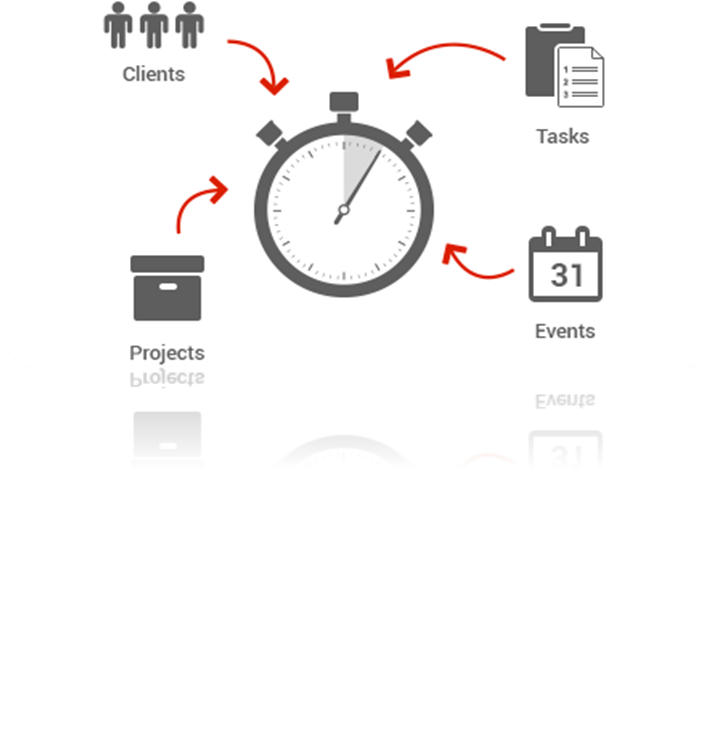
****

**Computer Science Project**

**Comp - 4910**

**Fall, 2017**

**Time Tracking and Task Management Project**

****

**Submitted to: Kevin’O Neil**

**Submitted by:**

**Karanbir Singh**

**Mohammad Alsharida**

**Agyapal Singh**

**Ramandeep Kaur**

**Table of Contents**

[**Project Outline**](#_9dobu11iix0m)3

[**Design and Testing**](#_m5pqxeijl9j3)4

[**Administrative Guide**](#_21ldloofpshu)9

[**User guide**](#_gfxsr31uvuhu)16

[**Comments**](#_2g28b7ajql0a)20

[**Appendix: Weekly Reports**](#_r6n6a5k54pn)22

# 

# **Project Outline**

The goal of this project is to create a new Time Tracking and Task Management web application that is optimized for use in a diverse website development, SEO and graphic design business. All user functions of the application must be designed to require as little interaction with the user as possible while simultaneously collecting valuable information so that management and decision makers are able to ensure smooth operation and generate useful reports. The web application will be focused on carefully condensed information displays allowing for quick and easy access to view and manipulate important information.

Project Requirements:

* The ability for employee and admin to create and manage tasks.
* The ability to switch between tasks without losing time tracking data.
* The ability for employee and admin to generate reports.
* Friendly interface interaction.

Project Platform and Used Languages:

This project is made up using MySQL and PHP server. The web app interface uses Javascript, AJAX, jQuery,HTML/CSS for faster updates of information and to allow the server to push updates to users all the time. The platform runs efficiently. All interfaces are accessible on all major browsers for both computers and mobile devices.

Languages:

* PHP
* Javascript
* jQuery
* CSS
* HTML
* MySql

External Libraries used:

* Bootstrap
* SweetAlerts
* DateAndTimePicker

Browsers Supported:

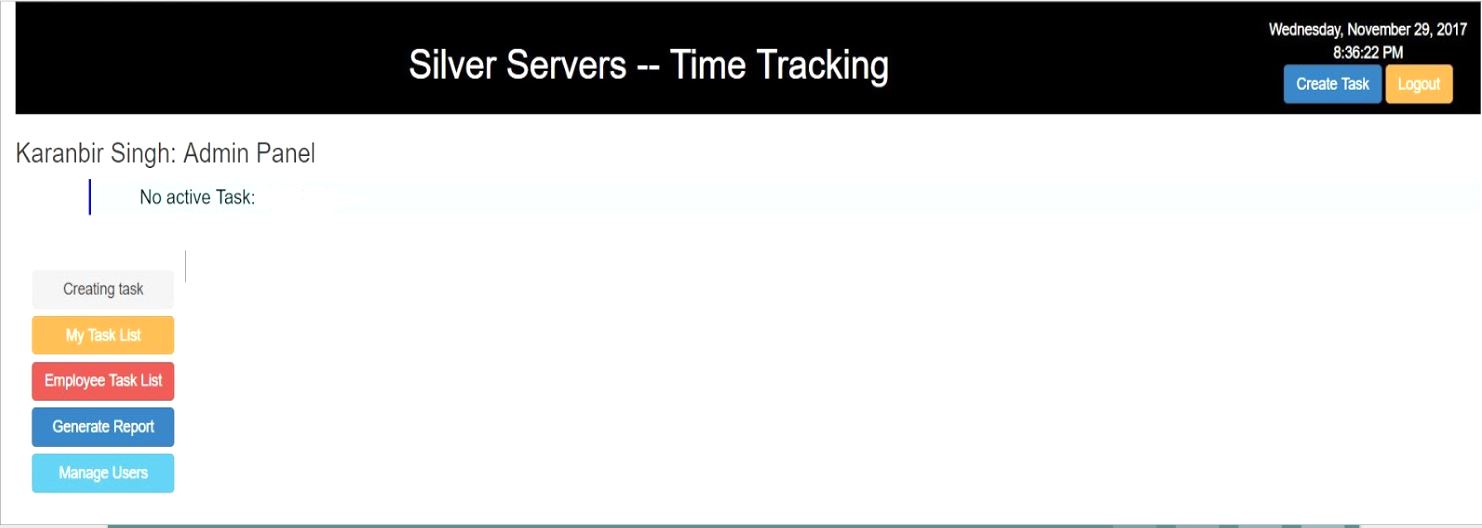
* Google Chrome
* Mozilla Firefox
* Microsoft Edge

# 

# **Design and Testing**

Create-Task Phase**:**

The goal here is to give the employee and Admin an easy and friendly way to create task at a fast pace in busy environment. This button interacts with front and back-end, where the data is entered and the values are stored in the database immediately. This privilege of creating new tasks is available on the top right side of the home screen of every user as shown in the figure 1.0.



