Team Members

Project Leader - Thesath Wijayasiri (33848612)

Program Manager - Pitchaigani Farjhana Begam (33787952)

Programmer - Kailasam Sasikumar (33787255)

Lead Programmer - Sai Karthik Miryala (33804652)

System Analysts - Abdul Rahman Suhail (33738202)

Document Version: 1.0

18th June 2021

Client/ Organization Name: Mr. Peter Cole, Murdoch University

Project Supervisor – Dr. Florence Mwagwabi

Project Name: MeetMe

DESIGN DOCUMENT

**Table of Contents**

[1. Executive Summary 4](#_Toc74919206)

[2. Introduction: 5](#_Toc74919207)

[3. Data Design: 5](#_Toc74919208)

[3.1 Entity Relationship Diagram: 5](#_Toc74919209)

[3.2 Data Dictionary: 7](#_Toc74919210)

[3.3 CRUD Matrix: 7](#_Toc74919211)

[4. Process Design: 8](#_Toc74919212)

[4.1. Use Case Lists 8](#_Toc74919213)

[4.1.1.Actor- Convenor 8](#_Toc74919214)

[4.1.2 Actor- Organiser 9](#_Toc74919215)

[4.1.3 Actor-Attendee 9](#_Toc74919216)

[4.2 Event Decomposition Diagram: 10](#_Toc74919217)

[4.3 Use Case List: 11](#_Toc74919218)

[4.3.1 Actor – Convenor/Organiser 11](#_Toc74919219)

[4.3.1.1 Register Account 11](#_Toc74919220)

[4.3.1.2 Login to Account 12](#_Toc74919221)

[4.3.1.3 Schedule the preferred time 12](#_Toc74919222)

[4.3.1.4 Modify the time slot 13](#_Toc74919223)

[4.3.1.5 Upload Attendee List 14](#_Toc74919224)

[4.3.1.6 Check Schedule list 15](#_Toc74919225)

[4.3.1.7 Modify Schedule list 16](#_Toc74919226)

[4.3.1.8 Save Schedule list to computer 16](#_Toc74919227)

[4.3.1.9 Add Schedule list 17](#_Toc74919228)

[4.3.1.10 Reset Password 18](#_Toc74919229)

[4.4 Actor – Attendee 19](#_Toc74919230)

[4.4.1 Register for Meeting 19](#_Toc74919231)

[5. Architecture/Infrastructure Design: 20](#_Toc74919232)

[5.1 Architectural Design: 20](#_Toc74919233)

[5.2 Software components: 21](#_Toc74919234)

[5.3 Infrastructural Design: 22](#_Toc74919235)

[5.4 Infrastructure requirements: 23](#_Toc74919236)

[5.5 Network Topology: 25](#_Toc74919237)

[6. Interface Design: 26](#_Toc74919238)

[Placing users in control of the interface 26](#_Toc74919239)

[Make interactions comfortable 26](#_Toc74919240)

[Reduce cognitive load 27](#_Toc74919241)

[Consistent user interface 27](#_Toc74919242)

[Color Scheme 27](#_Toc74919243)

[7. Appendices: 29](#_Toc74919244)

[7.1 Appendix A: Deliverable task Breakdown Statement 29](#_Toc74919245)

[7.2 Appendix B: Data Dictionary 30](#_Toc74919246)

[7.3 Appendix C: Glossary of terms and definitions and acronyms 33](#_Toc74919247)

Document Version: 1.0

04th June 2021

REQUIREMENTS AND ANALYSIS

# Executive Summary

This document will function as the first version of the Design document. FT group 1 will design the MeetMe website project, an online booking system for meeting. This website will address the current system that has been used by Murdoch University whereby the academic staff create a calendar of the meeting. The challenges that has been faced by the current system are as follows:

* Time consuming in creation of calendar events
* Distribution of meeting links individually through sorting in the calendar events
* Changes in the meeting is a long process and time consuming

MeetMe website will tackle all the issues that has been faced by the current system and certain design objectives are considered. The design objectives of this system are listed below:

* User friendly UI
* Lesser processing time
* Interactive UI
* Secure database
* Automated personalised email to students
* Ability to upload the schedule into the system and it auto generates the calendar
* One-time link for students to participate in the meeting (Virtual meeting)
* Email links which adds as a calendar event in the outlook

The main objective that MeetMe website is reduce the time and process of the current system and automate most of the process. As such this document will detail the design blueprint of the MeetMe project, this document helps to rebuild when the system is destroyed or corrupted.

As stated in the introduction below, the goal of this project is to develop a web based, meeting scheduling software for the use of Mr. Peter Cole and his organization, Murdoch University.

This document consists of

Introduction: an introduction to the project and the system

Data Design: it’s a database design consisting of data flows and data structures of the database.

Process Design: it’s a use case diagram which describes each process flow.

Architecture/Infrastructure Design: it’s a model to execute the system architecture.

Interface Design: it’s a user interface, it provides an interaction between the human and computer

# Introduction:

The design document provides a blueprint of the MeetMe system. It provides a clear picture of the design and structure of the system. Before this document, the team prepared the Requirements and Analysis Document and the PMP. The former document describes the system's functional and non-functional requirements while the subsequent outlines the processes in the knowledge fields of project management followed by the stakeholders during the life cycle of the project to effectively complete the system. The process of development follows the policies and procedures stated in the relevant publications. The team works simultaneously to build and develop a test strategy. The test plan ensures that every aspect of the system is carried out with high quality and that the scope indicated in those documents is complied with. The design document describes the database structure and design. The fields in each table are specified together with their data types and table relationships. The following diagrams are represented in an Entity Relationship Diagram (ERD). It consists of a CRUD matrix that sets out the activities that a user can do and how this impacts the database. The following section includes the process design describing how the workflow, equipment and implementation requirements for a process are defined. In this design document major factors are discussed such as the Data design, Process Design, Architecture/Infrastructure Design ad Interface Design.

# Data Design:

## 3.1 Entity Relationship Diagram:

MeetMe ERD diagram below in figure 3.1.1 represents the system data design; each attribute represents an entity-relationship with each other, cardinality of the entity relations and included with Legends.

