# **Index**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Topic** | **Page Number** |
| 1 | Introduction | 1 |
| 2 | Technologies Used | 1 |
| 3 | Steps to run application | 2 |
| 4 | UI Screenshots | 3 |
| 5 | Unit Test Case Report | 6 |
| 6 | Validations & Error Handling | 6 |

# **Introduction**

This project is built for simple task management which allows users to create, update, and delete tasks which also implements smooth and responsive user interactions, including validation, authentication and authorization.

# **Technologies Used**

* Front End – React JS
* Back End – Java, Spring Boot.
* Authentication & Authorization – Sprint Security - JWT
* Back End Server – Docker Container.
* Front End Server – Node JS
* Database – MySQL
* Unit Test FrameWork – Mockito
* Unit Test Plugin - JaCoCo
* Version Control – Git & GitHub
* RestAPI Documentation - Openapi: 3.0

# **Steps to run Application**

**Database Setup:**

1. Install MySQL server.
2. Create below database and tables using the below script,

* create database taskdb
* CREATE TABLE `tasks` (

`TaskId` int NOT NULL AUTO\_INCREMENT,

`Title` varchar(100) NOT NULL,

`Description` varchar(500) NOT NULL,

`Status` varchar(50) NOT NULL,

PRIMARY KEY (`TaskId`),

UNIQUE KEY `Title` (`Title`)

)

* CREATE TABLE `userinfo` (

`id` int NOT NULL AUTO\_INCREMENT,

`name` varchar(100) DEFAULT NULL,

`password` varchar(100) DEFAULT NULL,

`email` varchar(100) DEFAULT NULL,

`roles` varchar(100) DEFAULT NULL,

PRIMARY KEY (`id`)

)

UserInfo table is used to capture the user login credentials.

Tasks Table are used for CRUD operations of task related data.

**Back End:**

1. Download the Back-end application to your workspace from GitHub.
2. Install docker desktop and start the application.
3. In the below path, Task-Management-Application\Back End Task Management Application\src\main\resources\ application.properties – Update the password based on your MySQL Db connection password.
4. Go to Project folder and run the below commands,

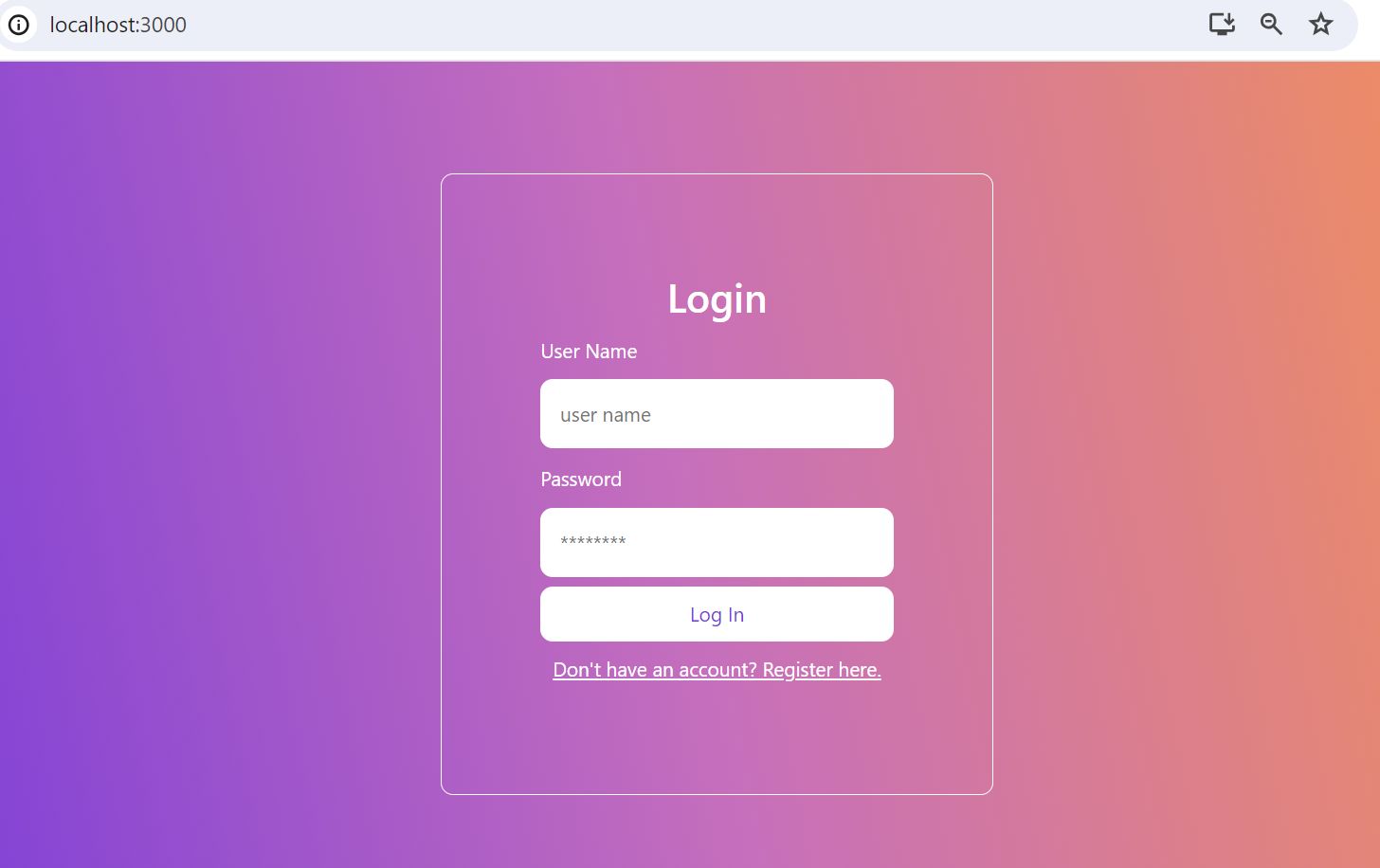
* docker build -f Dockerfile -t task-management-app:1.0 .
* docker-compose up

