

MeetMe Project Portfolio

(By Hacker Entourage)

Deepika Krishnakumar

Senthamil Sindhu Balasubramanian

Soham Shirgaonkar

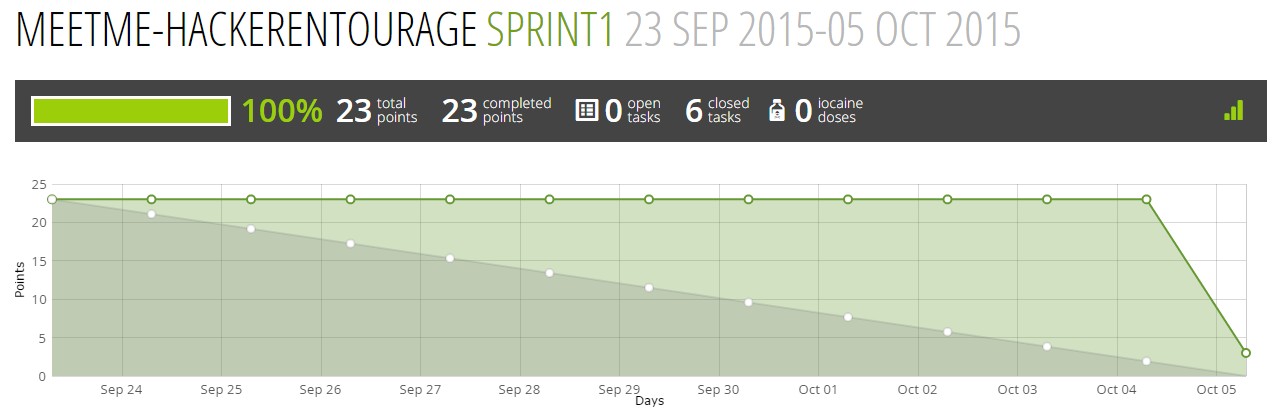
Vidyaabharathi Vasudevan

Vinothkumar Raja

deepika krishna kumar

[COMPANY NAME] [Company address]

