TimeLive Help

1 — Last update: 2018/08/09

Livetecs

Table of Contents

1. Welcome to TimeLive

The Industry Standard in Web-based Timesheet management tools

About TimeLive:

TimeLive is a web-based timesheet management application which not only runs on SaaS but also on local and Hosted Server and provides Timesheet Control and Project Management for workgroups of any size or complexity. With TimeLive, you can record time and expenses quickly and easily at any time, from anywhere with Internet access or can. Once submitted, Timesheets and Expenses are automatically forwarded to the appropriate supervisor or manager for approval. Additionally, reports can be generated to track data such as project costs, client billing, utilization, and time off.

Unlike traditional PC applications or client-server applications, the only software that is required on each user's PC is a standard Javascript-enabled web browser.

2. Installation For On-Premises Version

2.1. TimeLive Installation Types

TimeLive Installation Types:

TimeLive is available from Livetecs in a variety of installations:

On-Demand/Cloud-Based (Hosted) implementation:

The Cloud version of TimeLive is implemented on servers managed by Livetecs. The application is available to users through the Internet, using a web address provided by Livetecs or through the Customer Login section of the TimeLive website which can access any place of the world with internet access. By signing up for the Cloud version, you are saved from the overhead of purchasing and maintaining application and database servers. For more details, Please Click Here.

1. **Free Version up to 3 users with 3 Projects:**
   The Hosted Free Version is designed to assist the user in evaluating the application as well as it is ideal for small companies to benefit from this powerful robust application. It is limited to 3 users and 3 Projects but enables unlimited use. Any data entered in the Free Version can be migrated to the full version. For Trial & Free Cloud-based Version, Click Here.

2. **Full Version:**
   The full Cloud-based version is provided when a user is ready to purchase and implement TimeLive for use in the organization. It supports the unlimited number of users (depending on Purchase Plan), Projects and tasks. For Pricing, please Click Here. To Sign up Click Here.

On-Premises installation:

With the standard installer, Administrator can install and manage the TimeLive application and database within the organization. It is installed using a file provided by Livetecs and Application is available for users through the network. For more details, please Click Here.
There are two types of standard installations available:

1. **Free Version up to 3 users with 3 Projects:**
   The free version is designed to assist a user in evaluating the application as well as it is ideal for small companies to benefit from this powerful, robust application. The simplified installation process allows a user to install it on their local computer. It is limited to 3 users and 3 Projects but enables

unlimited use. Any data entered in the Free Version can be migrated to the full version anytime. For Free On-Premises Application, Click Here.

2. **Full Version:**
   The full version is provided when a user is ready to purchase and implement TimeLive for use in the organization. It is installed on a local server. It supports an unlimited number of users (depending on Purchase Plan), Projects and tasks. For Pricing, please Click Here. To download On-Premises Application, Click Here.

For information on evaluating or purchasing either type of TimeLive Application, visit our website at Livetecs.com, contact us through Live Chat or e-mail us on sales@livetecs.com

2.2. System requirements – Non Hosted Version

System requirements (On-Premises version):

The sections that follow provide the system requirements for each type of TimeLive installation.

Standard Full Version:

Below are the requirements for the full version of the standard (non-hosted) installation, which requires an application server, a database platform, and client machines.

Server Requirements

The server on which TimeLive is installed must meet the following requirements.

Operating System:

- Windows 7 or later.
- Windows Server 2008 R2 or later.

Hardware Requirement:

- Intel Core 2 Duo Processor or higher
- 4 GB of RAM or more
- 10 GB of hard disk space or more
- SMTP-compliant E-mail Server

Software (higher is better)

- Microsoft Windows Installer 3.1 (Built-in)
- Microsoft .NET Framework 4.5.1 (Built-in)
- Microsoft Data Access Components (MDAC) 2.8 (Built-in)
- Microsoft SQL Server 2008 Express Edition (Built-in)

- Optional: Microsoft Internet Information Services (IIS) 5.0 (built-in). TimeLive setup automatically installs the TimeLive internal web server (Cassini), in case if IIS will not available on a system where TimeLive is being installed.

Web Server requirements for different OS:

Operating System	Web Server
Windows 7 (32 bit and 64 bit) or later	Microsoft IIS 7.0 or later
Windows Server 2008 (R1, R2) (32 bit and 64 bit) or later	Microsoft IIS 7.0 or later

Database Requirements:

TimeLive requires any of the following database platforms be installed, either on the TimeLive server or on a server accessible by TimeLive.

- Microsoft SQL Server 2008 or later.

2.3. Client Requirements

Client Requirements:

Each computer being used to access the TimeLive system must have

Operating System (Any) :

- Any Windows version
- Macintosh OS X
- Linux workstations and UNIX terminals running X Windows

Minimum Hardware Requirement (Higher is better):

- Pentium III, 450 MHz processor
- 256 MB RAM
- Monitor resolution (1024 × 768)

Web Browser (Higher is better):

An application may require any one of the following,

- Microsoft Edge
- Google Chrome (Version 31.0)
- Safari (Version 7.1)
- Microsoft Internet Explorer (Version 11.0)
- Mozilla Firefox (Version 33.0)
- Mobile Browser (Any)

2.4. Installing Web Server (IIS)

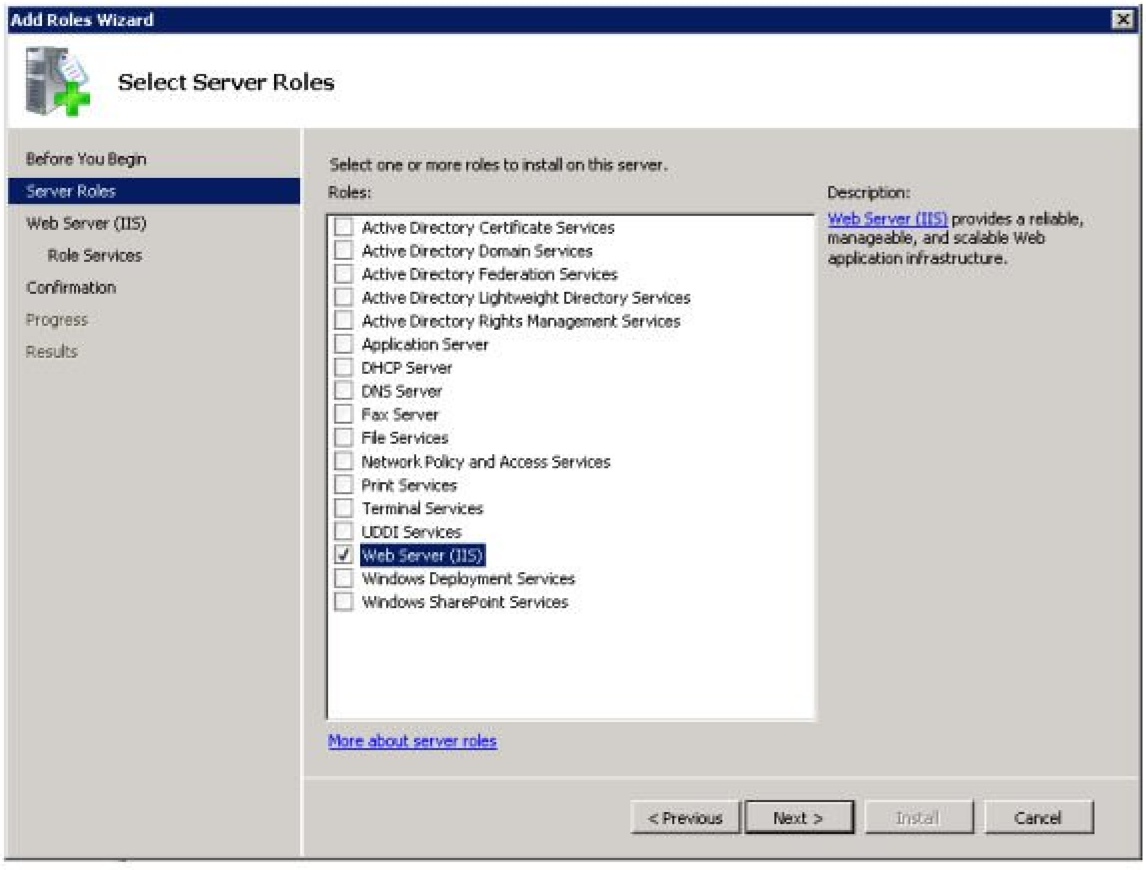
2.4.1. Installing Web Server (IIS) on Windows Server 2008

Installing Web Server (IIS) on Windows Server 2008 (32 bit & 64bit):

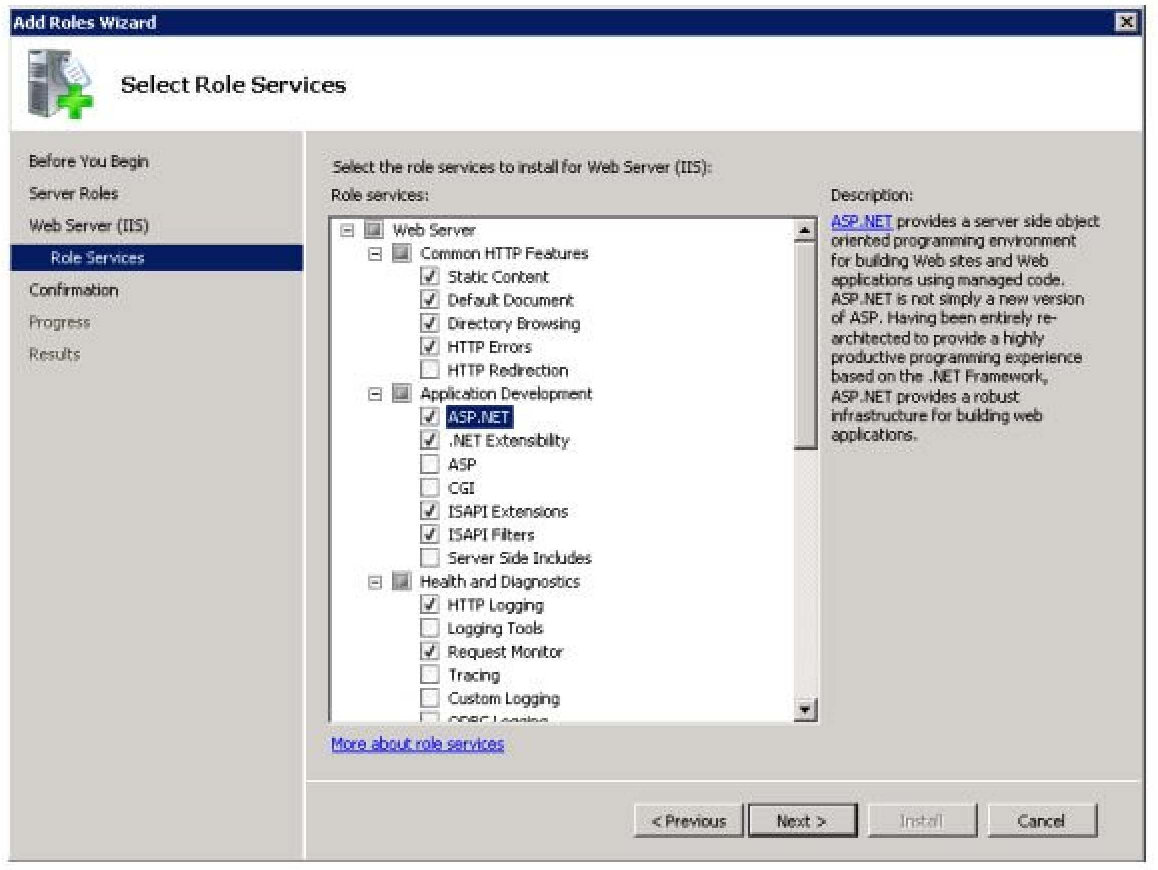
IIS on Windows Server 2008:

IIS 7.0 is one of the Windows Server 2008 Server Roles. IIS can be installed through the GUI using the new Server Manager interface after the Windows Server 2008 OS is installed. Server Manager provides a single dashboard to install or uninstall Server Roles and Features. It also gives an overview of all currently installed Roles and Features. When IIS 7.0 is chosen from the Server Manager, the basic components, and services needed for IIS are automatically selected.
Installing IIS 7.0 on Windows Server 2008 is a little different from installing IIS 6.0, but it's still pretty straightforward using Server Manager.
You can install IIS via Server Manager found in:

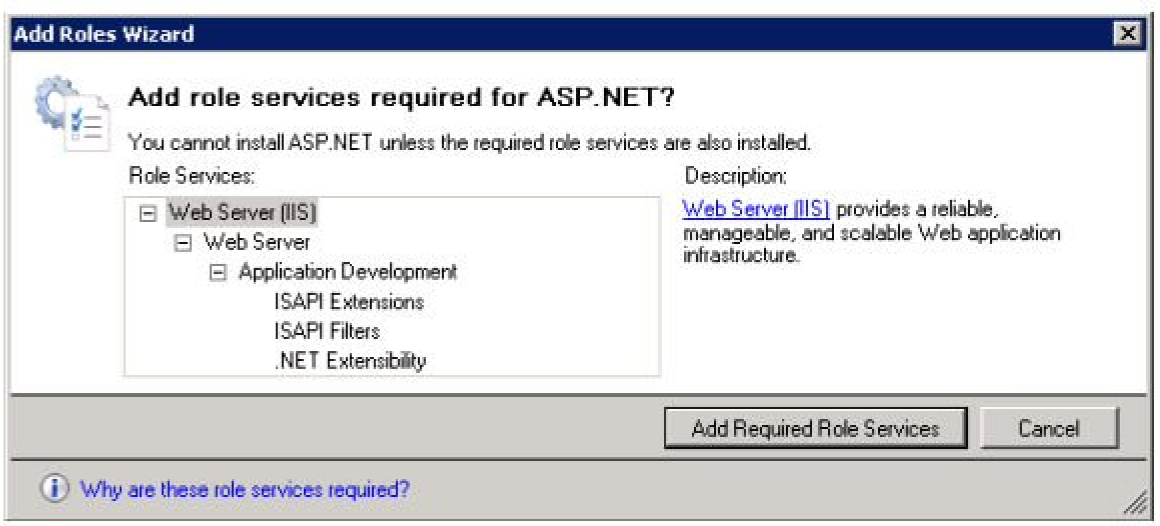
- Start Menu – Server Manager (it can also be found under Start Menu – Administrative Tools – Server Manager)
- Select the **Add Roles** under Roles
- Select **Web Server (IIS)** on the Server Roles page:



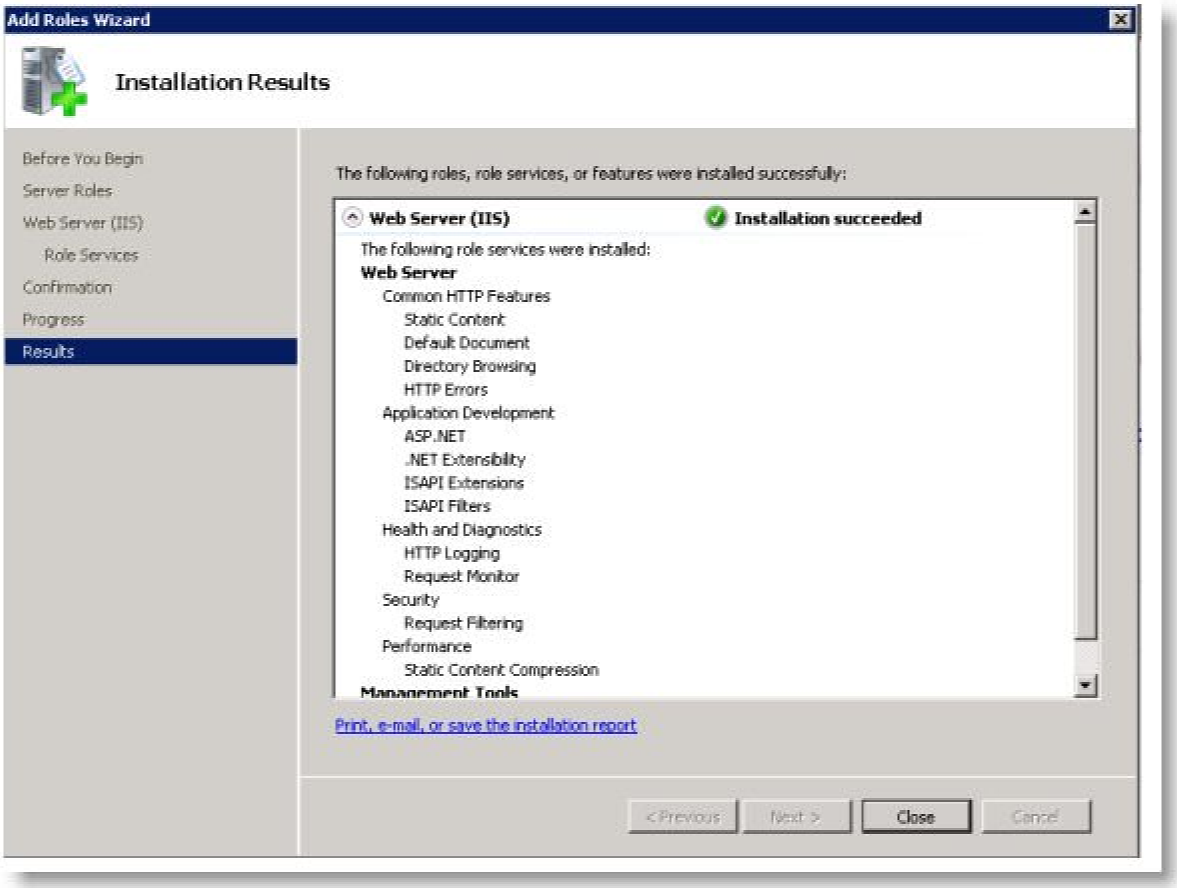
- Choose the IIS Services to be installed. In this case, ASP.NET is to be installed. Select the services required by your applications:



- When choosing the above services, you might be prompted with a warning to install some required services for the selection above, if the services are not already installed:



- IIS 7.0 is now installed with a default configuration for hosting ASP.NET on your Windows Server 2008:

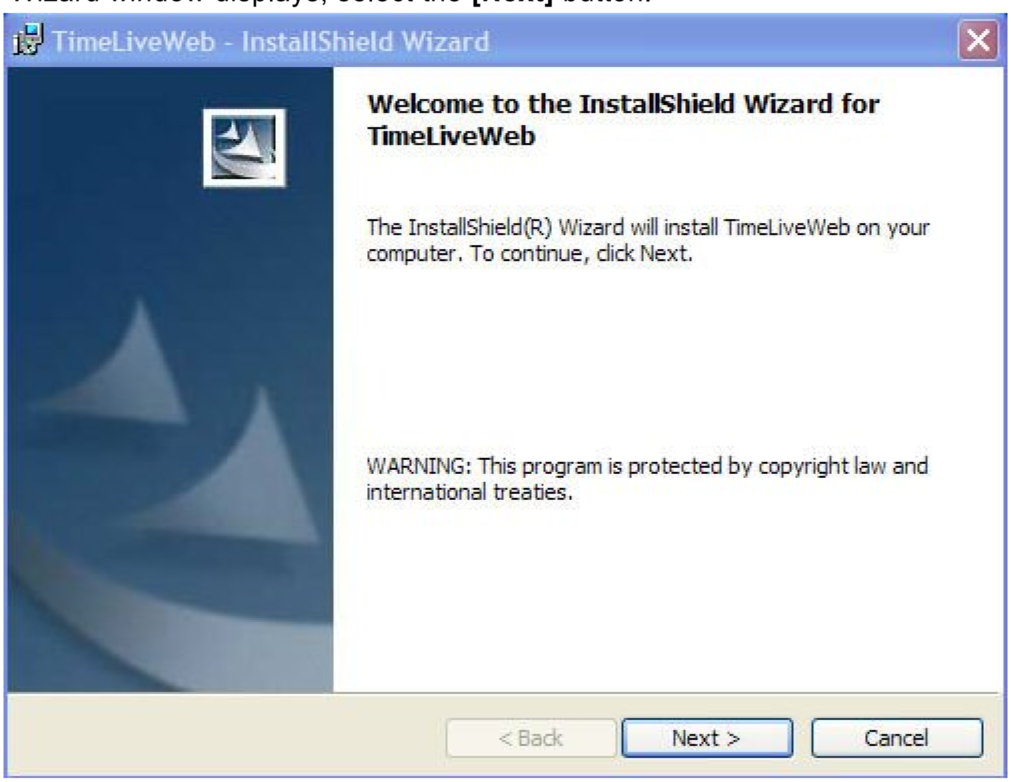

Add Roles Wizard
Installation Results
Before You Begin
Server Roles
Web Server (IIS)
Role Services
Confirmation
Progress
Results
The following roles, role services, or features were installed successfully:
Web Server (IIS)
Installation succeeded
The following role services were installed:
Web Server
Common HTTP Features
Static Content
Default Document
Directory Browsing
HTTP Errors
Application Development
ASP.NET
.NET Extensibility
ISAPI Extensions
ISAPI Filters
Health and Diagnostics
HTTP Logging
Request Monitor
Security
Request Filtering
Performance
Static Content Compression
Management Tools
Print, e-mail, or save the installation report
< Previous
Next >
Close
Cancel

2.5. Installing TimeLive on local server

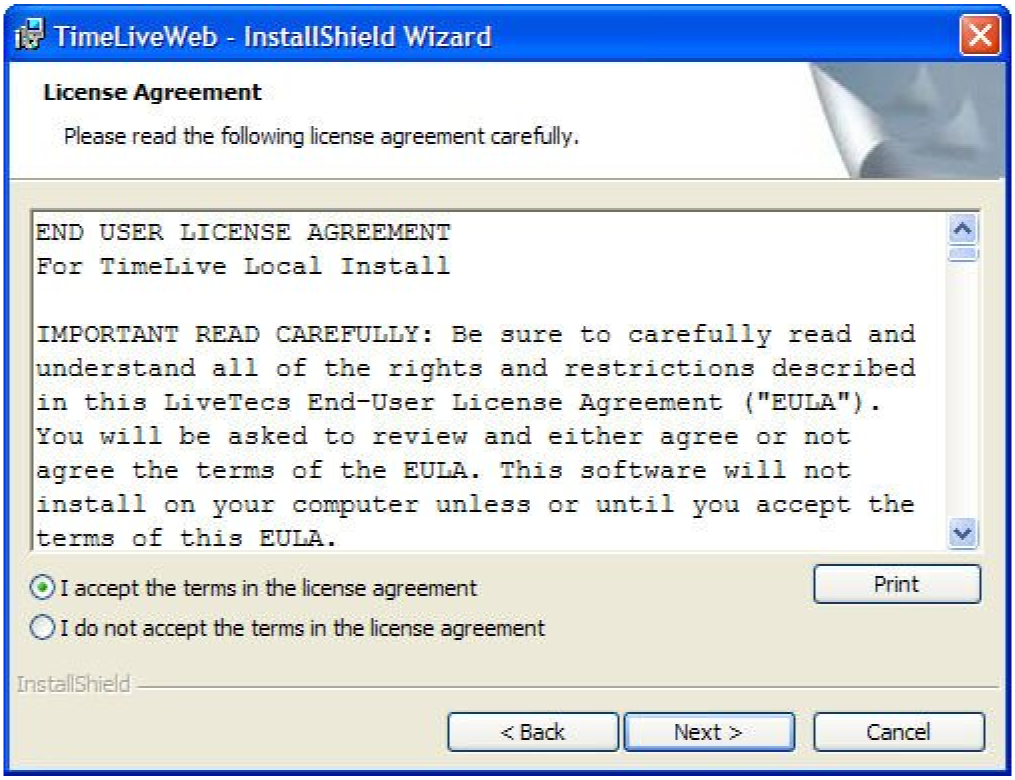
Installing TimeLive on local server

To install the local version of TimeLive:

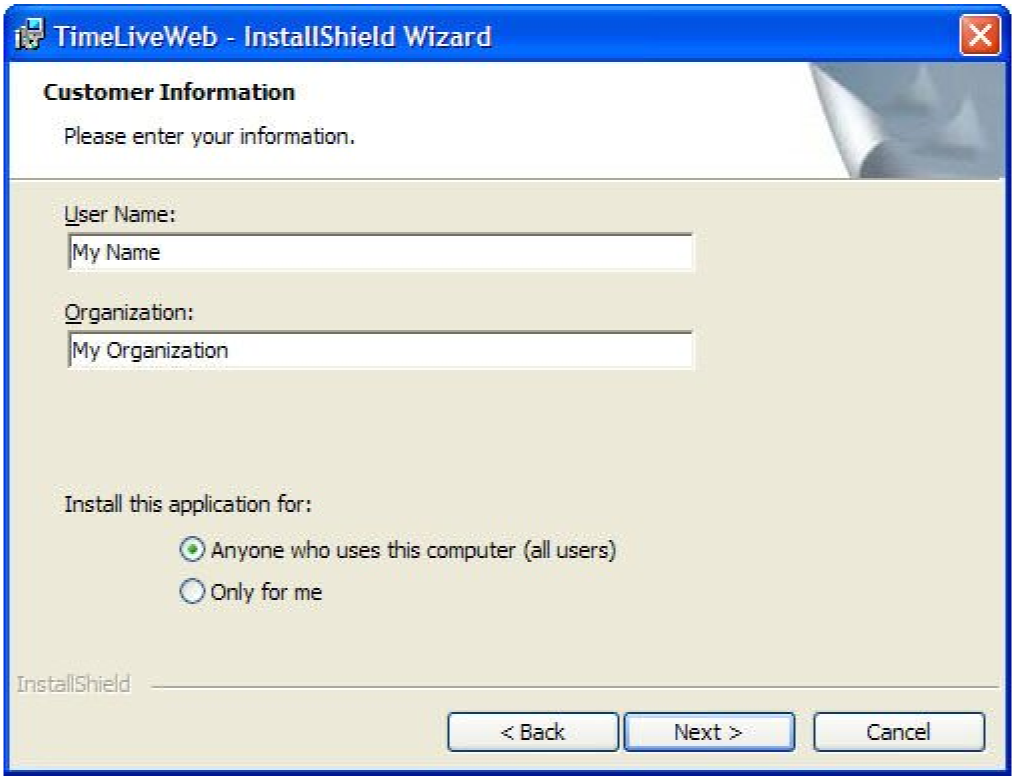
1. TimeLive installer automatically installs **[SQL Server 2008 Express Database]** engine during standard installation. SQL Server 2008 Express can be removed from the installation by choosing “Custom” installation in feature selection dialog box.
2. If **[SQL Express 2008 Express Database]** is not selected in installation, TimeLive then first show a “TimeLive database setup page” which allows creating and setup TimeLive database on any standard SQL Server 2000, 2005, 2008 server which is already running in the organization. This setup page creates a new TimeLive database with all necessary tables and default data.
3. TimeLive automatically install its own internal web server (Cassini), if there is no IIS already available on the system where TimeLive is being installed.
4. Download the free version of TimeLive from Livetecs website and save it to your local computer.
5. Double click on TimeLive installation file to launch TimeLive Setup Wizard. When the TimeLiveWeb – InstallShield Wizard window displays, select the **[Next]** button.



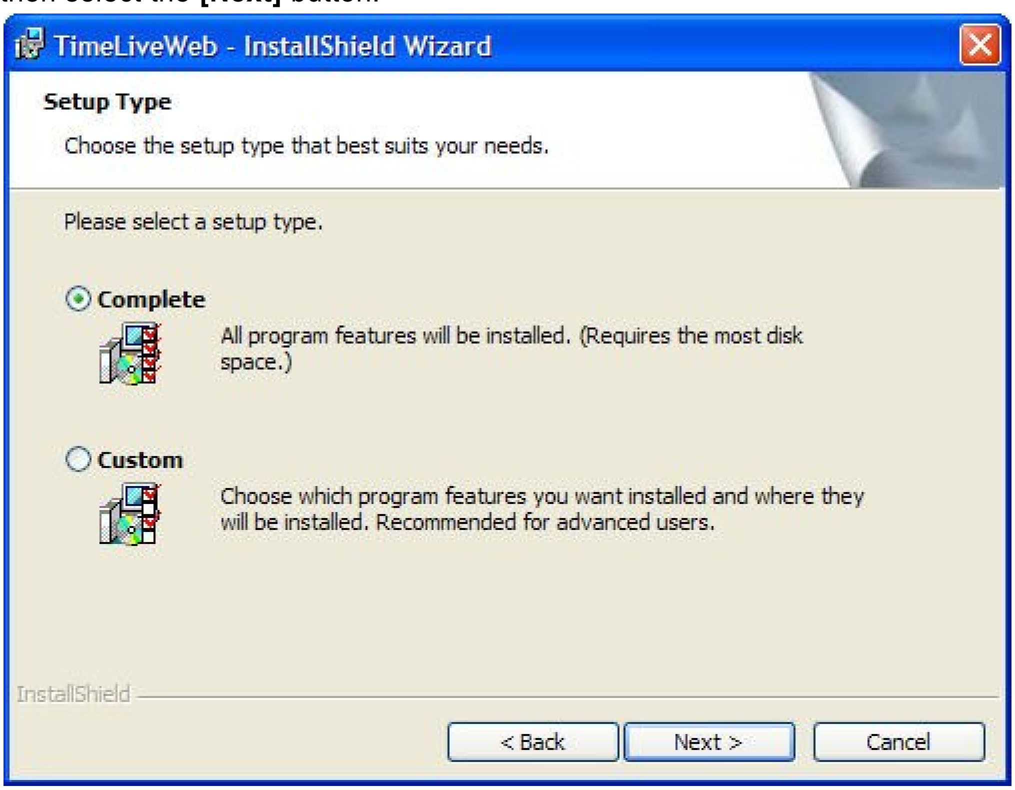
6. Read the license agreement. If you agree to its terms, select the **[I accept the terms of the license agreement]** radio button, and then select the Next button.



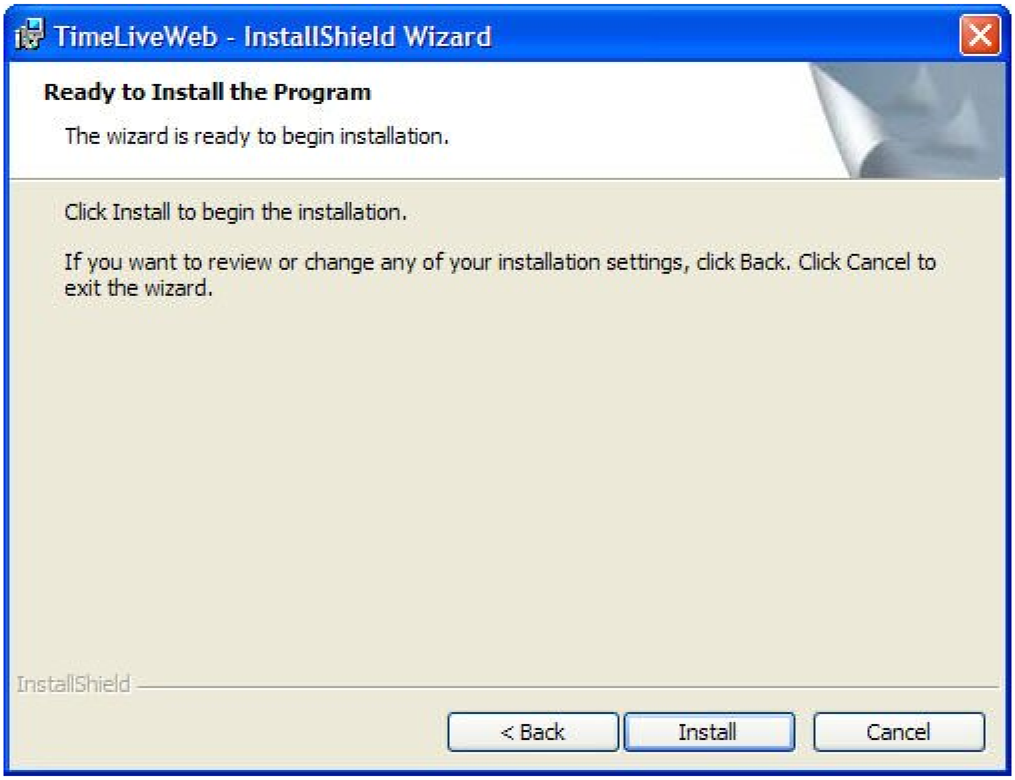
7. The installation program checks for the presence of Microsoft Windows Installer, Microsoft .NET Framework 2. If these necessary components are not present, the following window displays.
8. Enter your name and your organization name and then select the Next button.



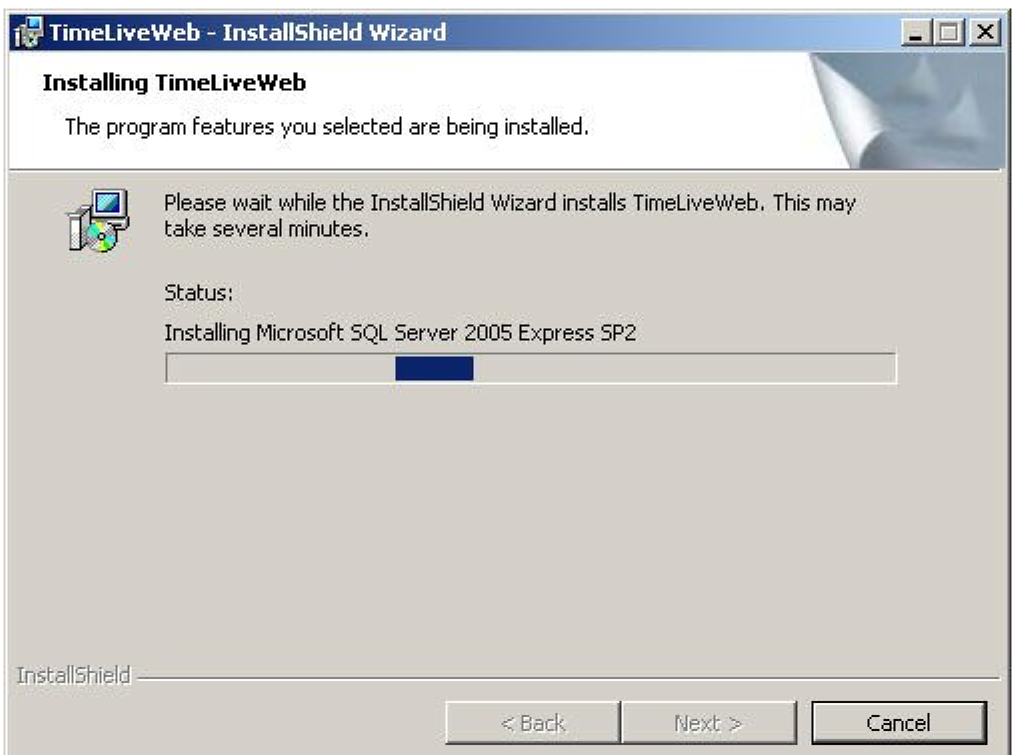
9. Select **[Custom]** if SQL Express engine is not required in the installation. Select the **[Complete]** radio button, and then select the **[Next]** button.



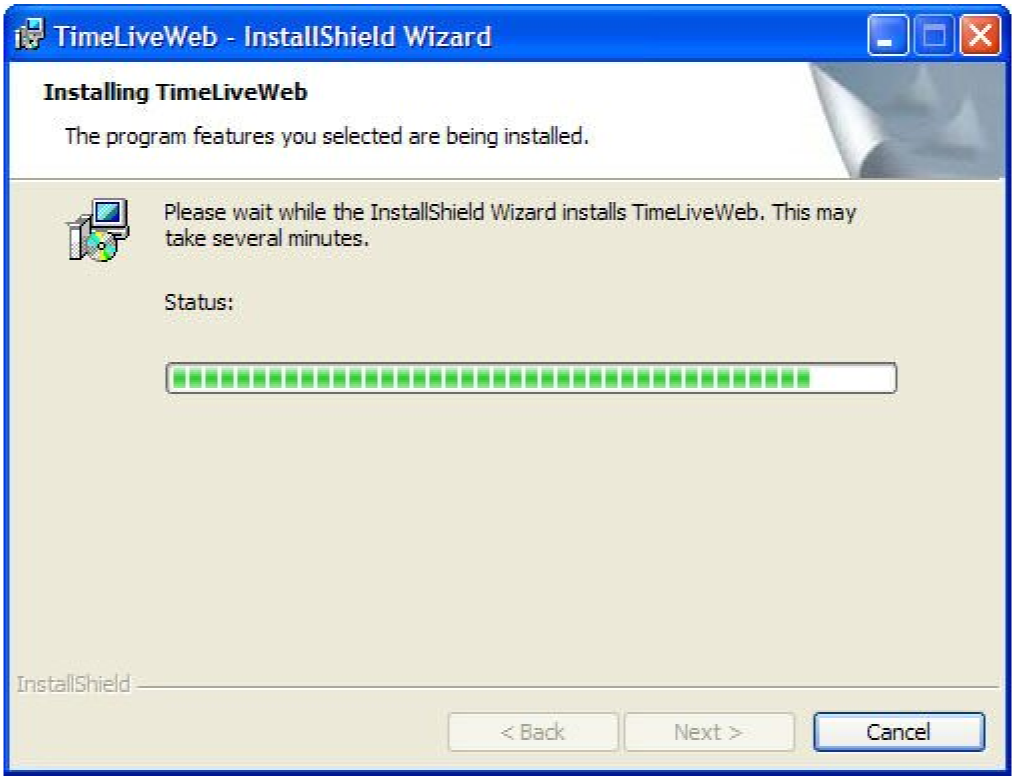
10. Now click on **[Next]** button to start TimeLive installation



11. A status showing that TimeLive is first installing "Microsoft SQL Server 2008 Express" on the machine.

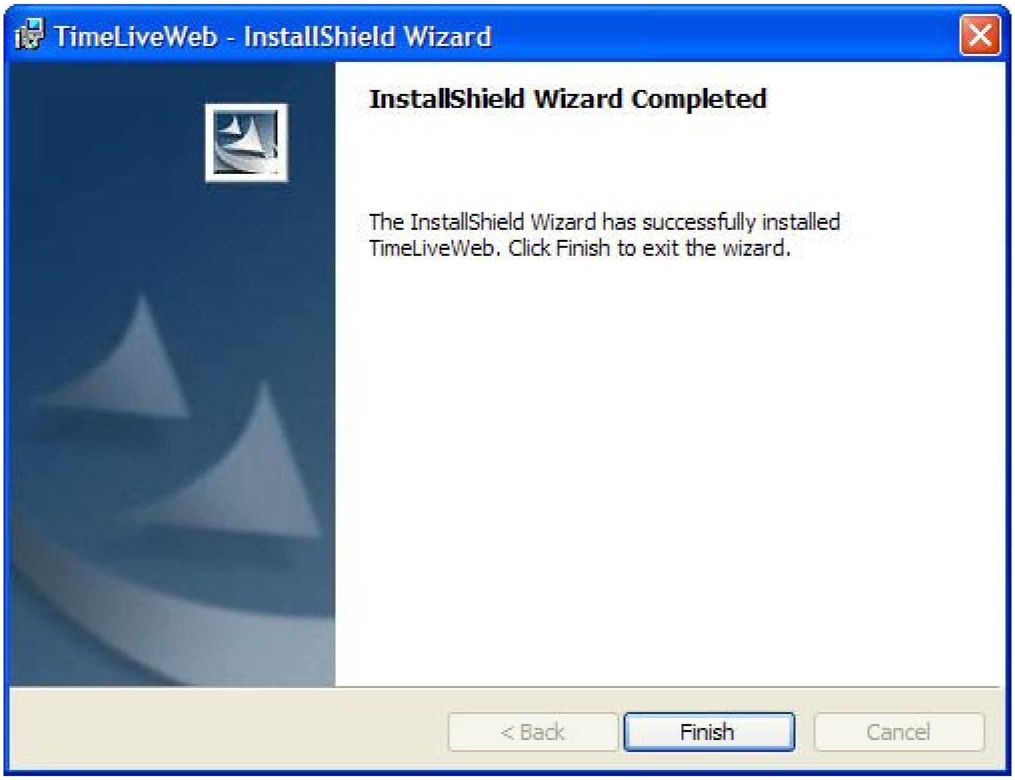


12. A Setup Status window displays that track the progress of the installation.



TimeLive requires that Microsoft Data Access Components (MDAC) be installed on your machine. At this point, the installation program will check for the presence of MDAC and automatically install it if it is absent. This may take a few minutes.
Once MDAC is installed, the computer may need to be restarted. The installation will automatically resume after the computer is restarted.

13. The InstallShield Wizard Complete window displays to indicate that installation was successful. To exit the installation program, click on **[Finish]** button.



14. TimeLive install SQL Server 2008 Express engine with instance name “TimeLive”. This instance can be removed from the system using add/remove program in your control panel.
15. Please continue reading to “Launching TimeLive (First Time)” in order to launch TimeLive.
16. In case of installation without SQL Server 2008 Express engine, TimeLive first open database setup page when launch first time. Using database setup page, a new database can be created as well as the already existing database can be pointed to the TimeLive application. Please continue reading to “TimeLive database setup” for setting up the TimeLive database.

2.6. Upgrading TimeLive on local server

Upgrading TimeLive on local server

To upgrade locally installed version of TimeLive:

1. Take a backup of TimeLive folder and database before the upgrade.
2. Uninstall existing TimeLive version from add/remove program in control panel. Uninstall will not remove TimeLive configuration files and database files. Do not delete any files manually from the TimeLive installation folder.
3. Install TimeLive new version.
4. Launch TimeLive. On the first launch, TimeLive will first upgrade existing database with new changes and then will reach to login page after upgrade. This process will take about 1 or 2 minutes.
5. User can now login in TimeLive as usual with their existing TimeLive login and password.

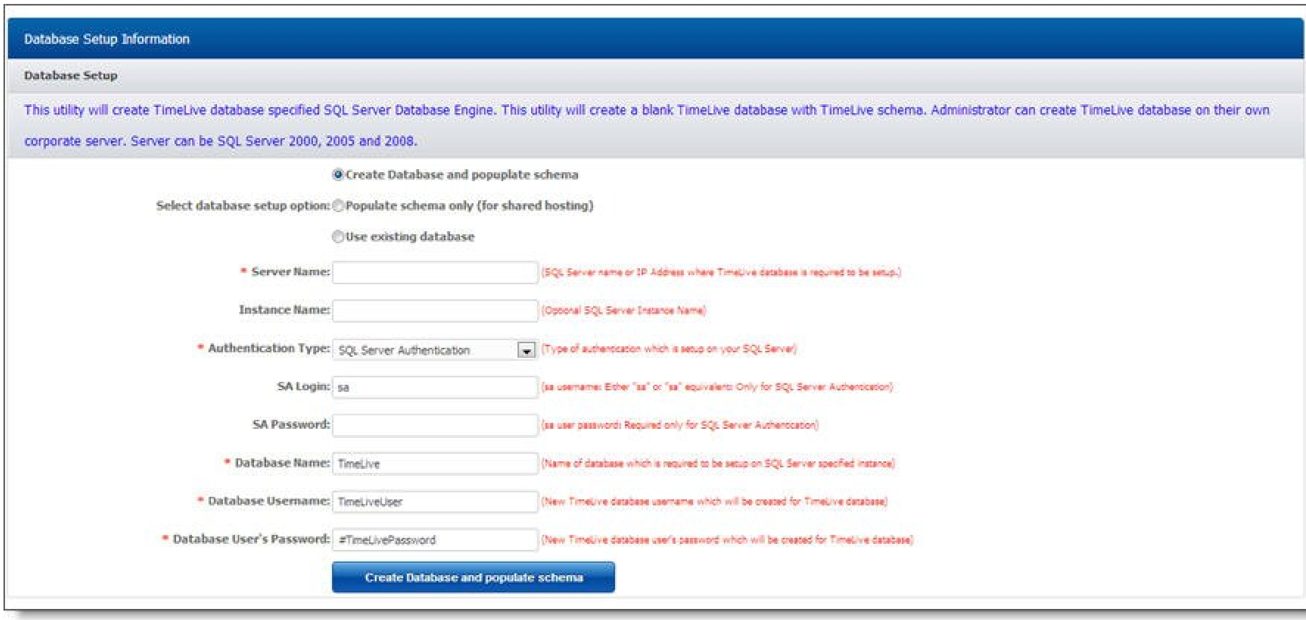
2.7. TimeLive Database Setup

TimeLive Database Setup:

At first launch of TimeLive, system administration should create and setup TimeLive database, if **[SQL Server 2008 Express]** engine is excluded in feature selection dialog box. In this case, a system administrator can setup TimeLive database by providing these parameters. Using this setup page, a system administrator can create a TimeLive database on any local or remote SQL server instance. SQL Server instance can be either SQL Server 2000, 2005 or 2008.

This utility can be launch separately also by opening below mentioned URL in your browser:
http://timeliveurl/Home/SetupDatabase.aspx
Using database setup page, an administrator can perform following database-related functionalities.

1. **Create a new database and populate schema:** (for setting up the new database) This option creates a new database, new database user and populate tables and default data in it. This also updates TimeLive configuration file (web.config) with this newly defined database connection settings.
2. **Populate schema only (For shared hosting):** This button creates database tables, objects and default data in the already created blank TimeLive database. This is useful for users who are setting up the TimeLive database in a shared hosting environment where there is the separate tool available for creating a database. This also updates TimeLive configuration file (web.config) with this newly defined database connection settings.
3. **Use existing database:** This option will allow the administrator to configure TimeLive with an existing TimeLive database.



Database setup parameter definition:

Server Name:	SQL Server database server name or IP address where SQL Server is installed. It can be any SQL Server 2000, 2005 or 2008 server address.
Instance Name:	(Optional) SQL Server instance name, if SQL Server is set up with some named instance.
Authentication Type:	SQL Server authentication type. TimeLive recommended authentication type is **[SQL Server Authentication]**, but system administrator can configure TimeLive database on windows authentication also.
SA Login:	As this utility creates a new TimeLive database and a new TimeLive database user, only SQL server user with **[System Administrator]** can create a new database on SQL server.
SA Password:	Password of “sa” or any system administrator equivalent login.
Database Name:	Database name which is required to be created on SQL Server.
TimeLive Username:	Name of the user which will be created automatically by this utility for accessing the TimeLive database by a TimeLive application.
TimeLive DB Users’ password:	Creates TimeLive DB user with this password.

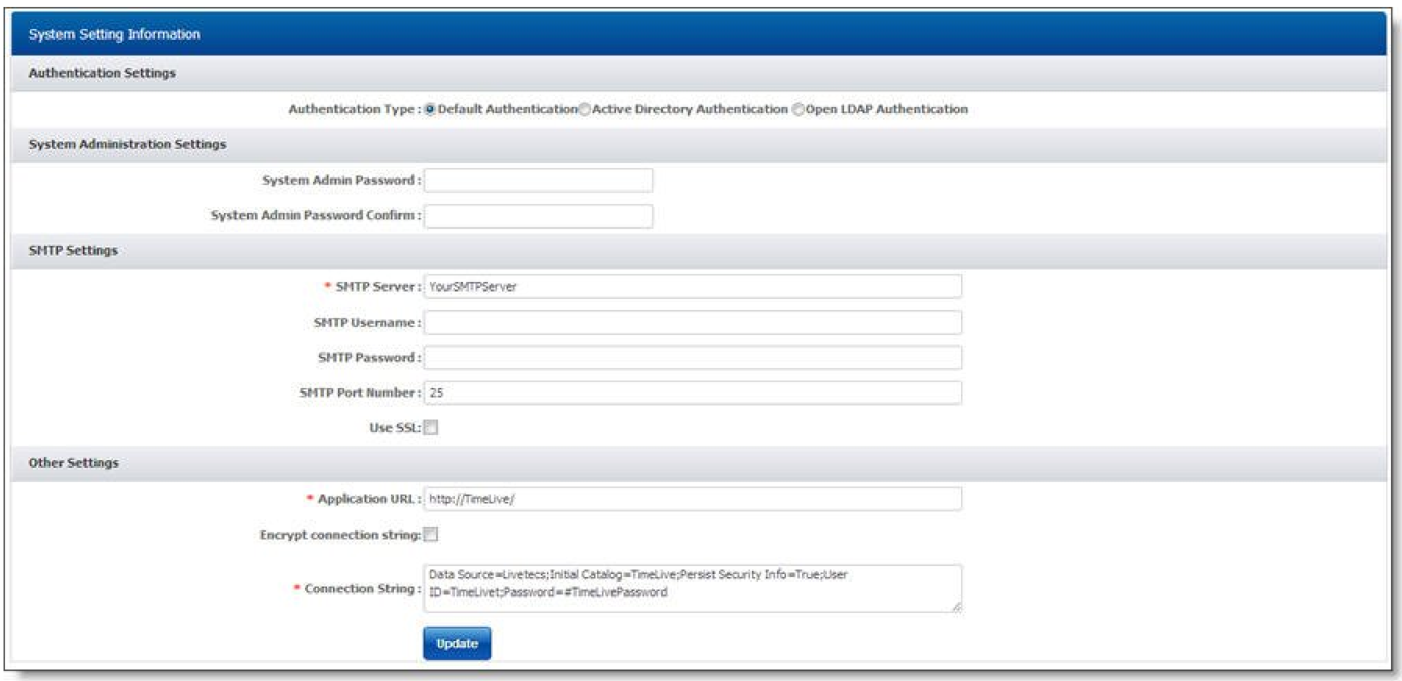
2.8. TimeLive System Settings

TimeLive System Settings

Specifying System parameters:

System configuration page can be launch from [Start —> All Programs —> TimeLive —> System Configuration] on system where TimeLive is installed. System setting can be open directly by opening this URL in your browser.
http://timeliveurl/home/systemsetting.aspx

Defining System parameters:



Authentication Type	TimeLive supports two types of authentication. One is default database-based authentication. Using default authentication, the user can login in TimeLive using their own username and password defined in the TimeLive application by Administrator. Second is [Active Directory Authentication] where the user can login in TimeLive using their corporate Active Directory username and password.
System Admin Password	TimeLive allows only system administrator user for accessing system configuration page. System administrator username is “SysAdmin”. An administrator can access system configuration area by entering username as “SysAdmin” and its password.

SMTP Server	SMTP Server name which you are using in your organization for sending emails.
SMTP Username	Username of SMTP server if any (optional)
SMTP Server Password	Password for SMTP server (optional)
SMTP Port Number	SMTP default port is 25. You can change it to some other port if SMTP server is configured on some other port other than 25.
Application URL	URL of TimeLive application which is accessible from your organization or from users from internet. It can be either local computer name of system where TimeLive is installed or it can be IP address of server or it can be full domain name.
Connection String	This is SQL Server connection string. By default, TimeLive installation always installs SQL Server 2008 Express edition on local server.

2.9. Understanding TimeLive Website URL

Understanding TimeLive Website URL:

TimeLive website URL depends on the type of server you are using, and has the following format:

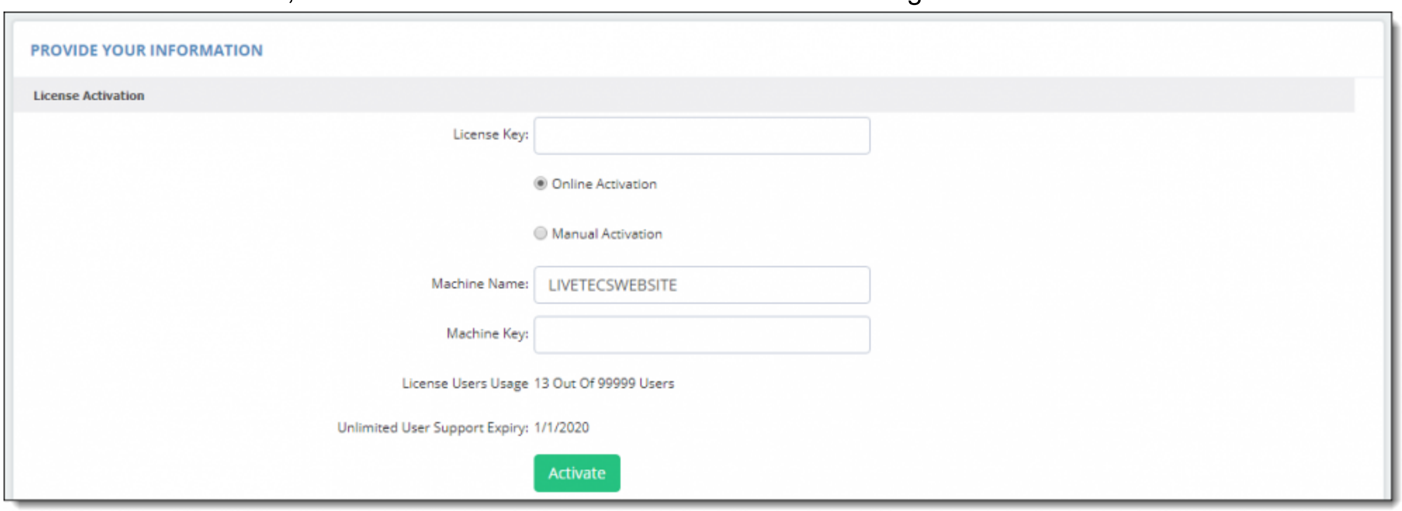
Server type	URL Format
Cassini WebServer	http://server_name:15395/ or http://IP_address:15395/
IIS	http://server_name/TimeLive or http://IP_address/TimeLive
ASP (Hosted version)	http://www.livetecs.com (login form of hosted version of TimeLive)

Once the application is installed, administrators should distribute the site's URL, together with login information, to all users so they can access TimeLive via their web browser.

2.10. TimeLive License Activation Guide

TimeLive License Activation Guide:

By default, TimeLive downloadable version comes with 30 days pre-activated trial license key. This pre-activated trial license key supports unlimited users up to 30 days. After 30 days, it converts to 5 users free TimeLive.
To ensure that TimeLive is secure and is only registered by authorized users, we have a systematic activation procedure in place to safeguard the software's use.
There are two types of license activation available in TimeLive. One is online activation and second is manual activation. Online activation is a process of activating TimeLive by using license key, received by the customer in TimeLive order email. Unfortunately, this online activation process can be blocked by either a firewall or the proxy server settings on your system. In case of failure of online activation, a customer can use manual activation, in order to activate TimeLive even without using of the internet.



Automatic Activation:

1. Navigate to **—>[License Manager]** while login as an admin user.
2. Enter your license keys in **[License Key]** field.
3. Just make sure that **[Online Activation]** is selected.
4. Click on **[Activate]** button to activate TimeLive.
5. You will receive a message **"License Activated Successfully"** in case of successful license activation.

6. You may receive a message **“Unable to activate license”** if TimeLive will not able to communicate with TimeLive license activation server. You can then try manual activation using instructions mentioned below.

Manual Activation:

There are two major steps of manual activation. First is to request your manual license activation key by sending an email to **[sales@livetecs.com]** with **[machine name]** of the system, where TimeLive is installed and order reference number. TimeLive machine name can be found in [Machine Name] field available in **Admin Options > License Manager**.
Once received a manual license key, navigate to **Admin Options > License Manager** while login as an admin user.
Select your activation type to **[Manual Activation]**.
Enter manual license keys in **[Machine Key]** field.
Click on **[Activate]** button to activate TimeLive.
You will receive a message **“License Activated Successfully”** in case of successful license activation.

2.11. Installation on Shared Hosting

TimeLive Installation on Shared Hosting Server

There are 3 major steps for Installation on Shared Hosting Server,

1. Introduction
2. Installation and Setup
3. Upgrading TimeLive on shared hosting

1. Introduction

TimeLive Edition is a timesheet and expense management system based on current Microsoft Technologies. It is designed to run on an IIS 7.0 or later web server that has ASP.Net 4.5.1 installed. In addition, the software is designed to work with Microsoft SQL Server, Microsoft SQL Server Express Edition. TimeLive can be set up on ASP.Net/SQL Server supported shared hosting.

2. Installation and Setup

- TimeLive require “Full Trust” permissions on your shared hosting.
- Download the manual installer file from http://www.livetecs.com/Release/TimeLiveWebForHosting.zip
- Unzip the software to a folder on your computer’s hard drive.
- Create a new (Domain or Website) from your hosting control panel for TimeLive.
- Upload, copy or FTP the entire TimeLive folder to your web server, including ALL the folders and files inside it. Do not change the file and folder structure. You would typically place the TimeLive folder in your web server’s wwwroot directory, but it can also be copied to any sub-directory under a web site’s root directory, as long as it is marked as an application starting point. You may have to ask your host for information about how to perform these steps.
  Your web server must have the Microsoft.Net Framework 4.5.1 installed before TimeLive will function.
- The following directories and files of the uploaded TimeLive installation need to have read, write, delete, and modify permissions in the file system:
  - /web.config (file)
  - /Uploads (folder)
  - /Log (folder)
  - /App_data (folder)
- The following directories need to have “execute” and “read” permissions in the file system
  - /bin

- Create a new SQL Server database with name “TimeLive” from your hosting control panel. After creating the database, note down the physical database name created using control panel.
- Create a new SQL Server database user from your hosting control panel with name “timelivedbuser” and assign this user to newly created “TimeLive” database.
- Launch “*TimeLive Database Setup*” page and setup TimeLive database on your shared hosting SQL Server.
- Now open “http://TimeLiveURL/Default.aspx” page where TimeLive URL should be URL pointing to a TimeLive folder in your file system. This will launch TimeLive account add a page where the administrator can define their organization and administrator information.

3. Upgrading TimeLive on shared hosting

- Download the manual installer file from http://www.livetecs.com/Release/TimeLiveWebForHosting.zip
- Unzip the software to a folder on your computer’s hard drive.
- Delete all files from TimeLive folder except
    - /web.config (file)
    - /Uploads (folder)
    - /Templates (folder)
- Upload, copy or FTP the entire TimeLive folder to your web server, including ALL the folders and files inside it except
    - /web.config (file)
    - /Uploads (folder)
    - /Templates (folder)
- Now open “http://TimeLiveURL/Default.aspx” page where TimeLive URL should be URL pointing to TimeLive folder in your file system. This will launch TimeLive login page after taking a while in executing database upgrade script on existing database.

3. Getting Started with TimeLive

3.1. 1.1 What you can do in TimeLive?

TimeLive Timesheet suite is integrated suite for managing project life cycle including projects, tasks, timesheet, expense, and attendance etc.

TimeLive Tools:

These are the top level tools for TimeLive timesheet suite:

1. **Employee Timesheet Management:**
    a. Time entry using multiple formats: Daily, Weekly, Bi-Weekly, Semi-Monthly, and Monthly.
    b. Time entry in time format (12h or 24h), decimal, and percentage.
    c. **Billing manager** can define separate billing rate for the different type of work type like overtime, travel etc. Employee can enter time entry for different work types like standard, overtime etc in time entry.
    d. Timesheet approval (customizable timesheet approval path by defining work-flow path).
    e. Allow external users (client) to approve their own project timesheet entered by employees.

2. **Billing Management:**
   Four different types of billing rate types configurable at the project level. Billing rate can be defined with its applicable date range with complete history. Timesheet then picks billing rate based on time entry date from billing rate history defined at the employee level, project role level, project employee level, or project task level.
    a. **Employee own billing rate** (System pick employee own billing rate).
    b. **Project-based employee billing rate** (System pick employee billing rate defined for each project separately).
    c. **Role-based billing rate** (System pick billing rate of the employee based on role wise billing rate defined for each project separately).
    d. **Task-based billing rate** (System pick billing rate of time entry which is defined in task's billing rate).

3. **Expense Management:**
    a. Multi-currency expense entry tracking with currency conversion.
    b. Defining of exchange rate history for each currency.
    c. Tax calculation with customizable formulas.
    d. Expense reimbursement management and tracking.
    e. Expense approval (customizable expense approval path by defining work-flow path).
    f. Allow external users (client) to approve their own project expenses entered by employees.

4. **Time Off Management**
   Using Time Off feature, employees can check available time off, make a request, get manager approval. TimeLive time off and leave management provides a comprehensive time off and leave management solution unmatched in the industry today.
   a. Leave records (sick leave, casual leave etc)

5. **Task Management:**
   Project manager, team lead, and employee can create tasks for their project and can assign those tasks to multiple users. And then the whole team which is assigned to that particular task can do these task management activities.
   a. Can add their own comments.
   b. Can attach files the task level.
   c. Can change the status.

6. **Employee Attendance:**
   a. Punch in / punch out

7. **Reporting**
   a. Fully customizable Reports
   b. Timesheet reports for timesheet and time billing.
   c. Expense report for expense reporting to the client.
   d. Attendance reports with yearly leave days summaries.

3.2. 1.2 Launching TimeLive

On-Premises TimeLive Version

Before starting TimeLive, Administrator should create a new TimeLive account. TimeLive can be launch using two way.

1. By double-clicking on the TimeLive shortcut on server desktop.
2. Or by typing the whole URL of TimeLive in your web browser. URL can be enter using these different way:
    a. **http://servername/TimeLive** or
    b. **http://server-ip-address/TimeLive**

On-Demand TimeLive Version

1. On-Demand TimeLive can be accessed using **https://yourcompany.livetecs.com**

3.3. 1.3 Creating new TimeLive Account

Creating TmeLive Account:

Before starting TimeLive, Administrator should create a new TimeLive account. If TimeLive is being launched for the first time, it will automatically redirect to account add a form where System Administrator can create new TimeLive Account.

Account Information:

There are two types of information which are required to be entered on new account Add Form. First, organization information and second is personal along with login information of TimeLive System Administrator. After sign up, Administrator receives a notification email containing Login Email / Password of TimeLive account.

3.4. 1.4 First time login in TimeLive

After new Account Add, the system administrator can log in to TimeLive system using login email and password.

Steps of login:

1. Enter your Login Email address
2. Enter your password
3. Select UI (User Interface) Language according to your preference
4. Click on Login button to login in the TimeLive Application
5. On successful login, the user will get access to the homepage.

TimeLive Dashboard is by default set as homepage:

3.5. 1.5 Using Menus in TimeLive

As soon as you login TimeLive Account, by default, you will be navigated to Dashboard. On TimeLive Screen, there are 3 Navigation Bars through which you can navigate further to each and every feature/tool/option of TimeLive(if you have the permission).

Navigation Bars:
There are three Navigation Bars which gives a user access to different pages available in TimeLive:

Top-Right Navigation Bar:

1. Hamburger button (View/Hide Description of Left-Hand Side Navigation Bar)
2. Clients
3. Projects
4. Employees
5. Billing

Left-Hand Side Navigation Bar:

1. **Dashboard** (Graphical Representation of your work at a glance)
2. **My Projects** (Shows only those projects which are assigned to the user.)
3. **My Tasks** (View and quick access to user's assigned tasks. Through this page, the user can also create new tasks in his assigned project)
4. **My Timesheet** (User access his/her Period timesheet. You can switch Day View Timesheet by clicking on Day View Button)
5. **My Expenses** (User access daily expense entries. It allow to submit new expenses)
6. **My Time Off**
7. **My Reports** (Show main reporting page)
8. **My Approvals**

Right-Top Navigation Bar:

1. Admin Options (All system administration options)
2. Profile Drop-Down Menu
   a. User Image (By Default, Name initials appear on it in absence of picture).
   b. My Profile
      This link open User Profile Popup through which user can,
      - Update/Change Personal Information along with Timezone and UI Language
      - Upload Picture and enable Time In/Out Option

- User can enable API Authentication Token

3. My Email Preferences
   With this link, User will be directed to current user's Email Preference Page. A user can set his/her email preferences as per the requirement.
4. Old Version
   This link will redirect the user to the old version of TimeLive.
5. Logout

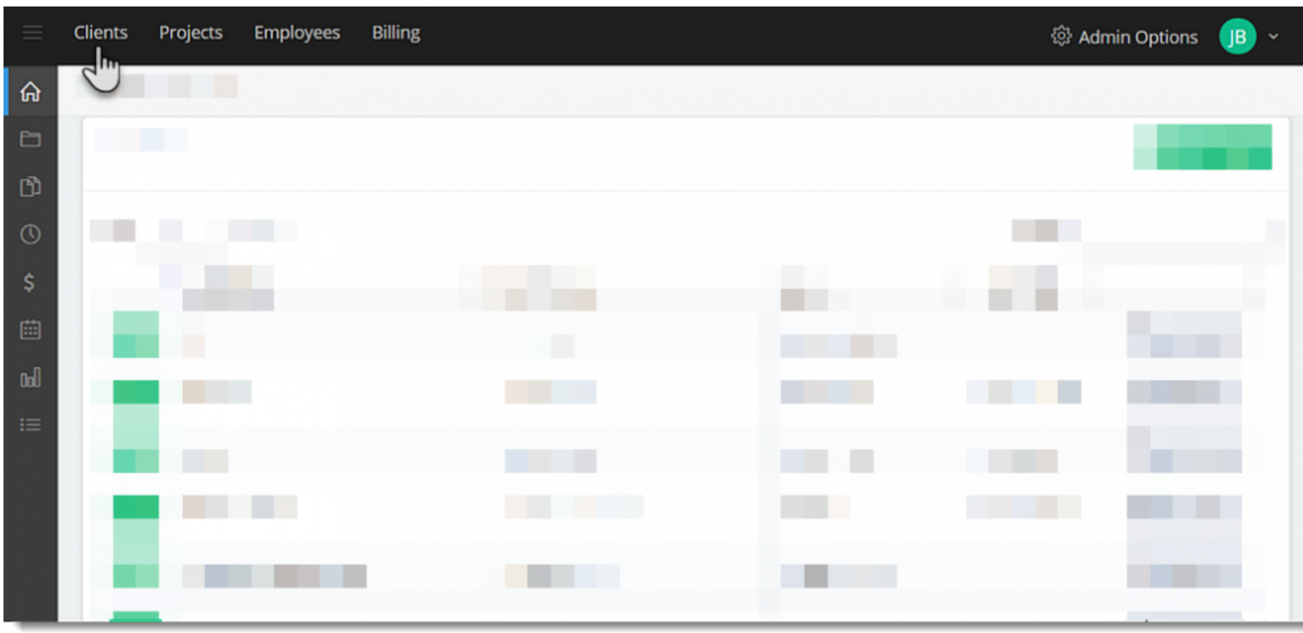
3.5.1. Clients in TimeLive

Clients are customers of your organization to whom you bill time or costs for projects. Clients can be external customers or departments within your organization. You should define your clients before setting up projects to eliminate the need to go back and assign the projects to the appropriate client.

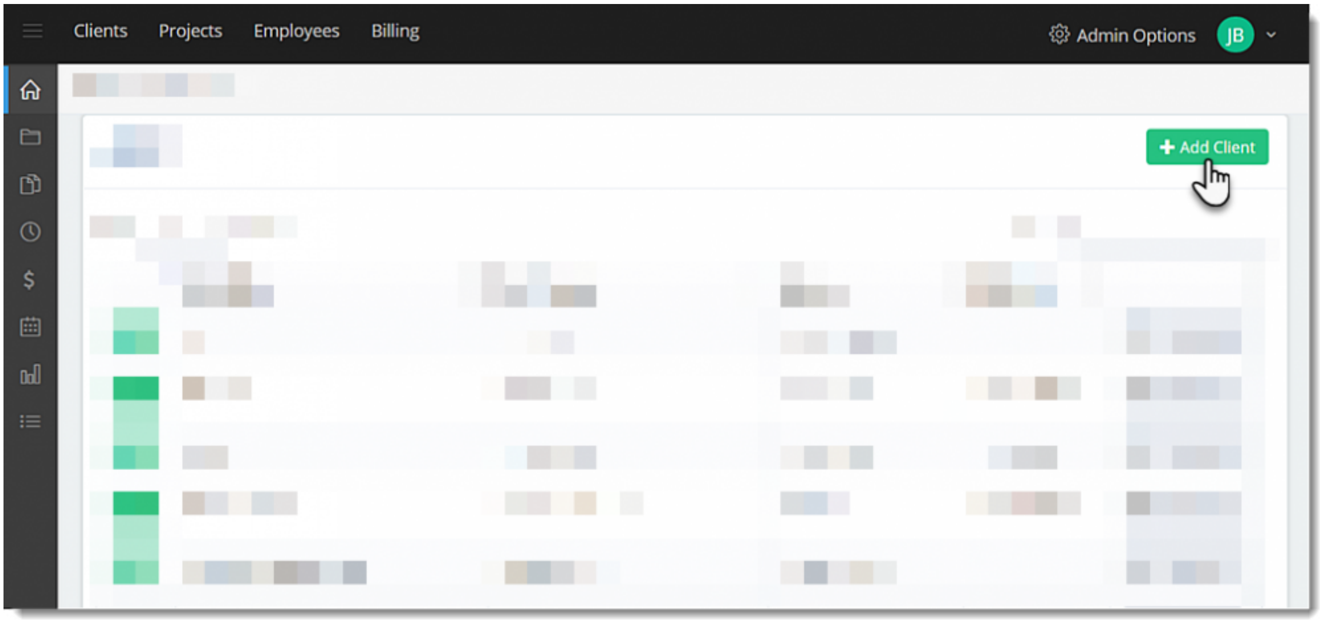
A System Administrator can set up Clients using **Clients** Link at the top-left of the application.

Add a new Client:

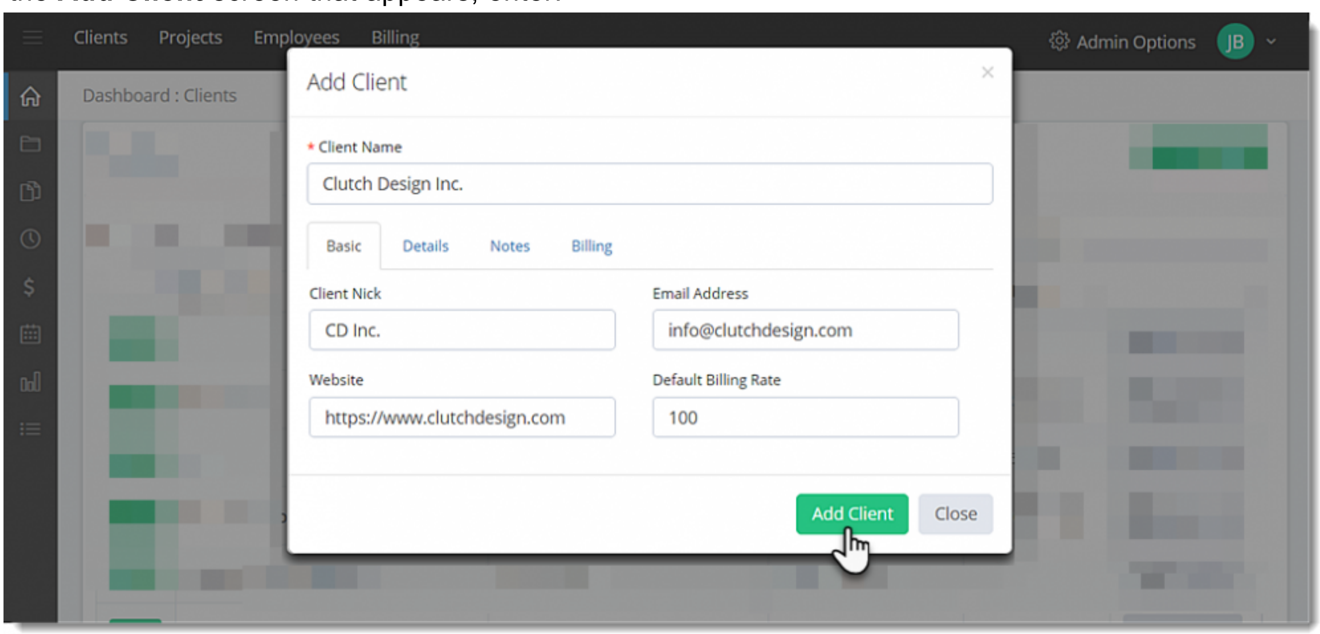
1. Login TimeLive with Admin Credentials.
2. Click on the **Clients** link at the top-left of the application.



3. Click **+ Add Client** green button on top-right of the application.



4. In the **Add Client** screen that appears, enter:



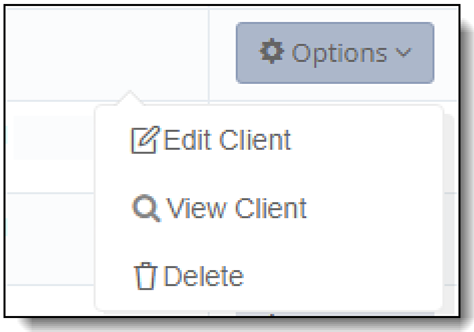
- Fill the **Client Name** (which is mandatory) along with other relevant data if you require.
- Click **Add Client** button to add this client to the application.

<u>Edit an existing Client</u>:

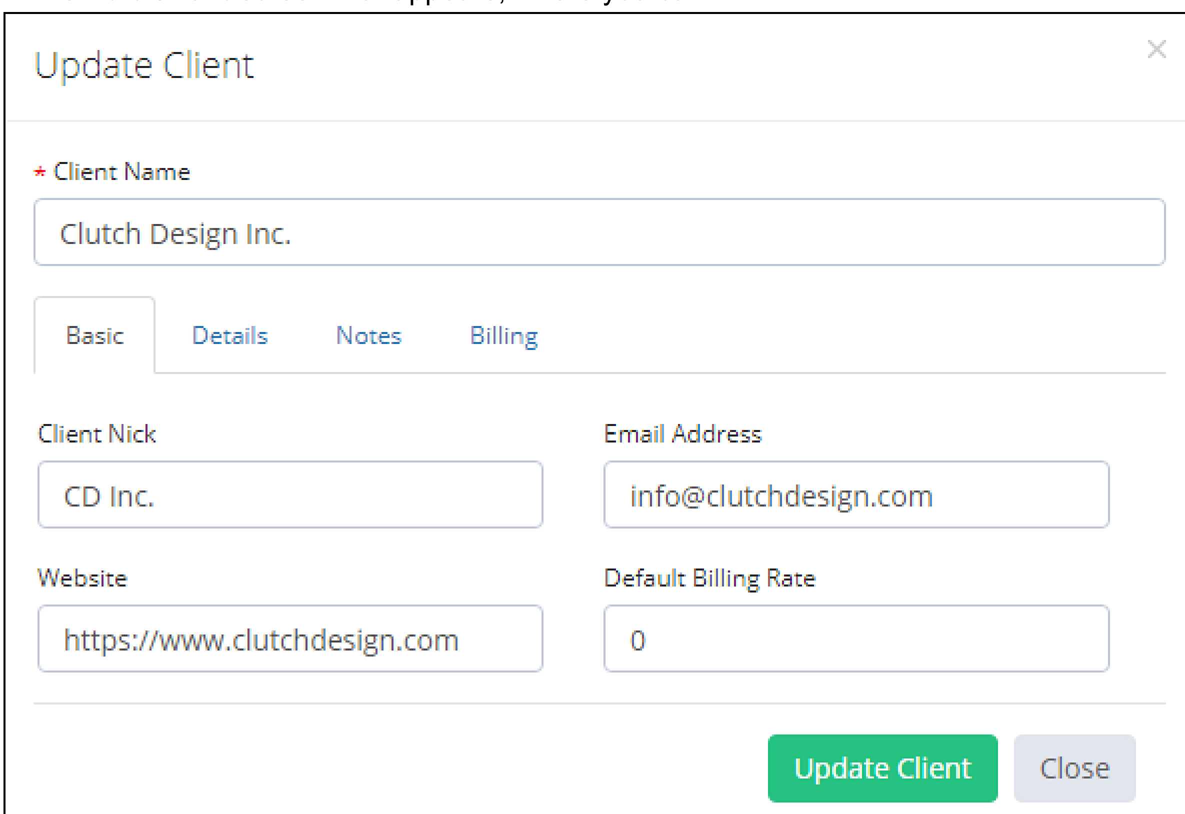
1. Login TimeLive with Admin Credentials.

2. Click on the **Clients** link at the top-left of the application.

3. In **Client List**, click on **Options** gear icon to the which you want to modify.
4. Select **Edit Client**.



5. In the **Edit Client** screen that appears, where you can:



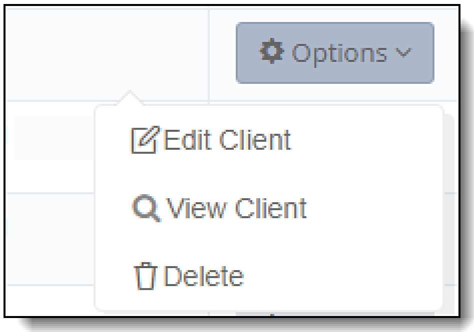
- Make the changes as per your requirement and click **Update Client**.

Delete an existing Client:

1. Login TimeLive with Admin Credentials.

2. Click on the **Clients** link at the top-left of the application.

3. In **Client List**, click on **Options** gear icon to the which you want to delete.
4. Select **Delete**.



3.5.2. My Projects in TimeLive

My Projects page shows all projects in which current logged in user is assigned using **Project Team** option in Project.

A user can change his own email preferences by clicking on **Options > Preferences** icon.
A user can click on **Options** gear icon for View Project, Task, Team and Milestone pages as well but these options will be available only for those projects in which current user is either Project Manager or Team Lead.

Project Email Preference:

Through project Email Preference page, a user can configure different email preferences as per the Project.

View Project:

With View Project, a user can view/add/edit all the tasks and their progress, status in a single view. Further, can also view and edit Milestones, Team their hours booked for this project, and attachments uploaded to this project.

3.5.3. My Tasks

My Tasks page provides a list of all tasks assigned to logged in user. A user can add new task for assigned project and he can assign those task to himself as well as other team members of the project.

- By default, **My Tasks** page only shows uncompleted tasks which are assigned to currently logged in user.
- A user can do the search on other tasks also from here by clicking on **Search** button.
- In **Search** popup page, a user can enter search parameters and can see other tasks also other than assigned to the current user
- When clicking on the **Task** or the **Options** gear icon button, a user can view Task summary and view/add **Task comments**, **Task Team**, **Attachments**. Also, with **Options** gear icon, a user can **Edit Task**, **Rate History** and also delete this task.

3.5.4. Projects in TimeLive

A project is a framework (structure) used to organize and manage jobs assigned to different or all employees. Projects identify the body of work to be completed by a team within your organization. Projects can have multiple levels and complexities, depending on the needs of your organization.

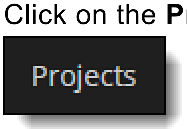
In TimeLive Administrator/Project Manager can:

- Create a Project for the Client.
- Create and set Project Team.
- Set Project Approvers, Timesheet and Expense Approval Path.
- Create many Milestones.
- Set Billing Rate Type.
- Set Project Estimation, budget.
- Set Project Start Date and End/Date.
- Attach File from the Attachment Tab

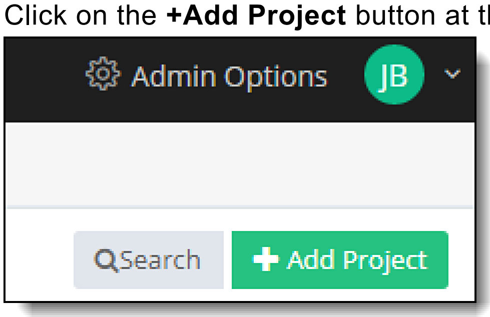
Creating a New Project

To create a new project, please follow below steps,

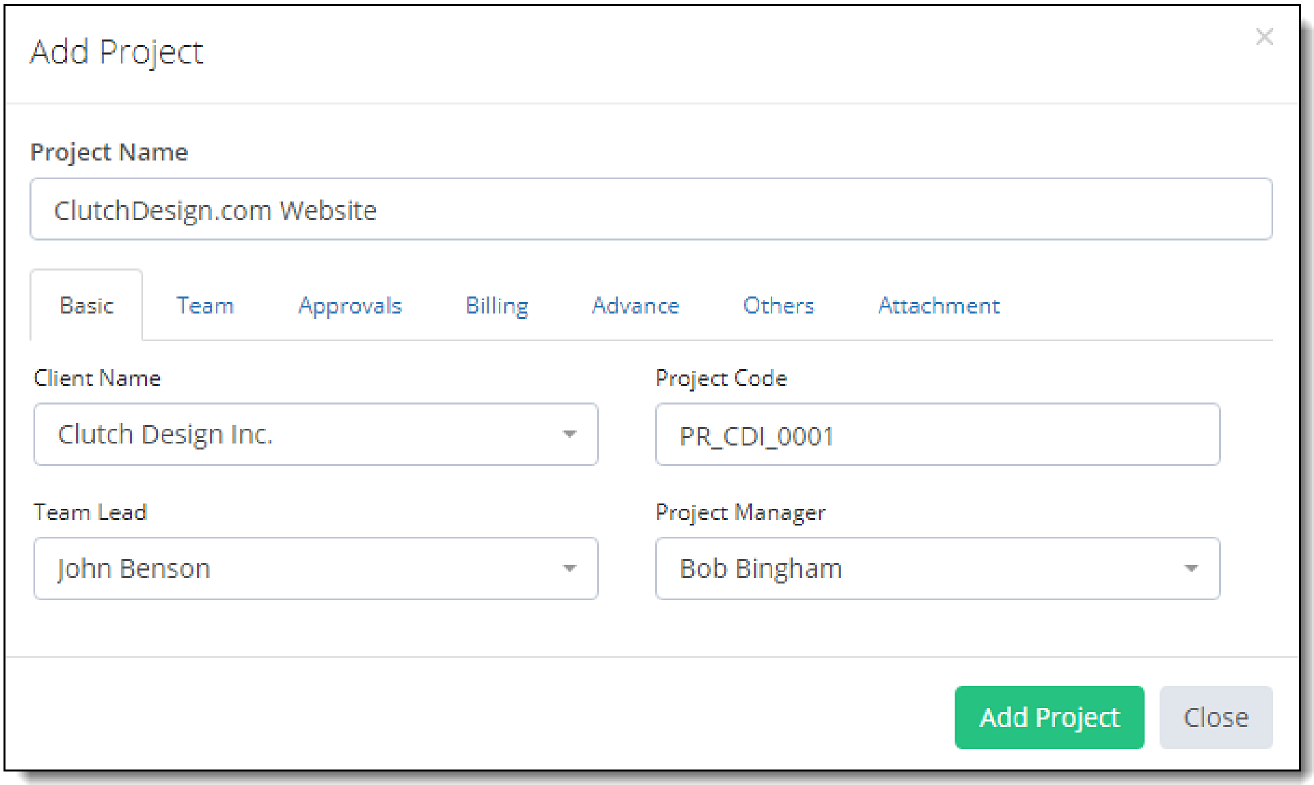
1. Click on the **Projects** link at the top-left of the application.



2. Click on the **+Add Project** button at the top-right of the application.



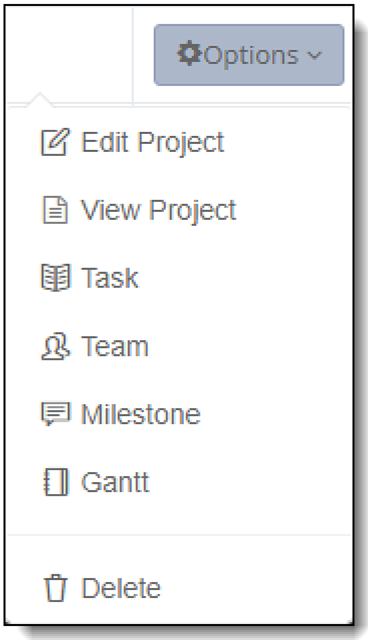
3. In the **Add Project** screen that appears, enter:



a. Fill the **Project Name** field and others as per the requirement.
b. Click **Add Project** button.

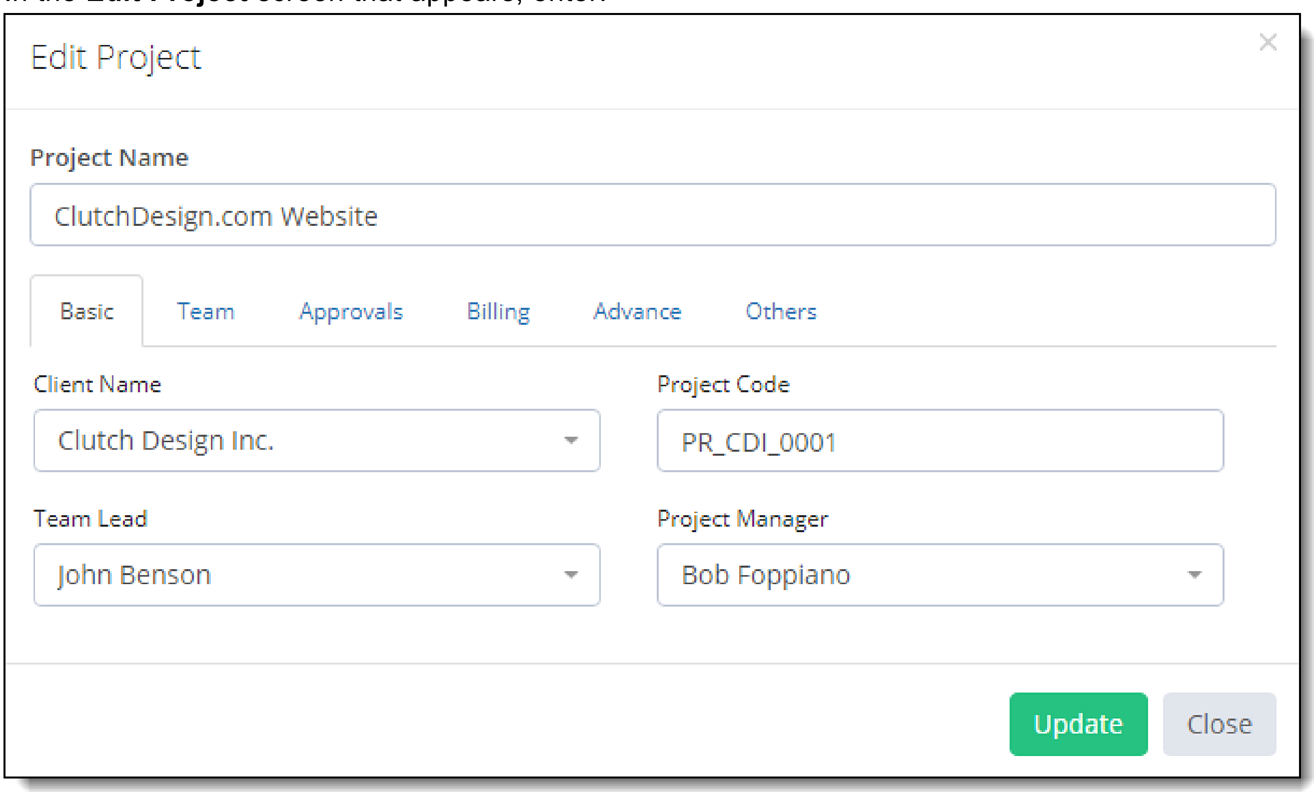
Edit existing Project

1. Navigate to the project which you want to edit and click **Options** gear icon.



   a. Select the option **Edit Project**.

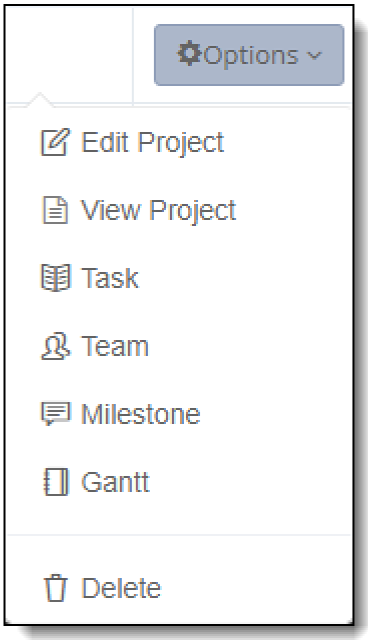
2. In the **Edit Project** screen that appears, enter:



a. Make the required changes in the project.
b. Click **Update** button.

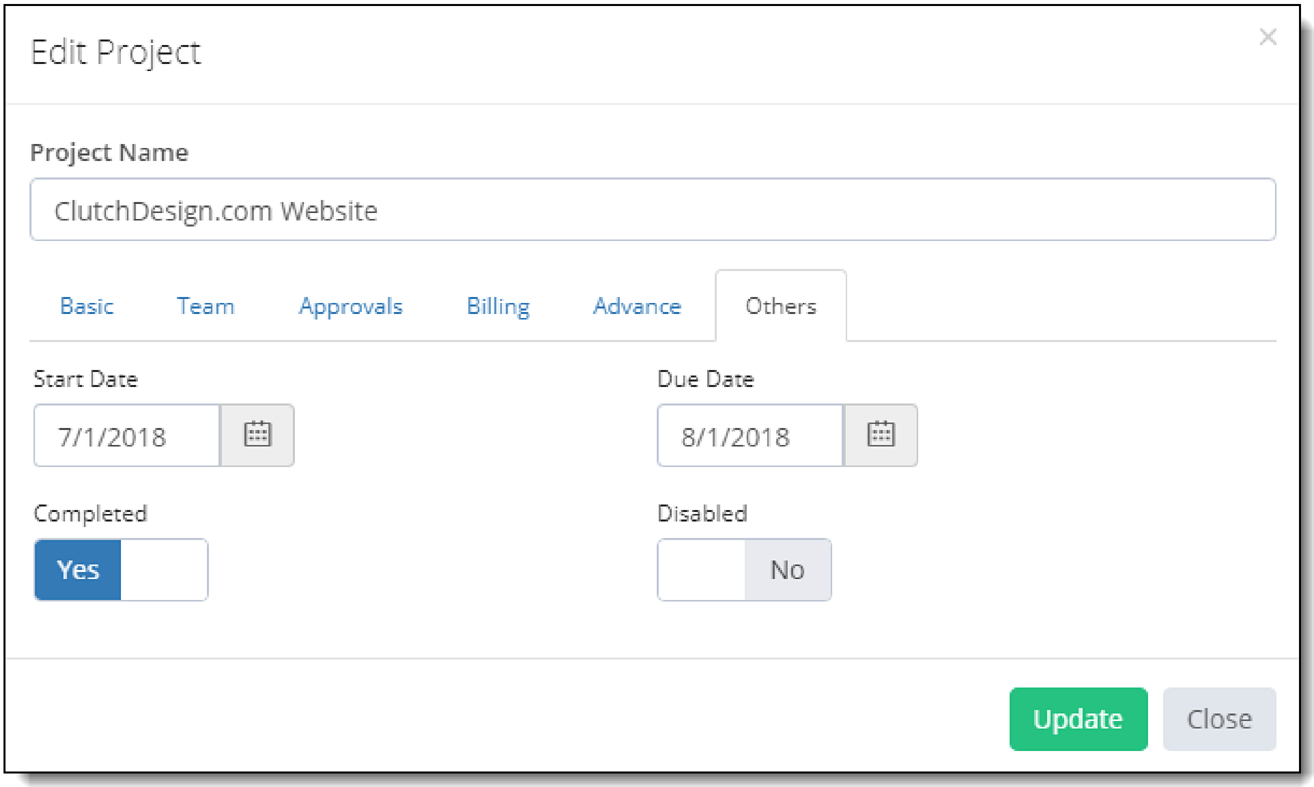
Complete/Disable the Project

1. Navigate to the project for which you want to Complete/Disable and click **Options** gear icon.



a. Select the option **Edit Project**.

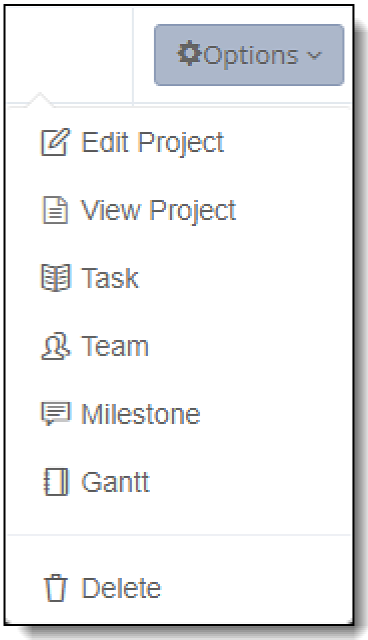
2. In the **Edit Project** screen that appears:



   a. Navigate and click on the **Others** Tab.
   b. Enable to Switch of **Completed/Disabled**.
   c. Click **Update** button.

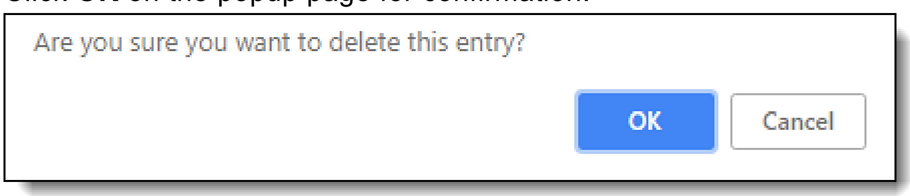
Delete the Project

1. Navigate to the project for which you want to delete and click **Options** gear icon.



   a. Select the option **Delete**.

2. Click **OK** on the popup page for confirmation.



Milestone in Project

See Milestone in Project

Create Tasks/Parent Tasks/Sub-Tasks in that project.

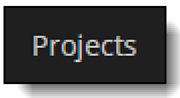
See Tasks/Parent Tasks/Sub-Tasks

3.5.4.1. Milestone in Project

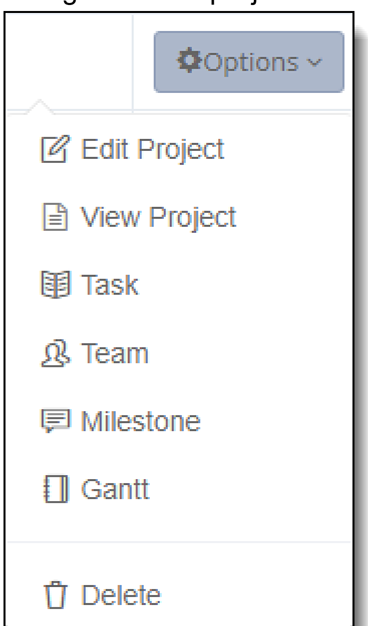
Milestones play a vital role when creating Project's Tasks. It allows administrator/project manager to manage tasks in sets which are called milestones. If 1 or more tasks created associated with milestone and that milestone is completed, all the tasks associated with this will be closed as well. Milestone can only be created after the project creation and should be created before any Project's Task.

Creating new Milestone

1. Click on the **Projects** link at the top-left of the application.

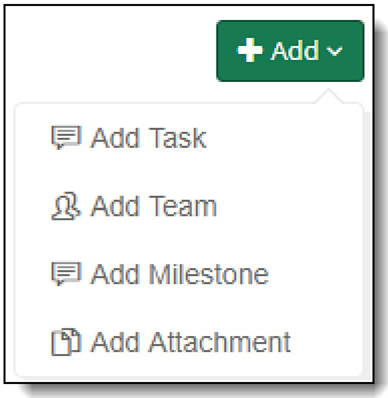


2. Navigate to the project for which you want to add the Milestone and click **Options** gear icon.



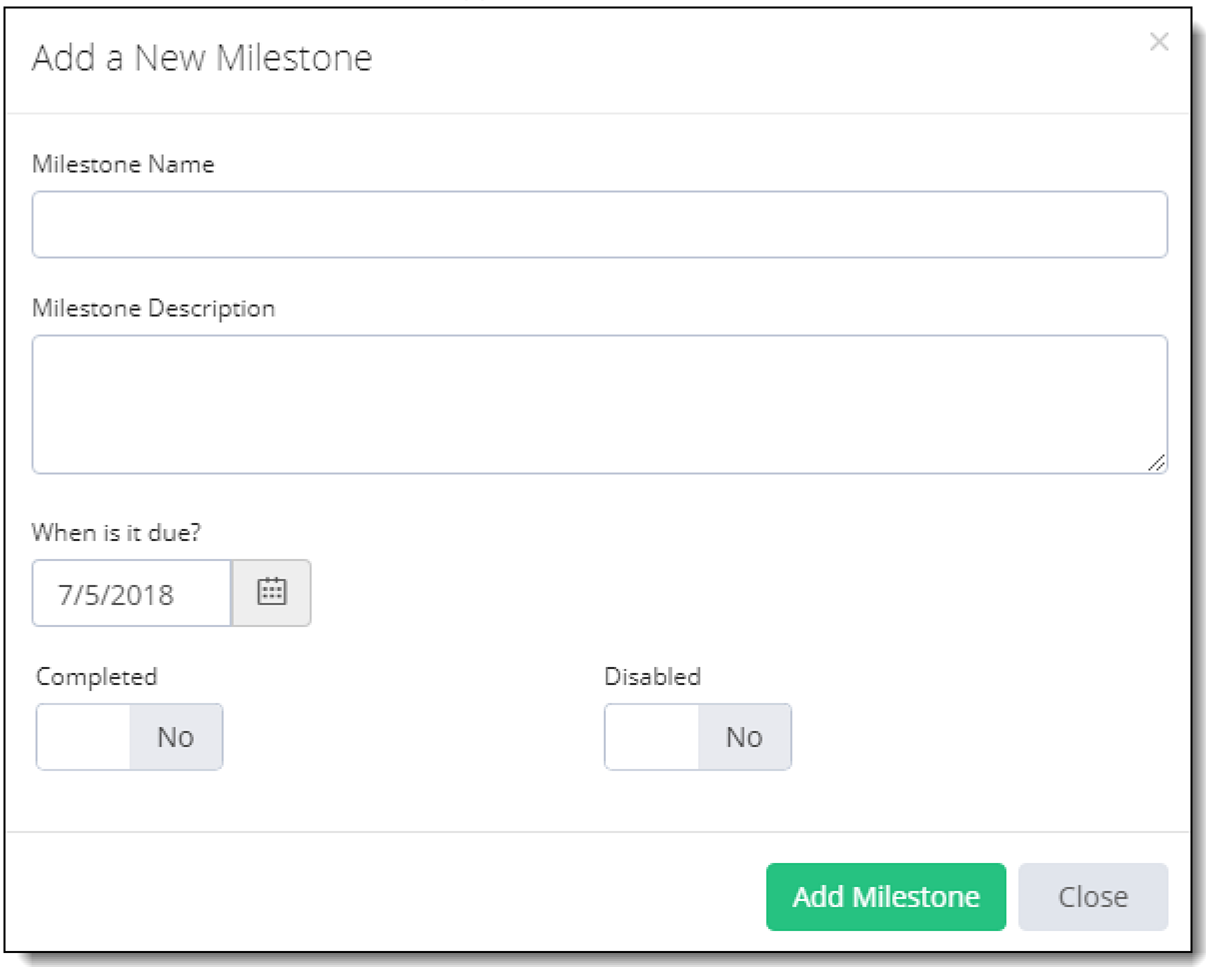
   a. Select the option **Milestone**

3. Click on the **+Add** button.



   a. Select the Add Milestone icon.

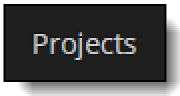
4. In the **Add Milestone** screen that appears:



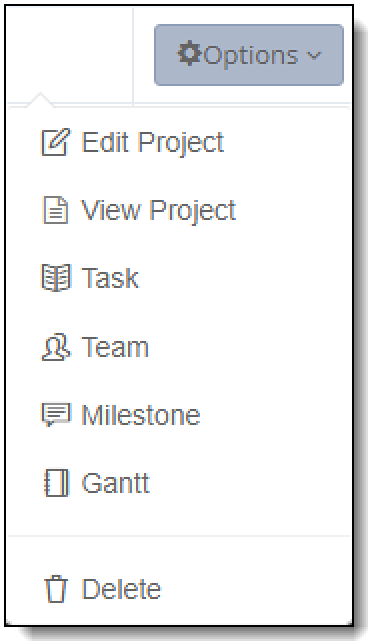
5. Fill the Milestone fields
6. Click **Add Milestone** button

Editing Milestone

1. Click on the **Projects** link at the top-left of the application.

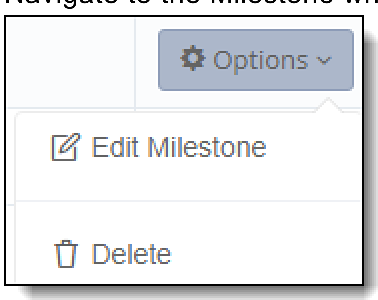


2. Navigate to the project for which you want to access the milestone and click **Options** gear icon.



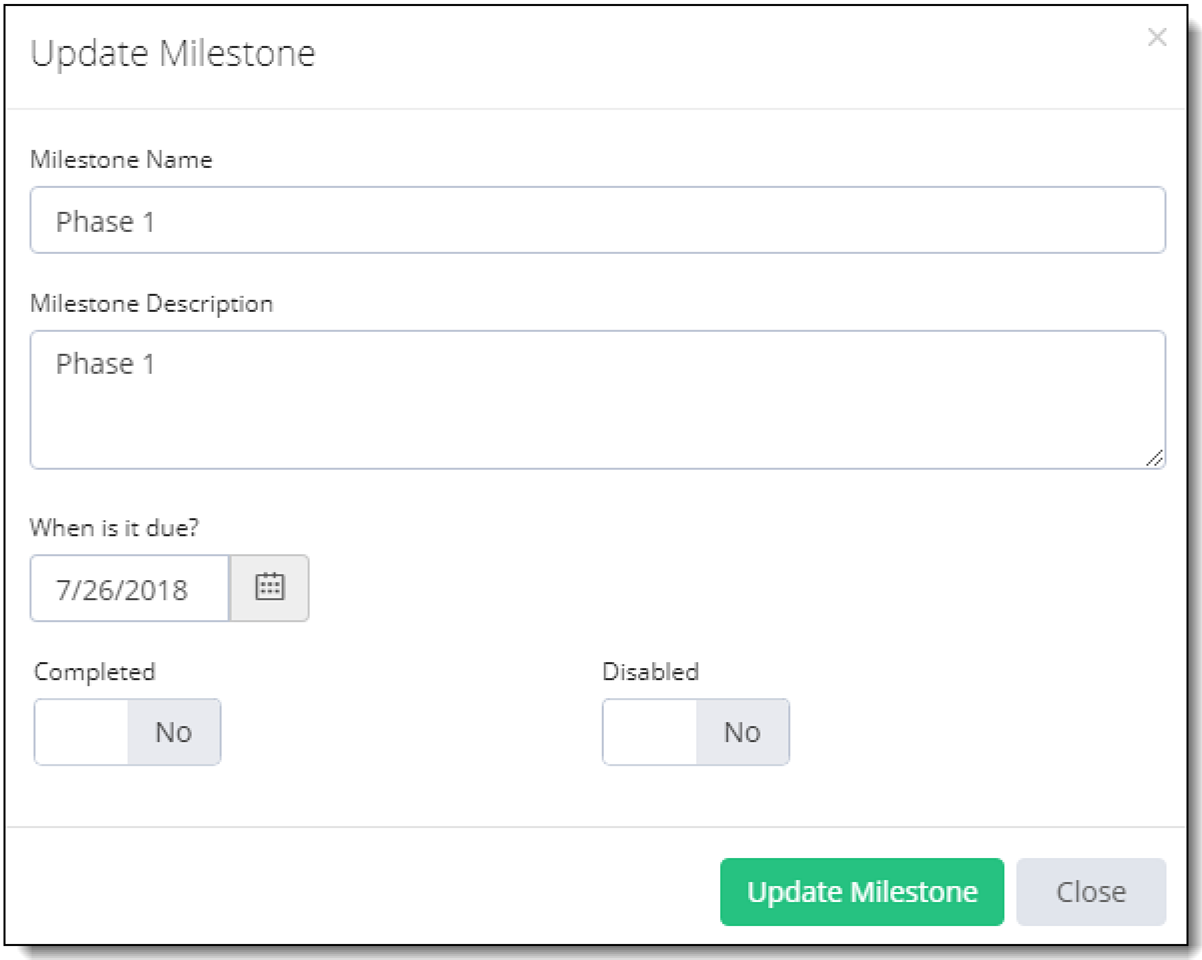
   a. Select the option **Milestone**.

3. Navigate to the Milestone which you want to edit the click **Options** gear icon.



   a. Select the option **Edit Milestone**.

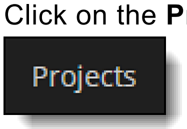
4. In the **Edit Milestone** screen that appears:



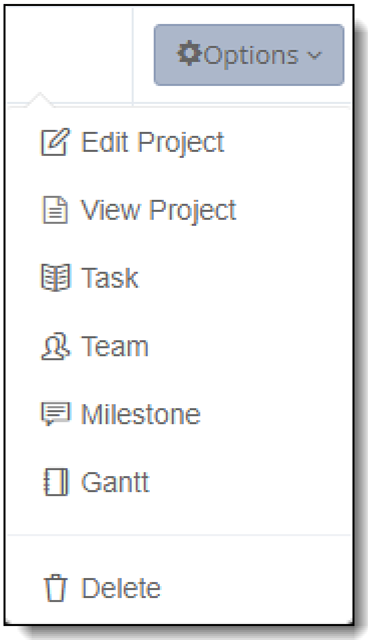
   a. Make the changes as per your requirement
   b. Click **Update Milestone** button.

Completing/Disabling Milestone

1. Click on the **Projects** link at the top-left of the application.

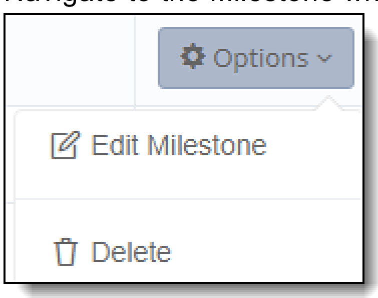


2. Navigate to the project for which you want to access the milestone and click **Options** gear icon.



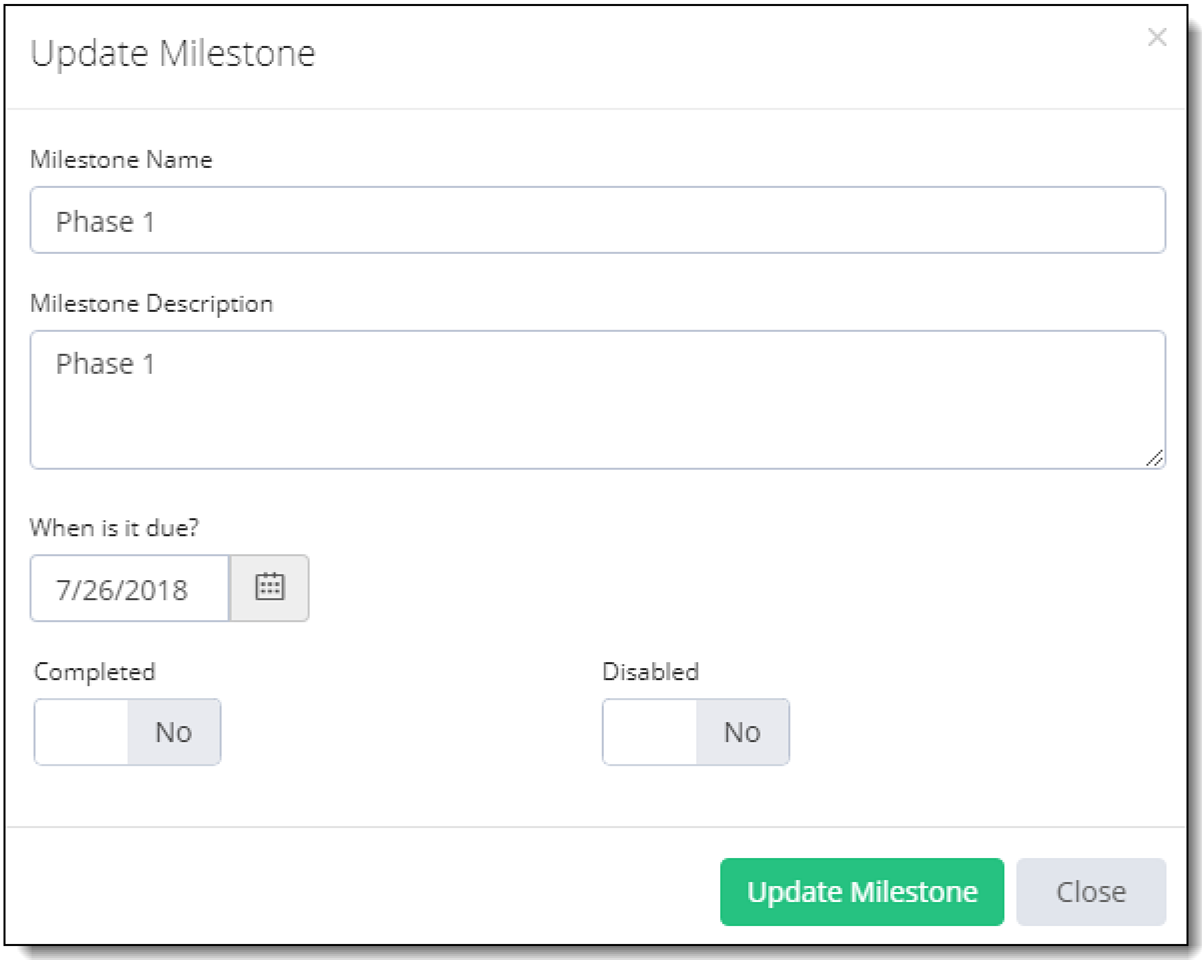
   a. Select the option **Milestone**.

3. Navigate to the Milestone which you want to Complete/disable and click **Options** gear icon.



   a. Select the option **Edit Milestone**.

4. In the **Edit Milestone** screen that appears:



   a. Click on the **Completed/Disabled** Switch.
   b. Click **Update Milestone** button.

3.5.4.2. Tasks

Create Task

If the project is being billed to a client, the project manager sets the billing rates for the project. The project manager creates tasks within the project and enters estimates. The project manager specifies which expenses can be claimed under the project. When entering time on their timesheet, users working on the project allocate their time against the project and task.

Setting Up Tasks

Projects can contain many tasks and up to ten hierarchical levels. Every employee who is assigned to a project can create a task(s) for that particular project. An administrator can customize task add permission using **Role Permission** option available in **Admin Options**.

More about task:

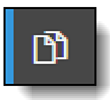
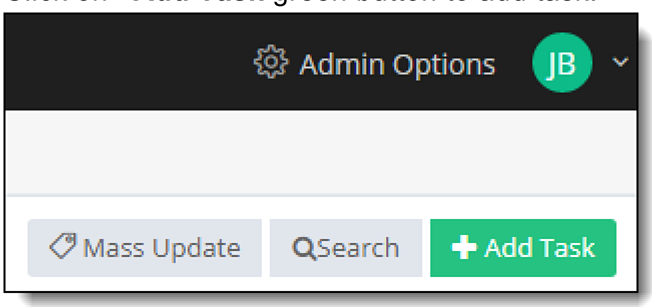
1. Task is the main source of project time sheet. Time entry can only be entered into a task of any project. An administrator should create projects and define project manager and team leader of projects.
2. **Administrator** or **Project Manager** or **Team Lead** can then define “Project Team”.
3. **Project Manager** and **Team Lead** then can create project tasks and assign them to project team in order to allow employees to enter time entry for that task.
4. Only user who is part of the project team can add a new task to a project.
5. A task can be created using **All Employee Task** checked. All employee tasks appear in time entry of all team members of the project.
6. A task can be created using **All Project Task** checked. This task will appear in Task column with all projects (it will only visible to those employees who are the part of this task’s Project).
7. A task can be created using both **All Employee Task** and All Project Task checked. Those tasks will be appearing for all projects and in all employees time entry.
8. Every task has their own status, priority, completed %, completed fields to monitor task progress for team collaboration purpose.
9. Every task has two fields Estimated Cost and Estimated Time spent for budgeting purpose.
10. Project manager can categorize their project tasks in different milestones (phases) for easier monitoring of individual milestones (phases) and then at project’s root level.

List of Task
A task can be viewed through these four different options.

