# Redmine – ERPmine User Guide

**Version 4.5**

Feb 09, 2022

# Revision History

| Date | Version | Description | Author |
|---|---|---|---|
| 09/06/2017 | 3.0 | Inventory module is introduced | R.Dhineshkumar |
| 10/28/2017 | 3.1 | Asset module is introduced | R.Dhineshkumar |
| 01/31/2018 | 3.2 | Schedule Shifts | R.Dhineshkumar |
| 12/21/2018 | 3.3 | Supervisor feature | M.Karthick |
| 01/08/2019 | 3.4 | Dashboards is introduced | T.Arun |
| 03/11/2019 | 3.5 | Survey module is introduced | T.Arun |
| 05/10/2019 | 3.6 | Entity Surveys | T.Arun |
| 07/03/2019 | 3.7 | Employee Performance Review | T.Arun |
| 11/01/2019 | 3.8 | Leave Request | T.Arun |
| 12/27/2019 | 3.9 | Income Tax Calculations | R.Amutha |
| 06/07/2020 | 4.0 | Record Geo Location | R.Amutha |
| 11/03/2020 | 4.1 | Individual approvers per project for timesheets | R.Amutha |
| 03/17/2021 | 4.2 | Referral | R.Amutha |
| 06/30/2021 | 4.3 | Generate invoice on selected hours | R.Amutha |
| 08/18/2021 | 4.3.1 | Serial number | R.Amutha |
| 10/14/2021 | 4.4 | Delivery | R.Amutha |
| 02/09/2022 | 4.5 | Cash Flow Report | C.Dhanalakshmi |

# Table of Contents

# 1. Introduction


ERPmine is a free open source ERP for Service Industries. It runs as a plugin within Redmine, it has the following modules:

## 1.1 Installation

Unpack the zip file to the plugins folder of Redmine. Starting from version 1.2, it requires db migration. So run the following command for db migration

rake redmine:plugins:migrate NAME=redmine_wktime RAILS_ENV=production

When uninstalling the plugin, be sure to remove the db changes by running

rake redmine:plugins:migrate NAME=redmine_wktime VERSION=0 RAILS_ENV=production

# 2. Time & Expense

## 2.1 Timesheet

Timesheet is used to log the spent time for projects by their members for a week. The member should have log time permission to fill out the timesheet.

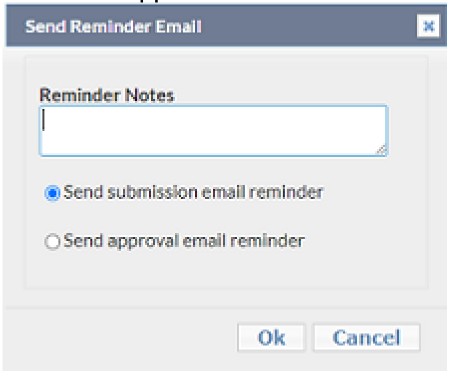
### *2.1.1 Timesheet List*

The list page by default displays the timesheets for the current month. Date range filters can be applied to the list page. The users with “Edit time logs” permission will have additional filters; project, member, group and status dropdowns. The list page displays the list of timesheets satisfying the filter criteria. The project/group dropdown is used for filtering the member dropdown. User custom fields which are marked as filters can also be used to filter time/expense sheets. In the Supervisor based approval system the supervisor has me, my reporters and my direct reporters filters additionally.

**Note:** Users who have edit permissions on all projects within a timesheet will be allowed to delete a timesheet from this list page.

### *2.1.2 Send Reminder*

Manager and TE admins can send reminders to the user to submit and approve the time/expense sheet. The reminders will be sent based on the filtered time/expense sheet list. The submission reminder will be sent to the corresponding user and the approval reminder will be sent to the user's manager.

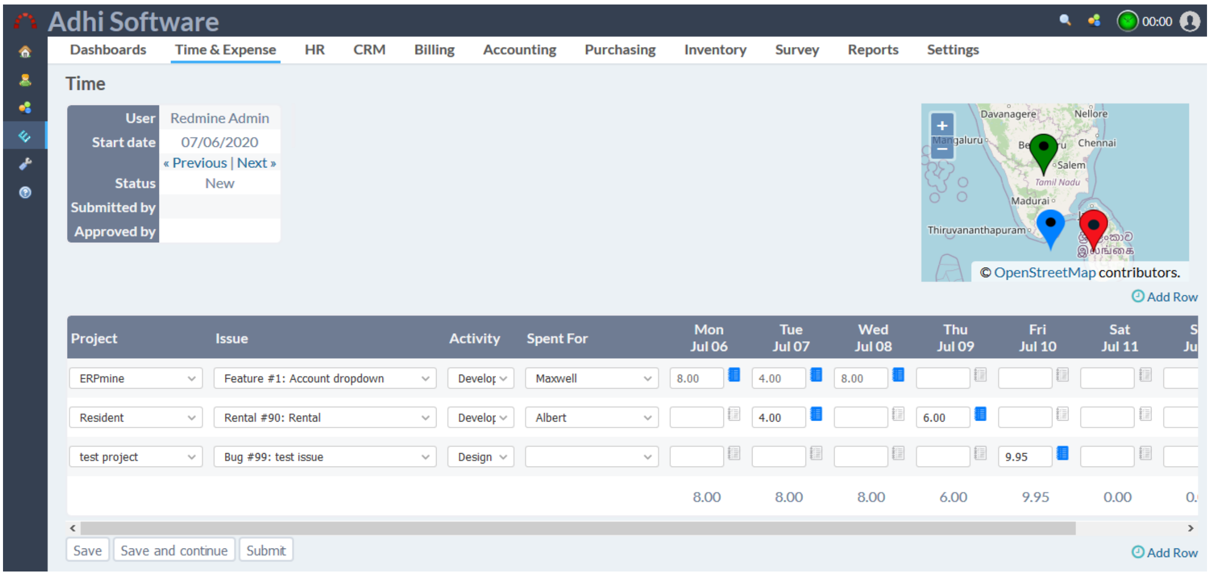



### *2.1.3 New timesheet*

Here are the steps to add a new timesheet.

- Click on the top menu "ERPmine",
- Select the tab "Time & Expense"
- Click on "New Timesheet" hyperlink
- Enter the start date of the week, sheet view for the new timesheet and if previous week template is needed check the previous week template checkbox, else uncheck it
- Click "Continue" to create timesheet.

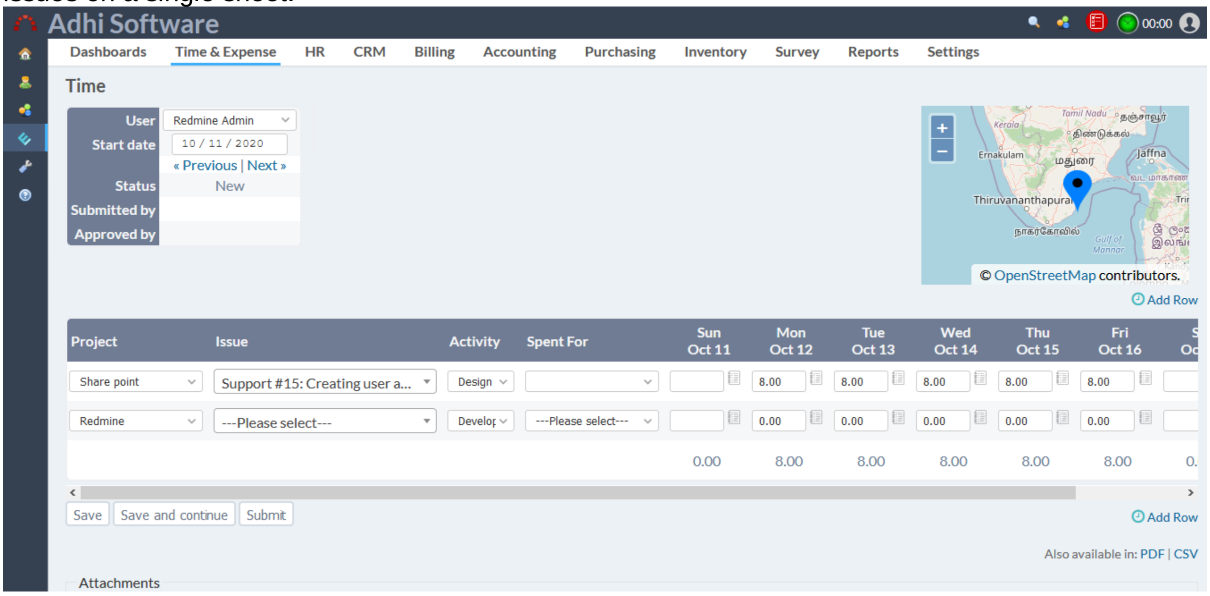
The screenshot represent the timesheet page. Using the "Add row" link, the user can enter new time entry. Choose the project, issue and activity from their respective dropdowns. Rows can be deleted using the delete icon at the end of the row. Enable the save geolocation, then we will view the current location in time sheet. we can add geo location. Map can be loaded even offline. The configuration timesheet Refer 15.2.8.




The Previous and Next link helps the user to navigate to the previous and next week timesheet.

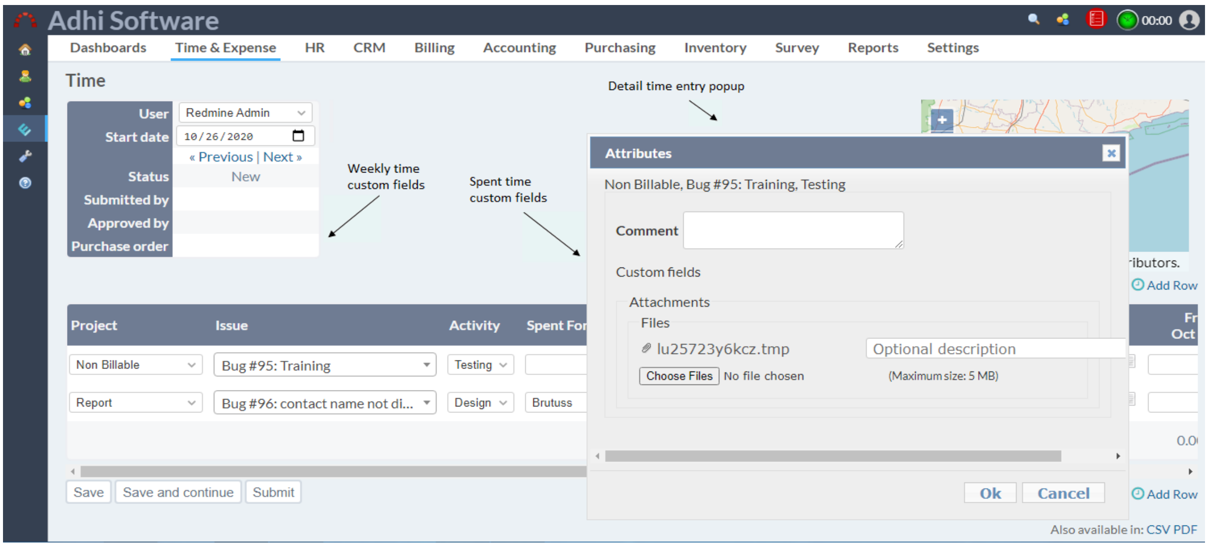
### 2.1.4 Issue view

Issue view helps us to enter the time entry for all assigned issues with respective spentfor. Chose issue view in new timesheet sheet view dropdown to get the issue view. It will list all the assigned issues on a single sheet.




### 2.1.5 Custom fields

Custom fields can be added at the Timesheet level and also at the individual spent time level.




Administrators can create custom fields by navigating to Administration ➔ Custom fields menu ➔ New custom field link. To create individual spent time custom fields choose the “Spent time” option and for timesheet custom fields choose the “Weekly Time” option from the Custom fields page.

#### 2.1.5.1 Spent time Custom Fields

The spent time custom fields are applicable to the individual cells within a timesheet; these are attributes of Spent Time entity. These custom fields can be configured for a row (discussed in section 6.5.7 and 6.5.8) instead of a cell and by default it will be shown in detail time entry popup (discussed in section 6.4).

#### 2.1.5.2 Timesheet Custom Fields

These custom fields are applicable to the whole timesheet and are created from the Weekly Time tab from the Custom fields page. For e.g., the Purchase Order field applies to the whole timesheet.

Refer the timesheet configuration 15.2

## 2.2 Expense Sheet

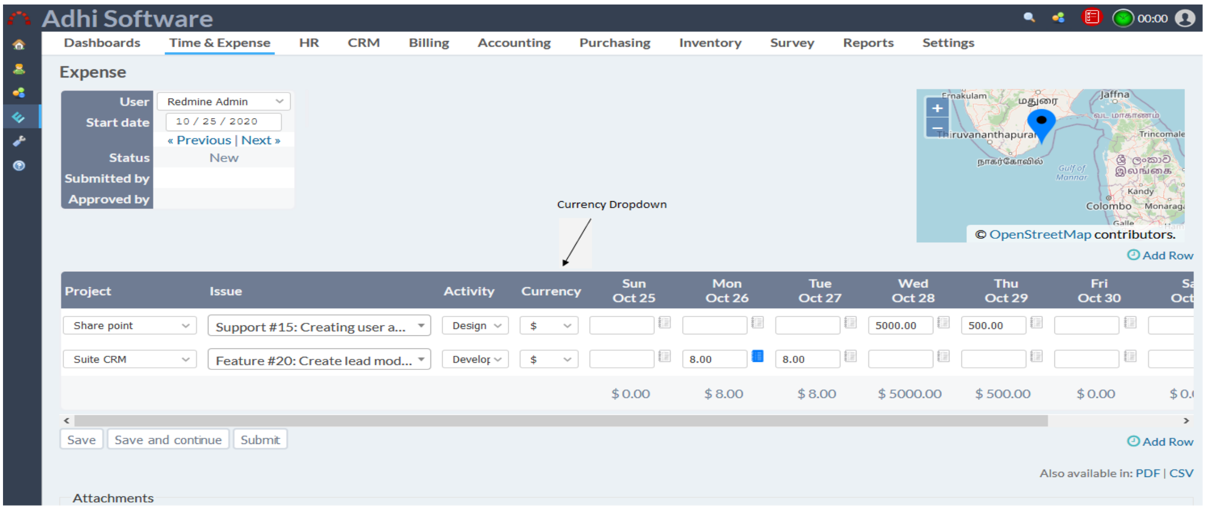
Expense sheet is used to fill out the expenses incurred on various projects. It works same as the timesheet. In addition to project, issue and activity fields, it has the currency field as well. Even though different currencies can be used for different rows, the currency from the last row is used for the day total and week total. Custom fields are not available for Expense Sheet.

### *2.2.1 Expense Sheet List*

This page list the expense sheets based on the filter criteria chosen on the page. This page is similar to the timesheet list page described in section 2.1.

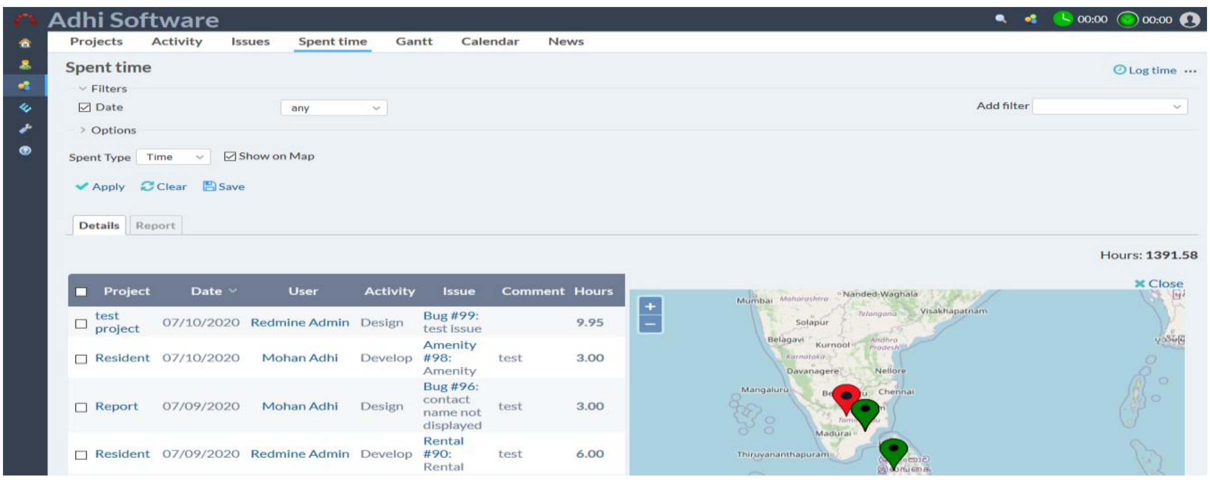
### *2.2.2 New Expense Sheet*

The steps for adding a new expense sheet is same as that of the timesheet. The configuration expense sheet Refer 15.2.18

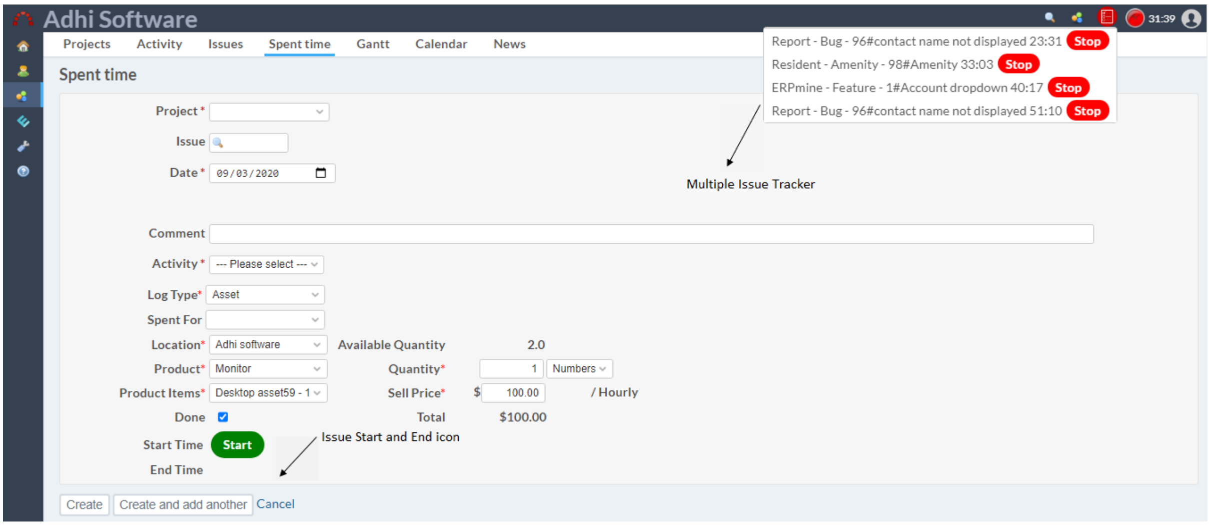



### *2.2.3 Detail and Report Page*

The expense detail and report page resembles the spent time detail and report page of redmine.

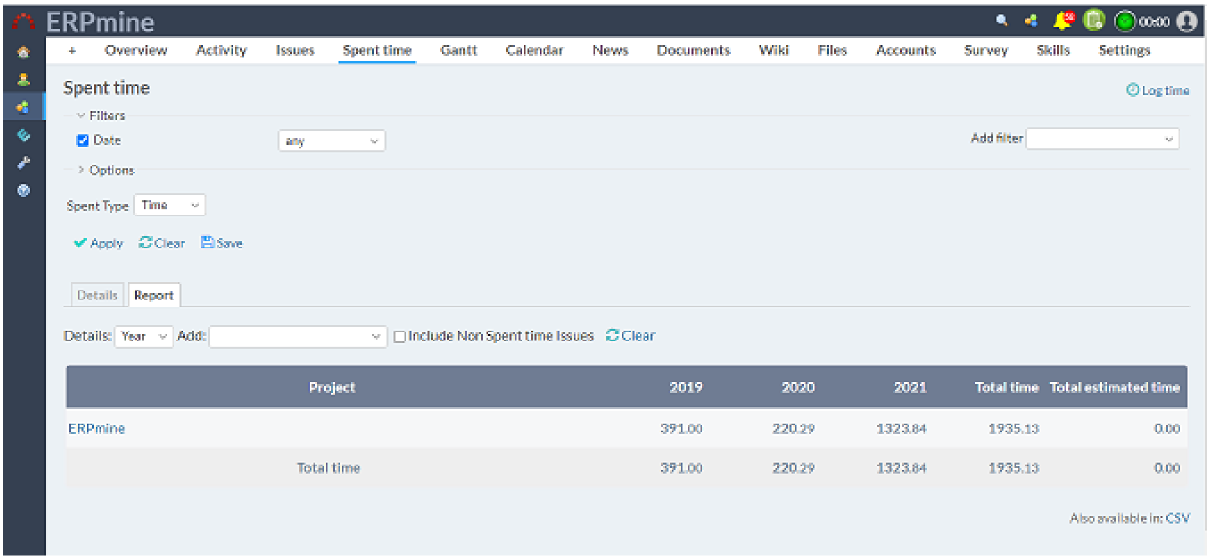



**Start/Stop issue** tracker helps to track the time for particular issue. we can add geo location.
In spent time, To Show the user location, Show on Map must be enabled.




The navigation for the expense Detail and Report page is Projects ➔ Edit project ➔ Expense Report. The project members with 'View Spent Time' permission have access to this page.

The details tab will list individual expense entries made for the project. We can edit and delete expense entries from here as well. The edit link will take the user to the corresponding expense sheet. Individual expense entries can be deleted from this page as well. The report tab can be used to summarize data at various levels; member, tracker etc. Report will show issues by enabling 'include non spent time Issue'. Added Total estimated time column in the spent time report.




## 2.3 Approval System

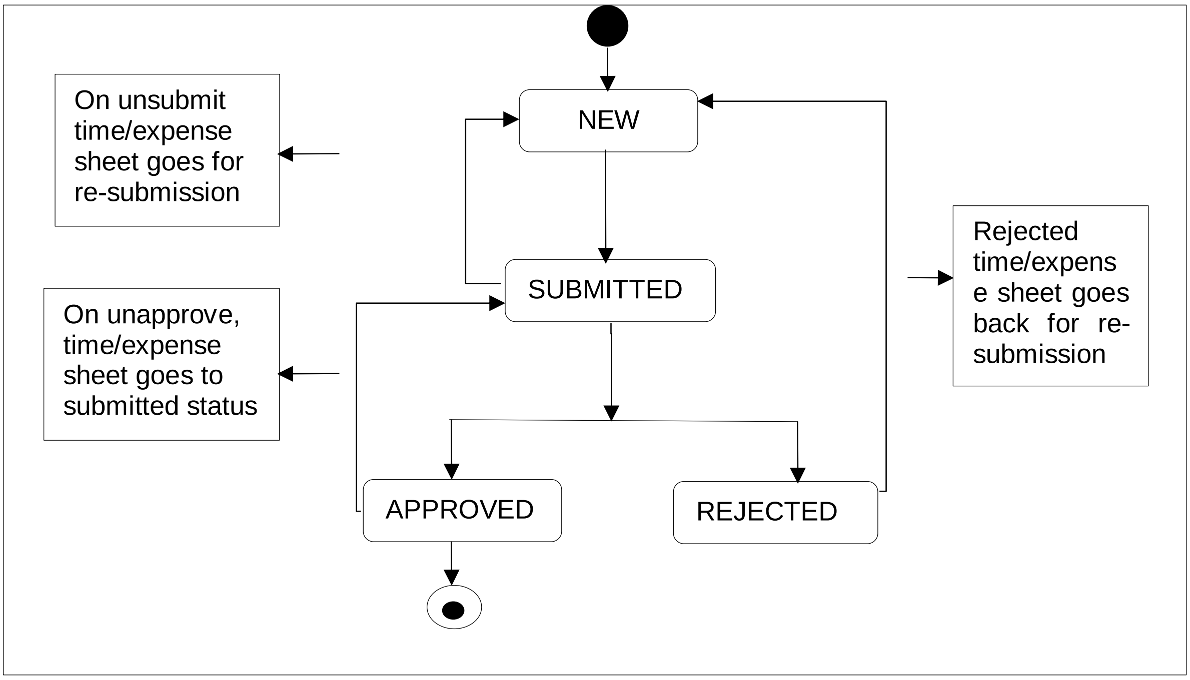
The ERPmine sheets can be approved / rejected by supervisors after it is submitted by Project Members.

### *2.3.1 Status*

The various statuses of the Time & Expense sheets are:

- New
- Submitted
- Approved
- Rejected.

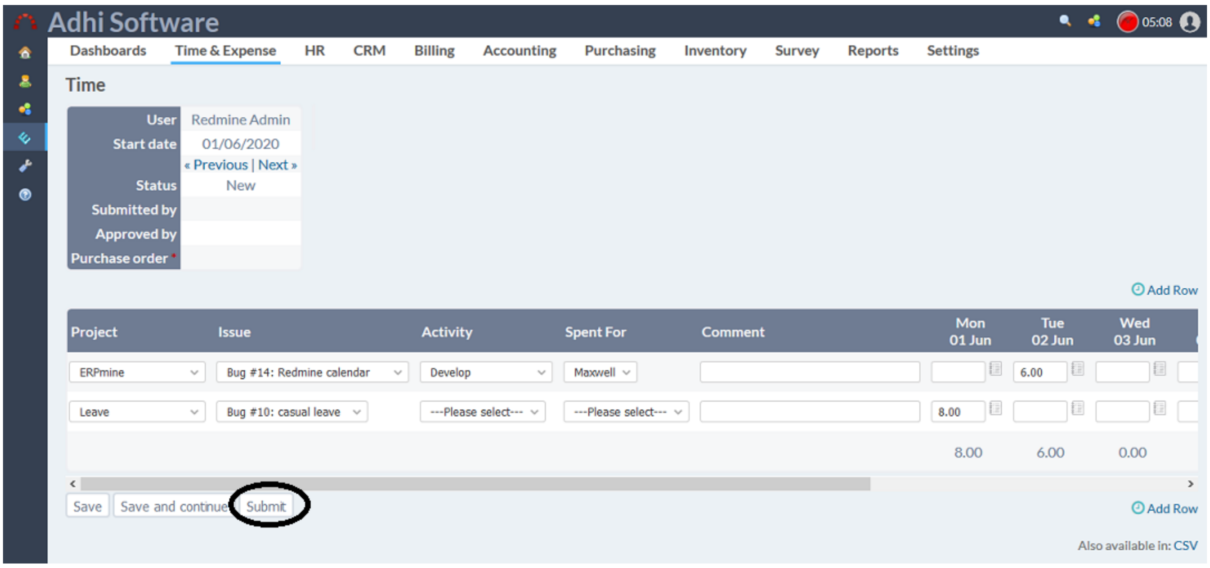
The time & expense sheet is in “New” status when it is created. Once the timesheet is submitted, it goes to the “Submitted” status. It goes to “Approved” status when the supervisor approves it. When the timesheet is rejected, it goes to the “Rejected” status and goes backs to the project member for resubmission. The Time & Expense Sheet returns to the “New” status when it gets un-submitted.




### 2.3.2 Workflow

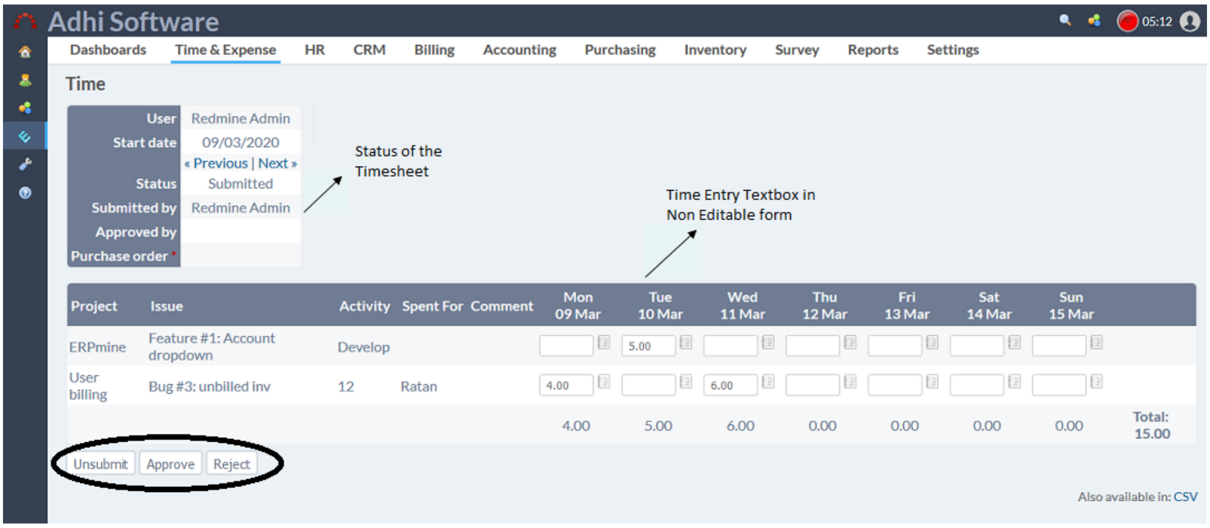
The workflow of Approval system is as follows.

i) The Time & Expense sheet has the Submit button in addition to the Save button.

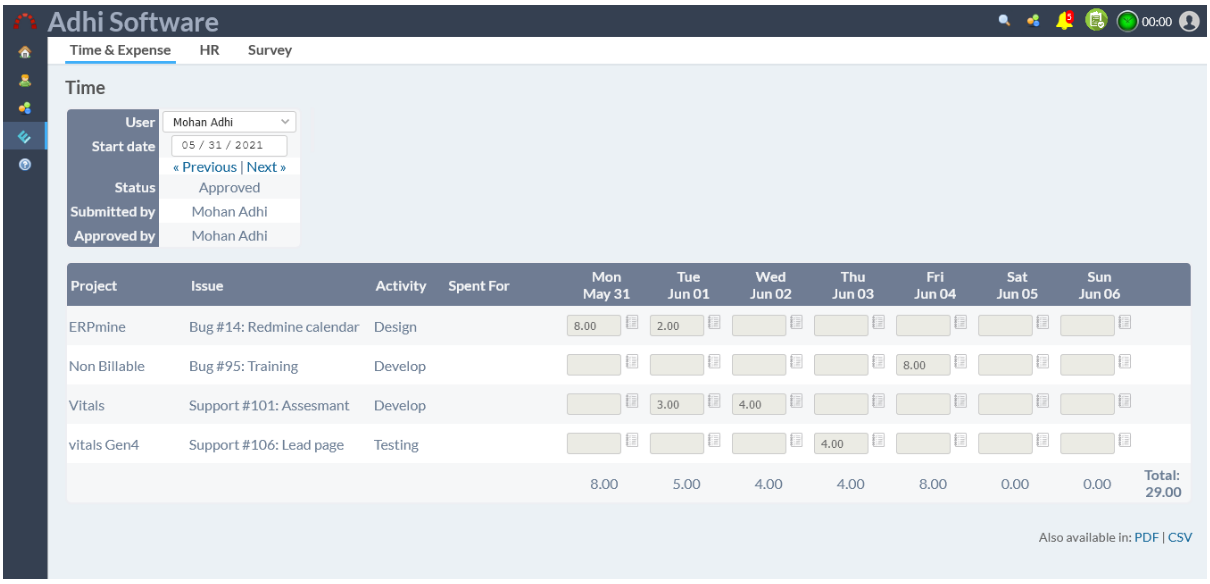



ii) A team member can save a time & expense sheet any number of times, but can only submit once.

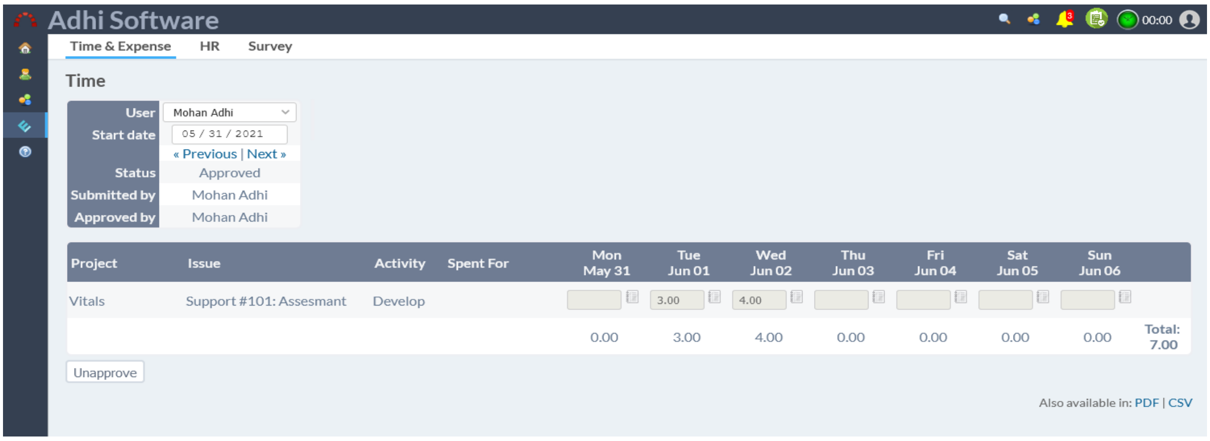
iii) Once a time & expense sheet is submitted, the user with “Approve time logs” permission can approve / reject it, the project member can also un-submit his time & expense sheet, if he wants to take back his submission.
iv) Once approved, the time & expense sheets can never be edited.
v) If the time & expense sheet is rejected, then it goes back to the project member for re-submission. An email about the rejection is sent to the Project Member.
vi) Once approved, a time & expense sheet can be unapproved; it sends the time & expense sheet back to submitted state.
vii) If the timesheet is not submitted before deadline, then an email notification will be sent to the project member.




## 2.3.3 Multiple Approvers




In case of multiple approvers for a time sheet, the approver can only approve the permitted time entries, please see the below two screenshots

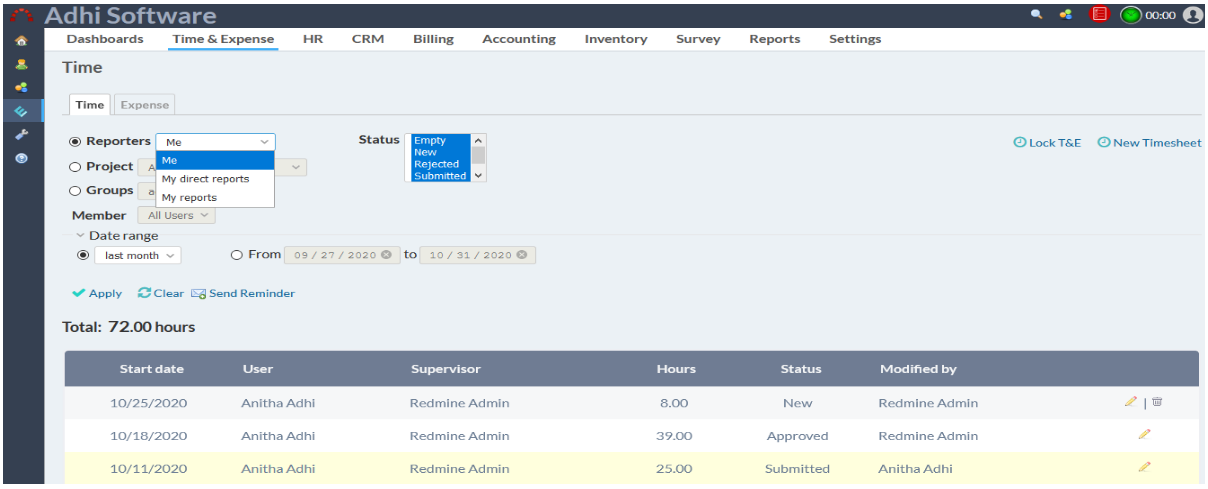



The approval system configuration Refer 15.1.2.20

## 2.4 Supervisor

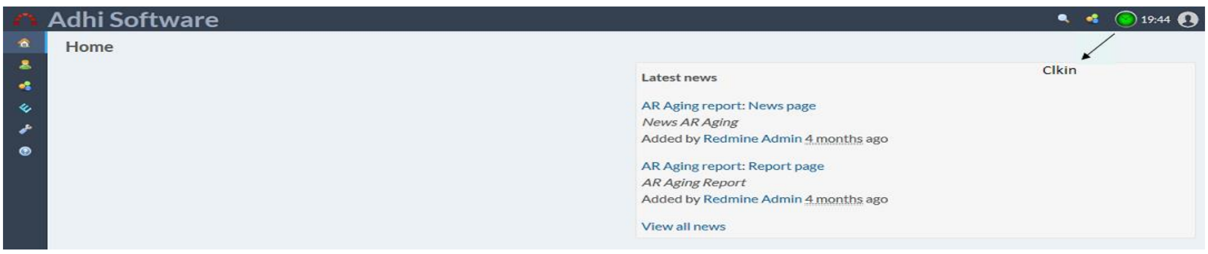
Supervisor is responsible for the productivity and actions of a small group of employees. Supervisor feature is enabled when "Use supervisor based approval system" is checked.

The supervisor can view their direct reporters and reporters using the filter. The supervisor also has the approval privilege. The configuration supervisor Refer 15.3

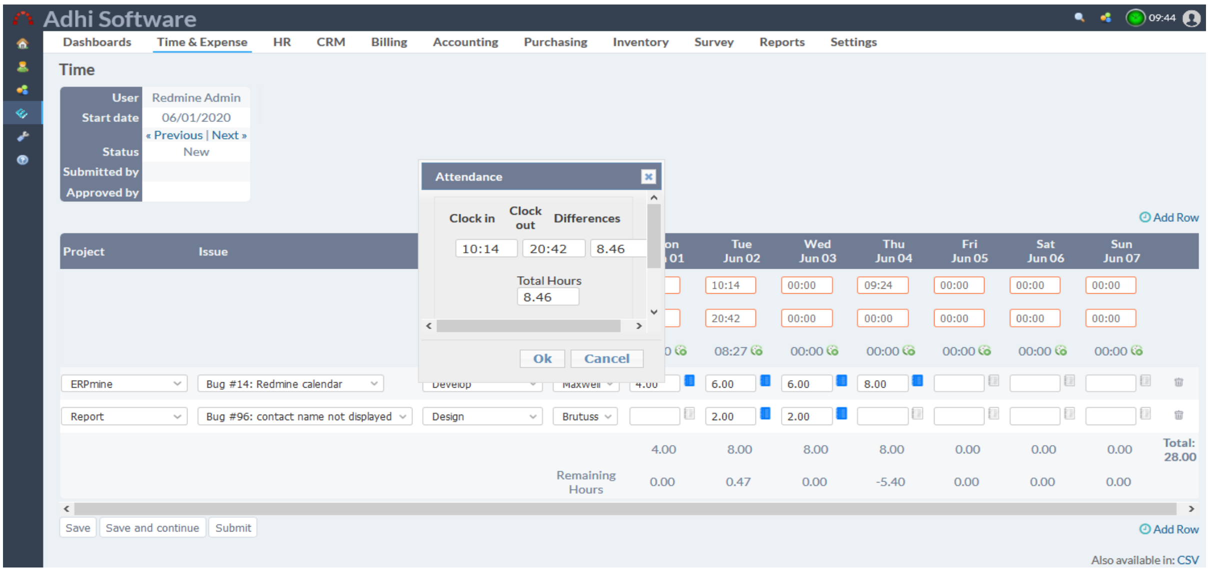



# 3. Attendance

Attendance module is enabled when "**Enable attendance module and Enable clock in / clock out**" is checked in the settings page. It allows the user to enable/disable attendance module and clock in and clock-out from the timesheet page. User is also allowed to do multiple clock-in and clock-out on the same day. The configuration attendance Refer 15.3

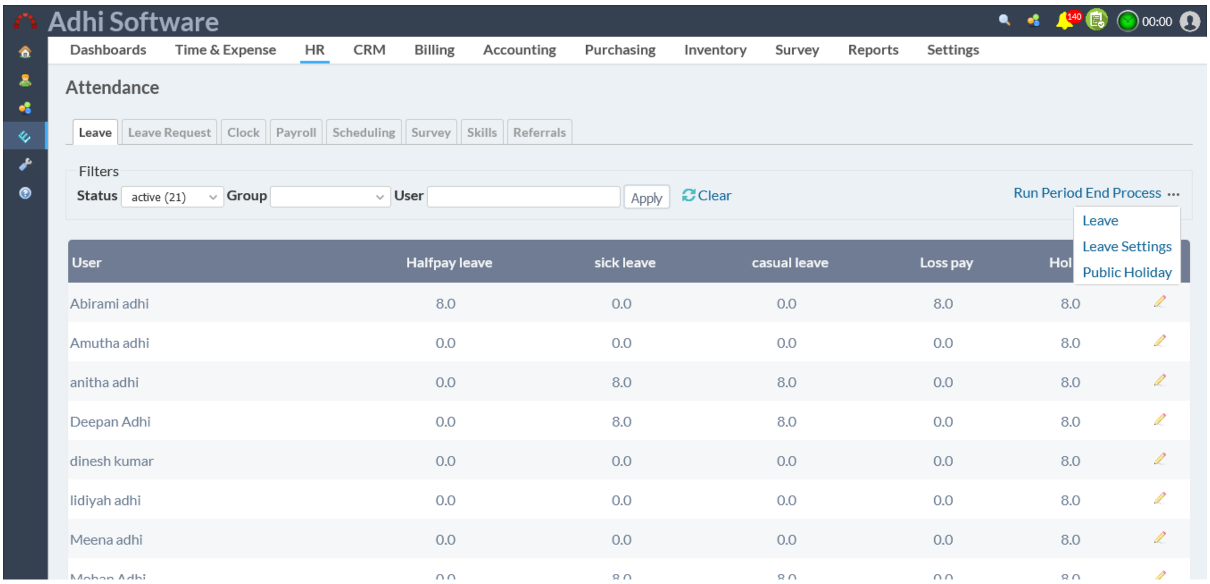


TE admin can add and edit other user’s clock in and clock out time.The Start textbox shows the clock in time, End textbox shows the clock out time, Total number of hours worked in a day excluding the break time hours is shown below the end time. Remaining hours are the number of hours left to be logged.



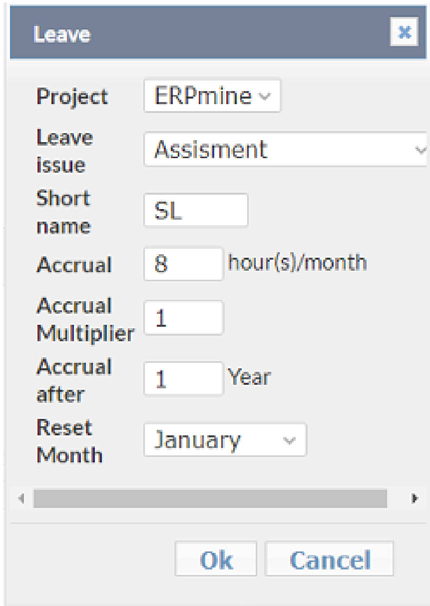

## 3.1 Leave Summary

The leave tab shows the summary of accrual, leave taken during the month. The list page shows closing balance for up to five configured leave information. User filters can be applied. Now, export list as CSV on all list pages.

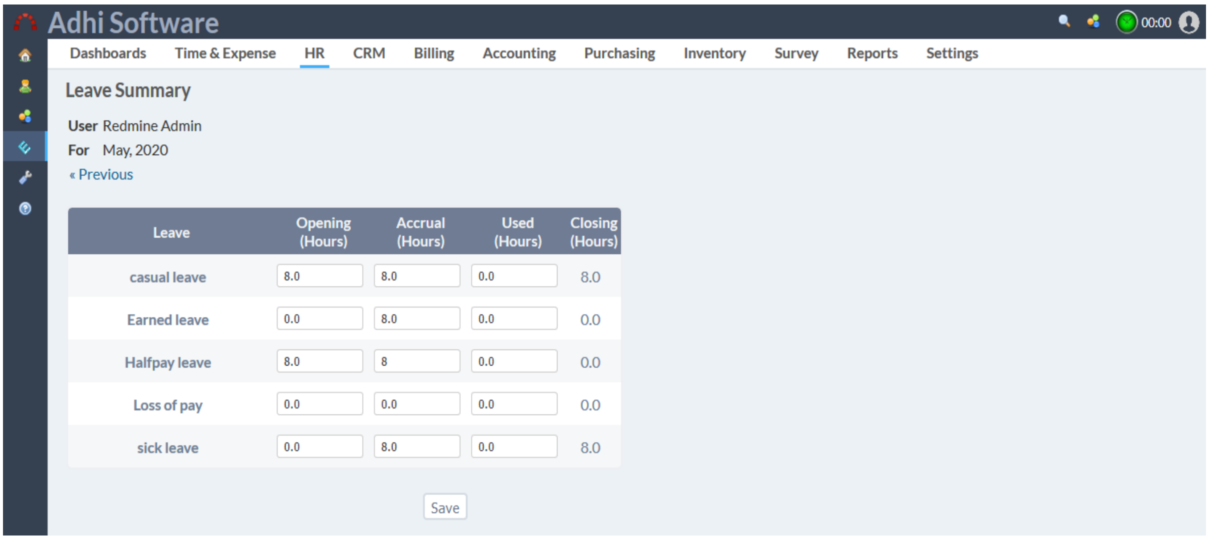



### 3.1.1 Leave Configuration

Leaves can be configured with short name, accrual, accrual after, reset month.




The detail page shows opening balance, accrual, leave taken in hours, remaining hours for each leave configured. TE admin are allowed to edit other user's leave information.




If the setting "**Approver Can Approve their own Time & Expense**" is enabled, then TE admin can edit his/her own leave information.

A scheduled job will be run at start of each month to calculate the leave accruals. The following rules are applied in calculating accruals

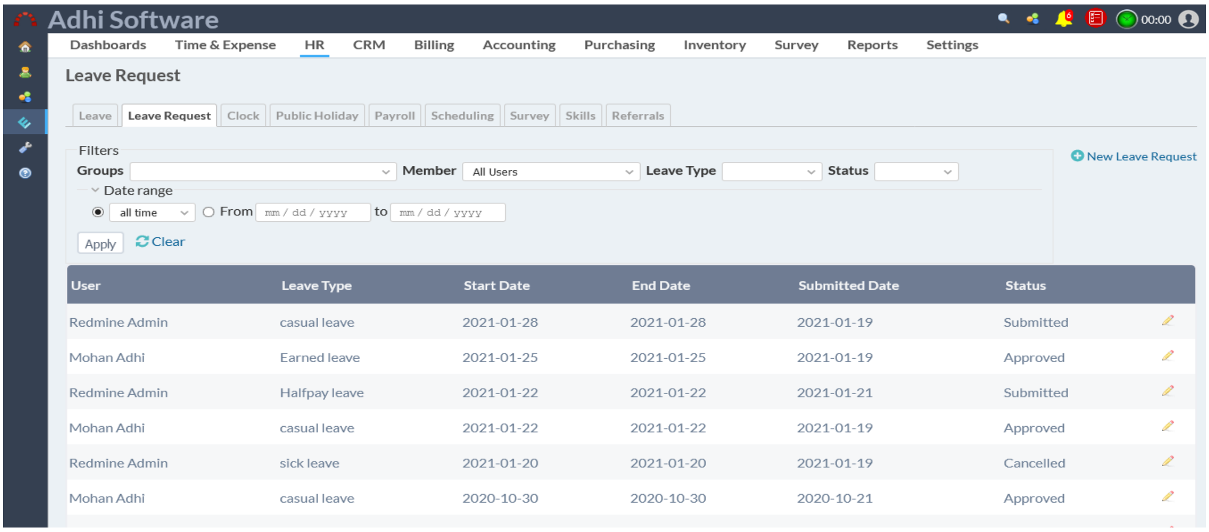
- Accrual will be given only when the user has attendance for at least 11 full days of month. For example, if the "**Default work time"** configured to 8 hours, then the user should have attendance for at least 88 hours to get accrual.

- If the 'reset month' is configured for a leave, then the balance will be reset to zero during that month. If the user has values in negative then it will be carry forwarded to the next period.

If the 'accrual after' is configured, then accrual will start only after passing the initial period configured.

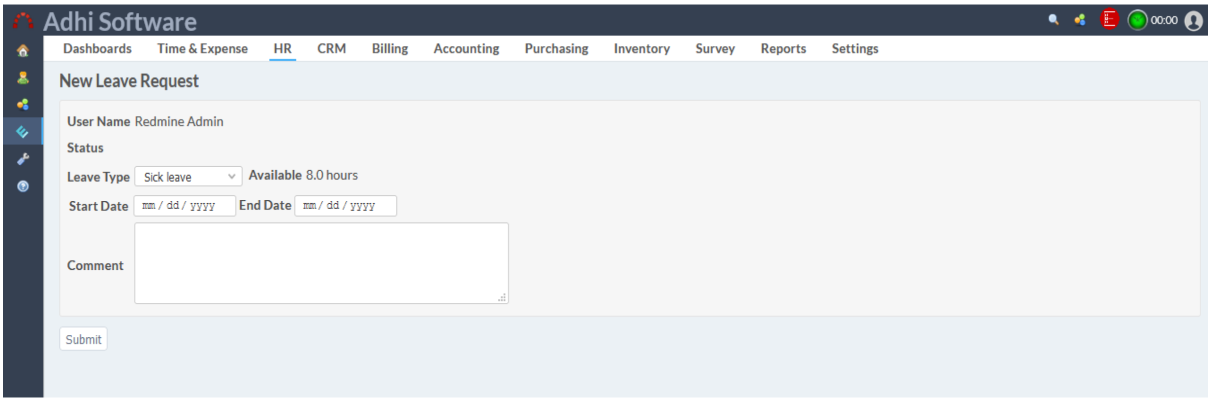
## 3.2 Leave Request

An employee can submit a leave request and can be approved by the supervisor. The list page by default displays all the leave requests. Leave type, status, date range filter and group filters can be applied to the list page.

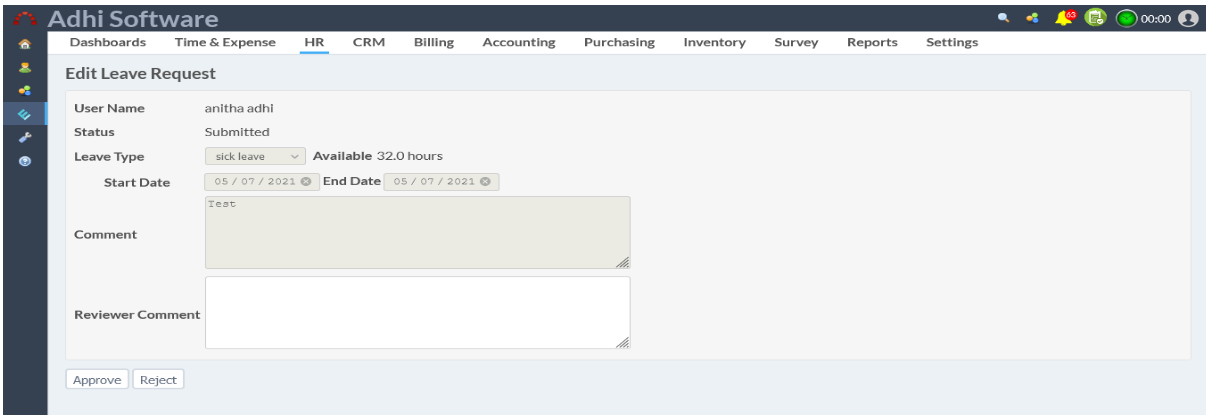



### 3.2.1 New/edit Leave Request

To add a new Leave Request, click on “New Leave Request” hyperlink. “Available hour” are automatically populated.




Once a leave request is submitted, the supervisor can approve the leave.