**Figure 1.0**

Working

The working begins as soon as the user hits create-task button. It consists of three different phases which are explained as under:-

1. Filling Dynamic Fields
2. Data Extraction
3. Finishing Settings

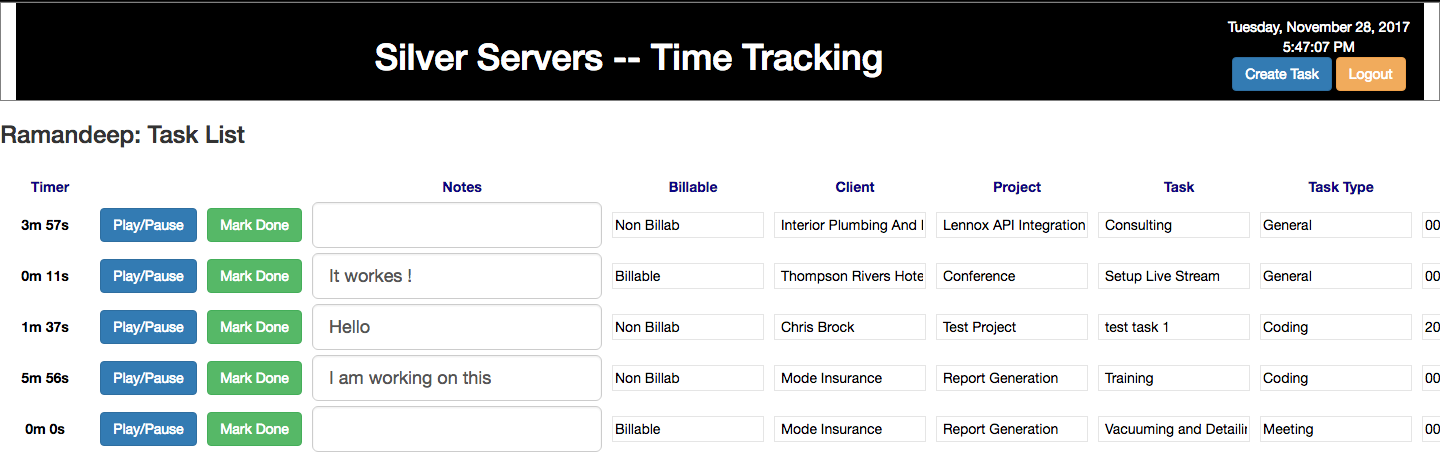
|  |  |
| --- | --- |
| **PHASE 1 - FILLING DYNAMIC FIELDS** | |
| **Dynamic Fields** | **Description** |
| **Task Name** | A user have to input the name of the task so that it can be tracked and worked on easily.  Note: Task names cannot be same, if there is a task name that already exists in the database then it will pop up a sweet alert. |
| **Client Name** | A user also have to input the name of the client they will be working for. New clients names will be stored in the database and previous clients will be shown to the user on the screen for whom user already worked for. |
| **Project Name** | Project name is an important dynamic field. There are two types of project fields depending on the client:   * Known Projects - User worked on the same project and for the same client are known projects. * Other known Projects - User working for a client but the project is new.   Note: These other known projects can be known to some clients because the user might have worked for a client on that project. |
| **PHASE 2 - DATA EXTRACTION** | In this phase the data is extracted from the database. The data extracted from the database are the dynamic fields which are filled in phase 1 by the user for creating a task. These fields are shown to the user on the left side of the screen in the last phase. |
| **PHASE 3 - FINISHING SETTINGS** | **DESCRIPTION** |
| **Select Type** | The user would have to select what type of work he is doing depending on the needs of the client. It can be Coding, General, Meeting, Q&A or more. In more the user have the privilege to add any other or new type of work done for the client. |
| **Select Worker** | In this, User would have to select the name of the worker who is assigned to the task. |
| **Select Billing** | User can select the type of billing which has to be done for the client at the end of the project. The available fields are Billable, Non-Billable, Discounted, Internal, Overtime or other. In Other the user again have the privilege to add any other type of billing which can be done for the client. |
| **Done & Start Timer** | This setting lets the user save all the dynamic fields and data selected to create the task in the database. After this setting is clicked the task will be started on the spot and timer will start for the task. |
| **Done & Save** | This setting lets the user save all the dynamic fields and data selected to create the task in the database. This task will be saved but not started. The user can come back to the task anytime later on to start working on the task and the timer will be started upon request. |
| **Show More** | In this setting, the user can add notes for the task or any instructions for the task including the due date and date assigned. |

Employee phase:

In this phase the main goal is to show the list of tasks that been created by the employee or the admin.

There are only three functions the employee could do in this phase:

* Play and pause.
* Insert note for each task anytime.
* Mark done and insert note within 30 seconds.



**Figure 2.0**

Features

* Each task could be updated multiple times.
* The note which is belong to each task could be updated by insert a text and click out of the label.
* All updates or changes in loges that would be recorded in the database with time.
* For more efficiency and accuracy the employee can only run ONE task at a time.
* The task that currently running will be swapped to the top of tasks list.
* While running certain task and click play button for another task, automatically the second task will be swapped to the top and pause the other task.

Instructions

1. After the task is created the task will be listed as it shown in **Figure 2.0**
2. By click play/pause button once the timer start counts beginning with seconds then add minutes and also hours.
3. By clicking play/pause while it is running the timer will stop and the logs will be updated in the database.
4. Notes label could be updated multiple times and to use it properly, user should insert short text note after pause the task every time, then click out of the label to save the note in the logs.
5. When the user click Mark Done button pop up window will show up and the user *must* insert a note within 30 seconds and click submit, then the task will be removed from the list.

