Sign Language Interpretation System QA Report

Zach Langford, Braden Bagby, Riley Hughes, David Gray, Robert Stonner

Table of Contents

Complete Test Plan

All 32 test cases made by our fellow classmates in Team 1 consists of the full testing plan that we have executed. We as a team felt that the test cases properly include most of how our software should function, apart from some functionality that we decided to get rid of. The test cases also successfully encapsulate all the requirements for our product that were included in the SRS document. The procedure to executing the following test cases was to test one test case multiple times, observe the results and try to pinpoint an issue if one appeared, then attempt to either fix the issue or discover why the issue was occurring. Finally, we would test the test case again and observe that result to be final, and document the result in this document, and if needed give an explanation as to why that was the result. Once all test cases were performed, the team would perform a method of smoke testing to validate the stability of the software. Below is the following test cases and their result.

Test Results

Test case number	**FR.1**	**Testing Results**
Test Description	When the program is running, the local environment should be able to distinguish a hand from other objects in the video feed input.	
Pre-condition	The application is running.	
Procedure		
Step 1	Launch the application and access camera	OK
Step 2	Wave hand in front of video camera	OK
Step 3	Verify hand is identified	OK

Post-condition	Program is able to distinguish a hand from other objects.	Overall: OK

Test case number	**FR. 2**	**Testing Results**
Test Description	The local environment should be able to track hand movements based off of 21 points on the wrist, palm, and fingers.	
Pre-condition	The local environment can detect a hand on the video feed.	
Procedure		
Step 1	Verify the program and environment are running	OK
Step 2	Verify the local environment can distinguish a hand from other objects	OK
Step 3	Local environment uses 21 predefined points on the wrist, palm, and fingers	OK
Step 4	The program can track the points on the hand and track hand movement	OK
Post-condition	The application can track hand movements.	Overall: OK

Test case number	**FR. 2.1**	**Testing Results**
Test Description	The local environment should output the 21 landmarks from the hand relative to a bounding box around a hand to a CSV structured file.	
Pre-condition	The local environment can detect a hand on the video feed.	
Procedure		
Step 1	Verify the program and environment are running	OK
Step 2	Verify the local environment can distinguish a hand from other objects	OK
Step 3	Local environment uses 21 predefined points on the wrist, palm, and fingers	OK
Post-condition	The application outputs the hand data to a CSV file.	Overall: OK

Test case number	**FR.3**	**Testing Results**
Test Description	The application can detect meaningful hand gestures apart from noise.	

Pre-condition	The program can detect and track hand movements.	
Procedure		
Step 1	Verify the application is running and collecting video feed.	OK
Step 2	Verify the application detects and tracks hand movements	OK
Step 3	The user can make a gesture with their hand for the application to calculate significant movements	OK
Step 4	Hand movements from small, insignificant, or unintended are distinguished from significant movements	OK
Post-condition	The application can distinguish meaningful hand gestures from noise.	Overall: OK; System will detect the hand and will not typically catch sign language gestures until one is being made

Test case number	**FR.4**	**Testing Results**
Test Description	The local environment must be able to send hand tracking data to the cloud environment.	

Pre-condition	The application must be running and collecting video feed.	
Procedure		
Step 1	The user makes a significant hand gesture in front of the video camera	OK
Step 2	The application distinguishes the hand movement from other objects	OK
Step 3	The hand movement data is collected	OK
Step 4	The data is sent to the cloud environment	OK
Post-condition	The hand movement data is successfully sent to the cloud environment.	Overall: OK; The system successfully sends the recorded data to the gesture interpreter, however since we do not use a cloud environment we took out the TCP communication.

Test case number	**FR.5**	**Testing Results**
Test Description	Verify the cloud environment is able to translate hand tracking data from hand gestures into letters and numbers.	

Pre-condition	The cloud environment has received hand gesture data collected from the video feed.	
Procedure		
Step 1	The cloud environment receives hand gesture data	OK
Step 2	The cloud environment translates the data into numbers or letters	POK
Step 3	An associated number or letter is returned from the cloud environment	OK
Post-condition	An alphabetic or numeric character is returned from the cloud environment.	Overall: POK; With the current state of the gesture interpretation model, the system will not always accurately translate the gesture.