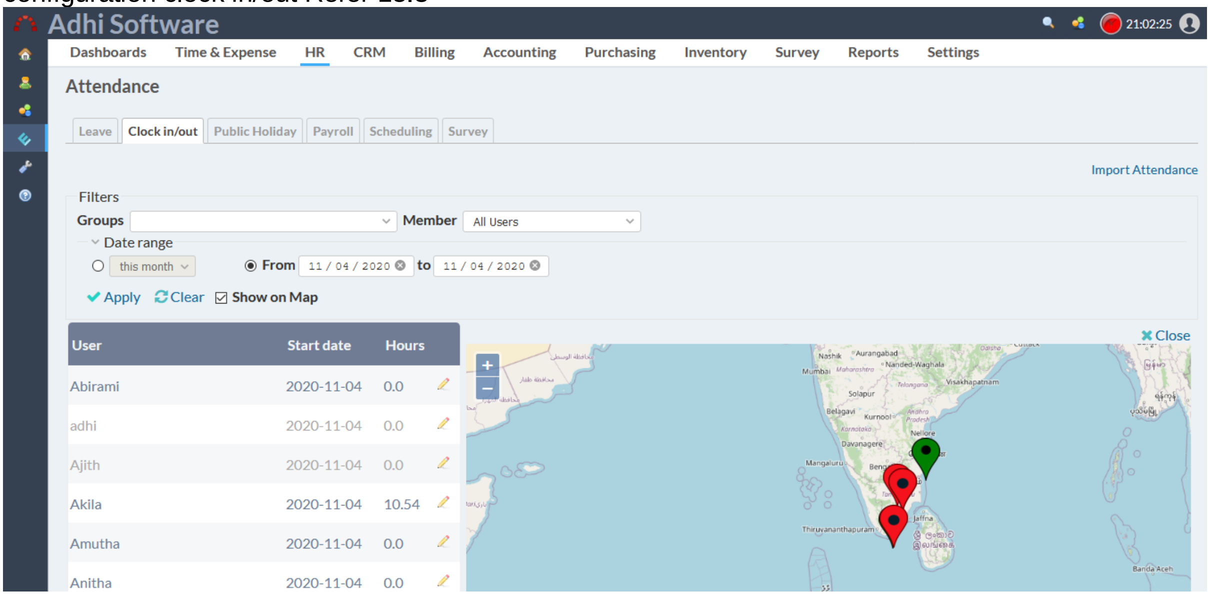


The supervisor will get the **Email notification**, Supervisor can be set in User detail page. To Configure User setting Refer 15.8

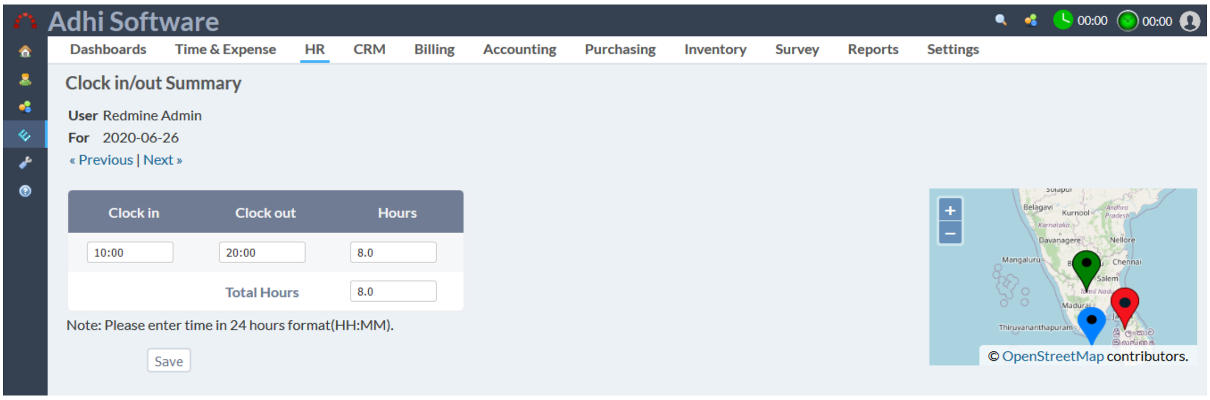
If additional people need to to be notified, a user group can be used. To setup the leave notification for a user group, please navigate to "Settings -> Permission -> Select Group -> Check Receive Leave Notification"
The configuration leave request Refer 15.3.4

## 3.3 Clock in/out

The list page by default displays the daily clock in/out for the current month. Date range and Group filters can be applied to the list page. In Plugins > Attendance, geolocation checkbox is enabled and HR > Attendance, ShowonMap is enabled, then Map shows the user clockin/out location. The configuration clock in/out Refer 15.3

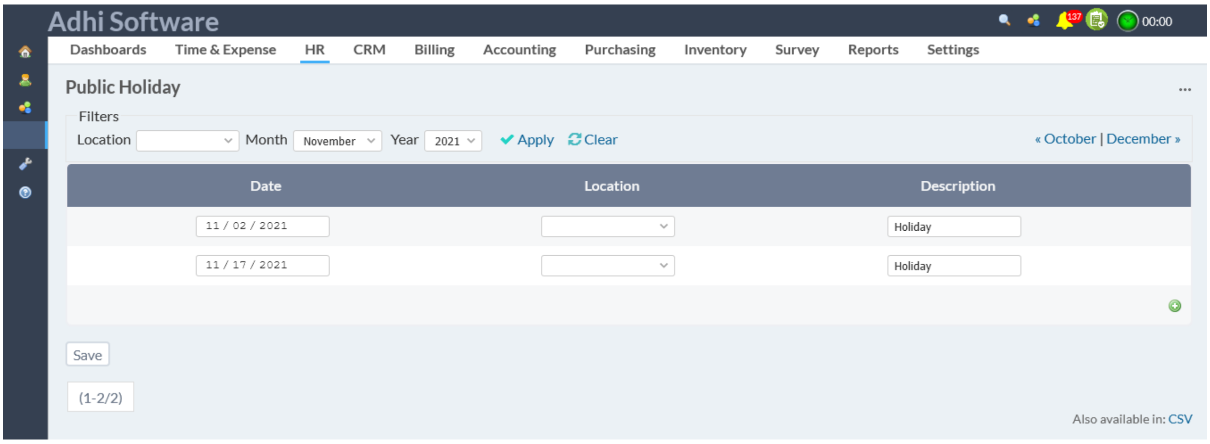



### 3.3.1 New/Edit Clock in/out

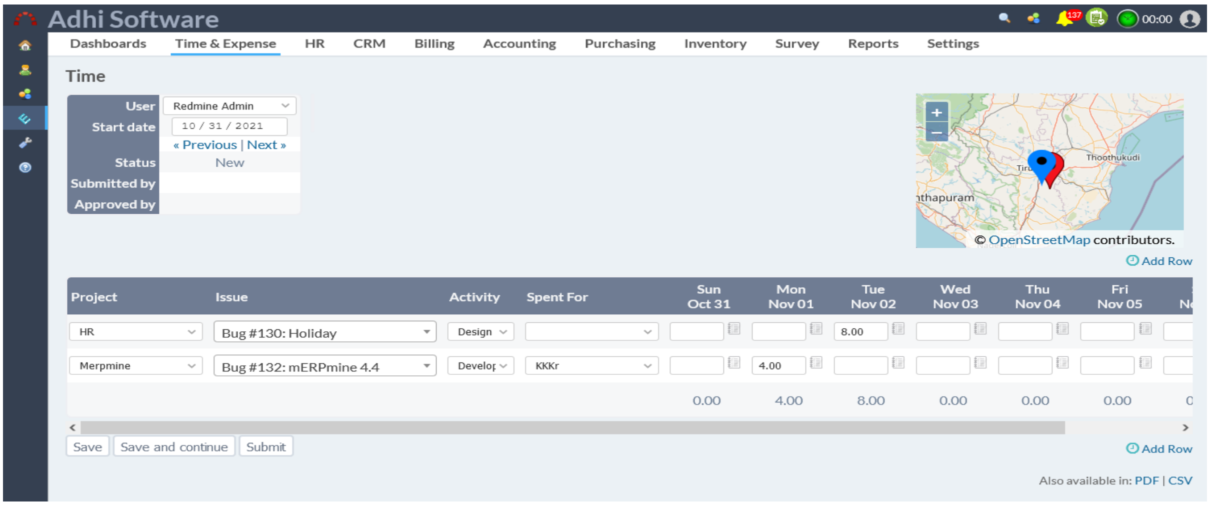



## 3.4 Public Holidays

A public holiday can be set for a particular location and considered non-working day during the year. Location and Date range filters can be applied to the list page.

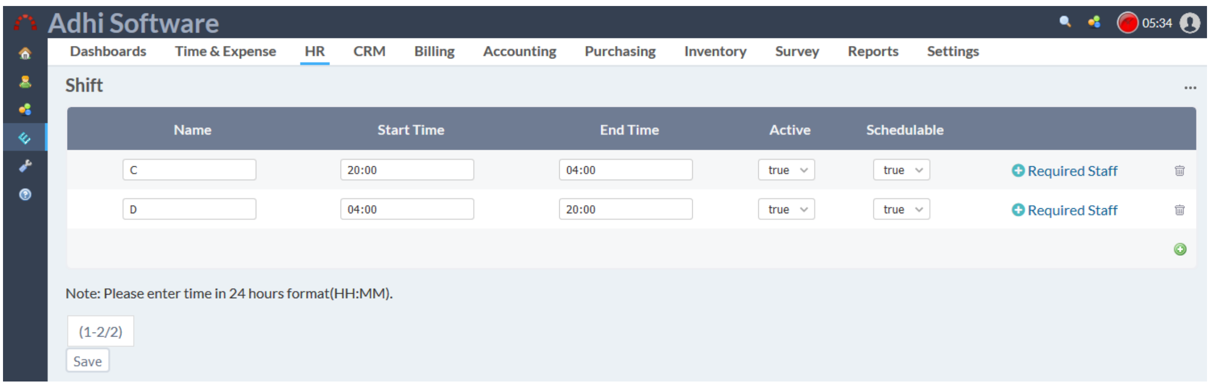



The configuration Public holiday Refer 16.3.



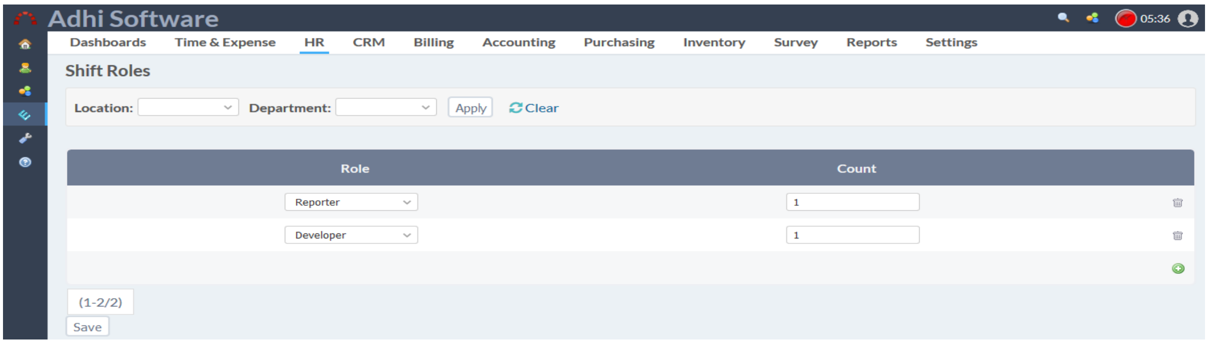

## 3.5 Shifts

Employee shifts can be configured in 24 hour format and shift can be activated and used in schedule generation.



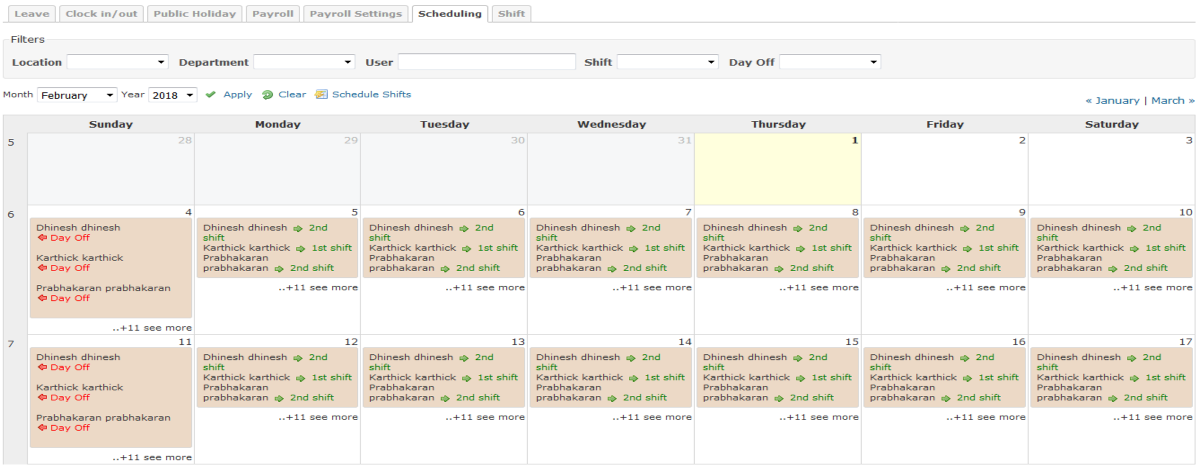

## 3.6 Shift Roles

Required staff per Role per shift can be setup under shift role page.



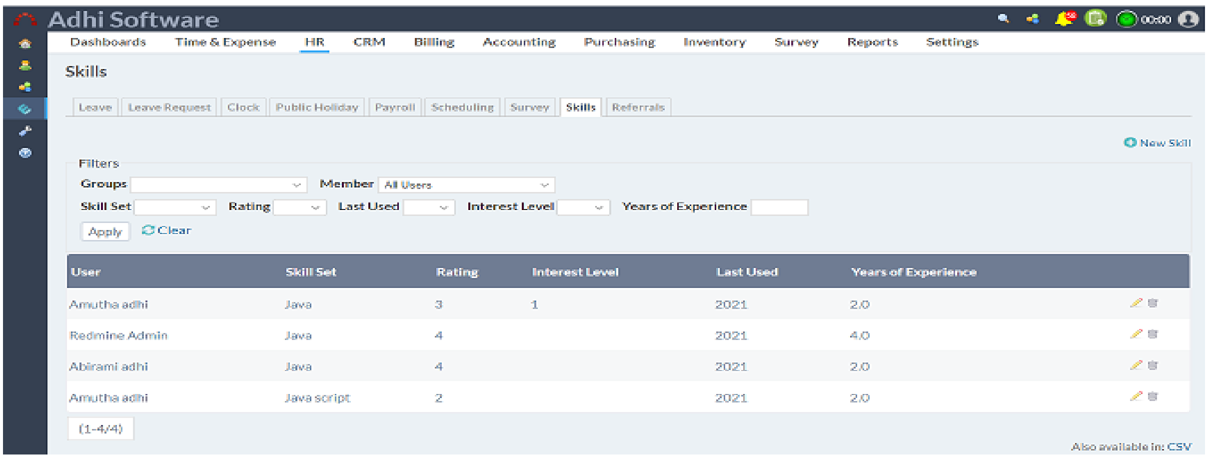

## 3.7 Scheduling

Shifts can be scheduled on a weekly basis. The calendar view displays the daily scheduled shifts for the current month by default.. Date range, location, department, user, shifts and day off filters can be applied to the view. 'Schedule shifts' link is used to schedules the shifts for employees by rotational basis. Employees can enter their shift preferences and the preferences will be taken into consideration while scheduling.



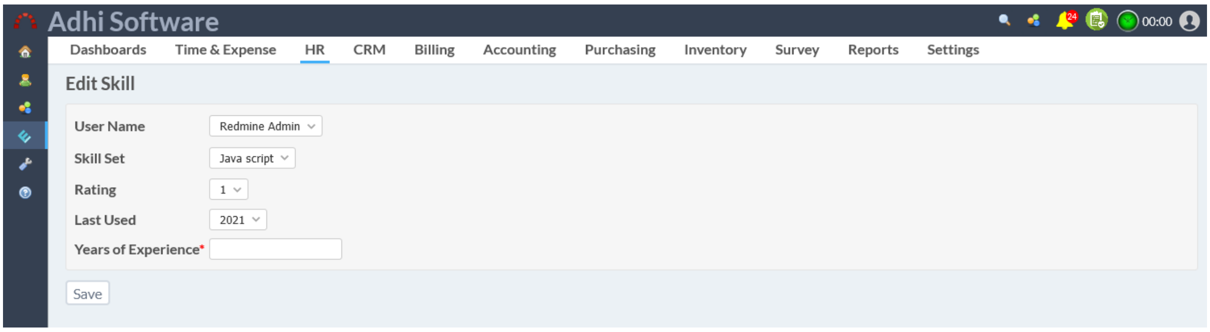

## 3.8 Skills

Employee skills can be captured in Erpmine, The list page by default displays the rating, skill set, interest level and years of experience.



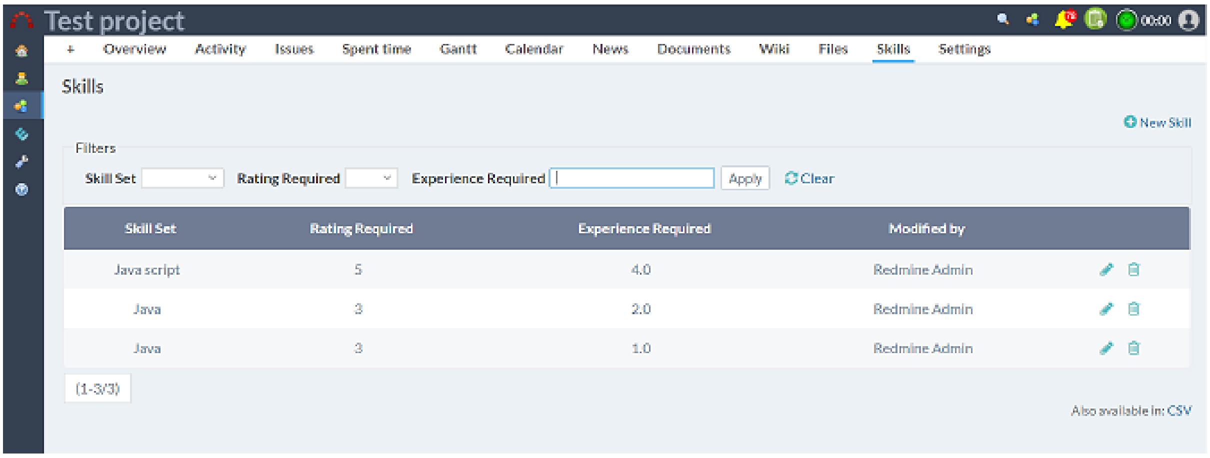
### 3.8.1 New/Edit Skills

To add a new new skill, click on "new skill" hyperlink. Skillsets can be added through settings > enumeration. Refer enumeration 13.1.



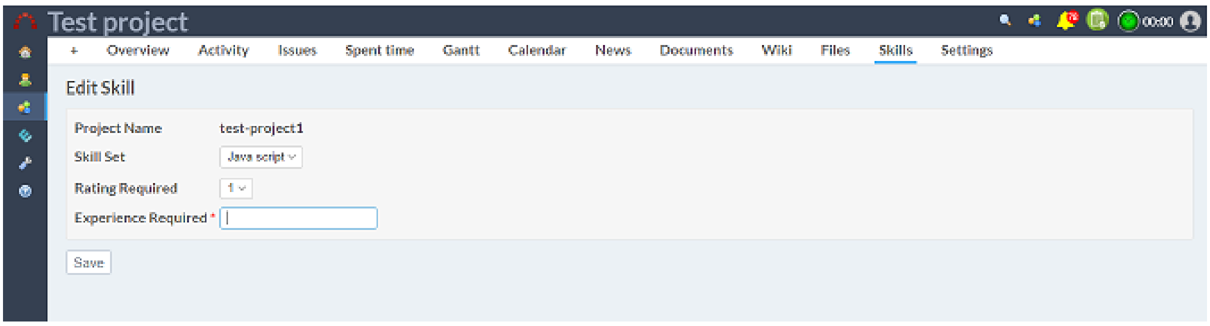

### *3.8.2 Project skills*

Project can list the skills required for its team members. The list page by default displays the rating , skill set, interest level and years of experience. To edit project skill user should have create and edit project permission.



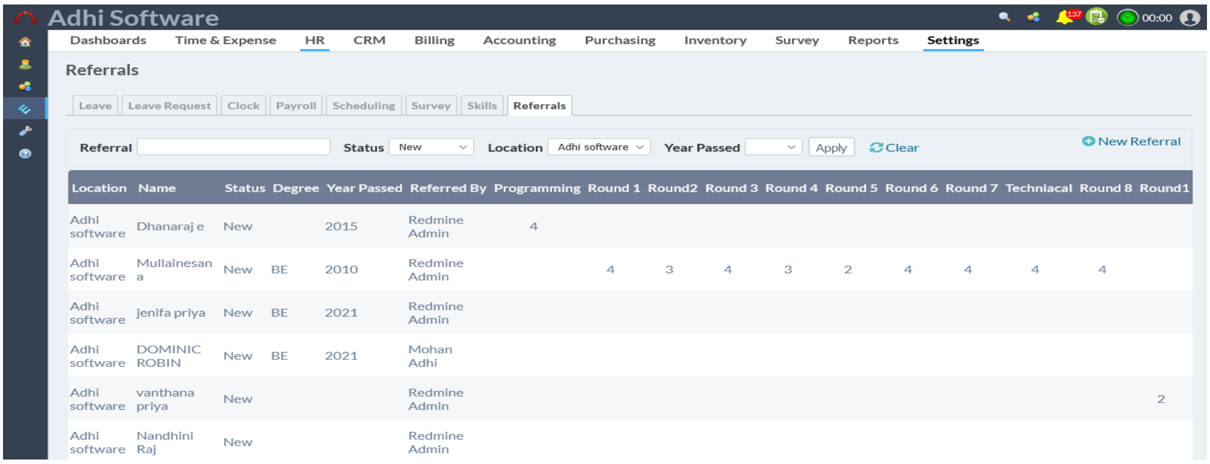

### 3.8.3 New/Edit Project Skills

To add a new project skill, click on "new skill" hyperlink. Skillsets can be added in settings > enumeration. Refer enumeration 13.1.



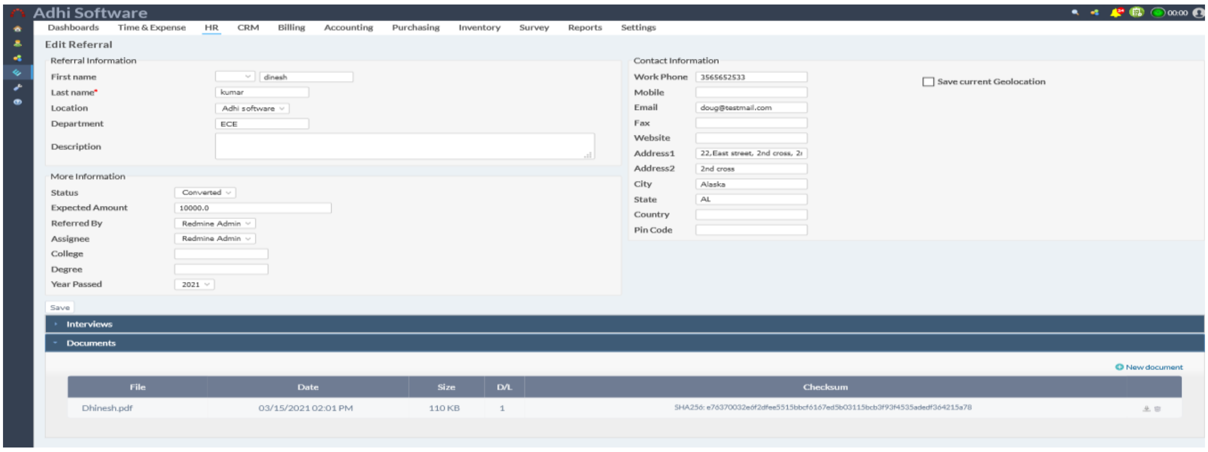

## 3.9 Referrals

The Employee referrals can be added into the system. The list page by default displays the Referred candidates' details. These referrals can later be interviewed and hired. Once candidates are hired, the referral details will be populated to user as Hired employees. Referral name can be wildcard searched and Status filter is also available for the list page.




### 3.9.1 New/Edit Referrals

The Edit referral page helps to update the referral information. Interview details can be added. The interview documents like resume, mark sheets, certificates can be uploaded to document section.

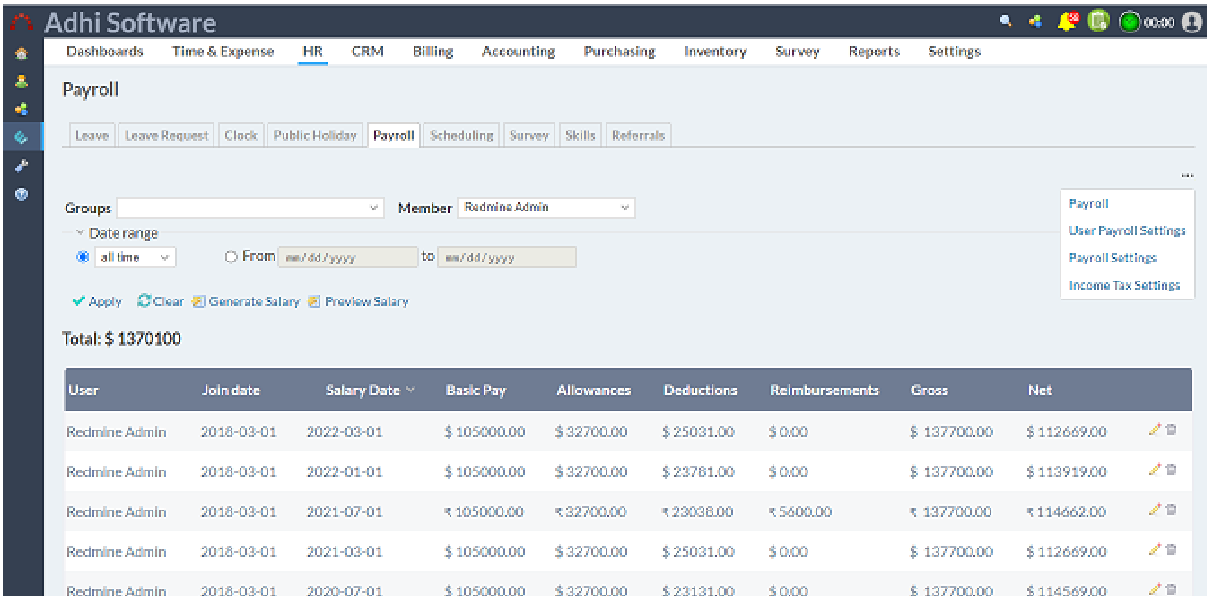



# 4 Payroll

This module allows the user to setup the payroll for the employees. The payroll settings can be done at the company level and can be overrides at the employee level. Once the payroll setup is done, the payroll can be generated on a periodic basis (monthly, weekly, bi-weekly etc). The payroll generation can be automated as well.

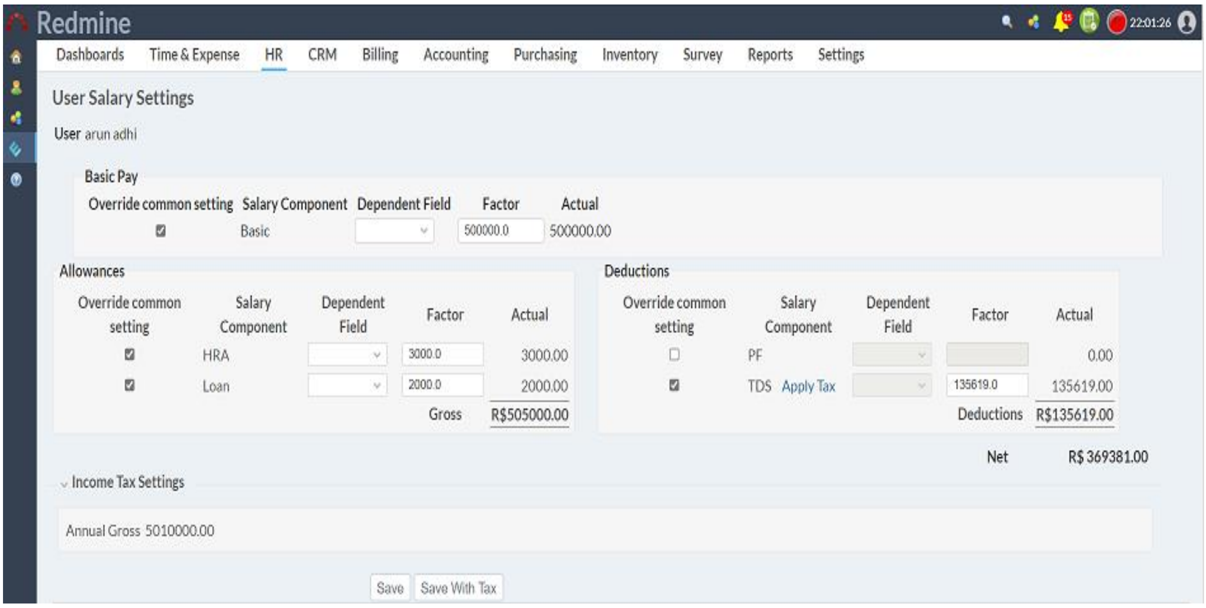
## 4.1 Payroll List

The list page by default displays the payroll for the current month. Date range and group filters can be applied to the list page and generate the salary, Plugin, configure the project. Individual payroll entries can be deleted from this page as well. To add projects in the reimbursement Plugins>payroll >Reimbursement Projects. The configuration Payroll Refer 15.4.

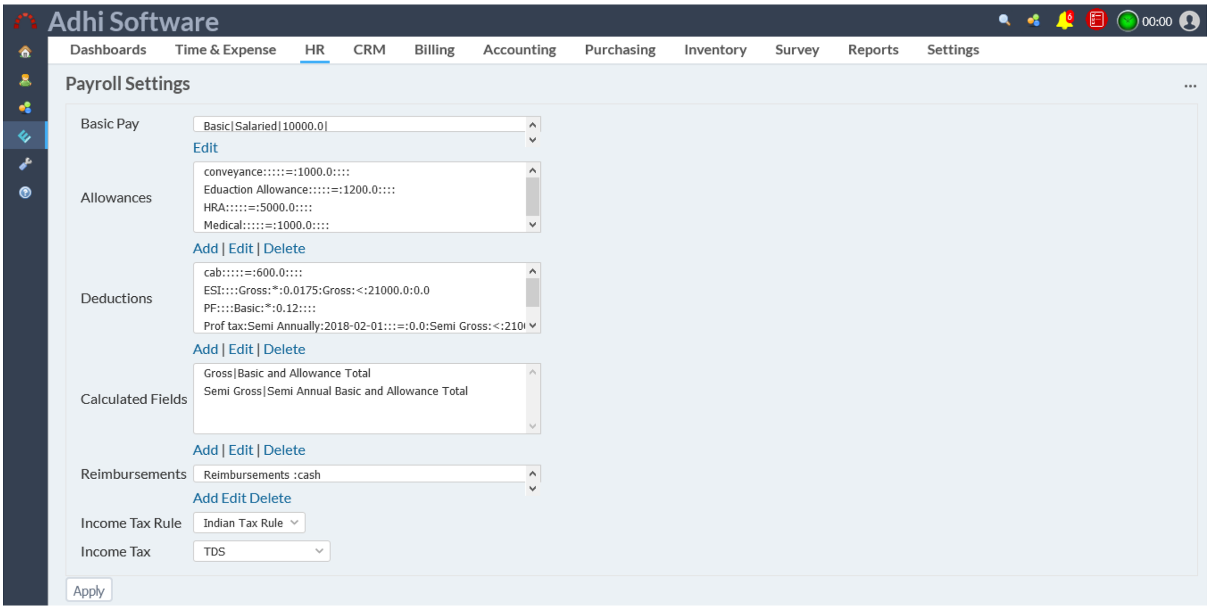



### 4.2 User Settings

To override an user payroll setting from the company level payroll setting, the User Setting tab is used. The list page by default displays the list of users. User, status and Group filters can be applied to the list page. The Edit page can be used to override the payroll setting for the individual user. “Income tax" is calculated based on Government rule. Taxes are automatically applied. The apply tax automatically calculate the tax into tds fields based on increment



### 4.3 Payroll Settings




#### *4.3.1 Basic pay*

Basic pay can be configured with name, salary type, debit Ledger and factor. Factor is nothing but the wages, it can be a hourly rate or wages.

#### *4.3.2 Allowances*

Allowance can be configured with name, frequency, start date, debit ledger, depends on and factor.

| # | field | description |
|---|---|---|
| 1 | name | Name of the allowance/deduction |
| 2 | Frequency | Frequency of the allowance, for monthly allowances, it can be left as blank |
| 3 | Start date | Applicable only when a frequency is set |
| 4 | Dr/Cr Ledger | Select the appropriate GL ledger |
| 5 | Depends on | Selected the field on which this depends on |
| 6 | Factor | Factor can be either a % or an amount |

#### *4.3.3 Detection*

Deduction can be configured with name, frequency, start date, depend on and factor, please refer the table above.

#### *4.3.4 Reimbursement*

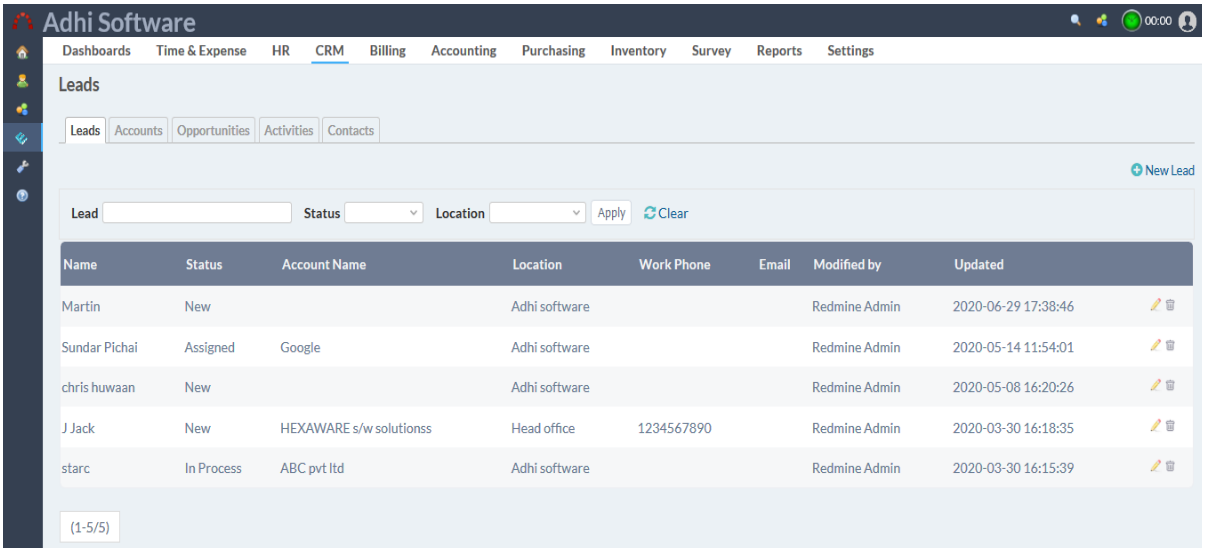
Reimbursement can be configured with name and ledger.

## 5 CRM

This Module allows the user to manage CRM entities like Leads, account contacts, opportunities and activities.

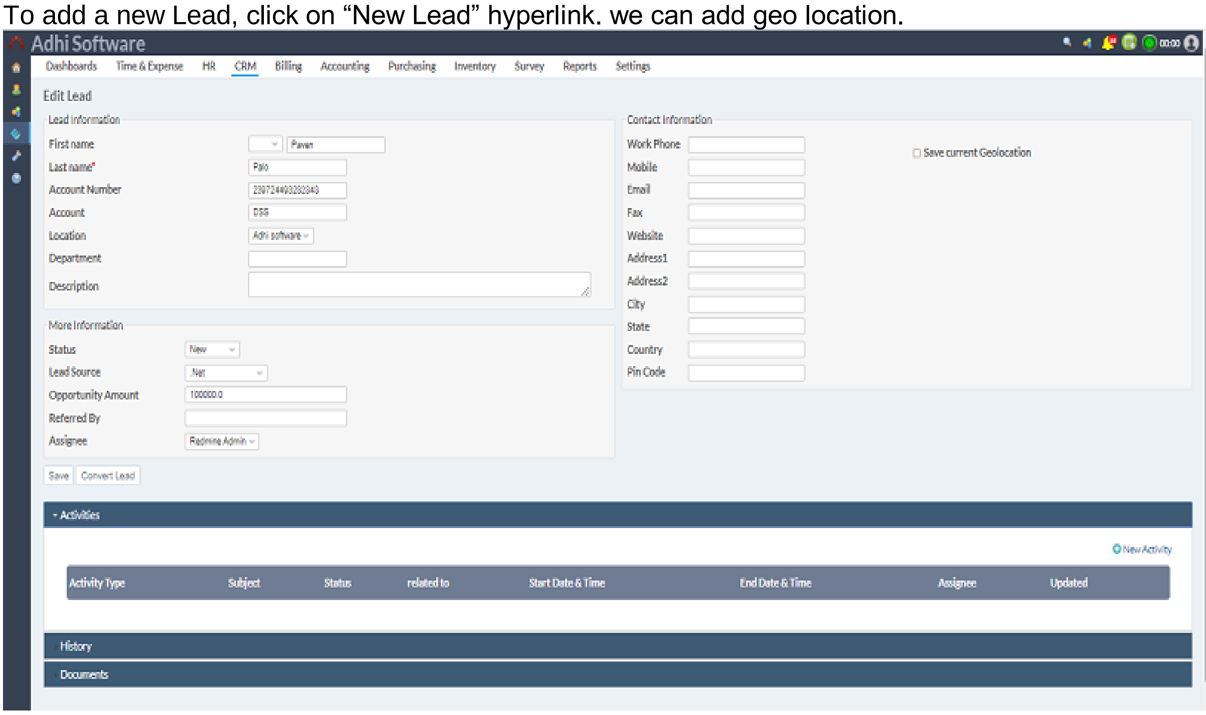
### 5.1 Leads

The list page by default displays the Leads details. Once Leads are converted, they are no longer visible in the Leads List View. Lead name wildcard search and Status filters can be applied to the list page.



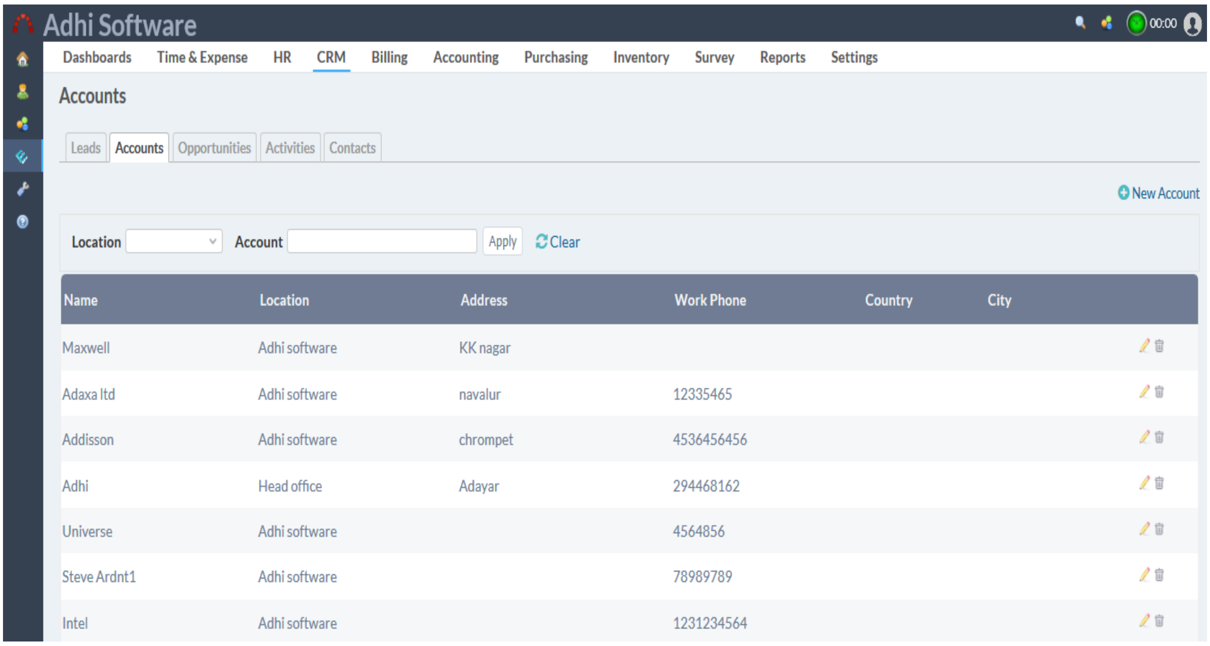

#### 5.1.1 New/Edit Lead

To add a new Lead, click on "New Lead" hyperlink. we can add geo location.



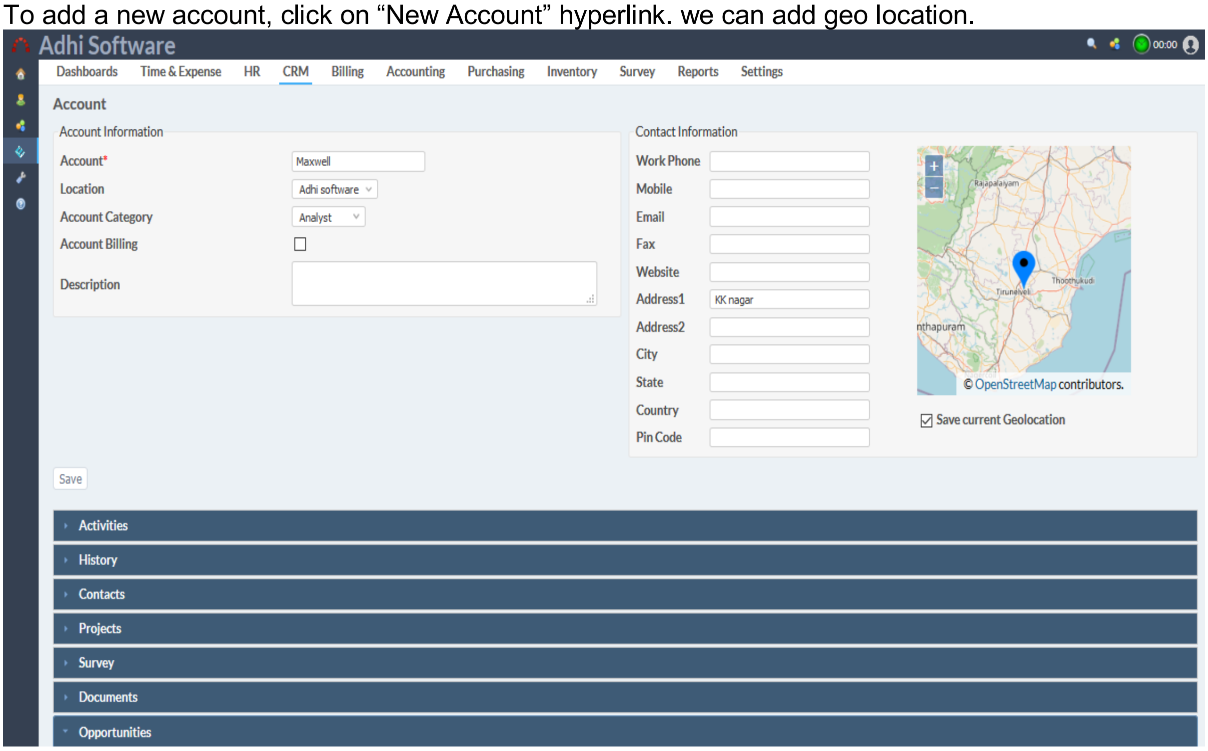

### 5.2 Accounts

Accounts are clients for whom the projects are executed, there can be multiple projects for an account. The list page by default displays the list of accounts. Account name wildcard search can be applied to the list page.



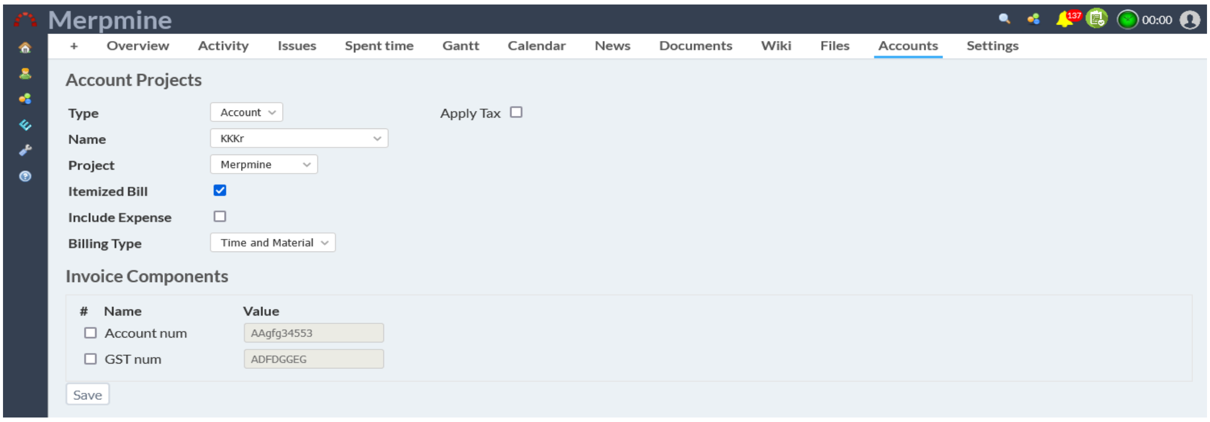

### 5.2.1 New Account

To add a new account, click on “New Account” hyperlink. we can add geo location.



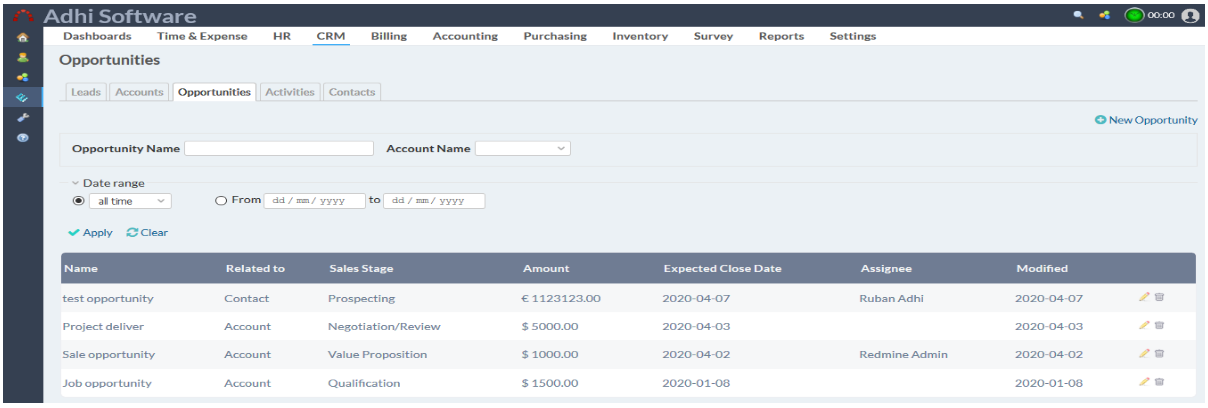

### *5.2.2 New Account Project*

To add a new Account Project, click on “New Account Project” hyperlink. The new/edit account project page associate the type, name and projects and define the properties such as; account/contact billing types, account/contact projects taxes, include expense and billing items and add the billing schedules. Billing schedules are required for Fixed Cost projects. invoice component can be edited



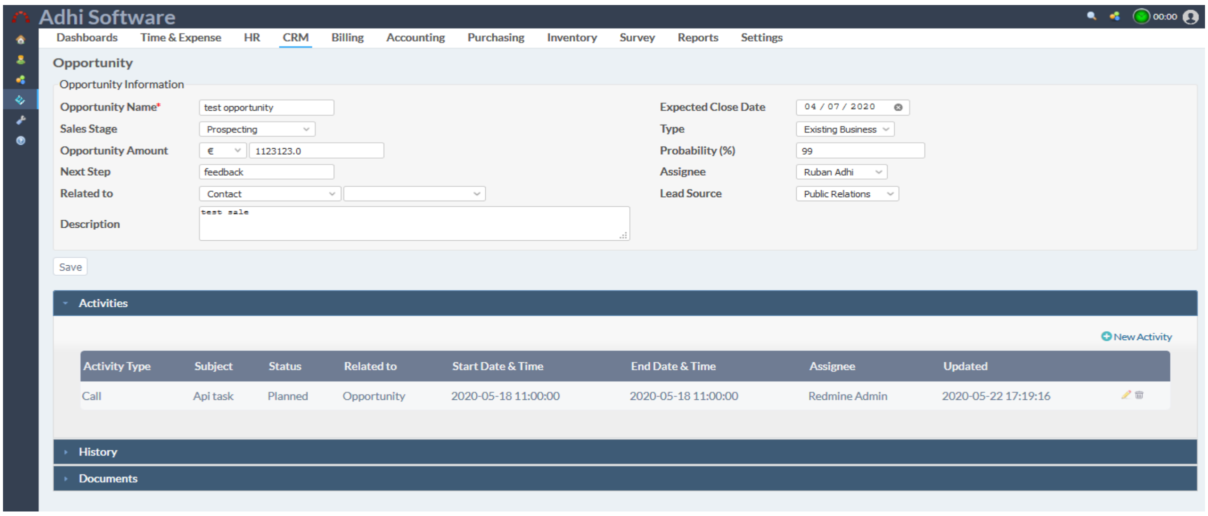

## 5.3 Opportunities

Any sales opportunities within an account can be captured. The list page by default displays the list of opportunities. Opportunity name wildcard search, account and Date range filters can be applied to the list page. Opportunity can be removed.



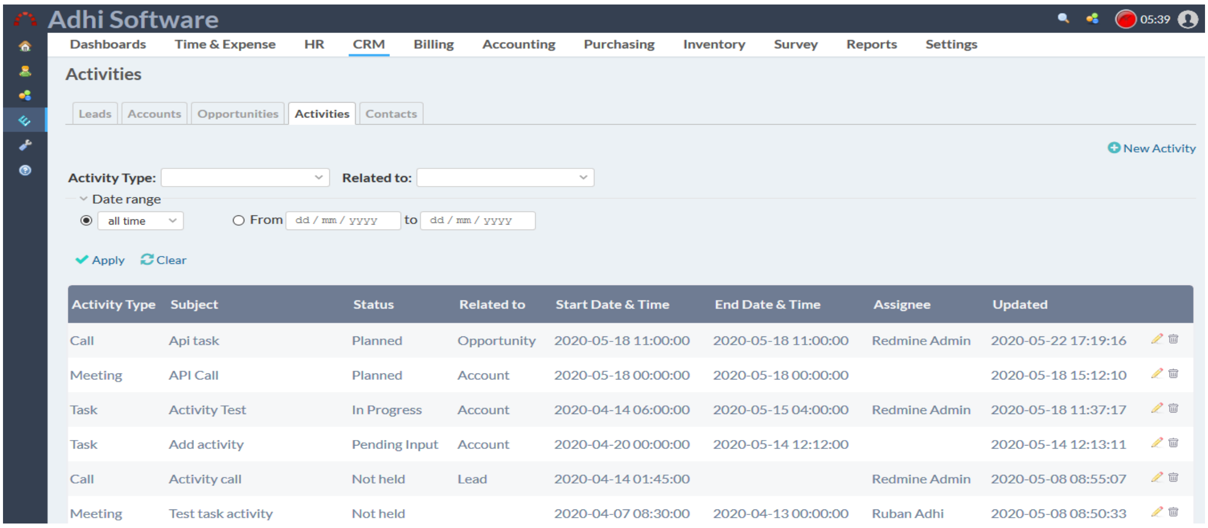

### 5.3.1 New/Edit Opportunities

To add a new Opportunity, click on "New Opportunity" hyperlink



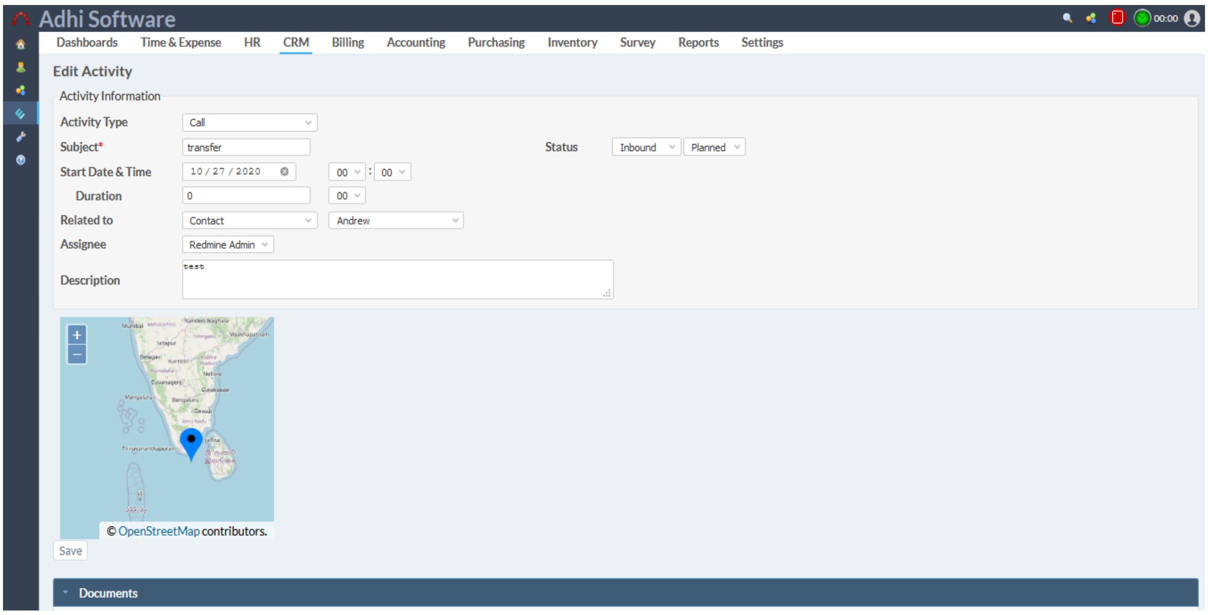
## 5.4 Activities

The sales activities performed by the sales team can be tracked. The list page by default displays the list of activities. Activity type, Related To and Date range filters can be applied to the list page. Activity can be removed.



