

# NASA EVA Gamification

## UNIT TEST SUMMARY REPORT

---

**PHASE II**



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

## Table of Contents

Introduction	3
Test Summary	3
1.0 Test Suite U01: Code and Data Coverage Tests for Point Class	3
1.1 Test Case TC_U01.1: Test Function getPointsGenerated()	4
1.2 Test Case TC_U01.2: Test Function setPointsGenerated(\$point_generated)	4
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)	5
1.5 Test Case TC_U01.5: Test Function setPointsGenerated(\$point_generated)	5
1.6 Test Case TC_U01.6: Test Function setPointsGenerated(\$point_generated)	6
1.7 Test Case TC_U01.7:Test Function getTotalPointsForAllUsers()	6
1.8 Test Case TC_U01.8:Test Function getTotalPointsForAllUsers() when Null	6
1.9 Test Case TC_U01.9:Test Function getTotalPointsFromUserId(\$id)	7
1.10 Test Case TC_U01.10:Test Function getTotalPointsFromUserId(\$id)	7
1.11 Test Case TC_U01.11:Test Function GetTotalPointsFromUserId(\$id)	8
1.12 Test Case TC_U01.12:Test Function getTotalPointsFromUserId(\$id) when User Id Not Found	8
1.13 Test Case TC_U01.13:Test Function getEditedPostFromUserId(\$id)	8
1.14 Test Case TC_U01.14:Test Function getEditedPostFromUserId(\$id) when Null	9
1.15 Test Case TC_U01.15:Test Function getEditedPostFromUserId(\$id) when 0	9
1.16 Test Case TC_U01.16:Test Function getEditedPostFromUserId(\$id) when negative	10
1.17 Test Case TC_U01.17: Test Function construct (\$page_is_new)	10
1.18 Test Case TC_U01.18: Test Function construct(\$page_is_new)	10
1.19 Test Case TC_U01.19:Test Function construct(\$page_is_new)	11
2.0 Test Suite U02: Code and Data Coverage Tests for LeaderBoard Class	11
2.1 Test Case TC_U02.1: Test Valid displayAddedPages(\$clickedUser)	11
2.2 Test Case TC_U02.2: Test Invalid displayAddedPages(\$clickedUser)	12
2.3 Test Case TC_U02.3: Test Null displayAddedPages(\$clickedUser)	12
2.4 Test Case TC_U02.4: Test Valid displayEditedPages(\$clickedUser)	12
2.5 Test Case TC_U02.5: Test Invalid displayEditedPages(\$clickedUser)	13
2.6 Test Case TC_U02.6: Test Null displayEditedPages(\$clickedUser)	13
2.7 Test Case TC_U02.7: Test Valid getBadges(\$totalPoints)	14
2.8 Test Case TC_U02.8: Test Zero getBadges(\$totalPoints)	14
2.9 Test Case TC_U02.9: Test Invalid getBadges(\$totalPoints)	14
2.10 Test Case TC_U02.10: Test Null getBadges(\$totalPoints)	15
2.11 Test Case TC_U02.11: Test Valid displayLeaderBoard(\$userId,\$userPoints)	15
2.12 Test Case TC_U02.12: Test Zero displayLeaderBoard(\$userId,\$userPoints)	15

2.13 Test Case TC_U02.13: Test Invalid displayLeaderBoard(\$userId,\$userPoints)	16
2.14 Test Case TC_U02.14: Test Null displayLeaderBoard(\$userId,\$userPoints)	16
Test Case Summary Results:	17
Bug Report Summary:	17

Version Number	Description of Change	Author	Date
1.0	Initial Creation of Document	Samia Alam	7/24/18
1.1	Updated Test Suite U02 - LeaderBoard Class	Adeola Odusola	7/31/18
1.2	Updated Test Suite U01- Point Class	Hung Pham	8/1/18

## Introduction

The purpose of this document is to provide the test scripts along with the results and overall summary of the unit testing for phase 2 of NASA EVA gamification project. The test was done on 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.

## Test Summary

The test cases for Unit Testing of phase two of the NASA EVA gamification project are created to ensure the validity of the codes based on requirements specifications. The planned test cases for Unit Testing have been executed and passed. The Unit test codes have been provided through TestUnitPhase2.php file. Both Code coverage and data coverage goals of Unit Test have been accomplished. Currently there is no concern related to the code validity of points and leadership classes.