Test case number	**FR.5.1**	**Testing Results**
Test Description	Check that the cloud environment throws an error if it is not able to translate hand tracking data from hand gestures into letters and numbers.	
Pre-condition	The cloud environment has received hand gesture data collected from the video feed.	

Procedure		
Step 1	The cloud environment receives hand gesture data	OK
Step 2	The cloud environment translates the data into numbers of letters	POK
Step 3	An associated number or letter is not returned from the cloud environment	POK
Post-condition	A default error symbol will be thrown to let the user know the gesture was not properly interpreted.	Overall: POK; This test case centers around the module interfacing with a cloud environment which we do not use.

Test case number	**FR.6**	**Testing Results**
Test Description	Verify the cloud environment is able to translate hand tracking data into words.	
Pre-condition	The cloud environment has received hand gesture data collected from the video feed.	
Procedure		
Step 1	The cloud environment receives hand gesture data	OK

Step 2	The cloud environment translates the data into letters or numbers (letters in this case)	POK
Step 3	A word is stringed together using the letters output from the data and a dictionary of sign language words	POK
Post-condition	An associated word is returned from the cloud environment.	Overall: POK; The system (gesture interpretation model) does not always accurately predict the right sign language letter, also we are not using a dictionary of sign language words, however the word is outputted.

Test case number	**FR.6.1**	**Testing Results**
Test Description	Check that the cloud environment returns an error if it is not able to translate hand tracking data into words.	
Pre-condition	The cloud environment has received hand gesture data collected from the video feed.	
Procedure		
Step 1	The cloud environment receives hand gesture data	OK
Step 2	The cloud environment translates the data into letters or numbers (letters in this case)	POK

Step 3	A word fails to be strung together using the letters output from the data and a dictionary of sign language words	POK
Post-condition	A default error symbol will be thrown to let the user know the gesture was not properly interpreted.	Overall: POK; The system will almost always output a word even if it's spelled incorrectly, it's when it reaches the formal grammar module where it will be corrected.

Test case number	**FR.7**	**Testing Results**
Test Description	Verify the system is able to understand a variety of different sign languages.	
Pre-condition	Hand tracking data is received by the cloud environment.	
Procedure		
Step 1	The user makes a gesture in a different sign language	NOK
Step 2	The cloud environment receives hand tracking data	NOK
Step 3	The gesture interpretation module interprets gesture from different sign language using a dictionary of sign language words	NOK

Post-condition	An accurate, associated word from a different sign language should be returned from the cloud environment.	Overall: NOK; The current state of the system does not interpret different sign languages.

Test case number	**FR.7.1**	**Testing Results**
Test Description	Check that the system throws an error if it is not able to understand a variety of different sign languages.	
Pre-condition	Hand tracking data is received by the cloud environment.	
Procedure		
Step 1	The user makes a gesture in a different sign language	NOK
Step 2	The cloud environment receives hand tracking data	NOK
Step 3	The gesture interpretation module fails to interpret gesture from different sign language using a dictionary of sign language words	NOK
Post-condition	A default error symbol will be thrown to let the user know the gesture was not properly interpreted.	Overall: NOK; Again, the system at this state does not interpret different sign languages.

Test case number	**FR.8**	**Testing Results**
Test Description	Verify the phrases output by the cloud environment are grammatically correct.	
Pre-condition	Hand gestures are translated from the hand gesture tracking data and letters and numbers are output.	
Procedure		
Step 1	Letters and numbers are received and translated in the cloud environment	OK
Step 2	Letters and numbers are put through a formal grammar recognition module	OK
Step 3	The module helps accurately form correct phrases	OK
Post-condition	Grammatically correct phrases from the hand gesture data are output.	Overall: OK; The grammar module will accurately translate misspelled words with about 1 to 2 letter thresholds.

