DELIVERABLE -4

# GROUP DETAILS:

**Project Title**: Hospital Management System

**Team Name**: Unt Ignitors

**LIST OF TEAM MEMBERS:**

|  |  |
| --- | --- |
| **Members** | **Student ID** |
| DHEERAJ REDDY AGUTHU | 11555619 |
| ABHAY ARORA | 11549231 |
| RAVI TEJA BALAJI | 11514148 |
| PRAVEEN NAKKA | 11532917 |
| CHARISHMA NAGA SAI SARADA BALUSU | 11525468 |
| GEETHA KRISHNA DODDA | 11558520 |
| SRIKANTH GOPI | 11514330 |
| MEGHANA JUNNUTULA | 11539646 |

1. **Test Cases for Phase 2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **S.**  **No** | **Requirement** | **Test Priority** | **Test Steps & pre-**  **conditions** | **Expected Result** | **Obtained Result** | **Success/ Failure** | **Tester** |
| 1 | View Doctor Dashboard | High | 1. User (Doctor) logins to the system 2. On successful login, the system will navigate to Doctor Dashboard Screen | The system will show the Doctor Dashboard with data from various sections (Patient, Doctors, Appointments, Medicines) | Doctor Dashboard is displayed | Success | Abhay |
| 2 | Side Menu in Doctor Dashboard | High | 1. User (Doctor) logins to the system 2. On successful login, the system will navigate to Doctor Dashboard Screen | The system will show the following sections on Side Nav Menu  Appointment, Treatment, Appointment Medicine | The side Menu section is displayed with Appointment, Treatment, Appointment Medicine Sections. | Success | Abhay |
| 3 | List Appointment Screen | High | 1. User (Doctor) logins to the system 2. On successful login, the system will navigate to Doctor Dashboard Screen 3. Click on the Appointment menu from Side Nav 4. The system will navigate to List Appointment Screen | Previously added records will be shown in the grid | The system is showed appointment records in a grid on the List Appointment Screen | Success | Abhay |
| 4 | Add Appointment Screen | High | 1. User (Doctor) logins to the system 2. On successful login, the system will navigate to Doctor Dashboard Screen 3. Click on the Appointment menu from Side Nav 4. The system will navigate to List Appointment Screen 5. Click on Add Appointment Button | The system will display a set of form fields with the button Create Appointment to submit the data | The system showed Form Fields to create a new appointment | Success | Abhay |
| 5 | Edit Appointment Screen | High | 1. User (Doctor) logins to the system 2. On successful login, the system will navigate to Doctor Dashboard Screen 3. Click on the Appointment menu from Side Nav 4. The system will navigate to List Appointment Screen 5. Click on the Edit Appointment Button from the grid | The system will display a set of form fields with the button Update Appointment to update the appointment data. | The system showed Form Fields to update existing appointment | Success | Abhay |
| 6 | Delete Appointment | High | 1. User (Doctor) logins to the system 2. On successful login, the system will navigate to Doctor Dashboard Screen 3. Click on the Appointment menu from Side Nav 4. The system will navigate to List Appointment Screen 5. Click on Delete Appointment Button from the grid | The system will ask for delete record confirmation. On Yes, the record will be deleted from the grid.  On No, the system remains in the same state | On clicking the delete button system asked for confirmation and on clicking yes, records get deleted and the grid refreshes with updated data from the server | Success | Abhay |
| 7 | User will be notified of successfully saved appointment record |  |  | Alert Box/Toast will be shown to the user with a success message after creating or updating the Appointment action | Toast messages appear on the screen with appropriate success messages after creating and updating the action |  | Abhay |
| 8 | The user will be notified of the successful deletion of the Appointment Record |  |  | Alert Box/Toast will be shown to the user with the success message. | Toast messages appear on the screen with an appropriate success message after the delete action |  | Abhay |

1. **User Manual**

**4.1 Installation of Required software**

To run the Hospital management system, we require a certain set of software and IDEs to run the code and provide an output.