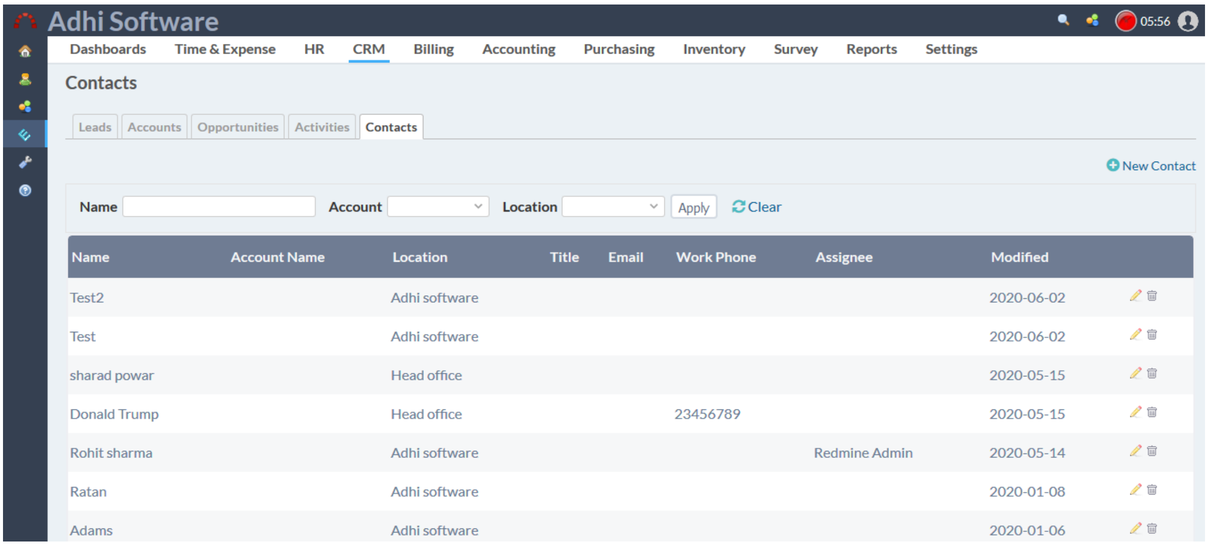

### 5.4.1 Add/ Edit Activities

To add a new Activity, click on "New Activity" hyperlink



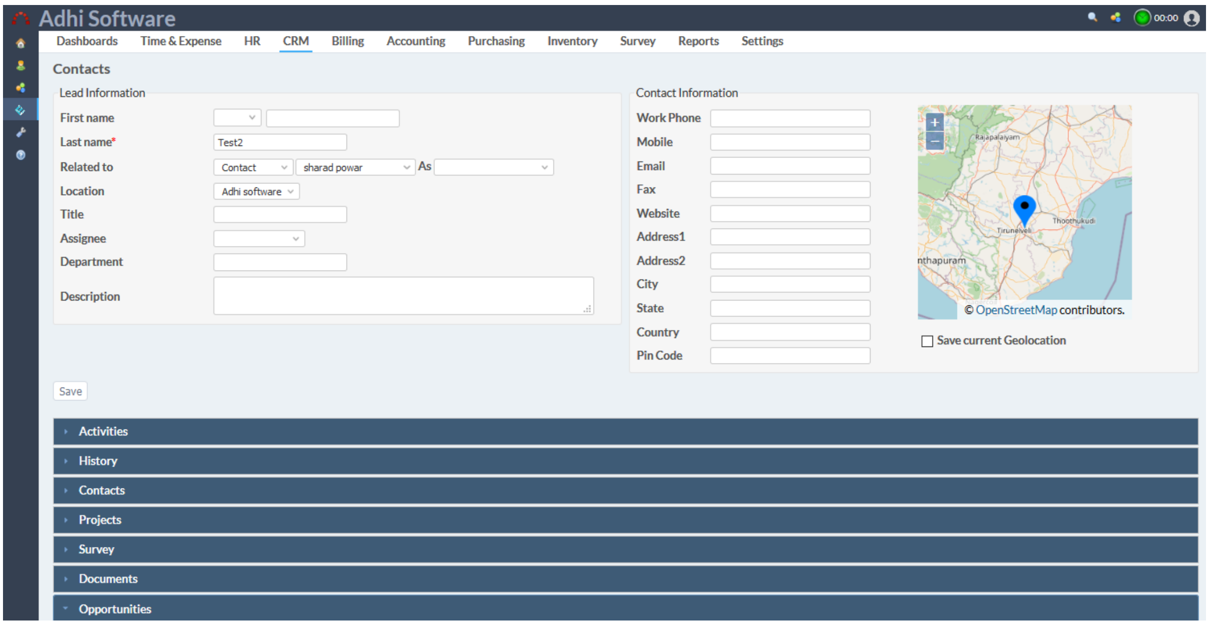

## 5.5 Contacts

A contact is an individual, they may work for an account. Contact can also related to many contacts. The list page by default displays the list of Contacts. Contact name wildcard search and account filters can be applied to the list page. Contact can be removed.




### 5.5.1 Add/ Edit Contacts

To add a new Contact, click on "New Contact" hyperlink. We can add geo location.

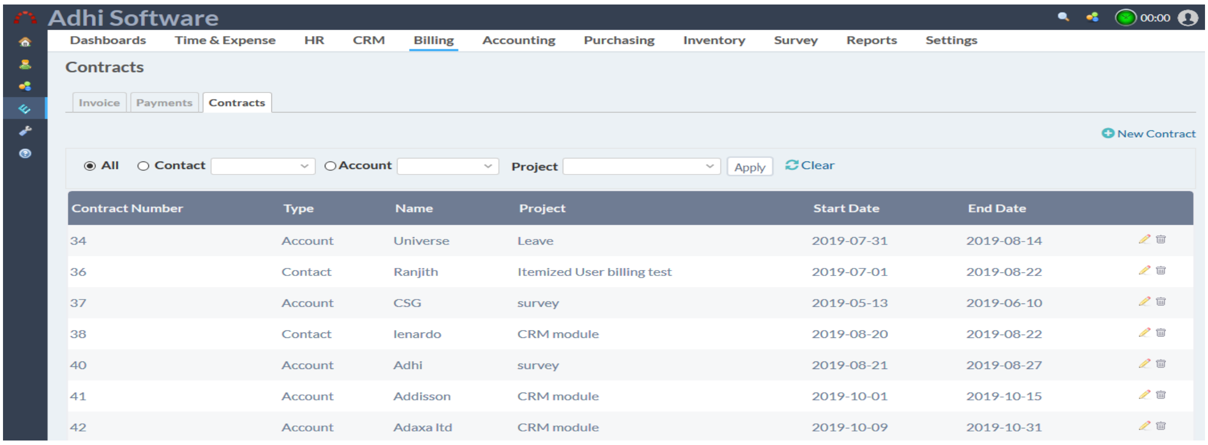



# 6 Billing

This module allows the Billing admin to setup the Billable projects, generate invoice and print the invoice. Billing admin can manage accounts, contracts, taxes and billable projects. Once the Billing setup is done, the invoice can be generated on a periodic basis (quarterly, monthly, weekly, bi-weekly etc). The invoice generation can be automated as well.

## 6.1 Contracts

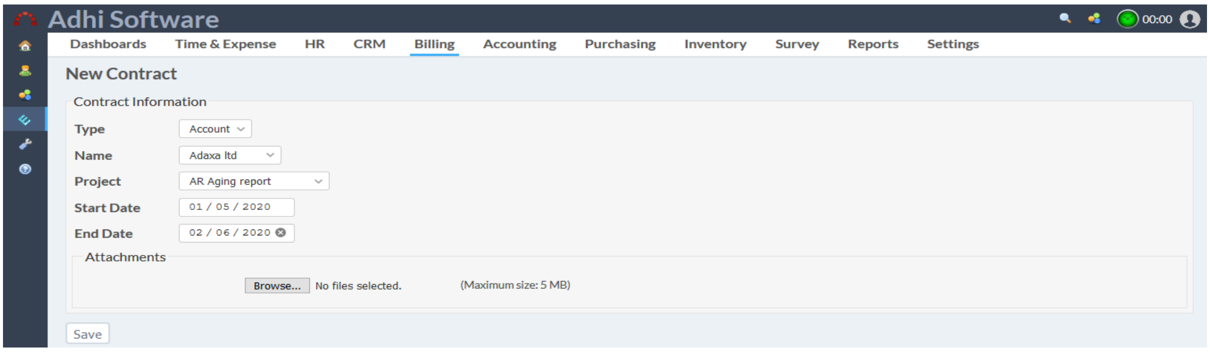
A contract is a signed agreement between an Account, Contact and the Company, they are typically signed before starting on a project. The list page by default displays the account and Contact contracts. Account, project filters can be applied to the list page.




.

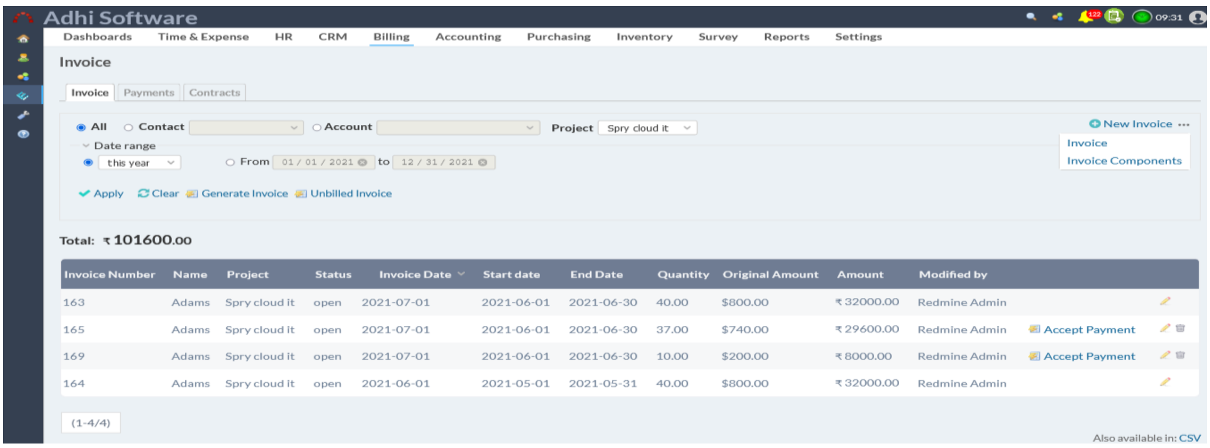
### 6.1.1 New contract

To add a new contract, click on “New Contract” hyperlink

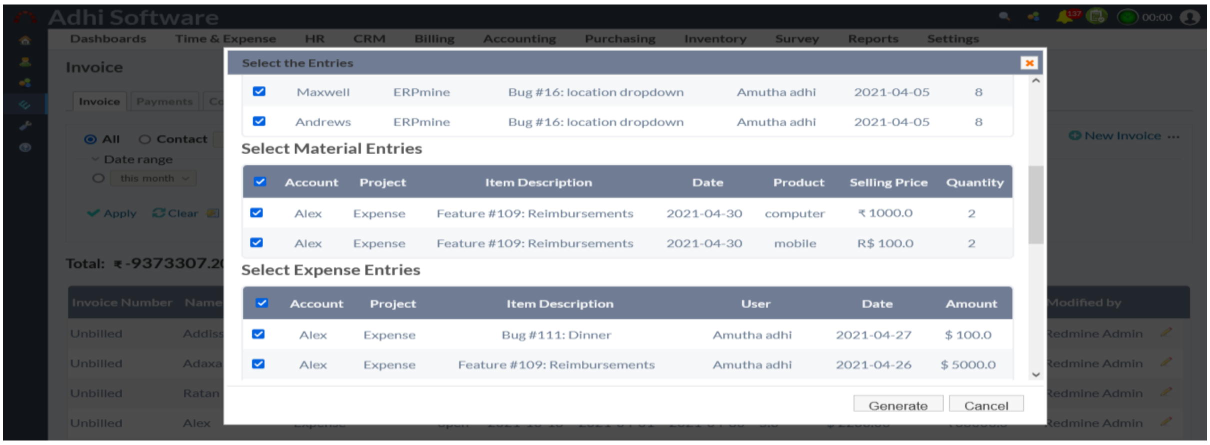



## 6.2 Invoice List

The list page by default displays the invoice for the current month. Date range, account and project filters can be applied to the list page. The invoices for a period can be generated from this page also. View the unbilled invoice using ’ Unbilled Invoice’ link. Individual invoice entries can be deleted from this page as well. Invoice component can be added in new invoice side bar and In each project also Invoice component can be added. The configuration invoice Refer 15.5.

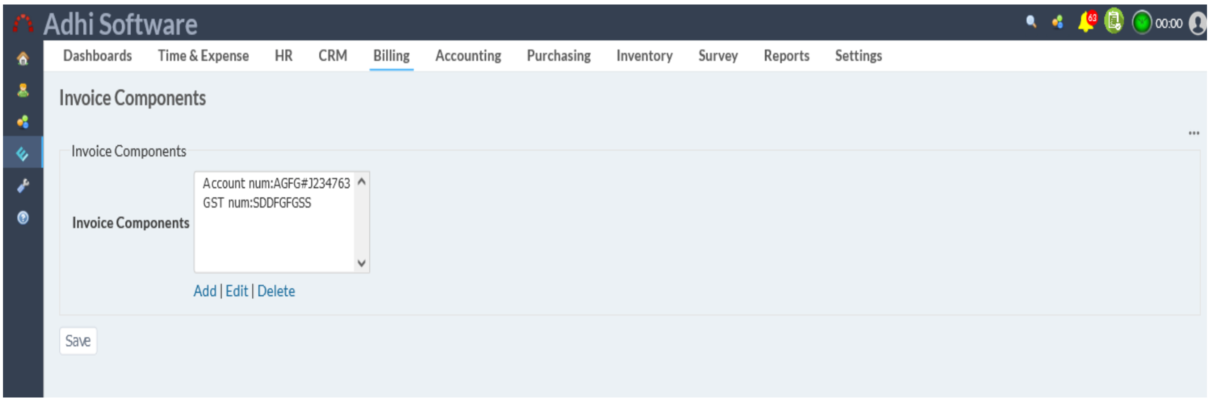



To **show billed hours** in the invoice, Click the detail icon next to the quantity textbox in invoice and unbilled invoice, clicking that opens a detailed pop up with the list of time entries,expense and material entries billed.



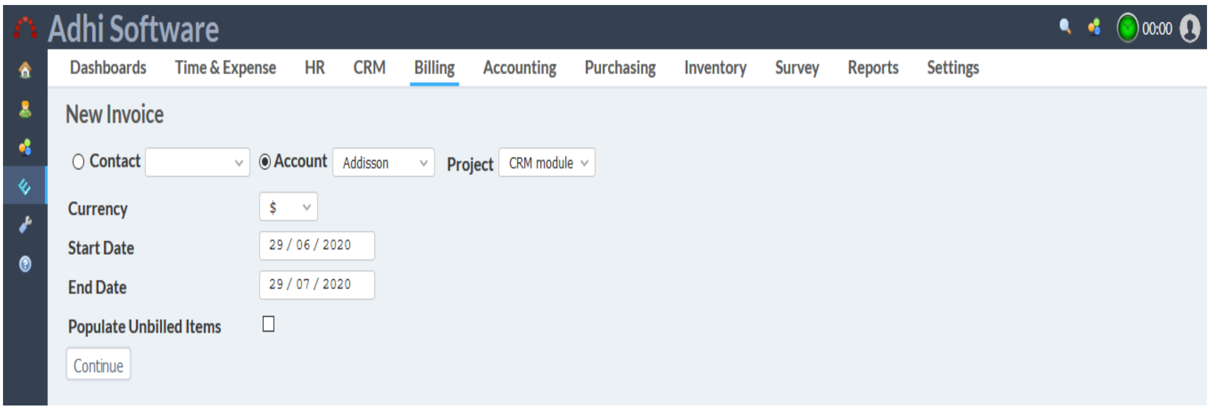
### 6.2.1 *Invoice Components*

The user can configure the invoice components, which will be included in the printed invoice.



### 6.2.2 *New Invoice*

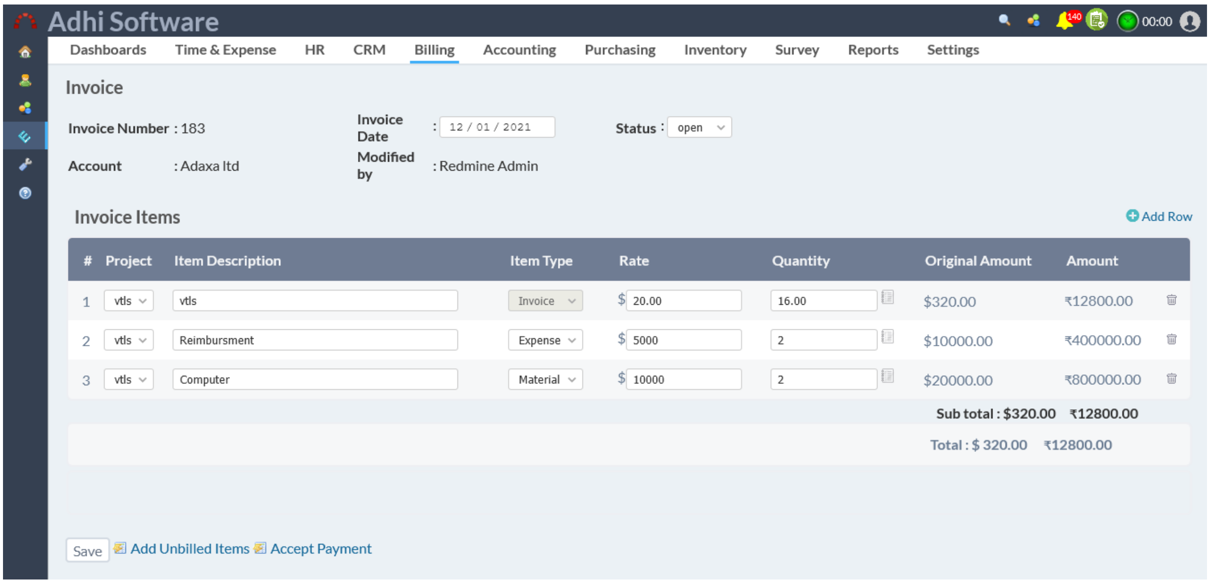
To add a new invoice click on “New Invoice” hyperlink



### *6.2.3 Edit Invoice*

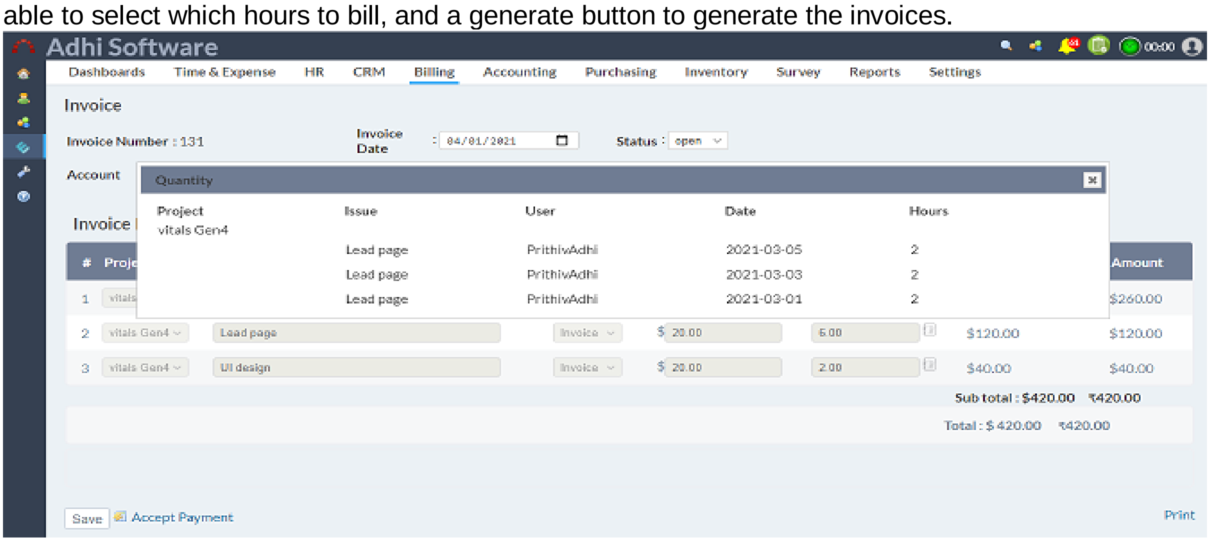
The invoice edit page displays the invoice details such as invoice number, project name, account name, invoice date, status, modified by and invoice items. Invoice items can be edited and new invoice items can be added also. Taxes are automatically applied. Invoices can be printed using 'print invoice' link. The update button inside the invoice detail page, which should include all the unbilled hours from prior to the end date of the invoice. Then upon hitting save, it should save the invoice.
After accepting payment, the update option should not be there.

'Add unbilled items' button inside the invoice detail page, which should include all the unbilled hours from prior to the end date of the invoice. Then upon hitting save, it should save the invoice. After accepting payment, the update option should not be there. Item can be exported as CSV and PDF.




### *6.2.4 Unbilled invoice*

All unbilled items can be viewed by clicking on the 'Unbilled Invoice' link. To **generate an invoice** on selected hours, it will open a dialog box with the list of time entries with a checkbox, you should be able to select which hours to bill, and a generate button to generate the invoices.



### *6.2.5 Print invoice*

**INVOICE**

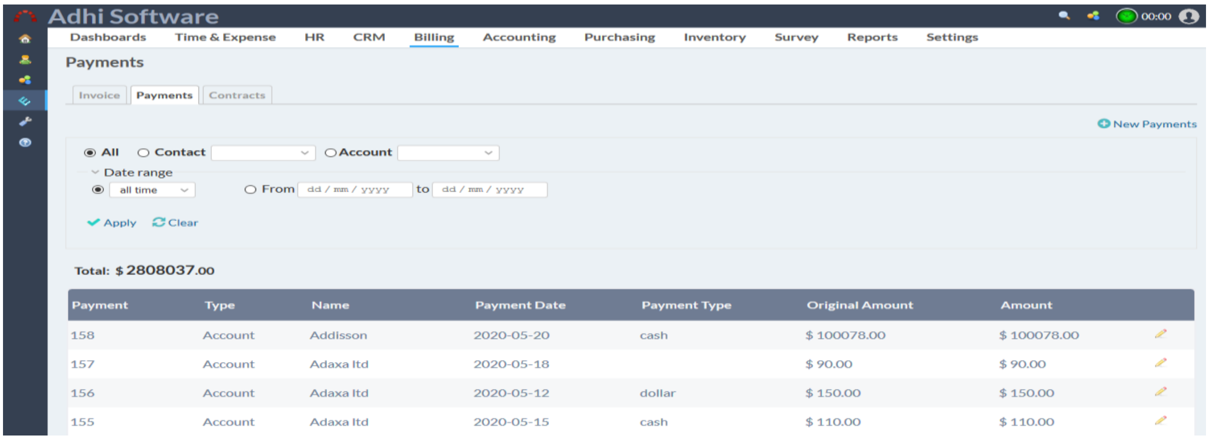
| Name & Address of the Supplier | Name & Address of the Customer | Invoice Number | Invoice Date | | |
|---|---|---|---|---|---|
| | sebakka | 6 | September 1, 2017 | | |
| **Contract / Purchase Order / Work Order / Agreement Date & Reference** | | | **Period** | | |
| 1 - April 4, 2017 | | | August 1, 2017 to August 31, 2017 | | |
| **Item Description** | **Billing Type** | **Rate** | **Quantity** | **Currency** | **Amount** |
| SPA - SPA - ISSUE - 1 | Hours | 20.00 | 5.00 | $ | 100.00 |
| GST - 1 | | 18.00% | | $ | 18.00 |
| GST - 2 | | 28.00% | | $ | 28.00 |
| | | **Total** | **5.00** | **$** | **146.00** |
| Iphone Apple Iphone 6 | | 3000.00 | 1.00 | $ | 3000.00 |
| Iphone - P-GST | | 18.00% | | $ | 540.00 |
| | | **Total** | **1.00** | **$** | **3540.00** |
| | | **Grand Total** | **6.00** | **$** | **3686.00** |
| **Amount in Words** | Three thousand six hundred and eighty six only. | | | | |

**Place :**
**Date :**

**Authorized Signatory**

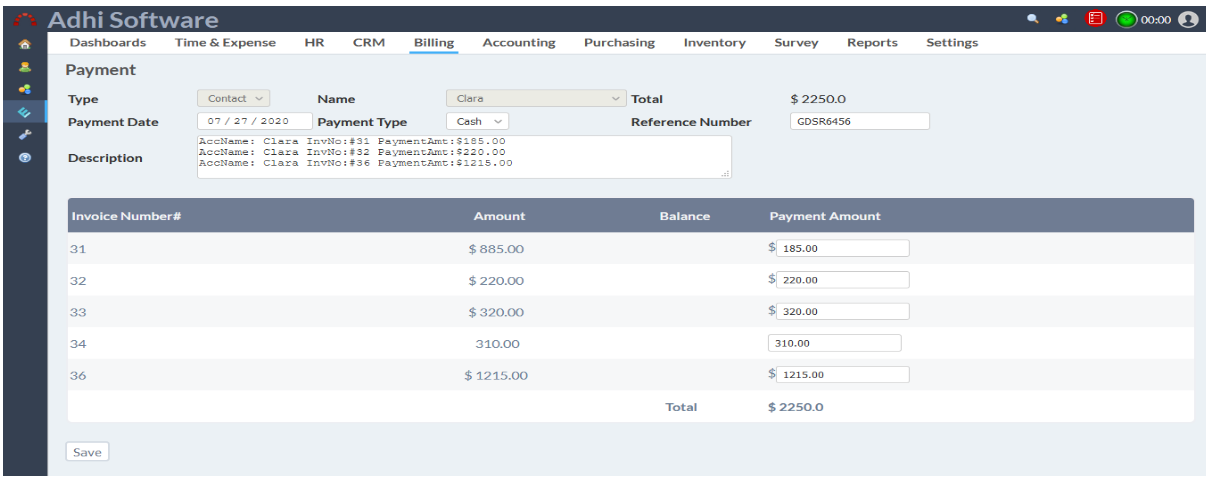
## 6.3 Payments

Payments can be accepted for invoices, multiple invoices can be paid by a single payment. The list page by default displays the payments for the current month. Contact, account, project and Date range filters can be applied to the list page. Added delete icon in the list page.




### *6.3.1 Edit Payment*

The payment edit page displays the payment details such as payment date, payment type, reference number, account and contacts and projects and payments items. Payment items can be edited.

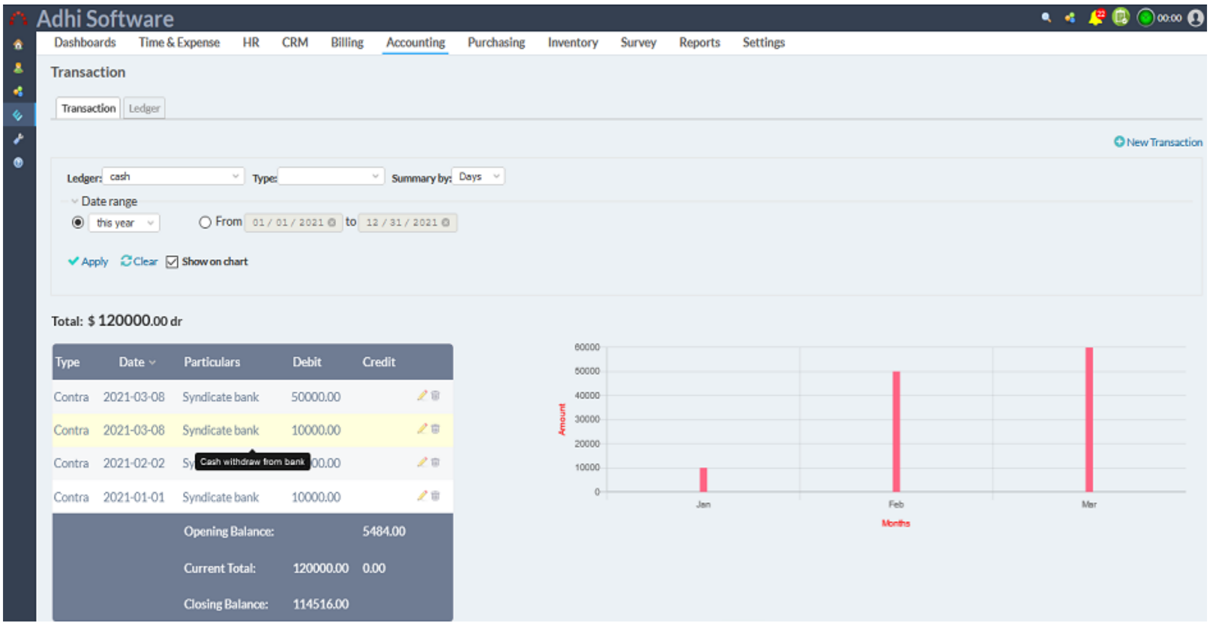


# 7 Accounting

This module allows the Accounting user to create/update the financial transaction and Ledgers.

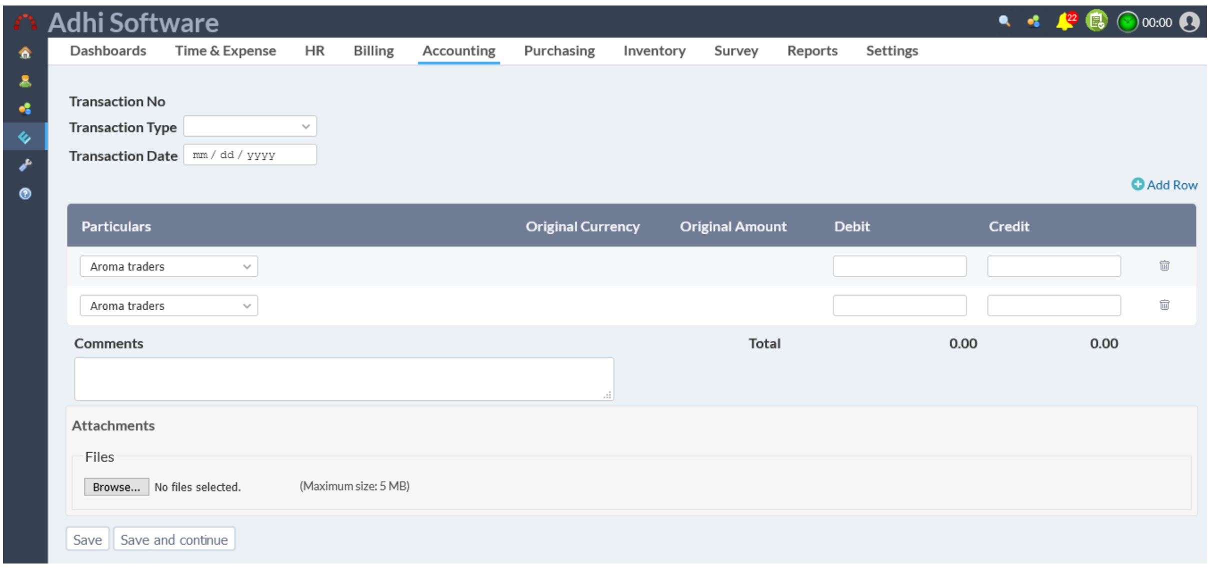
## 7.1 GL Transaction

The list page by default displays the transaction details for the current month. Date range and ledgers filters can be applied to the list page. Show the chart on Click the particular ledger.
View description on mouseover.




### 7.1.1 New/Edit GL Transaction

The transaction edit page displays the transaction details such as transaction number, type, date, comments and ledgers. Transaction details can be edited and new transaction details can be added also. we can attach file in the accounting under attachments.

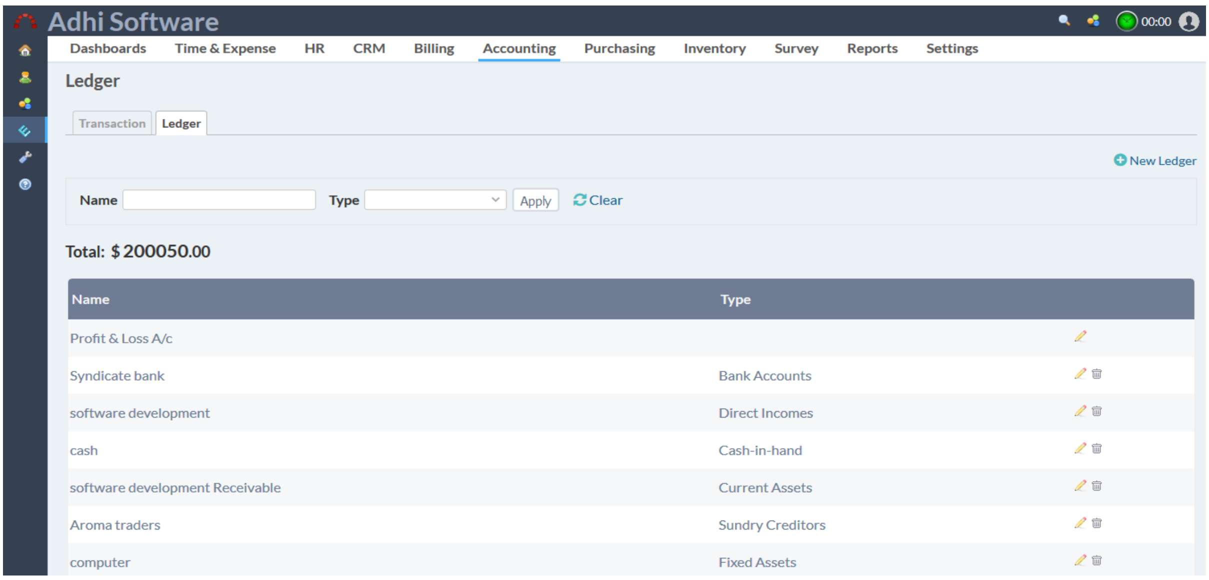


Following are some validation rules used when entering a GL transaction.

| # | Transaction Type | Cr | Dr | Rule |
|---|---|---|---|---|
| 1 | Contra | Bank Ac / Cash | Bank Ac / Cash | Contra should contain bank and cash ledgers only. |
| 2 | Payment | Bank Ac / Cash | | Payment should have bank or cash as Credit. |
| 3 | Receipt | | Bank Ac / Cash | Receipt should have bank or cash as Debit |
| 4 | Purchase | Sundry creditors Sundry debtors, Bank Ac / Cash | Purchase Group | Purchase should have sundry creditors/debtors, cash and bank as credit and purchase group as debit |
| 5 | Sales | Sales Group | Sundry creditors Sundry debtors, Bank Ac / Cash | Sales should have sundry creditors/debtors, cash and bank as debit and sales group as credit |
| 6 | Credit note | Sundry creditors Sundry debtors, Bank Ac / Cash | Except Bank Ac / Cash | Credit note should have sundry creditors/debtors, cash and bank as credit and should not have cash and bank as debit. |
| 7 | Debit note | Except Bank Ac / Cash | Sundry creditors Sundry debtors, Bank Ac / Cash | Debit note should have sundry creditors/debtors, cash and bank as debit and should not have cash and bank as credit. |

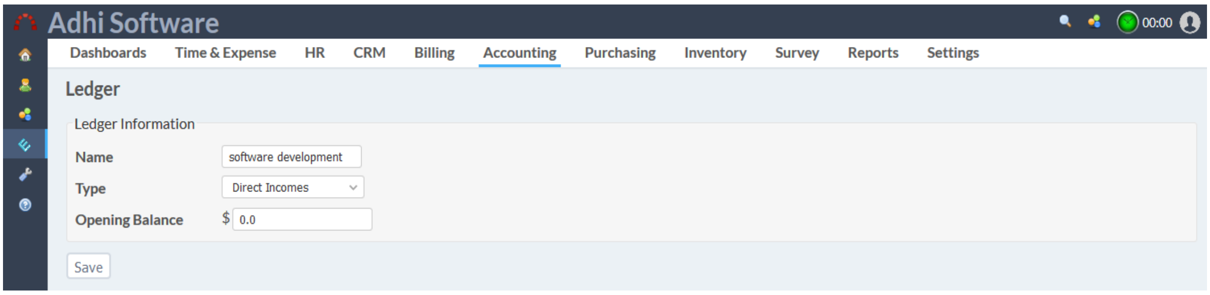
### 7.2 Ledgers

The list page by default displays the list of ledgers. Ledger name wildcard search and ledger type filters can be applied to the list page. Ledger can be removed.



### 7.2.1 New/Edit Ledger

To add a new tax click on “New Tax” hyperlink

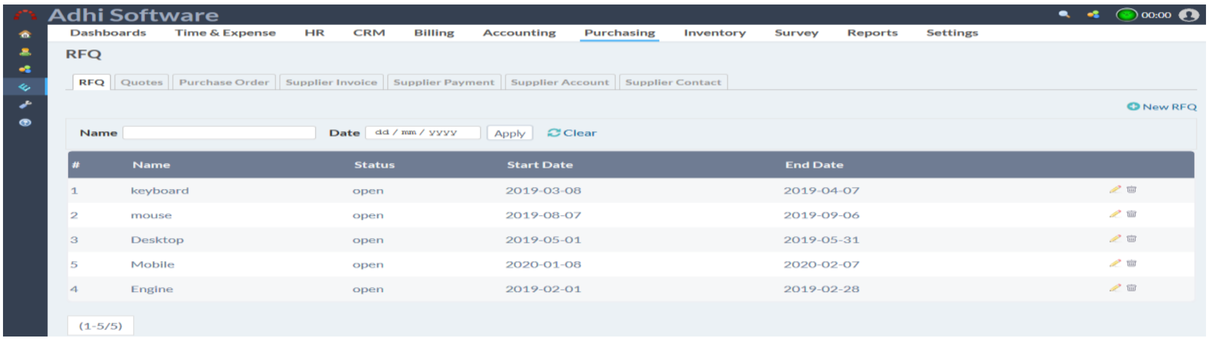



# 8 Purchasing

This module allows the Purchasing users and admin to create the Request for Quote (RFQ), quote, and purchase order.. A new supplier invoice can be accepted against a purchase order and a payment can be made against the supplier invoice.

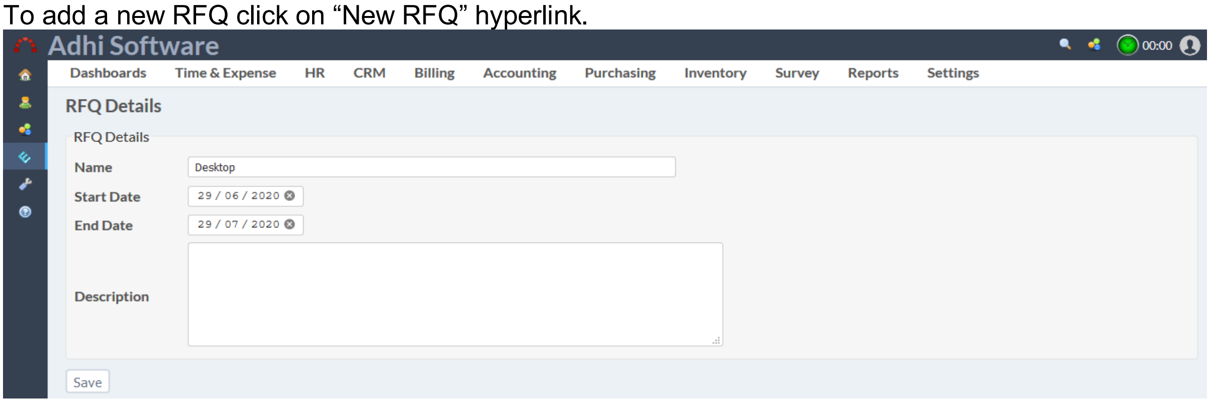
## 8.1 RFQ

The list page by default displays the list of RFQ. RFQ name wildcard search and Date filters can be applied to the list page. RFQ can be removed.



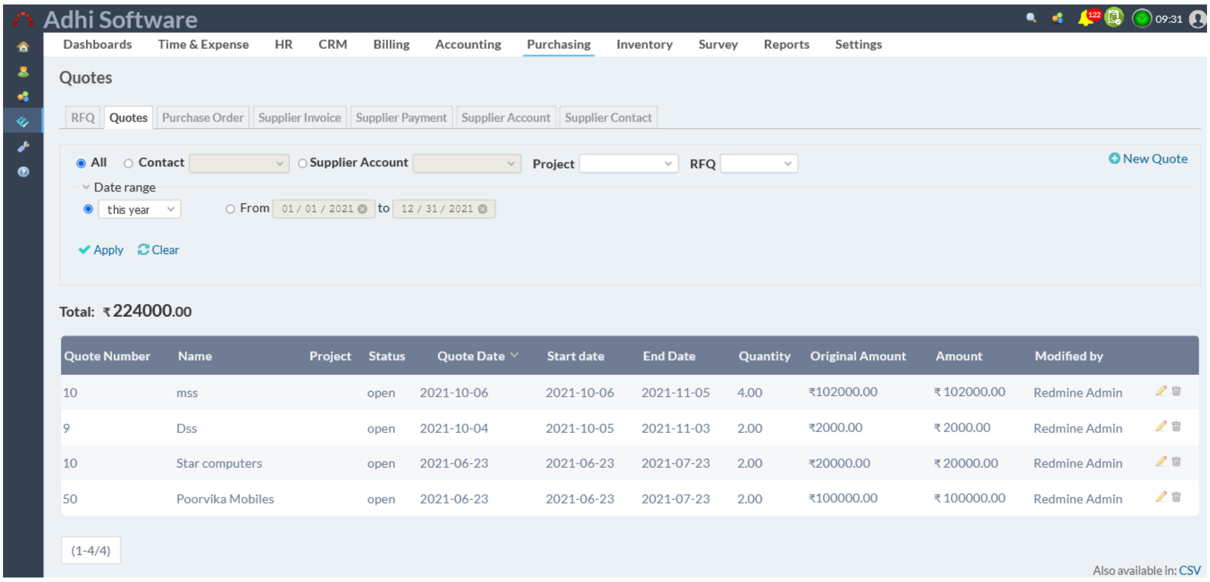

### 8.1.1 New/Edit RFQ

To add a new RFQ click on “New RFQ” hyperlink.



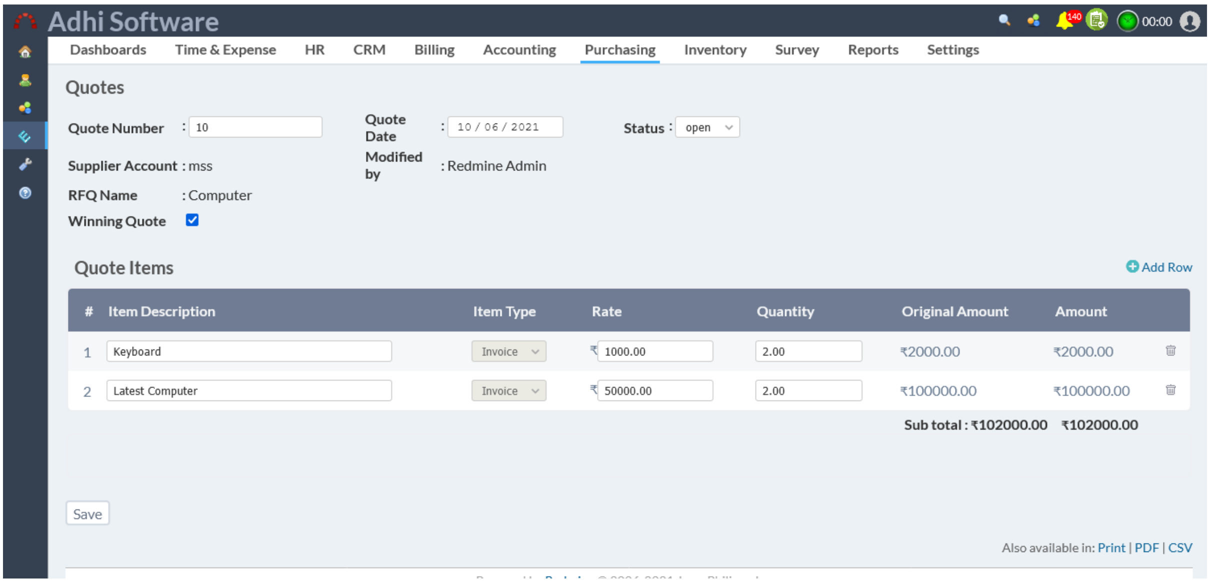
## 8.2 Quote

A quote is a proposal by a supplier to supply goods to a prospective customer. It will have items details with specification and price.The list page by default displays the list of quote. Date range, supplier account, contact, project and RFQ filters can be applied to the list page. Quote can be removed.




### 8.2.1 New/Edit Quote

To add a new Quote click on “New Quote” hyperlink. Quote can be edited to enable/disable the winning quote. If a quote is accepted for purchase it should be marked as a winning quote. Quotes can be exported as CSV and PDF.




### 8.2.2 Print Quote

A quote can be printed using the print link, it will open up the printer friendly Quote report.

**QUOTES**

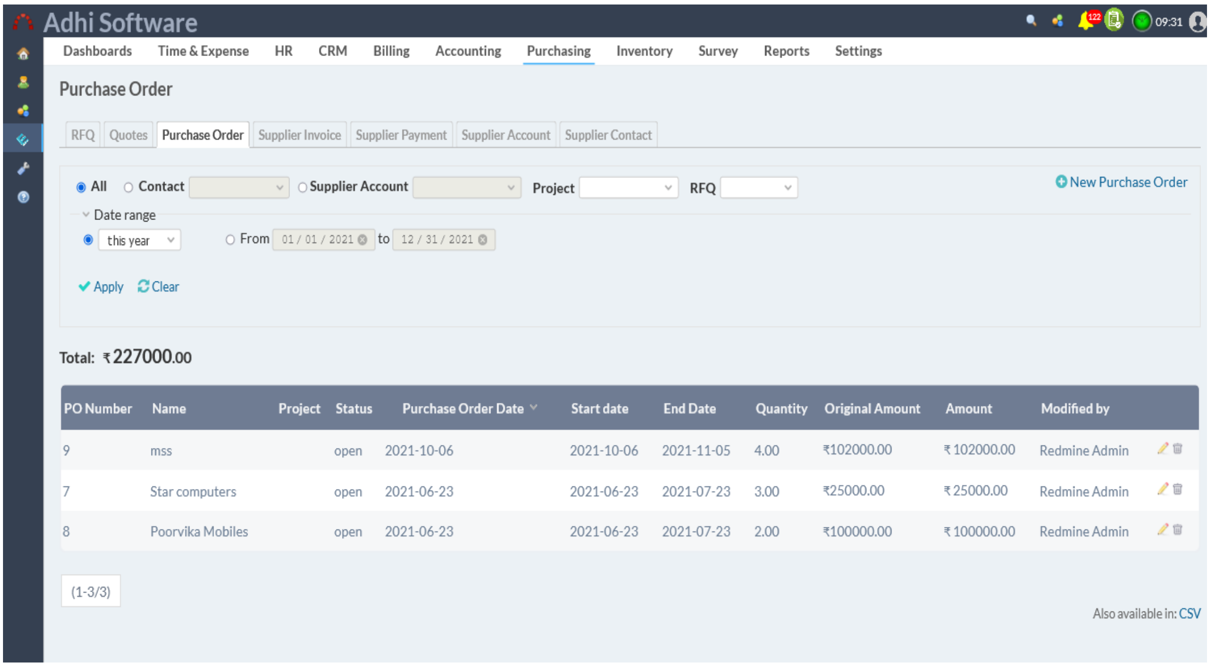
| Name & Address of the Supplier | Name & Address of the Customer | Quote Number | | Quote Date | | |
|---|---|---|---|---|---|---|
| DELL | ADHI SOFTWARE PVT LTD<br>SIPCOT IT PARK,<br>SIRUSERI,<br>CHENNAI. | 5 | | June 30, 2017 | | |
| **Contract / Purchase Order / Work Order / Agreement Date & Reference** | | | | **Period** | | |
| Keyboard(2) from June 30, 2017 to June 30, 2017 | | | | June 30, 2017 to July 30, 2017 | | |
| **Item Description** | **Billing Type** | | **Rate** | **Quantity** | **Currency** | **Amount** |
| DELL Keyboard V1.0 | | | 43.00 | 434.00 | $ | 18662.00 |
| GST | | | 1862.00 | 1.00 | $ | 1862.00 |
| | | | **Total** | **435.00** | **$** | **20524.00** |
| | | | **Grand Total** | **435.00** | **$** | **20524.00** |
| **Amount in Words** | Twenty thousand five hundred and twenty four only. | | | | | |
| **Quote - 1** | | QUOTE VAL - 1 | | | | |

**Place :**
**Date :**

**Authorized Signatory**

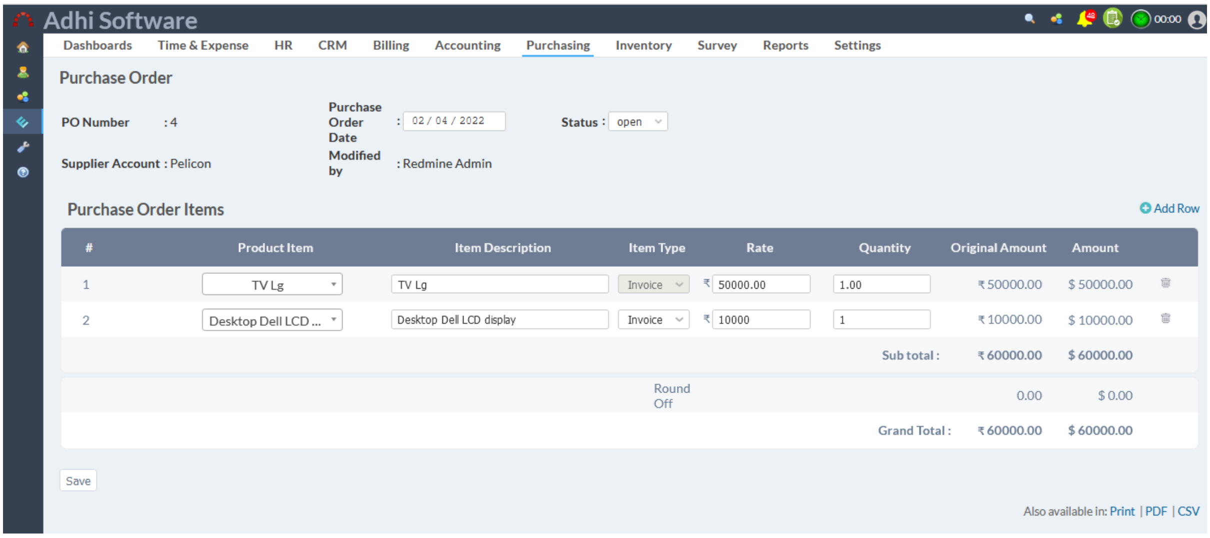
## 8.3 Purchase Order

A Purchase order is nothing but a confirmation from the customer purchasing goods from a supplier, it should be created from a winning quote. The list page by default displays the list of purchasing order. Date range, supplier account, contact, project and RFQ filters can be applied to the list page. Purchasing order can be removed.




### 8.3.1 New/Edit Purchase order

To add a new purchase order click on “New purchase order” hyperlink. A purchase order should be created from RFQ and winning quote. Purchase order items can be populated from winning quote. Purchase order items can be exported as CSV and PDF.




### 8.3.2 Print Purchase Order

A purchase order can be printed using the print link, it will open up the printer friendly purchase order report.

**PURCHASE ORDER**

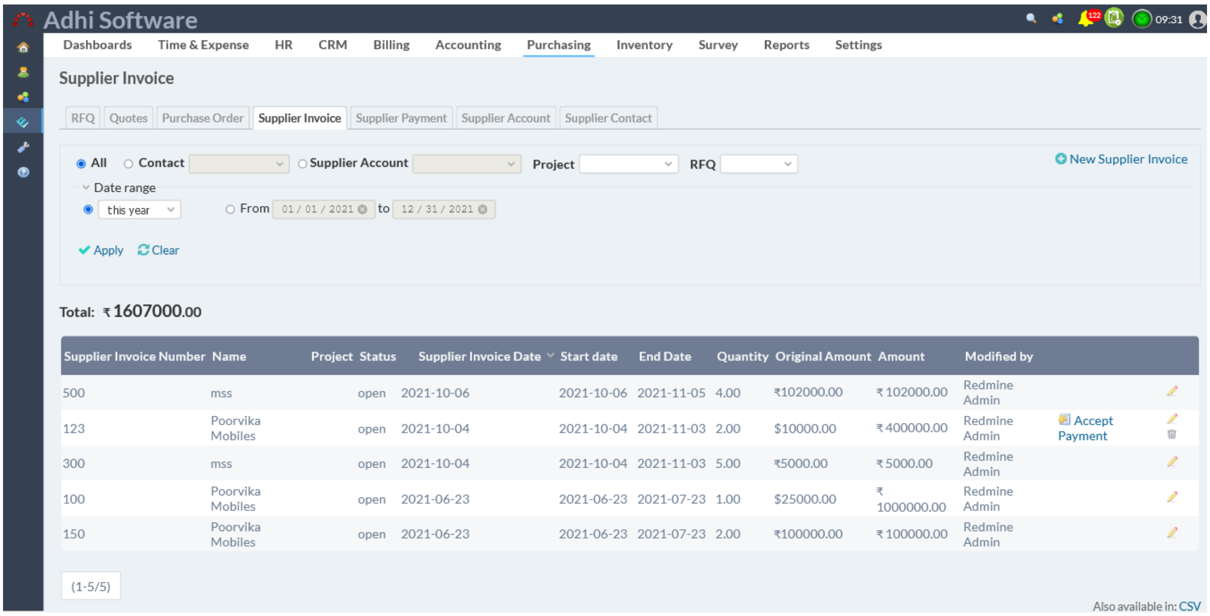
<table>
<tr><th>Name & Address of the Supplier</th><th>Name & Address of the Customer</th><th colspan="3">PO Number</th><th colspan="3">Purchase Order Date</th></tr>
<tr><td>Pelicon<br>121<br>xyz street<br>chennai tamilnadu<br>600026<br>india</td><td>Chennai<br>1st cross Road<br>Sirucheri IT Park<br>Navalur TamilNadu<br>600130<br>India</td><td colspan="3">4</td><td colspan="3">February 04, 2022</td></tr>
<tr><th colspan="5">Contract / Purchase Order / Work Order / Agreement Date & Reference</th><th colspan="3">Period</th></tr>
<tr><td colspan="5">1001 - February 04, 2022</td><td colspan="3">February 04, 2022 to March 06, 2022</td></tr>
<tr><th>Item Description</th><th colspan="2">Billing Type</th><th colspan="2">Rate</th><th>Quantity</th><th>Currency</th><th>Amount</th></tr>
<tr><td>TV Lg</td><td colspan="2"></td><td colspan="2">50000.00</td><td>1.00</td><td>₹</td><td>50000.00</td></tr>
<tr><td></td><td colspan="2"></td><td colspan="2">Total</td><td>1.00</td><td>₹</td><td>50000.00</td></tr>
<tr><td></td><td colspan="2"></td><td colspan="2">Grand Total</td><td>1.00</td><td>₹</td><td>50000.00</td></tr>
<tr><td>Amount in Words</td><td colspan="7">Fifty thousand only.</td></tr>
</table>

**Place :** **Authorized Signatory**

**Date :**

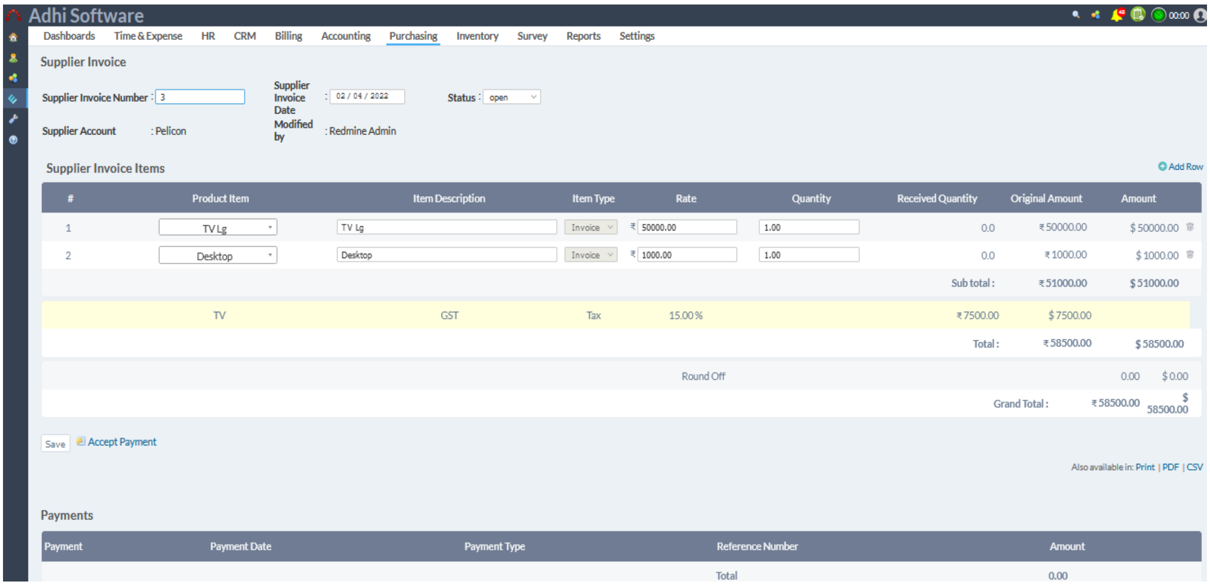
## 8.4 Supplier Invoice

Once the goods are received from the supplier, the supplier invoice can be entered into the system.. The list page by default displays the list of supplier invoice. Date range, supplier account, contact, project and RFQ filters can be applied to the list page. Supplier invoice can be removed.