Test case number	**FR.9**	**Testing Results**

Test Description	The cloud environment must be able to identify abbreviations of commands.	
Pre-condition	Letters and numbers translated from hand gesture data are received in the cloud environment.	
Procedure		
Step 1	Letters and numbers are received and translated in the cloud environment	NOK
Step 2	The data is input into the formal grammar recognition model to translate abbreviated commands	NOK
Step 3	Words and phrases are returned from the identified abbreviated commands	NOK
Post-condition	A sequence of accurate phrases and words is output.	Overall: NOK; The system will translate and interpret words, but no customization is added for abbreviations in this state.

Test case number	**FR.9.1**	**Testing Results**
Test Description	The cloud environment should return an error if it is not able to identify abbreviations of commands.	

Pre-condition	Letters and numbers translated from hand gesture data are received in the cloud environment.	
Procedure		
Step 1	Letters and numbers are received and translated in the cloud environment	NOK
Step 2	The data is input into the formal grammar recognition model to translate abbreviated commands	NOK
Step 3	Words and phrases are not returned from the identified abbreviated commands	NOK
Post-condition	An error notification will occur to inform the user that there is no valid corresponding abbreviation.	Overall: NOK; Again, the system does not interpret abbreviations of commands, but can be customized to.

Test case number	**FR.10**	**Testing Results**
Test Description	The cloud environment can translate phrases of letters and numbers into commands.	
Pre-condition	Letters and numbers translated from the hand gestures are received in the cloud environment.	

Procedure		
Step 1	Letters and numbers are received and translated in the cloud environment	OK
Step 2	A grammatically correct phrase is created from the letters and numbers	OK
Step 3	The phrase is checked against a library of key command phrases to create a command	OK
Post-condition	A command for controlling various technologies is output.	Overall: OK; When the system is given a full command, it will successfully rearrange it to a common smart home command which will then be compared to certain commands that are made.

Test case number	**FR.10.1**	**Testing Results**
Test Description	Verify that if the cloud environment can not translate phrases of letters and numbers into commands, it throws an error.	
Pre-condition	Letters and numbers translated from the hand gestures are received in the cloud environment.	
Procedure		

Step 1	Letters and numbers are received and translated in the cloud environment	OK
Step 2	A grammatically correct phrase is attempted to be created from the letters and numbers	OK
Step 3	A grammatically correct phrase fails to be created from the letters and numbers	NOK
Post-condition	An error is thrown to notify the user that there is no valid corresponding command.	Overall: POK; The system will not throw an error rather; it will simply not perform the command given.

Test case number	**FR.11**	**Testing Results**
Test Description	Verify the local environment can perform commands.	
Pre-condition	A command from hand gesture data is output from the cloud environment.	
Procedure		
Step 1	A command is translated from received hand tracking data in the cloud environment	POK

Step 2	The local environment receives the command data	OK
Step 3	The local environment performs the command on the corresponding technologies	OK
Post-condition	A pre-existing system will be able to interpret and perform the specified command.	Overall: POK; Since the system does not rely on a cloud environment, all the processes are performed locally on the server end.

Test case number	**NFR.1**	**Testing Results**
Test Description	Check to ensure the system performs the detection and translation of hand gestures into commands in real-time.	
Pre-condition	Hand gestures are received in the local environment.	
Procedure		
Step 1	The user makes a hand gesture in from of the video camera	OK
Step 2	Hand gesture data is received and put through the image processing module and formal grammar module to identify a command	OK

Step 3	Verify that a command has been output	OK
Post-condition	A command is output in real-time.	Overall: OK; The system performs its tasks such as recognition and interpretation in real time, there is a catch where when confirming a final letter is a word, the user will hold the letter for 4 seconds, which can easily be configured.

Test case number	**NFR.1.1**	**Testing Results**
Test Description	Check to ensure the system performs the detection and translation of hand gestures into commands in varying degrees of brightness.	
Pre-condition	Hand gestures are received in the local environment.	
Procedure		
Step 1	The user makes a hand gesture in front of the video camera with differing levels of brightness in the room	OK
Step 2	Hand gesture data is received and put through the image processing module and formal grammar module to identify a command	OK
Step 3	Verify that a command has been output	OK

Post-condition	A command is output in real-time with varying degrees of brightness.	Overall: OK

Test case number	**NFR.1.1a**	**Testing Results**
Test Description	Check to ensure the system throws an error if it cannot perform the detection and translation of hand gestures into commands in varying degrees of brightness.	
Pre-condition	Hand gestures are received in the local environment.	
Procedure		
Step 1	The user makes a hand gesture in from of the video camera with differing levels of brightness in the room	OK
Step 2	Hand gesture data is not received or put through the image processing module and formal grammar module to identify a command	OK
Step 3	The program is unable to output a command due to inability to view the hand gesture in a dark room	POK
Post-condition	An error is thrown to let the user know the hand gesture data was not successfully received and translated	Overall: POK; Rather than the system throwing an error, it will just keep trying to find a hand until prompted otherwise.