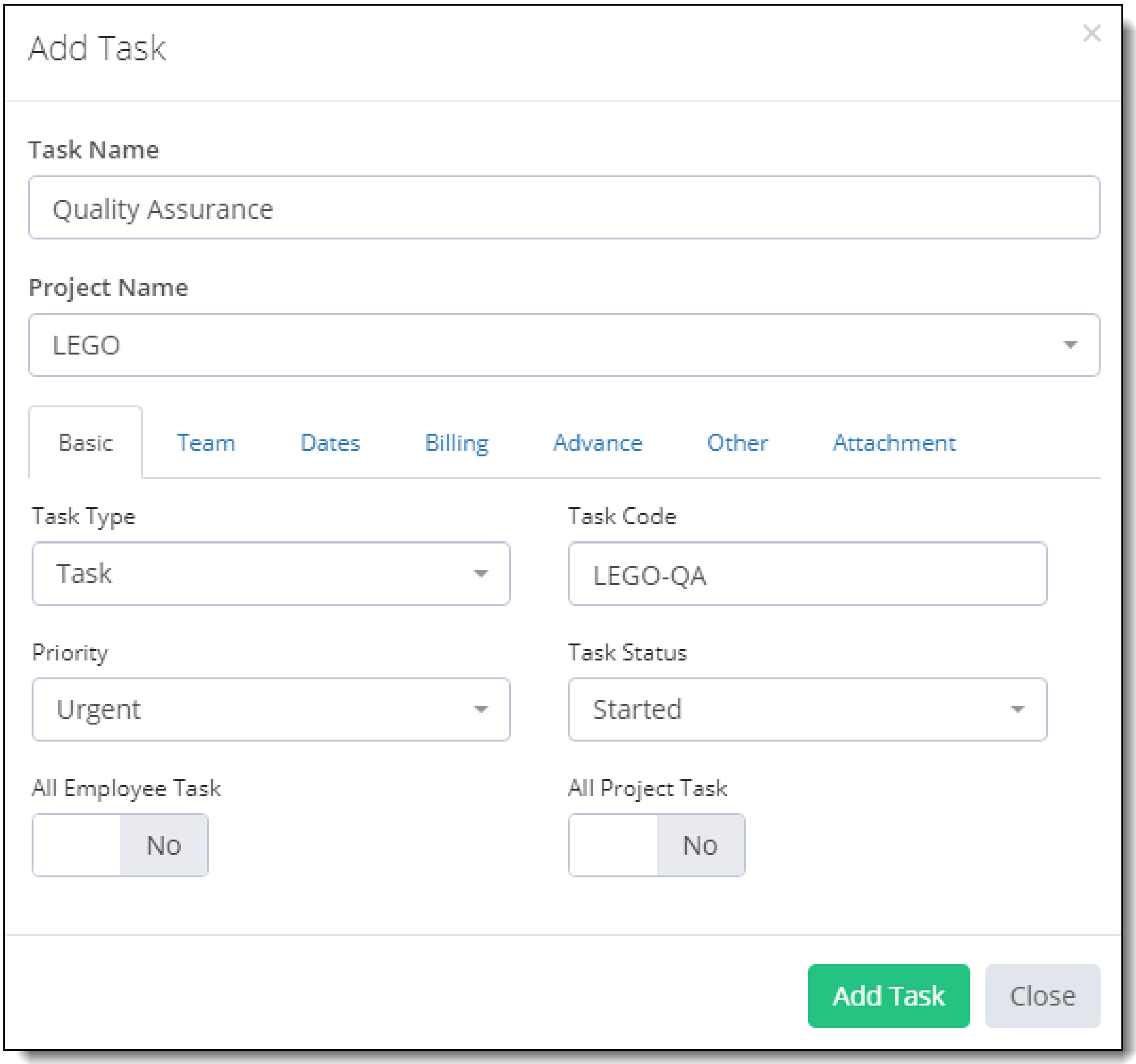
1. **My Task** (the only task which is assigned to the logged-in employee).
2. **My Tasks** (Task that is either assigned to logged in employee or task which are reported by the logged-in employee)
3. **Projects > Tasks** (Hierarchical view of all task for administration purpose). The administrator can add, edit and delete any task from there.
4. **My Projects > Task** (Hierarchical view of all task for administration purpose)

Adding New Task

A logged-in employee can add a new task in TimeLive and can assign this task to any other employee. To add a task:

1. Click on **My Task** icon from the left-side menu.

2. Click on **+Add Task** green button to add task.


3. In the **Add Task** screen that appears:



4. Fill the **Task Name**.
5. Select the **Project Name** from the drop-down list. (Note: If any project is not in the list, logged-in user is not in that Project team).
6. **Basic**
    a. Select **Task Type** (category of task. It depends upon usage to usage of TimeLive in different environments. Administrator can create new task types as per their organizational needs from **Admin options > Task Types**).
    b. Select **Task Code**.
    c. Select **Priority**.
    d. Select **Task Status**.
    e. Check **All Employee Task** (Check this field, if you want this task to have appeared in time entry of all employee. General routine tasks for which you don't want to add in all projects).

f. Check **All Project Task** (Check this field, if you want this task to have appeared in time entry of all projects. General routine tasks for which you don't want to add in all project)%(color-red)(it will only visible to those employees who are the part of this task's Project)%.

7. **Team**
   a. Select the Employees' names who should be the team members of this task.
8. **Dates**
   a. Select **Start Date**.
   b. Select **Due Date**.
9. **Billing**
   a. Select **Work Type**.
   b. Select the **Billable** Switch (if you want this Task hours to be billed to the client).
   c. Select **Billing Rate Start Date**.
   d. Select **Billing Rate End Date**.
   e. Select **Employee Rate Currency**.
   f. Set **Employee Rate**.
   g. Select **Billing Rate Currency**.
   h. Set **Employee Rate**.
10. **Advance**
   a. Select **Milestone** (Project manager can categorize their project tasks in different milestones i.e. phases for easier monitoring of individual milestones (phases) and then at project's root level).
   b. Select **Estimated Cost Currency**.
   c. Set **Estimated Cost**.
   d. Set **Estimated Time (Hours)**.
   e. Set **Duration (Hours)**.
   f. Select Switch **Parent Task** (check this field, if you want this field to be added as parent field. This field will then come in parent task dropdown during task add).
11. **Other**
   a. Fill **Task Description**.
   b. Set **Completed%**.
   c. Select **Completed** (This option is discussed below).
12. **Attachment**
   a. Select any File for attachment which is associated with this Task if required.
13. Click **Add Task** green button to add this Task.

Editing Existing Task:

A logged-in employee can modify any task by clicking on any task in **My task** task list. An employee can update task title, task comments, task status, completed, completed % in task modification form.

1. Click **My Task**.
2. Scroll to the task which needs to be changed.
3. Click on the **Options** gear icon against the task.
4. Click **View Task** option.
5. Click on the **Options** gear icon against the task.
6. Click **Edit Task** option.
7. Make the changes in the required field(s) then click **Update Task**.

Completing the existing Task:

When a task is completed, there are two ways to complete it in the system.

1. Complete the **Project Milestone** and all the tasks which are created under it will automatically be completed.
2. Complete the **Parent Task** and all its **Sub-Tasks** will be completed automatically.
3. Edit the Task, Navigate to **Other** Tab, Change the **Completed** switch to **Yes** and click **Update Task**.

Deleting the existing Task:

1. Click **My Task**.
2. Scroll to the task which needs to be deleted.
3. Click on the **Options** gear icon against the task.
4. Click **View Task** option.
5. Click on the **Options** gear icon against the task.
6. Click **Delete** option.
7. Click **Yes** for the confirmation.

Adding Task Comments:

Task Comments are the history of comments from different task related users in order to communicate different issues/updates/comments for that particular task within the team. Task comments cannot be deleted once it entered.

To enter task comments:

1. Open task by clicking on task either by **My Tasks**,**My Projects > Tasks** or **Project > Tasks**
2. In task comment form, enter comments title and comment description.
3. Click on **Add** to add that task comments.

3.5.4.2.1. Parent Task

Parent Task:

Parent Task is a high-level item within task breakdown that is specified with greater detail by a number of related sub-tasks or child tasks. It is the root task that subordinates its sub-tasks and defines how they should be performed and related to each other.

The term “parent task” is used in task planning practices to explain the central and major element of a task decomposition design. Parent task has the following characteristics:

- Splitting: the task can be split or divided into a series of simple operations or related sub-tasks
- Goal definition: it defines the direction and subordinated goals of related sub-tasks
- Forecast outcome: results of the task can be forecast by planning its sub-tasks ahead

Parent task is always at the top of task hierarchy. It may have several child tasks of the same level or multiple sub-tasks that belong to different hierarchical levels of the same hierarchy. Sometimes, parent tasks with multiple hierarchical levels and sub-tasks are regarded as projects or sub-projects. Such tasks are broad and included as elements of project breakdown structures and trees.

Making Task Parent:

1. Login TimeLive with Admin/managerial rights.
2. Navigate to **Projects** link at the top-left of the application.
3. Click on the **Options** gear icon against the project in which you want to create a sub-task.
4. Select **Task** option.
5. On the Task List, click on the Task in which you want to create a Sub-Task.
6. Click on the **Options** gear icon.
7. Click **Edit Task**.
8. Navigate to **Advance** tab.
9. Enable the switch of **Is Parent** to **Yes**.
10. Click **Update** green icon to confirm.

3.5.4.2.2. Sub-Task (Child Task)

A Subtask is also called Child Task. Creating a subtask can be helpful when on a project there are some major tasks but within that tasks, few of the small tasks which need to be dealt with a different approach/ team. Subtasks support people when they need to delegate parts of their assigned work to other people, but want to keep control over the overall result. They can also be used to invoke supporting services to help people accomplish the tasks that they are working on.

In TimeLive, in order to create a Sub-Task, a manager/administrator first make the task a Parent. The good thing about TimeLive is, there are no restrictions in how many levels of subtask manager/administrator create so any task can be a parent even the subtask so that further subtasks can be created in it. A manager/administrator can create a new task as a parent or make existing task parent.

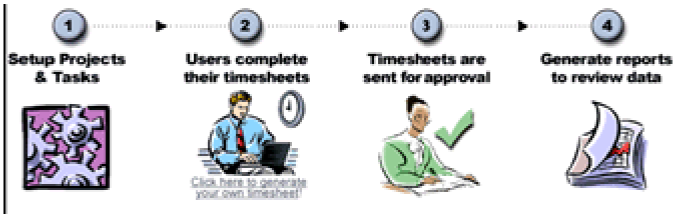
Creating a new Sub-Task/Child-Task

1. Login TimeLive with Admin/managerial rights.
2. Navigate to **Projects** link at the top-left of the application
3. Click on the **Options** gear icon against the project in which you want to create a sub-task.
4. Select **Task** option.
5. On the **Task List**, click **+ Add** drop-down button.
6. Select **Add Task** option.
7. Navigate to **Advance** tab and select **Parent Task** from the Parent Task drop-down field.
8. If you want to further subtask of this child task, enable the switch **Is Parent**.
9. Click **Add Task** button.

3.6. 1.6 Quick Basic Setup TimeLive

Understanding the Workflow:

The following diagram explains the general flow of work in TimeLive.



1. Create projects and tasks and assign project teams, rates, and billing options to allow users to enter billable and non-billable time and expenses. You will be able to track all time and expenses to calculate project costs and ensure accurate client billing.

1. Create generalized tasks with [All Employee Task] and [All Project Task] checked. Generalized tasks don’t require any assignment. Generalized tasks appear for all projects in time entry.
2. Employee completes their timesheets by entering time against projects or time off codes. Users can enter time only for those tasks for which they are part of the task team. All time is either billable or non-billable. Once completed, users submit their timesheets for approval.
3. Once submitted users’ timesheets are sent to the appropriate project leaders, clients, and supervisors for approval. Approvers can approve or reject the timesheets. Approved timesheets are closed so the data can be processed.
4. Any user with report access can generate reports to review data in the system including project status, timesheet status, billing, time off, and utilization.

How to perform quick setup steps:

Following are 4 simple steps which will allow you to get started using TimeLive. For more detail information for each of the setup,

1. **Creating of Clients:**
   Create Client from left-top link of Clients

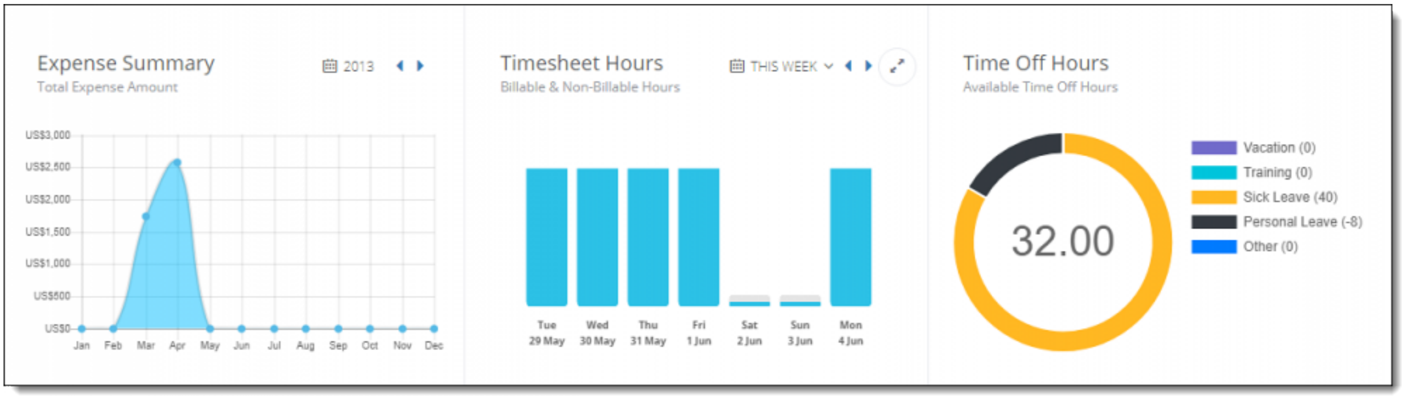
2. **Adding employees:**
   Add Employees from Top-Left Link of Employees. All newly created employees will receive email from TimeLive with their login details

3. **Creating Projects:** (Click to launch detail help of adding Clients)
    a. Time entry and expense entry can only be entered for project tasks.
    b. So you must have project setup before adding any task for your projects.
    c. You should also assign your project team in order to allow your team to enter time entry record for a particular project.

4. **Creating Tasks:** (Click to launch detail help of adding Clients)
    a. You can add tasks to your project and you can assign those tasks to the employee.
    b. Only tasks which are assigned to an employee will appear in time entry page for entering time entry record

3.7. 1.7 Dashboard in TimeLive

A dashboard is a user interface that organizes and display TimeLive Data in the graphical representation which is very convenient to view and comprehend for any user.



4. Timesheet

4.1. Timesheets

After you have completed the setup process, users can begin entering billable and non-billable time and expenses and tracking when they start and stop work. You will be able to track project costs and client billing, timesheet submission status, time off usage, and available vacation.

This topic will explains the actions involved in daily usage of timelive, including:

- Entering, saving time record using time entry **day view**. (Submit button is only available in period view).
- Entering, saving, and submitting time record using time entry **period view** (generally its a Week View).

More about TimeLive timesheet entry

1. Timesheets are used to enter time against specific projects and tasks or activities and time off using either day-view or week-view.
2. Users complete and submit their timesheets from week-view to supervisors, project leaders, and/or clients for approval.
3. The timesheets are then reviewed and either approved. When all items are approved, the time information is saved to the database.
4. Once a timesheet is submitted, the data can be analyzed by project leaders and supervisors to determine project status and costs, and to make more efficient resource planning decisions.
5. Billing detail can be generated after submitting of the timesheet.

Timesheet formats:

TimeLive provides **two timesheet formats** for entering time entry record in TimeLive.

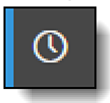
- **Standard Timesheet** (just with **total time** of a time entry record. It is useful in an organization where only **TotalTime** information is required for billing and analyzing)
- **Time entry with the Start time and End time** (additional start time/end time field with standard total time field can be enabled from **Admin Options > Preferences > Timesheet > “Show Clock Start/End”**)

Entering hours in Timesheet for the first time.

After creating Client, Employees, Project and its Tasks, billing rates, employee rates and assigning employees in Project and the task, users can book their hours through below steps,

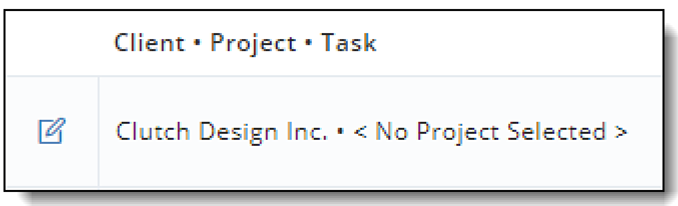
Filling Timesheet with Day View:

1. Navigate to My Timesheet.

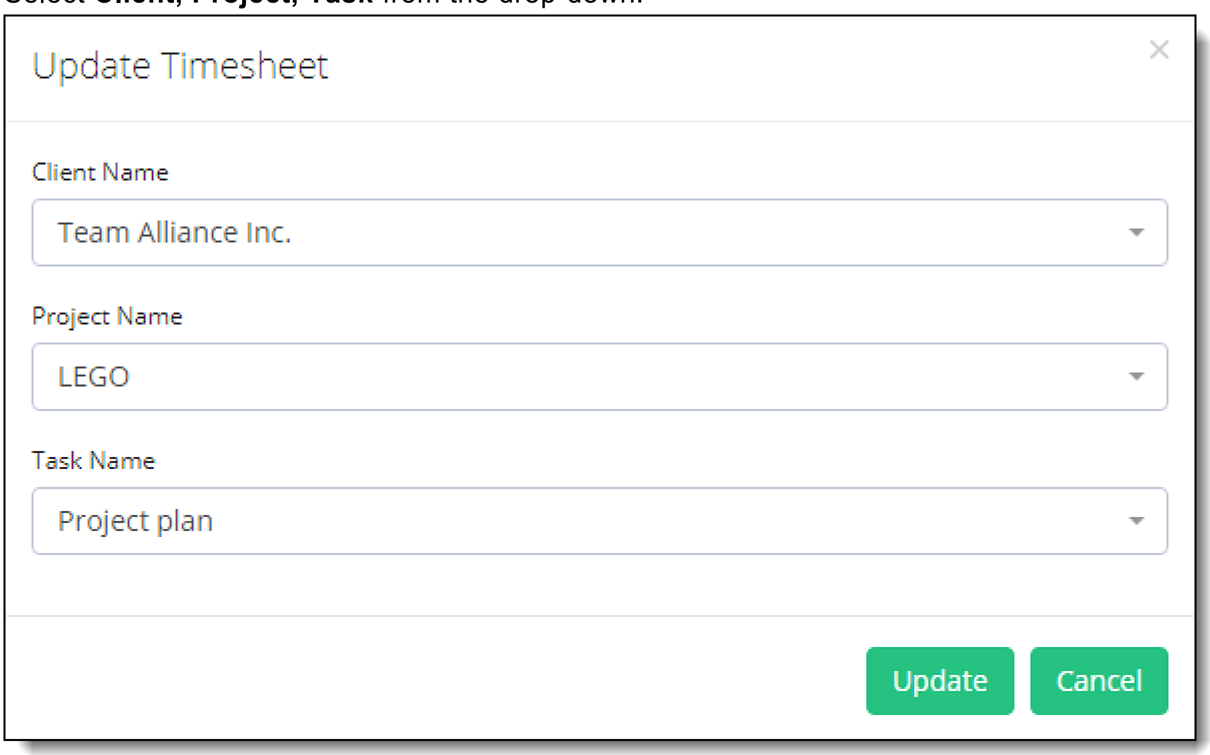


2. Click on **Day View**.

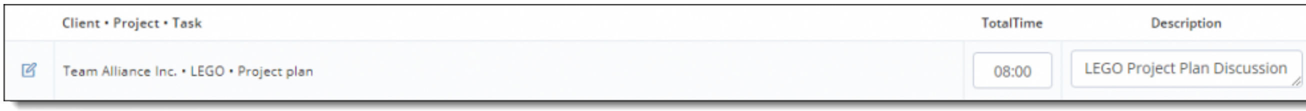
3. Click on the **Edit** Icon on the left of the screen.



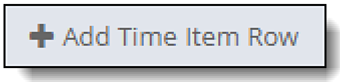
4. Select **Client, Project, Task** from the drop-down.



5. Enter the **Total Time** (Start/End Time if enabled from Preferences) and the description (optional).



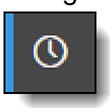
6. Click **+ Add Time Item Row** if required for more hours to enter.



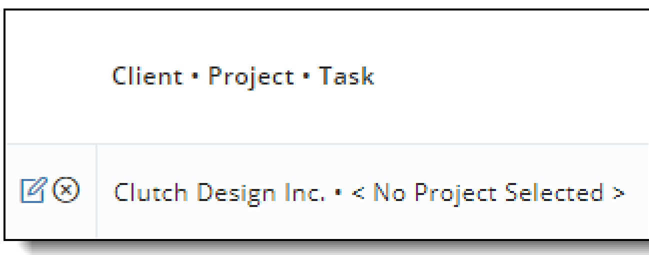
7. Click **+ Add Time Off Row** if required for Time Off hours to enter.
8. Click **Save** button on the top.

Filling Timesheet with Period View(Weekly View):

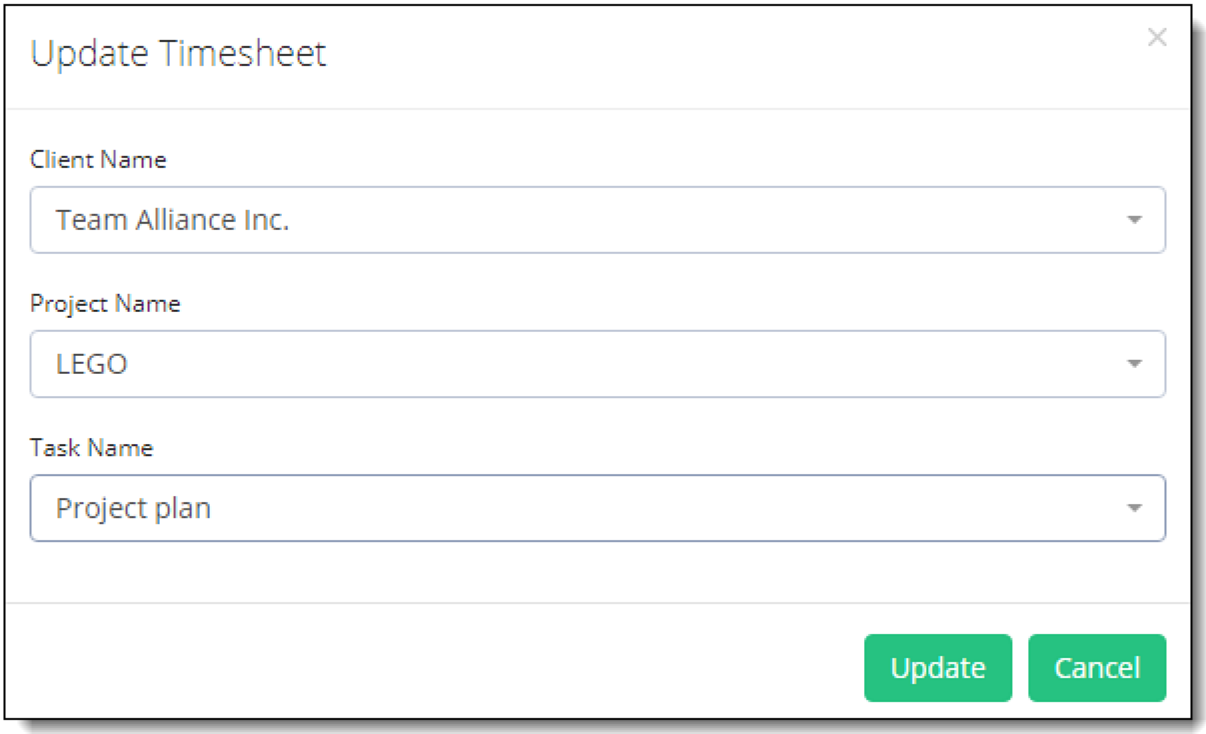
1. Navigate to My Timesheet.



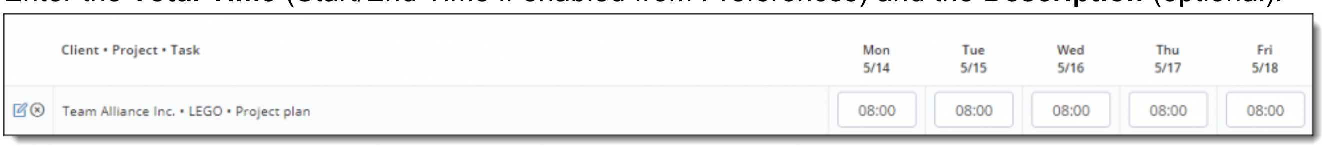
2. Click on the **Edit** Icon on the left of the screen.



3. Select **Client, Project, Task** from the drop-down.



4. Enter the **Total Time** (Start/End Time if enabled from Preferences) and the **Description** (optional).



5. Click **+ Add Time Item Row** if required for more hours to enter.

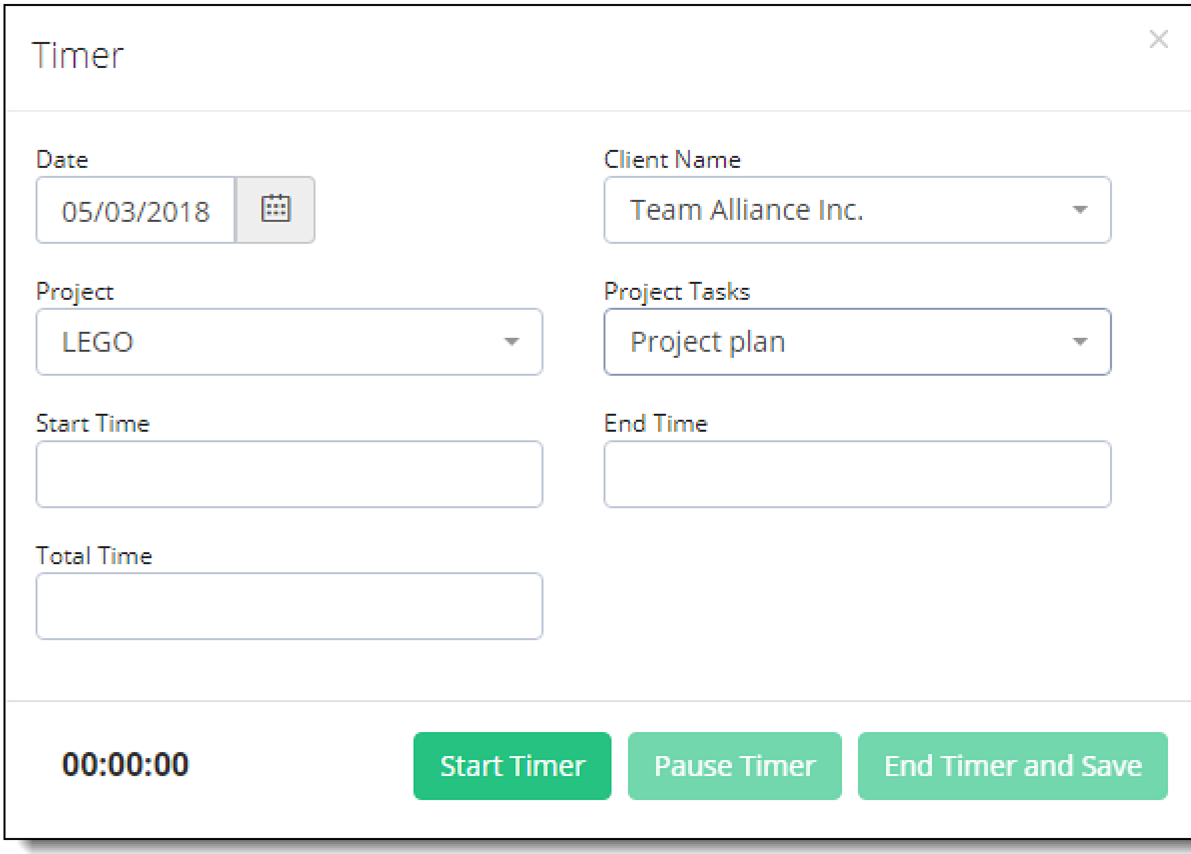
6. Click **+ Add Time Off Row** if required for **Time Off** hours to enter.
7. Click **Submit** button on the top to save the Timesheet and submit it for the approval process.

Filling Timesheet with Timer:

1. Navigate to My Timesheet.

2. Click on the **Start Timer** button.

3. Select the **Date, Client*(if enabled from *Admin Options > Preferences**) **Project, Task**.



4. Click **Start Timer** button.

5. Click **End Timer** when you are done with the task and want to save the worked hours.

6. Click **Submit** button on the top to save the Timesheet and submit it for the approval process.

4.2. Time Entry – Day View

About Day View:

Through Day View, a user can specifically focus on a particular day and fill timesheet of that day as per the assigned Projects and tasks and do not have to worry about any other day in this view. Day View Can be accessed through My Timesheet > Day View

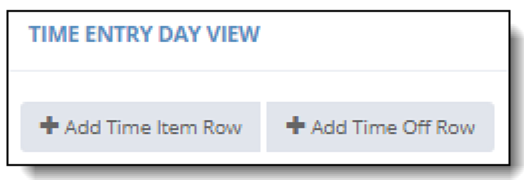
Features of Time Entry – Day View:

1. Easy single format of entering whole daytime entries for different project and task in single save.

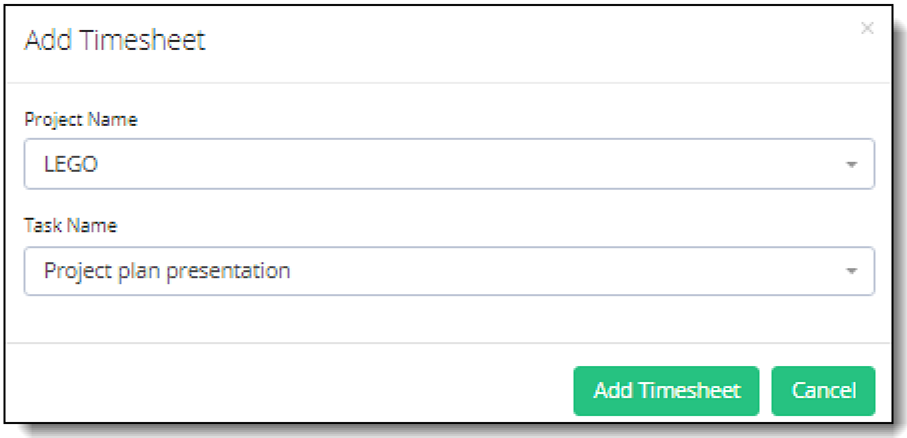
2. Day view of time entry also supports both two types of format. You can customize this option using **Admin option > Preferences (Show Clock Start/End]** option.
    - **Standard timesheet** (just with total time of a time entry record)
    - **Time entry with Start time and End time** (additional start time / end time field with standard total time field)

Adding Time entry using Day View

1. Click **Add Time Item Row** button on the left.



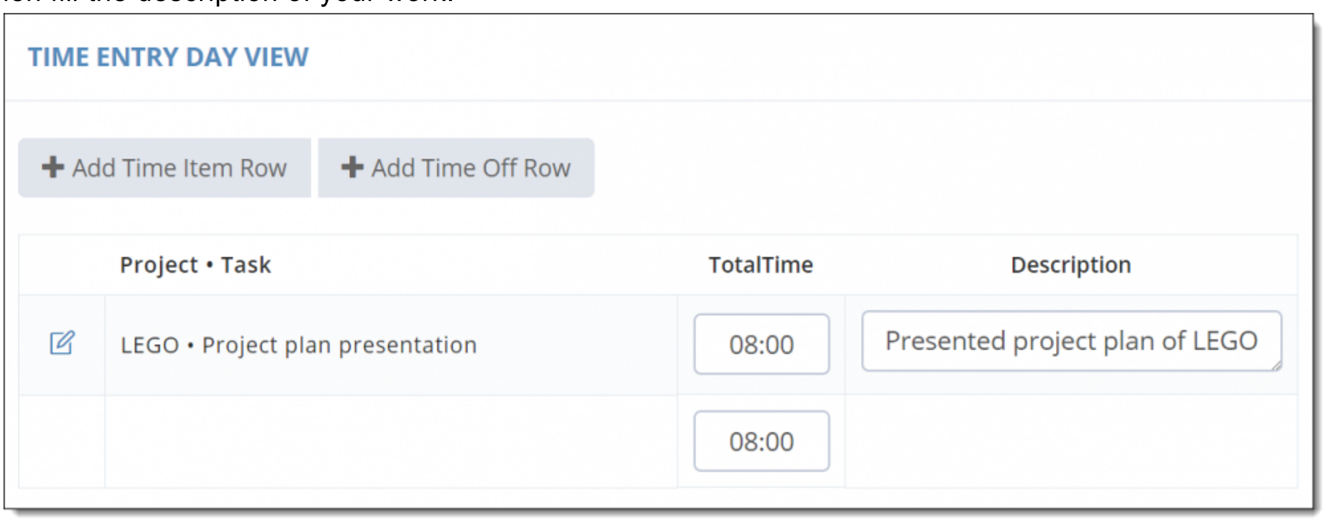
2. In the **Add Timesheet** screen that appears, enter:



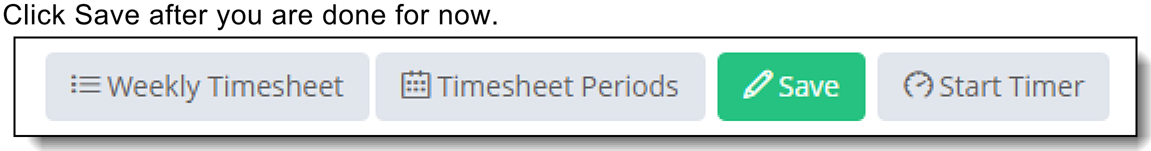
- Select Project Name from the popup option and the Task.
- Then click green button **Update**

3. Then fill the hours(if Start Time/End time option is enabled) fill the end time as well).

4. Then fill the description of your work.



5. Then Click Save after you are done for now.

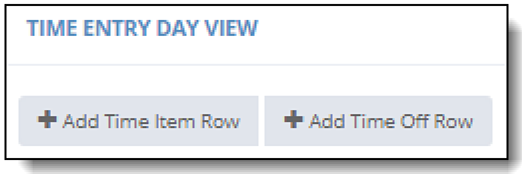


Your activity on this sheet will only be saved and will not be forwarded for approval till you submit it from My Timesheet view. Same activity will be shown in My Timesheet as well.

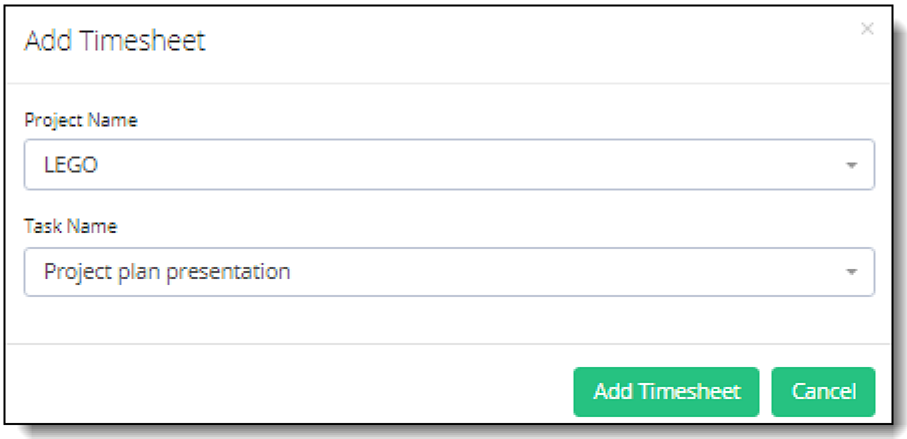
Adding More Lines to the Day View (Timesheet / Time Off)

Most of the time, users work more than one Project and tasks and sometimes take small offs while working like Lunch, Tea. Then through below steps they can add more lines in the day view and fill their timesheet as per their requirements.

1. **Click + Add Time Item Row** button.



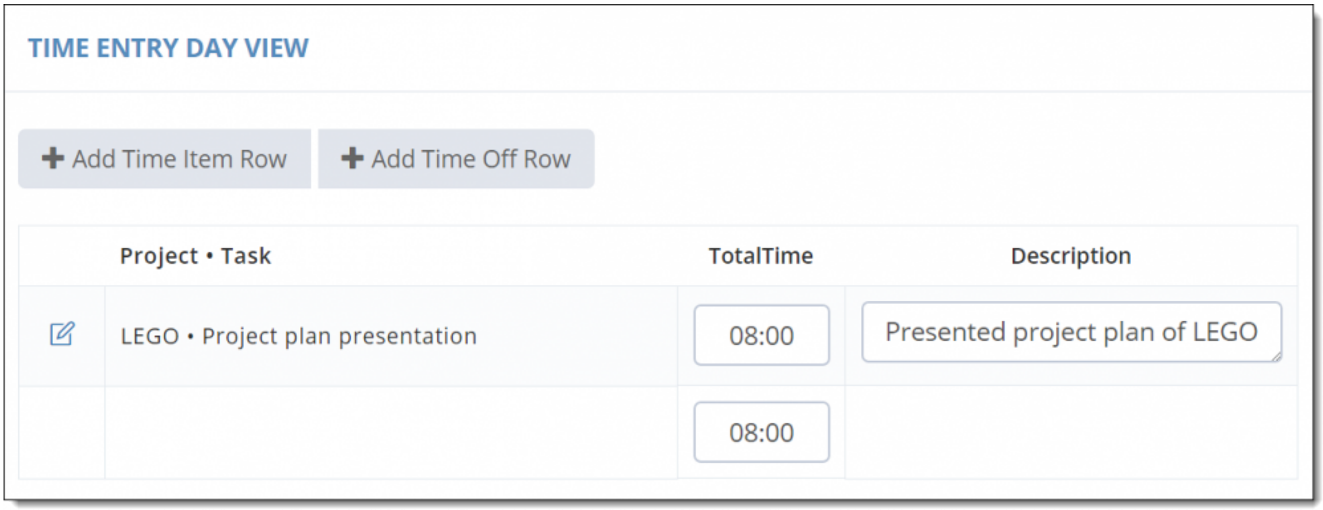
2. In the **Add Timesheet** screen that appears, enter:



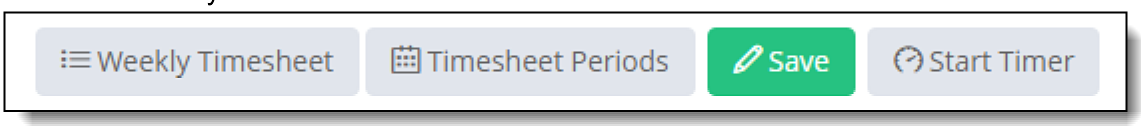
- Select Project Name from the popup option and the Task.
- Then click green button **Update**.

3. Then fill the hours(if Start Time/End time option is enabled) fill the end time as well).

4. Then fill the description of your work.



5. Then Click Save after you are done for now.



Deleting of time entry record:

1. Click on **[Delete]** Icon next to Edit Icon on the left of record which you want to delete.
2. Click on **[Ok]** to confirm this delete.

Updating time entry record:

1. Navigate to field which you want to change.
2. Click on **[Edit]** Icon next to Delete Icon on the left of record which you want to edit.
3. Select Project Name from the popup option and the Task
4. Then click green button **Update**
5. Then fill the hours(if Start Time/End time option is enabled) fill the end time as well)
6. Then fill the description of your work
7. Then Click Save after you are done

Submitting time entry record:

Submitting time entry can be done only from week-view(Period-View). Just navigate to week-view and click on submit button to submit that whole Period.

Adding Time entry record for Attendance using Day View:

1. Click on the green icon [+ Add Attendance]
2. Select the first drop-down In from In/Out Option
3. Then fill your Time-In
4. Then Select Present or any Status you want to fill against this time.
5. Then Click Add Attendance green button.

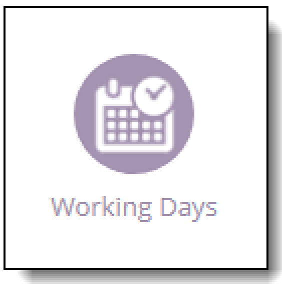
4.3. Time Entry – Period View

About Period View:

Using **Period View** of time entry, the employee can enter daily time sheet with full view of the whole period (usually Weekly View). An employee can submit these time entries for approval by clicking on **Submit** button. Time entry week view can be accessed using **My Timesheet > Period View**

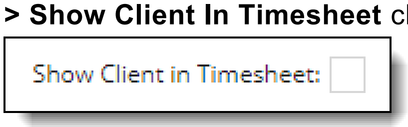
Features of Time Entry – Period View:

1. Easy single format of entering whole week time entries for different project and task in single save.
2. Period View can be configured using **Admin Options > Working Days**.



3. Period view of time entry also supports both two types of format (standard and with start time and end time). You can customize this option using **Admin option > Preferences (Show Clock Start/End)** Option.

4. A client selection drop-down can also be enabled in Time Entry using **Admin Options > Preferences > Show Client In Timesheet** check.



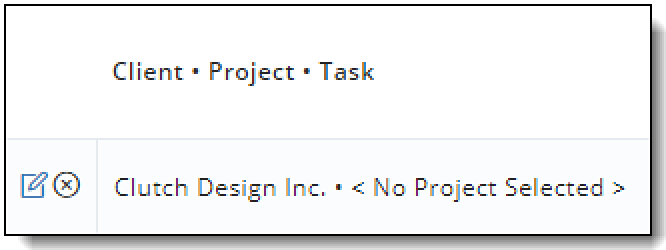
5. **Time entry with the Start time and End time** (additional start time/end time field with standard total time field)

Adding Time entry record using Period View:

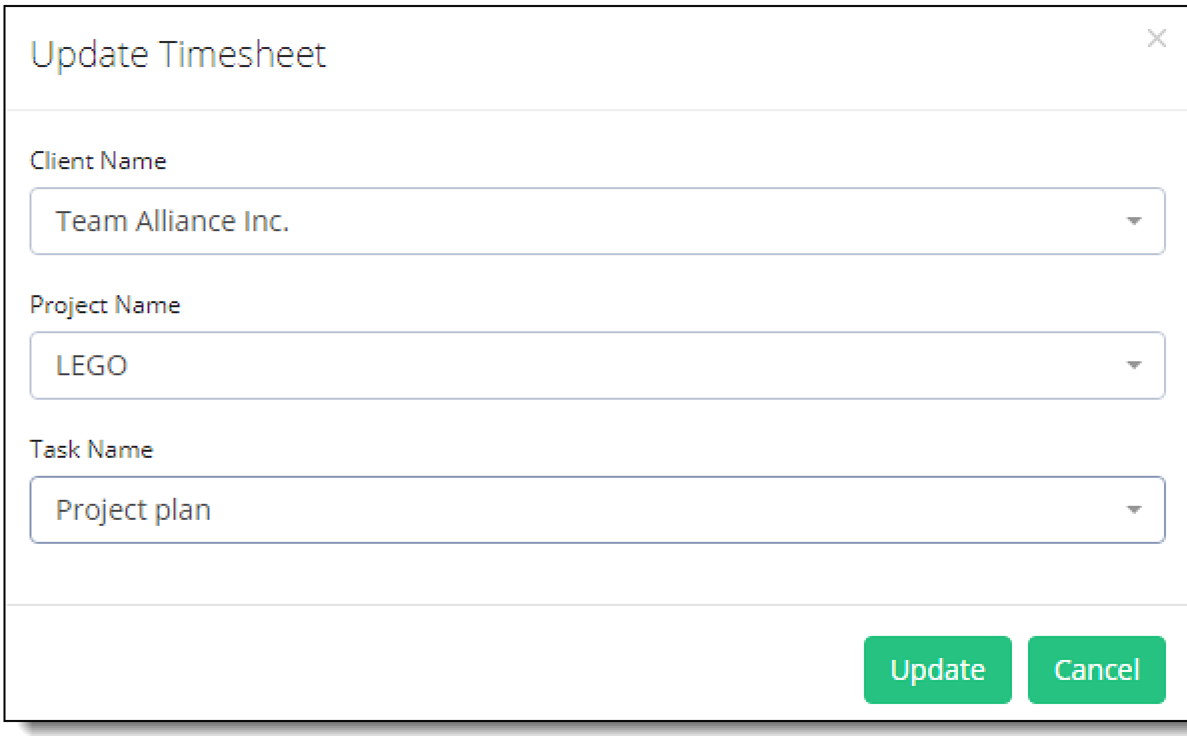
After creating Client, Employees, Project and its Tasks, billing rates, employee rates and assigning employees in Project and the task, users can book their hours through below steps,

1. By default, TimeLive always give one blank rows in week view to enter new timesheet record.

2. Click Edit Icon on the left.

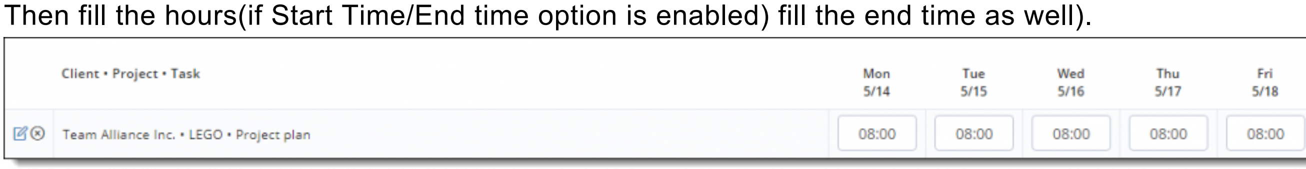


3. Select Client Name. Project Name from the popup option and the Task.



   a. Then click green button **Update**.

4. Then fill the hours(if Start Time/End time option is enabled) fill the end time as well).



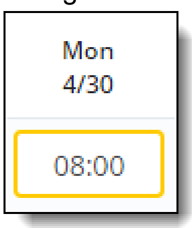
5. Then fill the description of your work.
6. Then Click **Save** after you are done for now. Your activity on this sheet will only be saved and will not be forwarded for approval.

7. Click on **Submit** Button to sent it for the Approval process.

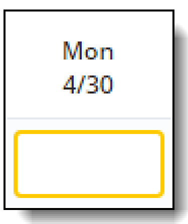
You can continue on adding a new row in time entry record from next row. If you want to add more row, save it first and then TimeLive system will give additional 1 row for entering more records.

Removing the saved time entry record:

1. Navigate to **Total Time** field either using a mouse or a keyboard.



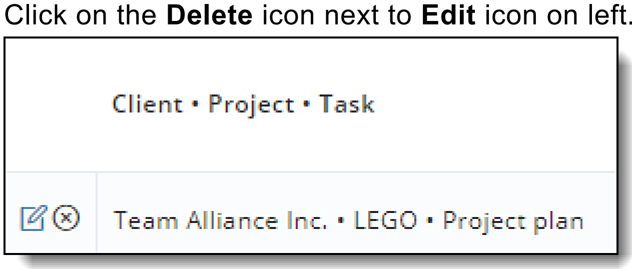
2. Remove information in **Total Time** field which is already entered.



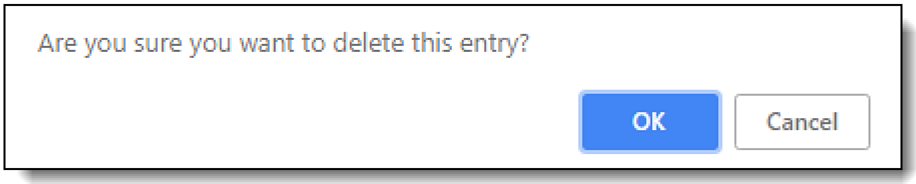
3. Click on **Save** button to remove the saved time entry record.

Removing the whole Entry with Project and Task:

1. Click on the **Delete** icon next to **Edit** icon on left.

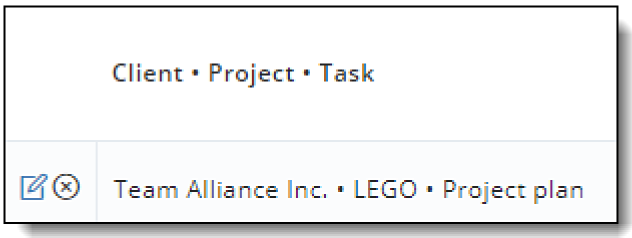


2. Click **Ok** on the popup for confirmation.

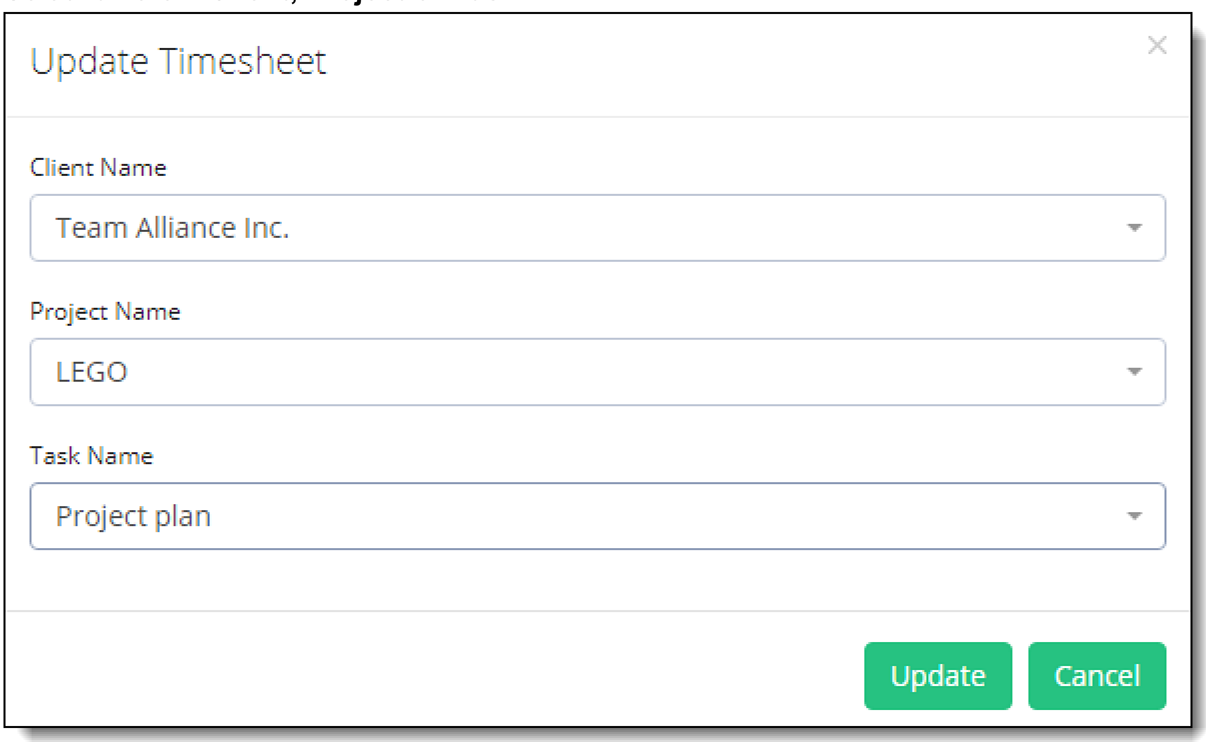


Changing the Project/task of the saved time entry record:

1. Navigate to **Time Entry** row which you want to change.
2. Click on the **Edit** Icon next to Delete button.



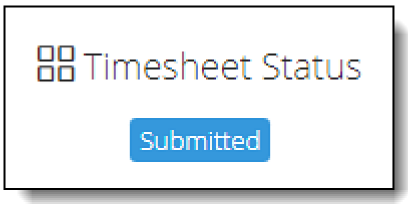
3. Select different **Client, Project or Task**.



   a. Click on **Update** button to save this change.

Submitting time entry record:

1. Click on **Submit** button to submit time entries for approvals. After submit, all time entries will be locked for users.


2. After submit, time entry status show **Submitted** and also reflects on Approval Status at the bottom of the Period View page.


Copy Timesheet, Copy Activities, Export Offline Timesheet, Import Offline Timesheet:

In Timesheet Period View, below the Save and Submit button, there are very useful options through which a lot of time can be saved.

- **Copy Timesheet**: While on any Timesheet period, the user can copy any existing Timesheet period by using Copy from Date. Through this option, all entries including, Projects, Tasks, hours, their descriptions will be copied from given period date to current period.
- **Copy Activities**: This option will copy only Project, Tasks to current Timesheet period. Hours will not be copied.
- **Export Offline Timesheet**: This option will save a CSV file of the current Timesheet period on your computer.
- **Import Offline Timesheet**: This option will upload saved CSV Timesheet file from the computer. For details, see **TimeLive FAQ**

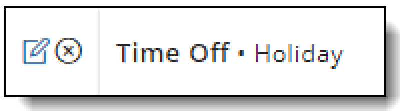
4.4. Time Off In Timesheet

There are two types of leaves we interact with any organization, one is planned and other is unplanned. For example, Annual Leaves, Marriage leaves and similar are pre-planned leaves but sick, casuals and some others are not planned.

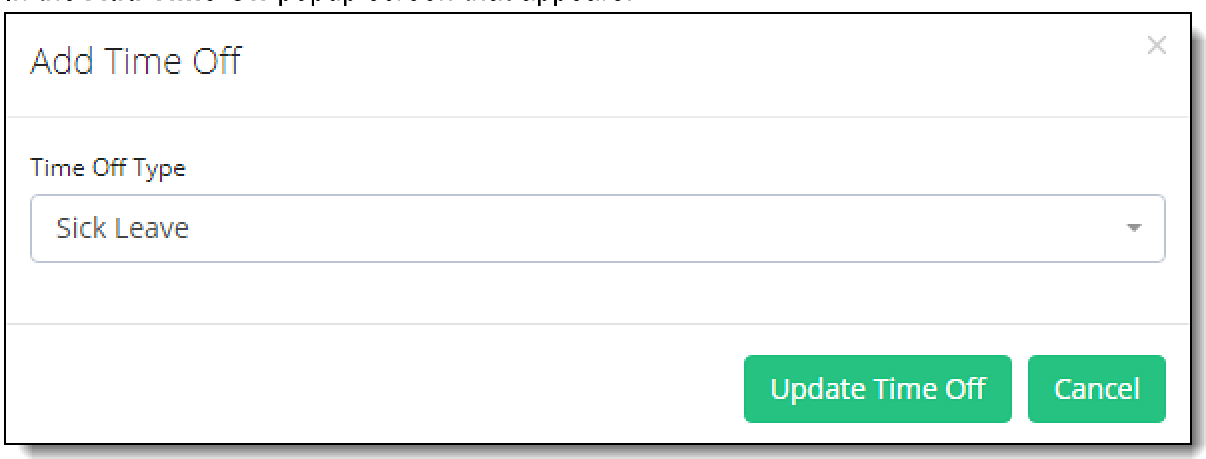
In TimeLive, the user can submit the request for pre-planned leaves and unplanned leaves through **Time Off Request** but that could only be done with the user login. On the other hand, through My Timesheet, same user or his/her manager, can fill and submit same leaves.

Filling Time Off through My Timesheet:

1. Click on **Edit** icon next to **Time Off** caption below the Time entry selection.

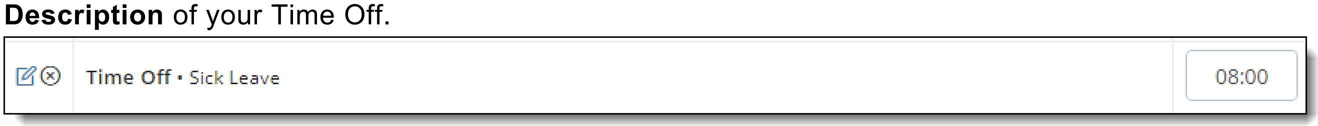


2. In the **Add Time Off** popup screen that appears:



   a. Select **Time Off Type** from drop-down list.
   b. Click green button **Update Time Off**.

3. Fill the hours(if Start Time/End time option is enabled, fill the end time as well) and fill the **Description** of your Time Off.



4. Click **Save** button after you are done.

5. Click **Submit** button for saving the timesheet and forward it for the approval process.

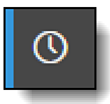
4.5. Submitting a Timesheet

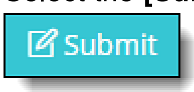
When a period ends, users submit their timesheets for approval. The approval process ensures that time is entered correctly for more efficient and accurate analysis.

TimeLive automatically submits users' timesheets to all necessary supervisors, project leaders, and clients based on both the system configuration set up by the administrator and the approval type defined at the project level.

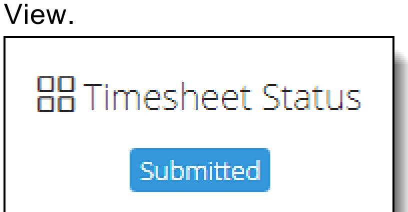
Submit button is only available in Period-View of time entry.

Submit your timesheet for approval:

1. Click on **[My Timesheet]** link on the left-hand side menu.


2. Select the **[Submit]** button.


3. After submitting, time entry records will become locked for users. **(Administrator can disable time entry locking after submitting by un-checking on preferences option "Lock Submitted Records")**.

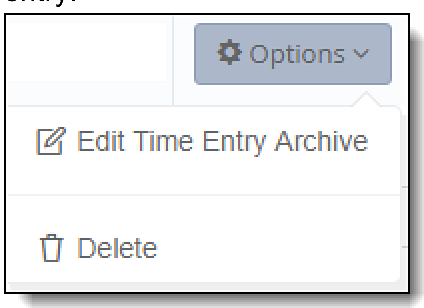
4. After submit, the timesheet will show submitted status and approval status at the bottom of Period View.


Un-submitting and Reopening Timesheets:

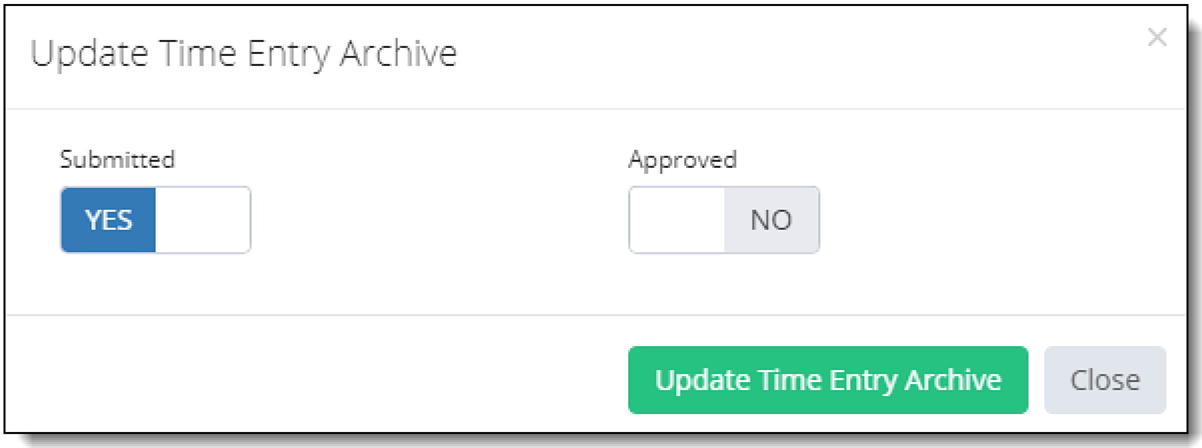
TimeLive Administrator can un-submit or un-approve user's time entries using **[Time Entry Archive]** option available under **[Admin Options]**. Unsubmitting/Unapproving a single record will open the whole period of time entry of that particular user. A user can update his time entry by login-in using his own username and password and can update any entry of the whole period again. He can then re-submit this whole period for approval.

1. Using **[Time Entry Archive]**, an administrator can search records which he wants to change using different filter options available.

2. Click on **[Edit]** on record on which administrator wants to change approval or submitted flag of a time entry.



3. In the **Update Time Entry Archive** popup screen that appears:



   a. Uncheck **[Approved]** or **[Submitted]** flag as per your requirements.
   b. Click on **[Update]** link for that time entry archive record to update this record. This will remove that same approved or submitted flag for the whole period.

Rejected Timesheets:

If a supervisor, project leader, or client rejects your timesheet, TimeLive will notify you by displaying a rejection notification in your timesheet. You can then review the approval details, correct the errors, and resubmit your timesheet for approval.

4.6. Timesheet Approval

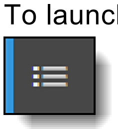
The approval process ensures that time is entered correctly for more efficient and accurate project and Time Off analysis. TimeLive provides a default approval process that you may customize for your organization.

More about Timesheet Approval:

1. If you choose to customize this process, you should change the approval settings before creating projects and allowing users to enter timesheets/expenses.
2. **Approver types** identify the individuals who will be approving users' timesheets and expenses. You can set up many different approver types, each of which will have a specific approval function within the approval path.
3. **Simultaneous / Alternate Approvals**. Multiple approvers can receive and approve the same item concurrently. For example, if you have defined two approval types named Supervisor and Administrator, you can assign them both 1st position in the approval sequence, and they will both receive approval items simultaneously.
4. Users complete their timesheets by entering time against projects or time off codes. Users can enter time only for those projects for which they are part of the project team. All time is either billable or non-billable. Once completed, users submit their timesheets for approval.
5. Once submitted users' timesheets are sent to the appropriate project leaders, clients, and supervisors for approval. Approvers can approve or reject the timesheets. Approved timesheets are closed so the data can be processed.
6. TimeLive sends e-mail notifications (if enabled) to inform approvers when timesheets/expenses are waiting for approval, and to inform users when timesheets and expenses have been rejected/ Approved.

Launching Timesheet Approval:

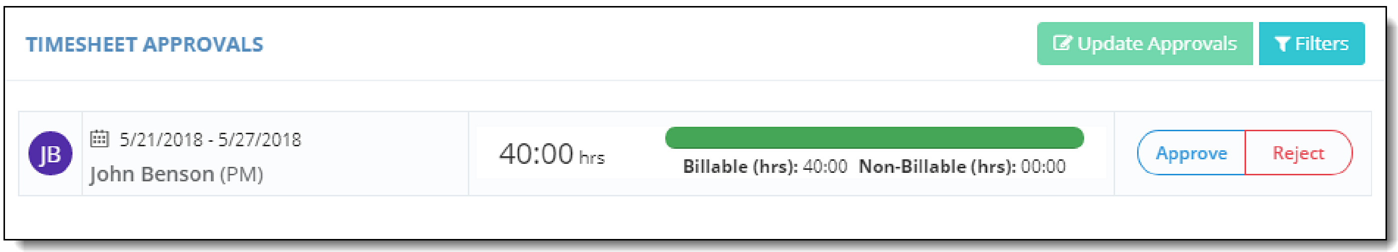
1. To launch timesheet approval, click on **My Approvals** at the left-hand navigation menu.

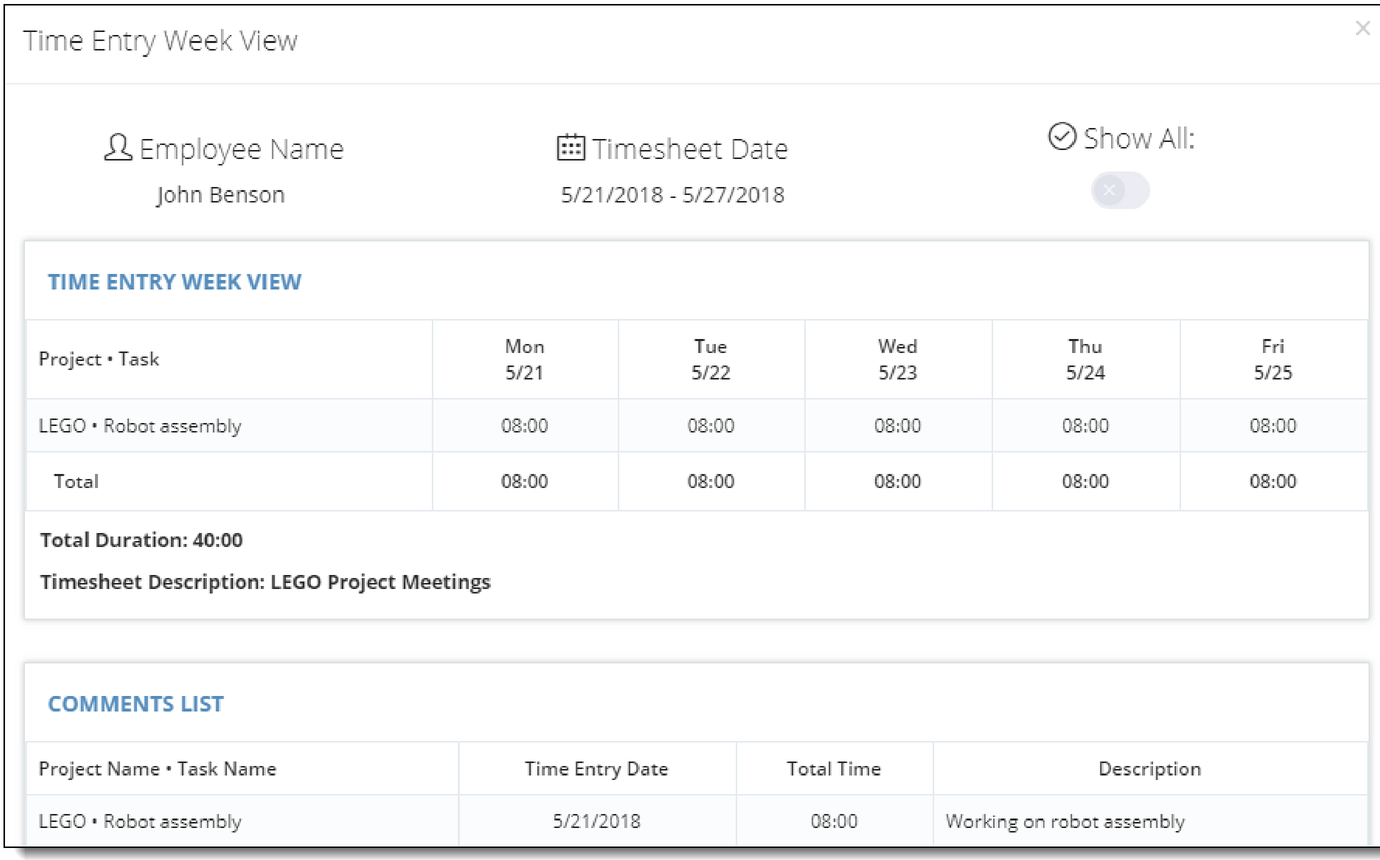


2. Click on **Timesheet Approval**

To review a time sheet record:

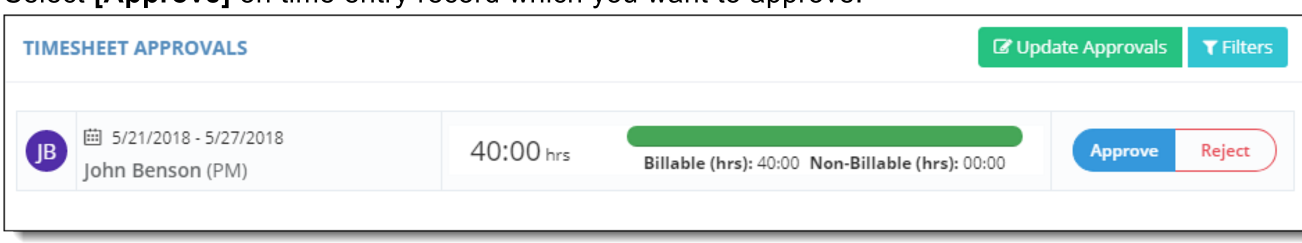
An approver can see whole period timesheet by clicking on **Employee Name** link. This will launch a read-only view of employee time-sheet period.





To Approve a time sheet record:

1. Select **[Approve]** on time entry record which you want to approve.

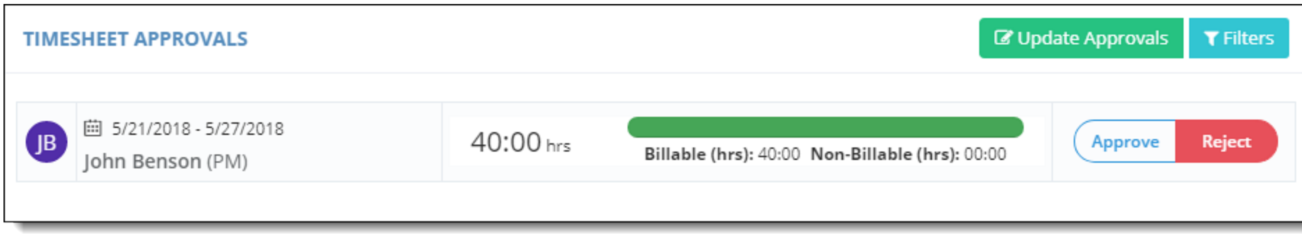


2. Enter your comments in “Approver’s Comments”.

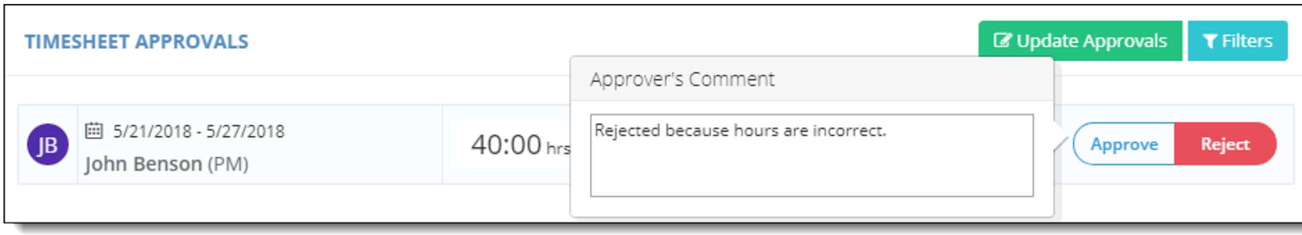
3. Click on **[Update Approvals]** button to confirm these approvals.

To Reject a time sheet record:

1. Select **[Reject]** on time entry record which you want to reject.



2. Enter your comments in “Approver’s Comments”.



3. Click on **[Update Approvals]** button to confirm these changes.

4.7. Approval Types (Workflow)

Approval types identify the individuals who will be approving users' timesheets and expenses. You can setup many different approver types, each of which will have a specific approval function within the approval path.

Workflows establish which manager(s) must approve an employee's time or expense submissions before those entries can be processed for project costs, billing, payroll, reports, etc. When multiple approvals are required for the same submission (e.g., first the Project Manager and then the HR Manager), the workflow also controls which manager approves first, second, and so on.

A workflow is first created in the **[Approval Types]** view and then assigned to individual projects. When the employee submits a time or expense entry, the designated manager then approves the time and expenses using the Time & Expense approval view. When signing on, an approving manager is notified when submitted entries are awaiting approval.

System administrator can setup **[Approval Types]** name using **[Admin Options] > [Approvals]**.

Approval Paths:

Approval paths define the approval process for the users in your organization. You can include five different types of approver in your approval path definition. You can define approval sequence for each of the approval paths in your approval paths definition.

Approval Types:

There are basically 3 types of Approvals

1. Timesheet Approvals
2. Expense Approvals
3. Time Off Approvals
   If you check **Is TimeOff Approval** while creating/editing the Approval type, it will be for Time Off Approvals

Approver types:

There are five types of approver in TimeLive which you can define in Approval Paths.

1. **Team Lead**
2. **Project Manager**
3. **Specific Employee** (if you want your timesheet/expenses to be approved by some specific employee)
4. **Specific External User** (if you want your timesheet/expenses to be approved by some specific external user (client)
5. **Employee Manager** (Employee own manager defined in employee form).

Simultaneous / Alternate Approvals

Multiple approvers can receive and approve the same item concurrently. For example, if you have defined two approval types named Supervisor and Administrator, you can assign them both 1st position in the approval sequence, and they will both receive approval items simultaneously.

4.8. Attendance Entry

In TimeLive, Administrator can cater company requirement of employee attendance through many ways.

Attendance through Employee Profile

In some companies, Users only require punch-in Time and punch-out Time for their presence verification. TimeLive allows user to **Time In/Time Out** in the most simple and basic way as soon as user login to his/her account.

To Achieve this, a user first needs to enable this feature from his/her profile.

1. **Login** to your **TimeLive Account**.
2. Click on your **Profile**.
3. Under Others Tab, Switch On **Enable Time In/Out Option** and click **Update**.
4. **Time In** and **Time Out** buttons will now show besides the User Profile.
5. **Click** on **Time In** button to input **Login Time** and Time Out button to input **Logout Time**. TimeLive will automatically record System Time.

Attendance through My Timesheet > Day View.

Attendance through a **punch in / punch out** of employees in order to track actual physical availability of work. Employee attendance and time entry are two different options to track two different things.

Attendance list and its form is available at **[My Timesheet > Day View]** which can be launch using **[My Timesheet]**

- A time entry record is an employee activity on a project for a particular duration. This is useful for calculating employee actual employee time spent on project task.
- **Time In** represents a time when an employee enters in an organization and **Time Out** represents when employee finishes his daily work.
- A case can be
    - 8:00 AM (Employee TimeIn through employee attendance)
        - Work starts on a project task from 8:30 AM to 12:30 PM. This is a case of timesheet record of recording employee activities on the project.
        - At 12:30 PM, Employee takes a lunch break for an hour.
        - Then start working again from 1:30 PM.

        - Perform a task from 1:30 PM to 4 PM (will be recorded in timesheet)
    - 4:00 PM (Out) (Employee leave with an attendance record of “Out : 4:00 PM)
    - This is a very typical case of attendance through which we can record one time in the record when an employee arrives and one timeout record when employee leaves.
- Attendance module of TimeLive also allow employees to enter their Paid Time Off (PTO) leave and Holiday.

How to add new attendance record:

1. Click on [My Timesheet] to launch time entry Day View.
2. Click on the green icon [+ Add Attendance]
3. Select the first drop-down In from In/Out Option
4. Then fill your Time-In
5. Then Select Present or any Status you want to fill against this time.
6. Then Click Add Attendance green button.

How to enter employee leave record:

1. Click on **[My Timesheet]** to launch time entry Day View.
2. Click on the green icon [+ Add Attendance]
3. Select the first drop-down In from In/Out Option
4. Then fill your Time-In
5. Then Select Time Off Type you want to fill against this time.
6. Then Click Add Attendance green button.

How to modify attendance record:

1. In employee attendance list, click on Gear Icon then Click **[Edit]** on record which you want to modify.
2. Update this record with your required modification in **[Update Attendance]** Popup.
3. Click on **[Update Attendance]** button to update this record.

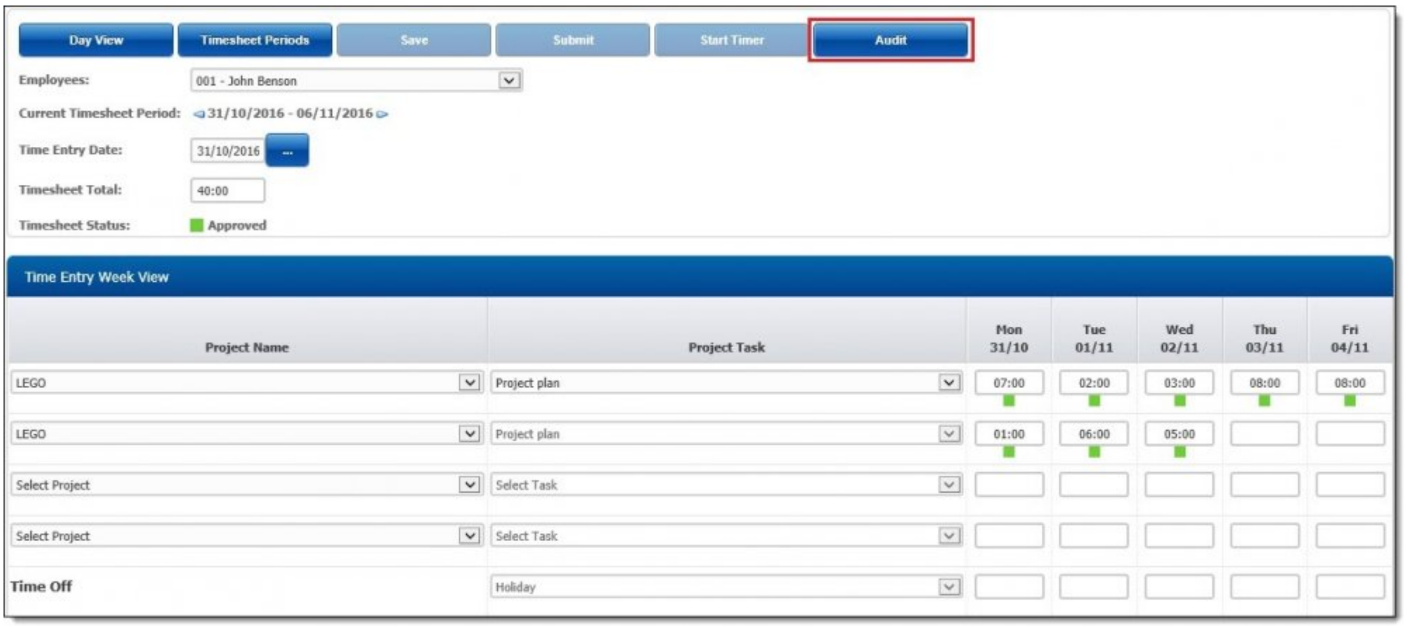
How to delete attendance record:

1. In employee attendance list, click on Gear Icon then Click **[Delete]** on record which you want to modify.
2. Click on **[Ok]** to confirm this delete on confirmation dialog box.

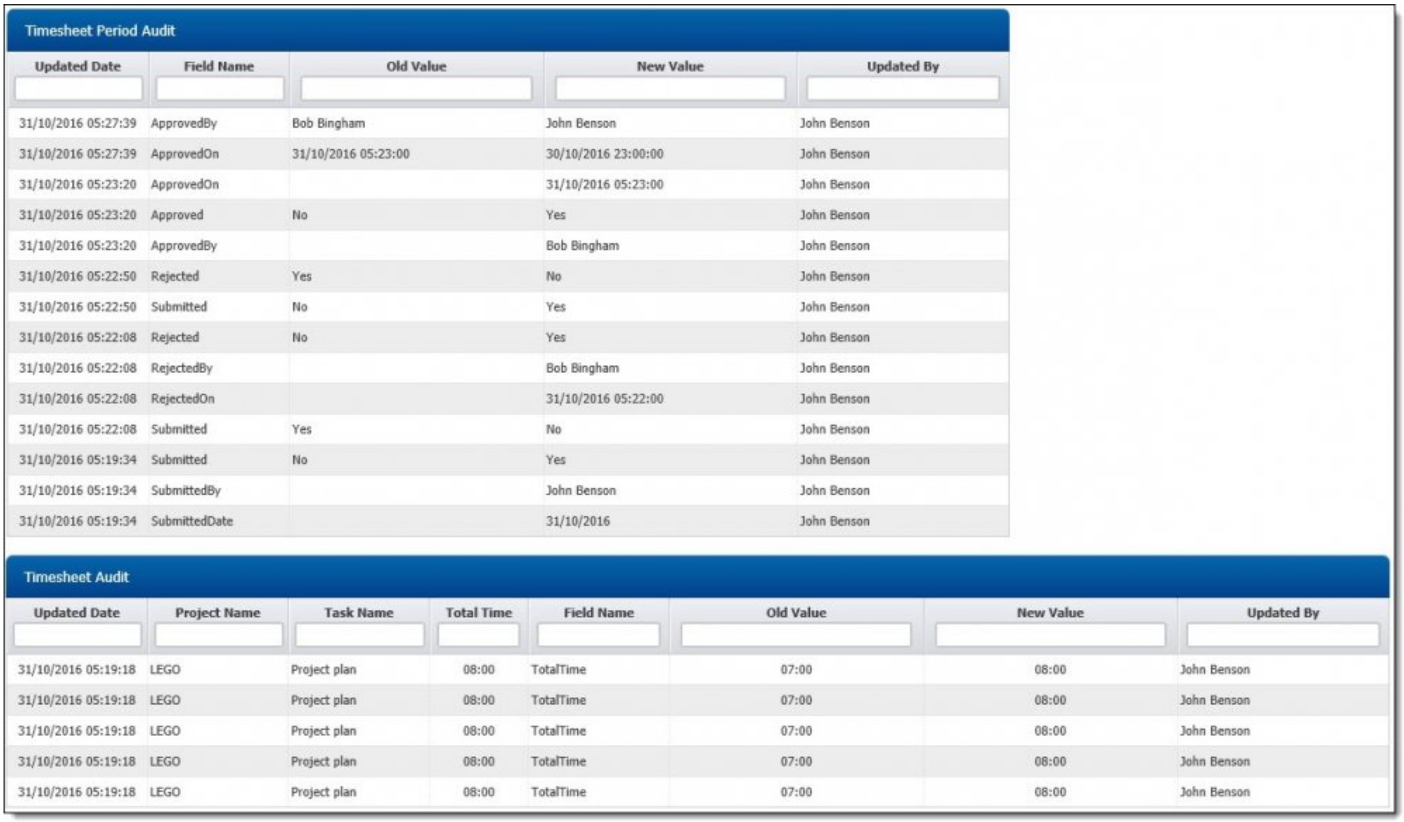
4.9. Timesheet Audit

Timesheet Audit in My Timsheet:

In My Timesheet User/Administrator (depends on Role Permission) can view Period Timesheet Audit.



Timesheet Audit button is visible for Users (depends on Role Permission) to view the period Activity. This Audit will show any changes done by user and approval process in detail.



Timesheet Period Audit

Updated Date	Field Name	Old Value	New Value	Updated By
31/10/2016 05:27:39	ApprovedBy	Bob Bingham	John Benson	John Benson
31/10/2016 05:27:39	ApprovedOn	31/10/2016 05:23:00	30/10/2016 23:00:00	John Benson
31/10/2016 05:23:20	ApprovedOn		31/10/2016 05:23:00	John Benson
31/10/2016 05:23:20	Approved	No	Yes	John Benson
31/10/2016 05:23:20	ApprovedBy		Bob Bingham	John Benson
31/10/2016 05:22:50	Rejected	Yes	No	John Benson
31/10/2016 05:22:50	Submitted	No	Yes	John Benson
31/10/2016 05:22:08	Rejected	No	Yes	John Benson
31/10/2016 05:22:08	RejectedBy		Bob Bingham	John Benson
31/10/2016 05:22:08	RejectedOn		31/10/2016 05:22:00	John Benson
31/10/2016 05:22:08	Submitted	Yes	No	John Benson
31/10/2016 05:19:34	Submitted	No	Yes	John Benson
31/10/2016 05:19:34	SubmittedBy		John Benson	John Benson
31/10/2016 05:19:34	SubmittedDate		31/10/2016	John Benson

Timesheet Audit

Updated Date	Project Name	Task Name	Total Time	Field Name	Old Value	New Value	Updated By
31/10/2016 05:19:18	LEGO	Project plan	08:00	TotalTime	07:00	08:00	John Benson
31/10/2016 05:19:18	LEGO	Project plan	08:00	TotalTime	07:00	08:00	John Benson
31/10/2016 05:19:18	LEGO	Project plan	08:00	TotalTime	07:00	08:00	John Benson
31/10/2016 05:19:18	LEGO	Project plan	08:00	TotalTime	07:00	08:00	John Benson
31/10/2016 05:19:18	LEGO	Project plan	08:00	TotalTime	07:00	08:00	John Benson

In the above image and under **1st Audit Grid**, date and time of Submission, Rejection and approval Date and Time along with Approver Name and submitter Name are shown along with old and new status of the Timesheet Period.

In the **2nd Audit Grid** it shows that there are 5 entries in total in this particular period along with their old and new values when submitted by Submitter Name.

5. Expenses

5.1. Expense Entry

An expense is a cost incurred from work-related tasks and/or purchases (travel, meals, office supplies, etc.). Expenses may be non-billable or billable, the latter applying to expenses for a project funded by a specific client.

Users complete and submit their expense sheets to team lead, project manager, and/or clients for approval. The expense sheets are then reviewed and either approved or rejected. When the expense sheet is approved, the expense information is saved to the database.

Once an expense sheet is submitted, the data can be analyzed by project leaders and administrator to determine project costs and client billing.

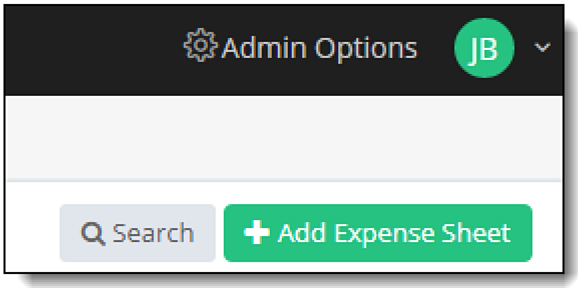
More about Expense sheet:

1. A single expense sheet allows a user to submit multiple expenses.
2. A user fills out and submits an expense sheet, uploading any available receipts.
3. The user can choose which project each expense was associated with.
4. User can print an expense sheet for the signature purpose.
5. The expense sheet is automatically sent to one or more approvers.
6. The approvers approve the expense sheet.
7. The finance department can reimburse the user based on the approved expense sheets.
8. The expenses can be exported for client billing or reviewed against project estimates.

Add new Expense Sheet:

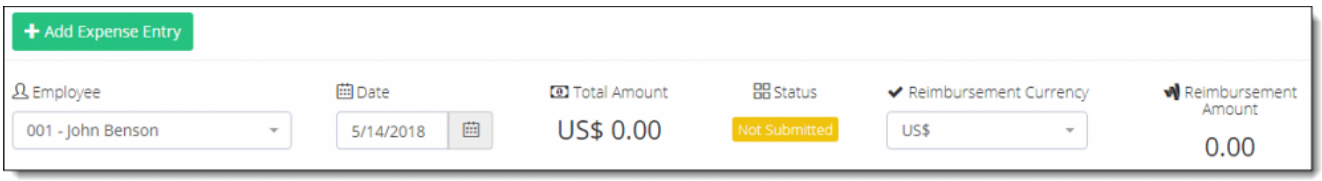
1. Click on the **My Expenses/$** Icon from the left navigation bar.


2. Click **Add Expense Sheet** green button on the top-right of the application and follow below steps for adding Expense Entry.



Add new Expense Entry :

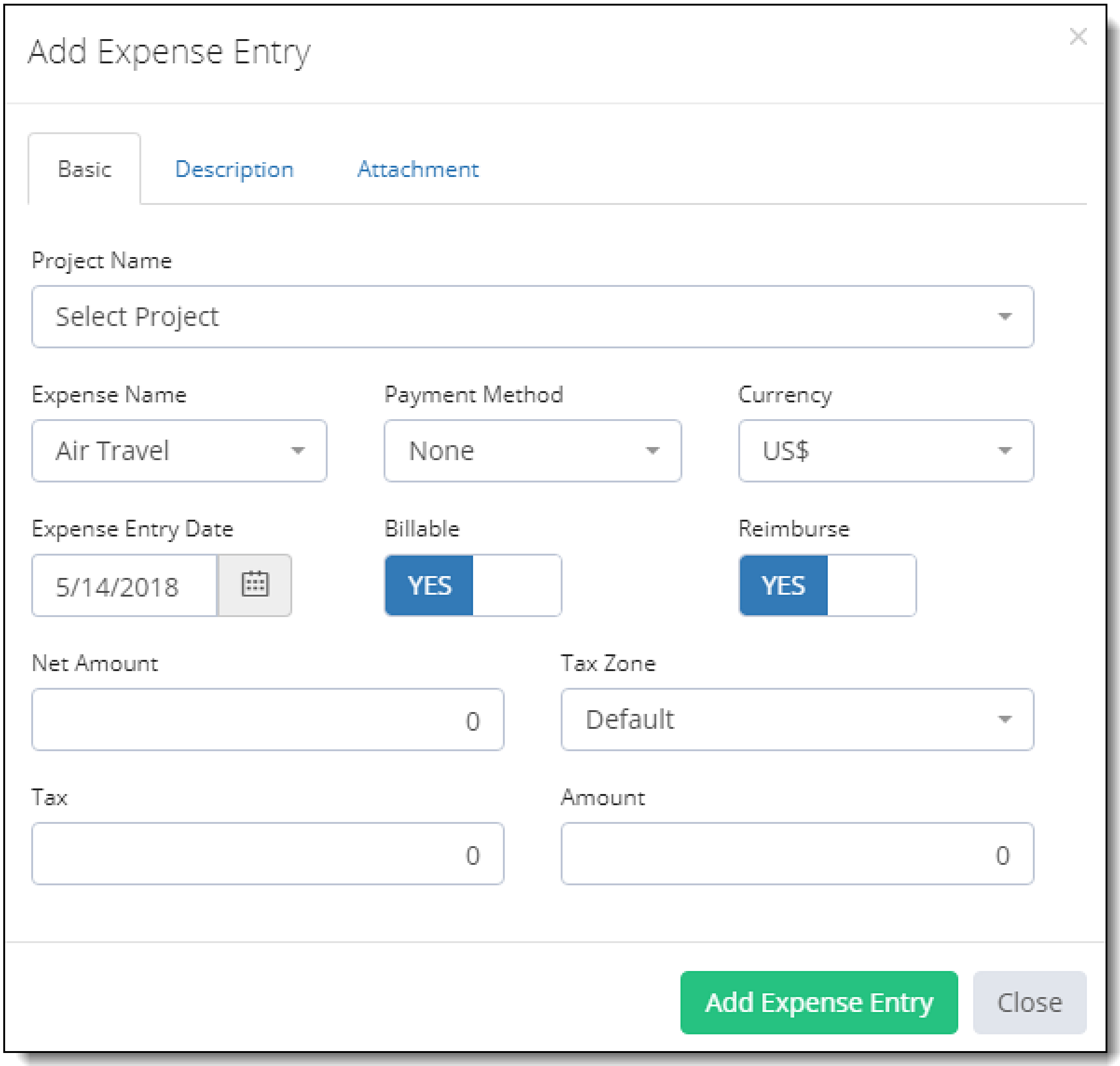
3. In the **Add Expense Entry** screen that appears, where you can:



    a. Click **Add Expense Entry** green button on the top-left of the application.

4. In the **Add Expense Entry** screen that appears:

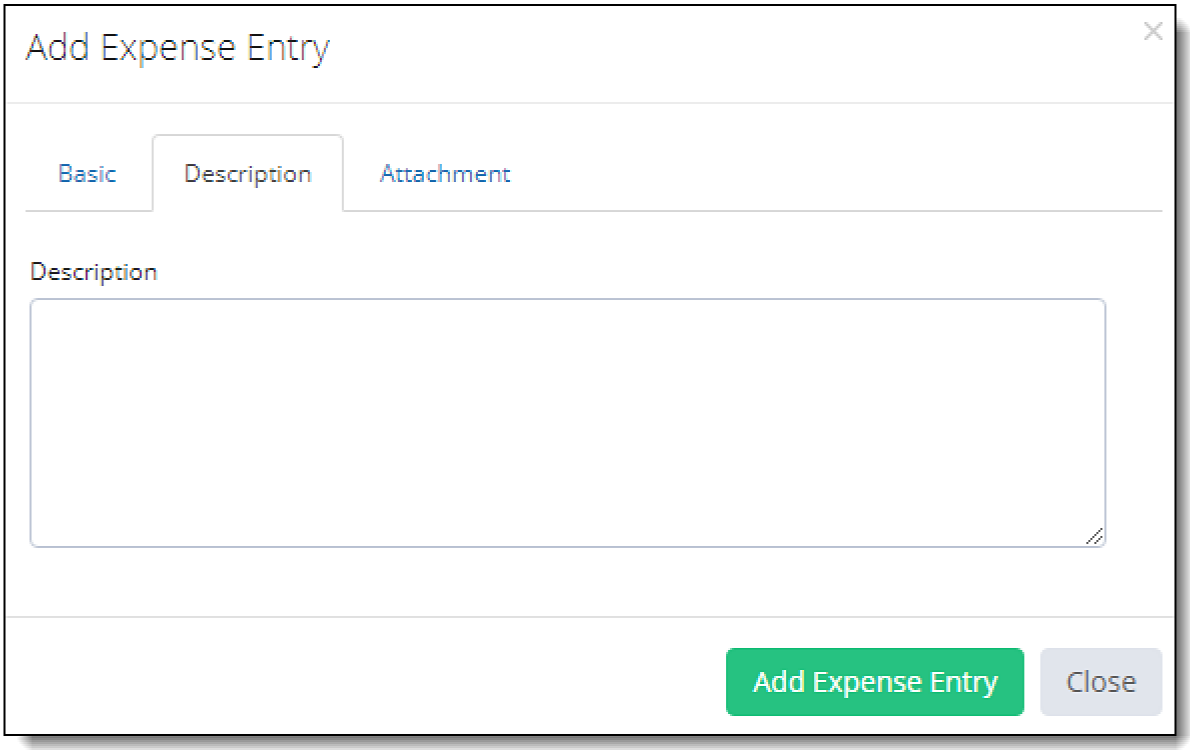
5. In **Basic** Tab.



a. Select the **Project Name**.
b. **Expense Name**.
c. **Payment Method** (different available payment methods like Cash, Credit Card etc.).
d. **Currency**.
e. **Expense Entry Date**.
f. **Billable** (Yes/No).
g. **Reimburse** (Yes/No) (if you paid for the expense and require reimbursement).
h. **Net Amount**.
i. **Quantity** (depends on Expense Type) (if this expense is configured to get input of quantity field).
j. **Rate** (depends on Expense Type) (if this expense is configured to get input of quantity field).

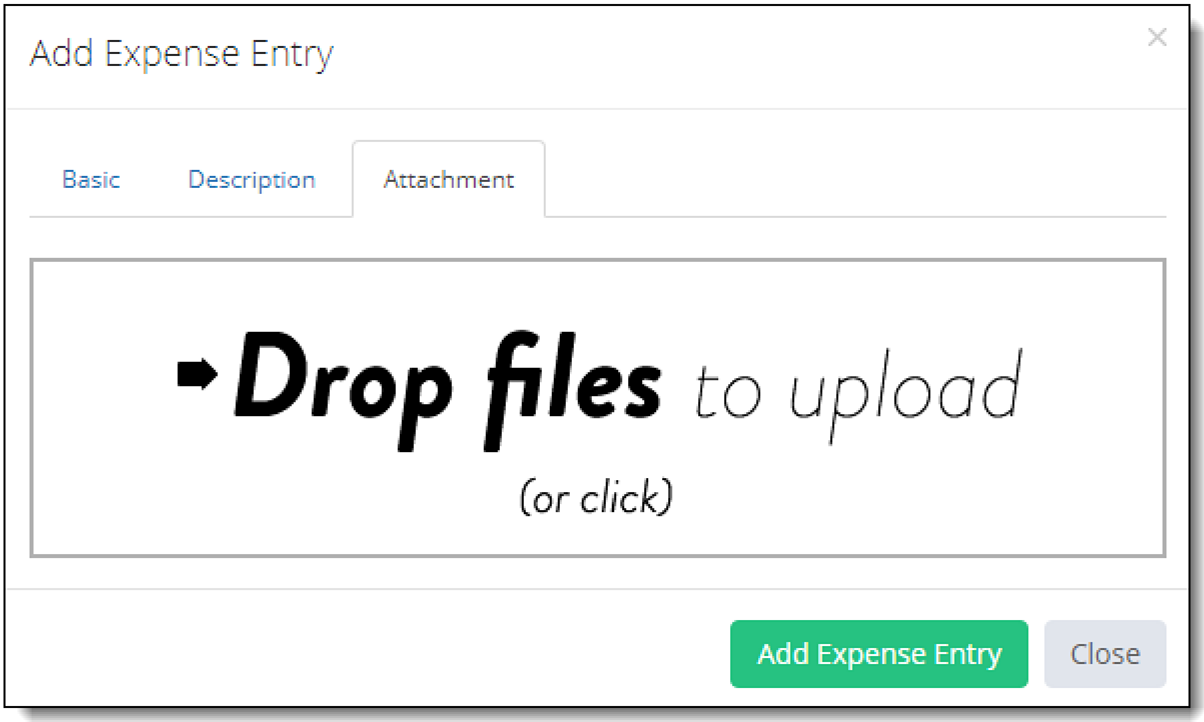
k. **Tax Zone** (depends on Expense Type) (will be appear only if this expense is configured as a tax able expense).
l. **Tax Amount** (depends on Expense Type) (will be appear only if this expense is configured as a tax able expense).
m. **Amount**.

6. In **Description** Tab.



a. Fill the **Description** as per the requirement.

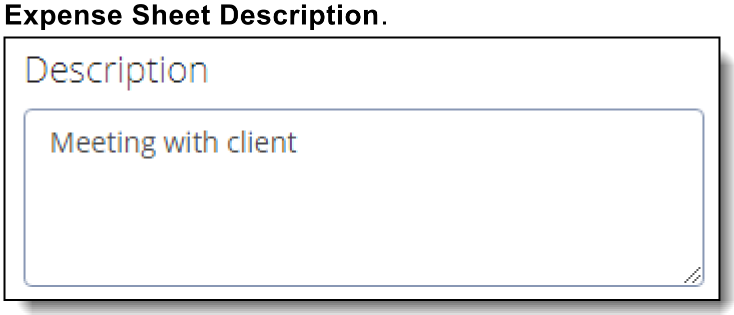
7. In **Attachment** Tab.



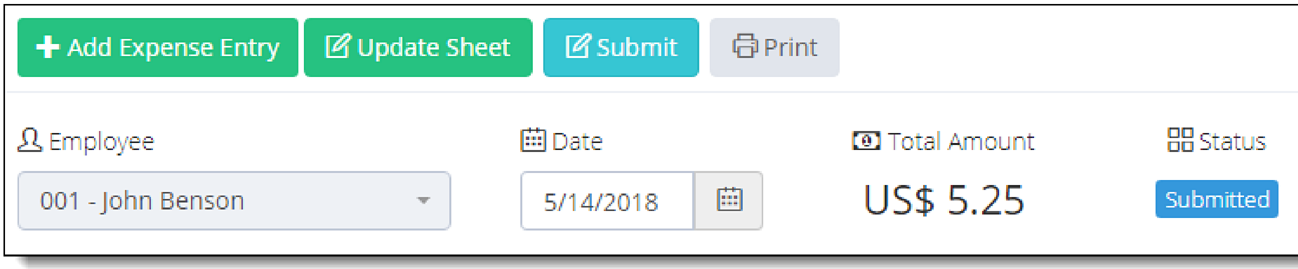
   a. Click on the box file to upload the related file.
   b. Select the file and click **Open**.
   c. Click **Add Expense Entry** button.

If you want to add another expense entry, please follow above process.

8. After expense entry, the user should navigate to the bottom of the Expense sheet page and fill the **Expense Sheet Description**.

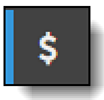


9. Click **Submit** blue button on top for the approval process.

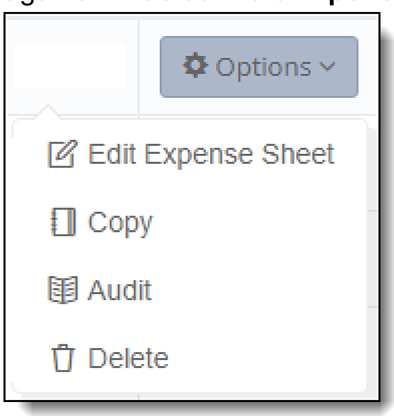


Editing Expense Entry:

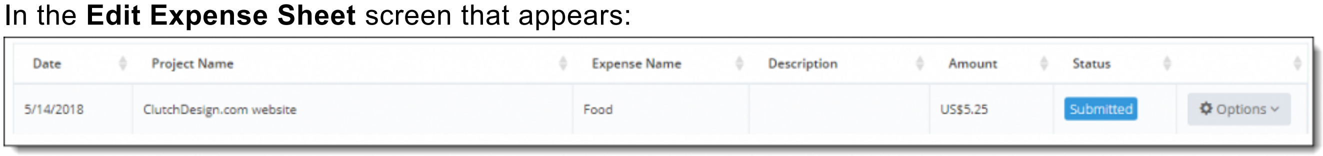
1. Click on the **My Expenses/$** Icon from the left navigation bar.



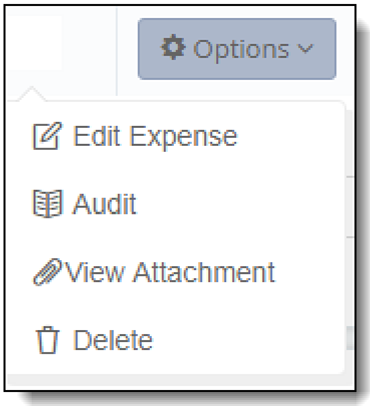
2. Scroll/search for the Expense Sheet which you want to Edit and click on the **Options** gear icon against it. Select **Edit Expense Sheet** option.



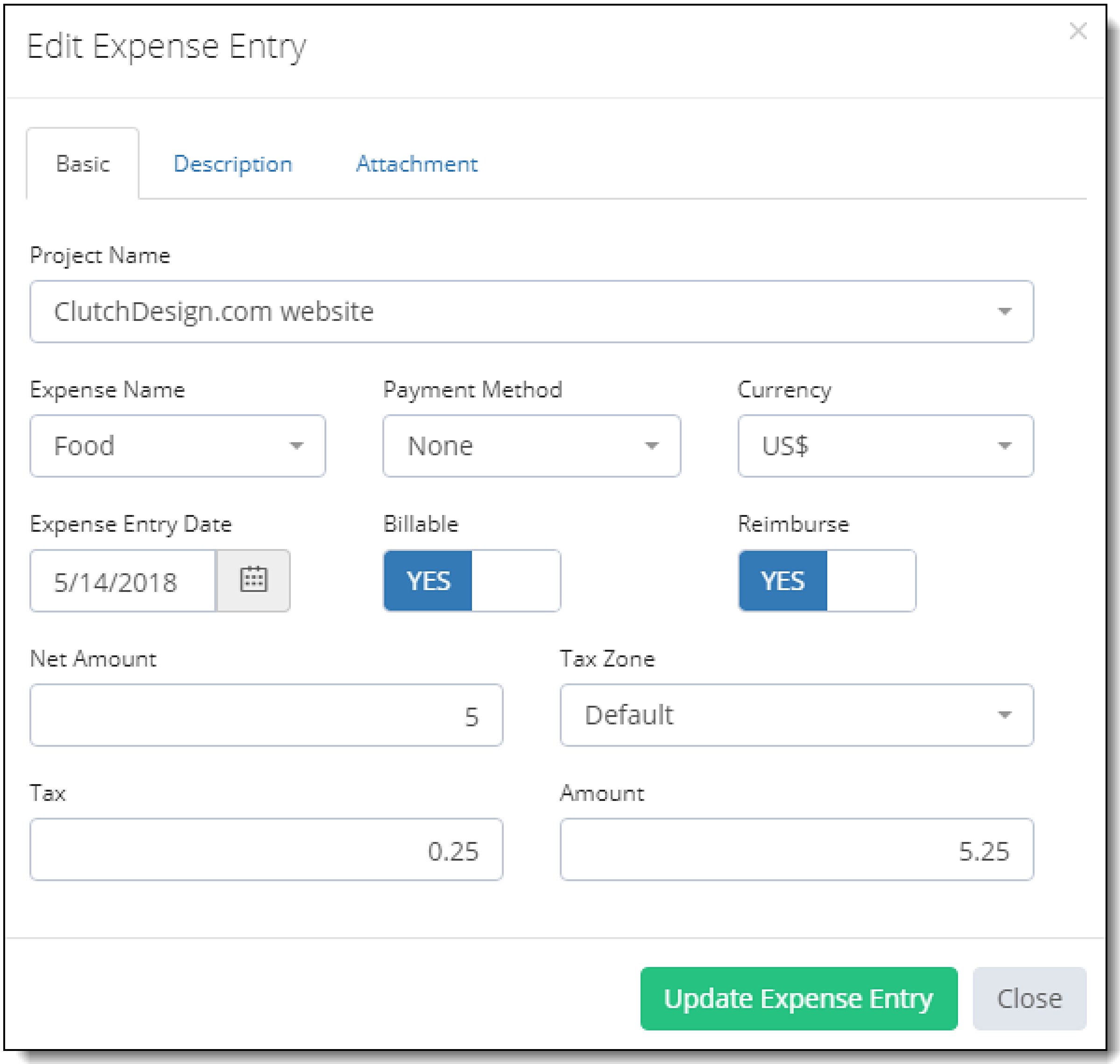
3. In the **Edit Expense Sheet** screen that appears:



4. If you want to modify the expense entry, click the **Options** gear icon against the expense entry. Select **Edit Expense** option.

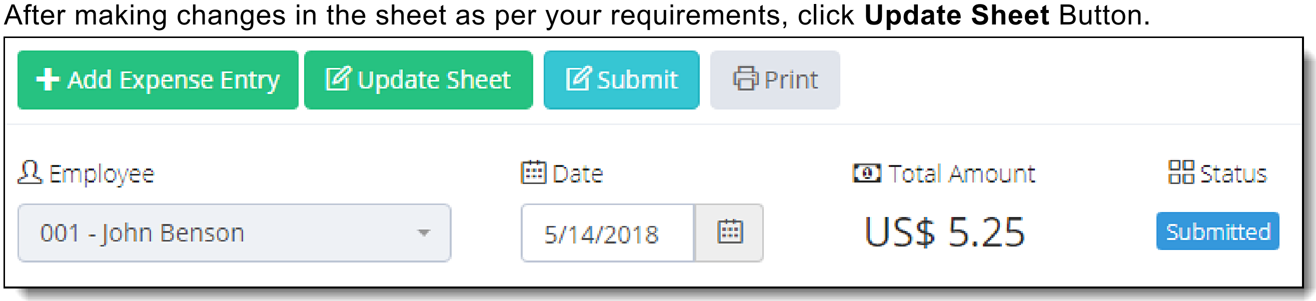


5. In the **Edit Expense Sheet** screen that appears:



   a. Make changes as per your requirements and click **Update Expense Entry**.

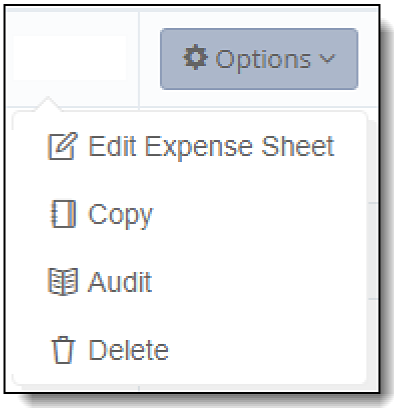
6. After making changes in the sheet as per your requirements, click **Update Sheet** Button.



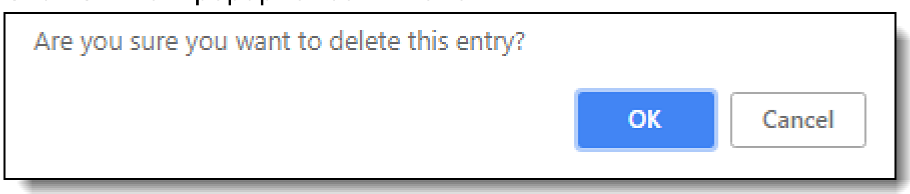
Deleting Expense Entry:

1. Click on the **My Expenses/$** Icon from the left navigation bar.

2. Scroll/search for the Expense Sheet which you want to delete and click on the **Options** gear icon against it.



   a. Select **Delete** option.
   b. Click **OK** from popup for confirmation.



5.2. Expense Approval

Launching Expense Approval:

To launch expense approval, click on **Project Management —> Expense Approval** option in left side navigation menu.

Approving expense records:

Expense Approval page shows four possible expense approval tables depend upon logged in employee role in different projects and expense approval path defined for those projects. These are the possible types of expense approval tables available in expense approval page.

1. **My Team Expense Entry Approvals** (Will be showed if a project's team lead expense entries are due to be approved)
2. **My Projects Expense Entry Approvals** (Will be showed if a project's project manager expense entries are due to be approved)
3. **Client Expense Entry Approvals** (Will be showed if a project's external client expense entries are due to be approved)
4. **Administrator Expense Entry Approvals** (Will be showed if system administrator expense entries are due to be approved)

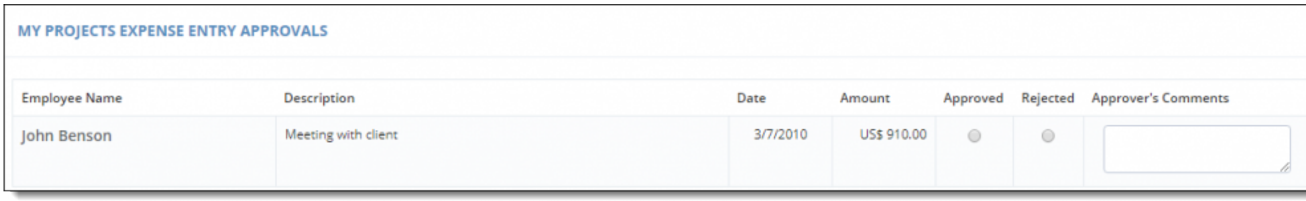
Access Expense Approval Page:

1. Click on the **My Approvals** Icon on the left Navigation Bar.

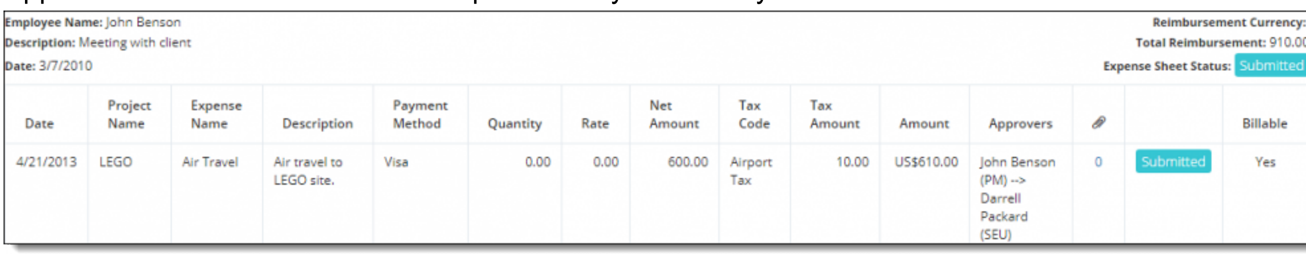
2. Click on the **Expense Approval** Icon.


To review expense sheet:

1. To view whole expense sheet before approval, an approver can click **Employee name** to open read-only view of expense sheet submitted by a user.

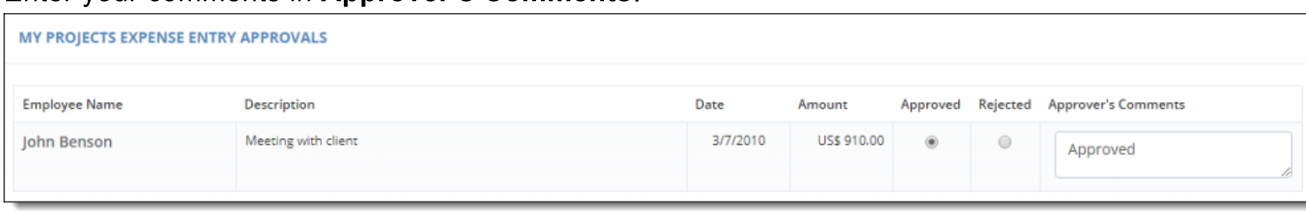


2. Approver can see attachments of expense entry records by click on attachment links.



To Approve a expense sheet records:

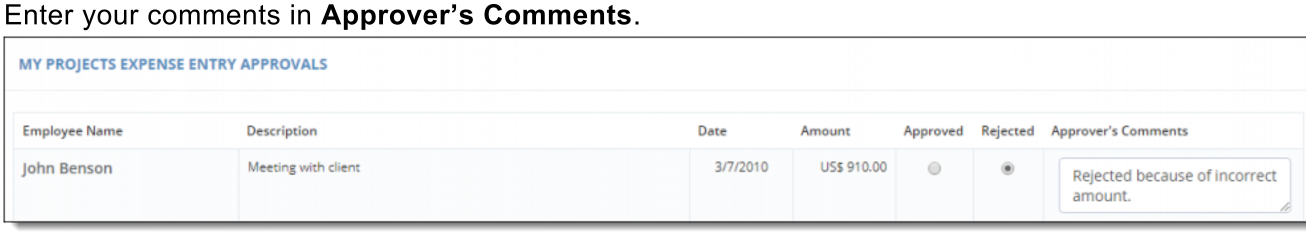
1. Select **Approved** radio box against the expense sheet record which you want to approve.
2. Enter your comments in **Approver's Comments**.



3. Click on **Update Expense Entry Approvals** button on the top of the application to confirm these approvals.

To Reject a expense sheet record:

1. Select **Rejected** radio box against the expense sheet record which you want to reject.
2. Enter your comments in **Approver's Comments**.



3. Click on **Update Expense Entry Approvals** button on the top of the application to confirm these approvals.

6. Time Off

6.1. Time Off Overview

TimeLive’s Time Off Feature provides a comprehensive and unmatched time off and absence management solution in the industry today. Using Time Off feature, employees can check available time off, make a request, get manager approval.

NOTE: Time Off Request will only be available if you have purchased Time Off feature.

Features Of Time Off Module in TimeLive:

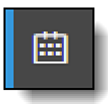
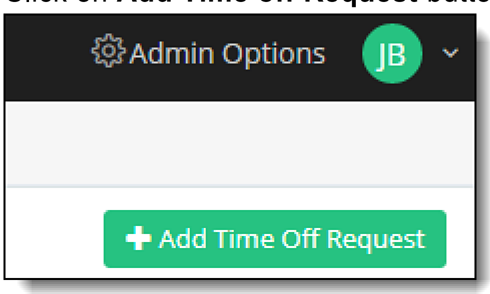
- Time off requests, approvals, and scheduling.
- Users can see how much time off they are entitled and request time off accordingly.
- Automated notifications: requests, approvals, rejections, approver notices.
- Unlimited time off types and balances
- Set up accrual rules that are automatically applied.
- Time off accruals engine.
- Multiple holiday schedules by country, country groups, or locales.
- Approvers can approve time off based on how much is entitled and how much has been requested.
- Multi-level Approval Paths for time off configurable at the employee level
- You may record time off against any enabled [Time Off] type. Some [Time Off Types] may require [Time Off Request], while others may not, as determined by your administrator.
- If the request is required for **[Time Off Type]**, you must submit **[Time Off Request]** for a time off type using the **[My Time Off]** page.
- Booked time off will automatically display on your timesheet.
- If “Request” is not required for a **[Time Off Type]**, you may enter it directly in your timesheet.
- User can see all **[Time Off Types]** balances while submitting **[Time Off Request]**.
- Administrator can check the Time Off status of each employee by clicking on **[Time Off]** link on ”Employee” page.