## 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

**Actual Results:** 0

**Status:** Success

**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.

**Actual Results:** 50

**Status:** Success

**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.  
**Actual Results:** 100  
**Status:** Success  
**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:** Display message ("Invalid Value") .  
**Actual Results:** Invalid Value  
**Status:** Success  
**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:** Display message ("Invalid Value").  
**Actual Results:** Invalid Value  
**Status:** Success

**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:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**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.

**Actual Results:** Array data

**Status:** Success

**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.

**Actual Results:** Null

**Status:** Success

**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.

**Actual Results:** 50

**Status:** Success

**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:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**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:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**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:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**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 .

**Actual Results:** 2

**Status:** Success

**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:** TestFunctionGetEditedPostFromNullUserIdReturnsInvalid()

**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 exists in the revision table in function getEditedPostFromUserId(\$id)

**Expected Results:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**Tester's Name:** Hung Pham

## 1.15 Test Case TC\_U01.15:Test Function getEditedPostFromUserId(\$id) when 0

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

**Test Case Name:** TestFunctionGetEditedPostFromZeroUserIdReturnInvalid()

**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:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**Tester's Name:** Hung Pham

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

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

**Test Case Name:** TestFunctionGetEditedPostFromNegativeUserIdReturnInvalid()

**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:** Display message ("Invalid Value").

**Actual Results:** Invalid Value

**Status:** Success

**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 and then call function getPointsGenerated.

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

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

**Actual Results:** 50

**Status:** Success

**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 and then call function getPointsGenerated.

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

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

**Actual Results:** 100

**Status:** Success

**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:** 0

**Actual Results:** 0

**Status:** Success

**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 Valid displayAddedPages(\$clickedUser)

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

**Test Case Name:** TestDisplayAddedPagesValid()

**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 page names is returned.

**Actual Results:** list of the added page names is returned

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.2 Test Case TC\_U02.2: Test Invalid displayAddedPages(\$clickedUser)

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

**Test Case Name:** TestDisplayAddedPagesInvalid()

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

**Short Description:** Test that the function displayAddedPages(\$clickedUser) does not display any page names but error message for an invalid user ID.

**Input Data to method being tested:** Create an instance of LeaderBoard class; Set user ID to -2. Provide the invalid value for \$clickedUser in the displayAddedPages(\$clickedUser) function

**Expected Results:** User ID must be valid

**Actual Results:** User ID must be valid

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.3 Test Case TC\_U02.3: Test Null displayAddedPages(\$clickedUser)

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

**Test Case Name:** TestDisplayAddedPagesNull()

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

**Short Description:** Test that the function displayAddedPages(\$clickedUser) does not display any page names but error message for a null user ID.

**Input Data to method being tested:** Create an instance of LeaderBoard class; Set a user ID to null. Provide the null value for \$clickedUser in the displayAddedPages(\$clickedUser) function

**Expected Results:** User ID must be valid

**Actual Results:** User ID must be valid

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.4 Test Case TC\_U02.4: Test Valid displayEditedPages(\$clickedUser)

**Test Case ID:** TC\_U02.4

**Test Case Name:** TestDisplayEditedPagesValid()

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

**Short Description:** Test that the function displayEditedPages(\$clickedUser) can display a list of edited 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 page names is returned.

**Actual Results:** List of the edited page names is returned

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.5 Test Case TC\_U02.5: Test Invalid displayEditedPages(\$clickedUser)

**Test Case ID:** TC\_U02.5

**Test Case Name:** TestDisplayEditedPagesInvalid()

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

**Short Description:** Test that the function displayEditedPages(\$clickedUser) does not display any page names but error message for an invalid user ID.

**Input Data to method being tested:** Create an instance of LeaderBoard class; Set user ID to -2. Provide the invalid value for \$clickedUser in the displayEditedPages(\$clickedUser) function

**Expected Results:** User ID must be valid

**Actual Results:** User ID must be valid

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.6 Test Case TC\_U02.6: Test Null displayEditedPages(\$clickedUser)

**Test Case ID:** TC\_U02.6

**Test Case Name:** TestDisplayEditedPagesNull()

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

**Short Description:** Test that the function displayEditedPages(\$clickedUser) does not display any page names but error message for a null user ID.

