Test Plan

Cases testing will be conducted in every iteration, once the deployment is done. All the past functionalities should be revisited also, to make sure all the current functionalities are met the least requirements.

1. **Functionality Requirements Analysis**

The whole group is supposed to read through the functionality requirements together during the planning process for the next iteration. To discuss the main test cases and find out any corner cases.

1. **Test Case Creation**

After the meeting, project manager will come up with the test cases based on the group discussion, and the coders would contribute by updating test cases during the coding time. Each test case contains the functionality name, description, input requirement, test procedure, expected results as well as actual results and the final results (pass or fail). To make the functionality accurate and easy to the following project manager to continue, the person in charge is supposed to come up with the test data also. The test cases should be finished before the relevant functionalities are done and the app is deployed on openshift. During the creation time, the details (functionality name, description, input requirement, test procedure, expected results) should be filled.

1. **System Testing**

The project manager will start to test the system and fill in the actual results and final results (pass or fail) cells once the app is deployed. Project manager should return the testing results to the coders. During the testing period, the project manager or coders can still come up some test cases if they missed any previously.

1. **Metrics Updating**

The project manager should update the debug metrics after testing as well as after debugging. To record the bugs need to be solved and the final results for each bug (whether it is solved or not). Coming up the final bug metrics score and action plan also, followed by the planning for the next step according to the action plan. During this period, the project manager would in charge to set the severity level for each bug, meanwhile, the other team member can contribute also if any particular bug needs to be discussed.

**Iteration 2:**

**(Project manager: Aayush)**

Generate test cases and conduct testing for:

* Login/logout (coder: Yuxuan & Eva)
* Bootstrap(unzip a zip file, validation) (coder: Max & Wu Di)

**Iteration 3:**

**(Project manager: Yuxuan)**

Generate test cases and conduct testing for:

* Basic Location Report (break down) (Coder: Aayush & Wu Di)
* Basic Location Report (Top-k popular places) (Coder: Aayush & Wu Di)
* Basic Location Report (Top-k popular next places) (Coder: Aayush & Wu Di)
* Basic Location Report (companions) (Coder: Aayush & Wu Di)
* Heatmap (Coder: Max & Eva)

Review the test cases or add on for:

* Login/logout
* Bootstrap

**Iteration 4:**

Generate test cases and conduct testing for:

**(Project manager: Wu Di)**

* Automatic Group Detection (Coder: Yuxuan & Eva)
* Web Service & JSON ( except for the groups) (Coder: Max & Aayush)
* Administrative Updating Data (Coder: Max & Aayush)

Review the test cases or add on for:

* Login/logout
* Bootstrap(unzip a zip file, validation)
* Basic Location Report (break down)
* Basic Location Report (Top-k popular places)
* Basic Location Report (Top-k popular next places)
* Basic Location Report (companions)
* Heatmap

**Iteration 5:**

**(Project manager: Wu Di)**

Generate test cases and conduct testing for:

* Group Aware Location Report ( Top-k popular places for groups) (Coder: Yuxuan & Aayush)
* Group Aware Location Report ( Top-k next places for groups) (Coder: Yuxuan & Aayush)
* Bootstrap changes ( check for duplicated lines) (Coder: Max & Eva)
* JSON (for groups & updated part for bootstrap) (Coder: Max & Eva)

Review the test cases or add on for:

* Login/logout
* Bootstrap(unzip a zip file, validation)
* Basic Location Report (break down)
* Basic Location Report (Top-k popular places)
* Basic Location Report (Top-k popular next places)
* Basic Location Report (companions)
* Heatmap
* Automatic Group Detection
* Web Service & JSON ( except for the groups)
* Administrative Updating Data

**Iteration 6:**

**(Project manager: Max)**

Generate test cases and conduct testing for:

* The improved UI

In this iteration, need to review all the test cases and keep testing all the functionalities with the JSON checker as well as the UAT cases and data.

**Iteration 7:**

**(Project manager: Eva)**

In this iteration, we need to test all the functionalities including UI.