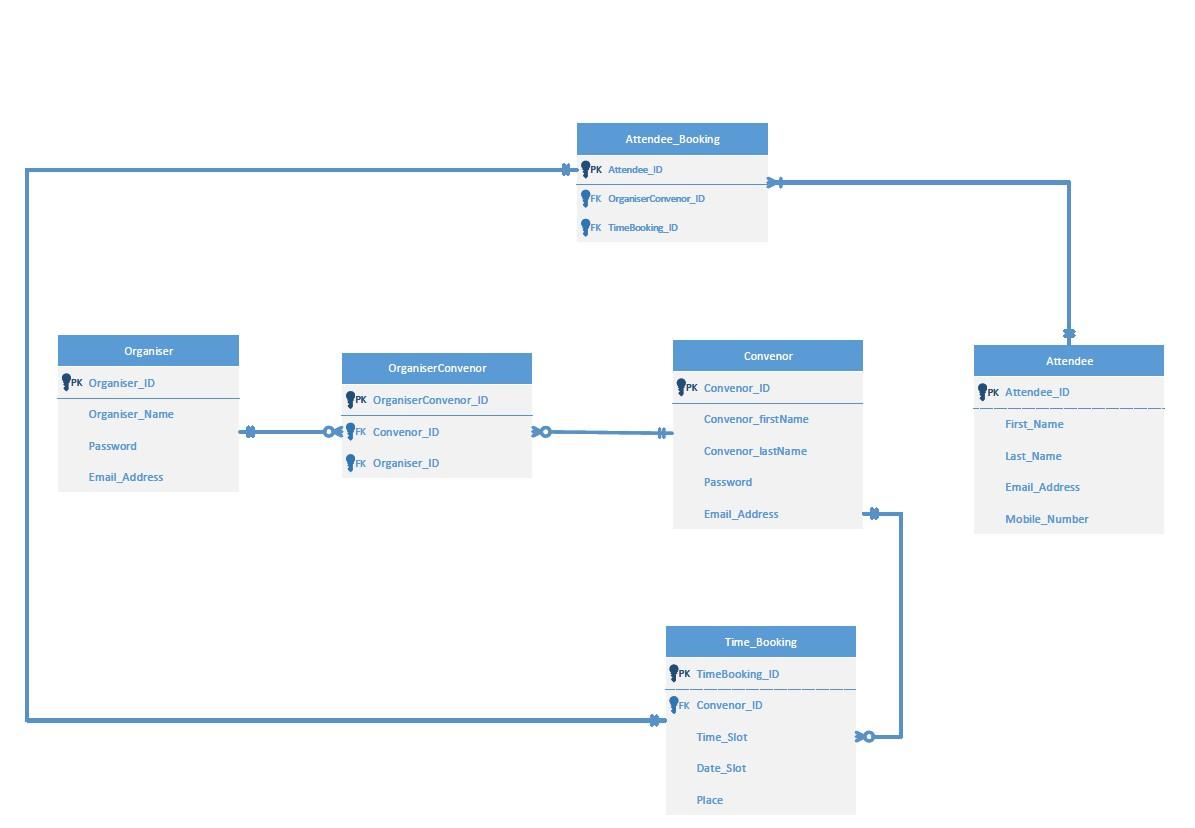


Figure 3.1.1 Meet Me Entity Relationship Diagram

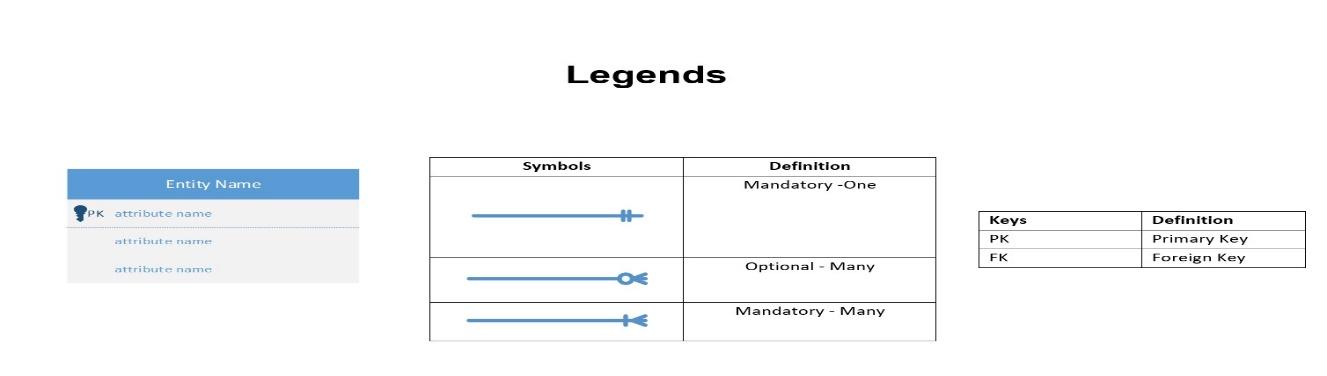


Figure 3.1.2 Legends

## 3.2 Data Dictionary:

Data Dictionary that represents Describes how each attribute represents: Description of the attributes, data type, required, NULL and Key. Below all table shows 7.2 Appendices B; Figure 7.2.1 is Organiser; Figure 7.2.2 is OrganiserConvenor, Figure 7.2.3 Convenor, Figure 7.2.4 Attendee, Figure 7.2.5 Time Booking, Figure 7.2.6 AttendeeBooking These are each function description with the detailed data dictionary.

## 3.3 CRUD Matrix:

CRUD Matrix is useful to capture role and permission within the MeetMe system Below in the table Figure 3.3.1

Create -Create a record

Read - Read data

Update - Update the data

Delete - Delete the data or remove the data

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Use Case Model** | **Organiser** | **OrganiserConvenor** | **Convenor** | **Attendee** | **Time\_Booking** | **Attendee\_Booking** |
| Register Account | C |  | C |  |  |  |
| Login to Account | R |  | R |  |  |  |
| Schedule the preferred Time |  |  |  |  | C | U |
| Modify the time slot | U |  | U |  | U |  |
| Upload Attendee list |  |  |  | UD |  |  |
| Check Schedule list | R | R | R |  |  |  |
| Modify Schedule list | RU |  | RU |  |  |  |
| Save Schedule list to computer | R | R | R |  |  |  |
| Add Schedule to Calendar | C | C | C |  |  |  |
| Reset Password | U | U | U |  |  |  |
| Meeting Registration link Sent | C | C | C |  |  |  |
| Register for Meeting |  |  |  | C |  |  |
| Schedule Available Time |  |  |  | C | C | C |
| Booking Confirmation Slot |  |  |  | C |  |  |