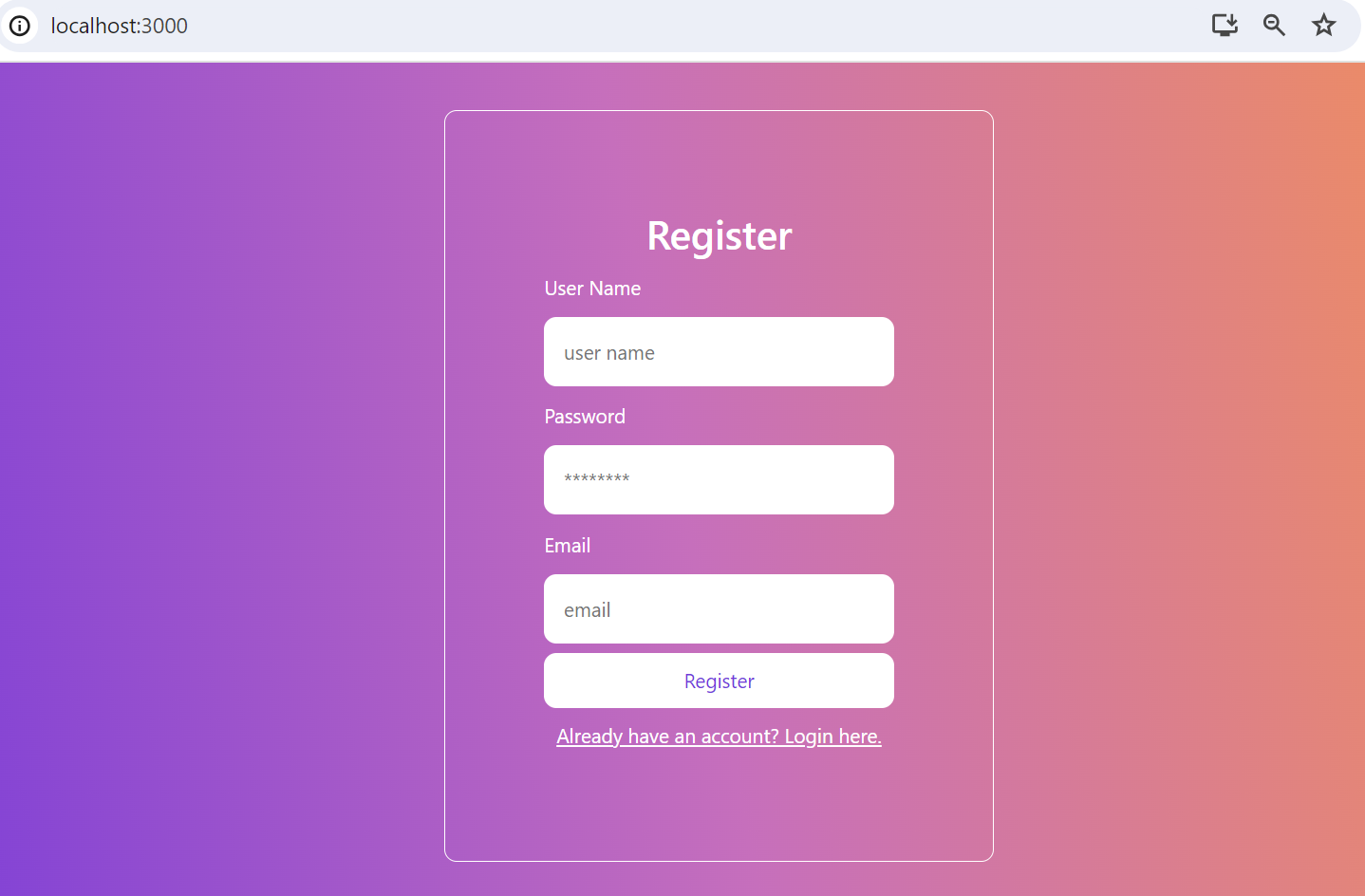
**Front End:**

1. Download the front-end application to your workspace from GitHub.
2. Go to Project folder and run command ‘npm start’.

