

---

# Implementation Document

for

## Mess Automation System

Version 0.1

Prepared by

**Group :7**

**Group Name: The Software Savants**

**Ashutosh Agrawal**  
**Sawan H N**  
**Mansi Singh**  
**Prapti Dagli**  
**Sanjay Kumar**  
**Archisha Singh**  
**Mokshagna Prattipati**  
**Charugundla Chethan**

<b>210219</b>	<b>ashutosha21@iitk.ac.in</b>
<b>210952</b>	<b>sawanh21@iitk.ac.in</b>
<b>210590</b>	<b>mansisingh21@iitk.ac.in</b>
<b>210748</b>	<b>praptid21@iitk.ac.in</b>
<b>210080</b>	<b>askumar21@iitk.ac.in</b>
<b>210181</b>	<b>archishas21@iitk.ac.in</b>
<b>210761</b>	<b>mokshagna21@iitk.ac.in</b>
<b>210279</b>	<b>chethan21@iitk.ac.in</b>

**Course: CS253**

**Mentor TA: Archit Gupta (garchit@cse.iitk.ac.in)**

**Date: 20 Feb 2023**



<b>CONTENTS</b>	<b>II</b>
<b>REVISIONS</b>	<b>II</b>
<b>1    IMPLEMENTATION DETAILS</b>	<b>1-4</b>
1.1 PRESENTATION TIER	
1.2 LOGIC TIER	
1.3 DATA TIER	
1.4 DEVELOPMENT AND VERSION CONTROL ENVIRONMENT	
1.5 HOSTING SERVICE	
1.6 AUTHENTICATION AND AUTHORIZATION	
<b>2    CODEBASE</b>	<b>5-6</b>
<b>3    COMPLETENESS</b>	<b>7</b>
<b>APPENDIX A - GROUP LOG</b>	<b>8</b>

## Revisions

Version	Primary Author(s)	Description of Version	Date Completed
0.1	Ashutosh Agrawal Sawan H N Mansi Singh Prapti Dagli Sanjay Kumar Archisha Singh Mokshagna Prattipati Charugundla Chethan	First Draft.	20/03/2023

# 1 Implementation Details

Mess Automation System implements a three-tier architecture, a popular pattern for user-facing applications. The tiers that comprise this architecture includes:

- **Presentation Tier:** This represents the components users directly interact with. In our case, this is the EJS based web application that runs inside a browser.
- **Logic Tier:** This contains the code required to translate user actions to the functionality, written in Java, that drives the application's behavior at the presentation tier.
- **Data Tier:** This consists of databases that hold the data relevant to the application.

## 1.1 Presentation Tier

The presentation layer is accessible to users via a browser and consists of user interface components that enable interaction with the system.

It is developed using three core technologies: HTML, CSS, and JavaScript. While HTML is the code that determines what the website will contain, CSS controls how it will look. JavaScript and its frameworks make the website interactive and responsive to a user's actions.

EJS simply stands for Embedded JavaScript. It is a simple templating language/engine that lets its user generate HTML with plain java script. EJS is preferred as compared to other frameworks due to its

- EJS was so similar to HTML so we found it easy.
- EJS uses all the JS jargon and logic, so if we are proficient in JS, we can use EJS right away.
- EJS is way faster than Jade and handlebars.
- EJS has a really smart error handling mechanism built right into it. It points out to us, the line numbers on which an error has occurred so that we don't end up looking through the whole template file wasting your time in searching for bugs.
- Simple template tags: `<% %>`.
- Custom delimiters (e.g., use `<? ?>` instead of `<% %>`).

The HTML, CSS and Javascript library used in EJS is Bootstrap. Bootstrap is a powerful,extensible,and feature-packed frontend toolkit. It is build and customized with Sass, utilized prebuild grid system and components,and brought projects to life with powerful Javascript plugins. The reason why we have used bootstrap library is:

- **Time-Saving:** When we are bound to an extremely confined timeline to build a website, we can take advantage of bootstrap and nail our project effortlessly because of the ready-made block built ready for us to use them.
- **Easy to Use:** We can swiftly do the bootstrap installation for ourselves without any hassle. Below are the ways to link HTML documents with our stylesheet and script file.
- **Responsive Grid System:** Responsive Grid Systems are of utmost priority as there is an increase in the usage of smartphones. This makes our website's front-end mobile-friendly.