**Burndown for Sprint1**



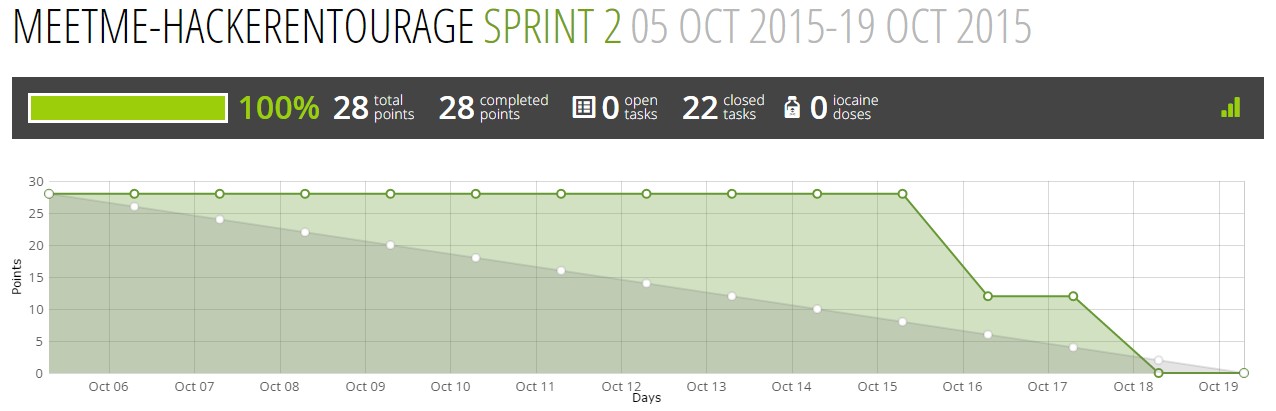
**Completed User Stories**

**1)** User can connect with his Email account

**2)** User can add contacts using their meetme\_id

**3)** User can log out from his profile.

**Burndown for Sprint2**



**Completed User Stories**

**1)** User can connect with his Google account

**2)** User can view his contacts in his profile

**3)** User can only view his own meetme calendar(events)

**4)** User can edit his own meet me calendar to create meeting and add participants

**Requirements Metadata**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement**  **Id** | **Source**  **Id** | **Requirement text** | **Date** | **Effectivity/Release** |
| A | [1](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/1?milestone=36312) | User can connect with his Email  account | 24-Sep | Sprint1 |
| B | [8](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/8?milestone=36312) | User can add contacts using their  meetme\_id | 24-Sep | Sprint1 |
| C | [14](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/14?milestone=36312) | User can log out from his profile. | 24-Sep | Sprint1 |
| D | [6](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/6?milestone=36312) | User can connect with his Google  account | 24-Sep | Sprint2 |
| E | [9](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/9?milestone=38218) | User can view his contacts in his  profile | 24-Sep | Sprint2 |
| E | [12](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/12?milestone=38218) | User can only view his own meetme  calendar(events) | 24-Sep | Sprint2 |
| G | [10](https://tree.taiga.io/project/ser515asu-meetme-hackerentourage/us/10?milestone=38218) | User can edit his own meet me  calendar to create meeting and add participants | 24-Sep | Sprint2 |

**Design documentation**

**1. Introduction**

**1.1 Purpose of this document**

This document describes the software design specifications for MeetMe-HackerEntourage, a meeting planner. It describes the overall system design in terms of user interface, dataflow and external and internal interactions.

**1.2 Scope of the development project**

This system is used for scheduling efficient meetings involving two or more participants based on

synchronizing events from the users’ google calendars and the MeetMe calendar. An algorithm is

used to find a common time based on the participants’ free slots and arrive at a meeting date and

time.

**1.3 Definitions, acronyms, and abbreviations**

ER: Entity Relationship

**1.4 References** <http://standards.ieee.org/findstds/standard/1016-1998.html> https://dzone.com/articles/requirements-metadata <http://html5up.net/>

<https://github.com/bitmazk/django-calendarium/tree/master/calendarium/templates/calendarium>

**1.5 Overview of document**

The chapters and their contents are given in the following table

|  |  |  |
| --- | --- | --- |
| Chapter | Name | Content Description |
| 1 | Introduction | This Chapter itself |
| 2 | System architecture  description | Overview of modules / components  ER Diagram  User interface – Sample screens  Software specifications |
| 3 | Detailed description  of components | Register  Login using MeetMe id Login using Google Adding contacts Creating a meeting  Adding participants to the meeting |
| 4 | Pseudocode for  components |  |
| 5 | Appendices |  |

**2. System architecture description**

**2.1 Overview of modules / components**

MeetMe consists of the following modules.

 Register

o The user can create a MeetMe profile with his email id, password and personal details.

 Login using MeetMe id

o The user can login with his credentials as created above.

 Login using Google

o The user can also login with his google account. Google api plugin of python is used.

 Adding contacts

o User can add other users having a MeetMe profile as their contact.

 Scheduling a meeting

o A user can create a meeting specifying the date, preferred time, duration of meeting and the description of the meeting.

 Adding participants to the meeting.

o The host of the meeting can choose from his contacts to add as participant to the meeting. He can choose the participant to be optional or required.

 Algorithm for picking suitable meeting time

o After the user selects required and optional participants for the meeting, the system checks the schedules of all participants and comes up with a list of possible meeting times

 Login with Facebook

o The user can also login with his Facebook account, the apis provided by Facebook for Django are used.