6.2. Time Off Request

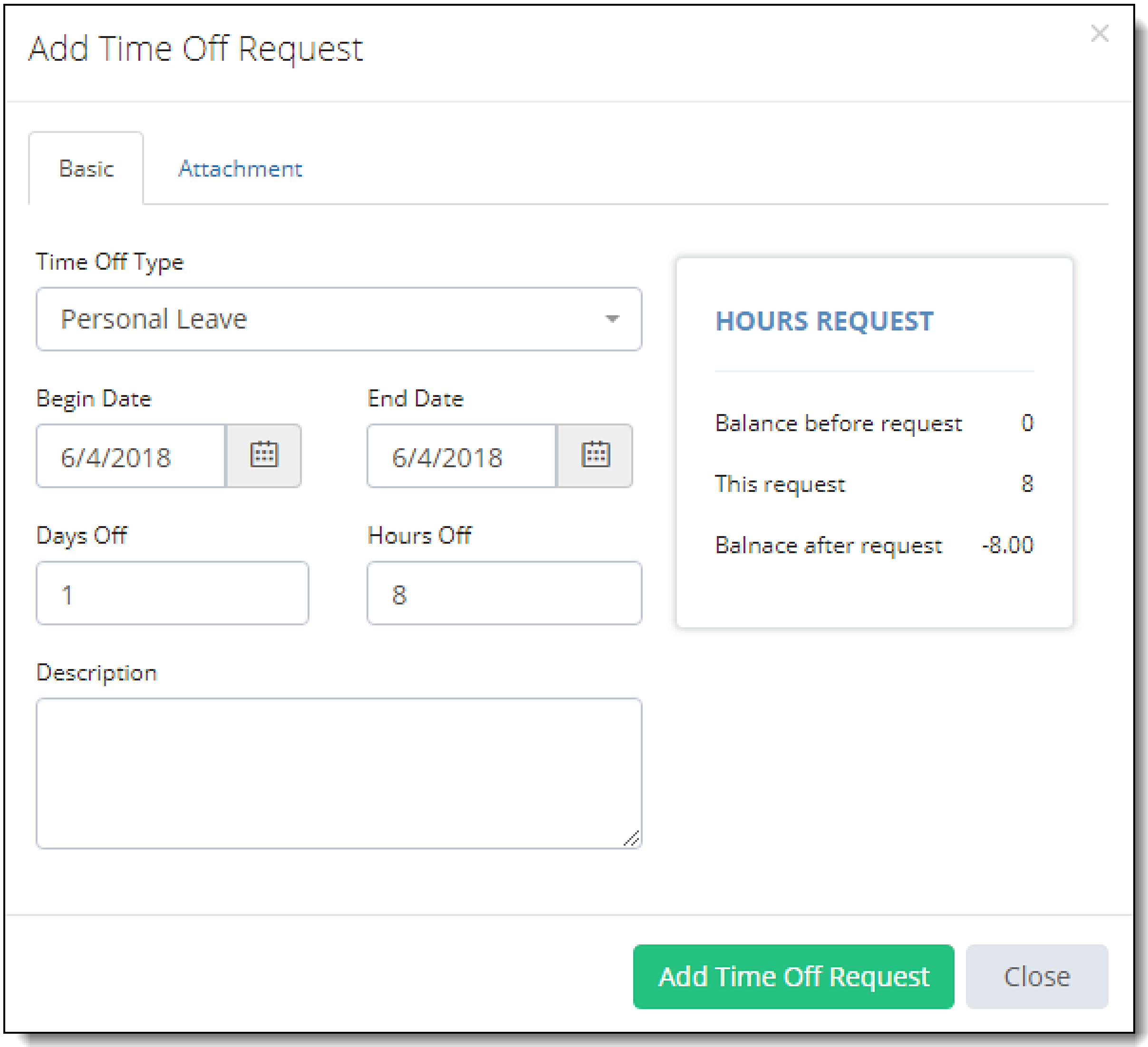
Time Off Request in advance allows your supervisors to schedule around the time when you will be away. Time off request will automatically display in your timesheet under the heading “Time Off” (even before it has been approved) (with its current approval status indicated by color coding).

NOTE: Time Off Request will only be available if you have purchased Time Off feature.

Adding New Time Off Request:

1. Employee can submit **Time Off Request** from **My Time Off** page. Time off request can be made only for those “Time Off Types” in which request is enabled.
2. Administrator can configure Time Off Types* from **Admin Options > Time Off Types**.
3. To add a new request, click on **My Time Off** menu option available on the left-hand side menu.

4. Click on **Add Time off Request** button to add Time off request.


5. In the **Add Time Off Request** screen that appears, enter:



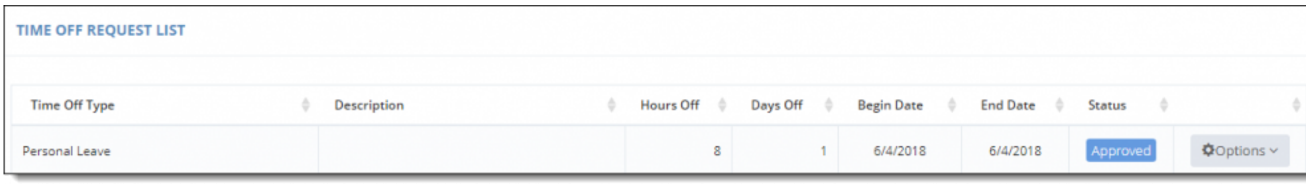
   a. Select **Time Off Type** from drop-down in **Time off Request Information** form.
   b. Select Begin date and End date for Time Off. (After setting the dates, Days off and Hours off will be calculated automatically).
   c. Enter Days off for Time Off. (After enter the Days off, Date range and Hours off will set automatically).
   d. Enter Hours off for Time Off. (After enter Hours off, Date range and Days off will set automatically).
   e. Select project (Optional) for which you want to enter Time Off. (A project selection drop-down can be enabled for Time Off Entry using **Admin Options > Preferences > Time Off** option name **Show Project for Time Off**.)
   f. Click on **Add Time Off Request** button, to save the changes and submit it for approval.

g. Once approved, “Available Balance” of that particular time off type will be decreased with a number of hours requested in **Time Off Request**. In case of no approval required, available balance will be decreased immediately after submission.

Viewing Time Off Request:

You can view:

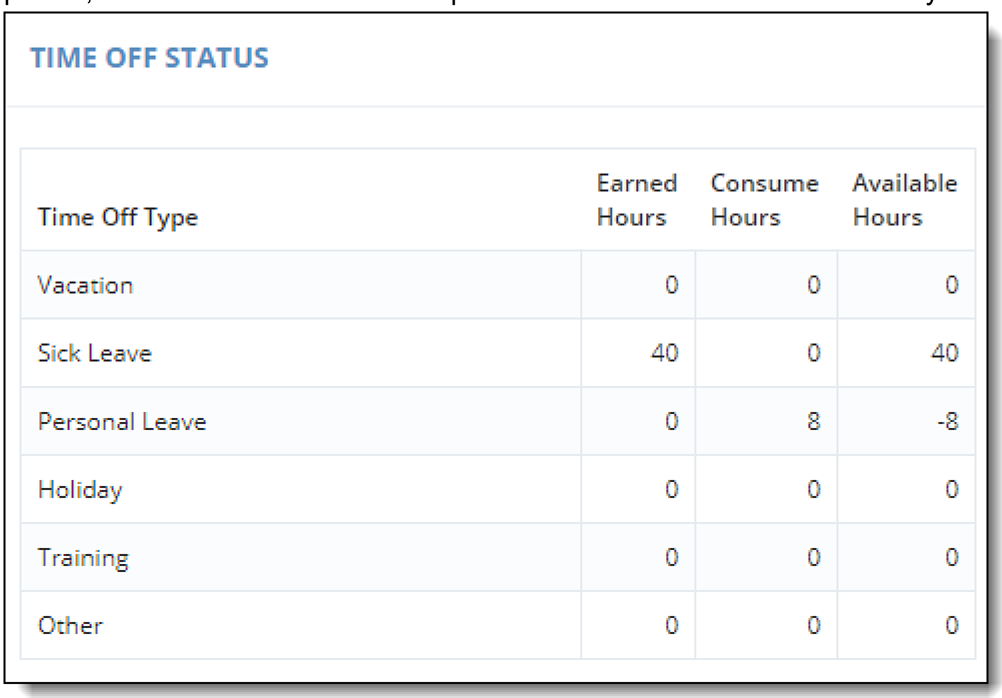
1. All details of your personal time off requests with their approval status, in your **My Time Off** page.



TIME OFF REQUEST LIST

Time Off Type	Description	Hours Off	Days Off	Begin Date	End Date	Status	
Personal Leave		8	1	6/4/2018	6/4/2018	Approved	Options

2. User can find available balances of their assigned **Time Off Types** with their “Earned” in the current period, “Consumed” in the current period and “Available Balance” in *My Time Off



TIME OFF STATUS

Time Off Type	Earned Hours	Consume Hours	Available Hours
Vacation	0	0	0
Sick Leave	40	0	40
Personal Leave	0	8	-8
Holiday	0	0	0
Training	0	0	0
Other	0	0	0

* page.

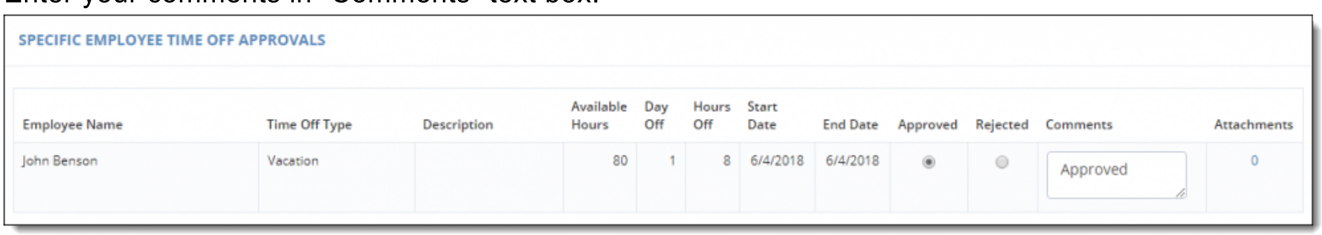
6.3. Time Off Approval

Depending on how your time off types are configured, time off entered against some or all time off types may require “Time Off Request”. To submit time off request, users must complete a time off request form and then submit it for approval. If approval is required, TimeLive sends each submitted **[Time Off Request]** through the sequence of approvers that comprise the user’s time off approval path for review. Each approver can either approve or reject the request.

If a **[Time Off Type]** does not require a request, users enter time against that type directly into their timesheets.

To Approve a Time Off Request:

1. **[Time Off Request]** approval type can be created from **[Admin Options] —> [Approvals]**. Approvals type should have **[Is Time Off Approval]** checkbox checked.
2. Administrator can then set that **[Approval Type]** in employee **[Time Off Approval Type]** dropdown, while editing of employee information.
3. Time Off Request records can be approved or rejected from **[Manage] —> [Time Off Approval]**.
4. To Approve a time off record, select **[Approved]** Radio Box on Time Off record which you want to approve.
5. Enter your comments in ”Comments” text box.


SPECIFIC EMPLOYEE TIME OFF APPROVALS

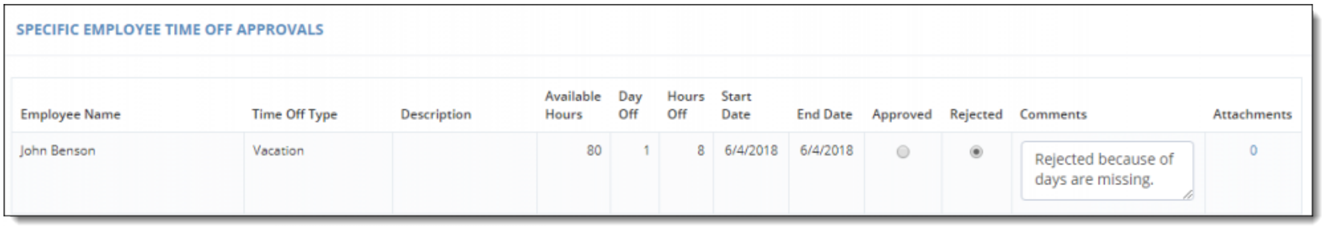
Employee Name	Time Off Type	Description	Available Hours	Day Off	Hours Off	Start Date	End Date	Approved	Rejected	Comments	Attachments
John Benson	Vacation		80	1	8	6/4/2018	6/4/2018			Approved	0

6. Click on **[Update Time Off Approvals]** button to confirm these approvals.

To Reject a Time Off Request:

1. Select **[Rejected]** Radio Box on “Time Off Request Record” record which you want to reject.

2. Enter your comments in ”Comments” text-box.



SPECIFIC EMPLOYEE TIME OFF APPROVALS

Employee Name	Time Off Type	Description	Available Hours	Day Off	Hours Off	Start Date	End Date	Approved	Rejected	Comments	Attachments
John Benson	Vacation		80	1	8	6/4/2018	6/4/2018			Rejected because of days are missing.	0

3. Click on **[Update Time Entry Approvals]** button to confirm these approvals.

7. Billing Rate Setup

7.1. TimeLive billing rate concept

Rate and Billing

There are four different types of **billing rate types** configurable at project level. Billing rate can be define with its applicable date range with complete history. Timesheet then pick billing rate based on timeentry date from billing rate history defined at employee level, project role level or project employee level.

When you set up a project, you will do the following in addition to setting the basic project characteristics.

1. Select type of billing rate pick option in **[Project]** from one of these options
    a. **[Use employee own billing rate]** TimeEntry will pick employee own billing rate defined in **[Employee]** form.
    b. **[Use Project Roles billing rate]** TimeEntry will pick role based billing rate defined in **[Project Roles]** configuration.
    c. **[Use Project employee billing rate]** TimeEntry will pick project based team member billing rate defined in **[Project Team]**
    d. **[Use Task based billing rate]** TimeEntry will pick billing rate defined in individual project tasks.
2. Define the rates to be used for the project
3. Assign a **[Project team]** and associate rates with each team member
4. Specify **[Billable]** during adding of new task in order to separate billing and non-billing time recording for billing purpose.

The Relationship Between Rates, Billing & Timesheet

It is very important that you understand the relationship between project rates, project billing options, and users' timesheets. Because the billing amount for a project is calculated based on the project rates and the time entered against the project by each member of the project team, you want to make sure that you set up the project effectively.

Remember that when you set up a project, you define the rates to be used for the project. You also assign a project team, associate rates with each team member, and specify billable and non-billable for each of the task.

7.2. Employee Own Billing Rate

An administrator or Project Manager (if given the permission) can navigate to billing rate history of employees and make changes in Employee Rates as per the requirement.

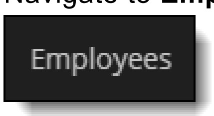
IMPORTANT NOTES:

- Billing history should be configured before entering timesheet record. Otherwise, time entry will not pick billing rate defined after recording time entry.
- In case if Administrator wants to modify time entry records which are already entered by other users, he can just add/edit billing rate record in history with the selection of checkbox **[Update all records within time range]**. It will update existing time entry records also with this newly updated billing rate in the specified date range.
- Billing Rates date range should not overlap with other rates in the same grid. This will result in wrong reporting and billing.

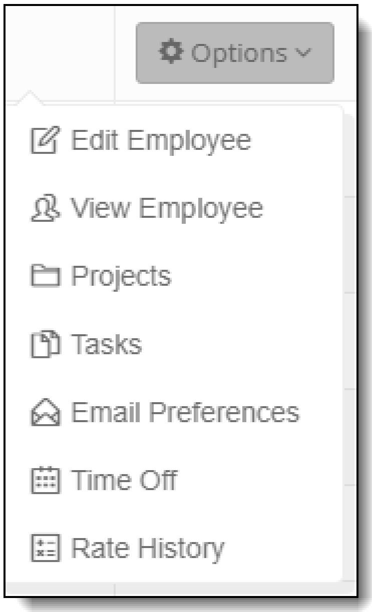
Billing Rate History access of Employee:

Below are the steps to access Billing Rate of an employee,

1. Navigate to **Employees** (Link is on the top-left).



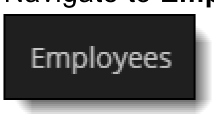
2. Click the **Options** gear icon on the employee you want to access Billing Rates.



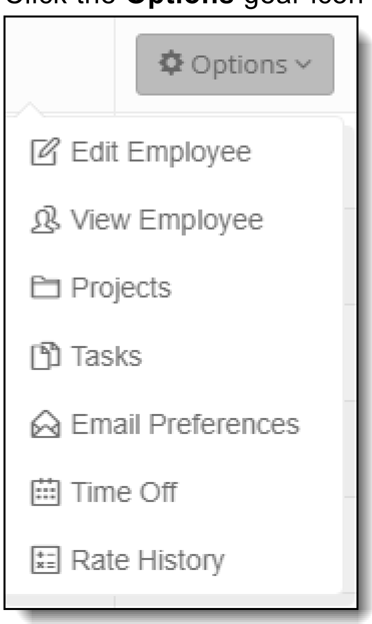
   a. Click on the **Rate History** option.

Adding New Billing Rate Employee own billing Rate:

1. Navigate to **Employees** (Link is on the top-left).



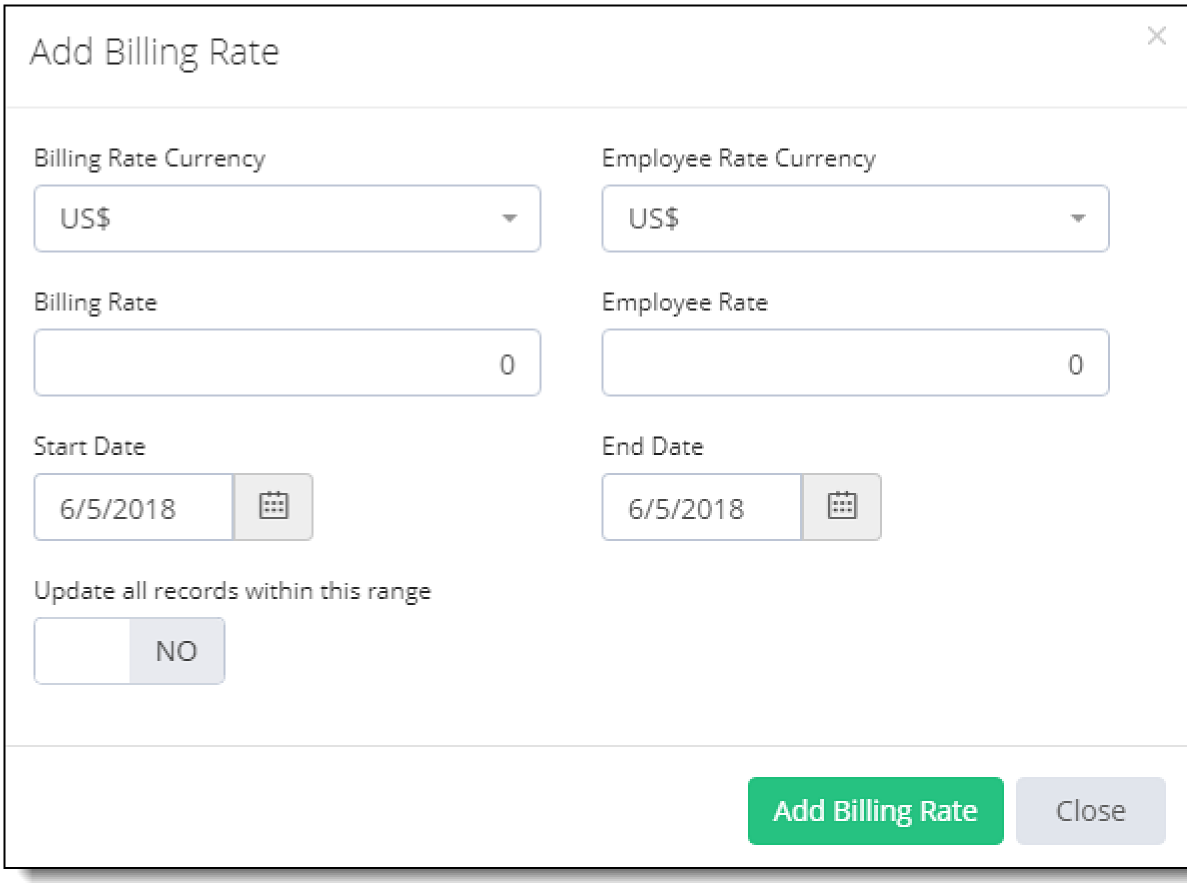
2. Click the **Options** gear icon on the employee you want to access Billing Rates.



a. Click on the **Rate History** option.

3. Click **+ Add Billing Rate** green button on the top right of the screen.

4. In the **Add Billing Rate** screen that appears:

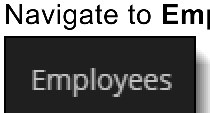


a. Select the Billing Rate Currency, Employee Rate Currency, Billing Rate, Employee Rate, Start and end date of this rate.

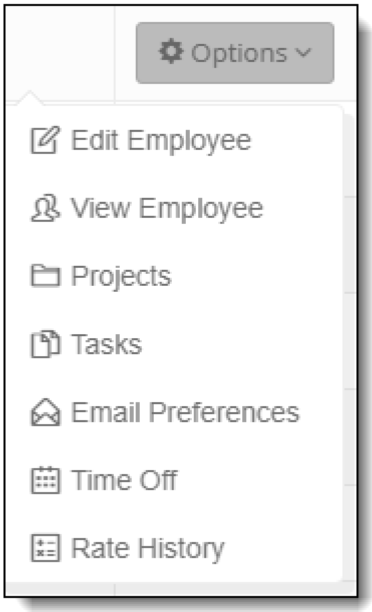
b. If there are some entries already exist of this employee and you want to update those entries with this rate, Click **Update all records within this range** and then Click **Add Billing Rate** Button.

Edit Existing Billing Rate:

1. Navigate to **Employees** (Link is on the top-left).

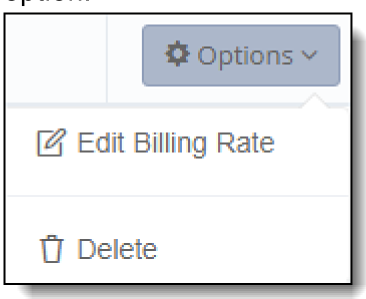


2. Click the **Options** gear icon on the employee you want to access Billing Rates.

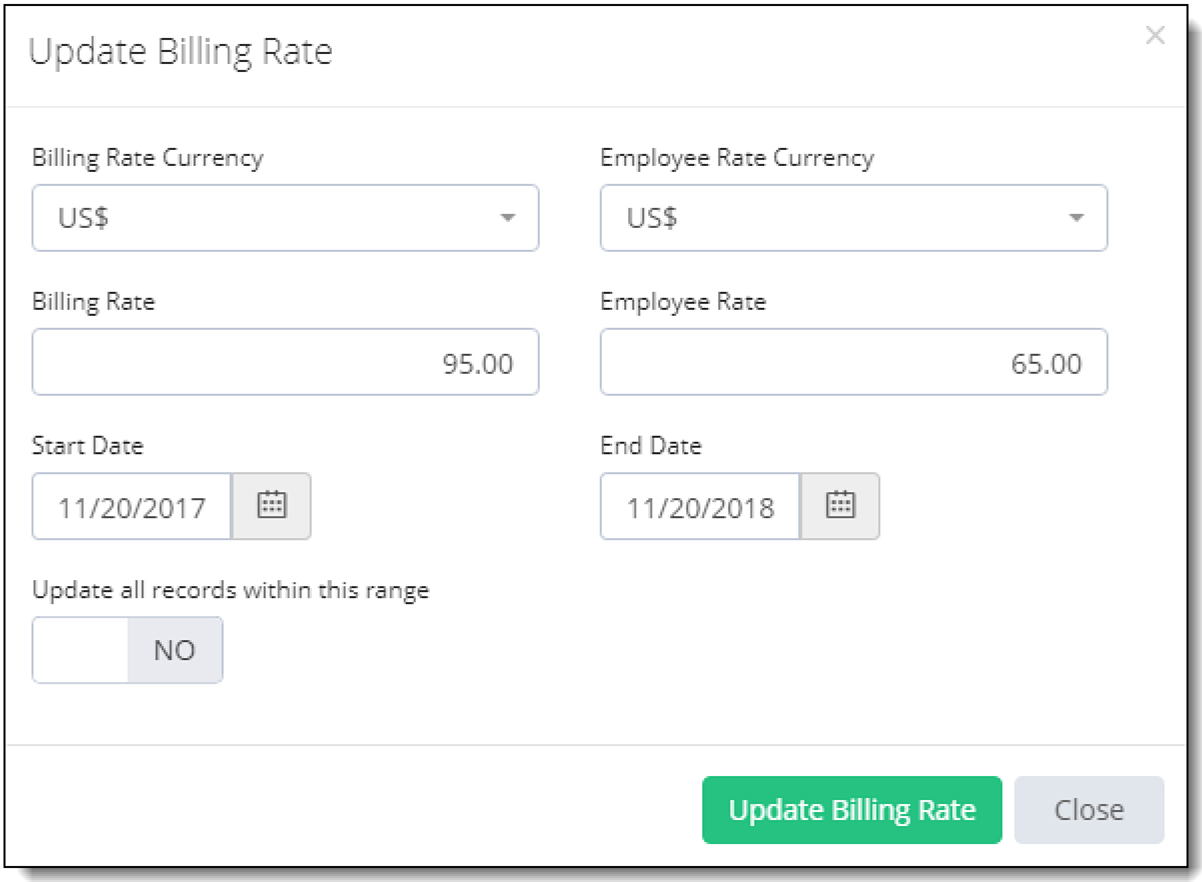


   a. Click on the **Rate History** option.

3. Click on the **Options** gear button against the billing rate you want to Edit and select **Edit Billing Rate** option.



4. In the **Edit Billing Rate** screen that appears:

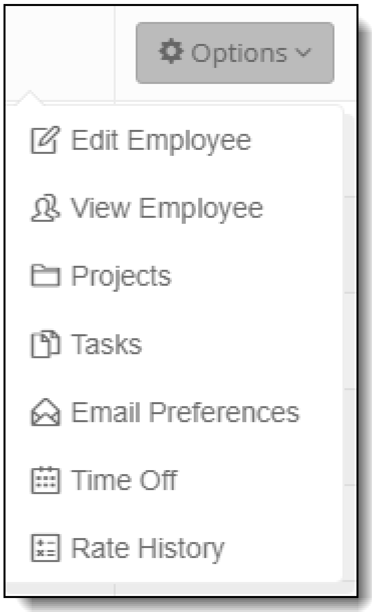


   a. Make the changes in the required field(s) and if there are some entries already exist of this employee and you want to update those entries with this rate, Click **Update all records within this range** and then Click **Update Billing Rate** Button.

Deleting Existing Billing Rate:

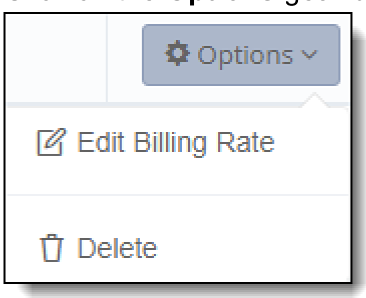
1. Navigate to **Employees** (Link is on the top-left).

2. Click the **Options** gear icon on the employee you want to access Billing Rates.

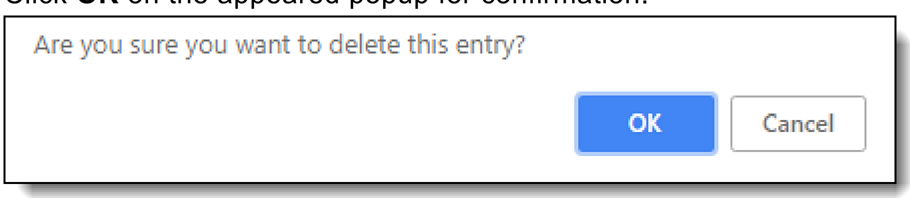


   a. Click on the **Rate History** option.

3. Click on the **Options** gear button against the billing rate you want to delete and select **Delete** option.



4. Click **OK** on the appeared popup for confirmation.

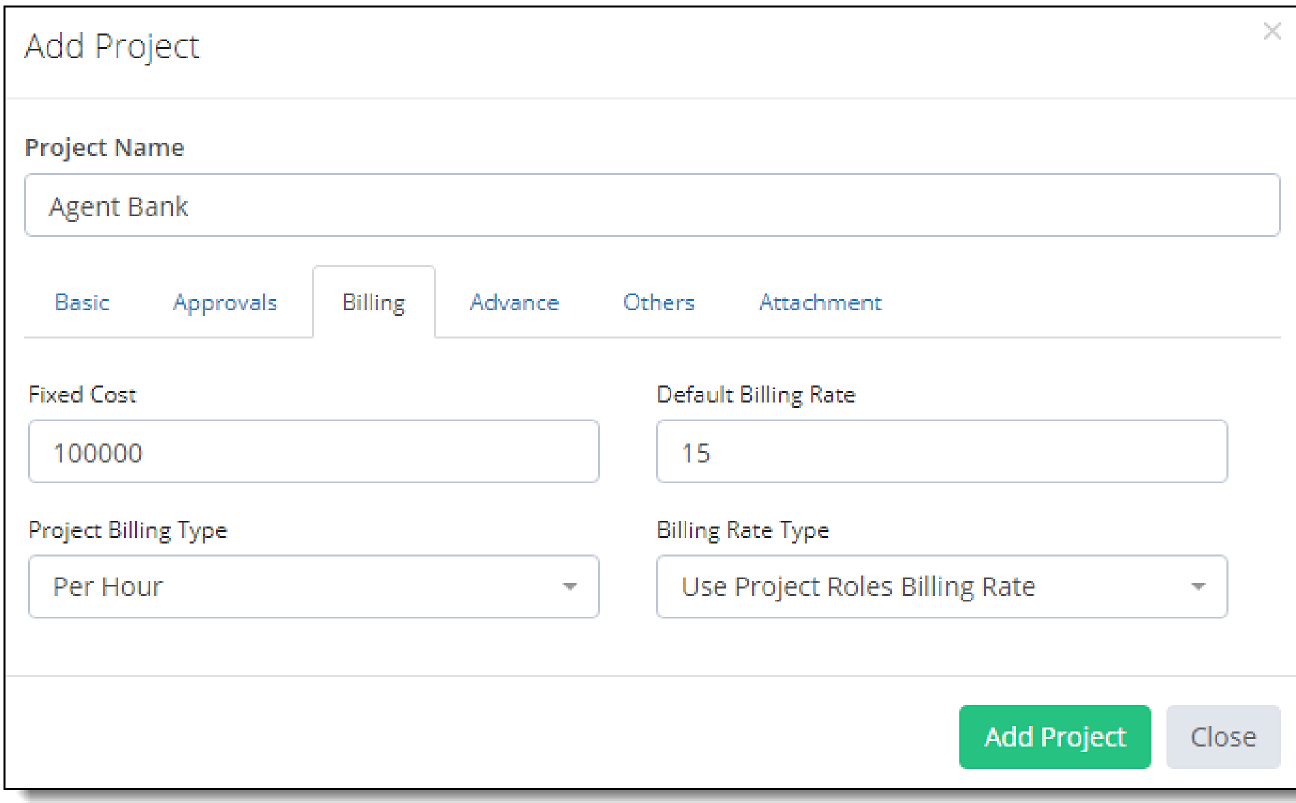


7.3. Project Role Billing Rate

System administrator can define **[Billing Rate Type]** as **[Project Roles Billing Rate]** to setup role wise billing rate for project. There are three steps for setting up role wise billing rate of project.

Setting up Role based billing rate:

1. Define **[Project Roles Billing Rate]** in **[Billing Rate Type]** when adding new project in TimeLive.



2. On click on **[Add]** button, TimeLive will open a **[Project Role]** definition page where use can setup number of resources required for each role and billing rate of every role. Just select your required **[Roles]** for which you want to define billing rate including its number of resources requirements.

3. On clicking on [Update] button, timelive will open employee role mapping page where system administrator can define employee for each of the role requirement defined in [Number of Resources] for every role. Just map your required employee for each of the role and check on **[Selected]** checkbox. After completing this role employee mapping, click on **[Update]** to update these changes.

Launching billing rate history page of Role billing rate:

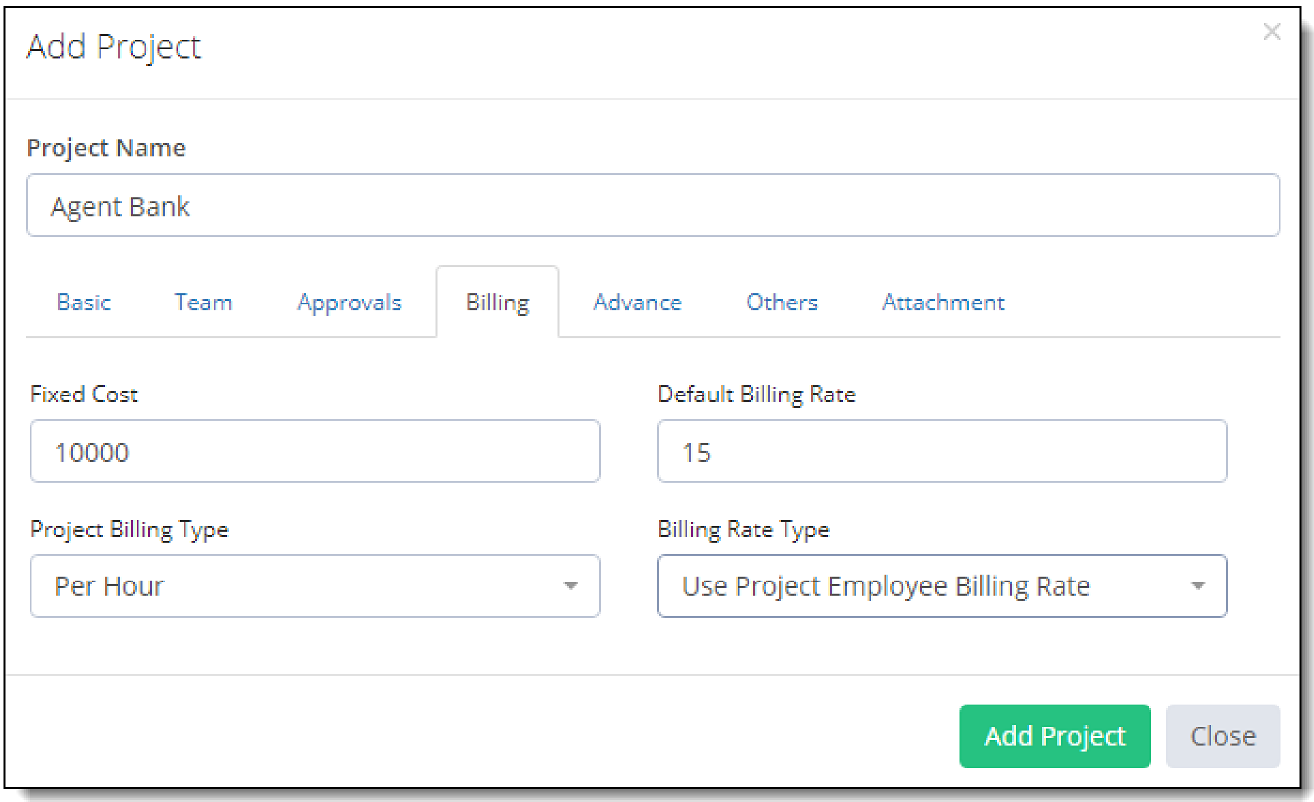
1. Using role billing rate history, administrator can add new billing record with its applicable date and he can also modify any of existing record also by clicking on **[Edit]** link.
2. Administrator can delete any of these billing history record by clicking on **[Delete]** link.
3. Billing history should be configure for in all area before entering timesheet record. Otherwise time entry will not pick billing rate defined after recording time entry.
4. In case if Administrator wants to modify billing rate of exiting time entry records, he can just add / edit billing rate record in history with selection of checkbox **[Update all records within time range]**. It will update existing time entry records also with this new updated billing rate in specified date range.
5. To launch billing rate history page, just click on **[History]** link of any role selected for current project.

7.4. Project Employee Billing Rate

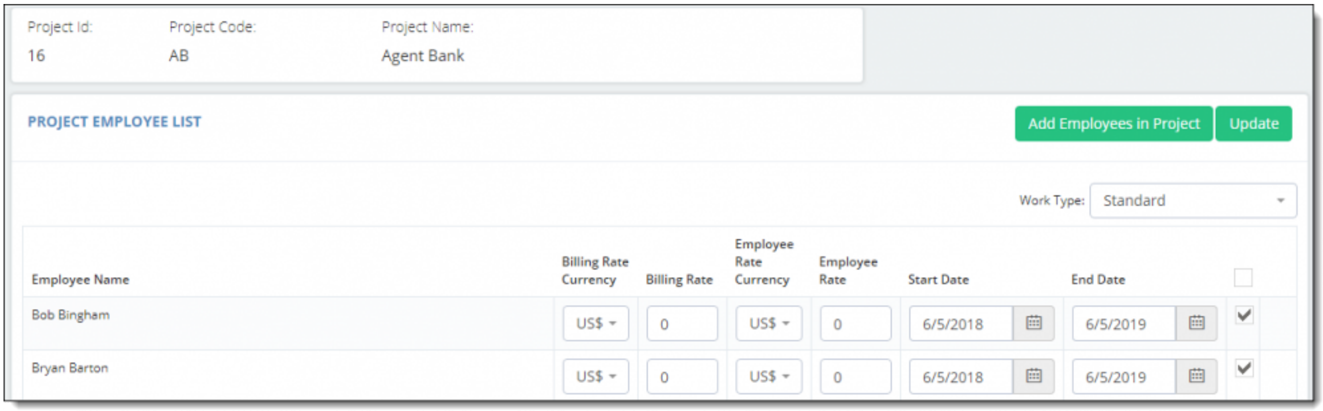
System administrator can define **[Billing Rate Type]** as **[Use Project Employee Billing Rate]** to setup project based employee billing rate. Project based employee billing rate is useful for projects where employee billing rate is different for every different project. There are two steps for setting up project based billing rate in a project.

Setting up Project based employee billing rate:

1. Define **[Use Project Employee Billing Rate]** in **[Billing Rate Type]** when adding new project in TimeLive.



2. On click on **[Add]** button, TimeLive will open a **[Project Team]** definition page where user can select project team required for this project with their billing rate.



3. After selecting employees which are required to assigned in this project, click on **[Update]** to update these changes.

Launching billing rate history page of Project based employee rate:

1. Using billing rate history, administrator can add new billing record with its applicable date and he can also modify any of existing record also by clicking on **[Edit]** link.
2. Administrator can delete any of these billing history record by clicking on **[Delete]** link.
3. Billing history should be configure before entering timesheet record. Otherwise time entry will not pick billing rate defined after recording time entry.
4. In case if Administrator wants to modify billing rate of exiting time entry records, he can just add / edit billing rate record in history with selection of checkbox **[Update all records within time range]**. It will update existing time entry records also with this new updated billing rate in specified date range.
5. To launch billing rate history page, just click on **[History]** link of any role selected for current project.

8. Invoicing

8.1. About Invoicing

Invoices are usually made for clients. It is a receipt which shows client name, invoice number, invoice date, billing rate, bill hour, timesheet description and amount. Generated invoices are stored in the database along with your time logs and expense and can be viewed or edited at any time. Invoices can be created based on the existing time entries and expenses or alternatively, new records can be added as you're working on the invoice. As you keep saving more invoices, you can view statistics including total due to be paid, total taxes collected.

Billing Managers can create invoices and can also check the previous records of the invoice by navigating to **Billing > Invoice Management** menu on the top-right side of the screen.

An invoice can then **Print** or **Generate PDF** for sending it to the client. Invoice use company logo can be uploaded from **Admin Options** > **Preferences** > **General Tab**.

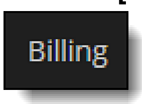
8.2. Manage Invoices

Billing managers can create invoices and can also check the previous records of the invoice by navigating to **[Billing] —> [Invoice Management]** menu on the left side menu.

Adding New Invoice:

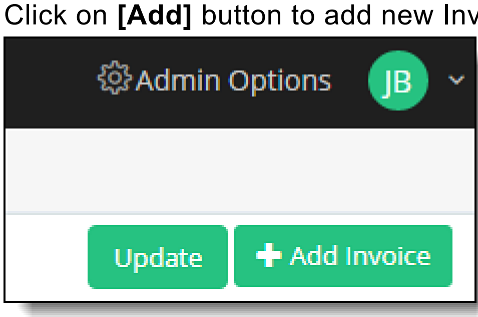
To add new invoice:

1. Click on **[Billing]** in top menu.



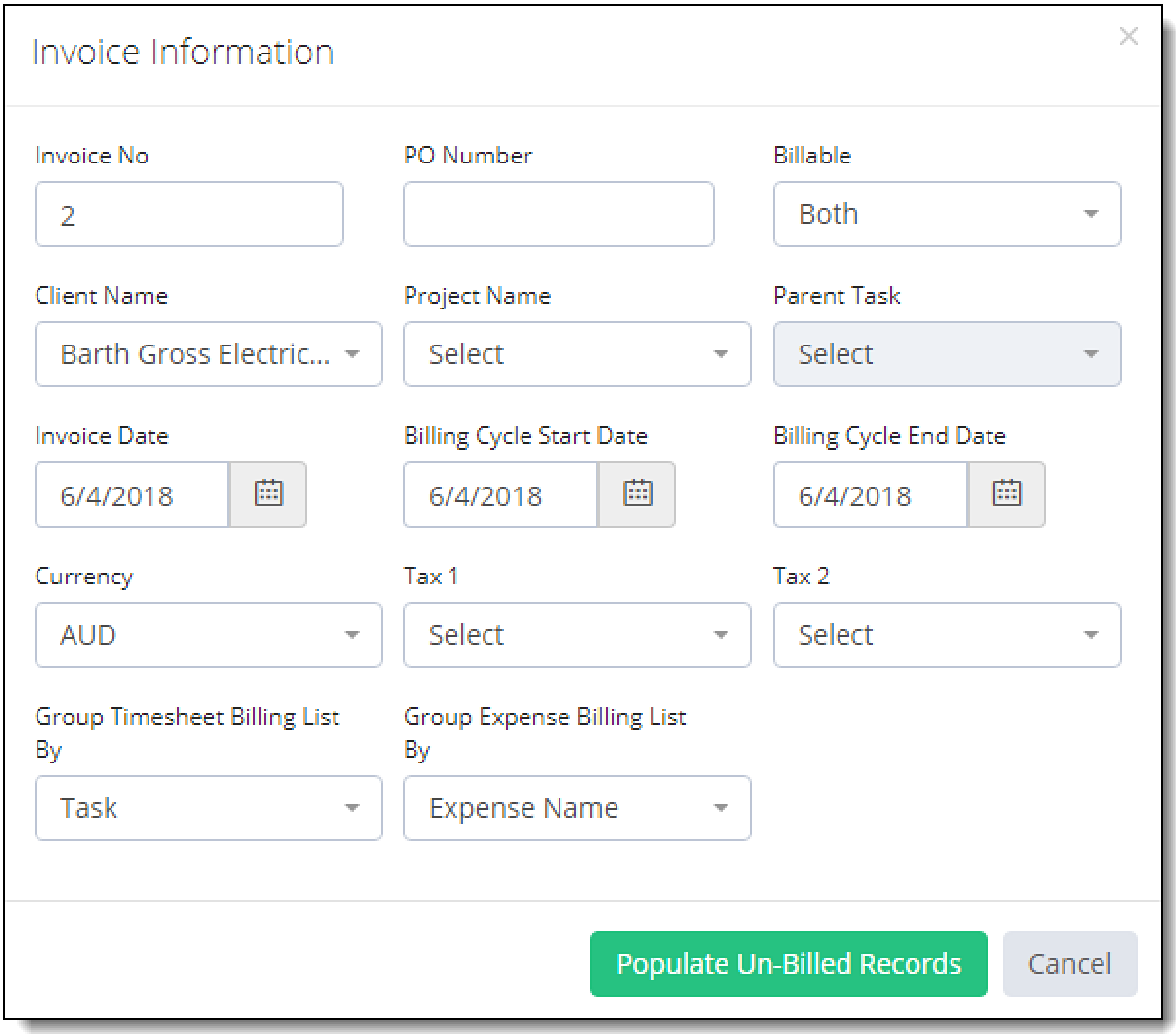
2. Click on **[Invoice Management]**.

3. Click on **[Add]** button to add new Invoice.



4. Click on **[Generate Invoice]** button.

5. In the **Invoice Information** screen that appears, select:



a. Select the **[Client Name]** from the list (to whom you want to make the invoice).
b. In order to create invoice for a single project, select **[Project]** from the list. Keep it as "All" if you want to create invoice for all projects for a client.
c. Enter the date range in **[Billing Cycle Start Date]** and **[Billing Cycle End Date]** (period to be billed).
d. Click on the **[Populate Un-billed Records]** to populate all un-billed time and expense entries in invoice for the given date range.
e. Click on **[Update]** button to update the record.
f. Click on **[Update Time Entry and Expense Entry as Billed]** once you finished creating your invoice. This will update all time and expense entries as "Billed" in database in order to avoid duplicate invoicing of same data.

Editing Invoice:

1. In **[Time Expense Invoice List]**, click on **[Edit]** link of record which you want to modify.
2. Update your required modification in **[Invoice information]** form.
3. Click on **[Update]** to update this record.

Deleting Invoice:

1. In **[Time Expense Invoice List]**, click on **[Delete]** link of record which you want to delete.
2. Click on **[Yes]** on delete confirmation dialog.

Printing Invoice:

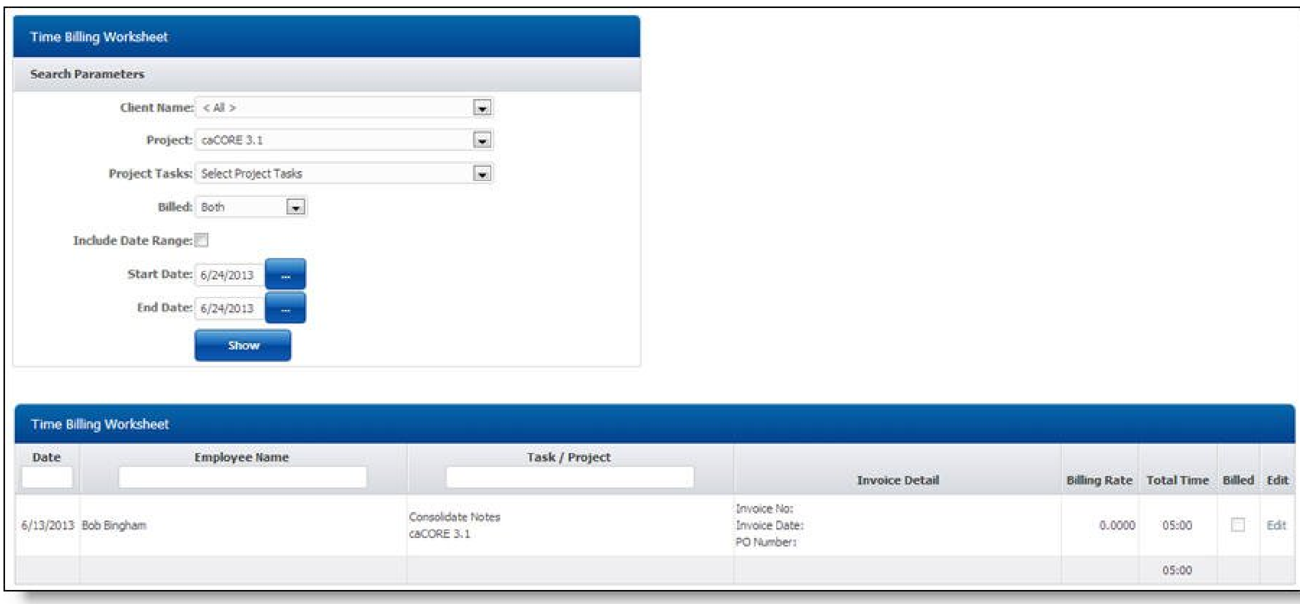
1. Click on **[Print]** link to print your invoice. Invoice can be exported to PDF or XLS on print page using export option available.
2. Invoice use company logo which is uploaded in **[Admin Options] —> [Preferences]**.

Updating Time and Expense Records as Billed:

1. At the end of finalizing the invoice, you need to mark ”Billed” for the time & expense records included in the current invoice to prevent double invoicing of already billed entries.
2. Navigate to **[Invoice Management]**.
3. Click on **[Edit]** link of record which you want to modify.
4. Click on **[Update Time Entry and Expense Entry Records As Billed]** to make all time & expense entries ”Billed”.
5. **[Billed]** status of time entry records and expense entry records can be changed from **[Billing] — [Time Billing Worksheet]** and **[Billing] —> [Expense Entry Worksheet]**.

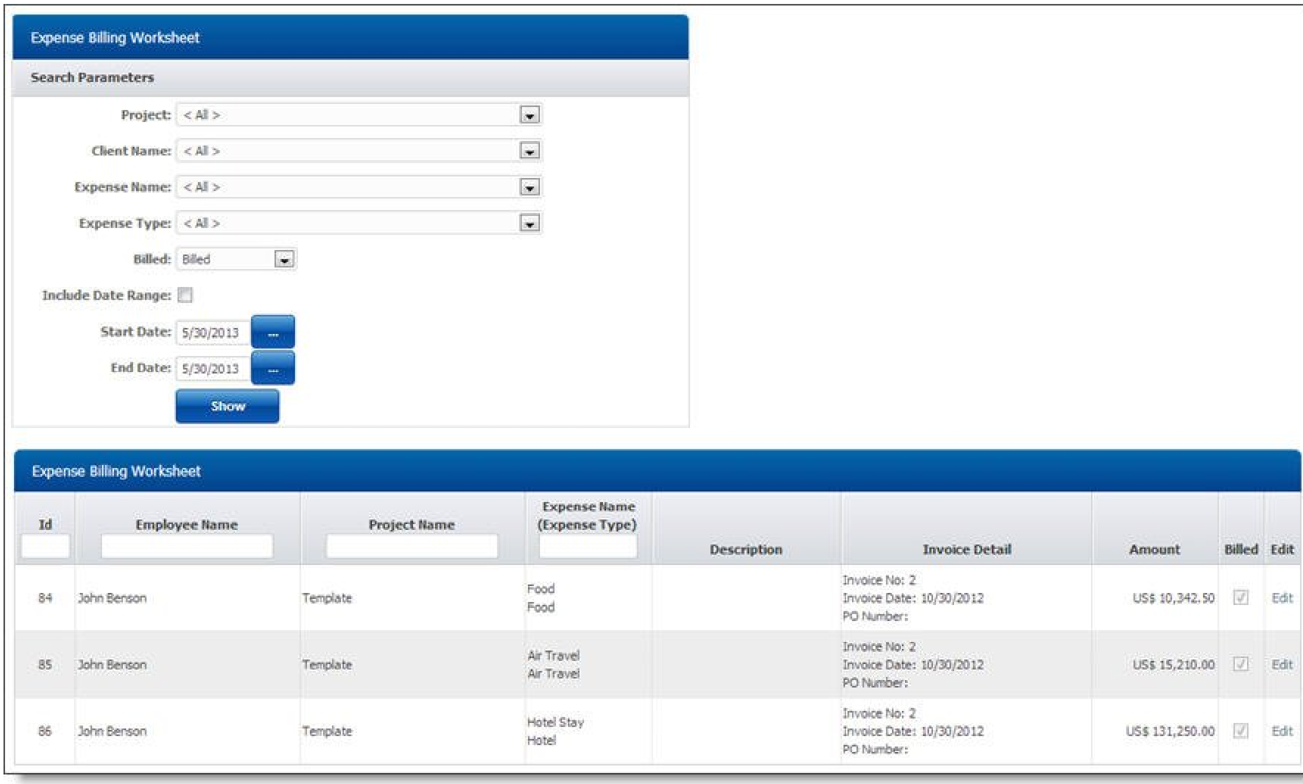
8.3. Time Billing Worksheet

- Time billing worksheet allow accountant to manage and track all billable time entries with their billing status and invoice number.
- It contains all billable time entries.
- Time billing worksheet contains all the invoices with total billing time, billing rate and all other invoice related information.
- You can change the status of invoices and marked them as “Billed” / “Un-billed”.
- To easily search for your required records using filters available there.
- To delete all invoices, use the option “Check All” and then click on **[Delete Selected Item]** to delete all checked entries.



8.4. Expense Billing Worksheet

- Expense billing worksheet allow accountant to manage and track all billable expense entries with their billing status and invoice number.
- It contains all billable expense entries and expense sheets.
- Expense billing worksheet contains all the invoices with total billing expense, amount and all other invoice related information.



9. Project Management

9.1. Gantt Chart

Viewing tasks in black and white cannot give that much info at a glance but viewing Tasks through Gantt chart is way better for understanding and task allocation.

Viewing Gantt Chart for Project Tasks

1. Login TimeLive with a user which has **Gantt View/Edit Permission** from **Admin Options > Role Permissions**
2. Navigate to **Projects** and search for the project of which Gantt View is required
3. Click the **Options** gear icon against the Project and click **Gantt** option

Through Gantt, User can add new tasks, allocate resources, its utilization and can make changes in the existing tasks.

It will automatically update the tasks and its changed details including Task Team.

Changing Task Start / End Date:

Click on Start/End Date in the required Task and change it to the required date.

Resource Allocation:

Double Click on the required task. Task Editing Window will pop-up. You can Add/Remove new resource and allocate hours for the same in this window.

10. Email Notifications and Alerts

10.1. Overview

A variety of e-mail notifications and alerts can be configured to be automatically sent to employees and managers when managers and employees are required to perform certain duties such as submitting or approving time and expenses.

Available E-mail Notifications include:

- **Send e-mail to approving managers when time/expenses are awaiting their approval**.
    - To help with getting time and expenses approved in a timely manner, you can setup notifications so the approving managers are sent an e-mail whenever time/expenses are awaiting their approval. Configure notifications to send one daily (batch) e-mail containing summary information for all sheets awaiting their approval.
- **Send e-mail to employees when time/expenses have been rejected by their manager**.
    - When a manager rejects time or expenses submitted by employees, there is usually a need to have the issue quickly resolved. By instantly sending e-mails to the employees, that alert them to the rejected time/expenses, both parties (approving managers and employees) can resolve any problems in an effective manner.
- **Send e-mail to employees detailing their project assignments**.
    - Assignment notifications inform the selected users about the projects and tasks that are assigned to them.
- **Send e-mail to employees when their timesheet are incomplete**
    - Incomplete Timesheet Notifications alert the selected users when their timesheet is incomplete based on a specified number of hours.

10.2. Email Settings

Following settings are required to set up e-mail notifications from TimeLive, if you are using self-hosted TimeLive version.

1. Defining of **[SMTP Server]**, **[SMTP Username]** and **[SMTP user's password]** in system setting page.

2. Download this free SMTP test utility and test your SMTP Settings using this utility on the same server where you have TimeLive installed.

3. Updating of **[Application URL]** with correct TimeLive self-hosted version URL in system setting page.

4. Updating of **[From E-mail address]** and **[From E-mail address display name]** in **[Admin Options] > [Preferences]** while logged-in using admin user in TimeLive.

11. Admin Setup

11.1. Launching Admin Options in TimeLive

TimeLive Administrator can access Administration Page by clicking on **Admin Options**. on Right-top of the Application. The administrator can set up all the features of TimeLive through below options.

Application Settings which can be configured using:

- **Organization Setup**
  - Locations
  - Departments
  - Roles
  - Employee Types
  - External Users

- **Application Setup**
  - Email Notification Preferences
  - Role Permissions
  - Manage Terminology
  - Preferences
  - Status Type
  - Custom Fields
  - Feature Management
  - License Manager
  - Approvals

- **Timesheet Setup**
  - Timesheet Period Types
  - Time Entry Archive
  - Holiday Types
  - Holidays
  - Absence Type
  - Work Types
  - Cost Center
  - Working Days

- **Project / Task Setup**
  - Task Types
  - Priorities

  - Project Types
  - Project Templates

- **Time Off Setup**
  - Time Off Types
  - Time Off Policies

- **Expense Setup**
  - Expense Types
  - Expense
  - Payment Method
  - Expense Entry Archive

- **Currencies and Tax**
  - Currencies
  - Tax Code
  - Tax Zone

- **Billing Setup**
  - Billing Types

- **Import / Export**
  - Import / Export CSV/Quickbooks / Microsoft Project
  - Attachment Archive

- **Integration**
  - QuickBooks Online
  - API
  - Zapier

- **Account Setup**
  - Disable Account

11.2. Organization Setup

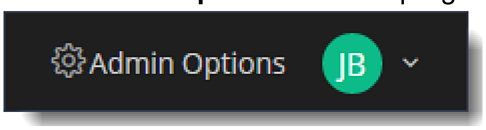
11.2.1. Locations

System administrator can setup **Locations** using **[Admin Options] —> [Locations]**

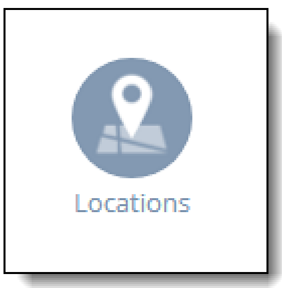
Adding Location:

If you plan to implement locations, you should define your locations before setting up your employees, allowing you to add an employee and assign the employee a location in one easy step.

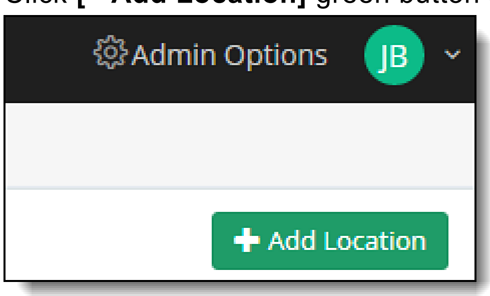
1. Click **Admin Options** at the top right.



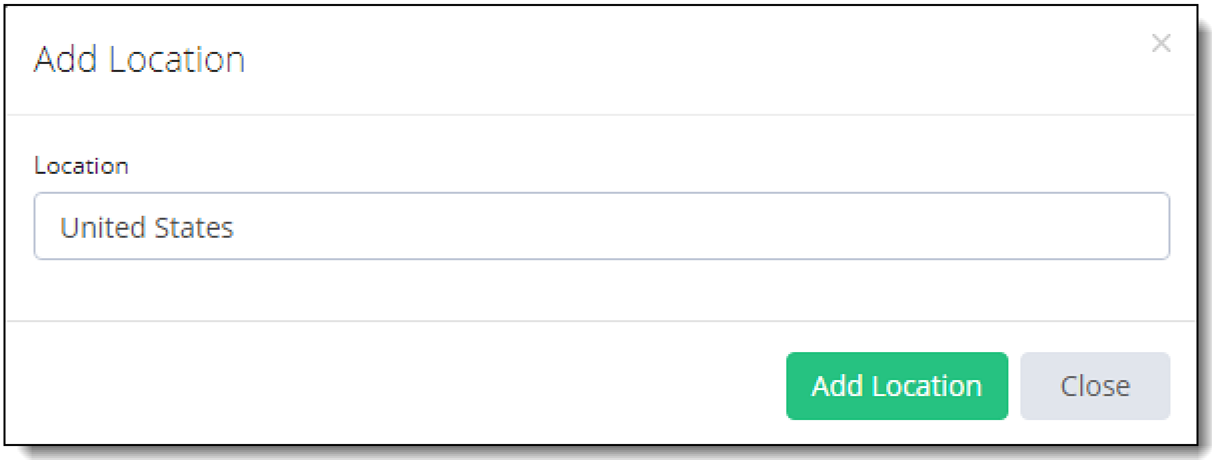
2. Click **Locations** icon under **Organization Setup**.



3. Click **[+ Add Location]** green button at the top right.



4. In the **Add Location** screen that appears, enter:

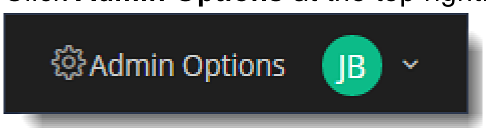


   a. The name of your new location. Don't worry, you can always change this later.
   b. Now click the green **Add Location** Button.

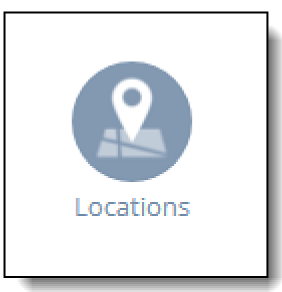
Your new location will be created and then immediately appear on your **Location** screen.

Editing/Changing Name of the Location:

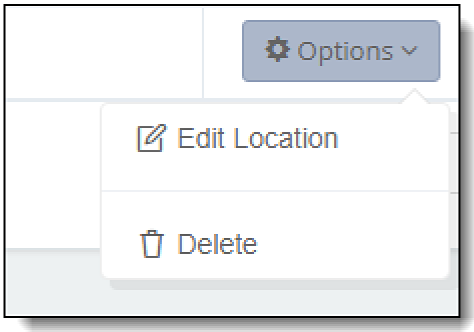
1. Click **Admin Options** at the top right.



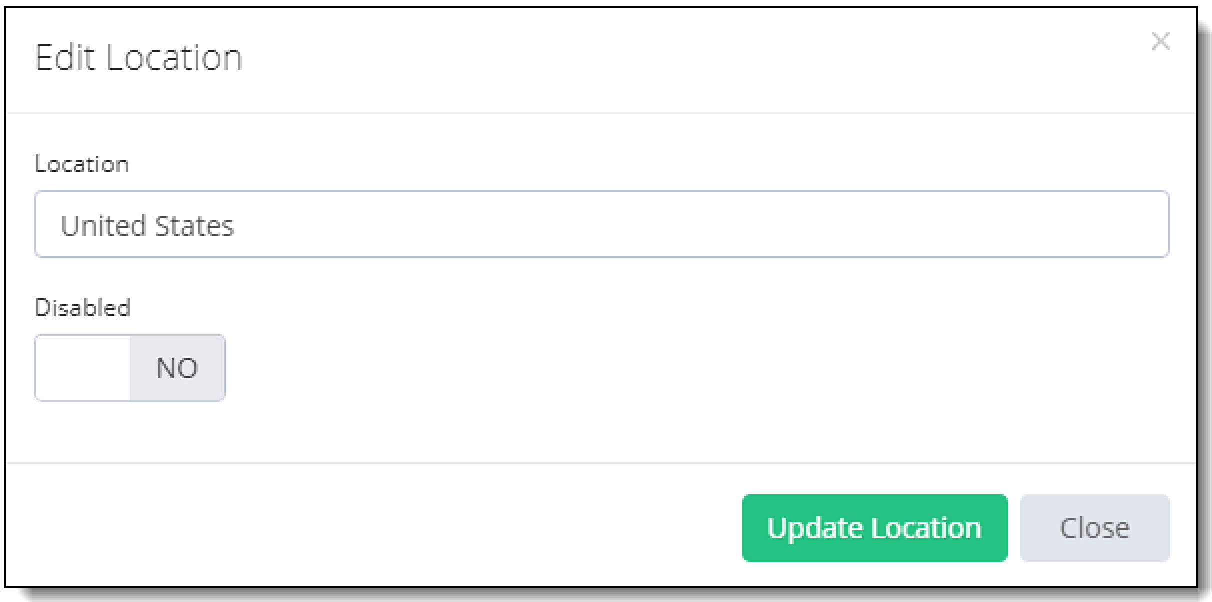
2. Click **Locations** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Location you want to **Edit/change name** then click **Edit Location** Button.



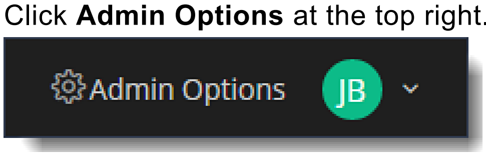
4. In the **Edit Location** screen that appears, where you can:



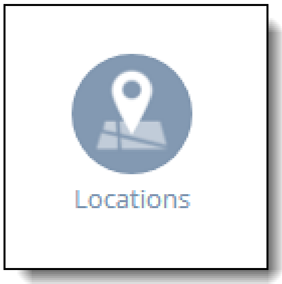
   a. Change the name of an existing **Location**.
   b. After making changes, click **Update Location** Button.

Disabling Existing Location:

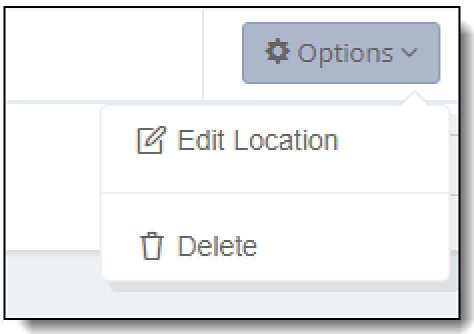
1. Click **Admin Options** at the top right.



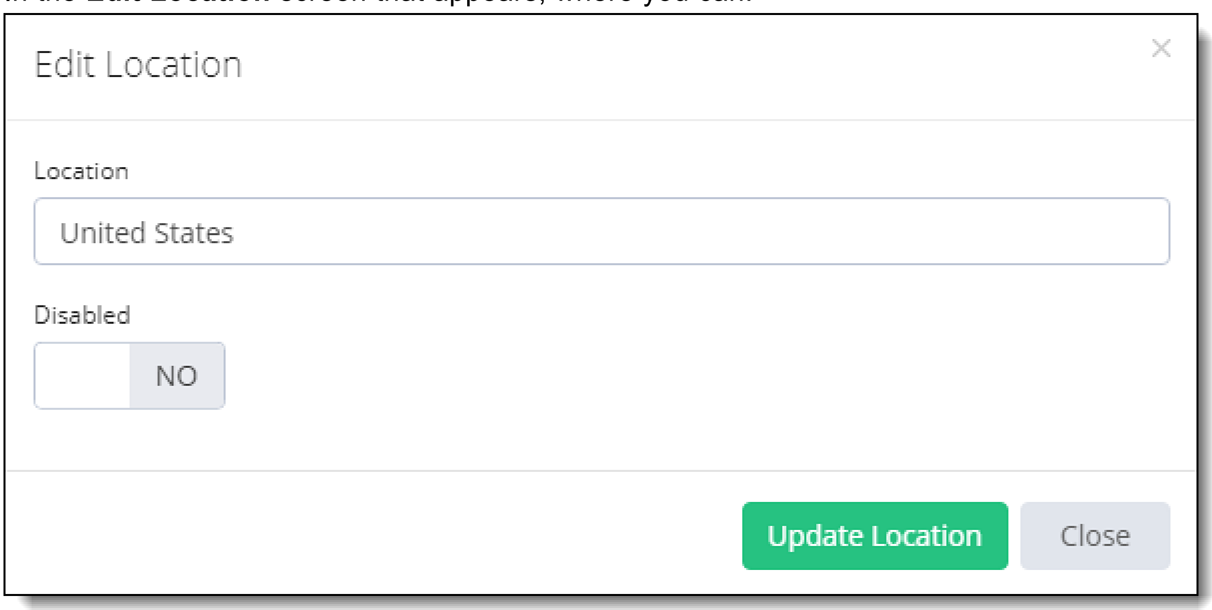
2. Click **Locations** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Location you want to **Edit/change name** then click **Edit Location** Button.



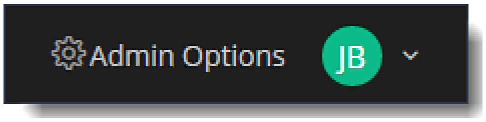
4. In the **Edit Location** screen that appears, where you can:



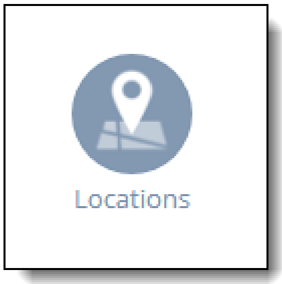
   a. Click **No** under the **Disabled** caption. It will be changed to **Yes**.
   b. Click **Update Location** button.

Deleting the Location:

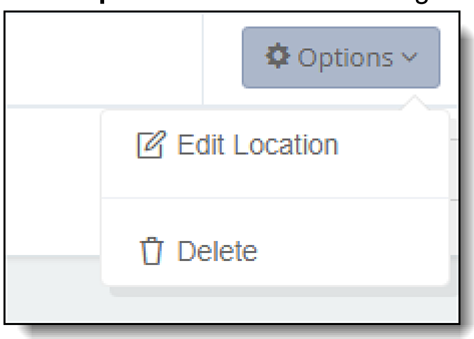
1. Click **Admin Options** at the top right.



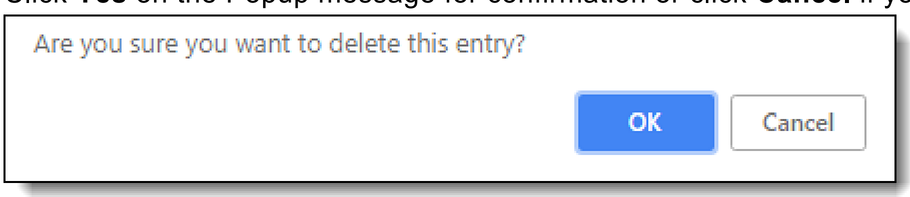
2. Click **Locations** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Location you want to **Delete** then click **Delete** Button.



4. Click **Yes** on the Popup message for confirmation or click **Cancel** if you do not want to delete.



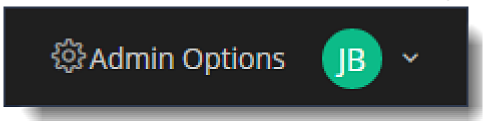
11.2.2. Departments

Department is a division of employees that may define the hierarchical structure, geographic regions, or functional groups within your organization.

Adding Department:

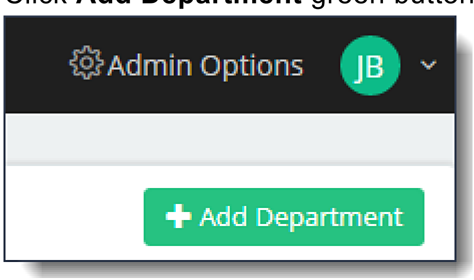
If you plan to implement departments, you should define your departments before setting up your employees, allowing you to add an employee and assign the employee a department in one easy step.

1. Click **Admin Options** at the top right.

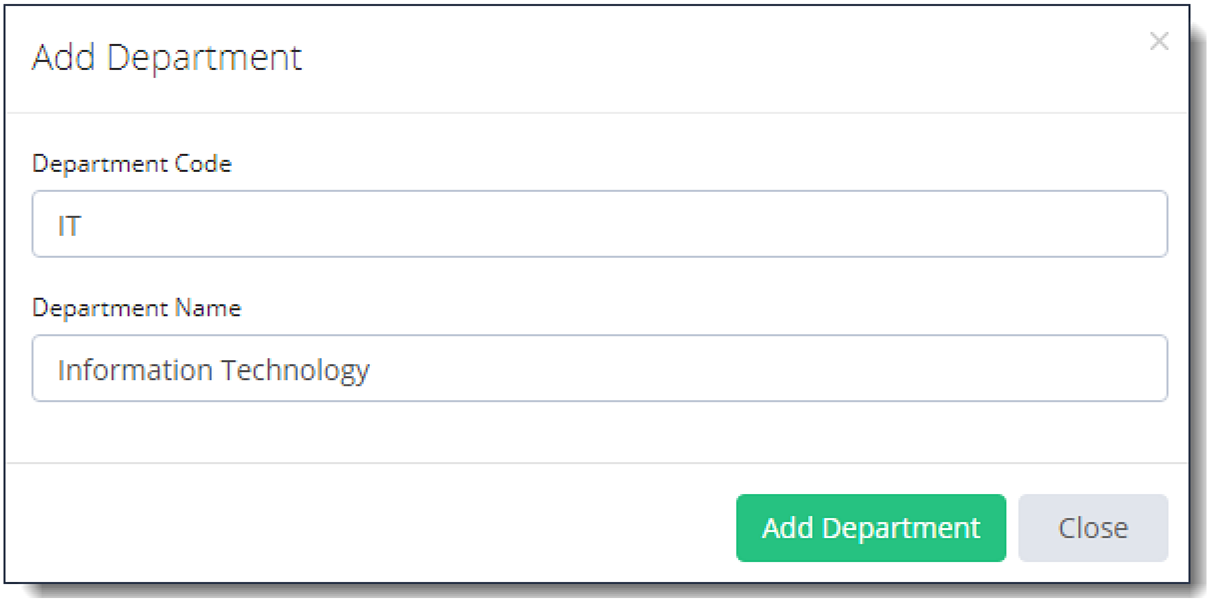


2. Click **Departments** icon under **Organization Setup**.

3. Click **Add Department** green button at the top right.



4. In the **Add Department** screen that appears, enter:

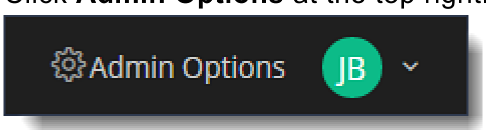


   a. The **Code** of your new department. e.g. you can enter short form of **Information Technology** like **IT**. If you're not using department code, just leave it blank.
   b. The **Name** of your new department. Don't worry, you can always change this later.
   c. Now click the green **Add Department** Button.

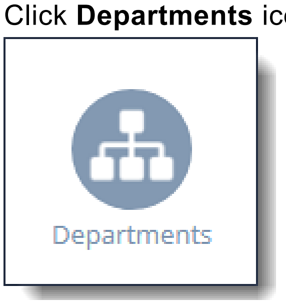
Your new department will be created and then immediately appear on your **Department** screen.

Editing/Changing Name of the Department:

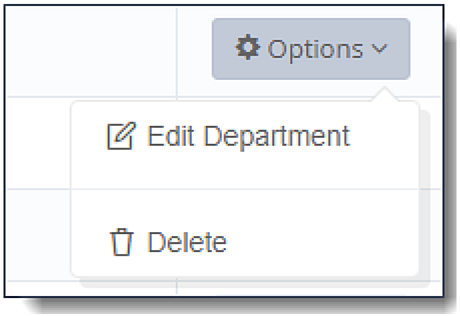
1. Click **Admin Options** at the top right.



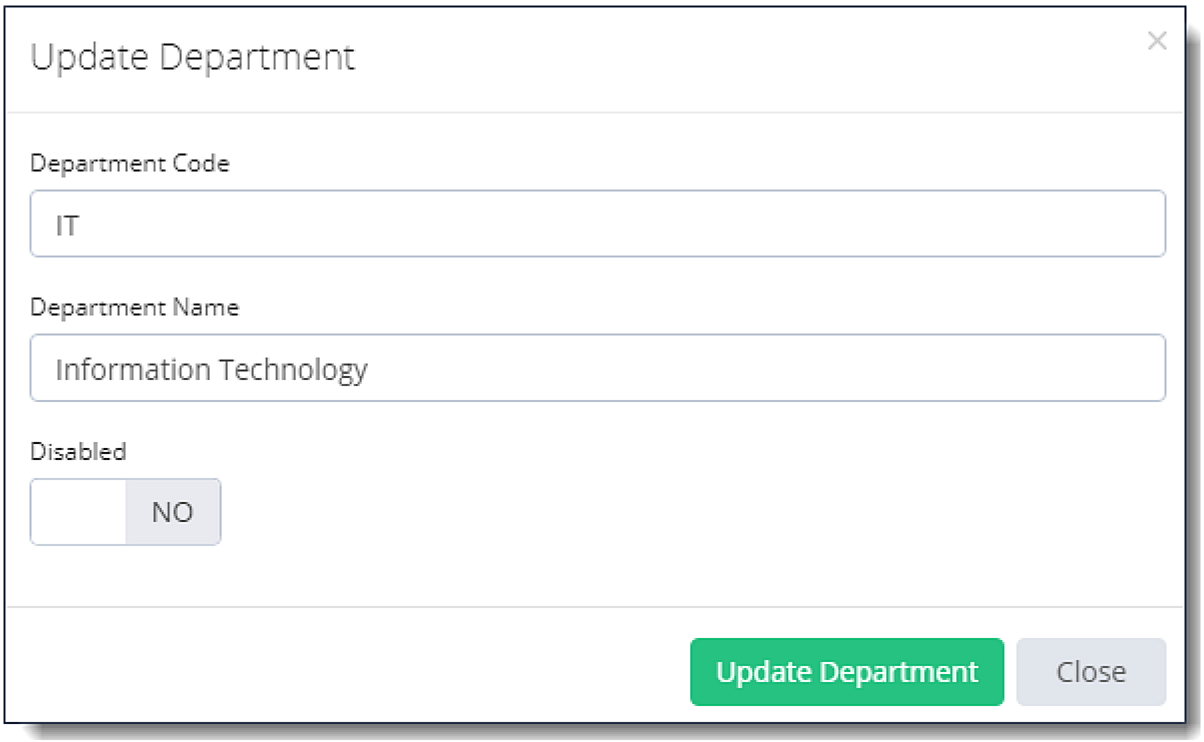
2. Click **Departments** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Department you want to **Edit/change name** then click **Edit Department**.



4. In the **Edit Department** screen that appears, where you can:

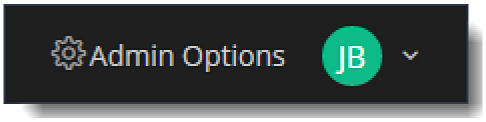


   a. Change the **Code** of an existing department.
   b. Change the **Name** of an existing department.
   c. After making changes, click **Update Department** Button.

Department will be updated and then immediately appear on your **Department** screen.

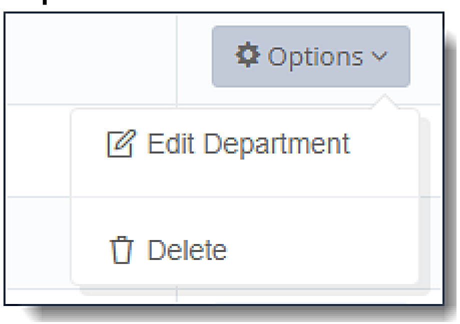
Disabling Existing Department:

1. Click **Admin Options** at the top right.

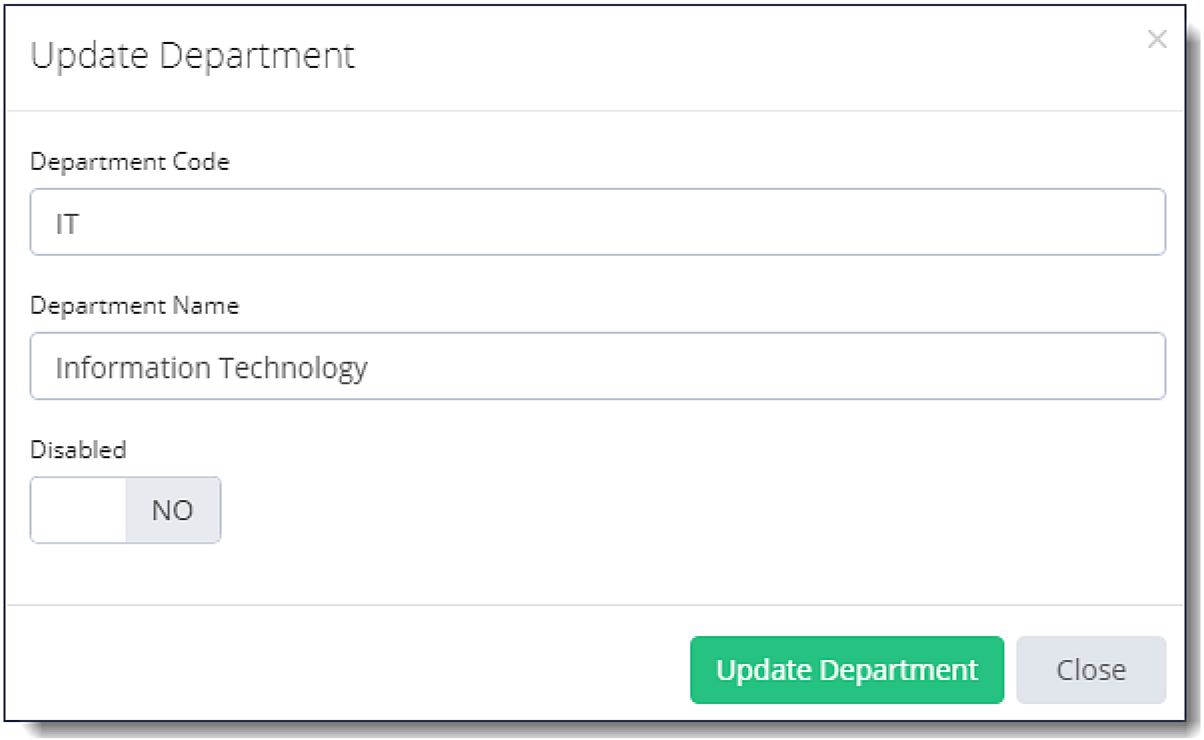


2. Click **Departments** icon under **Organization Setup**.

3. Click **Options** with **Gear Icon** against the Department you want to **Edit/change name** then click **Edit Department** Button.



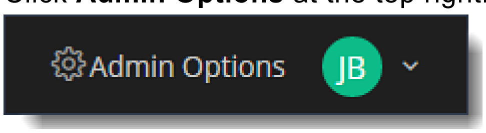
4. In the **Edit Department** screen that appears, where you can:



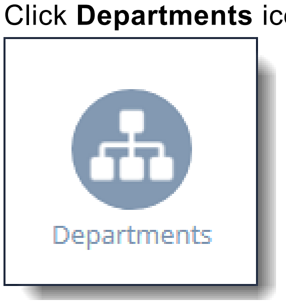
   a. Click **No** under the **Disabled** caption. It will be changed to **Yes**.
   b. Click **Update Department** button.

Deleting the Department:

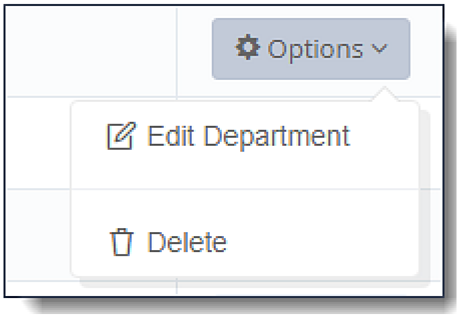
1. Click **Admin Options** at the top right.



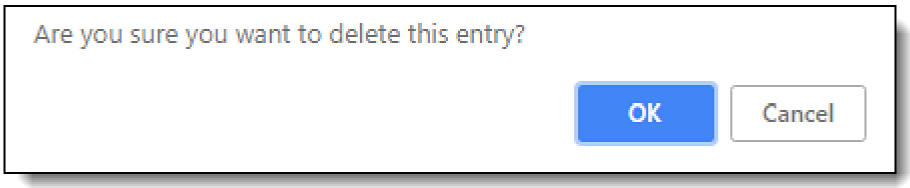
2. Click **Departments** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Department you want to **Delete** then click **Delete** Button.



4. Click **OK** on the Popup message for confirmation or click **Cancel** if you do not want to delete.



11.2.3. Roles

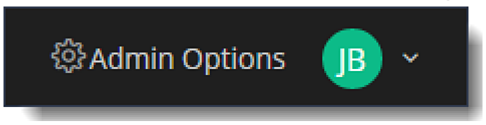
Roles are used to identifying types of work that are charged at specific rates. A role can describe a specific action/function or a level of expertise within the project team. Roles are initially created outside of the project and then assigned to the project and appropriate team members.

System administrator can setup **Roles** as per company requirements using **[Admin Options] —> [Roles]**

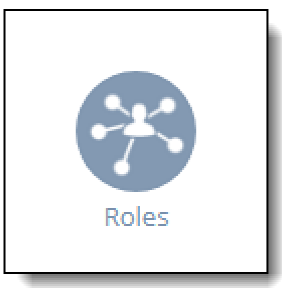
Adding Roles:

By default, the TimeLive application creates 4 default roles which TimeLive application used internally for its different functions. An administrator can add new roles based on their own organization requirements.

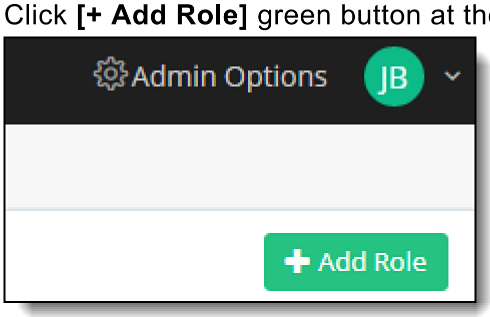
1. Click **Admin Options** at the top right.



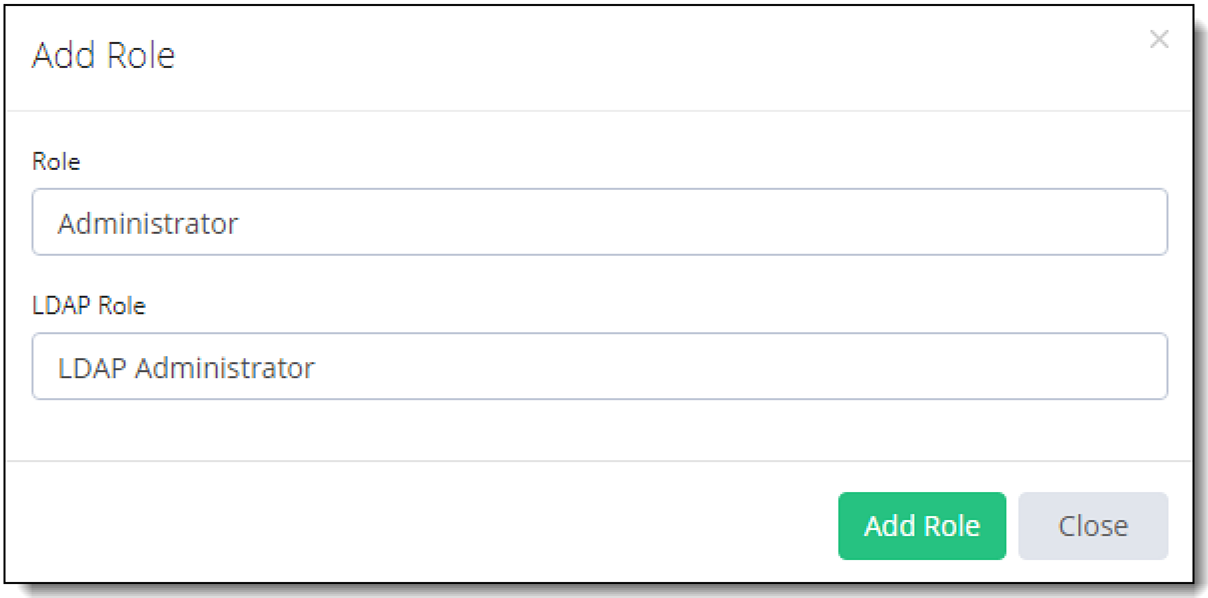
2. Click **Roles** icon under **Organization Setup**.



3. Click **[+ Add Role]** green button at the top right.



4. In the **Add Role** screen that appears, enter:

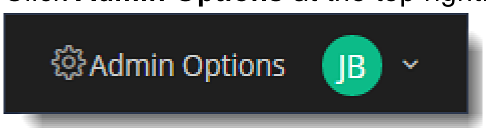


   a. The name of your new Role. Don't worry, you can always change this later.
   b. Now click the green **Add Role** Button.

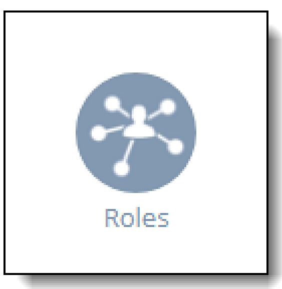
Your new Role will be created and then immediately appear on your **Roles** screen.

Editing/Changing Name of the Role:

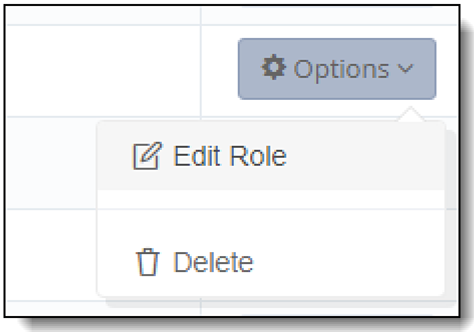
1. Click **Admin Options** at the top right.



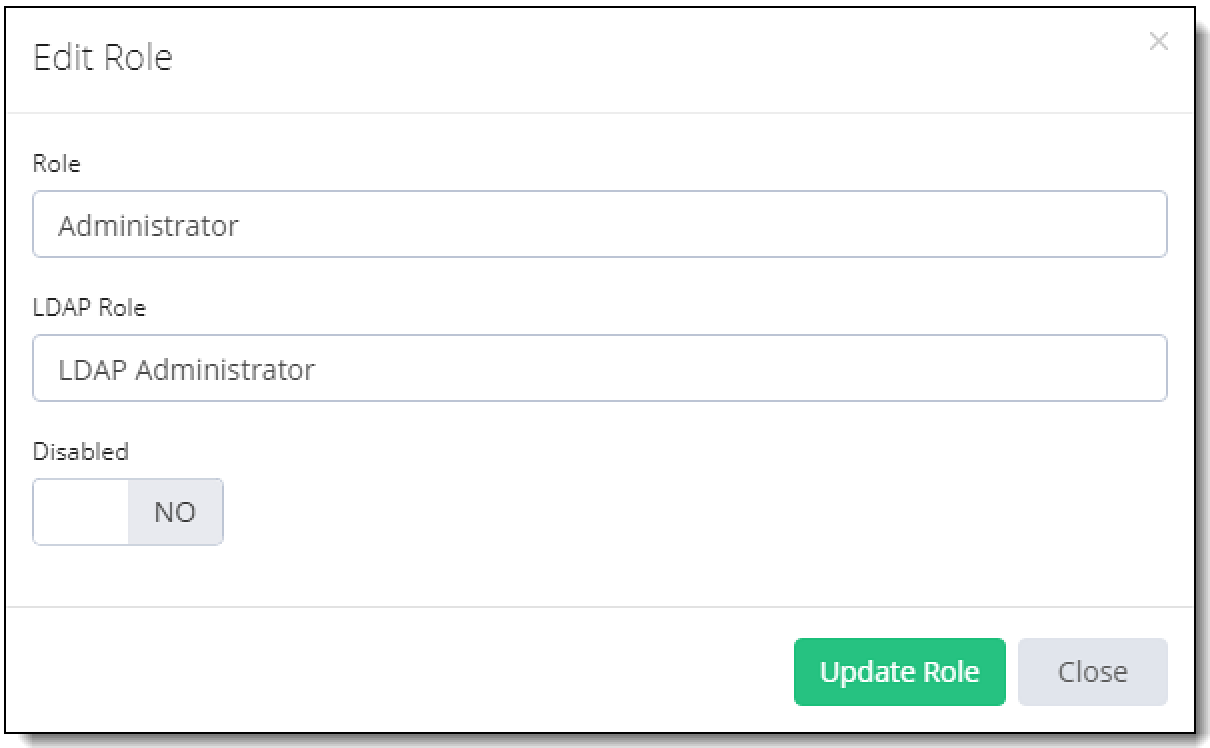
2. Click **Roles** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Location you want to **Edit/change name** then click **Edit Roles** Button.



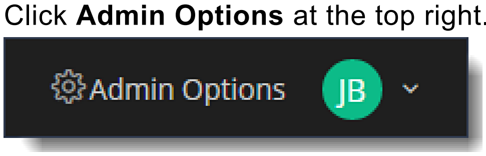
4. Role Popup will appear, where you can change the Role Name.



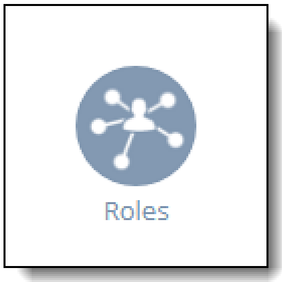
   a. After making changes, click **Update Role** Button.

Disabling Existing Role:

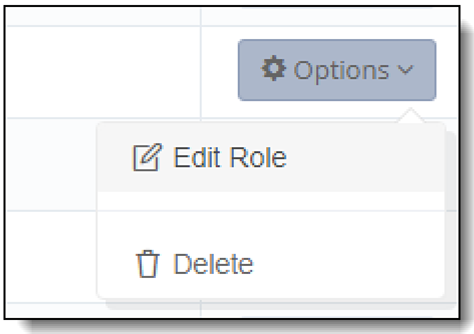
1. Click **Admin Options** at the top right.



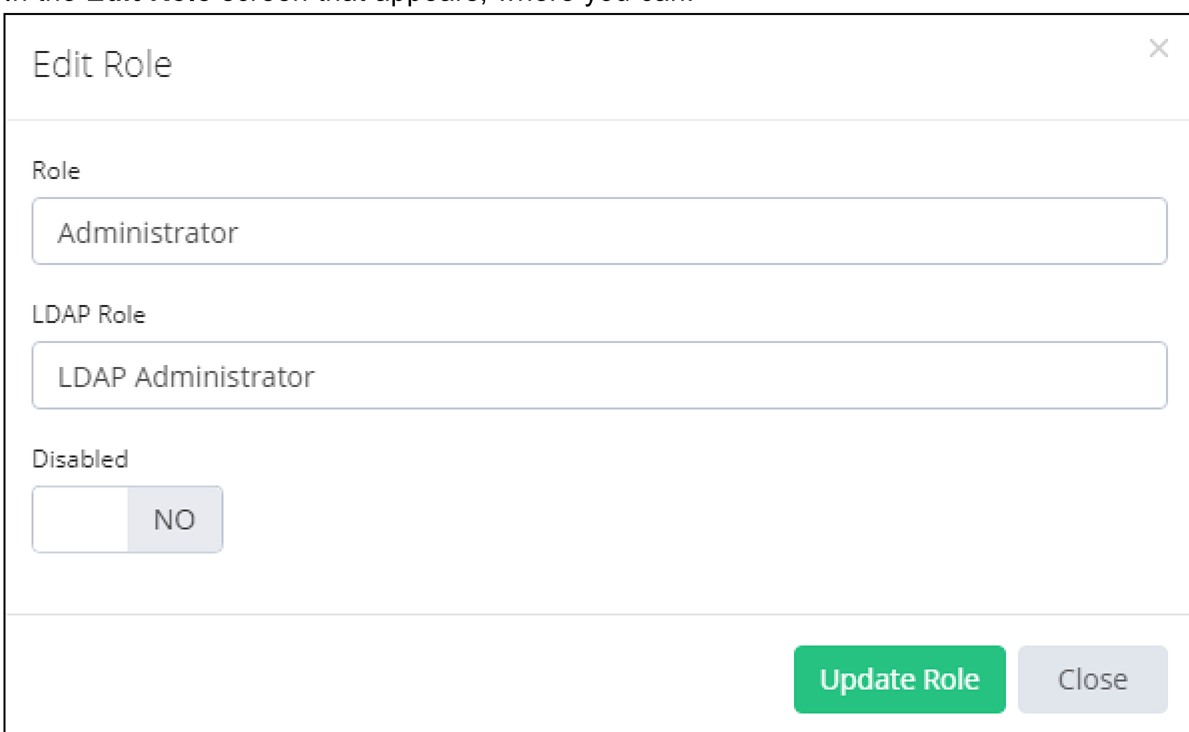
2. Click **Roles** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Role you want to **Disable** then click **Edit Role** Button.



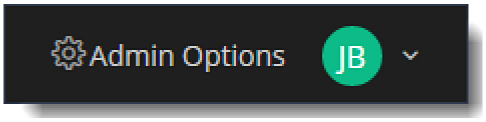
4. In the **Edit Role** screen that appears, where you can:



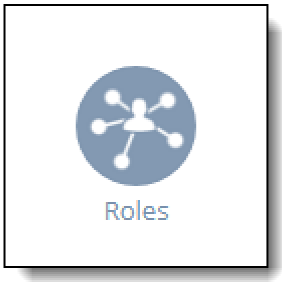
   a. Click **No** under the **Disabled** caption. It will be changed to **Yes**.
   b. Click **Update Role** button.

Deleting the Role:

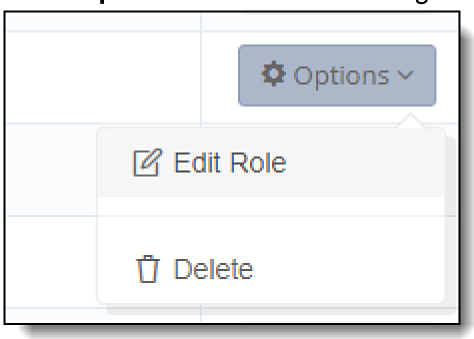
1. Click **Admin Options** at the top right.



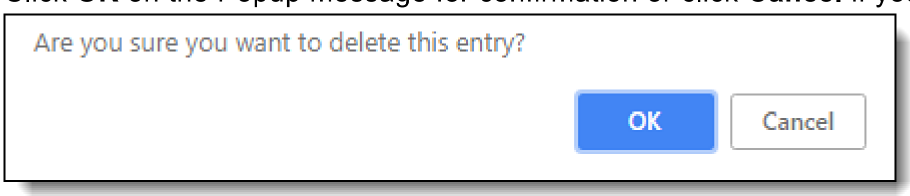
2. Click **Roles** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Role you want to **Delete** then click **Delete** Button.



4. Click **OK** on the Popup message for confirmation or click **Cancel** if you do not want to delete.



11.2.4. Employee Types

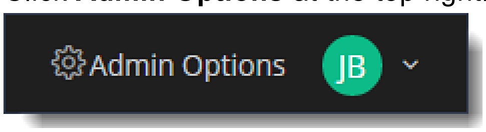
To provide an employee or other person access to TimeLive, you must add them as an Employee in TimeLive. Typically an employee record should be created in TimeLive to allow that employee to enter perform their daily operations through timelive including task management, time entry, expense entry, project management etc.

The system administrator can set up task type using **Admin Options > Employee Types**

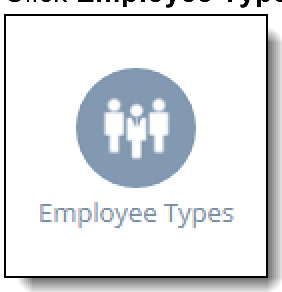
Adding New Employee Type:

By Default, there are 7 types of Employee Types already exist in the application. An administrator can delete/change it as per the company requirement. An administrator can add new Employee Type based on their own organization requirements.

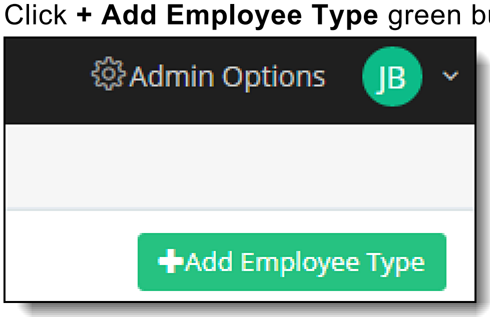
1. Click **Admin Options** at the top right.



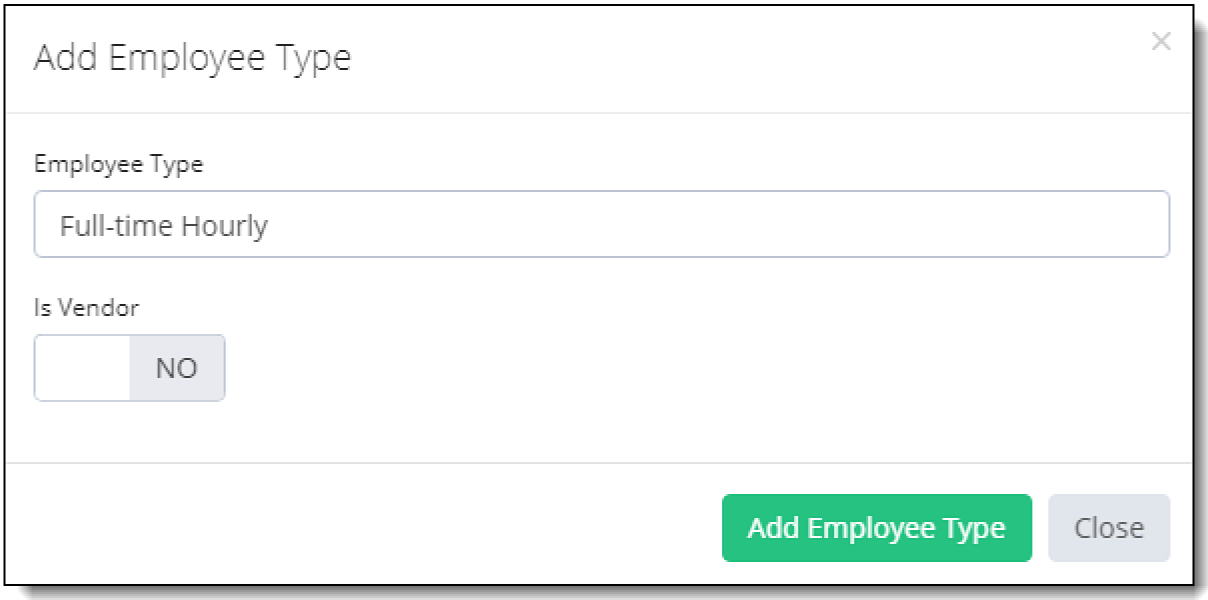
2. Click **Employee Types** icon under **Organization Setup**.



3. Click **+ Add Employee Type** green button at the top right.



4. In the **Add Employee Type** screen that appears, enter:

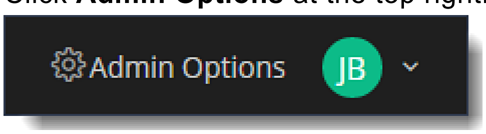


   a. The name of your new Employee Type. Don't worry, you can always change this later.
   b. Click **Is Vendor** if you are adding this for Quickbooks.
   c. Now click the green **Add Employee Type** Button.

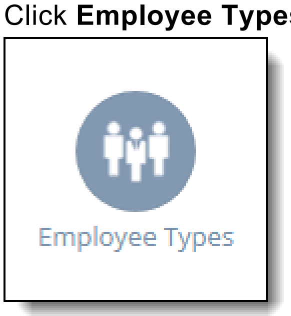
Your new Employee Type will be created and then immediately appear on your **Employee Types** grid.

Editing/Changing Name of the Employee Type:

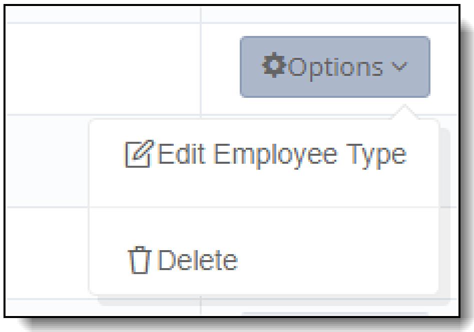
1. Click **Admin Options** at the top right.



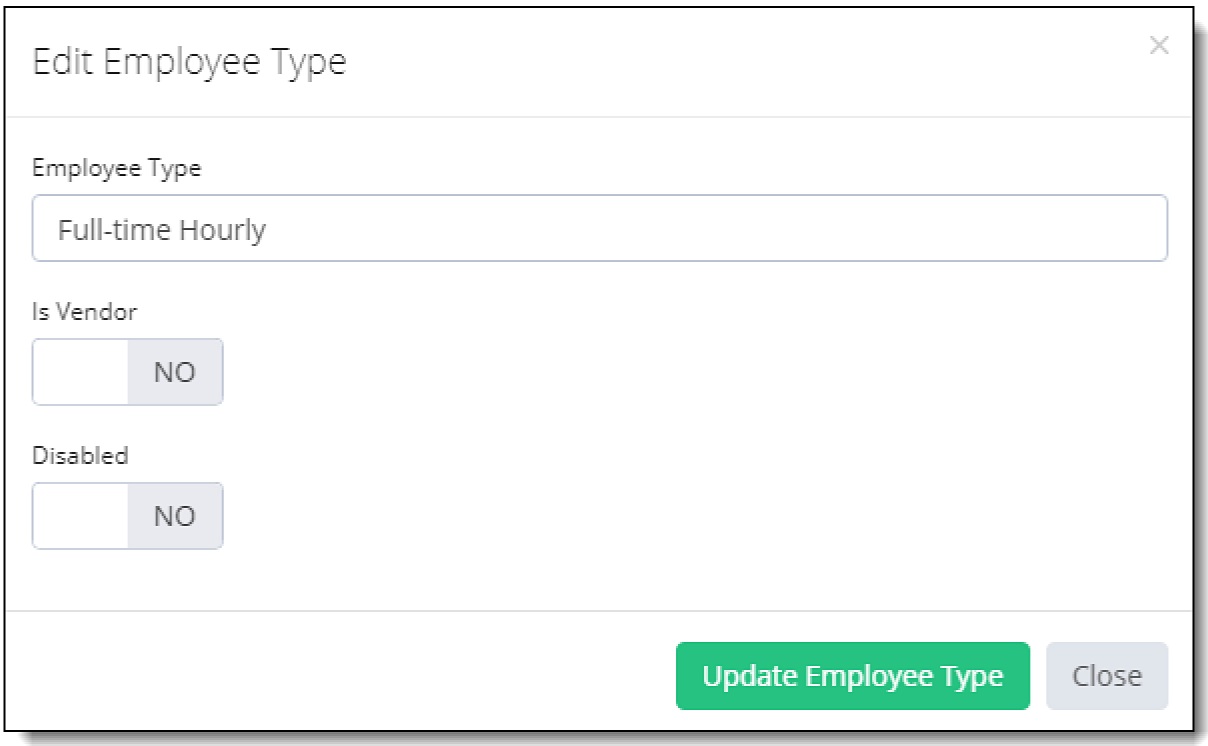
2. Click **Employee Types** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the Employee Type you want to **Edit/change name** then click **Edit Employee Type** Button.



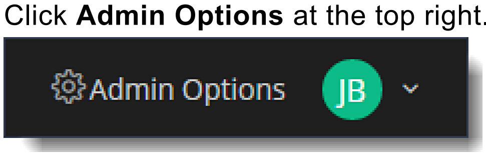
4. Employee Type Popup will appear, where you can change the Employee Type Name.



   a. After making changes, click **Update Employee Type** Button.

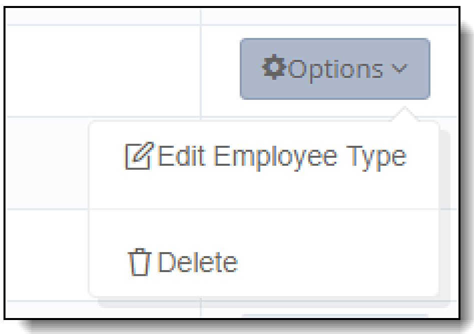
Disabling Existing Employee Type:

1. Click **Admin Options** at the top right.

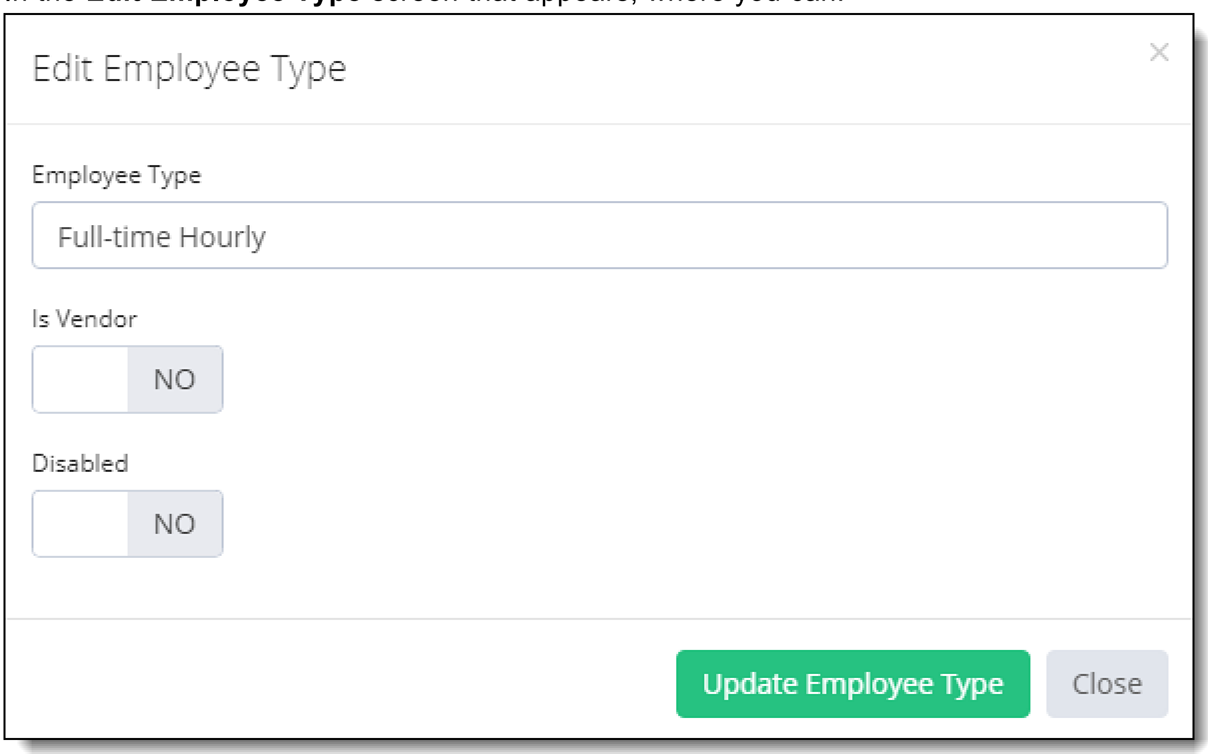


2. Click **Employee Types** icon under **Organization Setup**.

3. Click **Options** with **Gear Icon** against the Employee Type you want to **Edit/change name** then click **Edit Employee Type** Button.



4. In the **Edit Employee Type** screen that appears, where you can:

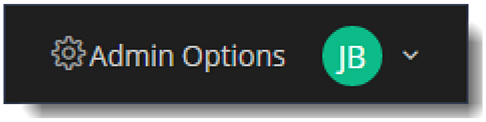


   a. Click **No** under the **Disabled** caption. It will be changed to **Yes**.

b. Click **Update Employee Type** button.

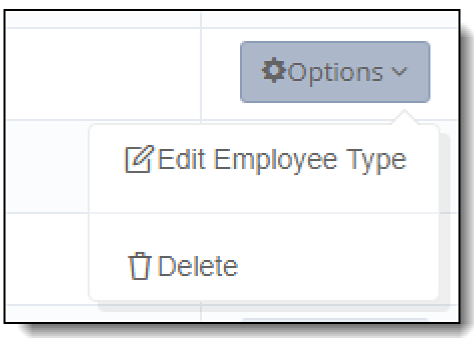
Deleting the Role:

1. Click **Admin Options** at the top right.

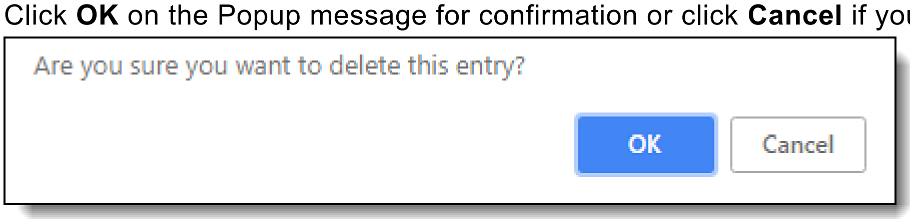


2. Click **Employee Types** icon under **Organization Setup**.

3. Click **Options** with **Gear Icon** against the Employee Type you want to **Delete** then click **Delete** Button.



4. Click **OK** on the Popup message for confirmation or click **Cancel** if you do not want to delete.



11.2.5. External Users

External users are typically external to the organization and, as a result, are much more limited in their access. External users are often created for those users who are required to approve time information and review report data on behalf of a Client.

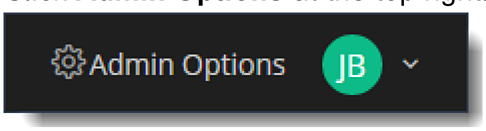
External users can be assigned as a representative/approver for projects associated with the client. Client approvers can approve timesheets in which time is entered into the client's projects (client approver sees only the entries of their projects).

System administrator can setup **[External Users]** through **[Admin Options] > [External Users]**

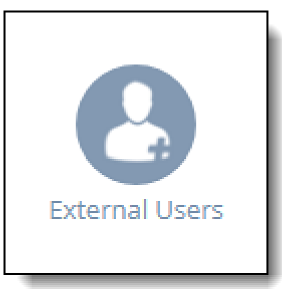
Adding New External User:

An administrator can add new External User through below steps,

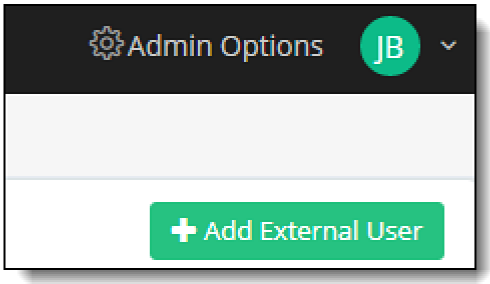
1. Click **Admin Options** at the top right.



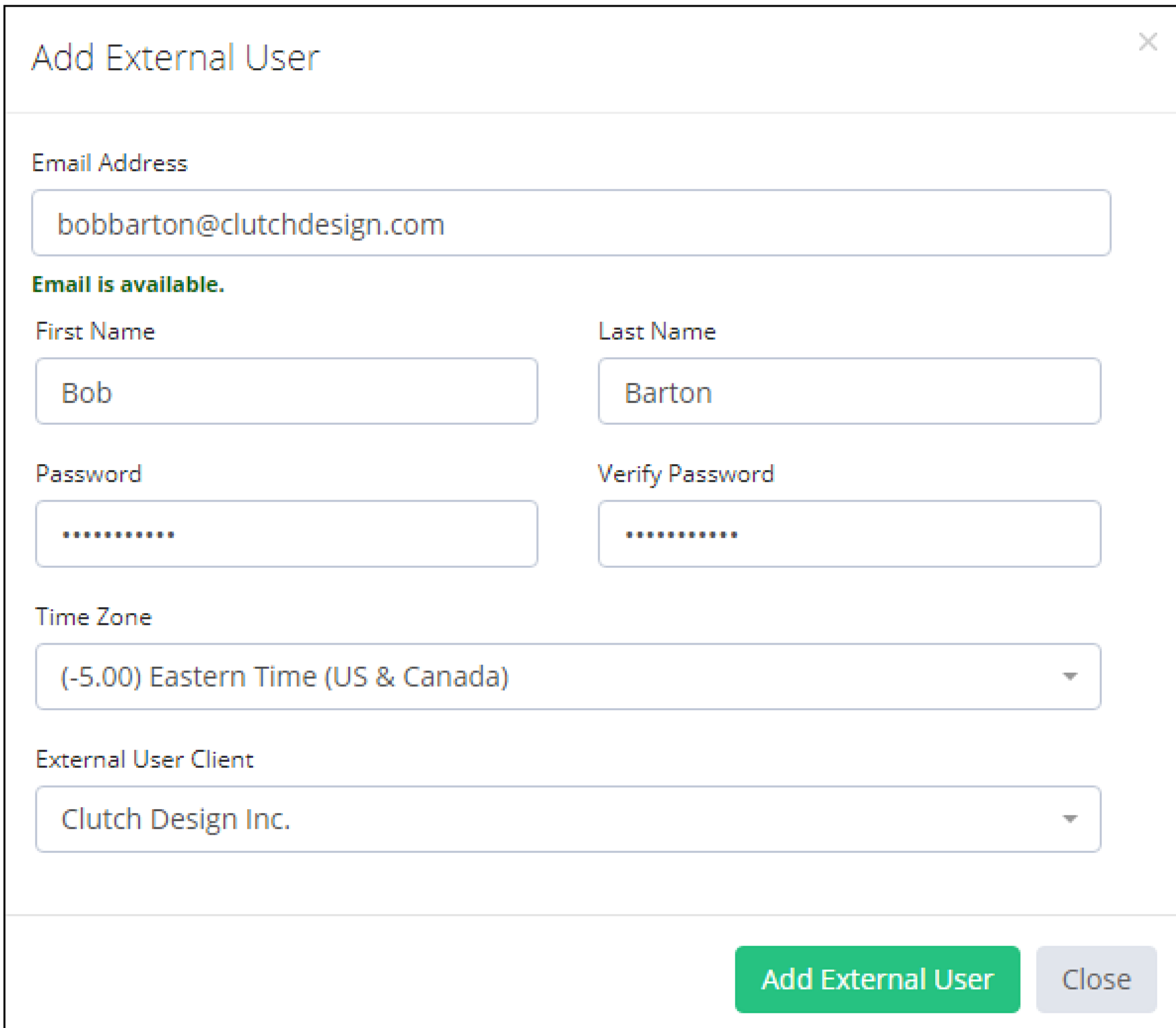
2. Click **External Users** icon under **Organization Setup**.



