SWE40001 Software Engineering Project

**Smart-glass based**

**Remote Guidance System**

**TEST PLAN**

Group 21

Lyndon Prado - 9740783

Tingcong Jimmy Li - 100017000

Keagan Foster - 101609822

Ayub Khan - 100667654

Dineth Gunawardena - 100862158

Kosala Edirisinghe - 101265981

Krishna Adhikari - 4953193

Liam Pan -101106174

Migara Gunarathne - 101002269

Shenal Nirushka – 101054998

**Table 1. Document Change Control**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Authors** | **Summary of Changes** |
| 1 | 14/04/2018 | Ayub Khan | Initial Draft |
| 2 | 29/04/2018 | Lyndon Prado | Addition of Test Cases |
| 2 | 02/05/2018 | Dineth Gunawardena | Addition of Product Level Pass/Fail Criteria |
| 2 | 05/05/2018 | Krishna Adhikari, Ayub Khan, Dineth Gunawardena | Fixing and completing test cases |
| 2 | 08/05/2018 | Dineth Gunawardena | Updated based on feedback from supervisor (Adding non-functional test cases and fixing errors) |
| 2 | 09/05/2018 | Ayub Khan | Adding more detail to scope with assumptions and constraints, adding environment requirements, making changes to test deliverable and features to be tested |
| 2.01 | 09/05/2018 | Liam Pan | Document Standards Review  Spelling and Grammar Review |

**Table 2. Document Sign Off**

|  |  |  |
| --- | --- | --- |
| **Name (Position)** | **Signature** | **Date** |
| Dr. Tony Huang |  |  |
| Lyndon Prado |  |  |
| Tingcong Jimmy Li |  |  |
| Keagan Foster |  |  |
| Ayub Khan |  |  |
| Dineth Gunawardena |  |  |
| Kosala Edirisinghe |  |  |
| Krishna Adhikari |  |  |
| Liam Pan |  |  |
| Migara Gunarathne |  |  |
| Shenal Nirushka |  |  |

[**1.) Introduction**](#_5jd57xs9l0jt) **3**

[1.1) Purpose of the test plan](#_ptc8dnl9waqr) 3

[1.2) Scope](#_jitsrj3tbhkv) 4

[1.3) References Material](#_r34fdlz5giap) 4

[1.4) Objective](#_x5zm8aijvdrl) 4

[1.5) Resources Required](#_cc1m3yp9sjg7) 4

[**2.) Test Items**](#_wrkhbe74lb8e) **5**

[2.1) Features to be Tested](#_z03aqpz2jnbq) 5

[2.2) Test Cases](#_j6syjet13esq) 5

[2.2.1) Functional Test Cases](#_qfqeibh76d3s) 5

[2.2.1) Non-functional Testing](#_yj37uwumz8tp) 9

[2.3) Features not to be Tested](#_4qu0upk8b53w) 12

[**3.) Strategy**](#_9mjoyidryh43) **12**

[3.1) Roles and Responsibilities](#_su6vt1x2n869) 13

[3.2) Test Deliverables](#_v3rq1s246ogu) 13

[3.3) Schedule](#_ksbov6lvq1th) 13

[3.4) Risk and Contingency](#_6ne794agkpc) 14

[3.5) Testing Tasks](#_9eb62rfk4zx3) 15

[**4.) Pass/Fail Criteria**](#_tg0p6jho208u) **15**

[4.1) Product Level](#_4s6iwddqilan) 15

[4.2) Testing Stages](#_si2y23b2qf7d) 16

[4.3) Suspension criteria and resumption requirements](#_4469vxb8knpp) 18

[Suspension Criteria](#_ah8y48p0clfb) 18

[Resumption Requirements](#_iq8q0u8xpeu5) 18

[4.4) Approvals](#_rwkt6yx0qvd2) 18

# **1.)** **Introduction**

## 1.1) Purpose of the test plan

The test plan is a document designed to outline the scope, resources, approach, strategy and criteria for all testing activities carried out during and after the development of the system. The plan will clearly identify the features that are to be tested and the features which will not be, and the stages at which tests are to be carried out.

## 1.2) Scope

Testing will be carried out at several stages of the life cycle as the system is developed. Testing will ensure system defects and bugs are caught early on in the development and also helps the team reevaluate their approach to developing a particular aspect of the project. The team will be following a four level testing strategy. This will allow the project to be tested at different development stages at maximum efficiency.