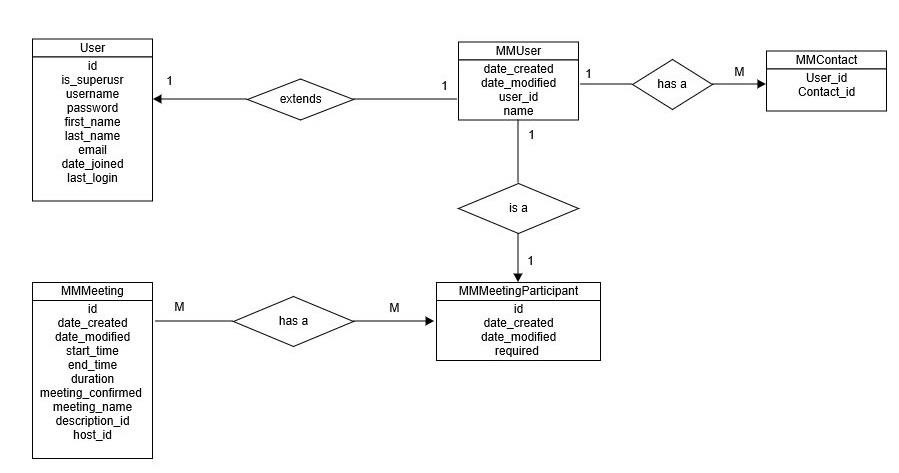
 Editing the MeetMe profile.