3. Click **+ Add External User** green button at the top right.



4. In the **Add External User** screen that appears, enter:

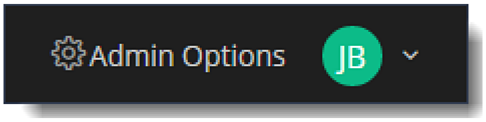


   a. Enter the name of your new External User's Email ID.
   b. First Name and Last Name
   c. Set Password and verify the same in the next box
   d. Select the Time Zone from the Drop-Down list

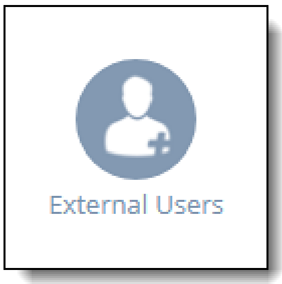
e. Select the Client from the Client Drop-Down of which this External User represent
f. Now click the green **Add External User** Button.

Editing External User Detail:

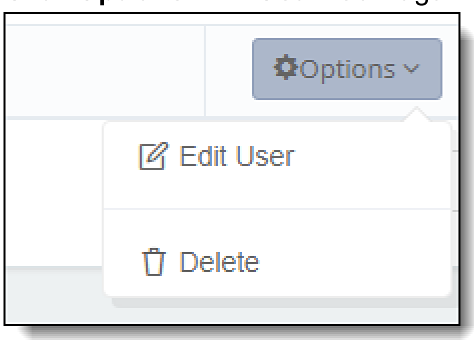
1. Click **Admin Options** at the top right.



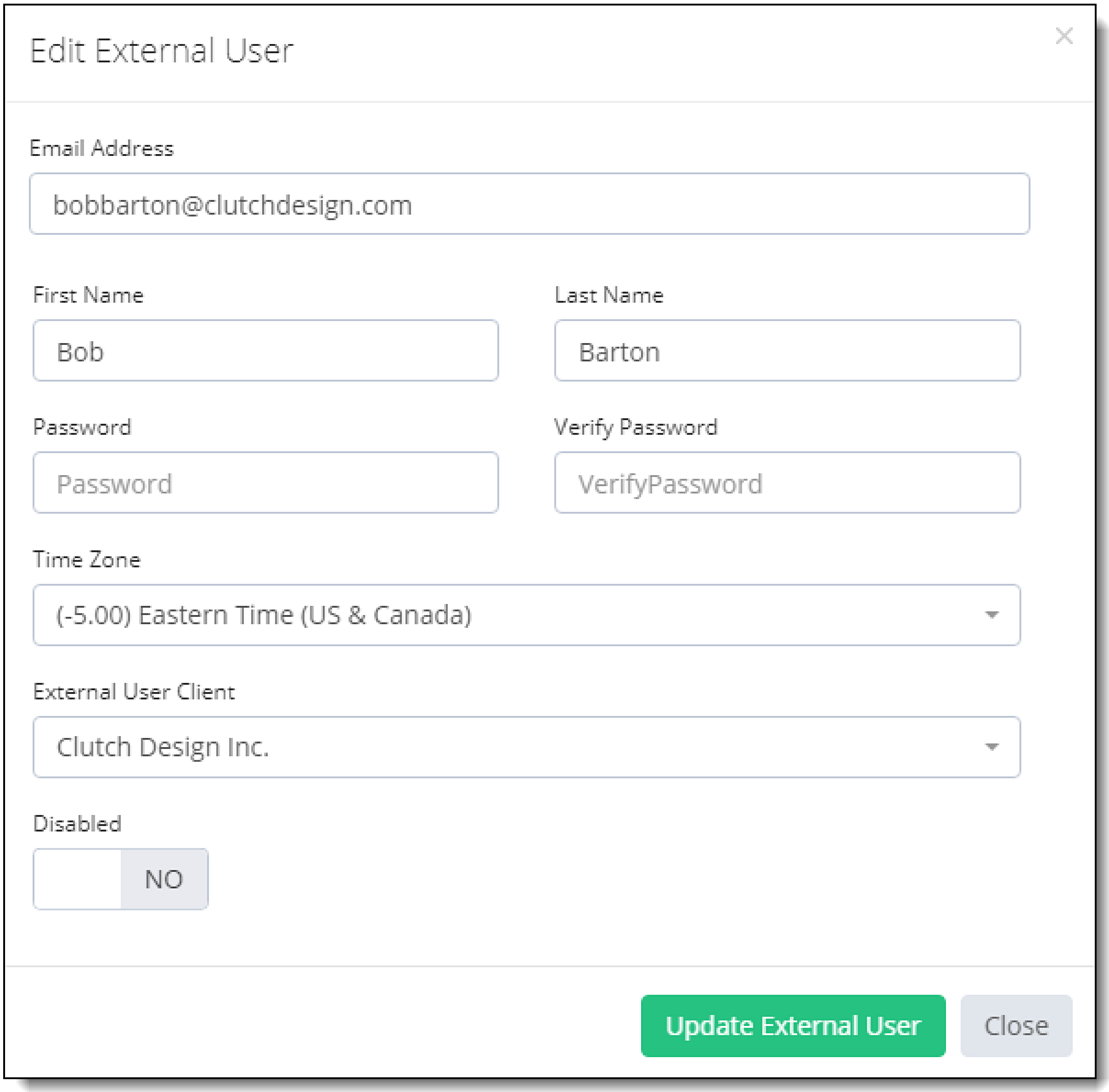
2. Click **External User** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the External User you want to **Edit** then click **Edit User** Button.



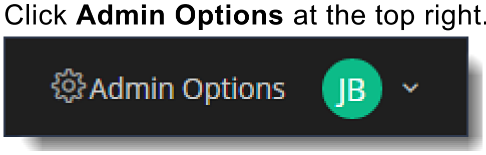
4. External User Popup will appear, where you can change the External User detail.



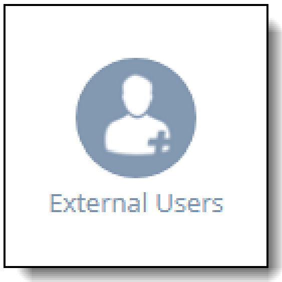
   a. After making changes, click **Update External User** Button.

Disabling Existing External User:

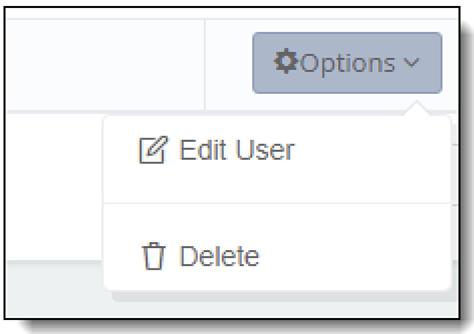
1. Click **Admin Options** at the top right.



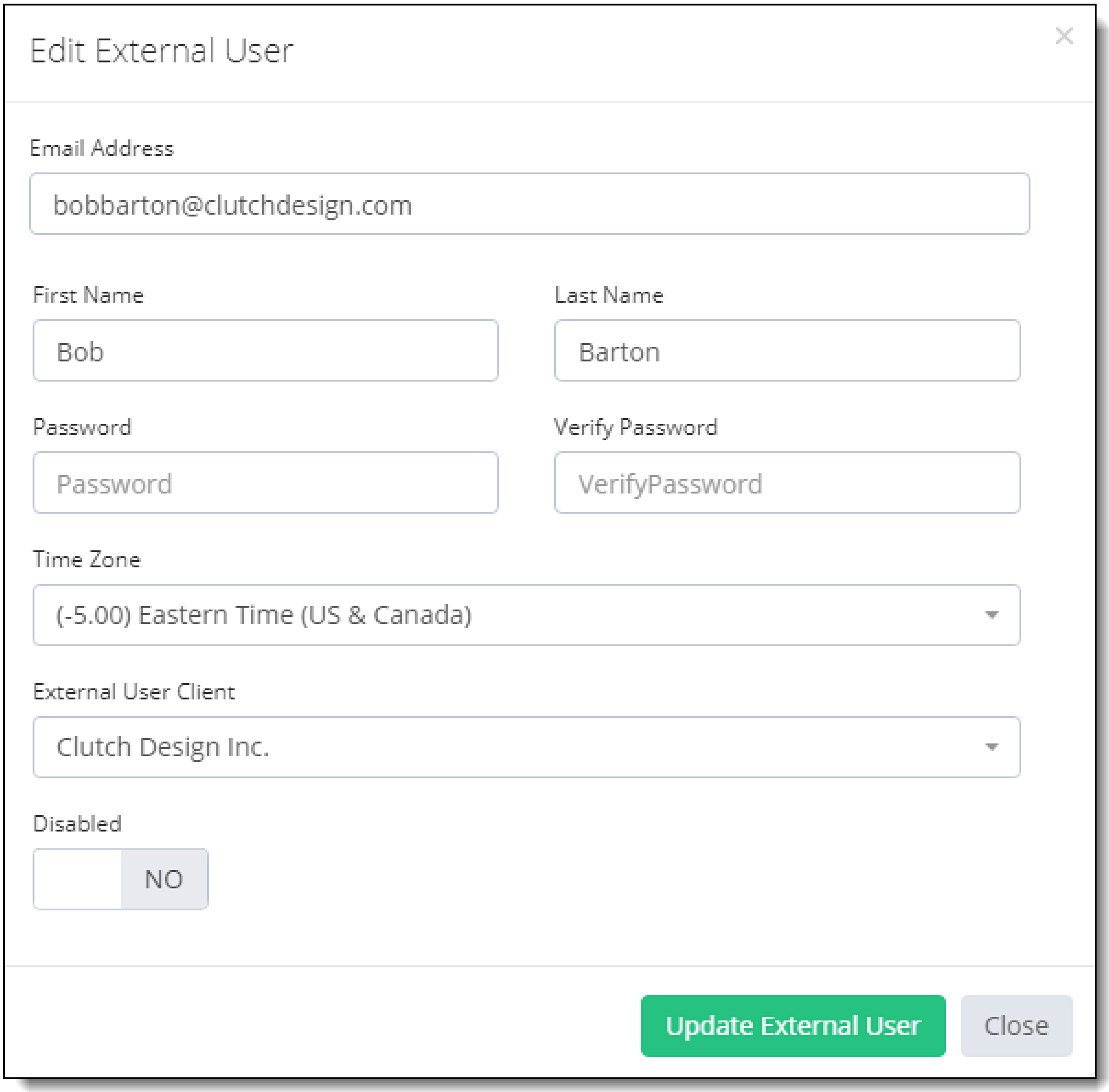
2. Click **External User** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the External User you want to **Disable** then click **Edit User** Button.



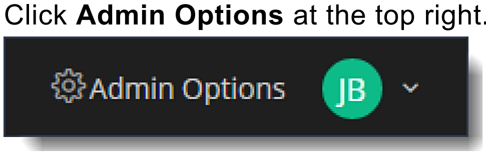
4. In the **Edit External User** screen that appears, where you can:



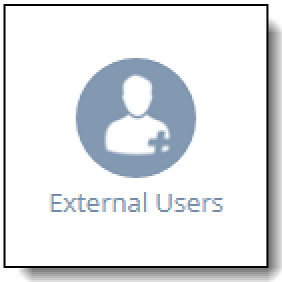
   a. Click **No** under the **Disabled** caption. It will be changed to **Yes**.
   b. Click **Update External User** button.

Deleting the External User:

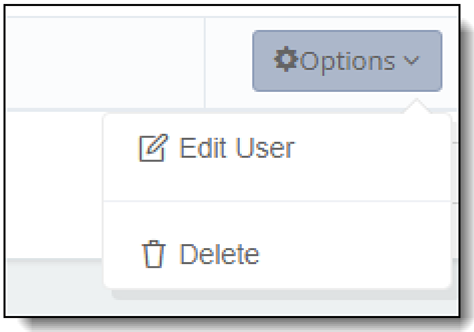
1. Click **Admin Options** at the top right.



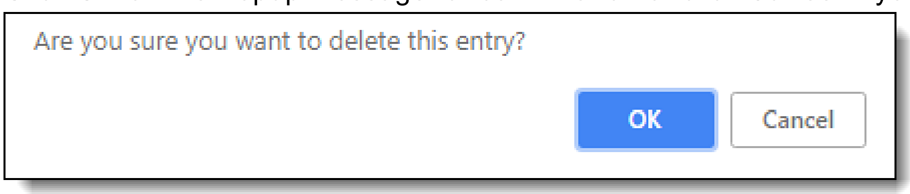
2. Click **External User** icon under **Organization Setup**.



3. Click **Options** with **Gear Icon** against the External User you want to **Delete** then click **Delete** Button.



4. Click **OK** on the Popup message for confirmation or click **Cancel** if you do not want to delete.



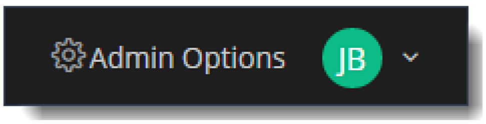
11.3. Application Setup

11.3.1. Email Notification Preferences

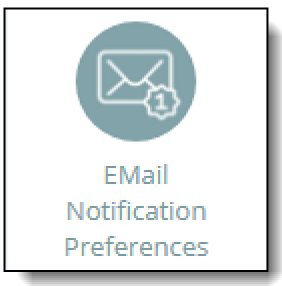
Emails are very important for updates and follow-ups. With it, we do not miss anything important. TimeLive have thought about it in detail and give you many system-generated notifications emails through which employees in organization keep themselves updated all the time.

Accessing Email Notification Preferences

1. Login TimeLive with Admin Credentials.
2. Navigate to Admin Options.



3. Click **Email Notification Preferences** icon under **Application Setup** section.



Enabling/Disabling Notifications Emails:

There are many notifications for users, administrators, Team Leads, Project Managers, and Approvers which Administrator can enable and disable as per company's policy. Some emails are generated on daily basis but their frequency can be set as per requirement.

Some emails are require notifications on Daily basis and some need urgent attention, which are as instant notifications in TimeLive. Enable them as per your convenience.

EMAIL NOTIFICATION PREFERENCES

Update Email Notification Preference

EMail Notification	Enable / Disable	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Attachment Add Notification	✓							
Daily Expense Approval Pending Notification	✓	✓	✓	✓	✓	✓	✓	✓
Daily Time Off Approval Pending Notification	✓	✓	✓	✓	✓	✓	✓	✓
Daily TimeEntry Notification	✓	✓	✓	✓	✓	✓	✓	✓

11.3.2. Role Permissions

Role Permission is used to define the activities and features users can access. Each role has a name and list of actions. The administrator assigns roles to each user when adding a new employee.
Also, an administrator can set a different default page for each role when they login TimeLive. For example, users usually need to enter time so the administrator can select **My Timesheet** as the default page. further, Managers have to view and approve Time Entries, Time Offs, and Expenses of their employees so the administrator can select **Manage Approvals** as their default page.

IMPORTANT NOTE:

- Role permissions allow an administrator to restrict or expand an employee’s access to TimeLive system. Consider each role carefully before assigning actions to that role.
- Some system-defined roles are not visible when trying to assign them to employees. Please refer to **TimeLive FAQ > System Defined Roles** page for reference.

Setting up Role Permissions
TimeLive provides 7 pre-defined roles which can be customized through role permission. Also, an administrator can create their own custom roles through **Admin Options > Roles** and can assign their own set of permissions.

Pre-defined roles:

- Administrator
- Project Manager
- Time Entry Approver
- Expense Entry Approver
- Team Lead
- External User
- User

Each TimeLive functionality can be customize using 4 level of access permissions:

- List / View (give access to view the grid)
- Add (allow this role to add new item)
- Edit (allow this role to edit existing item)

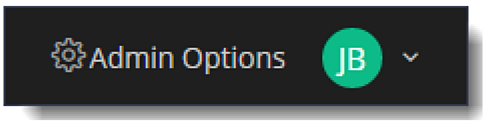
- Delete (allow this role to delete the existing item)

11.3.3. Manage Terminology

Manage Terminology option now allows TimeLive Administrator to change Timesheet default jargons to any of their own custom terminologies. An administrator can define which “TimeLive” jargons they want to modify with their own required name in **Admin Options > Manage Terminology**. After the update, TimeLive will then use those changed terminologies everywhere in the application including menus, timesheet, forms, reports etc.

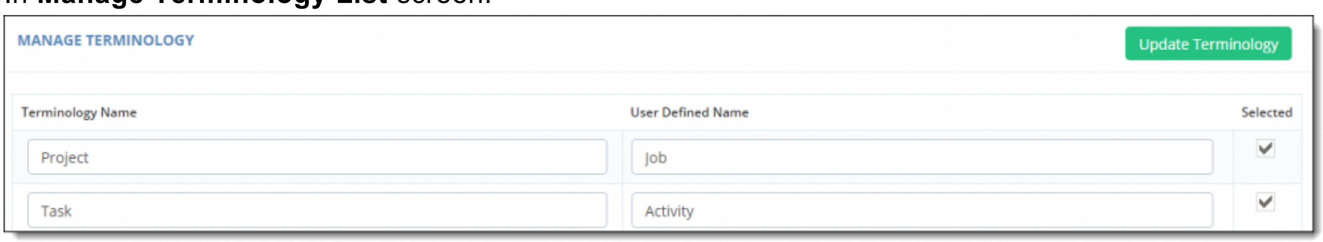
Defining new Manage Terminology:

1. Navigate to **Admin Options**.



2. Click **Manage Terminology**.

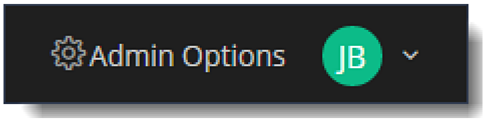
3. In **Manage Terminology List** screen:



1. Enter TimeLive built-in term in the **Terminology Name** field.
2. Enter your own custom name in **User Defined Name** which you want to show in place of **TimeLive built-in name**.
3. Click on the **Selected** checkbox in the same row where you have defined this record.
4. Click on the **Update Terminology** green button on the top right of the page to update this terminology.

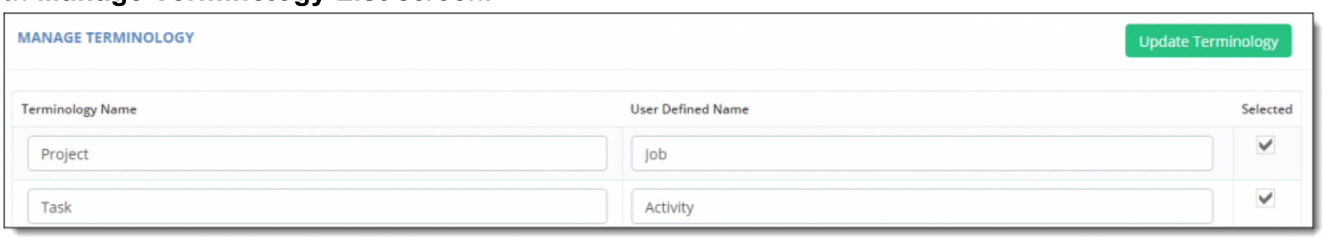
Switching back to the original terminology:

1. Navigate to **Admin Options**.



2. Click **Manage Terminology**.

3. In **Manage Terminology List** screen:



4. Uncheck on the **Selected** checkbox in the same row where you want to revert to the original terminology.
5. Click on the **Update Terminology** green button on the top right of the page to update this terminology.

11.3.4. Preferences

Preferences option allow Administrator to setup different application-level parameters which are required for their organization. **Admin Options > Preferences**

TimeLive preferences section has nine Sub-sections.

Organization:

Under **Organization**, Account Information is saved. The administrator can change the data according to the organizational needs along with Time Zone.

General:

Options under General Preferences plays a pivotal role in setting up account basic settings. Details are as below for each option,

Option	Description
Lock Submitted Records	This option lock any submitted records in Timesheet, Time Off, and Expense. A user cannot edit submitted records if this check is applied
Lock In-Approval/ Approved Records	If this check is applied, all records which are waiting for approval or already approved cannot be edited by user
Use Electronic Signature	If this check is applied, a user can see his/her Electronic signature on approvals and print paper
Show Additional Department Information in Employee	If this check is applied, Department Code will be included in Department column under Employee Grid
Show Employee Name with Code	If this check is applied, All list of employees drop-down will show Employee code along with Employee Name
Show Disabled Employees In Report	If this check is applied, All report will show disabled employees' list and their records, according to the report
Scheduled Email Sent Time	Through this field, Administrator can schedule email sending time. It could be **09:00** or **16:00**
Session Out Time	Due to security risk, Administrator can set minutes after TimeLive session may expire due to inactivity
Page Size	This drop-down, allow a user to view the number of Records on one page. (**10,20,30,40,50**)

From Email Display Name	This field allows the administrator to change Automatic email sender name according to organization need
From Email Address	This field allows the administrator to change sender default email address according to its organization
Standard and Formats	This drop-down option is crucial as it controls Time, Date and Text format (By default, it is <*Auto*> which automatically pick culture info from client browser default settings)
Show Employee Name By	This drop-down allows the administrator to use either First Name and Last Name or Last Name and First Name in TimeLive
Company Own Logo & Show Company Own Logo	Through this option, Administrator can upload Company logo to be shown in TimeLive Application and on Prints as well

Invoice Setup:

In this tab, Options related to Invoice can be modified,

Show Project Name In Invoice Report	Through this check, Project Name will be shown in the generated Invoice
Show Billing Rate In Invoice Report	Through this check, Billing Rate of the Project will be shown
Show Entry Date In Invoice Report	If this check is applied, Entry date will be shown on the Invoice
Show Employee Name In Invoice Report	If this check is applied, Employee name will reflect in the generated Invoice
Show Work Type In Invoice Description	Through this option, Work Type will reflect in Invoice Description
Round-Off Tax Value In Invoice	Through this option, Tax Value will be Round Off in the generated Invoice
Invoice No Prefix	With this option, Administrator can set Invoice No. Prefix
Invoice Billing Type	Through this Drop-Down, Administrator can select between Hours or Days Billing Type
Invoice Footer	In this TextBox, Administrator can input Footer for the generated Invoice. It would be general as it will reflect in all invoices

Task Setup:

Under Task Setup tab, Administrator can set some options which will reflect in Task Grid,

Show Project Name In Task	By enabling this check, Project Name will be shown in Task Grid
Show Client Name In Task	By enabling this check, Client Name will be shown in Task Grid
Sort Task By	Through this Drop-Down, Administrator has the option to sort Task Grid according to Deadline Date or Task ID

Project Setup:

Under Project Setup tab, Administrator can set some options which will reflect in Project Grid,

Auto Generate Project Code	Through this check, Project Code will be generated by System
Show Completed Project In Project List	If this check is enabled, Completed projects can be shown in Project Grid
Show Client Department In Project	By enabling this check, Client Department will appear in Project Grid
Include Current Year In Project Code	By enabling this check, TimeLive Application will include current year in Project Code. i.e. *2016*-XXX
Automatically Include Administrator In Project Team	If this check is enabled, Administrator will be automatically be added to every Project during creation. Note: Projects which are already created prior to this check enabled, will not make administrator as team member
Show Disable Project In Report	Through this check, All Disabled Projects will be shown in Reports and its Drop-Down
Project Template Mandatory	If this check is enabled, Administrator/User must select Project Template during Project Creation (See Project Template)
Project Code Prefix	Through this check, Administrator can assign Project Code Prefix for Project

Time Off Setup:

Under Time Off Tab, Administrator can manage few settings of Time Off,

Show TimeOff in Timesheet	Through this check, Time Off will be available in My Timesheet for users to input their Time Off
Show Project for TimeOff	Through this check, Project selection will be possible while submitting Time Off Request

Show Time Off in Days	Through this check, Time Off will be calculated in days with reference to hours set in Working Days
TimeOff Status Edit Mode	If this check is enabled, a user can edit their submitted TimeOff data from Time Off Request Page
Show Timeoff Types By	This drop-down allows an administrator to select between Account wise TimeOff type or Employee Wise

Expense Setup:

Under Expense Setup Tab, Administrator can enable some additional data fields for Expense Sheet,

Show Client In Expense Sheet	If this check is enabled, a user can select Client while creating new Expense Sheet
Show Task in Expense Sheet	If this check is enabled, a user can select Task for which they want to have Expense Sheet
Expense Approval Type By	Administrator can choose between **Project wise** or **Employee wise** that how Expense Approval shall be executed. Project is preferred
Expense Sheet Print Footer	This Footer will be visible on every Expense Sheet Print. Administrator can change its data according to organizational need

Timesheet Setup:

Under this option, Administrator can set many functions which TimeLive is offering which will reflect in My Timesheet of all users,

Show Clock Start/End	If this check is enabled, TimeLive allow users to input Start Time and End Time in their Time Entry. TimeLive will calculate Hours Spent itself
Show Client in Timesheet	Through this check, Client drop-down will be available for users to select prior to project and task
Show Description in Week View	Normally, Description is in Day View, but if this check is enabled, Description box will appear after entering End Time/Total Time
Show Completed Project in Timesheet	In TimeLive, Completed Projects will not appear in Project Drop-Down for selection but this check allows those project to reflect in the drop-down
Show Work Type in Timesheet	This feature is very useful. If a user has worked overtime or on Holidays, and have different rates for that, they can select Work type from My Timesheet while entering their Hours Spent
Show Cost Center in Timesheet	Through this check, Cost Center drop-down will be available for users to select

Show Copy Timesheet Button	This option allows Copy Timesheet Button to be visible in users timesheet (View My Timesheet)
Show Copy Activities Button In Timesheet	This option allows Copy Activities Button to be visible in users timesheet (View My Timesheet)
Enable Offline Timesheet	This option allow Offline Timesheet Button to be visible in users timesheet (View My Timesheet)
Show "Show All" Check in Approval	While in Approval, if the detail is clicked, Approver can see records which are pertaining to him/her, but if this check is enabled, Approver can view the whole week entry including TimeOffs as well. This will help Approver in monitoring users activities during the week
Show Task Percentage in Timesheet	This option allows users to input Percentage along with their time spent on that task. The percentage is linked directly to the task
Calculate Task percentage Automatically	This option automatically calculates task percentage
Auto Adjust Timesheet	This option will automatically adjust My Timesheet Layout according to enable fields and their width
Number of Blank Rows in Timesheet	This option allows the administrator to control the number of lines for TimeEntry. By default, it is 2. When both lines are filled and records are saved, 2 extra blank lines will appear for TimeEntry
Timesheet Period Overdue	Through this option, Administrator can set number of Overdue period reporting on Overdue email notification (See Email Notifications)
Show Clock Start/ End by	This drop-down allows the administrator to choose between Account/Employee
TimeEntry Hours Format	This drop-down option allows the administrator to select between Time format (HH: MM) or Decimal format (1.5 ie. 1 Hour 30 Mins. or 5.75 ie. 5 Hours 45 Mins.)
TimeEntry Clock Format	This drop-down option allows the administrator to select between **24 Hour format** (**HH: MM**) or 12 Hour format (**HH: MM AM/PM**)
Show additional Task Information Type in Timesheet	This drop-down option allows the administrator to select between some additional task information to be reflected in My Timesheet (**Parent Task Name, Parent Task Code, Task Code or Task Type**) Administrator can select any one.
Show additional Project Information Type in Timesheet	This drop-down option allows the administrator to select between some additional Project information to be reflected in My Timesheet (**Project Code**)
Default Time Entry Mode	This drop-down option allows Administrator to select between Day-View or Period View for default My Timesheet View.
Sort Timesheet By	This drop-down option allows the administrator to select sorting of My Timesheet according to Client or by Default (**Time Entry**)

Show Cost Center In Timesheet by	This drop-down option allows the administrator to select between **Account** or **Employee** to show their Cost CenterCenter
Timesheet Print Footer	This Footer will be visible in every Timesheet Print. The administrator can change its data according to organizational need.

Security:

Under this tab, Administrator can Enable Single Sign-On (**SSO**). This feature allows users to log in with their **ADFS/SAML ID** (for Details view **ADFS Integration**).

Enable Password Complexity	This option restricts users to input much secure password than simple ones. Password must consist of 6 digits including 1 numeric and 1 special characters if this check is applied
Password Expiry Period	Administrator can set the expiry period of employees password. After the expiry period, a user must need to change their existing password

11.3.5. Status Types

Status types are different types of status codes which TimeLive users can use to monitor their different types of project or task states from start to finish.

Different status type can be **Open**, **In Progress**, **Closed**. An administrator can add more types as per the requirement.

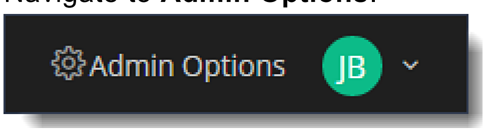
There are three status types which you can configure in Status Types setup:

1. Task (Status code of project tasks)
2. Project (Status code of projects)
3. Employee (Status code of employee)

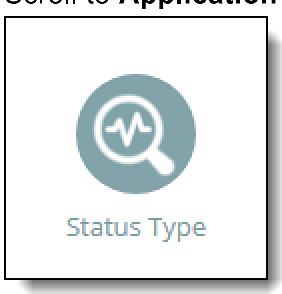
A system administrator can setup Status using **Admin Options> Status Type**

Adding new Status:

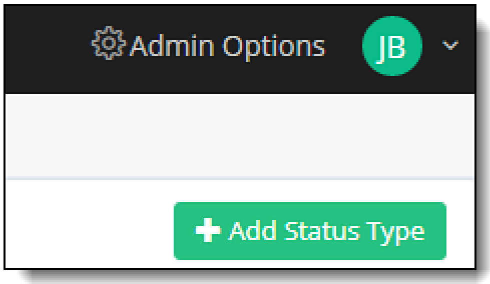
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options**.



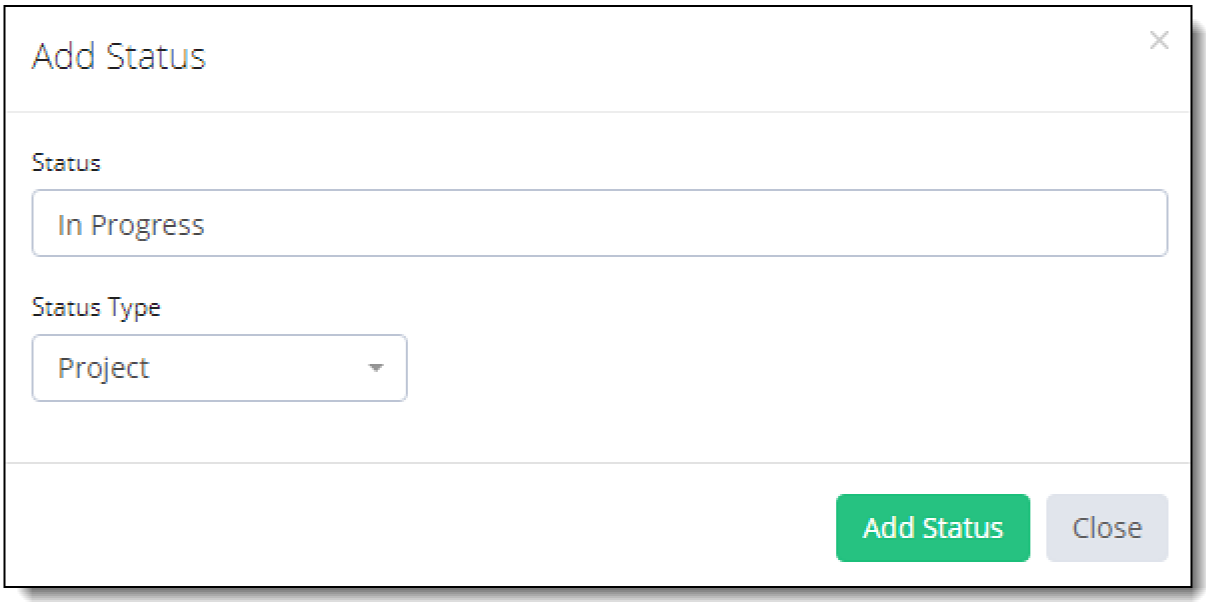
3. Scroll to **Application Setup** and click **Status Type**.



4. Click on the **+ Add Status Type** green button on the top-right side of the application.



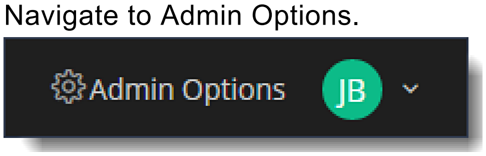
5. In the **Add Location** screen that appears, enter:



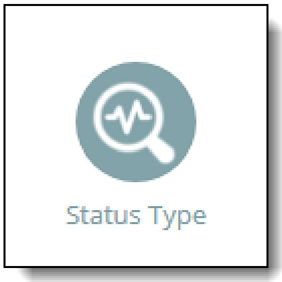
   a. Fill the first field with Status Name you want to give.
   b. Select status type (select Task, if you are going to add status of a task. Select Project if you want to add a status type for the project. Select Employee, if you want to add new status for employees).
   c. Click on **Add Status** button to add this status in TimeLive system.

Editing the existing Status type:

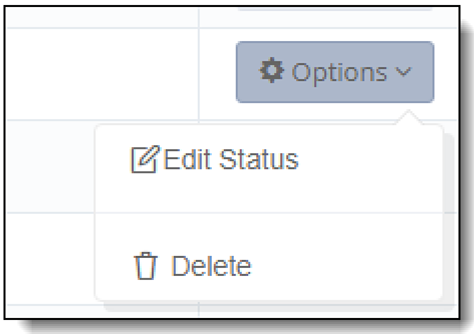
1. Login TimeLive with Admin Credentials.
2. Navigate to Admin Options.



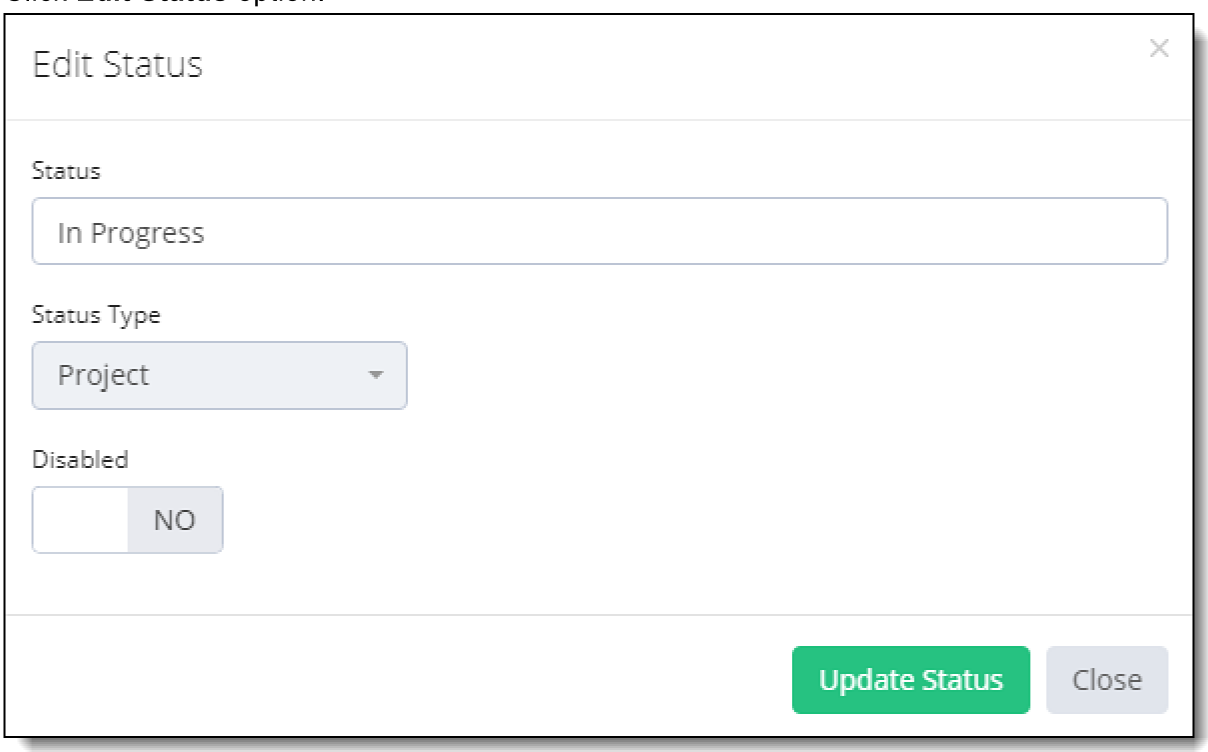
3. Scroll to **Application Setup** and click **Status Type**.



4. In **[Status List]**, click on **Options** gear icon of the type which you want to modify.

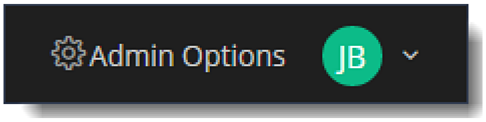


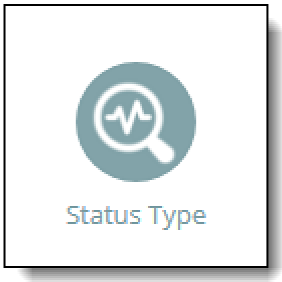
5. Click **Edit Status** option.

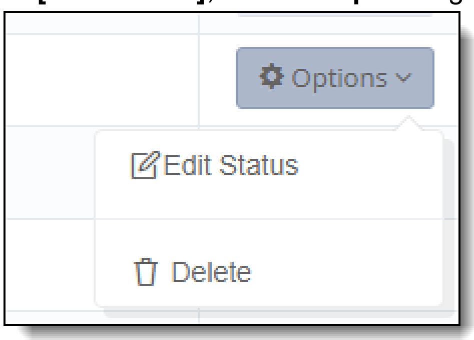


   a. Update your required field in the popup form.
   b. Click **Update Status** button to update this record.

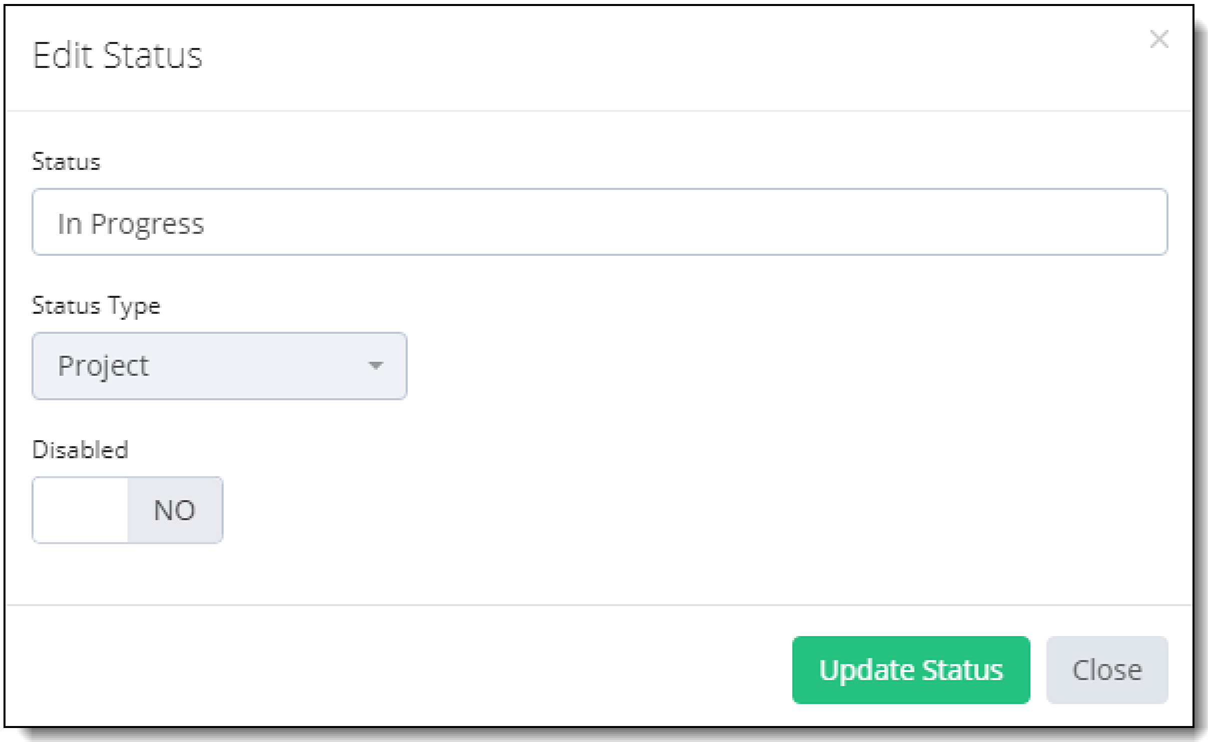
Disabling the existing Status type:

1. Login TimeLive with Admin Credentials.
2. Navigate to Admin Options.


3. Scroll to **Application Setup** and click **Status Type**.


4. In **[Status List]**, click on **Options** gear icon of the type which you want to disable.


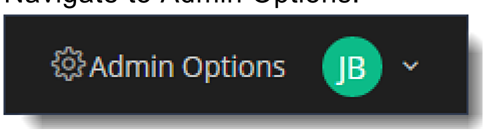
5. Click **Edit Status** option.



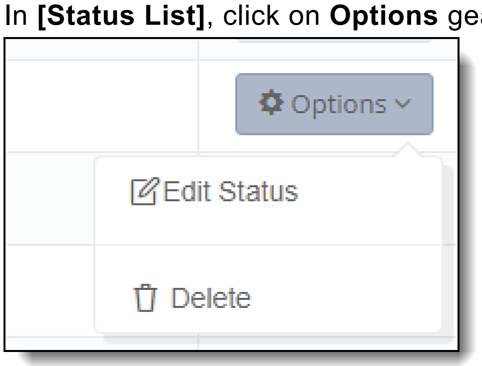
   a. Click on the **Disabled** switch and it should show **Yes**.
   b. Click **Update Status** button to update this record.

Deleting the existing Status type:

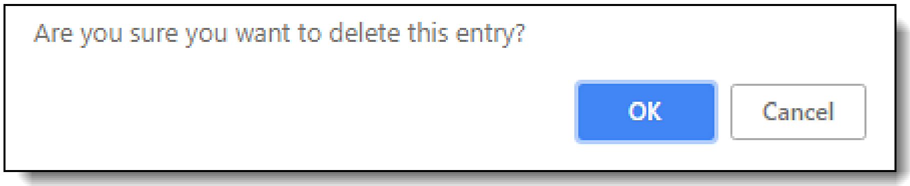
1. Login TimeLive with Admin Credentials
2. Navigate to Admin Options.



3. Scroll to **Application Setup** and click **Status Type**.
4. In **[Status List]**, click on **Options** gear icon of the type which you want to delete.



5. Click **Delete** option and click **OK** on the confirmation popup.



11.3.6. Custom Fields

Different organizations have different requirements which are specific to their own. It helps them to keep their records according to their needs.

TimeLive understands that and to cater their specific needs, Custom Field plays a key role in achieving this.

Navigate to **Admin Options > Custom Fields**.

The following picture shows a list of **Entities** under which Custom Fields can be created.

Through **Options > Manage Custom Field** link against each entity, Administrator can create custom fields according to organizational needs. At the moment, there are 15 custom fields can be created in each entity.

There are 4 Data Type through which, Custom Field can be created.

1. **Number**
2. **Text**
3. **Drop-Down**
4. **Date**

In below screenshot, 1 custom field “**Date of Birth**” is created. which is under Employee Entity. **IsRequired** check is a validation if this field is mandatory or not. Further, a new custom field is in the creation process. Data Type is selected as Drop-Down and list of options will be visible under employee record for custom field selection.

Creating a new Custom Field:

1. Login TimeLive with Admin Credentials
2. Navigate to **Admin Options > Custom Field**
3. Click on the **Options** gear icon against the entity in which you want to create a new custom field.
4. Click **Manage Custom Field**
5. Click on the “+ Add Custom Field* green button on the top-left side of the application
    a. Fill the first field with Status Name you want to give
    b. Select the Data Type
    c. Fill the Custom Field Name as per your requirement
    d. Fill the Custom Field Caption as per your requirement

   e. Fill the Default Value, Minimum Value, and Maximum Value if required
   f. Click on the **IsRequired** switch to **Yes** if you want to make it mandatory for users to fill it
6. Click **Add Custom Field** green button

Editing an existing Custom Field:

1. Login TimeLive with Admin Credentials
2. Navigate to **Admin Options > Custom Field**
3. Click on the **Options** gear icon against the entity in which you want to edit the existing custom field.
4. Click **Manage Custom Field**
5. Scroll to the Custom Field you want to edit and click on **Options** gear icon then select **Edit Custom Field**
6. Make the necessary changes in the fields
7. Click **Update Custom Field** Button

Disabling an existing Custom Field:

1. Login TimeLive with Admin Credentials
2. Navigate to **Admin Options > Custom Field**
3. Click on the **Options** gear icon against the entity in which you want to disable the existing custom field.
4. Click **Manage Custom Field**
5. Scroll to the Custom Field you want to disable and click on **Options** gear icon then select **Edit Custom Field**
6. Change the **Disabled** switch to **Yes**
7. Click **Update Custom Field** Button

Deleting an existing Custom Field:

1. Login TimeLive with Admin Credentials
2. Navigate to **Admin Options > Custom Field**
3. Click on the **Options** gear icon against the entity in which you want to delete the existing custom field.
4. Click **Manage Custom Field**
5. Scroll to the Custom Field you want to delete and click on **Options** gear icon
6. Click on the **Delete** option
7. Click **Yes** to confirm deletion

11.3.7. Approvals

The approval process ensures that time is entered correctly for more efficient and accurate project and Time Off analysis. TimeLive provides a default approval process that you may customize for your organization. In any organization, Time entries, Leaves, expenses are supposed to be approved by the managers or any specific employees. In most cases there could be one or more approvers to approve only one entry totally depending on the organizational needs.

In TimeLive, Administrator can use TimeLive default approval types or can easily create new Approval types of Timesheet, Time Off and Expense as per the requirement. There is no limit of creating such approvals.

By Default, besides "**Approval Not Required**" option, TimeLive offers below Approval Types,

- **Project Manager**
- **Team Lead > Project Manager**
- **Employee Manager**

In TimeLive, Administrator can create a new Approval Path from Admin Options > Approvals with many possibilities,

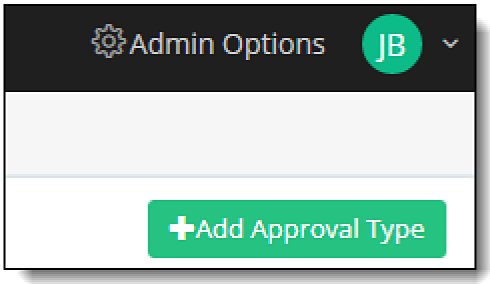
- There are 5 tiers of Approvals in each Approval Path. That means, maximum 5 approvals tier are available for defining the approvals but more than one employee can be set in each tier.
- Administrator can set 2 approvers on the same tier so that any one approves the request and it will be approved.
- Administrator can set more then 1 approver with different tiers so that request will be approved as per the tiers defined.
- Administrator can introduce any specific employee at any tier for approval even the external user.

Creating a new Approval Path:

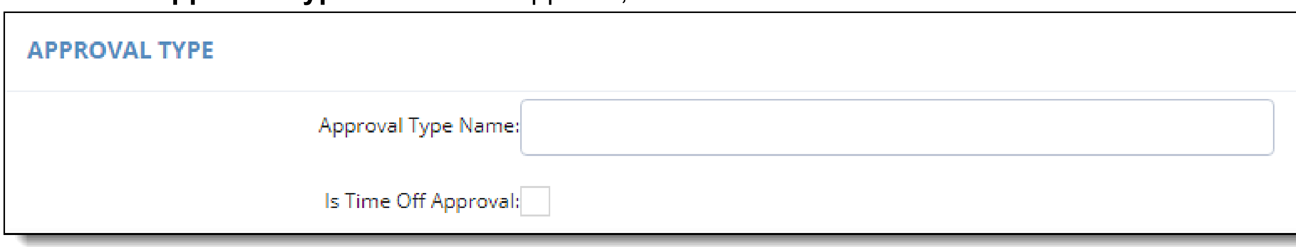
1. Login TimeLive with Admin Credentials.

2. Navigate to **Admin Options** scroll to **Application Setup** and click **Approvals**.

3. Click on the **+ Add Approval Type** green button on the top-right side of the application.



4. In the **Add Approval Type** screen that appears, enter:



   a. Approval Type Name.
   b. Enable the check **"Is Time Off Approval"** if it is for Time Off.

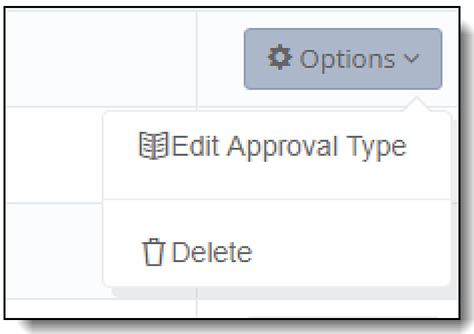
5. In the **Add Approval Path** screen that appears, enter:

   a. Select the **Approver Type** from the first column.
   b. External User column and Employee Column is dependent on the first column selection.
   c. Set the sequence(tier) of Approval Path. If you want it as 1st Project Manager and 2nd Employee Manager. select the first column as Project Manager and set the sequence as 1st then select the Employee Manager in the 2nd row and select the sequence as 2nd.
   d. Click **Add** button at the bottom-right of the of the page.

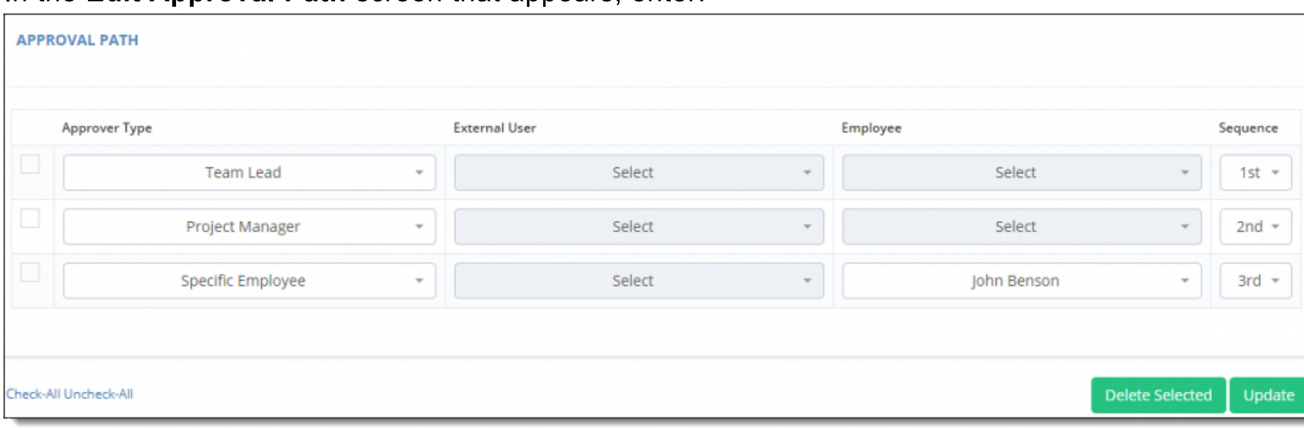
Editing the existing Approval Path:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** scroll to **Application Setup** and click **Approvals**.

3. Click on the **Options** gear icon against the path which needs to be changed.
4. Click the option **Edit Approval Path**.



5. In the **Edit Approval Path** screen that appears, enter:



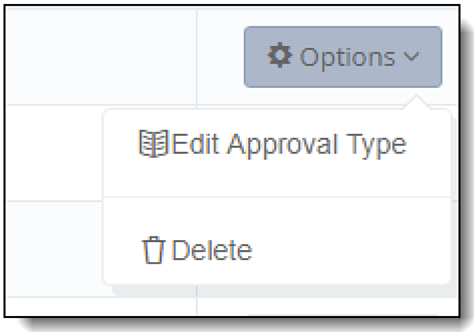
   a. Make the changes on the page as per your requirement and click **Update** button at the bottom of the page.

Deleting the existing Approval Path:

1. Login TimeLive with Admin Credentials.

2. Navigate to **Admin Options** scroll to **Application Setup** and click **Approvals**.

3. Click on the **Options** gear icon against the path which needs to be deleted.



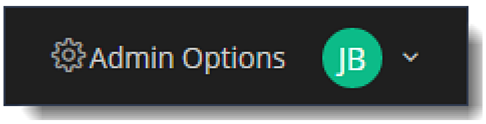
   a. Click **Delete** option.
   b. Click **Ok** for confirmation.

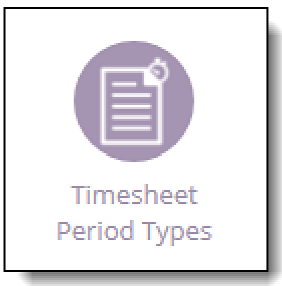
11.4. Timesheet Setup

11.4.1. Timesheet Period Types

Timesheet periods determine the length of user Timesheets. They should parallel either your **organization’s pay periods** or **billing cycle** so that timesheet data is available when needed for generating payroll or for invoicing clients. A **System timesheet** period is assigned to new users by default. An administrator can use different Timesheet Period for different employees, according to different Working Days.

This feature is under **Admin Options > Timesheet Period View**.





Under Timesheet Period Types, there are 5 types of Period view from which 1 can be used in Working Day.

TIMESHEET PERIOD TYPES LIST

	Timesheet Period Type	Initial Days Of The Period	Initial Days Of The Month	
Active	Daily			Options
Active	Weekly			Options
Active	BiWeekly			Options
Active	Semi-Monthly	1st And 16th		Options
Active	Monthly		1	Options

It does sound that Bi-weekly and Semi-monthly are similar but there is a difference.

In **BiWeekly**, TimeLive allows an administrator to change start period date according to the organizational need (as per below picture shows). Whereas, in **Semi-monthly**, periods are fixed from **1st to 15th** and **16th to the end of the month date**.

In **Monthly** Timesheet Period, TimeLive allows an administrator to set **Month Start Date** according to the organizational need.

11.4.2. Holiday Types

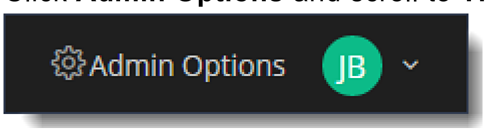
TimeLive provides few pre-defined holidays type such as US, UK etc. An administrator can define the holidays as per their organization rules & region by creating new **Holiday Type**. An administrator can create **Holiday Types** with different configurations and can assign these types separately to each individual employee. An administrator can also select **Default US Holiday** or delete it and create new Holiday Type. Following settings can be configured using **Holiday** options available in TimeLive.

Selecting Pre-defined Holidays Types:

TimeLive comes with different pre-defined holidays set. An administrator can select those pre-defined holidays set from *Admin Options > Holidays*. Once **Default** Holiday is selected, an administrator can fully edit those holidays types in **Admin Options > Holiday Types**.

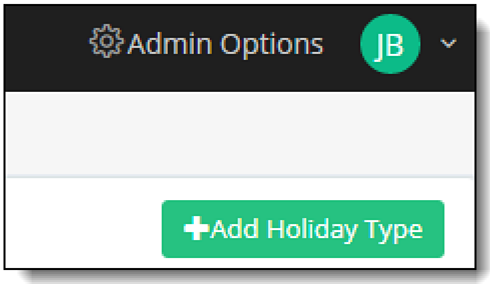
Adding Holiday Type:

1. Login TimeLive with Admin rights
2. Click **Admin Options** and scroll to **Timesheet Setup**

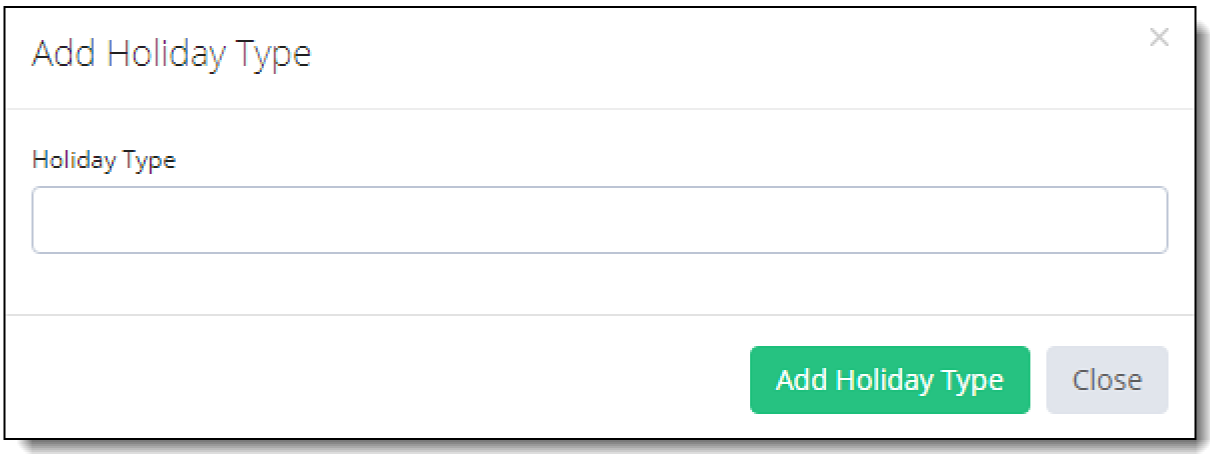


3. Click **Holiday Types** icon.

4. Click **+ Add Holiday Type** green button on the right-top of the application.



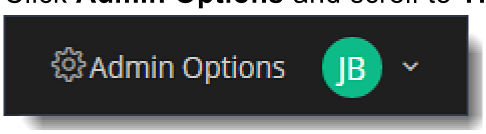
5. In the **Add Holiday Type** screen that appears, enter:



   a. Enter new holiday type name in the popup form.
   b. Click on **Add** to add this **Holiday Type** in TimeLive system.

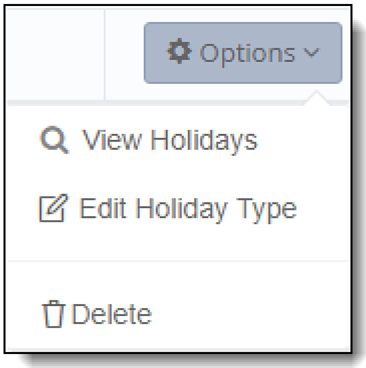
Filling new holidays in your newly created Holiday Type

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



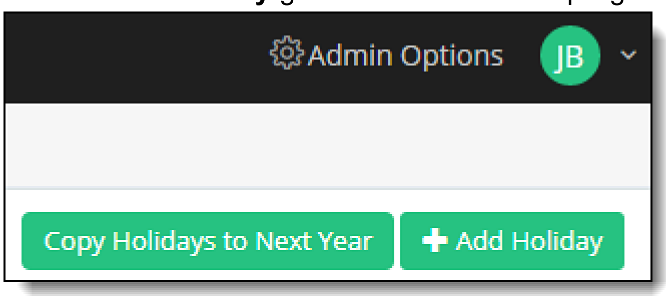
3. Click **Holiday Types** icon.

4. Scroll to the newly created Holiday Type, click **Options** gear icon.

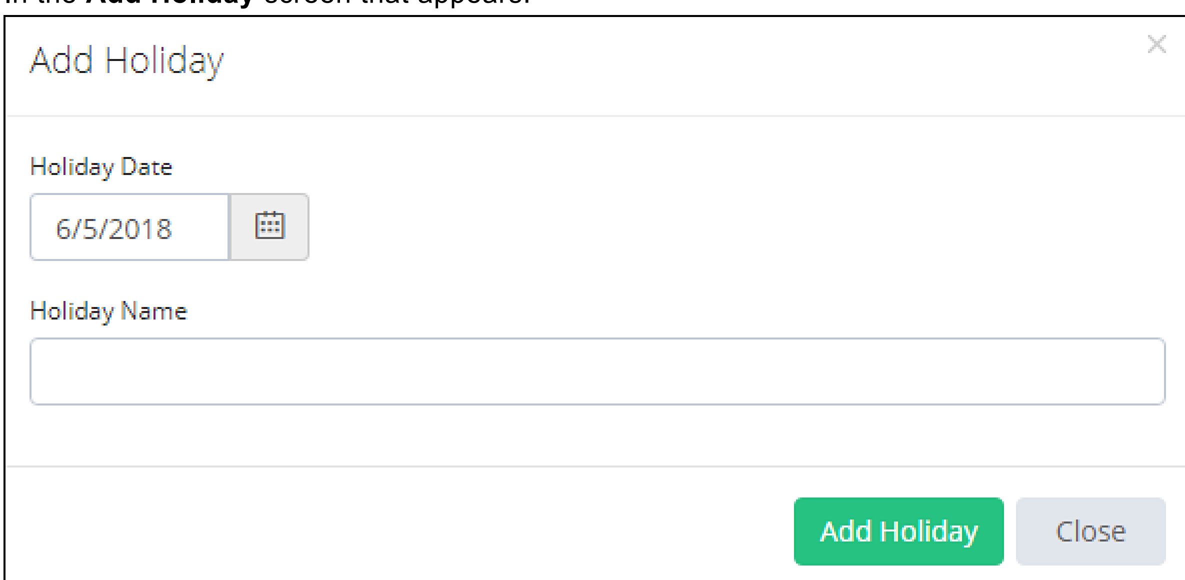


   a. Select **View Holidays**

5. Click **+ Add Holiday** green button on the top-right of the application.



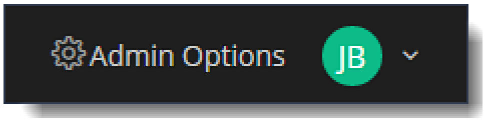
6. In the **Add Holiday** screen that appears:



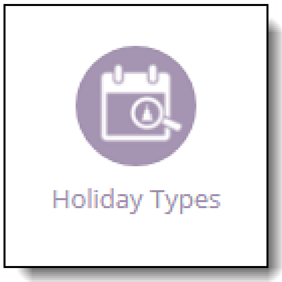
   a. Select the date in the Date Field and set the Name of the Holiday in the 2nd field.
   b. Click **Add Holiday** button.

Editing Holiday Type:

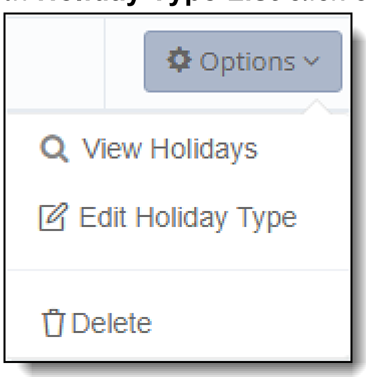
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



3. Click **Holiday Types** icon.

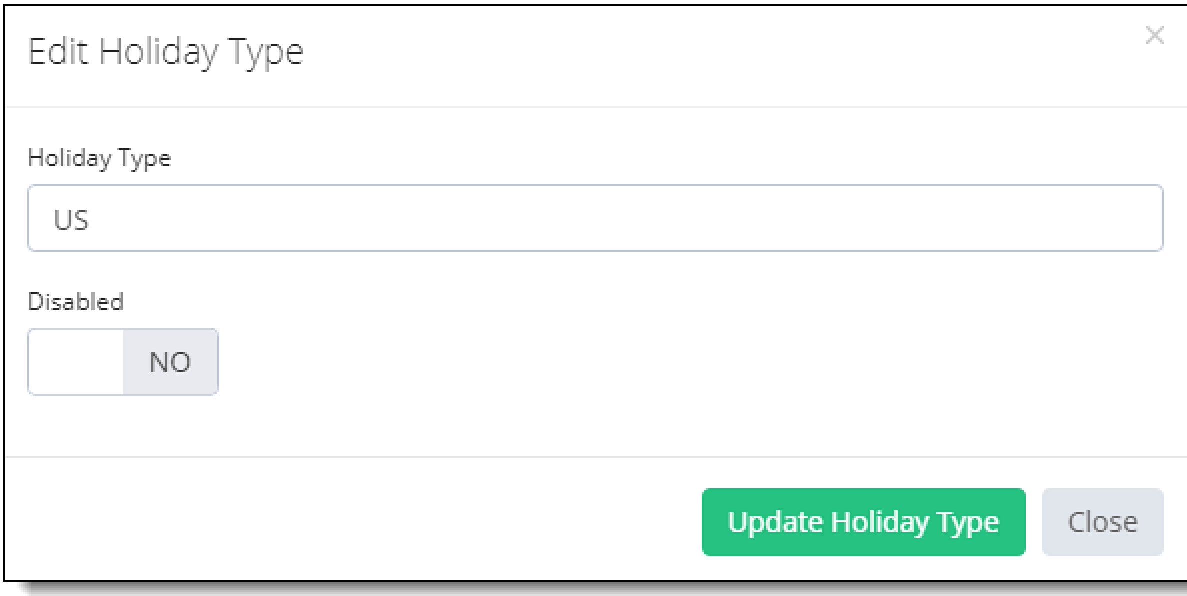


4. In **Holiday Type List** click on **Options** gear icon against the Holiday Type you want to modify.



   a. Select **Edit Holiday Type** option which you want to Edit.

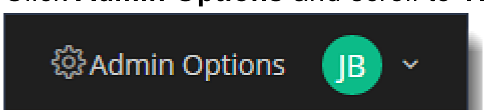
5. In the **Edit Holiday Type** screen that appears:



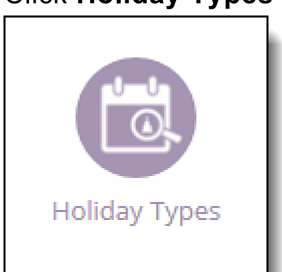
   a. Update your required modification in **Holiday Type Information** form.
   b. Click on **Update Holiday Type** button to update this record.

Deleting Holiday Type:

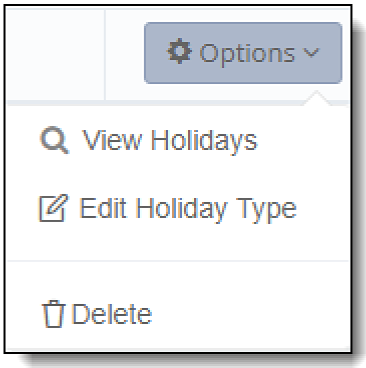
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



3. Click **Holiday Types** icon.

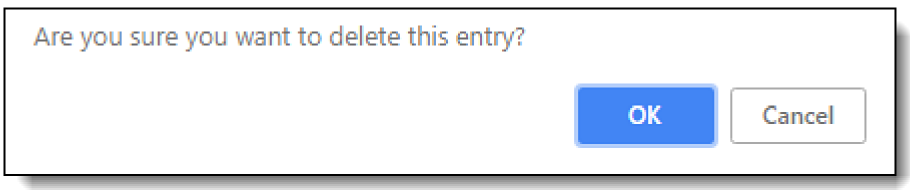


4. In **Holiday Type List** click on **Options** gear icon against the Holiday Type you want to delete.



   a. Select **Delete** option which you want to delete.

5. Click on **OK** to confirm the deletion.

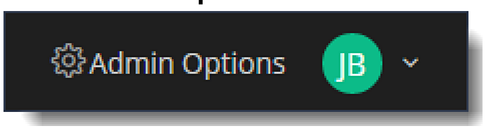


11.4.3. Holidays

TimeLive provides few pre-defined holidays type such as US, UK etc. An administrator can define the holidays as per their organization rules & region by creating new **Holiday Type**. An administrator can create **Holiday Types** with different configurations and can assign these types separately to each individual employee. An administrator can also select **Default US Holiday** or delete it and create new Holiday Type. Following settings can be configured using **Holiday** options available in TimeLive.

Assigning pre-defined/newly created Holidays to Employees

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



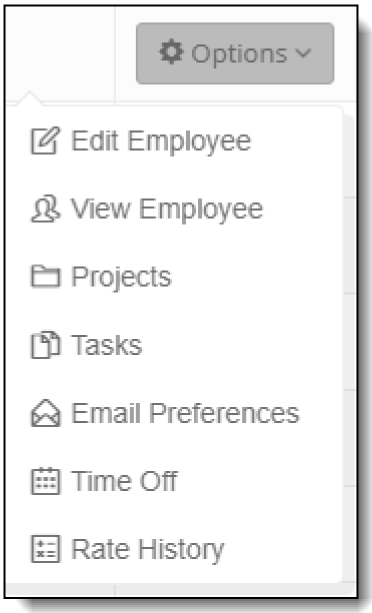
3. Click **Holidays** icon.

4. In the **Holidays** screen that appears:

   a. **Enable check** on the **Holiday Type** which you want to assign to your employees (only enable those types which you want to assign to your employees. They will reflect in Employees page for selection).
   b. Click **Update** button.

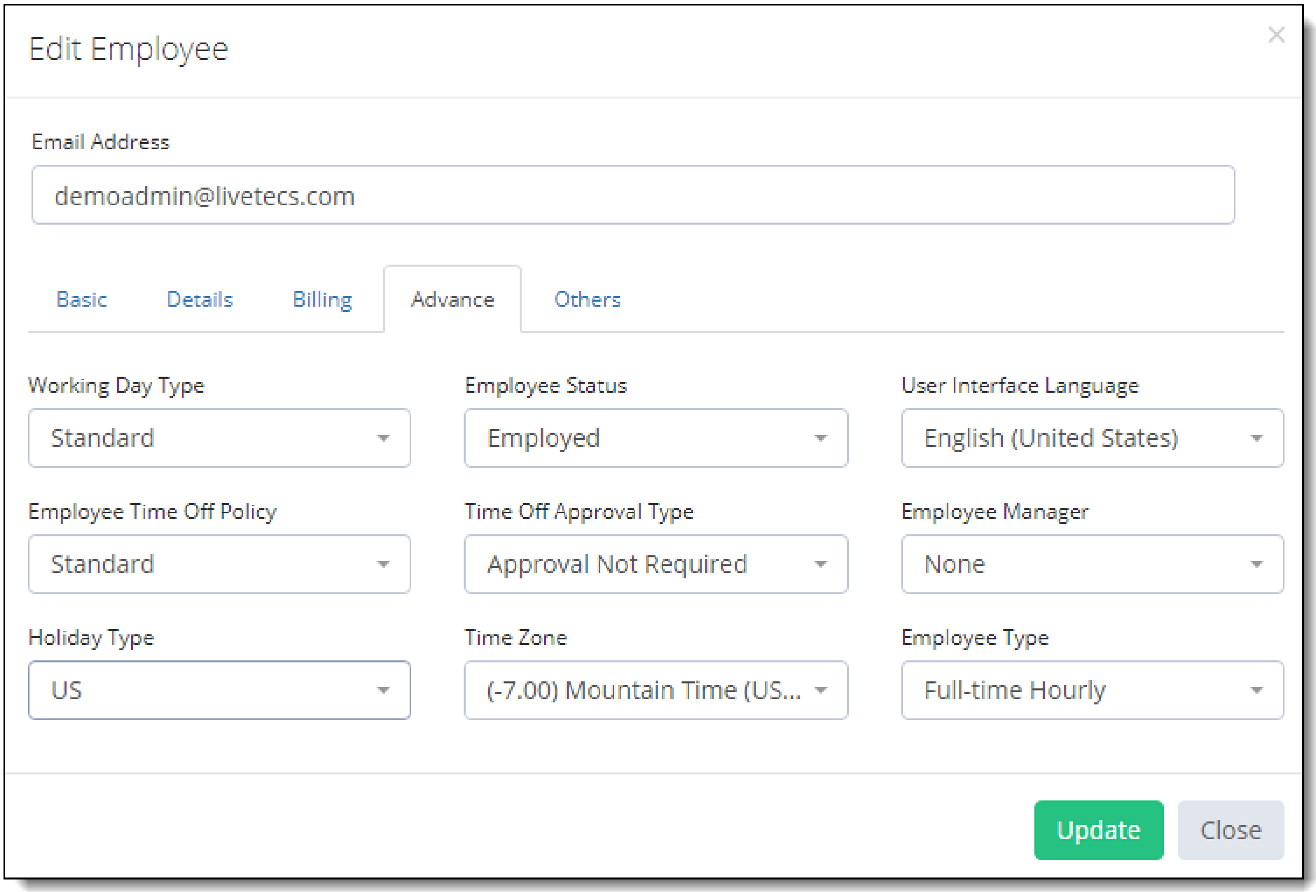
5. Navigate to **Employees** Link on the Left-top of the application.

6. Search for the employee to whom you want to assign the **Holiday Type** and click **Options** gear icon.



   a. Select **Edit Employee**.

7. In the **Edit Employee** screen that appears:



a. Navigate to **Advance** Tab.
b. Select the **Type** from the **Holiday Type** Field.
c. Click **Update** button.

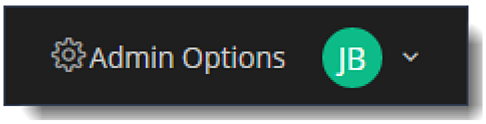
11.4.4. Absence Type

Absence Type is used in Attendance Management. Absence types can be configured using TimeLive Absence type setup. TimeLive provides few pre-defined Absence Types. An administrator can create their own Absence Types based on their organizational needs.

An administrator can view absence types using **Admin Options > Absence Types**

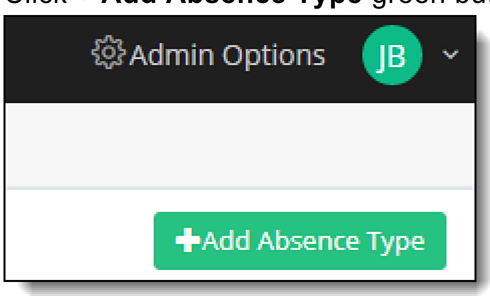
Adding new Absence type

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



3. Click **Absence Type** icon.

4. Click **+ Add Absence Type** green button on the top-right side of the application.

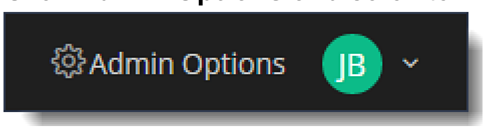


5. In the **Add Absence Type** screen that appears:

   a. Enter new Absence Type in **Absence Type Information** form which you want to add.
   b. Click on **Add** to add this absence type in TimeLive.

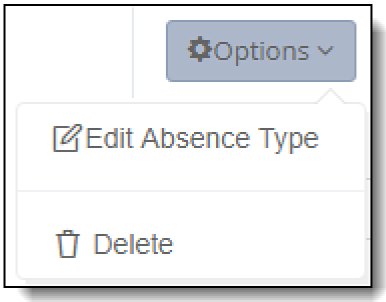
Editing Absence Type

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



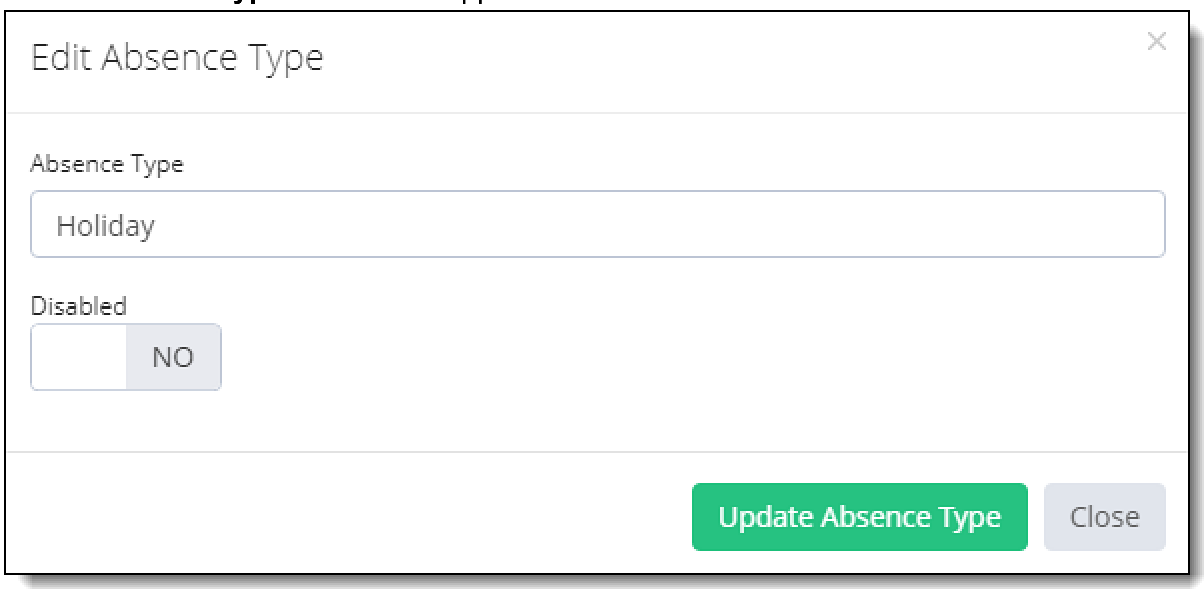
3. Click **Absence Type** icon.

4. In **Absence Type List**, click on **Options** gear icon of record which you want to modify.



   a. Select **Edit Absence Type**.

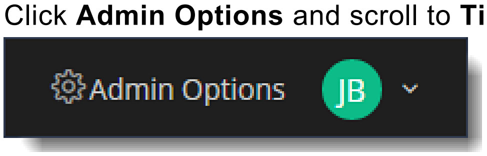
5. In the **Absence Type** screen that appears:



   a. Make the required changes in **Absence Type Information** form.
   b. Click on **Update Absence Type** button to update the changes.

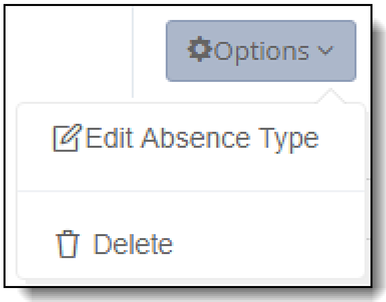
Disabling Absence Type

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



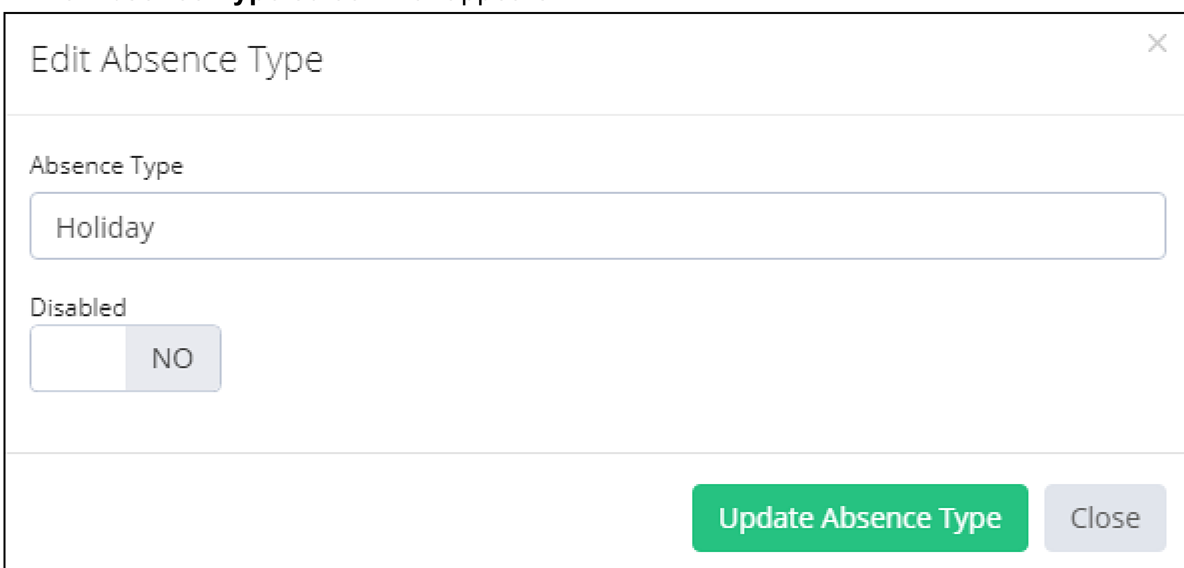
3. Click **Absence Type** icon.

4. In **Absence Type List**, click on **Options** gear icon of record which you want to disable.



   a. Select **Edit Absence Type**.

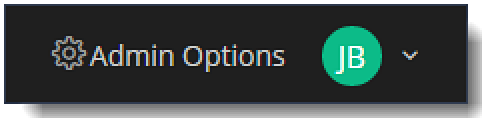
5. In the **Absence Type** screen that appears:



   a. Click on the **Disabled** switch and it will change to **Yes**.
   b. Click on **Update Absence Type** button to disable this type.

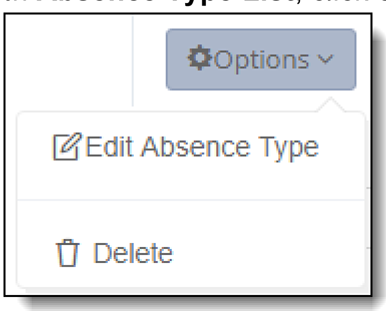
Deleting Absence Type

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.

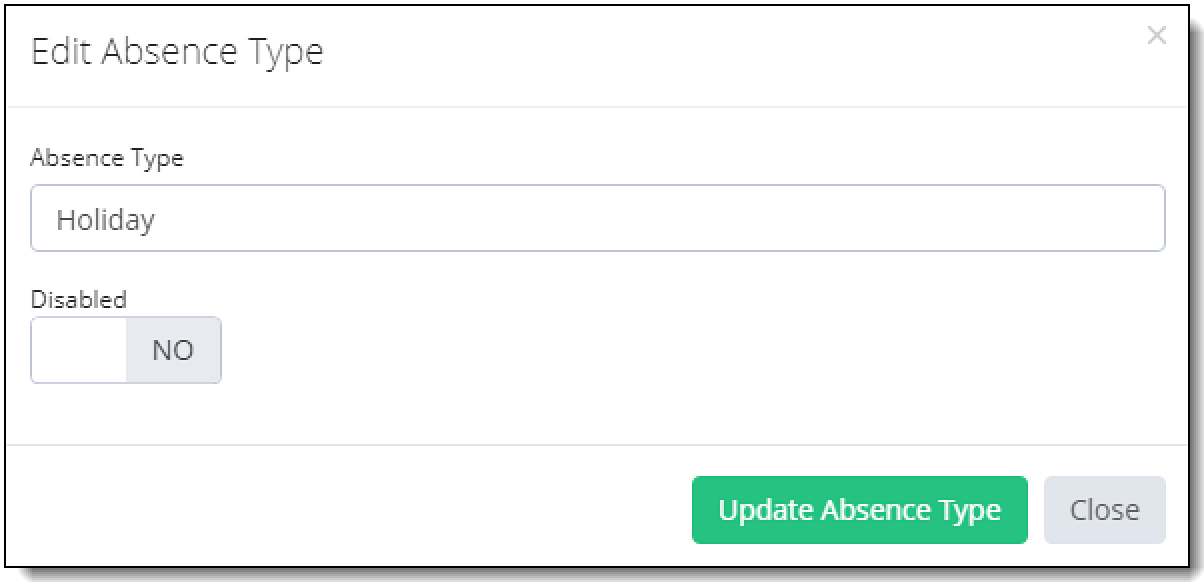


3. Click **Absence Type** icon.

4. In **Absence Type List**, click on **Options** gear icon of record which you want to disable.

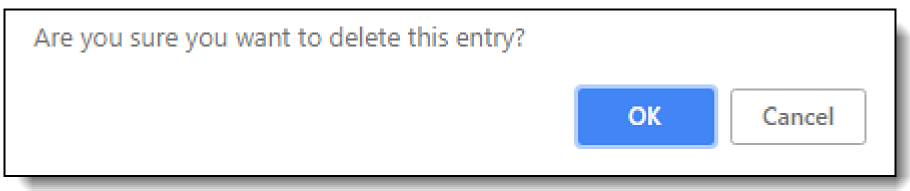


5. In the **Absence Type List** screen that appears:



   a. Select **Delete** Option.

6. Press **OK** to confirm the deletion.



11.4.5. Work Types

TimeLive Work Types implementation allow project manager and administrator to define separate billing rate and employee rate of every separate work type. Work type is something one level higher classification of time entry.

A typical case for separate billing rate for separate **Work Type** is

Employee Billing Rate (Standard): 80 per hour
Employee Billing Rate (Overtime): 100 per hour
Employee Billing Rate (Travel): 160 per hour.

In the above scenario, 1 employee have different working hour rates. With Work Type, an administrator can easily manage these and many others if need arrises.

Work Type in TimeLive:

An administrator can define separate billing rate for each Work Type. System Administrator can then enable **Work Type** field in My Timesheet through **Admin Options > Preferences > Timesheet Setup > "Show Work Type in TimeSheet"**. Employee/contractor can then select Work Type in time entry view. TimeLive by defaults come with 3 types of Work Types,

- Standard
- Overtime
- Travel

How to define separate employee and billing rate for different [work types]:

Work Type in Employee Own Billing Rate:

Select required **Work Type** in Employee Form and then define your Billing Rate and Employee Rate for that particular work type.

Work Type in Project-Based Employee Billing Rate:

Select required **Work Type** in drop-down and then define your Billing Rate and Employee Rate for that particular work type.

Work Type in Role-Based Billing Rate case

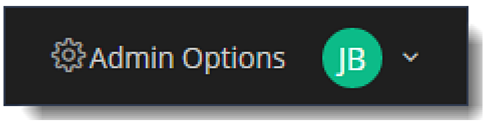
Select required **Work Type** in drop-down and then define your Billing Rate and Employee Rate for that particular work type.

Work Type in Task-Based Billing Rate] case

Select required Work Type in drop-down and then define your Billing Rate and Employer Rate for that particular work type.

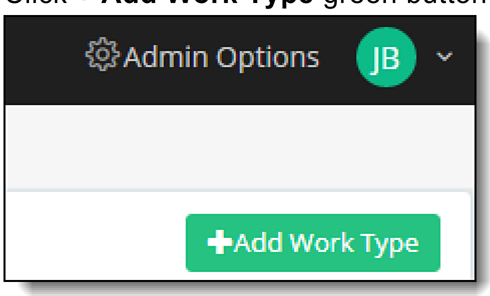
Adding new Work Type

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.

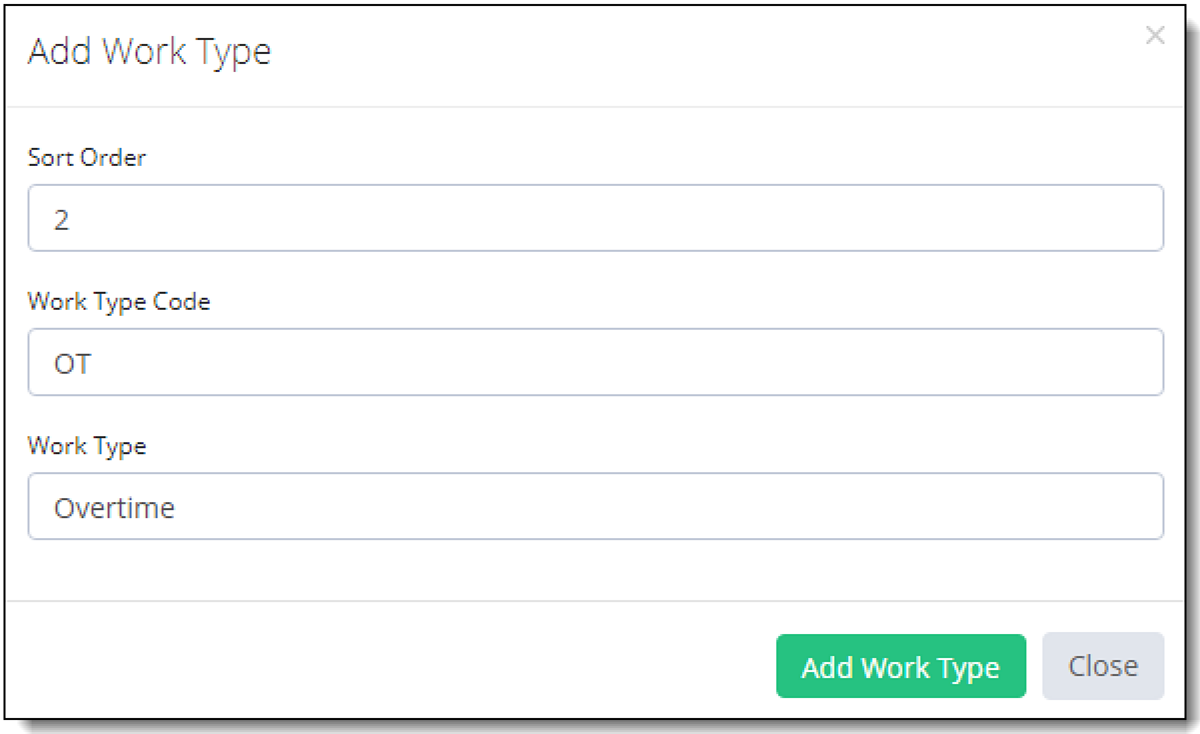


3. Click **Work Types** icon.

4. Click **+ Add Work Type** green button on the top-right side of the application.



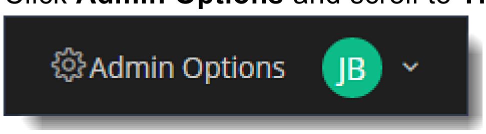
5. In the **Add Work Type** screen that appears:



   a. Enter new Work Type in **Work Type Information** form which you want to add.
   b. Short Work Type Code.
   c. Sort Order(Optional).
   d. Click on **Add Work Type** to add this in TimeLive.

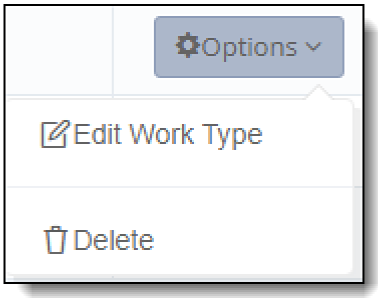
Editing Work Type:

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



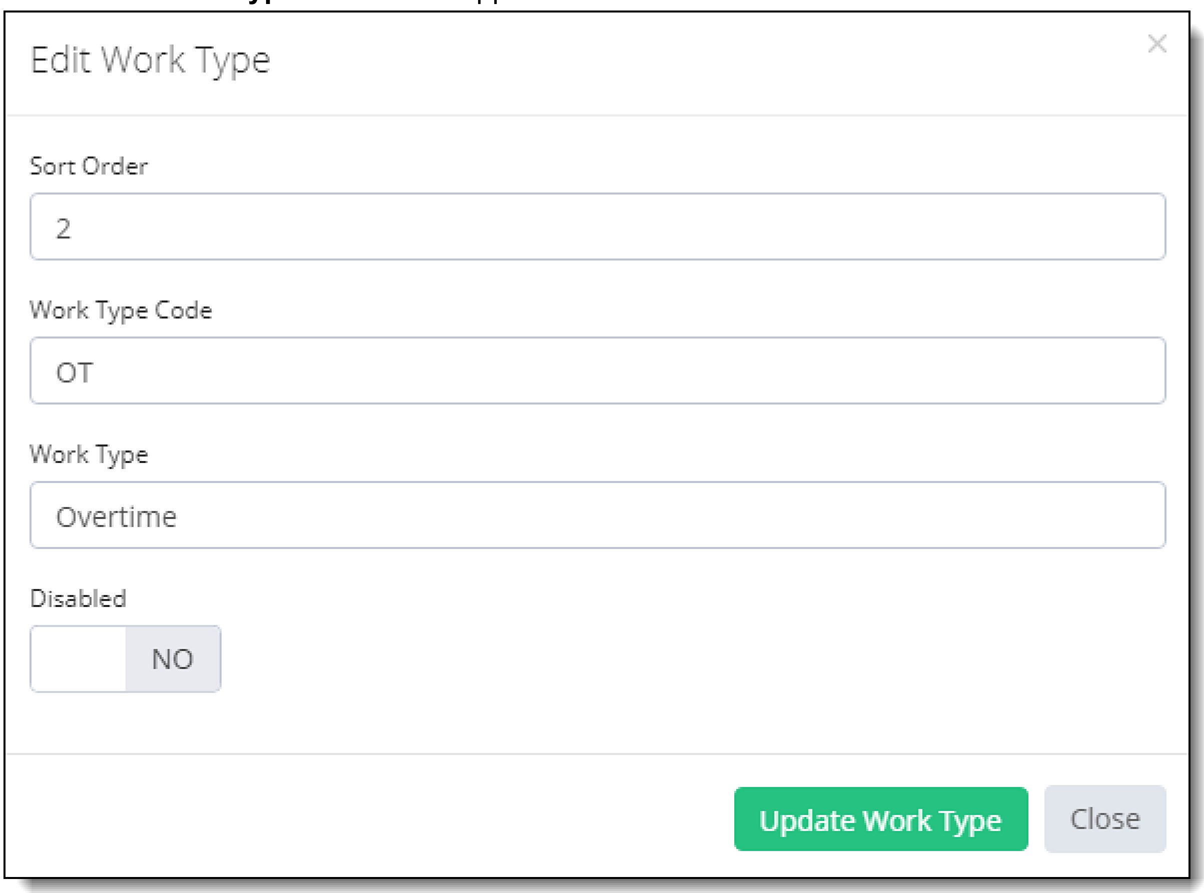
3. Click **Work Types** icon.

4. In **Work Type List**, click on **Options** gear icon of record which you want to modify.



   a. Select **Edit Work Type**.

5. In the **Edit Work Type** screen that appears:

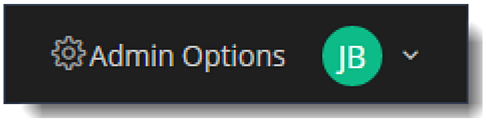


   a. Make the required changes in **Work Type Information** form.
   b. Click on **Update Work Type** button to update the changes.

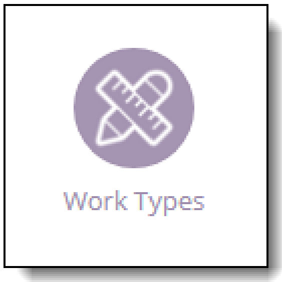
Disabling Absence Type:

1. Login TimeLive with Admin rights.

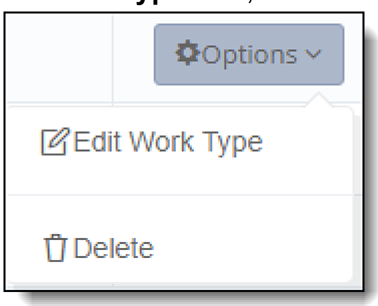
2. Click **Admin Options** and scroll to **Timesheet Setup**.



3. Click **Work Types** icon.

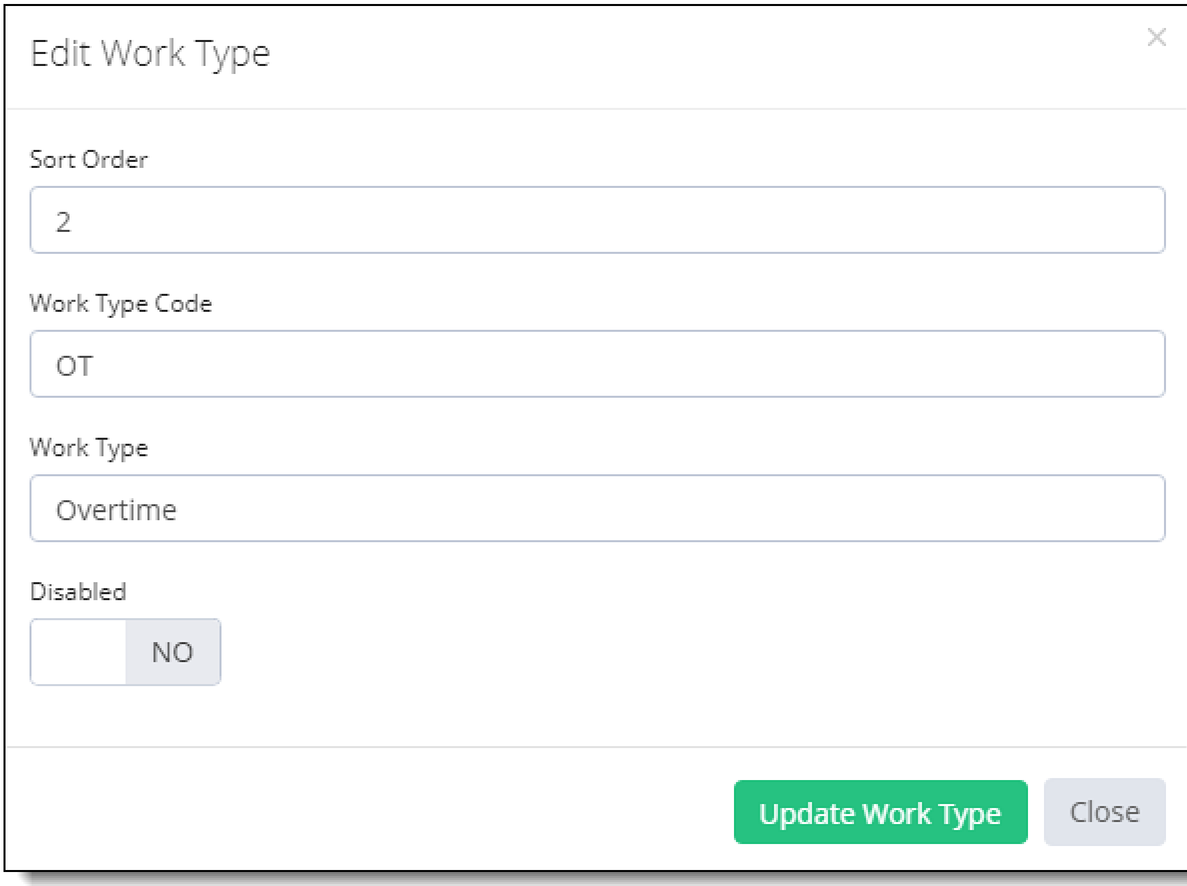


4. In **Work Type List**, click on **Options** gear icon of record which you want to disable.



   a. Select **Edit Work Type** option.

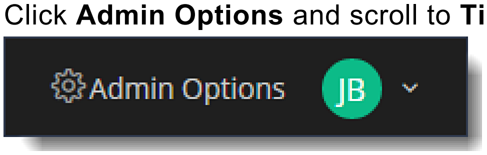
5. In the **Edit Work Type** screen that appears:



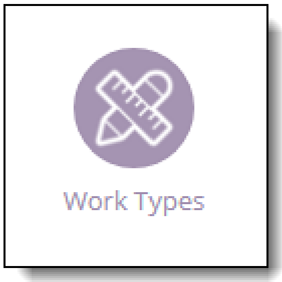
   a. Click on the **Disabled** switch and it will change to **Yes**.
   b. Click on **Update Work Type** button to update the changes.

Deleting Absence Type:

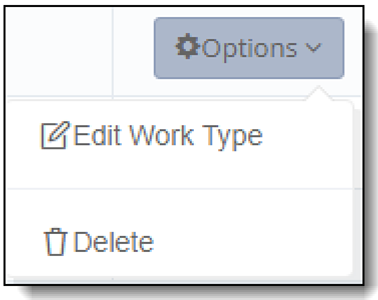
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Timesheet Setup**.



3. Click **Work Types** icon.

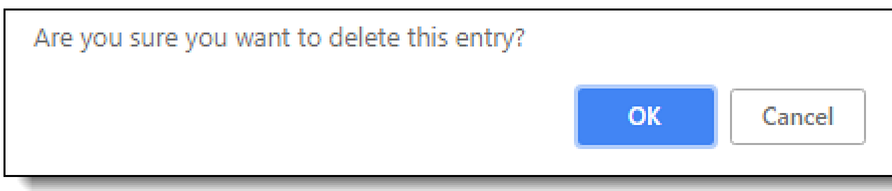


4. In **Work Type List**, click on **Options** gear icon of record which you want to delete.



   a. Select **Delete** option.

5. Press **OK** to confirm the deletion.



11.4.6. Working Days

Working Days in TimeLive plays a vital role in TimeLive. An administrator can manage a different kind of validations, configurations, and policies which applies to Timesheet. These sets of Working Day Type can be assigned to every individual employee on Employee profile option. Below are the functions/possibilities can be achieved with this tool,

- Administrator can manage different Working Days, Working Hours, Timesheet Hours Validation, TimeOff Validations, Timesheet Locking system, and most importantly Timesheet submission period(Daily/Weekly/Bi-Weekly/Semi-Monthly/Monthly) for a different levels/region/groups/department of employees
    - In US, Week starts with Top management may work 5 days a week whereas, labor may work 7 days a week. This could easily be managed by creating 2 separate **Working Days**. Again Top Management may work 7-8 hours a day whereas, labor may work 8-10 hours/day. This also could easily be managed by creating a separate **Working Days**. An administrator can restrict employees to enter time to their some or all previous/next timesheets on the specific date of the month.

The administrator can create or modify Working Days Types using **Admin Options > Working Days**. Each of the settings/configuration options is explained below.

Working Day Type Available Options.

Field Name	Description
Basic	
Working Day Type	Specify the name to be called for this Working Day Type.
Timesheet Overdue	Option can be used to inform the employee that Timesheet is pending. It can be set manually for the number of days. Empty or Zero is by default on Due Date.
Week Start Day	This is a drop box option used to identify the 1st working day of the week.
Timesheet Period Type	This is a drop box option used to assign employee Timesheet entry Daily/Weekly/Biweekly/Semi-Monthly/Monthly.
Show Clock Start/End	This checkbox allows the user to enter work start and end time in Time Entry.
Disable	It is a checkbox appear in Edit mode to allow an administrator to disable specific Working Day Type.

Hours Validation	
Hours Per Day	A numerical value which specifies the duty hours per day for that specific Working Day Type. This value will be used to convert days into hours and hours into days.
Minimum Hours Per Day	A numerical value which specifies the minimum allowed working hours per day. The system will validate the hour entered in Timesheet is not less than the minimum specified.
Maximum Hours Per Day	A numerical value which specifies the maximum allowed working hours per day. The system will validate the hour entered in Timesheet does not exceed than the maximum specified.
Minimum Hours Per Period	A numerical value which specifies the minimum allowed working hours per Period. The system will validate the hour entered in Timesheet is not less than the minimum specified. [Option can be changed via Admin Option – Preferences – Timesheet Setup]
Maximum Hours Per Period	A numerical value which specifies the maximum allowed working hours per Period. The system will validate the hour entered in Timesheet does not exceed than the maximum specified. [Option can be changed via Admin Option – Preferences – Timesheet Setup]
Enable Balance Validation for Time Off	It is a check-box option through which user can see the Time Off balance while entering Time Off.
Locking Timesheet	
Lock Previous Timesheet Periods	This option is used when to restrict employees edit the previous Timesheet Periods. Empty or Zero is by default.
Lock Next Timesheet Periods	This option is used when to restrict employees edit the Next Timesheet Periods. Empty or Zero is by default.
Lock All Periods Except Previous	This option is used when to allow employees to only edit the Previous Timesheet Periods. Empty or Zero is by default.
Lock All Periods Except Next	This option is used when to allow employees to only edit the Next Timesheet Periods. Empty or Zero is by default.
Lock Previous/ Next Timesheet On	This option is used when to restrict employees to edit Timesheet on any specific date of the month. Empty or Zero is by default.

Lock All Previous Timesheets	It is a check-box option through which All previous Timesheets can be locked.
Lock All Next Timesheets	It is a check-box option through which All next Timesheets can be locked.
Percentage Validation	
Minimum Percentage Per Day	A numerical value which specifies the minimum allowed working hours percentage per day. The system will validate the hour entered in Timesheet is not less than the minimum specified. [Option can be changed via Admin Option – Preferences – Timesheet Setup]
Maximum Percentage Per Day	A numerical value which specifies the maximum allowed working hours percentage per day. The system will validate the hour entered in Timesheet is not less than the maximum specified. [Option can be changed via Admin Option – Preferences – Timesheet Setup]
Minimum Percentage Per Period	A numerical value which specifies the minimum allowed working hours Percentage per Period. The system will validate the percentage entered in Timesheet is not less than the minimum specified. [Option can be changed via Admin Option – Preferences – Timesheet Setup]
Maximum Percentage Per Period	A numerical value which specifies the Maximum allowed working hours Percentage per Period. The system will validate the percentage entered in Timesheet does not exceed than the Maximum specified. [Option can be changed via Admin Option – Preferences – Timesheet Setup]
Working Days	
Working Days & Selection	Administrator can select working days for each Working days type.

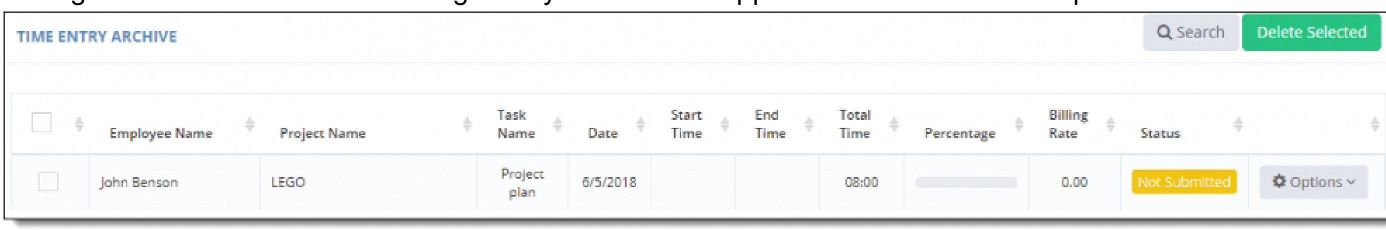
Assigning Working Day to the Employee:

1. Login TimeLive with Administrative Rights
2. Click on Employees link at the top-left of the Application
3. Click **Options** gear icon of the employee to which you want to assign the Working Day and select **View Employee** option
4. Again click on the **Options** gear icon and select **Edit Employee**
5. Navigate to **Others** tab and in the **Working Day Type** field, select the Working Day Type which you want to assign
6. Click Update to make the changes

11.4.7. Time Entry Archive

Sometimes, there are entries which have been submitted or approved by mistake or employee need to change something which is already submitted approved but unable make the changes. With this, **user can submit, Un-submit, Approve/reject, and delete the Timesheet entries and Time Off entries of any employee**. Administrator can access this feature through **Admin Options > Time Entry Archive**

NOTE: This feature of TimeLive is not for everyone and only should be allowed to Administrator or TimeLive Administrator/Power User. If any changes done from this feature, in the audit, it will be shown as changes done by the system rather then employee name. Further, the approval process will be override if done through this feature. So the tracking of any submission/approval/deletion will be impossible.



11.5. Project / Task Setup

11.5.1. Task Types

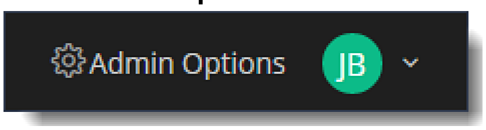
Task types is an identity of a task representing of the particular task group. It can be different for different industries and organizations. TimeLive provides few pre-defined tasks with an option of adding your own required task types in the system.

- **Task**
- **Bug**
- **Issue**

System Administrator can set up task type using **Admin Options > Task Types**

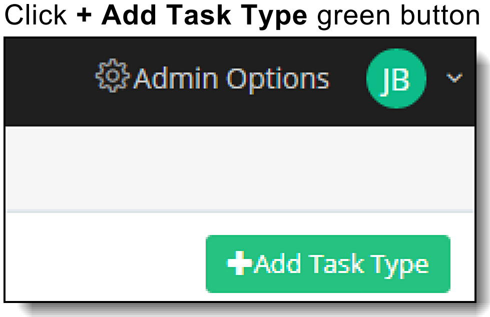
Adding new task type:

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.

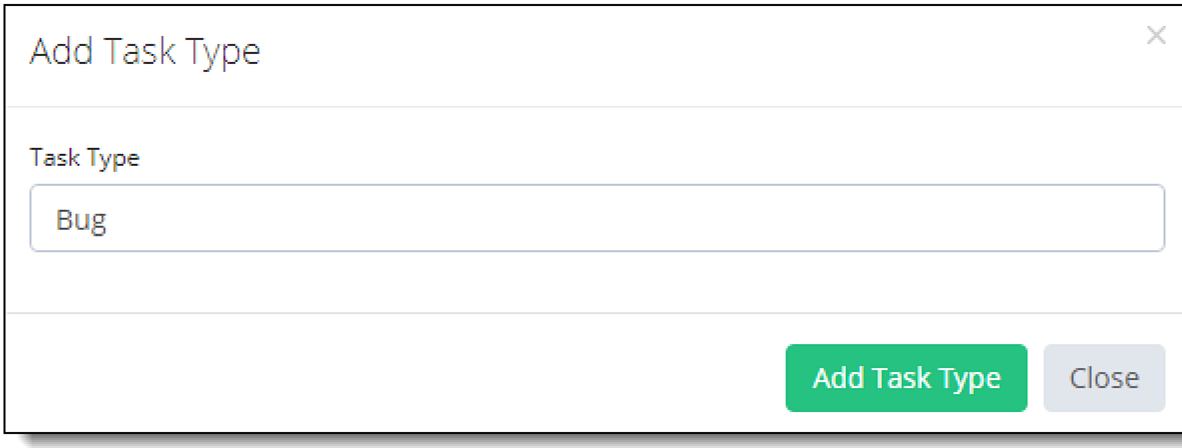


3. Click **Task Types** icon.

4. Click **+ Add Task Type** green button on the top-right side of the application.



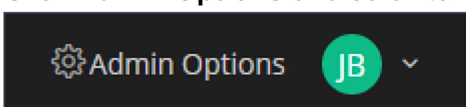
5. In the **Add Task Type** screen that appears:



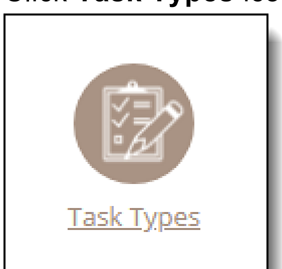
   a. Enter new Task Type in **Task Type** form which you want to add.
   b. Click on **Add Task Type** to add this in TimeLive.

Editing Task Type:

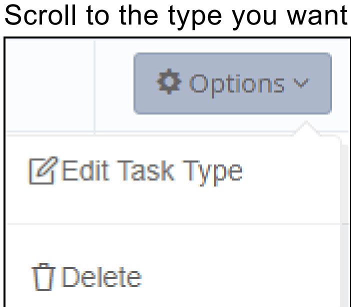
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.



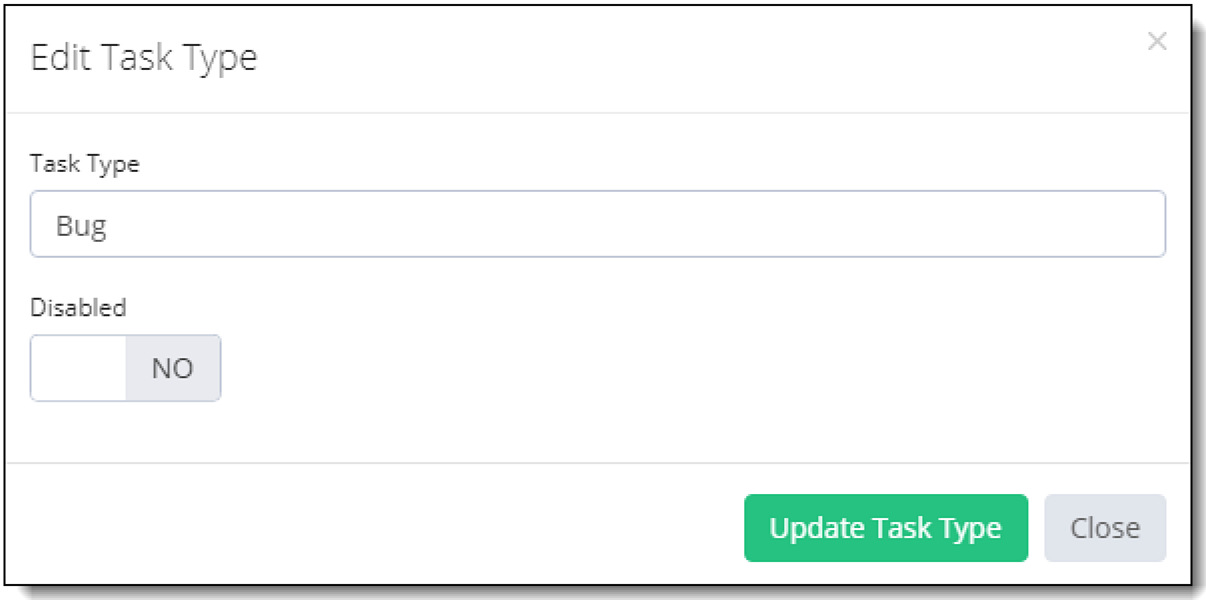
3. Click **Task Types** icon.