## 8.4.1 New/Edit Supplier Invoice

To add a new supplier invoice click on “New supplier invoice” hyperlink.. Supplier invoice items can be populating from purchase order. Supplier invoice can be exported as CSV and PDF.




## 8.4.2 Print Supplier Invoice

A supplier invoice can be printed using the print link, it will open up the printer friendly supplier invoice report.

SUPPLIER INVOICE

| Name & Address of the Supplier | Name & Address of the Customer | Supplier Invoice Number | Supplier Invoice Date | | |
|---|---|---|---|---|---|
| Pelicon<br>121<br>xyz street<br>chennai tamilnadu<br>600026<br>india | Chennai<br>1st cross Road<br>Sirucheri IT Park<br>Navalur TamilNadu<br>600130<br>India | 3 | February 04, 2022 | | |
| **Contract / Purchase Order / Work Order / Agreement Date & Reference** | | | **Period** | | |
| 4 - February 04, 2022 | | | February 04, 2022 to March 06, 2022 | | |

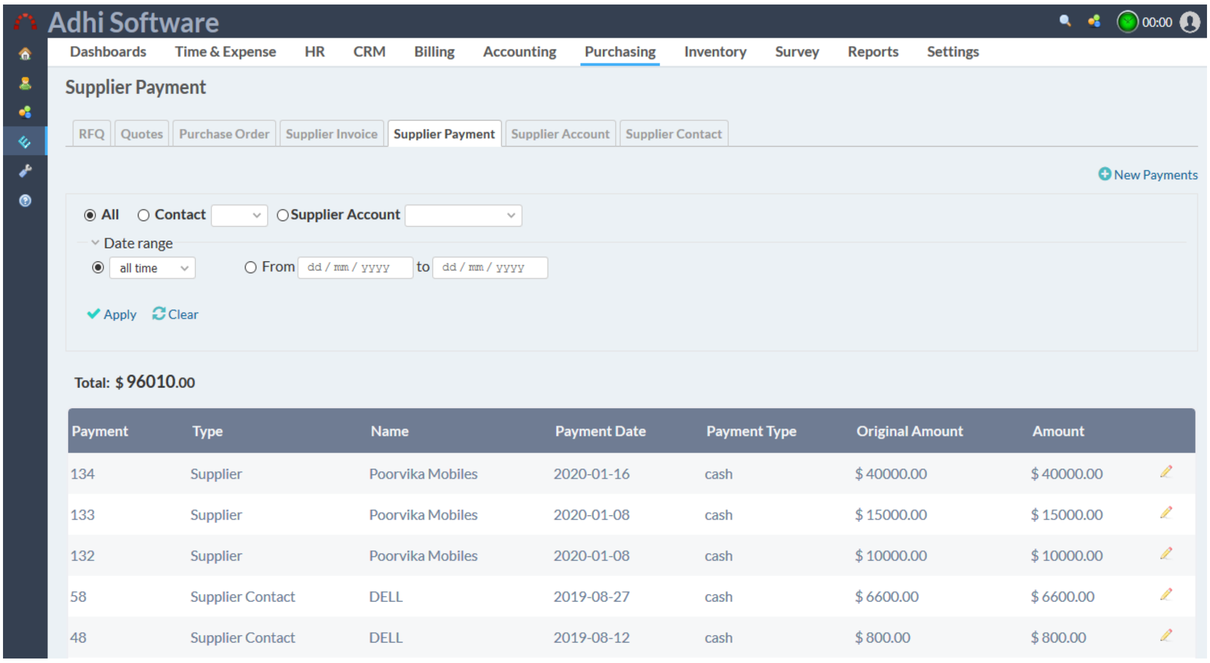
| Item Description | Billing Type | Rate | Quantity | Currency | Amount |
|---|---|---|---|---|---|
| TV Lg | | 50000.00 | 1.00 | ₹ | 50000.00 |
| GST | | 15.00% | | ₹ | 7500.00 |
| | | **Total** | **1.00** | **₹** | **57500.00** |
| Desktop | | 1000.00 | 1.00 | ₹ | 1000.00 |
| | | **Total** | **1.00** | **₹** | **1000.00** |
| | | **Grand Total** | **2.00** | **₹** | **58500.00** |
| **Amount in Words** | Fifty eight thousand five hundred only. | | | | |

**Place :**
**Date :**

**Authorized Signatory**

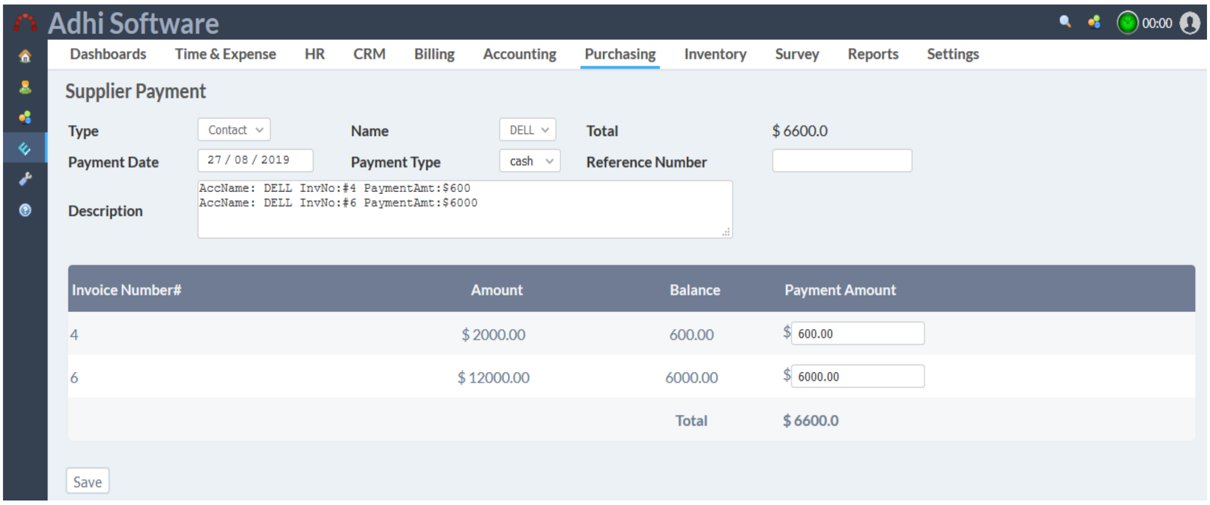
## 8.5 Supplier Payment

Supplier payments can be made against a supplier invoices, multiple supplier invoices can be paid by a single payment. The list page by default displays the payments for the current month. Contact, supplier account, project and Date range filters can be applied to the list page.



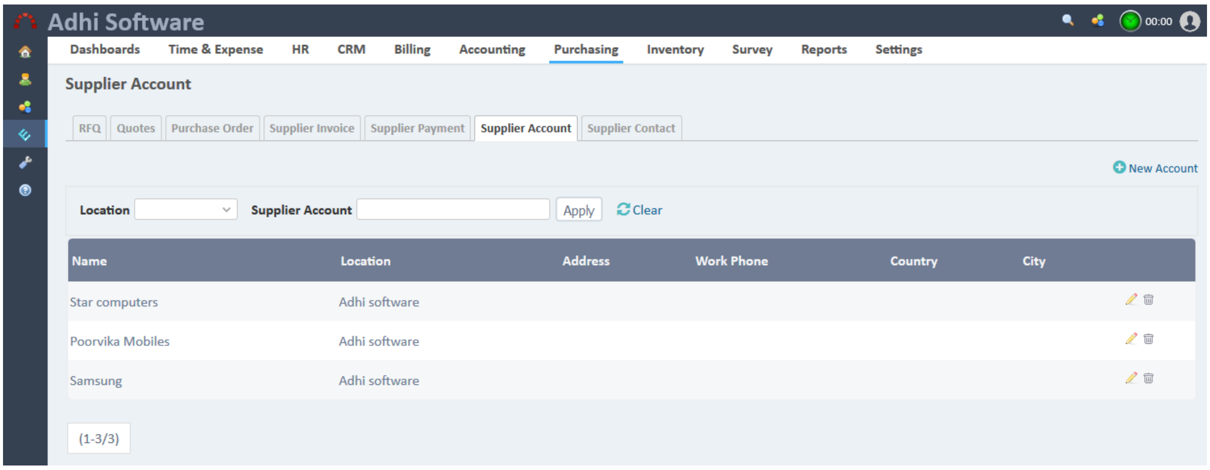

### 8.5.1 New/Edit Supplier Payment

The supplier payment edit page displays the payment details such as payment date, payment type, reference number, account and contacts and projects and payments items. Payment items can be edited.



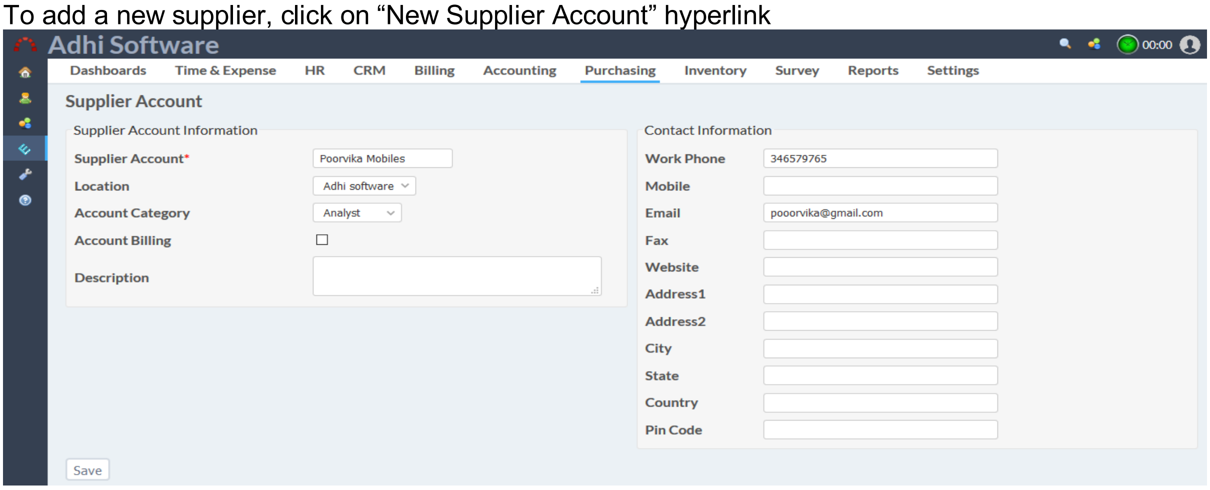

## 8.6 Supplier Accounts

Supplier accounts are Vendors for whom the goods are purchased, there can be multiple purchase order for a supplier. The list page by default displays the list of supplier accounts. Supplier Account name wildcard search can be applied to the list page.



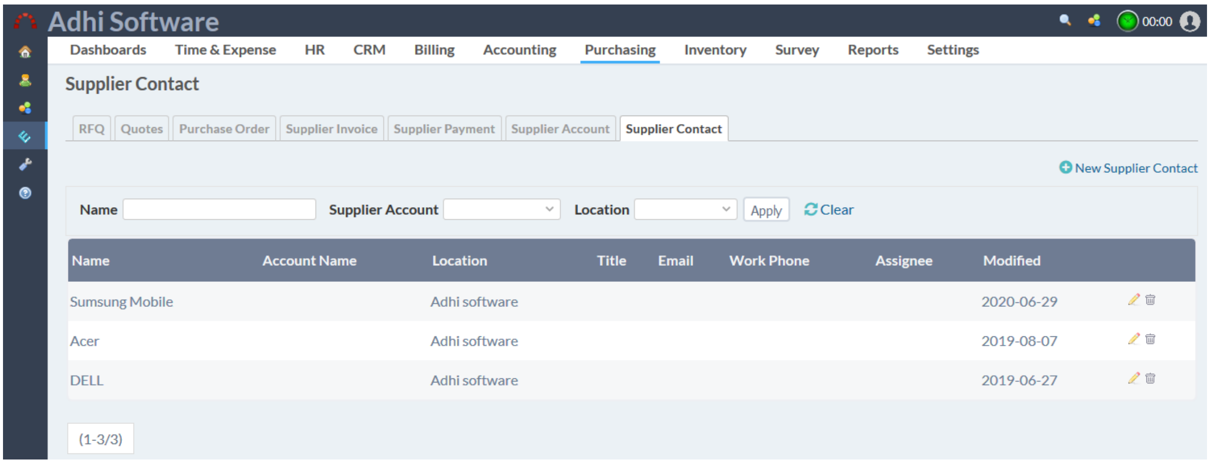

### 8.6.1 New/ Edit Supplier Account

To add a new supplier, click on "New Supplier Account" hyperlink



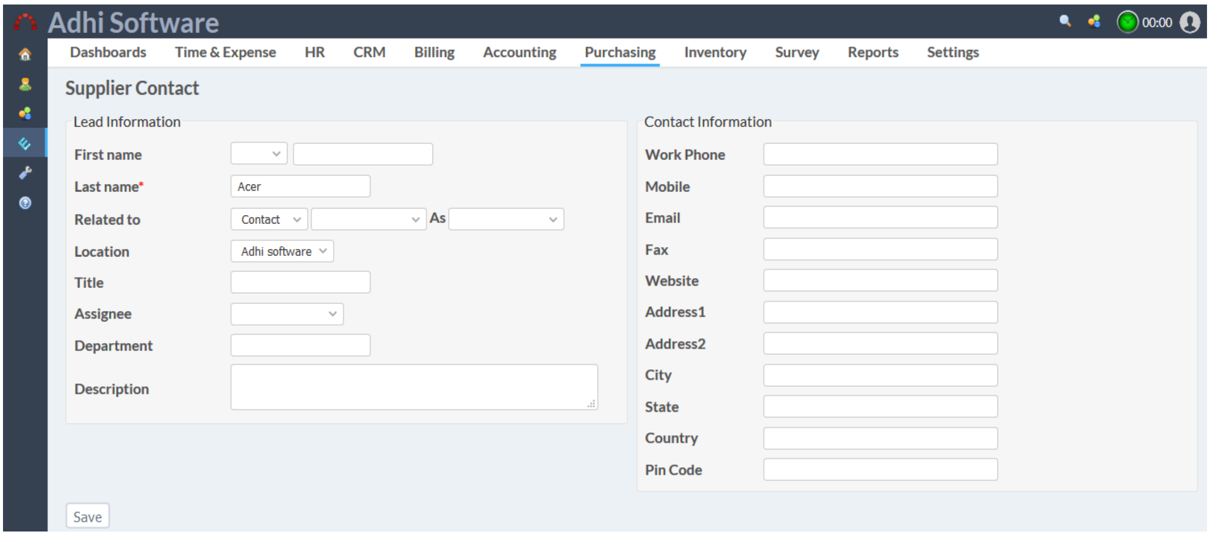
## 8.7 Supplier Contact

A supplier contact is an individual they may work for an Supplier. The list page by default displays the list of Supplier Contacts. Supplier Contact name wildcard search and account filters can be applied to the list page. Supplier Contact can be removed.




### 8.7.1 New/Edit Supplier Contact

To add a new Supplier Contact, click on "New Contact" hyperlink.

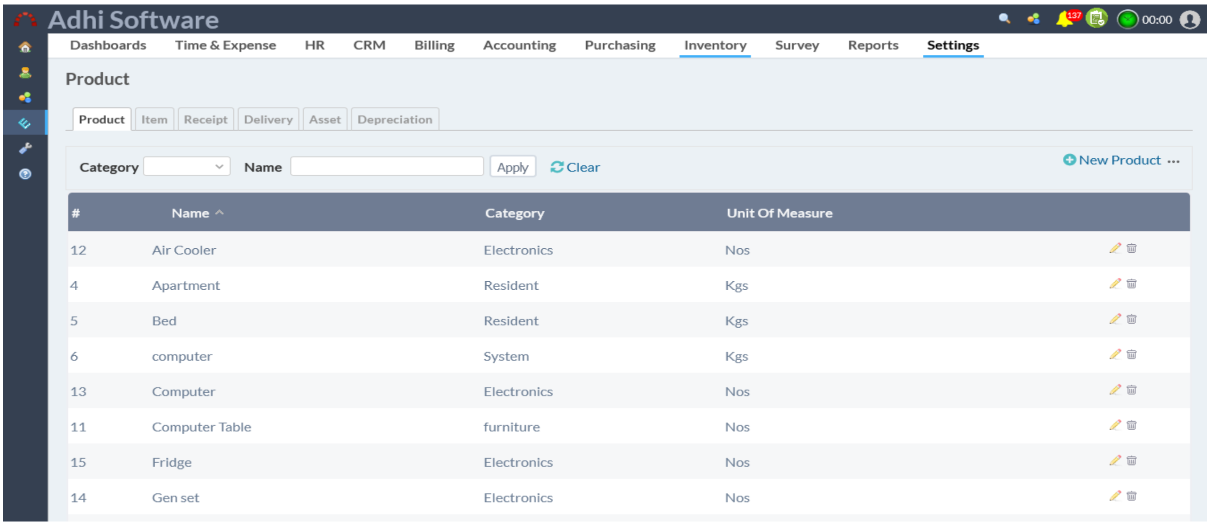



# 9 Inventory

This module allows the Inventory users and admin to create the Product, Items, shipments and log materials. A new shipment can be created against a supplier invoice.

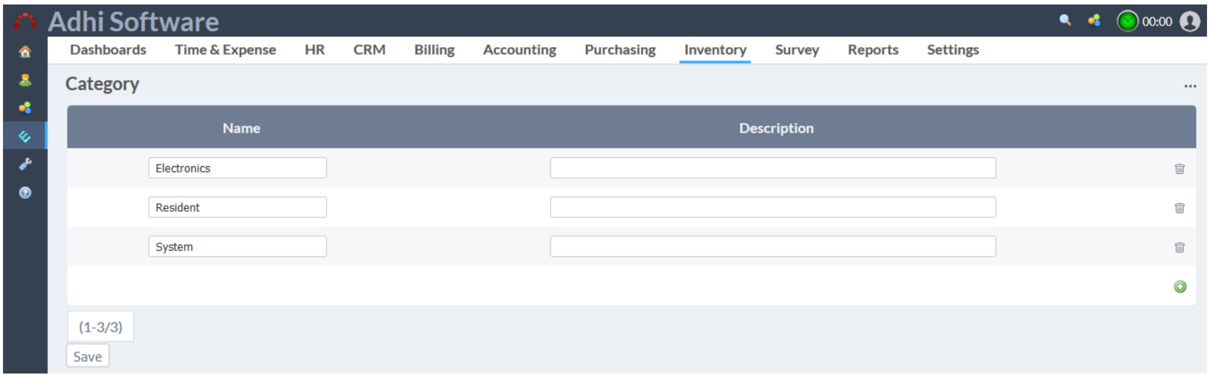
## 9.1 Products

The list page by default displays the list of Products. Products name and category filters can be applied to the list page. Products can also be removed.



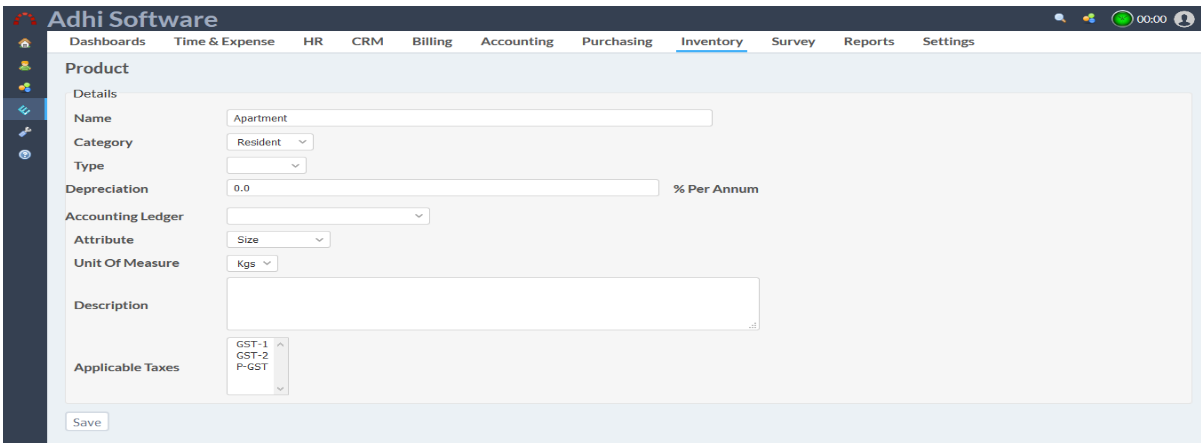

### *9.1.1 Category*

To add a new product category click on “category” hyperlink. The list page by default displays the list of categories. The category can be added/updated in the list page. Category can also be removed.



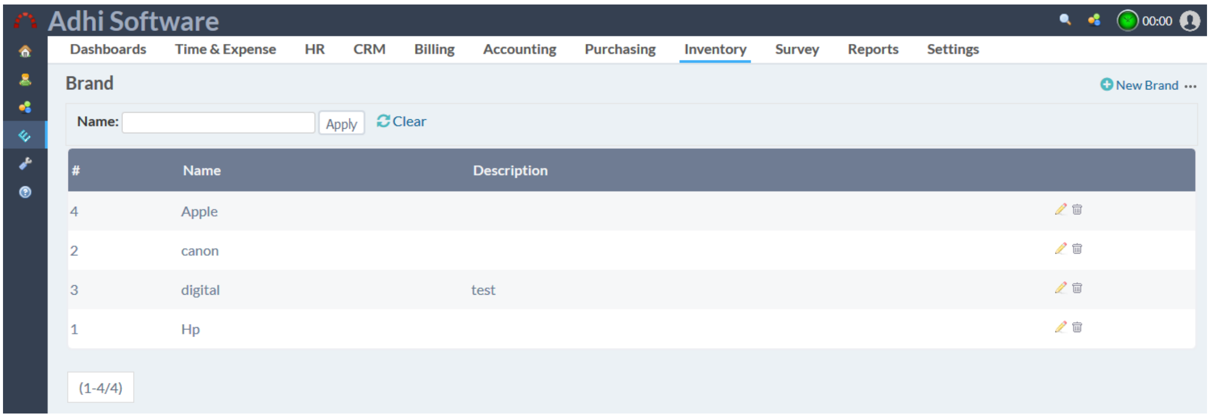

### 9.1.2 New/Edit Product

To add a new Product, click on “New Product” hyperlink. Product can be categorized as Inventory or Asset and apply depreciation percentage for asset product.



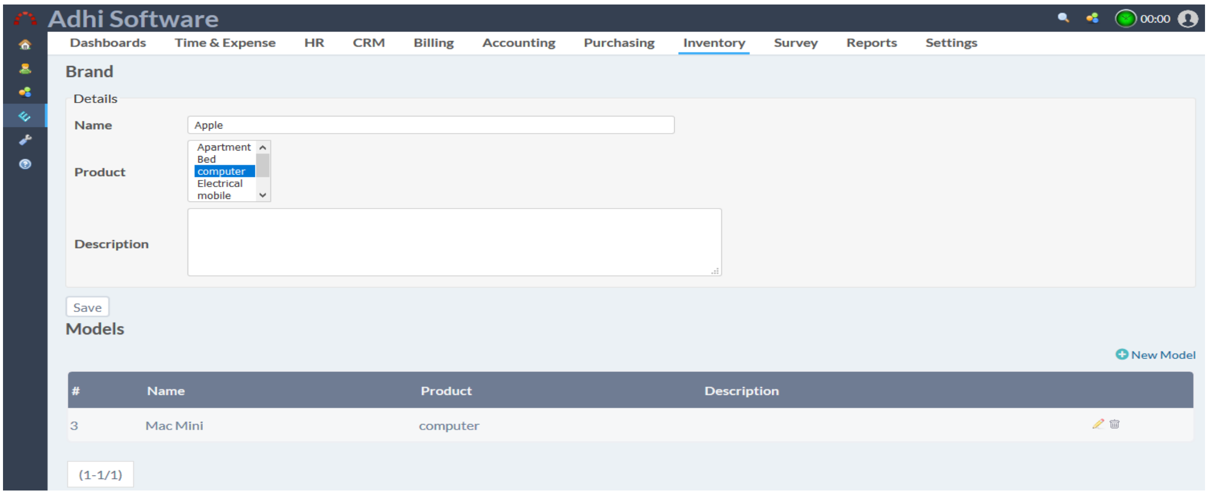
## 9.2 Brand

The list page by default displays the list of brands. Brand name (wildcard search) filter can be applied to the list page. Brand can also be removed.



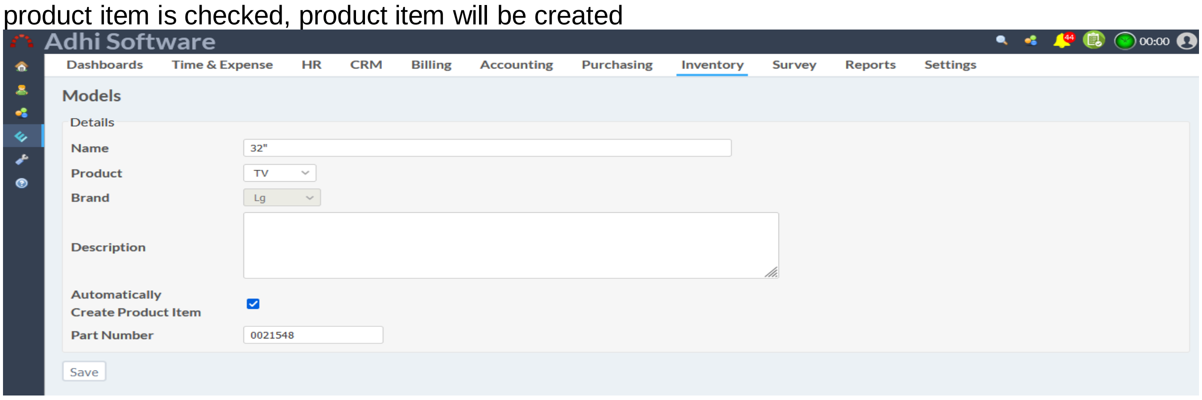

### 9.2.1 New/ Edit Brand

To add a new Brand, click on "New Brand" hyperlink. Models are listed on the Appropriate brand detail page. Model can also be removed.




### *9.2.2 New Model*

To add a new model click on "New Model" hyperlink in the brand detail page. If automatically create product item is checked, product item will be created

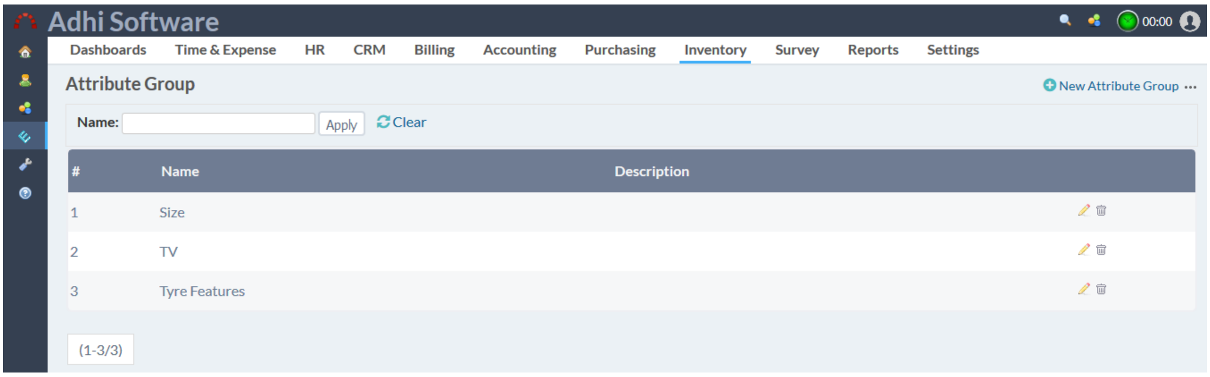


## 9.3 Attributes

These are attributes of a product, they are groped as attribute gropups, for example

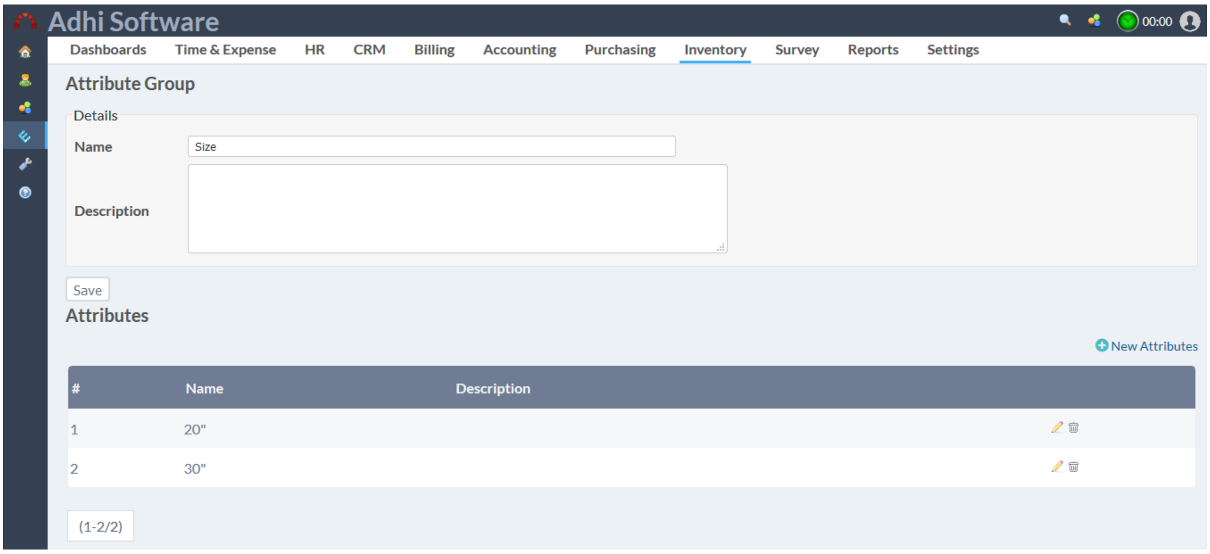
| $ | Attribute Group | Attributes |
|---|---|---|
| 1 | Size | S, M, L, XL, XXL |
| 2 | Style | S, R, L |
| 3 | Width | 12", 14", 16" |

The list page by default displays the list of Attribute groups. Attribute group name (wildcard search) filter can be applied to the list page. Attribute group can also be removed.



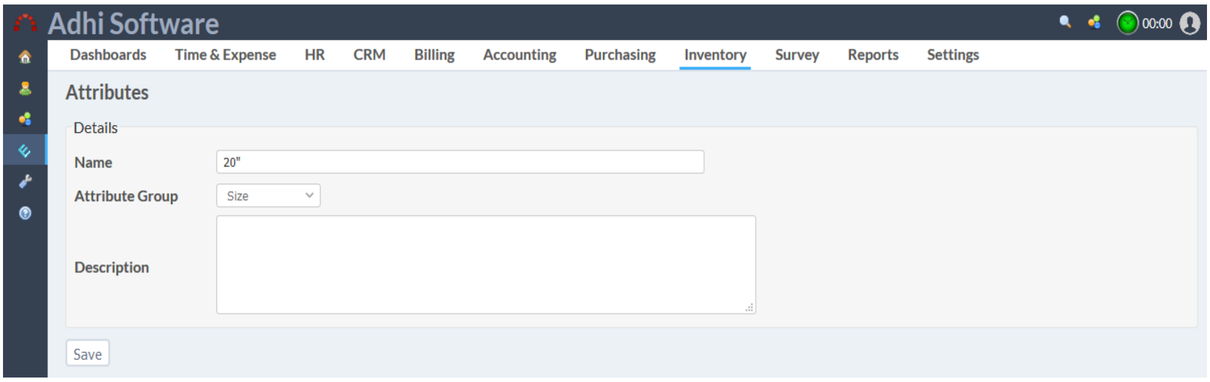

9.3.1 New/Edit Attribute group

To add a new Attribute, click on "New Attribute Group" hyperlink. Attribute are listed on the Appropriate Attribute group detail page.



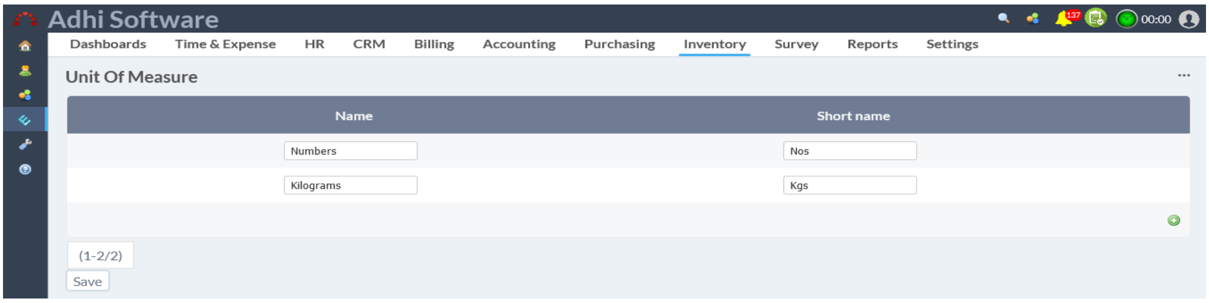

### *9.3.2 New Attribute*

To add a new attribute click on "New Attribute" hyperlink in the Attribute group detail page.



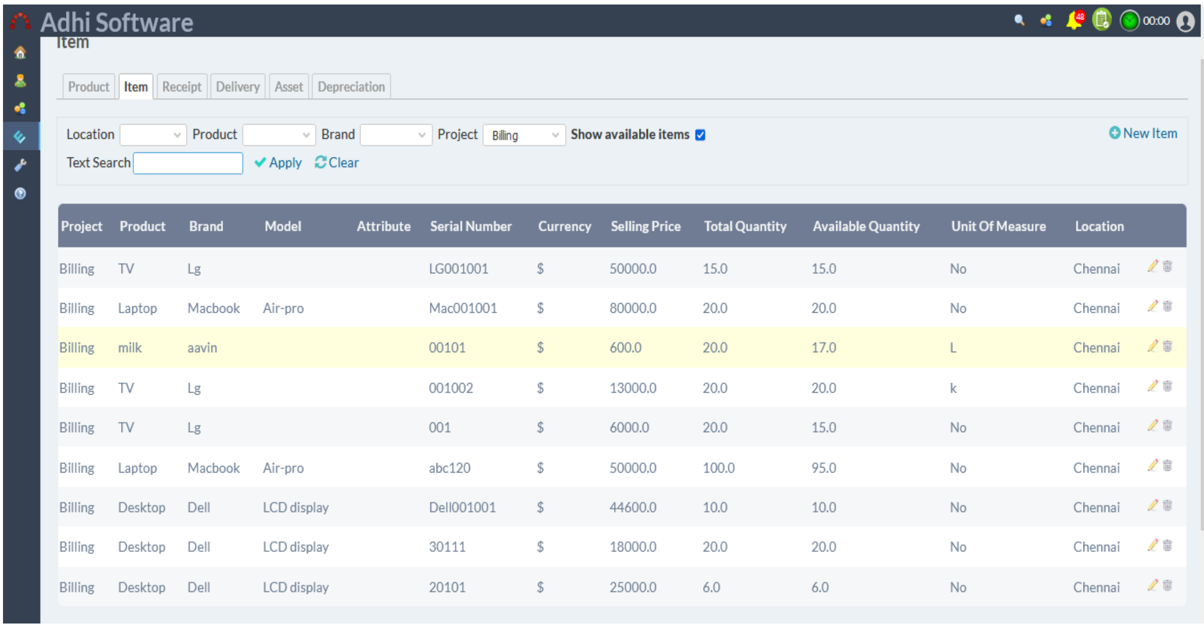

## 9.4 Unit of Measure

The list page by default displays the list of Unit of Measure (UOM). The UOM can be added/updated in the list page. UOM can also be removed.



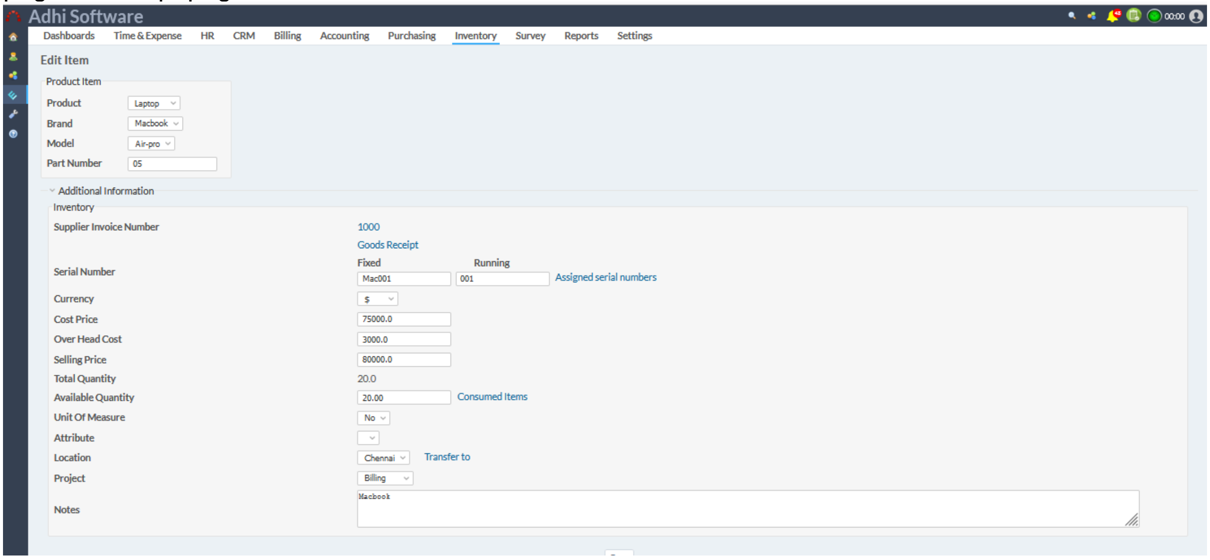

## 9.5 Product Items

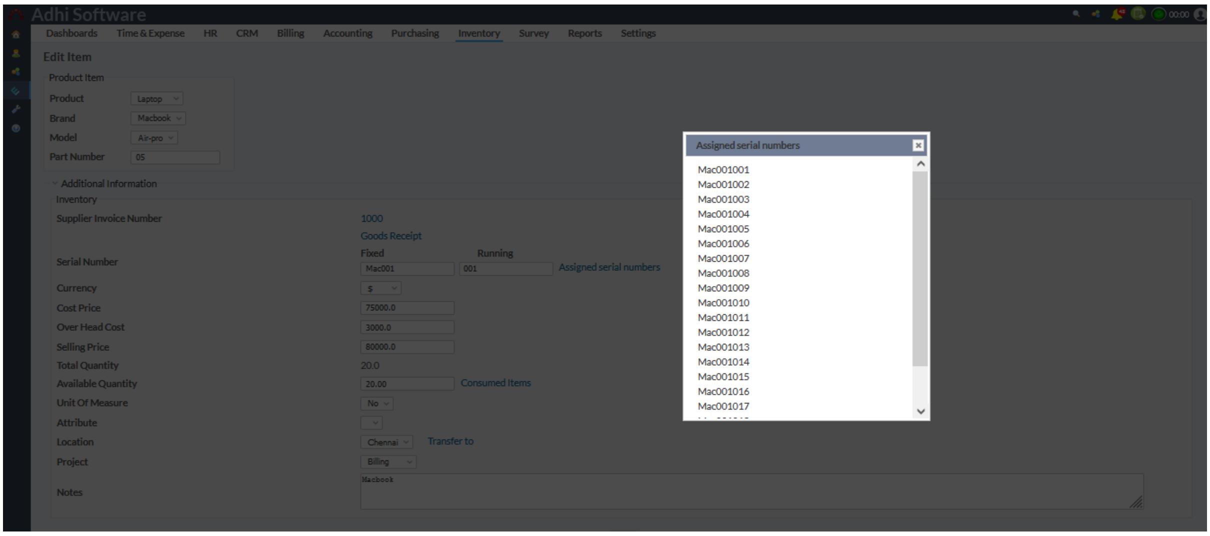
Product item is an instance of the product belonging to a particular brand and model, with a unique set of attributes. Products, brand, Text search and show available items filters can be applied to the list page. Product Items can also be removed.



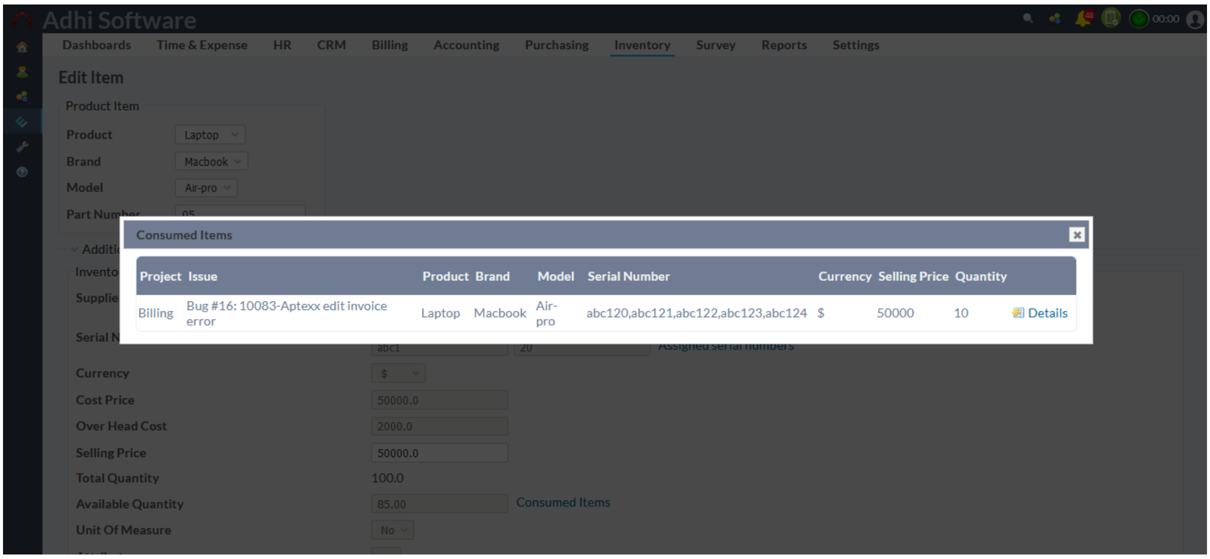
### 9.5.1 New/Edit Product Item

To add a new Product Item, click on “New Product Item” hyperlink. To transfer the items click on “Transfer To” hyper link in product item details page. The 'Assigned serial number' link will display the preview of serial numbers to be assigned for the product item. “Goods receipts" hyperlink redirect the page to receipt page.



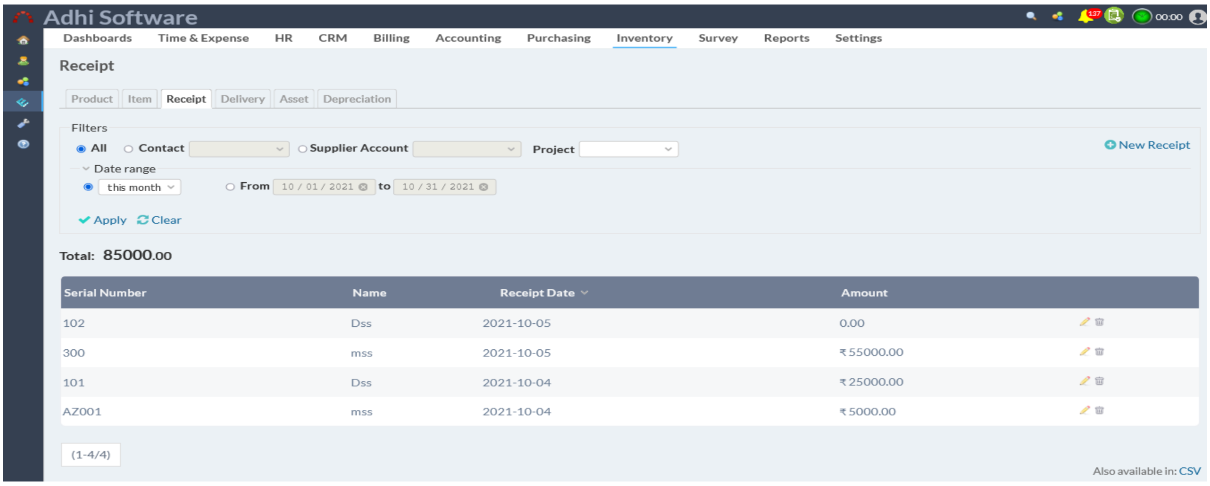




To show used material, Click the detail icon next to the quantity textbox, clicking on that opens a detailed pop up with the list of material used.



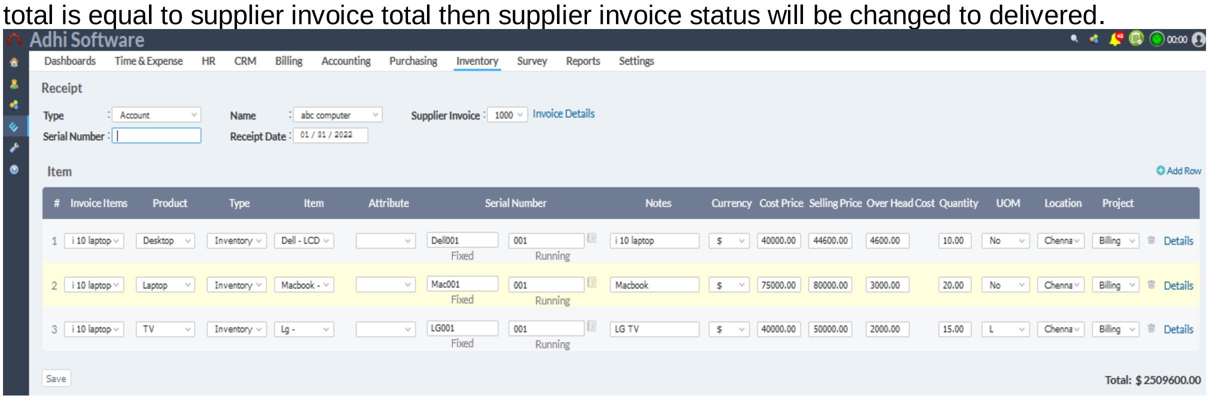

## 9.6 Receipts

A Receipt is a collection of product items received from a supplier. Date range, supplier account and contact filters can be applied to the list page. Shipment can also be removed. Create the item with the combination of brand and model before using that in the shipment.



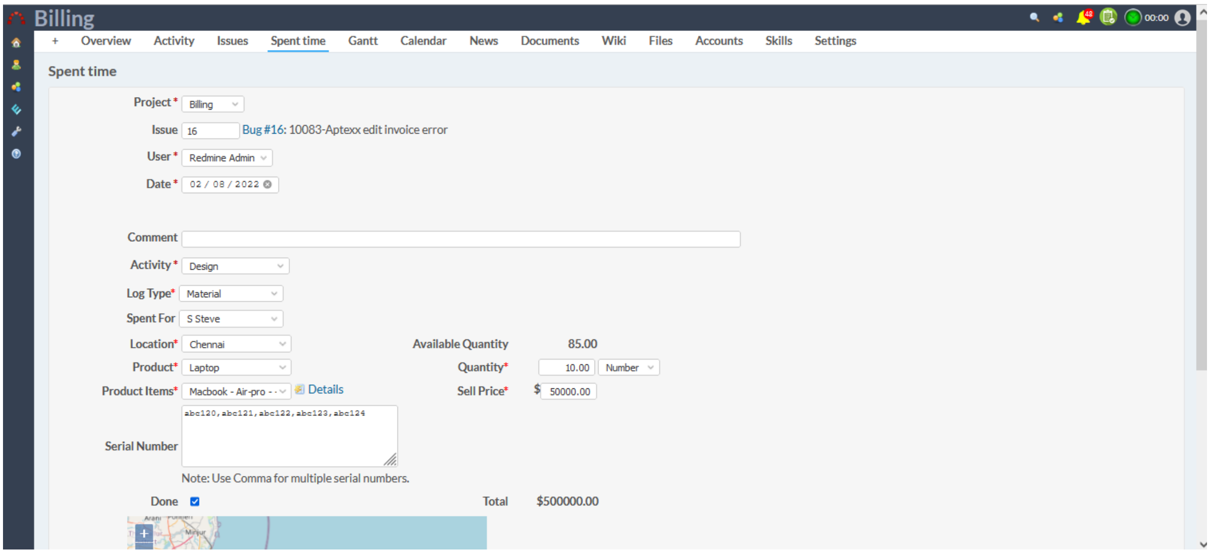

### 9.6.1 New/Edit Receipt

To add a new Receipt click on “New Receipt” hyperlink. A Receipt can be created from supplier invoice. "Details" hyperlink redirect the page to Inventory items /Asset page. When saving the receipt, if receipt total is equal to supplier invoice total then supplier invoice status will be changed to delivered.



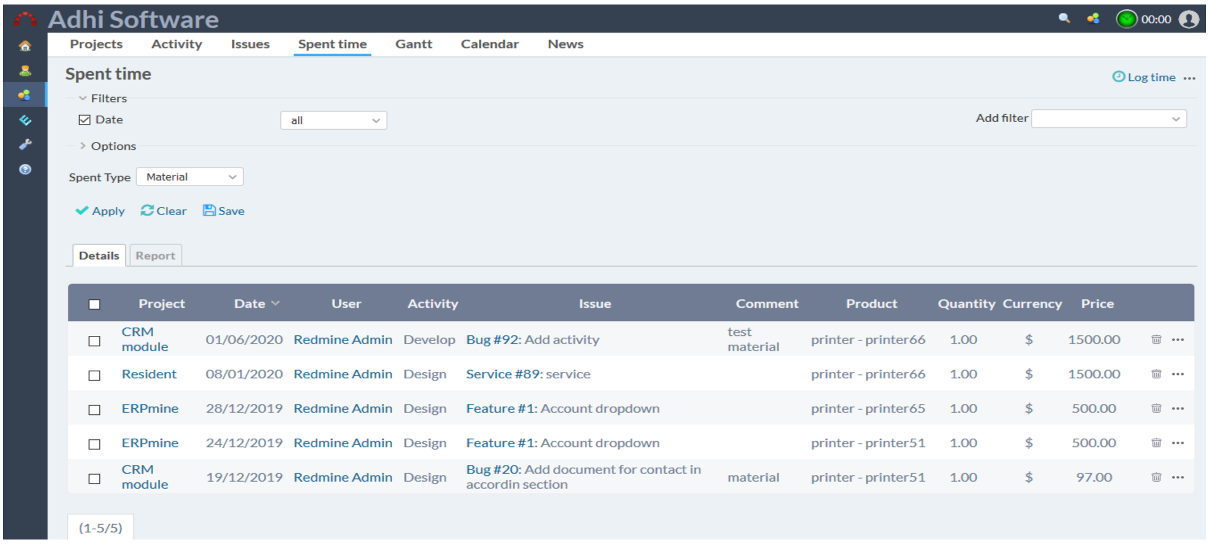
## 9.7 Log Material

The materials used on a task / issue can be logged using the log time page. Log Material page allows users to number of items used on a specific issue or project.

