

# NASA EVA Gamification

## TEST SUITE - UNIT TESTING

---

**PHASE II**



**Prepared By**  
Laura Addiego  
Samia Alam  
Kelli Corey  
Charles Milk  
Adeola Odusola  
Hung Pham

## Table of Contents

<b>Introduction</b>	<b>3</b>
<b>1.0 Test Suite U01: Code and Data Coverage Tests for Point Class</b>	<b>3</b>
1.1 Test Case TC_U01.1: Test Function GetPointsGenerated()	3
1.2 Test Case TC_U01.2: Test Function SetPointsGenerated(\$point_generated)	3
1.3 Test Case TC_U01.3: Test Function SetPointsGenerated(\$point_generated)	4
1.4 Test Case TC_U01.4: Test Function SetPointsGenerated(\$point_generated)	4
1.5 Test Case TC_U01.5: Test Function SetPointsGenerated(\$point_generated)	4
1.6 Test Case TC_U01.6: Test Function SetPointsGenerated(\$point_generated)	5
1.7 Test Case TC_U01.7: Test Function GetTotalPointsForAllUsers()	5
1.8 Test Case TC_U01.8: Test Function GetTotalPointsForAllUsers() when Null	5
1.9 Test Case TC_U01.9: Test Function GetTotalPointsFromUserId(\$id)	6
1.10 Test Case TC_U01.10: Test Function GetTotalPointsFromUserId(\$id)	6
1.11 Test Case TC_U01.11: Test Function GetTotalPointsFromUserId(\$id)	6
1.12 Test Case TC_U01.12: Test Function GetTotalPointsFromUserId(\$id) when User Id Not Found	7
1.13 Test Case TC_U01.13: Test Function getEditedPostFromUserId(\$id)	7
1.14 Test Case TC_U01.14: Test Function getEditedPostFromUserId(\$id) when Null	7
1.15 Test Case TC_U01.15: Test Function getEditedPostFromUserId(\$id) when Null	8
1.16 Test Case TC_U01.16: Test Function getEditedPostFromUserId(\$id) when Null	8
1.17 Test Case TC_U01.17: Test Function construct(\$page_is_new)	8
1.18 Test Case TC_U01.18: Test Function construct(\$page_is_new)	9
1.19 Test Case TC_U01.19: Test Function construct(\$page_is_new)	9
1.20 Test Case TC_U01.20: Test Function SetPointsGenerated(\$point_generated) with 0	9
1.21 Test Case TC_U01.21: Test Function SetPointsGenerated(\$point_generated) with Negative value	10
1.22 Test Case TC_U01.22: Test Function SetPointsGenerated(\$point_generated) with Null value	10
<b>2.0 Test Suite U02: Code and Data Coverage Tests for LeaderBoard Class</b>	<b>10</b>
2.1 Test Case TC_U02.1: Test Function displayAddedPages(\$clickedUser)	11
2.2 Test Case TC_U02.2: Test Function displayEditedPages(\$clickedUser)	11
2.3 Test Case TC_U02.3: Test Function getBadges(\$totalPoints)	11
2.4 Test Case TC_U02.4: Test Function displayLeaderBoard(\$userId,\$userPoints)	12

Version Number	Description of Change	Author	Date
1.0	Initial Creation of Document	Samia Alam	7/8/18
1.1	Updated leaderboard class test suite	Adeola Odusola	7/21/18

# Introduction

The purpose of this document is to describe the unit tests used to test the Point class, and Leaderboard class developed during phase 2 of NASA Gamification project. All methods of the classes will be tested in such a way that each line of code gets tested to meet 100% code coverage. Data type conditions such as positive and negative values, bad data conditions such as illegal data value and empty options will be tested to cover reasonable data coverage. Effective unit testing naming conventions will be used by including the name of the method being tested, adding the word test in the beginning, and adding the expectation of the test case outcome in the name. Each method will be tested through separate test cases, and there will be a single assert in each method. This will allow each test case to be simple, independent, and identify defects in each method and each line of code.

## 1.0 Test Suite U01: Code and Data Coverage Tests for Point Class

Following Test Case should be executed to test each line of code in Point Class.

