# To-DoListProject

Angular is a platform that makes it easy to build applications with the web. Angular combines declarative templates, dependency injection, end to end tooling, and integrated best practices to solve development challenges. Angular empowers developers to build applications that live on the web, mobile, or the desktop

This is a MEAN stack based application which is used to schedule the Meetings. There are two separate parts of the application:

1. ToDoListFrontend
2. ToDoListBackend

### Prerequisites

1. To start with this, install node and npm
2. Install git
3. Use npm to install Angular CLI . Exceute this command

>npm install -g @angular/cli

### Installing/ Running locally

1. Create a folder named as Meeting Planner at any local drive
2. change directory to

>cd /ToDoList/

1. Fetch the source code from my github library

>git init

>git remote add origin [https://github.com/thuteshubham/ToDoList.git](https://github.com/thuteshubham/MeetingPlanner.git)

>git pull origin master

Or

Refer a bellow link

<https://github.com/thuteshubham/ToDoList>

1. Install all the modules required to run the given application with following command in Client(ToDoListFrontend)

>npm install

1. Run the application by using following command in Client(ToDoListFrontend)

>ng serve

But in my package.json I have added in Client(frontEnd)

"scripts": {

"ng": "ng",

"start": "ng serve",

"build": "ng build",

"test": "ng test",

"lint": "ng lint",

"e2e": "ng e2e"

},

So you use can use > npm start

6. Navigate to <http://localhost:4200/> via browser . You will see the application is running on frontEnd(ToDoListFrontend)

1. By considering Mongo dB and mongoose installation run the below command

>mongod

8. Install all the modules required to run the given application with following command in Server (ToDoListBackend)

>npm install

9. Run the application by using following command

>npm start

Since in my package.json I have added below code

{

"name": "todolistproject-backend",

  "version": "1.0.0",

  "description": "This is backend for ToDo list",

  "main": "index.js",

  "scripts": {

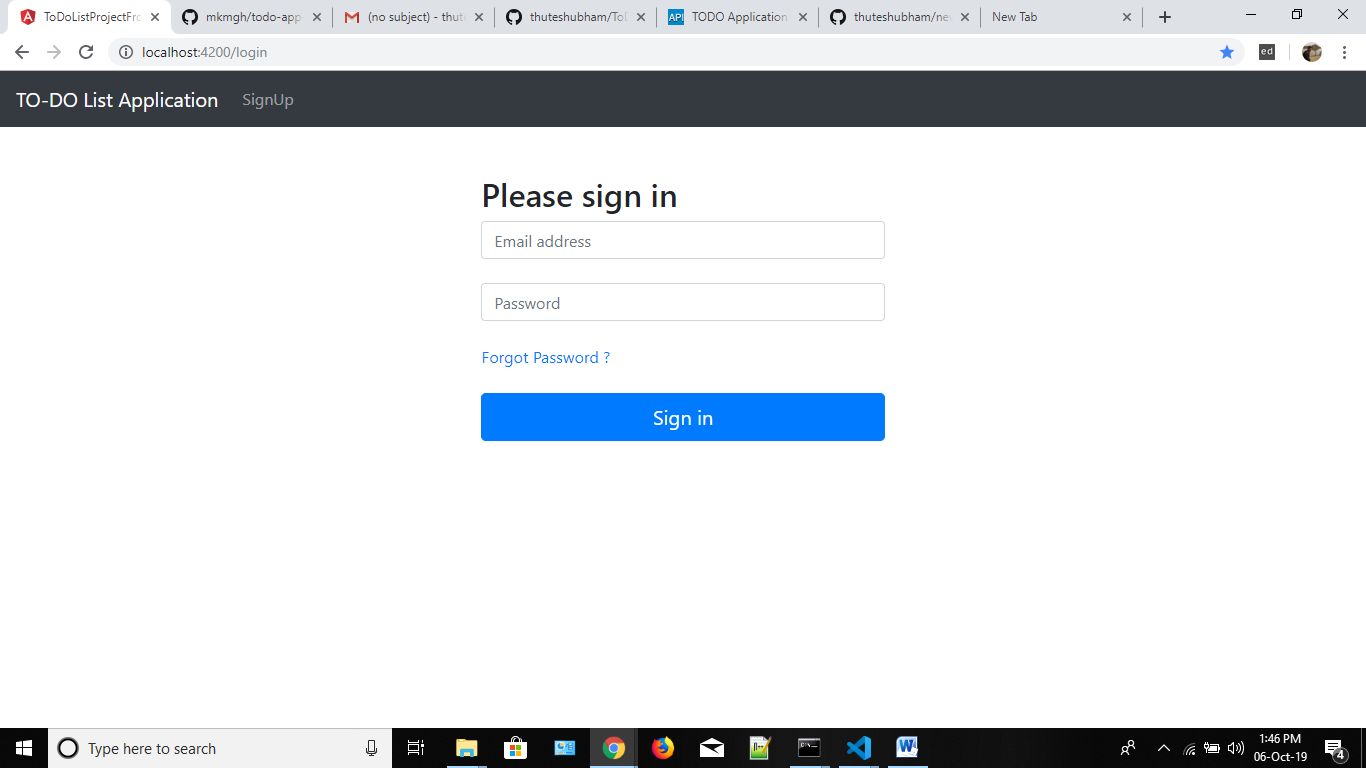
    "test": "echo \"Error: no test specified\" && exit 1",

