

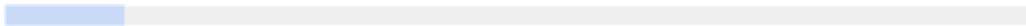
Milestones for Time Tracker Project

Turn Mockups into a Web Page

In this document, you'll find a set of milestones to follow that will help you deliver your project.

Milestone 1: Setting up the development environment

10% progress



Get your project off to a good start by installing a solid, rigorous development environment VueJs as well as setting up the files you will need for your development work. We also need the Its version of Vuetify and a router in our project.

 **Once this step is completed you will have:**

- A src folder in which we'll do all the code.
- A public folder with our main index.html file and logo of our website.

 **Recommendations:**

- Start by installing VSCode
- Install VueJs in your computer or laptop whatever you have by using one of these commands:

`npm install -g @vue/cli` **OR** `yarn global add @vue/cli`

- Add vuetify by using this command:

```
vue add vuetify
```

- Add router by using command:

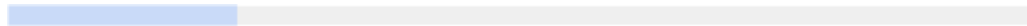
```
vue add router
```

Resources:

- For install [Vue](#)
- For install [Vuetify](#) in our project
- For install [Router](#) in our project

Milestone 2: Setting up folders and files structure of our project

20% progress



First of all, we will make all the folders or files in the views folder which we have to show on our header or in the sidebar. And we also add our files or folders to our components folder related to our views component. We also make an interface folder which we use for making interfaces of our different files.



Once this step is completed you will have:

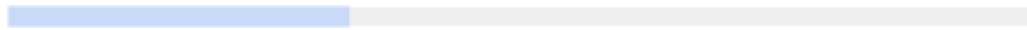
- An auth folder for user login, logout and signup.
- A client folder to show our clients.
- A team folder to show our team-mates.
- An invoices folder.
- A setting folder for user settings.

Resources:

- To see folders and file structure in [VS Code](#)

Milestone 3: Setting up the Login, Signup, Sign Out page and Forget password etc.

30% progress



You can now start embedding the Login, Signup, Sign Out and Forget password page in our auth folder.



Once this step is completed you will have:

- A Login.vue file
- A Signup.vue file
- A Logout.vue file
- A ResetPassword.vue file

Resources:

- Use [Vuetify](#)
- Use Vuetify [Form](#)

Issue to be aware of:

- Make responsive pages for mobile and other devices.
- Pages should not overflow from the screen.

Milestone 4: Setting up the Header and Sidebar

40% progress



You can now start designing the Header and Sidebar of the Project.

 **Once this step is completed you will have:**

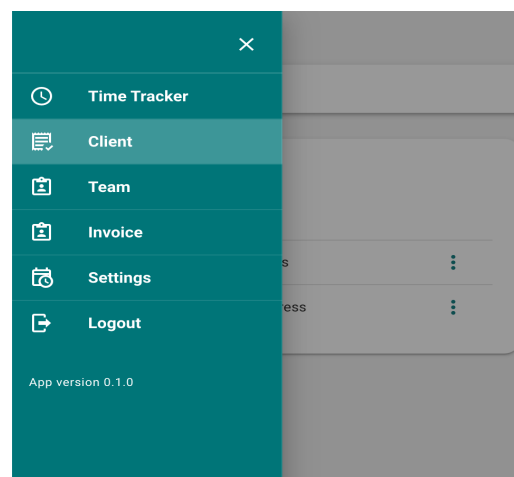
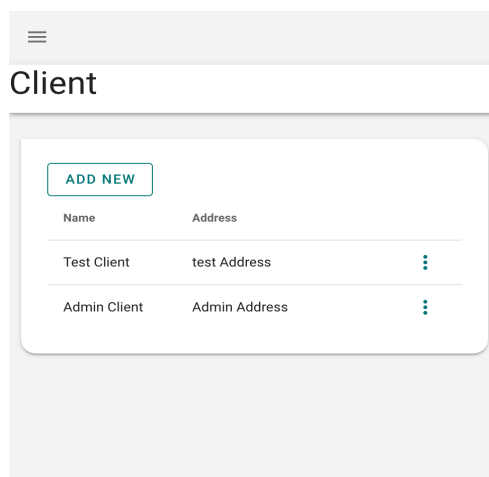
- A nav folder in Components folder
- A config.ts file.
- A SiteNav.vue file in which all of the Sidebar components will be run.
- A Header.vue file.