### 1.1 Test Case TC\_U01.1: Test Function GetPointsGenerated()

**Test Case ID:** TC\_U01.1

**Test Case Name:** TestFunctionGetPointsGeneratedReturnsPointGenerated()

**Method being tested:** GetPointsGenerated()

**Short Description:** Test that the function GetPointsGenerated() returns the generated points attribute

**Input Data to method being tested:** None

**Expected Results:** \$mPointsGenerated attribute is returned

**Tester's Name:** Hung Pham

### 1.2 Test Case TC\_U01.2: Test Function SetPointsGenerated(\$point\_generated)

**Test Case ID:** TC\_U01.2

**Test Case Name:** TestFunctionSet50PointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be 50

**Expected Results:** Return value should be 50.

**Tester's Name:** Hung Pham

### 1.3 Test Case TC\_U01.3: Test Function SetPointsGenerated(\$point\_generated)

**Test Case ID:** TC\_U01.3

**Test Case Name:** TestFunctionSet100PointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be 100

**Expected Results:** Return value should be 100.

**Tester's Name:** Hung Pham

### 1.4 Test Case TC\_U01.4: Test Function SetPointsGenerated(\$point\_generated)

**Test Case ID:** TC\_U01.4

**Test Case Name:** TestFunctionSetZeroPointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be 0

**Expected Results:** Throw an error exception ("Invalid value") .

**Tester's Name:** Hung Pham

### 1.5 Test Case TC\_U01.5: Test Function SetPointsGenerated(\$point\_generated)

**Test Case ID:** TC\_U01.5

**Test Case Name:** TestFunctionSetNegativePointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be -1

**Expected Results:** Throw an error exception ("Invalid value") .

**Tester's Name:** Hung Pham

## 1.6 Test Case TC\_U01.6: Test Function SetPointsGenerated(\$point\_generated)

**Test Case ID:** TC\_U01.6

**Test Case Name:** TestFunctionSetInvalidPointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be 500

**Expected Results:** Throw an error exception ("Invalid value") .

**Tester's Name:** Hung Pham

## 1.7 Test Case TC\_U01.7:Test Function GetTotalPointsForAllUsers()

**Test Case ID:** TC\_U01.7

**Test Case Name:** TestFunctionGetTotalPointsForAllUsersReturnTotalPointsFromAllusers()

**Method being tested:** getTotalPointsForAllUsers()

**Short Description:** Test that the function getTotalPointsForAllUsers() can return total points calculation for all users in the system

**Input Data to method being tested:** None

**Expected Results:** Total points from the revision table is returned.

**Tester's Name:** Hung Pham

## 1.8 Test Case TC\_U01.8:Test Function GetTotalPointsForAllUsers() when Null

**Test Case ID:** TC\_U01.8

**Test Case Name:** TestFunctionGetTotalPointsForAllUsersReturnsNullWhenUserNotFound()

**Method being tested:** getTotalPointsForAllUsers()

**Short Description:** Test that the function getTotalPointsForAllUsers() can return null when no such user is found

**Input Data to method being tested:** None

**Expected Results:** Null is returned when there is no user with points.

**Tester's Name:** Hung Pham

## 1.9 Test Case TC\_U01.9:Test Function GetTotalPointsFromUserId(\$id)

**Test Case ID:** TC\_U01.9

**Test Case Name:** TestFunctionGetTotalPointsFromUserIdReturnsPointsForUserId()

**Method being tested:** getTotalPointsFromUserId(\$id)

**Short Description:** Test that the function getTotalPointsFromUserId(\$id) can return total points for selected user Id

**Input Data to method being tested:** Create instance of Point class; get a user Id with 50 points from revision table. Insert the ID in the getTotalPointsFromUserId(\$id)

**Expected Results:** returned points for the given ID is 50.

**Tester's Name:** Hung Pham

## 1.10 Test Case TC\_U01.10:Test Function GetTotalPointsFromUserId(\$id)

**Test Case ID:** TC\_U01.10

**Test Case Name:** TestFunctionGetTotalPointsFromZeroUserIdReturnsPointsForUserId()

**Method being tested:** getTotalPointsFromUserId(\$id)

**Short Description:** Test that the function getTotalPointsFromUserId(\$id) can return total points for selected user Id

**Input Data to method being tested:** Create instance of Point class; pass 0 to user ID. Insert the ID in the getTotalPointsFromUserId(\$id)

**Expected Results:** Throw an error exception ("Invalid user ID") .

**Tester's Name:** Hung Pham

## 1.11 Test Case TC\_U01.11:Test Function GetTotalPointsFromUserId(\$id)

**Test Case ID:** TC\_U01.11

**Test Case Name:** TestFunctionGetTotalPointsFromNegativeUserIdReturnsPointsForUserId()

**Method being tested:** getTotalPointsFromUserId(\$id)

**Short Description:** Test that the function getTotalPointsFromUserId(\$id) can return total points for selected user Id

**Input Data to method being tested:** Create instance of Point class; pass -1 to user ID. Insert the ID in the getTotalPointsFromUserId(\$id)