    "start": "nodemon index.js"

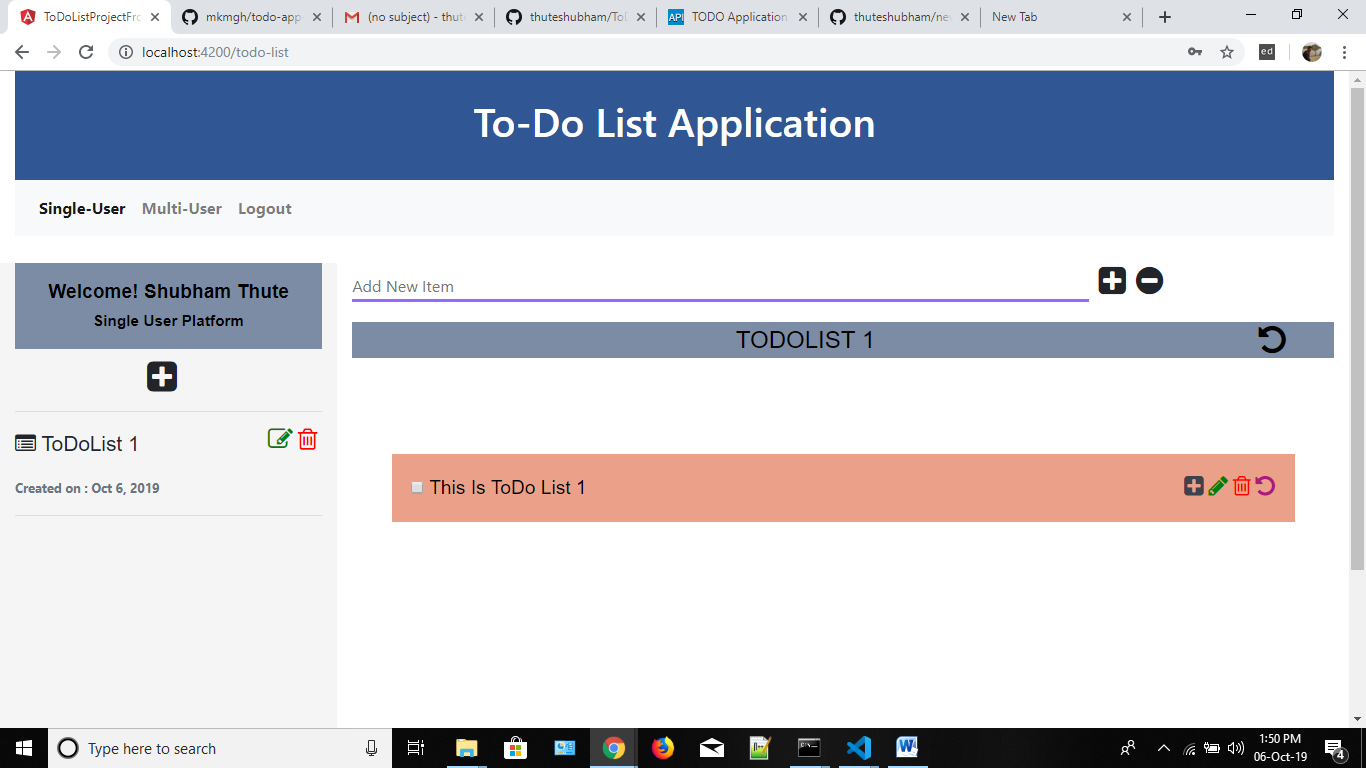
}

**Screenshots for your reference:**

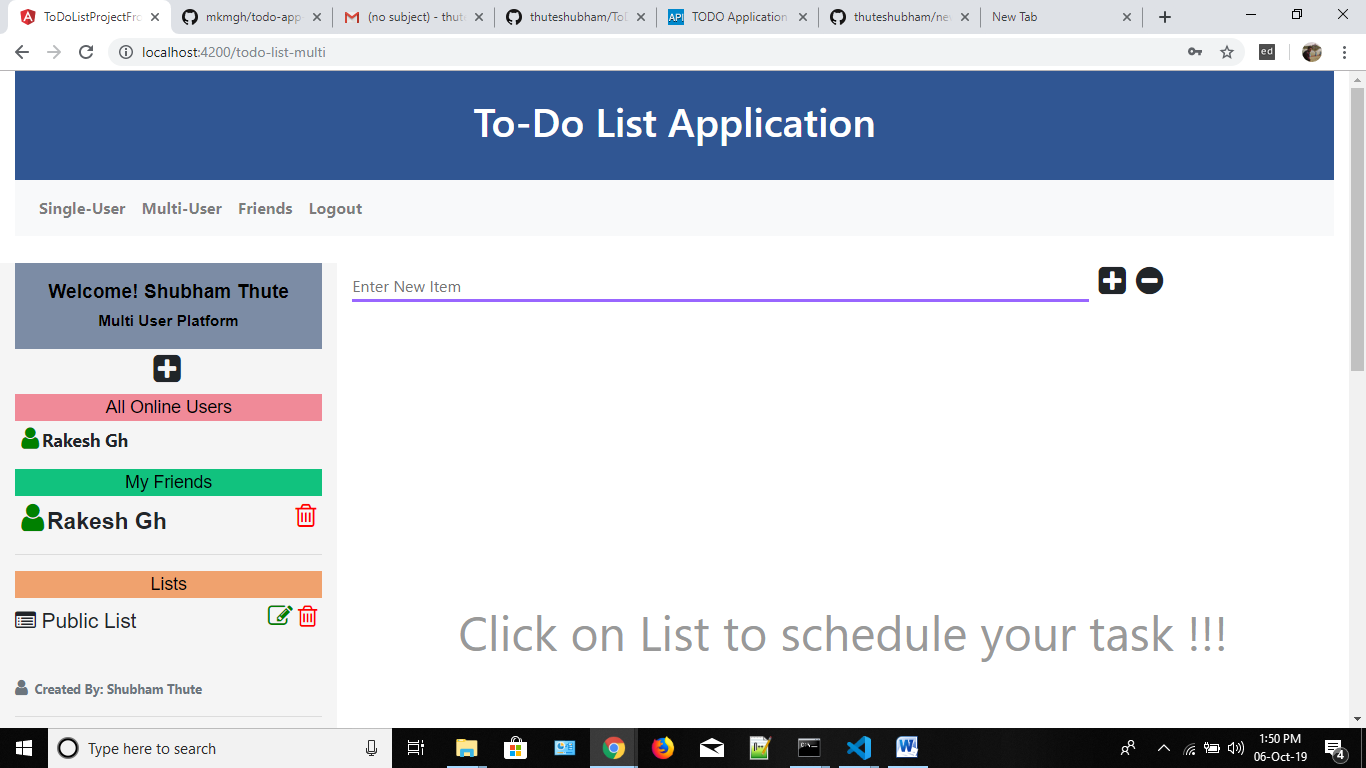
**Login Page:**



**Single user Dashboard:**

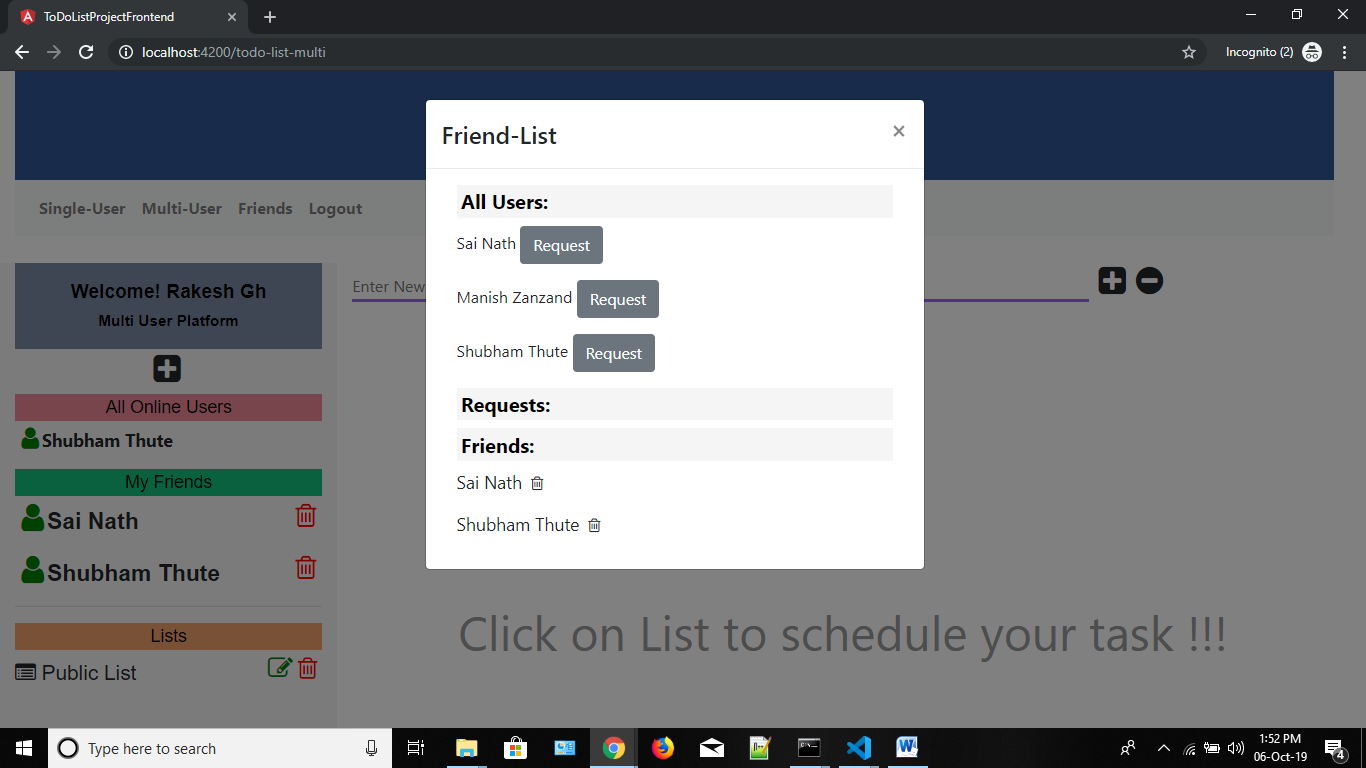


**For Multiuser:**



**While clicking by Admin on any of the user it will show dashboard of the particular user:**

**Friend Request PopUP:**



**More about the application(FrontEnd):**

USER MANAGEMENT PART:

i. Login Page

1. This is the first page to be loaded .
2. If new to application you can redirect to signup page by choosing signup option from navbar.
3. To get in to the TodoListApp use your Email Id and Password to login.

ii. SignUp Page

1. This page will have the form to sign up to the Todo List App.
2. You need to fill all the feilds and then click SignUp button which will then redirect you to Login page if Succesful.
3. Check your mail, you will get notification about registration in TO-DO List

To-Do LIST MANAGEMENT (single user):

1. This is default panel will open after successful login, You can create personal todo list by clicking (+) plus symbol.
2. After list creation, you can do edit and delete functionality on it.
3. Lists are selectable.
4. After selection of list, you can create Items(Task) in it (Note: You cannot create item(task) in list without selection).