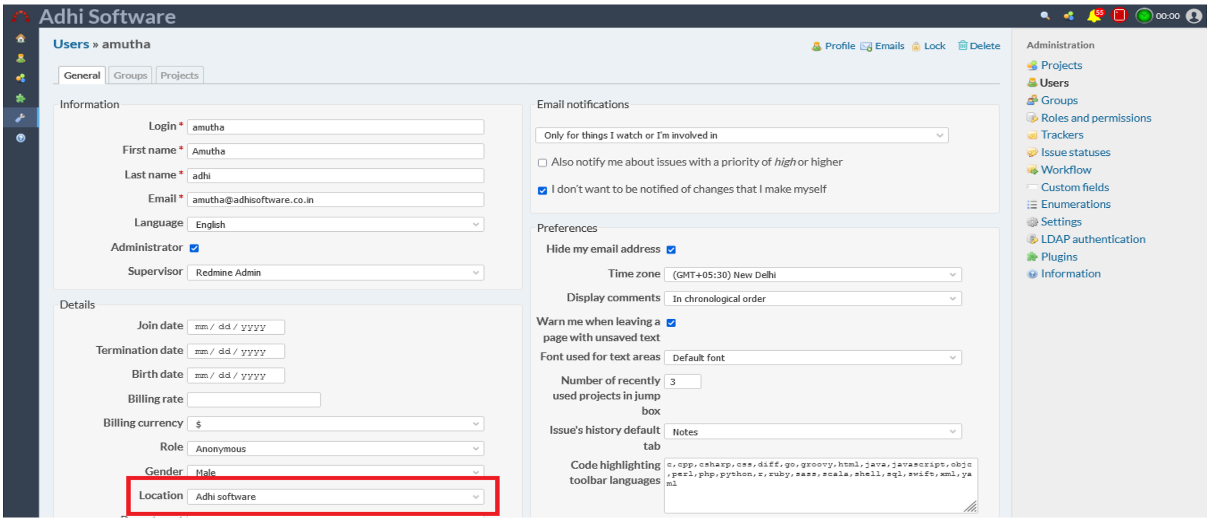


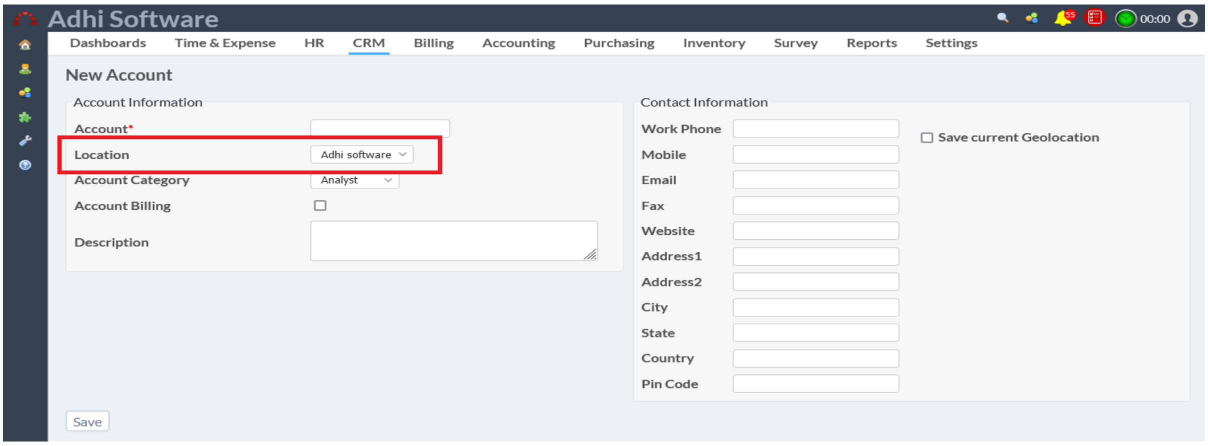

### *9.7.2 Material Log Details*

The spent material details can be viewed using the spent time page. The Material Details page is accessible from the Overview > Spent Time > Details menu.



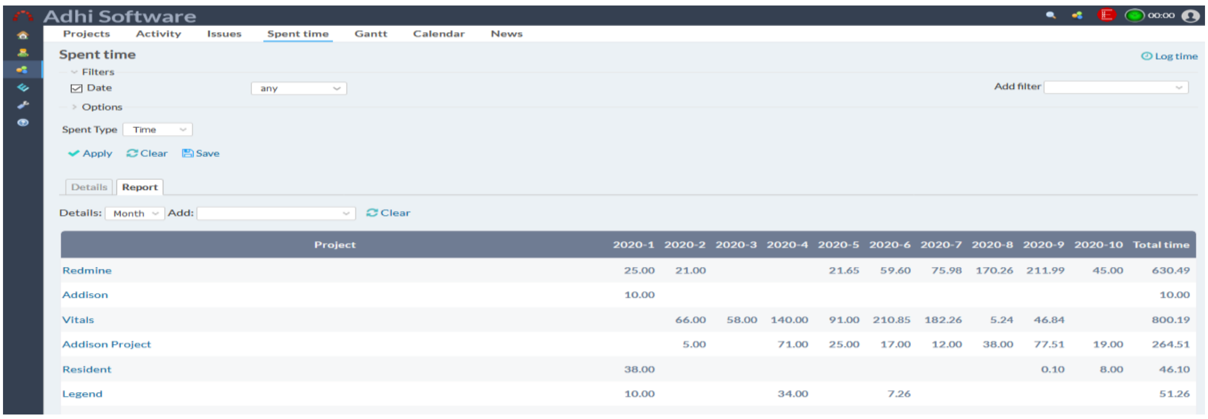

Account or contact location and the user location should be same, only then 'spent for' dropdown will populate. For material Entries to be included into an invoice, the 'Spent for" must be filled in




### 9.7.3 *Spent Material Report*

The Material report page is accessible from the Overview > Spent Time > Details menu. Product Item group by filters also added on the filter.

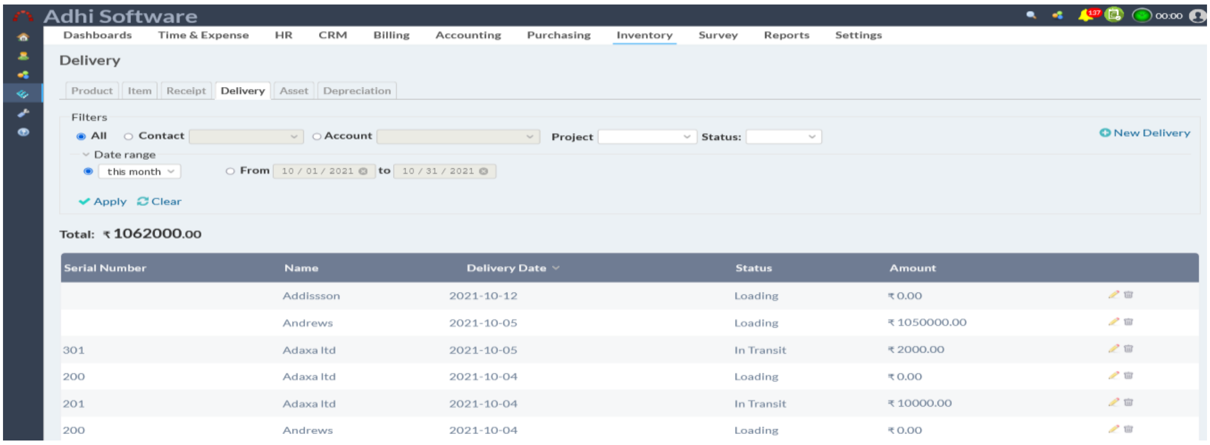



| Project | 2020-1 | 2020-2 | 2020-3 | 2020-4 | 2020-5 | 2020-6 | 2020-7 | 2020-8 | 2020-9 | 2020-10 | Total time |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Redmine | 25.00 | 21.00 | | | 21.65 | 59.60 | 75.98 | 170.26 | 211.99 | 45.00 | 630.49 |
| Addison | 10.00 | | | | | | | | | | 10.00 |
| Vitals | | 66.00 | 58.00 | 140.00 | 91.00 | 210.85 | 182.26 | 5.24 | 46.84 | | 800.19 |
| Addison Project | | 5.00 | | 71.00 | 25.00 | 17.00 | 12.00 | 38.00 | 77.51 | 19.00 | 264.51 |
| Resident | 38.00 | | | | | | | | 0.10 | 8.00 | 46.10 |
| Legend | 10.00 | | | 34.00 | | 7.26 | | | | | 51.26 |

# 10 Delivery

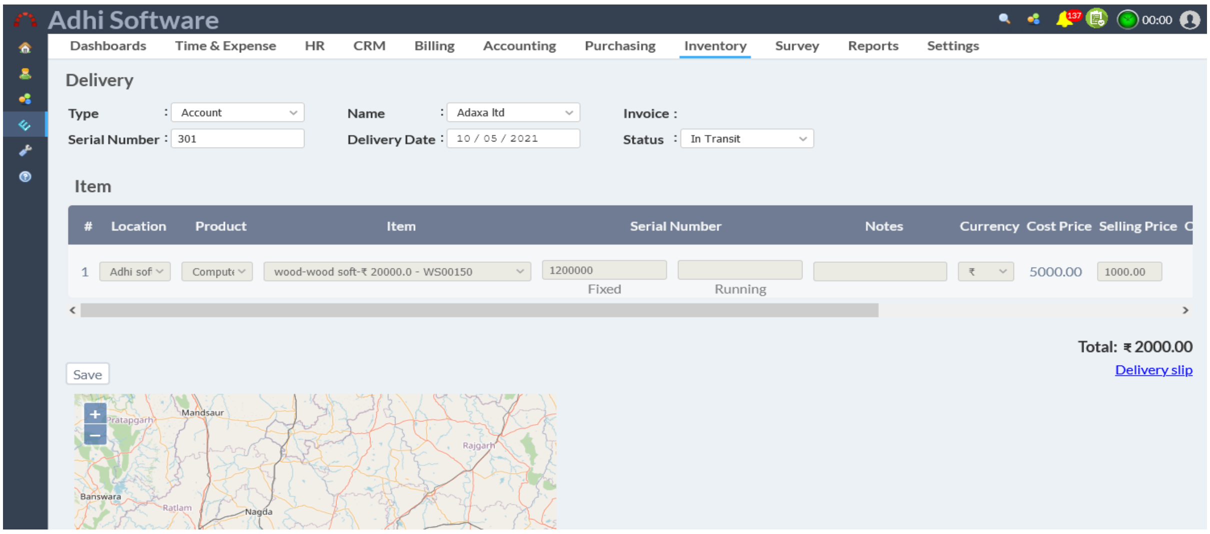
## 10.1 Delivery

Delivery feature will help to track the shipped products and also print delivery slip by the customer.




## 10.2 New/Edit Delivery

To add a new Delivery, click on "New Delivery" hyperlink. Status and location can be updated using the edit feature. The corresponding invoice can be attached to a Delivery.




## Delivery slip

**DELIVERY SLIP**

| Name & Address of the Supplier | Name & Address of the Customer | | | Date |
|---|---|---|---|---|
| Adhi software | Adaxa ltd<br>navalur | | | October 05, 2021 |
| **Item** | | **Rate** | **Quantity** | **Amount** |
| Computer Table-wood-wood soft | | ₹ 1000.0 | 2.0 | ₹ 2000.0 |
| | | **Grand Total** | **2.00** | **₹ 2000.00** |
| **Amount in Words** | Two thousand only. | | | |
| **Notes** | | | | |

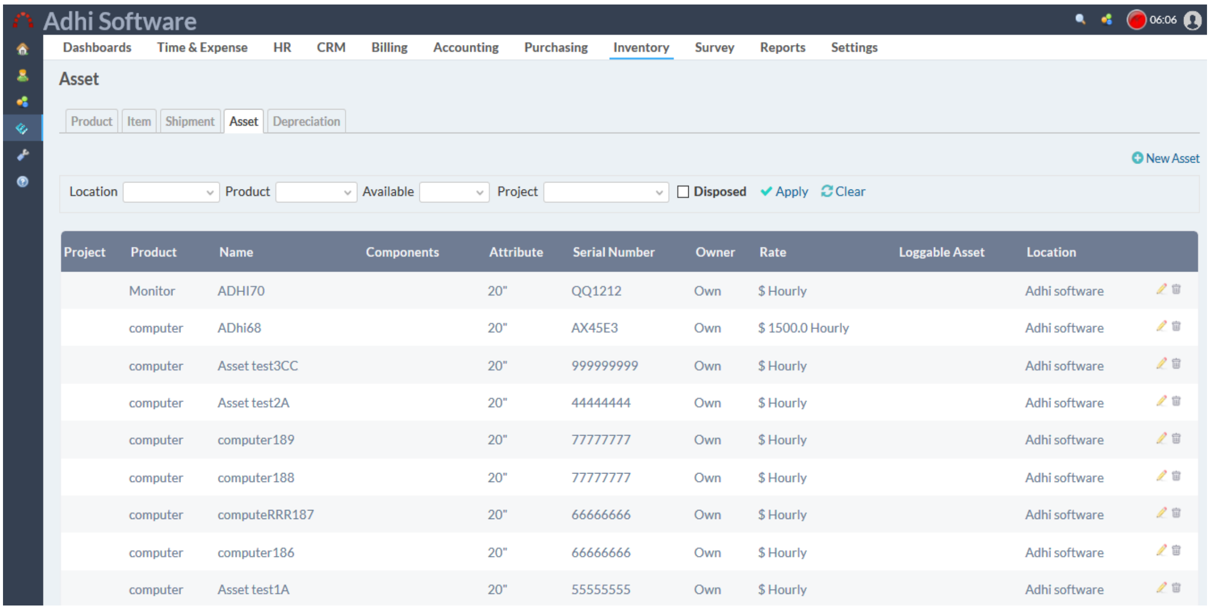
Place :
Date :

Authorized Signatory

# 11 Asset

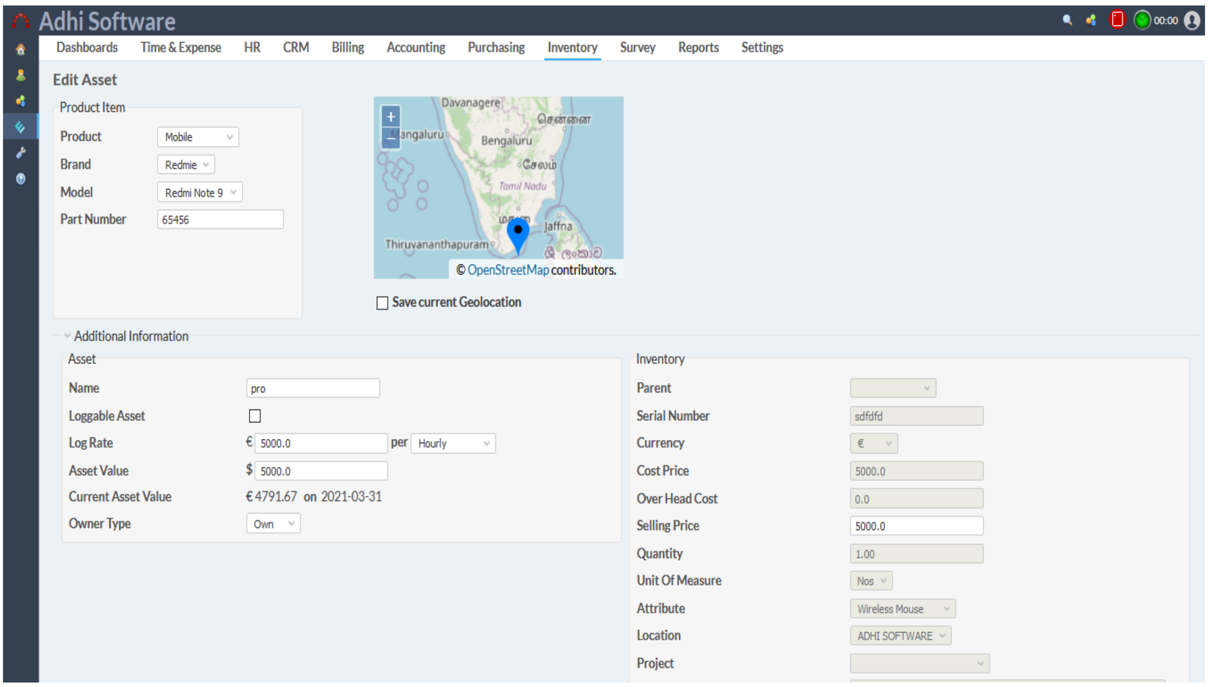
## 11.1 Asset

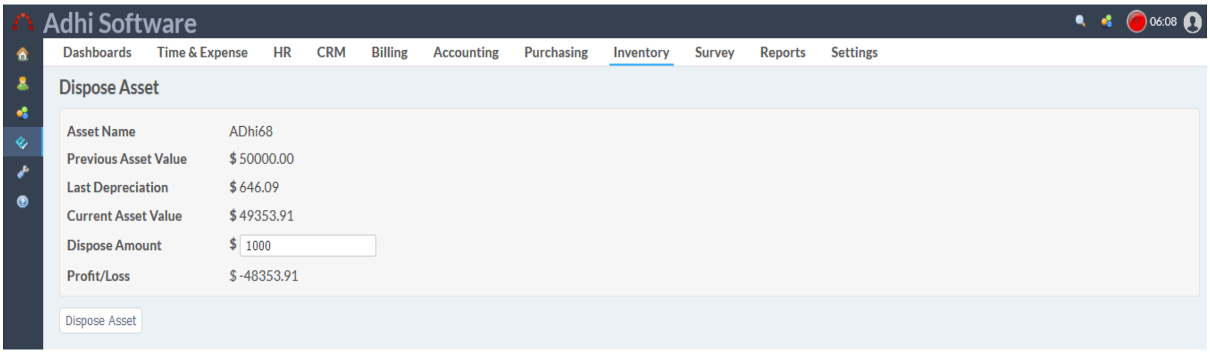
Asset are the wealth of the company, they can be of type building, furniture, equipment's, plant, machinery etc. They are modelled as a product belonging to a particular brand and model, with a unique set of attributes. Products and brand filters can be applied to the list page. Asset Items can also be removed.




### 11.1.2 New/Edit Asset

To add a new Asset Item, click on "New Product Item" hyperlink. To transfer the items click on "Transfer To" hyper link in Asset item details page. Dispose button helps to save the sold out assists.



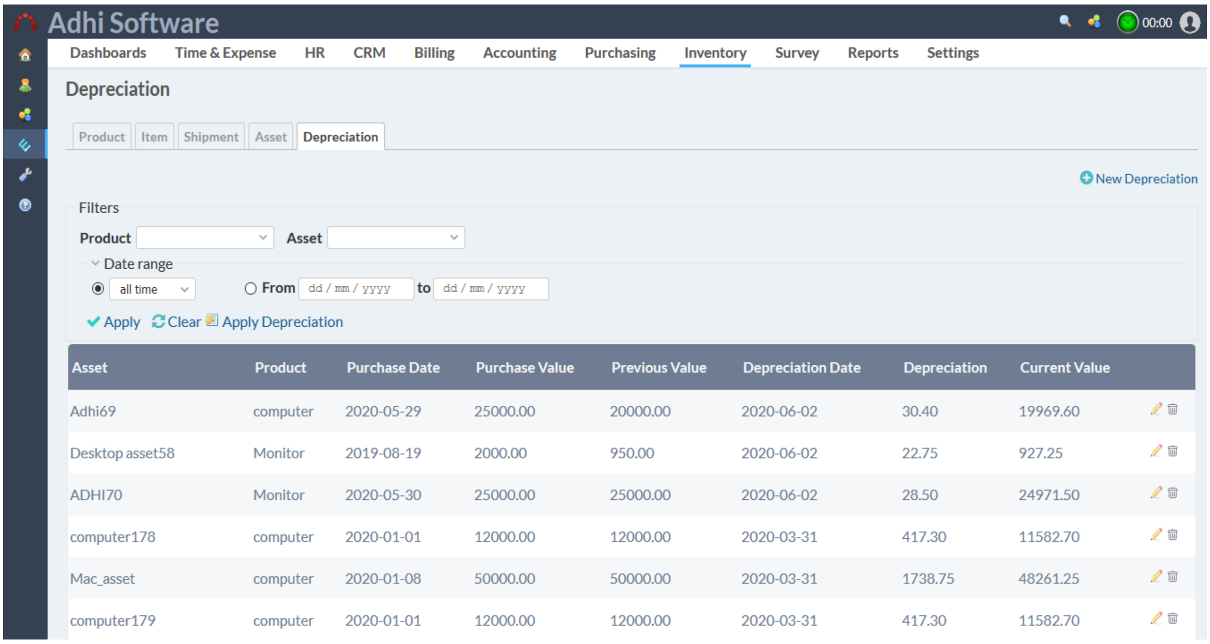



## 11.2 Depreciation

The Decrease in value of asset to the appropriate period. The asset methods are below

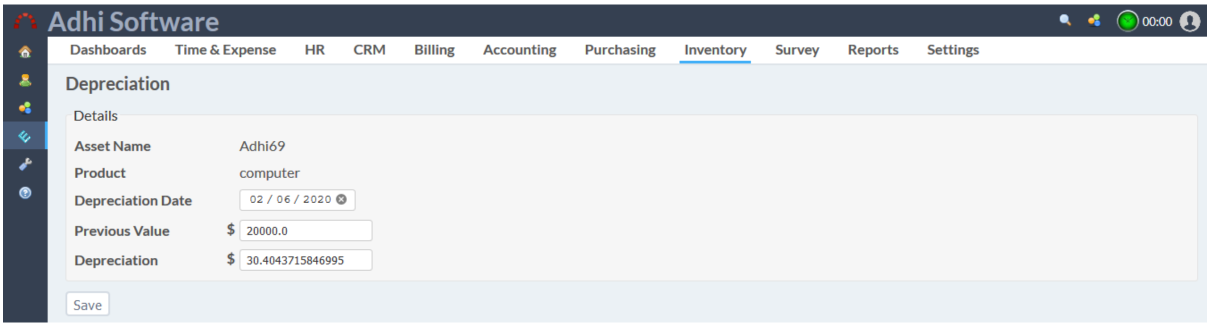
| S.No | Methods |
|---|---|
| 1 | Straight Line Method |
| 2 | Written-Down Value |

The list page by default displays the list of Depreciation. Product and Asset filters can be applied to the list page. The Depreciation of asset can be generated from this page by clicking on the 'Apply Depreciation' link.. Depreciation can be removed as well.



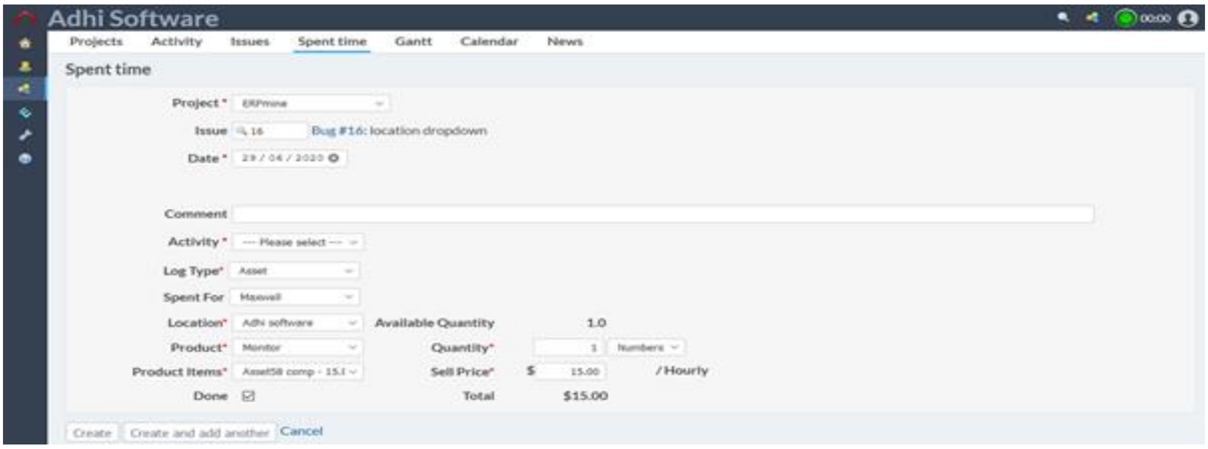

11.2.1 New/Edit Depreciation

To add a new Depreciation, click on "New Depreciation" hyperlink.



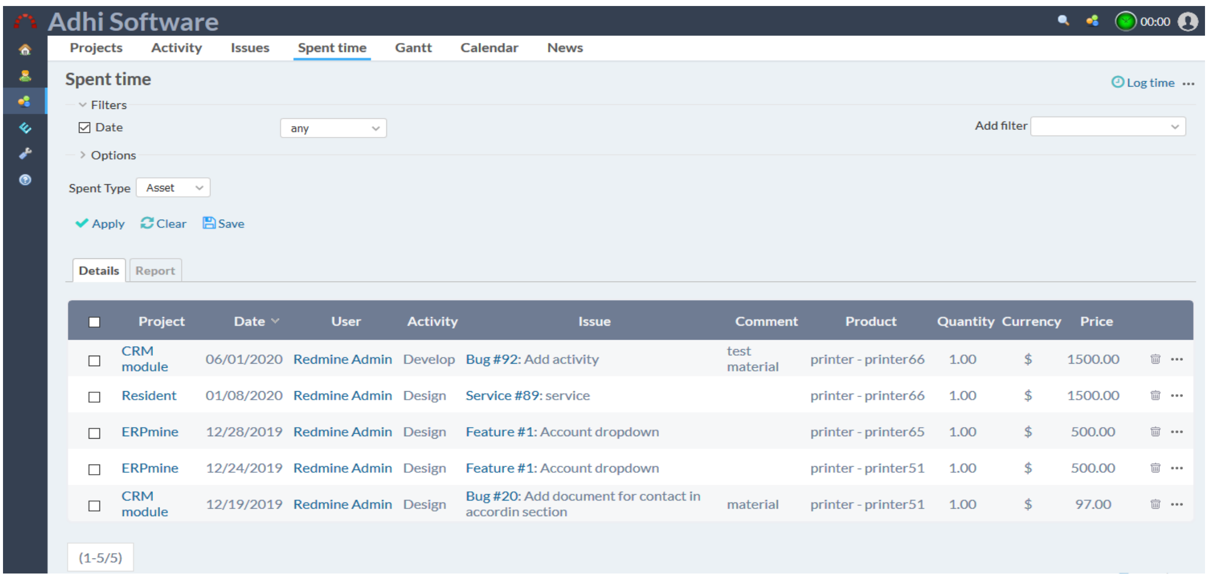

## 11.3 Log Asset

The asset used on a task / issue can be logged using the log time page. Log Asset page allows users to enter the usage of the asset on a specific issue or project



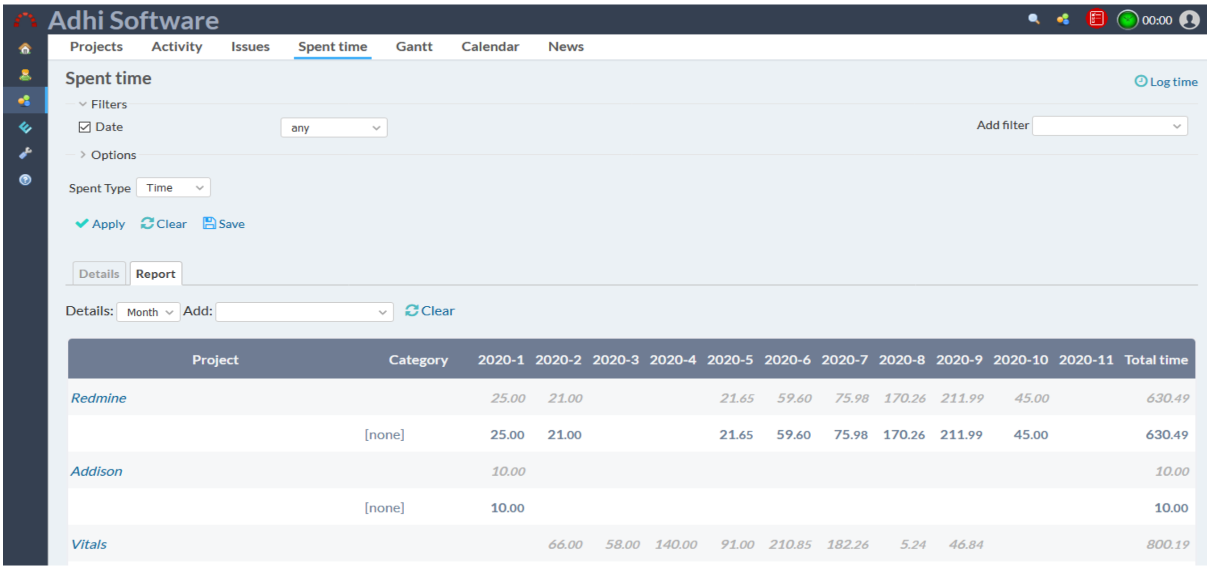

### *11.3.1 Asset Log Details*

The spent asset details can be viewed using the spent time page. Asset Details page is accessible from the Overview > Spent Time > Details menu.



### *11.3.2 Spent Asset Report*

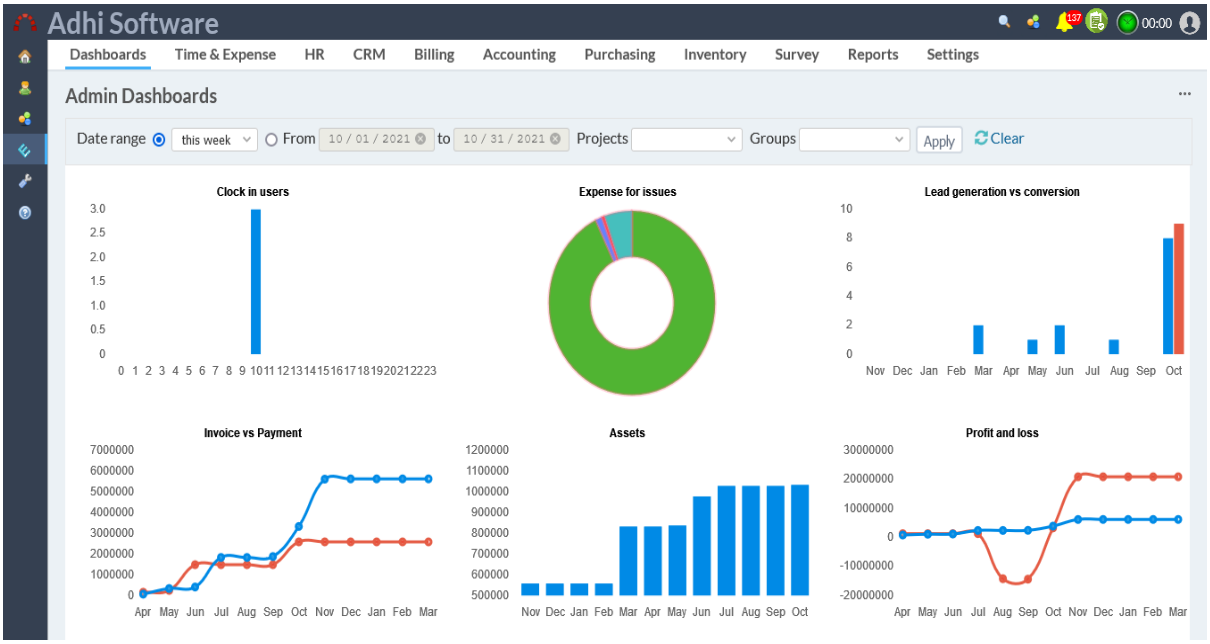
The Asset report page is accessible from the Overview > Spent Time > Details menu. The report can be grouped by Product Item.



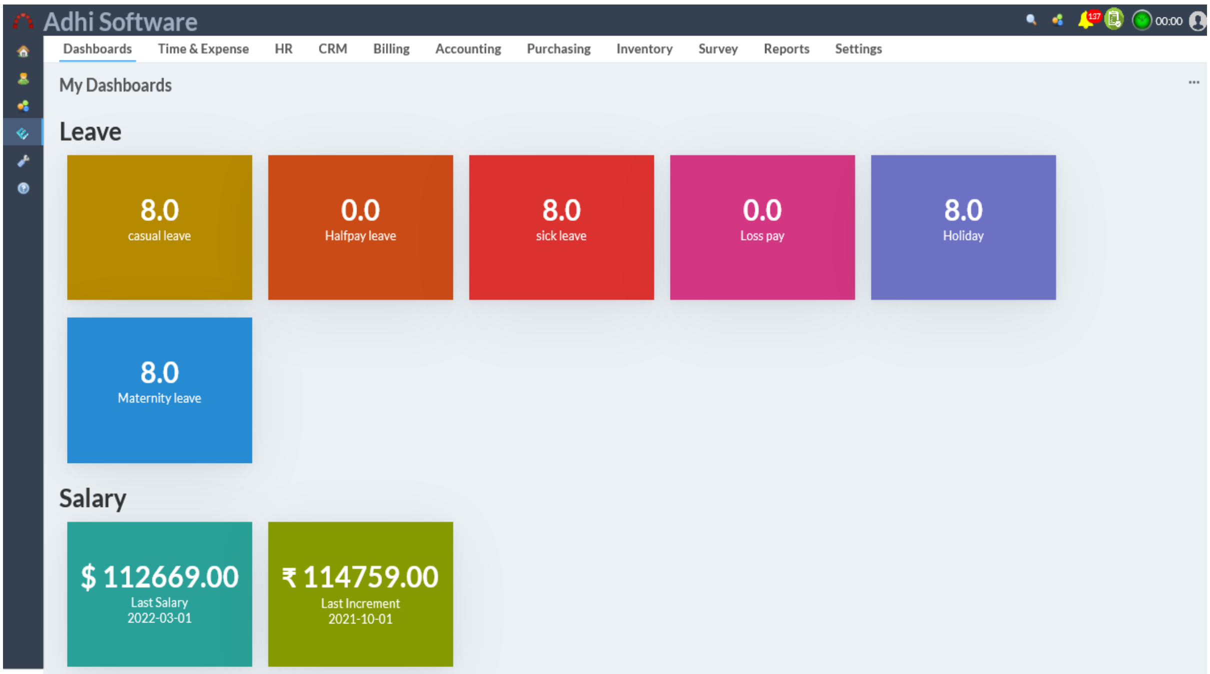

# 12 Dashboards

Dashboards visually tracks, analyzes and displays key performance indicators (KPI). It Organizes and presents information in graphical.

### Admin Dashboard



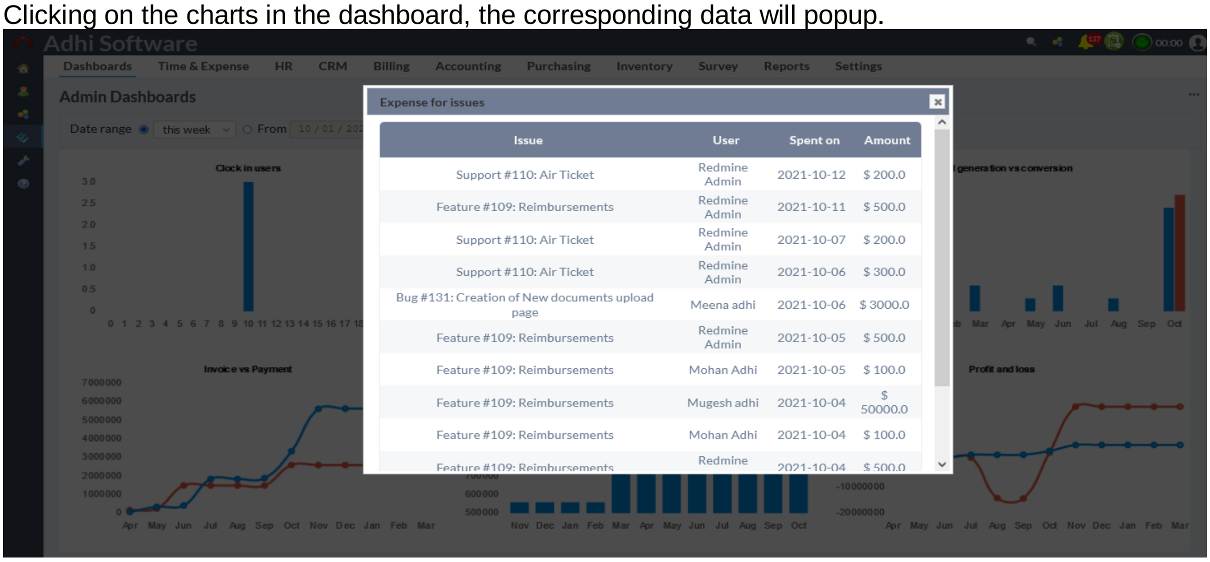

### My Dashboards




It has the following chart

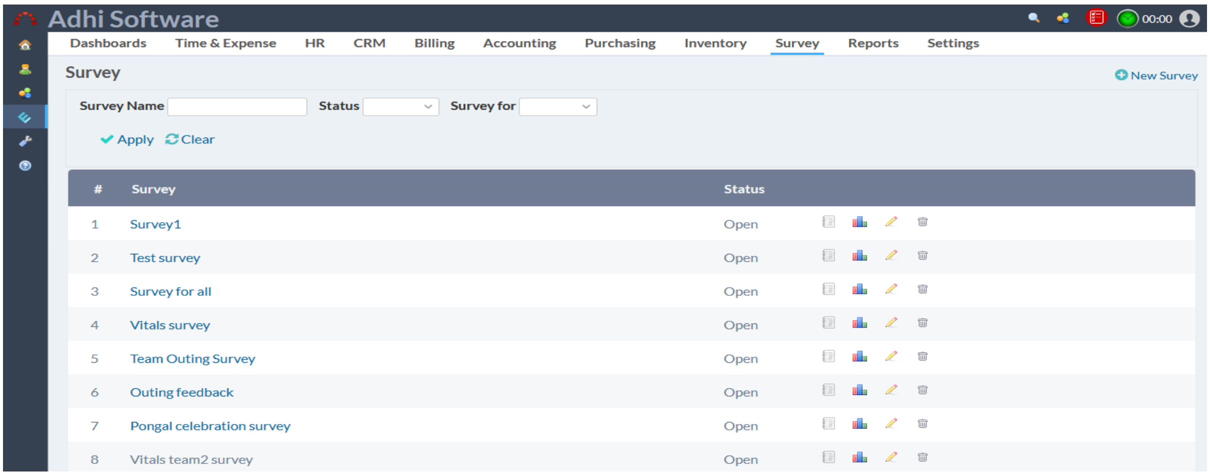
- Clock in Users
- Expense for Issues
- Lead Generation vs Conversion
- Invoice vs Payment
- Asset Values
- Profit and Loss

Clicking on the charts in the dashboard, the corresponding data will popup.

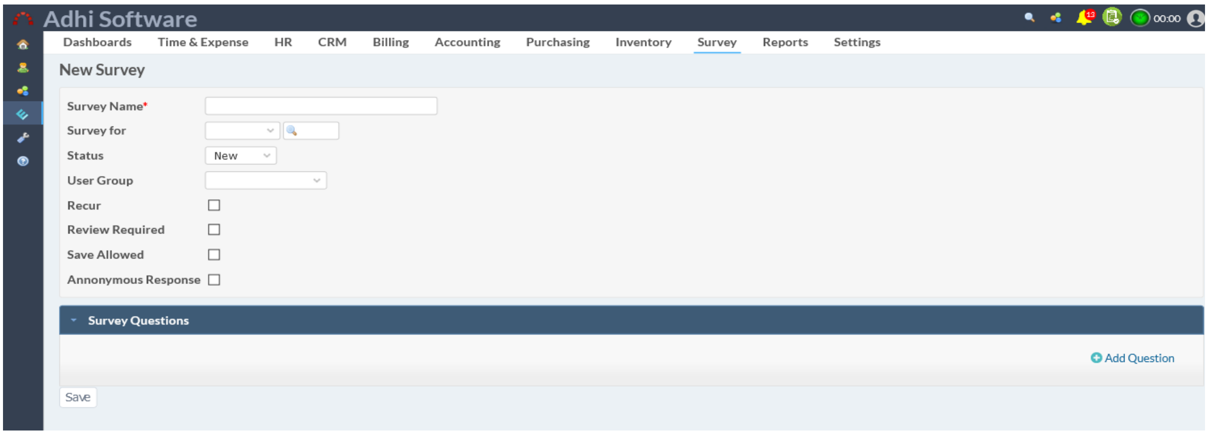



## 13 Survey

Survey can be used for polls and ratings. Survey can also be attached to Projects, Issues, Contacts, Accounts and Users. Survey name, status and "survey for" filters can be applied to the list page. Surveys can also be removed.



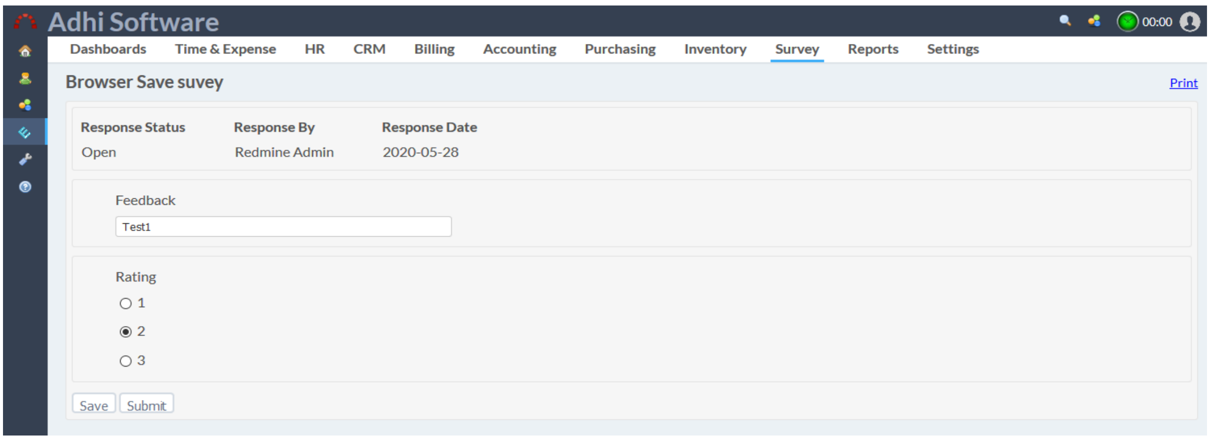

## 13.1 New Survey



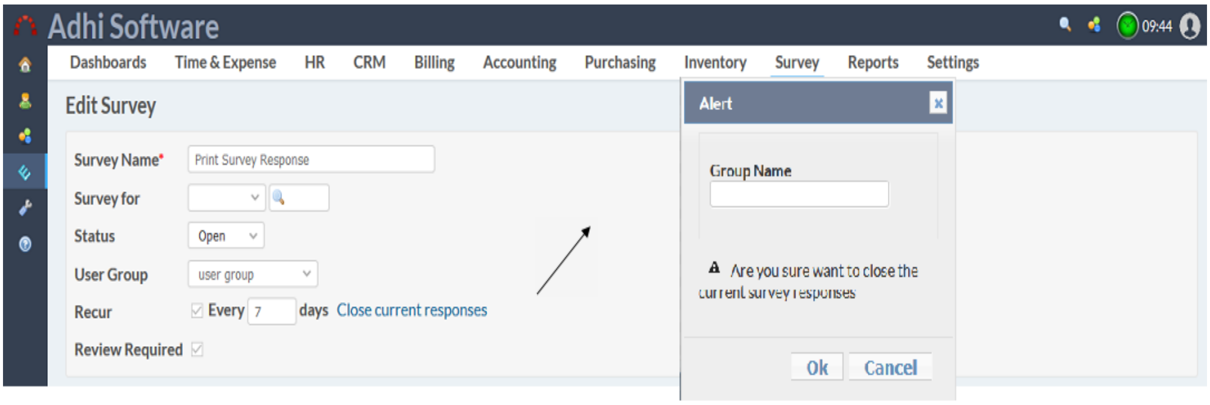

To add a new survey, click on “New survey” hyperlink.

## 13.2 Take Survey

To take a survey, click on survey name hyperlink.

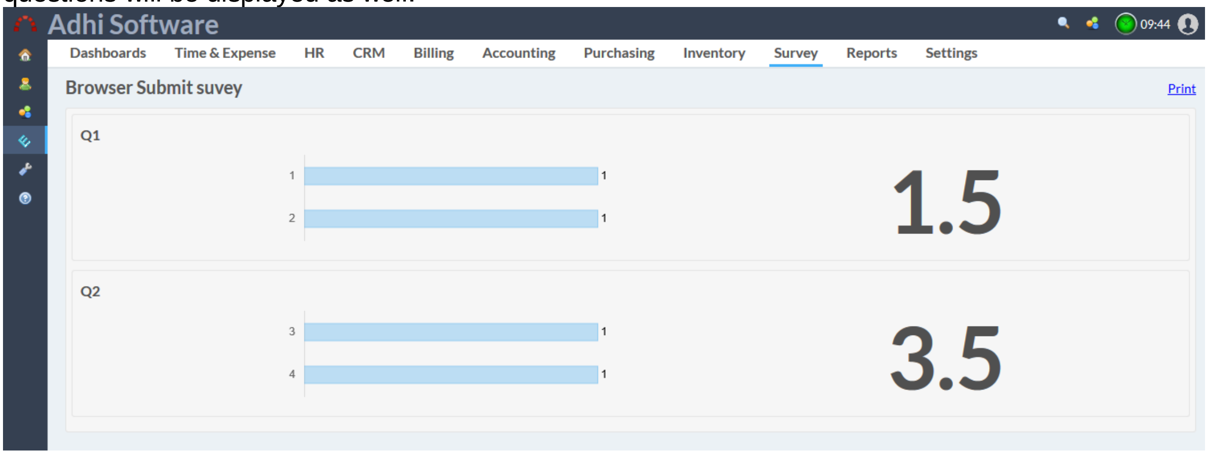



To close survey response click on "Close the current responses” hyperlink and then pop-up appears. Fill the group name in the pop-up and click ok to close the current responses.

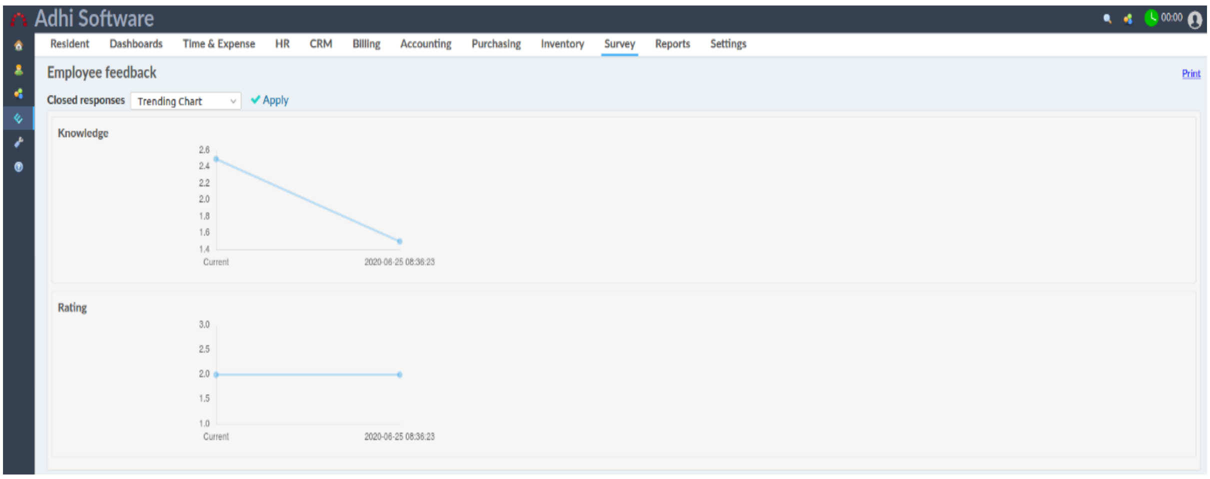



## 13.3 Survey Result

To see a survey result, click on the survey result icon. The average score for the each survey questions will be displayed as well.



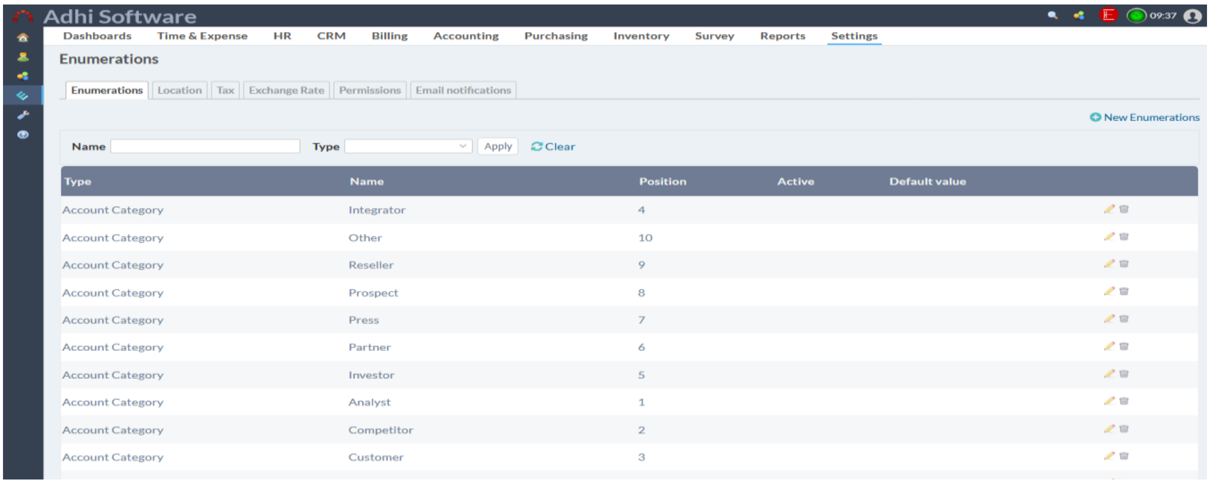

Survey responses can be printed and the "Trending chat" comparing with previous responses can be viewed as well..



# 14 Settings

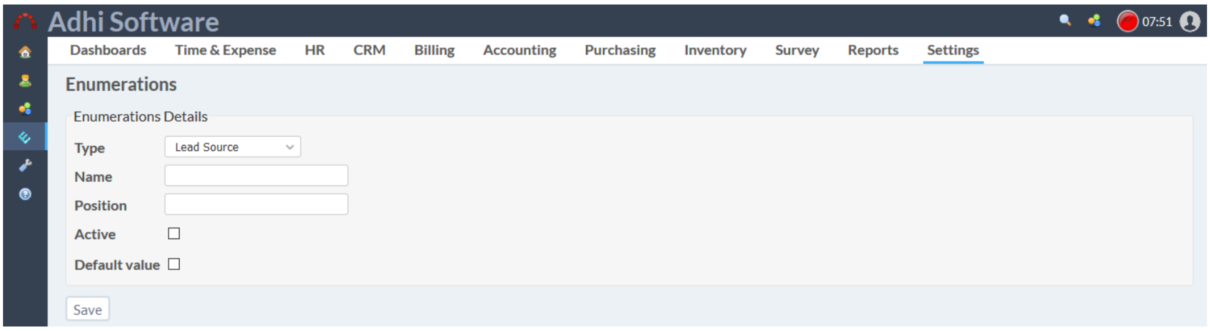
## 14.1 Enumeration

The group of drop down values can be altered using the Enumeration page. Enumeration name (wildcard search) and type filters can be applied to the list page. Enumerations can also be removed.



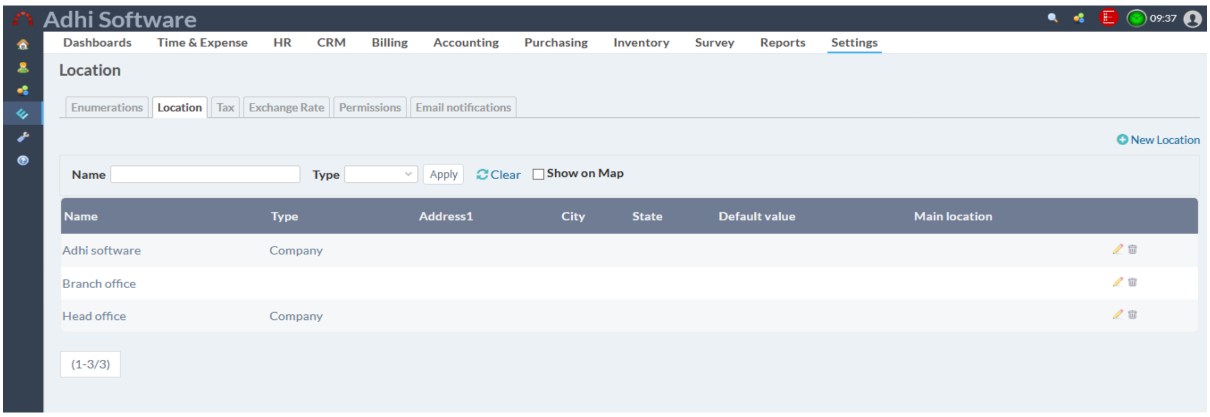

13.1.1 New/Edit Enumeration

To add a new Enumeration, click on “New Enumeration” hyperlink.