**Cascading Style Sheet(CSS):**

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css"
rel="stylesheet"
integrity="sha384-GLhITQ8iRABdZLI6O3oVMWSktQOp6b7In1Zl3/Jr59b6EGGo1aFkw7cmDA6j6
gD" crossorigin="anonymous">
```

**Javascript:**

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-w76AqPfDkMBDXo30jS1Sgez6pr3x5MIQ1ZAGC+nuZB+EYdgRZgiwxhTBTkF7
CXvN" crossorigin="anonymous"></script>
```

- **Cross-Browser Compatibility:** The Bootstrap team ensures the compatibility of the framework with all modern browsers versions and platforms.

## 1.2 Logic Tier

The Logic Tier accepts user requests from the browser, processes them and determines the routes through which the data will be accessed. The workflows by which the data and requests travel through the backend are encoded in this layer.

The logic tier is implemented in Java, one of the most widely used programming languages in the world.

The advantages of using Java are as follows:

- **Simplicity:** Java has been used by developers for more than 20 years and is considered to be one of the simplest languages to learn due to its less ambiguous syntax terminology.
- **Cross-platform:** Being an object-oriented compiled language, Java allows you to write the code once and run it anywhere on any platform (Windows, Mac OS, and Linux)
- **Multi-threading:** Java uses a multi-threaded web server that processes each request in a separate thread. That enables it to perform several tasks simultaneously without querying the events.
- **Robust & scalable:** The automatic memory management and garbage collection make Java highly scalable and speed up the development of web applications.

The framework used on the top of Java is Spring Boot. Spring Boot is an open-source Java-based framework used to create web services, especially microservices.

Spring Boot offers the following advantages to its developers –

- Easy to understand and develop spring applications
- It provides a flexible way to configure Java Beans, XML configurations, and Database Transactions.
- It provides powerful batch processing and manages REST endpoints.
- In Spring Boot, everything is auto-configured; no manual configurations are needed.
- It offers annotation-based spring application
- Eases dependency management
- It includes Embedded Servlet Container

## 1.3 Data Tier

The Data Tier, sometimes called Database Tier, is where the information processed by the application is stored and managed. We have used MongoDB, which is more than a database. It's a complete developer data platform. With MongoDB Atlas, the cloud offering by MongoDB, you have access to a collection of services that all integrate nicely with your database.

Below are the main advantages/benefits of MongoDB:

- Full cloud-based developer data platform.
- Flexible document schemas.
- Widely supported and code-native data access.
- Change-friendly design.
- Powerful querying and analytics.
- Easy horizontal scale-out with sharing.
- Simple installation.
- Cost-effective.
- Full technical support and documentation.

## 1.4 Development and version control environment

We have used Git as our version control system. It is the most commonly used version control system that is used for software development and other version control tasks. It tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to. It also makes collaboration easier, allowing changes by multiple people to all to be merged into one source.

We used GitHub, a web-based Git repository hosting service to manage our repositories and collaborate amongst ourselves. Jira is a proprietary issue tracking product developed by Atlassian that allows bug tracking and agile project management.

We have used Jira for issue/tickets tracking which helped us in bug tracking and agile project management.

## 1.5 Hosting Service

We have planned to use Heroku, a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud. All three tiers of our application, the frontend, the backend and the database run on the Heroku cloud.

Here are the pros/benefits of using Heroku:

- Allows the developer to focus on code instead of infrastructure
- Generous free-tier is useful for students.
- Enhance the productivity of cloud app development team
- Simple Horizontal & Vertical Scalability
- Offers a powerful dashboard and CLI
- Integrates with familiar developer workflows
- Beginner and startup-friendly