Test case number	**NFR.2**	**Testing Results**
Test Description	Test the accuracy of the gesture interpretation module (goal is 90% accuracy or higher.)	
Pre-condition	Hand gestures are received in the local environment.	
Procedure		
Step 1	A user makes a hand gesture in front of the camera	OK
Step 2	The hand gesture data is put through the image processing module using machine learning tools from MediaPipe	OK
Step 3	A character associated with the sign language gesture is output	POK
Post-condition	Accuracy is calculated using test samples and ratio and correct character associations.	Overall: POK; In the current state, with all alphabetic characters in the model it is about 60% accurate, however with different sets of letters, it will reach upwards of 90% accuracy. A big issue with this was our data and certain letters from our data are just simply harder to predict than others because of the quality.

Test case number	**NFR.3**	**Testing Results**
Test Description	The formal grammar module should enhance the system's ability to accurately predict words and phrases.	
Pre-condition	Hand gestures data has been received and translated into letters and numbers.	
Procedure		
Step 1	Letters and numbers are received and put into an autocomplete/search functionality using a library of predetermined, common smart home commands	OK
Step 2	A word or phrase is output after the prediction is made based from the library	OK
Step 3	Verify the prediction is accurate for the intended command	OK
Post-condition	An accurate word or phrase command is predicted.	Overall: OK; Regarding using a library, we ended up scrapping that and set up a module that will rearrange the command into a common smart home command. It is very accurate (upwards of 90%) with words that have 1 or 2 misinterpreted

		letters, however with a word that is almost unrecognizable, it will guess a different word which at that point would be something with the gesture interpreter that would need a fix.

Test case number	**NFR.4**	**Testing Results**
Test Description	Check that the hand tracking module continuously tracks the hand in view of the camera.	
Pre-condition	The video camera is picking up a live feed.	
Procedure		
Step 1	A user can make a gesture with their hand in front of the live video feed	OK
Step 2	The machine learning module that looks for a hand in the live video feed will track the hand in view	OK
Step 3	Verify that a data object is created for the hand tracking data and the video stream is uninterrupted	OK
Post-condition	A data object should be created for the hand tracking data containing position and orientation of various points on the hand.	Overall: OK

Test case number	**NFR.4a**	**Testing Results**
Test Description	Check that the hand tracking module throws an error if it does not continuously track the hand in view of the camera.	
Pre-condition	The video camera is picking up a live feed.	
Procedure		
Step 1	A user can make a gesture with their hand in front of the live video feed	OK
Step 2	The machine learning module that looks for a hand in the live video feed tries to track the hand in view	OK
Step 3	The video feed is interrupted	OK
Post-condition	An error is thrown to notify the user that the video stream has been interrupted and processes are stopped until the stream is resumed or reset.	Overall: OK

Test case number	**NFR.4.1**	**Testing Results**
Test Description	Check that the MediaPipe-Server Component slices each video frame a minimum of 20 times before it is sent over the UDP stream.	
Pre-condition	The video camera is picking up a live feed.	