Figure 3.3.1 CRUD Matrix table

# Process Design:

## 4.1. **Use Case Lists**

### ***4.1.1.Actor- Convenor***

|  |  |  |
| --- | --- | --- |
| **Primary Actor(Convenor)** | | |
| **No.** | **Use Case Name** | **Description** |
| 1. | Register Account | The actor Register an account to Access the MeetMe to Schedule the meet. |
| 2. | Login to Account | The actor Login to access the MeetMe Account. |
| 3. | Schedule the preferred Time | The actor has booked the preferred time for the meeting. |
| 4. | Modify the time slot | The actor can change the time slot for the meeting. |
| 5. | Upload Attendee list | The actor can Upload the attendee list such as Excel spreadsheet etc. |
| 6. | Check Schedule list | The actor can check schedule list of the Meet |
| 7. | Modify Schedule list | The actor can change the schedule list. |
| 8. | Save Schedule list to computer | The actor can save the schedule list into the computer. |
| 9. | Add Schedule to Calendar | The actor can Add the schedule to his calendar. |
| 10. | Reset Password | The actor can reset the password when forgot password |

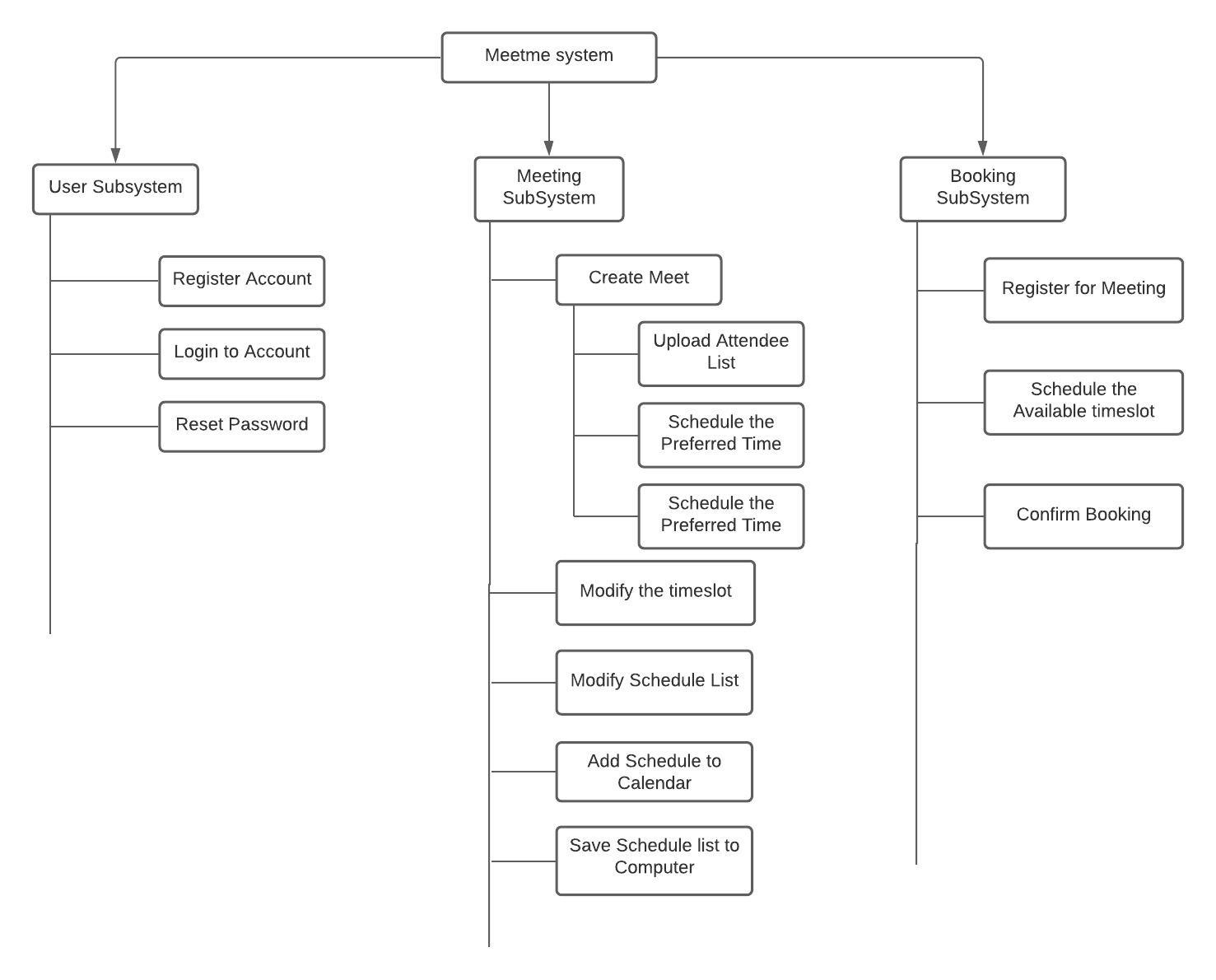
### 4.1.2 Actor- Organiser

|  |  |  |
| --- | --- | --- |
| **Primary Actor(Organiser)** | | |
| **No.** | **Use Case Name** | **Description** |
| 1. | Register Account | The actor Register an account to Access the MeetMe to Schedule the meet. |
| 2. | Login to Account | The actor Login to access the MeetMe Account. |
| 3. | Upload Attendee list | The actor can Upload the attendee list such as Excel spreadsheet etc. |
| 4. | Add Schedule to Calendar | The actor can Add the schedule to his calendar. |
| 5. | Reset Password | The actor can reset the password when forgot or changed |

### 4.1.3 Actor-Attendee

|  |  |  |
| --- | --- | --- |
| **Primary Actor(Attendee)** | | |
| **No.** | **Use Case Name** | **Description** |
| 1. | Meeting Registration link Sent | The Actor can receive with registration link to register for the meeting |
| 2. | Register for Meeting | The Actor can Enter the details to Register for meeting |
| 3. | Schedule the Available Time | The actor can schedule the Available time. |
| 4. | Booking Confirmation Slot | Attendee Confirms his meeting. |

## 4.2 Event Decomposition Diagram:

****

## 4.3 Use Case List:

### 4.3.1 Actor – Convenor/Organiser

### 4.3.1.1 Register Account

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Register Account | |
| Use-Case ID | 1 | |
| Scenario | The Actor is a new User | |
| Description | The actor registers with the MeetMe System | |
| Triggering Event | Actor clicks “Create an Account” Button in the MeetMe website in the Login Page / Actor Visits the MeetMe Signup Page | |
| Actors | Convenor, Organiser | |
| Stakeholders | Convenor, Organiser, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor does not have an account previously with same credentials | |
| Post-condition | A new Actor is added to the system | |
| Flow of Activities | Actor | System |
| 1. Enters his details in the form  3. Submits the form | 2.Validates the users input  4.Checks if the user exists  4.1 Adds a new User Account |
| Exception Conditions: | 4a) If User Account Exists already  1) Display Use Account Exists  2) Redirect to Login Page | |

