EGR 402 – Capstone Design

Deliverable 7: Testing

Team Name: Blueprints

Team Members: Jordan Ziegler

  Jeremy Fischer

  Justus Karenzi

  Sudi Nsengiyumva

  Dylan Shanahan

Client’s Name: Dr. Mark Gordon

Advisor’s Name: Professor Thomas Renck

Version: 2

Date: April 9, 2015

**7.1 Unit Testing**

Unit Testing is the process in which a module’s functionality is observed through testing. A module is an individual and independent part of the application that is able to be tested. Thus these modules or units are the smallest testable portions of the application. To perform a unit test the application is broken down into these modules and then tested for its functionality based on a single purpose.

7.1.1 Button Testing

This unit test takes the application and tests the buttons responses to being pressed. The button is considered a success if the actual result is the same as the expected result. In this case the button should take the user to the buttons location only when clicked. The test is a failure should the button take the user to the wrong location, should it not react when clicked, or should it take the user to a location without clicking. The table containing the information for this test can be seen below in Table 7.1.1. Should there be any failures they will be discussed after the table along with plans to rectify the failure.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Number (BP-UT-1)** | **Test Procedure** | **Expected Result** | **Actual Result** | **Pass/Fail** | **Comments** |
| 1 | “Start Button” | When clicked moves to Building Activity | Performed as expected | Pass | No comments |
| 2 | “Engineering” | When clicked moves to Engineering Activity | Performed as expected | Pass | No comments |
| 3 | “James” or “Yeager” | When clicked moves to an “Unfinished Page” | Performed as expected | Pass | No comments |
| 4 | “Room” | When clicked moves to the camera activity | Performed as expected | Pass | No comments |
| 5 | “Back” | Returns the user to the previous page | Performed as expected | Pass | No comments |

Table 7.1.1 - Testing information regarding Button Success or Failure

7.1.2 Scanner Testing

This unit test checks to see that the scanner activity is working properly. For the Scanner to be a success it must activate after the user selects a room destination. This test is a failure if the Scanner fails to activate or if it activates improperly (such as only partially opens on the screen or is unable to initialize a scan phase). Table 7.1.2 (seen below) shows the information regarding the Scanner’s testing. Any failures and plans to fix said failures can be seen after the table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Number**  **(BP-UT-2)** | **Test Procedure** | **Expected Result** | **Actual Result** | **Function Pass/Fail** | **Program Pass/**  **Fail** | **Comments** |
| 1 | “100J” | When pressed the application jumps to the Scanner Activity | Jumps to the Camera Application on the phone | Pass | Fail | Application does what it’s asked but not what is needed for the program. |
| 2 | “100K” | When pressed the application jumps to the Scanner Activity | Jumps to the Camera Application on the phone | Pass | Fail | Application does what it’s asked but not what is needed for the program. |
| 3 | “100L” | When pressed the application jumps to the Scanner Activity | Jumps to the Camera Application on the phone | Pass | Fail | Application does what it’s asked but not what is needed for the program. |
| 4 | “110D” | When pressed the application jumps to the Scanner Activity | Jumps to the Camera Application on the phone | Pass | Fail | Application does what it’s asked but not what is needed for the program. |
| 5 | “130H” | When pressed the application jumps to the Scanner Activity | Jumps to the Camera Application on the phone | Pass | Fail | Application does what it’s asked but not what is needed for the program. |

Table 7.1.2 - Testing information regarding Scanner Success or Failure

This test passes its function test because it does what is act of it, meaning that when this portion of the application is started the program tells it to open the camera on the phone. However, using the camera function is not the same as a Scanner field much like barcode scanner apps. Due to Android Studios’ continuous updates as well as the limited documentation on it our team has had difficulty finding information on how to initialize a scanner view. By using information on the web our team hopes that creating a separate project and then integrating the scanner into the team application.