**Expected Results:** Throw an error exception ("Invalid user ID") .

**Tester's Name:** Hung Pham

## 1.12 Test Case TC\_U01.12:Test Function

### GetTotalPointsFromUserId(\$id) when User Id Not Found

**Test Case ID:** TC\_U01.12

**Test Case Name:** TestFunctionGetTotalPointsFromUserIdReturnsNullWhenUserIdNotFound()

**Method being tested:** getTotalPointsFromUserId(\$id)

**Short Description:** Test that the function getTotalPointsFromUserId(\$id) can return Null when user id does not exist

**Input Data to method being tested:** Create instance of Point class; pass null to user ID. Insert the user ID in the getTotalPointsFromUserId(\$id)

**Expected Results:** Null is returned.

**Tester's Name:** Hung Pham

## 1.13 Test Case TC\_U01.13:Test Function

### getEditedPostFromUserId(\$id)

**Test Case ID:** TC\_U01.13

**Test Case Name:** TestFunctionGetEditedPostFromUserIdReturnsEditedPost()

**Method being tested:** getEditedPostFromUserId(\$id)

**Short Description:** Test that the function getEditedPostFromUserId(\$id) can return edited posts for selected user Id

**Input Data to method being tested:** Create instance of Point class; Insert a user ID with edited post of 2 from revision table in the getEditedPostFromUserId(\$id)

**Expected Results:** returned edited post count is 2 .

**Tester's Name:** Hung Pham

## 1.14 Test Case TC\_U01.14:Test Function

### getEditedPostFromUserId(\$id) when Null

**Test Case ID:** TC\_U01.14

**Test Case Name:** TestFunctionGetEditedPostFromUserIdReturnsNullWhenUserIdNotFound()

**Method being tested:** getEditedPostFromUserId(\$id)

**Short Description:** Test that the function getEditedPostFromUserId(\$id) can return Null when user id does not exist

**Input Data to method being tested:** Create instance of Point class; Insert a user ID that does not exist in the revision table in function getEditedPostFromUserId(\$id)

**Expected Results:** Null is returned when User Id is not found.

**Tester's Name:** Hung Pham

## 1.15 Test Case TC\_U01.15:Test Function getEditedPostFromUserId(\$id) when Null

**Test Case ID:** TC\_U01.15

**Test Case Name:** TestFunctionGetEditedPostFromZeroUserIdReturnsEditedPost()

**Method being tested:** getEditedPostFromUserId(\$id)

**Short Description:** Test that the function getEditedPostFromUserId(\$id) will throw an error exception when user id is 0.

**Input Data to method being tested:** Create instance of Point class; pass 0 to user ID. Insert that user ID in function getEditedPostFromUserId(\$id)

**Expected Results:** Throw an error exception ("Invalid user ID") .

**Tester's Name:** Hung Pham

## 1.16 Test Case TC\_U01.16:Test Function getEditedPostFromUserId(\$id) when Null

**Test Case ID:** TC\_U01.16

**Test Case Name:** TestFunctionGetEditedPostFromNegativeUserIdReturnsEditedPost()

**Method being tested:** getEditedPostFromUserId(\$id)

**Short Description:** Test that the function getEditedPostFromUserId(\$id) will throw an error exception when user id is negative.

**Input Data to method being tested:** Create instance of Point class; pass -1 to user ID. Insert that user ID in function getEditedPostFromUserId(\$id)

**Expected Results:** Throw an error exception ("Invalid user ID") .

**Tester's Name:** Hung Pham

## 1.17 Test Case TC\_U01.17: Test Function construct(\$page\_is\_new)

**Test Case ID:** TC\_U01.17



**Test Case Name:** TestFunctionConstructCreates50PointsInstance()

**Method being tested:** construct(\$page\_is\_new)

**Short Description:** Test that when the function construct(\$page\_is\_new) is called, instance of point call is generated.

**Input Data to method being tested:** Set \$page\_is\_new = 0

**Expected Results:** 50 Points is returned.

**Tester's Name:** Hung Pham

## 1.18 Test Case TC\_U01.18: Test Function construct(\$page\_is\_new)

**Test Case ID:** TC\_U01.18

**Test Case Name:** TestFunctionConstructCreates100PointsInstance()

**Method being tested:** construct(\$page\_is\_new)

**Short Description:** Test that when the function construct(\$page\_is\_new) is called, instance of point call is generated.

**Input Data to method being tested:** Set \$page\_is\_new = 1

**Expected Results:** 100 Points is returned.

**Tester's Name:** Hung Pham

## 1.19 Test Case TC\_U01.19:Test Function construct(\$page\_is\_new)

**Test Case ID:** TC\_U01.19

**Test Case Name:** TestFunctionConstructCreatesNegativePointsInstance()

**Method being tested:** construct(\$page\_is\_new)

**Short Description:** Test that when the function construct(\$page\_is\_new) is called, instance of point call is generated.