### 4.3.1.2 Login to Account

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Login to Account | |
| Use-Case ID | 2 | |
| Scenario | The Actor is a registered User | |
| Description | The actor Logs into the MeetMe System. | |
| Triggering Event | Actor visits the MeetMe Login Page/ Actor clicks “Sign In” Button in Signup Page | |
| Actors | Convenor, Organiser | |
| Stakeholders | Convenor, Organiser, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor is a Registered User | |
| Post-condition | A Dashboard is displayed to the Actor | |
| Flow of Activities | Actor | System |
| 1. Enters his details in the form  3. Submits the form | 2.Validates the users input  4.Checks if the user exists  4.1 Displays the Dashboard to the User |
| Exception Conditions: | 4a) If User Account does not exist  1) Display Use Account Exists  2) Redirect to SignUp page | |

### 4.3.1.3 Schedule the preferred time

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Schedule the preferred time | |
| Use-Case ID | 3 | |
| Scenario | The Actor inputs the preferred date and time for a meeting | |
| Description | The Actor sets the date and the time where he is available for the meeting. | |
| Triggering Event | Actor Creates a new Meeting | |
| Actors | Convenor, Organiser | |
| Stakeholders | Convenor, Organiser, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor is logged in with the MeetMe System | |
| Post-condition | System stores the preferred date and time. | |
| Flow of Activities | Actor | System |
| 1. Creates a new Meeting  3.Enters the preferred date and time  3.1 Press “Next” | 2.Prompts to set the preferred date and time  4. System stores the preferred date and time. |
| Exception Conditions: | 3a) If Actor does not enter the preferred date and time  1) Next is disabled | |

### 4.3.1.4 Modify the time slot

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Modify the time slot | |
| Use-Case ID | 4 | |
| Scenario | The Actor modifies the chosen timeslot | |
| Description | Actor can modify the preferred date and time slots for the meeting | |
| Triggering Event | Actor clicks modify button available on the meeting column in the table | |
| Actors | Convenor | |
| Stakeholders | Convenor, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor chooses the preferred date and time slots before. | |
| Post-condition | System sets the modified date and time slot for the meeting | |
| Flow of Activities | Actor | System |
| 1. Click “Modify” button on the meeting column    3. Inputs the preferred date and time slots  3.1. Click “Next” | 2.Prompts to set the preferred date and time slots    4. System stores the new preferred date and time. |
| Exception Conditions: | 3a) If Actor does not enter the preferred date and time  1) Next is disabled | |

### 4.3.1.5 Upload Attendee List

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Upload Attendee List | |
| Use-Case ID | 5 | |
| Scenario | Actor uploads the Attendee List to the System | |
| Description | Actor uploads an excel file with all the Attendees Details and the meeting timings to the system | |
| Triggering Event | Actors create a new Meeting | |
| Actors | Convenor, Organiser | |
| Stakeholders | Convenor, Organiser, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor chooses the preferred date and time slots before. | |
| Post-condition | System sends the Meeting Invitation to all Attendees | |
| Flow of Activities | Actor | System |
| 1. Uploads the Attendees List  3. Users clicks “Next” Button. | 2. Checks if the List file is Valid  2.1 Stores all the Meeting Details  4. Finish Creating a new meeting  4.1 Send Invitation Link to all the Attendees |
| Exception Conditions: | 2a) If the file is Invalid  1) Displays error Message  2) Disable Next Button | |

### 4.3.1.6 Check Schedule list

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Check Schedule List | |
| Use-Case ID | 6 | |
| Scenario | Actor want to check all the Meeting Schedules Lists | |
| Description | Actor can check all the Meeting Schedules List the  Browser | |
| Triggering Event | Click “Schedule” Section in the Dashboard | |
| Actors | Convenor | |
| Stakeholders | Convenor, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor must create a new Meeting | |
| Post-condition | System must display the Schedule Meeting Details | |
| Flow of Activities | Actor | System |
| 1. Click “View Schedule” in the Dashboard | 2. Checks if any Meeting is available  2.1 Displays the Meeting Details |
| Exception Conditions: | 2a) If Meeting is not Available  1) Displays “No Meeting Available” Message | |

### 4.3.1.7 Modify Schedule list

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Modify Schedule list | |
| Use-Case ID | 7 | |
| Scenario | Actor access to modify existing schedule lists in the system. | |
| Description | Actors can edit the existing list and modify the schedule list in the system. | |
| Triggering Event | Click “Modify Schedule” Section in the Dashboard | |
| Actors | Convenor | |
| Stakeholders | Convenor, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor can Modify the Schedule in the system | |
| Post-condition | System must show the Schedule list | |
| Flow of Activities | Actor | System |
| 1. Click “View Schedule” in the Dashboard | 2. Check the schedule list  3. Modify the schedule list |
| Exception Conditions: | 2a) If Meeting is not Available  1) Displays “No Schedule list” Message | |

### 4.3.1.8 Save Schedule list to computer

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Save Schedule list to computer | |
| Use-Case ID | 8 | |
| Scenario | Actor access save the schedule list in the computer | |
| Description | The actor can save the schedule list into the computer. | |
| Triggering Event | Click “Export” Section in the Dashboard | |
| Actors | Convenor | |
| Stakeholders | Convenor, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor can download and save the schedule list into computer | |
| Post-condition | System must show the schedule list then we can Export the schedule list. | |
| Flow of Activities | Actor | System |
| 1. Click “Export ” in the Dashboard | 2. Save the Schedule list to computer |
| Exception Conditions: | 2a) If Meeting is not Available  1) Displays “No Schedule list” Message | |

### 4.3.1.9 Add Schedule list

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Add Schedule list | |
| Use-Case ID | 9 | |
| Scenario | Actor access to add all schedule list to his calendar | |
| Description | The actor can save the schedule list into the computer. | |
| Triggering Event | Click “Export” Section in the Dashboard | |
| Actors | Convenor | |
| Stakeholders | Convenor, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor can download and save the schedule list into computer | |
| Post-condition | System must show the schedule list then we can Export the schedule list. | |
| Flow of Activities | Actor | System |
| 1. Click “Export ” in the Dashboard | 2. Save the Schedule list to computer |
| Exception Conditions: | 2a) If Meeting is not Available  1) Displays “No Schedule list” Message | |