· Unit Testing

· System Testing

· Integration Testing

· Acceptance Testing

## 1.3) References Material

The following document has been created according to the details specified in the documents below. The documents below will provide further information on the project.

· Project Plan

· Software Quality Assurance Plan

· Software Requirements Specification

· IEEE 829 Standard template (Test Plan)

## 1.4) Objective

Carry out appropriate tests at appropriate stages of development.

The test plan shall act as a guideline for performing these different types of tests required for maintaining the quality and efficiency of the system.

## 1.5) Resources Required

· 2 Smart Glasses

· May require Smart mobiles (for higher processing power)

· 2 users acting as Operator and Instructor

· A ruler (for point of reference)

· Piece of meat (for the user test)

· Hotspot or Wifi

· Glove (to identify the users hand)

· IDE (to create and run unit tests)

# **2.)** **Test Items**

· The product to be tested is the Smart Glass Based Remote Guidance System

## 2.1) Features to be Tested

· Both users must be able to begin transmission.

· Both users must be able to end transmission.

· The operator must be able to take a picture during the live video streaming between the operator and instructor.

· The operator must be capable of annotating the image during the live video stream.

· The operator must also be able to share the pictures taken.

· The instructor must be able to send his hand gestures to the operator’s glasses in real time.

· The system must be able to identify a piece of meat and assess its quality via colour and dimensions.

## 2.2) Test Cases

### 2.2.1) Functional Test Cases

**Table 3. Functional Test Cases**

|  |  |  |
| --- | --- | --- |
| **Test Case(s)** | **Steps** | **Expected Results** |
| Test case: | Select Role |  |
| 1) Check if users can open the applications without issues  **Preconditions:** User must have application installed. | Open the application | The application should run and not freeze or crash |
| 2a) Check if the device can be set as an instructor device  **Preconditions:** User must have instructor privileges | Select the option “Instructor” | The glasses should have instructor features.  The glasses should then be ready to request connection or accept a request |
| 2b) Check if device can be set as the operator device  **Preconditions:** User must have operator privileges | Select the option “Operator” | The glasses should have operator features. The glasses should then be ready to request connection or accept a request |
|  | Connect with user |  |
| 1a) Check if instructor can view the other user(operator) in instructor mode  **Preconditions:** Operator devices must be visible and open for connections. | Instructor Select “connect with user” as an Instructor | The program should open up a window with a list of users that are operators. |
| 1b) Check if operator can view the other user(instructor) in operator mode  **Preconditions:** Instructor devices must be visible and open for connections | Operator Select “connect with user”as an operator | The program should open up a window with a list of users that are instructors. |
| 2a)Check if instructor can connect with an operator in instructor mode  **Preconditions:** Instructor and operator devices must be open for connections | Instructor Select 1 device from list of users(operators) | There should only be one listed operator in the list, as one device has been set up as an operator |
| 2b)Check if operator can connect with an instructor in operator mode  **Preconditions:** Operator and instructor devices must be open for connections | Operator Select 1 device from list of users (instructors) | There should only be one listed instructor in the list, as one device has been set up as an instructor |
|  | Send video feed |  |
| 1)Check if Operator can send video feed to Instructor  **Preconditions:** Connection with instructor has to be established | Operator Select “Send video feed” option | Receiving user is prompted to receive video streaming |