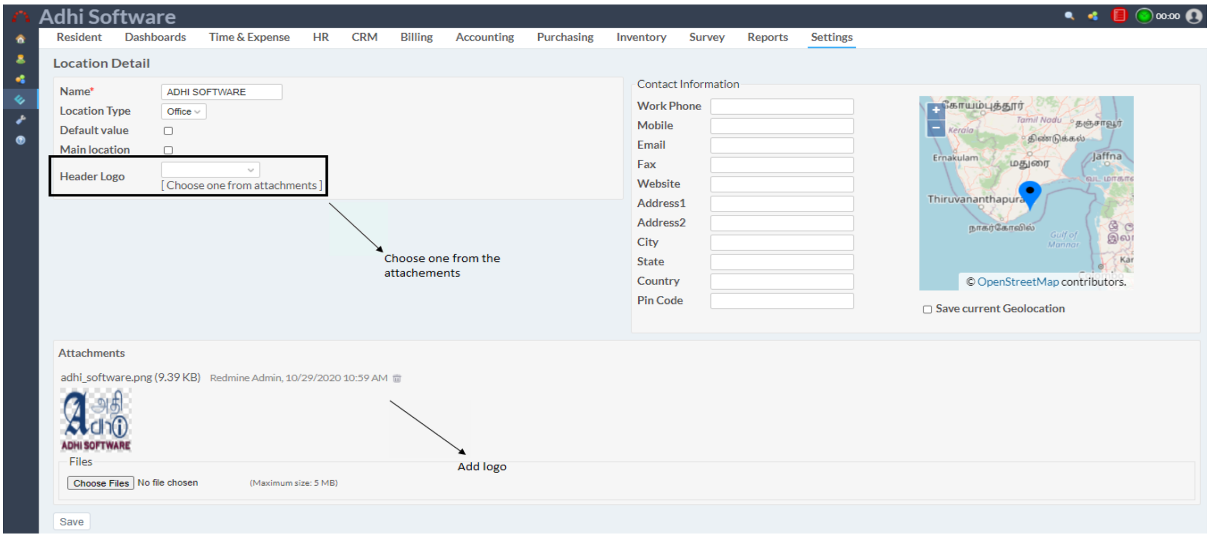

## 14.2 Location

These are the different locations / branches of the company. Location name (wildcard search) and type filters can be applied to the list page. Location can also be removed.



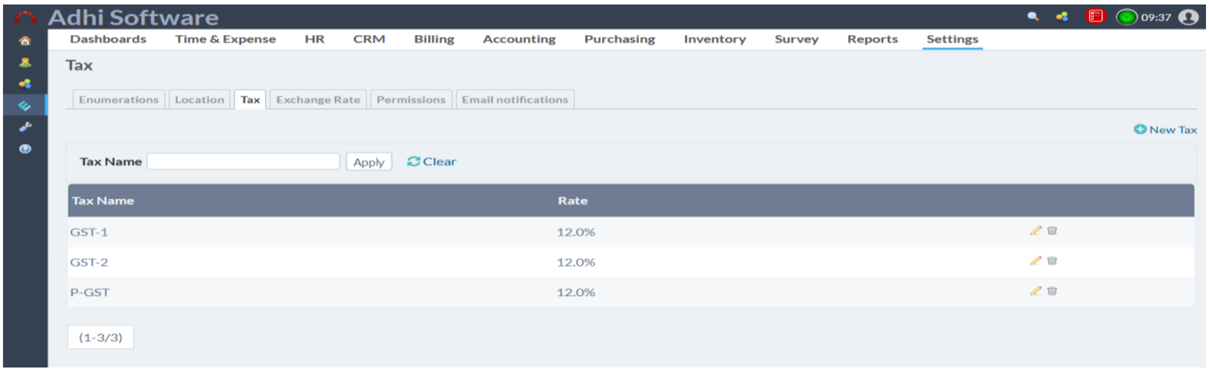
### 13.2.1 New/Edit Location

To add a new Enumeration, click on “New Location” hyperlink. We can add geo location.



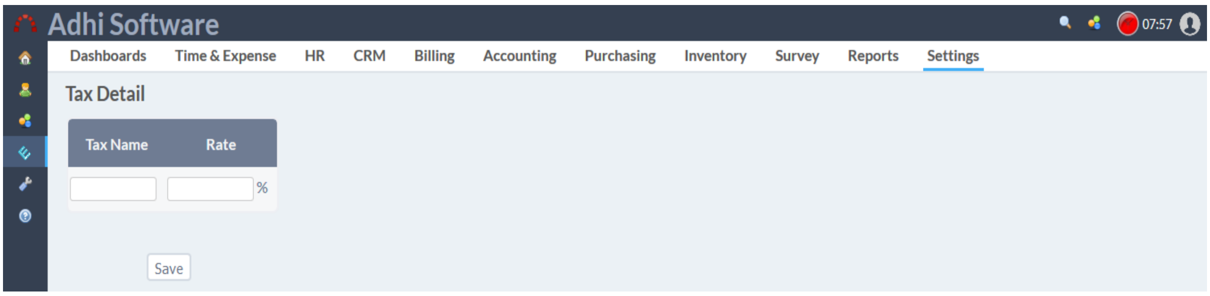

## 14.3 Taxes

The list page by default displays the list of taxes. Tax name wildcard search can be applied to the list page. Tax can be removed.




### 13.3.1 New/Edit Taxes

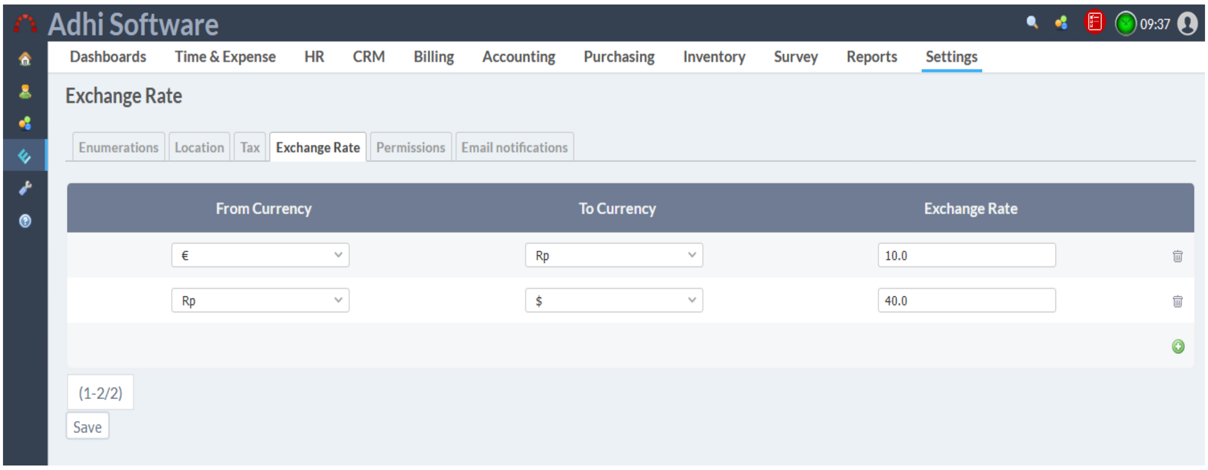
To add a new tax click on “New Tax” hyperlink




The tax name and rate can added/updated.

### 14.4 Exchange Rate

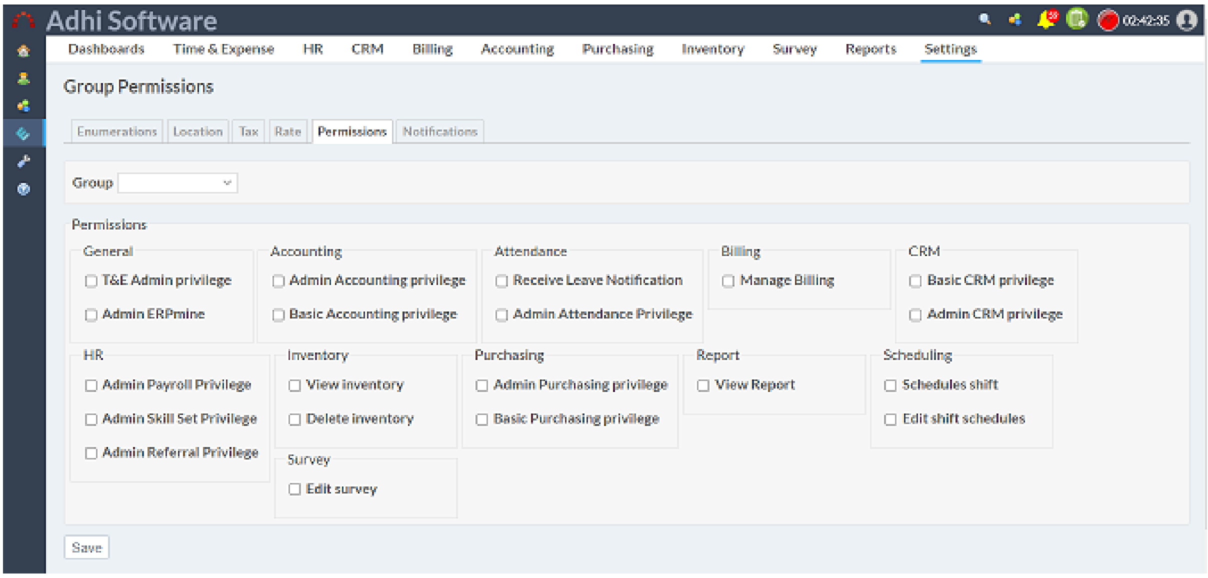
When multiple currencies are involved, the exchange rate needs to be keyed in. The list page by default displays the list of exchanges rates. The exchange rate can be added/updated in the list page. Exchange rate can be removed.




### 14.5 Permissions

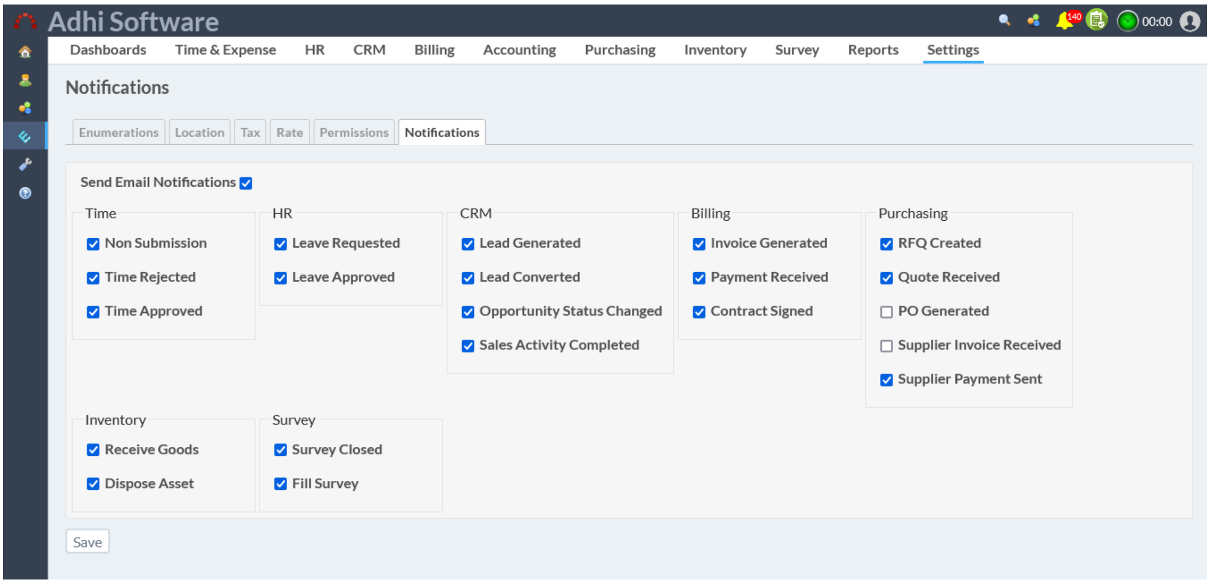
The ERPmie permissions can be assigned to user groups. The following permissions are available

| S.No | Permissions |
|---|---|
| 1 | Admin ERPmine |
| 2 | T&E Admin privilege |
| 3 | Basic Accounting privilege |
| 4 | Admin Accounting privilege |
| 5 | Receive Leave Notification |
| 6 | Admin Attendance Privilege |
| 7 | Manage Billing |
| 8 | Basic CRM privilege |
| 9 | Admin CRM privilege |
| 10 | View inventory |
| 11 | Delete inventory |
| 12 | Admin Payroll Privilege |
| 13 | Basic Purchasing privilege |
| 14 | Admin Purchasing privilege |
| 15 | Schedules shift |
| 16 | Edit shift schedules |
| 17 | Edit Survey |
| 18 | View Report |




## 14.6 Notifications

Events can be selected for notifications under settings.




# 15 Reports

## 15.1 Attendance

Attendance report shows the monthly view of attendance. Date range and Group filters can be applied. The report displays the general information of the user along with the leave summary and attendance entries for the month. The leave taken by the user will be marked with short name of the leave. TE Admins have the privilege to look into other user’s attendance. All reports can be exported us PDF and CSV.

FORM-Q
REGISTER OF EMPLOYMENT FOR SH
Name and Address of the Establishment: Re

| Sl. No. | User | Date of entry into service | Age / Date of birth | Designation | Leave Credit at the Beginning of the Month | | | Leave Availed During the Month | | | Leave Balance | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | SL | CL | H | SL | CL | H | SL | CL | H | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 33 | Suganya Thulasiraman | 05/03/2010 | | SSE | 24.0 | 32.0 | 8.0 | 16.0 | 0.0 | 0.0 | 16.0 | 40.0 | 16.0 | 6.62 | 4.25 | 8.5 | 9.0 | | | | 8.0 |
| | Karthick Madhiazhagan | 03/03/2014 | | Program Analyst | -13.0 | 48.0 | 8.0 | | | | | | | 4.8 | 8.5 | | 8.5 | | | | CL |
| | Jeya Sundari | | | | | | | 0.0 | 0.0 | 0.0 | 8.0 | 8.0 | 8.0 | 5.75 | 7.5 | | | | | | |
| | Dhinesh R | 06/02/2014 | | Prgm Analyst | | | | 0.0 | 0.0 | 0.0 | 8.0 | 8.0 | 8.0 | 6.5 | 8.0 | 8.25 | 5.75 | | | CL | |
| | Redmine Admin | | | | | | | | | | | | | 4.75/ SL | 6.7 | | | | | CL | CL |
| | janaki janaki | | | Admin | | | | | | | | | | | | | | | | | |
| | Veeralakshmi D | | 04/07/1985 | SSEngineer | 48.0 | 32.0 | 16.0 | | | | | | | | | | | | | | |
| 34 | Chandra D | 05/03/2010 | 01/11/1988 | SSE | | | | 0.0 | 0.0 | 0.0 | 8.0 | 8.0 | 8.0 | 8.5 | 8.5 | 9.5 | 8.75 | 9.17 | | 7.5 | |
| 32 | Saravana C | 03/18/2009 | | SSE | | | | | | | | | | | | | | | | | |
| 40 | Aravind S | 03/01/2012 | | SE | | | | 0.0 | 8.0 | 0.0 | 8.0 | 0.0 | 8.0 | | | | | | | | |
| | Arul Baskar | 12/01/2010 | | SSE | | | | 0.0 | 0.0 | 0.0 | 8.0 | 8.0 | 8.0 | | | | | | | | |
| | Prabhakar Selvam | 12/01/2010 | 01/30/2016 | Sr s/w Tester | 0.0 | 24.0 | 8.0 | 0.0 | 0.0 | 8.0 | 8.0 | 32.0 | 8.0 | | | | | | | | |
| | dhana lakshmi | 11/01/2010 | 12/17/1987 | | | | | | | | | | | | | | | | | | |

**Note:** The user with termination date less than the report date will not be shown in report. And also locked users will not be shown in report.

## 15.2 Timesheet

Time report shows the weekly view of spent time. Date range and Group filters can be applied. The report displays the name, week, projects, issues, activity, comment, custom fields and spent time.

**Timesheet**

**Name** : suganya T
**Week** : 2016-07-10 - 2016-07-16

| Project | Issue | Activity | Comment | Estimated hours | Difficulties | Sun Jul 10 | Mon Jul 11 | Tue Jul 12 | Wed Jul 13 | Thu Jul 14 | Fri Jul 15 | Sat Jul 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SPA | GI Interface | Design | | | | | 8.00 | 8.00 | 8.00 | 8.00 | 8.00 | |
| | | | | | Total = 40.00 | 0.00 | 8.00 | 8.00 | 8.00 | 8.00 | 8.00 | 0.00 |

I Acknowledge that the hours entered are accurate to the best of my knowledge

**Signature :**
**Submitted By** ____________________
**Approved By** ____________________

## 15.3 Expense Sheet

Expense report shows the weekly expense entries made for the project. Date range and Group filters can be applied. The report displays the name, week, projects, issues, activity, comment and currency.

**Expensesheet**

**Name** : suganya T
**Week** : 2016-07-10 - 2016-07-16

| Project | Issue | Activity | Comment | Currency | Sun Jul 10 | Mon Jul 11 | Tue Jul 12 | Wed Jul 13 | Thu Jul 14 | Fri Jul 15 | Sat Jul 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| SPA | GI Interface | Design | | $ | | 10.00 | 10.00 | 10.00 | 10.00 | 10.00 | |
| | | | | Total = $ 50.00 | $ 0.00 | $ 10.00 | $ 10.00 | $ 10.00 | $ 10.00 | $ 10.00 | $ 0.00 |

I Acknowledge that the hours entered are accurate to the best of my knowledge

**Signature :**
**Submitted By** ______________________
**Approved By** ______________________

## 15.4 Spent Time

Attendance report shows the monthly view of attendance. Date range and Group filters can be applied. The report displays the general information of the user along with the leave summary and spent time entries for the month. The leave taken by the user will be marked with short name of the leave. TE Admins have the privilege to look into other user's attendance.

**FORM-Q**
**REGISTER OF EMPLOYMENT FOR SHOPS AND ESTABLISHMENTS**
**Name and Address of the Establishment:** Adhi software pvt ltd **Month:** July **Year:** 2016

| Sl. No. | User | Date of entry into service | Age / Date of birth | Designation | Leave Credit at the Beginning of the Month | | | Leave Availed During the Month | | | Leave Balance | | | Da | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | SL | CL | WH | SL | CL | WH | SL | CL | WH | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| | Redmine Admin | | | | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | | | |
| 33 | Suganya Thulasiraman | 05/18/2010 | 05/13/1987 | | 0.0 | 24.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 24.0 | 0.0 | 4.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.( |
| 55 | Karthick Adhi | | | | 0.0 | 40.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 40.0 | 0.0 | | | | | | | | | | | | |
| 62 | Jeya Sundari | 06/11/2014 | 07/03/1989 | | 0.0 | 32.0 | 0.0 | 0.0 | 16.0 | 0.0 | 0.0 | 16.0 | 0.0 | | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | | | CL | CL |
| 61 | Dhinesh Kumar | 04/15/2015 | 01/09/1989 | | 0.0 | 24.0 | 0.0 | 4.0 | 20.0 | 0.0 | -4.0 | 4.0 | 0.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | | 5.0 | 5.( |
| 55 | Jeya Priya | 03/18/2014 | 09/10/1990 | Software analysis | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 4.0 | 5.0 | | 5.0 | 4.0 | 3.0 | 2.0 | 1.0 | 5.0 | | | |
| 52 | Divya Bharathi | 09/07/2015 | | | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 8.0 | | | 8.0 | 8.0 | 8.0 | 8.0 | 8.0 | | | 5.0 | 5.( |
| 56 | janaki adhi | 04/16/2014 | 03/14/1989 | Admin | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | | | |
| 51 | Gandhi Mathi | 01/01/2016 | 06/19/1991 | Software analysis | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | | | |
| 25 | Veera Lakshmi | 01/15/2008 | 04/09/1985 | senior Software | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | | | |
| 34 | Chandra Durairaj | 05/12/2010 | 01/11/1988 | senior Software | | | | 0.0 | 1.0 | 0.0 | 0.0 | -1.0 | 0.0 | | | | | | CL | | | | | | |
| 43 | Arul Baskar | 01/19/2011 | 01/22/1986 | senior Software | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | 2.0 | 3.0 | 4.( |
| 63 | Anitha adhi | 11/24/2015 | 07/15/1989 | | 0.0 | 5.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 5.0 | 0.0 | | | | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | | 5.0 | 5.0 | 5.( |
| 64 | Thanga Raj | 12/01/2015 | 01/09/1980 | Test Analyst | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | | | |
| 41 | Prabhakaran adhi | 09/07/2011 | | | 0.0 | 32.0 | 0.0 | 0.0 | 0.0 | 0.0 | 8.0 | 40.0 | 8.0 | 8.0 | | | 8.0 | 8.0 | 8.0 | 8.0 | 8.0 | | | 8.0 | 8.( |
| 71 | krish raghunandan | 06/01/2015 | 06/24/2016 | Developer | 0.0 | 24.0 | 0.0 | 0.0 | 15.0 | 0.0 | 8.0 | 17.0 | 8.0 | | | | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | | | 6.0 | 6.( |
| 66 | meena adhi | 06/01/2016 | | | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | | | | | | | | | |
| | Berret Hinn | 04/01/2015 | | | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | CL | | 4.0 | 4.0 | 4.0 | 5.0 | 4.0 | 4.0 | 4.0 | 5.0 | 5.0 | 5.( |
| | Boopathi adhi | | | | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | CL | | 4.5 | 5.0 | 4.0 | 5.0 | 5.0 | 4.0 | | 5.0 | 5.0 | 5.( |
| 31 | Saravana Adhi | | | | 16.0 | 40.0 | 0.0 | 0.0 | 0.0 | 0.0 | 16.0 | 40.0 | 8.0 | 6.0 | | 8.0 | 8.0 | 8.0 | 8.0 | 8.0 | 8.0 | | | 5.0 | |
| 68 | Dhana Lakshmi | | | | | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 4.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | 5.( |
| 49 | Suresh adhi | | | | 8.0 | 24.0 | 0.0 | 0.0 | 0.0 | 0.0 | 8.0 | 24.0 | 8.0 | 5.0 | | | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | | | 5.0 | 5.( |
| 47 | Aravind adhi | 10/01/2015 | | | 24.0 | 40.0 | 0.0 | 0.0 | 25.0 | 0.0 | 24.0 | 23.0 | 8.0 | 8.0 | | | 5.0 | 5.0 | 5.0 | 5.0 | 5.0 | | | CL | CL |

(1-23/23)

## 15.5 Payroll Report

Payroll report shows the view of salary register. Date range and Group filters can be applied. The report displays the general information of the user along with the basic, allowances and deduction for the salary period. TE Admins have the privilege view other user's salary data.

**FORM-R**
**Register of Wages**
**See Sub Rule(5) of rule 11 of Tamilnadu Shops and Establishments Rule, 1948**
**Name and Address of the Establishment:** Adhi software Pvt Ltd 12/B-35, 6th cross Road SIPCOT IT Park, Siruseri Kancheepuram **Wages Period Month:** 2016-09-01

ADHI SOFTWARE

| Id | Name | Gender | Designation | Basic Pay | Medical | conv | HRA | SPL(1) | SPL(F) | Edu | Leave | Bonus | PF | Cab | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 33 | Suganya | | | 27500.0 | 1250.0 | 1600.0 | 15250.0 | 0.0 | 9400.0 | | | | 3300.0 | 0.0 | 1 |
| 61 | Dhinesh | | | 10000.0 | 1250.0 | 1600.0 | 5950.0 | 0.0 | 1200.0 | | | | 1200.0 | 0.0 | |
| 55 | Jeya | | Software analysis | 15000.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 1800.0 | 600.0 | |
| 52 | Divya | | | 15000.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 1800.0 | 600.0 | |
| 56 | janaki | | Admin | 10000.0 | 1250.0 | 1600.0 | 5950.0 | 0.0 | 1200.0 | | | | 1200.0 | 0.0 | |
| 25 | Veera | | senior Software | 29000.0 | 3000.0 | 2920.0 | 17000.0 | 0.0 | 12000.0 | | | | 3480.0 | 0.0 | 1 |
| 43 | Arul | | senior Software | 22500.0 | 1250.0 | 2800.0 | 11250.0 | 0.0 | 8400.0 | | | | 2700.0 | 0.0 | 1 |
| 63 | Anitha | | | 100.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 12.0 | 600.0 | |
| | Redmine | | | 10000.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 1200.0 | 0.0 | |
| | Berret | | | 2490000.0 | 207500.0 | 265600.0 | 1319700.0 | 0.0 | 199200.0 | | | | 298800.0 | 0.0 | |
| 64 | Thanga | | Test Analyst | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | 0.0 | 0.0 | |
| 55 | Karthick | | | 15000.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 1800.0 | 0.0 | |
| 41 | Prabhakaran | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | 0.0 | 0.0 | |
| 66 | meena | | | 10000.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 1200.0 | 0.0 | |
| | Boopathi | | | 8000.0 | 100000.0 | 128000.0 | 636000.0 | 0.0 | 96000.0 | | | | 960.0 | 0.0 | |
| 49 | Suresh | | | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | | | | 0.0 | 0.0 | |
| 47 | Aravind | | | 2415000.0 | 172500.0 | 220800.0 | 1207500.0 | 0.0 | 814200.0 | | | | 289800.0 | 0.0 | 1 |
| 31 | Saravana | | | 3045000.0 | 315000.0 | 168000.0 | 1785000.0 | 0.0 | 1260000.0 | | | | 365400.0 | 0.0 | 42 |
| 68 | Dhana | | | 15000.0 | 1250.0 | 1600.0 | 7950.0 | 0.0 | 1200.0 | | | | 1800.0 | 0.0 | |
| 6 | Dhanasingh | | Director | | | | | | | | | | | | |

## 15.6 Payslip Report

Payslip report shows the view of salary payslip for individual user . Date range and Group filters can be applied. The report displays the general information of the user along with the basic, allowances and deduction for the month. TE Admins have the privilege to view other user's Payslip.

**Payslip**

Adhi software

| | | | | | |
|---|---|---|---|---|---|
| **Salary Date** | 2021-03-01 | **Employee Name** | Redmine Admin | **Employee ID** | |
| **Join Date** | 2018-03-01 | | | | |

| Earnings | | | Deductions | | |
|---|---|---|---|---|---|
| | Monthly | YTD | | Monthly | YTD |
| **Basic** | $ 105000.00 | $ 390000.00 | **Prof tax** | $ 1250.00 | $ 1250.00 |
| **HRA** | $ 18500.00 | $ 92500.00 | **ESI** | $ 0.00 | $ 0.00 |
| **Eduaction Allowance** | $ 1200.00 | $ 6000.00 | **TDS** | $ 20813.00 | $ 59939.00 |
| **Medical** | $ 5000.00 | $ 25000.00 | **PF** | $ 2368.00 | $ 11840.00 |
| **conveyance** | $ 8000.00 | $ 40000.00 | **cab** | $ 600.00 | $ 3000.00 |
| **Total Earnings** | $ 137700.00 | $ 553500.00 | **Total Deductions** | $ 25031.00 | $ 76029.00 |
| **Net Earnings** | $112669.00 | $477471.00 | | | |

| Reimbursements | | |
|---|---|---|
| | Monthly | YTD |
| **Reimbursment** | $ 2000.00 | $ 2000.00 |
| **Net Earnings + Reimbursements** | $114669.00 | $479471.00 |

YTD - Year to Date from April 01, 2020 to March 31, 2021

## 15.7 Profit & Loss A/c Report

Profit & Loss a/c shows the view of income and expense of financial transaction. Date range filter can be applied. Accounting Admins and Users have the privilege to view the data.

Adhi software Pvt Ltd
Profit & Loss A/c
December 1, 2016 To December 31, 2016

| Particulars | | December 1, 2016 To December 31, 2016 |
|---|---|---|
| Trading Account : | | |
| Sales Accounts | | |
| Direct Incomes | | 100000.00 |
| S/w Development | 100000.00 | |
| Cost of sales | | 260.00 |
| Purchase Accounts | | |
| Direct Expenses | 260.00 | |
| Canteen | -240.00 | |
| Depreciation | 500.00 | |
| Gross Profit : | | 99740.00 |
| Income Statement : | | |
| Indirect Incomes | | |
| | | 99740.00 |
| Indirect Expenses | | 27000.00 |
| Salary | 27000.00 | |
| Net Profit : | | 72740.00 |

## 15.8 Balance Sheet

Balance Sheet shows the view of inflow and outflow of financial transaction. Date range filter can be applied. Accounting Admins and Users have the privilege to view the data.

Adhi software
Balance Sheet
As at March 31, 2021

| | | |
|---|---|---|
| Source of Funds : | | |
| Capital Account | | 300000.00 |
| capital | 300000.00 | |
| Loans (Liability) | | -125.00 |
| Unsecured Loans | -125.00 | |
| test ledger | -125.00 | |
| Current Liabilities | | -9572.60 |
| Provisions | 1000.00 | |
| Deferred tax | 1000.00 | |
| Sundry Creditors | -10572.60 | |
| Aroma traders | -10572.60 | |
| Profit & Loss A/c | | 1473038.11 |
| Opening Balance | 1485258.11 | |
| Current Period | -12220.00 | |
| Total : | | 1763340.51 |
| Application of Funds : | | |
| Fixed Assets | | 351587.94 |
| computer | 351587.94 | |
| Current Assets | | -2078412.13 |
| Bank Accounts | -3137154.55 | |
| Syndicate bank | -3137154.55 | |
| Cash-in-hand | 104516.00 | |
| cash | 104516.00 | |
| Sundry Debtors | 390800.00 | |
| Asset Sale Recievables | 390800.00 | |
| software development Receivable | 563426.42 | |
| Total : | | -1726824.19 |

## 15.9 Lead Conversion

Lead Conversion report shows the view a list of all leads. Conversion ratio is number of converted leads over total number of leads.. Date range and user filter can be applied. CRM Admins and Users have the privilege to view the data.

**Adhi software Pvt Ltd**
**Lead Conversion Report**
01-Jan-2017 To 31-May-2017

| Lead | Status | Created | Converted | Sales Cycle Days | Assignee |
|---|---|---|---|---|---|
| james | Converted | 2017-02-07 10:38:24 | 2017-02-07 10:39:58 | 0.0 | |
| Warner | Converted | 2017-02-07 13:47:32 | 2017-02-07 13:58:42 | 0.01 | Saravana Adhi |
| kerry | New | 2017-02-07 15:29:59 | | | |
| Wellesy | New | 2017-02-07 17:49:23 | | | |
| John | Converted | 2017-02-07 17:18:12 | 2017-05-25 15:04:47 | 106.91 | |
| | | | | **Conversion Rate: 60.0%** | |

## 15.10 Sales Activity

Sales activity report shows the view a list of all activities. Date range and user filter can be applied. CRM Admins and Users have the privilege to view the data.

**Adhi software Pvt Ltd**
**Sales Activity Report**
01-Jan-2017 To 31-May-2017

| Activity Type | Subject | Status | Related to | Name | Start Date & Time | Completed Date & Time | Assignee | Duration Days |
|---|---|---|---|---|---|---|---|---|
| Meeting | contact meeting | Planned | Contact | contact1 | 2017-01-01 10:15:00 | | janaki adhi | |
| Call | Direct call | Planned | Opportunity | mail opportunity | 2017-02-05 07:00:00 | | janaki adhi | |
| Call | forward call | Planned | Account | Hooper | 2017-02-01 09:30:00 | | Saravana Adhi | |
| Task | new task | Not Started | Account | Hooper | 2017-01-01 11:45:00 | | Saravana Adhi | |
| Meeting | Intro meeting | Held | Account | Hooper | 2017-01-01 05:30:00 | 2017-02-07 11:21:40 | Saravana Adhi | 37.24 |
| Call | Inquiry call | Held | Account | Hooper | 2017-02-06 07:00:00 | 2017-02-07 11:12:42 | Saravana Adhi | 1.18 |
| | | | | | | | **Average Duration: 6.4 Days** | |

## 15.11 AR Aging

AR aging report shows the view a list of all outstanding balance for account and contact. Date range and project filter can be applied. Billing Admins have the privilege to view the data.

**Adhi software Pvt Ltd**
**AR Aging**
April 1, 2017 To May 31, 2017

| Account | | Previous Balance | 2017 Apr | 2017 May | Current Balance |
|---|---|---|---|---|---|
| **Steve** | | | | | |
| | Invoice | | 236.00 | 5791.00 | |
| | Payment | | 0.00 | -4745.00 | |
| | Balance | 0.00 | 236.00 | 1046.00 | 1282.00 |
| **Lewis** | | | | | |
| | Invoice | | 1141.00 | 0.00 | |
| | Payment | | 0.00 | -1230.00 | |
| | Balance | -288.00 | 1141.00 | -1230.00 | -377.00 |
| **Andrew** | | | | | |
| | Invoice | | 984.00 | 547.00 | |
| | Payment | | -252.00 | -560.00 | |
| | Balance | 0.00 | 732.00 | -13.00 | 719.00 |
| **Kyle** | | | | | |
| | Invoice | | 0.00 | 2257.00 | |
| | Payment | | 0.00 | -1730.00 | |
| | Balance | 0.00 | 0.00 | 527.00 | 527.00 |
| | | | | **Total:** | 2151.00 |

## 15.12 Purchasing Cycle

Purchasing cycle report shows the list of purchased goods and average time taken for every purchasing stage.

**Adhi software Pvt Ltd**
**Purchase Cycle Report**
01-Jun-2017 To 30-Jun-2017

| RFQ | Purchase Cycle In Days | | | |
|---|---|---|---|---|
| | Winning Quote | Purchase Order | Supplier Invoice | Supplier Payment |
| Projector | 26.00 | 6.00 | 11.00 | |
| Test RFQ | 33.00 | 2.00 | 4.00 | 3.00 |
| Plasma Tv | 0.00 | 3.00 | 2.00 | |
| Samsung LED Tv | | | | |
| Samsung Projector | 22.00 | 0.00 | 0.00 | |
| Test 236 | | | | |
| **Average:** | 20.25 | 2.75 | 4.25 | 3.00 |

## 15.13 Asset Report

Asset report shows the list of assets, purchase value, depreciation value and current value.

**Bangalore**
**Asset Report**
As at November 07, 2020

| # | Project | Asset Name | Product | Purchase Date | Purchase Value | Depreciation | Current Value | Last Depreciation on |
|---|---|---|---|---|---|---|---|---|
| 1 | Vitals | Keyboard7 | Keyboard | 2020-05-01 | R$77000.00 | R$0.00 | R$77000.00 | |
| 2 | Vitals | Keyboard12 | Keyboard | 2020-05-01 | R$77000.00 | R$0.00 | R$77000.00 | |
| 3 | | Projector26 | Projector | 2020-05-01 | R$5500.00 | R$0.00 | R$5500.00 | |
| 4 | | Projector test | Projector | 2020-07-06 | R$5500.00 | R$500.00 | R$5000.00 | |
| 5 | | projector adhi | Projector | 2020-07-06 | R$4500.00 | R$500.00 | R$4000.00 | |
| 6 | | Mobile33 | Mobile | 2020-08-01 | R$16000.00 | R$0.00 | R$16000.00 | |
| 7 | | Mobile34 | Mobile | 2020-08-01 | R$16000.00 | R$0.00 | R$16000.00 | |
| 8 | | Laptop45 | Laptop | 2020-09-01 | R$82000.00 | R$0.00 | R$82000.00 | |
| 9 | | Laptop46 | Laptop | 2020-09-01 | R$16000.00 | R$0.00 | R$16000.00 | |
| 10 | | Laptop47 | Laptop | 2020-09-01 | R$16000.00 | R$0.00 | R$16000.00 | |
| 11 | | Laptop48 | Laptop | 2020-09-01 | R$16000.00 | R$0.00 | R$16000.00 | |
| 12 | | Laptop44 | Laptop | 2020-09-01 | R$82000.00 | R$0.00 | R$82000.00 | |
| 13 | | Laptop49 | Laptop | 2020-09-01 | R$16000.00 | R$0.00 | R$16000.00 | |
| 14 | Redmine | Mobile41 | Mobile | 2020-09-01 | R$42000.00 | R$0.00 | R$42000.00 | |
| 15 | Redmine | Mobile42 | Mobile | 2020-09-01 | R$42000.00 | R$0.00 | R$42000.00 | |
| 16 | Redmine | Mobile43 | Mobile | 2020-09-01 | R$42000.00 | R$0.00 | R$42000.00 | |
| 17 | Redmine | Mobile39 | Mobile | 2020-09-01 | R$42000.00 | R$0.00 | R$42000.00 | |
| 18 | Redmine | Mobile40 | Mobile | 2020-09-01 | R$42000.00 | R$0.00 | R$42000.00 | |
| 19 | | pro | Mobile | 2020-10-30 | €5000.00 | €0.00 | €5000.00 | |
| 20 | Vitals | Mouse1 | Mouse | 2019-11-01 | R$5100.00 | R$212.50 | R$4887.50 | 2020-03-31 |
| | | | | | **R$649600.00** | **R$1212.50** | **R$648387.50** | |

## 15.14 Stock Report

Stock report shows the list of product items in he warehouse and its stock value..

.

**Adhi software Pvt Ltd**
**Stock Report**

| Product | Brand | Model | Attribute | Quantity | Unit Of Measure | Currency | Stock value |
|---|---|---|---|---|---|---|---|
| Washing Machine | Samsung | Sam Washing machine 1222 | | 25.0 | Numbers | R$ | 445000.0 |
| Tv | Sony | Sony Bravia124 | 40" inch Tv | 25.0 | Numbers | R$ | 315000.0 |
| Water Purifier | Aqua Guard | Aqua 1205 | 10ltr Purifier | 15.0 | Numbers | R$ | 94500.0 |
| Tv | Philips | Philips Tv 1202 | 29 inches TV | 23.0 | Numbers | R$ | 343620.0 |
| CD ROM Disks | Sony | Sony CD123 | 16 GB CD ROM | 20.0 | Numbers | R$ | 30600.0 |
| CD ROM Disks | Sony | Sony CD123 | 32 GB CD ROM | 15.0 | Numbers | R$ | 31050.0 |
| Air Conditioner | Hitachi | Hitachi 12352 | 1.5 ton split AC | 15.0 | Numbers | R$ | 189000.0 |
| MRF Tyres | MRF | MRF 013 | 12 inch tyre | 21.0 | Numbers | $ | 1365.0 |
| | | | | | | | **1450135.0** |

## 15.15 Payroll Bank Report

Payroll bank report shows the list of users, routing number, account number and total salary.

**Adhi software**
**Payroll Bank Report**
**Wages Period Month:** 2019-07-01

| # | Name | Routing Number | Account Number | Net |
|---|---|---|---|---|
| 1 | arun t | SYB000898 | 6568321478569 | Rp18500.0 |
| 2 | tharma t | SYB000898 | 7458962148569 | Rp18500.0 |
| 3 | munish m | SYB000898 | 456781245799 | Rp18500.0 |
| 4 | Ruban Adhi | SYB000898 | 6100089803 | Rp22900.0 |
| 5 | Mohan Adhi | SYB000443 | 61007777443 | Rp18500.0 |
| 6 | Deepan Adhi | SYB000444 | 6100022206 | Rp18500.0 |
| 7 | Redmine Admin | | | Rp27300.0 |
| | | | **Total:** | **Rp142700.0** |

## 15.16 User Utilization Report

User utilization report shows the list of users and utilization percentage.

**Chennai**
**User Utilization Report**
01-May-2019 To 30-Jun-2019

| User | 2019 May | 2019 Jun | Average |
|---|---|---|---|
| Redmine | 31.15% | 55.88% | 43.52% |
| mohan | 100.0% | 68.85% | 84.43% |
| arun | 75.0% | 100.0% | 87.5% |
| ruban | 80.77% | 36.84% | 58.81% |
| **Average** | 71.73% | 65.39% | 68.56% |

## 15.17 Project Profitability

Project profitability report shows the list of billable projects, revenue, expense and profit percentage.

**Adhi Software Pvt LTD**
**Project Profitability**
June 01, 2019 To July 31, 2019

| Project | 2019 Jun | 2019 Jul | Total |
|---|---|---|---|
| **Vitals** | | | |
| Revenue | R$ 29760.00 | R$ 14880.00 | **R$ 44640.00** |
| Expense | R$ 3671.43 | R$ 3671.43 | **R$ 7342.86** |
| Profit | R$ 26088.57 (87.66%) | R$ 11208.57 (75.33%) | **R$ 37297.14 (83.55%)** |
| **Legend** | | | |
| Revenue | R$ 0.00 | R$ 43400.00 | **R$ 43400.00** |
| Expense | R$ 0.00 | R$ 9178.57 | **R$ 9178.57** |
| Profit | R$ 0.00 (0%) | R$ 34221.43 (78.85%) | **R$ 34221.43 (78.85%)** |
| **Training Connection** | | | |
| Revenue | R$ 28000.00 | R$ 7000.00 | **R$ 35000.00** |
| Expense | R$ 13965.00 | R$ 13965.00 | **R$ 27930.00** |
| Profit | R$ 14035.00 (50.13%) | R$ -6965.00 (0%) | **R$ 7070.00 (20.2%)** |
| **Total** | | | |
| **Profit** | **R$ 40123.57 (69.47%)** | **R$ 38465.00 (58.92%)** | **R$ 78588.57 (63.87%)** |

## 15.18 AP Aging

AP Aging report shows the list of invoice, payment and balance.

**Adhi software**
**AP Aging**
June 01, 2020 To July 31, 2020

| Supplier Account | | Previous Balance | 2020 Jun | 2020 Jul | Current Balance |
|---|---|---|---|---|---|
| **Star computers** | | | | | |
| | Invoice | | $ 0.00 | $ 0.00 | |
| | Payment | | $ 0.00 | $ 0.00 | |
| | Balance | $ 12000.00 | $ 0.00 | $ 0.00 | $ 12000.00 |
| **Poorvika Mobiles** | | | | | |
| | Invoice | | $ 0.00 | $ 0.00 | |
| | Payment | | $ 0.00 | $ 0.00 | |
| | Balance | $ 150000.00 | $ 0.00 | $ 0.00 | $ 150000.00 |
| **DELL** | | | | | |
| | Invoice | | $ 0.00 | $ 0.00 | |
| | Payment | | $ 0.00 | $ 0.00 | |
| | Balance | $ 8100.00 | $ 0.00 | $ 0.00 | $ 8100.00 |
| **Acer** | | | | | |
| | Invoice | | $ 0.00 | $ 0.00 | |
| | Payment | | $ 0.00 | $ 0.00 | |
| | Balance | $ 47750.00 | $ 0.00 | $ 0.00 | $ 47750.00 |
| | | | | **Total:** | $ 217850.00 |

## 15.19 Tax Report

Tax Report shows the list of Employee ID, Tax ID, User name, Gross, TDS, Cess and Total.

Adhi software
Tax Report
29-Jun-2020 To 05-Jul-2020

June 2020

| Employee ID | Tax ID | User Name | Gross | TDS | Cess | Total |
|---|---|---|---|---|---|---|
| | Total | | | | | |

July 2020

| Employee ID | Tax ID | User Name | Gross | TDS | Cess | Total |
|---|---|---|---|---|---|---|
| | | Redmine | 137700.0 | 19388 | 775.0 | 20163.0 |
| 73 | | anitha | 72700.0 | 4655 | 186.0 | 4841.0 |
| | | Mohan | 51200.0 | 2051 | 82.0 | 2133.0 |
| | Total | | 261600.0 | 26094 | 1043.0 | 27137.0 |

## 15.20 PF Report

PF Report shows the list of Employee UAN, User name, basic pay and Total.

**Adhi software**
**PF Report**
**Wages Period Month:** November 01, 2021

| Sl. No. | UAN | Name | Wages | | | | Contribution Remitted | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | Basic Pay | EPS | EPF | EDLI | EE | EPS | ER |
| 1 | | Redmine Admin | 105000 | 15000 | 15000 | 15000 | 12600 | 1250 | 551 |
| 2 | | Ruban Adhi | 10000 | 10000 | 10000 | 10000 | 1200 | 833 | 367 |
| 3 | | Mohan Adhi | 1000000 | 15000 | 15000 | 15000 | 120000 | 1250 | 551 |
| 4 | | Deepan Adhi | 11000 | 11000 | 11000 | 11000 | 1320 | 916 | 404 |
| 5 | | anitha adhi | 90000 | 15000 | 15000 | 15000 | 10800 | 1250 | 551 |
| 6 | | Amutha adhi | 20000 | 15000 | 15000 | 15000 | 2400 | 1250 | 551 |
| 7 | | Abirami adhi | 10000 | 10000 | 10000 | 10000 | 1200 | 833 | 367 |
| 8 | | Sathiya adhi | 25000 | 15000 | 15000 | 15000 | 3000 | 1250 | 551 |
| 9 | | Mugesh adhi | 50000 | 15000 | 15000 | 15000 | 6000 | 1250 | 551 |
| 10 | 34576890987 | Meena adhi | 50000 | 15000 | 15000 | 15000 | 6000 | 1250 | 551 |
| 11 | | dinesh kumar | 10000 | 10000 | 10000 | 10000 | 1200 | 833 | 367 |
| 12 | | Visa arun | 10000 | 10000 | 10000 | 10000 | 1200 | 833 | 367 |
| 13 | | Riya ani | 50000 | 15000 | 15000 | 15000 | 6000 | 1250 | 551 |
| 14 | | Ranjith adhi | 2500000 | 15000 | 15000 | 15000 | 300000 | 1250 | 551 |
| 15 | | Visa adhi | 5000000 | 15000 | 15000 | 15000 | 600000 | 1250 | 551 |
| 16 | | priya Adhi | 10000 | 10000 | 10000 | 10000 | 1200 | 833 | 367 |
| 17 | | lidiyah adhi | 2500000 | 15000 | 15000 | 15000 | 300000 | 1250 | 551 |
| 18 | | RANDAL TOMMIE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Total** | | | 11451000 | 226000 | 226000 | 226000 | 1374120 | 18831 | 8300 |

### 15.21 Cash Flow Report

Cashflow Report shows the view of inflow and outflow of cash transaction. Date range filter can be applied. Accounting Admins and Users have the privilege to view the data.

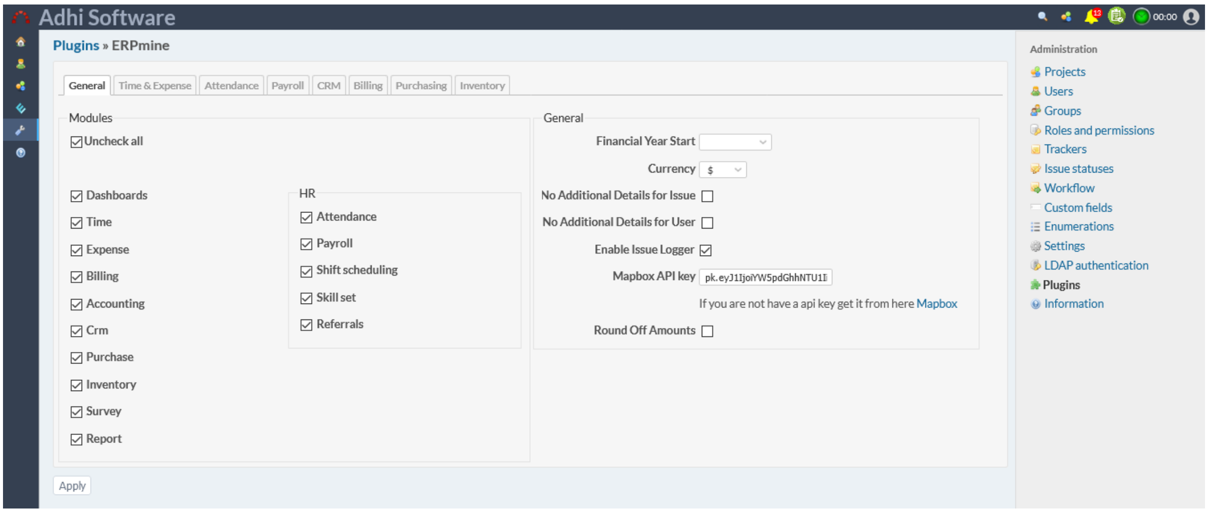
**Chennai**
**Cash Flow Report**
April 01, 2021 To April 30, 2021

| Particulars | Cash Inflow | Cash Outflow |
|---|---|---|
| **Current Liabilities** | **4462.00** | **34462.00** |
| EPFO | 2100.00 | 2100.00 |
| ESIC | 2362.00 | 2362.00 |
| **Sundry Creditors** | | |
| PC World | | 30000.00 |
| **Fixed Assets** | **400.00** | **3500.00** |
| Computer | 400.00 | 2500.00 |
| Furniture | | 1000.00 |
| **Current Assets** | **115000.00** | **50000.00** |
| **Deposits(Asset)** | | |
| CB Bank Deposit | | 50000.00 |
| **Sundry Debtors** | | |
| Client Legend | 25000.00 | |
| Client Silver | 80000.00 | |
| Client Vi | 10000.00 | |
| **Direct Expenses** | **0.00** | **35000.00** |
| Salary | | 35000.00 |
| **Indirect Incomes** | **1500.00** | **0.00** |
| Interest Income | 1500.00 | |
| **Indirect Expenses** | **0.00** | **4000.00** |
| Internet Charges | | 1500.00 |
| Tax | | 2500.00 |
| **Sales Accounts** | **15650.00** | **0.00** |
| Software Development | 15650.00 | |
| **Purchase Accounts** | **0.00** | **10000.00** |
| Software Purchase | | 10000.00 |
| **Total:** | **137012.0** | **136962.0** |
| **Net InFlow:** | | **50.0** |

# 16 ERPmine Configuration

The plugin can be configured from the plugin settings page to customize its behavior. Only the administrators can access this page. The plugin configurations are grouped into the following categories; Display Settings, Export (for pdf) Settings, Time & Expense Settings, Approval system Settings, Attendance Settings.The settings defined in the "Time & Expense" section are common to both timesheet and expense sheet. There is also separate section for "Time" and "Expense" as well, the settings which comes under these sections is specific to its sheets.

## 16.1 General Settings



#### *16.1.1 Modules*

This setting can be used to enable / disable the various modules; Expense, Attendance, Payroll, Billing, Accounting, CRM, Purchase, Inventory and Reports.

#### *16.1.2 Company Name*

This settings can be used to define the organization name.

#### *16.1.3 Address*

This settings can be used to define the organization address.

#### *16.1.4 Header Logo*

This settings can be used to define the organization address.

### 16.2 Time & Expense Settings

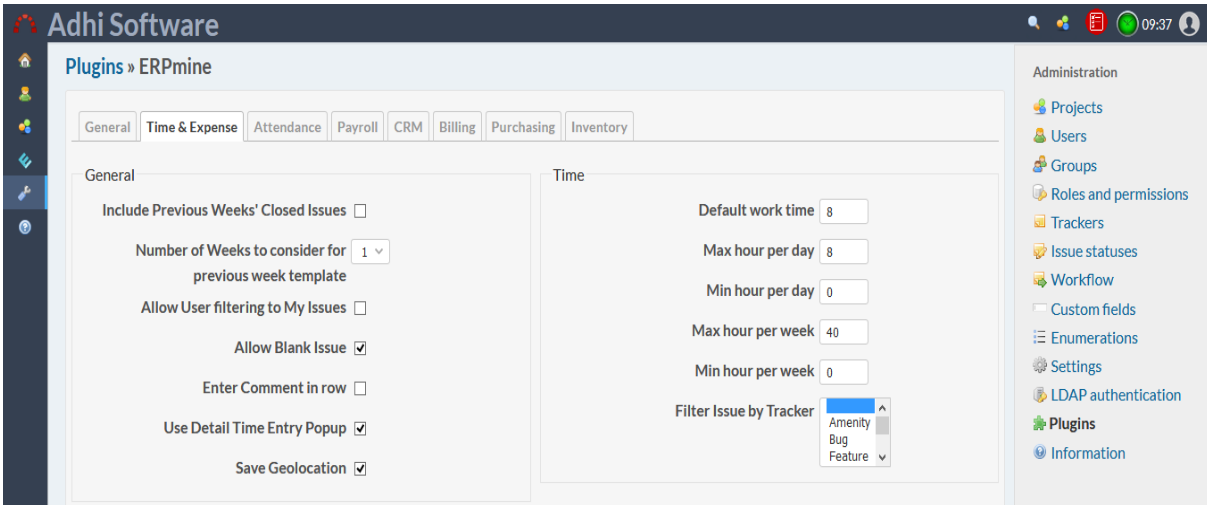
#### *16.2.1 Include closed issues*

The issue dropdown in time & expense sheet will list only issues which are currently open. To include previous weeks closed issues, enable the configuration "**Include Previous Weeks' Closed Issues**". **Note:** The current week's closed issue is not affected by this setting it will always be shown in issue dropdown. Instead of

#### *16.2.2 Previous week template*

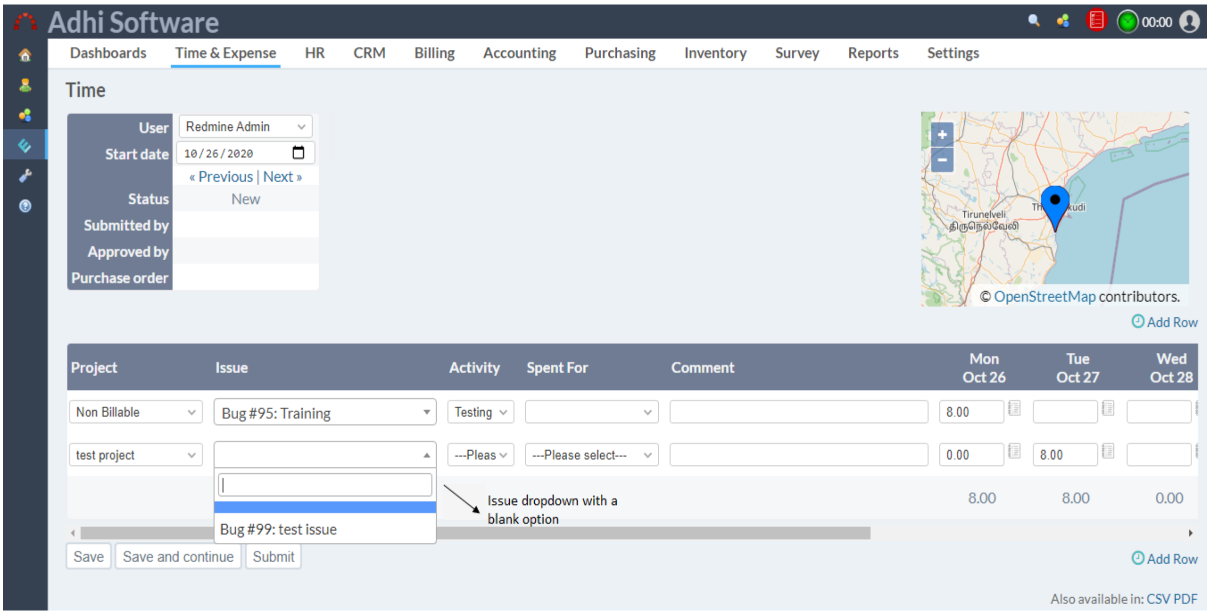
There is a checkbox "**Previous week's template**" in the new time & expense sheet page. If it is checked, then previous week(s) time & expense sheet are used as template.