### 4.3.1.10 Reset Password

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Reset Password | |
| Use-Case ID | 10 | |
| Scenario | Actor wants to reset his password or forgot his password | |
| Description | The actor can reset his password in case he forgets the password, or he wish to change his password | |
| Triggering Event | Click “Reset Password” in the login Page | |
| Actors | Convenor, Organiser | |
| Stakeholders | Convenor, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor must have a valid registered email id. | |
| Post-condition | System should let the actor login with the new password. | |
| Flow of Activities | Actor | System |
| 1. Click “Reset Password” link in the login page  3. Actor inputs the user id or email  5. Click the link in the mail.  5.1 Input new password | 2.Prompts for the User Id / email  4. Send the reset link to the registered mail  6. System updates the password |
| Exception Conditions: | 4a) If Mail is not found  1) Displays “No Email Found” Message | |

## 4.4 Actor – Attendee

### 4.4.1 Register for Meeting

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Register for Meeting | |
| Use-Case ID | 2 | |
| Scenario | Users gets the Invitation Link from the MeetMe System and wants to register for a meeting | |
| Description | Actor clicks the Invitation Link in the Mail and fill the form as needed and confirms his meeting | |
| Triggering Event | Actor Clicking the Invitation Link in the Mail | |
| Actors | Attendee | |
| Stakeholders | Attendee, Organisation, MeetMe Project Team (Project Manager, Developer, UI Designer, System Designer) | |
| Precondition | Actor must get the Invitation for the Meeting | |
| Post-condition | Confirmation of the Attendees Booking | |
| Flow of Activities | Actor | System |
| 1. Clicks the Link  1.1 Enter email  3. Chooses the Available slot  3.1 Click Next | 2. Verifies if the Attendee is in the List  2.1 Prompts to choose the preferred date and time from the dropdown  4. Confirms the Booking  4.1) Sends Booking Confirmation to the Actor |
| Exception Conditions: | 2a) If the Attendee Email is not in the List  1) Display error message  2) Does not prompt to enter the preferred date and the time slot | |

# Architecture/Infrastructure Design:

## 5.1 Architectural Design:

Architecture design of a system is the structural design of a system. Architecture is the overall design of a system and the logical and physical interrelationship between its components. The Architecture of a system illustrates the hardware, software and protocols used in the system. Architecture can also be defined as the framework and a set of guidelines to follow to build a new system. There are many types of architectural systems that can be used in developing a new system, the architecture that has been used in this project MeetMe is three-tier architecture. Three- tier architecture is the framework in which a system is divided into three computing tiers such as Interface layer, Application layer and Data layer.

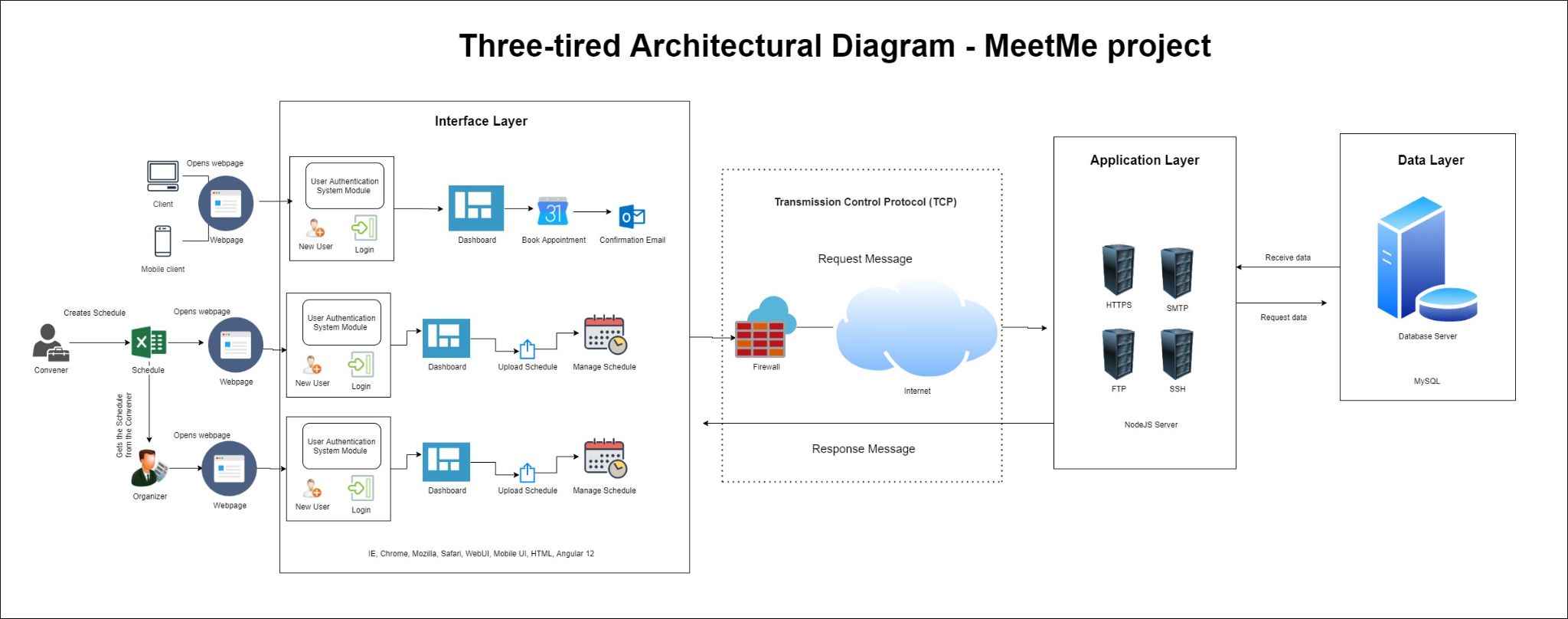


Figure 5.1.1 Three-tiered Architecture Diagram - MeetMe Project

The diagram above is the proposed architecture for the system MeetMe, this architecture is divided into three layers such as Interface layer, Application layer and Data layer. In the Interface layer in the data link layer in the system, this layer provides the physical interface between the client system and application server. In the interface layer the webpage produces the website information to the users such as the user interface, system functionality, dashboard, authentication module and the appointment booking page. The Interface layer is based on Angular 12. Application layer is responsible for the communication between the data layer and interface layer. In the application layer it uses transmission control protocol to communicate between the server and the webpage. The webpage that is present in the interface layer will request for http page, images, file, and UI from the server in application, the information is transmitted through the internet. The application layer is based on the NodeJS server. Data layer is the layer in which the database is located, this database consists of secure data and information about the meeting, client, and convener details. The system MeetMe works by data transmitting and communicating through the above-mentioned layers.