Graphical user interface, text, application

Description automatically generated

*Fig 4*

* Please install the “Node.js” software to run the script, from the following link
* [**https://nodejs.org/en/download/**](https://nodejs.org/en/download/)

**Step 2: Install the MySQL Workbench**

Graphical user interface, text

Description automatically generated

*Fig 5*

* Please install the “MySql” workbench to connect the database to the UI through WebAPI**,** [**https://www.mysql.com/downloads/**](https://www.mysql.com/downloads/)

**Step 3: Install Visual studio**

Timeline

Description automatically generated with medium confidence

*Fig 6*

* Please install Visual Studio to connect the database and WebAPI using the below link [**https://visualstudio.microsoft.com/downloads/**](https://visualstudio.microsoft.com/downloads/)

1. **Instructions to compile and run the program**

**Step 1**

* Download the code repository from GitHub. To download follow the below steps
* Create a new folder HMS\_code\_base and open the folder in “CMD” or Terminal.
* Run command: **git clone** [**https://github.com/abhayarora23UNT/UntSeProjects2022.git**](https://github.com/abhayarora23UNT/UntSeProjects2022.git)
* This will clone all the project code into the created folder.

**Step 2**

* To compile the program in a local machine you should have node and angular installed.
* Download and install node js from <https://nodejs.org/en/download/>
* Open Cmd/terminal and run the command npm install –g @angular/cli
* User node -v and ng –version commands to check whether the above software is installed.
* Go to path “HMS\_code\_base /UntSeProjects2022/HMS/Source Code/Front\_End/HmsApp/” open Cmd/Terminal.
* And run the command npm install. This will install all the required dependencies to run the application.
* After successful installation, run command ng serve to run the application in the local machine. This will run your application in localhost:4203 (port can be changed from angular. json)

Text

Description automatically generated

*Fig 7*

Text

Description automatically generated

*Fig 8*

**Step 3**

Please open the browser and search as **“localhost:4203”** to access the application

Graphical user interface, application

Description automatically generated

*Fig 9*

Users are required to enter their credentials to log in to the website. To login into the system. Users are required to register first in the system by providing the basic information given in the below screenshot

* The user of Type(Doctor), logins into the system
  + On Success, System will show Doctor Dashboard Screen

A screenshot of a computer

Description automatically generated with medium confidence

* The user clicked on Appointment Menu from Side Nav Section.
  + The system will navigate to List Appointment Screen

A picture containing graphical user interface

Description automatically generated

* User clicks on Add Appointment Button
  + The system will navigate to Add Appointment Screen

Graphical user interface, application, Teams

Description automatically generated

* The user clicks on Delete Button
  + The system will show the confirmation to delete the record

Graphical user interface

Description automatically generated with medium confidence

1. **Peer Review Feedback**

.

1. **Accomplishments/Challenges**

**8. Member Contribution Table**

|  |  |  |
| --- | --- | --- |
| **Member Name** | **Contribution Description** | **Overall Contribution (%)** |
| DHEERAJ REDDY AGUTHU | * Deliverable 4 Documentation   + - 1. UML (Class Diagram)       2. Test Cases       3. User Manual * Backend Database creation * Testing | 12.5 |
| ABHAY ARORA | * Updated the Minutes of Meeting in Repo * Deliverable 4 Documentation  1. Code Inspection Doc 2. Test Cases 3. User Manual  * Front End Development(Initial Architecture, Doctor Dashboard, Appointment(List, Edit, Delete) * Deployment and Unit Testing * Managing Git and Trello Board * Deliverable 4 Task Management | 12.5 |
| RAVI TEJA BALAJI | * Development phases and Member Contribution tables. * Deliverable 4 Documentation  1. UML (Class Diagram) 2. Peer Review Section Feedback 3. Test Cases | 12.5 |
| PRAVEEN NAKKA | * Deliverable 4 Documentation  1. Requirements 2. Peer Review Section Feedback  * Front End Development (Home Screen, Login Screen) | 12.5 |
| CHARISHMA NAGA SAI SARADA BALUSU | * Deliverable 4 Documentation  1. Sequence Diagrams  * Testing | 12.5 |
| GEETHA KRISHNA DODDA | * Deliverable 4 Documentation  1. Sequence Diagrams  * Testing | 12.5 |
| SRIKANTH GOPI | * Deliverable 4 Documentation  1. Requirements 2. UML (Use Case Diagram) 3. Test Cases | 12.5 |
| MEGHANA JUNNUTULA | * Deliverable 4 Documentation  1. Requirements 2. UML (Use Case Diagram) 3. User Manual 4. Code Inspection  * API Initial Structure + API implementation (Admin and Doctor Module) * Front End Development(Doctor-Treatment Screen ) * DB Schema and Records changes | 12.5 |

**9. Minutes of Meeting**

Minutes of the meeting are updated below path in the project repository

<https://github.com/abhayarora23UNT/UntSeProjects2022/tree/main/HMS/MOM>