Recommendations:

- Use Vuetify [app-bar](#) Component
- Use Vuetify [side-bar](#) component

Issue to be aware of:

- App-bar should be responsive for mobile and other related devices.
- It should be like this:



Milestone 5: Embedding the Client Component in Header and Side-bar.

50% progress



During this stage you will have to embed a data-table in our Client Component with CRUD options. Also add a [Dialog](#) to add more clients in clients tables.



Once this step is completed you will have:

- A client folder in our views folder.
- A Client.vue file in our client folder.
- A dialog in our Client.vue file.

Resources:

- Use Vuetify [data-table](#) to display all of the clients.
- Use the Vuetify [dialog](#) component.

Issue to be aware of:

- Make responsive data-table for mobile and other devices.
- Pages should not overflow from the screen.

Milestone 6: Embedding the Invoices Component in Header and Side-bar.

60% progress



During this stage you will have to embed a data-table in our Invoices Component with CRUD options. Also add a button for downloading the pdf of the table.



Once this step is completed you will have:

- A invoices folder in our views folder.
- A Invoices.vue file in our invoices folder.

Resources:

- Use Vuetify [data-table](#) to display all of the invoices data.

Issue to be aware of:

- Make responsive data-table for mobile and other devices.
- Pages should not overflow from the screen.
- Button should work properly.

Milestone 7: Embedding the Team Component in Header and Side-bar.

70% progress



During this stage you will have to embed a data-table in our Team Component with CRUD options. Also add two dialogs. One for adding a user and one for adding his role. i.e: admin etc.

Once this step is completed you will have:

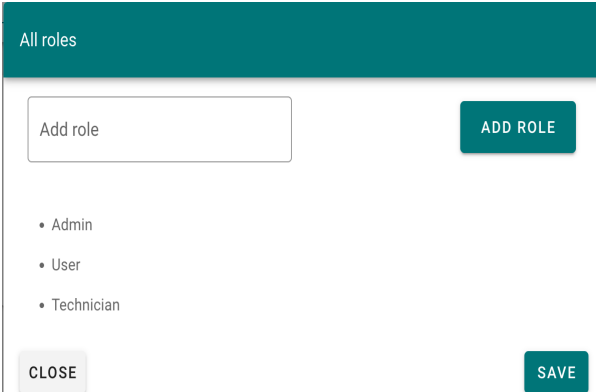
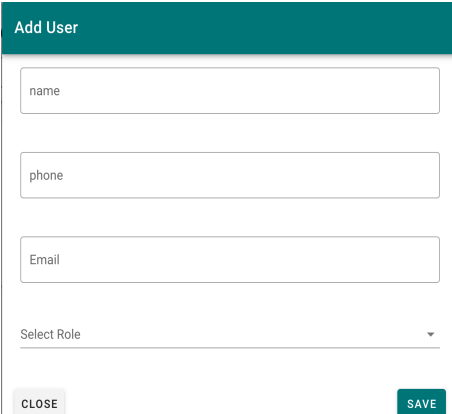
- A team folder in our views folder.
- A Team.vue file in our Team folder.
- A dialog for adding more users.
- A dialog for adding the role of users.

Resources:

- Use Vuetify [data-table](#) to display all of the invoices data.
- Use the Vuetify [dialog](#) component.

Issue to be aware of:

- Make responsive data-table for mobile and other devices.
- Pages should not overflow from the screen.
- Only the admin can add more users and assign their roles.



Milestone 8: Embedding the Settings Component in Header and Side-bar.

80% progress



During this stage you will have to embed a simple form in our Settings Component.



Once this step is completed you will have:

- A Profile folder in our views folder.
- A Settings.vue file in our Profile folder.

Resources:

- Use the Vuetify [form](#).

Issue to be aware of:

- Make responsive data-table for mobile and other devices.
- Input bars should not overflow from the screen.
- Data should be displayed on the screen of the user who is logged in.

Milestone 9: Embedding the Time-Tracker Component in Header and Side-bar.

100% progress



During this stage you will have to embed a data-table with CRUD options. Also need some input bars for selecting project, user, starting and ending time etc and a button to save the new data.



Once this step is completed you will have:

- A time folder in our views folder.
- A TimeTracker.vue file in our time folder.

Resources:

- Use the Vuetify [data-table](#).
- Use the Vuetify [input-bars](#).
- Use the Vuetify [select-bar](#).

Issue to be aware of:

- Make responsive data-table for mobile and other devices.
- Input bars and tables should not overflow from the screen.