Procedure		
Step 1	A user can make a gesture with their hand in front of the live video feed	OK
Step 2	The protocol handles dropped packets with a timeout on each frame	OK
Step 3	If the frame is not completely reconstructed after n milliseconds, the receiver moves on to the next frame	OK
Post-condition	The frame is sliced at least 20 times and sent over the UDP stream.	Overall: OK

Test case number	**NFR.4.1a**	**Testing Results**
Test Description	Check that the MediaPipe-Server Component receives a JPEG image stream as input via a TCP connection.	
Pre-condition	The video camera is picking up a live feed and sending image data to MediaPipe.	
Procedure		
Step 1	MediaPipe has a TCP thread listening for a new frame	OK

Step 2	A TCP socket opens and listen to save number of byte data from the data it receives	OK
Step 3	The size of the images is detected with this data	OK
Step 4	The socket is filled with incoming data after preparing a buffer of detected image size length	OK
Post-condition	The MediaPipe-Server Component receives a JPEG image stream via TCP.	Overall: OK; This functionality does work, however we are strictly taking the landmark data from the image itself for the machine learning model to interpret.

Test case number	**NFR.4.2**	**Testing Results**
Test Description	Check that the hand tracking module continuously tracks the hand movements regardless of skin tone.	
Pre-condition	The video camera is picking up a live feed.	
Procedure		
Step 1	A user can make a gesture with their hand in front of the live video feed	OK

Step 2	The machine learning module that looks for a hand in the live video feed tries to track the hand in view	OK
Step 3	Verify that a data object is created for the hand tracking data and the video stream is uninterrupted	OK
Post-condition	A data object should be created for the hand tracking data containing position and orientation of various points on the hand.	Overall: OK

Test case number	**NFR.4.2a**	**Testing Results**
Test Description	Check that the program throws an error if the hand tracking module does not track the hand movements regardless of skin tone.	
Pre-condition	The video camera is picking up a live feed.	
Procedure		
Step 1	A user can make a gesture with their hand in front of the live video feed	OK
Step 2	The machine learning module that looks for a hand in the live video feed tries to track the hand in view	OK
Step 3	The feed cannot pick up hand gesture data and a data object is not created for the hand tracking data	POK

Post-condition	An error is thrown to let the user know the hand gesture data was not picked up.	Overall: POK; The current state of the system will just continue to search for a hand in position, not throwing an error.

Test case number	**NFR.5**	**Testing Results**
Test Description	Check that the command output module is able to tailor command outputs to compatible devices.	
Pre-condition	Hand gesture tracking data is translated into commands.	
Procedure		
Step 1	Command outputs are output from the command output module	OK
Step 2	Compatible device API commands are read and compared to command outputs	OK
Post-condition	The device API outputs are matched to the command outputs.	Overall: OK

Test case number	**NFR.5.1**	**Testing Results**

Test Description	Check that the smart home device and program are connected via TCP connection over a network.	
Pre-condition	Hand gesture tracking data is translated into commands.	
Procedure		
Step 1	Command outputs are output from the command output module	OK
Step 2	A TCP connection is established to send the command string to the smart home device	NOK
Post-condition	The device is successfully connected via TCP connection and ready to receive command strings.	Overall: POK; The current state of the system will not utilize TCP since we scrapped the use of a cloud service, instead it will just communicate with the API via get/post request.

Test case number	**NFR.5.1a**	**Testing Results**
Test Description	Return an error if the smart home device and program are not connected via TCP connection over a network.	
Pre-condition	Hand gesture tracking data is translated into commands.	

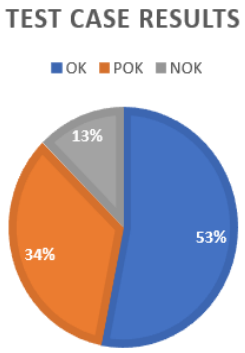
Procedure		
Step 1	Command outputs are output from the command output module	OK
Step 2	A TCP connection is not established	NOK
Post-condition	The software returns an error to the user to let them know the device did not successfully connect via TCP connection.	Overall: POK; Again, the system does not utilize TCP for sending the smart home command.

