

System and Unit Test Report - CMPS 115

Planit by Team Rocket

12 - 02 - 2019

Sprint 1 Test

User Stories:

- User story 1 from sprint 1: As a user, I want an interface that allows me to input data into text fields and submit them to be stored, as well as a “generate activity plan” button, so I can easily interact with the application.
- User story 2 from sprint 1: As a user, I want the application to prompt and allow me to enter in activities I enjoy, along with the maximum time, ideal time, and maximum cost of each activity, so that these activities can become the basis of my future activity plans.

Scenario:

start Planit app; select ‘Edit Activities’; type/select

- name = <Test Planit App>
- max time = <60.0 minutes>
- ideal time = <30.0>
- max cost = <0.0>
- activity type = <Misc>
- press +Create Activity button

user should see verification message that activity has been inserted.

Sprint 2 Test

User Stories:

- User story 1 from sprint 2: As a user, I want the application to draw from a saved database of activities, and fill up the given time with activities that fit within the given cost, so that I will have a list of activities I can do.

Scenario:

start Planit app; select 'Edit Activities'. If the table is empty, execute the test above (under Sprint 1 Test) to add a new 'test' activity

1. on the main window under 'Generate a New Activity Plan', select
 - Select amount of free hours = <0.5>
 - Select current budget \$ = <0>
2. Press the 'Generate Plan' button. User should see a window pop up with a new random activity plan that fits the bounds set on the main window (amount of free hours and current budget). If no other activities have been added yet, the plan should only contain 'Test Planit App'.

Sprint 3 Test

User Stories:

- User story 1 from sprint 3: As a user, I want the final list to be saveable, including the cost/time for each activity, as well as the total time and cost, so I can take my plan on the go.

Scenario:

start Planit app; select 'Edit Activities'. If the table is empty, execute the test above (under Sprint 1 Test) to add a new 'test' activity

1. on the main window under 'Generate a New Activity Plan', select
 - Select amount of free hours = <0.5>
 - Select current budget \$ = <0>
2. Press the 'Generate Plan' button. User should see a window pop up with a new random activity plan
3. In the Plan Generator window, press the 'Save Plan' button and choose a directory to save your plan as a pdf.
4. Navigate to the directory you saved the pdf in and open the pdf. Verify that the contents match the plan generated in the Plan Generator window.

Sprint 4 Test

Developer Stories:

- As a Developer, I want to make sure the code always runs without errors.
- I want to easily be able to add or modify my tests to make refactoring more effective.

- Acceptance criteria: Unit code is set up to test the entire project, and project runs without errors.

Scenario:

Open terminal, navigate to the planit directory where pom.xml is located,

1. To test the unit code, use the command `mvn -Dtest=TestAll test`
2. The terminal should show that all tests have been successfully run and there are no errors or failures.