7.1.3 Image Recognition Testing

This phase of the unit testing is designed to test the image Recognition function of the Application. For this to be a success this portion needs to be able to recognize the image given to it. This test is considered a failure should the application fail to perceive the appropriate image or misinterprets the image given. Seen below is Table 7.1.3 which contains the testing information for this unit test. As with the other tests the discussion and plan for any failures can be seen after the table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Number**  **(BP-UT-3)** | **Test Procedure** | **Expected Result** | **Actual Result** | **Pass/Fail** | **Comments** |
| 1 | Given Image:  “100J” | Application returns the Image from the outside environment. | Returns information about the image. | Pass | No comments |
| 2 | Given Image:  “100K” | Application returns the Image from the outside environment. | Returns information about the image. | Pass | No comments |
| 3 | Given Image:  “100L” | Application returns the Image from the outside environment. | Returns information about the image. | Pass | No comments |
| 4 | Given Image:  “110D” | Application returns the Image from the outside environment. | Returns information about the image. | Pass | No comments |
| 5 | Given Image:  “130H” | Application returns the Image from the outside environment. | Returns information about the image. | Pass | No comments |

Table 7.1.3 - Testing Information regarding Image Recognition Success or Failure

This test passes because having integrated the Vuforia SDK into Unity Vuforia is now able to utilize image recognition. As it scans a picture it is able to correctly identify it provided that the picture is within its programming database. This testing is updated since version 1 of this document due to the use of Unity.

7.1.4 Augmented Reality Output Testing

This unit test is designed to test the Augmented Reality Output created by the application. This test is considered a success if the application is able to output the correct Augmented Reality Output, meaning that the application is able to impose the object its supposed to on top of the camera view. For the purpose of testing the object can be anything designed within the code but for the purpose of the final application the object will be a navigational arrow. This test utilizes multiple objects to make sure the code is able to differentiate between the correct objects. This test is a failure should the code be unable to output an augmented object, if the code is unable to impose the object over the camera view (meaning the object is output but there is no camera view visible on the screen), or if the output is of the wrong object. Table 7.1.4 shows the information for this unit test. Any failure’s discussion and the plans to fix said failures can be seen after the table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Number** | **Test Procedure** | **Expected Result** | **Actual Result** | **Pass/Fail** | **Comments** |
| 1 | Application is asked to output a given object. | Outputs an Augmented Reality object. | An object was outputted. | Pass | No comments |
| 2 | Application is asked to output a given object. | Outputs an Augmented Reality object. | An object was outputted. | Pass | No comments |
| 3 | Application is asked to output a given object. | Outputs an Augmented Reality object. | An object was outputted. | Pass | No comments |
| 4 | Application is asked to output a given object. | Outputs an Augmented Reality object. | An object was outputted. | Pass | No comments |
| 5 | Application is asked to output a given object. | Outputs an Augmented Reality object. | An object was outputted. | Pass | No comments |

Table 7.1.4 - Testing Information regarding Augmented Reality Output Success or Failure

Since the integration of Unity and the Vuforia SDK our group has been able to output an Augmented Reality object.

**7.2 Integration Testing**

Integration Testing is the process through which the individual modules from the unit testing are integrated together into large subcomponents and tested to verify that they are able to operate correctly together.

7.2.1 Scanner/Image Recognition Testing

This integration testing utilizes the Scanner and Image Recognition modules to test their ability to operate together. The Scanner activity calls the Image Recognition activity so this is a key component of the application. Should the Scanner activity not be able to recognize the image or the image recognition within the Scanner activity is wrong then this test fails. The discussion for any failures can be seen after Table 7.2.1.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Writer: Jordan Ziegler** | | | | | | | | | |
| **Test Case Name:** | | **Scanner and Image Recognition Test** | | | | **Test ID#:** | BP-IT-1 | | |
| **Description:** | | **Tests the Scanner’s ability to recognize an Image** | | | | **Type:** | ☐  ☐ | | White  **Black** |
| **Tester Information** | | | | | | | | | |
| **Application:** | | BluePrints Augmented Reality App | | | | | **Date:** | 3/23/15 | |
| **Software Ver:** | | 0.1 | | | | | **Time:** | 3:00 PM | |
| **Setup:** | | The application starts at the room selection | | | | | | | |
| **Step** | **Action** | **Expected Result** | **P** | **F** | **N/A** | **Comments** | | | |
| 1 | Click room button | Application goes to Scanner View | x |  |  | For the purpose of testing we’re using Camera View | | | |
| 2 | Input an Image for Scanner | Application returns information regarding the image | x |  |  |  | | | |
| 3 | Click back button to return to room selection |  | x |  |  |  | | | |
| **Overall Test Results:** | | | x |  |  |  | | | |

Table 7.2.1 - Step-by-Step Testing Table regarding Scanner/Image Recognition Integration

Our team used this testing system to check the application’s ability to recognize an image and extract meaningful data from that image. The application was able to go into the scanner view where we could see the image being sent and then was able to recognize the image using the Vufroia SDK with Unity.