## **1.6 Authentication and Authorization**

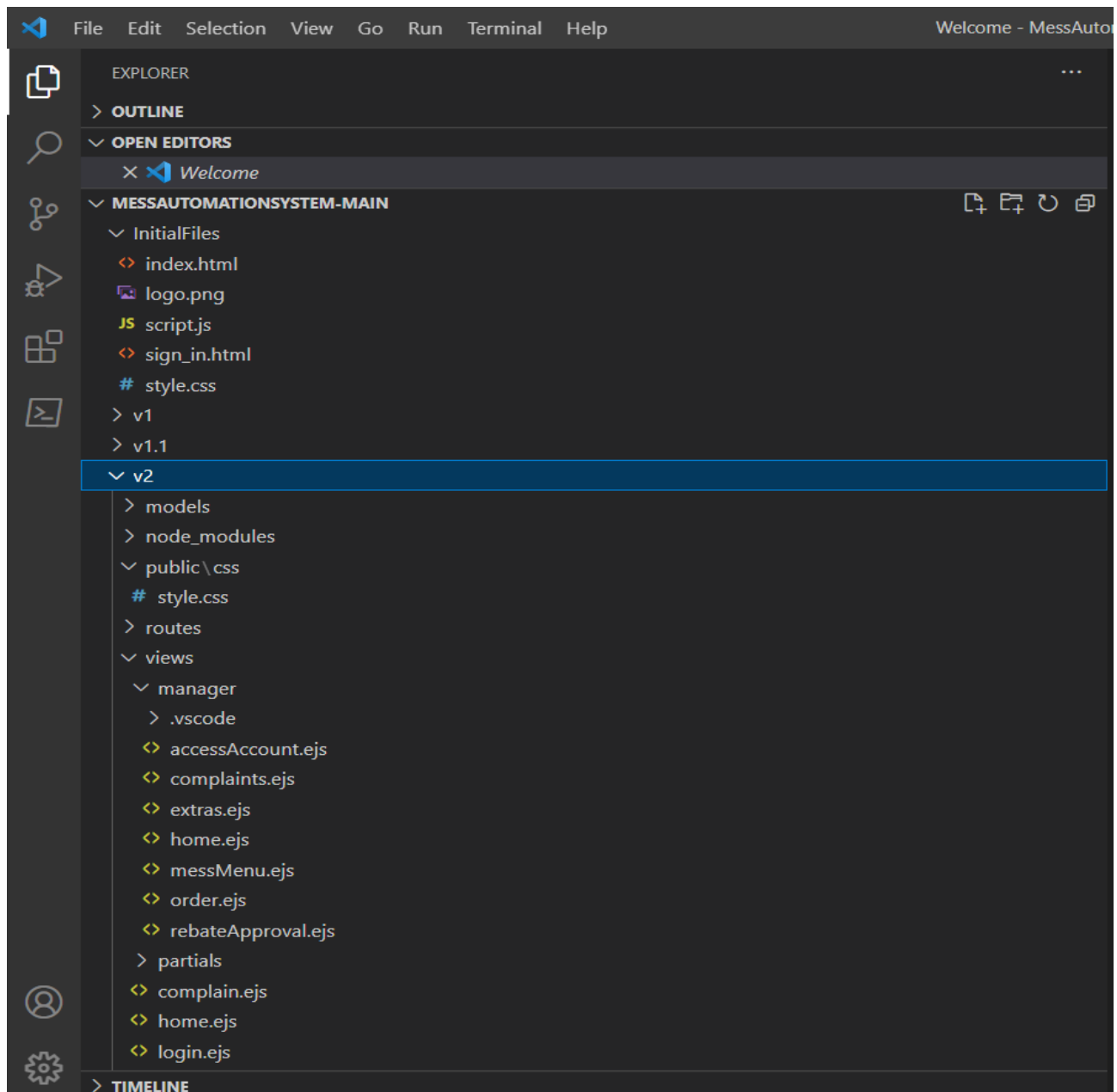
- To implement authentication we have stored the email and password in our database. The password uses cryptographic techniques of hashing and salting to prevent the data being exposed in case of data leaks.
- To implement authorization, we have made use of Passport Local Strategy