o The user can edit his personal details such as name, email, username etc

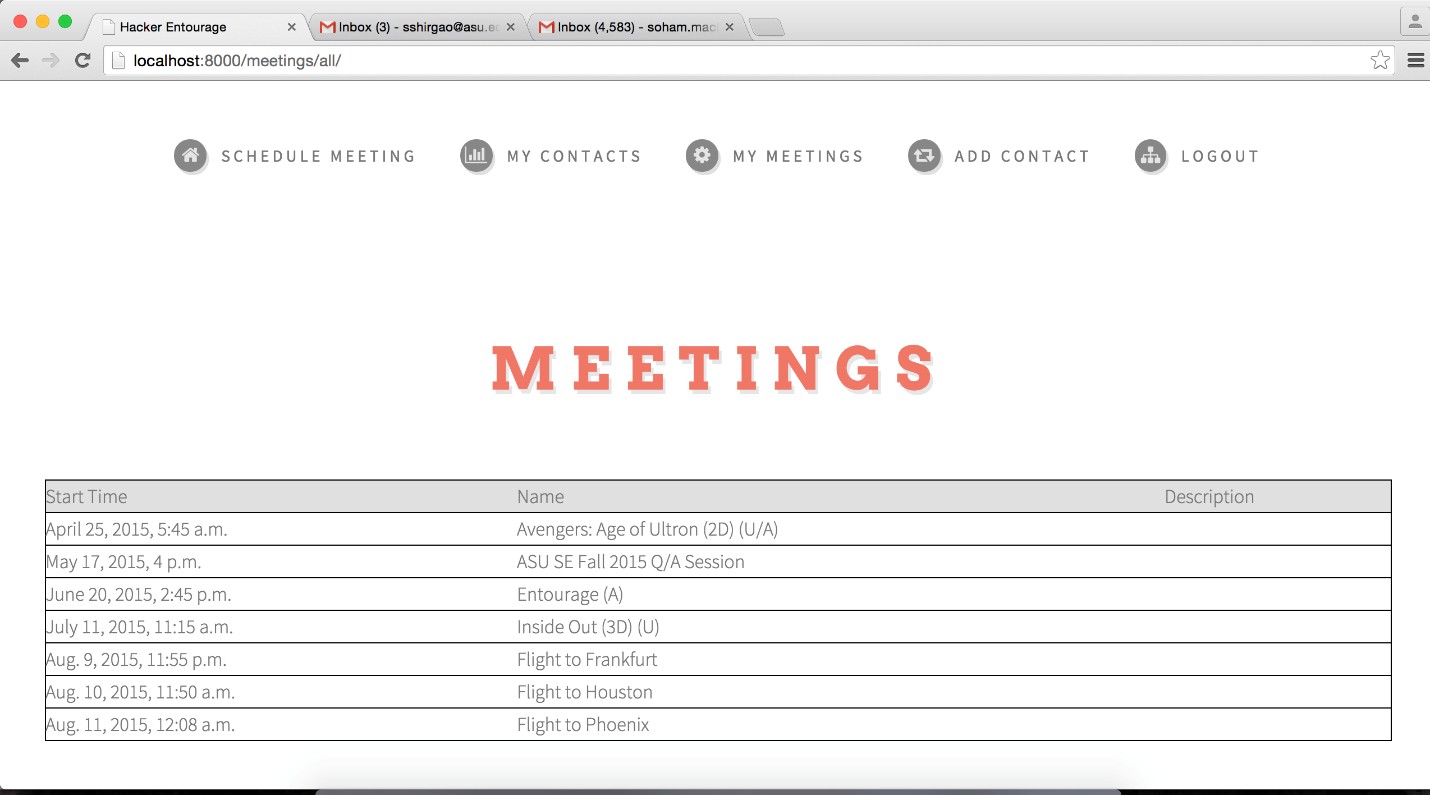