4. Scroll to the type you want to edit, click the **Options** gear icon and select **Edit Task Type** option.



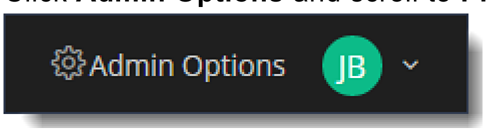
5. In the **Edit Task Type** screen that appears:



   a. Make the changes in the **Task Type** field.
   b. Click on **Update Task Type** button to update changes.

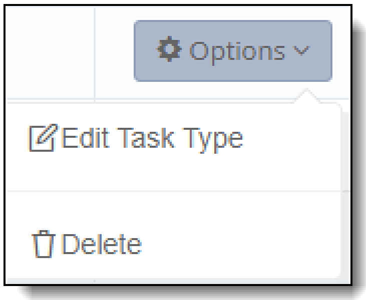
Disabling Task Type:

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.

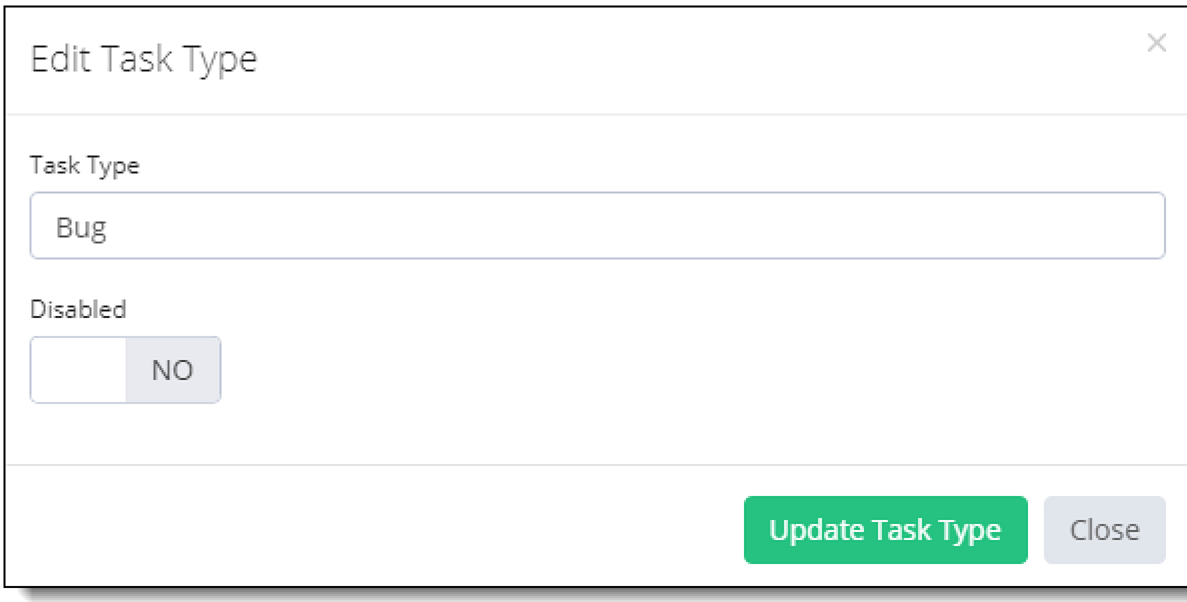


3. Click **Task Types** icon.

4. Scroll to the type you want to disable, click the **Options** gear icon and select **Edit Task Type** option.



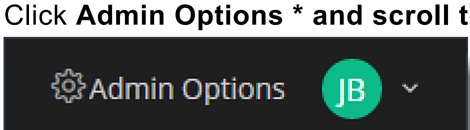
5. In the **Edit Task Type** screen that appears:



   a. Click the **Disabled** switch to convert it to **Yes**.
   b. Click on **Update Task Type**.button to update changes.

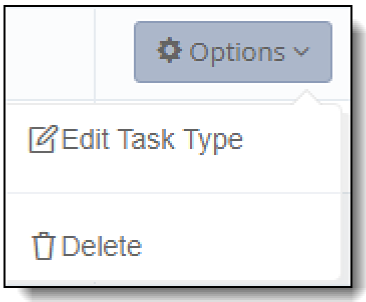
Deleting Task Type:

1. Login TimeLive with Admin rights.
2. Click **Admin Options * and scroll to *Project / Task Setup**.

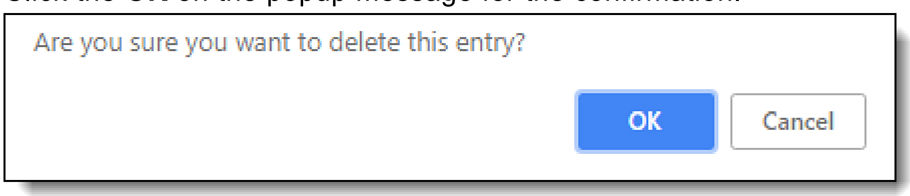


3. Click **Task Types** icon.

4. Scroll to the type you want to delete, click the **Options** gear icon and select **Delete** option.



5. Click the **OK** on the popup message for the confirmation.



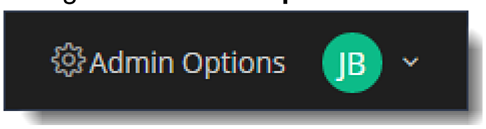
11.5.2. Priorities Setup

Priorities setup allow the system administrator to create own custom priorities of tasks. TimeLive provides few pre-defined priorities which can be renamed or edited according to organization needs.

A system administrator can set up priorities using **Admin Options > Priorities**

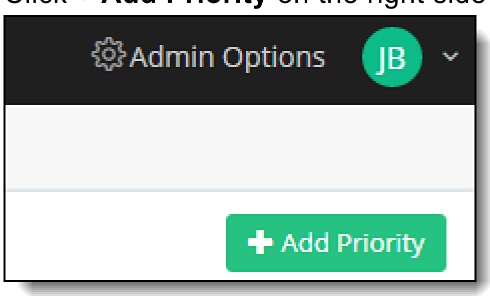
Adding new priority:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Project / Task Setup**.

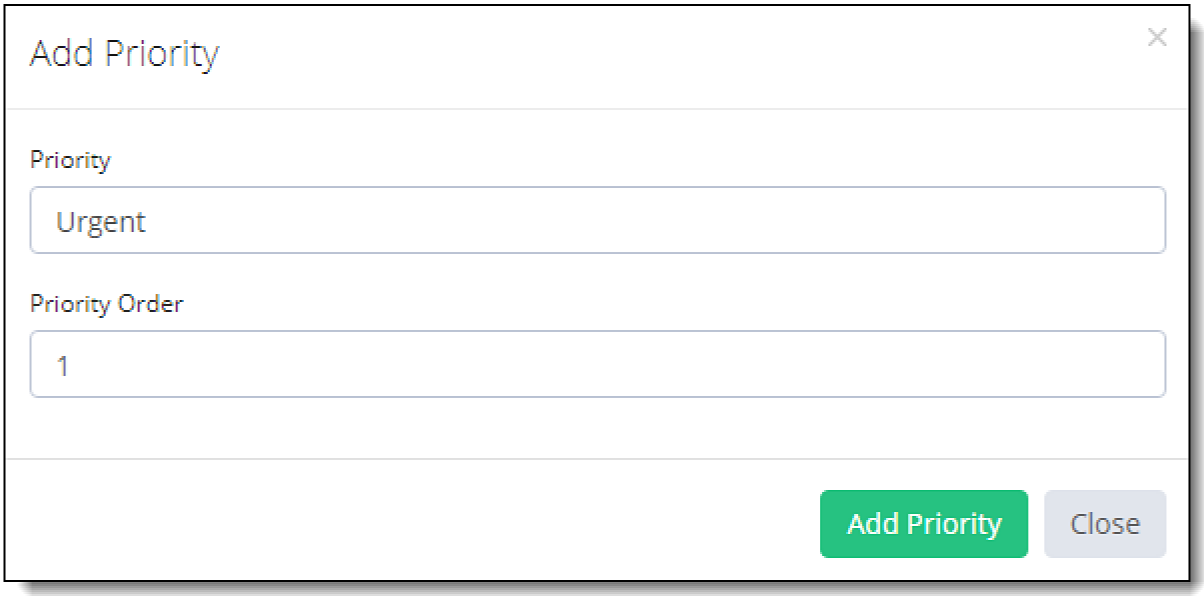


3. Click **Priorities** icon.

4. Click **+ Add Priority** on the right side of the application.



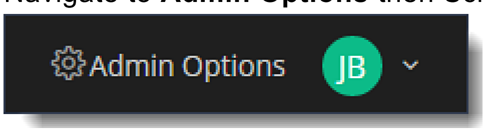
5. In the **Add Priority** screen that appears:



   a. Set the **Priority Name** in the first field.
   b. Set the **Priority Order**.
   c. Click on **Add Priority** green button to add this priority in the TimeLive system.

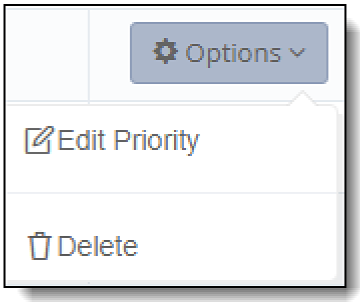
Editing Existing Priority:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Project / Task Setup**.



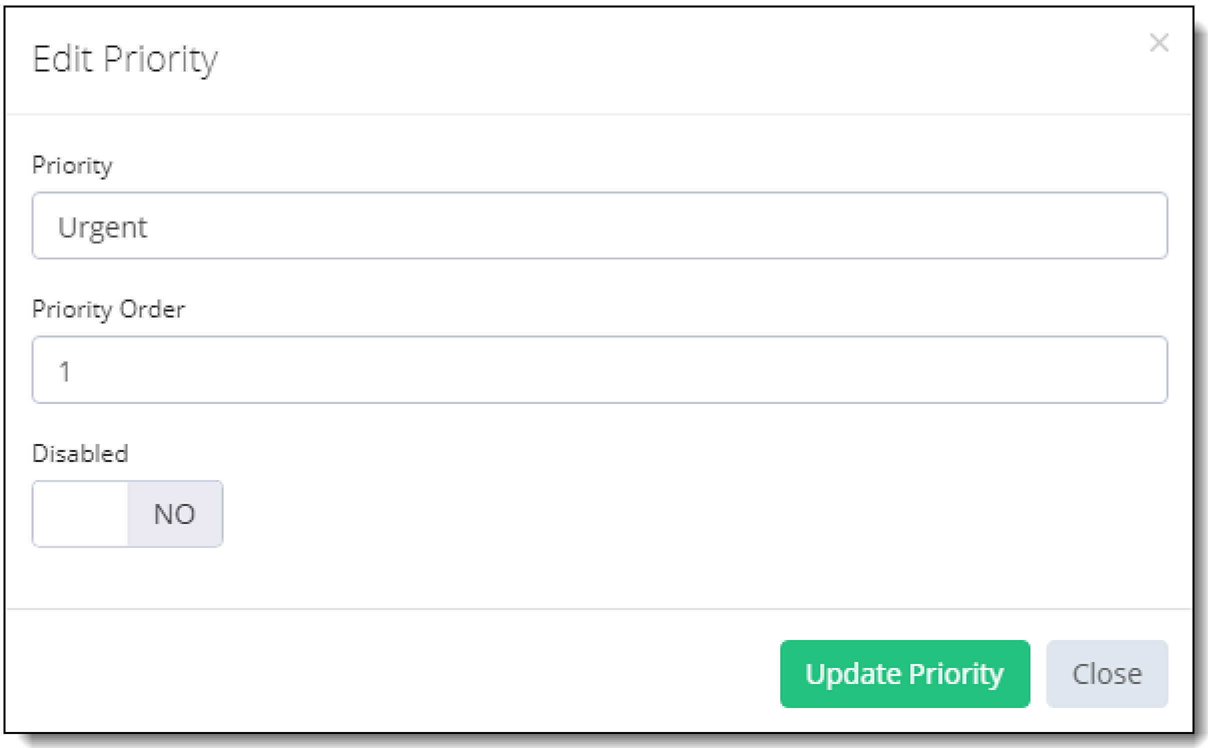
3. Click **Priorities** icon.

4. In **[Priority List]**, navigate to the Priority Type you want to modify and click the Options gear icon against this Priority Type.



   a. Click **Edit Priority** Option.

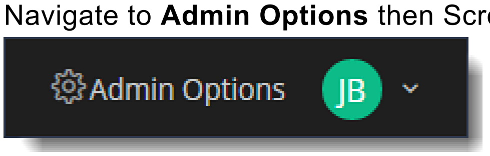
5. In the **Edit Priority** screen that appears:



   a. Make the changes in the required filed.
   b. Click on **Update Priority** button to update changes.

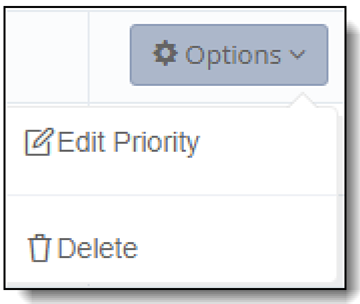
Disabling Existing Priority:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Project / Task Setup**.



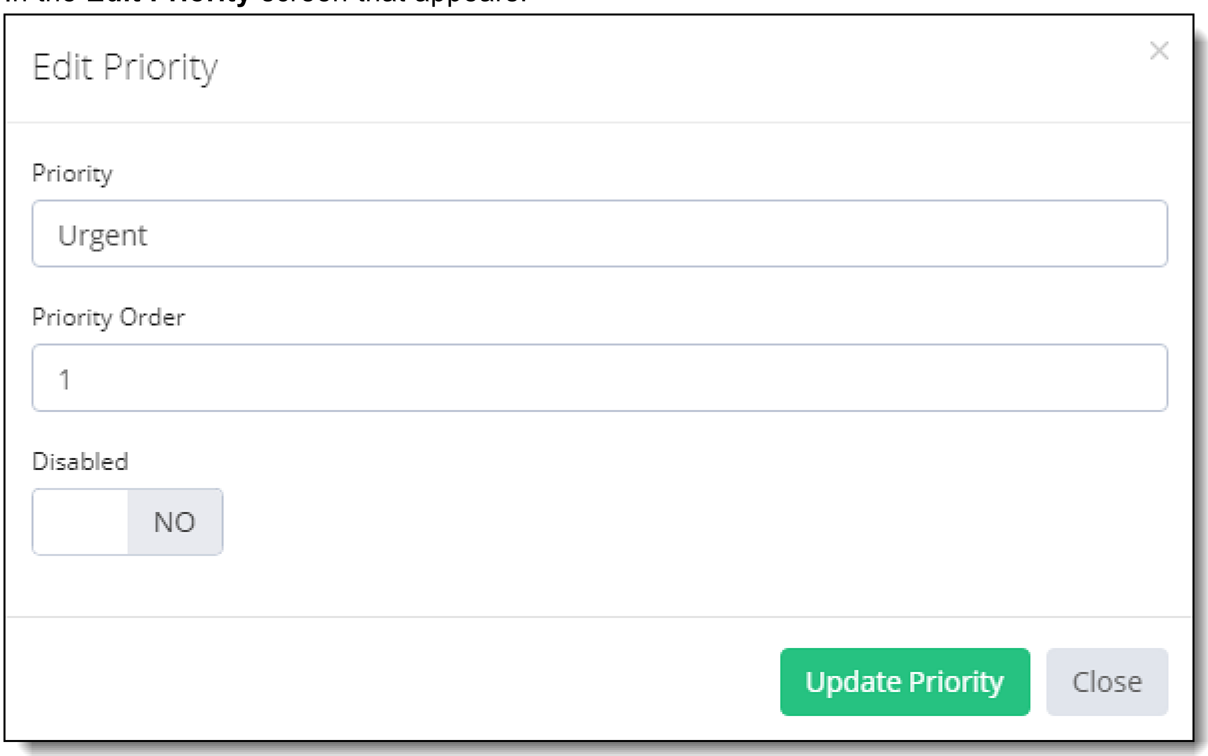
3. Click **Priorities** icon.

4. In **[Priority List]**, navigate to the Priority Type you want to disable and click the Options gear icon against this Priority Type.



   a. Click **Edit Priority** Option.

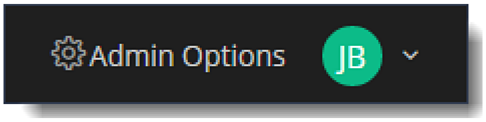
5. In the **Edit Priority** screen that appears:



a. Click on **Disabled** Option and change it to **Yes**.
b. Click on **Update Priority** button to update changes.

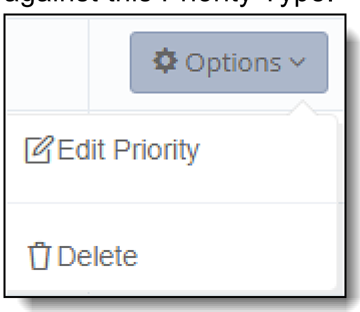
Deleting priority:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Project / Task Setup**.

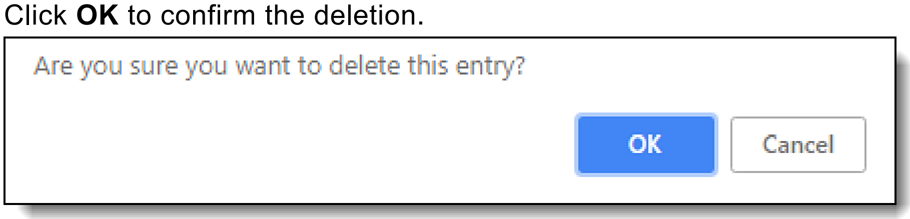


3. Click **Priorities** icon.

4. In **[Priority List]**, navigate to the Priority Type you want to disable and click the Options gear icon against this Priority Type.



a. Click on **Delete** Option.
b. Click **OK** to confirm the deletion.

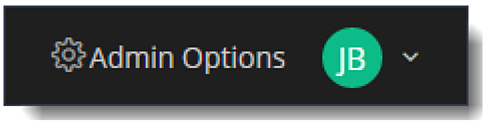


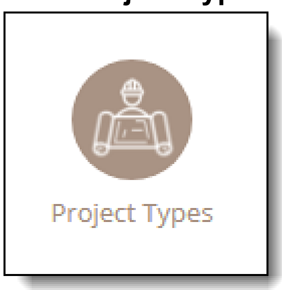
11.5.3. Project Types

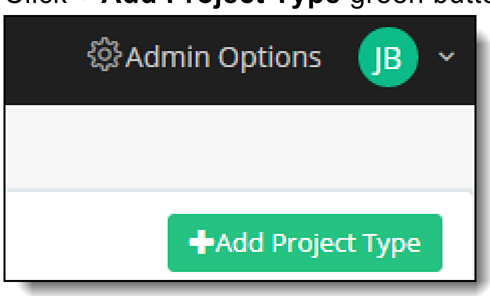
Project types are the different categories of projects which an organization is executing. This depends on the organization need. For example, a software consulting company can have a different type of projects like “Web Development”, “Application Development” etc.

System Administrator can set up Project Types through **Admin Options > Project Types**

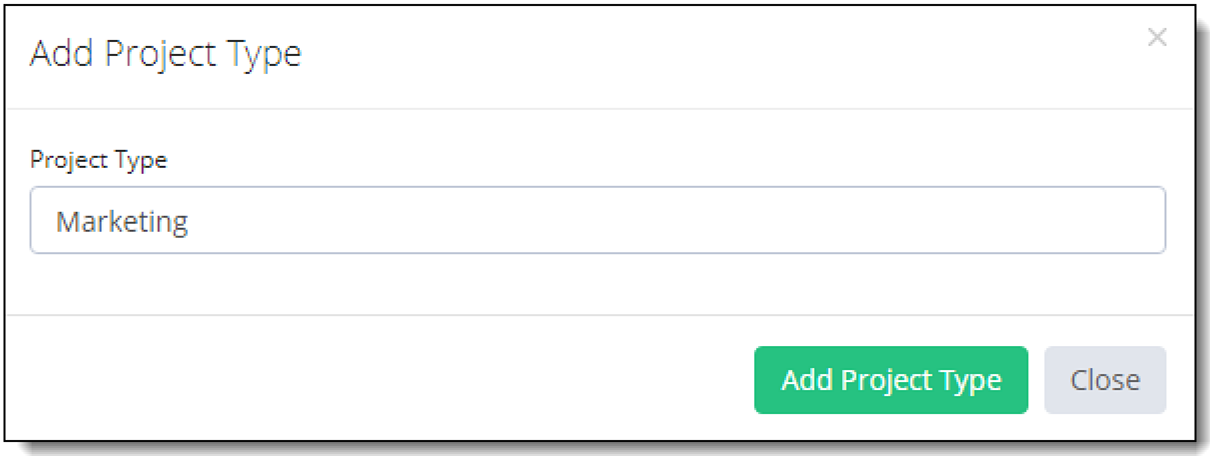
Adding new Project type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Project / Task Setup**.


3. Click **Project Types** icon.


4. Click **+ Add Project Type** green button on the top-right side of the application.


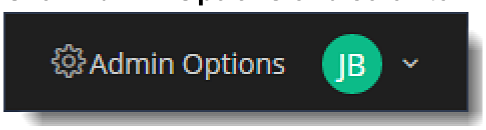
5. In the **Add Project Type** screen that appears:



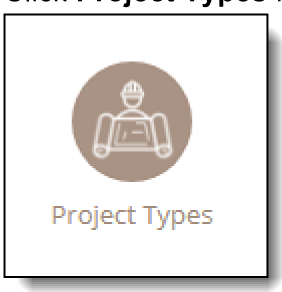
   a. Enter new Project Type in **Project Type** form which you want to add.
   b. Click on **Add Project Type** to add this in TimeLive.

Editing Project Type:

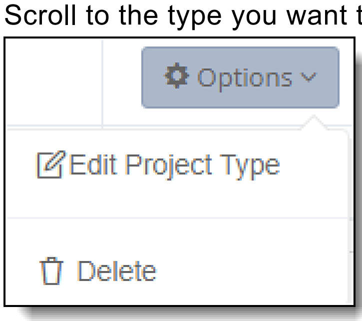
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.



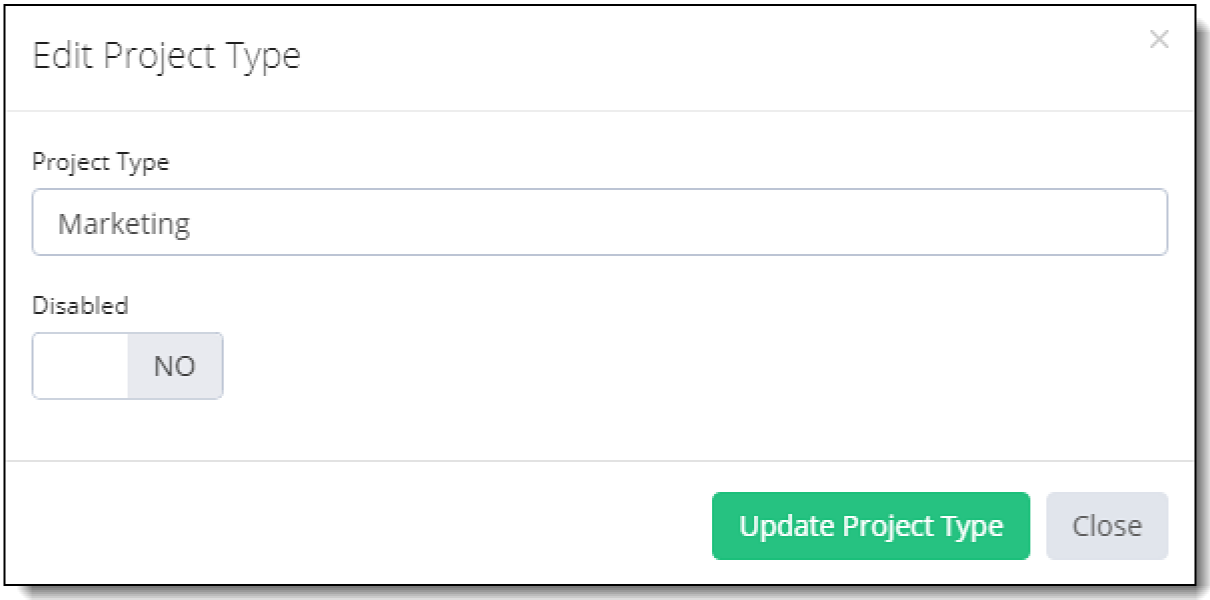
3. Click **Project Types** icon.



4. Scroll to the type you want to edit, click the **Options** gear icon and select **Edit Project Type** option.



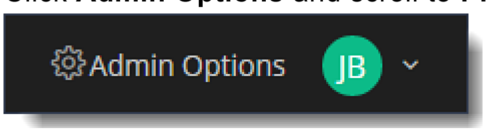
5. In the **Edit Project Type** screen that appears:



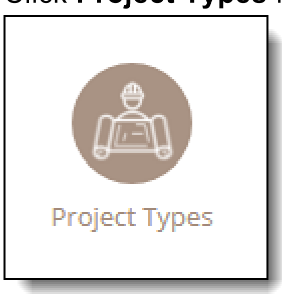
   a. Make the changes in the field.
   b. Click **Update Project Type** button to update changes.

Disabling Project Type:

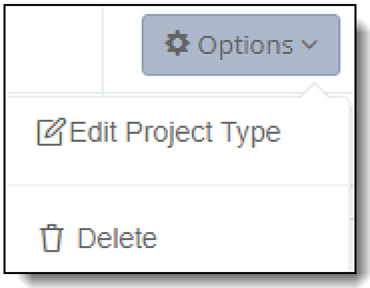
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.



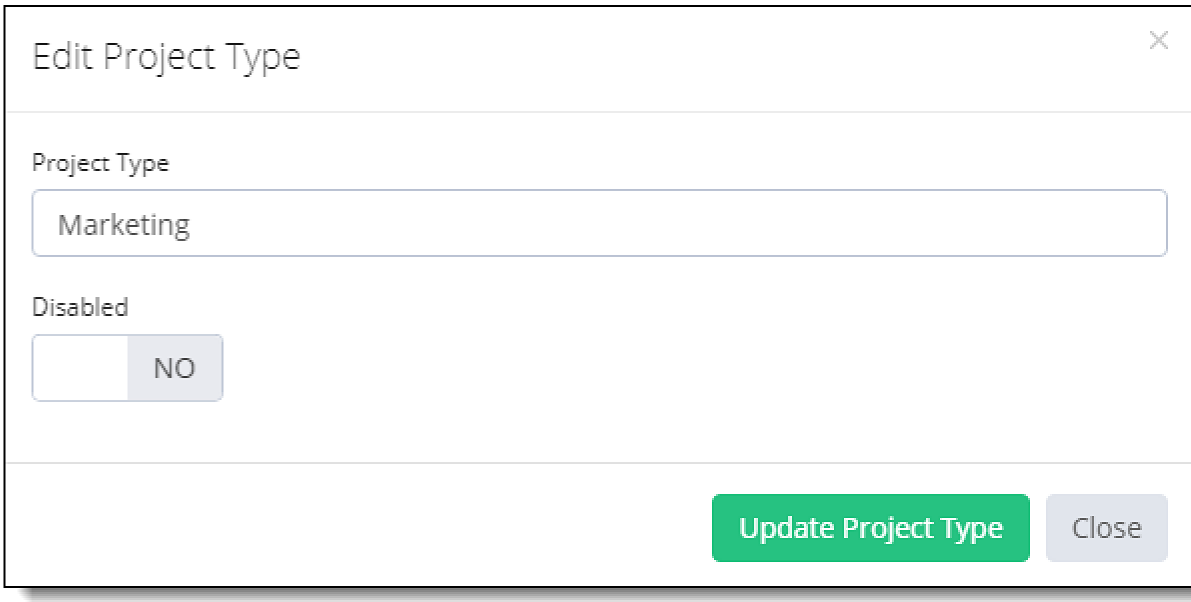
3. Click **Project Types** icon.



4. Scroll to the type you want to disable, click the **Options** gear icon and select **Edit Project Type** option.



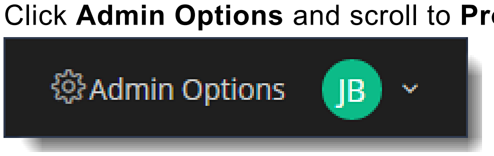
5. In the **Edit Project Type** screen that appears:



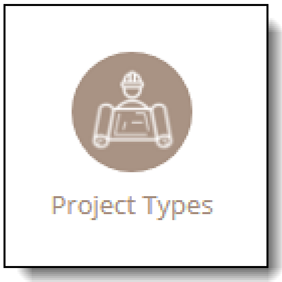
   a. Click the **Disabled** switch to convert it to **Yes**.
   b. Click **Update Project Type** button to update changes.

Deleting Project Type:

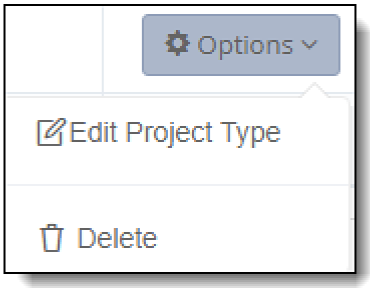
1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.



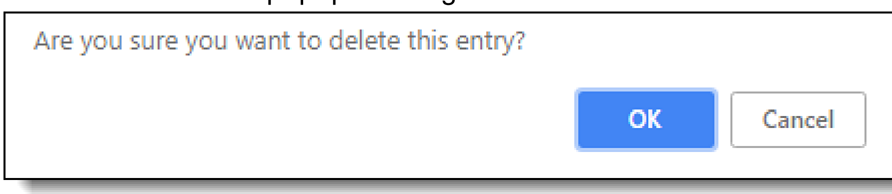
3. Click **Project Types** icon.



4. Scroll to the type you want to delete, click the **Options** gear icon and select **Delete** option.



   a. Click the **OK** on the popup message for the confirmation.



11.5.4. Project Templates

Some organizations have different projects but have same tasks and same team for those projects. Creating every project with same tasks and the team could be frustrating and some data can be missed out easily. In TimeLive, Administrator can create **Project Templates** as per requirement without any limitation to cater such needs. Not only the template allows Power User to create task and team, it also allows to create Milestones and attachments for the same. To access Project Template page navigate to **Admin Options > Project Template**.

Creating new Project Template:

Creating a New Project Template is as easy as creating a New Project. All the steps are the same including creating team and tasks.(Creating New Project)

After creating a new project template user can now use this template in creating a new project. User will observe that creating Project by using Project Template, Tasks, Team members, Milestones are added automatically (if created in Project Template) into the newly created project.

Editing the Project Template:

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.
3. Click **Project Templates**.
4. Scroll to the type you want to edit, click the **Options** gear icon and select **View Project** option.
5. Click on the **Options** gear icon and select *Edit Project*option.
6. Make the changes as per your requirements and click **Update**.

Disabling Project Template:

1. Login TimeLive with Admin rights.
2. Click **Admin Options** and scroll to **Project / Task Setup**.
3. Click **Project Templates**.
4. Scroll to the type you want to disable, click the **Options** gear icon and select **View Project** option.
5. Click on the **Options** gear icon and select **Edit Project** option.
6. Navigate to **Advance Tab**, click the switch **Disabled** to **Yes** then click **Update** button.

Deleting Project Type:

1. Login TimeLive with Admin rights.

2. Click **Admin Options** and scroll to **Project / Task Setup**.
3. Click **Project Templates**.
4. Scroll to the type you want to delete, click the **Options** gear icon and select **View Project** option.
5. Click on the **Options** gear icon and select **Delete** option.
6. Click **Yes** on the popup to confirm deletion of this project template.

11.6. Time Off Setup

11.6.1. Time Off Types

Time off types are categories that employees use to request or enter time off against. Commonly used time off types includes vacation time, sick time, jury duty, or banked time.

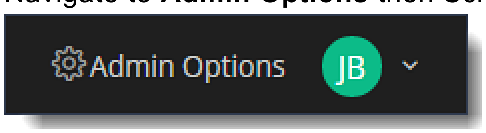
TimeLive provides few pre-defined Time off Types. An administrator can create their own Time off Type based on their own organizational needs.

Times off Types are of two types.

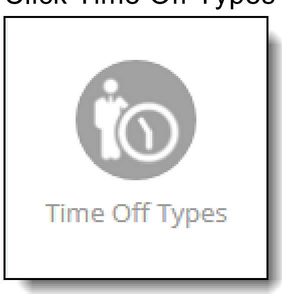
1. Time Off Types with “Request Required” enabled. A user will be required to submit **[Time Off Request]** for these **[Time Off Types]**.
2. Time Off Types with “Request Required” disabled. A user will be required to enter time off entries in timesheet directly for these **[Time Off Types]**.

Adding New Time Off Type:

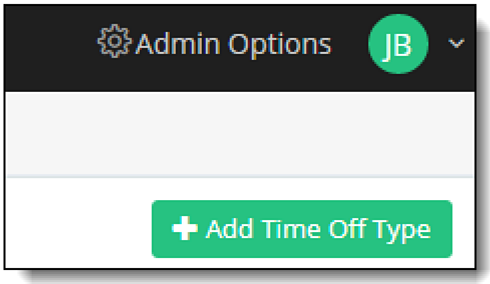
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.



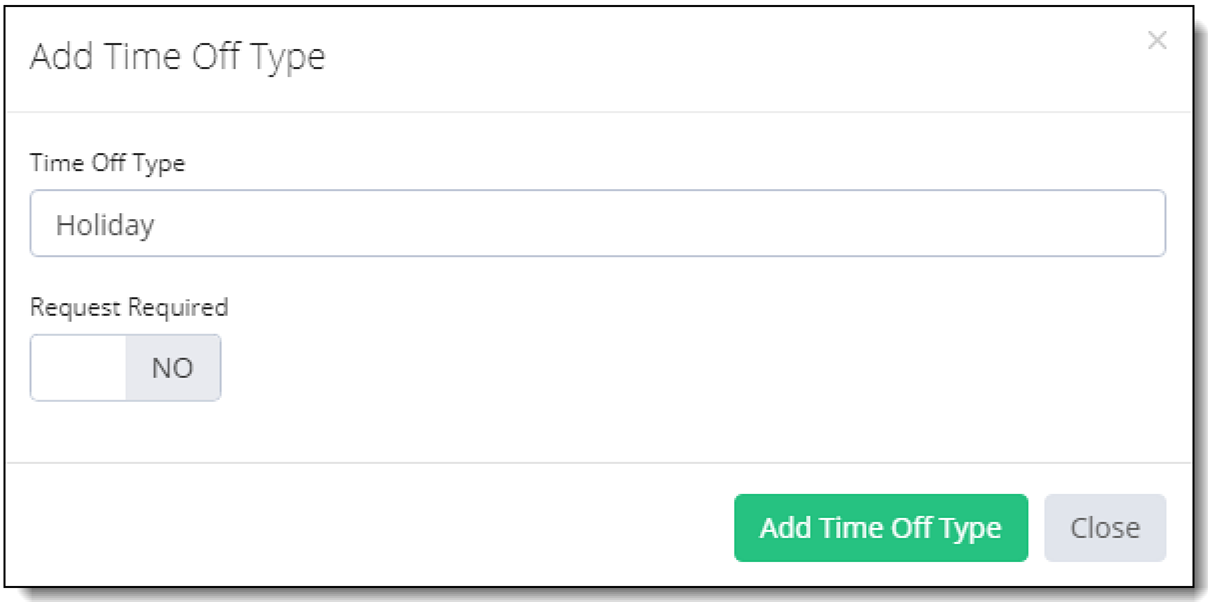
3. Click Time Off Types icon.



4. Click **+ Add Time Off Type** green button on the top-right of the screen.



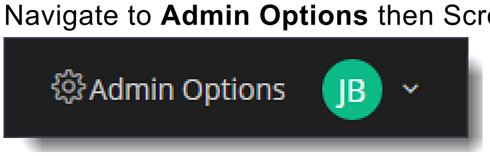
5. In the **Add Time Off Type** screen that appears:



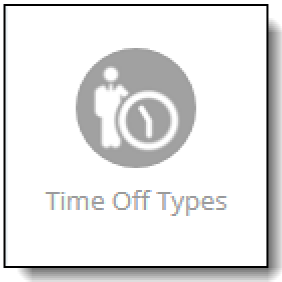
   a. Fill the Time Off Type Field with the Time Off name you want to set.
   b. If you want this Time Off Type to be requested from Time Off Request, Click on the box **Request Required**. If you keep this as **No** it will only show in My Timesheet of the Employee.
   c. Click on **Add Time Off Type** to add this Time Off Type.

Editing Time Off Type:

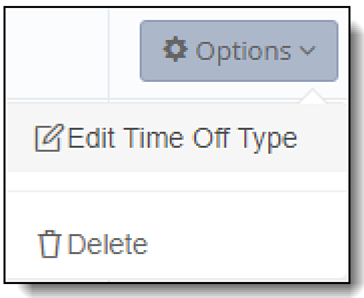
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.



3. Click Time Off Types icon.

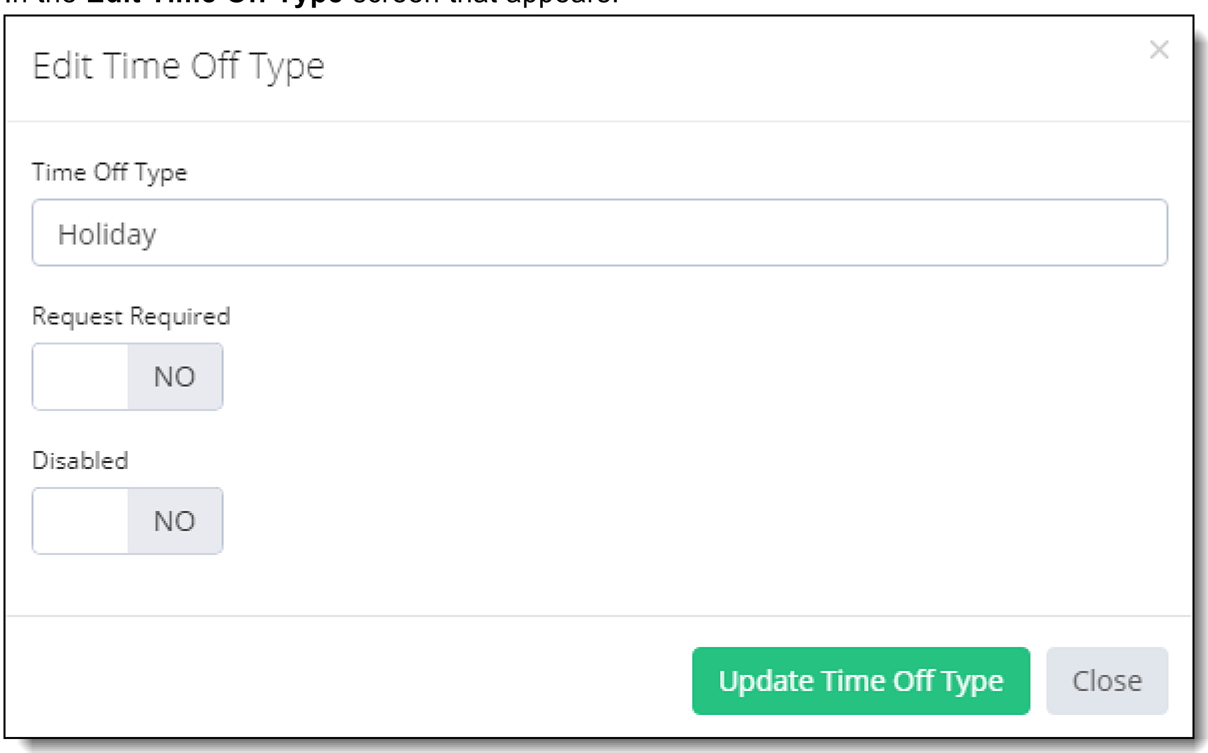


4. In **Time off Type List**, navigate to the Time Off Type you want to modify and click the **Option** gear icon against this Time Off Type.



   a. Click **Edit Time Off Type**.

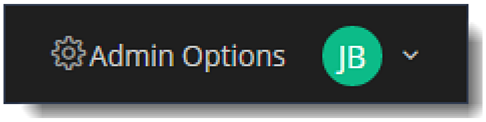
5. In the **Edit Time Off Type** screen that appears:



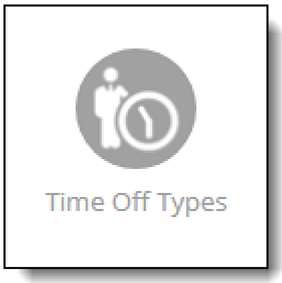
a. Update your required modification in **Time off Type Information** form.
b. Click on **Update Time Off Type** button to update this record.

Disabling Time Off Type:

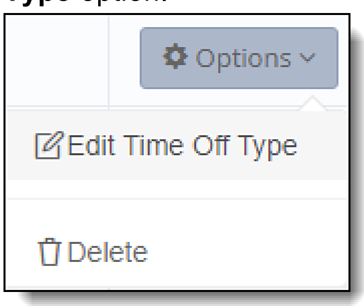
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.



3. Click Time Off Types icon.

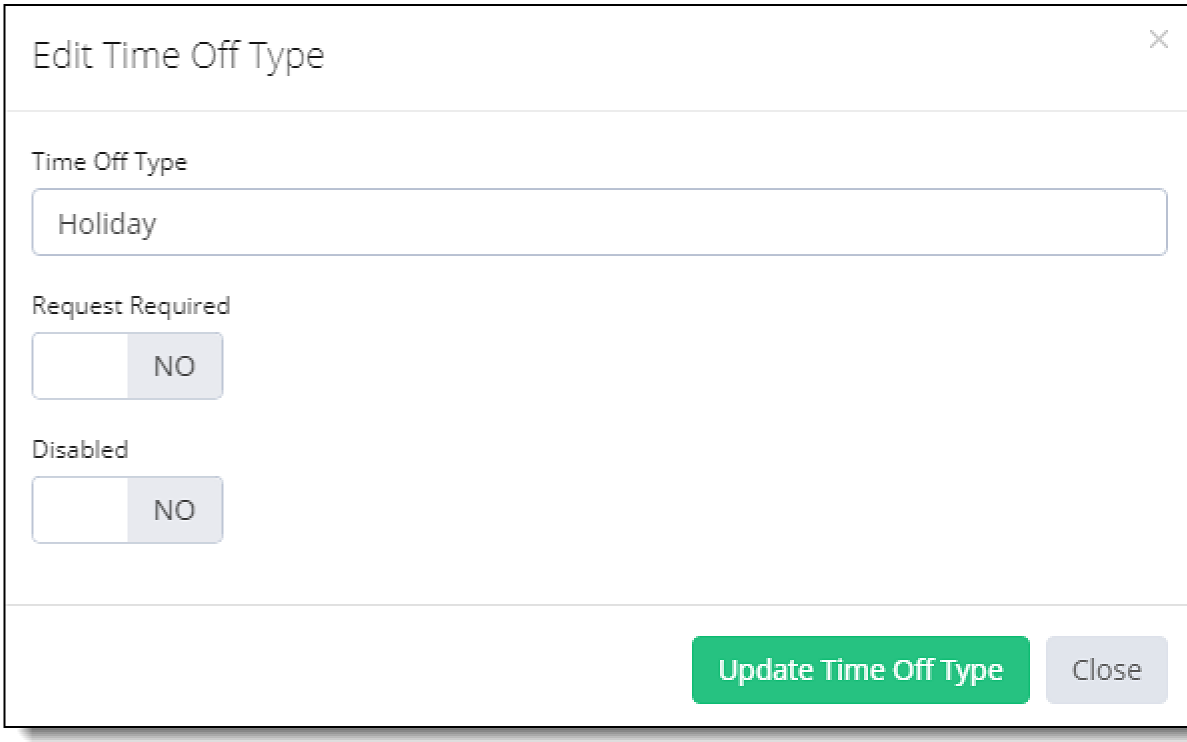


4. Scroll to the **Time Off Type** you want to disable, click the **Options** gear icon and select **Edit Time Off Type** option.



a. Click **Edit Time Off Type**.

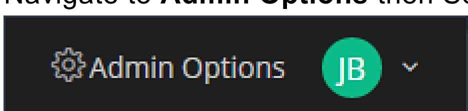
5. In the **Edit Time Off Type** screen that appears:



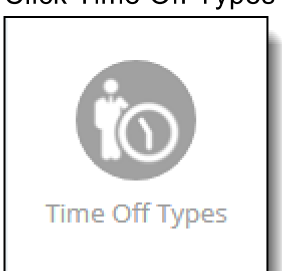
   a. Click on the **Disabled** Option in the popup.
   b. Click on **Update Time Off Type** button to update this record.

Deleting Time Off Type:

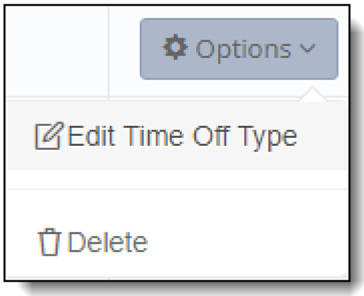
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.



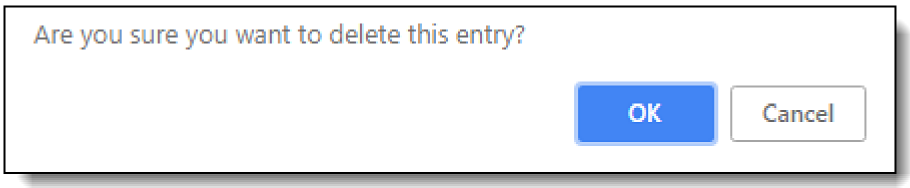
3. Click Time Off Types icon.



4. In **Time off Type List**, navigate to the Time Off Type you want to delete and click the **Option** gear icon against this Time Off Type.



   a. Click **Delete** option.
   b. Click on **OK** on delete confirmation dialog.



11.6.2. Time Off Policies

Time Off accrues is based on the accrual settings configured using **Time Off Policies** available in **Admin Options > Time off Policies**. Administrator can create “Time off Policies” with different configurations and can assign these types separately to each individual employee. Following settings can be configured using Time Off Policy option.

By default, TimeLive comes with one pre-populated time off policy name “Standard”. An Administrator can rename this Time Off Policy and can also add new time off policies as per organizational requirements.

In Time off Policy, there are several fields. Every field has its own work. Following are the fields of Time off Policy.

1. **INITIAL SET TO HOURS**: Time Off Available balance will be set to this “Initial Hours” when Time Off policy will be first executed by “Time Off Accrual Engine”
2. **EARNED HOURS**: Total hours which will be earned after the finish of every “Earned Period” (Weekly, Monthly, Yearly, Employee Anniversary Period).
3. **EARNED PERIOD**: TimeLive provides few pre-defined Earned Period which are.
    a. Never: When “Never” is used, the time off policy does not works.
    b. Each Week: When Administrator use this option, a user will get earn hours after every week by Time Off Policy scheduler.
    c. Each Month: When Administrator uses this option, a user will get earn hours after every month by Time Off Policy scheduler.
    d. Each Year: When Administrator uses this option, a user will get earn hours on every year by Time Off Policy scheduler.
    e. Each Year Anniversary: When Administrator uses this option, a user will get earn hours on each year anniversary according to hired date of the user.
4. **RESET AT**: There are four pre-defined **Reset At** which are.
    a. Never: When Never is selected, reset hours will not be set.
    b. Every Year: In this option, employee available balance will be set to “Reset Hours” value, every year by Time Off Policy scheduler.
    c. Every Week: In this option, employee available balance will be set to “Reset Hours” value, every week by Time Off Policy scheduler.
    d. Every Month: In this option, employee available balance will be set to “Reset Hours” value by Time Off Policy scheduler.

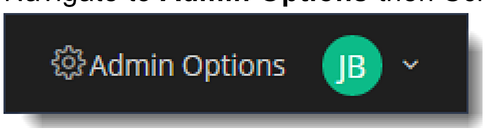
   e. Each Year Anniversary: In this option, employee available balance will be set to “Reset Hours” value at every anniversary date of the employee, according to hired date defined in employee form.
5. **RESET HOURS**: In this field, you can define your “Reset Hours” value which will be set to employee available balance at end of every “Reset At” period.
6. **MAXIMUM AVAILABLE**: Available balance of an employee cannot be increased more than maximum available hours, defined in [Time Off Policy].

How to reset Policy:

Reset button available in “Employee Time Off Page” which allows an administrator to reset all current period earned hours back to 0. And then at end of “Earned Period”, TimeLive Time Off scheduler automatically add new earned hours based on the policy defined for that particular employee.

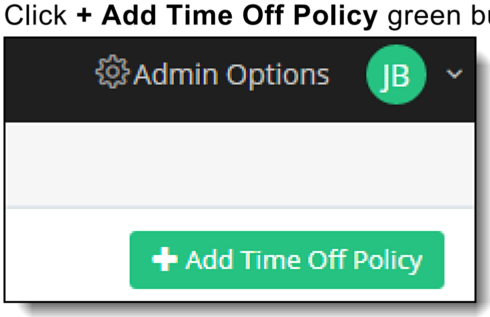
Add New Time Off Policy:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.

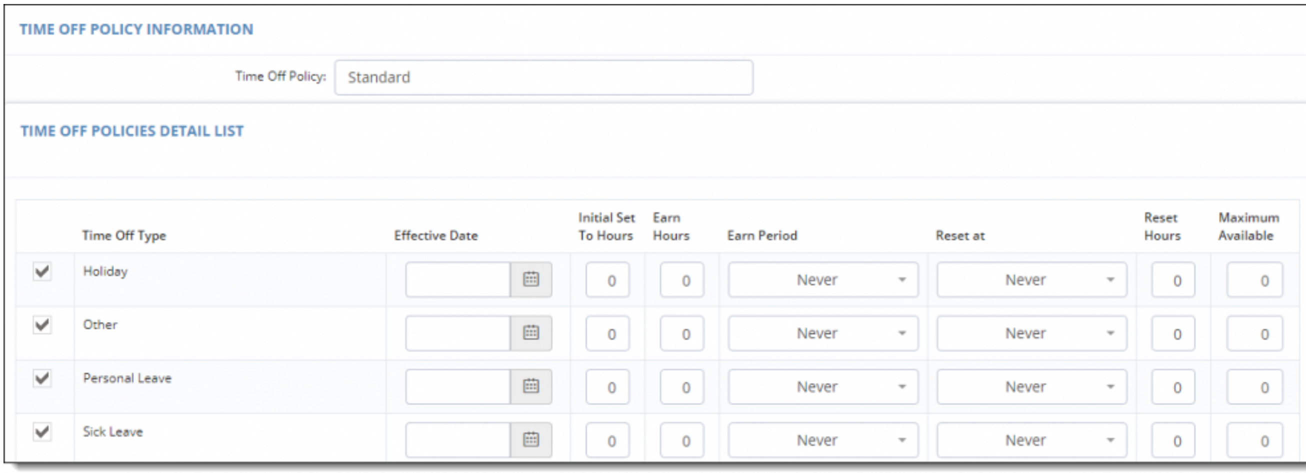


3. Click Time Off Policies icon.

4. Click **+ Add Time Off Policy** green button on the top-right of the screen.



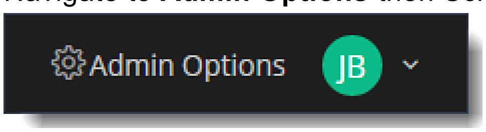
5. In the **Add Time Off Policy** screen that appears:



   a. Give a unique name to this Policy as you will have to assign this policy to the employees who will be entitled to this policy from Employee Profile.
   b. Fill the following fields as described above and then click **Add** green button.

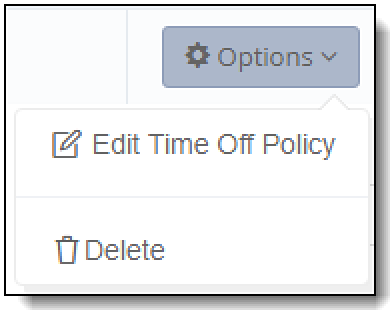
Edit Time Off Policy:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.



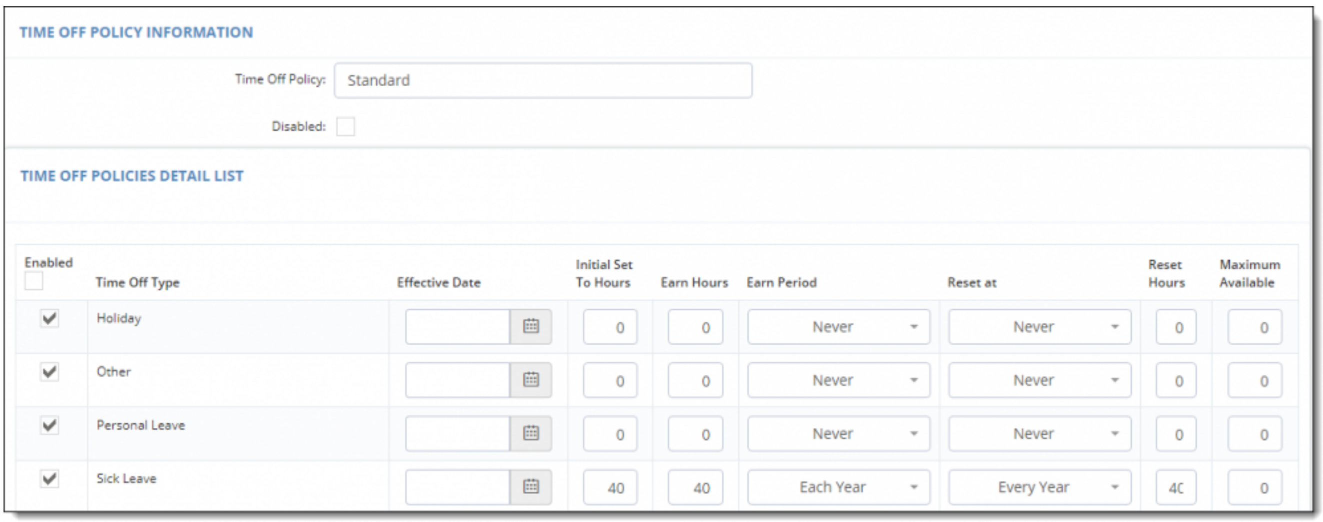
3. Click Time Off Policies icon.

4. In **Time off Policy List**, navigate to the Time Off Policy you want to Edit and click the **Option** gear icon against this Time Off Policy.



   a. Click **Edit Time Off Policy** option.

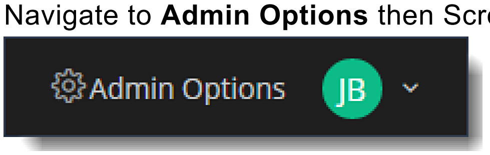
5. In the **Edit Time Off Policy** screen that appears:



   a. Make the desired changes.
   b. Click on **Update** button to update this record.

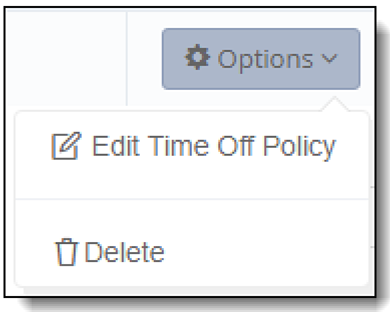
Disabling Time Off Type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.



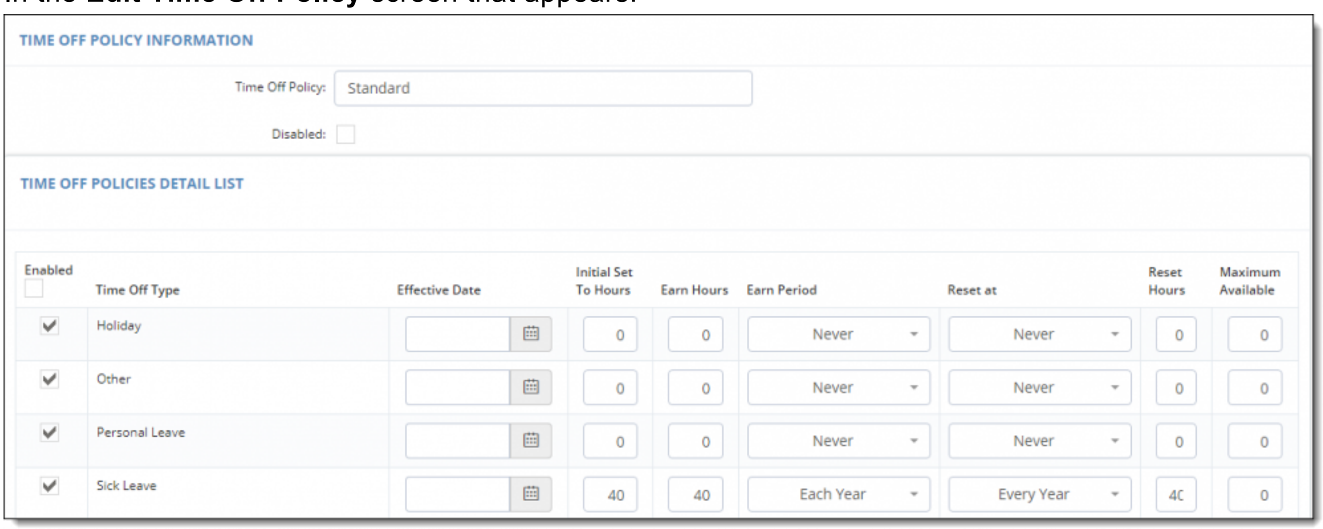
3. Click Time Off Policy icon.

4. In **Time off Policy List**, navigate to the Time Off Policy you want to **Disable** and click the **Option** gear icon against this Time Off Policy.



   a. Click **Edit Time Off Policy** option

5. In the **Edit Time Off Policy** screen that appears:

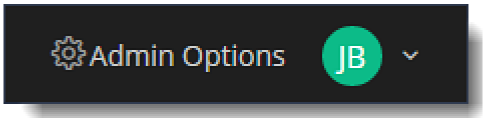


   a. Enable the check **Disabled**.
   b. Click on **Update** button to update this record.

Deleting Time Off Policy:

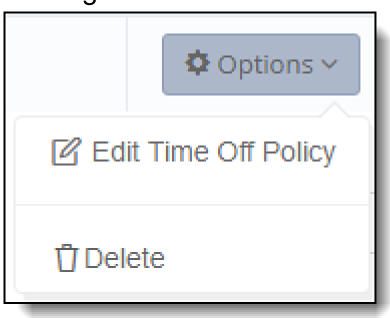
1. Login TimeLive with Admin Credentials.

2. Navigate to **Admin Options** then Scroll to **Time Off Setup**.

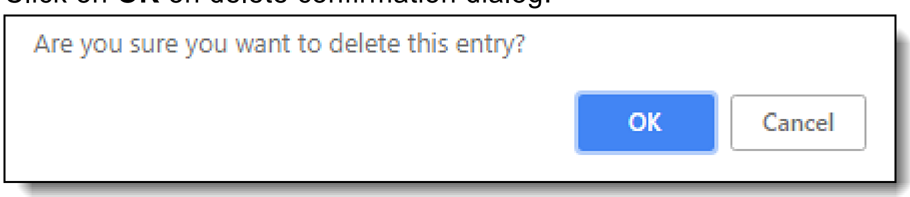


3. Click Time Off Policy icon.

4. In **Time off Policy List**, navigate to the Time Off Policy you want to **Delete** and click the **Option** gear icon against this Time Off Policy.



   a. Click **Delete** option.
   b. Click on **OK** on delete confirmation dialog.



11.7. Expense Setup

11.7.1. Expense Types Setup

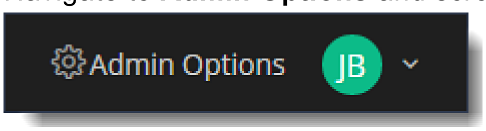
Expense types are the different categories of expenses. The application administrator can create different expense types according to their organization needs.

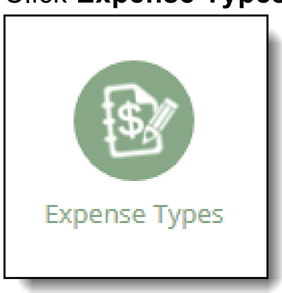
Expense Type also allows the administrator to define either that expense types require quantity input in expense entry. In case of “Car Mileage” where users are required to enter “Miles/Kilometer” in expense entry, an administrator can enable quantity field with its own defined caption in expense type.

A system administrator can setup Expense Types **Admin Options > Expense Types**

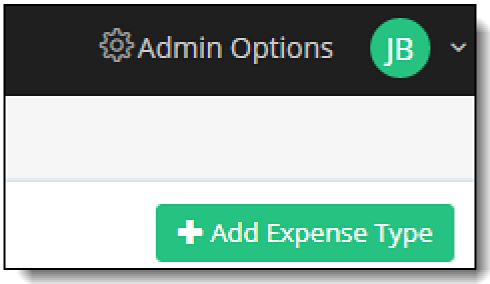
Adding new Expense type:

To add new expense type:

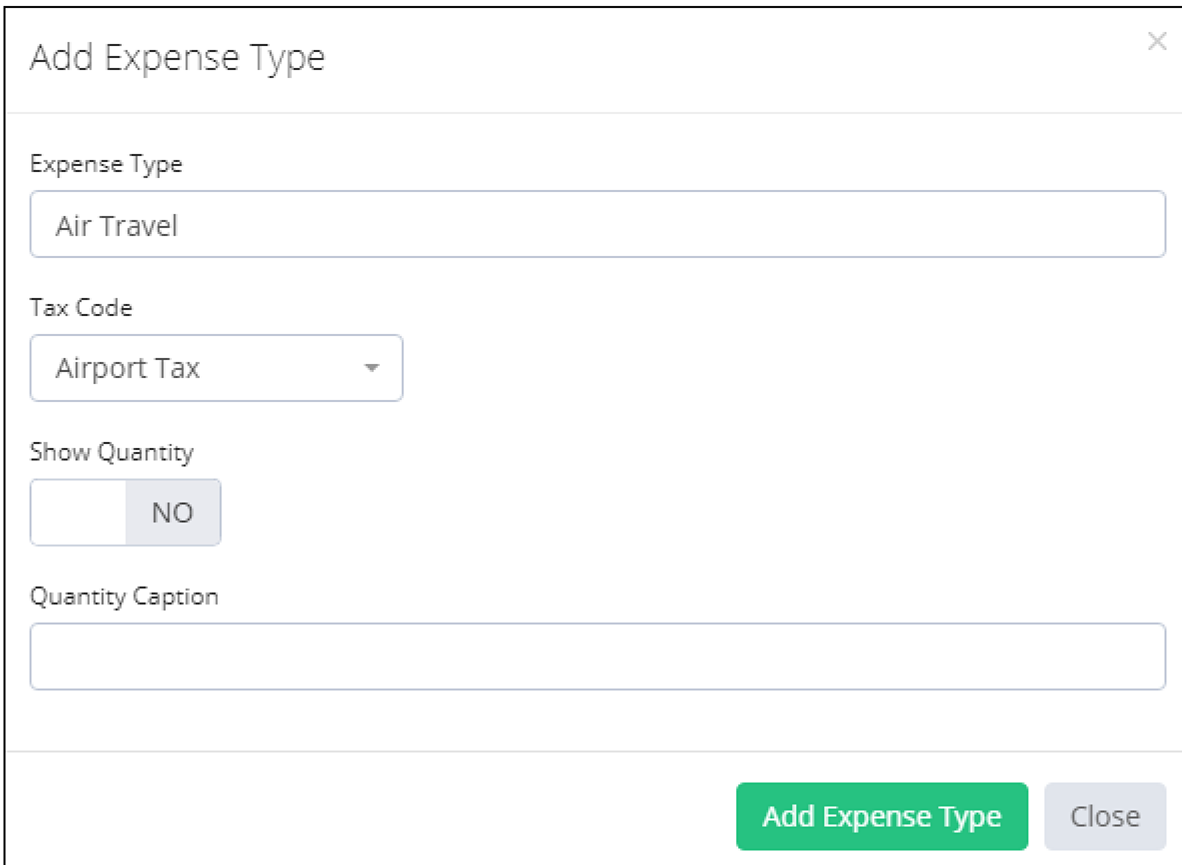
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.


3. Click **Expense Types** icon.


4. Click **+ Add Expense Type** button.



5. In the **Add Expense Type** screen that appears:

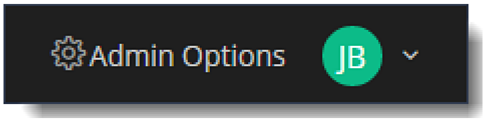


   a. Enter expense type title in "Expense Type Field" form which you want to add.
   b. Select Tax Code (Tax codes define different types of tax, that can be applied to the expense codes defined in your system. An administrator can create region based taxes in order to support tax calculation based on user own region.)
   c. Enable **Show Quantity** Box if an administrator wants to input quantity in expense entry for this particular expense type. Example of quantity field. is **Car Mileage** where users are required to enter **Miles/Kilometer** in expense entry.
   d. Enter **Quantity Caption** in case if this expense type is quantity based.

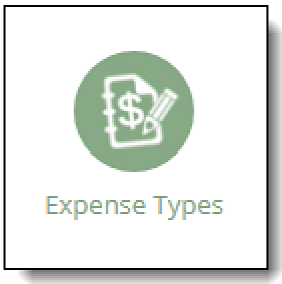
e. Click on **Add Expense type** button to add this expense type in TimeLive system.

Editing Expense Type:

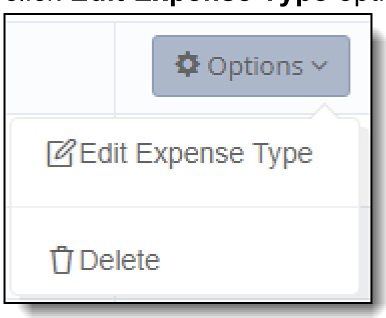
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.



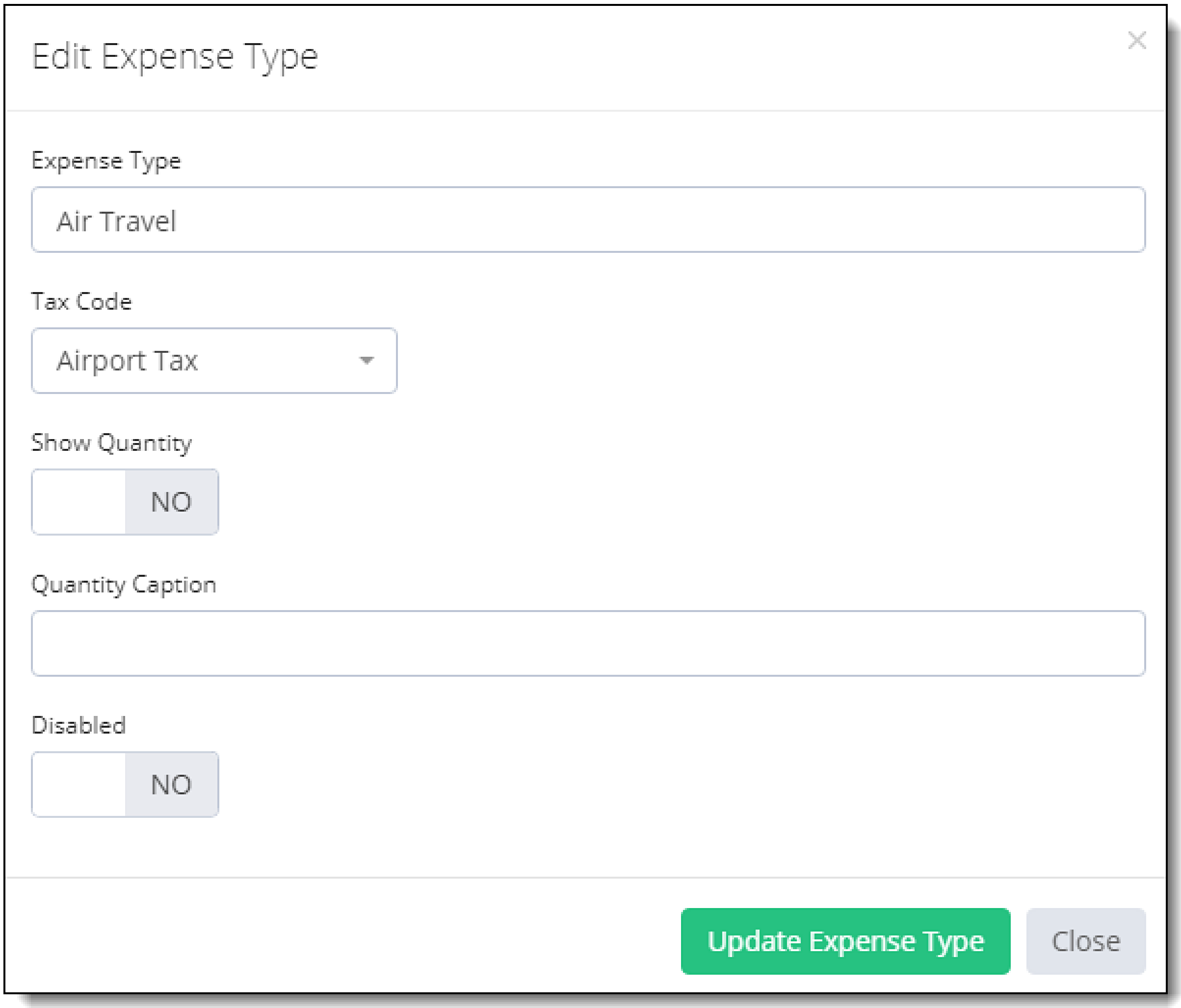
3. Click **Expense Types** icon.



4. In **Expense Type List**, scroll to the Expense type you want to edit and click **Options** gear icon then click **Edit Expense Type** option.



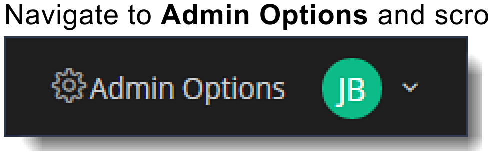
5. In the **Edit Expense Type** screen that appears:



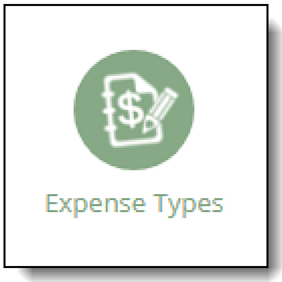
   a. Make changes as per your requirement.
   b. Click **Update Expense Type** button to update changes.

Deleting Expense Type:

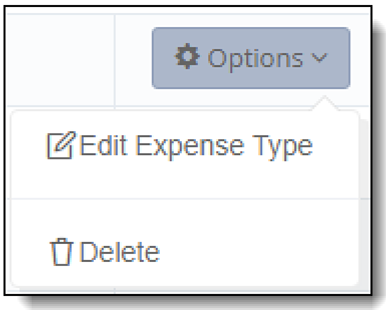
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.



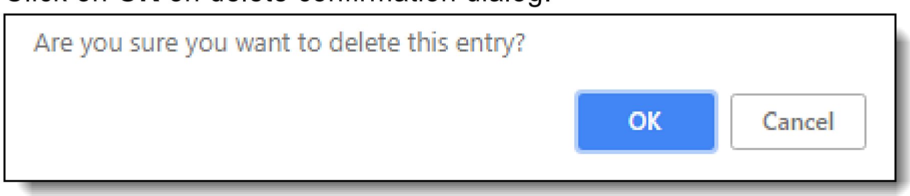
3. Click **Expense Types** icon.



4. In **Expense Type List**, scroll to the Expense type you want to delete and click **Options** gear icon.



   a. Click **Delete** option.
   b. Click on **OK** on delete confirmation dialog.



11.7.2. Expense Code Setup

Expense codes define the different costs incurred for business activities and purchases. You can create your required expense codes using Expenses option of administration.

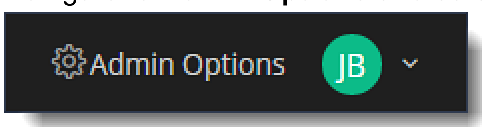
Adding expense codes will allow you to identify the types of expenses that may be entered against specific projects. Expense codes will also allow you to track the kinds of expenses being recorded by TimeLive users.

A system administrator can setup expenses name using **Admin Options > Expenses**

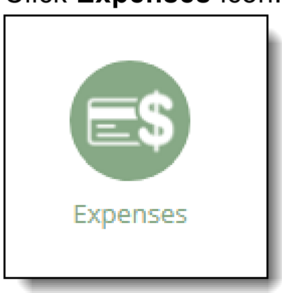
Adding new expenses:

To add a new expense, follow these instructions.

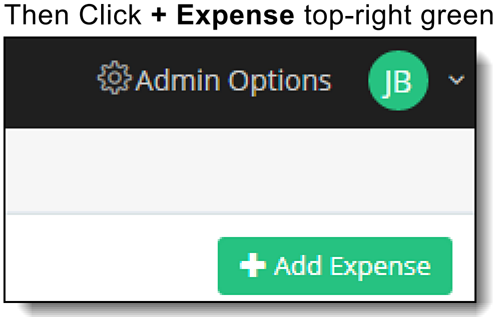
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.



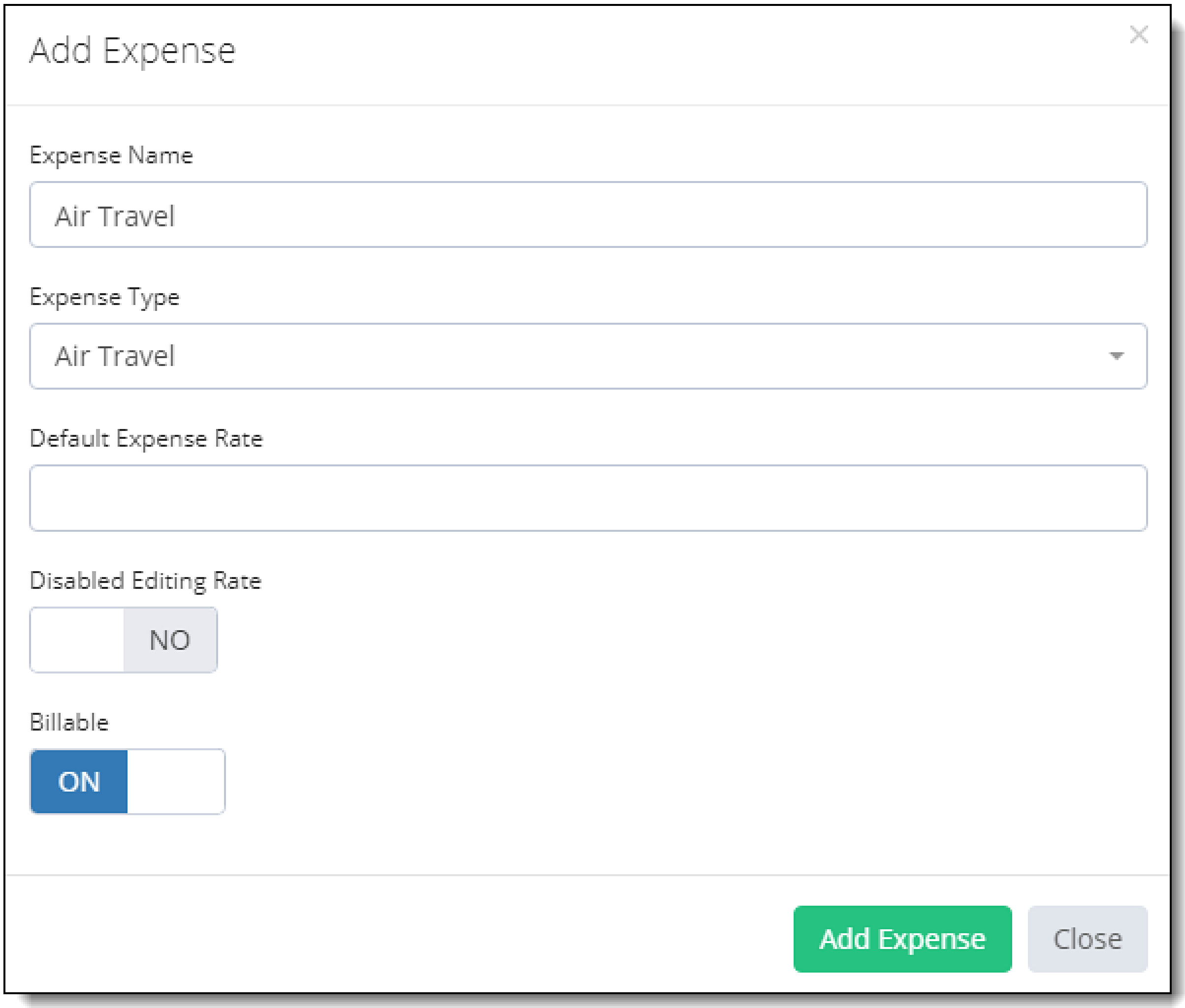
3. Click **Expenses** icon.



4. Then Click **+ Expense** top-right green button.



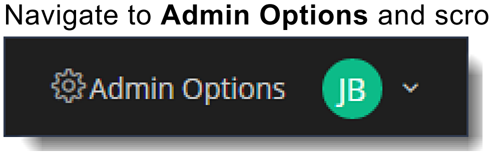
5. In the **Add Expense** screen that appears:



   a. Enter default expense rate for expense codes which are quantity based. Example of default expense rate is mileage rate for expense **Car Mileage**. An administrator can define the rate for that expense here.
   b. If an administrator doesn't want the user to change rates in expense entry, an administrator can disable milage rate editing in expense entry by users, by enabling the **Disabling Editing Of Rate** Box.
   c. Click on **Add Expense** to add this expense.

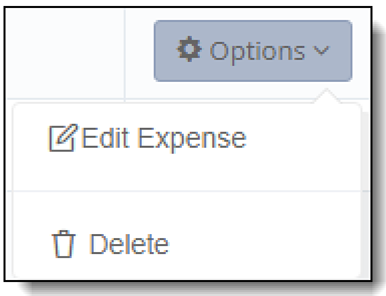
Editing Expense:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.

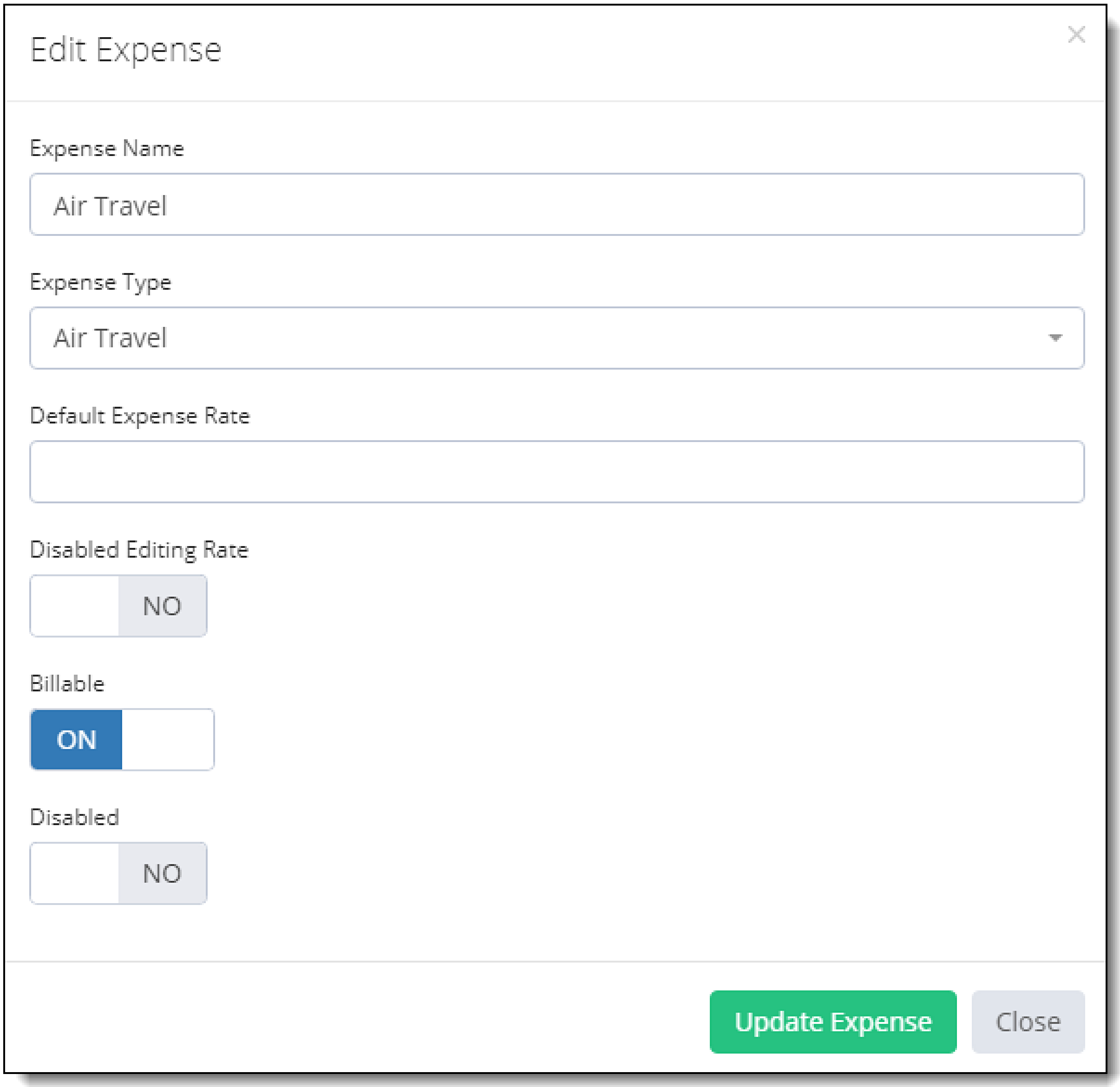


3. Click **Expenses** icon.

4. In **Expense List**, click on **Gear Icon** of **Options** to the one which you want to Edit then click **Edit Expense** option.



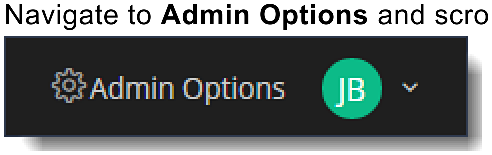
5. In the **Edit Expense** screen that appears:



   a. Update your required modification in [Expense Information] form.
   b. Click **Update Expense** green button to update this record.

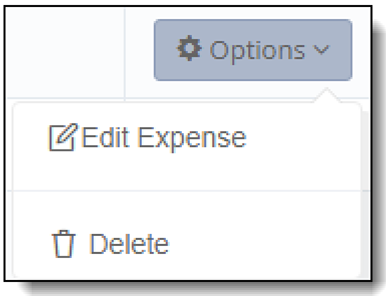
Deleting Expense:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.



3. Click **Expenses** icon.

4. In [Expense List], click on **Gear Icon** of **Options** to the one which you want to delete.



   a. Click **Delete** option.
   b. Click on **OK** on delete confirmation dialog.

11.7.3. Payment Methods

Payment methods specify how expenses were paid. TimeLive includes a set of predefined payment methods that you can edit to meet the needs of your organization. You can also create new payment methods.

The system administrator can setup organization departments using **Admin Options > Payment Methods**.

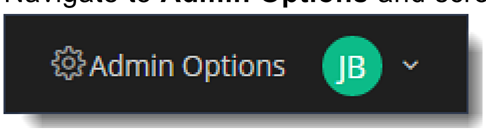
Setting up Payment Methods:

By default, four default payment methods are available in TimeLive with name “American Express,Cash, Master Card,Visa”. Administrator can rename these payment methods and can also add new payment methods.

Adding new payment method:

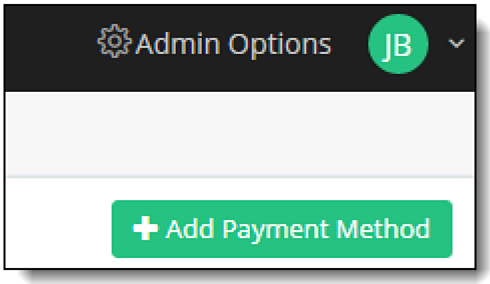
To add new payment method, follow these instructions.

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.

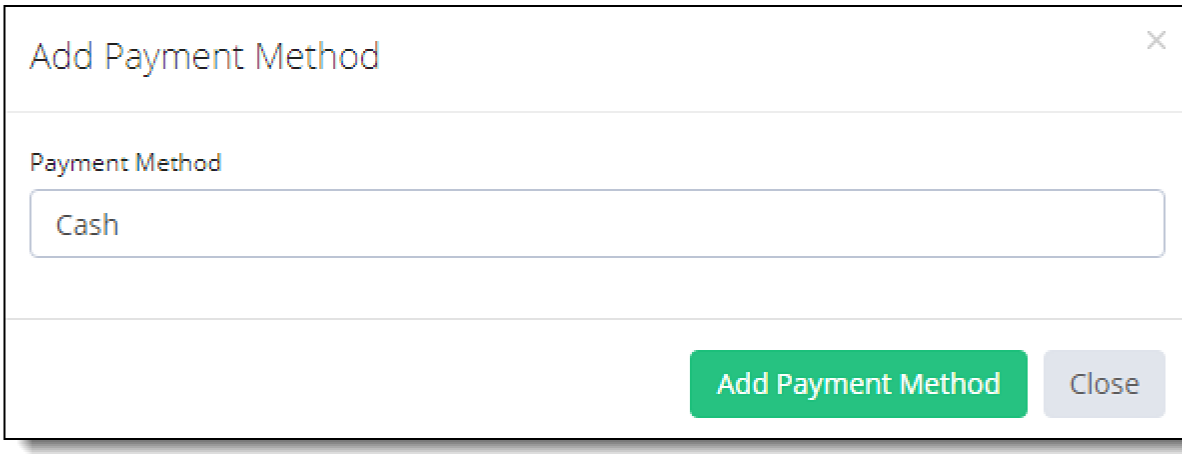


3. Click **Payment Method** icon.

4. Click **+ Add Payment Method** button.



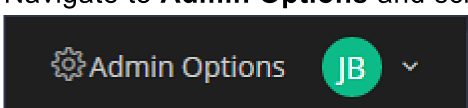
5. In the **Payment Method** screen that appears:



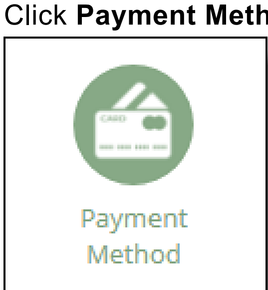
   a. Fill the Payment Method form.
   b. Click **Add Payment Method** button to add this record.

Editing Payment Method:

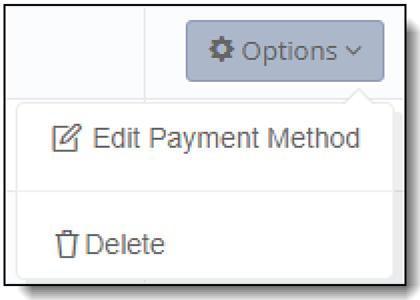
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.



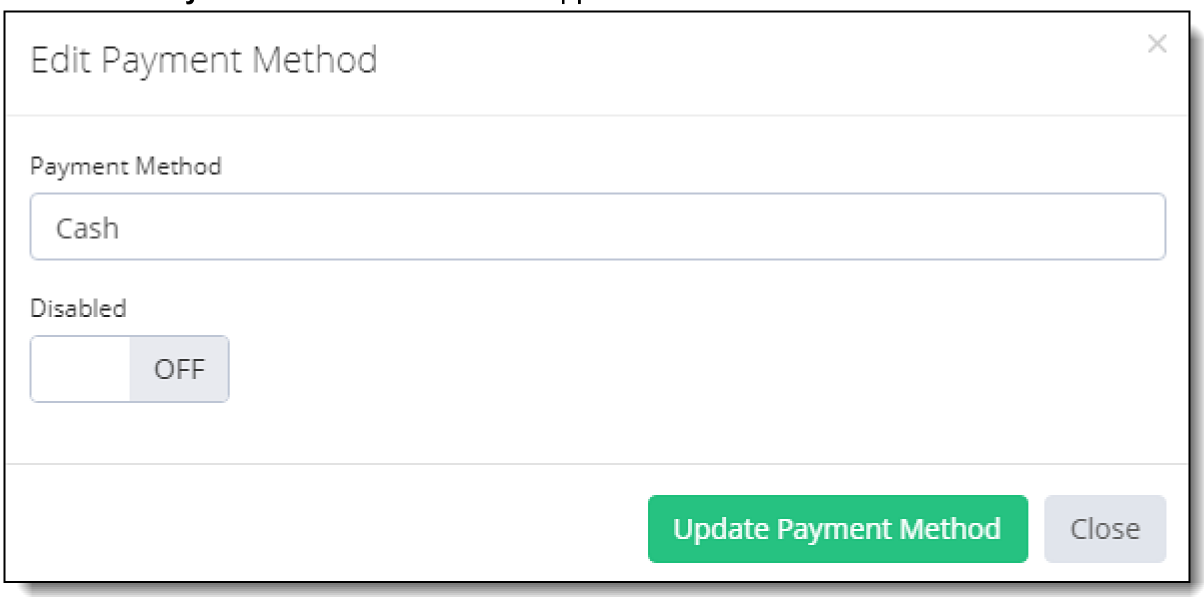
3. Click **Payment Method** icon.



4. In **Payment Method List**, click on **gear icon** of **Options** against the Payment Method you want to edit and click **Edit Payment Method** option.



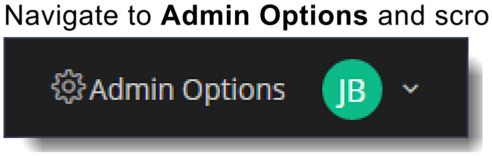
5. In the **Edit Payment Method** screen that appears:



   a. Update your required modification in **[Payment Method Information]** form.
   b. Click on **Update Payment Method** button to update this record.

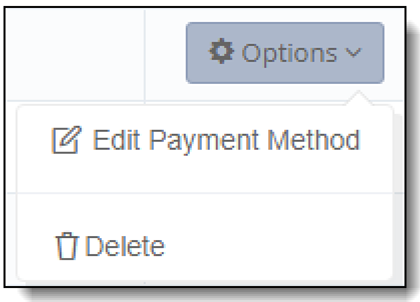
Deleting Payment Method:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Expense Setup**.

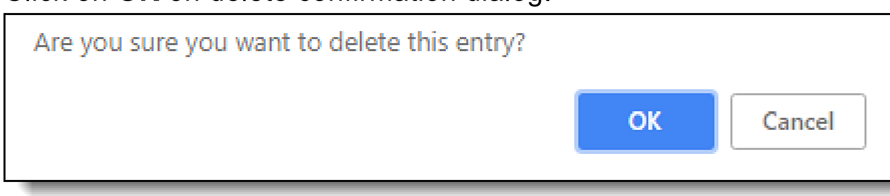


3. Click **Payment Method**

4. In **Payment Method List**, click on **gear icon** of **Options** against the Payment Method you want to delete.



   a. Click **Delete** option.
   b. Click on **OK** on delete confirmation dialog.



11.8. Currencies and Tax

11.8.1. Currencies

TimeLive allows multiple currencies for each organization. This feature is under **Admin Options > Currencies**.

Currencies play a very important role in Timesheet and Expense Billing. It also manages rates fluctuations for any length of time under multiple dates ranges for each currency.

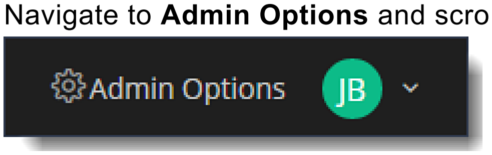
By default, most commonly used currency are predefined in TimeLive,

- **Australian Dollar**
- **Canadian Dollar**
- **Euro**
- **Japanese Yen**
- **Swiss Franc**
- **Pound**
- **USA Dollar**

But TimeLive is not limited to these only. User can add/remove any currency according to their need. By default, **USA Dollar** is the **Base Currency** and can be changed by the Administrator. **Identification of Base Currency is that its conversion rate would be 1**. An administrator can and should check the conversion rates and date ranges of all currencies in order to avoid any miscalculation.

Adding New Currency Type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.



3. Click **Currencies** icon.

4. Click on the button **+Add Currency** at the top-right of the screen.
5. Select the new currency from the **Name drop-down field** and the **Exchange Rate** in the 2nd field(as per the base currency. If you are adding the base currency then set it to 1 and change already existing currency exchange rates).
6. Click on **Add Currency** to add.

Disabling Currency Type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.
3. Click **Currencies**.
4. In **Currency List**, click on the **gear icon** on the right of the Currency you want to disable and click **Edit Currency** option.
5. Click the **Disabled** labeled switch to **Yes**.
6. Click **Update Currency** button.

Adding new Exchange Rate with the different date range in the existing Currency:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.
3. Click **Currencies**.
4. In **Currency List**, click on the **gear icon** on the right of the Currency you want to add exchange rate and click **History** option.
5. Click on the button **+Add Exchange Rate** at the top-right of the screen.
6. Fill the Exchange Rate field and select the exchange rate date range as per your requirement.
7. Click **Add Exchange Rate** button.

Deleting existing Exchange Rate of a currency:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.

3. Click **Currencies**.
4. In **Currency List**, click on the **gear icon** on the right of the Currency you want to add exchange rate and click **History** option.
5. Click on the **gear icon** on the right of the Exchange Rate you want to delete and select **Delete** option.
6. Click **Yes** on the Dialog Box for confirmation.

NOTE:

There are two main issues which should always be on the check that.

1. No Date ranges should overlap in the same currency type.
2. During updating/managing existing rate, Administrator/Power User must enable **Update all records within this range** check to avoid any miscalculations.

11.8.2. Tax Codes

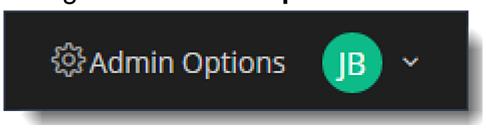
Tax codes define different types of tax, that can be applied to the expense codes defined in your system.

When you add a tax code, you will be required to enter its tax formula. This formula, which is based on the net amount of the expense, determines how TimeLive calculates the tax.

System administrator can setup organization departments using **Admin Options > Tax Codes**

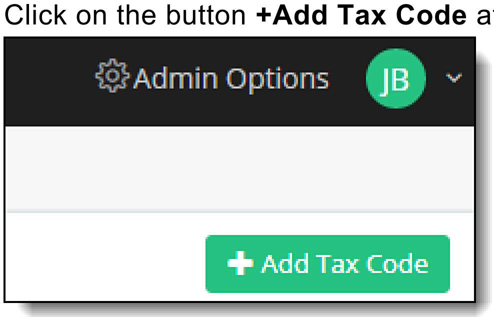
Adding New Tax Code:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.

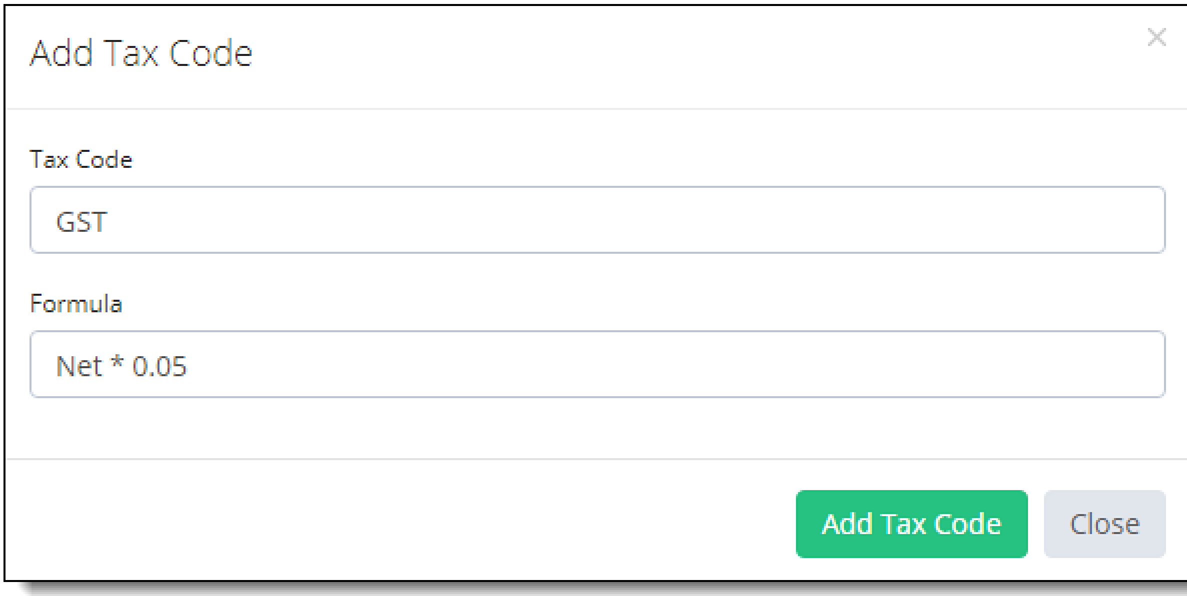


3. Click **Tax Code** icon.

4. Click on the button **+Add Tax Code** at the top-right of the screen.



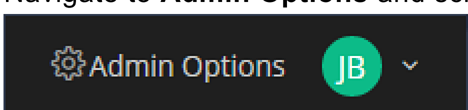
5. In the **Add Tax Code** screen that appears:



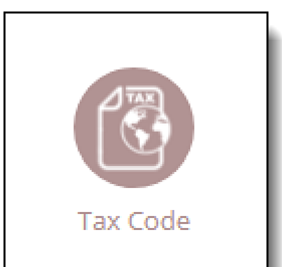
   a. Enter new **Tax Code** name in the 1st field and the formula in the 2nd field.
   b. Click on **Add Tax Code** button to add a record.

Editing Tax Code:

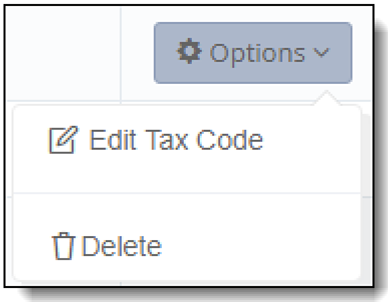
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.



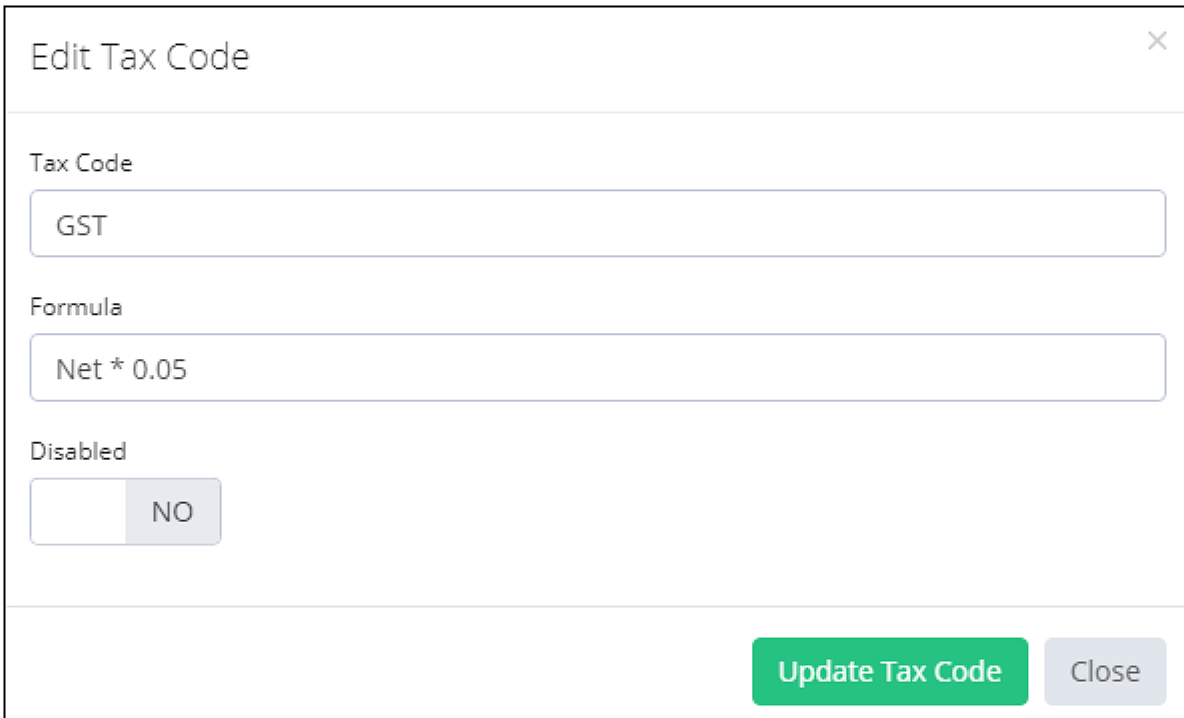
3. Click **Tax Code** icon.



4. In **Tax Code List**, click on the **gear icon** on the right of the Tax code you want to edit and click **Edit Tax Code** option.



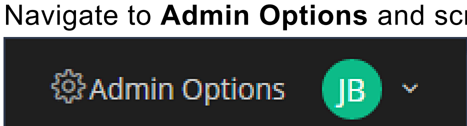
5. In the **Edit Tax Code** screen that appears:



   a. Update your required modification in **[Tax Code Information]** form.
   b. Click on **Update Tax Code** button to update this record.

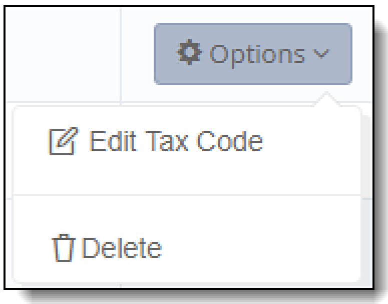
Deleting Tax Code:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.

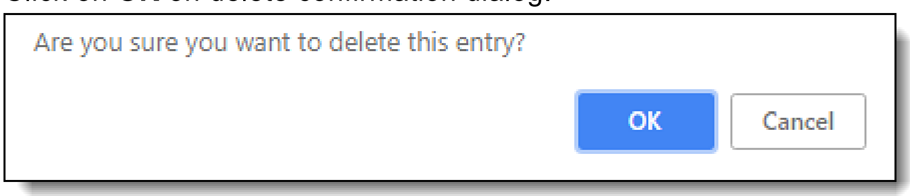


3. Click **Tax Code** icon.

4. In **Tax Code List**, click on the **gear icon** on the right of the Tax code you want to delete.



   a. Click **Delete** option.
   b. Click on **OK** on delete confirmation dialog.

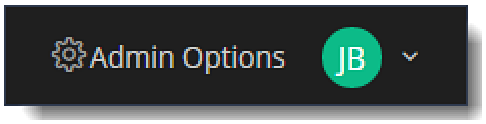


11.8.3. Tax Zone

Different companies have different Tax calculations in different regions/countries/states/zones. To cater this, an administrator can create different Tax Zones as per the requirement. Tax Zone is the key point when setting up Tax Codes. Tax Codes will be different for each Tax Zone.

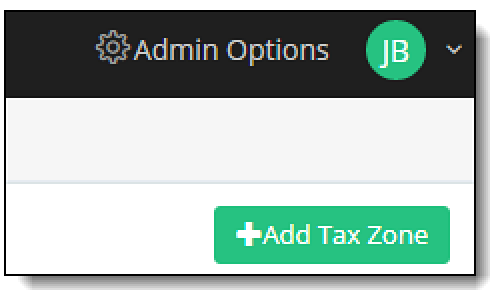
Adding New Tax Code:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.

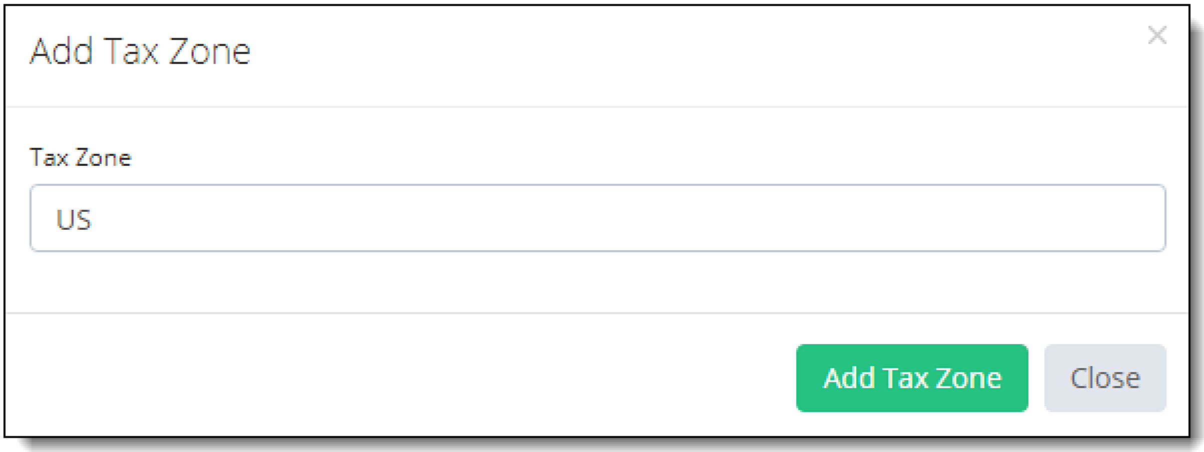


3. Click **Tax Zone** icon.

4. Click on the button **+Add Tax Zone** at the top-right of the screen.



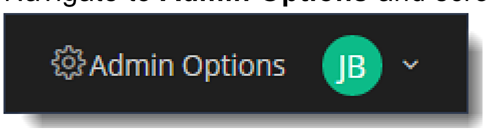
5. In the **Add Tax Zone** screen that appears:



   a. Fill the Tax Zone Field.
   b. Click **Add Tax Zone** button to a record.

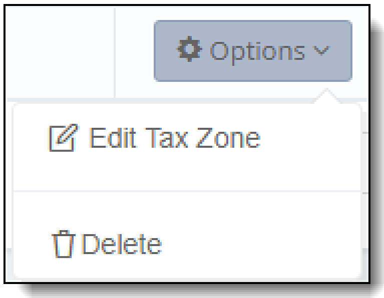
Editing Tax Zone:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.

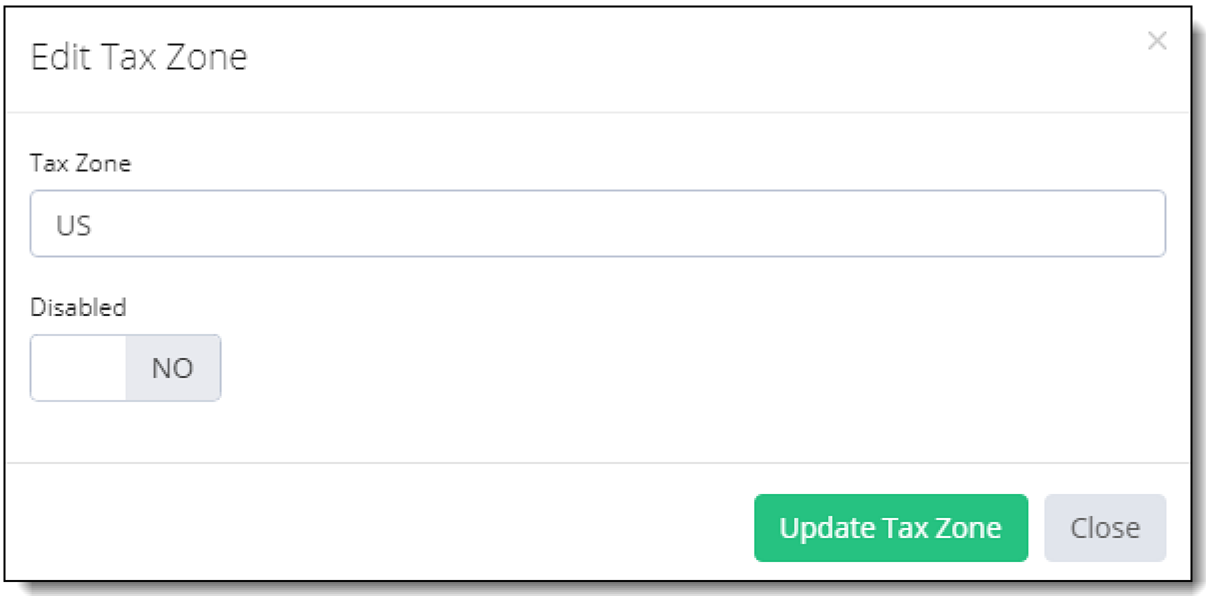


3. Click **Tax Zone** icon.

4. In **Tax Zone List**, click on the **gear icon** on the right of the Tax code you want to edit and click **Edit Tax Zone** option.



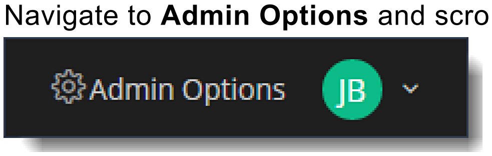
5. In the **Edit Tax Zone** screen that appears:



   a. Update your required modification in **[Tax Zone Information]** form.
   b. Click on **Update Tax Zone** button to update this record.

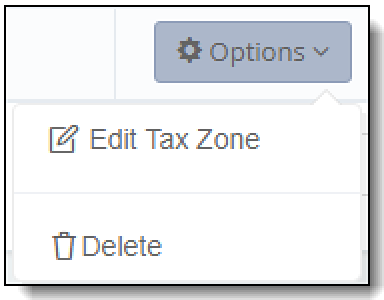
Deleting Tax Zone:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Currencies and Tax**.



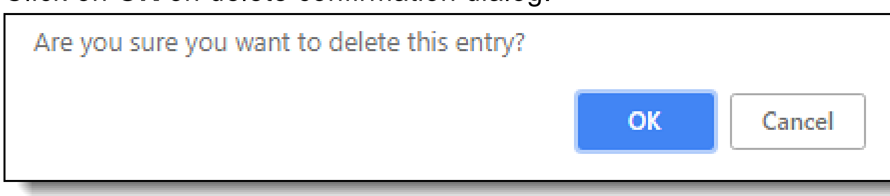
3. Click **Tax Zone** icon.

4. In **Tax Zone List**, click on the **gear icon** on the right of the Tax Zone you want to delete.



   a. Click **Delete** option.

5. Click on **OK** on delete confirmation dialog.



11.9. Billing Setup

11.9.1. Billing Types

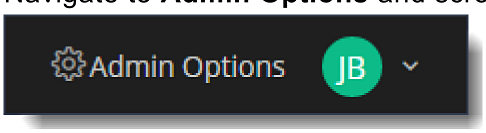
Billing types are system configurable types which you can define and TimeLive allow System Administrator to create own billing types. TimeLive provides few pre-defined billing types which can be renamed or edited according to organization needs.

System Administrator can set up billing types using **Admin Options > Billing Types**
There are two types of billing category which you can set up using billing type setup.

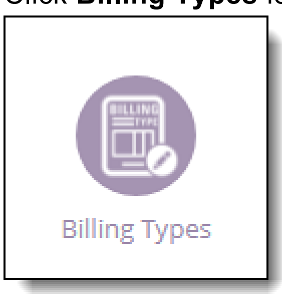
1. Employee
    a. Hourly
    b. Monthly
2. Project / Client
    a. Per Hour
    b. Fixed Bid

Adding New Billing Type:

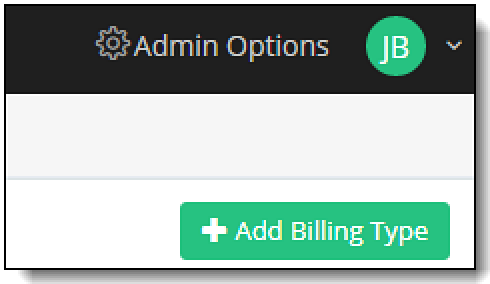
1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Billing Setup**.



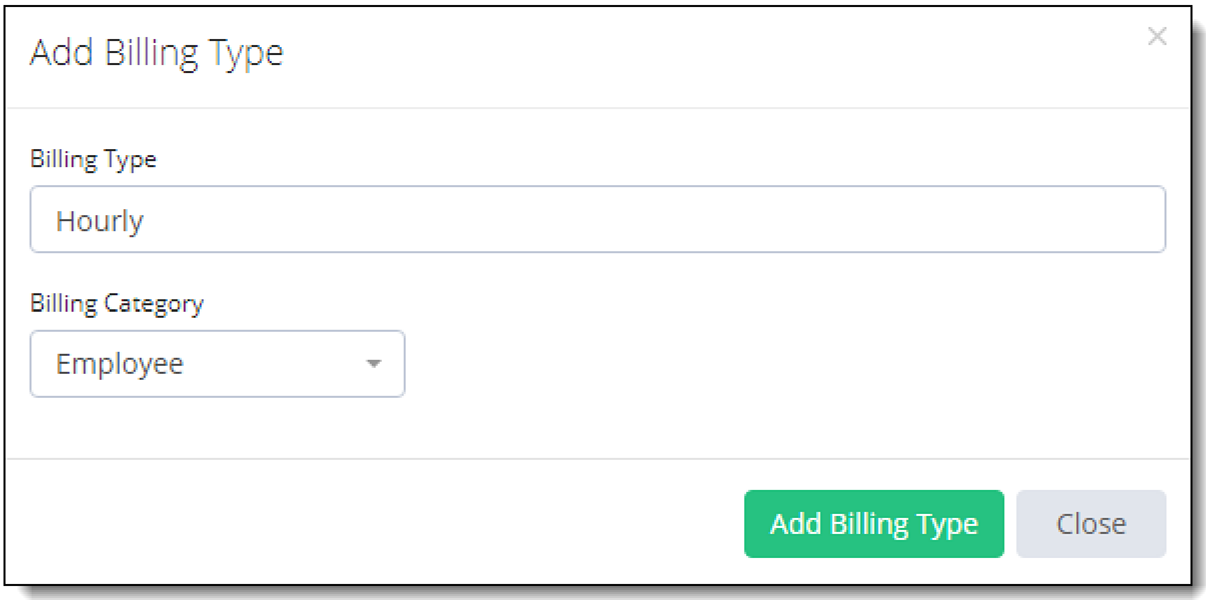
3. Click **Billing Types** icon.



