**Technical Documentation**

**Installation**

**Prerequisites**

In order to run to-do list app test you need nodejs to install the dependencies. Hence, install the npm at this https://nodejs.org/en/download/

**Downloading**

The zip of the app is located in Github, You can download the apps by navigating to the repository of the app using this link and click “Clone or Download”. When the files are downloaded unzip the file.

Next, go to your terminal and go to the directory in which the files are being unzipped. Then run this command “**npm install”.** After the installation succeed, open index.html and you can start to use the app.

**How to Use**

**Adding a task**

1. Click to input field that has a placeholder of “What needs to be done?”
2. Input the task that you need to do
3. Press enter to add the task to the list

**Marking a task as completed**

1. There are 2 types of filter “All” and “Active”, make sure you set the types correctly to your reference
2. Click the checkbox to the left of the task that you want to be marked as completed

**Deleting a task**

1. Hover the mouse on top of the task that you want to delete, you can see that there is a cross [x] icon shows up on the right of the task
2. Click the cross [x] icon to delete the task (Be careful as there is no confirmation to delete the task)

**Editing a task**

1. Double click on the task that you want to complete
2. Input it with the edited task

**UML Diagram**



**System Documentation**

The apps is using several components and dependencies to make it work, below are the listed components and dependencies

**Dependencies:**

1. **todomvc-common:** Some common CSS and JS
2. **tomomvc-app-css:** Css Styling
3. **jasmine-core:** Testing framework

**Components:**

1. **Model:** A file to handle data for the application such as Create, Read, Update, and Delete operations also known as (CRUD).
2. **View:** A file that present our processed data that is visually can be seen to the user
3. **Controller:** Handles the request coming from user interaction which in this case coming from view files
4. **Store:** A file that handle the interaction which data is stored
5. **Template:** A file that handling the template of the data that is being shown
6. **Helpers:** A file that is containing of several function that is used repeatedly