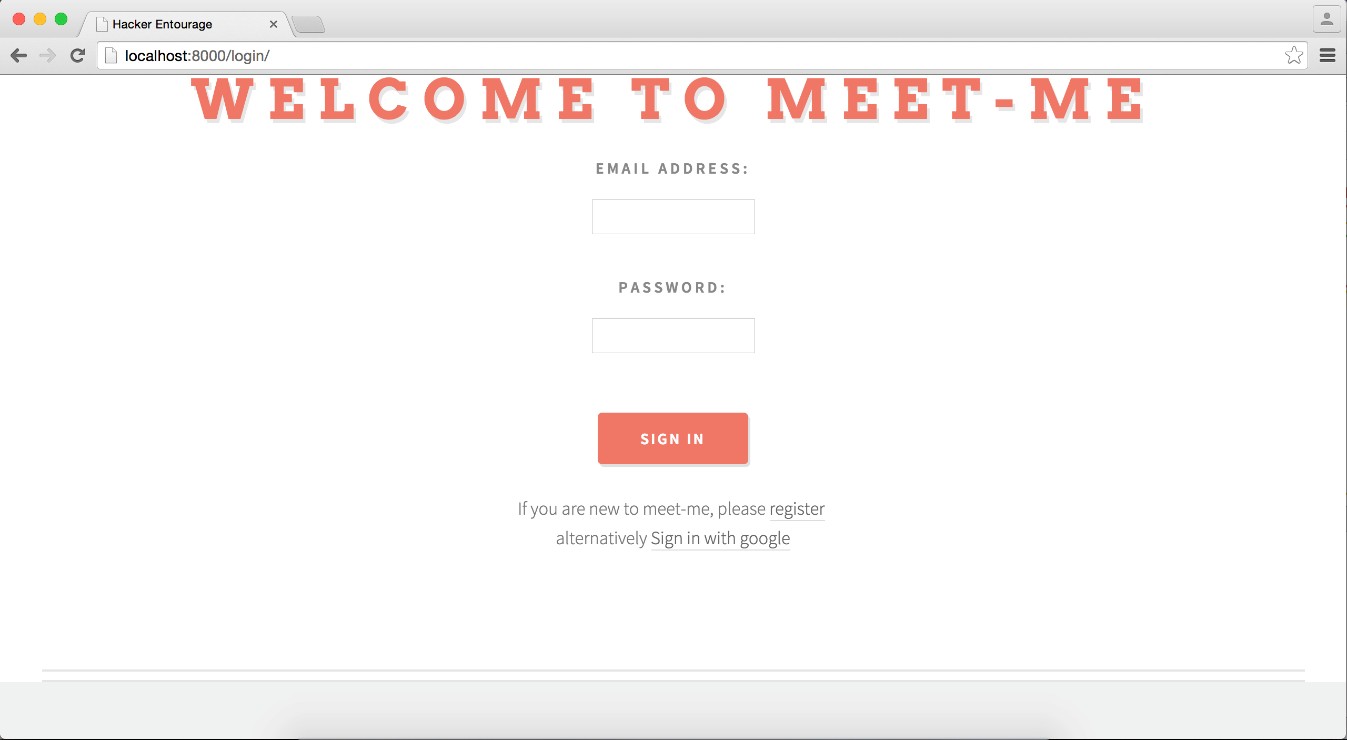
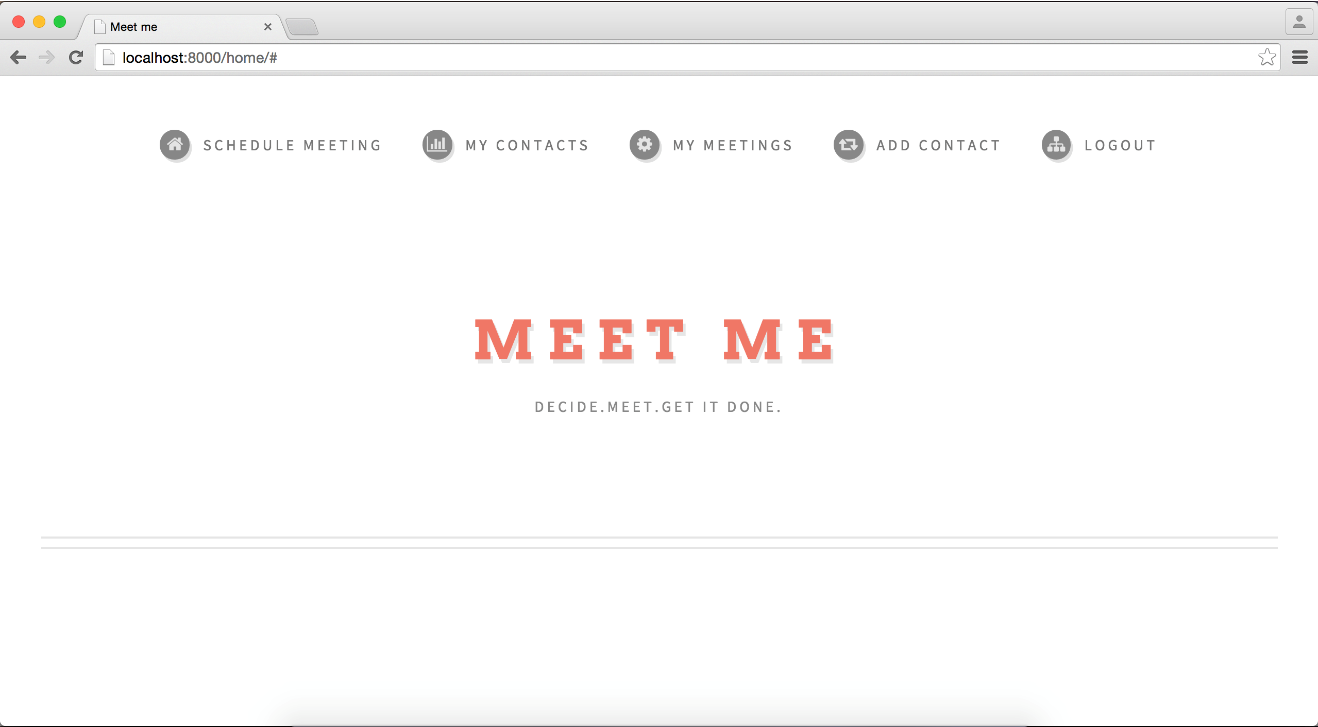
 Displaying the user’s calendar.