| 2) Check if Instructor can receive video stream  **Preconditions:** Connection with operator has to be established | Instructor Select “accept” to video streaming request | Video stream connection is successfully setup and video feed would stream from the operator’s device to the instructor’s device |
| 2b)Check if Instructor can refuse video streaming  **Preconditions:** Operator has to send video request | Instructor Select “decline” to video streaming request | Video stream setup is dropped and sending user is notified of the video stream request being denied. |
|  | View video feed |  |
| 1)Check if Instructor can view received video feed  **Preconditions:** Instructor has to receive video feed | Select the option “view video” | Sending user begins to send video feed to the system. The system is meant to process the feed and send it to the receiving user. The receiving user should be able to view the feed of the sending user. |
|  | Send hand gestures |  |
| 1) Check if Instructor can send hand gestures to Operator  **Preconditions:** The system has to have the video feed of the operator | Select the option “send hand gesture” | The system should start obtaining the video feed of the operator and also the video feed of the instructor. The video feed of the instructor is then processed to obtain footage only of the instructor's hand and not the background surroundings. The image of the hands should then be superimposed on the operator’s video feed and then sent back to the operator. |
|  | Make sketch |  |
| 1) Check if Operator can capture image  **Preconditions:** Operator must have video feed to capture image from | Select “Capture Image” button | System captures image of operator’s video feed |
| 1b) Check if operator can edit image | Select “Edit” button | The image is opened up in an editable format |
| 1c) Check if operator can draw sketch on image | Draw sketch on image by using their fingers | System registers finger movements and traces it onto image |
| 1d) Check if operator can undo sketch | Select “undo” button | System reverts image back to original condition |
| 1e) Check if operator can save edited image | Save edited image | System saves edited image and prompts user to send it to paired device |
| 1f) Check if operator can delete edited image | Delete edited image | System deletes edited image and goes back to video feed |
|  | Send sketch |  |
| 2) Check if operator can send sketched image to instructor  **Preconditions:** Operator must have stable connection with instructor device | Select “send to instructor” option to send to device it is paired with | System prepares image and sends image to the connected instructor |
|  | View sketch |  |
| 3) Check if Instructor can view image  **Preconditions:**Instructor device must receive image | Select “view image” | System displays image sent by the operator. |
|  | Switch to Instructor |  |
| \*Note: User1 - Current Operator / User2 - Current Instructor | | |
| 1)Check if Operator can switch modes to Instructor  **Preconditions**: Operator and instructor device have been assigned beforehand | User1 select “switch to instructor” | System notifies User2 of the requested switch |
| 2)Check if instructor can accept switch request | User2 select “accept” to the system’s switch permission | System notifies User1 of the switch agreement, User1 is switched to Instructor mode and vice versa for User2 |
| 2b)Check if instructor can decline switch request | User2 select “decline” to system switch permission | System notifies User1 of switch disagreement |
|  | Switch to Operator |  |
| \*Note: User1 - Current Instructor / User2 - Current Operator | | |
| 1)Check if Instructor can switch modes to Operator  **Preconditions**: Operator and instructor device have been assigned beforehand | User1 select “switch to operator” | System notifies User2 of the requested switch |
| 2)Check if operator can accept switch request | User2 select “accept” to the system’s switch permission | System notifies User1 of the switch agreement, User1 is switched to Operator mode and vice versa for User2 |
| 2b)Check if operator can decline switch request | User2 select “decline” to system switch permission | System notifies User1 of switch disagreement |