**Input Data to method being tested:** Create an instance of LeaderBoard class; Set the user ID value to null. Provide the invalid value for \$clickedUser in the displayEditedPages(\$clickedUser) function

**Expected Results:** User ID must be valid

**Actual Results:** User ID must be valid

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.7 Test Case TC\_U02.7: Test Valid getBadges(\$totalPoints)

**Test Case ID:** TC\_U02.7

**Test Case Name:** TestGetBadgesValid()

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

**Short Description:** Test that the function getBadges(\$totalPoints) can return an array of badge names based on valid 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 bronze is returned.

**Actual Results:** An array of silver and bronze is returned

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.8 Test Case TC\_U02.8: Test Zero getBadges(\$totalPoints)

**Test Case ID:** TC\_U02.8

**Test Case Name:** TestGetBadgesZero()

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

**Short Description:** Test that the function getBadges(\$totalPoints) returns an empty array for zero points.

**Input Data to method being tested:** Create an instance of LeaderBoard class; pass 0 points to the getBadges(\$totalPoints) function

**Expected Results:** An empty array is returned.

**Actual Results:** An empty array is returned

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.9 Test Case TC\_U02.9: Test Invalid getBadges(\$totalPoints)

**Test Case ID:** TC\_U02.9

**Test Case Name:** TestGetBadgesInvalid()

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

**Short Description:** Test that the function getBadges(\$totalPoints) returns an empty array for an invalid point value

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

**Expected Results:** An empty array is returned.

**Actual Results:** An empty array is returned

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.10 Test Case TC\_U02.10: Test Null getBadges(\$totalPoints)

**Test Case ID:** TC\_U02.10

**Test Case Name:** TestGetBadgesNull()

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

**Short Description:** Test that the function getBadges(\$totalPoints) returns an empty array for a null point value.

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

**Expected Results:** An empty array is returned.

**Actual Results:** An empty array is returned

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.11 Test Case TC\_U02.11: Test Valid displayLeaderBoard(\$userId,\$userPoints)

**Test Case ID:** TC\_U02.11

**Test Case Name:** TestDisplayLeaderBoardValid()

**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

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

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.12 Test Case TC\_U02.12: Test Zero displayLeaderBoard(\$userId,\$userPoints)

**Test Case ID:** TC\_U02.12

**Test Case Name:** TestDisplayLeaderBoardZero()

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

**Short Description:** Test that the function displayLeaderBoard(\$userId,\$userPoints) displays no data found on user ID of zero and points of zero

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

**Expected Results:** display says: No data found!

**Actual Results:** display says: No data found!

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.13 Test Case TC\_U02.13: Test Invalid displayLeaderBoard(\$userId,\$userPoints)

**Test Case ID:** TC\_U02.13

**Test Case Name:** TestDisplayLeaderBoardInvalid()

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

**Short Description:** Test that the function displayLeaderBoard(\$userId,\$userPoints) displays no data found on invalid user ID and point values

**Input Data to method being tested:** Create an instance of LeaderBoard class; pass user ID = -2 and points = -100 to displayLeaderBoard(\$userId,\$userPoints) function

**Expected Results:** display says: No data found!

**Actual Results:** display says: No data found!

**Status:** Pass

**Tester's Name:** Adeola Odusola

## 2.14 Test Case TC\_U02.14: Test Null displayLeaderBoard(\$userId,\$userPoints)

**Test Case ID:** TC\_U02.14

**Test Case Name:** TestDisplayLeaderBoardNull()

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

**Short Description:** Test that the function displayLeaderBoard(\$userId,\$userPoints) displays no data found on null user ID and point values

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

**Expected Results:** display says: No data found!

**Actual Results:** display says: No data found!

**Status:** Pass

**Tester's Name:** Adeola Odusola



## Test Case Summary Results:

Test Assessment	Total Number of test Cases	% of Total Planned	Comments
Test Cases Planned	33	100%	
Test Cases Run	33	100%	
Test Cases Passed	33	100%	
Test Cases Failed	0	0%	

## Bug Report Summary:

Severity Level	Total Reported	# Total Resolved	% Total resolved	#Total Unresolved	% Total Unresolved
1 (Critical)	0	0	0	0	0
2 (High)	0	0	0	0	0
3 (Medium)	0	0	0	0	0
4 (Low)	0	0	0	0	0