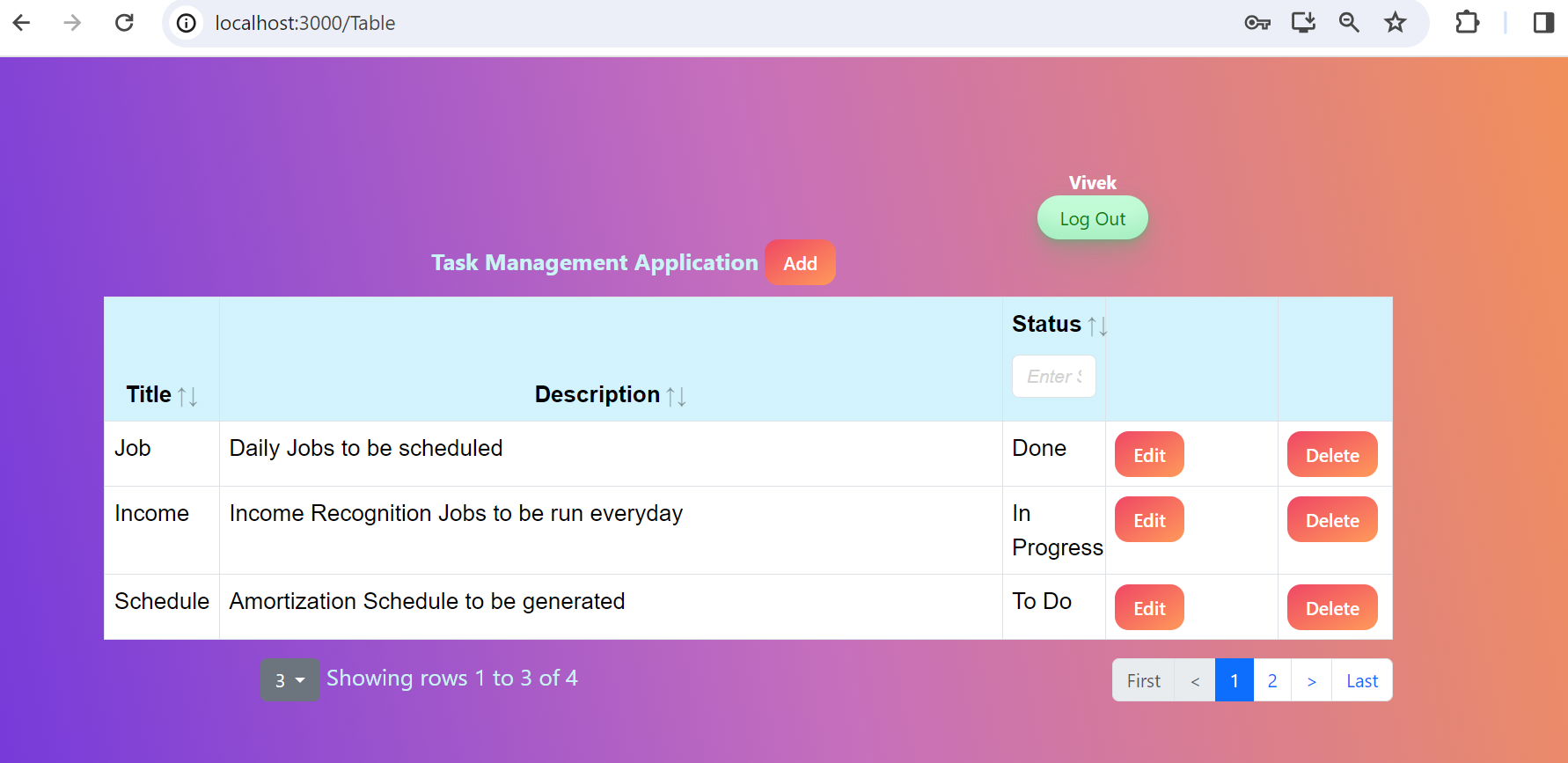
Please refer the below UI screenshots to create new user and navigate through the application.