### 

### 2.2.1) Non-functional Testing

**Testing Goal:** The purpose of the following test cases is to make sure that the application we develop would not only be functional, but also be user friendly with a small learning curve allowing the average user to quickly adapt to the user interface.

**Testing Procedure:** The test would be performed on a set of participants, who will be asked to perform basic tasks using the applications. Upon completing the task, the user would be expected to be able to give feedback on the difficulty of the task and what their suggestion to improve the user interface would be.

**Pass/Fail Criteria:** The tasks given are very simple, and would have constant pass criteria. Each task should be completed within 30 seconds and would have a difficulty goal of being less than 5 (1 being very easy, and 10 being very hard).

**Testing Constants**: All of the tests will be carried out on the application on the Smart Glasses, using sample data provided to them.

**Testing Assumptions:** The tests will be carried out in pairs of participants where one of the participants will play the role of the “operator” while the other acts as the “instructor”.

**Table 4. Non-functional testing**

|  |  |  |
| --- | --- | --- |
| **Test Case(s)** | **Steps** | **Expected Results** |
| 1a) Check if user can select role.  **User Task:** Select role as “operator” | 1. Open the application 2. Select the option “Operator” | User successfully selects operator role. |
| 1b) Check if user can select role.  **User Task:** Select role as “instructor” | 1. Open the application 2. Select the option “instructor” | User successfully selects instructor role. |
| 2a) Check if user can connect with the other user as an instructor  **Use Task:** Pair up your pair of glasses with another pair of glasses | 1. Select “connect with user” as an Instructor 2. Select 1 device from list of users(operators) | User successfully connects with other operator as an instructor |
| 2b) Check if user can connect with the other user as an operator  **Use Task:** Pair up your pair of glasses with another pair of glasses | 1. Select “connect with user”as an operator 2. Select one device from list of users (instructors) | User successfully connects with other instructor as an operator |
| 3a) Check if user can send video feed request as an operator  **User Task:** Send a video feed request to connected device. | 1. Select “Send video feed” option | User successfully sends video feed request |
| 3b) Check if user can receive video feed request as an instructor  **User Task:** Send a video feed request to connected device. | 1. Select “accept” to video streaming request | User successfully accepts video feed request |
| 4) Check if user can view received video feed as an instructor  **User Task:** View operator’s video feed | 1. Select the option “view video” | User is able to view video |
| 5) Check if user can make a sketch.  **User Task:** Capture and edit an image | 1. Select “Capture Image” button 2. Select “Edit” button 3. Draw sketch on image by using their fingers 4. Save edited image | User successfully edits and saves image into gallery |
| 6a) Check if user can send sketch  **User Task:** Send edited image to instructor | 1. Select “send to instructor” | User successfully sends edited image |
| 6b) Check if user can view image  **User Task:** View image operator has sent you | 1. Select “view image” | User is able to view image sent by operator |
| 7a) Check if user can switch modes to Instructor  **User Task:** Switch your role from an “operator” to an “instructor” | 1. Select “switch to instructor” | Request is sent after user selects “switch to instructor” |
| 7b)Check if user can accept switch request  **User Task**: Accept operator’s request to switch roles | 1. Select “accept” to the operator’s switch request | Users successfully switch roles |
| 8a) Check if user can switch modes to operator  **User Task:** Switch your role from an “instructor” to an “operator” | 1. Select “switch to operator” | Request is sent after user selects “switch to operator” |
| 8b) Check if user can accept switch request  **User Task**: Accept instructor’s request to switch roles | 1. Select “accept” to the instructor’s switch request | Users successfully switch roles |

## 

## 2.3) Features not to be Tested

Since all of the features are being implemented for the first time and are all capable of being tested, there are currently no features that will not be tested.

# 3.) Strategy

As stated the project will be tested following a four level testing strategy

**· Unit Testing –** This stage of testing will be carried out during the initial coding stage when there are multiple modules of code being developed by different members of the team. This will be carried out by generating multiple unit tests in order to test pieces of code. Will help detect errors early on.

**· Integration Testing –** This stage of testing will be carried out when the system reaches a low level design and all prior unit tests have been passed. Multiple modules of the system will be tested together using appropriate test cases. This will confirm if various modules function effectively altogether.

**· System Testing –** This stage will be carried out when the system reaches a functional level performing the core functions it was developed for. This will ensure if specified requirements are reached and product works as a whole. The system will undergo usability testing according to the non-functional test cases.

**· Acceptance Testing –** The last stage of testing to make sure the client is happy with the product and all requirements have been achieved successfully.

## 3.1) Roles and Responsibilities

**· Unit Testing –** Must be carried out by individual developers on a continuous basis by creating unit tests throughout the development of the system.

**· Integration Testing –** May be carried out by the developers as well the team leader/testing managers by using appropriate test cases.

**· System Testing –** Must be carried out by a separate testing team under supervision of the team leader. Users testing the system are preferred to be candidates who have experience with testing systems and interfaces.

**· Acceptance Testing –** Must be carried out by separate testing team as well the team leader under supervision of the client. Requires the users to be non - developers.

## 3.2) Test Deliverables

· Test Plan

· Test Cases

· Defect/Enhancement logs

· Test Reports

## 3.3) Schedule

**Table 5. Semester 1 Schedule**

|  |  |
| --- | --- |
| **Features to test** | **Estimated Testing** |
| Video Streaming | Week 9-10 |
| Ability to switch viewing perspective | Week 9-10 |
| Both user can begin and end stream | Week 9-10 |

**Table 6. Semester 2 Schedule**

|  |  |
| --- | --- |
| **Features to test** | **Estimated Testing** |
| Annotation/Sketching | - |
| Ability to take and send picture during transmission | - |
| Hand gesture recognition and transmission | - |
| Object Recognition (piece of meat) | - |

## 

## 3.4) Risk and Contingency

**Table 7. Risks and Contingencies**

|  |  |
| --- | --- |
| **Risks** | **Contingency** |
| May be shortage of time which may result in less time for testing. | Carry out tests frequently as the system is developed instead of assigning a particular date. |
| Requirements may change during the course of development. | Keep updating requirements from client as frequent as possible to avoid last minute chaos. |
| Team member leaves | Have each module worked on and tested by 2 members. |
| Design lacks flexibility to make changes | When designing allow room for accommodating changes. |

## 3.5) Testing Tasks

· Testing scenarios shall be taken from the client and documented. The test scenarios will be required before creating the test cases.

