

## INT\_ELLIGENCE SECOND ITERATION DEMO

G. Chandana Priya – cg3111

Varun Chandra – vc2491

C Sai Srujan – sc4401

Jiayi Wang- jw3704

1. We completed our second iteration demo on Dec 3<sup>rd</sup> (Monday) at 4:40pm. We faced one issue during “IA test-drive” regarding new user creation. We however, figured out and fixed the bug. This was mainly because the server wasn’t loading the right data.

2. The use cases that we demonstrated are as below.

Create Account: As a user, I want to be able to create an account so that I can sync my schedule and use the system. My conditions of satisfaction are that system stores my basic profile information and allows me to set a login password.

Use-Case for Create Account:

Title: Create Account

Actor: User of Meeting Scheduler

Description: Customer creates a new account using valid credentials

Precondition: Application displays the create account page

Basic Flow:

- User enters their first name and last name in the respective fields.
- User enters unique username and valid email address.
- User enters a password of their choice which conforms to the requirement (regex testing) and re-enters it accurately in the confirm password field.
- Upon clicking the create account button, if all the fields are entered correctly, the user account is created and all the details are added to the database.
- The user is now re-directed to the login page.

Alternate Flow:

- If any of the fields are left empty, the application throws an error.
- If the password and confirm password values do not match, an error message is displayed.
- If the username is not unique, the user account creation fails.
- E-mail address needs to match the standard template for email address (regular expression check is performed to enforce this)
- If the password does not satisfy certain pre-defined requirements, an error message is displayed.

Postcondition: User account is created and the details are added to the database.

Sign In: As an authorized user, I want to be able to log-in to my account so that I can access, update my information, find upcoming meetings, schedule new meetings, find best time slot for the meeting and get comprehensive report on things covered. My condition of satisfaction are that the system asks for my login credentials (user name, password) and allows me in on valid entries.

Use-Case for Sign In:

Title: Sign-In

Actor: User of Meeting Scheduler

Description: User logs in to their account using valid credentials

Precondition: Application displays the login page (default page)

Basic Flow:

- User enters their username and password.
- Upon clicking the login button, the user is re-directed to the home page.

Alternate Flow:

- If the user leaves either field empty, the application displays an error message.
- If invalid credentials are entered, the user is prompted to enter the correct credentials.
- If the user is not connected to the internet, an error message will be displayed relaying this information.

Forgot-password: As a registered user, I want to be able to reset my password so that I can I don't have to fear the loss of credentials. My condition of satisfaction is the system should send me link for password reset to my email provided at the time of registration.

Create groups: As event host/ participant, I want to be able to create new groups for people attending same events/meetings and allow sharing of information within that group. My condition of satisfaction is that I should be able to search for people (based on their usernames) and add them to a new group.

Use Case for Create Groups:

Title: Create Groups

Actor: Group admin

Description: User creates a group to later schedule a meeting

Precondition: The user clicks the create group button and is currently in the page which lets them create a group

Basic Flow:

- The user types in the partial username of users that he wants to add to a group.
- A list of users with similar usernames is displayed and the user selects the people he wants in their group.
- After adding all the users the admin wishes to, they click the create group button. At this point, a new group is created with all the selected users and this information is stored in the database.

Alternate Flow:

- If the user attempts to create a group without any other members, an error message is displayed which does not allow the group creation.

Post-Condition: The user is re-directed to the home page which displays all the groups that the user belongs to.

Location (from admin perspective): As an admin, I want to set the location of a particular meeting. My conditions of satisfaction are a seamless interface to select the location.

Use case for Location (admin perspective):

Title: Location (Admin perspective)

Actor: Admin

Description: Admin sets the location of the meeting

Precondition: The user is currently in the meeting tab

Direct Flow:

- The admin clicks the set location button upon which a new browser window opens.
- Google Maps will run in this web browser and the admin can navigate and select the location that he wants.

- The latitude and longitude of the selected location gets stored in the database and gets associated with that particular meeting.

Alternate Flow:

- If there is no internet connection, the admin gets an error message detailing the problem.

Postcondition: The user returns to the meeting tab.

Location (User perspective): As a user, I want to be able to access the location of a particular meeting so as to be prepared for it. My conditions of satisfaction is that the system should show me a map location for the meeting venue.

Use case for location (user perspective):

Title: Location (User Perspective)

Actor: User

Description: User accesses the location of the meeting

Direct Flow:

- User clicks the get location button for the meeting of their choice. This opens up a browser window wherein the Google Maps API clearly displays the location of the meeting.

Alternate Flow:

- If the user is not connected to the internet, an appropriate error message will pop up.

Scheduling (from organiser perspective): As an organiser, I want to be able to schedule a meeting as per the convenience of all the attendees so that everyone is informed beforehand about the meeting. My conditions for satisfaction are being able to: toggle between taking a consensus on a list of locations for the meeting and enforcing a particular location for the meeting, and being able to bypass rules in case of emergency meetings.

Use Case for Scheduling:

Title: Scheduling from User perspective

Actor: Admin

Description: Deciding if consensus or single-point decision

Direct Flow:

- The admin gets to decide if they want the group to vote on the location of the meeting or if the admin gets to enforce the location of the meeting.

Alternate Flow:

- If the admin leaves this field blank, the application will throw an error message.

The improvements made from the first to the second iteration are from the location use-case.

### 3. CI

We used travis CI for continuous integration. We connected our GitHub repo with our travis account. The .travis.yml file contains scripts to run our program and test cases.

For bug finder we used prospector, a python library that integrated tools such as pylint. We used unittest and pytest library for testing. We used coverage.py for the unittest coverage reports.

### 4. GitHub link

[https://github.com/ASE-Int-Elligence/Meeting\\_Scheduler](https://github.com/ASE-Int-Elligence/Meeting_Scheduler)