## 5.2 Software components:

Software components are the essential tools which are required for the system to function and sustain throughout. The software components together with the hardware components allows the developers to create the system. The list of software components is listed below.

|  |  |  |
| --- | --- | --- |
| # | Software Components | Description |
| 1 | Network and Internet Services | Network and internet services provide a vital tool in functioning of the system as the system is a web-based application which requires the internet to function. For communication between different layers certain protocols are needed such as:  1. TCP/IP (Transmission Control Protocol/ Internet Protocol)  2. DHCP (Dynamic Host Configuration Protocol)  3. DNS (Domain Name System)  4. HTTP (HyperText Transfer Protocol)  5. SMTP (Simple Mail Transfer Protocol)  6. FTP (File Transfer Protocol)  7. SSH (Secure Shell Protocol) |
| 2 | Web browser | Web browser is the application in which it can render html pages via http protocol to be displayed to the user. The common web browsers are:  1. Google Chrome  2. Mozilla Firefox  3. Microsoft Edge  4. Safari  5. UC browser  6. Opera  The above-mentioned web browser is available in both desktop client and mobile client. |
| 3 | User Interface (UI) | User interface (UI) is the point of human-computer interaction in communicating between the website in ease. The MeetMe project is based on Graphical-user interface (GUI) and menu-driven user interface. |
| 4 | User Experience (UX) | User experience (UX) is the design in which the UI is developed upon. This UX makes the end user use the system in ease and confidently. |
| 5 | Web Server | Web server is the computer software that has been developed on a hardware that will accept the request via Http and it sends webpages to the requested devices through TCP/IP. |
| 6 | Database Management System (DBMS) | Database is the collection of data and information that is stored securely in a server. DBMS is the software that interacts with the application and the end user. |
| 7 | Putty | Putty is an implementation of SSH, which is a terminal emulator to access the server, virtual machine and maintain the website. |
| 8 | Cisco- Any connect | It a VPN that is provided by Murdoch to facilitate the project members in accessing the Virtual machine |
| 9 | Html editor | To create the html webpage, the tools that will be used in this project to code and test the system are  1. Notepad++  2. Sublime Text 3  3. Visual studio code |

Figure 5.2.1 Software Components Table

## 5.3 Infrastructural Design:

Infrastructure design in Information technology is a combination of components such as hardware, software, network, and facilities form the foundation, that are used to develop, monitor, and troubleshoot an Information Technology service. The Infrastructure that has been followed in this system is Platform-as-a-service (PaaS). In PaaS where the hardware and the application server are managed by a third party and we manage the website and database.

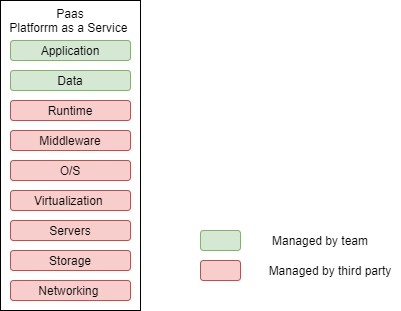


Figure 5.3.1 Infrastructural Design in Platform as a service

From the above diagram we can depict that the MeetMe website is designed on Platform as a service infrastructure. In this infrastructure the development/ project team handles the coding, testing, implementation, monitoring and controlling of the webpage and the database. The third party manages the rest of the components such as Runtime, Middleware, O/S, Virtualization, Servers, Storage and Networking for the system.

## 5.4 Infrastructure requirements:

Infrastructure requirements are the essential components of the system for it to be functional and to be usable to the end user.

|  |  |  |
| --- | --- | --- |
| # | Infrastructure | Description |
| 1 | Capacity | Capacity of the system is the most crucial part of this project as this capacity deals with the amount of traffic to the system. Since it is a web page the capacity plays a part for the web page to be used in a large sector. Website capacity is the number of visitors that can support with each visitor performing representative tasks (such as booking, signing in, and browsing)   1. Data volumes: 30 -40 Gb of space are allocated to Database for this system 2. Approximate number of users at one time: |
| 2 | Performance | Performance of the website is the speed and response time to the user when the user clicks any links. The ideal performance level this project is trying to achieve is Google insight speed of 90+. |
| 3 | Integration and compatibility | The MeetMe project is compatible with any device which has a web-browser with stable internet connection. As this system is being developed in 2020 old browsers such as IE will have trouble in viewing all the content. The MeetMe website will be accessible through desktop client and mobile client. |
| 4 | Platform strategy | The MeetMe project is based on a webpage-based application thus this system runs on a website. |
| 5 | Security | Security is taken into consideration when development of this website, thus several security features are implemented in this system manually or readily available, they are as follows :  1. Passwords will be encrypted and stored in a secure vault of separate database  2. In built firewall of the client devices  3. HTTP  4. Intrusion detection system  5. Secure socket layers  6. SMTP  7. Secure database with two-factor authentication |
| 6 | Back-up & recovery | Backup and recovery of data are important in any system that handles small to large amounts of data to preserve in case of a disaster. For this system regular duplication of data into a secure database will be the backup and recovery in case of disaster or failure in the system. |
| 7 | Scalability | Scalability is the term used to determine how well a website can cope up with increased number of usages. For this website if the number of users are increasing and servers couldn’t cope up with it the solution would be to have multiple servers and having a load balancer from the website where it directs the users to the different servers which host the same information. |
| 8 | Future proofing | Future proofing a website is hard to achieve because the growth of the IT sector is phenomenal, and it will be quite challenging for this system to cope with the far future in 10-20 years’ time. The MeetMe website will be able to sustain up to 5 years if the system is properly maintained, website patching and bug fixtures will increase the lifespan of the website.  The next option would be to host the website by cloud hosting, if the storage space goes lower, we could always purchase additional storage to cater to the users. |

Figure 5.4.1 Interface Requirements Table

## 5.5 Network Topology:

Network topology is the arrangement of the hardware to form a communication network. Network topology is an important factor in the infrastructure of a system to determine the networking workflow of the system.