7.2.2 Image Recognition/Database/Augmented Reality Output Testing

The final integration testing takes the Image Recognition module and combines it with the Augmented Reality Output module while using the Database held within the application. This test will utilize the image recognition to obtain the information, check it with the database, and then output the appropriate augmented reality output. Should the Image Recognition not recognize the given image, should the given image not be correctly identified with the database, should the incorrect augmented output be given instead, or any combination of the above then this test can be considered a failure. Any failure that is incurred during this test will be discussed after Table 7.2.2.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Writer: Dylan Shanahan** | | | | | | | | | |
| **Test Case Name:** | | **Comprehensive Test #1** | | | | **Test ID#:** | BP-IT-2 | | |
| **Description:** | | **This test checks the ability to obtain information from the camera, and output the AR output** | | | | **Type:** | ☐  ☐ | | White  **Black** |
| **Tester Information** | | | | | | | | | |
| **Application:** | | BluePrints Augmented Reality App | | | | | **Date:** | 3/23/2015 | |
| **Software Ver:** | | 0.1 | | | | | **Time:** | 3:00pm | |
| **Setup:** | | For this test the application starts at the information collection from an image it receives through the scanner (for a generic test the room selected was 100J). | | | | | | | |
| **Step** | **Action** | **Expected Result** | **P** | **F** | **N/A** | **Comments** | | | |
| 1 | Input an Image for Scanner | Application returns information regarding the image | x |  |  |  | | | |
| 2 | Wait for application | Application returns an Augmented Reality Object | x |  |  |  | | | |
| 3 | Click back button to return to room selection |  | x |  |  |  | | | |
| **Overall Test Results:** | | | x |  |  |  | | | |

Table 7.2.2 - Step-by-Step Testing Table regarding Image Recognition/Database/Augmented

Reality Output Integration

This test was used to check the programs ability to gather information from an image and then outputting an Augmented Reality Object. Not only was the application able to recognize an image but it was also able to output an Augmented Reality Object. It was also able to utilize the button controls.

**7.3 System Testing**

This test is the completed systems test. The purpose of this test is to test the application as a whole from beginning to end. The application is a success should the application complete from start to finish without complications. This test is a failure should any part of the application fail or should the application perform in some way that was not desired. Table 7.3 shows the testing information regarding the System Testing. Any failures noted will be discussed after the table as well as the plans to correct these failures.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Number** | **Test Procedure** | **Expected Result** | **Actual Result** | **Pass/Fail** | **Comments** |
| 1 | The application will start from the beginning as if being used by a user in the Engineering building | Accepts a building and room input from user. Accepts an image. Outputs an Augmented Reality Object. | It goes to a camera which shows them where they’re going but doesn’t do augmented reality. | fail | Still some minor details causing errors when all systems are working in tandem. |
| 2 | The application will start from the beginning as if being used by a user in the Yeager building. | Accepts a building input from user. Leads to a “Not Currently Working“ page. | Accepts a building input from user. Leads to a “Not Currently Working“ page. | pass |  |
| 3 | The application will start from the beginning as if being used by a user in the Engineering building to test a different room. | Accepts a building and room input from user. Accepts an image. Outputs an Augmented Reality Object. | It goes to a camera which shows them where they’re going but doesn’t do augmented reality. | fail | Still some minor details causing errors when all systems are working in tandem. |
| 4 | The application will start from the beginning as if being used by a user in the Engineering building to test a different room. | Accepts a building and room input from user. Accepts an image. Outputs an Augmented Reality Object. | It goes to a camera which shows them where they’re going but doesn’t do augmented reality. | fail | Still some minor details causing errors when all systems are working in tandem. |
| 5 | The application will start from the beginning as if being used by a user in the Engineering building to test a different room. | Accepts a building and room input from user. Accepts an image. Outputs an Augmented Reality Object. | It goes to a camera which shows them where they’re going but doesn’t do augmented reality. | fail | Still some minor details causing errors when all systems are working in tandem. |

Table 7.3 - Testing Information regarding Complete System Success or Failure

This test was supposed to be a final test for the application. It was supposed to run from the beginning of the application to the end without difficulties. Each test would accept different inputs with the hope that the application would be able to work through each one. Even though many of the other tests passed their portion of the testing we’re still having some minor difficulties when incorporating the entire application into one system. Our team is currently working on this to rectify the minor bugs and have a completed proof of concept application.