Admin Phase

Admin phase consists of day-to-day activities which are performed by the Admin and employees of SilverServers while working for the client on any project. Admin profile has the authority and permissions to access clients, projects and tasks. Adding to this, the admin has all the privileges to keep track of employees work done for the clients. It shows the admin a full task list displaying the completed and incomplete projects.

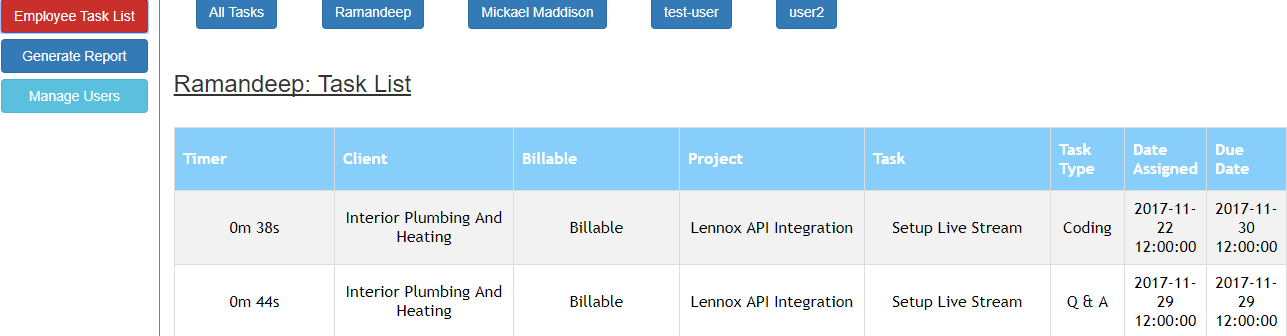
There are set of different options which are available to the admin as following:

* Creating Task
* My Task List
* Employee Task List
* Generate Report
* Manage Report

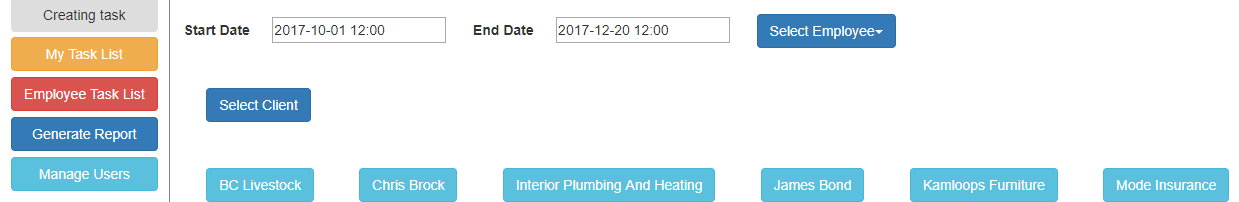
**Creating Task-** When Admin would like to create a task, there will be button on the top right stating “Create Task” which would be same as described above in Create-Task Phase and its working.

**My Task List**- This option gives the admin to view all his tasks. These tasks are created by the admin only and it helps to keep a track of projects date started, due date, total time spend and time remaining to complete the task.

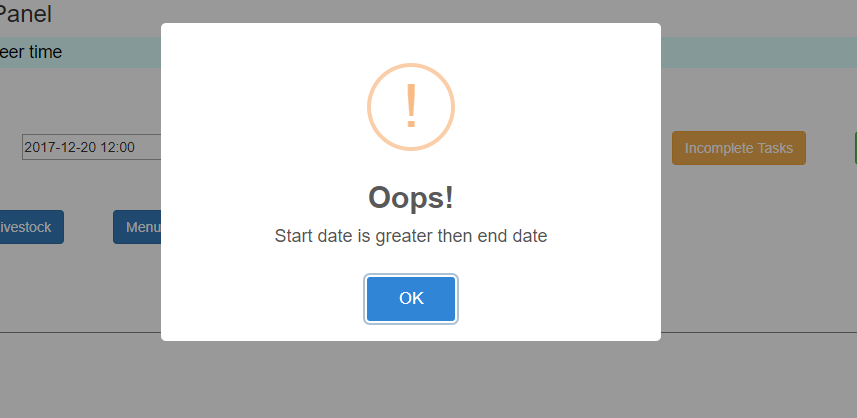
**Employee task list-** A function where if the admin wants to see what are all the tasks currently employees working on, and it has a special feature that is there are different button on top by the employee name. You can also short list all the work by clicking on the name button and after that it’ll only show you the tasks currently working under that employee name.



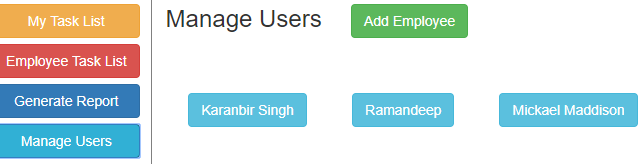
**Generate Report-** Generate report is a another feature for the admin page where if the admin wants to see completed or the uncompleted work for a specific client, he has to click on the client’s name button on top of page and it’ll show you all current work. Moreover , if you want to see projects for any client you can click on client+ project button, to see the current projects running. More specifically client + project + tasks To see all tasks running under the client's name.



Another main feature of the admin page is that you cannot put invalid dates for ex: if you put wrong dates, it will show you a pop-up message saying that your “Start date is greater then End date”.



Manage Users:Manage users is an authentication feature that provides administrators with the ability to identify and control the state of users logged into the network and also has the ability to add more users in the project or create a new employee.



# 

# 

# 

# 

# **Administrative Guide**

Time Tracking and Task Management is a web based software. Time Tracking software will enable employees to keep track of the hours that they worked on the given project, task or other deliverable. This software is made for company’s internal use only, There could be two different type of users, employee and a admin.

**Employee could do the following things:**

* Can see all uncompleted task in the list
* Start timer of the task from the list.
* Add notes to the task
* Mark the task as done
* Create a task
* Also, employee is able to see the all the information about the task

**Where as Admin has more privileges, which are as follows:**

* Admin always get notified, which employee is working on which task.
* Admin can view all the uncompleted tasks from all the employees and can also filter the list by selecting the specific employee
* Admin can generate three different types of reports, which is generate report by client, generate report by selecting client and a specific project, and generate report by selecting a client, selecting client’s specific project task.
* Admin can manage all the users that are currently using the software, which means admin can change their username, password, display name, mark the selected user as admin, and force logout from the software.
* Also, admin as all the privileges that the other user(employee) has.