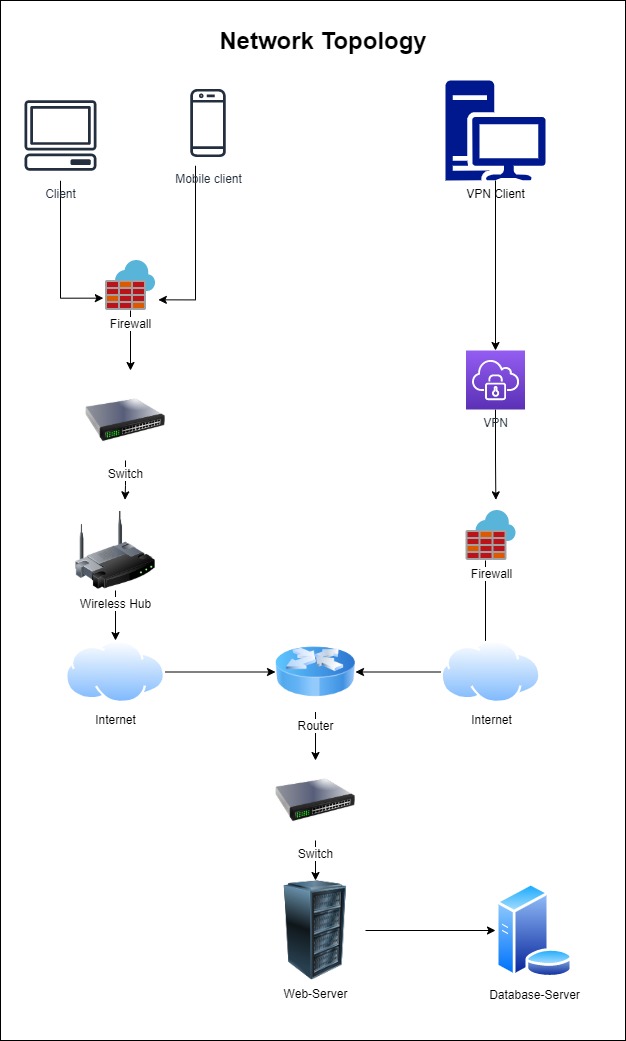


Figure 5.5.1 Network Topology Diagram

In this project MeetMe we are following a central network topology where the web server is the central system which will distribute to different devices. In this network topology if a device fails such as the client or the admin device fails it doesn’t affect the entire system as the problem is isolated in the device itself, but however if the server crashes or fails the entire network of devices are affected and unable to access the webpage. Centralized network topology is the preferred network system for web-based application

# Interface Design:

Based on and by summarizing Jakob Nielsens 10 usability heuristics, Ben shneidermans eight golden rules of interface design and Bruce Tognazzinis principles of interaction design, the following usability rules can be extracted.

* Place users in control of the interface
* Make interactions comfortable
* Reduced cognitive load
* Consistent user interface

## Placing users in control of the interface

To ensure users feel in control of the interface,

* Most actions will be reversible as shown by the undo last action and reset page action available in the modular pages as shown below.

This lets users undo a mistake, or start the main action that they are currently involved in (eg: restart new schedule process)

* This has been done carefully to ensure that it does not increase the cognitive load.
* The interface’s navigation is clear, with visible and discernable icons.
* There is also a feedback system which lets users know which step of the procedure they are currently on and how much is left.

## Make interactions comfortable

To ensure that the interactions are comfortable for the users,

* Removed any information that made no discernable difference whether it was present or not.
* Terms and metaphors were selected that users are familiar with, instead of coming up with terms that users may not be familiar with.
* More screen space is allocated to major information or buttons that impact major decisions (Fitt’s law)(eg next step). This is made to ensure that they are not oversized either.

## Reduce cognitive load

In order to reduce the cognitive load on the user

* Modular pages are introduced as below. The main page remains the same, with the menu on the side.
* With the modular page having all the information regarding the task the user is performing.
* The modular page is simply a container for the information relating to the task such as follows.
* This separation of information with the container lets the user focus better, while still showing considerable information without increasing the cognitive load. (scheduling systems require a lot of information on the screen)

## Consistent user interface

In order to ensure that the user interface is consistent,

* Only the modular page changes when the user is performing different tasks. This ensures that the information outside the modular page is always consistent, both visually and functionally.
* Positioning of objects in the modular pages will be relative to other objects in the same page (alignment) as well as similar objects in other pages (eg next buttons would be found in the same placement on different pages)

## Color Scheme

The main color present would be white, with grayscale colors used for most of the information.

Shades of blue would be used on occasion to show selection or to provide a visual contrast so that information can be segregated (eg titles)