The number of previous weeks to be considered can be configured using the setting "**Number of Weeks to consider for previous week template**".



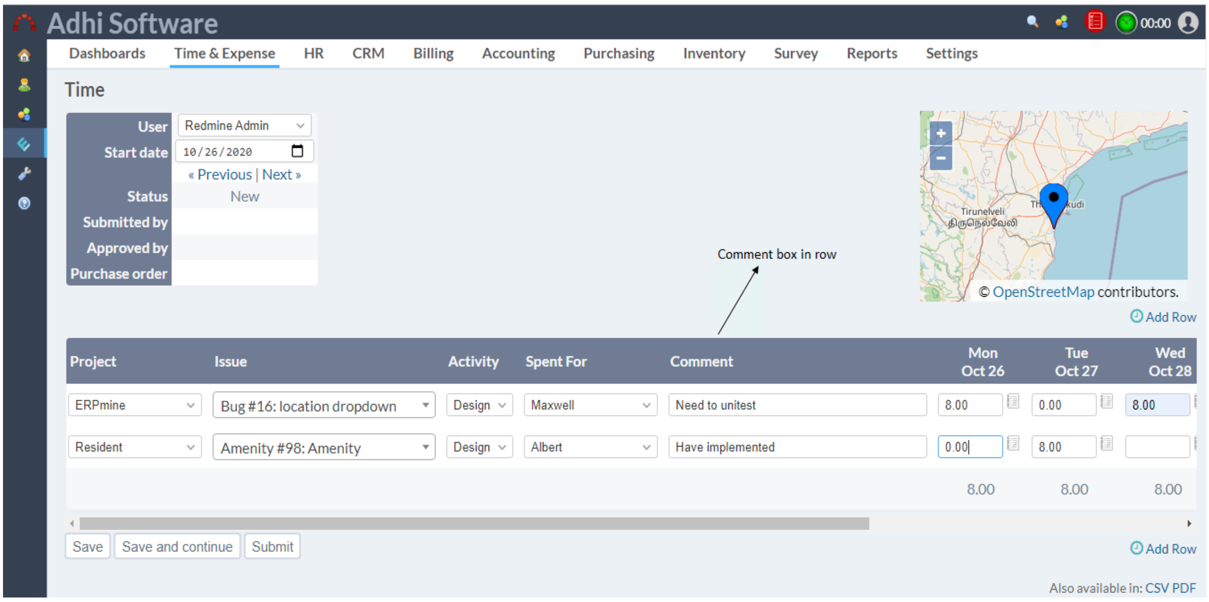

#### *16.2.3 Allow Blank Issue*

This Time & Expense entries can be made directly to a project without selecting an issue and this configuration "**Allow Blank Issue**" enables that. If the setting is enabled, then the issue dropdown in time & expense sheet will contain a blank option.



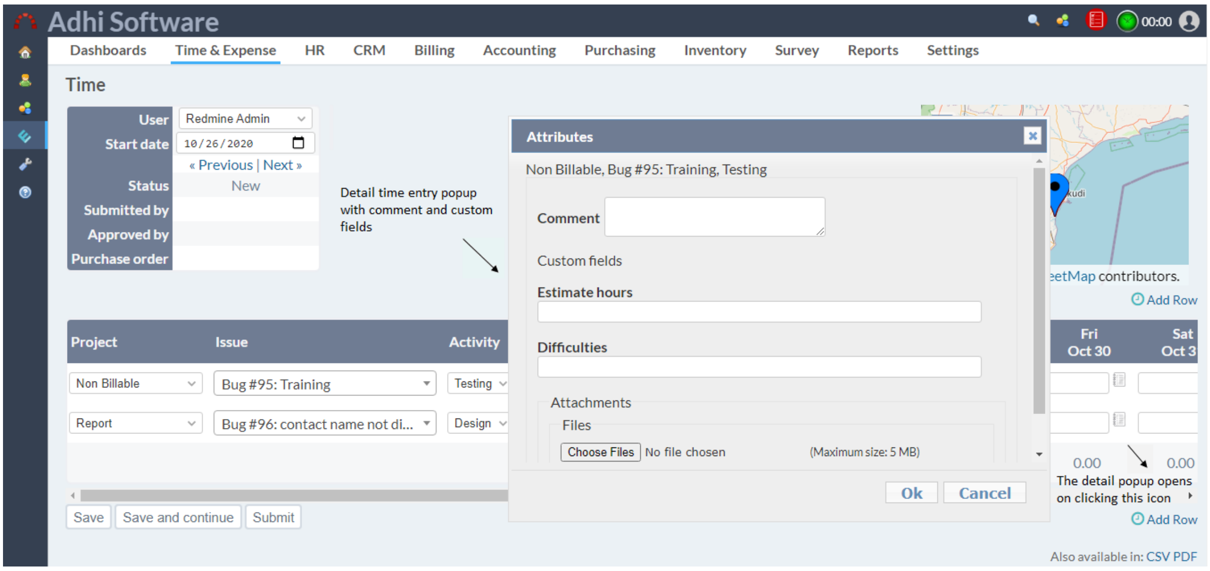

### 16.2.4 Enter comment in row

Comments on the Time & expense entries can be made from the row using this setting "**Enter comment in row**". This setting allows for one common comment for all entries of a unique combination of project, issue and activity. If this setting is enabled, there will be a textbox for comment in addition to project, issue and activity dropdown.




### 16.2.5 Use detail time entry popup

The setting "**Use Detail Time Entry Popup**" can be used for entering additional details like comments and custom fields on a time & expense entry. If this setting is enabled there will be an icon next to the entry textbox for popping up a detail dialog box.




### *16.2.6 Allow User Filtering to My Issues.*

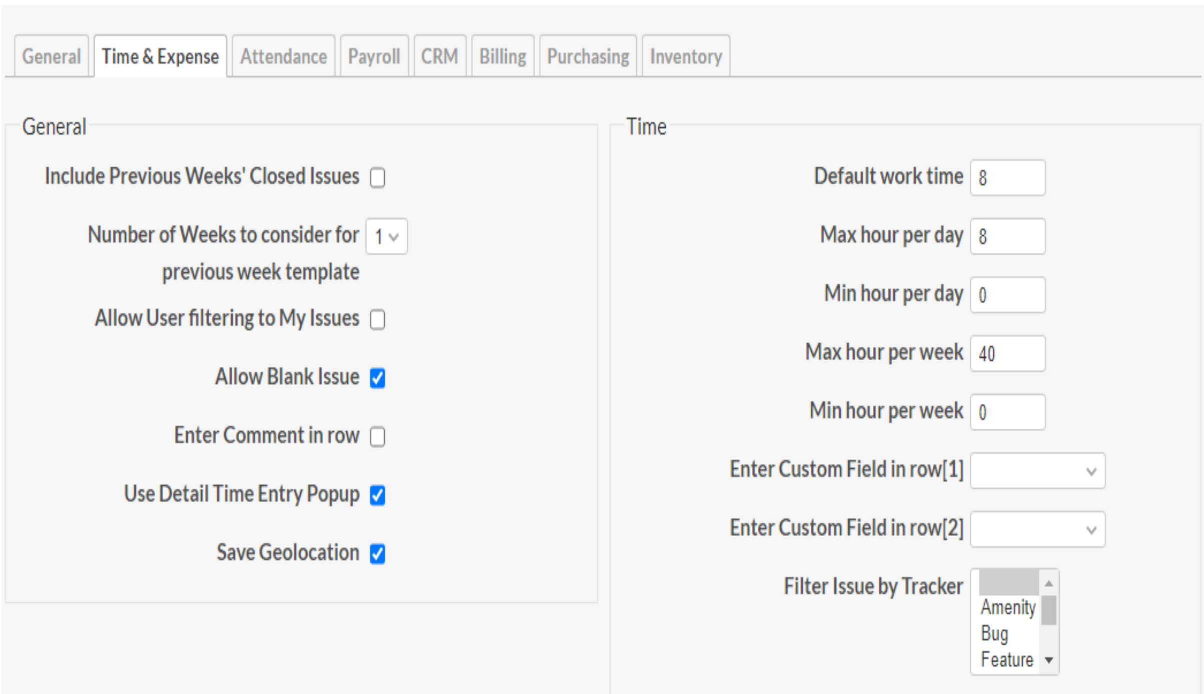
The setting, "**Allow User Filtering to My Issues**" is used to filter issues which are assigned to them or created by them in time and expense sheet page.

### *16.2.7 T&E Admin Groups*

The user group configured as "**T&E Admin Groups**" can view, modify and approve time and expense sheets without needing to have time tracking permissions.

### *16.2.8 Time Settings*

The settings which are defined under this section are only applicable to Time sheets.



#### *16.2.9 Default work time*

The setting "**Default work time"** is used to define the standard work time for a day.

#### *16.2.10 Max hour*

Maximum hour for a day can be configured using the "**Max hour per day**". This is used to limit the number of hours entered by a project member for a particular day.

#### *16.2.11 Enforce Max hour*

If the setting "**Enforce Max hour per week/day**" is enabled, then validation will be done to check if the entered time per week/day is above the configured "**Max hour per week/day**". Min hour
Minimum hour for a day can be configured using the "**Min hour per week/day**".

#### *16.2.12 Enforce Min hour*

If the setting "**Enforce Min hour per week/day**" is enabled, then validation will be done to check if the entered time per week/day is below the configured "**Min hour per week/day**". Validation will be done only for working days, for the non-working days validation will be done only when user enters time.
**Note**:

1. The min and max configurations are only applicable to timesheet.
2. Non working days can be configured by Administrators (Administration ➔ Settings ➔ Issue tracking ➔ Non-working days).
3. Min hour settings are applicable only when approval system is enabled
4. By default, the max hour is 8 and min hour is 0
5.

#### *16.2.13 Filter Issue by Tracker*

Tracker is basically used to categorize issues like bug, support etc. This setting "**Filter Issue by Tracker**" is set separately for time and expense. This setting is used to restrict entries for a particular set of trackers. Only the issues from the configured trackers will be listed in the issue dropdown in the time & expense sheet. If no tracker is configured then all issues will be listed in issue dropdown.

#### *16.2.14 Allow User Filtering by Tracker*

If "**Allow User Filtering by Tracker**" is enabled then user will be allowed to override the tracker filters set on the settings page.

#### *16.2.15 Enter custom field in row1*

The spent time custom fields are entered through detail popup by default. The setting "**Enter custom field in row1**" can be used to enter a spent time custom field from the row. If this setting is enabled, there will be a textbox for the configured custom field in the row.

#### *16.2.16 Enter custom field in row2*

Same as Enter custom field in row1

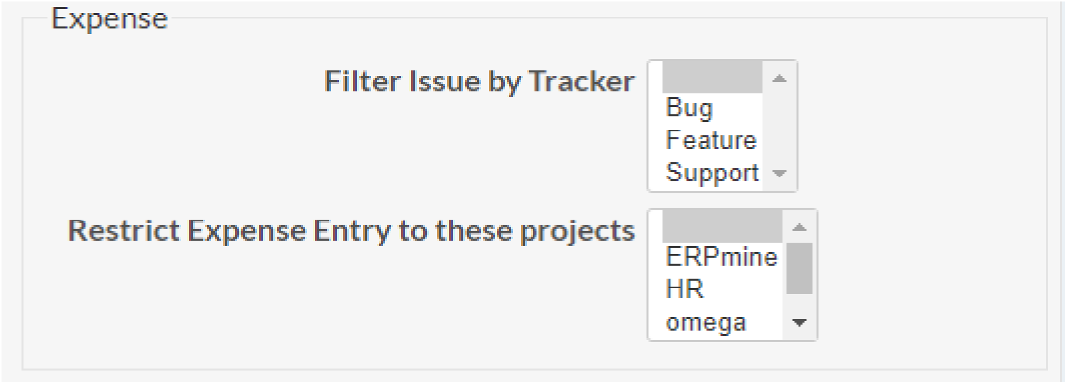
**Note:** Only two spent time custom fields can be configured to be in the row.

#### *16.2.17 Expense Settings*

This section contains settings that are applicable to expense sheets.

#### *16.2.18 Expense project*

Separate list of projects can be configured for Expenses. This setting "**Restrict Expense Entry to these projects**" allow members to charge expense only to those configured projects

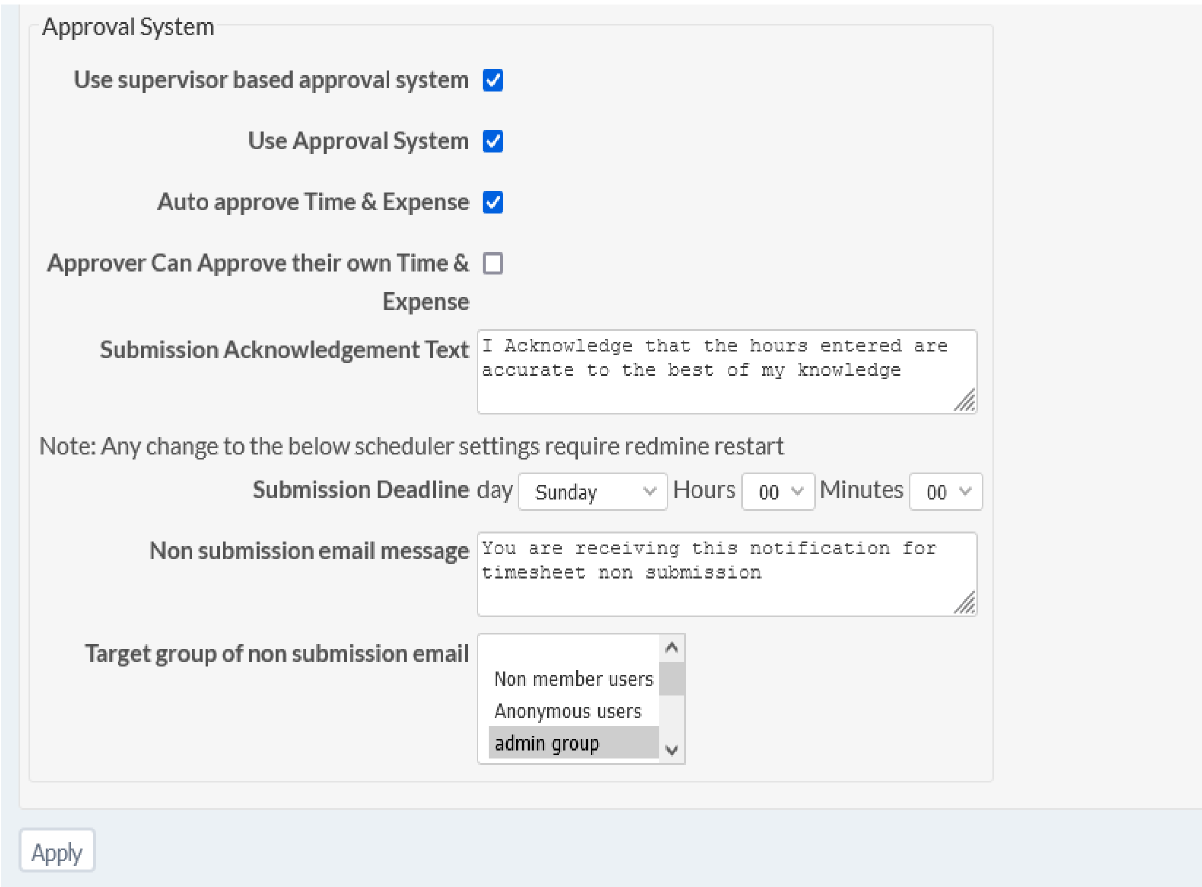



### *16.2.19 Filter issue by tracker*

This is same as section 6.5.5, but in the context of an expense.

### *16.2.20 Approval Settings*

Approval System is common to both Time and Expense except for email notification on non-submission which is applicable only to Time.



### *16.2.21 Use Approval System*

If “**Use Approval System**” checkbox is checked, then approval system is used.

### *16.2.22 Use Supervisor based approval system*

Enable this setting for the supervisor based approval system.

### *16.2.23 Auto approve*

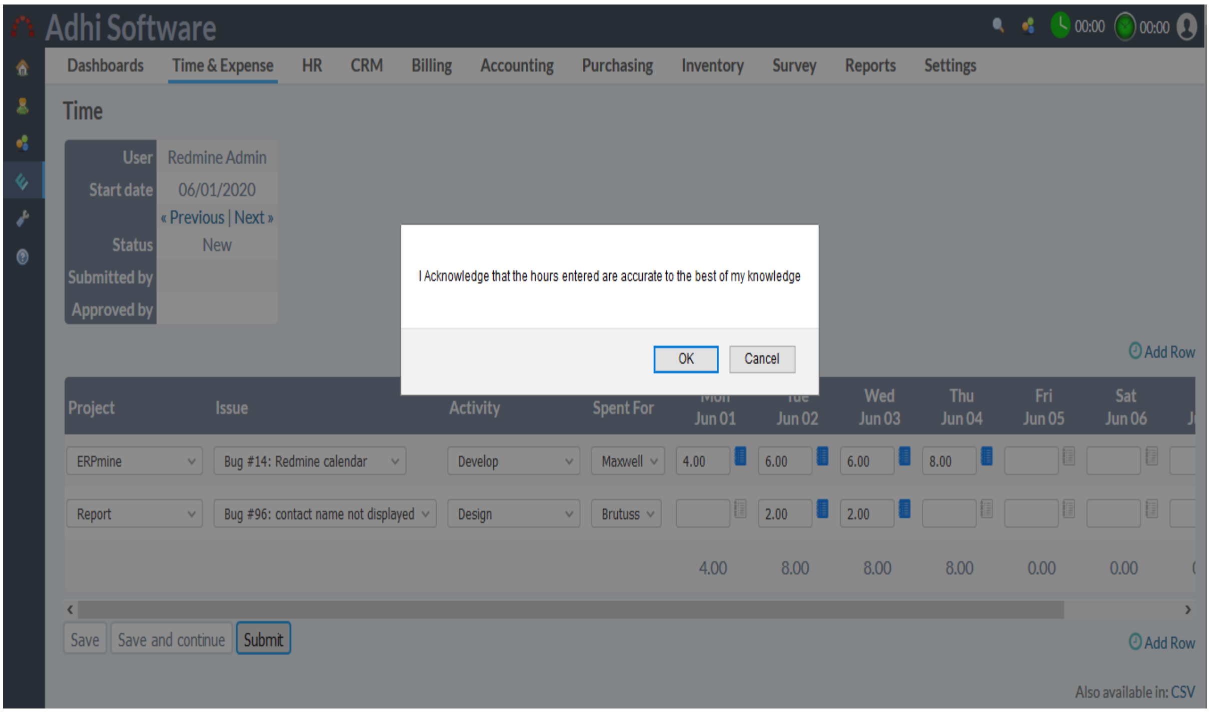
If the setting “**Auto approve Time & Expense**” is enabled, then the time & expense sheet gets automatically approved by the system upon submission. The project member will not be able to unsubmit his time/expense sheet since it goes to approved status immediately.

### *16.2.24 Approve own time/expense sheet*

If the setting “**Approver Can Approve their own Time & Expense**” is enabled, then the project member with “Approve time logs” along with “Edit time logs” and “Log spent time” permission can approve/reject his own time & expense sheet.

### *16.2.25 Submission ask text*

The text given in the “**Submission Acknowledgement Text**” is the message shown during Time & Expense submission. (Please see screenshot below)



#### *16.2.26 Email Notification*

The setting "**Send email for non submission"** is only for Time. If this setting is enabled, then an email notification is sent to the user about his non submission of timesheet for the previous week. Please note, this setting required rufus scheduler gem.

#### *16.2.27 Submission deadline*

Timesheet submission deadline can be configured using this setting. If the project member did not submit the timesheet before the configured ""**Submission Deadline**", then an email notification is sent only when the configuration "**Send email for non submission"** is enabled.

#### *16.2.28 Non submission email message*

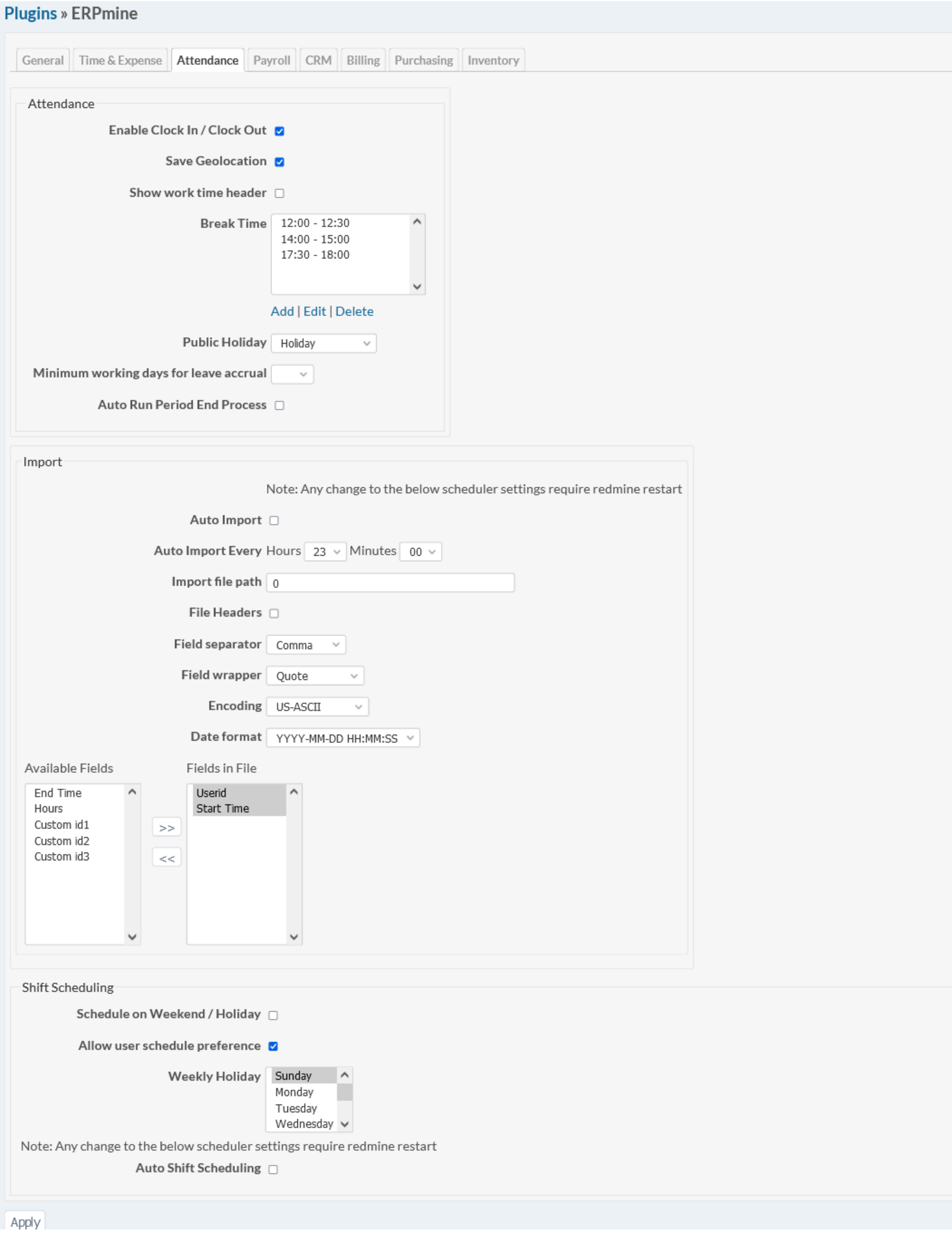
The text for non-submission timesheet notification email message can be configured using the setting "**Non submission email message**". The text given in this configuration will be used as the email message body along with the name of the project member, submission deadline, the week for which timesheet need to be submitted.

#### *16.2.29 Target group of Non submission email*

The selected user group will receive the email notification about non submission of timesheet for the previous week.

### 16.3 Attendance

This section contains settings that are applicable to Attendance. To auto-populate, the public holiday entries in a timesheet need to configure the plugin setting.




### *16.3.1 Enable clock in / clock out*

This setting can be used to enable and disable attendance module for Time & Attendance.

### *16.3.2 Break Time*

Break time for a day like lunch intervals can be configured.

#### *16.3.3 Auto import*

Attendance data can be automatically imported from an external system / Attendance Devices / Time Clocks. The following file formats are supported.

**Type 1:**

| Userid | Punchtime |
|---|---|
| 1 | 10:00 |
| 2 | 10:00 |
| 1 | 20:00 |
| 2 | 20:00 |

**Type 2:**

| Userid | Clockintime | Clockouttime |
|---|---|---|
| 1 | 10:00 | 20:00 |
| 2 | 10:15 | 18:15 |
| 3 | 10:30 | 19:30 |

**Type 3:**

| Userid | Clockintime | Clockouttime | hours |
|---|---|---|---|
| 1 | 10:00 | 20:00 | 10.0 |
| 2 | 10:15 | 18:15 | 8.0 |
| 3 | 10:30 | 19:30 | 9.0 |

#### *16.3.4 Auto Import Every*

This is the frequency of the Attendance Import Scheduler,. Auto import will be performed for the given interval of time.

#### *16.3.5 Import File Path*

This setting is used to locate the file directory for attendance auto import.

#### *16.3.6 File Headers*

This setting is used to determine whether import file contains header or not.

#### *16.3.7 Field Separator*

The Type of field separator used in import file .

#### *16.3.8 Field Wrapper*

The type of field wrapper used in import file.

#### *16.3.9 Encoding*

The type of encoding used in import file

*16.3.10 Date format*

The date time format used in import file.

*16.3.11 Available Fields and Fields in File*

This settings shows the list of available fields,. User can pick the fields used in the import file from the available fields .

*16.3.12 Schedule on weekend / Holiday*

If the setting is enabled then it will schedule on weekends and holidays.

*16.3.13 Allow user schedule preference*

If the settings is enabled then it will consider the employee preference for scheduling

*16.3.14 Weekly Holidays*

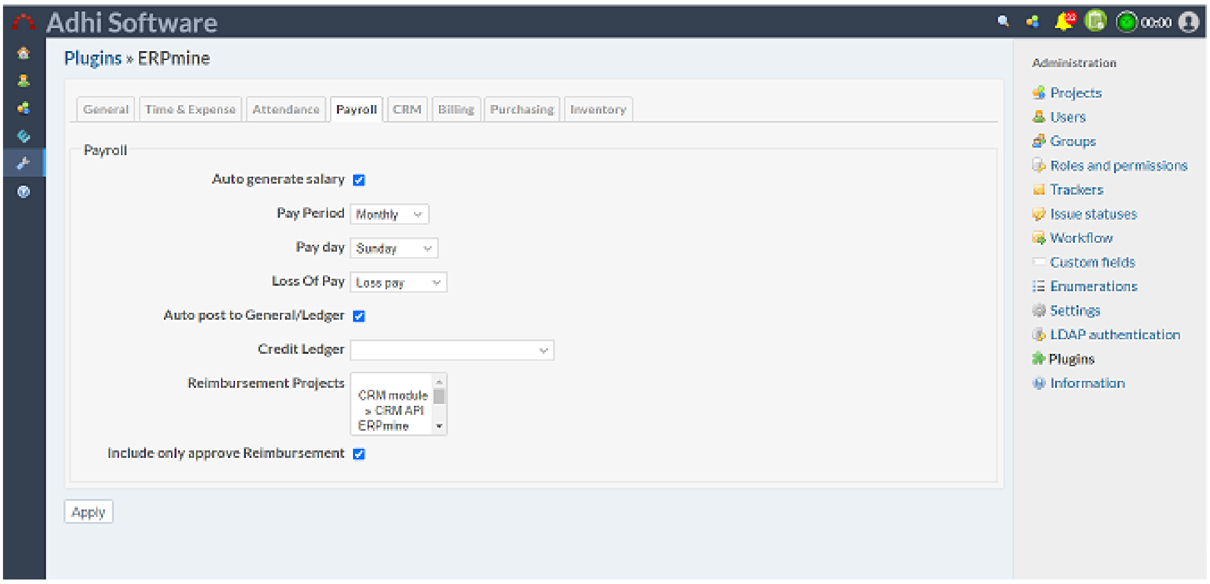
Allow the user to select their weekly holidays

*16.3.15 Auto Shift Scheduling*

The scheduler job automatically schedule shifts at the beginning of the month.

## 16.4 Payroll

This section contains settings that are applicable to Payroll.




*16.4.1 Auto generate salary*

The scheduler job runs automatically to generate the salary at the end of pay period.

*16.4.2 Currency*

This settings can be used to set currency of payroll.

### *16.4.3 Pay Period*

User can set the pay period.

### *16.4.4 Pay day*

If the pay period is weekly or bi-weekly then the user set the pay day.

### *16.4.5 Loss of pay*

The selected leave (from Attendance Module) will be used as loss of pay.

### *16.4.6 Financial year start*

They user can set the financial start.

### *16.4.7 Auto post to General/Ledger*

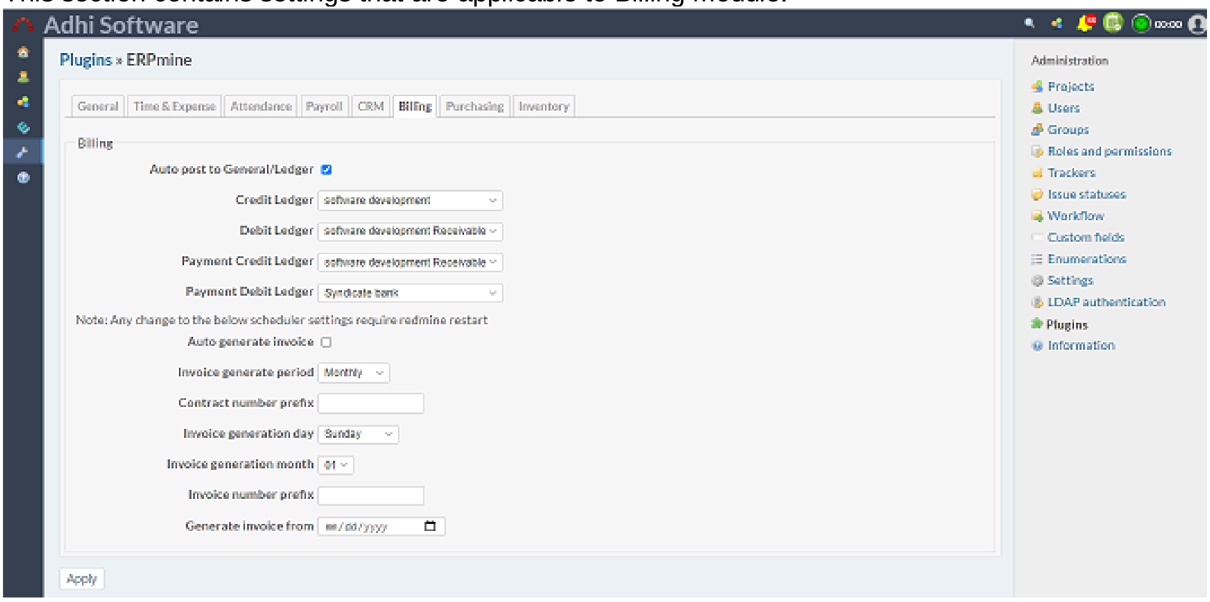
When salary is generated, it will automatically post salary to General Ledger

### *16.4.8 Credit Ledger*

Select the appropriate CL credit ledger for the above setting

## 16.5 Billing

This section contains settings that are applicable to Billing module.




### *16.5.1 Auto post to General/Ledger*

When generate invoice post invoice to financial transaction.

### *16.5.2 Credit Ledger*

Select the appropriate transaction credit ledger.

### *16.5.3 Debit Ledger*

Select the appropriate transaction debit ledger.

#### *16.5.4 Payment Credit Ledger*

Select the appropriate transaction credit ledger for payments.

#### *16.5.5 Payment Debit Ledger*

Select the appropriate transaction debit ledger for payments.

#### *16.5.6 Auto generate invoice*

The scheduler job runs automatically to generate the invoice at the end of invoice period.

#### *16.5.7 Invoice generation period*

User can set the invoice generation period (Quarterly, Monthly, bi-weekly, weekly etc).

#### *16.5.8 Contract number prefix*

This setting can be used to add prefix of contract number.

#### *16.5.9 Invoice generation day*

If the invoice generation period is weekly or bi-weekly then the user should set the invoice generation day.

#### *16.5.10 Invoice number prefix*

This setting can be used to add prefix of invoice number.

#### *16.5.11 Generate Invoice from*

The user can set the invoice start date for the whole of application

#### *16.5.12 Time Entry Billing id*

The user has to select spent time custom field to store the billed invoice item id for the appropriate spent time.

#### *16.5.13 Rate fields*

The Billing rate can come from Project or User. Project custom fields can be mapped to the following settings

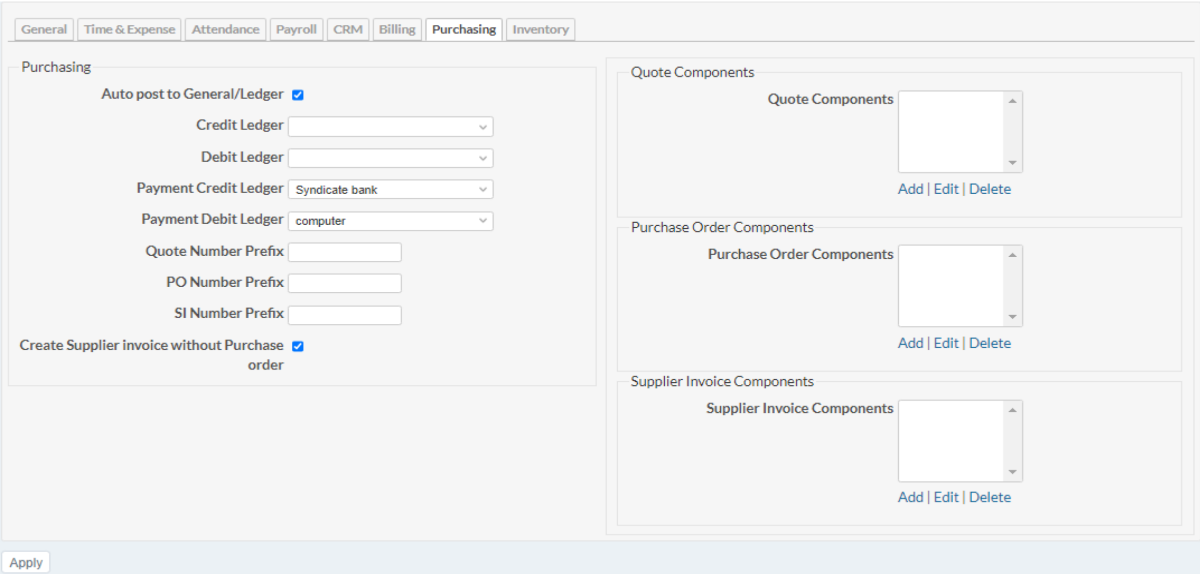
- Project Billing Rate
- Project Billing Currency

User custom fields can be mapped to the following settings

- User Billing Rate
- User Billing Currency

### 16.6 Purchasing

This section contains settings that are applicable to Purchase module.




### *16.6.1 Auto post to General/Ledger*
When supplier invoice and supplier payment post to financial transaction.

### *16.6.2 Credit Ledger*
Select the appropriate transaction credit ledger.

### *16.6.3 Debit Ledger*
Select the appropriate transaction debit ledger.

### *16.6.4 Payment Credit Ledger*
Select the appropriate transaction credit ledger for payments

### *16.6.5 Payment Debit Ledger*
Select the appropriate transaction debit ledger for payments

### *16.6.6 Quote Number Prefix*
This setting can be used to add prefix of quote number.

### *16.6.7 PO Number Prefix*
This setting can be used to add prefix of purchase order number.

### *16.6.8 SI Number Prefix*
This setting can be used to add prefix of supplier invoice number.

### *16.6.9 Quote Components*
The user can configure the quote components, which will be included in the printed quote.

#### *16.6.10 Purchase Order Components*

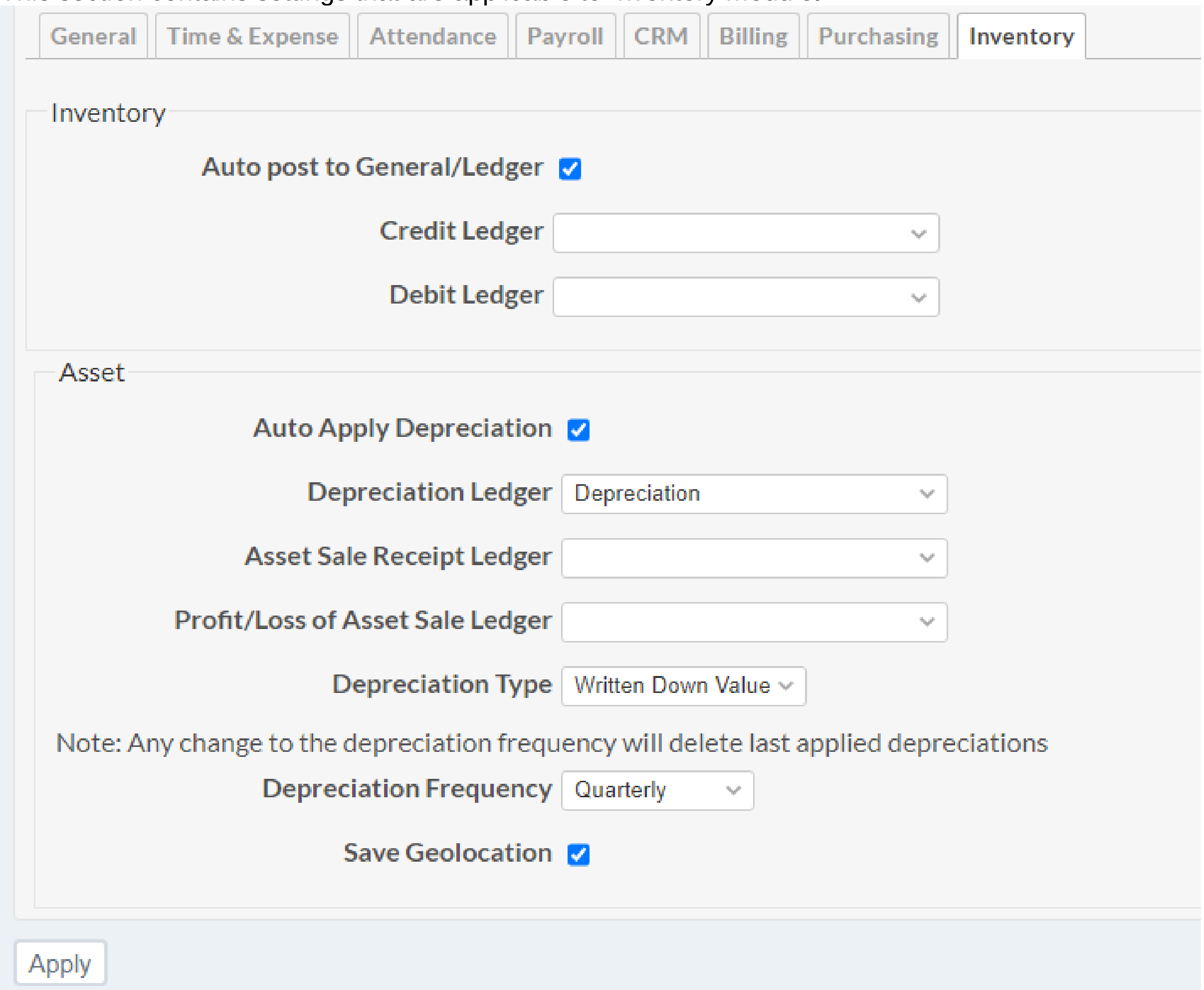
The user can configure the purchase order components, which will be included in the printed purchase order.

#### *16.6.11 Supplier Invoice Components*

The user can configure the supplier invoice components, which will be included in the printed supplier invoice.

### 16.7 Inventory

This section contains settings that are applicable to Inventory module.




#### *16.7.1 Auto post to General/Ledger*

When supplier invoice and supplier payment post to financial transaction.

#### *16.7.2 Credit Ledger*

Select the appropriate transaction credit ledger.

### *16.7.3 Debit Ledger*

Select the appropriate transaction debit ledger.

### *16.7.4 Auto Apply Depreciation*

The scheduler job runs automatically to generate the asset depreciation at the end of depreciation period.

### 16.7.5 Depreciation Ledger

The depreciation ledger in accounting module

### 16.7.6 Depreciation Type

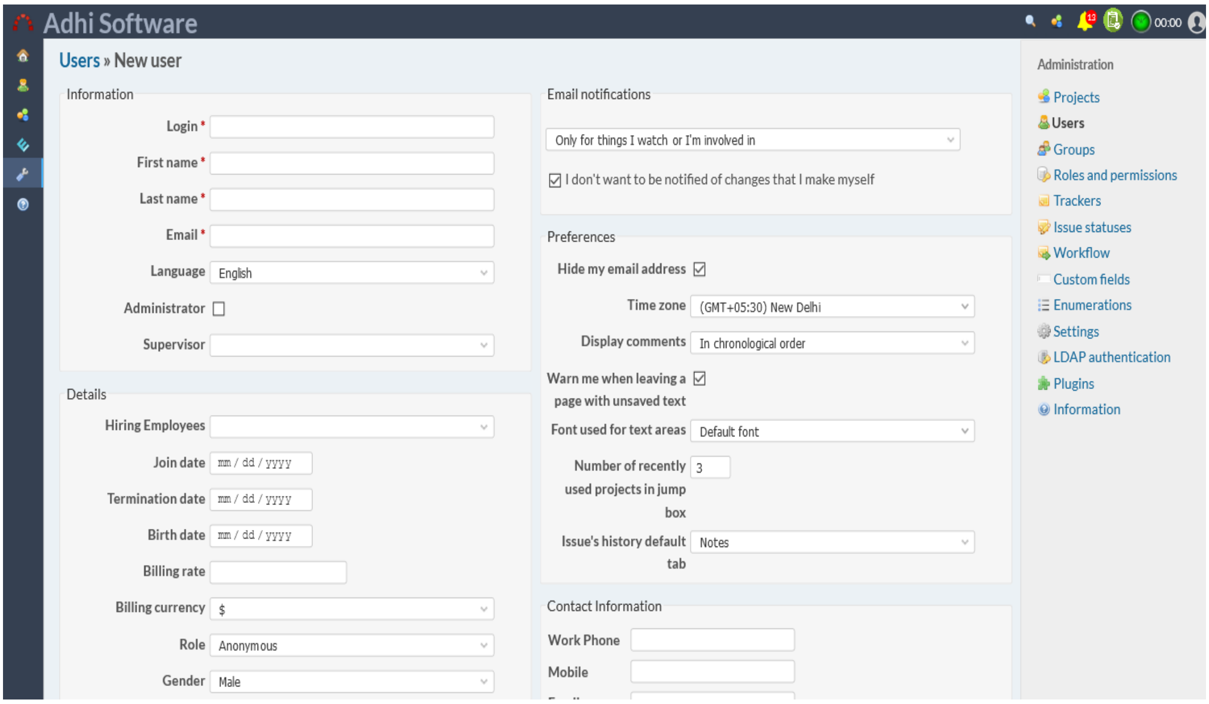
Type of depreciation; Straight line and WDV.

### 16.7.7 Depreciation Frequency

User can set the depreciation frequency.

## 16.8 ERPmine User fields

Add some additional fields for user like bank name, bank code, account number, joining date, termination date and etc




**Fields Description:**
**Supervisor:** Approves timesheet and leave request.
**Join date:** Displayed in payroll Reports, Attendance Reports, payslip Report.
**Termination Date:** If the Termination date is filled in, the payroll will not be generated for him.
**Birthdate:** Displayed in Attendance Reports.
**Billing Rate and Billing Currency:** used in the Billing module to generate an invoice.
**Role:** Used in the shift schedule configuration and displayed in Attendance Reports.
**Gender:** Displayed in payroll Report.
**Department:** Used in the shift schedule configuration

**Location:** used in the timesheet
**Bank name, Bank account number, Bank code:** Displayed in a payroll bank report
**Tax ID:** Displayed in Tax report.

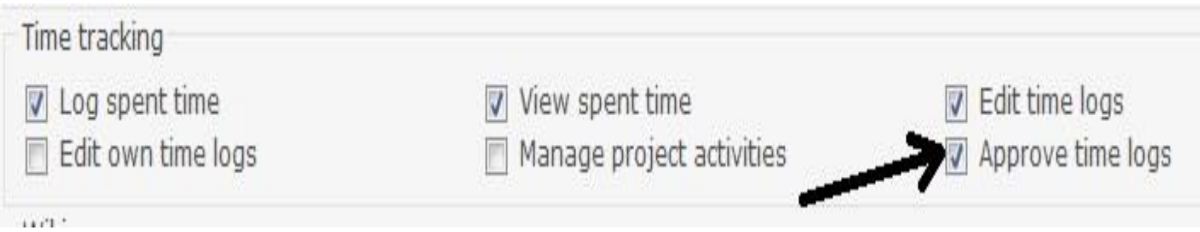
If you would like to have a demo setup, we will show you a demo of erpmine.

# 17 Redmine Settings

There are certain settings of redmine which influence the functionality of the Time & Attendance plugin.

## 17.1 Approve Time Logs Permission

A new permission is defined for time/expense sheets approval under "Time tracking" section. The user with "**Approve time logs"** permission along with "**Log spent time"** and "**Edit time logs**" permission will be allowed to approve, unapproved and reject time and expense sheets.




## 17.2 Permissions

The roles (manager, developer, reporter etc.) defined in redmine are assigned with a set of permissions. Administrators can define roles and configure its permission by navigating to **Administration → Roles and permissions**. A user can have different roles for different projects and also a project member can have multiple roles assigned for single project.

Following are some of the privileges defined in time tracking section **[An extract from Redmine User Guide]**

Log spent time - Allow user to log time on the project
View spent time - Allow user to view the time logs on the project
Edit time logs - Allow user to edit any time log
Edit own time logs - Edit own time logs

The rules for Roles and permission in the context of **Time & Attendance** is as follows

| # | Feature | View spent time | Log spent time | Edit own time logs | Edit time logs | Approve time logs |
|---|---|---|---|---|---|---|
| 1 | Time & Expense List Page | X | X | | | |
| 2 | New Time & Expense Sheet | | X | | | |
| 3 | Edit Time & Expense sheet | | | X | | |
| 4 | Edit Other's Time & Expense Sheet | | | | X | |
| 5 | Approve Time & Expense | | | | | X |

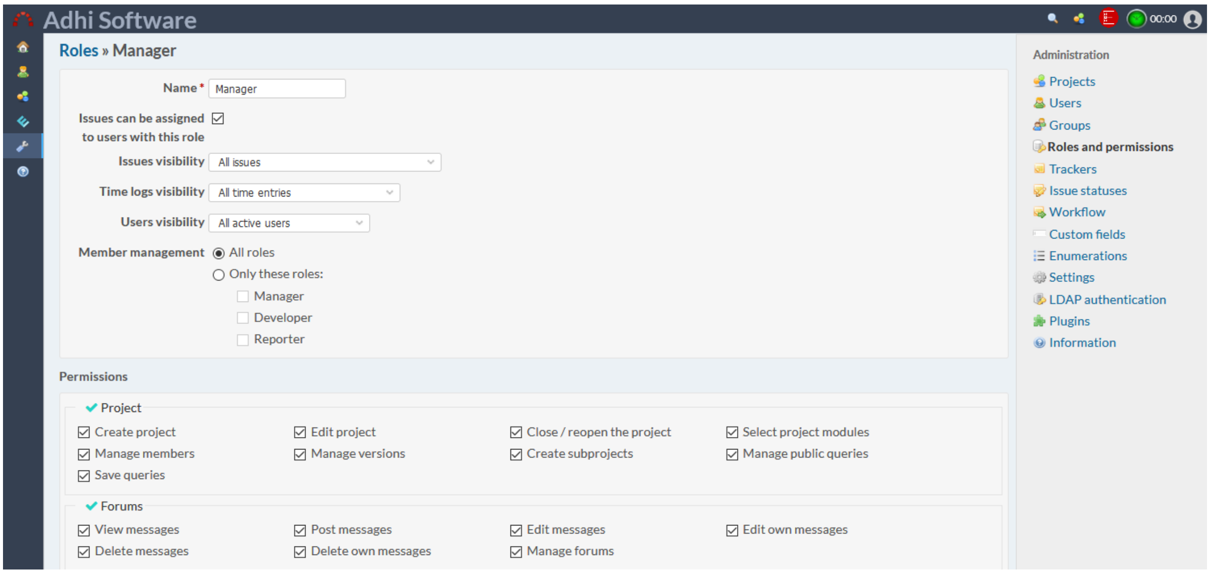
**Note:** There are no separate permissions for expense; it uses the permission from spent time.

*17.2.1 Issue Visibility*

*17.2.2*

**[An extract from Redmine User Guide]**
Assuming that the role includes the *View Issues* permission, the following rules apply to the issues of the projects the user is tied to through it:

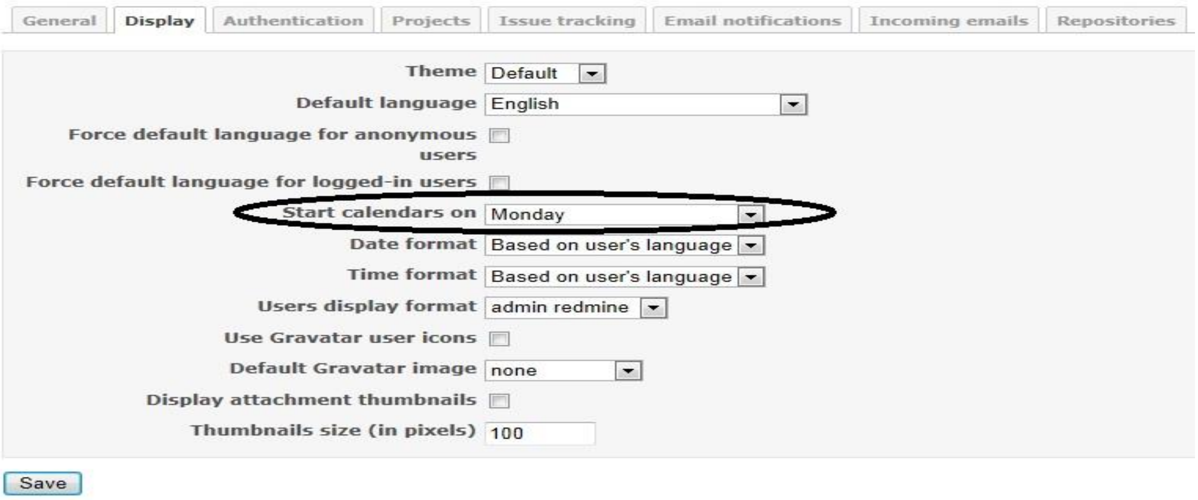
*All issues* - the user can see all issues. This is the default.
*All non-private issues* - the user can see all issues which are not marked as private.
*Issues created by or assigned to the user* - the user can only see issues created by or assigned to




The issue dropdown in time & expense sheet is filled based on Issue Visibility for the role assigned to project members.

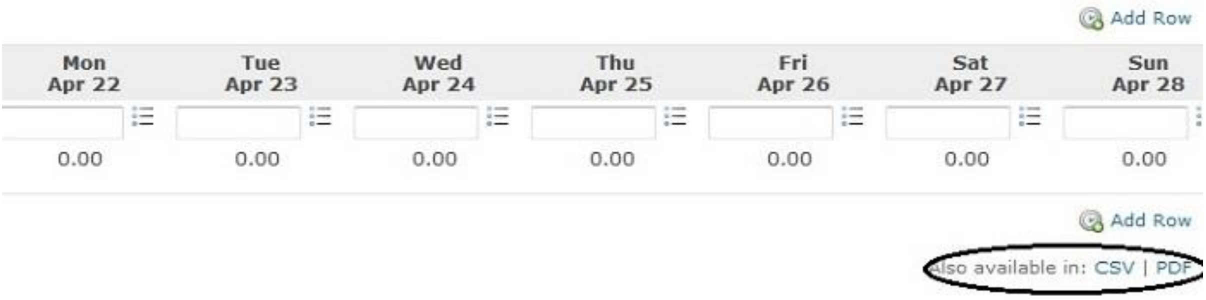
## 17.3 Start of Week

The start of the week on the time & expense sheet can be configured to “Monday”, “Saturday”, “Sunday” or “Based on user’s language”. If “Based on user’s language” is set, then start of the week will be the day defined in language file. Start of the week can be configured by navigating to **Administration ➔ Settings ➔ Display ➔ Start calendars on**



## 18 PDF & CSV

The time & expense sheet can be exported into pdf or csv format. To take a csv report click on the csv link and for the pdf report click on the pdf link.