**More about the Software:**

Time tracking and task management software is very user friendly. The user could easily and quickly understand the interface even with minimal knowledge of the software. It is also a *responsive website*, this approach to website design makes the web pages render on variety of devices, window and the screen sizes.

One of the most important feature of this software is that it is *secured*, To make the software secure and to avoid all the possible attacks on the software, parameterized queries has been used. Parameterized query is a means of pre-compiling a SQL statements. ***It is used to prevent SQL attacks.***

Other important feature of the software is it does not allow the same user to be logged in multiple browsers or machines. Which means only one active screen is possible. If the same user tries to log in on the different window/browser or even on the different machine, the software won’t allow the user to login, until the user logs out from there current active window. Database keeps the track of the user, how gets logged in.

Using Time Tracking and task management software data can’t be lost. Which means database gets anonymously updated whenever the user makes some changes, and saves everything without notifying to the user. For example when a user starts timer of the task, watch the timer clicks, the software saves data in the database.

Another Important feature of the software is, Whatever the user do while they are logged in will be stored in the database. Which means, what time the employee started or stopped working on a task, which time the user added the note on a task also when the user marked the task as done.

**How it Works:**

This software follows MVC architecture. Using this pattern it divides the software into three different interconnected parts This is accomplished by separating internal representation of information from the ways information is presented to, accepted from, and the user.

View

Frontend

Controller

Model

Database

The *Model* is the central component of this application. It explains the software’s behaviour. Independent of the user, It directly manages the logic, data and the concept or the rules of the application. The Model stores data that is retrieved according to the commands from the controller and displays in the view. Model also interacts with the database which means it retrieve and send data to the database.

The *View* is the output representation of the information, such as tables and lists etc. view generates new output to the user based on request to model and the response from the model.

The *Controller* accepts inputs and converts it to commands for the view ot the controller. When the controller sends the command to the model to update the state. It can also send commands to its associated view to change the view’s presentation of the model.

**Software structure**

**USERS**

**Individual Task List**

**Admin**

Create Task

Employee task list

Generate Report

Manage Users

Employee

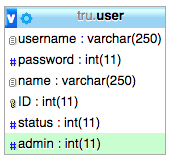
**There are two types of users and things that could be done by the user:**

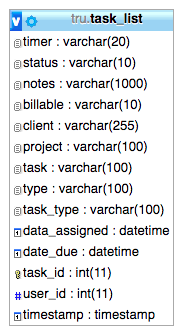
* **User: Employee**
  + View Task list
  + Create new Task
* **User: Admin**
  + View Task list
  + Create new Task
  + Employees Task List
  + Generate report
  + Manage users

Database structure:

**Database uses 8 different tables to store data, Which are as follows:**

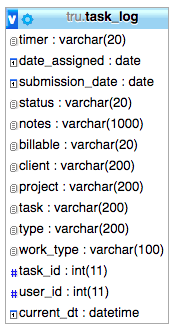
1. User
2. Task\_list
3. Task\_log
4. Clients
5. Projects
6. Tasks
7. Task\_type
8. Billing\_info

**User:** It stores all information about the user. Its stores the following

* Username - With varchar datatype which can store upto 250 characters
* Passwords - It has a int datatype which can store upto 11 digit long integer
* Name -- A varchar data type with 250 character limit.
* ID - It is primary key of user table, has a int data type, stores the unique id of every user, auto increment is used to keep each user unique
* Admin -- It has int data type and stores upto integer length of 11, this field just has the value of 0 and 1. 1 means the user is a admin, and 0 means the user is a employee.
* Status-- It has int data type and stores upto integer length of 11, this field just stores the value of 0 and 1 where 1 means the user is a logged in, and 0 means the user is a logged off.

**Task\_list:** it stores all the information about the task, which are described as follows:

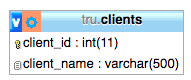
* Timer: With varchar datatype which can store upto 20 characters. It is used to store time of all the individual tasks in the database that the user is working or worked on before.
* Status: Its varchar datatype which can store upto 10 characters. It can only has 3 valued in this field, play, pause and done. Play means the task is currently active. And some user is currently working on it. Pause demonstrate that user is not currently working on the task
* Notes: With varchar datatype which can store upto 1000 characters. This field is used keep the last added comment on the given task to the user
* Billable: With varchar datatype which can store upto 10 characters. This field values are taken from other database table (billing\_info), after the user creates the task in frontend
* Client: With varchar datatype which can store upto 255 characters. This field values are taken from other database table (clients), after the user creates the task in frontend.
* Project: With varchar datatype which can store upto 100 characters. This field values are taken from other database table (projects), after the user creates the task in frontend
* Task: With varchar datatype which can store upto 100 characters. This field values are taken from other database table (tasks), after the user creates the task in frontend
* Type: With varchar datatype which can store upto 100 characters. This field values are taken from other database table (tasks), after the user creates the task in frontend
* Data\_assigned: With datetime datatype. This field values are taken from, after the user creates the task in frontend
* Date\_due: With datetime datatype. This field values are taken from, after the user creates the task in frontend
* Task\_id: It is primary key of this table, has a int data type, stores the unique id of every task, auto increment is used to keep each task unique
* User\_id: is has a int data type, this field describes which user is working on which task, this field is the primary key of user database. Whenever the user create the task, each task is assigned to one or multiple users.
* Timestamp: With datetime datatype. Date and time are stored in this field whenever the user plays or pause the task, it keeps the track of the last action made on the task.