4. Click on the button **+Add Billing Type** at the top-right of the screen.



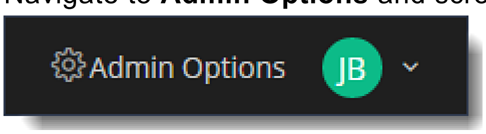
5. In the **Add Billing Type** screen that appears:



   a. Enter the name of the **Billing Type** in the first field.
   b. Select billing category (select Employee if you are defining billing type of employee billing. Select Project/Client if you want to add billing type for the project).
   c. Click on **Add Billing Type** to add this to the system.

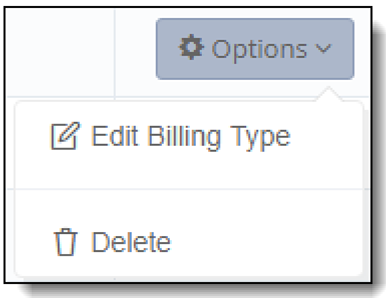
Editing Billing Type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Billing Setup**.

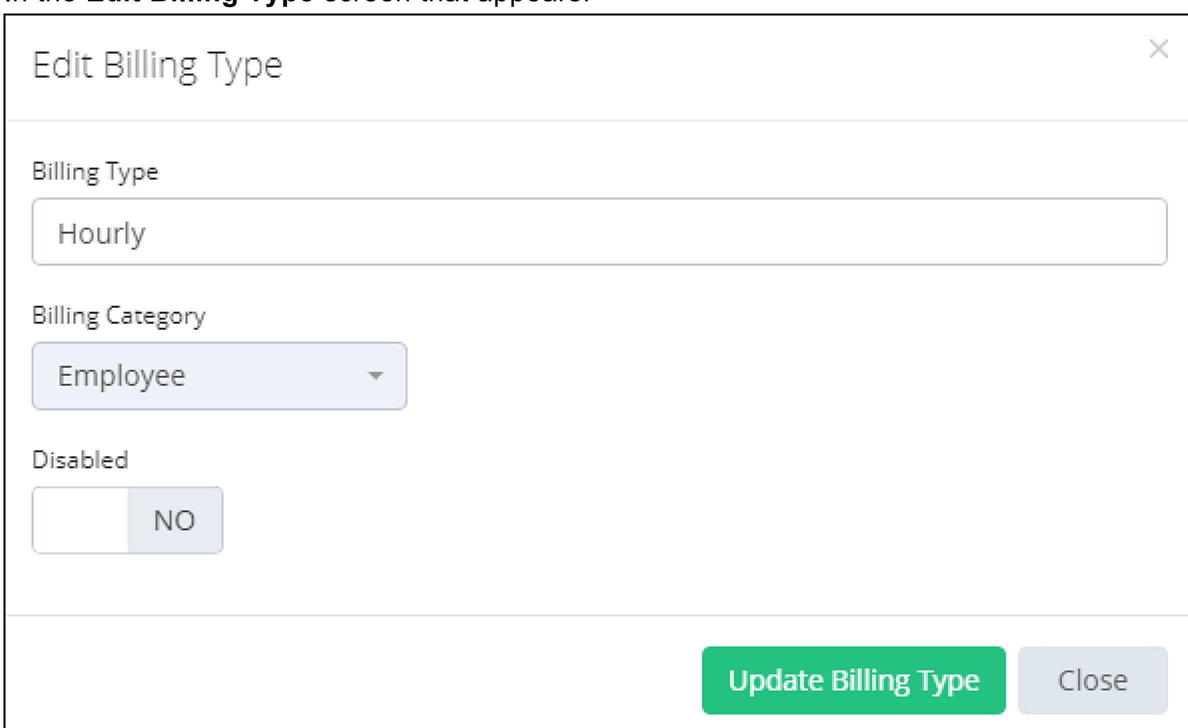


3. Click **Billing Types** icon.

4. In **Billing type List**, click on **Options** gear icon against the type which you want to edit and select **Edit Billing Type**.



5. In the **Edit Billing Type** screen that appears:

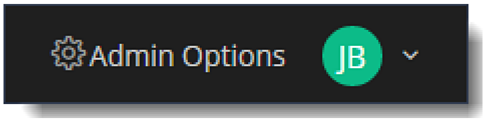


    a. Update your required modification in **Billing Type** form.

b. Click on **Update Billing Type** button to update this record.

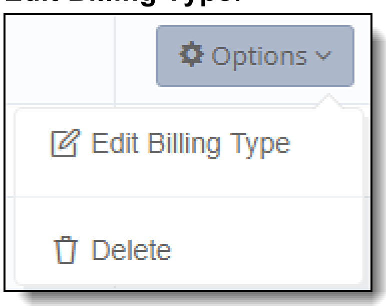
Disabling Billing Type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Billing Setup**.

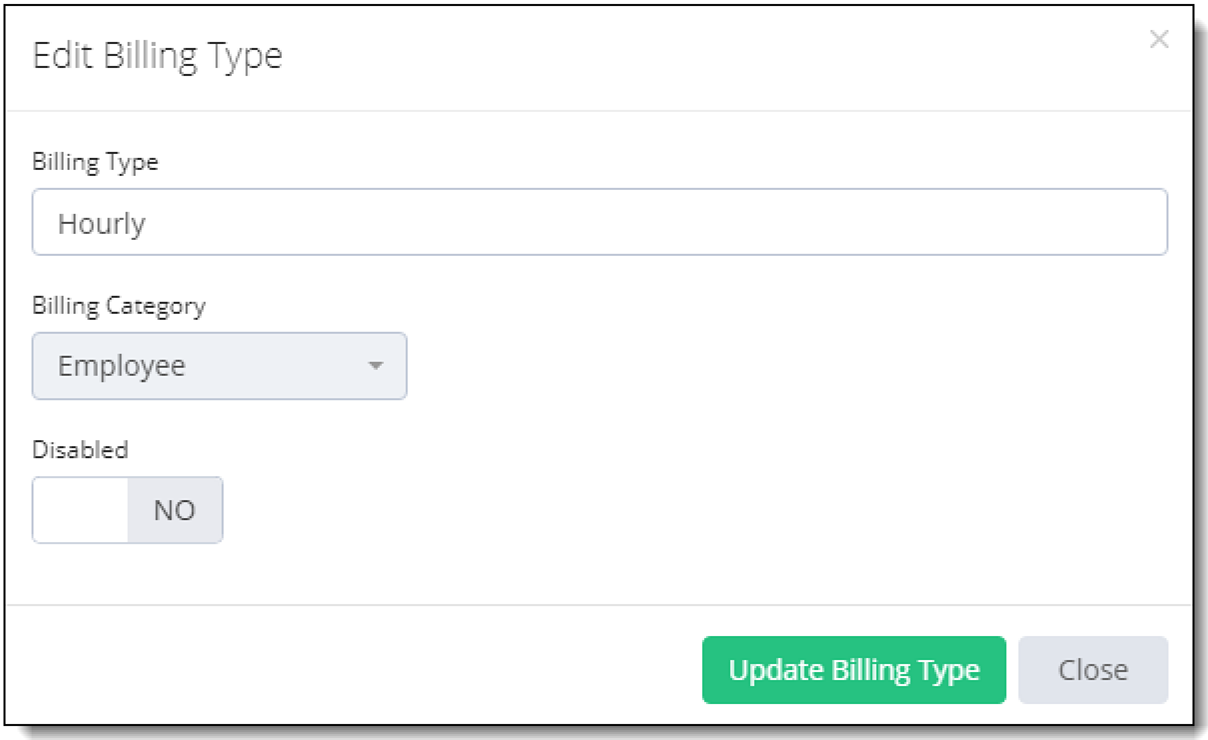


3. Click **Billing Types** icon.

4. In **Billing type List**, click on **Options** gear icon against the type which you want to disable and select **Edit Billing Type**.



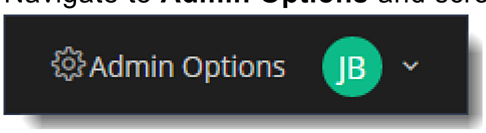
5. In the **Edit Billing Type** screen that appears:



   a. Click on the **Disabled** switch and change it to **Yes**.
   b. Click on **Update Billing Type** button to update this record.

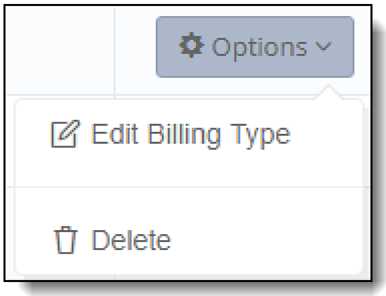
Deleting billing type:

1. Login TimeLive with Admin Credentials.
2. Navigate to **Admin Options** and scroll to **Billing Setup**.

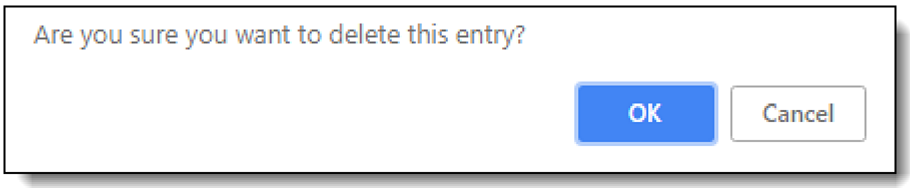


3. Click **Billing Types** icon.

4. In **Billing type List**, click on **Options** gear icon against the type which you want to delete.



   a. Click on **Delete**.
   b. Click on **OK** to confirm deletion on the dialog box.



12. Reports

12.1. Understanding Reports

Reports are helpful for determining how time is utilized throughout your organization and for reviewing project costs and client billing. You can generate reports to review detailed or summarized data. TimeLive allows you to create highly-customized reports to assist with tracking and management of time, expenses, users, and projects. Reports can be viewed within TimeLive or saved to a PDF or Excel (CSV) file.

A number of default reports have been installed with TimeLive; refer to Default Reports for a list of these reports and their uses. You can run these reports without modification, or customize these reports.(see Bottom)

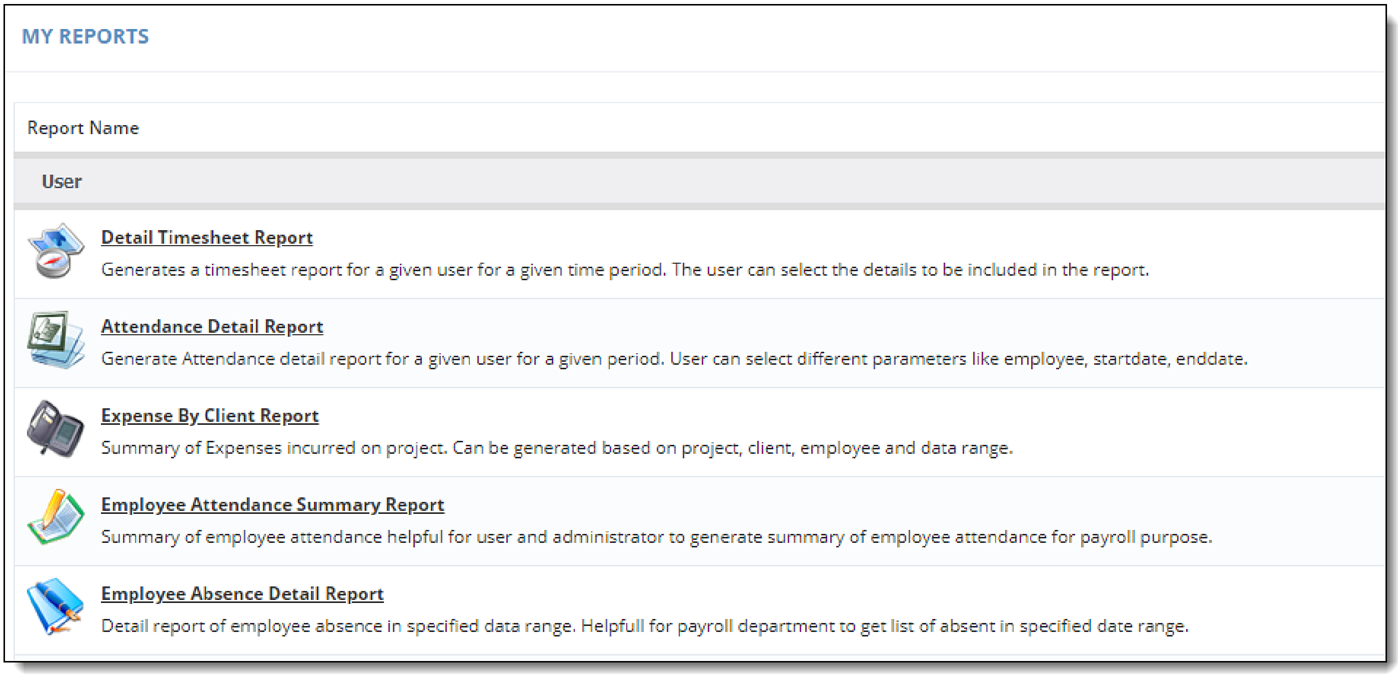
If you are running a text report with Summary Options selected, choose the Data Display Style:

- **Detailed**: each report entry will display in a separate row. Plus, a summary row will display for each row grouping and for the report as a whole.
- **Consolidated**: only summary rows will display – for each row grouping and for the report as a whole

The best way to start learning which reports work best for you is by viewing the report previews and running test reports.

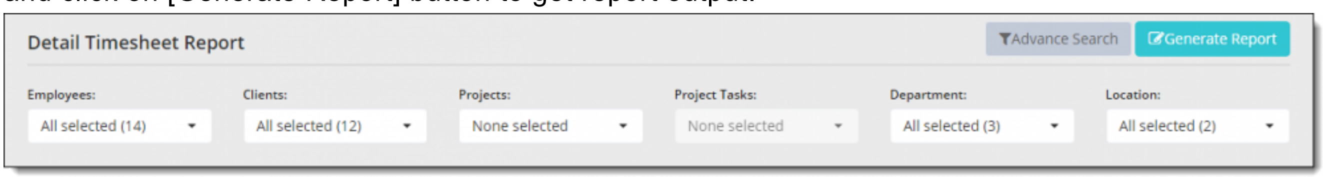
Reports can be launched by clicking on **[My Reports]** icon in the left navigation menu.

Report Home Page:

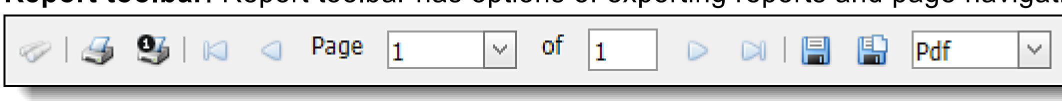


Parts of Report:

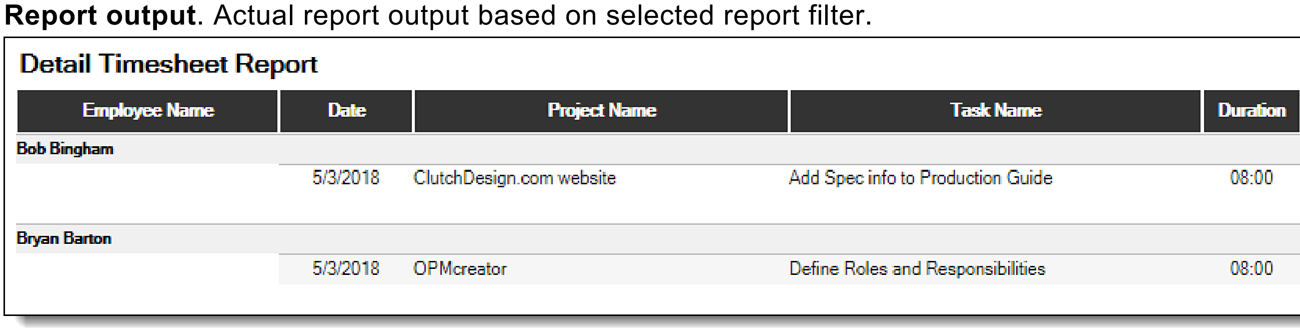
1. **Filter option of report**. These filter options are available in all reports. Just select your required filter and click on [Generate Report] button to get report output.



2. **Report toolbar**. Report toolbar has options of exporting reports and page navigation buttons.



3. **Report output**. Actual report output based on selected report filter.

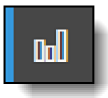

Detail Timesheet Report

Employee Name	Date	Project Name	Task Name	Duration
Bob Bingham				
	5/3/2018	ClutchDesign.com website	Add Spec info to Production Guide	08:00
Bryan Barton				
	5/3/2018	OPMcreator	Define Roles and Responsibilities	08:00

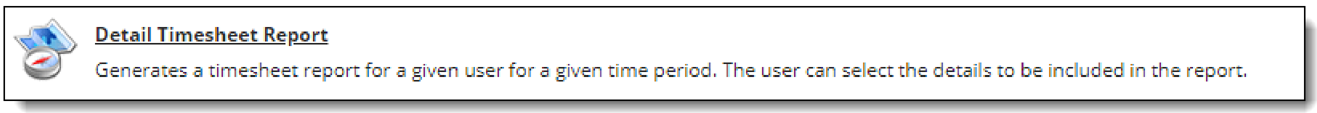
Running a Report

To run a report:

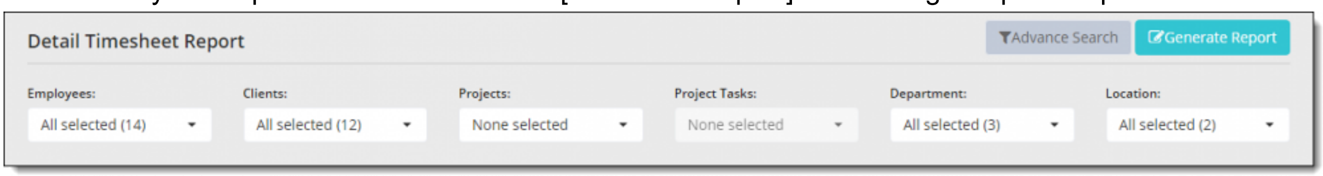
1. Navigate report home page by clicking on **[My Reports]** icon in the left navigation menu.



2. Select Reports from report page which you want to run.



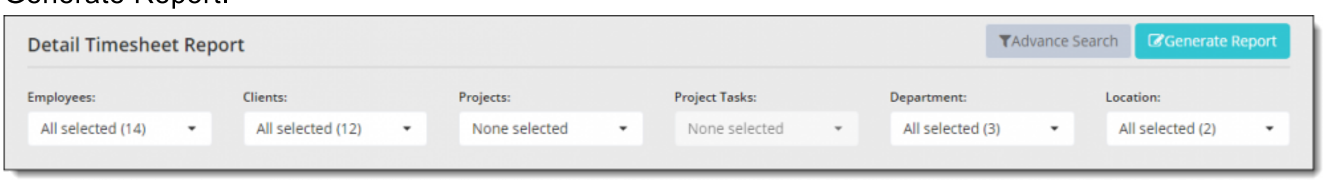
3. Just select your required filter and click on [Generate Report] button to get report output.



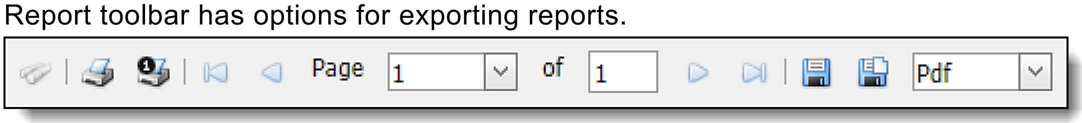
Exporting a Report

To export a report:

1. Generate Report.



2. Report toolbar has options for exporting reports.

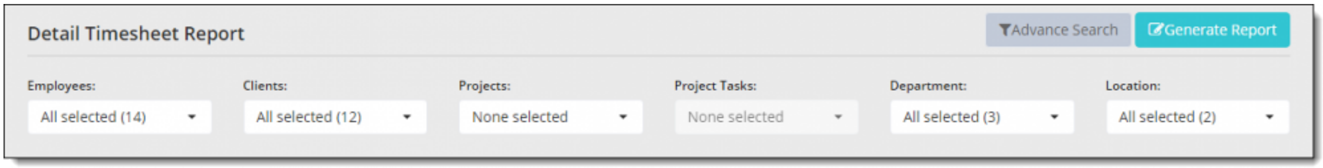


   a. Select your desired exportable format [PDF, HTML, XLS, RTF].
   b. Click on **[Export]** icon just right on exportable format drop-down.

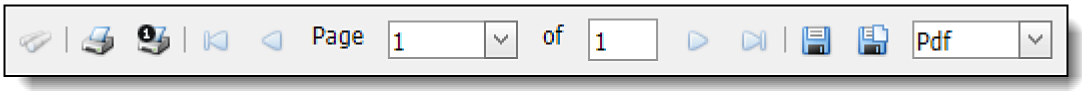
Printing a Report

To print a report

1. Generate Report.



2. Report toolbar has options for exporting reports.



   a. Click on **[Print]** icon in report toolbar to print currently selected reports.

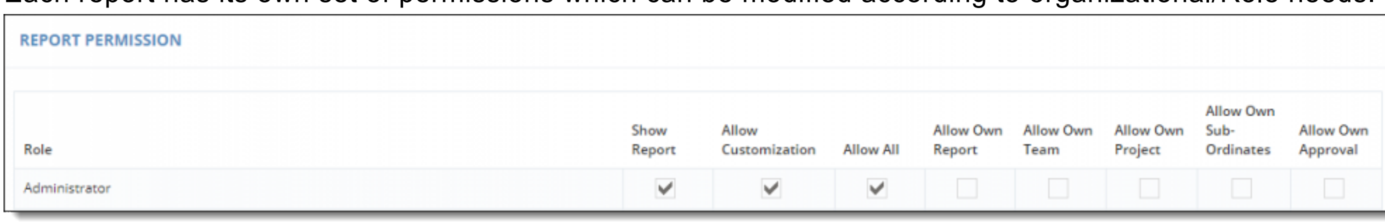
Report Customization:

This feature allow users to edit Name, Fields, grouping and layout of report. Each report can be customized according to organizational needs. To customize any report please click **Customize** link against required report. For Understanding Report Customization or creating new report, please review Understanding Report Designer.

Report Permission:

This feature allow users to access and modify reports according to Roles assigned in their profile.

Each report has its own set of permissions which can be modified according to organizational/Role needs.



Report Permission in detail:

- **Show Report**: This is basic permission which allow User to see this report in My Reports. If check is not applied, user will not be able to see that particular report in My Report Grid.
- **Allow Customization**: This permission give access to the Customization option in My Report for only that particular report (See Understanding Report Designer for Customization). Show Report check is mandatory if this check is enabled.
- **Allow All**: This permission gives access to view all employees in the organization.
- **Allow Own Report**: This permission gives access to user to view his/her own report.
- **Allow Own Team**: This permission gives access to user to view his/her own Team Report. For Example if he/she is the team lead or PM in any project, user can view their teams' report.

- **Allow Own Project**: This permission give access to user to view his/her own Project Report.
- **Allow Own Sub-Ordinates**: This permission give access to user to view Report of those employees of whom he/she is Employee Manager.

12.2. System Reports

TimeLive includes a large number of default reports. You can use the **[Report Datasources]** to create your own customized versions of reports. Alternatively, you can use pre-defined default reports, which are based on the **[Report Datasources]**. You can also customize default reports by modifying their settings.

TimeLive currently have following system reports which are available by default:

Users Reports:

- Detail Timesheet Report
- Detail Expense Report
- Attendance Detail Report
- Employee Attendance Summary Report
- Employee Absence Detail Report
- Absence Summary Report
- Detail Expense Report

Project Management Reports:

- Timesheet Approval Activity
- Expense Approval Activity
- Attendance Detail Report
- Project Cost And Revenue Report
- Detail Task Report
- Task Summary Report
- Task Status Summary Report
- Timesheet Submission Report
- Project Activity Summary Report

Billing Reports:

- Billing Report
- Task Billing By Projects/Clients
- Expense By Client Report
- Department Wise Timesheet Report
- Project Expense Detail Report

Administrator Reports:

- All Locations Report
- All Departments Report
- All Clients Report
- All Employees Report
- All Projects Of Organization
- All External Users Report

12.3. Understanding Report Designer

TimeLive reports are generating using pre-defined report data-sources. From these data sources, the user can select/deselect columns which they want to be showed in reports. The user can also define grouping, different summarized functions and custom formulas on all available Datasource columns.

Report Data-source:

These are the report Datasource currently available in TimeLive.

1. Absence
2. Attendance
3. Clients
4. Departments
5. Employee Time Off Audit Report
6. Employee Time Off Details
7. Employee Time Off Summary
8. Employees
9. Expense Entry
10. Expense Entry Approval Activity
11. Expense Sheet
12. Expense Sheet Audit Trail
13. External Users
14. Locations
15. Missing Time Entry Period
16. Project Task
17. Project Team
18. Projects
19. Tasks
20. Time Entry
21. Time Entry Approval Activity
22. Time Entry Period
23. Time Entry Periodic Submission
24. Time Off Request
25. TimeLive Invoice
26. Timesheet Audit Trail

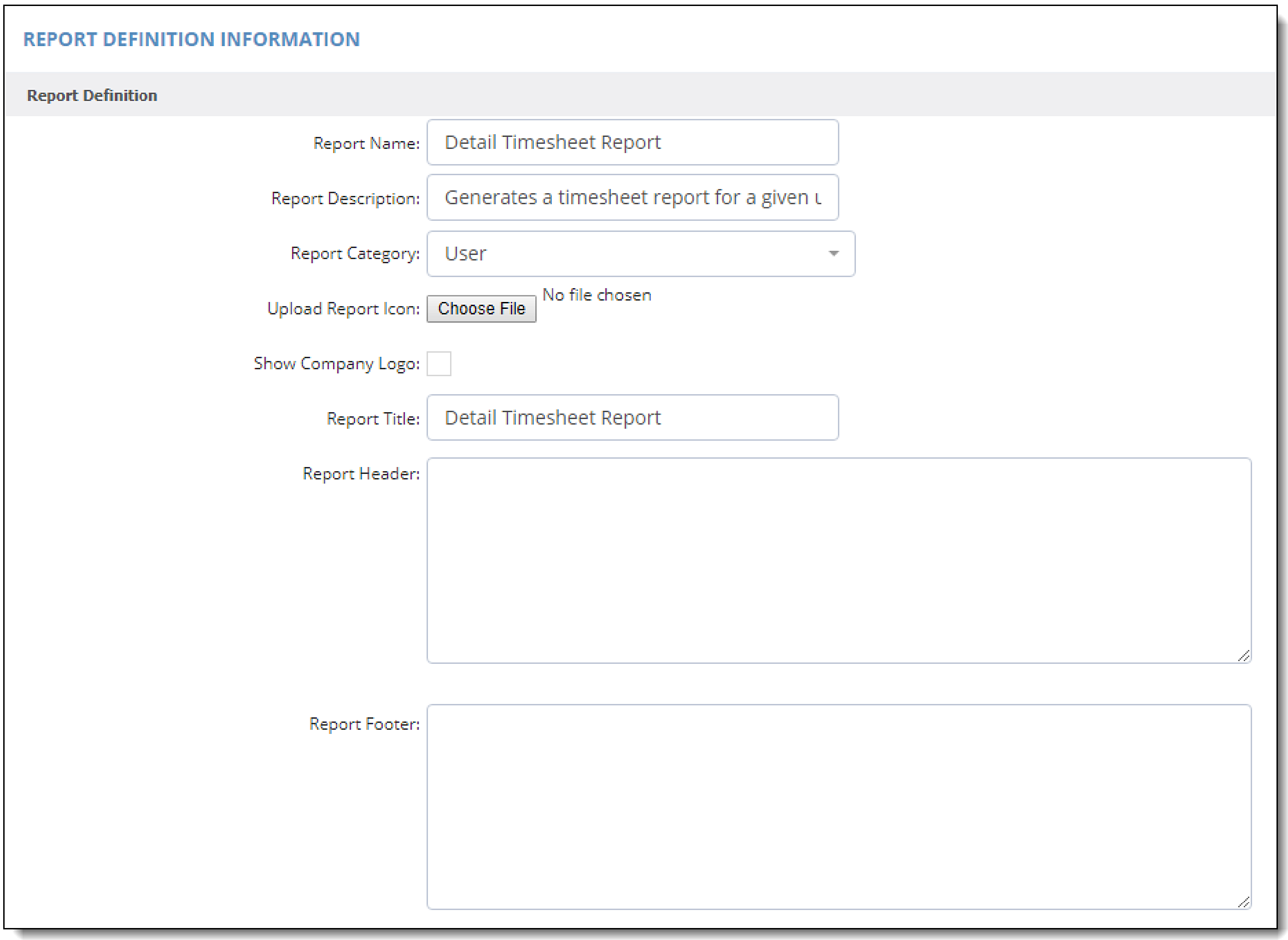
Report Designer Steps:

TimeLive Report Designer consists of 3 important wizard steps. Following are the name of steps and a brief description of every step:

Step 1: Report Description

For defining report name, report icon, report category and report type (detailed or consolidated).

Further, Unique Header and Footer can be described for identification of report during View, Export, or in Print.



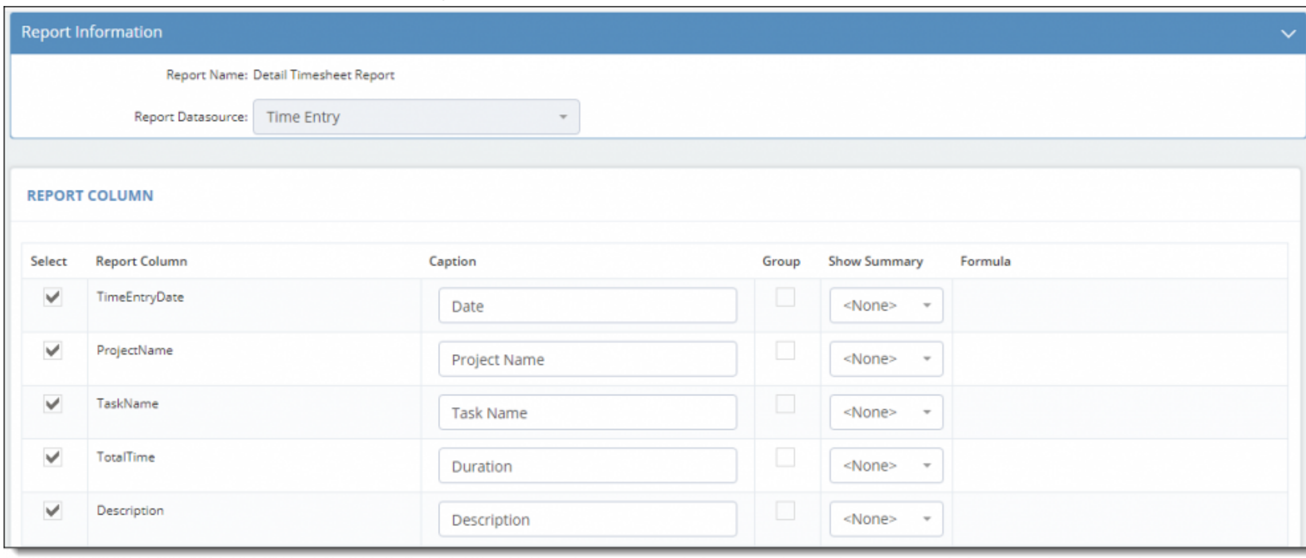
Step 2: Report Datasource Selection

Different Datasource contains some similar fields but most of the unique ones related to its Datasource to generate the required report.

A user can Edit existing Report or can create a new report through this Wizard page according to the requirements.

Following are the list of actions can be performed in this Wizard Page,

- Can select/ de-select Datasource column from the report as per requirement.
- Can give different Caption for each field according to organization's terminology.
- Can Group single or multiple fields to extract complex reports. (EmployeeName is grouped in below screenshot)
- Can select group summary type (sum, count) etc for each report column.
- Formula Fields (see Formula Column Syntax) and Custom Fields (if defined under Admin Options > Custom Fields see Custom Fields) can be added to extract more customized report.

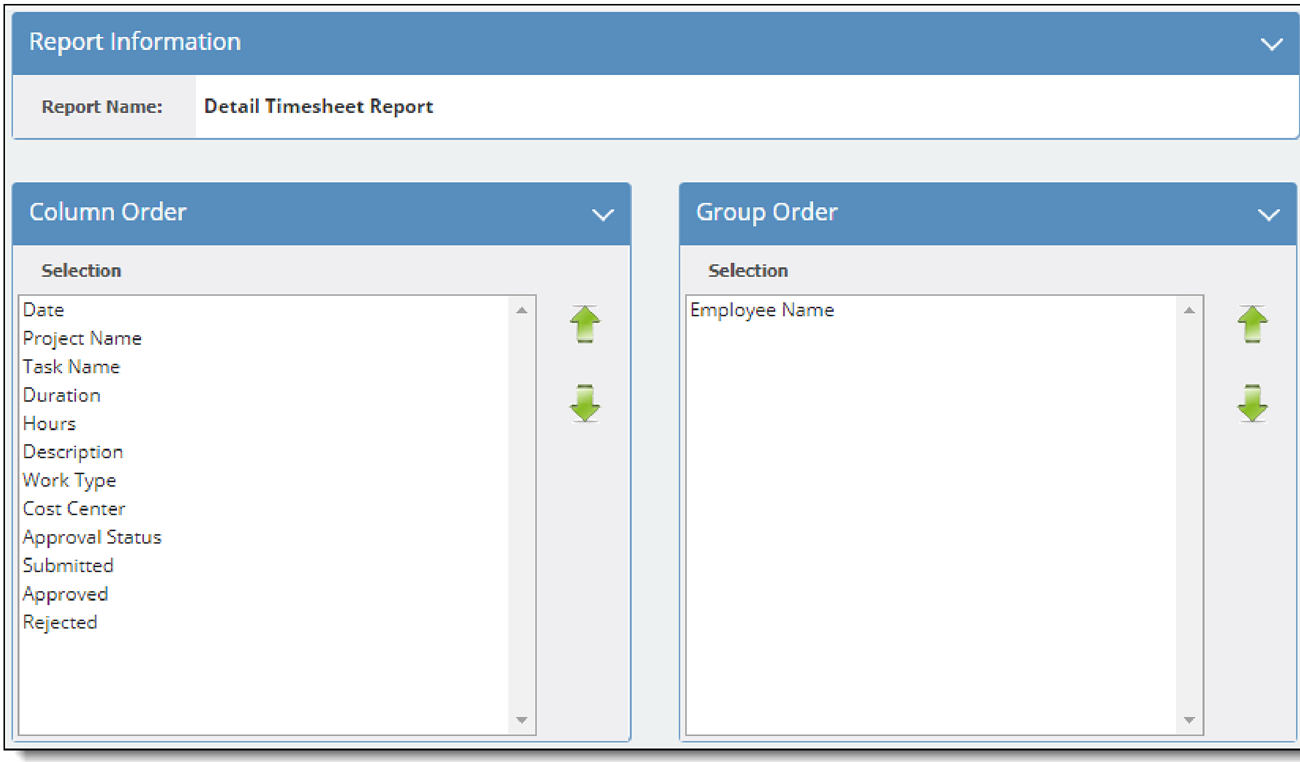


Step 3: Column and Group Ordering

The last page of Report Wizard, there are two tables. Both tables play a crucial part in generating the report according to the requirements.

- Column Order: Allows the user to set the order of fields in which report view is required. Topmost field will be the first column of the report and the last Field in the order will be the last column of the report.
- Group Order: If on the second page, Group check is enabled in any field, those fields will be visible in this table. That too can be sorted according to the requirement.
  In below screenshot, Employee Name was grouped to view Report according to Employee name then

the date which is the first field of Column order.



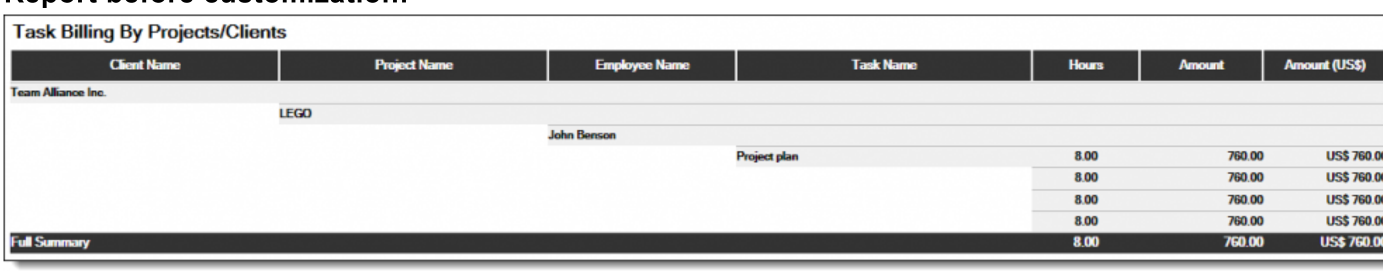
12.4. Understanding Formula Columns

TimeLive allows 5 formula fields in every report data source. A user can enter their own custom formula for these columns. And they can perform other reports functions like grouping, calculations (sum etc) and show in reports.

A typical case for Formula Columns:

Billing departments want to submit client billing reports with 10% extra overhead charges. TimeLive administrator then decided to **[customize] [Task Billing By Projects/Clients]** in **[Billing Reports]** category.

Report before customization:


Task Billing By Projects/Clients

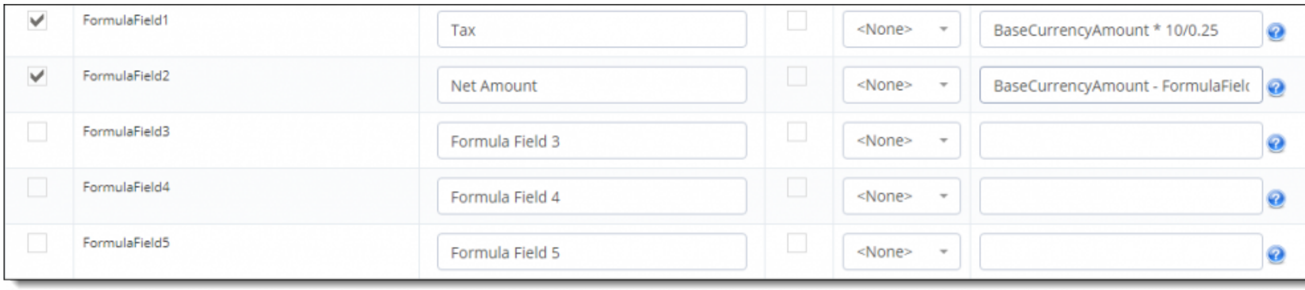
Client Name	Project Name	Employee Name	Task Name	Hours	Amount	Amount (US$)
Team Alliance Inc.						
	LEGO					
		John Benson				
			Project plan	8.00	760.00	US$ 760.00
				8.00	760.00	US$ 760.00
				8.00	760.00	US$ 760.00
				8.00	760.00	US$ 760.00
Full Summary				8.00	760.00	US$ 760.00

Here are the steps required to be followed for customizing **[Task Billing By Projects/Client]** report:

1. Click on **[My Reports]** to launch **[My Reports]** page.
2. Click on **[Customize]** link of report **[Task Billing By Projects/Clients]** in **[Billing Reports]** category.

3. Click on **[Next]** in **[Report Definition Information]** wizard step.

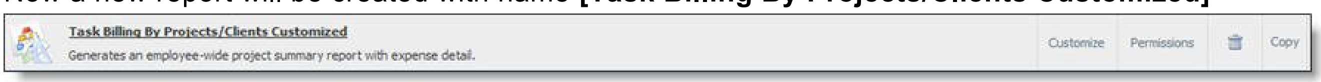
4. In **[Report Data Selection]** wizard step: follow these steps to customize **[Task Billing By Project/ Clients]** reports to achieve above-mentioned requirements.



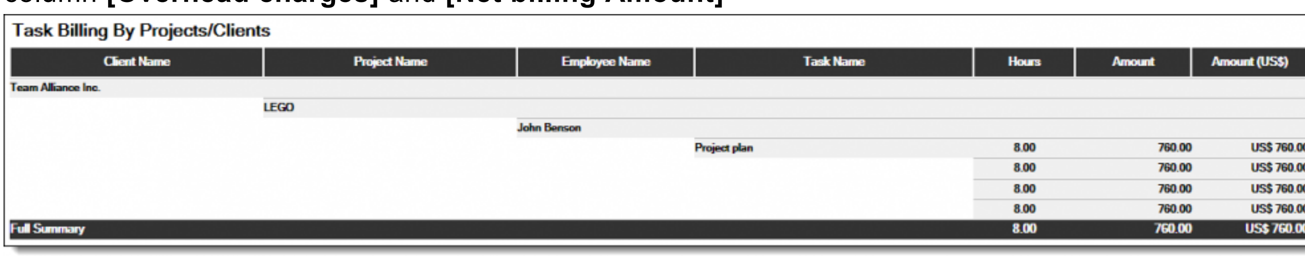
☑	FormulaField1	Tax	☐	<None>	BaseCurrencyAmount * 10/0.25
☑	FormulaField2	Net Amount	☐	<None>	BaseCurrencyAmount - FormulaFiel(
☐	FormulaField3	Formula Field 3	☐	<None>	
☐	FormulaField4	Formula Field 4	☐	<None>	
☐	FormulaField5	Formula Field 5	☐	<None>	

   a. Check on **[FormulaField1]** and update its caption to **[Overhead Charges]**.
   b. Select **[Sum]** in **[Show Summary]** dropdown of **[FormulaField1]**.
   c. Enter formula for calculating 10% of Amount: Amount * 10 / 100.
   d. Check on **[FormulaField2]** and update its caption to **[Net Billing Amount]**.
   e. Select **[Sum]** in **[Show Summary]** dropdown of **[FormulaField2]**.
   f. Enter formula for sum of original amount field and calculated **[Overhead Charges]**: Amount + FormulaField1.
   g. Click on **[Next]** to update these changes.

5. Click on **[Finish]** button in final wizard step **[Column and Group order]**.
6. Now a new report will be created with name **[Task Billing By Projects/Clients Customized]**



7. Clicking on this report in **[My Reports]** will show TimeLive customized report with two additional column **[Overhead charges]** and **[Net billing Amount]**



Task Billing By Projects/Clients

Client Name	Project Name	Employee Name	Task Name	Hours	Amount	Amount (US$)
Team Alliance Inc.						
	LEGO					
		John Benson				
			Project plan	8.00	760.00	US$ 760.00
				8.00	760.00	US$ 760.00
				8.00	760.00	US$ 760.00
				8.00	760.00	US$ 760.00
Full Summary				8.00	760.00	US$ 760.00

8. Click on **[Permissions]** link to give this customized reports to your required Roles.

12.5. Formula Column Syntax

Formula columns are derived from user-defined formulas. Users create these formulas by combining columns (fields) available in the report.

When might I need to create a custom column?

Custom columns are useful for creating such numeric fields as:

- Values with non-standardized modes of calculation that are therefore not included in TimeLive as default columns (for example, overhead cost etc)
- Calculations specific to your organization
- Highly-specialized calculations

Numeric Formula Examples:

- A formula for 10% Overhead cost in the **[Detail Timesheet Report]**: (Amount * 10 / 100)
- A formula for calculating profit margin in the **[Detail Timesheet Report]**. (BillingRate – EmployeeRate) * TotalHours

Text Formula Examples:

- To identify the location and their departments in a single field, you could combine both location and department by creating a formula like this: (AccountLocation+'-' DepartmentName)

Numeric Operators:

Operator	Use this operator to specify…
+	Addition
-	Subtraction
	Multiplication
/	Division
(…)	Parentheses
=	Two values are equal
<	Less than
>	Greater than

<=	Less than or equal
>=	Greater than or equal
!=	Not equal
%	Modulus

Operators follow standard order of operations rules. For example:

- 2 + 2 * 2 = 6, but (2 + 2) * 2 = 8
- 2 + 2 / 2 = 3, but (2 + 2) / 2 = 2

Functions:

Functions	Description	Syntax	Example
IIF	Gets one of two values depending on the result of a logical expression	Iif (expr, truepart, falsepart)	Iif (billingrate>500,‘expensive’,‘dear’)
LEN	Gets the length of a string	LEN (expression)	Len (EmployeeName)
CONVERT	Converts particular value to a specified .NET Framework Type.	Convert(expression, type)	Convert(total, ‘System.Int32’), Convert(id,‘System.String’)
SUBSTRING	Gets a sub-string of a specified length, starting at a specified point in the string.	SUBSTRING (expression, start, length)	SUBSTRING (phone, 7, 8)

13. Integration in TimeLive

13.1. Importing / Exporting Raw Data in TimeLive

TimeLive’s raw data imports/exports allow you to import / export raw **[timesheet]**, **[project]**, and **[expense]** data to spreadsheets using a CSV file. CSV file can be opened in Excel as well as the standard way of transferring data from different applications.

TimeLive support these types of data in CSV Import / Export:

1. Client
2. Employee
3. Department
4. Projects
5. Time Entry
6. Expense Entry
7. Project Tasks
8. Task Users
9. Project Team
10. Project Roles

13.2. QuickBooks Integration wtih TimeLive

13.2.1. Using TimeLive QuickBooks Integration Tool

13.2.1.1. Using TimeLive's QuickBooks Integration Tool

[TimeLive QuickBoooks Integration Manager] provides users the ability to transfer information between **TimeLive** and **QuickBooks**. With the ability to transfer time and expense data, the integration module is an ideal solution for those working on projects or tasks that are to be billed to clients. Using TimeLive, you can track the actual time spent working on projects, then transfer the data to QuickBooks, to invoice clients for the work completed. Time data entered by employees can also be transferred for use in generating **payroll timesheets** in QuickBooks. Plus, expenses incurred against the project can also be entered in TimeLive, and transferred to QuickBooks to **reimburse employees**, **pay vendors** and **charge clients**. Additionally, you can use the integration module to save the time and effort associated with entering the user and project data in both QuickBooks and TimeLive. This duplication can be eliminated by entering the data in one application and transferring to the other using the **[TimeLive QuickBooks Integration Manager]**.

Data Type	TimeLive Term	QuickBooks Term
Employees	Employee (ones paid on a regular and on-going basis).	Employee
Vendors	Employees with "IsVendor=Yes in Employee Types"	Vendor
Customers	Clients	Customers
Jobs/Items	Project / Task	Job/Item, Job/Sub Job or Item/ Sub Item
Time Tracking	Timesheet	Time Tracking
Expense Tracking	Expense Sheet	Vendor Bills

Steps for setting up TimeLive's QuickBooks Integration Tool:

1. Download the module
2. Installing the module
3. Configuring QuickBooks for use with the Module.
4. Launching the Module

Downloading the module:

TimeLive's QuickBooks integration module can be download from download page from http://www.livetecs.com/Home/Download.aspx

Installing the module

Installation of the module is a simple, wizard-guided process. Before beginning installation, ensure that:

1. Your machine meets the integration module’s minimum system requirements.
2. You are installing the integration module on the computer where QuickBooks is installed.
3. Your version of QuickBooks is updated to include all available patches (recommended).

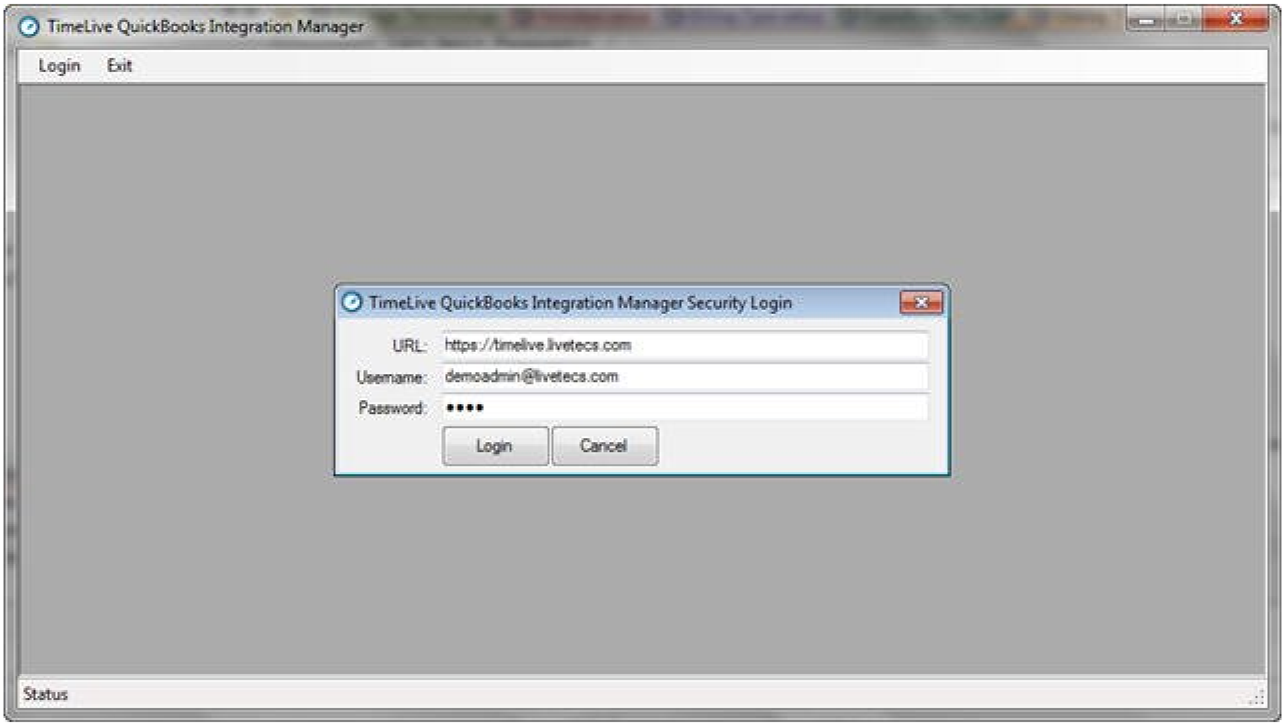
Configuring QuickBooks for use with the Module

Before attempting to integrate with TimeLive, follow the steps below to ensure QuickBooks is properly configured for integration.

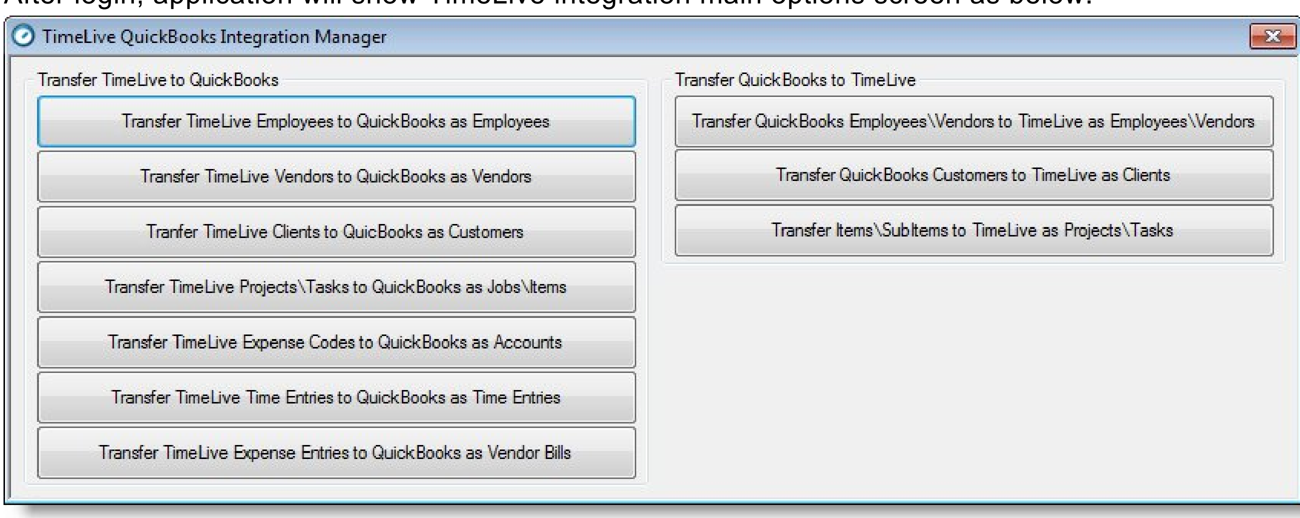
1. Launch **[QuickBooks]**.
2. Open a company file.
3. When you integrate with TimeLive, you will transfer data to and from the open QuickBooks company file.
4. If a dialog box displays indicating that TimeLive is attempting to access the QuickBooks company file without a certificate: a. Select the **[Yes]**, **[Always]** button. This will allows TimeLive to access the company file for data transfer purposes.

Launching the Module:

1. To launch the integration module:
   a. Double-clicking a shortcut name **[TimeLive Quickbooks Integration Manager]** created during installation
   b. Selecting **[TimeLive QuickBooks Integration Manager]** from the Windows Start menu.
2. On login window, enter **[TimeLive URL]** which you are using to access your TimeLive application, TimeLive username and password.



3. Click on **[Login]** button to launch Integration tool.
4. When a QuickBooks dialog box stating the module is attempting to access the company file displays, enable the check box labeled **[Allow]** this application to access. This will give TimeLive permission to transfer employee and vendor data to the company file. Select the **[Yes]** or **[Always]**. Select **[Yes]** if asked to confirm. The main **[TimeLive Integration Manager for QuickBooks]** window displays.
5. After login, application will show TimeLive integration main options screen as below.



13.2.1.2. TimeLive QuickBooks Integration Manager System Requirements

Each computer with **[TimeLive QuickBooks Integration Tool]** must meet the following requirements:

Hardware:

- Core 2 Duo or higher.
- 2 GB RAM.
- 512 MB free hard disk space.
- Screen resolution of 1024 × 768 or higher.

Operating system:

One of the following:

- Windows Server 2008 (R1/R2) or higher.
- Windows 2003
- Windows XP Professional SP2
- Windows Vista
- Windows 7
- Windows 8 or 8.1
- Windows 10

Software:

One of the following:

- QuickBooks Pro or Premier U.S. Edition 2006 or later
- QuickBooks Pro or Premier Canadian Edition 2006 or later
- QuickBooks Pro or Premier UK Edition 2008 or later
- QuickBooks Enterprise Solutions U.S. Edition 6.0 or later
- QuickBooks Premier Canadian Multicurrency Edition or later

13.2.1.3. Defining the Relationship between Data Structures

Understanding Project / Task relationship with QuickBooks:

Defining the Relationship between Data Structures Before using the integration module to transfer data, you must determine how the data structure in TimeLive relates to the data structure in QuickBooks. Understanding the Relationship between Data Structures When transferring certain data types, you must define how the data structure in TimeLive relates to the data structure in QuickBooks. When transferring **[project]** data from TimeLive to QuickBooks, the relationship tells the integration module where the projects and tasks should be transferred to in Quick- Books. For example, TimeLive **[Projects]** can be transferred to QuickBooks as **[Jobs]** or as **[Items]**. If the integration module is configured to transfer **[Projects]** as **[Jobs]**, then TimeLive Tasks can be transferred as **[Sub Jobs]** or **[Items]**. Depending on the selection data.

Understanding Time Entries relationship with QuickBooks:

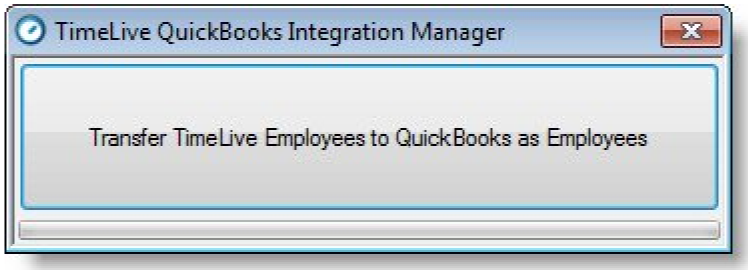
When transferring **[time entries]**, the integration module will use the specified relationship to locate the projects and tasks selected for the time entries in QuickBooks. For example, if a TimeLive **[project]** exists in QuickBooks as a **[job]**, you would typically assign a relationship between **[Projects]** and **[Jobs]**. When transferring a time entry for that project, the wizard will then attempt to locate the **[project]** in the QuickBooks **[Jobs]** list. Once the **[project]** is found as a job, the time entry can be associated with the **[job]** and will be successfully transferred.

13.2.1.4. Transferring Employees

To transfer **[Employees records from TimeLive to QuickBooks]**, just click on **[Transfer TimeLive Employees to QuickBooks as Employees]**
button in main option screen of integration tool.

Steps for Transfer:

1. Click on **[Transfer TimeLive Employees to QuickBooks as Employee]** button.



2. After successful, application will show a Successful transferred message.

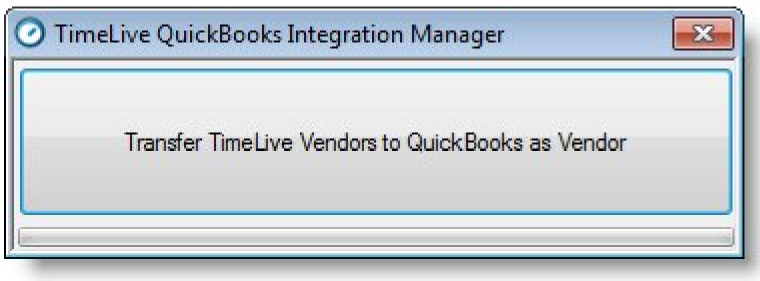
13.2.1.5. Transferring Vendors

To transfer **[Vendors records from TimeLive to QuickBooks]**, just click on **[Transfer TimeLive Vendors to QuickBooks as Vendors]**
button in main option screen of integration tool.

You can define an **[Employee]** as **[Vendor]** in TimeLive by checking on **[Is Vendor]** field in **[Admin Options] —> [Employee Types]**. And then you can assign that **[Employee Type]** to your user who should be treated as **[Vendor]**.

Steps for Transfer:

1. Click on **[Transfer TimeLive Vendors to QuickBooks as Vendor]** button.



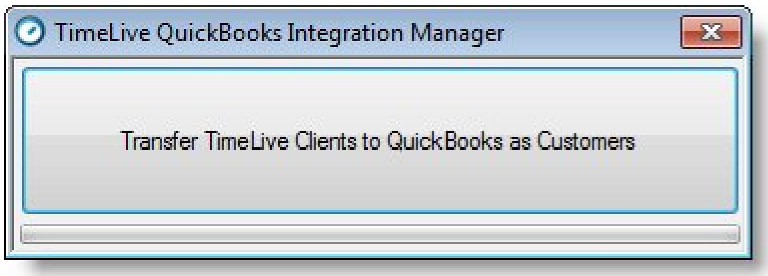
2. After successful, application will show a Successful transferred message.

13.2.1.6. Transferring Customers

To transfer **[Clients/Customers records from TimeLive to QuickBooks]**, just click on **[Transfer TimeLive Clients to QuickBooks as Customers]**
button in main options screen of integration tool.

Steps for Transfer:

1. Click on **[Transfer TimeLive Clients to QuickBooks as Customers]** button.



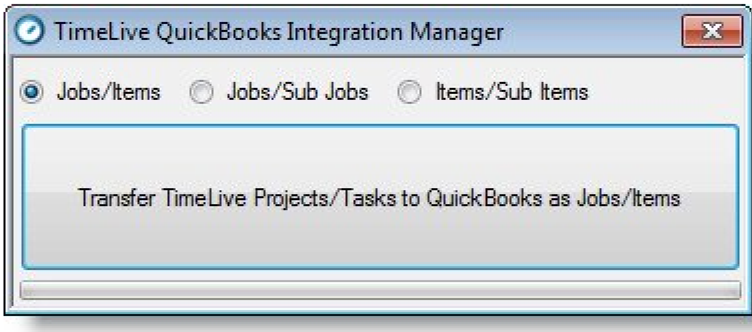
2. After successful, application will show a Successful transferred message.

13.2.1.7. Transferring Projects/Tasks

To transfer **[Projects/Tasks records from TimeLive to QuickBooks]**, just click on **[Transfer Project/ Tasks to QuickBooks as Jobs/Items]** button in main options screen of integration tool.

Steps for Transfer:

1. Select any mapping option of TimeLive Project / Tasks to QuickBooks which you want in this transfer.
   a. **Jobs / Items** (Projects Tasks will be transferred as **[Jobs/Items]** in QuickBooks
   b. **Jobs / Sub Jobs** (Project Tasks will be transferred as **[Jobs/Sub Jobs]**
   c. **Items / SubItems** (Project will be transferred as **[Items/Sub Items]**
2. Click on **[Transfer TimeLive Project/Tasks to QuickBooks as Jobs/Items]** button.



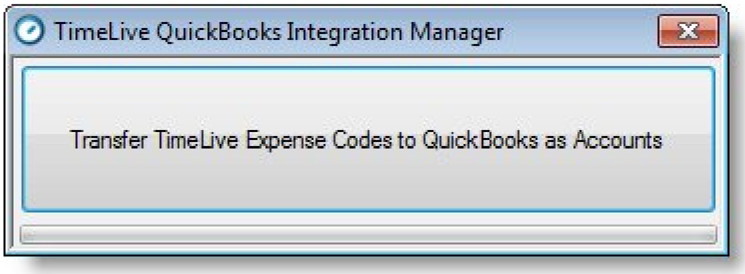
3. After successful, application will show a Successful transferred message.

13.2.1.8. Transferring Expense Codes

To transfer **[Clients/Customers]** records from TimeLive to QuickBooks, just click on **[Transfer TimeLive Expense Codes to QuickBooks as Accounts]** button in main options screen of integration tool.

Steps for Transfer:

1. Click on **[Transfer TimeLive Expense Codes to QuickBooks as Accounts]** button.



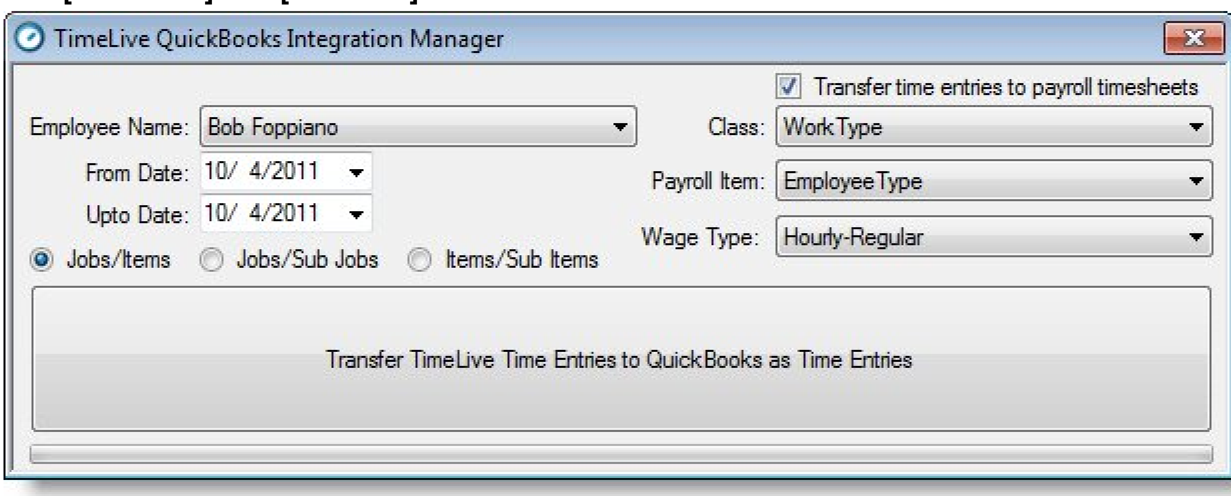
2. After successful, an application will show a Successful transferred message.

13.2.1.9. Transferring Time Entries

To transfer **[Time Entries]** records from TimeLive to QuickBooks, just click on Transfer **[TimeLive Time Entries to QuickBooks as Time Entries]** button in main option screen of integration tool.

Steps for Transfer:

1. Select **[Employee Name]** if you want to transfer any selected employees timesheet. Just select **[All]**, if you want to select time entries of all employees.
2. Select **[start date]** and **[end date]**



3. Select type of mapping for your **[Project / Tasks]** with QuickBooks which you used while transferring your projects / tasks to QuickBooks
   a. **Jobs / Items** (Projects Tasks will be transferred as “Jobs/Items” in QuickBooks
   b. **Jobs / Sub Jobs** (Project Tasks will be transferred as “Jobs/Sub Jobs”
   c. **Items / SubItems** (Project will be transferred as “Items/Sub Items”
4. Tick on “Transfer time entries to payroll timesheets”, if you want to transfer TimeLive time entries with payroll related items.
5. Select mapping of QuickBooks **[Class Items]** with TimeLive data. You can select either of these option for mapping of Class data.
   a. None
   b. Cost Center
   c. Department
   d. Employee Type
   e. Milestone
   f. Work Type

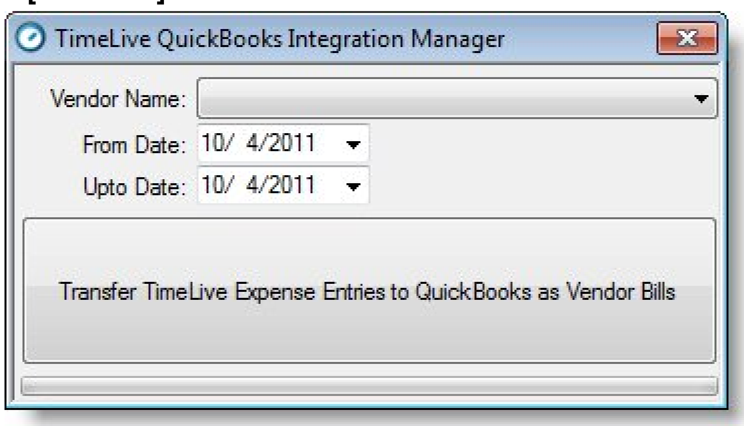
6. Select mapping of QuickBooks **[Payroll Item]** mapping with TimeLive data. You can select either of these option for mapping of Payroll Items data.
   a. None
   b. Cost Center
   c. Department
   d. Employee Type
   e. Milestone
   f. Work Type
7. Select **[Wage Type]** which you want to be populated with your time entries
8. Click on **[Transfer TimeLive Expense Codes to QuickBooks as Accounts]** button.
9. After successful, application will show a Successful transferred message.

13.2.1.10. Transferring of Expense Entries as Vendor Bills

To transfer **[Expense Entries]** records from TimeLive to QuickBooks as **[Vendor Bills]**, just click on **[Transfer TimeLive Expense Entries to QuickBooks as Vendor Bills]** button in main option screen of integration tool.

Steps for Transfer:

1. Select **[Vendor Name]** if you want to transfer any selected vendor expense sheet. Just select All, if you want to select time entries of all vendors.
2. Select **[start date]** and **[end date]**



3. Click on **[Transfer TimeLive Expense Entries to QuickBooks as Vendor Bills]** button.
4. After successful, application will show a Successful transferred message.

13.2.2. Exporting Time to QuickBooks

Overview:

The QuickBooks Pro product has the capability of importing data from an external source. Data can include lists that correspond to QuickBooks' lists, budgets, and individual transactions. We will be utilizing the IIF imports to facilitate the importing of time data into QuickBooks. This is facilitated by creating a comma (or tab) separated file in a format that QuickBooks can understand (the QuickBooks IIF format). This format includes the use of column heading tags (keywords) and row heading tags to help identify the beginning and end of each transaction.

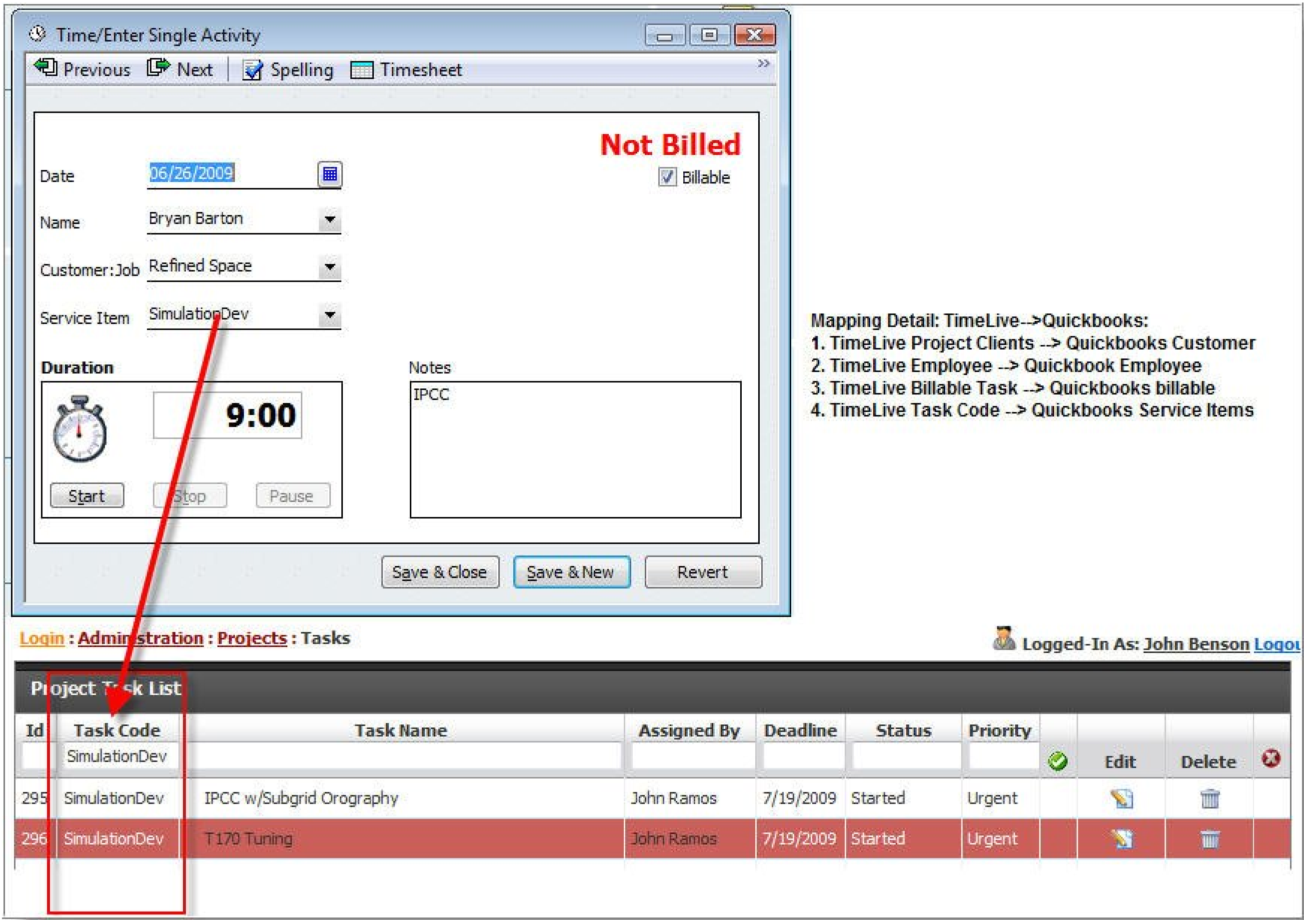
IIF File Naming:

When you are exporting the data out of TimeLive, in preparation for the import into QuickBooks, the extension given to the file is important. QuickBooks is expecting the file extension to be "IIF" or "iif". When on the Export — Time screen, one of the options is to supply the output filename. You can at this point provide the appropriate the file name with IIF extension.

For example, an appropriate file name may look like: june_time.iif

TimeLive —> Quickbooks Field Mapping (Time Entry Import) :

1. TimeLive Project Clients —> Quickbooks Customer
2. TimeLive Employee —> Quickbooks Employee
3. TimeLive Billable Task Flag —> Quickbooks Billable Flag
4. TimeLive Task Code —> Quickbooks Service Items



Preparing Data for input into QuickBooks:

The following steps may be followed to:

1. Export Data from TimeLive in a Quickbooks ready format (IIF).
2. Import the exported time data into Quickbooks.
3. Define billing rates for **[Items]** in Quickbooks.
4. Generate an Invoice in Quickbooks based on imported TimeLive Time data.

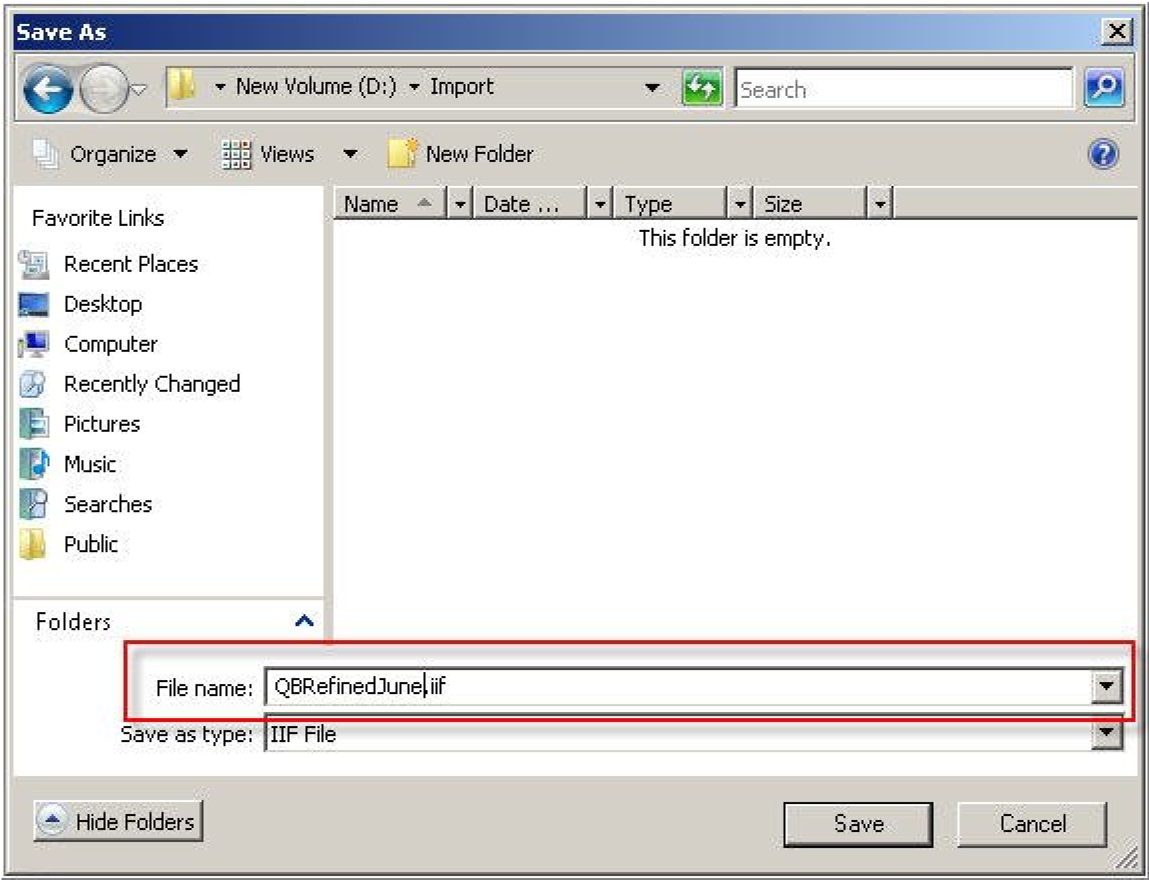
Step 1: Export Data from TimeLive in a Quickbooks ready format

The following steps may be followed to:

1. Navigate to **[Admin Options] —> [Import / Export CSV/Quickbooks]** in **[Import/Export]** section.
2. Select **[Import/Export]** type as "**Export**"
3. Select **[File Type]** as "**Quickbooks**'

4. Select **[Data Source]** as "**Time Entry**"
5. Define your data export start date and end date.
6. Select your project whose time entries you want to export.

7. Now click on **[Proceed]** button.
8. Give some best name for your exported IIF file and note down your IIF file saved location.

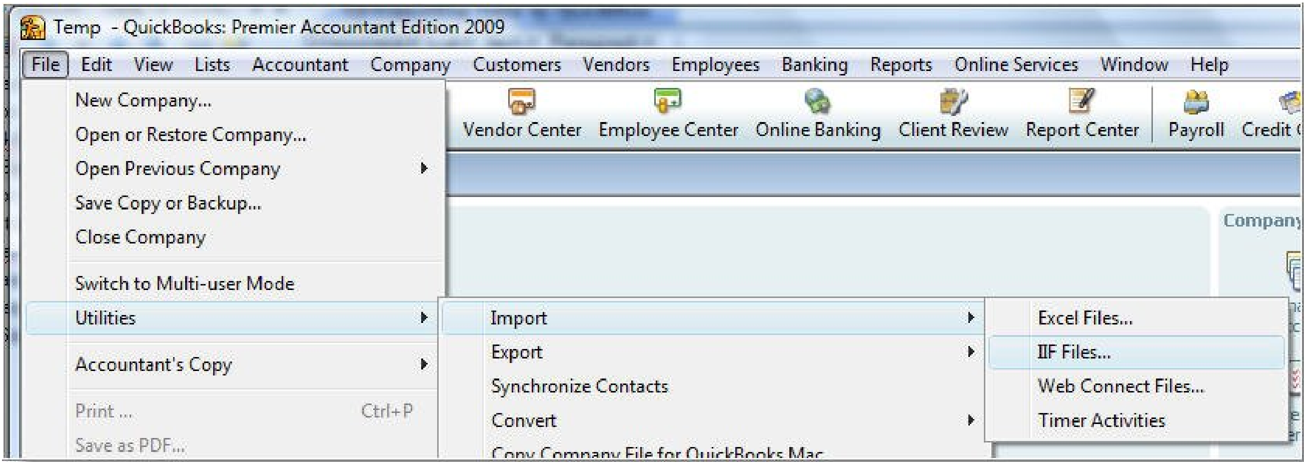


9. This is end of exporting time data to Quickbooks IIF File.

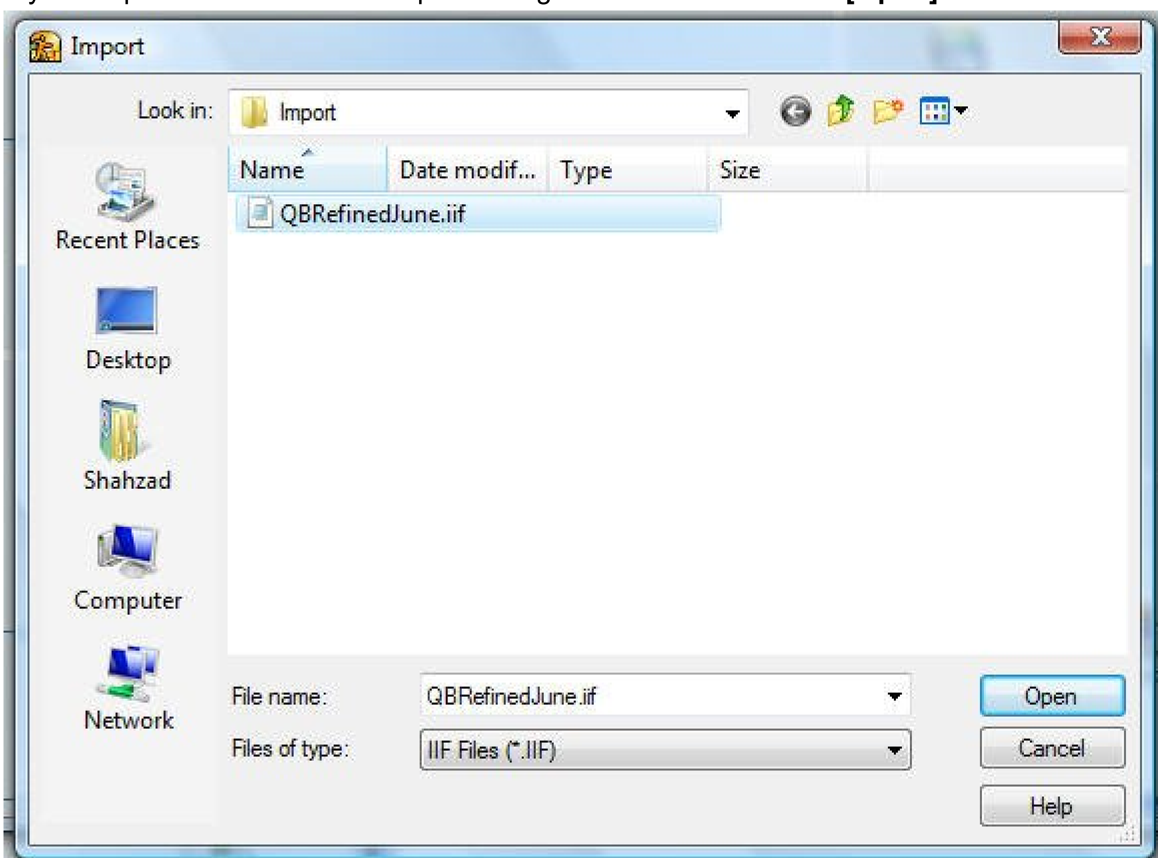
Step 2: Import the exported time data into Quickbooks.

The following steps may be followed to:

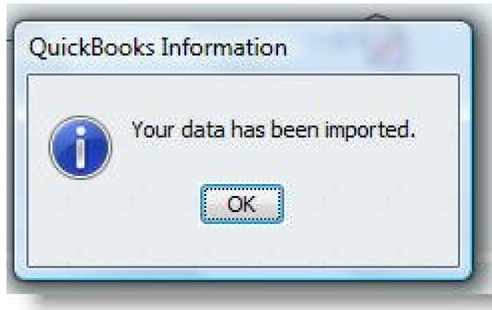
1. Launch Quickbooks
2. Open **[File] —> [Import]—> [IIF Files…]**



3. Select your exported IIF File in File open dialog box and then click on **[Open]**.

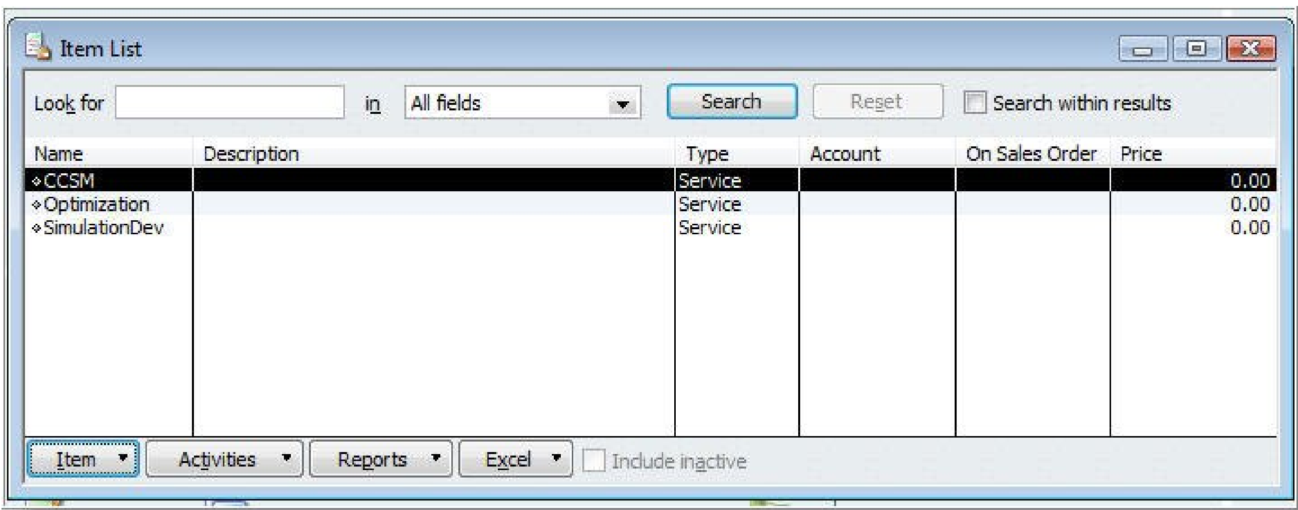


4. Quickbooks will show completed messaged after import.

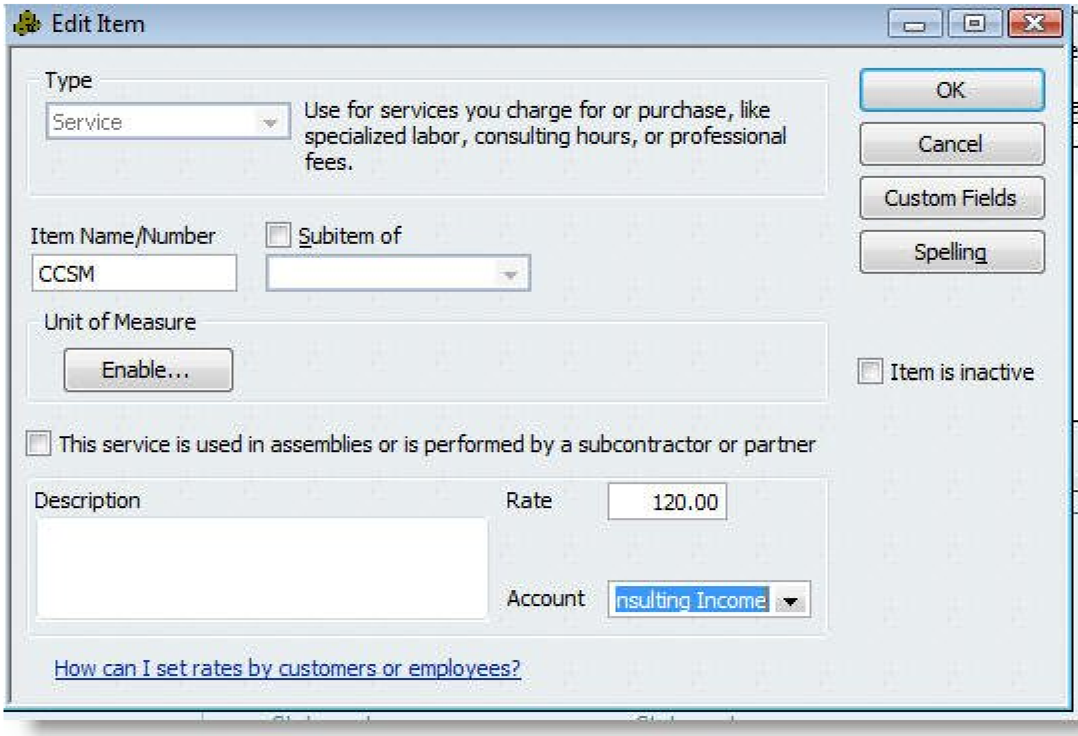


Step 3: Define billing rates for [Items] in Quickbooks.

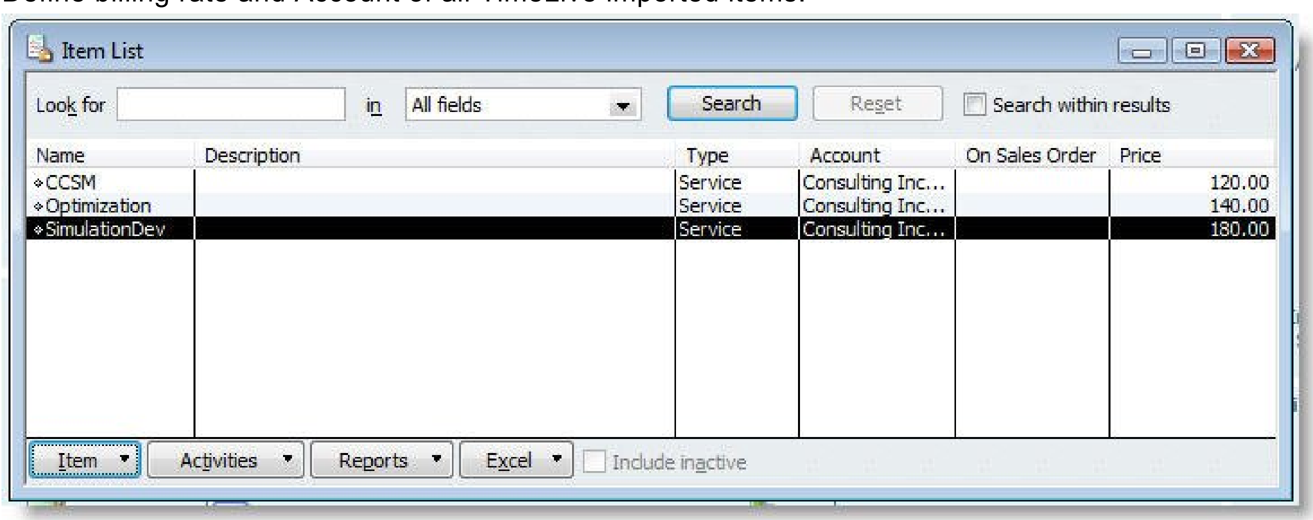
1. Open **[Customers] —> [Item List]** in Quickbooks.
2. Double click on all imported **[Items]** and define their **[Account]** and **[Rate]**



3. Define item's **[Billing Rate]** and **[Account]** in Quickbooks. Click on **[OK]** when done.

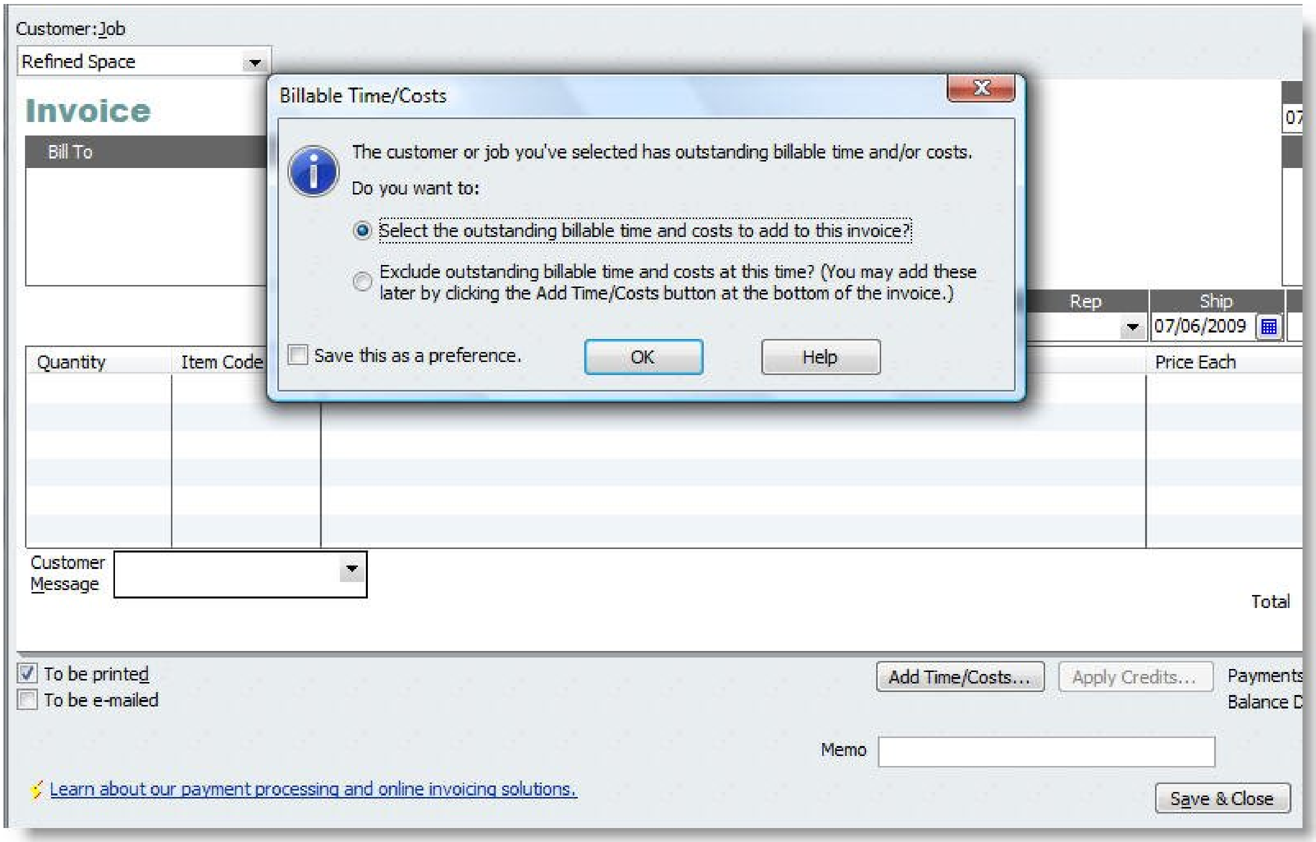


4. Define billing rate and Account of all TimeLive imported items.

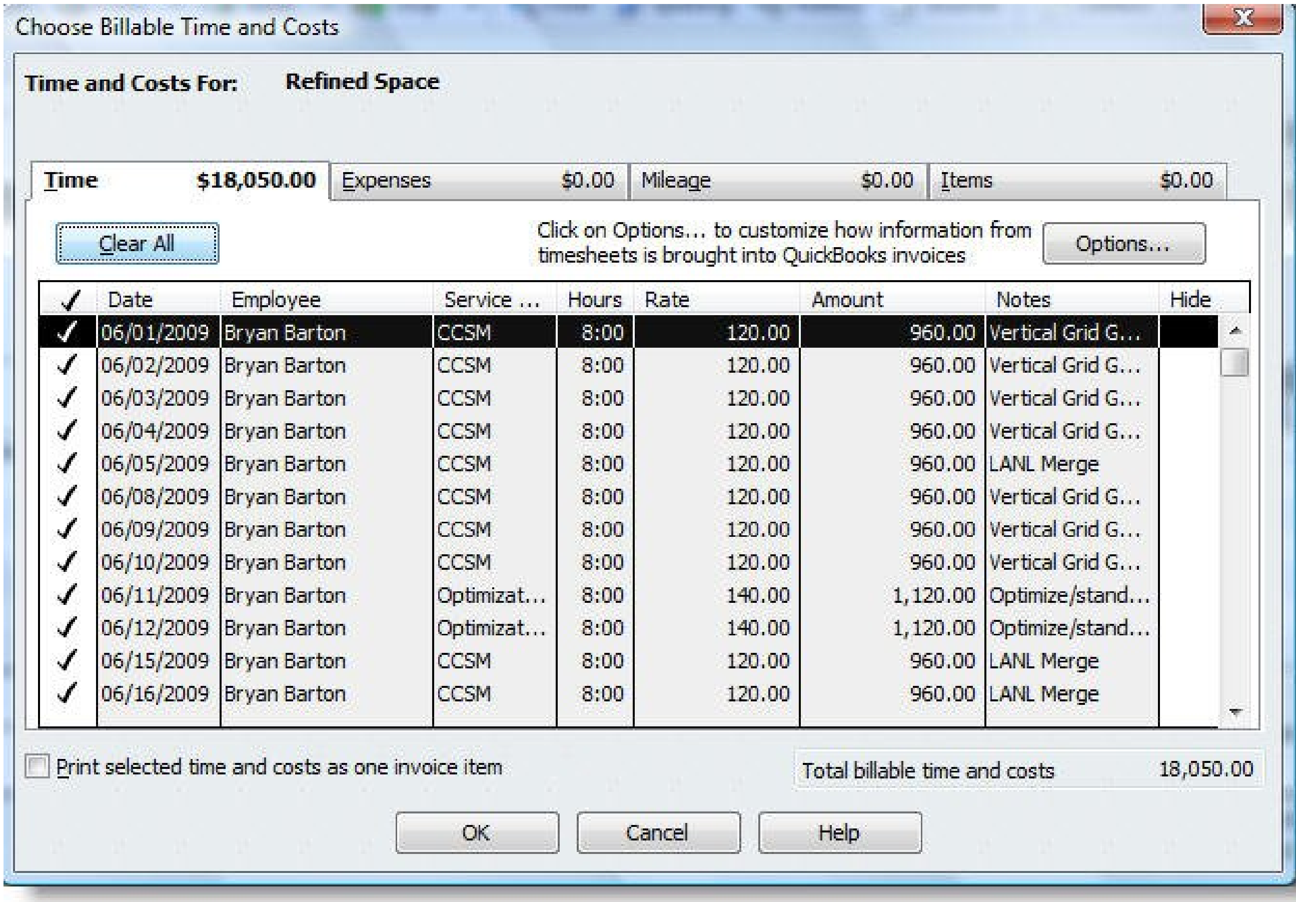


Step 4: Generate an Invoice in Quickbooks based on imported TimeLive Time data

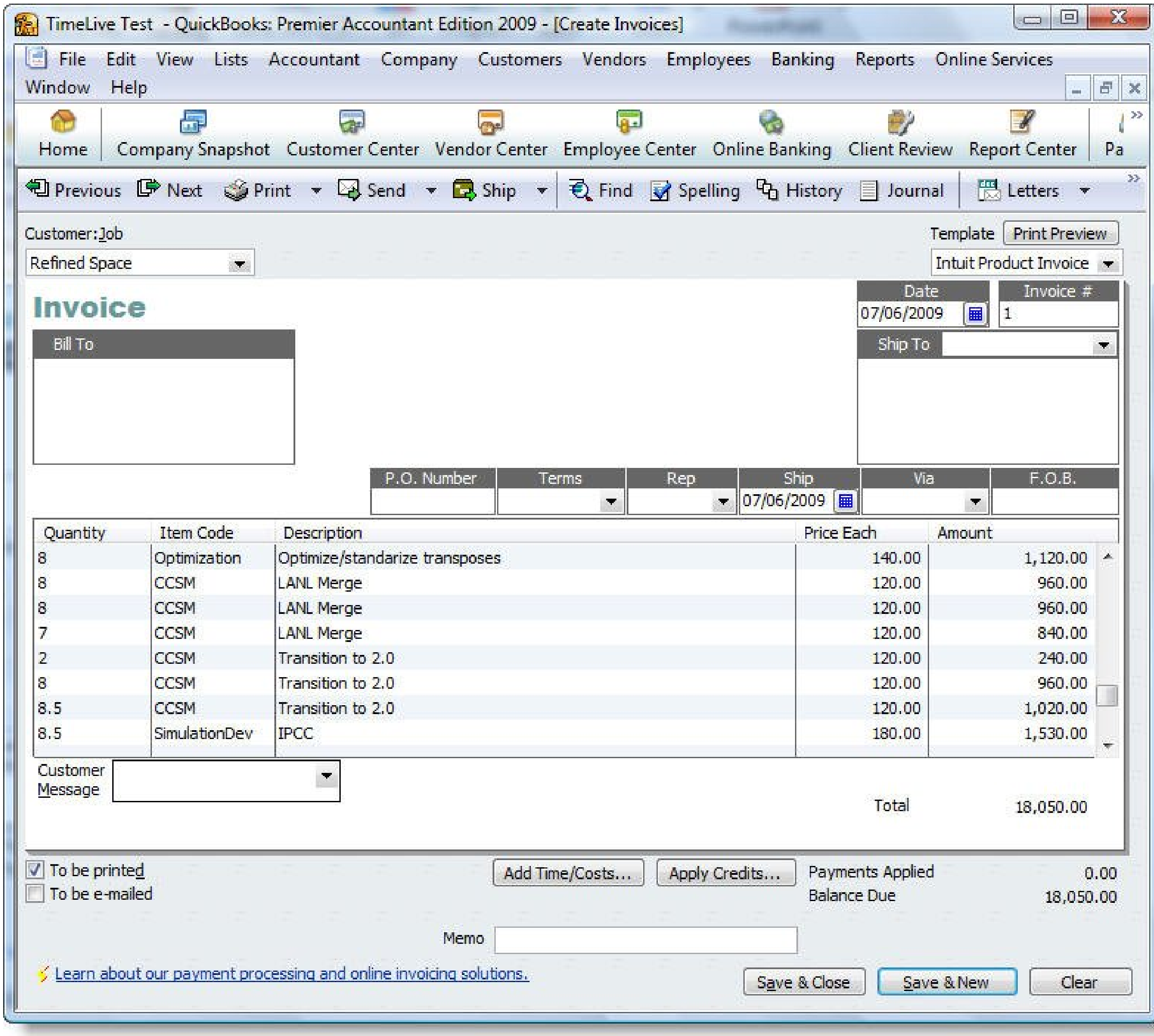
1. Open **[Customers] —> [Create Invoices]** in Quickbooks.
2. Select your customer whose Invoice you want to create in **[Customer:Job]** dropdown.
3. Quickbooks will prompt for importing all pending time and cost items.



4. Click on **[OK]** to import all outstanding billable time and cost in Quickbooks Invoice.
5. Quickbooks now show all outstanding billing items. Just click on **[Select All]** to select all time entries.



6. Click on **[OK]** to import all selected time records in Quickbooks.
7. Quickbooks will import all-time records in Customer Invoice. An accountant can then proceed this invoice as per their own defined organization policies.


TimeLive Test - QuickBooks: Premier Accountant Edition 2009 - [Create Invoices]
File Edit View Lists Accountant Company Customers Vendors Employees Banking Reports Online Services
Window Help
Home Company Snapshot Customer Center Vendor Center Employee Center Online Banking Client Review Report Center
Previous Next Print Send Ship Find Spelling History Journal Letters
Customer:Job
Refined Space
Template Print Preview
Intuit Product Invoice
Invoice
Date 07/06/2009
Invoice # 1
Bill To
Ship To
P.O. Number Terms Rep Ship Via F.O.B.
07/06/2009
Quantity Item Code Description Price Each Amount
8 Optimization Optimize/standarize transposes 140.00 1,120.00
8 CCSM LANL Merge 120.00 960.00
8 CCSM LANL Merge 120.00 960.00
7 CCSM LANL Merge 120.00 840.00
2 CCSM Transition to 2.0 120.00 240.00
8 CCSM Transition to 2.0 120.00 960.00
8.5 CCSM Transition to 2.0 120.00 1,020.00
8.5 SimulationDev IPCC 180.00 1,530.00
Customer Message
Total 18,050.00
To be printed
To be e-mailed
Add Time/Costs...
Apply Credits...
Payments Applied 0.00
Balance Due 18,050.00
Memo
Learn about our payment processing and online invoicing solutions.
Save & Close
Save & New
Clear

13.2.3. QuickBooks Online Integration with TimeLive

TimeLive Hosted Account can easily be integrated with QuickBooks online.
To Integrate TimeLive with QuickBooks, Administrator must have existing TimeLive Hosted Account and QuickBooks Intuit Account.

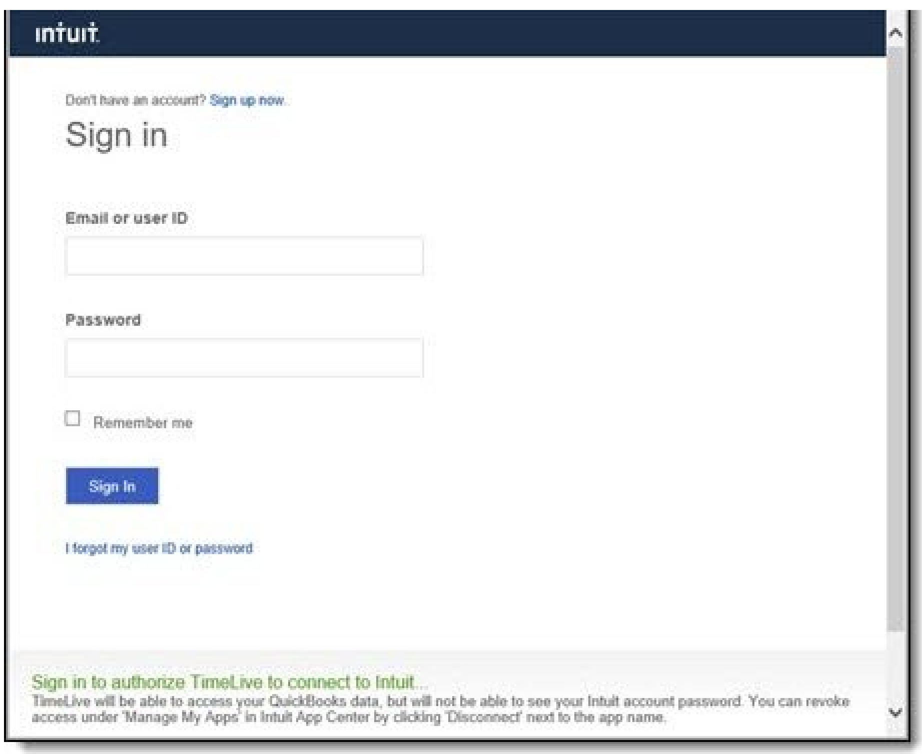
Setup:

Setting up TimeLive requires few simple steps. which are mentioned below,

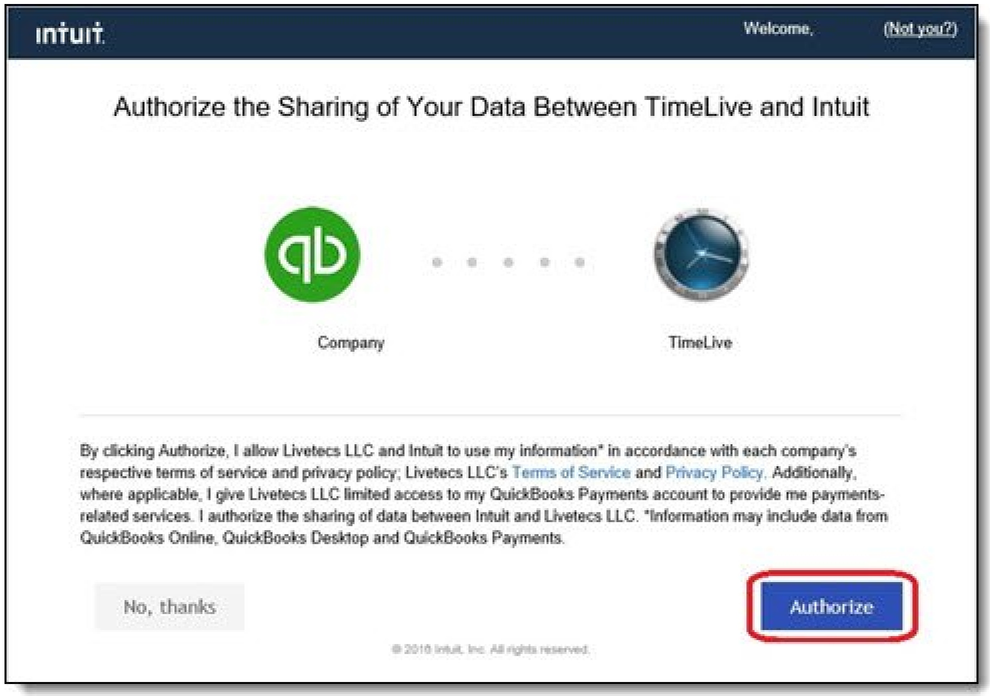
1. Login to **TimeLive Hosted Account**.
2. Navigate to **Admin Options > QuickBooks Online** under Integration

1. Click on “**Connect to QuickBooks**” Button

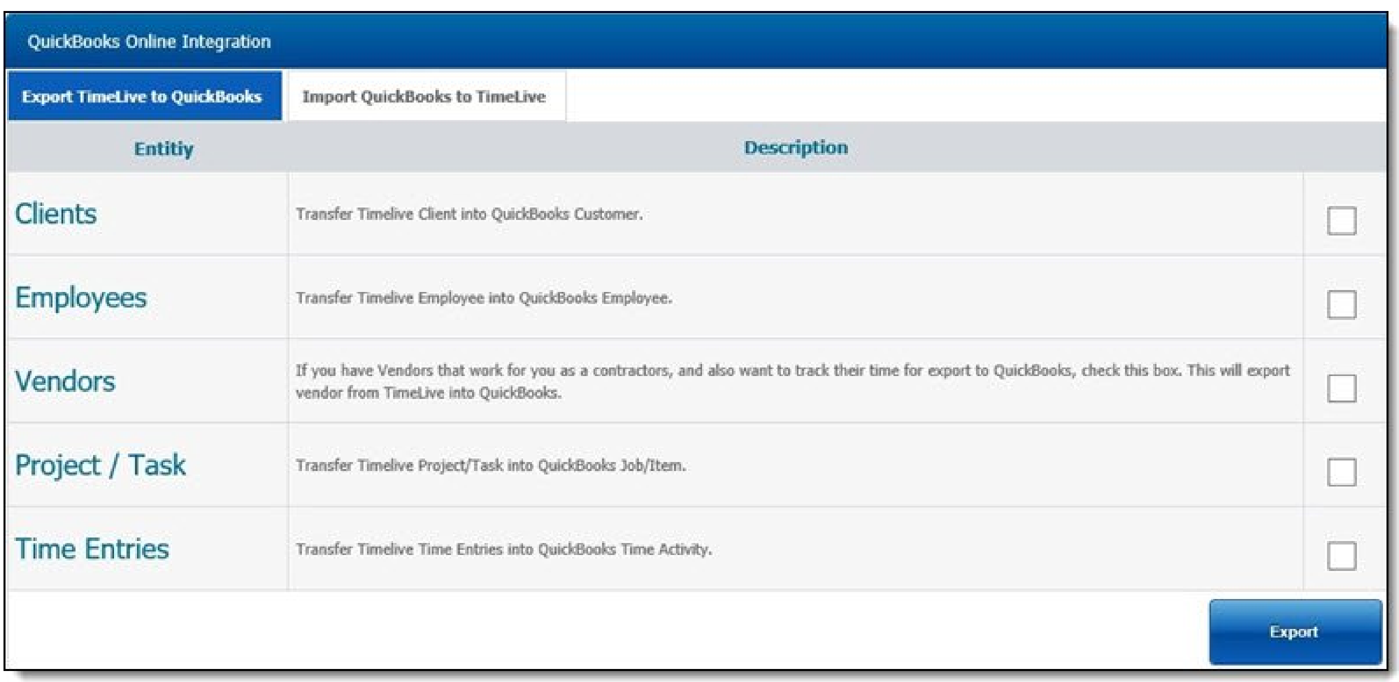
1. Login to **QuickBooks Intuit Account**



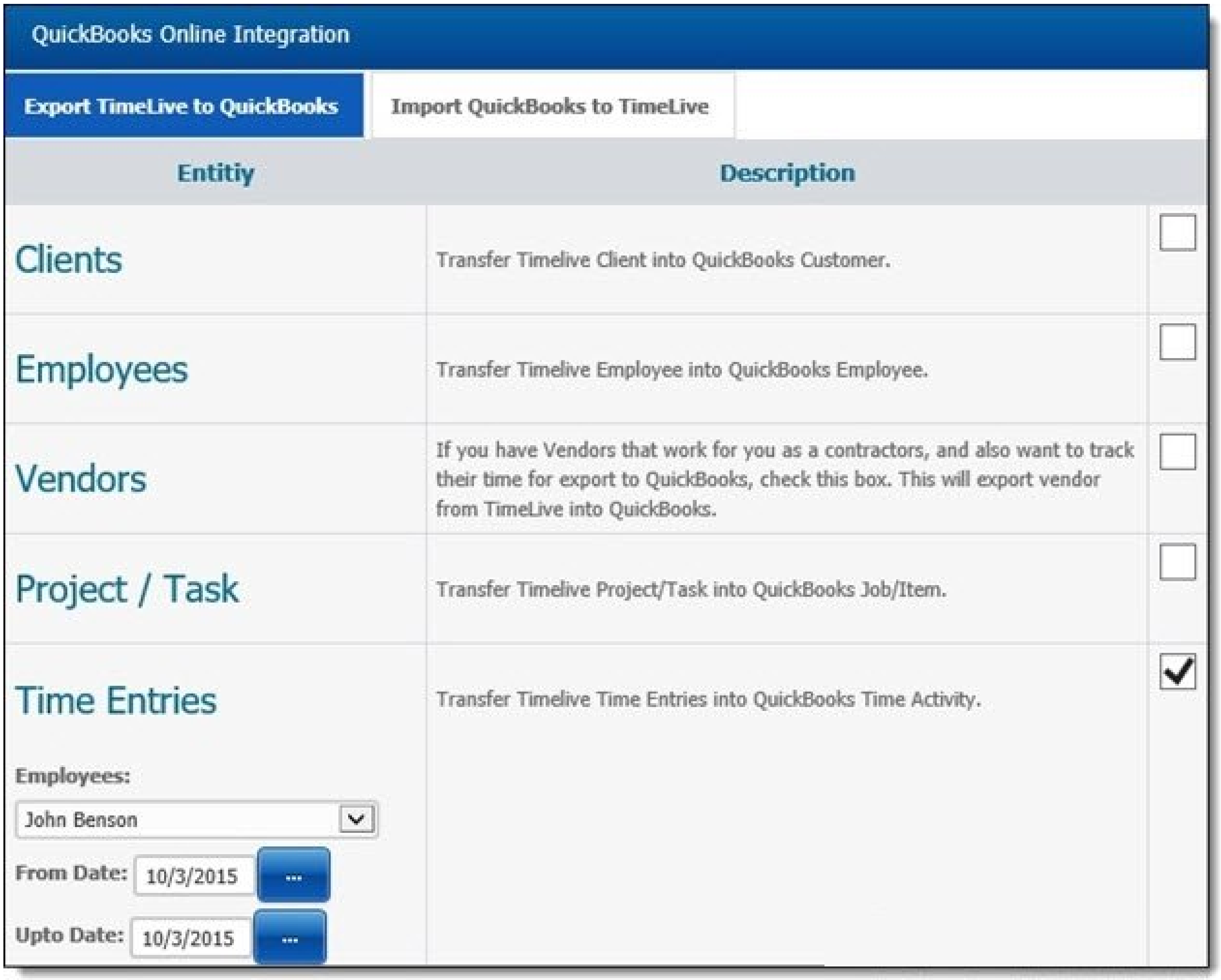
2. Click **Authorize** Button to Integrate both TimeLive and QuickBooks Account



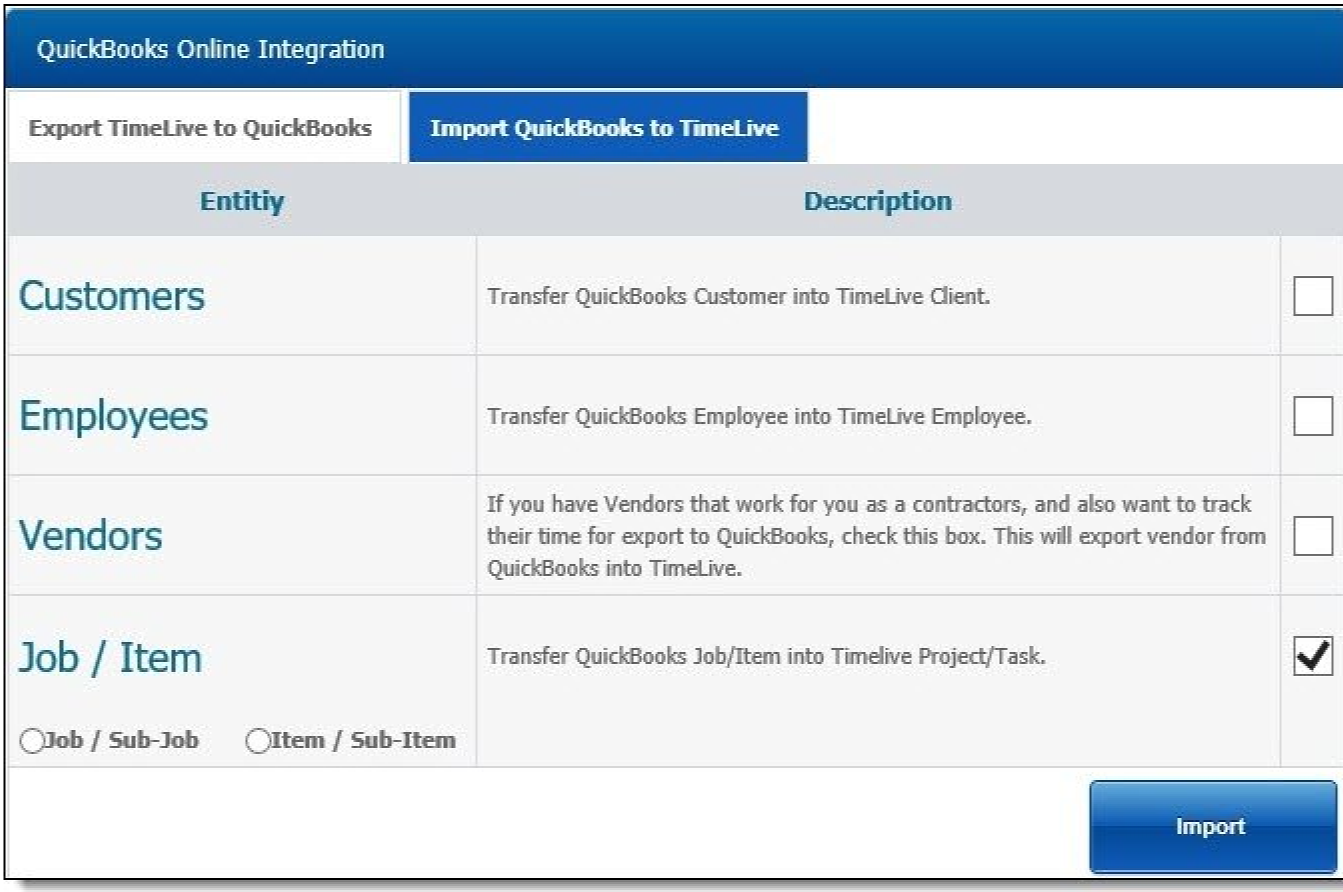
Now, you are able to **Import/Export** your data between both accounts.