White

- #FFFFFF

Grayscale colors

* #979DAC
* #F6F6F6
* #E8E8E8
* #333333

Blue shades

* #CAF0F8
* #0077B6

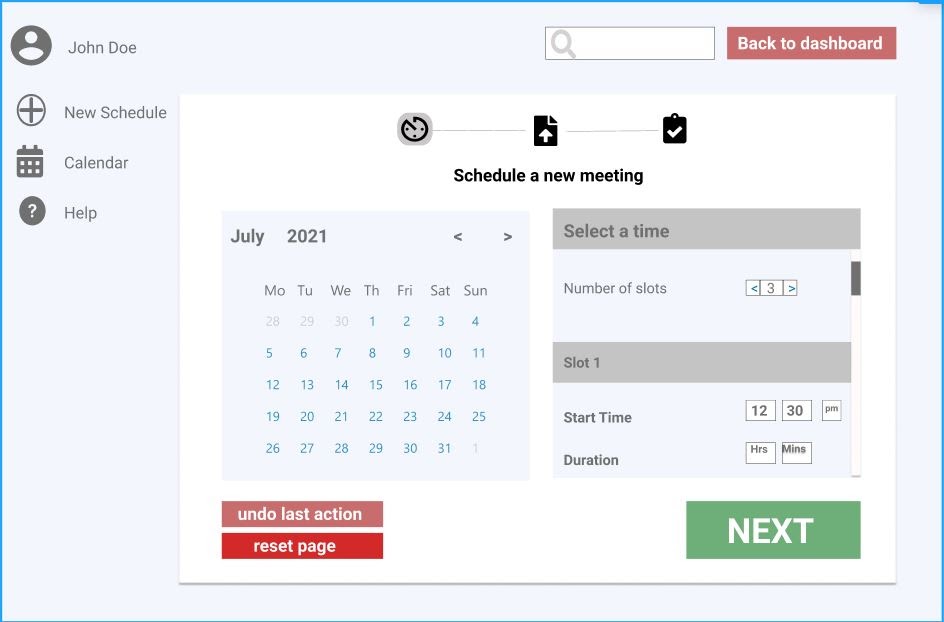


Figure 6.1 Interface Page

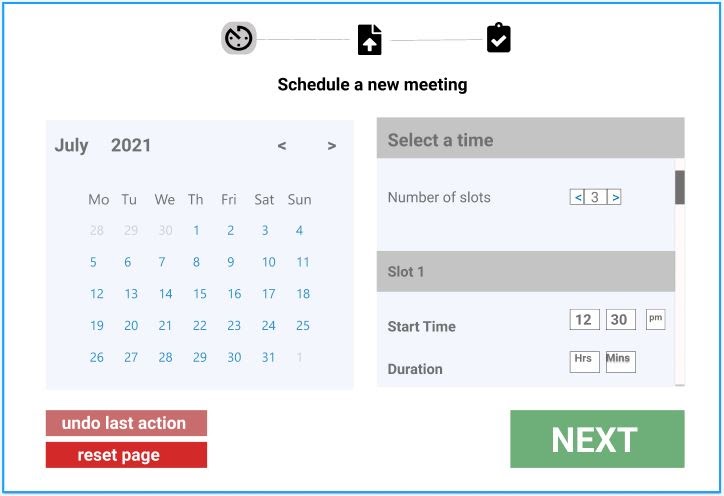


Figure 6.2 Interface (Modular)

# Appendices:

## 7.1 Appendix A: Deliverable task Breakdown Statement

|  |  |
| --- | --- |
| Project Team Name | Team Member |
| Software Comedians | FT01 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Team Member Names** | **Deliverable Percentage Completed** | **Date** | **Signature** |
| Thesath wijayasiri (Team Leader) | 100% | 03/06/2021 |  |
| Kailasam Sasikumar | 100% | 03/06/2021 |  |
| Sai Karthik Miryala | 100% | 03/06/2021 |  |
| Abdul Rahman Suhail | 100% | 03/06/2021 |  |
| Pitchaigani Farjhana Begam | 100% | 03/06/2021 |  |

## 7.2 Appendix B: Data Dictionary

**Organiser:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Description** | **Required?** | **Data Type** | **NULL?** | **Key?** |
| Organiser\_ID | Organiser\_ID is Unique Identifier | Yes | VARCHAR2(50) | NOT NULL | PRIMARY KEY |
| Organiser\_Name | Name of the organiser | Yes | VARCHAR2(50) | NOT NULL |  |
| Password | Password of the organiser | Yes | VARCHAR2(50) | NOT NULL |  |
| Email\_Address | Email Address for the organiser | Yes | VARCHAR2(50) | NOT NULL |  |

Figure 7.2.1 Organiser Table

**OrganiserConvenor:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Description** | **Required?** | **Data Type** | **NULL?** | **Key?** |
| OrganiserConvenor\_ID | Unique OrganiserConvenor ID | Yes | VARCHAR2(25) | NOT NULL | PRIMARY KEY |
| Convenor\_ID | This is Convenor Identification number | Yes | VARCHAR2(50) | NOT NULL | FOREIGN KEY |
| Organiser\_ID | This is Organiser Identification number | Yes | VARCHAR2(50) | NOT NULL | FOREIGN KEY |

Figure 7.2.2 OrganiserConvenor Table

**Convenor:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Description** | **Required?** | **Data Type** | **NULL?** | **Key?** |
| Convenor\_ID | Unique Convenor ID | Yes | VARCHAR2(50) | NOT NULL | PRIMARY KEY |
| Convenor\_firstName | Convenor is first name | Yes | VARCHAR2(50) | NOT NULL |  |
| Convenor\_lastName | Convenor last Name | Yes | VARCHAR2(50) | NOT NULL |  |
| Password | Convenor Password | Yes | VARCHAR2(50) | NOT NULL |  |
| Email\_Address | Email address of the Convenor | Yes | VARCHAR2(50) | NOT NULL |  |

Figure 7.2.3 Convenor Table

**Attendee:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Description** | **Required?** | **Data Type** | **NULL?** | **Key?** |
| Attendee\_ID | Unique Attendee ID | Yes | VARCHAR2(50) | NOT NULL | PRIMARY KEY |
| First\_Name | Attendee first name | Yes | VARCHAR2(50) | NOT NULL |  |
| Last\_Name | Attendee last name | Yes | VARCHAR2(50) | NOT NULL |  |
| Email\_Address | Attendee Email address | Yes | VARCHAR2(50) | NOT NULL |  |
| Mobile\_Number | Attendee Mobile number | Yes | VARCHAR2(10) | NOT NULL |  |

Figure 7.2.4 Attendee Table

**Time\_Booking:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Description** | **Required?** | **Data Type** | **NULL?** | **Key?** |
| TimeBooking\_ID | Unique TimeBooking ID | Yes | VARCHAR2(50) | NOT NULL | PRIMARY KEY |
| Convenor\_ID | Identification for Convenor | Yes | VARCHAR2(50) | NOT NULL | FOREIGN KEY |
| Time\_Slot | Time slot | yes | TIMESTAMP | NOT NULL |  |
| Date\_Slot | Date Slot | yes | TIMESTAMP |  |  |
| Place | Place | yes | VARCHAR2(50) | NOT NULL |  |

Figure 7.2.5 Time\_Booking

**Attendee\_Booking:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Description** | **Required?** | **Data Type** | **NULL?** | **Key?** |
| Attendee\_ID | Unique TimeBooking ID | Yes | VARCHAR2(50) | NOT NULL | PRIMARY KEY |
| OrganiserConvenor\_ID | This is OrganiserConvenor Identification number | Yes | VARCHAR2(25) | NOT NULL | PRIMARY KEY |
| TimeBooking\_ID | This is TimeBooking identification number | Yes | VARCHAR2(50) | NOT NULL | FOREIGN KEY |

Figure 7.2.6 Attendee\_Booking

## 7.3 Appendix C: Glossary of terms and definitions and acronyms

**Data Dictionary:** The Collection of names, attributes, data elements and models. Data Dictionary for metadata in the database.

**CRUD:** A CRUD Matrix with ERD entities on the Left side and use case functions has Top. Each Entities execute has Create, Read, Update, and delete.

**Process Design:** The process Design contain the DFD context diagram, Event Decomposition diagram, use case name list, use case Description

**Architecture/Infrastructure Design:** The Architecture design contains the system architecture, Architecture diagram and Infrastructure Requirements.

**Interface Design:** Interface Design contains the prototype of the website design.