**task \_log:** this database table is very identical to task\_list table. This table is used to store all the activities that the user make while they are logged in, for instance when the user played the specific task, when the user paused, when the user adds note the note to a task, when the user mark the task as done. This database collects all the task\_list current field values and stores it the database along with the real time. Fields are described as follows:

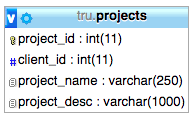
* Timer: With varchar datatype which can store upto 20 characters. It is used to current time of all the individual tasks in the database that the user is working.
* Status: Its a varchar datatype which can store upto 20 characters. Stores the current status of the task
* Notes: With varchar datatype which can store upto 1000 characters. This field is used keep the last added comment of the task
* Billable: With varchar datatype which can store upto 20 characters.
* Client: With varchar datatype which can store upto 200 characters.
* Project: With varchar datatype which can store upto 200 characters.
* Task: With varchar datatype which can store upto 100 characters. This field values are taken from other database table (tasks), after the user creates the task in frontend
* Type: With varchar datatype which can store upto 100 characters.
* Date\_assigned: With datetime datatype.
* submission\_date: With datetime datatype.
* Task\_id: has a int data type, which tells the following logs belongs to which task.
* User\_id: is has a int data type, this field describes which user is making changes on task.

**Clients:** this database table is used to store the list of the companies clients that they are working for. It is described as follows:



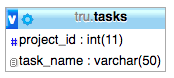
* Client\_id: primary key of the table and stores int datatype, to keep each client with unique id, auto increment is used.
* Client\_name: with varchar data type and can store upto 500 characters. This field will just store the name of the client

**Projects:** This table is used to store all the list of the projects that the company is working or worked before for the client.the fields are describe as follows:

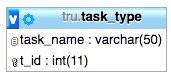


* Project\_id: it stores int data type with length upto 11, this is a primary key to keep each project with a unique ID auto increment is used
* Client\_id: this field stores int value and describes which project belongs to which project.
* Project\_name: With varchar datatype which can store upto 250 characters. This field is used keep store names of the project
* Project\_desc: With varchar datatype which can store upto 1000 characters. This field is used store description of the project.

**Tasks:** This table is used to store all the list of the tasks that the company is working or worked before for the client.the fields are describe as follows:

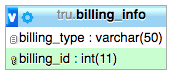


* Project\_id: This field stores int data type and is used to describe, to which project the task belongs to.
* Task\_name: Stores varchar data type, with character limit of 50 and is used to store names of the task.

**Task\_type:** This table is used to store all the list of different types of task that the company is working or worked before for the client.the fields are describe as follows:

* Task\_name: stores varchar data type and upto 50 characters. And stores name of the task type
* I\_id: it stores int data type, it’s the primary key of this table and to keep the id’s unique auto increment is used.

**billing \_info:** This table is used to store the billing info for the company is working or worked before for the client.the fields are describe as follows:



* billing \_type: it stores varchar data type, upto 50 characters. This field stores the billing type info.
* billing\_id : This a primary key of this table and stores int datatype. To keep the id’s unique auto increment is used.

# 

# **User guide**

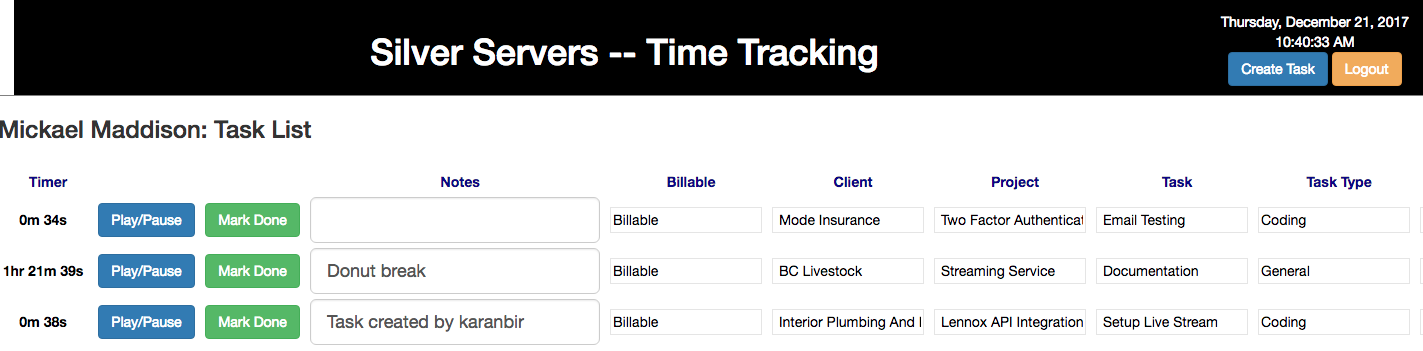
**Step 1: Login**

* Enter valid username and password
* Press the *Sign In* button to log in

**Step 2:** Based on the user login , there are 2 possibilities, user maybe admin or an Employee.

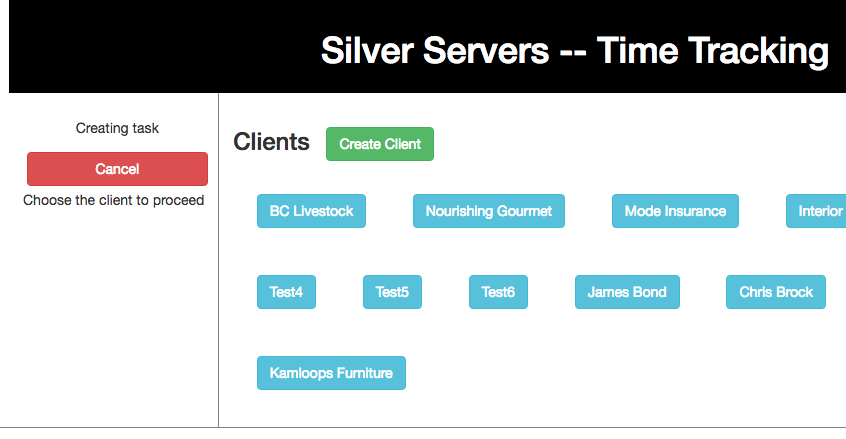
Following are the steps are for employee user:

**Step 1: Employee also for admin**

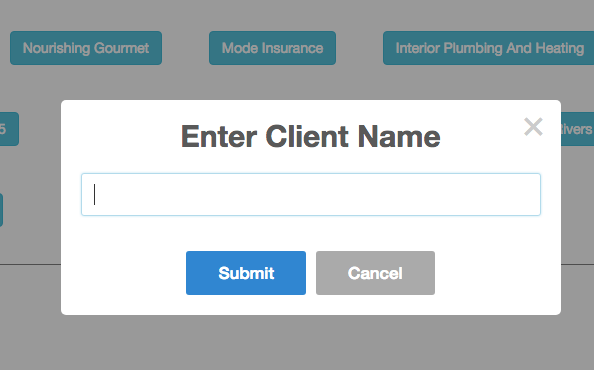
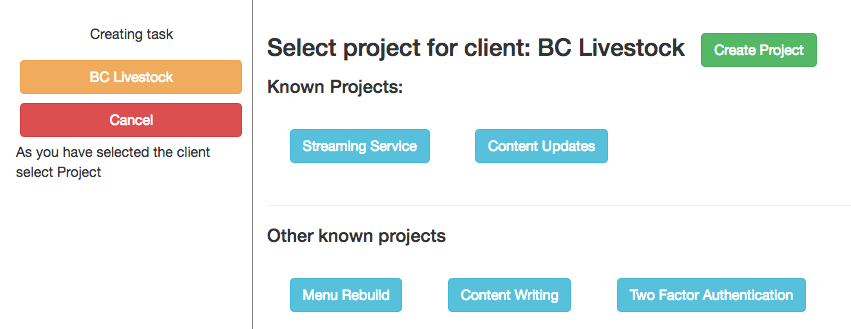
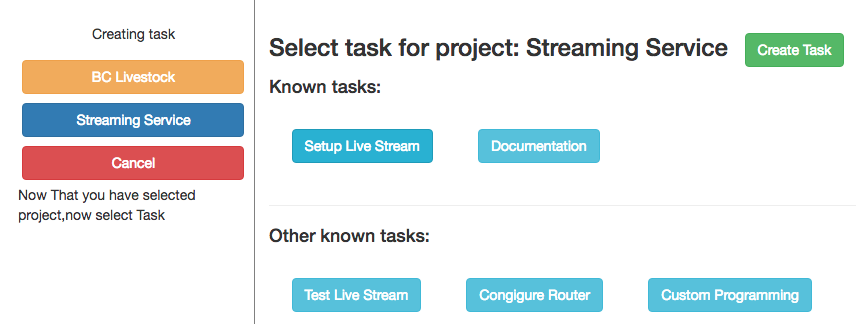
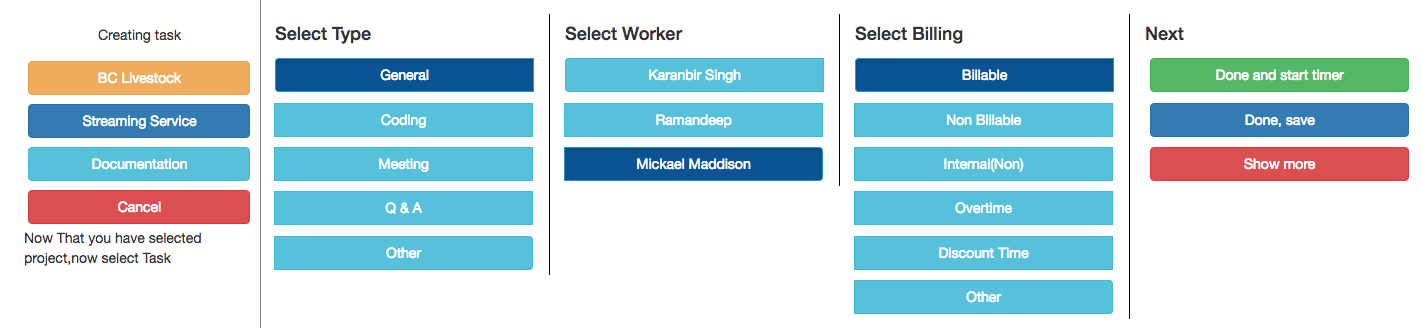
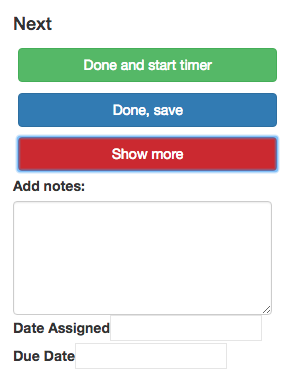


* After login task list will be displayed for the logged in employee
* Employee can start working on a task by simply clicking play/pause button
* Only one task can be active at one time.
* Timer will start clicking on the screen after task has been started.
* User can pause the active by just clicking on play/pause button.
* Notes can be added any time to any task, notes to submitted simply when you click outside the textbox.
* Mark Done, whenever the user finish the task, they can mark the task as done.
  + There will be pop up window, where you have to insert final notes within 30 seconds and submit the task, unable to do so, will not submit task
* Other information is also provided to the user, but employee won't be able to edit it.
* Create new task, by clicking the button on top