## 2 Codebase

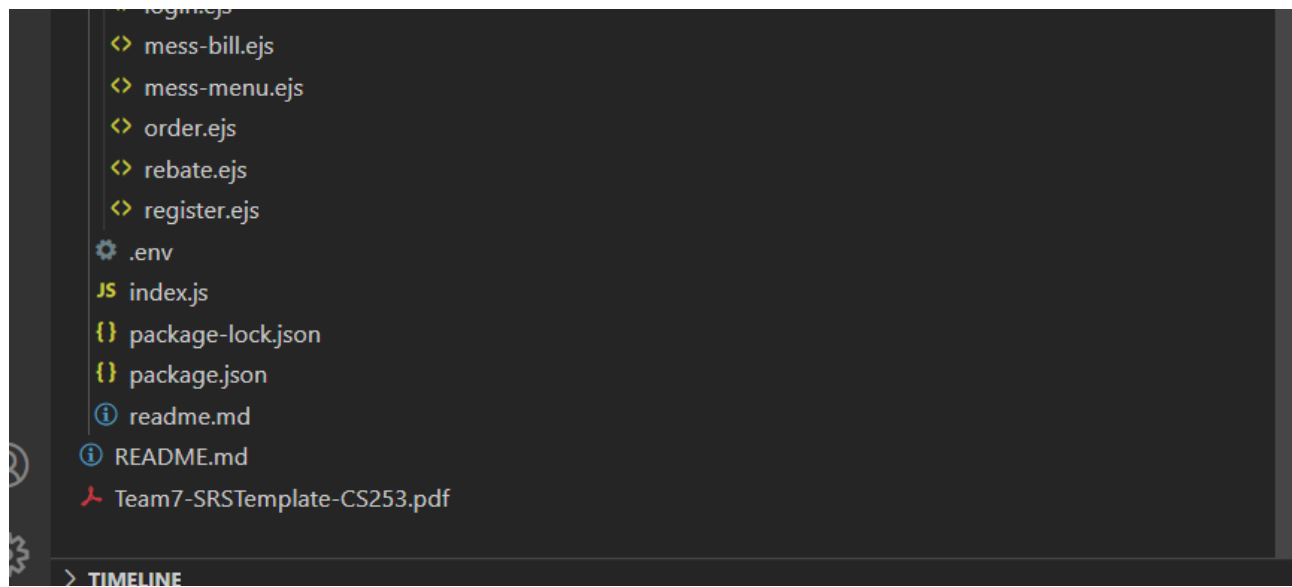
An organization has been set up on GitHub for storing and collaborating on the source code of this project <https://github.com/Ashu3208/MessAutomationSystem>.

v1 and v1.1 contains initial files of the project. All the latest files and directories are in v2. To locally run the project, switch to v2 directory and run 'npm run start' in the terminal. The project will then be live at port 3000 and can be accessed by going to 'localhost:3000'.

Repository Contains the following files:







## 3 Completeness

The “Section 3: Specific Requirements” of Software Requirements Specification Document lists all the desired product functionality.

- “Section 3.1: External Interface Requirements” is implemented in **Mess Automation System-frontend**.

Different user interfaces displayed in section 3.1 of the SRS document are implemented using React. The different views include - Homepage, Sign-in, Student’s interface( orders, complaint, apply for rebate, history, Mess menu, Extras, Mess bill and dues), Manager’s interface( orders, complaints, rebate approval, Mess menu, extras, access account).

- All the features listed in “Section 3.2: Functional Requirements” all the points of hall residence and few points of hall manager are implemented in **Mess Automation System-backend**.

The application - “**Mess Automation System**” is aimed at digitizing various physical hall messes and service providers at the IIT Kanpur hall messes. Currently, the implementation is done to provide the basic features for both the Hall Manager and hall residence, i.e., book mess food and extras(for hall resident), managing orders(for hall manager), checking order history for particular day, feedback on mess food and extras, checking mess rebate, etc. In addition to these features, we have a future development plan to improve the functionality of our product by adding more features which include:

- Adding a SBI collect link in the mess bill page for the ease of students to pay for food and extras.
- Deploying a corresponding guest interface for enhancing the ease of use and accessibility for visitors.
- Possibility of adding an online payment option via credit/debit cards, net banking, UPI to increase the ease of transactions.
- We can also add a feedback page which would provide options for the hall residence so that they can opt for the food and extras they want to have in the mess.
- Providing the contact information of hall mess staff and mess secretary for ease of contacting and fast implementation of complaints.
- Providing the hall residence an option to edit/delete their extras order. This will help mess staff and hall residences.
- Notifying the hall residence about their mess dues and also about their remaining mess dues. We will also notify the hall resident to pay the mess bill before the deadline.
- Providing the contact information of hall mess staff and mess secretary for ease of contacting and fast implementation of complaints.

## Appendix A - Group Log

- The group activities during implementation were asynchronous, where we kept constant communication via our WhatsApp group.
- After people came to campus after mid-sem break, we started working together in RM, CSE Dept. Building.