## 19 REST API

REST API supports both XML and JSON. It supports following functionalities.

- Get list of time/expense sheets
- Get list of clock in/out
- Get list of payroll
- Get list of leads
- Get list of account
- Get list of activities
- Get list of contacts
- Get list of invoice
- Get list of payments
- Create time/expense sheet
- Update a time/expense sheet
- Update a clock in/out
- Update a leads
- Update a account
- Update a activities
- Update a contacts
- Update a invoice
- Update a payment
- Get a time/expense sheet
- Get a clock in/out
- Get a payroll
- Get a lead
- Get a account
- Get a activities
- Get a contact
- Get a invoice
- Get a payment
- Delete a time/expense sheet
- Delete a time/expense entries

API requires authentication for each request. The API validates the user and responds with 401 Unauthorized if user did not have required privileges. For more information on REST API configuration, please refer to the Redmine User Guide.

### 19.1 List Time/Expense Sheets

List of time and expense sheet can be retrieved using the list API. By default, it returns time and expense sheets for the current month if the parameters **from** and **to** is not specified. If **from** and **to** is specified then time/expense sheet for that particular date range will be retrieved.

| Request Type | URL | Response |
|---|---|---|
| **Time** | | |
| GET | /wktime/index.xml?user_id=9999&from=2013-01-01<br>&to=2013-08-01 | list of timesheets of a user as xml/json |
| | /wktime/index.json?user_id=9999 | |
| **Expense** | | |
| GET | /wkexpense/index.xml?user_id=9999<br>&from=2013-01-01&to=2013-08-01 | list of expense sheets of a user as xml/json |
| | /wkexpense/index.json?user_id=9999<br>&from=2013-01-01&to=2013-08-01&limit=25&offset=0 | |
| | /wkpayment/index.json? | |

| **Parameters** | | |
|---|---|---|
| user_id | Required | user_id for whom, list of time/expense sheet is to be retrieved.<br>If 0, then list of time/expense sheets for all user is retrieved. |
| project_id | Optional | Use this parameter, if project_id = 0 is used i.e., to get list of time/expense sheets of all user under one project. |
| from | Optional | Specifies the start of the date range |
| to | Optional | Specifies the end of the date range |
| offset | Optional. | The offset of the first item to retrieve |
| limit | Optional | Number of records to be retrieved (default is 25) |

**Sample XML Response (for Time)**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<wk_times total_count="1" offset="0" limit="25" type="array">
  <wk_time>
    <user id="5" name="Chandra Durairaj"/>
    <hours>10.0</hours>
    <startdate>2013-08-19</startdate>
    <status>n</status>
  </wk_time>
</wk_times>
```

User Billing Rate
User Billing Currency

**Sample JSON Response (for Time)**

```json
{"wk_times":
        [
                {
                 "user":{"id":5,"name":"Chandra Durairaj"},
                 "hours":10.0,
                 "startdate":"2013-08-19",
                 "status":"n"
                }
        ],
```

"total_count":1,
"offset":0,
"limit":25
}

## 19.2 List of Clock in/out

List of Clock in/out can be retrieved using the list API. By default, it returns clock in/out for the current month if the parameters from and to is not specified. If from and to is specified then clock in/out for that particular date range will be retrieved.

| Request Type | URL | Response |
|---|---|---|
| Clock in/out | | |
| GET | /wkattendance/clockindex.xml?user_id=1<br>/wkattendance/clockindex.json?user_id=1 | list of Clockin/out sheets of a user as xml/json |

| Parameters | | |
|---|---|---|
| group_id | Optional | group_id for whom, list of clock in/out is to be retrieved. |
| user_id | Required | user_id for whom, list of clock in/out is to be retrieved. |
| Period | Optional | Specifies the date range |
| from | Optional | Specifies the start of the date range |
| to | Optional | Specifies the end of the date range |

**Sample XML Response (for Clock in/out)**

```
&lt;wk_clkinout total_count="32" offset="0" limit="25" type="array"&gt;
&lt;clk_entry&gt;
&lt;id&gt;1&lt;/id&gt;
&lt;name&gt;Redmine Admin&lt;/name&gt;
&lt;clock_in&gt;00:00&lt;/clock_in&gt;
&lt;clock_out&gt;09:44&lt;/clock_out&gt;
&lt;hours&gt;9.74&lt;/hours&gt;
&lt;startdate&gt;2020-06-30&lt;/startdate&gt;
&lt;/clk_entry&gt;
&lt;/wk_clkinout&gt;
```

**Sample JSON Response (for Clock in/out)**

```
{
  “wk_clkinout”: [
    {
      “id”: 1,
      “name”: “Redmine Admin”,
      “clock_in”: “00:00”,
      “clock_out”: “09:44”,
      “hours”: 9.74,
      "startdate": "2020-06-30"
    },]
}
```

## 19.3 List of Payroll

List of payroll can be retrieved using the list API. By default, it returns payroll for the current month if the parameters from and to is not specified. If from and to is specified then payroll for that particular date range will be retrieved.

| **Request Type** | **URL** | **Response** |
|---|---|---|
| Payroll | | |
| GET | /wkpayroll/index.xml?user_id=1 | list of Payroll of a user as xml/json |
| | /wkpayroll/index.json?user_id=1 | |

| **Parameters** | | |
|---|---|---|
| group_id | Optional | group_id for whom, list of payroll is to be retrieved. |
| User_id | Required | user_id for whom, list of payroll is to be retrieved. |
| Period | Optional | Specifies the date range |
| from | Optional | Specifies the start of the date range |
| to | Optional | Specifies the end of the date range |

**Sample XML Response (for Payroll)**


```
<wk_payroll type="array">
<entry>
<id>1</id>
<name>Redmine Admin</name>
<joindate>2018-03-01</joindate>
<saldate>2020-06-01</saldate>
<basic>60000</basic>
<allowance>32700</allowance>
<deduction>9289</deduction>
<currency>$</currency>
</entry>
</wk_payroll>
```


**Sample JSON Response (Payroll)**


```
{
  "wk_payroll": [
    {
      "id": 1,
      "name": "Redmine Admin",
      "joindate": "2018-03-01",
      "saldate": "2020-06-01",
      "basic": 60000,
      "allowance": 32700,
      "deduction": 9289,
      "currency": "$"
    },]
}
```


### 19.4 List of Leads

List of Leads can be retrieved using the list API. By default, it returns payroll for the current month if the parameters from and to is not specified. If from and to is specified then payroll for that particular date range will be retrieved.

| **Request Type** | **URL** | **Response** |
|---|---|---|
| Lead | | |

| | | |
|---|---|---|
| GET | /wklead/index.xml<br>/wklead/index.json | list of Lead of a user as xml/json |

| **Parameters** | | |
|---|---|---|
| lead_name | Optional | lead_name for whom, list of lead is to be retrieved. |
| status | Optional | Status of the lead |
| Location-id | Optional | Specifies the location of the lead |

**Sample XML Response (for Leads)**


```
<wk_lead total_count="5" offset="0" limit="25" type="array">
<entry>
<id>128</id>
<status>New</status>
<name> Martin</name>
<location>Adhi software</location>
</entry>
</wk_lead>
```


**Sample JSON Response (for Leads)**


```
{
  "wk_lead": [
    {
      "id": 128,
      "status": "New",
      "name": " Martin",
      "location": "Adhi software"
    },]
}
```


### 19.5 List of Accounts

List of accounts can be retrieved using the list API. By default, it returns accounts for the current month if the parameters from and to is not specified. If from and to is specified then accounts for that particular date range will be retrieved.

| **Request Type** | **URL** | **Response** |
|---|---|---|
| Accounts | | |
| GET | /wkcrmaccount/index.xml<br>/wkcrmaccount/index.json | list of account of a user as xml/json |

| **Parameters** | | |
|---|---|---|
| location_id | Optional | Specifies the location of the account |
| Account name | Optional | Specifies the account name |

**Sample XML Response (for Accounts)**


```
<wk_accounts total_count="12" offset="0" limit="25" type="array">
<entry>
<id>78</id>
<name>Maxwell</name>
<location>Adhi software</location>
```

```
<address>KK nagar</address>
<work_phone/>
<country/>
<city/>
</entry>
</wk_accounts>
```


**Sample JSON Response (for Accounts)**


```
{
  "wk_accounts": [
    {
      "id": 78,
      "name": "Maxwell",
      "location": "Adhi software",
      "address": "KK nagar",
      "work_phone": "",
      "country": "",
      "city": ""
    },]
}
```


## 19.6 List of Activities

List of activities can be retrieved using the list API. By default, it returns activities for the current month if the parameters from and to is not specified. If from and to is specified then activities or that particular date range will be retrieved.

| Request Type | URL | Response |
|---|---|---|
| Activity | | |
| GET | /wkcrmactivity/index.xml | list of activity of a user as xml/json |
| | /wkcrmactivity/index.json | |

| Parameters | | |
|---|---|---|
| activity_type | Optional | activity_type for whom, list of activity is to be retrieved. |
| Related_to | Optional | Specifies the related to activity |
| period | Optional | Specifies the date range |
| from | Optional | Specifies the start of the date range |
| to | Optional | Specifies the end of the date range |

**Sample XML Response (for Activities)**


```
<wk_activities total_count="1" offset="0" limit="25" type="array">
<entry>
<id>44</id>
<activity_type>Meeting</activity_type>
<subject>API Call</subject>
<status>Planned</status>
<related_to>Account</related_to>
<start_date>2020-05-18 00:00:00</start_date>
<end_date>2020-05-18 00:00:00</end_date>
</entry>
```

```
</wk_activities>
```


**Sample JSON Response (for Activities)**


```
{
  "wk_activities": [
    {
      "id": 44,
      "activity_type": "Meeting",
      "subject": "API Call",
      "status": "Planned",
      "related_to": "Account",
      "start_date": "2020-05-18 00:00:00",
      "end_date": "2020-05-18 00:00:00"
    }
  ],
  "total_count": 1,
  "offset": 0,
  "limit": 25
}
```


## 19.7 List of Contacts

List of contacts can be retrieved using the list API. By default, it returns contacts for the current month if the parameters from and to is not specified. If from and to is specified then contacts or that particular date range will be retrieved.

| Request Type | URL | Response |
|---|---|---|
| Contacts | | |
| GET | /wkcrmcontact/index.xml | list of contact of a user as xml/json |
| | /wkcrmcontact/index.json | |

| Parameters | | |
|---|---|---|
| contact_name | Optional | Specifies the contact name |
| Account_id | Optional | Specifies the account_id |
| Location_id | Optional | Specifies the location_id |

**Sample XML Response (for Contacts)**


```
<wk_contact total_count="11" offset="0" limit="25" type="array">
<entry>
<id>188</id>
<name> Test2</name>
<location>Adhi software</location>
<title/>
</entry>
</wk_contact>
```


**Sample JSON Response (for Contacts)**


```
{
  "wk_contact": [
    {
```

```
    "id": 188,
    "name": " Test2",
    "location": "Adhi software",
    "title": ""
  },]
}
```


### 19.8 List of invoice

List of invoice can be retrieved using the list API. By default, it returns invoice for the current month if the parameters from and to is not specified. If from and to is specified then invoice or that particular date range will be retrieved.

| Request Type | URL | Response |
|---|---|---|
| Invoice | | |
| GET | /wkinvoice/index.xml | list of invoice of a user as xml/json |
| | wkinvoice/index.json | |

| Parameters | | |
|---|---|---|
| contact_id | Optional | contact_id for whom, list of invoice is to be retrieved |
| Account_id | Optional | account_id for whom, list of invoice is to be retrieved |
| Project_id | Optional | project_id for whom, list of invoice is to be retrieved |
| Period | Optional | Specifies the date range |
| from | Optional | Specifies the start of the date range |
| to | Optional | Specifies the end of the date range |

**Sample XML Response (for invoice)**


```
<wk_invoice total_count="106" offset="0" limit="25" type="array">
<entry>
<id>1</id>
<invoice_number>1</invoice_number>
<name> Bhargav </name>
<project>survey</project>
<invoice_date>2019-03-08</invoice_date>
<start_date>2019-01-01</start_date>
<end_date>2019-03-31</end_date>
<status>open</status>
<quantity>5.5000</quantity>
<amount>714.00</amount>
<org_amount>714.00</org_amount>
<org_currency>$</org_currency>
<currency>$</currency>
</entry>
</wk_invoice>
```


**Sample JSON Response (for invoice)**


```
{
  "wk_invoice": [
   {
    "id": 1,
    "invoice_number": "1",
```

"name": " Bhargav ",
"project": "survey",
"invoice_date": "2019-03-08",
"start_date": "2019-01-01",
"end_date": "2019-03-31",
"status": "open",
"quantity": "5.5000",
"amount": "714.00",
"org_amount": "714.00",
"org_currency": "$",
"currency": "$"
},]

}

## 19.9 List of Payments

List of Payments can be retrieved using the list API. By default, it returns Payments for the current month if the parameters from and to is not specified. If from and to is specified then Payments or that particular date range will be retrieved.

| Request Type | URL | Response |
|---|---|---|
| Payment | | |
| GET | /wkpayment/index.xml | list of payment of a user as xml/json |
| | /wkpayment/index.json | |

| Parameters | | |
|---|---|---|
| contact_id | Required | contact_id for whom, list of payment is to be retrieved. |
| Account_id | Optional | account_id for whom, list of payment is to be retrieved. |
| period | Optional | Specifies the date range |
| from | Optional | Specifies the start of the date range |
| to | Optional | Specifies the end of the date range |

**Sample XML Response (for Payments)**

```
&lt;wk_payment total_count="10" offset="0" limit="25" type="array"&gt;
&lt;entry&gt;
&lt;id&gt;158&lt;/id&gt;
&lt;name&gt;Addisson&lt;/name&gt;
&lt;type&gt;Account&lt;/type&gt;
&lt;payment_date&gt;2020-05-20&lt;/payment_date&gt;
&lt;payment_type&gt;cash&lt;/payment_type&gt;
&lt;amount&gt;100078.00&lt;/amount&gt;
&lt;org_amount&gt;100078.00&lt;/org_amount&gt;
&lt;org_currency&gt;$&lt;/org_currency&gt;
&lt;currency&gt;$&lt;/currency&gt;
&lt;/entry&gt;
&lt;/wk_payment&gt;
```

**Sample JSON Response (for Payments)**

{
"wk_payment": [

```
  {
    "id": 158,
    "name": "Addisson",
    "type": "Account",
    "payment_date": "2020-05-20",
    "payment_type": "cash",
    "amount": "100078.00",
    "org_amount": "100078.00",
    "org_currency": "$",
    "currency": "$"
  },]
}
```


### 19.10 Create Time/Expense Sheet

Time and Expense sheet can be created using the create API.

| Request Type | URL | Response |
|---|---|---|
| **Time** | | |
| POST | /wktime/update.xml?wktime_save=Save | 200 OK.<br>Creates a timesheet |
| | /wktime/update.xml?wktime_submit=Submit | 200 OK.<br>Creates and set timesheet to submitted state |
| **Expense** | | |
| POST | /wkexpense/update.xml?wktime_save=Save | 200 OK.<br>Creates an expensesheet |
| | /wkexpense/update.json?wktime_submit=Submit | 200 OK.<br>Creates and set expensesheet to submitted state |

**Sample XML data to be posted (for Time)**


```
<?xml version="1.0" encoding="UTF-8"?>
<wk_time>
 <user name="Chandra Durairaj" id="5"/>
 <startday>2013-08-19</startday>
 <status>New</status>
 <total>6.0</total>
 <custom_fields type="array">
  <custom_field name="Purchase Order" id="12">
    <value/>
  </custom_field>
</custom_fields>
 <time_entries type="array">
  <time_entry>
    <project name="Vitals Software" id="1"/>
    <issue id="1"/>
    <user name="Chandra Durairaj" id="5"/>
    <activity name="Design" id="10"/>
    <hours>4.0</hours>
    <comments/>
    <spent_on>2013-08-19</spent_on>
    <custom_fields type="array">
     <custom_field name="Difficulties" id="15">
```

```
      <value/>
    </custom_field>
  </custom_fields>
</time_entry>
<time_entry>
  <project name="Vitals Software" id="1"/>
  <issue id="1"/>
  <user name="Chandra Durairaj" id="5"/>
  <activity name="Design" id="10"/>
  <hours>2.0</hours>
  <comments/>
  <spent_on>2013-08-21</spent_on>
  <custom_fields type="array">
    <custom_field name="Difficulties" id="15">
      <value/>
    </custom_field>
  </custom_fields>
</time_entry>
</time_entries>
</wk_time>
```


**Sample XML data to be posted (For Expense)**


```
<?xml version="1.0" encoding="utf-8"?>
<wk_expense>
 <user id="5" name="Chandra Durairaj"/>
 <startday>2013-08-26</startday>
 <status>New</status>
 <total>75.0</total>
 <wk_expense_entries type="array">
  <wk_expense_entry>
   <id>84</id>
   <project id="22" name="Expense"/>
   <issue id="50"/>
   <user id="5" name="Chandra Durairaj"/>
   <activity id="8" name="Design"/>
   <amount>25.0</amount>
   <currency>$</currency>
   <comments/>
   <spent_on>2013-08-26</spent_on>
  </wk_expense_entry>
  <wk_expense_entry>
   <id>85</id>
   <project id="22" name="Expense"/>
   <issue id="51"/>
   <user id="5" name="Chandra Durairaj"/>
   <activity id="8" name="Design"/>
   <amount>50.0</amount>
   <currency>$</currency>
   <comments/>
   <spent_on>2013-08-27</spent_on>
  </wk_expense_entry>
 </wk_expense_entries>
</wk_expense>
```

### 19.11 Update Time/Expense Sheet

Time & Expense sheet can be updated suing update API. Any one of the following action and its value should be sent for the corresponding functionalities.

- wktime_save=Save
- wktime_submit=Submit
- wktime_unsubmit=Unsubmit
- wktime_approve=Approve
- wktime_reject=Reject
- wktime_unapprove=Unapprove

| Request Type | URL | Response |
|---|---|---|
| **Time** | | |
| POST | /wktime/update.xml?wktime_save=Save | 200 OK.<br>Updates a timesheet |
| | /wktime/update.xml?wktime_submit=Submit | 200 OK.<br>Update and set timesheet status to “Submitted” |
| | /wktime/update.xml? wktime_unsubmit=Unsubmit | 200 OK.<br>Updates the timesheet status to “New” |
| | /wktime/update.xml?wktime_approve=Approve | 200 OK.<br>Approves a timesheet |
| | /wktime/update.xml?wktime_reject=Reject | 200 OK<br>Rejects a timesheet |
| | /wktime/update.xml?wktime_unapprove=Unapprove | 200 OK.<br>Unapproves a timesheet |
| **Expense** | | |
| POST | /wkexpense/update.xml?wktime_save=Save | 200 OK.<br>Updates an expensesheet |
| | /wkexpense/update.xml?wktime_submit=Submit | 200 OK.<br>Update and set timesheet status to “Submitted” |
| | /wkexpense/update.xml?wktime_unsubmit=Unsubmit | 200 OK.<br>Updates the expensesheet status to “New” |
| | /wkexpense/update.xml?wktime_approve=Approve | 200 OK.<br>Approves an expensesheet |
| | /wkexpense/update.json?wktime_reject=Reject | 200 OK<br>Rejects an expensesheet |
| | /wkexpense/update.xml?wktime_unapprove=Unapprove | 200 OK.<br>Unapproves an expensesheet |

**Note:** If validation failure occurs, then API response will be **422 Unprocessable Entity** otherwise response will be **200 OK** for successful updates.

**Sample XML data to be posted (for Time)**

In the below sample, two time_entry object contains id and one did not have id, then API will update the two entries which has id and creates a new time_entry for which id is not specified.

```
<?xml version="1.0" encoding="UTF-8"?>
<wk_time>
 <user name="Chandra Durairaj" id="5"/>
```

```
 <startday>2013-08-19</startday>
 <custom_fields type="array">
  <custom_field name="Purchase Order" id="12">
   <value/>
  </custom_field>
       </custom_fields>
 <time_entries type="array">
  <time_entry>
   <id>1506</id>
   <project name="Vitals Software" id="1"/>
   <issue id="1"/>
   <user name="Chandra Durairaj" id="5"/>
   <activity name="Design" id="10"/>
   <hours>4.0</hours>
   <comments/>
   <spent_on>2013-08-19</spent_on>
   <custom_fields type="array">
    <custom_field name="Difficulties" id="15">
     <value/>
    </custom_field>
   </custom_fields>
  </time_entry>
  <time_entry>
   <id>1507</id>
   <project name="Vitals Software" id="1"/>
   <issue id="1"/>
   <user name="Chandra Durairaj" id="5"/>
   <activity name="Design" id="10"/>
   <hours>2.0</hours>
   <comments/>
   <spent_on>2013-08-21</spent_on>
   <custom_fields type="array">
    <custom_field name="Difficulties" id="15">
     <value/>
    </custom_field>
   </custom_fields>
  </time_entry>
  <time_entry>
   <project name="Vitals Software" id="1"/>
   <issue id="2"/>
   <user name="Chandra Durairaj" id="5"/>
   <activity name="Design" id="10"/>
   <hours>4.0</hours>
   <comments/>
   <spent_on>2013-08-20</spent_on>
   <custom_fields type="array">
    <custom_field name="Difficulties" id="15">
     <value/>
    </custom_field>
   </custom_fields>
  </time_entry>
 </time_entries>
</wk_time>
```

## 19.12 Update a Clock in/out

Clock in/out can be update using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Clock in/out** | | |
| POST | /wkattendance/update.xml | 200 OK. |
| | /wkattendance/update.json | |

**Sample XML Response (for clock in/out)**


```
<clock_entries type="array">
<clk_entry>
<id>1217</id>
<clock_in>10:00</clock_in>
<clock_out>20:00</clock_out>
<hours>8.0</hours>
</clk_entry>
</clock_entries>
```


**Sample JSON Response (for clock in/out)**


```
{
  "clock_entries": [
    {
      "id": 1217,
      "clock_in": "10:00",
      "clock_out": "20:00",
      "hours": 8
    }
  ]
}

}
```


## 19.13 Update a leads

Leads can be updated using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Leads** | | |
| POST | /wklead/ update.xml | 200 OK. |
| | /wklead/ update.json | |

**Sample XML Response (for leads)**


```
<lead>
<lead_id>128</lead_id>
<account_id/>
<status>N</status>
```

```
<opportunity_amount/>
<lead_source_id>1</lead_source_id>
<referred_by/>
<contact_id>175</contact_id>
<first_name/>
<last_name>Martin</last_name>
<department/>
<location_id>5</location_id>
<description>default location</description>
<assigned_user_id>0</assigned_user_id>
</lead>
```


**Sample JSON Response (for Leads)**


```
{
 "lead": {
  "lead_id": 128,
  "account_id": null,
  "status": "N",
  "opportunity_amount": null,
  "lead_source_id": 1,
  "referred_by": "",
  "contact_id": 175,
  "first_name": "",
  "last_name": "Martin",
  "department": "",
  "location_id": 5,
  "description": "default location",
  "assigned_user_id": 0
 }
}
```


## 19.14 Update a Accounts

Accounts can be updated using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Accounts** | | |
| POST | /wkcrmaccount/update.xml | 200 OK. |
| | /wkcrmaccount/update.json | |

**Sample XML Response (for Account)**


```
<wk_account>
<account_id>78</account_id>
<account_name>Maxwell</account_name>
<account_billing>false</account_billing>
<account_category>26</account_category>
<location_id>5</location_id>
<description/>
<address id="18" address1="KK nagar" address2="" work_phone="" mobile="" email="" fax="" city=""
country="" state="" pin="" website=""/>
</wk_account>
```

**Sample JSON Response (for Accounts)**


```
{
  "wk_account": {
    "account_id": 78,
    "account_name": "Maxwell",
    "account_billing": false,
    "account_category": "26",
    "location_id": 5,
    "description": "",
    "address": {
      "id": 18,
      "address1": "KK nagar",
      "address2": "",
      "work_phone": "",
      "mobile": "",
      "email": "",
      "fax": "",
      "city": "",
      "country": "",
      "state": "",
      "pin": null,
      "website": ""
    }
  }
}
```


## 19.15 Update a Activities

Activities can be updated using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Activities** | | |
| POST | /wkcrmactivity/update.xml | 200 OK. |
| | /wkcrmactivity/update.json | |

**Sample XML Response (for Activities)**


```
<wk_activity>
<crm_activity_id>44</crm_activity_id>
<activity_type>M</activity_type>
<activity_subject>API Call</activity_subject>
<related_parent>7</related_parent>
<related_to>WkAccount</related_to>
<assigned_user_id/>
<status>P</status>
<activity_direction>I</activity_direction>
<activity_start_date>2020-05-18T00:00:00+05:30</activity_start_date>
<activity_end_date>2020-05-18T00:00:00+05:30</activity_end_date>
<activity_duration>0</activity_duration>
<location/>
```

<activity_description/>
</wk_activity>

**Sample JSON Response (for Activities)**

{
"wk_activity": {
"crm_activity_id": 44,
"activity_type": "M",
"activity_subject": "API Call",
"related_parent": 7,
"related_to": "WkAccount",
"assigned_user_id": null,
"status": "P",
"activity_direction": "I",
"activity_start_date": "2020-05-18T00:00:00+05:30",
"activity_end_date": "2020-05-18T00:00:00+05:30",
"activity_duration": 0,
"location": null,
"activity_description": null
}
}

## 19.16 Update a Contacts

Activities can be updated using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Contacts** | | |
| POST | /wkcrmcontact/update.xml | 200 OK. |
| | /wkcrmcontact/update.xml | |

**Sample XML Response (for Contact)**

<wk_contact>
<contact_id>188</contact_id>
<first_name/>
<last_name>Test2</last_name>
<address_id/>
<contact_title/>
<department/>
<assigned_user_id>0</assigned_user_id>
<salutation/>
<description/>
<related_to>WkCrmContact</related_to>
<related_parent>176</related_parent>
<contact_type>C</contact_type>
<location_id>5</location_id>
<relationship_id>0</relationship_id>
</wk_contact>

**Sample JSON Response (for Contact)**

```
{
  "wk_contact": {
    "contact_id": 188,
    "first_name": "",
    "last_name": "Test2",
    "address_id": null,
    "contact_title": "",
    "department": "",
    "assigned_user_id": 0,
    "salutation": "",
    "description": "",
    "related_to": "WkCrmContact",
    "related_parent": 176,
    "contact_type": "C",
    "location_id": 5,
    "relationship_id": 0
  }
}
```


## 19.17 Update a Invoice

Invoice can be updated using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Payment** | | |
| POST | /wkinvoice/update.xml | 200 OK. |
| | /wkinvoice/update.json | |

**Sample XML Response (for Invoice)**


```
<invoice>
<invoice_id>1</invoice_id>
<invoice_number>1</invoice_number>
<inv_date>2019-03-08</inv_date>
<inv_start_date>2019-01-01</inv_start_date>
<inv_end_date>2019-03-31</inv_end_date>
<field_status>o</field_status>
<modifier_id>1</modifier_id>
<modifier_name>Redmine Admin</modifier_name>
<parent_name> Bhargav </parent_name>
<gl_transaction_id>969</gl_transaction_id>
<parent_type>WkCrmContact</parent_type>
<parent_id>1</parent_id>
<invoice_type>I</invoice_type>
<invoice_num_key>1</invoice_num_key>
<invoiceItemEntries type="array">
<invoice_item>
<item_id>1</item_id>
<name>sprint2</name>
<project_id>1</project_id>
<rate>200.0</rate>
<amount>600.0</amount>
<quantity>3.0</quantity>
<hd_item_type>i</hd_item_type>
```

<currency>$</currency>
<original_currency>$</original_currency>
<original_amount>600.0</original_amount>
</invoice_item>
</invoiceItemEntries>
</invoice>

**Sample JSON Response (for Invoice)**

```
{
 "invoice": {
  "invoice_id": 1,
  "invoice_number": "1",
  "inv_date": "2019-03-08",
  "inv_start_date": "2019-01-01",
  "inv_end_date": "2019-03-31",
  "field_status": "o",
  "modifier_id": 1,
  "modifier_name": "Redmine Admin",
  "parent_name": " Bhargav ",
  "gl_transaction_id": 969,
  "parent_type": "WkCrmContact",
  "parent_id": 1,
  "invoice_type": "I",
  "invoice_num_key": 1,
  "invoiceItemEntries": [
   {
    "item_id": 1,
    "name": "sprint2",
    "project_id": 1,
    "rate": "200.0",
    "amount": "600.0",
    "quantity": 3,
    "hd_item_type": "i",
    "currency": "$",
    "original_currency": "$",
    "original_amount": "600.0"
   },]
}}
```

## 19.18 Update a Payment

Payment can be updated using the update API.

| Request Type | URL | Response |
|---|---|---|
| **Payment** | | |
| POST | /wkpayment/update.xml | 200 OK. |
| | /wkpayment/update.json | |

**Sample XML Response (for Payments)**

<wk_payment>
<payment_id>158</payment_id>
<payment_type_id>36</payment_type_id>

```
<payment_date>2020-05-20</payment_date>
<reference_number/>
<description>AccName:Addisson InvNo:#26 PaymentAmt:$100078.00</description>
<related_parent>1</related_parent>
<related_to>WkAccount</related_to>
<payment_entries type="array">
<entry>
<invoice_id>71</invoice_id>
<payment_item_id>438</payment_item_id>
<credit_issued>false</credit_issued>
<invoice_no>26</invoice_no>
<invoice_org_amount>168.0</invoice_org_amount>
<payment_org_amount>100168.0</payment_org_amount>
<invoice_org_currency>$</invoice_org_currency>
<paid_amount>100078.0</paid_amount>
<amount>100078.0</amount>
</entry>
</payment_entries>
</wk_payment>
```


**Sample JSON Response (for Payment)**


```
{
 "wk_payment": {
  "payment_id": 158,
  "payment_type_id": 36,
  "payment_date": "2020-05-20",
  "reference_number": "",
  "description": "AccName:Addisson InvNo:#26 PaymentAmt:$100078.00",
  "related_parent": 1,
  "related_to": "WkAccount",
  "payment_entries": [
   {
    "invoice_id": 71,
    "payment_item_id": 438,
    "credit_issued": false,
    "invoice_no": "26",
    "invoice_org_amount": "168.0",
    "payment_org_amount": "100168.0",
    "invoice_org_currency": "$",
    "paid_amount": "100078.0",
    "amount": "100078.0"
   }
  ]
 }
}
```


## 19.19 Get Time/Expense Sheet

| Request Type | URL | Response |
|---|---|---|
| **Time** | | |
| GET | /wktime/edit.xml?user_id=9999&startday=2013-07-29 | Timesheet of a user for a week |

| | | |
|---|---|---|
| | /wktime/edit.json?user_id=9999&startday=2013-07-29 | |
| **Expense** | | |
| GET | /wkexpense/edit.xml?user_id=9999&startday=2013-07-29 | Expensesheet of a user for a week |
| | /wkexpense/edit.json?user_id=9999&startday=2013-07-29 | |

| **Parameters** | | |
|---|---|---|
| user_id | Required | User_id for whom time/expense sheet is to be retrieved |
| startday | Required | Specifies the start of the week |

**Sample XML Response (for Time)**


```
<?xml version="1.0" encoding="UTF-8"?>
<wk_time>
 <user name="Chandra Durairaj" id="5"/>
 <startday>2013-08-19</startday>
 <status>New</status>
 <total>6.0</total>
 <custom_fields type="array">
  <custom_field name="Purchase Order" id="12">
   <value/>
  </custom_field>
 </custom_fields>
 <time_entries type="array">
  <time_entry>
   <id>1506</id>
   <project name="Vitals Software" id="1"/>
   <issue id="1"/>
   <user name="Chandra Durairaj" id="5"/>
   <activity name="Design" id="10"/>
   <hours>4.0</hours>
   <comments/>
   <spent_on>2013-08-19</spent_on>
   <custom_fields type="array">
    <custom_field name="Difficulties" id="15">
     <value/>
    </custom_field>
   </custom_fields>
  </time_entry>
  <time_entry>
   <id>1507</id>
   <project name="Vitals Software" id="1"/>
   <issue id="1"/>
   <user name="Chandra Durairaj" id="5"/>
   <activity name="Design" id="10"/>
   <hours>2.0</hours>
   <comments/>
   <spent_on>2013-08-21</spent_on>
   <custom_fields type="array">
    <custom_field name="Difficulties" id="15">
     <value/>
    </custom_field>
   </custom_fields>
  </time_entry>
 </time_entries>
```

</wk_time>

**Sample JSON Response**


```
{"wk_time":
	{
		"user":{"id":5,"name":"Chandra Durairaj"},
		"startday":"2013-08-19",
		"status":"New",
		"total":6.0,
		"custom_fields":[
		{"id":12,"name":"Purchase Order","value":"233"}
		],
		"time_entries":[
		{
			"id":1506,
			"project":{"id":1,"name":"Vitals Software"},
			"issue":{"id":1},
			"user":{"id":5,"name":"Chandra Durairaj"},
			"activity":{"id":10,"name":"Design"},
			"hours":4.0,
			"spent_on":"2013-08-19",
			"custom_fields":[
			{"id":15,"name":"Difficulties"}
			]
		},
		{
			"id":1507,
			"project":{"id":1,"name":"Vitals Software"},
			"issue":{"id":1},
			"user":{"id":5,"name":"Chandra Durairaj"},
			"activity":{"id":10,"name":"Design"},
			"hours":2.0,
			"spent_on":"2013-08-21",
			"custom_fields":[
			{"id":15,"name":"Difficulties"}
			]
		} ]
	}
}
```


## 19.20 Get Clock in/out

| Request Type | URL | Response |
|---|---|---|
| Clkin/out | | |
| GET | /wkattendance/clockedit.xml?date=2020-06-26&user_id=1 | 200 OK. |
| | /wkattendance/clockedit.json?date=2020-06-26 &user_id=1 | |

| Parameters | | |
|---|---|---|
| user_id | Required | User_id for whom clock in/out is to be retrieved |

**Sample XML Response (for clock in/out)**

<clock_entries type="array">
<clk_entry>
<id>1217</id>
<clock_in>10:00</clock_in>
<clock_out>20:00</clock_out>
<hours>8.0</hours>
</clk_entry>
</clock_entries>

**Sample JSON Response (for clock in/out)**

```
{
  "clock_entries": [
    {
      "id": 1217,
      "clock_in": "10:00",
      "clock_out": "20:00",
      "hours": 8
    }
  ]
}

}
```

## 19.21 Get payroll

| Request Type | URL | Response |
|---|---|---|
| Payroll | | |
| GET | /wkpayroll/edit.xml?isPreview=false&salary_date=2019-12-01&tab=wkpayroll&user_id=1 | 200 OK. |
| | /wkpayroll/edit.json?isPreview=false&salary_date=2019-12-01&tab=wkpayroll&user_id=1 | |

| Parameters | | |
|---|---|---|
| user_id | Required | User_id for whom payroll is to be retrieved |

**Sample XML Response (for Payroll)**

<wk_payroll>
<basic>[["Basic", 40000, "$"]]</basic>
<allowance>
[["HRA", 18500, "$"], ["conveyance", 8000, "$"], ["Eduaction Allowance", 1200, "$"], ["Medical", 5000, "$"]]
</allowance>
<deduction>
[["ESI", 0, "$"], ["PF", 4800, "$"], ["Prof tax", 0, "$"], ["TDS", 4615, "$"], ["cab", 600, "$"]]
</deduction>
</wk_payroll>

**Sample JSON Response (for Payroll)**

```
   {
  "wk_payroll": {
```

```
  "basic": [
    [
      "Basic",
      40000,
      "$"
    ]
  ],
  "allowance": [
    [
      "HRA",
      18500,
      "$"
    ],
    [
      "conveyance",
      8000,
      "$"
    ],
    [
      "Eduaction Allowance",
      1200,
      "$"
    ],
    [
      "Medical",
      5000,
      "$"
    ]
  ],
  "deduction": [
    [
      "ESI",
      0,
      "$"
    ],
    [
      "PF",
      4800,
      "$"
    ],
    [
      "Prof tax",
      0,
      "$"
    ],
    [
      "TDS",
      4615,
      "$"
    ],
    [
      "cab",
      600,
      "$"
    ]
  ]
}
```

}

## 19.22 Get Leads

| Request Type | URL | Response |
|---|---|---|
| Lead | | |
| GET | /wklead/edit.xml?address_id=&lead_id=128 | 200 OK. |
| | /wklead/edit.json?address_id=&lead_id=128 | |

| Parameters | | |
|---|---|---|
| Lead_id | Required | Lead_id for whom leads is to be retrieved |
| address_id | Required | address_id for whom lead address is to be retrieved |

**Sample XML Response (for leads)**

```
<lead>
<lead_id>128</lead_id>
<account_id/>
<status>N</status>
<opportunity_amount/>
<lead_source_id>1</lead_source_id>
<referred_by/>
<contact_id>175</contact_id>
<first_name/>
<last_name>Martin</last_name>
<department/>
<location_id>5</location_id>
<description>default location</description>
<assigned_user_id>0</assigned_user_id>
</lead>
```

**Sample JSON Response (for Leads)**

```
{
 "lead": {
  "lead_id": 128,
  "account_id": null,
  "status": "N",
  "opportunity_amount": null,
  "lead_source_id": 1,
  "referred_by": "",
  "contact_id": 175,
  "first_name": "",
  "last_name": "Martin",
  "department": "",
  "location_id": 5,
  "description": "default location",
  "assigned_user_id": 0
 }
}
```

## 19.23 Get Account

| Request Type | URL | Response |
|---|---|---|
| Account | | |
| GET | /wkcrmaccount/edit.xml?account_id=78&address_id=18 | 200 OK. |
| | /wkcrmaccount/edit.json?account_id=78&address_id=18 | |

| Parameters | | |
|---|---|---|
| account_id | Required | account_id for whom leads account is to be retrieved |
| Address_id | Required | address_id for whom leads address is to be retrieved |

**Sample XML Response (for Accounts)**

<wk_account>
<account_id>78</account_id>
<account_name>Maxwell</account_name>
<account_billing>false</account_billing>
<account_category>26</account_category>
<location_id>5</location_id>
<description/>
<address id="18" address1="KK nagar" address2="" work_phone="" mobile="" email="" fax="" city="" country="" state="" pin="" website=""/>
</wk_account>

**Sample JSON Response (for Accounts)**

```
{
  "wk_account": {
    "account_id": 78,
    "account_name": "Maxwell",
    "account_billing": false,
    "account_category": "26",
    "location_id": 5,
    "description": "",
    "address": {
      "id": 18,
      "address1": "KK nagar",
      "address2": "",
      "work_phone": "",
      "mobile": "",
      "email": "",
      "fax": "",
      "city": "",
      "country": "",
      "state": "",
      "pin": null,
      "website": ""
    }
  }
}
```

### 19.24 Get Activities

| Request Type | URL | Response |
|---|---|---|
| Activity | | |
| GET | /wkcrmactivity/edit.xml? activity_id=44 | 200 OK. |
| | /wkcrmactivity/edit.json? activity_id=44 | |

| Parameters | | |
|---|---|---|
| activity_id | Required | activity_id for whom leads activity is to be retrieved |

**Sample XML Response (for Activities)**

```
&lt;wk_activity&gt;
&lt;crm_activity_id&gt;44&lt;/crm_activity_id&gt;
&lt;activity_type&gt;M&lt;/activity_type&gt;
&lt;activity_subject&gt;API Call&lt;/activity_subject&gt;
&lt;related_parent&gt;7&lt;/related_parent&gt;
&lt;related_to&gt;WkAccount&lt;/related_to&gt;
&lt;assigned_user_id/&gt;
&lt;status&gt;P&lt;/status&gt;
&lt;activity_direction&gt;I&lt;/activity_direction&gt;
&lt;activity_start_date&gt;2020-05-18T00:00:00+05:30&lt;/activity_start_date&gt;
&lt;activity_end_date&gt;2020-05-18T00:00:00+05:30&lt;/activity_end_date&gt;
&lt;activity_duration&gt;0&lt;/activity_duration&gt;
&lt;location/&gt;
&lt;activity_description/&gt;
&lt;/wk_activity&gt;
```

**Sample JSON Response (for Activities)**

```
{
  "wk_activity": {
    "crm_activity_id": 44,
    "activity_type": "M",
    "activity_subject": "API Call",
    "related_parent": 7,
    "related_to": "WkAccount",
    "assigned_user_id": null,
    "status": "P",
    "activity_direction": "I",
    "activity_start_date": "2020-05-18T00:00:00+05:30",
    "activity_end_date": "2020-05-18T00:00:00+05:30",
    "activity_duration": 0,
    "location": null,
    "activity_description": null
  }
}
```

### 19.25 Get Contacts

| Request Type | URL | Response |
|---|---|---|

| Contact | | |
|---|---|---|
| GET | /wkcrmcontact/edit.xml?contact_id=188 | 200 OK. |
| | /wkcrmcontact/edit.json?contact_id=188 | |

| **Parameters** | | |
|---|---|---|
| contact_id | Required | contact_id for whom leads contact is to be retrieved |

**Sample XML Response (for Contact)**

```
&lt;wk_contact&gt;
&lt;contact_id&gt;188&lt;/contact_id&gt;
&lt;first_name/&gt;
&lt;last_name&gt;Test2&lt;/last_name&gt;
&lt;address_id/&gt;
&lt;contact_title/&gt;
&lt;department/&gt;
&lt;assigned_user_id&gt;0&lt;/assigned_user_id&gt;
&lt;salutation/&gt;
&lt;description/&gt;
&lt;related_to&gt;WkCrmContact&lt;/related_to&gt;
&lt;related_parent&gt;176&lt;/related_parent&gt;
&lt;contact_type&gt;C&lt;/contact_type&gt;
&lt;location_id&gt;5&lt;/location_id&gt;
&lt;relationship_id&gt;0&lt;/relationship_id&gt;
&lt;/wk_contact&gt;
```

**Sample JSON Response (for Contact)**

```
{
  "wk_contact": {
    "contact_id": 188,
    "first_name": "",
    "last_name": "Test2",
    "address_id": null,
    "contact_title": "",
    "department": "",
    "assigned_user_id": 0,
    "salutation": "",
    "description": "",
    "related_to": "WkCrmContact",
    "related_parent": 176,
    "contact_type": "C",
    "location_id": 5,
    "relationship_id": 0
  }
}
```

### 19.26 Get invoice

| **Request Type** | **URL** | **Response** |
|---|---|---|

| Invoice | | |
|---|---|---|
| GET | /wkinvoice/edit.xml?invoice_id=1 | 200 OK. |
| | /wkcrmcontact/edit.json?contact_id=188 | |

| **Parameters** | | |
|---|---|---|
| invoice_id | Required | invoice_id for whom invoice is to be retrieved |

### Sample XML Response (for Invoice)


```xml
<invoice>
<invoice_id>1</invoice_id>
<invoice_number>1</invoice_number>
<inv_date>2019-03-08</inv_date>
<inv_start_date>2019-01-01</inv_start_date>
<inv_end_date>2019-03-31</inv_end_date>
<field_status>o</field_status>
<modifier_id>1</modifier_id>
<modifier_name>Redmine Admin</modifier_name>
<parent_name> Bhargav </parent_name>
<gl_transaction_id>969</gl_transaction_id>
<parent_type>WkCrmContact</parent_type>
<parent_id>1</parent_id>
<invoice_type>I</invoice_type>
<invoice_num_key>1</invoice_num_key>
<invoiceItemEntries type="array">
<invoice_item>
<item_id>1</item_id>
<name>sprint2</name>
<project_id>1</project_id>
<rate>200.0</rate>
<amount>600.0</amount>
<quantity>3.0</quantity>
<hd_item_type>i</hd_item_type>
<currency>$</currency>
<original_currency>$</original_currency>
<original_amount>600.0</original_amount>
</invoice_item>
</invoiceItemEntries>
</invoice>
```


### Sample JSON Response (for Invoice)


```json
{
  "invoice": {
    "invoice_id": 1,
    "invoice_number": "1",
    "inv_date": "2019-03-08",
    "inv_start_date": "2019-01-01",
    "inv_end_date": "2019-03-31",
    "field_status": "o",
    "modifier_id": 1,
    "modifier_name": "Redmine Admin",
    "parent_name": " Bhargav ",
    "gl_transaction_id": 969,
```

"parent_type": "WkCrmContact",
"parent_id": 1,
"invoice_type": "I",
"invoice_num_key": 1,
"invoiceItemEntries": [
{
"item_id": 1,
"name": "sprint2",
"project_id": 1,
"rate": "200.0",
"amount": "600.0",
"quantity": 3,
"hd_item_type": "i",
"currency": "$",
"original_currency": "$",
"original_amount": "600.0"
},

### 19.27 Get Payments

| Request Type | URL | Response |
|---|---|---|
| Payments | | |
| GET | /wkpayment/edit.xml?payment_id=158 | 200 OK. |
| | /wkpayment/edit.json?payment_id=158 | |

| Parameters | | |
|---|---|---|
| Payment_id | Required | payment_id for whom payment is to be retrieved |

**Sample XML Response (for Payments)**

<wk_payment>
<payment_id>158</payment_id>
<payment_type_id>36</payment_type_id>
<payment_date>2020-05-20</payment_date>
<reference_number/>
<description>AccName:Addisson InvNo:#26 PaymentAmt:$100078.00</description>
<related_parent>1</related_parent>
<related_to>WkAccount</related_to>
<payment_entries type="array">
<entry>
<invoice_id>71</invoice_id>
<payment_item_id>438</payment_item_id>
<credit_issued>false</credit_issued>
<invoice_no>26</invoice_no>
<invoice_org_amount>168.0</invoice_org_amount>
<payment_org_amount>100168.0</payment_org_amount>
<invoice_org_currency>$</invoice_org_currency>
<paid_amount>100078.0</paid_amount>
<amount>100078.0</amount>
</entry>
</payment_entries>

</wk_payment>

**Sample JSON Response (for Payment)**


```
{
  "wk_payment": {
    "payment_id": 158,
    "payment_type_id": 36,
    "payment_date": "2020-05-20",
    "reference_number": "",
    "description": "AccName:Addisson InvNo:#26 PaymentAmt:$100078.00",
    "related_parent": 1,
    "related_to": "WkAccount",
    "payment_entries": [
      {
        "invoice_id": 71,
        "payment_item_id": 438,
        "credit_issued": false,
        "invoice_no": "26",
        "invoice_org_amount": "168.0",
        "payment_org_amount": "100168.0",
        "invoice_org_currency": "$",
        "paid_amount": "100078.0",
        "amount": "100078.0"
      }
    ]
  }
}
```


### 19.28 Delete Time/Expense Sheet

The project member with edit permission will be allowed to delete time and expense sheet using delete API.

| Request Type | URL | Response |
|---|---|---|
| **Time** | | |
| DELETE | /wktime/destroy.xml?user_id=9999&startday=2013-08-12 | 200 OK.<br>Deletes a timesheet |
| | /wktime/destroy.json?user_id=9999&startday=2013-08-12 | |
| **Expense** | | |
| DELETE | /wkexpense/destroy.xml?user_id=9999&startday=2013-08-12 | 200 OK.<br>Deletes an expensesheet |
| | /wkexpense/destroy.json?user_id=9999&startday=2013-08-12 | |

| **Parameters** | | |
|---|---|---|
| user_id | Required | user_id for whom time/expense sheet is to be deleted |
| startday | Required | Specifies the start of the week |

### 19.29 Delete Time/Expense Entries

The project member with edit permission will be allowed to delete time and expense entries using deleteEntries API.

| Request Type | URL | Response |
|---|---|---|
| **Time** | | |
| DELETE | /wktime/deleteEntries.xml | 200 OK.<br>Deletes a time entries |
| | /wktime/deleteEntries.json | |
| **Expense** | | |
| DELETE | /wkexpense/deleteEntries.xml | 200 OK.<br>Deletes an expense entries |
| | /wkexpense/deleteEntries.json | |

**Sample XML data to be posted (for Time)**


```xml
<?xml version="1.0" encoding="utf-8"?>
<time_entries type="array">
 <time_entry>
  <id>1535</id>
  <project id="1" name="Vitals Software"/>
  <issue id="1"/>
  <user id="3" name="Suganya Thulasiraman"/>
  <activity id="10" name="Design"/>
  <hours>4.0</hours>
  <comments/>
  <spent_on>2013-12-23</spent_on>
  <custom_fields type="array">
   <custom_field id="15" name="Difficulties">
    <value/>
   </custom_field>
  </custom_fields>
 </time_entry>
 <time_entry>
  <id>1537</id>
  <project id="1" name="Vitals Software"/>
  <issue id="1"/>
  <user id="3" name="Suganya Thulasiraman"/>
  <activity id="10" name="Design"/>
  <hours>4.0</hours>
  <comments/>
  <spent_on>2013-12-25</spent_on>
  <custom_fields type="array">
   <custom_field id="15" name="Difficulties">
    <value/>
   </custom_field>
  </custom_fields>
 </time_entry>
</time_entries>
```


## 20 Code Hooks

T&E supports code hooks so it can be extended seamlessly. The following are the hooks

| # | Name | Description | Context |
|---|---|---|---|
| 1 | view_te_filter | Used to specify filters for the time/expense sheet list page. | |
| 2 | view_member_dropdown | Used to fill member dropdown in New time/expense sheet page. | project_id |
| 3 | view_wktime_menu | Used to show or hide "Time & Attendance" menu. | |
| | view_additional_lead_info | Used to add additional fields on the lead page | leadObj |
| | view_product_item | Used to add additional fields on the product item page | |
| | view_asset_inventory | Used to add additional fields on the inventory item page | |
| | view_accordion_section | Used to add a new section in the accordion view | Entity, curObj |
| | add_report_filters | Used to add filters in reports tab | |
| | additional_contact_info | Used to add additional fields on the contact item page | Address, contactobj |
| | additional_contact_type | Used to add more contact types | |
| | modify_product_log_type | Used to add additional log types | params |
| | get_invoice_issue_period | Used to set the invoice period | Issue, attributes |
| | payment_additional_where_query | Used to add additional contact type in payment query | |
| | additional_spent_type | Used to add additional spent types | |
| | additional_product_type | Used to add additional product types | |
| | external_enum_type | Used to add additional enum types | |
| 4 | controller_project_member | Used to fill member dropdown for the selected project on project dropdown change. | project_id |
| 5 | controller_group_member | Used to fill member dropdown for the selected group on group dropdown change | group_id |
| 6 | controller_get_member | Used to fill member dropdown on list page on page load. | filter_type |
| 7 | controller_check_permission | Used to check permission to view time/expense sheet. | |
| 8 | controller_check_approvable | Used to check permission to approve time/expense sheet. | |
| 9 | controller_edit_timelog_permission | Used to check permission for edit/delete time/expense log. | |
| 10 | controller_set_view_projects | Used to fill project dropdown in list page. | |
| 11 | controller_set_manage_projects | Used to fill project dropdown in new time/expense sheet page. | |
| 12 | controller_check_editable | Used to check whether time/expense sheet is editable. | editable, user |
| 13 | controller_check_locked | Used to check whether time/expense sheet is locked. | Startdate |
| 14 | controller_get_manager | Used to get the manager of a member | user, approval |
| 15 | controller_get_permissible_projs | Used to get the allowed project for the user | user |
| 16 | controller_after_save_invoice | Call back after save invoice | Attributes |
| 17 | controller_convert_contact | Used to set contact type on convert | Params, |

| | | | |
|---|---|---|---|
| | | | leadObj, contactObj |
| 18 | controller_updated_contact | Used to throw some error message on lead conversion | Params, leadObj, contactObj |

# 21 Customization

For any Customization / Support, please contact us, our consulting team will be happy to help you

Adhi Software Pvt Ltd
http://www.adhisoftware.co.in
info@adhisoftware.co.in
+1 732 661 8294
+91 44 27470401
+91 44 27470402

Here are the Customizations we have done for our clients:

1. Monthly Calendar – Puny Human
2. Supervisor Approvals – Fotonation

# 22 Troubleshooting

1. *The issues associated with each project do not show up in the drop down list*
   This might be the issue visibility configuration issue. Check under Administration ➔ Roles and Permissions➔ Edit user role ➔ Issue Visibility.
   (For further detail refer section 5.1.1)

2. *Is it possible to change the default currency of expenses*
   First one from the currency dropdown is the default. To have the currency of your choice to be the default in currency dropdown, add the following lines within number in the language file (for e.g., config/locale/cs.yml )

```
currency:
         format:
         format:"%u%n"
         unit: "CZK"
```

3. *There is no submit button*
   The submit button will be visible only when “Approval System” is enabled.

4. *How to approve the manager time-sheet?*
5. *User Billing Rate*
6. *User Billing Currency*
7. 
   If the setting “Approve own time/expense sheet” is enabled, then the manager can approve his own timesheet.
8. *Why is the plugin not maintaining comments separately for each time entry?*
   If we have set “Enter comment in row”, it will allow only one comment per row. The timesheet will allow only one combination of project, issue and activity on multiple rows. To allow for separate comments per time entry, do not use the setting “Enter comment in row”.

9. *The pdf export is not working. It is showing Internal error*

The rmagick gem is required for displaying the logo in the pdf. If the rmagick gem is not installed, this error is encountered. Other alternative is to remove the logo from the pdf. To remove the logo, go to the plugin configuration page and empty the report logo field.

10. *Some of the time entries are disabled on edit for project manager*
The project manager (with "Edit time logs" permission) can add/edit a time & expense sheet for their project members.

User Billing Rate
User Billing Currency