**Input Data to method being tested:** Set \$page\_is\_new = -1

**Expected Results:** Throw an error exception "Invalid value"

**Tester's Name:** Hung Pham

## 1.20 Test Case TC\_U01.20: Test Function SetPointsGenerated(\$point\_generated) with 0

**Test Case ID:** TC\_U01.20

**Test Case Name:** TestFunctionSetPointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute of 0

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be 0

**Expected Results:** Throw an error exception ("Invalid value") .

**Tester's Name:** Hung Pham

## 1.21 Test Case TC\_U01.21: Test Function SetPointsGenerated(\$point\_generated) with Negative value

**Test Case ID:** TC\_U01.21

**Test Case Name:** TestFunctionSetPointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute of negative value

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be -50

**Expected Results:** Throw an error exception ("Invalid value") .

**Tester's Name:** Hung Pham

## 1.22 Test Case TC\_U01.22: Test Function SetPointsGenerated(\$point\_generated) with Null value

**Test Case ID:** TC\_U01.21

**Test Case Name:** TestFunctionSetNullPointsGeneratedReturnsGivenPoints()

**Method being tested:** SetPointsGenerated(\$point\_generated)

**Short Description:** Test that the function SetPointsGenerated() can set the accurate points attribute of negative value

**Input Data to method being tested:** Create instance of Point class; Set \$point\_generated parameter to be null

**Expected Results:** Return null value

**Tester's Name:** Hung Pham

## 2.0 Test Suite U02: Code and Data Coverage Tests for LeaderBoard Class

Following Test Case should be executed to test each line of code in LeaderBoard Class.

## 2.1 Test Case TC\_U02.1: Test Function displayAddedPages(\$clickedUser)

**Test Case ID:** TC\_U02.1

**Test Case Name:** TestFunctionDisplayAddedPagesReturnsAddedPagesList()

**Method being tested:** displayAddedPages(\$clickedUser)

**Short Description:** Test that the function displayAddedPages(\$clickedUser) can display a list of added pages by selected user

**Input Data to method being tested:** Create an instance of LeaderBoard class; Select or set a user with 2 added posts. provide the valid value for \$clickedUser in the displayAddedPages(\$clickedUser) function

**Expected Results:** list of the added posts is returned.

**Tester's Name:** Adeola Odusola

## 2.2 Test Case TC\_U02.2: Test Function displayEditedPages(\$clickedUser)

**Test Case ID:** TC\_U02.2

**Test Case Name:** TestFunctionDisplayEditedPagesReturnsEditedPagesList()

**Method being tested:** displayEditedPages(\$clickedUser)

**Short Description:** Test that the function displayEditedPages(\$clickedUser) can display a list of added pages by selected user

**Input Data to method being tested:** Create an instance of LeaderBoard class; Select or set a user with 2 edited posts. provide the valid value for \$clickedUser in the displayEditedPages(\$clickedUser) function

**Expected Results:** List of the edited posts is returned.

**Tester's Name:** Adeola Odusola

## 2.3 Test Case TC\_U02.3: Test Function getBadges(\$totalPoints)

**Test Case ID:** TC\_U02.3

**Test Case Name:** TestFunctiongetBadgesReturnArrayOfBadgeNames()

**Method being tested:** getBadges(\$totalPoints)

**Short Description:** Test that the function getBadges(\$totalPoints) can return an array of badge names based on the total points

**Input Data to method being tested:** Create an instance of LeaderBoard class; pass 150 points to the getBadges(\$totalPoints) function

**Expected Results:** An array of silver and gold is returned.

**Tester's Name:** Adeola Odusola

## 2.4 Test Case TC\_U02.4: Test Function displayLeaderBoard(\$userId,\$userPoints)

**Test Case ID:** TC\_U02.3

**Test Case Name:** TestFunctiondisplayLeaderBoardReturnsScaledleaderBoardWithData()

**Method being tested:** displayLeaderBoard(\$userId,\$userPoints)

**Short Description:** Test that the function displayLeaderBoard(\$userId,\$userPoints) can return a scaled leaderboard displaying username, added post, edited post, points and badges.

**Input Data to method being tested:** Create an instance of LeaderBoard class; pass valid user ID and points into displayLeaderBoard(\$userId,\$userPoints) function

**Expected Results:** A leaderboard with username, added post, edited post, points and badges appear

**Tester's Name:** Adeola Odusola