# **UI Screenshots**

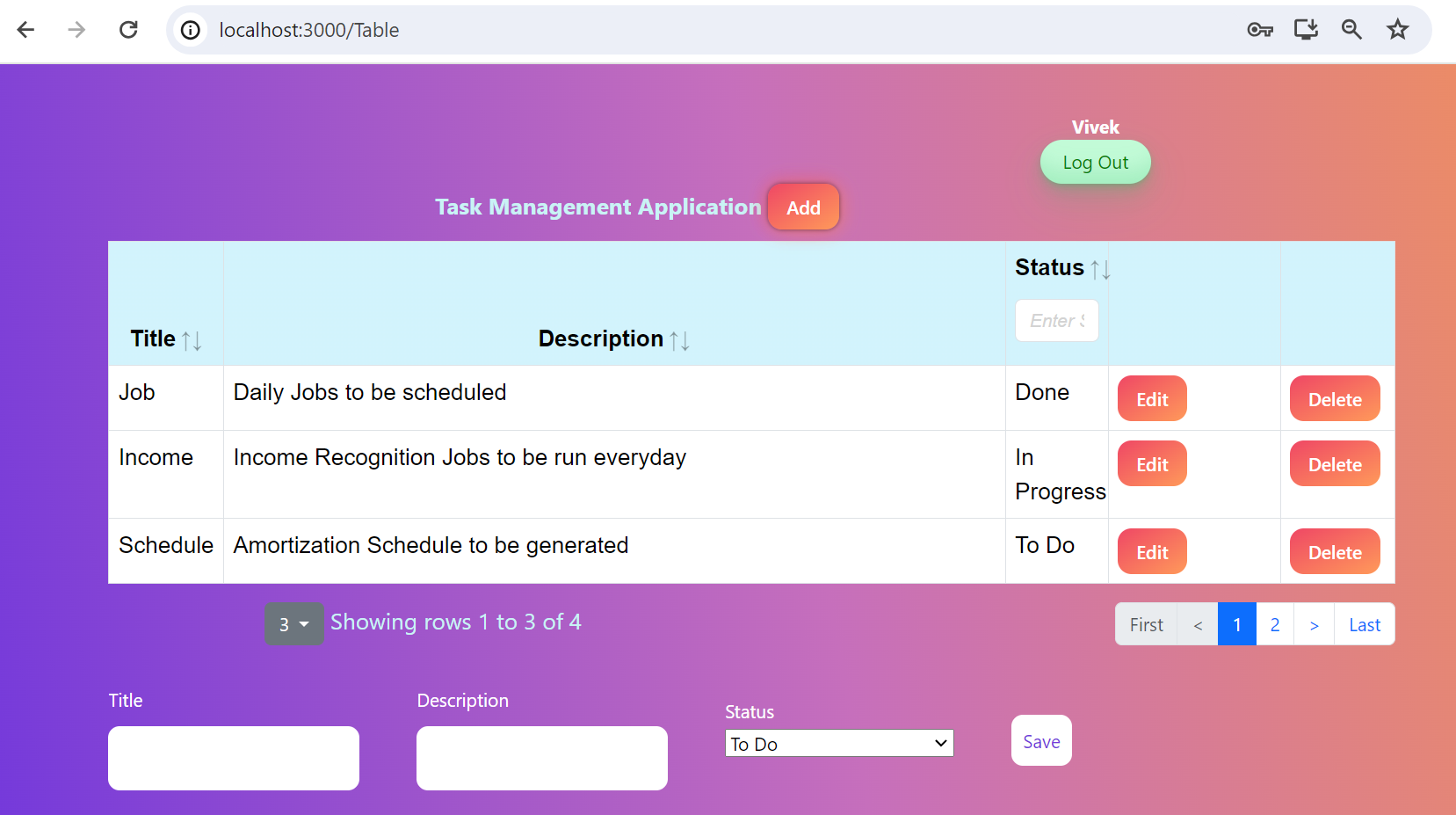
Once the application is launched, below login page will be loaded where user can login with existing credentials or register if new user.