**Step 2: Create new task → Employee and admin**

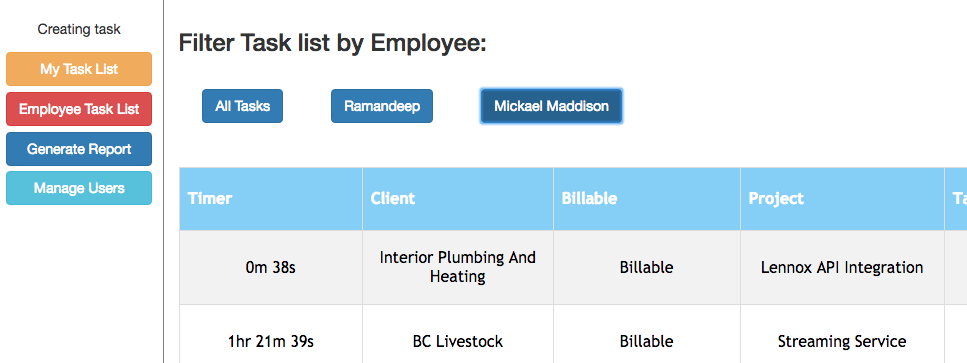
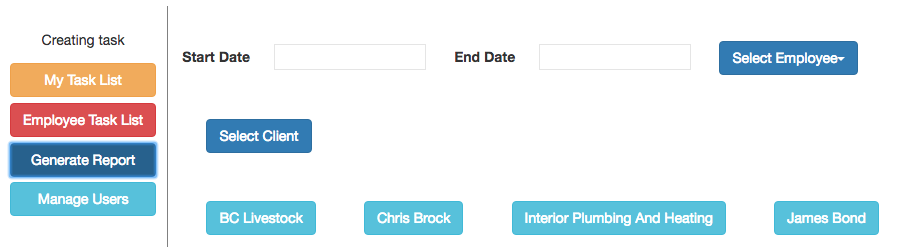
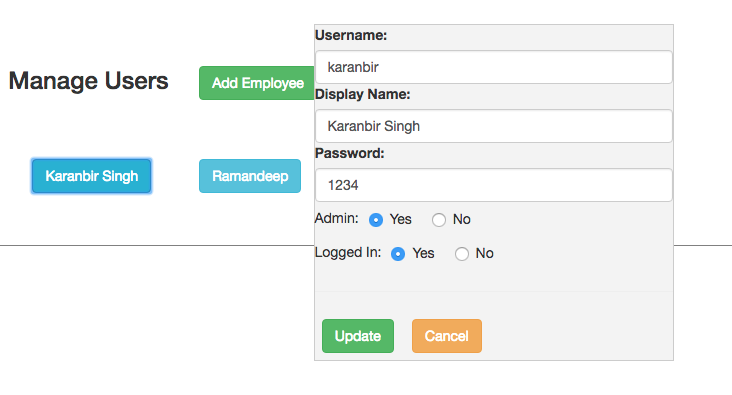


There is a screen partition, on left side you will see all your selected values, and the left your will get display next step for creating the task User can cancel the task any time , and can get back to the there task list

1. Step 1 is to select the client or to create a new client
   1. To create new client just click on the create client
   2. Click submit or cancel,
   3. If there is a existing client, then a message will be displayed to the user.
   4. If it is a new client then, after you submit, it will be displayed in the client list for you selection.
2. Step 2 is to select the project or create new project
   1. Known projects → projects that user have already worked for the client in the past. Or when the user creates a new project, then also project it will appear in the known task
   2. Other known tasks → Other list of tasks other then known tasks.
   3. Creating a new project is similar to creating a new client
3. Step 3 is to select the task or create a new task
   1. This step is similar to selection of a project.
4. Step 4 is to select task type, assigned a worker to task, and select billing
   1. Default value for select type → General
   2. Default value for select worker → logged in user , task any also be assigned to multiple users
   3. Default for billing → Billable
5. Step 5 is to add notes , date assigned and due date. There are optional fields. Click on show now to view these options
   1. Add Notes: Starting note to the user who is going to work on the task
   2. Date Assigned/ Due Date: Used to store the start date and end of the task. Format is date and time, user has to select the date and the time.
6. Step 6 user selects one of the two options, → Save or save and start working on the task
   1. Saving the task will help the user to save the task.After saving the user will will be redirected to the step 1 of the create task.
   2. Save and start is used to save the task and start working on the same task right after the button is clicked

**Step 3: Admin**

**Admin can has some additional privileges then the regular user. Which are explained as follows:**

1. View Employee Task list:
   1. By Default when you click this button all uncompleted task list will be displayed. But there is a list of all the users and by choosing the specific user you will be able to filter task list of the employees.
   2. Admin won't be able to see his/her task list there, because they have their own task list under *my task*
2. **Generate Report:** 
   1. Choose valid date and select client to proceed.
   2. After selecting client → then either you generate report or select project to narrow down report. Or you can choose specific task of of the selected project to get very precise report
   3. Select employee to generate a report for the specific client.
   4. Report can be generated based on completed or uncompleted tasks
3. **Manage Users:**
   1. You can create a new user 
   2. Edit and updating the the existing user
   3. Make user admin or make the user as employee
   4. Force logout for logged in employee

# 

# **Comments**

This project has helped me a lot to learn new things and has much prepared me for future job. It was a great experience working with silverservers, they were really helpful and encouraging. Thanks kevin for all your efforts, it's not easy to do such thing for students. Thanks again.

*-- Karanbir Singh*

This project was great opportunity to gain and improve my knowledge and give me the real job atmosphere stage. There was some challenges and difficulties that we faced during the first few weeks however, we realize that this is the real situation which we are moving to for the next station of life. Thanks kevin for all your helps and thanks Silver Server for providing this chance.

*-- Mohammed Alsharida*

This opportunity was a very good grab for us. We were always interested in this proposal over others projects, the time we saw it. We kind of knew “We Got This”. Luckily we got Silver Servers as our client and it was a bliss working under them. We had no communication problems and they were a huge support and very patience. Kevin as an Instructor was a great help. He guided us and helped us get to our goal with minimum problems. Compared to my other Internships, It was a huge lot of experience working with my team. They rock.

-- Agyapal Singh

Working on a project with such a good company was a positive and a great experience, as it gave me the strength to deal with difficulties.   
It helped me learnt a lot of new and different things which definitely going to help me in my future jobs. Special thanks to Kevin as because of him, me and my team-mates were able to work with the co-ordination.

*-- Ramandeep Kaur*

# 

# **Appendix:** Weekly Reports

Week 1

**Summary:**

We had our first meeting with the company(SilverServers), and discussed in detail about the project. They are going to set up the resources and also they are going to provide us documentation from where we can get started.

Task Completion from Last Week

* Our team had our first meeting with client.
* Discussed in depth about the project, gathered requirements and all the necessary tools that will be needed for the project completion.

Task for This Week

* We will have group meetings next week, will discuss more about the project after we get documentation from silverservers and task will be assigned to the team members accordingly

Week 2

**Summary:**

Got the start-up documentation from Siverservers. Got resources (server and access to database) from them. We had couple of team meeting to discuss the project, and Individual task was assigned to team members

Task Completion from Last Week

* Our team had our first meeting with client.
* Got documentation from client
* Resources are all provided
* Task assigned to each team member

**Task for This Week**

* Create login page and display contents such as username from database. Display current date and time, and create timer that will be used in task
* Create notes textbox, and its working(update database in note log database, which consists of current time on timer, task due date, current data and time, task type, work type)
* Create task list, and its working (display current working task on top, next most recent task should display next on the list, tasks that are not being started should display on the bottom of the list).

