

# Functional Testing - CMPS 115

Planit by Team Rocket

12 - 02 - 2019

The following are the test classes used to automate the testing process using maven and junit suite.

## Test Classes - ActivityTest.java, ActivityPlan.java, TestAll.java

### ActivityTest.java

- ActivityTest.java tests the Activity class by initializing a new Activity then testing the methods associated with the class
  - Tests include:
    - Testing getName() function - sets activity name to 'test', then verifies that getName() returns 'test'
    - Testing getActivityType() function - sets activity type to 'test', then verifies that getActivityType() returns 'test'
    - Testing getMaxTime function - sets max time to 50, then verifies that getMaxTime() returns 50
    - Testing getIdealTime() function - sets ideal time to 30, then verifies that getIdealTime() returns 30
    - Testing getActualTime() function - sets actual time to 40, then verifies that getActualTime() returns 40
    - Testing getMaxCost() - sets max cost to 100, then verifies that getMaxCost() returns 100
    - Testing compareTo() function - copies activity to a temporary holder, then compares the copy to the original initialized activity

### ActivityPlanTest.java

- ActivityPlanTest.java tests the ActivityPlan class by initializing a new Activity list then testing the methods associated with generating a new plan. This test class initializes variables 'availableMoney' and 'availableCost' to use when generating. Also initialized is an array list of Activities 'myActivityList' to hold a generated plan.
  - Tests include:
    - Testing generatePlan() function - calls the generatePlan() function, which returns a boolean 'generated' set to true if a plan has been successfully

generated. myActivityList holds the generated plan. The test verifies that 'generated' is true.

- Testing that plan fits boundaries - cycles through the plan generated, keeping track of the total time and total cost of the plan, then verifies that the total time and total cost are less than the available time and available cost respectively

#### TestAll.java

- This class combines all tests so that the tests can be run all together