An administrator can export **Time Entries** of **Specific Employee** to **QuickBooks** with **Date Range**. See below image.



During **Import**, Administrator can select option of **Job / Sub-Job** or **Item / Sub-Item**


QuickBooks Online Integration
Export TimeLive to QuickBooks
Import QuickBooks to TimeLive
Entitiy
Description
Customers
Transfer QuickBooks Customer into TimeLive Client.
Employees
Transfer QuickBooks Employee into TimeLive Employee.
Vendors
If you have Vendors that work for you as a contractors, and also want to track their time for export to QuickBooks, check this box. This will export vendor from QuickBooks into TimeLive.
Job / Item
Transfer QuickBooks Job/Item into Timelive Project/Task.
Job / Sub-Job
Item / Sub-Item
Import

13.3. Configuring Active Directory Authentication in TimeLive

You can configure TimeLive to authenticate the requester login with the active directory (AD). This provides you an advantage of not having to remember too many passwords. For this, you should configure AD authentication, then any password change that is made in the AD will also reflect in TimeLive. So the requesters can login using the login name and password of the system.

TimeLive Active Directory integration concepts:

TimeLive supports two types of active directory authentication.

1. First option is automated way through which user will be automatically added as an employee in TimeLive if that particular user is the member of the corresponding mapped group in Active Directory.
    a. Using role management in admin options, you can configure TimeLive role mapping with Active Directory group. A user will automatically be assigned to that role in TimeLive if he is the member of the corresponding group member in active directory.
    b. For example, if an employee is a member of the TimeLive User group in **[Active Directory]**, he will be automatically added as an employee in timelive on the first login with **[User]** role in TimeLive.
    c. By default, TimeLive comes with two TimeLive roles with their active directory corresponding roles mapping. You can define your AD Group in TimeLive Roles management page.
        i. AD Group **[TimeLiveAdministrator]** —> map to TimeLive **[Administrator]** role.
        ii. AD Group **[TimeLiveUser]** —> map to TimeLive **[User]** role.
2. Second way is to add all your employee manually in TimeLive. The administrator can define active directory username of employees during new employee creation. After adding an employee with their active directory username, they can login in TimeLive using their active directory username and password.

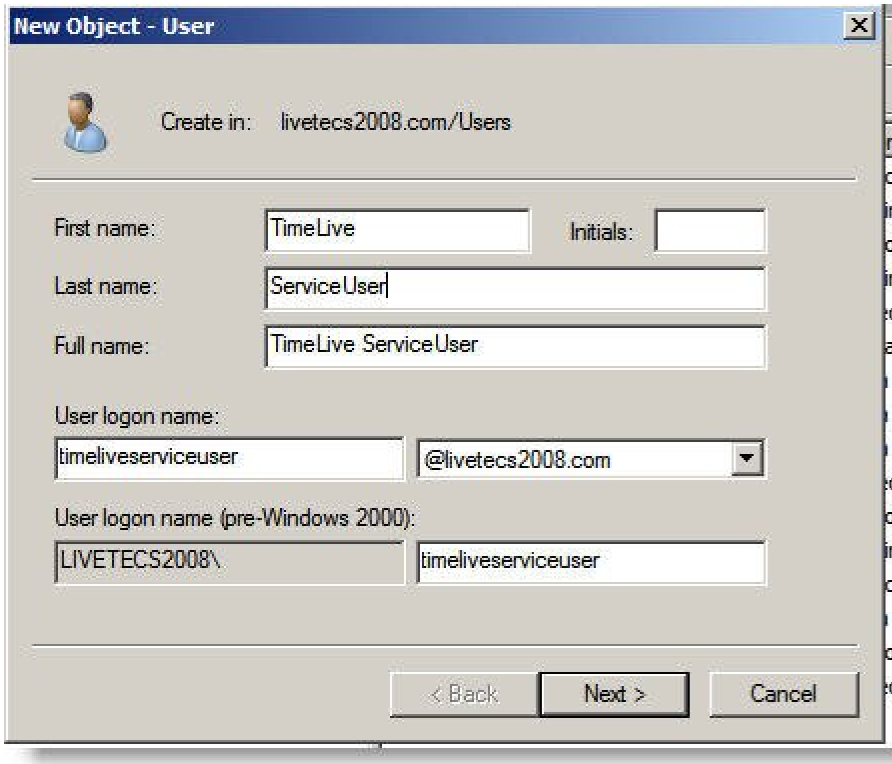
Step by step: Active Directory integration:

TimeLive active directory integration required setup on two places. One is in active directory and second is defining server parameters in system configuration pages.

Step1: Changes required in Active Directory:

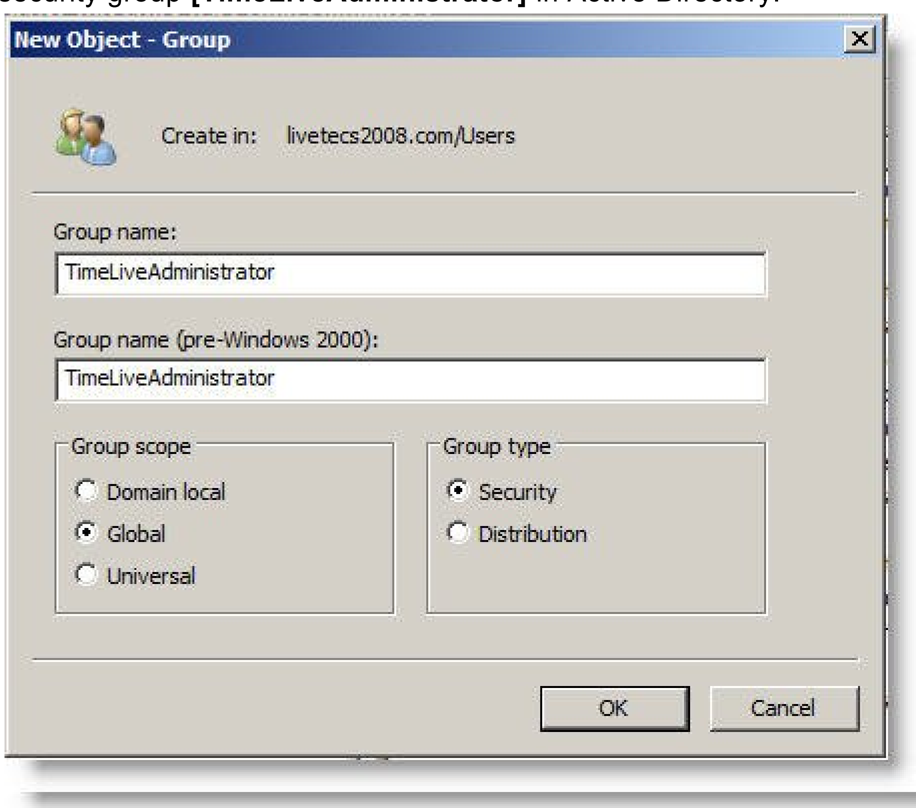
Here are the steps which are required to be done on active directory side.

1. IT Administrator should decide first, exactly which Active Directory username will work as TimeLive Administrator. In this help section, we have assumed one AD user with username **[LivetecsIT]**
2. Create a new **[TimeLive Service User]**. TimeLive APIs will use this username and password to communicate to Active Directory. Create a service user with name "**timeliveserviceuser**".

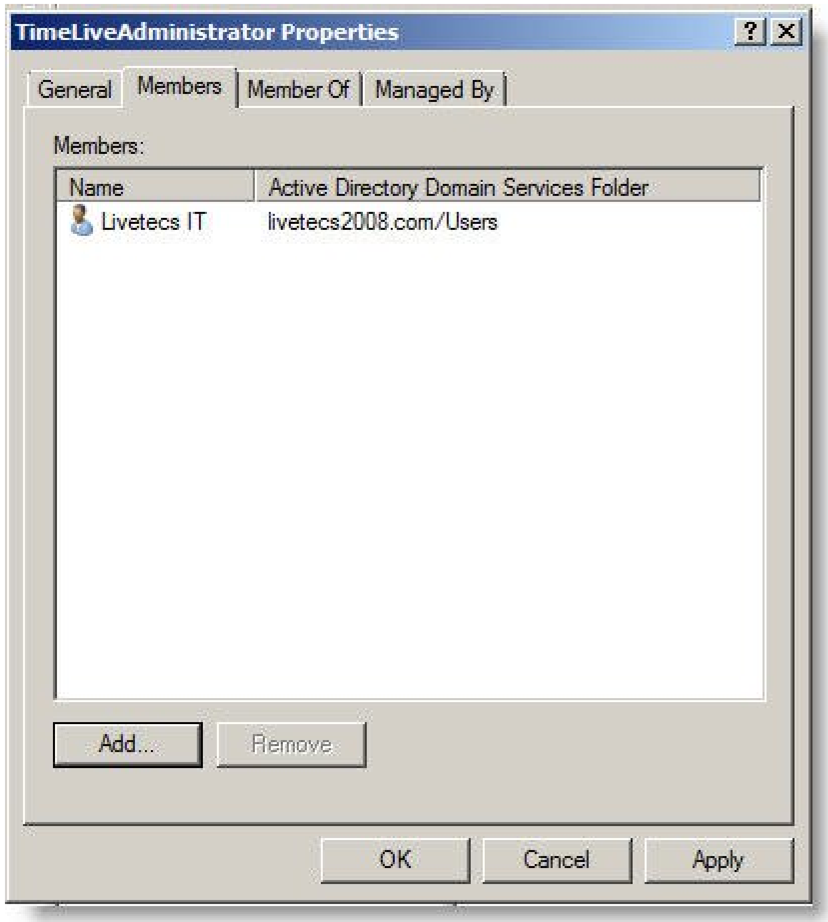


3. Define some password for "**TimeLive Service User**" and make sure that **[Password never expires]** should be checked and **[User must change password at next login]** should be unchecked.

4. Create a new security group **[TimeLiveAdministrator]** in Active Directory.



5. Now assign your user which you want to work as TimeLive Administrator, in **[TimeLiveAdministrator]** group. This user will become **[Administrator]** in TimeLive. Make sure that this user should have **[First Name]**, **[Last Name]** and **[EmailAddress]** information are filled in Active Directory.



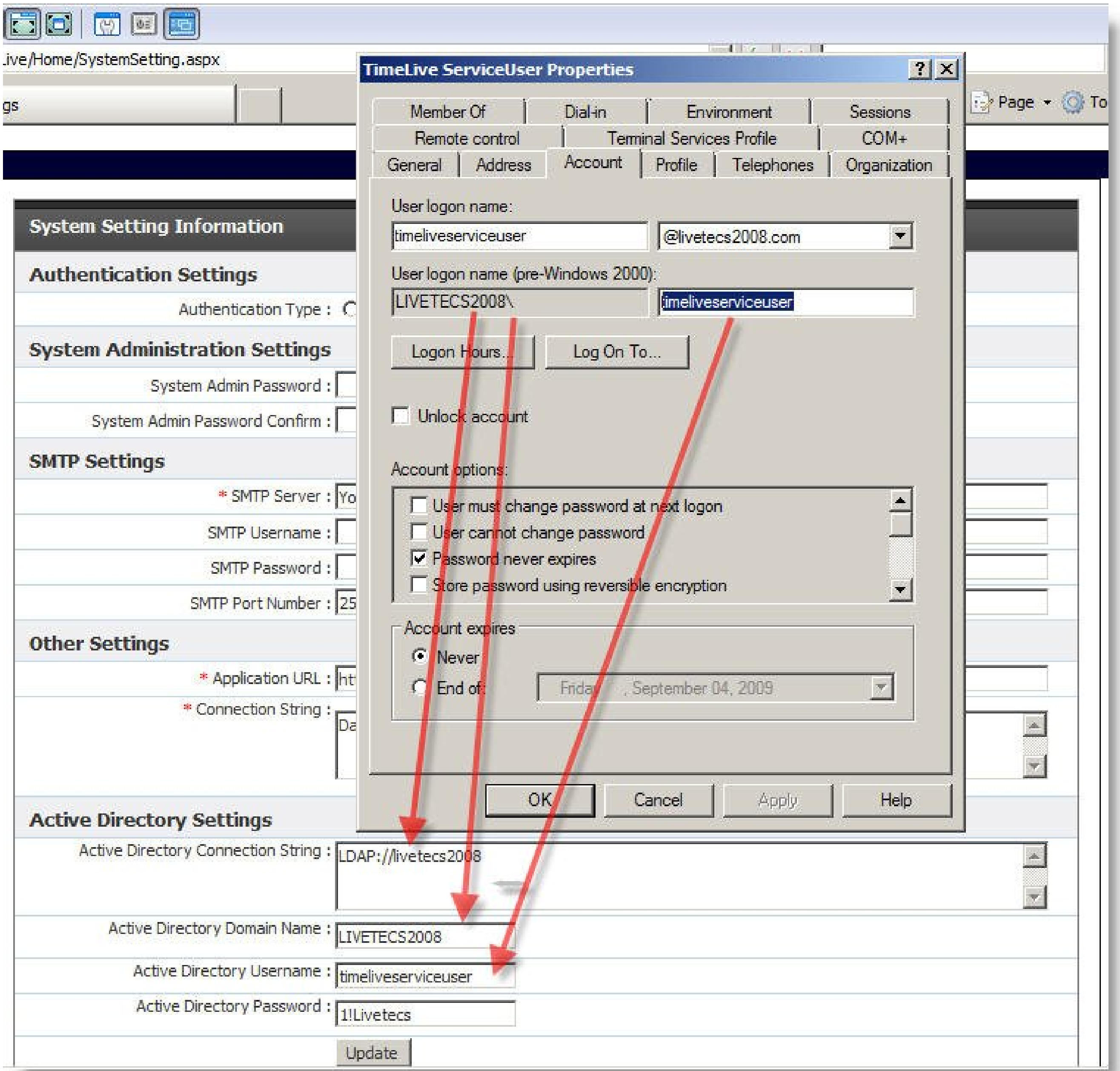
Step 2: Changes required in TimeLive:

1. On first-time execution, after database setup, TimeLive first open **[Account Add]** page where a user can enter their organization and administrator user information. Do not fill this form if you are going to setup on **[Active Directory]** integration.
2. Open **[System Configuration]** page (**http://timeliveurl/home/systemsetting.aspx**), where you can define system level parameters like active directory integration, database connection string and SMTP server.

[More about system setting page]

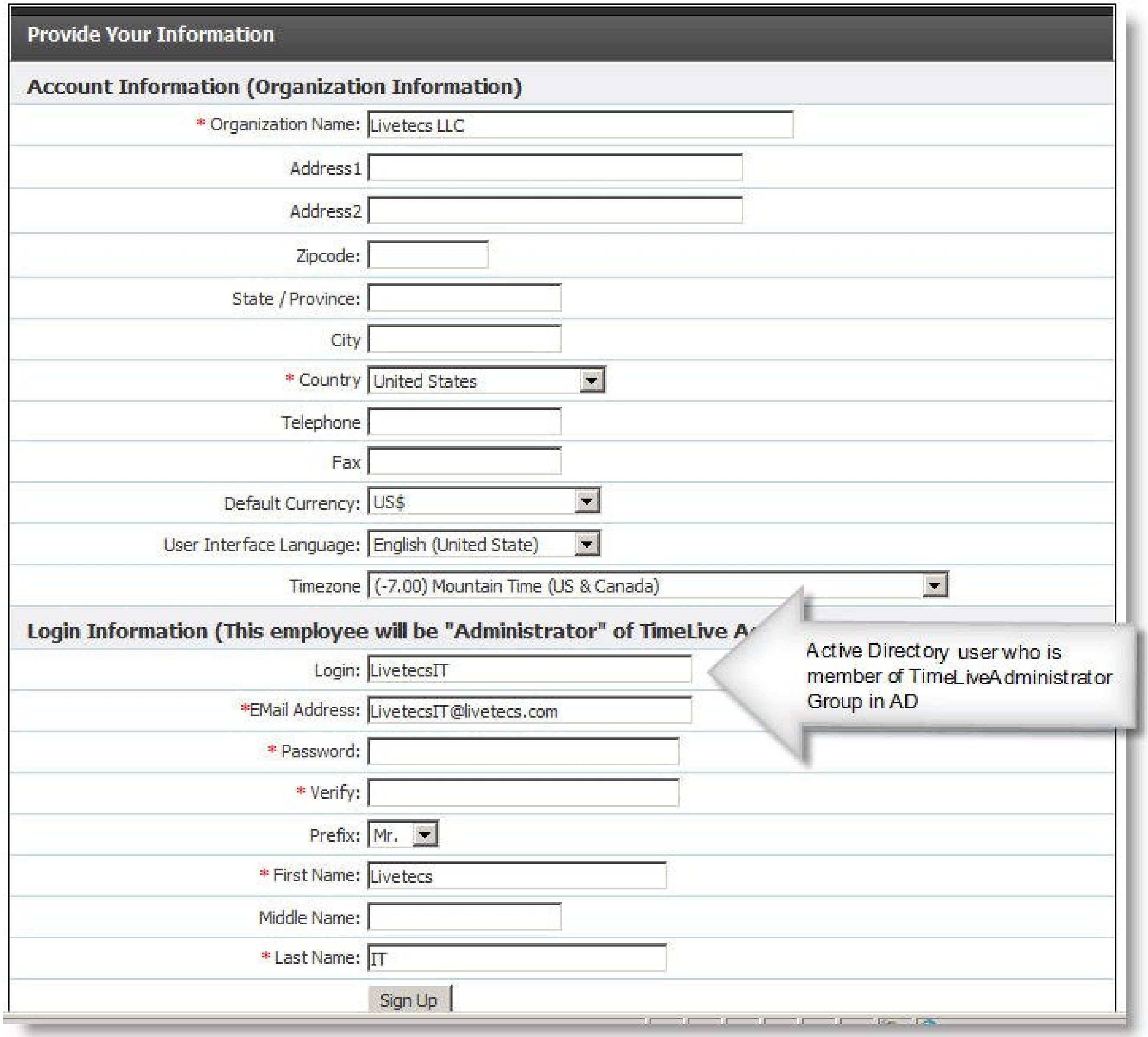
1. Select **[Active Directory Authentication]** checkbox to select your authentication mode as Active Directory.

2. Enter “**LDAP://YourServerName**” in **[Active Directory Connection String]**. Yourservername should be replaced with physical server name where the Active Directory is installed. Please see below screenshots to get an exact idea of which value will appear where.



3. Enter domain name in **[Active Directory Domain Name]** field. The domain name should be pre-windows 2000 server name instead of actual domain name.
4. Enter your **[TimeLive Service User]** username in **[Active Directory Username]** field. Username should be in an exact same case which is in Active Directory. Make sure that your (pre-windows 2000) username and your actual username are same.
5. Enter **[TimeLive Service User]** user’s password in **[Active Directory Password]** field.

6. Click on **[Update]** to update these changes.
7. After **update**, TimeLive will open new account add page.



8. Enter your organization information in the top portion.
9. Enter **[TimeLive Administrator username]** which you earlier assigned in **[TimeLiveAdministrator]** group in Active Directory. TimeLive will automatically populate FirstName, LastName and email address from Active Directory.
10. Enter Active Directory password and verify password of **[TimeLive Administrator user]**. This should be active directory password of the TimeLive administrator user.
11. Enter First Name, Middle Name, and Last Name.
12. Click on **[Sign up]** to complete Active directory integration steps.

13. This administrator can now sign-in in TimeLive using his **[TimeLive Admin]** (LivetecsIT in above case) active directory username and password.
14. Now administrator can add other employees using **[Administration] —> [Employees]** option by specifying AD username in **[User name]** field.
15. New employee will be automatically added if they are member AD Group whose mapping is defined with TimeLive roles.

Note: Migrating from standard authentication to Active Directory authentication:

1. In case, if Active Directory is being set up for switching from already setup standard authentication to Active Directory authentication, system will redirect to login page directly instead of account add page. An administrator can log in with TimeLive admin user created using instructions mentioned above.
2. Just make sure that email address of TimeLiveAdministrator user should not be already defined to some other user.
3. This administrator can now login in TimeLive using his **[TimeLive Admin]** (LivetecsIT in above case) active directory username and password.
4. After login, the administrator should edit every employee which are already defined in TimeLive and change value of “**username**” field from their email address to Active Directory login id.
5. After having Active Directory login id in username field in employee form, the employee then will be able to log in using their Active Directory username and Active Directory password with their data, which they already have in TimeLive.

13.4. Active Directory authentication with TimeLive On-Cloud

TimeLive supports single sign-on (SSO) logins through SAML 2.0/W-Federation if you're on the Professional or Enterprise plans. A SAML 2.0/W-Federation identity provider (IDP) can take many forms, one of which is a self-hosted Active Directory Federation Services (ADFS) server. ADFS is a service provided by Microsoft as a standard role for Windows Server that provides a web login using existing Active Directory credentials.

Requirements

To use ADFS to log in to your TimeLive instance, you need the following components:

1. An Active Directory instance where all users have an email address attribute.
2. A TimeLive instance with Hosted plans.
3. A server running Microsoft Server 2012 or 2008. This guide uses screenshots from Server 2012 R2, but similar steps should be possible on other versions.
4. A SSL certificate to sign your ADFS login page and the fingerprint for that certificate.
5. If you're using host mapping in your TimeLive instance, an installed certificate for hosted SSL.

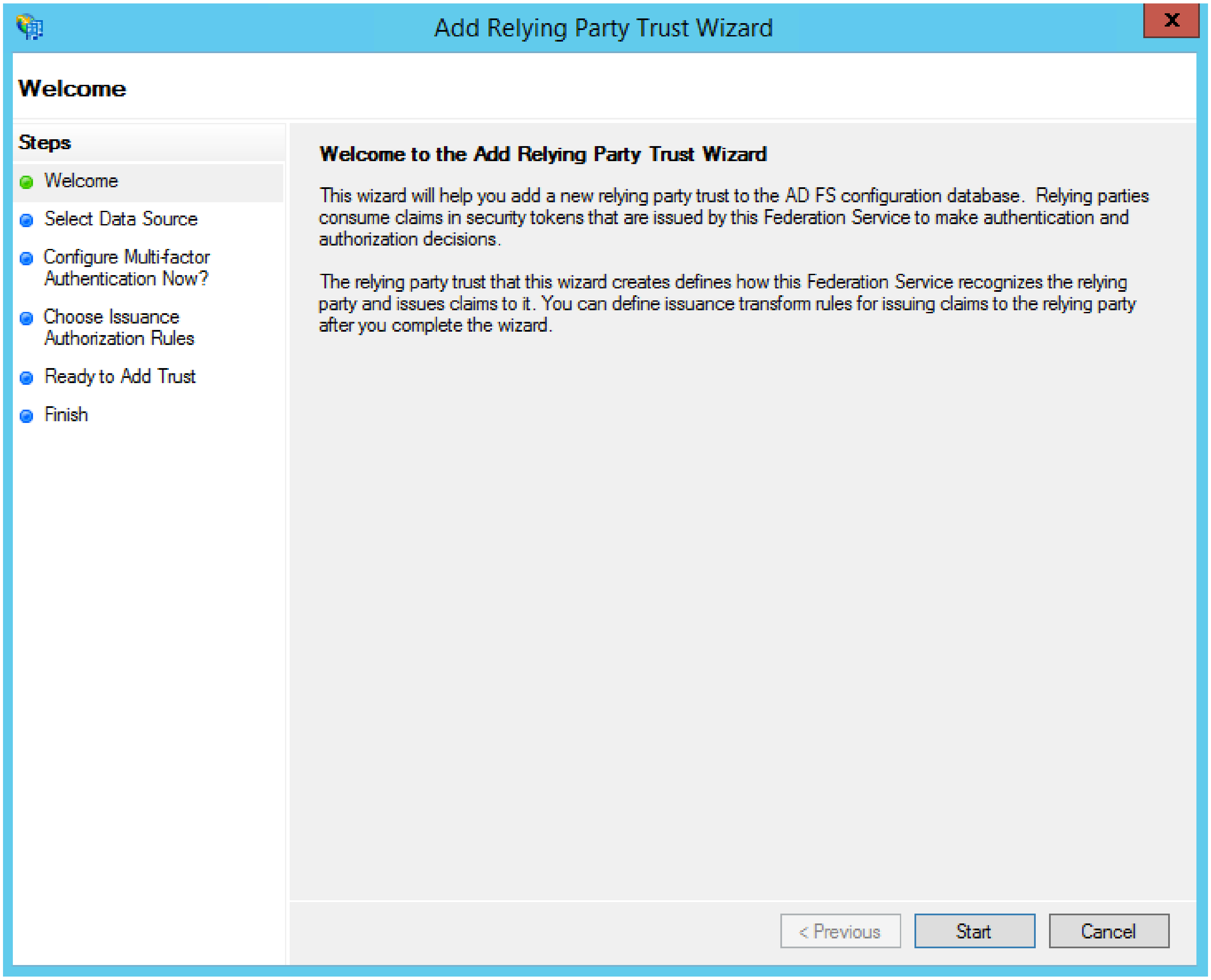
After you meet these basic requirements, you need to install ADFS on your server. Configuring and installing ADFS is beyond the scope of this guide, but is detailed in a Microsoft KB article.

When you have a fully installed ADFS installation, note down the value for the 'SAML 2.0/W-Federation' URL in the ADFS Endpoints section. If you chose the defaults for the installation, this will be '/adfs/ls/'.

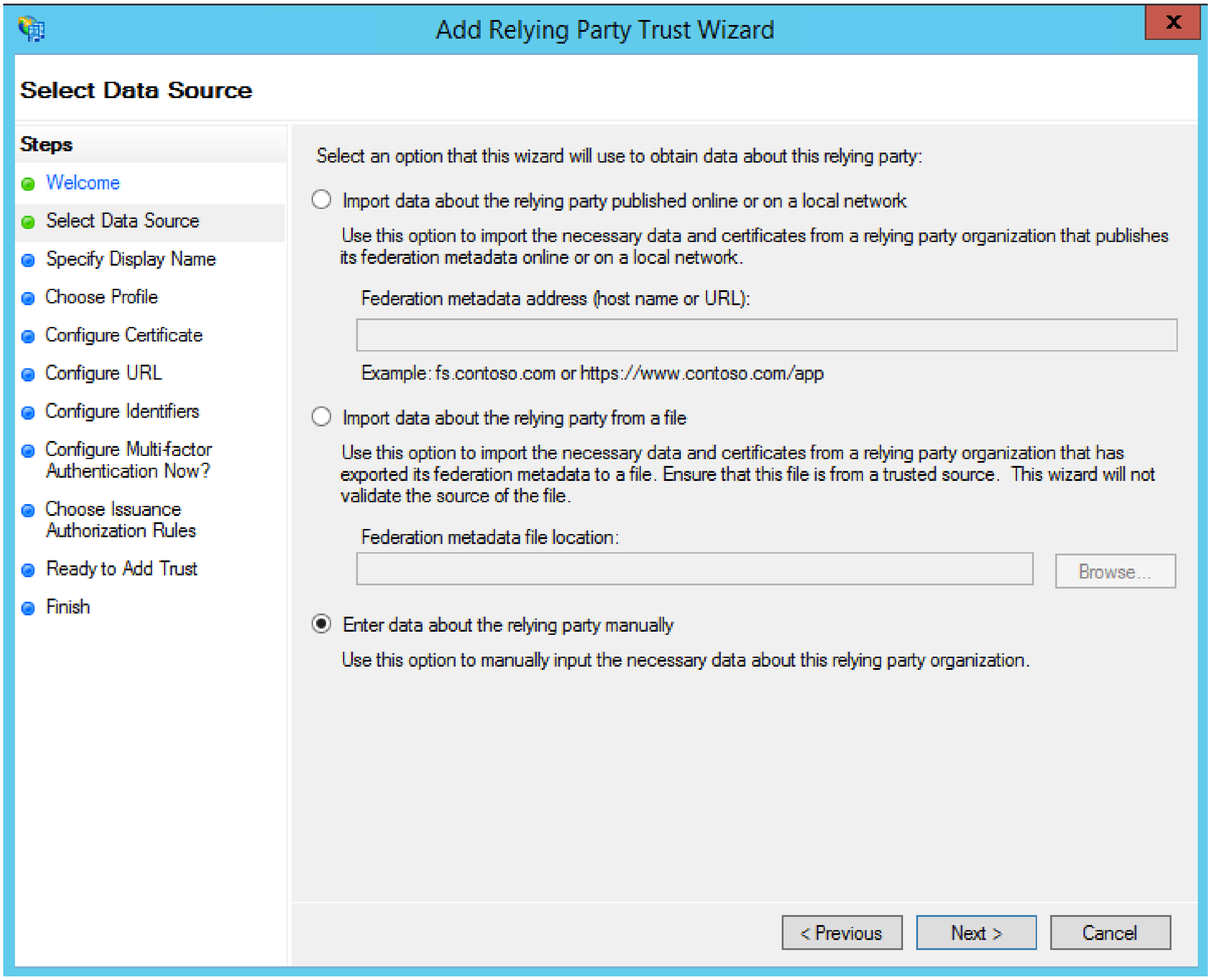
Step 1 – Adding a Relying Party Trust

At this point you should be ready to set up the ADFS connection with your TimeLive instance. The connection between ADFS and TimeLive is defined using a Relying Party Trust (RPT).

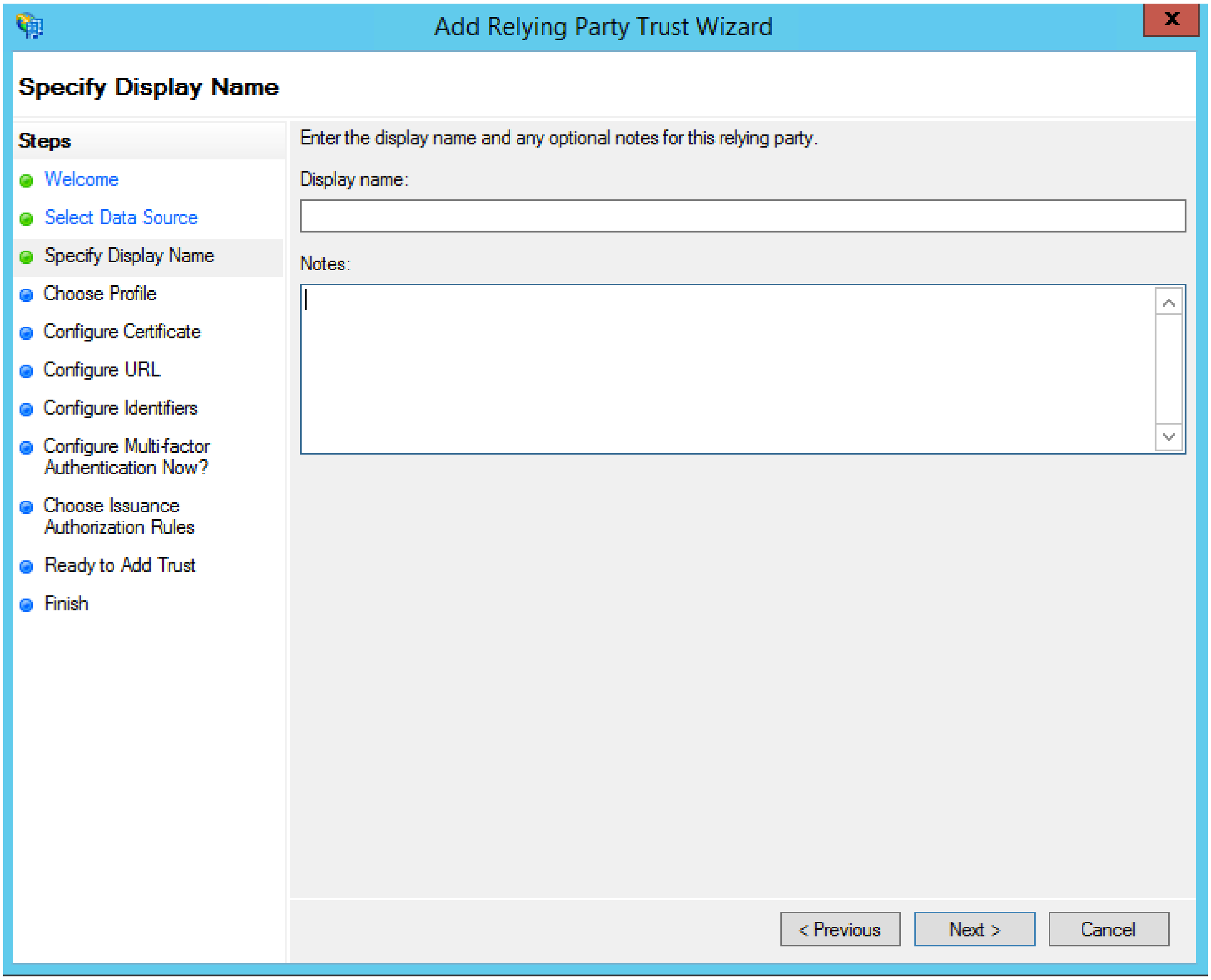
Select the **Relying Party Trusts** folder from **AD FS Management**, and add a new **Standard Relying Party Trust** from the Actions sidebar. This starts the configuration wizard for a new trust.



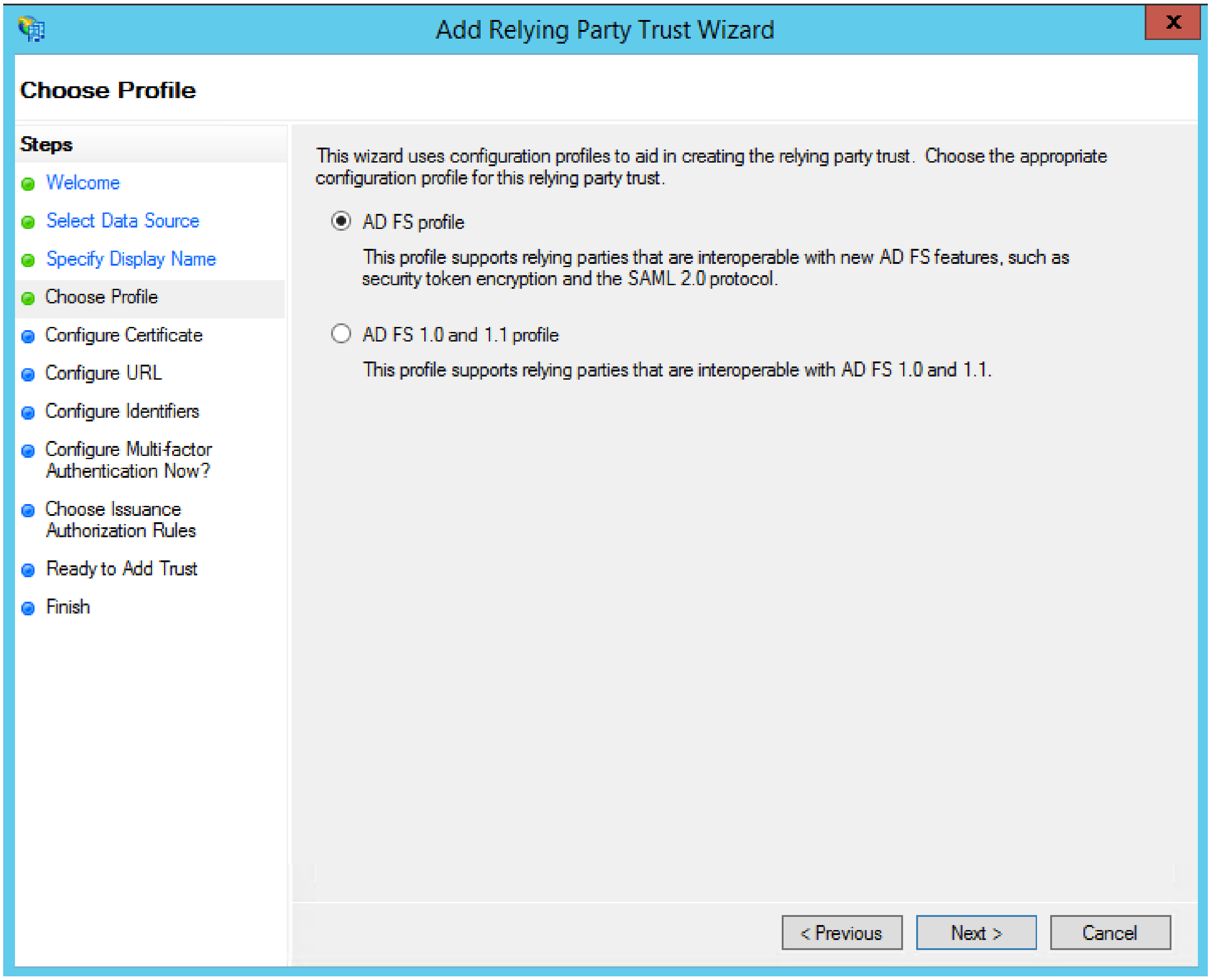
1. In the **Select Data Source** screen, select the last option, **Enter Data About the Party Manually**.



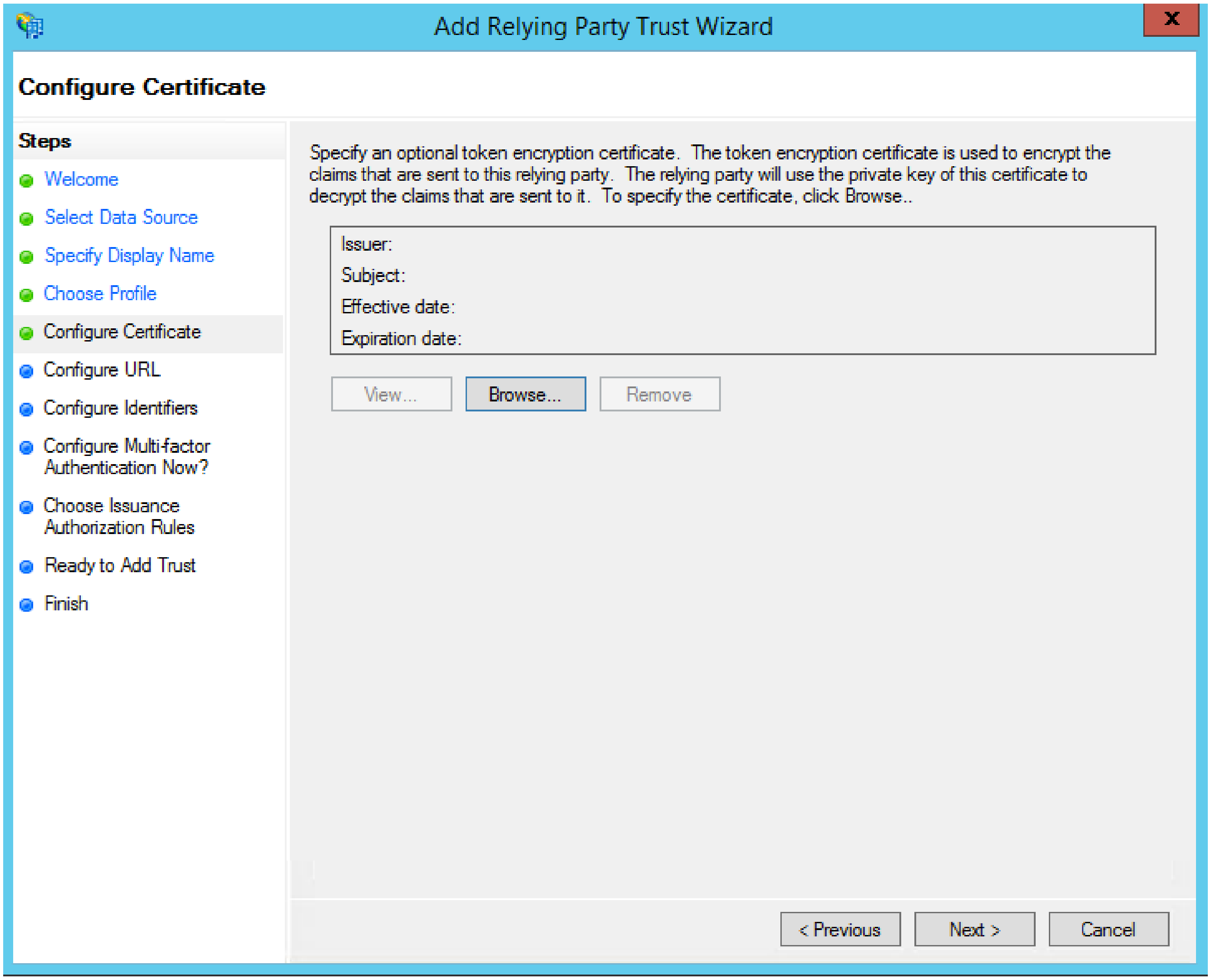
2. On the next screen, enter a **Display name** that you'll recognize in the future, and any notes you want to make.



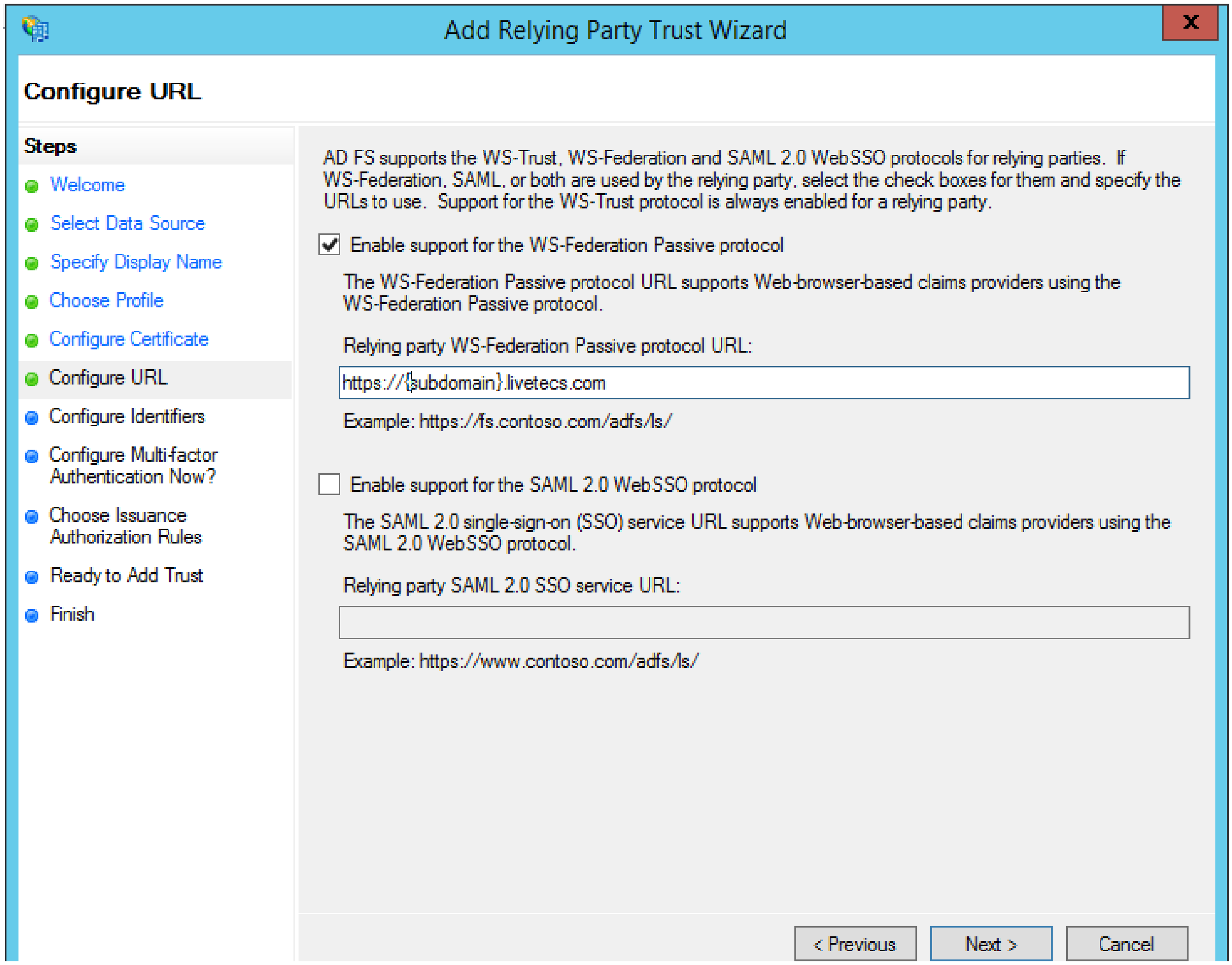
3. On the next screen, select the **ADFS FS profile** radio button.



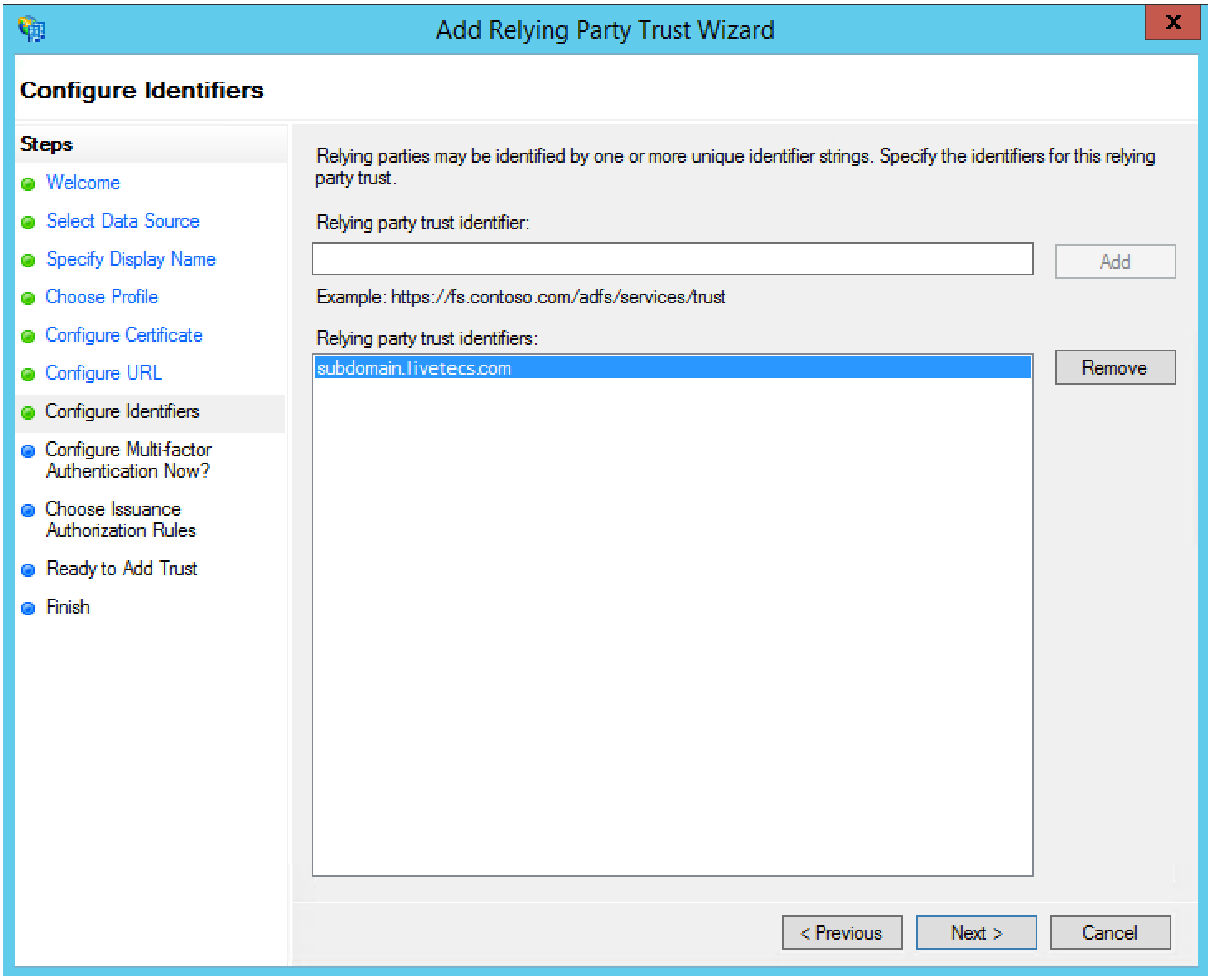
4. On the next screen, leave the certificate settings at their defaults.



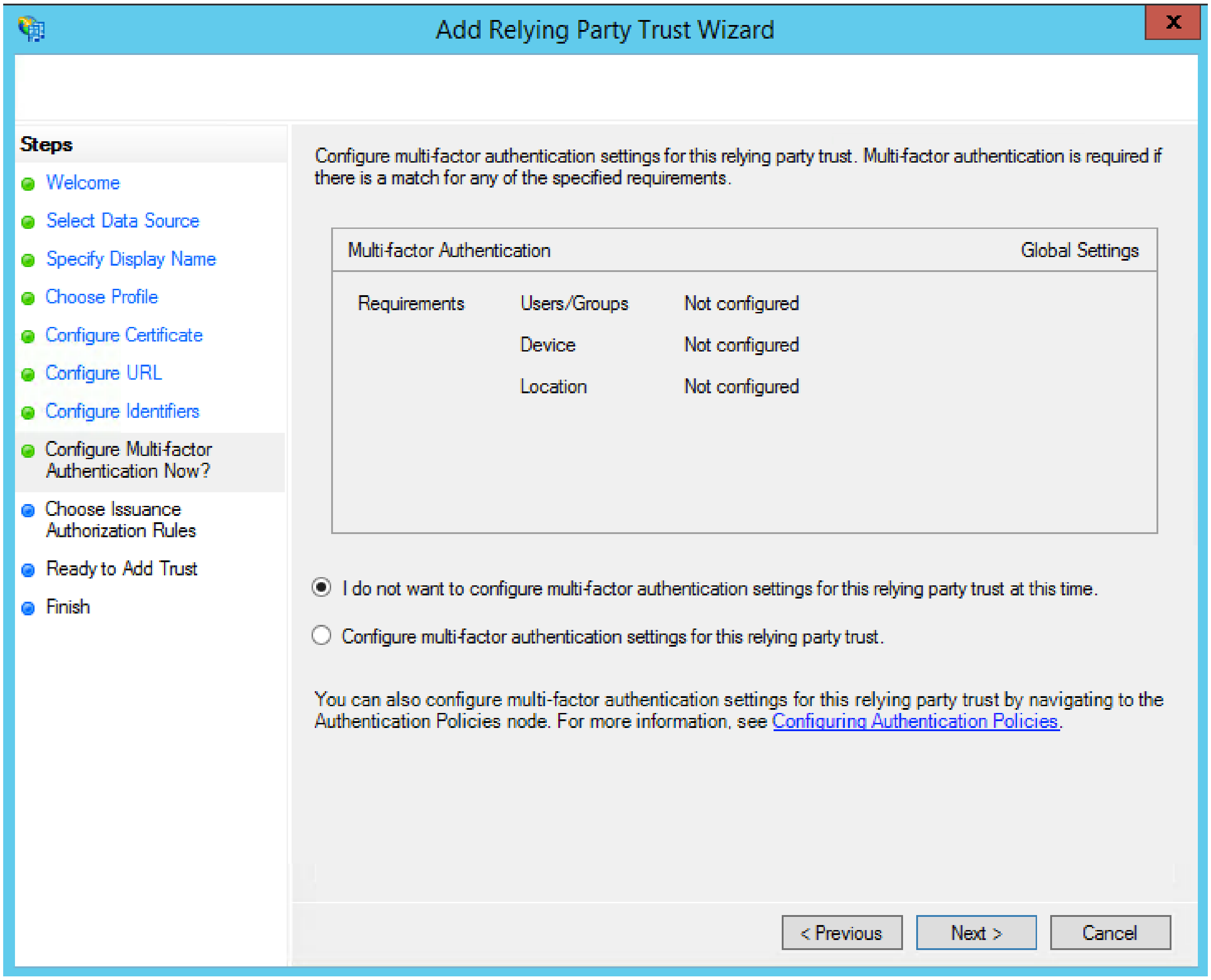
5. On the next screen, check the box labeled **Enable Support for the WS-Federation Passive Protocol**. The service URL will be https://*subdomain*.livetecs.com/, replacing **[subdomain]** with your TimeLive's subdomain. Note that there's no trailing slash at the end of the URL.



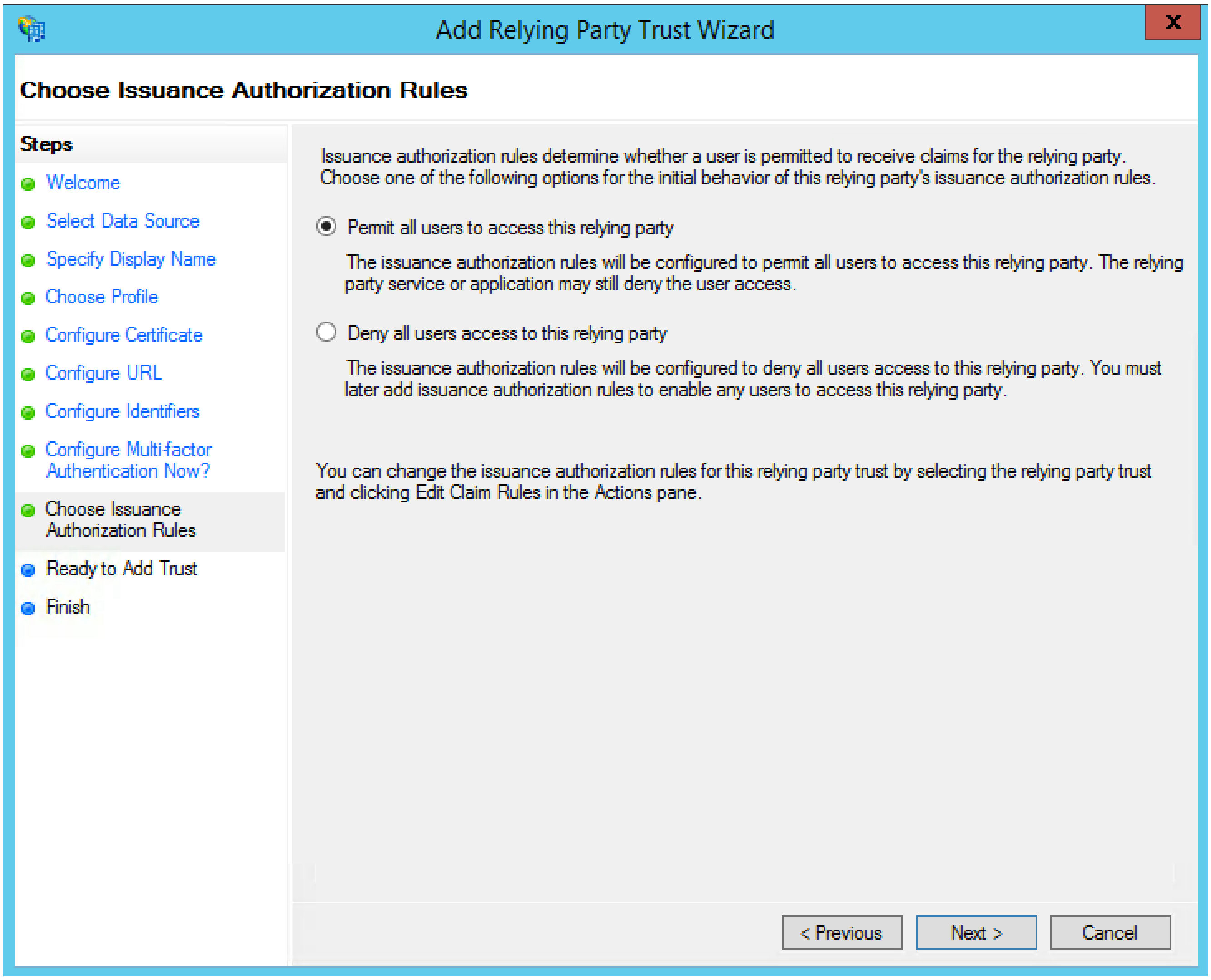
6. On the next screen, add a **Relying party trust identifier** of **subdomain**.livetecs.com, replacing **subdomain** with your TimeLive subdomain.



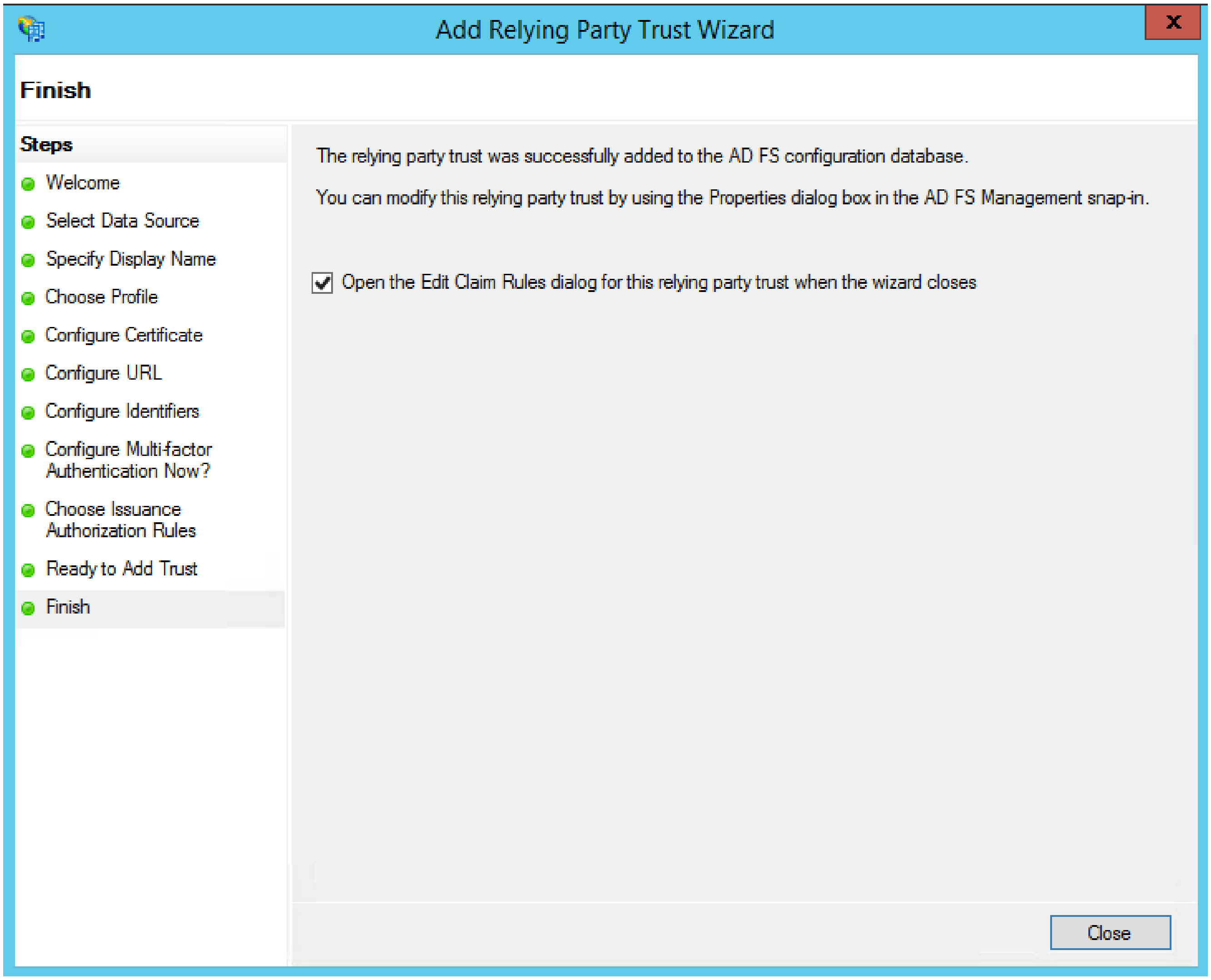
7. On the next screen, you may configure multi-factor authentication but this is beyond the scope of this guide.



8. On the next screen, select the **Permit all users to access this relying party** radio button.

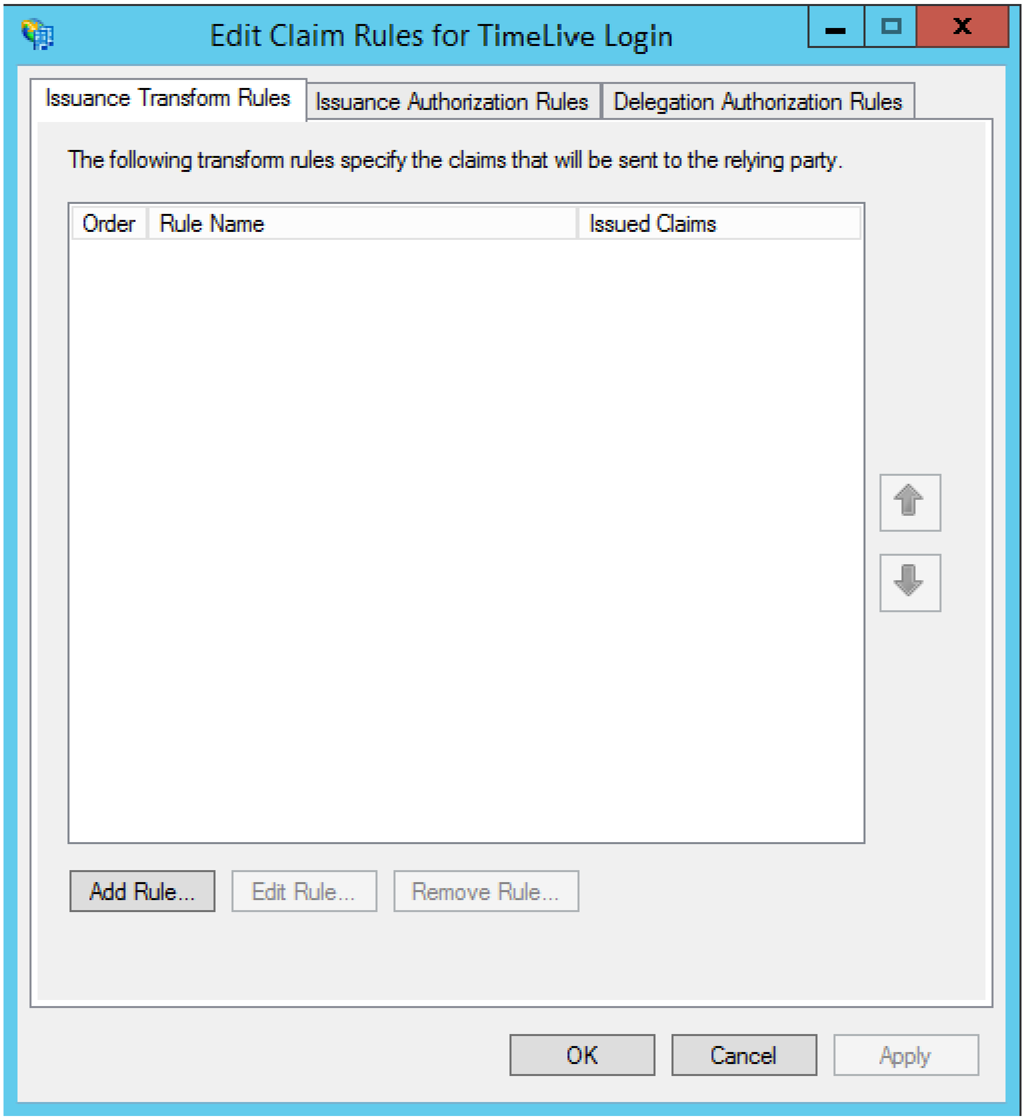


9. On the next two screens, the wizard will display an overview of your settings. On the final screen use the **Close** button to exit and open the Claim Rules editor.

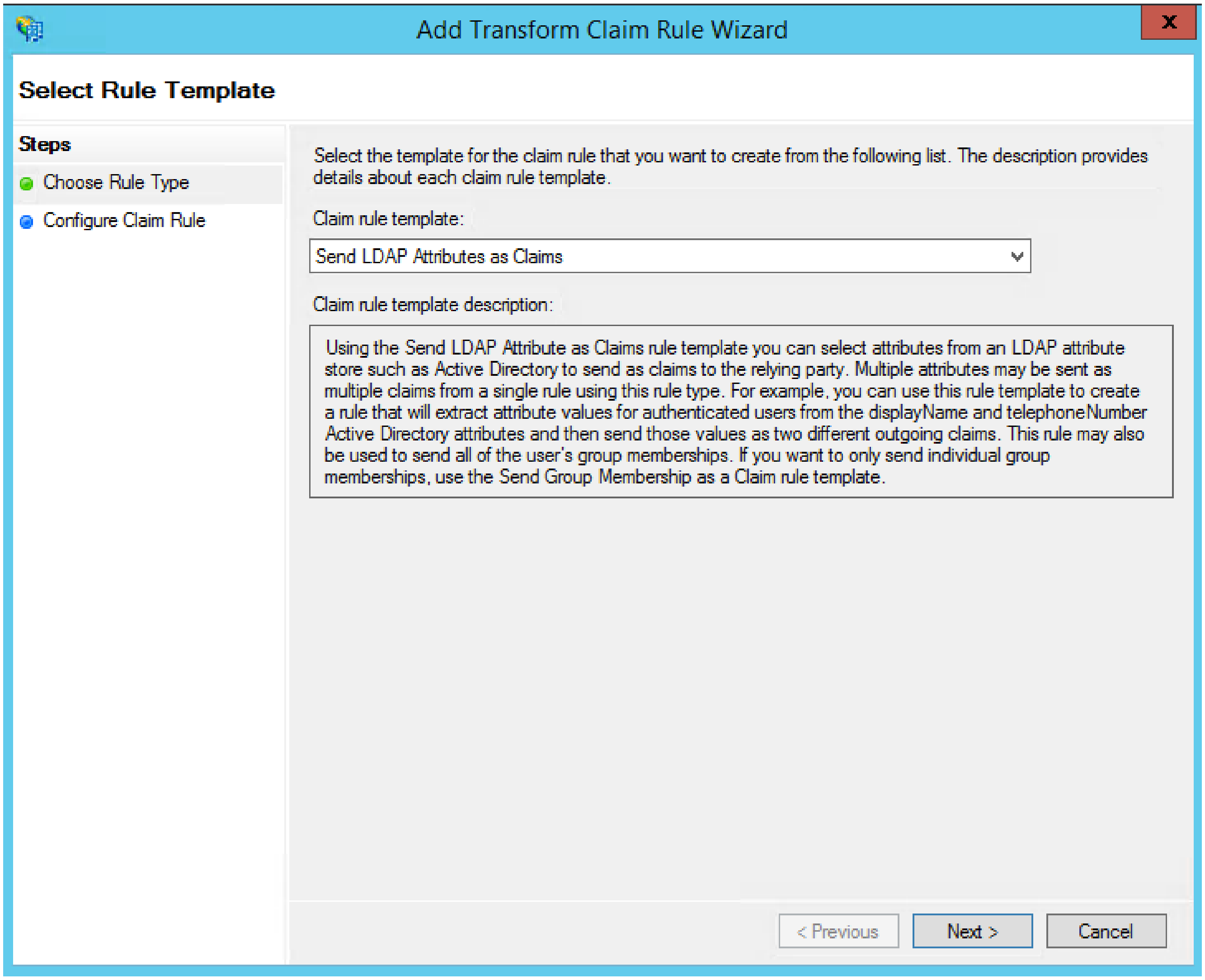


Step 2 – Creating claim rules

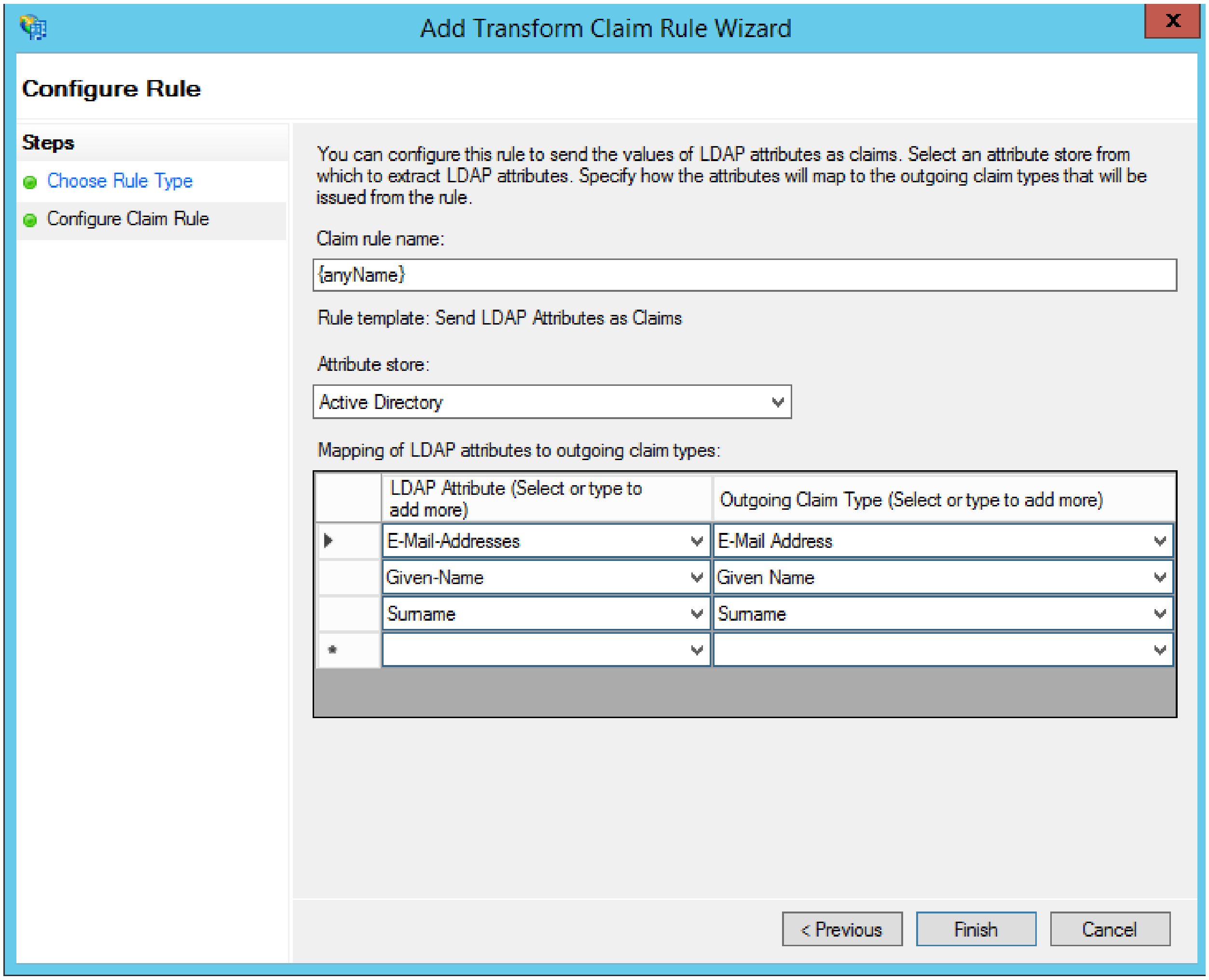
Once the relying party trust has been created, you can create the claim rules and update the RPT with minor changes that aren't set by the wizard. By default the claim rule editor opens once you created the trust. If you want to map additional values beyond authentication



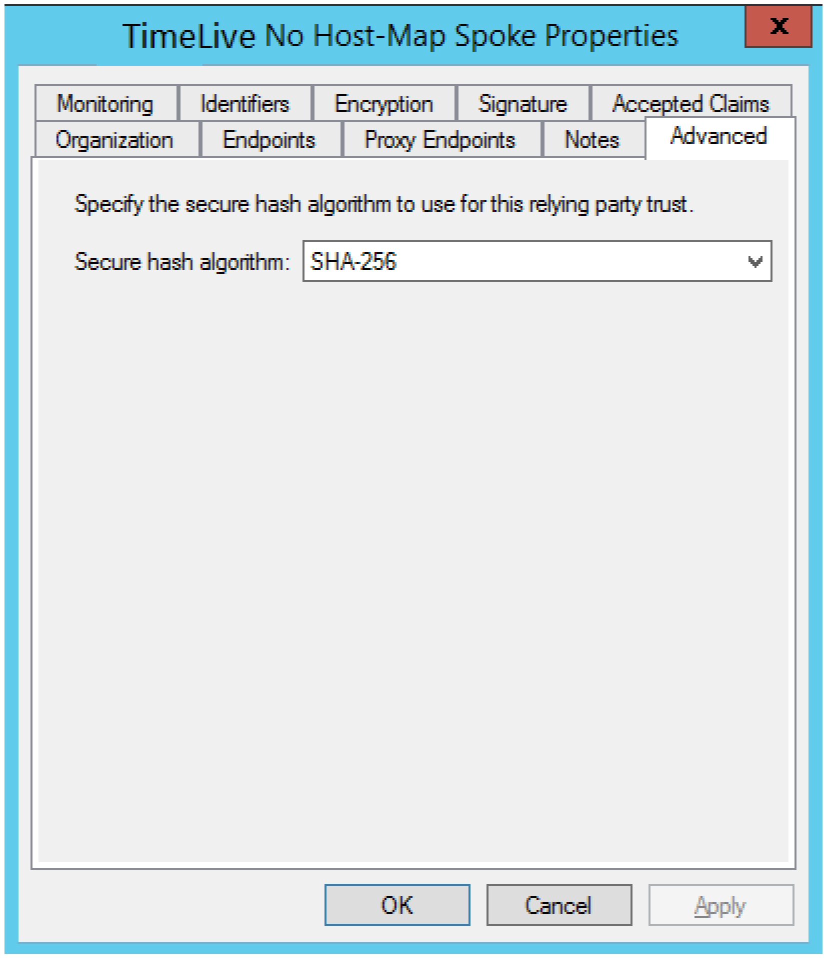
1. To create a new rule, click on **Add Rule**. Create a **Send LDAP Attributes as Claims** rule.



2. On the next screen, using **Active Directory** as your attribute store, do the following:
    a. In 1st row, under the **LDAP Attribute** column and **Outgoing Claim Type**, select **E-Mail Addresses** from Drop-down list.
    b. In 2nd row, select **Given-Name** from Drop-down list.
    c. In 3rd row, select **Surname** from Drop-down list.



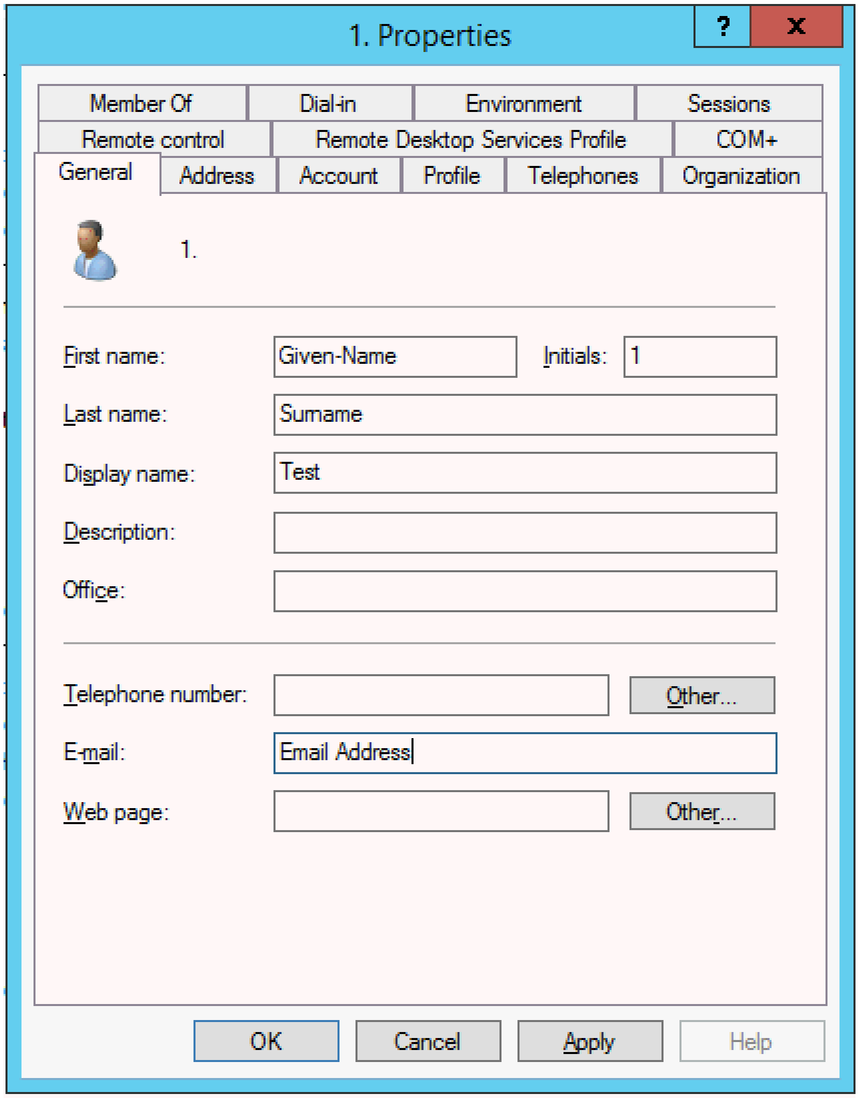
3. Click on **Finish** to save the new rule.
   You still need to adjust a few settings on your relying party trust. To access these settings, select **Properties** from the **Actions** sidebar while you have the RPT selected.
   In the **Advanced** tab, switch from **SHA-1** to **SHA-256** and click **OK**.



Step 3 – Configuring User settings with LDAP

In Active Directory, User attributes should be identical to LDAP Attributes,

1. First Name is Given-Name (Mandatory)
2. Last Name is Surname (Mandatory)
3. Email is E-mail-Address (Mandatory)



Step 4 – Configuring TimeLive

After setting up ADFS, you need to configure your TimeLive instance to authenticate using SAML. You'll use your full ADFS server URL with the SAML endpoint as the SSO URL, and the login endpoint you created as the logout URL.

After you're done, the Security page in the TimeLive admin interface (**Admin Options > Preferences > Security**) should look like this:

You need to provide SAML SSO URL Here. You should now have a working ADFS SSO implementation for your TimeLive.

13.5. LDAP Integration with TimeLive

What is LDAP how to use it:

LDAP directory service is based on a client-server model. One or more LDAP servers contain the data making up the LDAP directory tree or LDAP backend database. An LDAP client connects to an LDAP server and asks it a question. The server responds with the answer, or with a pointer to where the client can get more information (typically, another LDAP server).

No matter what LDAP server a client connects to, it sees the same view of the directory; a name presented to one LDAP server references the same entry it would at another LDAP server. This is an important feature of a global directory service, like LDAP.

Configuring LDAP Authentication

You can configure TimeLive to authenticate the requester login with the LDAP. This provides you an advantage of not having to remember too many passwords. For this, you should configure LDAP authentication, then any password change that is made in the LDAP will also reflect in TimeLive. So the requesters can login using the login name and password of the system.

LDAP Host must be specified exactly the same as **Server Name** or **IP** used in **LDAP** configuration (As shown in above picture) as well as fill **LDAP Port** with same port used in LDAP configuration and **LDAP Base DN** should match with **Base DN** of LDAP configuration. After giving same LDAP username and password, Click update. LDAP Integration is now configured with TimeLive Account. LDAP users can also now login with there already assigned usernames and passwords.

14. Mobile Apps

14.1. Introduction, Download and Settings

Introduction to TimeLive Mobile App.

With the advancement of technology and increase in the use of mobile phones, everyone is looking for ways they can bring the business in their hands. Many of the tasks were performed but building the timesheet for the projects was very difficult. If one is unable to complete the job in the official hours then the maintenance of the extra hours worked outside the office hours was not possible. TimeLive facilitates Timesheet mobile application to overcome this problem and to give the freedom of managing timesheet wherever an employee is.

Time tracking using TimeLive Mobile App.

Timesheet gives the opportunity to track the time against the projects and Tasks. The updates to the records are circulated in the complete circle of the team involved in the project and notification through the e-mail is also sent. Before TimeLive, it was very difficult to coordinate with other members of the project and a rapid change was witnessed with the introduction of the application.

Now the timesheet can be updated and managed anywhere by using the mobile application.

Download TimeLive Mobile Time Tracking For Android & iPhone

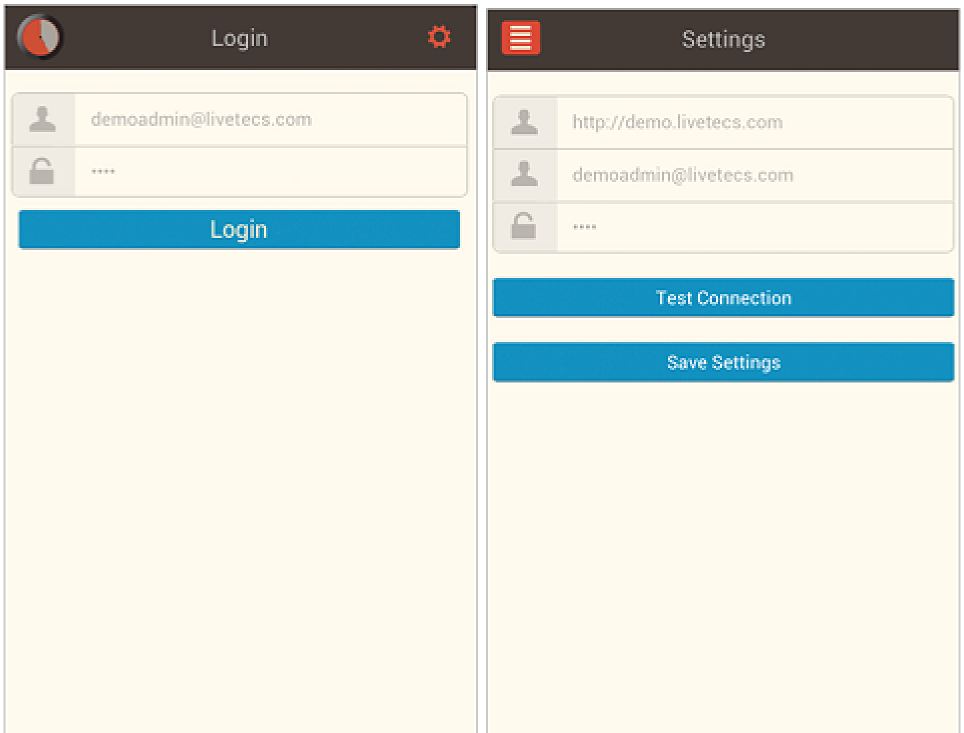
Capture your time against your clients, projects, and tasks while on the go.

- Empower employees to work seamlessly across all devices – desktop, tablet, and smartphone.
- Allows your field staff to enter time against jobs that have been assigned to them.
- Timesheets are updated automatically in TimeLive web and assigned to their associated jobs.
- Timesheet comments as easy as in the web browser.
- Easily enter time against the selected projects, tasks and activities.
- Enter data just once and it reflects on all devices and web timesheet.

The application requires downloading (Android / IOS), once done it is connected to the account, and any changing in the entries and records are automatically updated on the account. Then from whichever device the employee login, the updated information.

Setup:

After the installation of TimeLive App on your mobile, please launch Mobile App and go to **Settings (ICON)** before attempting to log in.

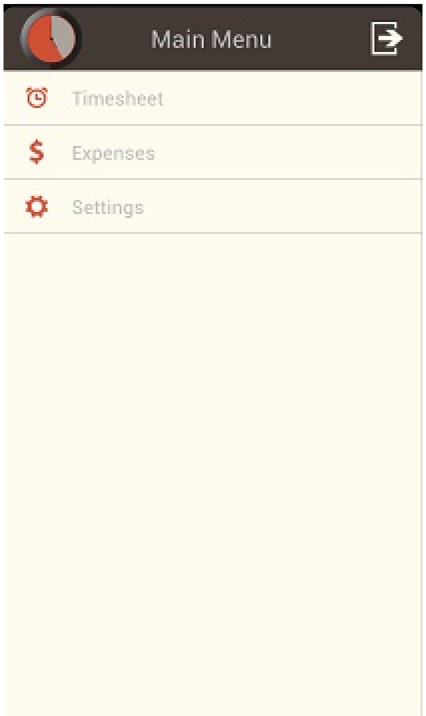


Under **Settings**, Change URL to **https://timelive.livetecs.com**, and change **Email ID**, masked characters with your **Login Credentials**.
You can **Test Connection** for credentials confirmation. **Save Settings** after Test confirmation.

14.2. Android App Entries

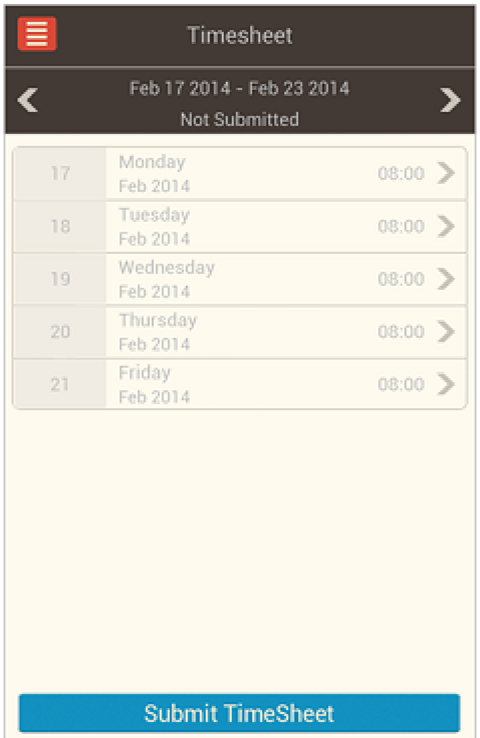
TimeLive Android App Entries:

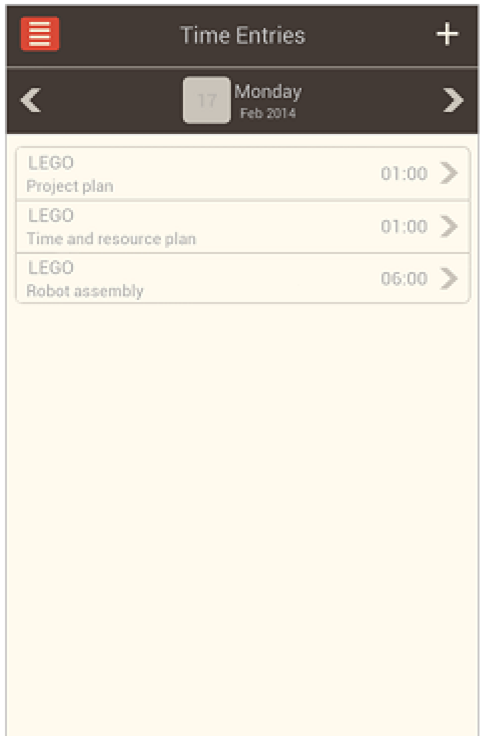
After the successful login to TimeLive App, user will be prompt with 3 options, as shown below.

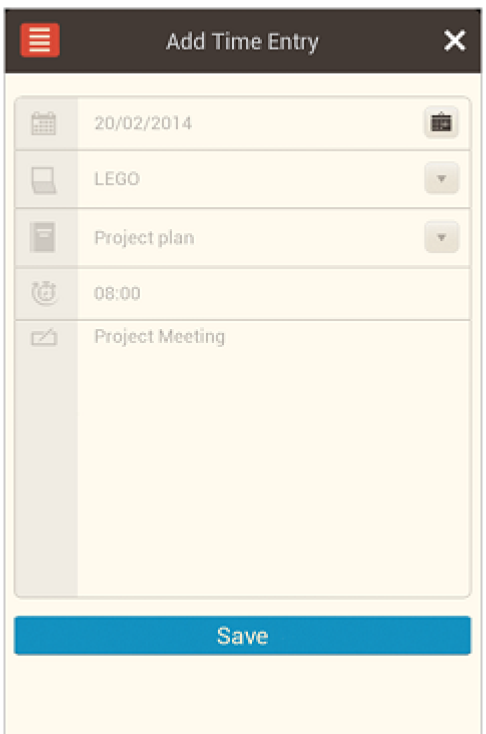


Timesheet Entries:

User can easily make time entries and submit Timesheets from mobile.







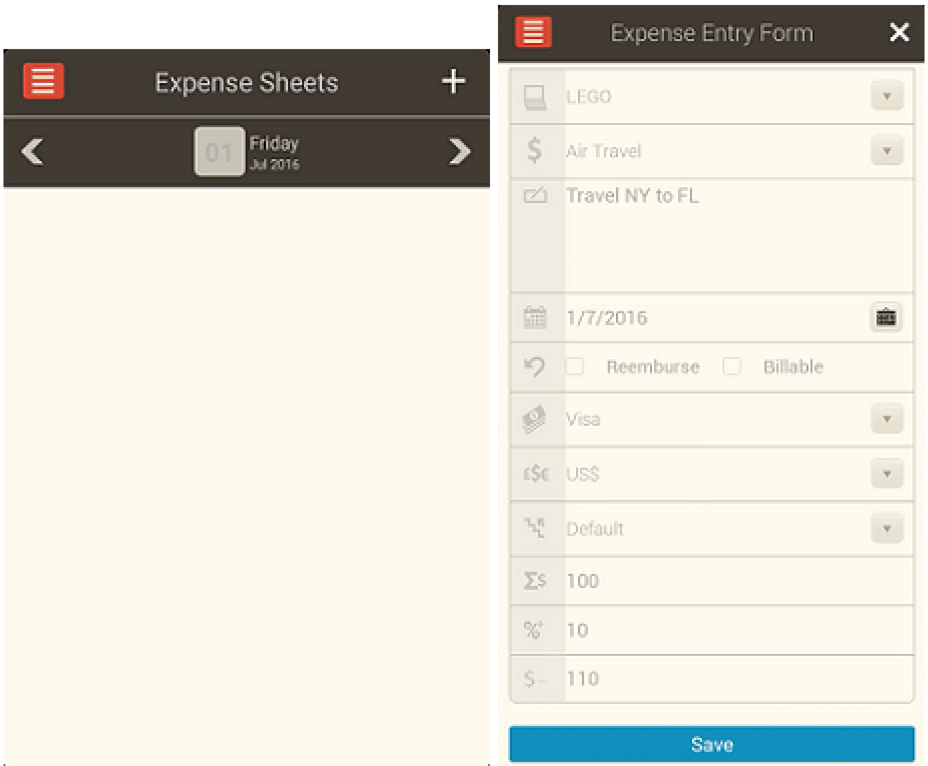
To make new TimeEntry, user need to click **Timesheet** from **Menu** and click on “**+**” Sign.

- Select Date on first row
- Select Project from 2nd row Drop-Down

- Select Task From 3rd row Drop-Down
- Select Time from analog screen clock
- After filling the last description field, click “Save” button

Expense Entries:

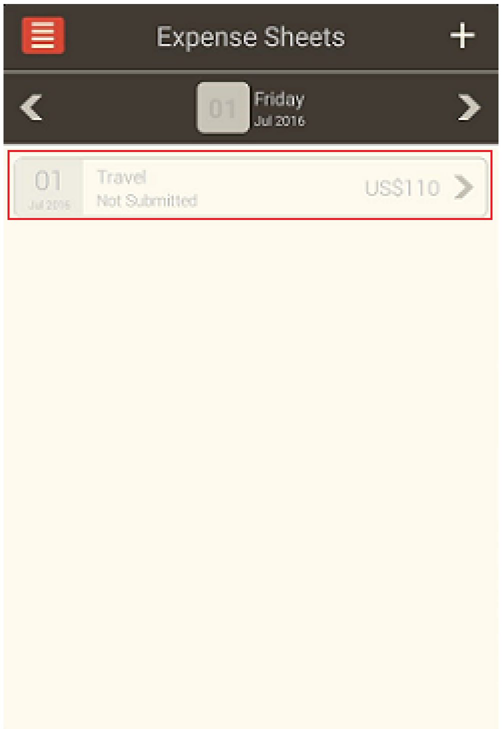
User can easily input & submit Expense entry from their mobiles.

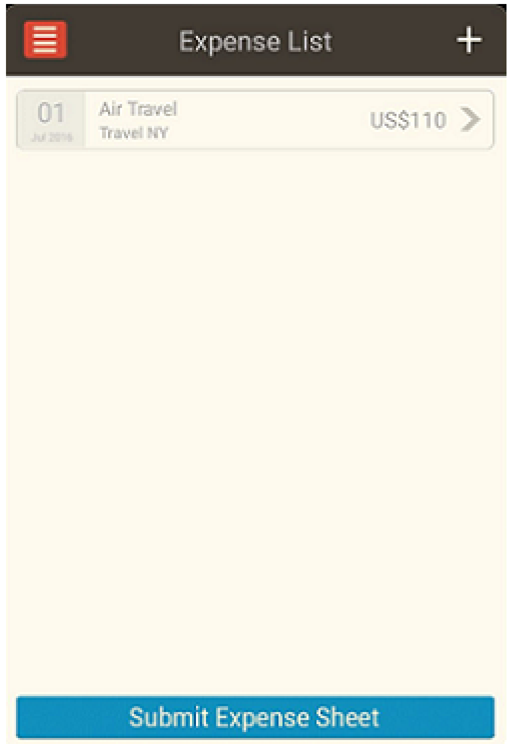


To make new Expense Entry, user need to click **Expenses** from **Menu** and click on “**+**”.

- Select Project from first Drop-Down
- Select Expense Type from 2nd Drop-Down
- Fill the Expense Description
- Select the Expense Date
- Select the option of Reimburse and Billable
- Select Mode of Payment from Drop-Down
- Select Currency in which payment was made
- Input Amount spent in the next row
- Extra charges/taxes will be calculated automatically according to expense se parameters
- After filling the last field, click “Save” Button

To **submit** saved Expense Entry, user need to **click** on **Expense sheet** and click **Submit Expense Sheet**.

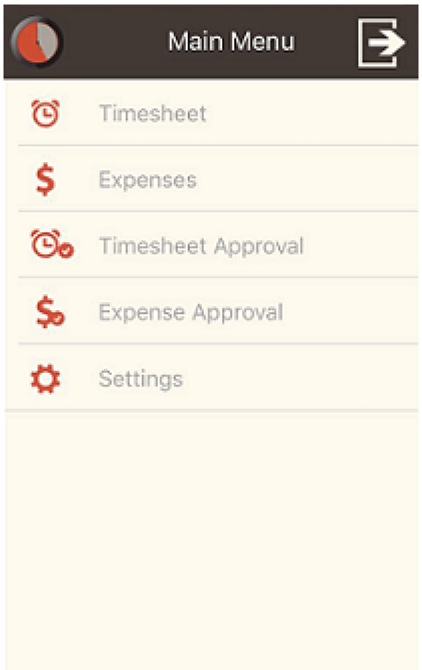




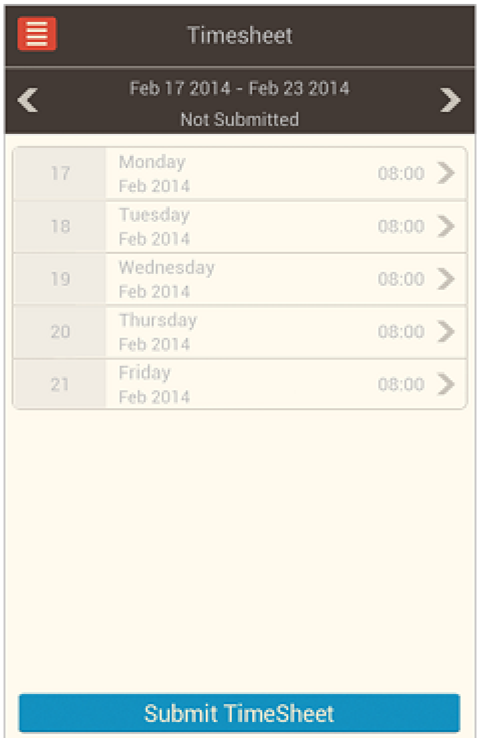
14.3. IOS App Entries and Approvals

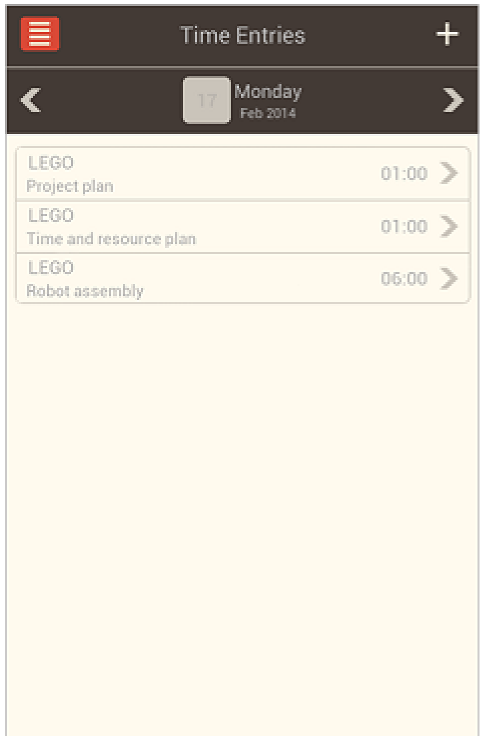
Timesheet Entries:

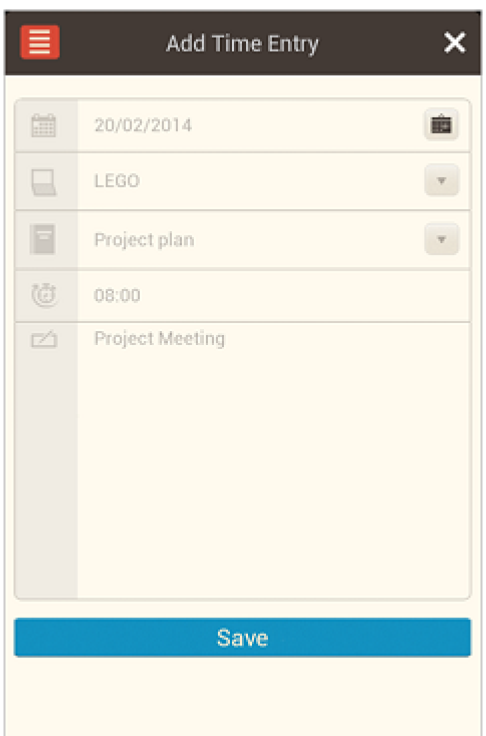
After the successful login to TimeLive App, user will be prompt with 5 options, as shown below.



User can easily input/submit Timesheet & Expense entry from their mobiles.







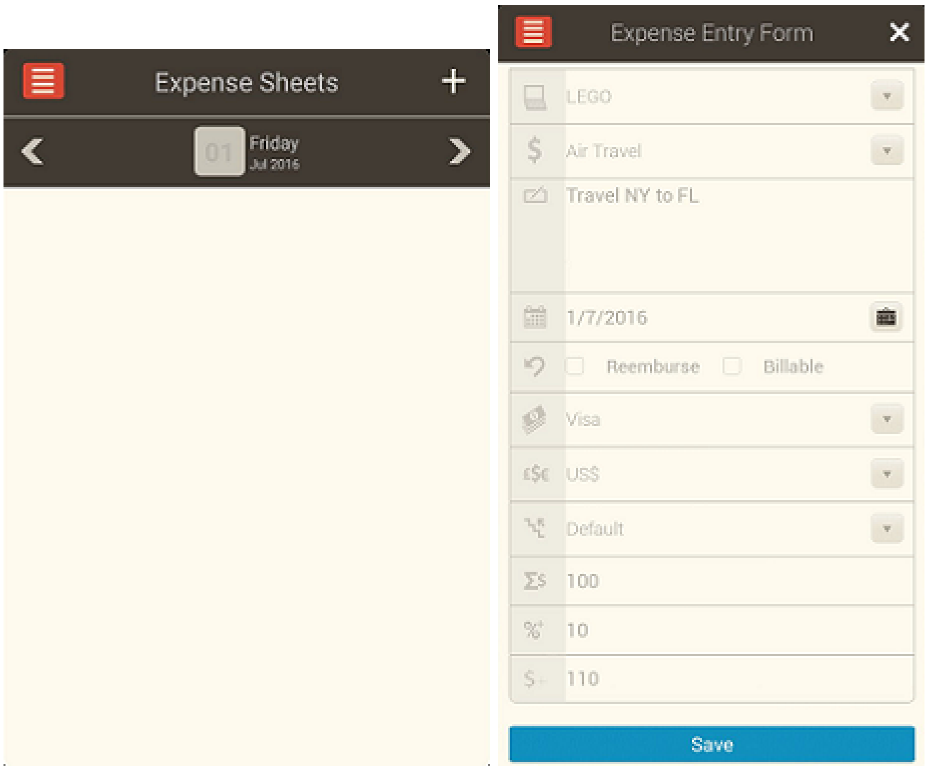
To make new TimeEntry user need to click on "**+**" Sign under Timesheet.

- Select Date on first row
- Select Project from 2nd row Drop-Down

- Select Task From 3rd row Drop-Down
- Select Time from analog screen clock
- After filling the last description field, click “Save” button

Expense Entries:

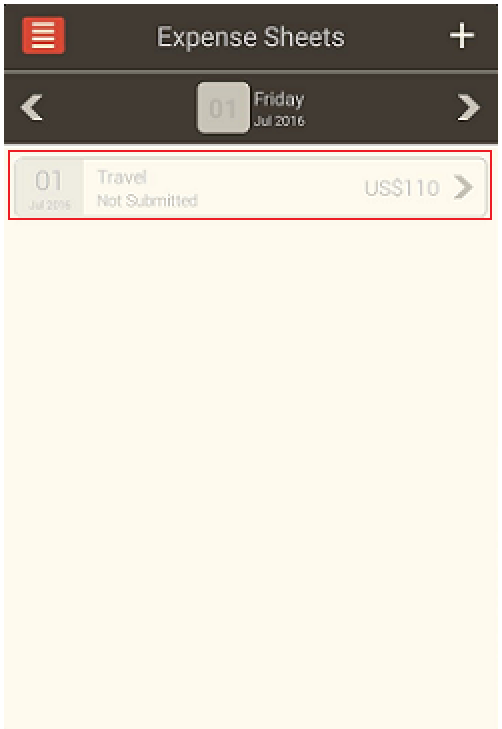
User can easily input & submit Expense entry from their mobiles.

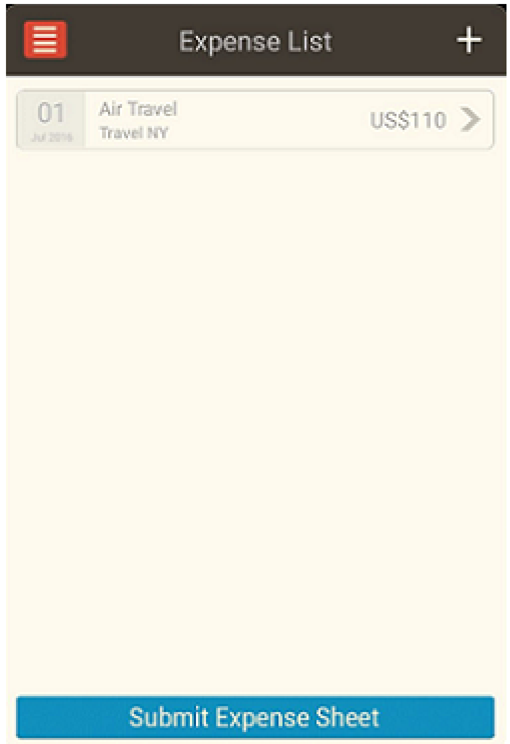


To make new Expense Entry, user need to click **Expenses** from **Menu** and click on “**+**”.

- Select Project from first Drop-Down
- Select Expense Type from 2nd Drop-Down
- Fill the Expense Description
- Select the Expense Date
- Select the option of Reimburse and Billable
- Select Mode of Payment from Drop-Down
- Select Currency in which payment was made
- Input Amount spent in the next row
- Extra charges/taxes will be calculated automatically according to expense parameters
- After filling the last field, click “Save” Button

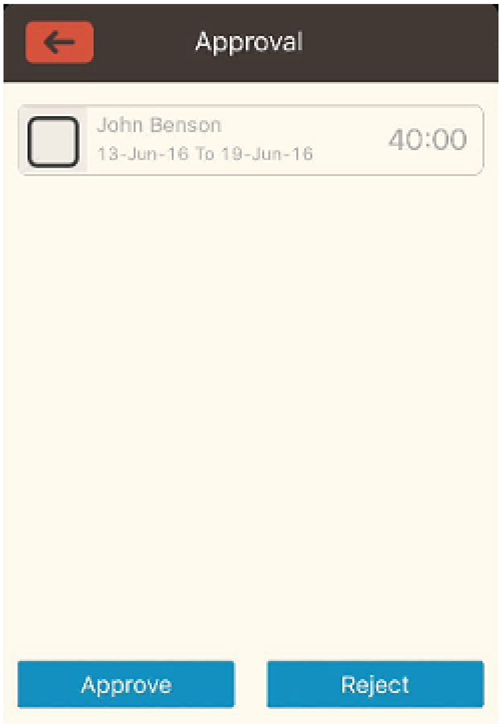
To **submit** saved Expense Entry, user need to **click** on **Expense sheet** and click **Submit Expense Sheet**.

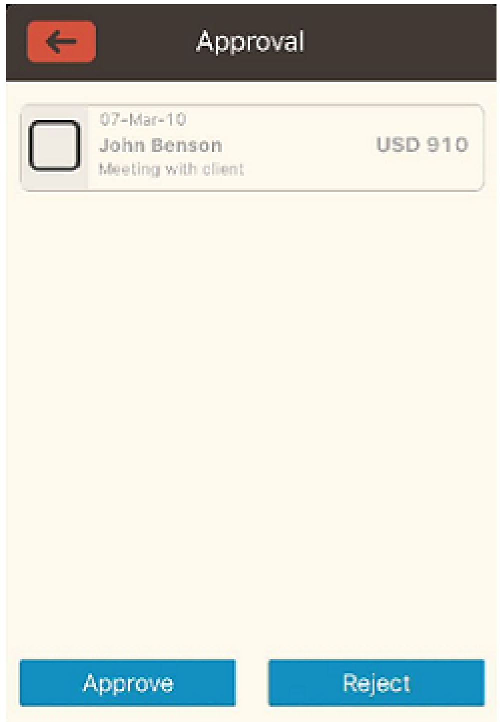




Timesheet & Expense Approval:

User can Approve / Reject Timesheet / Expense of his subordinates from mobile.
To do that user needs to click Timesheet / Expense Approval from Main Menu, and click on Approve / Reject after check box selection.
Approver can select multiple Timesheets / Expense at a time for Approval / Rejection.


Approval
John Benson
13-Jun-16 To 19-Jun-16
40:00
Approve
Reject


Approval
07-Mar-10
John Benson
USD 910
Meeting with client
Approve
Reject

15. TimeLive API

15.1. Overview

The TimeLive API allows you to interact with your TimeLive account from any third party application or programmatically. You can track time, log expenses, create employees, create clients, create projects, create tasks, and more.

15.2. Authentication

The TimeLive API requires an **API Key** and **Authentication Token** to authenticate requests.

API Key

Please follow given steps to get an **API Key**

1. Login to your **Administrator** TimeLive account.
2. Click **Admin Options** at the top right.
3. Click **API** icon under **Integration** section.
4. In the **API Key** screen that appears:
    a. Click **Enable API** button, then copy generated **API Key** and click **Update** button.

Authentication Token

Please follow given steps to get an **Authentication Token**

1. Login to your TimeLive account.
2. Click **Profile** Image or Initials at the top right, then click on **My Profile**.
3. In the **Profile** screen that appears:
    a. Navigate to **API** tab.
    b. Click **Enable Token** button, then copy generated **Authentication Token** and click **Update** button.

Authentication Example

Please follow given **TimeLive API** sample code document for **Authentication**.