o The monthly, weekly and daily meetings of the user are displayed in a calendar view using a calendar app.

**2.2 ER Diagram**



**2.3 User interface – Sample screens**



**2.4 Software specifications**

Programming language: Python

Framework: Django

User Interface: HTML, Javascript

Server: Database: Sqlite

Misc: Celery, RabbitMQ***.***

**3. Detailed description of components**

**3.1 Register**

A model, view and form is created for a User consisting of the fields as specified in the ER diagram. User entered data is saved to the database. We use the base model auth\_user of the framework which consists of basic user profile data, password encryption and authenticating methods,

**3.2 Login using MeetMe id**

The user enters his credentials and we query our database to authenticate. User is redirected to home page on success.

**3.2 Login using Google**

The user can also login to MeetMe using his google account credentials. We make use of Google Api and Oauth to access the required data from the google server. We also pull the required calendar data in the background to sync with our Meeting table in the database. The user’s google account details are saved into the User table of our database.

**3.3 Adding contacts**

A model, view and controller is created for a Contact which stored id of a user and his contact. Both the fields are foreign keys of the User table. The user enters the MeetMe user name of the contact that he wishes to add which is used to query the User table object and add the contact table in our database. It is redirected to an error page if the user tries to add a contact who does not have a MeetMe profile.

**3.4 Creating a meeting**

A model, view and controller is created for a Meeting that consists of the fields specified in the ER diagram. The user can choose the date range in which he wants to schedule the meeting, the duration and the preferred time. These details along with confirmed column as default No is saved to the database. After syncing with events of the other participants, the specific meeting time is arrived at and the confirmed is set to Yes. The user id i.e the host id is set as the logged in user’s id.

**3.5 Adding participants to the meeting**

A model, view and controller is created for a Meeting Participant that consists of the meeting id and participant id. Meeting\_id is a foreign key of the meeting table and the participant\_id is the foreign key with the users table. A drop down is populated with the list of meetings scheduled by the user and another drop down listing his contacts is populated to choose the meeting and its participants respectively. The participant can be optional or required.

**3.6 Algorithm for picking suitable meeting time**

The meeting details of the participants are retrieved from the database. The start and end times of

each confirmed meeting is compared to arrive at a suitable slot. The same is repeated for all the participants of the meeting.