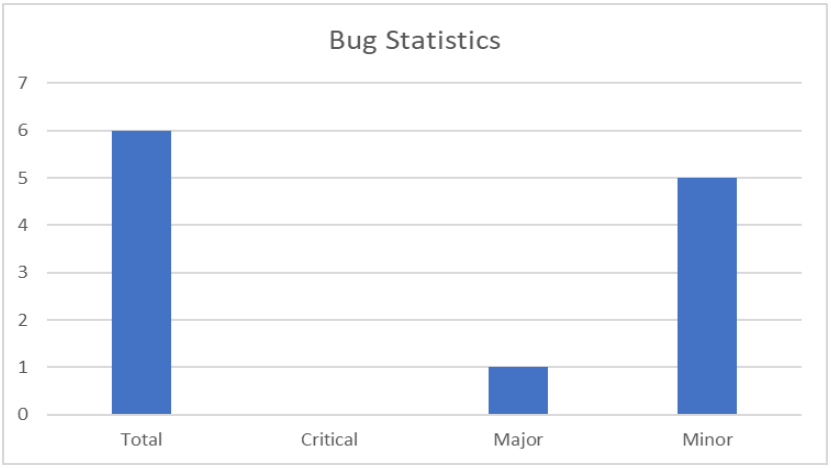
Test case number	**NFR.6**	**Testing Results**
Test Description	Ensure the hand tracking module is only able to track one hand in the video stream at once.	
Pre-condition	The video camera is picking up a live video feed.	
Procedure		
Step 1	A user can make a gesture with their hand in front of the camera	OK
Step 2	The machine learning module looks for a hand and a data object for the position and orientation of various points of only one hand is created	OK
Post-condition	Verify the hand tracking module tracks only one hand at a time.	Overall: OK

Statistical Representation of Testing Results and Bugs

Test Case Results

OK	17
POK	11
NOK	4
Total	32





Bug Overview

With the current state that the software is in, there is only one major bug and that has to with the machine learning model in the gesture interpreter. With the data we have used to train the model with all letters of the alphabet, the model has a hard time accurately predicting certain sign language letters that look like others. This issue causes the accuracy of the model to drop to roughly 60% making it very hard to test other modules of the software. However, when modifying the model to recognize letters from the data that are not causing the issue (14 out of 26 letters) the accuracy comes to an outstanding 98%. There is one minor bug with the

formal grammar module and that is if there are no prepositions in the command being signed (words like: in, the, at, etc.), the parser will sometimes remove a random noun needed for the command. The other 4 minor bugs are referring to some error handling that does not occur in the software, in the next section there is more detail referring to the summary of the testing statistics.

Overall Assessment of Test Execution and Delivered Product

- Summary: All test cases with the functionality of our camera, UDP connection with client and server, MediaPipe communication, and smart home command output work great. However, there is one major bug regarding the gesture interpretation model, a minor bug with the formal grammar module, and a few outlying minor bugs as well that hinder the overall accuracy of our software. Given the setbacks our team has had and a little more time, these bugs could be ironed out, but with the current state that the product is in we would personally not deploy it as a finished product quite yet.
    - Details: As mentioned above in the Bug Overview section, we have one major bug that has plagued our software and that is the quality of the data used to train our gesture interpretation model. This bug is significant in that it brings the accuracy of our product down to around 60% when using all of the letters of the alphabet which in turn does not meet our requirement of an accuracy of 90% or higher. For demonstration, testing, and quick use purposes we have found a work around, with training a model that uses 14 of the 26 letters of the alphabet with clean data we can create a model with upwards of 90% accuracy. There is also a minor bug with the formal grammar module where if a command with no prepositions is sent, sometimes a needed noun for the smart home command will be parsed out (words such as: light, front, door, etc.). This bug only occurs when there are no prepositions in the command (in, the, at, etc.), when a command is sent with at least one preposition it works flawlessly. The other outlying bugs are simple error handling messages that do not get sent when they should (Test Cases: NFR 4.2a, NFR 1.1a). Another point that should be mentioned is the amount of change that this product has gone through. Some of the test cases do not accurately account for the changes with the main change being that we got rid of the implementation of a cloud-based service. All the functionality of the gesture interpretation model and formal grammar module are done on the server side, so there is no need for a TCP connection for the two to interface with each other. These changes are what caused for a lot more test cases to be partially okay (POK) since they were centered around ideas that we scrapped. Like previously mentioned, with some setbacks like having to create a new machine learning model for our gesture interpretation module, the quality of the data we had access to, had we caught these setbacks earlier or started with the

functional implementation we have now, we would have been presenting a rounded out product. However, setbacks are always going to happen, and we are still very proud of what we have developed.