5. You can add, delete and edit actions on item.
6. Moreover you can add sub-items to items.
7. There are functionality like add, delete and edit functionality.
8. Also, there is undo functionality for sub-item also.
9. You can undo actions on sub-items by clicking the undo symbol on their parent item.
10. If you logout and login again, you can see your old details again.

FRIEND LIST:

1. In this platform there is "Friends" section in nav-bar.
2. In this friend section, you can see all user to whom you can send friend request.
3. When you click on the request buttom, that perticular user get notification regarding friend request.
4. If that your accept your friend request, You will get notification regarding request acceptance.
5. Also that user will automatically move to your friend list.

To-Do LIST MANAGEMENT (multi-user):

1. On multiuser platform, you can use todoListApp as a real time application.
2. Whereas all the actions on lists, items and sub-items are same, except there will be real time notifications to users regarding each action performed by whom.
3. All lists, Items and Sub-Items are accessible to friends.
4. If any user performed any action on list, Item or Sub-Item, All the other users will get notification of it in real time.
5. You can see list of users which are online.
6. On this platform, there is one more extra functionality added which is clear undos(like clear history).
7. Sometimes we need to add items or sub-items in the lists which should not be modified by any other user, At that time clear undo option will be useful.
8. If you logout and login again, you can see your old details again.

.

**More about the application(Backend):**

1. I have used Mongo Db as a database, used node.js as a platform to build backend server and Express.js is framework on node.js platform to build backend meeting planner application.
2. Hashpassword method use for encryption by using bycrypt.
3. Token Used: Jwt token
4. Logger Library: Pino
5. Socket.io for notification
6. Node Mailer for mailing system.
7. Redis for Cache

## Built With

* [Angular](https://angular.io/) - The web framework used for Frontend Design
* [NPM](https://www.npmjs.com/) - Most of the modules are used
* [nodemailer](https://nodemailer.com/about/) - NPM module to send the mails
* [apiDoc](http://apidocjs.com/) - NPM module to create the apiDoc and eventDoc

## Authors

* **Shubham S thute**

**Guided by:**

* **Edwisor.com**

## Acknowledgments

* Thanks Edwisor for helping to improve my skills ,many more to learn ☺.