**Test Cases**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Case**  **#** | **Test Title** | **Test**  **Summary** | **Test Steps** | **Test Data** | **Expected**  **Result** | **Status** |
| 1 | Login using google | Login to meetme using google | 1. Open the meetme  URL.  2. Enter the authentication details.  3. Click Login | Correct set of authenticatio n details | User must be successfully directed to his home page | Passed |
| 2 | Login using google | Login to meetme using google | 1. Open the meetme  URL.  2. Enter the authentication details.  3. Click Login | Incorrect username or password | User will be directed to error page | Passed |
| 3 | Enter invalid email address | Entering  invalid characters  in the email field | 1. Open the meetme  URL  2. Enter the email address  3. Click login | Invalid characters(eg  :skipping @  symbol) | Error message will be thrown | Passed |
| 4 | Add contacts | Adding contacts not in the database | 1. Go to the user's  home page.  2. Click on add contacts  3. Enter the contact name | Invalid contact name | Error message will be thrown | Passed |
| 5 | Add participant | Adding participant not present in the database | 1. Go to the user's  home page.  2. Click on add participant  3. Enter the participant name | Invalid participant name | Error message will be thrown | Passed |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 6 | Enter wrong password | Entering  invalid password while signing in | 1. Go to the meetme  URL  2. Enter the authentication details 3. Click Login | Incorrect password | User will be asked to sign in again | Passed |
| 7 | Add existing contacts | Adding existing contacts to the add contacts list | 1. Go to the user's  home page.  2. Click on add contacts  3. Enter the contact name | Enter already existing contact | The contact won't be added to the database | Passed |
| 8 | Add a meeting | Add a meeting for the user | 1. Go to the user's  homepage  2. Click on add a meeting  3. Enter the details of meeting | Leave the time field blank | User gets a warning to include all the necessary fields | Passed |
| 9 | Add a meeting | Add a meeting for the user | 1. Go to the user's  homepage  2. Click on add a meeting  3. Enter the details of meeting | Leave the date field blank | User gets a warning to include all the necessary fields | Passed |
| 10 | Register for a meetme account | Register by providing all the details required | 1. Go to the meetme  URL  2. Enter the required details  3. Click Register | Enter the appropriate number of characters for username | User gets a  warning to include the necessary number of characters (less than 30) | Passed |
| 11 | Register for a meetme account | Register by  providing all the details required | 1. Go to the meetme  URL  2. Enter the required details  3. Click Register | Enter the email address without '@' symbol | User gets a  warning to enter the email address correctly | Passed |
| 12 | Enter valid email and password | Login to the  meetme account using a valid email and password | 1. Go to the meetme  URL  2. Enter the authentication details 3. Click Login | Enter the correct combination of email and password | User gets to login to his meetme account | Passed |
| 13 | Add contacts | Add contacts present in the database | 1. Go to the user's  home page.  2. Click on add contacts  3. Enter the contact name | Enter the correct contact name | User can add his contacts which is reflected in the database | Passed |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 14 | Add a meeting | | Add a meeting for the user | 1. Go to the user's homepage  2. Click on add a meeting  3. Enter the details of meeting | Enter the  starting time and date, ending time and date, duration etc. along with the participants' name | The meeting details will be saved in the database. | Passed |
| 15 | Edit profile | Edit profile to change user details | | 1. Go to user’s home page 2. Click on edit profile 3. Enter the new field values. 4. Click on submit | Enter the username, email id, first name, last name | The profile details will be modified and saved to the database. | Passed. |
| 16 | View Notifications | View notifications of meeting invites/updates | | 1. Click on the notifications menu. 2. Click on notification to accept or decline. | Click on the notifications menu to view the meeting notifications | Whenever a new meeting is scheduled, notifications are received | Passed |
| 17 | View meeting details | View individual meeting details by clicking on the particular meeting name. | | 1. Go to user’s homepage. 2. Click on meetings 🡪 My meetings 🡪 meeting name. 3. The meeting details are displayed. | On clicking the meeting name, the details of that meeting are displayed. | The meeting details are displayed for that particular meeting | Passed |
| 18 | Cancel meeting | Cancel a meeting after its scheduled | | 1. Go to Meetings 🡪 My meetings 2. Select meeting name 3. Click on cancel | Select the meeting to be cancelled. | The particular meeting which is selected will be cancelled and removed from the database. | Passed |
| 19 | Withdraw meeting | Withdraw from a meeting after its scheduled | | 1. Go to Meetings 🡪 My meetings 2. Select meeting name 3. Click on withdraw | Select the meeting to be withdrawn from. | The particular meeting which is selected will be withdrawn from and the participant is removed from the meeting | Passed |
| 20 | Prompt the user with email notifications | Send an email to meetme user for viewing the notifications. | | 1. Login to email. | Meetme users who are a part of the meeting will receive an email notifying them that they have been added to the meeting. | Email Notifications are received when a new meeting is created with the users as participants. | Passed |
| 21 | Accept Email Meeting Invite | User has the ability to accept an invite to a meeting from the notification email. | | 1. Open Notification email. 2. Click on accept | User is redirected to the meetme homepage. | User is successfully redirected to the meetme homepage | Passed |
| 22 | Reject Email Meeting Invite | User has the ability to reject an invite to a meeting from the notification email. | | 1. Open Notification email. 2. Click on reject | User is redirected to the meetme homepage. | User is redirected to the homepage. | Passed |
| 23 | Connect account | User has the ability to connect his account with an existing social account | | 1. Create an account on meetme using email address. 2. Go to profile 3. Click on connect profile 4. Type the username of a valid meetme the account. | User should get an email at the email address linked to the account that he is trying to connect to. | An email is received at the email address linked to the account that the user is trying to connect to. | Passed |
| 24 | Connect account | User has the ability to connect his account with an existing social account | | 1. Create an account on meetme using email address. 2. Go to profile 3. Click on connect profile 4. Type the username of an invalid meetme account. | Error message should be displayed indicating that the account that the user is trying to connect to does not exist. | Error message ‘No such profile exists’ is displayed | Passed |
| 25 | Connect account | User has the ability to approve the request to connect his account with an existing meetme account | | 1. Open connect request email received from meetme. 2. Click on accept. | All the meetings from the account that the user is trying to connect to, should be synced with the users account. | All the meetings from the account that the user is trying to connect to, get synced with his meetme account. | Passed |
|  |  |  | |  |  |  |  |
|  |  |  | |  |  |  |  |
|  |  |  | |  |  |  |  |
|  |  |  | |  |  |  |  |