TimeLive API Sample Code – C#

15.3. API

15.3.1. Departments

List All Departments

Returns a list of your all **Departments** in your **TimeLive** account.

```
GET /API/Departments
```

Example Request

```
using System;
namespace TimeLiveAPISample
{
    class Program
    {
        static void Main(string[] args)
        {
            const string URL = "http://demo.livetecs.com/api/departments";
            try
            {
                var webRequest = System.Net.WebRequest.CreateHttp(URL);
                if (webRequest != null)
                {
                    webRequest.Method = "GET";
                    webRequest.UserAgent = "C# TimeLive API Sample";
                    webRequest.Headers.Add("AuthToken",
<TIMELIVE_AUTHENTICATION_TOKEN>);
                    webRequest.Headers.Add("APIKey", <TIMELIVE_API_KEY>);
                    using (System.IO.Stream s =
webRequest.GetResponse().GetResponseStream())
                    {
                        using (System.IO.StreamReader sr = new
System.IO.StreamReader(s))
                        {
                            var jsonResponse = sr.ReadToEnd();
                            Console.WriteLine(jsonResponse);
                        }
                    }
                }
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.ToString());
            }
        }
```

```
    }
}
```

Example Response


```
[
  {
    "AccountDepartmentId": 125,
    "AccountId": 151,
    "DepartmentName": "Administration",
    "IsDisabled": false,
    "IsDeleted": false,
    "CreatedOn": "2007-02-03T16:39:29.083+00:00",
    "ModifiedOn": "2007-02-04T11:00:57.467+00:00",
    "CreatedByEmployeeId": 0,
    "ModifiedByEmployeeId": 129,
    "DepartmentCode": "ADMIN "
  },
  {
    "AccountDepartmentId": 127,
    "AccountId": 151,
    "DepartmentName": "Development ",
    "IsDisabled": false,
    "IsDeleted": false,
    "CreatedOn": "2007-02-03T18:19:44.273+00:00",
    "ModifiedOn": "2007-02-03T18:19:44.273+00:00",
    "CreatedByEmployeeId": 129,
    "ModifiedByEmployeeId": 129,
    "DepartmentCode": "DEV"
  },
  {
    "AccountDepartmentId": 128,
    "AccountId": 151,
    "DepartmentName": "Quality Assurance",
    "IsDisabled": false,
    "IsDeleted": false,
    "CreatedOn": "2007-02-03T18:19:53.637+00:00",
    "ModifiedOn": "2007-02-03T18:19:53.637+00:00",
    "CreatedByEmployeeId": 129,
    "ModifiedByEmployeeId": 129,
    "DepartmentCode": "QA"
  }
]
```

15.3.2. Roles

List All Roles

Returns a list of your all **Roles** in your **TimeLive** account.

```
GET /API/Employees/GetActiveRoles
```

Example Request

```
using System;
namespace TimeLiveAPISample
{
    class Program
    {
        static void Main(string[] args)
        {
            const string URL = "http://demo.livetecs.com/api/employees/
getactiveroles";
            try
            {
                var webRequest = System.Net.WebRequest.CreateHttp(URL);
                if (webRequest != null)
                {
                    webRequest.Method = "GET";
                    webRequest.UserAgent = "C# TimeLive API Sample";
                    webRequest.Headers.Add("AuthToken",
<TIMELIVE_AUTHENTICATION_TOKEN>);
                    webRequest.Headers.Add("APIKey", <TIMELIVE_API_KEY>);
                    using (System.IO.Stream s =
webRequest.GetResponse().GetResponseStream())
                    {
                        using (System.IO.StreamReader sr = new
System.IO.StreamReader(s))
                        {
                            var jsonResponse = sr.ReadToEnd();
                            Console.WriteLine(jsonResponse);
                        }
                    }
                }
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.ToString());
            }
```

```
        }
    }
}
```

Example Response


```
[
  {
    "AccountRoleId": 207,
    "AccountId": 151,
    "Role": "Administrator",
    "IsDisabled": false,
    "IsDeleted": false,
    "CreatedOn": "2007-02-03T16:39:42.45+00:00",
    "CreatedByEmployeeId": 0,
    "ModifiedOn": "2007-02-03T16:39:42.45+00:00",
    "ModifiedByEmployeeId": 0,
    "MasterAccountRoleId": 1,
    "IsSystemRole": null
  },
  {
    "AccountRoleId": 208,
    "AccountId": 151,
    "Role": "User",
    "IsDisabled": false,
    "IsDeleted": false,
    "CreatedOn": "2007-02-03T16:39:42.45+00:00",
    "CreatedByEmployeeId": 0,
    "ModifiedOn": "2007-02-03T16:39:42.45+00:00",
    "ModifiedByEmployeeId": 0,
    "MasterAccountRoleId": 2,
    "IsSystemRole": null
  }
]
```

15.3.3. Employees

The Employee Object

Object Property	Required	Object Type	Description	Max Character Length
FirstName	**Yes**	**String**	**First Name Of Employee**	**150**
LastName	**Yes**	**String**	**Last Name Of Employee**	**150**
EMailAddress	**Yes**	**String**	**Email Address Of Employee**	**100**
AccountEmployeeId	No	Integer	Unique Id Of Employee	10
MiddleName	No	String	Middle name Of Employee	40
AccountDepartmentId	No	Integer	Department Id Of Employee	10
AccountRoleId	No	Integer	Role Id Of Employee	10
AddressLine1	No	String	Address Line 1 Of Employee	300
AddressLine2	No	String	Address Line 2 Of Employee	300
City	No	String	City Of Employee	100
Zip	No	String	Zip Of Employee	50
CountryId	No	Integer	CountryId Of Employee	3
HomePhoneNo	No	String	Home Phone No Of Employee	40
WorkPhoneNo	No	String	Work Phone No Of Employee	100
MobilePhoneNo	No	String	Mobile Phone No Of Employee	100
TerminationDate	No	DateTime	Termination Date Of Employee	19
IsDisabled	No	Boolean	Disabled Status Of Employee	1
TimeZoneId	No	Integer	TimeZone Id Of Employee	3
Prefix	No	String	Prefix Of Employee	50
AccountLocationId	No	Integer	Location Id Of Employee	10
Notes	No	String	Notes Of Employee	1000
BillingTypeId	No	Integer	Billing Type Id Of Employee	10
State	No	String	State Of Employee	40
AllowedAccessFromIP	No	String	Allowed Access From IP Of Employee	40

AccountBillingRateId	No	Integer	Billing Rate Id Of Employee	10
EmployeeCode	No	String	Employee Code Of Employee	100
EmployeeManagerId	No	Integer	Employee Manager Id Of Employee	10
JobTitle	No	String	Job Title Of Employee	200
EmployeePayTypeId	No	GUID	Employee Pay Type Id Of Employee	36
HiredDate	No	DateTime	Hired Date Of Employee	19
StatusId	No	Integer	Status Id Of Employee	10
AccountWorkingDayTypeId	No	GUID	Working Day Type Id Of Employee	36
AccountTimeOffPolicyId	No	GUID	TimeOff Policy Id Of Employee	36
TimeOffApprovalTypeId	No	Integer	TimeOff Approval Type Id Of Employee	10
AccountHolidayTypeId	No	GUID	Holiday Type Id Of Employee	36
IsForcePasswordChange	No	Boolean	Force Password Change Status Of Employee	1
CustomField1	No	String	CustomField 1 Of Employee	2000
CustomField2	No	String	CustomField 2 Of Employee	2000
CustomField3	No	String	CustomField 3 Of Employee	2000
CustomField4	No	String	CustomField 4 Of Employee	2000
CustomField5	No	String	CustomField 5 Of Employee	2000
CustomField6	No	String	CustomField 6 Of Employee	2000
CustomField7	No	String	CustomField 7 Of Employee	2000
CustomField8	No	String	CustomField 8 Of Employee	2000
CustomField9	No	String	CustomField 9 Of Employee	2000
CustomField10	No	String	CustomField 10 Of Employee	2000
CustomField11	No	String	CustomField 11 Of Employee	2000
CustomField12	No	String	CustomField 12 Of Employee	2000
CustomField13	No	String	CustomField 13 Of Employee	2000
CustomField14	No	String	CustomField 14 Of Employee	2000
CustomField15	No	String	CustomField 15 Of Employee	2000
UserInterfaceLanguage	No	String	User Interface Language Of Employee	50

IsShowEmployeeProfilePicture	No	Boolean	Show Employee Profile Picture Status Of Employee	1
IsTimeInOutAvailable	No	Boolean	Time In Time Out Available Of Employee	1
BillingRate	No	Decimal	Billing Rate Of Employee	10
EmployeeRate	No	Decimal	Employee Rate Of Employee	10
StartDate	No	DateTime	Start Date of Billing Of Employee	19
EndDate	No	DateTime	End Date of Billing Of Employee	19
AccountWorkTypeId	No	Integer	Work Type Id Of Employee	10
BillingRateCurrencyId	No	Integer	Currency Id Of Billing Rate Of Employee	10
EmployeeRateCurrencyId	No	Integer	Currency Id Of Employee Rate Of Employee	10

List All Employees

Returns a list of your all **Employees** in your **TimeLive** account.

```
GET /API/Employees
```

Example Request

```
curl "https://demo.livetecs.com/api/employees" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
  {
    "AccountEmployeeId": 130,
    "EmployeeCode": "001",
    "FirstName": "Bob",
    "LastName": "Bingham",
    "UserName": "bobbingham@livetecs.com",
    "EMailAddress": "bobbingham@livetecs.com",
    "DepartmentName": "Development",
    "JobTitle": null,
```

```
    "Role": "User",
    "AccountLocation": "US(HeadOffice)",
    "BillingType": null,
    "HiredDate": "2018-04-03T00:00:00+00:00",
    "CreatedOn": "2018-04-03T00:00:00+00:00",
    "ModifiedOn": "2018-04-03T00:00:00+00:00",
    "AccountEmployeeType": "accountant",
    "AccountTimeOffPolicyId": null,
    "TimeOffApprovalTypeId": null,
    "AllowedAccessFromIP": null,
    "AccountWorkingDayTypeId": "55dedd23-7d27-4300-be08-2982079da47d",
    "UserInterfaceLanguage": "en-US",
    "Department": "DEV-Development",
    "MobilePhoneNo": null,
    "WorkPhoneNo": "",
    "HomePhoneNo": null,
    "Zip": null,
    "City": null,
    "State": null,
    "AddressLine1": null,
    "AddressLine2": null,
    "TerminationDate": null,
    "CustomField1": null,
    "CustomField2": null,
    "CustomField3": null,
    "CustomField4": null,
    "CustomField5": null,
    "CustomField6": null,
    "CustomField7": null,
    "CustomField8": null,
    "CustomField9": null,
    "CustomField10": null,
    "CustomField11": null,
    "CustomField12": null,
    "CustomField13": null,
    "CustomField14": null,
    "CustomField15": null,
    "IsDisabled": "No",
    "AccountLocationId": 12948,
    "AccountRoleId": 83569,
    "AccountDepartmentId": 15706,
    "EmployeeCountryId": 170,
    "StatusId": 149541,
    "EmployeePayTypeId": "00aa8186-c0f4-4b61-ba25-ddf857c04ffd",
    "EmployeeTimeZoneId": 21,
    "EmployeeManagerId": null,
    "AccountHolidayTypeId": null,
    "IsForcePasswordChange": "No",
    "IsShowEmployeeProfilePicture": "No"
```

```
  }
]
```

Retrieve an Employee

Retrieves the Employee with the given ID. Returns a Employee object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Employees/{EMPLOYEE_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Employees/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
  {
    "AccountEmployeeId": 130,
    "EmployeeCode": "001",
    "FirstName": "Bob",
    "LastName": "Bingham",
    "UserName": "bobbingham@livetecs.com",
    "EMailAddress": "bobbingham@livetecs.com",
    "DepartmentName": "Development",
    "JobTitle": null,
    "Role": "User",
    "AccountLocation": "US(HeadOffice)",
    "BillingType": null,
    "HiredDate": "2018-04-03T00:00:00+00:00",
    "CreatedOn": "2018-04-03T00:00:00+00:00",
    "ModifiedOn": "2018-04-03T00:00:00+00:00",
    "AccountEmployeeType": "accountant",
    "AccountTimeOffPolicyId": null,
    "TimeOffApprovalTypeId": null,
    "AllowedAccessFromIP": null,
    "AccountWorkingDayTypeId": "55dedd23-7d27-4300-be08-2982079da47d",
    "UserInterfaceLanguage": "en-US",
    "Department": "DEV-Development",
    "MobilePhoneNo": null,
    "WorkPhoneNo": "",
```

```
    "HomePhoneNo": null,
    "Zip": null,
    "City": null,
    "State": null,
    "AddressLine1": null,
    "AddressLine2": null,
    "TerminationDate": null,
    "CustomField1": null,
    "CustomField2": null,
    "CustomField3": null,
    "CustomField4": null,
    "CustomField5": null,
    "CustomField6": null,
    "CustomField7": null,
    "CustomField8": null,
    "CustomField9": null,
    "CustomField10": null,
    "CustomField11": null,
    "CustomField12": null,
    "CustomField13": null,
    "CustomField14": null,
    "CustomField15": null,
    "IsDisabled": "No",
    "AccountLocationId": 12948,
    "AccountRoleId": 83569,
    "AccountDepartmentId": 15706,
    "EmployeeCountryId": 170,
    "StatusId": 149541,
    "EmployeePayTypeId": "00aa8186-c0f4-4b61-ba25-ddf857c04ffd",
    "EmployeeTimeZoneId": 21,
    "EmployeeManagerId": null,
    "AccountHolidayTypeId": null,
    "IsForcePasswordChange": "No",
    "IsShowEmployeeProfilePicture": "No"
  }
]
```


Retrieve an Employee by Employee Code

Retrieves the Employee with the given Code. Returns a Employee object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Employees/GetEmployeeByEmployeeCode/{Employee_Code}
```

Example Request

```
curl "https://demo.livetecs.com/api/Employees/GetEmployeeByEmployeeCode/001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
  {
    "AccountEmployeeId": 130,
    "EmployeeCode": "001",
    "FirstName": "Bob",
    "LastName": "Bingham",
    "UserName": "bobbingham@livetecs.com",
    "EMailAddress": "bobbingham@livetecs.com",
    "DepartmentName": "Development",
    "JobTitle": null,
    "Role": "User",
    "AccountLocation": "US(HeadOffice)",
    "BillingType": null,
    "HiredDate": "2018-04-03T00:00:00+00:00",
    "CreatedOn": "2018-04-03T00:00:00+00:00",
    "ModifiedOn": "2018-04-03T00:00:00+00:00",
    "AccountEmployeeType": "accountant",
    "AccountTimeOffPolicyId": null,
    "TimeOffApprovalTypeId": null,
    "AllowedAccessFromIP": null,
    "AccountWorkingDayTypeId": "55dedd23-7d27-4300-be08-2982079da47d",
    "UserInterfaceLanguage": "en-US",
    "Department": "DEV-Development",
    "MobilePhoneNo": null,
    "WorkPhoneNo": "",
    "HomePhoneNo": null,
    "Zip": null,
    "City": null,
    "State": null,
    "AddressLine1": null,
    "AddressLine2": null,
    "TerminationDate": null,
    "CustomField1": null,
    "CustomField2": null,
    "CustomField3": null,
    "CustomField4": null,
    "CustomField5": null,
    "CustomField6": null,
    "CustomField7": null,
```

```
    "CustomField8": null,
    "CustomField9": null,
    "CustomField10": null,
    "CustomField11": null,
    "CustomField12": null,
    "CustomField13": null,
    "CustomField14": null,
    "CustomField15": null,
    "IsDisabled": "No",
    "AccountLocationId": 12948,
    "AccountRoleId": 83569,
    "AccountDepartmentId": 15706,
    "EmployeeCountryId": 170,
    "StatusId": 149541,
    "EmployeePayTypeId": "00aa8186-c0f4-4b61-ba25-ddf857c04ffd",
    "EmployeeTimeZoneId": 21,
    "EmployeeManagerId": null,
    "AccountHolidayTypeId": null,
    "IsForcePasswordChange": "No",
    "IsShowEmployeeProfilePicture": "No"
  }
]
```


Create an Employee

Creates an **Employee** in your **TimeLive** account and returns the Response Message.

```
POST /API/Employees
```

Object Property	Required	Object Type	Description	Max Character Length
FirstName	**Yes**	**String**	**First name of the newly created employee**	**150**
LastName	**Yes**	**String**	**Last name of the newly created employee**	**150**
EmailAddress	**Yes**	**String**	**Email of the newly created employee**	**100**

Example Request

```
curl "https://demo.livetecs.com/api/employees" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d
"{'EmailAddress':'someone@example.com','FirstName':'john','LastName':'Doe'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "Employee Added Successfully"
    }
]
```

Update an Employee

Update an **Employee** in your **TimeLive** account and returns the Response Message.

```
POST /API/Employees/{EMPLOYEE_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/employees/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'LastName':'Smith'}"
```

Example Response

```
[
    {
        "id": "00003",
        "message": "Employee Updated Successfully"
    }
]
```

Delete an Employee

Delete an **Employee**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Employee/{EMPLOYEE_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/employees/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00003",
        "message": "Employee Deleted Successfully"
    }
]
```

15.3.4. Clients

The Task Object

Object Property	Required	Object Type	Description	Max Character Length
ClientName	**Yes**	**String**	**Name Of Client**	**200**
AccountclientId	No	Integer	Unique Client Id Of Client	10
ClientCode	No	String	Client Code Of Client	4000
EMailAddress	No	Integer	Email Address Of Client	10
Address1	No	Decimal	Address 1 Of Client	10
Address2	No	String	Address 2 Of Client	20
CountryId	No	Date Time	Country Id Of Client	19
State	No	Float	State Of Client	10
City	No	Boolean	City Of Client	1
ZipCode	No	Boolean	ZipCode Of Client	1
Telephone1	No	Boolean	Telephone 1 Of Client	1
Telephone2	No	Integer	Telephone 2 Of Client	10
Fax	No	Integer	Fax No Of Client	10
DefaultBillingRate	No	Integer	Default Billing Rate Of Client	10
Website	No	Boolean	Website Of Client	1
Notes	No	Float	Notes Of Client	10
IsDisabled	No	Float	Disabled Status Of Client	10
CustomField1	No	Boolean	CustomField 1 Of Client	1
CustomField2	No	Boolean	CustomField 2 Of Client	1
CustomField3	No	Integer	CustomField 3 Of Client	10
CustomField4	No	Boolean	CustomField 4 Of Client	1
CustomField5	No	String	CustomField 5 Of Client	30
CustomField6	No	Integer	CustomField 6 Of Client	10
CustomField7	No	Integer	CustomField 7 Of Client	10
CustomField8	No	Date Time	CustomField 8 Of Client	19

CustomField9	No	String	CustomField 9 Of Client	50
CustomField11	No	String	CustomField 10 Of Client	2000
CustomField10	No	String	CustomField 11 Of Client	2000
CustomField12	No	String	CustomField 12 Of Client	2000
CustomField13	No	String	CustomField 13 Of Client	2000
CustomField14	No	String	CustomField 14 Of Client	2000
CustomField15	No	String	CustomField 15 Of Client	2000
FixedCost	No	Float	Fixed Cost Of Client	10

List All Clients

Returns a list of your all **Clients** in your **TimeLive** account.

```
GET /API/Clients
```

Example Request

```
curl "https://demo.livetecs.com/api/clients" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountClientId": 00001,
        "ClientName": "Barth Gross Electric Company Inc",
        "ClientCode": "BGE",
        "EMailAddress": "info@barthgross.com",
        "Address1": "110 West 26th Street",
        "Address2": null,
        "State": "NY",
        "City": "New York",
        "Country": "United States"
        "ZipCode": 10001,
        "Telephone1": "+1 (212) 645-7452",
        "Telephone2": null,
        "Fax": null,
        "Website": "http://www.example.com",
```

```
        "Notes": null,
        "CreatedOn": "2017-12-27T13:57:34.78+00:00",
        "ModifiedOn": "2017-12-27T13:57:34.78+00:00",
        "CustomField1": null,
        "CustomField2": null,
        "CustomField3": null,
        "CustomField4": null,
        "CustomField5": null,
        "CustomField6": null,
        "CustomField7": null,
        "CustomField8": null,
        "CustomField9": null,
        "CustomField11": null,
        "CustomField10": null,
        "CustomField12": null,
        "CustomField13": null,
        "CustomField14": null,
        "CustomField15": null
    },
    {
        "AccountClientId": 00002,
        "ClientName": "Clutch Design Inc.",
        "ClientCode": "CLUTCH",
        "EMailAddress": "info@clutchdesign.com",
        "Address1": "320 Wilshire Blvd",
        "Address2": null,
        "State": "CA",
        "City": "Santa Monica",
        "Country": "United States"
        "ZipCode": 90401,
        "Telephone1": "+1 (310) 899-0522",
        "Telephone2": null,
        "Fax": null,
        "Website": "http://www.example.com",
        "Notes": null,
        "CreatedOn": "2017-12-27T13:57:34.78+00:00",
        "ModifiedOn": "2017-12-27T13:57:34.78+00:00",
        "CustomField1": null,
        "CustomField2": null,
        "CustomField3": null,
        "CustomField4": null,
        "CustomField5": null,
        "CustomField6": null,
        "CustomField7": null,
        "CustomField8": null,
        "CustomField9": null,
        "CustomField11": null,
        "CustomField10": null,
        "CustomField12": null,
        "CustomField13": null,
```

```
        "CustomField14": null,
        "CustomField15": null
    }
]
```

Retrieve a Client

Retrieves the client with the given ID. Returns a client object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Clients/{CLIENT_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/clients/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountClientId": 00001,
        "ClientName": "Barth Gross Electric Company Inc",
        "ClientCode": "BGE",
        "EMailAddress": "info@barthgross.com",
        "Address1": "110 West 26th Street",
        "Address2": null,
        "State": "NY",
        "City": "New York",
        "Country": "United States"
        "ZipCode": 10001,
        "Telephone1": "+1 (212) 645-7452",
        "Telephone2": null,
        "Fax": null,
        "Website": "http://www.example.com",
        "Notes": null,
        "CreatedOn": "2017-12-27T13:57:34.78+00:00",
        "ModifiedOn": "2017-12-27T13:57:34.78+00:00",
        "CustomField1": null,
        "CustomField2": null,
        "CustomField3": null,
        "CustomField4": null,
```

```
        "CustomField5": null,
        "CustomField6": null,
        "CustomField7": null,
        "CustomField8": null,
        "CustomField9": null,
        "CustomField11": null,
        "CustomField10": null,
        "CustomField12": null,
        "CustomField13": null,
        "CustomField14": null,
        "CustomField15": null
    }
]
```

Create a Client

Creates a new client. Returns Response message and a 201 Created response code if the call succeeded.

```
POST /API/Clients
```

Object Property	Required	Object Type	Description	Max Character Length
TaskName	Yes	String	Task Name Of Task	200
AccountProjectId	Yes	Integer	Project Id Of Task	10

Example Request

```
curl "https://demo.livetecs.com/api/clients" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'clientname':'LEGO'}"
```

Example Response

```
[
    {
        "id": "00003",
        "message": "Client Added Successfully"
    }
]
```

Update a Client

Updates the specific client by setting the values of the parameters passed. Any parameters not provided will be left unchanged. Returns Response message and a 200 OK response code if the call succeeded.

```
POST /API/Clients/{CLIENT_id}
```

Example Request

```
curl "https://demo.livetecs.com/api/clients/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'clientcode':'LEGO'}"
```

Example Response

```
[
    {
        "id": "00003",
        "message": "Client Updated Successfully"
    }
]
```

Delete a Client

Delete a client. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Clients/{CLIENT_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/clients/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

bc.[{ “id”: “00003”, “message”: “Client Deleted Successfully” }
]

15.3.5. Projects

The Project Object

Object Property	Required	Object Type	Description	Max Character Length
ProjectName	**Yes**	**String**	**Name Of Project**	**600**
AccountClientId	**Yes**	**Integer**	**Client Id Of Project**	**10**
AccountProjectId	No	Integer	Id Of Project	10
AccountProjectTypeId	No	Integer	Type Id Of Project	10
ProjectBillingTypeId	No	Integer	Billing Type Id Of Project	10
ProjectDescription	No	String	ProjectDescription Of Project	4000
StartDate	No	Date Time	StartDate Of Project	19
AccountMilestoneId	No	Integer	Milestone Id Of Project	10
Deadline	No	Date Time	Deadline Date Of Project	19
LeaderEmployeeId	No	Integer	Leader Employee Id Of Project	10
ProjectManagerEmployeeId	No	Integer	Project Manager Employee Id Of Project	10
EstimatedTime	No	Date Time	Estimated Time Of Project	19
EstimatedDuration	No	Float	Estimated Duration Of Project	10
EstimatedDurationUnit	No	String	Estimated Duration Unit Of Project	20
ProjectCode	No	String	Code Of Project	100
DefaultBillingRate	No	Decimal	Default Billing Rate Of Project	10
IsActive	No	Boolean	Active Status Of Project	1
ProjectStatusId	No	Integer	Status Id Of Project	10
AccountPriorityId	No	Integer	Priority Id Of Project	10
TimeSheetApprovalTypeId	No	Integer	TimeSheet Approval Type Id Of Project	10
ExpenseApprovalTypeId	No	Integer	Expense Approval Type Id Of Project	10
ProjectBillingRateTypeId	No	Integer	Billing Rate Type Id Of Project	10
AccountPartyContactId	No	Integer	Project Contact Id Of Project	10

AccountPartyDepartmentId	No	Integer	Project Department Id Of Project	10
IsDisabled	No	Boolean	Disabled Status Of Project	1
IsTemplate	No	Boolean	Template Status Of Project	1
IsProject	No	Boolean	Is This a Project	1
AccountProjectTemplateId	No	Integer	Project Template Id Of Project	10
Completed	No	Boolean	Completed Status Of Project	1
CustomField1	No	String	CustomField 1 Of Project	2000
CustomField2	No	String	CustomField 2 Of Project	2000
CustomField3	No	String	CustomField 3 Of Project	2000
CustomField4	No	String	CustomField 4 Of Project	2000
CustomField5	No	String	CustomField 5 Of Project	2000
CustomField6	No	String	CustomField 6 Of Project	2000
CustomField7	No	String	CustomField 7 Of Project	2000
CustomField8	No	String	CustomField 8 Of Project	2000
CustomField9	No	String	CustomField 9 Of Project	2000
CustomField10	No	String	CustomField 10 Of Project	2000
CustomField11	No	String	CustomField 11 Of Project	2000
CustomField12	No	String	CustomField 12 Of Project	2000
CustomField13	No	String	CustomField 13 Of Project	2000
CustomField14	No	String	CustomField 14 Of Project	2000
CustomField15	No	String	CustomField 15 Of Project	2000
ProjectPrefix	No	String	Prefix Of Project	100
ProjectEstimatedCost	No	Float	Estimated Cost Of Project	10
IsForAllProjectProject	No	Boolean	Project is for all Projects	1
FixedCost	No	Float	Fixed Cost Of Project	10

List All Projects

Returns a list of your all **Projects** in your **TimeLive** account.

```
GET /API/Projects
```

Example Request

```
curl "https://demo.livetecs.com/api/Projects" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountProjectId": 00001,
        "ProjectName": "Demo Project",
        "StartDate": null,
        "ProjectCode": null,
        "AccountClientName": "Demo Client",
        "LeaderName": "Employee 1",
        "ProjectManagerName": "Employee 2",
        "Deadline": "2018-07-20T13:52:00+00:00",
        "EstimatedDuration": null,
        "EstimatedDurationUnit": null,
        "Status": "Started",
        "ProjectDescription": " ",
        "CreatedOn": "2018-06-20T13:52:21.957+00:00",
        "ModifiedOn": "2018-06-20T13:52:21.957+00:00",
        "ProjectType": "Technology",
        "CustomField1": null,
        "CustomField2": null,
        "CustomField3": null,
        "CustomField4": null,
        "CustomField5": null,
        "CustomField6": null,
        "CustomField7": null,
        "CustomField8": null,
        "CustomField9": null,
        "CustomField10": null,
        "CustomField11": null,
        "CustomField12": null,
        "CustomField13": null,
        "CustomField14": null,
        "CustomField15": null,
        "Completed": "No"
    }
]
```

Retrieve a Project

Retrieves the Project with the given ID. Returns a Project object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Projects/{Project_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Projects/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountProjectId": 00001,
        "ProjectName": "Demo Project",
        "StartDate": null,
        "ProjectCode": null,
        "AccountClientName": "Demo Client",
        "LeaderName": "Employee 1",
        "ProjectManagerName": "Employee 2",
        "Deadline": "2018-07-20T13:52:00+00:00",
        "EstimatedDuration": null,
        "EstimatedDurationUnit": null,
        "Status": "Started",
        "ProjectDescription": " ",
        "CreatedOn": "2018-06-20T13:52:21.957+00:00",
        "ModifiedOn": "2018-06-20T13:52:21.957+00:00",
        "ProjectType": "Technology",
        "CustomField1": null,
        "CustomField2": null,
        "CustomField3": null,
        "CustomField4": null,
        "CustomField5": null,
        "CustomField6": null,
        "CustomField7": null,
        "CustomField8": null,
        "CustomField9": null,
        "CustomField10": null,
        "CustomField11": null,
        "CustomField12": null,
```

```
        "CustomField13": null,
        "CustomField14": null,
        "CustomField15": null,
        "Completed": "No"
    }
]
```

Create a Project

Creates a new Project. Returns Response message and a 201 Created response code if the call succeeded.

```
POST /API/Projects
```

Object Property	Required	Object Type	Description	Max Character Length
ProjectName	**Yes**	**String**	**Name Of Project**	**600**
AccountClientId	**Yes**	**Integer**	**Client Id Of Project**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/Projects" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountClientId':00001,'Projectname':'LEGO'}"
```

Example Response

```
[
  {
    "id": "00003",
    "message": "Project Added Successfully"
  }
]
```

Update a Project

Updates the specific Project by setting the values of the parameters passed. Any parameters not provided will be left unchanged. Returns Response message and a 200 OK response code if the call succeeded.

```
POST /API/Projects/{Project_id}
```

Example Request

```
curl "https://demo.livetecs.com/api/Projects/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'ProjectDescription':'Demo Description of Project'}"
```

Example Response

```
[
  {
    "id": "00003",
    "message": "Project Updated Successfully"
  }
]
```

Delete a Project

Delete a Project. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Projects/{Project_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Projects/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
  {
    "id": "00003",
    "message": "Project Deleted Successfully"
```

```
    }
]
```

15.3.6. Tasks

The Task Object

Object Property	Required	Object Type	Description	Max Character Length
TaskName	**Yes**	**String**	**Name Of Task**	**200**
AccountProjectId	**Yes**	**Integer**	**Project Id Of Task**	**10**
AccountProjectTaskId	No	Integer	Unique Task Id Of Task	10
ParentAccountProjectTaskId	No	Integer	Parent Task Id Of Task	10
TaskDescription	No	String	Task Description Of Task	4000
AccountTaskTypeId	No	Integer	Task Type Id Of Task	10
Duration	No	Decimal	Duration Of Task	10
DurationUnit	No	String	Duration Unit Of Task	20
DeadlineDate	No	Date Time	Deadline Date Of Task	19
CompletedPercent	No	Float	Completed Percent Of Task	10
Completed	No	Boolean	Completed Of Task	1
IsParentTask	No	Boolean	Parent Task Status Of Task	1
IsForAllEmployees	No	Boolean	Task is for All Employee Status Of Task	1
AccountPriorityId	No	Integer	Priority Id Of Task	10
TaskStatusId	No	Integer	Status Id Of Task	10
AccountProjectMilestoneId	No	Integer	Milestone Id Of Task	10
IsReOpen	No	Boolean	ReOpen Status Of Task	1
EstimatedCost	No	Float	Estimated Cost Of Task	10
EstimatedTimeSpent	No	Float	Estimated Time Spent Of Task	10
EstimatedTimeSpentUnit	No	String	Estimated Time Spent Unit Of Task	20
IsBillable	No	Boolean	Billable Status Of Task	1
IsDisabled	No	Boolean	Disabled Status Of Task	1
AccountBillingRateId	No	Integer	Billing Rate Id Of Task	10
IsForAllProjectTask	No	Boolean	All Project Task Status Of Task	1

TaskCode	No	String	Task Code Of Task	30
AccountProjectTaskTemplateId	No	Integer	Project Task Template Id Of Task	10
EstimatedCurrencyId	No	Integer	Estimate Currency Id Of Task	10
StartDate	No	Date Time	Start Date Of Task	19
Predecessors	No	String	Predecessors Of Task	50
CustomField1	No	String	CustomField 1 Of Task	2000
CustomField2	No	String	CustomField 2 Of Task	2000
CustomField3	No	String	CustomField 3 Of Task	2000
CustomField4	No	String	CustomField 4 Of Task	2000
CustomField5	No	String	CustomField 5 Of Task	2000
CustomField6	No	String	CustomField 6 Of Task	2000
CustomField7	No	String	CustomField 7 Of Task	2000
CustomField8	No	String	CustomField 8 Of Task	2000
CustomField9	No	String	CustomField 9 Of Task	2000
CustomField10	No	String	CustomField 10 Of Task	2000
CustomField11	No	String	CustomField 11 Of Task	2000
CustomField12	No	String	CustomField 12 Of Task	2000
CustomField13	No	String	CustomField 13 Of Task	2000
CustomField14	No	String	CustomField 14 Of Task	2000
CustomField15	No	String	CustomField 15 Of Task	2000
FixedCost	No	Float	Fixed Cost Of Task	10

List All Tasks

Returns a list of your all **Tasks** in your **TimeLive** account.

```
GET /API/Tasks
```

Example Request

```
curl "https://demo.livetecs.com/api/Tasks" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
```

```
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
  {
        "AccountProjectId": 00001,
        "TaskName": "gather requirements",
        "TaskDescription": "Sample requirements for project",
        "CompletedPercent": 0,
        "AccountProjectTaskId": 00001,
        "CreatedOn": "2017-12-29T12:39:14.813+00:00",
        "ModifiedOn": "2017-12-29T12:39:14.813+00:00",
        "Duration": 0,
        "DurationUnit": "Hours",
        "DeadlineDate": "2017-12-29T00:00:00+00:00",
        "EstimatedCost": 0,
        "EstimatedTimeSpentUnit": null,
        "EstimatedTimeSpent": 0,
        "TaskCode": "GR",
        "StartDate": "2017-12-29T00:00:00+00:00",
        "CustomField1": null,
        "CustomField2": null,
        "CustomField3": null,
        "CustomField4": null,
        "CustomField5": null,
        "CustomField6": null,
        "CustomField7": null,
        "CustomField8": null,
        "CustomField9": null,
        "CustomField10": null,
        "CustomField11": null,
        "CustomField12": null,
        "CustomField13": null,
        "CustomField14": null,
        "CustomField15": null,
        "Completed": "No",
        "IsParentTask": "No",
        "IsForAllEmployees": "No",
        "IsBillable": "No"
  }
]
```


Create a Task

Creates a **Task** in your **TimeLive** account and returns the Response Message.

```
POST /API/Tasks
```

Object Property	Required	Object Type	Description	Max Character Length
TaskName	**Yes**	**String**	**Name Of Task**	**200**
AccountProjectId	**Yes**	**Integer**	**Project Id Of Task**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/Tasks" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountProjectId':00001,'TaskName':'Demo Task'}"
```

Example Response

```
[
    {
        "id": "00003",
        "message": "Task Added Successfully"
    }
]
```

Update a Task

Update an **Task** in your **TimeLive** account and returns the Response Message.

```
POST /API/Tasks/{Task_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Tasks/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'TaskDescription':'Sample Description for Task'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "Task Updated Successfully"
    }
]
```

Delete a Task

Delete a **Task**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Task/{Task_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Tasks/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
  {
    "id": "00003",
    "message": "Task Deleted Successfully"
  }
]
```

15.3.7. Timesheet

The Timesheet Object

Object Property	Required	Object Type	Description	Max Character Length
Year	**Yes**	**Integer**	**Year Of Timesheet**	**4**
Month	**Yes**	**Integer**	**Month Of Timesheet**	**2**
Day	**Yes**	**Integer**	**Day Of Timesheet**	**2**
AccountProjectId	**Yes**	**Integer**	**Project Id Of Timesheet**	**10**
AccountProjectTaskId	**Yes**	**Integer**	**Task Id Of Timesheet**	**10**
TotalTime	**Yes**	**String**	**Total Time Of Timesheet**	**5**
StartTime	No	String	Start Time Of Timesheet	5
EndTime	No	String	End Time Of Timesheet	5
Description	No	String	Description Of Timesheet	100
AccountWorkTypeId	No	Integer	Work Type Id Of Timesheet	10
AccountCostCenterId	No	Integer	Cost Center Id Of Timesheet	10
Latitude	No	Double	Latitude Of Employee	10
Longitude	No	Double	Longitude Of Employee	10
IPAddress	No	String	IPAddress Of Employee	15

Create a Timesheet Entry

Creates a **Timesheet Entry** in your **TimeLive** account and returns the Response Message.

```
POST /API/Timesheets
```

Object Property	Required	Object Type	Description	Max Character Length
Year	**Yes**	**Integer**	**Year Of Timesheet**	**4**
Month	**Yes**	**Integer**	**Month Of Timesheet**	**2**
Day	**Yes**	**Integer**	**Day Of Timesheet**	**2**
AccountProjectId	**Yes**	**Integer**	**Project Id Of Timesheet**	**10**
AccountProjectTaskId	**Yes**	**Integer**	**Task Id Of Timesheet**	**10**

TotalTime	**Yes**	**String**	**Total Time Of Timesheet**	**5**

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountProjectId':00001,'TimesheetName':'Demo
Timesheet''Year':2018,'Month':1,'Day':1,'AccountProjectId':00001,'AccountProjectTaskId':00
```

Example Response

```
[
    {
        "message": "Insert Record Successfully."
    }
]
```

Update a Timesheet Entry

Update an **Timesheet Entry** in your **TimeLive** account and returns the Response Message.

```
POST /API/Timesheets/{Timesheet_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountProjectId':00001,'TimesheetName':'Demo
Timesheet''Year':2018,'Month':1,'Day':1,'AccountProjectId':00001,'AccountProjectTaskId':00
```

Example Response

```
[
    {
          "message": "Update Record Successfully."
```

```
    }
]
```

Delete a Timesheet Entry

Delete a **Timesheet Entry**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Timesheet/{Timesheet_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/00003" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
  {
    "id": "00003",
    "message": "Deleted Successfully"
  }
]
```

Get Timesheet Entries By Employee and Date Range

Display **Time Entries**. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/{Employee_ID}/{Start_Date}/{End_Date}
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/
TimeEntriesByEmployeeIdAndDateRange/00001/2018-01-01/2018-01-31" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "TimeEntryDate": "2018-01-01T00:00:00+00:00",
        "TotalTime": "2018-01-01T02:03:00+00:00",
        "ProjectName": "Demo Project",
        "ProjectCode": "DP",
        "TaskName": "Demo Task",
        "IsBillable": false,
        "TaskCode": "DT",
        "PartyName": "Demo Client",
        "PartyNick": "DC",
        "EmployeeName": "User 01",
        "EmployeeCode": "U-1",
        "MilestoneDescription": "Default Milestone",
        "AccountEmployeeType": "Administrator",
        "DepartmentName": "Admin Department",
        "AccountWorkType": "Standard",
        "AccountCostCenter": null,
        "ServiceItemName": null,
        "AccountProjectId": 00001,
        "AccountProjectTaskId": 00001,
        "TimeEntryDescription": "",
        "TaskDescription": "",
        "IsVendor": false,
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "AccountCostCenterId": null,
        "AccountPartyId": 00001,
        "AccountEmployeeTimeEntryId": 00001,
        "AccountTaskTypeId": 00001,
        "TaskType": "Task",
        "EmployeeManager": "",
        "ApprovalStatus": "Not Submitted",
        "Submitted": false,
        "ProjectType": "Marketing",
        "Approved": false,
        "Rejected": false
    }
]
```


Get Timesheet Entries By Timesheet Id

Display **Time Entries** of specific Timesheet Id. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/{Timesheet_Id}
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
{
    "Timesheets": {
        "ID": "00000000-0000-0000-0000-000000000000",
        "EmployeeId": 00001,
        "EmployeeName": "User 01",
        "StartDate": "2018-01-01T00:00:00",
        "EndDate": "2018-01-31T00:00:00",
        "TimesheetType": "Weekly",
        "Status": "Submitted",
        "SubmittedDate": "",
        "TotalHours": 2,
        "TotalMinutes": 120,
        "TotalSeconds": 7200,
        "ApprovalDate": "",
        "TimeEntryRows": [
            {
                "TimeEntryDate": "2018-01-01T00:00:00",
                "ProjectId": 00001,
                "ProjectName": "Demo Project",
                "TaskId": 00001,
                "TaskName": "Demo Task",
                "StartTime": null,
                "EndTime": null,
                "TotalTime": "2018-01-01T01:00:00",
                "Minutes": 60,
                "Hours": 1,
                "Description": "",
                "ClientId": 00001,
                "ClientName": "Demo Client",
                "BillingType": "Hourly",
                "IsBillable": false,
                "BillingRate": 0,
                "EmployeeRate": 50
            },
            {
                "TimeEntryDate": "2018-01-10T00:00:00",
                "ProjectId": 00002,
```

```
                "ProjectName": "Sample Project",
                "TaskId": 00002,
                "TaskName": "Sample Task",
                "StartTime": null,
                "EndTime": null,
                "TotalTime": "2018-01-10T01:00:00",
                "Minutes": 60,
                "Hours": 1,
                "Description": null,
                "ClientId": 00002,
                "ClientName": "Client 2",
                "BillingType": "Hourly",
                "IsBillable": false,
                "BillingRate": 0,
                "EmployeeRate": 50
            }
        ]
    }
}
```


Get Timesheet Period From Date

Display **Time Period** of specific Date. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetTimesheetPeriod/{ Year }/{ Month }/{ Day }
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetTimesheetPeriod/2018/01/01" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
{
    "StartDate": "2018-01-01T00:00:00+00:00",
    "EndDate": "2018-01-07T00:00:00+00:00",
    "TimesheetPeriodType": "Weekly",
    "TimesheetPeriodId": "00000000-0000-0000-0000-000000000000",
    "TimesheetStatus": "Not Submitted",
    "DisableTimeEntry": "No"
}
```

Get All Timesheet Entries by Date Range

Display **Timesheet Entries** within Date Range. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetAllTimeEntriesByDateRange/{Start_Date}/{End_Date}
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetAllTimeEntriesByDateRange/
2018-01-01/2018-01-02" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "TimeEntryDate": "2018-01-01T00:00:00+00:00",
        "TotalTime": "2018-01-01T01:00:00+00:00",
        "ProjectName": "Demo Project",
        "ProjectCode": "DP",
        "TaskName": "Demo Task",
        "TaskCode": "DT",
        "AccountClientName": "Demo Client",
        "AccountClientNick": "DC",
        "EmployeeName": "User 01",
        "EmployeeCode": "U-1",
        "MilestoneDescription": "Default Milestone",
        "AccountEmployeeType": "Administrator",
        "DepartmentName": "Admin Department",
        "AccountWorkType": "Standard",
        "AccountCostCenter": "Default Cost Center",
        "ServiceItemName": null,
        "AccountProjectId": 00001,
        "AccountProjectTaskId": 00001,
        "TimeEntryDescription": "Sample Time Entry Description",
        "TaskDescription": "Sample Task Description",
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "AccountCostCenterId": 0001,
        "AccountPartyId": 00001,
        "AccountEmployeeTimeEntryId": 00001,
        "AccountTaskTypeId": 00001,
```

```
        "TaskType": "Task",
        "EmployeeManager": "",
        "ApprovalStatus": "Not Submitted",
        "ProjectType": "Marketing",
        "IsBillable": "No",
        "IsVendor": "No",
        "Submitted": "No",
        "Approved": "No",
        "Rejected": "No"
    },
    {
        "TimeEntryDate": "2018-01-02T00:00:00+00:00",
        "TotalTime": "2018-01-01T01:00:00+00:00",
        "ProjectName": "Demo Project",
        "ProjectCode": "DP",
        "TaskName": "Demo Task",
        "TaskCode": "DT",
        "AccountClientName": "Demo Client",
        "AccountClientNick": "DC",
        "EmployeeName": "User 01",
        "EmployeeCode": "U-1",
        "MilestoneDescription": "Default Milestone",
        "AccountEmployeeType": "Administrator",
        "DepartmentName": "Admin Department",
        "AccountWorkType": "Standard",
        "AccountCostCenter": "Default Cost Center",
        "ServiceItemName": null,
        "AccountProjectId": 00001,
        "AccountProjectTaskId": 00001,
        "TimeEntryDescription": "Sample Time Entry Description",
        "TaskDescription": "Sample Task Description",
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "AccountCostCenterId": 0001,
        "AccountPartyId": 00001,
        "AccountEmployeeTimeEntryId": 00001,
        "AccountTaskTypeId": 00001,
        "TaskType": "Task",
        "EmployeeManager": "",
        "ApprovalStatus": "Not Submitted",
        "ProjectType": "Marketing",
        "IsBillable": "No",
        "IsVendor": "No",
        "Submitted": "No",
        "Approved": "No",
        "Rejected": "No"
    }
]
```

Get All Timesheet WorkDays with Hours by Week of Specific Date

Display **Timesheet WorkDays and Hours** by Week of Specific Date. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetTimesheetWorkingDaysWithHours/{ Year }/{ Month }/{ Day }
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetTimesheetWorkingDaysWithHours/
2018/08/01" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "TimeEntryDate": "2018-01-01T00:00:00+00:00",
        "TotalHours": "01:00",
        "ShowClientInTimesheet": "No",
        "ShowWorkTypeInTimesheet": "No",
        "ShowClockStartEnd": "No",
        "ShowCostCenterInTimesheet": "No"
    },
    {
        "TimeEntryDate": "2018-01-02T00:00:00+00:00",
        "TotalHours": "01:00",
        "ShowClientInTimesheet": "No",
        "ShowWorkTypeInTimesheet": "No",
        "ShowClockStartEnd": "No",
        "ShowCostCenterInTimesheet": "No"
    }
]
```


Get Timesheet Entry Details of Specific Date

Display **Timesheet Entry Details** of Specific Date. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetTimeEntryDetail/{ Year }/{ Month }/{ Day }
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetTimeEntryDetail/2018/08/01" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "AccountEmployeeTimeEntryID": 00001,
        "AccountProjectID": 00001,
        "AccountProjectTaskID": 00001,
        "ProjectName": "Demo Project",
        "TaskName": "Demo Task",
        "Description": "Sample Time Entry Description",
        "StartTime": "01/01/2018 1:00:00 PM",
        "EndTime": "01/01/2018 4:00:00 PM",
        "TotalHours": "03:00",
        "ShowClientInTimesheet": "No",
        "ShowWorkTypeInTimesheet": "No",
        "ShowCostCenterInTimesheet": "No",
        "ShowClockStartEnd": "No",
        "AccountClientId": 00001,
        "ClientName": "Demo Client",
        "WorkTypeName": "Standard",
        "AccountWorkTypeID": 0001,
        "TaskTypeID": 00001,
        "TaskType": "Task"
    },
    {
        "AccountEmployeeTimeEntryID": 00002,
        "AccountProjectID": 00001,
        "AccountProjectTaskID": 00001,
        "ProjectName": "Demo Project",
        "TaskName": "Demo Task",
        "Description": "Sample Time Entry Description",
        "StartTime": "01/01/2018 9:00:00 AM",
        "EndTime": "01/01/2018 11:00:00 AM",
        "TotalHours": "02:00",
        "ShowClientInTimesheet": "No",
        "ShowWorkTypeInTimesheet": "No",
        "ShowCostCenterInTimesheet": "No",
        "ShowClockStartEnd": "No",
        "AccountClientId": 00001,
        "ClientName": "Demo Client",
```

```
        "WorkTypeName": "Standard",
        "AccountWorkTypeID": 0001,
        "TaskTypeID": 00001,
        "TaskType": "Task"
    }
]
```

Get Types of Timesheet Approvals

Display **Approvals** of **Time Sheet**. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetTimesheetApprovalTypes
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetTimesheetApprovalTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "AccountApprovalTypeId": 00001,
        "ApprovalTypeName": "Team Lead --> Project Manager",
        "IsTimeOffApprovalTypes": "No"
    },
    {
        "AccountApprovalTypeId": 00002,
        "ApprovalTypeName": "Team Lead",
        "IsTimeOffApprovalTypes": "No"
    }
]
```

Get List of Submitted Timesheets Who are Not Approved Yet

Display **Approvable Time Sheet**. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetTimesheetApprovals
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetTimesheetApprovals" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "AccountEmployeeTimeEntryPeriodId":
"00000000-0000-0000-0000-000000000000",
        "TimeEntryAccountEmployeeId": 00001,
        "TimeEntryStartDate": "2018-01-01T00:00:00+00:00",
        "TimeEntryEndDate": "2018-01-07T00:00:00+00:00",
        "TimeEntryViewType": "Weekly",
        "EmployeeName": "Employee 01",
        "TotalMinutes": 120,
        "TimeEntryDate": "2018-01-01T00:00:00+00:00",
        "BillableTotalMinutes": 60,
        "NonBillableTotalMinutes": 60,
        "EMailAddress": "someone@example.com",
        "AccountProjectId": 00001,
        "Percentage": 50,
        "SystemApproverTypeId": 1
    }
]
```


Get Timesheet Period of Specific Employee by Date

Display **Timesheet Period** for the Week by Date of Employee . Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/GetTimesheetPeriodByEmployeeId/{Employee_Id}/{ Year }/{ Month
}/{ Day }
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/GetTimesheetPeriodByEmployeeId/
00001/2018/01/01" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
```

```
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "StartDate": "2018-01-01T00:00:00+00:00",
        "EndDate": "2018-01-07T00:00:00+00:00",
        "TimesheetPeriodType": "Weekly",
        "TimesheetPeriodId": "00000000-0000-0000-0000-000000000000",
        "TimesheetStatus": "Not Submitted",
        "DisableTimeEntry": "No"
    }
]
```


GET Submit Timesheet Of Employee

Submit **Timesheet** for specific PeriodId of Employee . Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/
SubmitTimesheetByEmployeeId/{Employee_Id}/{AccountEmployeeTimeEntryPeriod_Id}
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/SubmitTimesheetByEmployeeId/00001/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "Message": "Record Updated"
    }
]
```

GET Submit Own Timesheet

Submit Own **Timesheet** for specific PeriodId. Returns a 200 OK response code if the call succeeded.

```
GET /API/Timesheet/SubmitTimesheetByEmployeeId/{
AccountEmployeeTimeEntryPeriodId }
```

Example Request

```
curl "https://demo.livetecs.com/api/Timesheets/SubmitTimesheetByEmployeeId/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X GET \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "Message": "Record Updated"
    }
]
```

15.3.8. Absence

The absence Object

Object Property	Required	Object Type	Description	Max Character Length
AbsenceDescription	**Yes**	**String**	**Description of Absence**	**25**
AccountAbsenceTypeId	No	Integer	AbsenceTypeId of Absence	10
IsDisabled	No	Boolean	Disabled of Absence	1

List All absences

Returns a list of your all **absences** in your **TimeLive** account.

```
GET /API/absences
```

Example Request

```
curl "https://demo.livetecs.com/api/absences" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountAbsenceTypeId": 00001,
        "AbsenceDescription": "Sick Leave"
    },
    {
        "AccountAbsenceTypeId": 00002,
        "AbsenceDescription": "Holiday"
    }
]
```

Retrieve a absence

Retrieves the absence with the given ID. Returns a absence object and a 200 OK response code if a valid identifier was provided.

```
GET /API/absences/{absence_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/absences/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountAbsenceTypeId": 00002,
        "AbsenceDescription": "Holiday"
    }
]
```

Create a absence

Creates a **absence** in your **TimeLive** account and returns the Response Message.

```
POST /API/absences
```

Object Property	Required	Object Type	Description	Max Character Length
AbsenceDescription	**Yes**	**String**	**Description of Absence**	**25**

Example Request

```
curl "https://demo.livetecs.com/api/absences" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d "{'AbsenceDescription':'Demo absence'}"
```

Example Response

```
[
    {
        "id": "00000",
        "message": "Absence Type Added Successfully"
    }
]
```

Update a absence

Update an **absence** in your **TimeLive** account and returns the Response Message.

```
POST /API/absences/{absence_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/absences/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AbsenceDescription':'Sample absence'}"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "Absence Type Update Successfully"
    }
]
```

Delete a absence

Delete a **absence**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/absence/{absence_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/absences/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response


```
[
  {
    "id": "00002",
    "message": "Absence Type Delete Succeed"
  }
]
```

15.3.9. Billing Rates

The BillingRate Object

Object Property	Required	Object Type	Description	Max Character Length
BillingRate	**Yes**	**Decimal**	**Billing Rate of Billing Rate**	**10**
AccountBillingRateId	No	Integer	Billing Rate Id of Billing Rate	10
AccountWorkTypeId	No	Integer	Work Type Id of Billing Rate	10
AccountProjectEmployeeId	No	Integer	Project Employee Id of Billing Rate	10
AccountProjectRoleId	No	Integer	Project Role Id of Billing Rate	10
StartDate	No	DateTime	Start Date of Billing Rate	19
EndDate	No	DateTime	End Date of Billing Rate	19
AccountEmployeeId	No	Integer	Employee Id of Billing Rate	10
AccountProjectTaskId	No	Integer	Project Task Id of Billing Rate	10
EmployeeRate	No	Decimal	Employee Rate of Billing Rate	10
BillingRateCurrencyId	No	Integer	Billing Rate Currency Id of Billing Rate	10
EmployeeRateCurrencyId	No	Integer	Employee Rate Currency Id of Billing Rate	10

List All BillingRates

Returns a list of your all **BillingRates** in your **TimeLive** account.

```
GET /API/BillingRates
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountBillingRateId": 00001,
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "BillingRate": 50,
        "EmployeeRate": 50,
        "BillingRateCurrencyCode": "USD",
        "EmployeeRateCurrencyCode": "USD",
        "StartDate": "2017-11-01T00:00:00+00:00",
        "EndDate": "2018-11-01T00:00:00+00:00"
    }
]
```

Retrieve a BillingRate

Retrieves the BillingRate with the given ID. Returns a BillingRate object and a 200 OK response code if a valid identifier was provided.

```
GET /API/BillingRates/{BillingRate_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountBillingRateId": 00001,
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "BillingRate": 50,
        "EmployeeRate": 50,
        "BillingRateCurrencyCode": "USD",
        "EmployeeRateCurrencyCode": "USD",
        "StartDate": "2017-11-01T00:00:00+00:00",
        "EndDate": "2018-11-01T00:00:00+00:00"
```

```
    }
]
```

Retrieve BillingRates of Employee

Retrieves the BillingRates with the given ID. Returns a BillingRate object and a 200 OK response code if a valid identifier was provided.

```
GET /API/BillingRates/GetByEmployee/{Employee_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates/GetByEmployee/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountBillingRateId": 00001,
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "BillingRate": 50,
        "EmployeeRate": 50,
        "BillingRateCurrencyCode": "USD",
        "EmployeeRateCurrencyCode": "USD",
        "StartDate": "2017-11-01T00:00:00+00:00",
        "EndDate": "2018-11-01T00:00:00+00:00"
    }
]
```

Retrieve BillingRates of Employee for given WorkType

Retrieves the BillingRates with the given ID. Returns a BillingRate object and a 200 OK response code if a valid identifier was provided.

```
GET /API/BillingRates/GetByEmployeeAndWorkType/{Employee_ID}/{WorkType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates/GetByEmployeeAndWorkType/00001/
00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountBillingRateId": 00001,
        "AccountEmployeeId": 00001,
        "AccountWorkTypeId": 00001,
        "BillingRate": 50,
        "EmployeeRate": 50,
        "BillingRateCurrencyCode": "USD",
        "EmployeeRateCurrencyCode": "USD",
        "StartDate": "2017-11-01T00:00:00+00:00",
        "EndDate": "2018-11-01T00:00:00+00:00"
    }
]
```


Create a BillingRate

Creates a **BillingRate** in your **TimeLive** account and returns the Response Message.

```
POST /API/BillingRates
```

Object Property	Required	Object Type	Description	Max Character Length
BillingRate	**Yes**	**Decimal**	**Billing Rate of Billing Rate**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{"BillingRate":50}"
```

Example Response

```
[
    {
        "id": "00000",
        "message": "Billing Rate Added Successfully"
    }
]
```

Update a BillingRate

Update an **BillingRate** in your **TimeLive** account and returns the Response Message.

```
POST /API/BillingRates/{BillingRate_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{"BillingRate":60}"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "BillingRate Update Successfully"
    }
]
```

Delete a BillingRate

Delete a **BillingRate**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/BillingRate/{BillingRate_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingRates/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "Billing Rate Delete Succeed"
    }
]
```

15.3.10. Billing Types

The BillingType Object

Object Property	Required	Object Type	Description	Max Character Length
BillingType	**Yes**	**String**	**Billing Type of Billing Type**	**15**
AccountBillingTypeId	No	Integer	Billing Type Id of Billing Type	10
IsDisabled	No	Boolean	Disabled Status of Billing Type	1

List All BillingTypes

Returns a list of your all **BillingTypes** in your **TimeLive** account.

```
GET /API/BillingTypes
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountBillingTypeId": 0001,
        "BillingType": "Hourly",
        "BillingCategory": "Employee",
        "IsDisabled": "No"
    },
    {
        "AccountBillingTypeId": 0002,
        "BillingType": "Monthly",
        "BillingCategory": "Employee",
        "IsDisabled": "No"
    }
]
```

Retrieve a BillingType

Retrieves the BillingType with the given ID. Returns a BillingType object and a 200 OK response code if a valid identifier was provided.

```
GET /API/BillingTypes/{BillingType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingTypes/0002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountBillingTypeId": 0002,
        "BillingType": "Monthly",
        "BillingCategory": "Employee",
        "IsDisabled": "No"
    }
]
```

Create a BillingType

Creates a **BillingType** in your **TimeLive** account and returns the Response Message.

```
POST /API/BillingTypes
```

Object Property	Required	Object Type	Description	Max Character Length
BillingType	**Yes**	**String**	**Billing Type of Billing Type**	**15**

Example Request

```
curl "https://demo.livetecs.com/api/BillingTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
```

```
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'BillingType':"Hourly"}"
```

Example Response

```
[
    {
        "id": "00000",
        "message": "Billing Type Added Successfully"
    }
]
```

Update a BillingType

Update an **BillingType** in your **TimeLive** account and returns the Response Message.

```
POST /API/BillingTypes/{BillingType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingTypes/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{"BillingType":"Monthly"}"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "Billing Type Update Successfully"
    }
]
```

Delete a BillingType

Delete a **BillingType**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/BillingType/{BillingType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/BillingTypes/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "id": "00002",
        "message": "Succeed"
    }
]
```

15.3.11. Cost Centers

The CostCenter Object

Object Property	Required	Object Type	Description	Max Character Length
AccountCostCenterCode	**Yes**	**String**	**Cost Center Code of Cost Center**	**7**
AccountCostCenter	**Yes**	**String**	**Name of Cost Center**	**50**
AccountCostCenterId	No	Integer	Cost Center Id of Cost Center	10
IsDisabled	No	Boolean	Disabled Status of Cost Center	1

List All CostCenters

Returns a list of your all **CostCenters** in your **TimeLive** account.

```
GET /API/CostCenters
```

Example Request

```
curl "https://demo.livetecs.com/api/CostCenters" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountCostCenterId": 0001,
        "AccountCostCenterCode": "Default",
        "AccountCostCenter": "Default Cost Center",
        "IsDisabled": "No"
    }
]
```

Retrieve a CostCenter

Retrieves the CostCenter with the given ID. Returns a CostCenter object and a 200 OK response code if a valid identifier was provided.

```
GET /API/CostCenters/{CostCenter_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/CostCenters/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountCostCenterId": 0001,
        "AccountCostCenterCode": "Default",
        "AccountCostCenter": "Default Cost Center",
        "IsDisabled": "No"
    }
]
```

Create a CostCenter

Creates a **CostCenter** in your **TimeLive** account and returns the Response Message.

```
POST /API/CostCenters
```

Object Property	Required	Object Type	Description	Max Character Length
AccountCostCenterCode	**Yes**	**String**	**Cost Center Code of Cost Center**	**7**
AccountCostCenter	**Yes**	**String**	**Name of Cost Center**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/CostCenters" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountCostCenterCode': 'tcc','AccountCostCenter': 'temp Cost
Center'}"
```

Example Response

```
[
    {
        "id": "00000",
        "message": "Cost Center Added Successfully"
    }
]
```

Update a CostCenter

Update an **CostCenter** in your **TimeLive** account and returns the Response Message.

```
POST /API/CostCenters/{CostCenter_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/CostCenters/00001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountCostCenter': 'temp Cost Center'}"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "Cost Center Update Successfully"
    }
]
```

Delete a CostCenter

Delete a **CostCenter**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/CostCenter/{CostCenter_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/CostCenters/00002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "Cost Center Delete Succeed"
    }
]
```

15.3.12. Employee Types

The EmployeeType Object

Object Property	Required	Object Type	Description	Max Character Length
AccountEmployeeType	**Yes**	**String**	**Employee Type of EmployeeType**	**50**
AccountEmployeeTypeId	No	Guid	Employee Type Id of EmployeeType	32
IsDisabled	No	Boolean	Disabled Status of EmployeeType	1
IsVendor	No	Boolean	Vendor Status of EmployeeType	1

List All EmployeeTypes

Returns a list of your all **EmployeeTypes** in your **TimeLive** account.

```
GET /API/EmployeeTypes
```

Example Request

```
curl "https://demo.livetecs.com/api/EmployeeTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountEmployeeTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountEmployeeType": "Demo",
        "IsDisabled": "No",
        "IsVendor": "No"
    }
]
```

Retrieve a EmployeeType

Retrieves the EmployeeType with the given ID. Returns a EmployeeType object and a 200 OK response code if a valid identifier was provided.

```
GET /API/EmployeeTypes/{EmployeeType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/EmployeeTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountEmployeeTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountEmployeeTypeCode": "Demo",
        "AccountEmployeeType": "Demo User",
        "IsDisabled": "No"
    }
]
```

Create a EmployeeType

Creates a **EmployeeType** in your **TimeLive** account and returns the Response Message.

```
POST /API/EmployeeTypes
```

Object Property	Required	Object Type	Description	Max Character Length
AccountEmployeeType	**Yes**	**String**	**Employee Type of EmployeeType**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/EmployeeTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountEmployeeType': 'Demo'}"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Employee Type Added Successfully"
    }
]
```

Update a EmployeeType

Update an **EmployeeType** in your **TimeLive** account and returns the Response Message.

```
POST /API/EmployeeTypes/{EmployeeType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/EmployeeTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountEmployeeType': 'Sample'}"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "Employee Type Update Successfully"
    }
]
```

Delete a EmployeeType

Delete a **EmployeeType**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/EmployeeType/{EmployeeType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/EmployeeTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Employee Type Deleted successfully"
    }
]
```

15.3.13. Expense Type

The ExpenseType Object

Object Property	Required	Object Type	Description	Max Character Length
ExpenseType	**Yes**	**String**	**Expense Type of expenseType**	**50**
AccountExpenseTypeId	No	Integer	Expense Type Id of expenseType	32
IsQuantityField	No	Boolean	Quantity Field Status of expenseType	1
IsDisabled	No	Boolean	Disabled Status of expenseType	1

List All expenseTypes

Returns a list of your all **expenseTypes** in your **TimeLive** account.

```
GET /API/expenseTypes
```

Example Request

```
curl "https://demo.livetecs.com/api/expenseTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountExpenseTypeId": 0001,
        "ExpenseType": "Demo Expense",
        "QuantityFieldCaption": null,
        "IsQuantityField": "No",
        "IsDisabled": "No"
    },
    {
        "AccountExpenseTypeId": 0002,
        "ExpenseType": "sample expense",
        "QuantityFieldCaption": null,
```

```
        "IsQuantityField": "No",
        "IsDisabled": "No"
    }
]
```


Retrieve a expenseType

Retrieves the expenseType with the given ID. Returns a expenseType object and a 200 OK response code if a valid identifier was provided.

```
GET /API/expenseTypes/{expenseType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/expenseTypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountExpenseTypeId": 0001,
        "ExpenseType": "Demo Expense",
        "QuantityFieldCaption": null,
        "IsQuantityField": "No",
        "IsDisabled": "No"
    }
]
```


Create a expenseType

Creates a **expenseType** in your **TimeLive** account and returns the Response Message.

```
POST /API/expenseTypes
```

Object Property	Required	Object Type	Description	Max Character Length
ExpenseType	**Yes**	**String**	**Expense Type of expenseType**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/expenseTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'ExpenseType': 'Demo'}"
```

Example Response

```
[
    {
        "id": "0",
        "message": "expense Type Added Successfully"
    }
]
```

Update a expenseType

Update an **expenseType** in your **TimeLive** account and returns the Response Message.

```
POST /API/expenseTypes/{expenseType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/expenseTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountexpenseType': 'Sample'}"
```

Example Response

```
[
    {
        "id": "00002",
        "message": "expense Type Update Successfully"
    }
]
```

Delete a expenseType

Delete a **expenseType**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/expenseType/{expenseType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/expenseTypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "expense Type Deleted successfully"
    }
]
```

15.3.14. Expenses

The expense Object

Object Property	Required	Object Type	Description	Max Character Length
AccountExpenseName	**Yes**	**String**	**Expense Name of expense**	**50**
AccountExpenseTypeId	**Yes**	**String**	**Expense Type Id of expense**	**50**
AccountExpenseId	No	Integer	Expense Id of expense	32
DefaultExpenseRate	No	Boolean	Default Expense Rate of expense	1
DisabledEditingOfRate	No	Boolean	Disabled Editing Of Rate of expense	1
IsBillable	No	Boolean	Billable Status of expense	2
IsDisabled	No	Boolean	Disabled Status of expense	3

List All expenses

Returns a list of your all **expenses** in your **TimeLive** account.

```
GET /API/expenses
```

Example Request

```
curl "https://demo.livetecs.com/api/expenses" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountExpenseId": 0001,
        "AccountExpenseName": "Air Travel",
        "ExpenseType": "Air Travel",
        "DefaultExpenseRate": null,
        "DisabledEditingOfRate": "No",
```

```
        "IsBillable": "Yes",
        "IsDisabled": "No"
    },
    {
        "AccountExpenseId": 0002,
        "AccountExpenseName": "Travel Amount",
        "ExpenseType": "Air Travel",
        "DefaultExpenseRate": null,
        "DisabledEditingOfRate": "No",
        "IsBillable": null,
        "IsDisabled": "No"
    }
]
```

Retrieve a expense

Retrieves the expense with the given ID. Returns a expense object and a 200 OK response code if a valid identifier was provided.

```
GET /API/expenses/{expense_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/expenses/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountExpenseId": 0001,
        "AccountExpenseName": "Air Travel",
        "ExpenseType": "Air Travel",
        "DefaultExpenseRate": null,
        "DisabledEditingOfRate": "No",
        "IsBillable": "Yes",
        "IsDisabled": "No"
    }
]
```

Create a expense

Creates a **expense** in your **TimeLive** account and returns the Response Message.

```
POST /API/expenses
```

Object Property	Required	Object Type	Description	Max Character Length
AccountExpenseName	**Yes**	**String**	**Expense Name of expense**	**50**
AccountExpenseTypeId	**Yes**	**String**	**Expense Type Id of expense**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/expenses" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountExpenseName': 'Demo','AccountExpenseTypeId': 0001}"
```

Example Response

```
[
    {
        "id": "0",
        "message": "expense Added Successfully"
    }
]
```

Update a expense

Update an **expense** in your **TimeLive** account and returns the Response Message.

```
POST /API/expenses/{expense_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/expenses/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
```

```
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountExpenseName': 'Sample'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "expense Update Successfully"
    }
]
```

Delete a expense

Delete a **expense**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/expense/{expense_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/expenses/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "expense Deleted successfully"
    }
]
```

15.3.15. Holiday Type

The HolidayType Object

Object Property	Required	Object Type	Description	Max Character Length
AccountHolidayType	**Yes**	**String**	**Holiday type of Billing**	**50**
AccountHolidayTypeId	No	Integer	Holiday Type Id of Billing	10
IsDisabled	No	Boolean	Disabled Status of Billing	1

List All HolidayTypes

Returns a list of your all **HolidayTypes** in your **TimeLive** account.

```
GET /API/HolidayTypes
```

Example Request

```
curl "https://demo.livetecs.com/api/HolidayTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountHolidayTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountHolidayType": "Demo",
        "IsDeleted": null,
        "IsDisabled": "No"
    },
    {
        "AccountHolidayTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountHolidayType": "Demo 2",
        "IsDeleted": null,
        "IsDisabled": "No"
    }
]
```

Retrieve a HolidayType

Retrieves the HolidayType with the given ID. Returns a HolidayType object and a 200 OK response code if a valid identifier was provided.

```
GET /API/HolidayTypes/{HolidayType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/HolidayTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountHolidayTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountHolidayType": "Demo",
        "IsDeleted": null,
        "IsDisabled": "No"
    }
]
```

Create a HolidayType

Creates a **HolidayType** in your **TimeLive** account and returns the Response Message.

```
POST /API/HolidayTypes
```

Object Property	Required	Object Type	Description	Max Character Length
AccountHolidayType	**Yes**	**String**	**HolidayType type of Billing**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/HolidayTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
```

```
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountHolidayType': 'Sample'}"
```

Example Response

```
[
    {
        "id": "0",
        "message": "HolidayType Added Successfully"
    }
]
```

Update a HolidayType

Update an **HolidayType** in your **TimeLive** account and returns the Response Message.

```
POST /API/HolidayTypes/{HolidayType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/HolidayTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountHolidayType': 'Demo'}"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "HolidayType Update Successfully"
    }
]
```

Delete a HolidayType

Delete a **HolidayType**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/HolidayType/{HolidayType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/HolidayTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "HolidayType Deleted successfully"
    }
]
```

15.3.16. Holidays

The Holiday Object

Object Property	Required	Object Type	Description	Max Character Length
AccountHolidayId	**Yes**	**Guid**	**Holiday Type Id of Holiday**	**32**
HolidayName	**Yes**	**String**	**Holiday Name of Holiday**	**25**
HolidayDate	**Yes**	**DateTime**	**Holiday Date of Holiday**	**19**
AccountHolidayDetailId	No	Guid	Holiday Type Detail Id of Holiday	32

List All Holidays

Returns a list of your all **Holidays** in your **TimeLive** account.

```
GET /API/Holidays
```

Example Request

```
curl "https://demo.livetecs.com/api/Holidays" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountHolidayDetailId": "00000000-0000-0000-0000-000000000000",
        "AccountHolidayId": "00000000-0000-0000-0000-000000000000",
        "HolidayName": "Demo",
        "HolidayDate": "2018-01-01T00:00:00+00:00"
    },
    {
        "AccountHolidayDetailId": "00000000-0000-0000-0000-000000000000",
        "AccountHolidayId": "00000000-0000-0000-0000-000000000000",
        "HolidayName": "Demo 2",
        "HolidayDate": "2018-01-01T00:00:00+00:00"
    }
]
```

Retrieve a Holiday

Retrieves the Holiday with the given ID. Returns a Holiday object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Holidays/{Holiday_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Holidays/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountHolidayDetailId": "00000000-0000-0000-0000-000000000000",
        "AccountHolidayId": "00000000-0000-0000-0000-000000000000",
        "HolidayName": "Demo 2",
        "HolidayDate": "2018-01-01T00:00:00+00:00"
    }
]
```

Create a Holiday

Creates a **Holiday** in your **TimeLive** account and returns the Response Message.

```
POST /API/Holidays
```

Object Property	Required	Object Type	Description	Max Character Length
AccountHolidayId	**Yes**	**Guid**	**Holiday Type Id of Holiday**	**32**
HolidayName	**Yes**	**String**	**Holiday Name of Holiday**	**25**
HolidayDate	**Yes**	**DateTime**	**Holiday Date of Holiday**	**19**

Example Request

```
curl "https://demo.livetecs.com/api/Holidays" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'HolidayName':
'Sample','AccountHolidayId':'00000000-0000-0000-0000-000000000000','HolidayDate','2018-01
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Holiday Added Successfully"
    }
]
```

Update a Holiday

Update an **Holiday** in your **TimeLive** account and returns the Response Message.

```
POST /API/Holidays/{Holiday_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Holidays/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'HolidayDate','2018-01-01'}"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Holiday Update Successfully"
```

```
    }
]
```

Delete a Holiday

Delete a **Holiday**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Holiday/{Holiday_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Holidays/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Holiday Deleted successfully"
    }
]
```

15.3.17. Locations

The Location Object

Object Property	Required	Object Type	Description	Max Character Length
AccountLocation	**Yes**	**String**	**Location Name of Location**	**50**
AccountLocationId	No	Integer	Id of Location	11
IsDisabled	No	Boolean	Disabled Status of Location	1

List All Locations

Returns a list of your all **Locations** in your **TimeLive** account.

```
GET /API/Locations
```

Example Request

```
curl "https://demo.livetecs.com/api/Locations" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountLocationId": 0001,
        "AccountLocation": "Default Location",
        "IsDisabled": "No"
    },
    {
        "AccountLocationId": 0002,
        "AccountLocation": "Other",
        "IsDisabled": "No"
    }
]
```

Retrieve a Location

Retrieves the Location with the given ID. Returns a Location object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Locations/{Location_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Locations/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
   {
        "AccountLocationId": 0001,
        "AccountLocation": "Default Location",
        "IsDisabled": "No"
    }
]
```

Create a Location

Creates a **Location** in your **TimeLive** account and returns the Response Message.

```
POST /API/Locations
```

Object Property	Required	Object Type	Description	Max Character Length
AccountLocation	**Yes**	**String**	**Location Name of Location**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/Locations" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d "{'AccountLocation': 'Sample'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Location Added Successfully"
    }
]
```

Update a Location

Update an **Location** in your **TimeLive** account and returns the Response Message.

```
POST /API/Locations/{Location_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Locations/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountLocation': 'Sample'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Location Update Successfully"
    }
]
```

Delete a Location

Delete a **Location**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Location/{Location_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Locations/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Location Deleted successfully"
    }
]
```

15.3.18. Payment Methods

The PaymentMethods Object

Object Property	Required	Object Type	Description	Max Character Length
PaymentMethod	**Yes**	**String**	**Payment Method of PaymentMethod**	**50**
AccountPaymentMethodId	No	Integer	Id of PaymentMethod	10
IsDisabled	No	Boolean	Disabled Status of PaymentMethod	1

List All PaymentMethods

Returns a list of your all **PaymentMethods** in your **TimeLive** account.

```
GET /API/PaymentMethods
```

Example Request

```
curl "https://demo.livetecs.com/api/PaymentMethods" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountPaymentMethodId": 0001,
        "PaymentMethod": "American Express",
        "IsDisabled": "No"
    },
    {
        "AccountPaymentMethodId": 0002,
        "PaymentMethod": "Cash",
        "IsDisabled": "No"
    }
]
```

Retrieve a PaymentMethods

Retrieves the PaymentMethods with the given ID. Returns a PaymentMethods object and a 200 OK response code if a valid identifier was provided.

```
GET /API/PaymentMethods/{PaymentMethods_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/PaymentMethods/0002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
   {
        "AccountPaymentMethodId": 0002,
        "PaymentMethod": "Cash",
        "IsDisabled": "No"
    }
]
```

Create a PaymentMethods

Creates a **PaymentMethods** in your **TimeLive** account and returns the Response Message.

```
POST /API/PaymentMethods
```

Object Property	Required	Object Type	Description	Max Character Length
PaymentMethod	**Yes**	**String**	**Payment Method of PaymentMethod**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/PaymentMethods" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d "{'PaymentMethod': 'demo'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "Payment Method Added Successfully"
    }
]
```

Update a PaymentMethods

Update an **PaymentMethods** in your **TimeLive** account and returns the Response Message.

```
POST /API/PaymentMethods/{PaymentMethods_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/PaymentMethods/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'PaymentMethod': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Payment Method Update Successfully"
    }
]
```

Delete a PaymentMethods

Delete a **PaymentMethods**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/PaymentMethods/{PaymentMethods_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/PaymentMethods/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Payment Method Deleted successfully"
    }
]
```

15.3.19. Priorities

The Priorities Object

Object Property	Required	Object Type	Description	Max Character Length
Priority	**Yes**	**String**	**Name of priority**	**50**
AccountPriorityId	No	Integer	Id of priority	11
PriorityOrder	No	Integer	Order of priority	2
IsDisabled	No	Boolean	Disabled Status of priority	1

List All Priorities

Returns a list of your all **Priorities** in your **TimeLive** account.

```
GET /API/Priorities
```

Example Request

```
curl "https://demo.livetecs.com/api/Priorities" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountPriorityId": 0001,
        "Priority": "Urgent",
        "PriorityOrder": 1,
        "IsDisabled": "No"
    },
    {
        "AccountPriorityId": 0002,
        "Priority": "High",
        "PriorityOrder": 10,
        "IsDisabled": "No"
    }
]
```

Retrieve a Priorities

Retrieves the Priorities with the given ID. Returns a Priorities object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Priorities/{Priorities_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Priorities/0002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
   {
        "AccountPriorityId": 0002,
        "Priority": "Urgent",
        "PriorityOrder": 1,
        "IsDisabled": "No"
    }
]
```

Create a Priorities

Creates a **Priorities** in your **TimeLive** account and returns the Response Message.

```
POST /API/Priorities
```

Object Property	Required	Object Type	Description	Max Character Length
priority	**Yes**	**String**	**Payment Method of priority**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/Priorities" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
```

```
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'priority': 'demo','PriorityOrder':3}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "Priority Added Successfully"
    }
]
```

Update a Priorities

Update an **Priorities** in your **TimeLive** account and returns the Response Message.

```
POST /API/Priorities/{Priorities_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Priorities/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'priority': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Priority Updated Successfully"
    }
]
```

Delete a Priorities

Delete a **Priorities**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Priorities/{Priorities_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Priorities/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Priority Delete Successfully"
    }
]
```

15.3.20. Project Types

The Projecttypes Object

Object Property	Required	Object Type	Description	Max Character Length
ProjectType	**Yes**	**String**	**Type of Project**	**50**
AccountProjectTypeId	No	Integer	Id of Project	10
IsDisabled	No	Boolean	Disabled Status of Project	1

List All Projecttypes

Returns a list of your all **Projecttypes** in your **TimeLive** account.

```
GET /API/Projecttypes
```

Example Request

```
curl "https://demo.livetecs.com/api/Projecttypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
    {
        "AccountProjectTypeId": 0001,
        "ProjectType": "Technology",
        "IsDisabled": "No"
    },
    {
        "AccountProjectTypeId": 0002,
        "ProjectType": "Marketing",
        "IsDisabled": "No"
    }
]
```

Retrieve a Projecttypes

Retrieves the Projecttypes with the given ID. Returns a Projecttypes object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Projecttypes/{Projecttypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Projecttypes/0002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
   {
        "AccountProjectTypeId": 0001,
        "ProjectType": "Technology",
        "IsDisabled": "No"
    }
]
```

Create a Projecttypes

Creates a **Projecttypes** in your **TimeLive** account and returns the Response Message.

```
POST /API/Projecttypes
```

Object Property	Required	Object Type	Description	Max Character Length
ProjectType	**Yes**	**String**	**Type of Project**	**50**

Example Request

```
curl "https://demo.livetecs.com/api/Projecttypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d "{'ProjectType': 'demo'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "ProjectType Added Successfully"
    }
]
```

Update a Projecttypes

Update an **Projecttypes** in your **TimeLive** account and returns the Response Message.

```
POST /API/Projecttypes/{Projecttypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Projecttypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'ProjectType': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "ProjectType Updated Successfully"
    }
]
```

Delete a Projecttypes

Delete a **Projecttypes**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Projecttypes/{Projecttypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Projecttypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "ProjectType Delete Successfully"
    }
]
```

15.3.21. Status

The statuses Object

Object Property	Required	Object Type	Description	Max Character Length
Status	**Yes**	**String**	**Status Name of Status**	**50**
SystemStatusType	**Yes**	**String**	**System Status Type of Status**	**9**
AccountStatusId	No	Integer	Id of Status	11
IsDisabled	No	Boolean	Disabled Status of Status	1

List All statuses

Returns a list of your all **statuses** in your **TimeLive** account.

```
GET /API/statuses
```

Example Request

```
curl "https://demo.livetecs.com/api/statuses" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
```

“AccountStatusId”: 0001, “Status”: “Started”, “SystemStatusType”: “Task”, “IsDisabled”: “No” }, { “AccountStatusId”: 0002, “Status”: “OnHold”, “SystemStatusType”: “Task”, “IsDisabled”: “No” }]

Retrieve a statuses

Retrieves the statuses with the given ID. Returns a statuses object and a 200 OK response code if a valid identifier was provided.

```
GET /API/statuses/{statuses_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/statuses/0002" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
    {
        "AccountStatusId": 0002,
        "Status": "OnHold",
        "SystemStatusType": "Task",
        "IsDisabled": "No"
    }
]
```

Create a statuses

Creates a **statuses** in your **TimeLive** account and returns the Response Message.

```
POST /API/statuses
```

Object Property	Required	Object Type	Description	Max Character Length
Status	**Yes**	**String**	**Status Name of Status**	**50**
SystemStatusType	**Yes**	**String**	**System Status Type of Status**	**9**

Example Request

```
curl "https://demo.livetecs.com/api/statuses" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'Status': 'demo','SystemStatusType':'project'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "status Added Successfully"
    }
]
```

Update a statuses

Update an **statuses** in your **TimeLive** account and returns the Response Message.

```
POST /API/statuses/{statuses_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/statuses/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'status': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "status Updated Successfully"
    }
]
```

Delete a statuses

Delete a **statuses**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/statuses/{statuses_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/statuses/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "id": "0001",
        "message": "status Delete Successfully"
    }
]
```

15.3.22. Task Types

The TaskType Object

Object Property	Required	Object Type	Description	Max Character Length
TaskType	**Yes**	**String**	**Task Type of Task Type**	**10**
AccountTaskTypeId	No	Integer	Id of Task Type	10
IsDisabled	No	Boolean	Disabled Status of Task Type	1

List All TaskType

Returns a list of your all **TaskType** in your **TimeLive** account.

```
GET /API/TaskType
```

Example Request

```
curl "https://demo.livetecs.com/api/TaskType" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
     {
        "AccountTaskTypeId": 0001,
        "TaskType": "Task",
        "IsDisabled": "No"
    },
    {
        "AccountTaskTypeId": 0002,
        "TaskType": "Bug",
        "IsDisabled": "No"
    }
]
```

Retrieve a TaskType

Retrieves the TaskType with the given ID. Returns a TaskType object and a 200 OK response code if a valid identifier was provided.

```
GET /API/TaskType/{TaskType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaskType/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTaskTypeId": 0001,
        "TaskType": "Task",
        "IsDisabled": "No"
    }
]
```

Create a TaskType

Creates a **TaskType** in your **TimeLive** account and returns the Response Message.

```
POST /API/TaskType
```

Object Property	Required	Object Type	Description	Max Character Length
TaskType	**Yes**	**String**	**Task Type of Task Type**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/TaskType" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d "{'TaskType': 'demo'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "TaskType Added Successfully"
    }
]
```

Update a TaskType

Update an **TaskType** in your **TimeLive** account and returns the Response Message.

```
POST /API/TaskType/{TaskType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaskType/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'TaskType': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "TaskType Updated Successfully"
    }
]
```

Delete a TaskType

Delete a **TaskType**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/TaskType/{TaskType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaskType/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "TaskType Delete Successfully"
    }
]
```

15.3.23. Tax Code

The TaxCode Object

Object Property	Required	Object Type	Description	Max Character Length
TaxCode	**Yes**	**String**	**TaxCode of Task Type**	**10**
Formula	**Yes**	**String**	**Formula of Task Type**	**10**
AccountTaxCodeId	No	Integer	Id of Task Type	10
IsDisabled	No	Boolean	Disabled Status of Task Type	1

List All TaxCodes

Returns a list of your all **TaxCodes** in your **TimeLive** account.

```
GET /API/TaxCodes
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxCodes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTaxCodeId": 0001,
        "TaxCode": "Airport Tax",
        "Formula": "10.00",
        "IsDisabled": "No"
    },
    {
        "AccountTaxCodeId": 0002,
        "TaxCode": "GST",
        "Formula": "Net * 0.05",
        "IsDisabled": "No"
    }
]
```

Retrieve a TaxCode

Retrieves the TaxCode with the given ID. Returns a TaxCode object and a 200 OK response code if a valid identifier was provided.

```
GET /API/TaxCodes/{TaxCode_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxCodes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTaxCodeId": 13949,
        "TaxCode": "Airport Tax",
        "Formula": "10.00",
        "IsDisabled": "No"
    }
]
```

Create a TaxCode

Creates a **TaxCode** in your **TimeLive** account and returns the Response Message.

```
POST /API/TaxCodes
```

Object Property	Required	Object Type	Description	Max Character Length
TaxCode	**Yes**	**String**	**TaxCode of Task Type**	**10**
Formula	**Yes**	**String**	**Formula of Task Type**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/TaxCodes" \
      -H "APIKey: {Your API Key}" \
```

```
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'TaxCode': 'demo','Formula':'Net*0.15'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "TaxCode Added Successfully"
    }
]
```

Update a TaxCode

Update an **TaxCode** in your **TimeLive** account and returns the Response Message.

```
POST /API/TaxCodes/{TaxCode_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxCodes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'TaxCode': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "TaxCode Updated Successfully"
    }
]
```

Delete a TaxCode

Delete a **TaxCode**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/TaxCodes/{TaxCode_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxCodes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "TaxCode Delete Successfully"
    }
]
```

15.3.24. Tax Zones

The TaxZone Object

Object Property	Required	Object Type	Description	Max Character Length
AccountTaxZone	**Yes**	**String**	**TaxZone of Taxzone**	**10**
AccountTaxZoneId	No	Integer	Id of Taxzone	10
IsDisabled	No	Boolean	Disabled Status of Taxzone	1

List All TaxZones

Returns a list of your all **TaxZones** in your **TimeLive** account.

```
GET /API/TaxZones
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxZones" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTaxZoneId": 0001,
        "AccountTaxZone": "Default",
        "IsDisabled": "No"
    },
    {
        "AccountTaxZoneId": 0002,
        "AccountTaxZone": "Sample",
        "IsDisabled": "No"
    }
]
```

Retrieve a TaxZone

Retrieves the TaxZone with the given ID. Returns a TaxZone object and a 200 OK response code if a valid identifier was provided.

```
GET /API/TaxZones/{TaxZone_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxZones/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTaxZoneId": 0001,
        "AccountTaxZone": "Default",
        "IsDisabled": "No"
    }
]
```

Create a TaxZone

Creates a **TaxZone** in your **TimeLive** account and returns the Response Message.

```
POST /API/TaxZones
```

Object Property	Required	Object Type	Description	Max Character Length
AccountTaxZone	**Yes**	**String**	**TaxZone of Taxzone**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/TaxZones" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
```

```
      -H "Content-Type: application/json" \
      -d "{'AccountTaxZone': 'demo'}"
```

Example Response

```
[
    {
        "id": "00001",
        "message": "Tax Zone Added Successfully"
    }
]
```

Update a TaxZone

Update an **TaxZone** in your **TimeLive** account and returns the Response Message.

```
POST /API/TaxZones/{TaxZone_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxZones/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountTaxZone': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Tax Zone Updated Successfully"
    }
]
```

Delete a TaxZone

Delete a **TaxZone**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/TaxZones/{TaxZone_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/TaxZones/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "id": "0001",
        "message": "Tax Zone Deleted Successfully"
    }
]
```

15.3.25. Timeoff Types

The AccountTimeOffType Object

Object Property	Required	Object Type	Description	Max Character Length
AccountTimeOffType	**Yes**	**String**	**Name of Timeoff Type**	**10**
AccountTimeOffTypeId	No	Guid	Id of Timeoff Type	32
IsTimeOffRequestRequired	No	Boolean	Request Required Status of Timeoff Type	1
IsDisabled	No	Boolean	Disabled Status of Timeoff Type	1

List All AccountTimeOffTypes

Returns a list of your all **AccountTimeOffTypes** in your **TimeLive** account.

```
GET /API/AccountTimeOffTypes
```

Example Request

```
curl "https://demo.livetecs.com/api/AccountTimeOffTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTimeOffTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountTimeOffType": "Holiday",
        "IsTimeOffRequestRequired": "No",
        "IsDisabled": "No"
    },
    {
        "AccountTimeOffTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountTimeOffType": "Other",
        "IsTimeOffRequestRequired": "Yes",
        "IsDisabled": "No"
```

```
    }
]
```

Retrieve a AccountTimeOffType

Retrieves the AccountTimeOffType with the given ID. Returns a AccountTimeOffType object and a 200 OK response code if a valid identifier was provided.

```
GET /API/AccountTimeOffTypes/{TimeOffTypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/AccountTimeOffTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountTimeOffTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountTimeOffType": "Holiday",
        "IsTimeOffRequestRequired": "No",
        "IsDisabled": "No"
    }
]
```

Create a AccountTimeOffType

Creates a **AccountTimeOffType** in your **TimeLive** account and returns the Response Message.

```
POST /API/AccountTimeOffTypes
```

Object Property	Required	Object Type	Description	Max Character Length
AccountTimeOffType	**Yes**	**String**	**Type of Timeoff Type**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/AccountTimeOffTypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountTimeOffType': 'demo'}"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Time Off Type Added Successfully"
    }
]
```

Update a AccountTimeOffType

Update an **AccountTimeOffType** in your **TimeLive** account and returns the Response Message.

```
POST /API/AccountTimeOffTypes/{AccountTimeOffType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/AccountTimeOffTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'IsDisabled': true}"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Time Off Type Updated Successfully"
    }
]
```

Delete a AccountTimeOffType

Delete a **AccountTimeOffType**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/AccountTimeOffTypes/{AccountTimeOffType_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/AccountTimeOffTypes/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "Time Off Type Deleted Successfully"
    }
]
```

15.3.26. Work Types

The Worktype Object

Object Property	Required	Object Type	Description	Max Character Length
AccountWorkTypeCode	**Yes**	**String**	**Code of Worktype**	**10**
AccountWorkType	**Yes**	**String**	**Name of Worktype**	**10**
AccountWorkTypeId	No	Guid	Id of Worktype	32
IsDisabled	No	Boolean	Disabled Status of Worktype	1

List All Worktypes

Returns a list of your all **Worktypes** in your **TimeLive** account.

```
GET /API/Worktypes
```

Example Request

```
curl "https://demo.livetecs.com/api/Worktypes" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountWorkTypeId": 0001,
        "AccountWorkTypeCode": "ST",
        "AccountWorkType": "Standard",
        "IsDisabled": "No"
    },
    {
        "AccountWorkTypeId": 0002,
        "AccountWorkTypeCode": "OT",
        "AccountWorkType": "Overtime",
        "IsDisabled": "No"
    }
]
```

Retrieve a Worktype

Retrieves the **Worktype** with the given ID. Returns a Worktype object and a 200 OK response code if a valid identifier was provided.

```
GET /API/Worktypes/{Worktype_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Worktypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response

```
[
     {
        "AccountWorkTypeId": 0001,
        "AccountWorkTypeCode": "ST",
        "AccountWorkType": "Standard",
        "IsDisabled": "No"
    }
]
```

Create a Worktype

Creates a **Worktype** in your **TimeLive** account and returns the Response Message.

```
POST /API/Worktypes
```

Object Property	Required	Object Type	Description	Max Character Length
AccountWorkTypeCode	**Yes**	**String**	**Code of Worktype**	**10**
AccountWorkType	**Yes**	**String**	**Name of Worktype**	**10**

Example Request

```
curl "https://demo.livetecs.com/api/Worktypes" \
      -H "APIKey: {Your API Key}" \
```

```
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'AccountWorkTypeCode': 'dm',''AccountWorkType': 'demo'}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Work Type Added Successfully"
    }
]
```

Update a Worktype

Update an **Worktype** in your **TimeLive** account and returns the Response Message.

```
POST /API/Worktypes/{Worktypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Worktypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'IsDisabled': true}"
```

Example Response

```
[
    {
        "id": "0001",
        "message": "Work Type Updated Successfully"
    }
]
```

Delete a Worktype

Delete a **Worktype**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/Worktypes/{Worktypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/Worktypes/0001" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "id": "0001",
        "message": "Work Type Deleted Successfully"
    }
]
```

15.3.27. Working Days

The Working Day Object

AccountWorkingDayType	**Yes**	**String**	**Name of Working Days**	**25**
HoursPerDay	**Yes**	**Integer**	**Hours Per Day of Working Days**	**2**
MinimumHoursPerDay	**Yes**	**Integer**	**Minimum Hours Per Day of Working Days**	**2**
MaximumHoursPerDay	**Yes**	**Integer**	**Maximum Hours Per Day of Working Days**	**2**
MinimumHoursPerWeek	**Yes**	**Integer**	**Minimum Hours Per Week of Working Days**	**2**
MaximumHoursPerWeek	**Yes**	**Integer**	**Maximum Hours Per Week of Working Days**	**2**
WeekStartsOn	**Yes**	**String**	**Week Start day of Working Days**	**9**
WorkingDays	**Yes**	**String**	**Working Days of Working Days** [coma seperated values (e.g mon,tue,wed)]	**70**
AccountWorkingDayTypeId	No	Guid	Id of Working Days	32
IsDisabled	No	Boolean	Disable Status of Working Days	1
EnableBalanceValidationForTimeOff	No	Boolean	Balance Validation For TimeOff Status of Working Days	1
ShowClockStartEndEmployee	No	Boolean	Show Clock Start / End for Employees of Working Days	1
AccountTimesheetPeriodTypeId	No	Guid	Timesheet PeriodType Id of Working Days	32
TimesheetOverdueAfterDays	No	Integer	Timesheet Overdue After Days of Working Days	2
MinimumPercentagePerDay	No	Integer	Minimum Percentage Per Day of Working Days	3
MaximumPercentagePerDay	No	Integer	Maximum Percentage Per Day of Working Days	3
MinimumPercentagePerWeek	No	Integer	Minimum Percentage Per Week of Working Days	3
MaximumPercentagePerWeek	No	Integer	Maximum Percentage Per Week of Working Days	3
LockPreviousTimesheetPeriods	No	Integer	Lock Number of Previous Timesheet Periods of Working Days	2
LockFutureTimsheetPeriods	No	Integer	Lock Number of Future Timsheet Periods of Working Days	2
LockPreviousNextMonthTimesheetOn	No	Integer	Disable edit Timesheet on Specific date of the month of Working Days	2
LockAllPeriodExceptPrevious	No	Integer	Lock All Timesheet Periods Except Previous of Working Days	2

LockAllPeriodExceptNext	No	Integer	Lock All Timesheet Periods Except Next Period of Working Days	2
LockAllPreviousTimesheets	No	Boolean	Allow Timesheet Entry in previous Period of Working Days	1
LockAllFutureTimesheets	No	Boolean	Allow Timesheet Entry in Future Periods of Working Days	1

List All Working Days

Returns a list of your all **Working Days** in your **TimeLive** account.

```
GET /API/WorkingDays
```

Example Request

```
curl "https://demo.livetecs.com/api/WorkingDays" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
     {
        "AccountWorkingDayTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountId": 7606,
        "AccountWorkingDayType": "Standard",
        "HoursPerDay": 8,
        "CreatedByEmployeeId": 0,
        "CreatedOn": "2017-11-01T10:55:31.523+00:00",
        "ModifiedByEmployeeId": 0,
        "ModifiedOn": "2017-11-01T10:55:31.523+00:00",
        "MasterWorkingDayTypeId": "00000000-0000-0000-0000-000000000000",
        "MinimumHoursPerDay": 0,
        "MaximumHoursPerDay": 24,
        "MinimumHoursPerWeek": 0,
        "MaximumHoursPerWeek": 60,
        "AccountTimesheetPeriodTypeId": "00000000-0000-0000-0000-000000000000",
        "TimesheetOverdueAfterDays": null,
        "MinimumPercentagePerDay": 0,
        "MaximumPercentagePerDay": 100,
        "MinimumPercentagePerWeek": 0,
        "MaximumPercentagePerWeek": 500,
```

```
        "LockPreviousTimesheetPeriods": null,
        "LockFutureTimsheetPeriods": null,
        "LockPreviousNextMonthTimesheetOn": null,
        "LockAllPeriodExceptPrevious": null,
        "LockAllPeriodExceptNext": null,
        "WeekStartsOn": "Monday",
        "WorkingDays": "Monday , Tuesday , Wednesday , Friday , Saturday",
        "IsDisabled": "No",
        "LockAllPreviousTimesheets": null,
        "LockAllFutureTimesheets": null,
        "EnableBalanceValidationForTimeOff": "Yes",
        "ShowClockStartEndEmployee": null
    },
    {
        "AccountWorkingDayTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountId": 7606,
        "AccountWorkingDayType": "demo",
        "HoursPerDay": 8,
        "CreatedByEmployeeId": 17739,
        "CreatedOn": "2018-06-01T16:27:06.043+00:00",
        "ModifiedByEmployeeId": 17739,
        "ModifiedOn": "2018-06-01T16:31:15.98+00:00",
        "MasterWorkingDayTypeId": null,
        "MinimumHoursPerDay": 5,
        "MaximumHoursPerDay": 8,
        "MinimumHoursPerWeek": 25,
        "MaximumHoursPerWeek": 40,
        "AccountTimesheetPeriodTypeId": "00000000-0000-0000-0000-000000000000",
        "TimesheetOverdueAfterDays": 0,
        "MinimumPercentagePerDay": 0,
        "MaximumPercentagePerDay": 100,
        "MinimumPercentagePerWeek": 0,
        "MaximumPercentagePerWeek": 500,
        "LockPreviousTimesheetPeriods": 0,
        "LockFutureTimsheetPeriods": 0,
        "LockPreviousNextMonthTimesheetOn": 0,
        "LockAllPeriodExceptPrevious": 0,
        "LockAllPeriodExceptNext": 0,
        "WeekStartsOn": "Monday",
        "WorkingDays": "Monday , Tuesday , Wednesday",
        "IsDisabled": "No",
        "LockAllPreviousTimesheets": "No",
        "LockAllFutureTimesheets": "No",
        "EnableBalanceValidationForTimeOff": "Yes",
        "ShowClockStartEndEmployee": "No"
    }
]
```

Retrieve a Working Day

Retrieves the Working Day with the given ID. Returns a Working Day object and a 200 OK response code if a valid identifier was provided.

```
GET /API/WorkingDays/{TimeOffTypes_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/WorkingDays/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "Content-Type: application/json" \
      -H "User-Agent: MyApp (yourname@example.com)"
```

Example Response


```
[
     {
        "AccountWorkingDayTypeId": "00000000-0000-0000-0000-000000000000",
        "AccountId": 7606,
        "AccountWorkingDayType": "Standard",
        "HoursPerDay": 8,
        "CreatedByEmployeeId": 0,
        "CreatedOn": "2017-11-01T10:55:31.523+00:00",
        "ModifiedByEmployeeId": 0,
        "ModifiedOn": "2017-11-01T10:55:31.523+00:00",
        "MasterWorkingDayTypeId": "00000000-0000-0000-0000-000000000000",
        "MinimumHoursPerDay": 0,
        "MaximumHoursPerDay": 24,
        "MinimumHoursPerWeek": 0,
        "MaximumHoursPerWeek": 60,
        "AccountTimesheetPeriodTypeId": "00000000-0000-0000-0000-000000000000",
        "TimesheetOverdueAfterDays": null,
        "MinimumPercentagePerDay": 0,
        "MaximumPercentagePerDay": 100,
        "MinimumPercentagePerWeek": 0,
        "MaximumPercentagePerWeek": 500,
        "LockPreviousTimesheetPeriods": null,
        "LockFutureTimsheetPeriods": null,
        "LockPreviousNextMonthTimesheetOn": null,
        "LockAllPeriodExceptPrevious": null,
        "LockAllPeriodExceptNext": null,
        "WeekStartsOn": "Monday",
        "WorkingDays": "Monday , Tuesday , Wednesday , Friday , Saturday",
```

```
        "IsDisabled": "No",
        "LockAllPreviousTimesheets": null,
        "LockAllFutureTimesheets": null,
        "EnableBalanceValidationForTimeOff": "Yes",
        "ShowClockStartEndEmployee": null
    }
]
```

Create a Working Day

Creates a **Working Day** in your **TimeLive** account and returns the Response Message.

```
POST /API/WorkingDays
```

Object Property	Required	Object Type	Description	Max Character Length
AccountWorkingDayType	**Yes**	**String**	**Name of Working Days**	**25**
HoursPerDay	**Yes**	**Integer**	**Hours Per Day of Working Days**	**2**
MinimumHoursPerDay	**Yes**	**Integer**	**Minimum Hours Per Day of Working Days**	**2**
MaximumHoursPerDay	**Yes**	**Integer**	**Maximum Hours Per Day of Working Days**	**2**
MinimumHoursPerWeek	**Yes**	**Integer**	**Minimum Hours Per Week of Working Days**	**2**
MaximumHoursPerWeek	**Yes**	**Integer**	**Maximum Hours Per Week of Working Days**	**2**
WeekStartsOn	**Yes**	**String**	**Week Start day of Working Days**	**9**
WorkingDays	**Yes**	**String**	**Working Days of Working Days** [coma seperated values (e.g mon,tue,wed)]	**70**

Example Request

```
curl "https://demo.livetecs.com/api/Working Days" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d
"{'WeekStartsOn':'Monday','WorkingDays':'mon,tue','AccountWorkingDayType':'Demo','MinimumI
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "WorkingDayType Added Successfully"
    }
]
```

Update a Working Day

Update a **Working Day** in your **TimeLive** account and returns the Response Message.

```
POST /API/WorkingDays/{Working Days_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/WorkingDays/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X POST \
      -H "Content-Type: application/json" \
      -d "{'IsDisabled': true}"
```

Example Response

```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "WorkingDayType Updated Successfully"
    }
]
```

Delete a Working Day

Delete a **Working Day**. Returns a 200 OK response code if the call succeeded.

```
DELETE /API/WorkingDays/{WorkingDays_ID}
```

Example Request

```
curl "https://demo.livetecs.com/api/WorkingDays/
00000000-0000-0000-0000-000000000000" \
      -H "APIKey: {Your API Key}" \
      -H "AuthToken: {Your Authentication Token}" \
      -H "User-Agent: MyApp (yourname@example.com)" \
      -X DELETE \
      -H "Content-Type: application/json"
```

Example Response


```
[
    {
        "id": "00000000-0000-0000-0000-000000000000",
        "message": "WorkingDayType Deleted Successfully"
    }
]
```

15.4. Sample Codes

15.4.1. TimeLive API Sample Code – C#

```
using System;
namespace TimeLiveAPISample
{
    class Program
    {
        static void Main(string[] args)
        {
            const string URL = "http://demo.livetecs.com/api/departments";
            try
            {
                var webRequest = System.Net.WebRequest.CreateHttp(URL);
                if (webRequest != null)
                {
                    webRequest.Method = "GET";
                    webRequest.UserAgent = "C# TimeLive API Sample";
                    webRequest.Headers.Add("AuthToken",
<TIMELIVE_AUTHENTICATION_TOKEN>);
                    webRequest.Headers.Add("APIKey", <TIMELIVE_API_KEY>);
                    using (System.IO.Stream s =
webRequest.GetResponse().GetResponseStream())
                    {
                        using (System.IO.StreamReader sr = new
System.IO.StreamReader(s))
                        {
                            var jsonResponse = sr.ReadToEnd();
                            Console.WriteLine(jsonResponse);
                        }
                    }
                }
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.ToString());
            }
        }
    }
}
```

16. TimeLive FAQ

16.1. Disabling Pop-Up Blockers

To ensure full functionality of the application, you should disable pop-up blockers for the TimeLive X website. To allow pop-ups for TimeLive X website:

Microsoft Internet Explorer:

1. Select Tools > Pop-up Blocker > Pop-up Blocker Settings.
2. Enter the TimeLive X URL in the field provided.
3. Select the Add button, and then the Close button, and then the OK button.

Mozilla Firefox:

1. Select Tools > Options.
2. Select the Content icon.
3. Select the Exceptions button next to the Block pop-up windows checkbox.
4. Enter the TimeLive X URL in the field provided.
5. Select the Allow button, and then the Close button, and then the OK button

Other pop-up blocking application

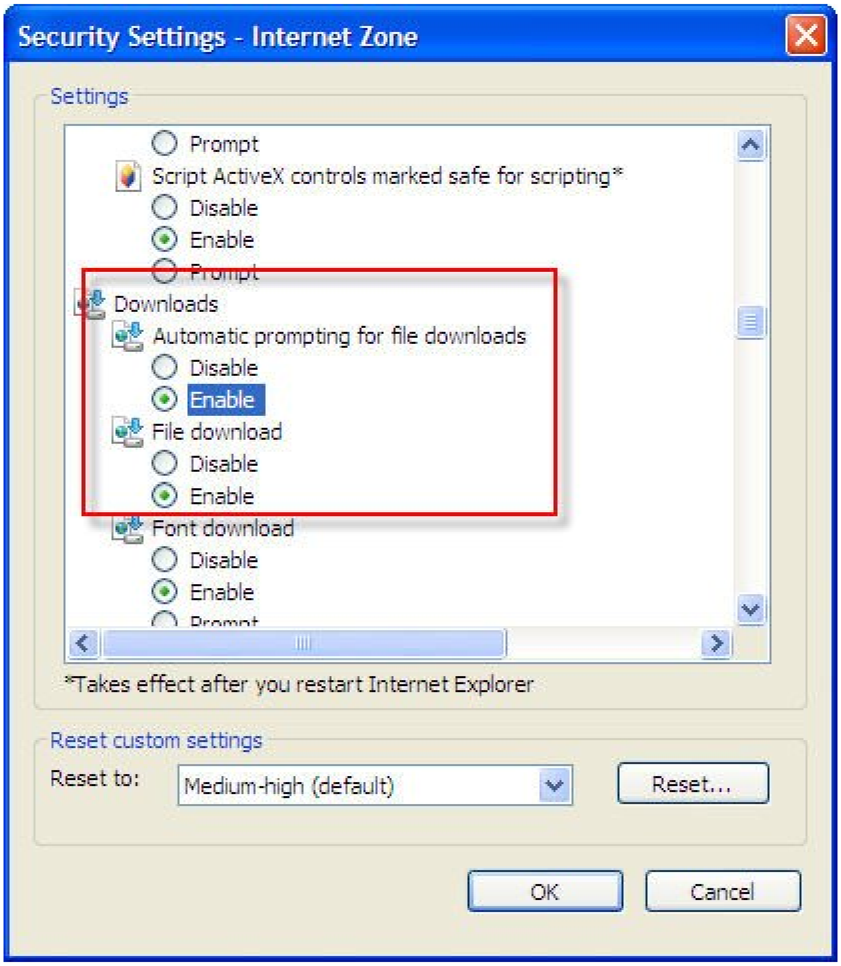
1. Refer to that application’s documentation for information on how to disable pop-up blocking for a specific site.

16.2. Settings required on Internet Explorer 7

Because of recent security changes in Internet Explorer 7, these settings are required for exporting reports in IE7.

Instructions required to be done in IE7:

- From the **[Tools]** menu – top right on IE7 – select **[Internet Options]**.
- Select the **[Security]** Tab.
- Select the **[Internet Zone]** (click the globe on the left).
- Hit **[Custom Level]** button (bottom of dialogue box)
- Scroll down to the **[Downloads]** section and find the **[Automatic prompting -for file downloads]** setting and set it to **[Enable]**, then click on **[OK]**.



16.3. Frequently Logout Issue By Users

SYMPTOMS:

TimeLive X users can experience frequently log out.

CAUSE:

There are many possible causes for this problem. This article addresses one possible cause: session data that is lost because of antivirus software activity.

This problem can occur when antivirus software scans the Web application files. During the scanning process, the antivirus software may mark the Global.asax, the Web.config, and/or the Machine.config files as modified. This modification prompts the Microsoft .NET Framework restart the Web application. If the session data is stored in-process, all session data is lost whenever a Web application is restarted.

RESOLUTION:

To resolve this problem, use the following method:

- Configure your antivirus software so that it does not scan .asax and .config files.
  http://kbalertz.com/316148/Session-Variables-Intermittently-Applications.aspx

16.4. TimeLive Resource File Updates

TimeLive use .Net Resource files for showing international languages based contents in TimeLive. In case of any issue in translation, a customer can correct that resource file and can send it to us for merging that resource file in TimeLive released version.

To Edit Resource file:

1. Send an email to support@livetecs.com with a request for a resource file in your required language.
2. Once received resource file, open that resource file in ResourceEditor tool using **[Browse]** button. ResourceEditor can be download from https://www.livetecs.com/release/ResourceEditor.zip.
3. Enter comments “Updated by Your name” in all translation updated by you.
4. Click on **[Save]** in ResourceEditor once done all changes.
5. Send back that resource file back to support@livetecs.com.

16.5. System Defined Roles

In Timelive, there are some System Defined Roles which cannot be assigned to an employee manually. In Employee form, such system roles do not appear in the [Role] list. The Detail and workflow of these roles are defined below.

Employee Manager

In employee form, there is an option to select specific employee manager for that particular employee. Once the employee has been specified as an employee manager of any user, the system will automatically grant the permissions and privileges of Role “Employee Manager” to that particular employee.

Assigning Employee Manager to the existing Employee:

1. Click on **Employees** link at the top-right side of the application.
2. Click on **Options** gear icon against the employee to whom you want to assign Employee Manager.
3. Click **View Employee** option
4. Click **Options** icon and select **Edit Employee** option
5. Click **Other** Tab and select the employee name from **Employee Manager**
6. Click Update.

Team Lead

In project form, there is an option to select specific team lead for that particular project. Once the employee has been specified as the team lead of any project, the system automatically grants the permissions and privileges of Role “Team Lead” to that particular employee.

Steps to assign Team Lead:

1. Click on **Projects** link at the top-right side of the application.
2. Scroll to the project in which you want to set **Team Lead** and click **Options** gear icon then click **View Project** option.
3. Click **Options** and click **Edit Project** option.
4. On the **Basic** tab, under **Team Lead** field, select the employee name to which you want to assign **Team Lead Role**.
5. Click **Update** button.

Project Manager

In Project form there is an option to select the project manager for that particular project. Once the employee has been specified as the project manager of any project, the system automatically grants the permissions and privileges of Role “Project Manager” to that particular employee.

Steps to assign Project Manager:

1. Click on **Projects** link at the top-right side of the application.
2. Scroll to the project in which you want to set **Project Manager** and click **Options** gear icon then click **View Project** option.
3. Click **Options** and click **Edit Project** option.
4. On the **Basic** tab, under **Project Manager** field, select the employee name to which you want to assign **Project Manager Role**.
5. Click **Update** button.

Time Entry Approver

In Project form, there is an option to select the approval type of timesheet for that particular project. Once the approval type of timesheet is selected for any project, all of the employees defined in that particular approval type (approval flow) become time entry approver for that project, and the system automatically grants the permissions and privileges of Role “Time Entry Approver” to that particular employees. It can be system defined role or custom approval type created from **Admin Options > Approvals**.

Steps to select Timesheet Approval Type:

1. Click on **Projects** link at the top-right side of the application.
2. Scroll to the project in which you want to set **Timesheet Approval Type** and click **Options** gear icon then click **View Project** option.
3. Click **Options** and click **Edit Project** option.
4. On the **Approval** tab, under **Timesheet Approval Type** field, select the role/approval type to which you want to assign **Timesheet Approval Type**.
5. Click **Update** button.

Expense Entry Approver

In Project form, there is an option to select the approval type of expense sheet for that particular project. Once the approval type of expense sheet is selected for any project, all of the employees defined in that particular approval type (approval flow) become expense entry approver for that project, and the system automatically grants the permissions and privileges of Role “Expense Entry Approver” to that particular employees. It can be system defined role or custom approval type created from **Admin Options > Approvals**.

Steps to select Expense Sheet Approval Type:

1. Click on **Projects** link at the top-right side of the application.
2. Scroll to the project in which you want to set **Expense Approval Type** and click **Options** gear icon then click **View Project** option.
3. Click **Options** and click **Edit Project** option.
4. On the **Approval** tab, under **Expense Approval Type** field, select the role/approval type to which you want to assign **Expense Approval Type**.
5. Click **Update** button.

External User

In TimeLive, external users are not considered as employees hence, doesn’t appear in employee list as well. External users can be added/edited from **External Users** available in **Admin Options**. The system automatically grants the permissions and privileges of Role **External User** to all the external users.

16.6. Timesheet Copy, Activities Copy, Export/Import Offline Timesheet

Copy Timesheet and Activity:

There are many entries which are generic and users do these entries or similar ones on daily basis. Selecting same Clients, Projects, Tasks and hours does not take much time if it is only for one time but on daily basis, it could be frustrating. TimeLive also thinks the same. With our Copy Timesheet/activities feature, user can copy same Timesheet or the activities and carry it in the current period.

Copy Timesheet from existing entries:

Below are the steps to **Copy Timesheet** from any existing Time entry period (NOTE: This will copy all the selection of Client, Project, Tasks, work type, custom fields, and hours, description to the current period),

1. Login TimeLive and navigate to My Timesheet.
2. Navigate to fresh Timesheet period in which you want to copy the Timesheet.
3. Scroll to the bottom of the page and select the period starting date from which you want to copy the Timesheet.
4. Click Copy Timesheet.

Copy Activity from existing entries:

Below are the steps to **Copy Activity** from any existing Time entry period to the current period (NOTE: This will copy all the selection of Client, Project, Tasks, work type, custom fields but not the hours, description, and Timesheet custom Fields),

1. Login TimeLive and navigate to My Timesheet.
2. Navigate to fresh Timesheet period in which you want to copy the **Activity**.
3. Scroll to the bottom of the page and select the period starting date from which you want to copy the Activity.
4. Click Copy Activity.

Import/Export Offline Timesheet:

Sometimes due to internet issue, users cannot fill their timesheet but with this option, users can keep already filled timesheet in a saved file and upload it at their convenience.

Exporting Offline Timesheet:

1. Login TimeLive and navigate to My Timesheet.
2. Navigate to the filled Timesheet period which you want to Export.
3. Scroll to the bottom of the page and click on the button **Export Offline Timesheet**.
4. Save the file in a secure location for future extraction(It is recommended that you save the file with some reference name for easy understanding. Also you can save many other for your convenience).

Importing Offline Timesheet:

1. Login TimeLive and navigate to My Timesheet.
2. Navigate to the empty Timesheet period in which you want to import exported Timesheet.
3. Click on the button **Import Offline Timesheet** and select the file which you want to import.

16.7. Attachment in Timesheet/TimeOff Request

Some details are required for Timesheet Entry or Time Off which is not sufficient only with Description and some documentation are required for approver to view and approve accordingly, specially leaves and other.

In TimeLive, you can attach any document as per your company requirement with your Timesheet or Time Off Request through below steps,

Attachment in Timesheet Period:

1. Navigate to **My Timesheet**.
2. Click on **Timesheet Periods** Icon.
3. Click on the **Options** gear icon in the period you want to attach the document.
4. Click **View Attachment** option.
5. Fill the name of the attachment(It could be a reference for your understanding).
6. Click on the **Choose File** button to upload the file.
7. Click **Upload** button.

Attachment in Time Off Request:

1. Navigate to **My Time Off**.
2. Click on **Add Time Off Request** Icon.
3. After filling the Time Off Request Basic Information, Navigate to Attachment Tab.
4. Click on the **DropFile** Box.
5. Click **Add Time Off Request** Button

16.8. Licensing DLL files (32 bit and 64bit)

TimeLive licensing dll files are operating system specific and different for 32bit and 64bit operating system. TimeLive installer automatically copy platform specific dlls while installing TimeLive.

In case of manual installation of TimeLive, an administrator should copy platform specific license file in TimeLive\bin folder. An administrator can download the platform-specific license file from

32bit license dll file
https://www.livetecs.com/Release/IsLicense4032bit.zip
64bit License dll file
https://www.livetecs.com/Release/IsLicense4064bit.zip