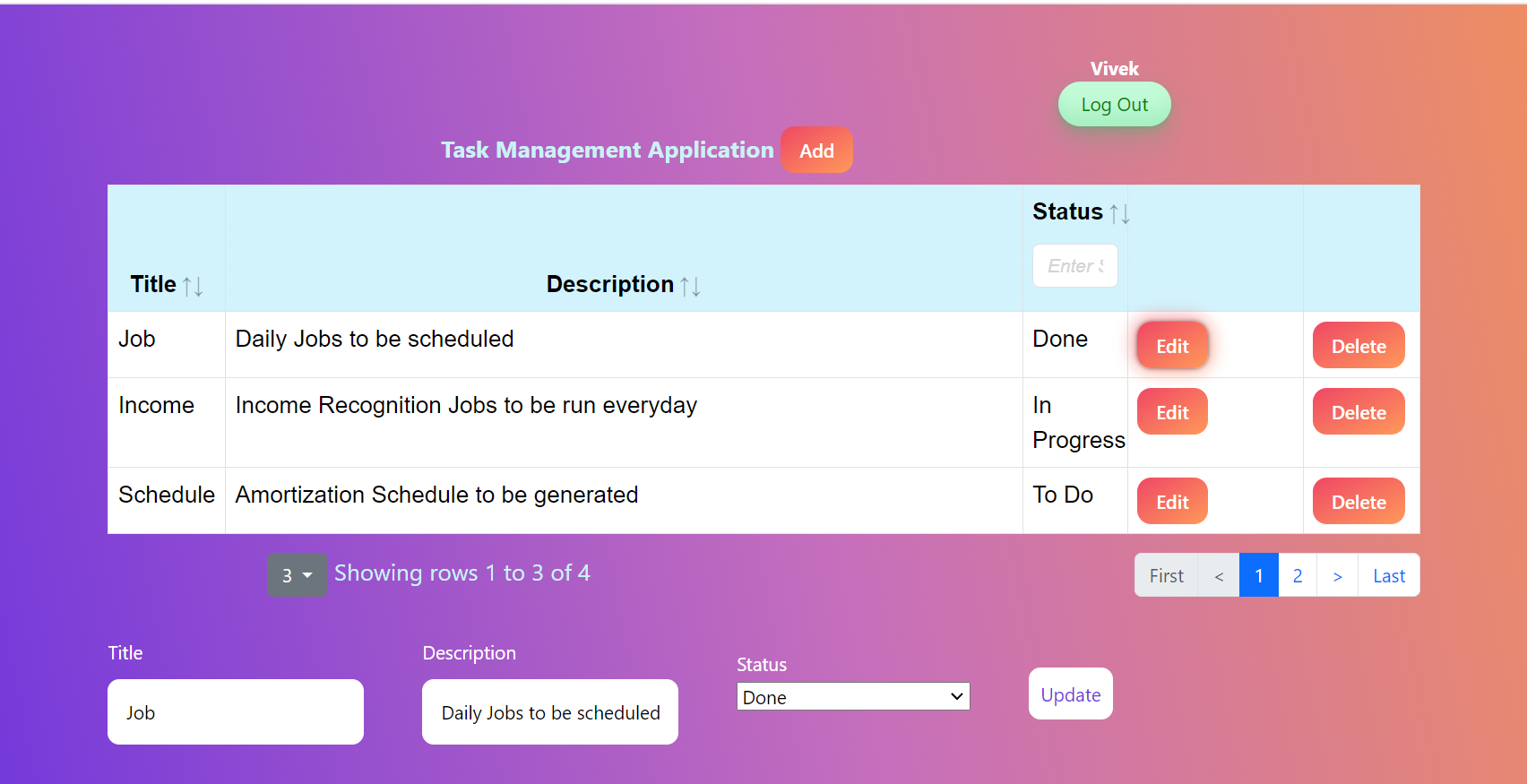


Once the user is logged in successfully after Authentication below form will be displayed for user to Add/View/Edit/Delete the existing tasks.



**Adding new task:**

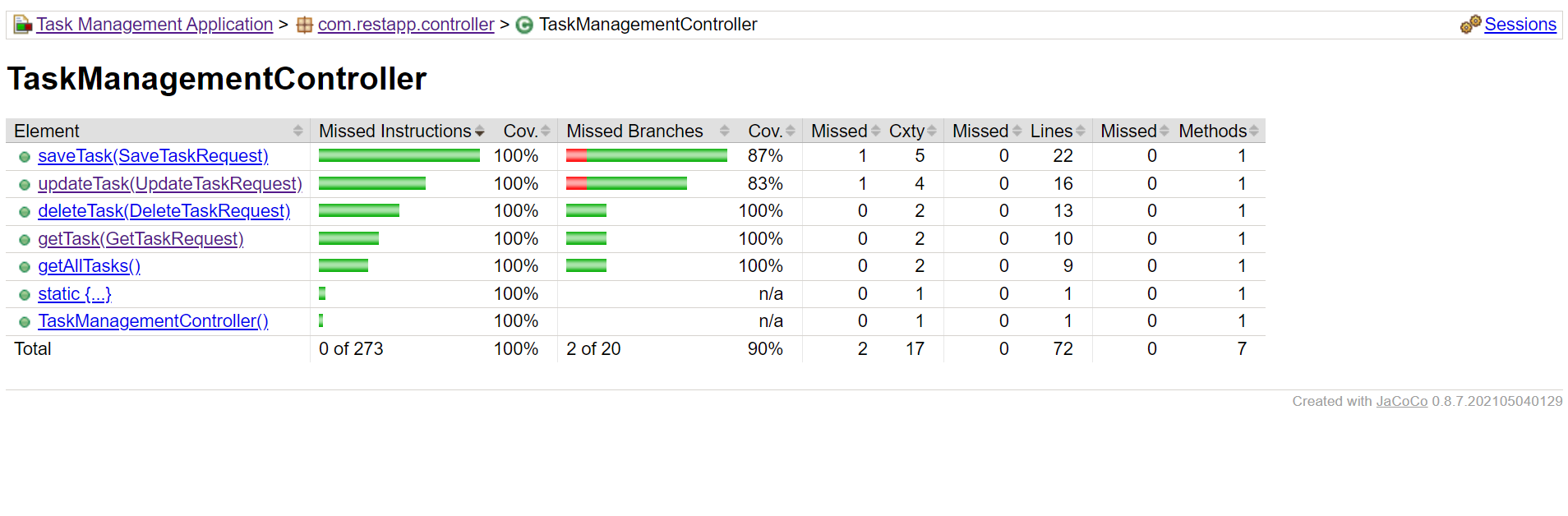


**Updating existing Task:**

# **Unit Test Case Report**

Unit Test cases has been written for critical parts of the application, such as API endpoints and data validation using Mockito.

Below Unit test case report has been generated using JaCoCo plugin. Report has been generated for all the end points.



# **Validations & Error Handling**

Necessary validations and error handling has been added in Login, Register and Add/Edit forms. Back-end API also does validations and error handling for incoming parameters. Please find the below screenshot as reference.