Week 3

**Summary:**

Worked on the structure of the project, created database, frontend page, controller and the main pain page. Had a meeting the client and discussed on few confusions.

**Task Completion from Last Week**

* Database created
* Login page created. With interaction with the database
* Created the main display page where the user will see all the listed tasks.
* Timer working.
* Displaying the list from the database
* Had a meeting with the client discussed our doubt on few tasks.

**Task for This Week**

* working on note\_log database. Working : after the user enter text in notes and clicks outside the box the note\_box database should be updated with data
* working on task\_log database: whenever the user clicks on play/pause button. The timer should respond and data should be passed to the task\_log database
* Create overall database, timer working, Display real time and current date. Connection with the database with php. Create over all structure of the project. Pages such as: controller, module and login page, main page and there interaction with other. Display list of task from the current user.
* Agyapal: Managing task lisk. The order of the list to be shown to the user. And Marking task as done

**Known issues / things blocking progress**

AJAX -- our team has no prior experience in AJAX.

Week 4

**Summary:**

Working on functionality of project. Had a meeting the client and discussed on next phase of the project.

**Task Completion from Last Week**

* Timer working.
* Functionality of notes textbox. Data being retrieve in backend
* Added functionality on Mark as done.
* Had a meeting with the client discussed our doubt on few tasks.

**Task for This Week**

* Store values in note\_log database that are being retrieved from frontend.
* working on play/pause button. (just one task to be played at time from the list). Retrieve values from frontend and store it in the task\_log database
* Managing order of task list.

Week 5

**Summary:**

Added backend functionality to the project. Had a meeting the client and asked us to make changes to the databases plus added functionality to the current phase of the project. We were also given documentation for the next phase of the project and talked about it.

**Task Completion from Last Week**

* Connected database connectivity for note logs and task logs tables.
* connectivity with database for Mark as done.
* Had a meeting with the client discussed about current and next phase.

**Task for This Week**

* updating database.
* Working on task timer. before functionality was storing the values in string, like “10m 5s”. now we asked to store the values only in seconds in database, and do the math on frontend. But still need to display the timer in the same format.
* Managing order of task list. there is a reported bug from silver servers. Whenever the user logs in. Tasks are displayed from only for the user at index 0.

Week 6

**Summary:**

Added full functionality for display task phase. Had a meeting the client and discussed about the progress for the project so far. Also they wanted was create a secure connection with the database.

**Task Completion from Last Week**

* Secure connection with database
* Only one active session at a time.
* Frontend and backend connectivity of the next phase (“create task”)
* Had a meeting with the client about the progress.

**Task for This Week**

* Creating frontend for the create task phase
* Creating secure connection with the database and one active session
* Connectivity with backend for create task. And helping Mohammed in creating frontend.

Week 7

**Summary:**

Updated all the database queries to make then secure. Completed the frontend and the backend for the create task page. Had a meeting with a client and were asked to add the functionality to the create phase task and update timer to store values only in seconds in the database

**Task Completion from Last Week**

* Secure connection
* Completion of (“create task”) phase
* Project review with the client and Kevin.

**Task for This Week**

* Styling frontend of display task list page
* Additional functionality in create task such as: create new client, create new project, create new task, cancel task
* Storing the timer only in seconds in the database and display timer according to the user format.
* Submit create task data to the database and display/start task according to the user input.

Week 8

**Summary:**

Additional functionality in create task such as: create new client, create new project,

create new task, cancel task. Had a Meeting with the client, and discussed about the

progress of the project.

**Task Completion from Last Week**

* Added functionality in create task
* Programs checks from the previous data to check if the value entered still exists in the database
* Frontend for the create task phase
* Updated timer

**Task for This Week**

* Mohammed Alsharida: allow only one active page per user.
* working on the frontend of the display task page.
* Submit create task data to the database and display/start task according

to the user input.

Week 9

**Summary:**

Completed tasks → Styling of display list, Restrict user to only one active session, Submit task buttons(create and display task on the user list). Had a Meeting with the client, Submit buttons wasn’t working due to last minute updates. We have emailed them with the updates, and waiting to hear from client. We are waiting for there response at the moment, and few tasks

**Task Completion from Last Week**

* Styling of display list
* Restrict user to only one active session
* Save and done button (when creating task)
* Save and start timer button (Restrict user to only one active session)

**Task for This Week**

* Defaults for checkboxes and also radio buttons.
* Testing current software.

Week 10

**Summary:**

We had a meeting with a client on Friday; we had a discussion on the current working of the software. They have asked has to create next phase (admin page) to the software, but we are not asked to complete the phase. Also there is one bug found in the software, will be fixing that this week

**Task Completion from Last Week**

* Create task → Defaults for checkboxes and radio buttons.
* Restrict user to only one active session
* Save and done button (when creating task)
* Save and start timer button (Restrict user to only one active session)

**Task for This Week**

styling of the admin page.

* On user login → check user in database if the user is admin then display admin page, also working on the bug that was found in the meeting
* Display current tasks that user are working on in the admin page. Along with the task description.
* Update table when some user start working on different task or new user start there task.

**Additional Information**

We are also going to create a functionality → generate report in the admin page.

Week 11

**Summary:**

Created next phase (admin page) to the software, we were asked to add additional features when admin generate report also add feature to display Employee task list.

**Task Completion from Last Week**

* Created Admin page
* Admin → view all employee active task list, and user can filter individual employee task by selecting a employee
* Admin → Generate report, admin can generate 3 type of different report from specified date range Selecting Client, Selecting Client à Project, Selecting Client à Project à Task
* Admin → There will always be a display message on admin page, which will display which employee is working on which task, if there is no task currently being played then, then it displays message accordingly

**Task for This Week**

* Add feature to Generate report, where the admin can select to display incomplete or only completed tasks.
* In generate report, filter report by employee. In employee task listàwhen all task are displayed, there should be different tables of each employee tasks