· Test cases shall be created for individual features of the system.

· Required resources shall be accounted for before carrying out the tests.

· Bugs will be logged and reported

# 

# **4.)** **Pass/Fail Criteria**

## 4.1) Product Level

Instructor and operator can switch roles upon accepting the connected user’s request.

**Table 8. Test Case Pass/Fail Criteria**

|  |  |  |
| --- | --- | --- |
| **Testing Criteria** | **Pass Criteria** | **Fail Criteria** |
| Pairing the glasses over a wireless connection/ Bluetooth connection | Glasses are paired up with a stable wireless connection with a third device acting as a server. | Glasses don’t pair up successfully or the connection isn’t stable enough. |
| Instructor can see hand gestures on operator’s video feed. | Request sent from operator glasses and accepted by instructor glasses. | Request isn’t sent or doesn’t prompt instructor with video request. |
| Receiving and viewing operator video feed from operator glasses on instructor glasses | Video is sent with a delay of less than 1000ms and a minimum frame rate of 20. | Video sent, has loss of frames. |
| Sending hand gestures from instructor glasses | Instructor can see hand gestures on operator’s video feed. | Instructor’s video feed does not show hand gestures. |
| Receiving only hand gestures(without video) from instructor glasses and viewing them on operator video feed. | Operator can see instructor’s hand gestures on operator video feed. | Instructor’s hand gestures are not shown at all/ or not shown clearly on operator’s video stream. |
| Make sketch | Operator can capture a still image of video feed and draw a sketch on it using his/her finger. | Operator can capture a still image of video feed and draw a sketch on it using his/her finger. |
| Send sketch | Operator is able to send a saved image of a sketch to the instructor. | Unable to save sketched image or send it to instructor |
| View sketch | Instructor can download and view sketched image. | Instructor is unable to view image. |
| Switching roles | Instructor and operator can switch roles upon accepting the connected user’s request. | Instructor and operator can switch roles upon accepting the connected user’s request. |

## 4.2) Testing Stages

During each development stage test will be conducted and judged according to the following criteria

**Table 9. Testing Stage Pass/Fail Criteria**

|  |  |  |
| --- | --- | --- |
| **Testing Level** | **Pass criteria** | **Fail criteria** |
| Unit Testing | · All unit tests have been passed successfully  · At least 70% of all code written has been covered during testing.  · All bugs and errors found have been logged and been accounted for. | · Not all unit tests have been passed  · Not enough code written has been included in testing. |
| Integration Testing | · 90% of all modules developed have been tested.  · Modules perform their assigned function successfully when tested together.  · All issues have been logged and corrected | · Modules fail to carry out their function when put together.  · Too many critical issues found. |
| System Testing | · Entire System has been tested as a whole.  · 100% of all specified requirements have been successfully achieved.  · Minor Issues found have been logged and fixed.  · 100% of all system features functioning appropriately. | · Not all specified requirements have been achieved.  · Critical issues and defects found during the test.  · System features are not functioning accurately. |
| Acceptance  Testing | · When client is satisfied with the product. | · Does not achieve the requirements specified.  · Critical issues found by client.  · Does not satisfy the client. |

## 4.3) Suspension criteria and resumption requirements

### Suspension Criteria

* The design of the system is found to be defective.
* The client has changed his mind about the specified requirements.
* The module tested is found to have critical issues which need to be fixed before proceeding further.

### Resumption Requirements

* All critical issues found have been fixed.
* Modified or fixed modules have passed testing successfully.
* The new changes have gained the approval of the testing manager as well as the Team Leader.

## 4.4) Approvals

* Both the Team leader and the Testing manager must agree to the completion of a testing level before moving on to the next level.
* Any changes or addition to the system features would have to be approved by the client.