Black Box Testing Document

Virtual Museum

Theme – World War II

Nikunj Mittal (43) *Mayank Yadav (34)* *Harshit Bansal (26)*

Supervised by Prof. Samit BHATTACHARAYA

Black-box Testing Document

Purpose:

The document records the the process and results of System Testing conducted on the “World War 2 Virtual Museum app” using BlackBox Testing method.

Introduction:

There are various methods that can be used to Test Working of a product.

Black box testing method which is a method of software testing that examines the functionality of an application without peering into its internal structures or workings.

This method of test can be applied virtually to every level of software testing. This method has been used here for System Testing.

There are various methods of doing system testing like Equivalence Class testing, Boundary Value Testing.

Equivalence partitioning or **equivalence class partitioning** (**ECP**) is a software testing technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived. In principle, test cases are designed to cover each partition at least once.

Boundary value analysis is a software testing technique in which tests are designed to include representatives of boundary values in a range. The idea comes from the boundary

Equivalence Class method has been used in the document for testing.

Equivalence class partitioning

Equivalence partitioning focuses of dividing the input data from the user into equivalent classes and then deriving test cases from those classes to check the functioning of the system .

Partitioning into classes

The Virtual Museum System uses touch interaction with various buttons and exhibits from user as input so it is convenient to divide all the inputs into classes on the basis of Button types that are initiated by that input. The following table shows the classes generated using this method.

S.no	**Class name**
1	Start Tour Button
2	Joystick Movement Buttons
3	Joystick Rotate
4	Joystick Jump Buttons
5	Map Input Buttons
6	Add Notes Buttons
7	View Exhibit Input Buttons
8	Exhibit Resize Input Buttons
9	Move Exhibit Input Buttons
10	View exhibit discription Buttons
11	Manage Wishlist Buttons
12	Move to next Exhibit Buttons
13	Quit Tour Button

Correctness of the Partitioning

The Virtual Museum System takes in input from user as touch and sense interactions only, Each touch or sense interaction is interact with some button so each inputs will belong one of the button classes. Therefore ours classes cover all the inputs and each input can be put into one of these classes uniquely.

Equivalence Classes and Test Cases

Convention for testing:

Result-Pass means expectations were met by the system.

Result-Fail means expectations were not met completely by the system

1.Start Button Class:

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Start Auto Tour by clicking Start Auto Tour Button	Auto tour should be loaded	Auto Tour Begins.	Pass

2.Joystick Movement Buttons Class:

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Click on move right in Joystick move right when no exhibit on right	Player moves one step to right	Player moves one step to right	Pass
Task2	Click on move right in Joystick when a exhibit or boundary on right	Player doesn't move as motion is restricted	Player doesn’t move and stays in his place	Pass

3.Joystick Rotate

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Rotate rightward by tilting device towards right	Player should rotate rightward without any change in position	Player rotates right	Pass

4.Joystick Jump Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task 1	Click on Jump button while player is stationary	Player should Jump and remain in its place	Player jumps in his own place	Pass
Task2	Click on Jump button while player is moving forward continuously	Player should Jump and continue to move forward	Player jumps and stops	**Fail**

5.Map Input Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Click on MAP button to view the map	The Map should be loaded and should be displayed on the screen	The Map is loaded and displayed	Pass
Task2	View Map and move player to right	The player Location pointer on the map should also move right	The Location pointer moves right	Pass
Task3	View Map and click on a section to move to that section	The player should moves to the new section and the location pointer on the map also moves to new location	Player moves to the new section but the Location pointer on the map doesn't move	**Fail**

6.Add Notes Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Click on the add note button to add a new note	Notes window appear where user can add his note	Notes Window appears	Pass

7.View Exhibit Input Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Click on any exhibit to see details about the exhibit	View Exhibit window should open displaying detailed information	View exhibit window opens	Pass

8.Exhibit Resize Input Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	**Precondition: View** Exhibit Window is open. **Task**: Resize the exhibit so that it remains within window boundaries	exhibit should get resized	exhibits is resized	Pass
Task2	**Precondition: View** Exhibit Window is open**.** **Task:**Enlarge the exhibit beyond the window boundaries	exhibit should not get enlarged further	exhibit is not enlarged further	Pass

9.Move Exhibit Input Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	**Precondition :**View Exhibit Window is open**.** **Task**: Move exhibit to right	Exhibit should move one unit to the right	Exhibit moves right by one unit	**Pass**
Task2	**Precondition :**View Exhibit Window is open**.** **Task**: Move exhibit to right when exhibit is at window boundary	Exhibit should not move beyond the window boundaries as motion beyond boundaries is restricted	Exhibit doesn't move beyond window boundaries	**Pass**

10. View exhibit description Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	**Precondition**: View Exhibit Window is open. **Task**: Click on View Description Button to display	Exhibit Description should appear	Exhibit Description is displayed	Pass

11.Manage Wishlist Buttons Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Add a Exhibit to wishlist	Exhibit should be added to wishlist	Exhibit added to wishlist	Pass
Task2	Remove Exhibit from wishlist	Exhibit should be removed from wishlist	Exhibit removed from wishlist	Pass

12. Move to next Exhibit Buttons

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	**Precondition**: Player doing auto tour. **Task**: Click on next exhibit button to move to next exhibit	View Exhibit window should open for the next exhibit	View Exhibit window opens for the next exhibit	Pass
Task2	**Precondition:** Player doing auto tour **Task**: Click on next exhibit button when already on last exhibit	Prompt shows telling there are no more exhibits to display	Stays on the current screen and no change	Fail

13.Quit Button Class

S.no	Task Assigned	Expected Output	Actual Output	Result
Task1	Quit Tour by clicking Start Auto Tour Button	Tour should be ended	Tour ends.	Pass

Conclusion:

The Blackbox testing was conducted to test the system and useful information from about the shortcomings of the system were discovered from those test tasks which failed.

Number of testcases(Tasks):21

Number of testcases that failed:3

Number of testcases that passed:18