



CECS 491A
Requirement & Functional Specification
Khanh Nguyen (Leader)
Angel Franco
Christopher Imantaka
Ryan Valdriz

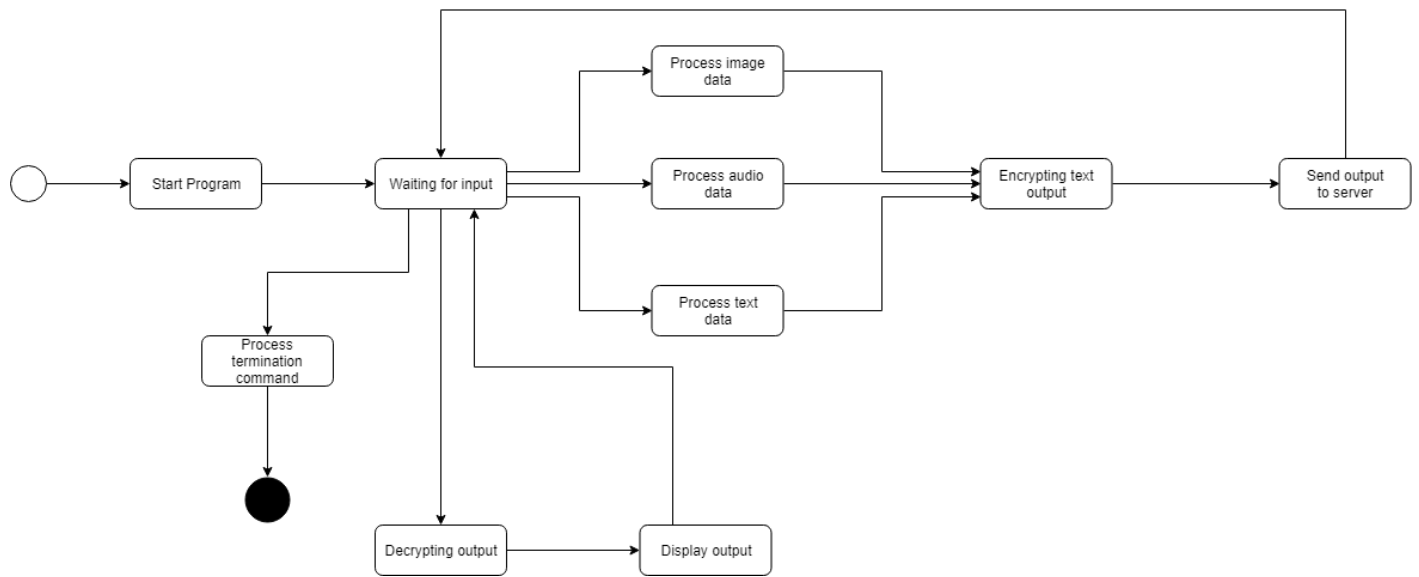
Table Of Content

Purpose.....	3
State Diagram.....	4
Sequence Diagram.....	5
Functional Specification.....	6
1 Communicate By Texting.....	6
2 Video Chat.....	8
3 ASL to Text.....	11
4 ASL to Audio.....	14
5 Audio to Text.....	18
Non-Functional Specifications.....	21

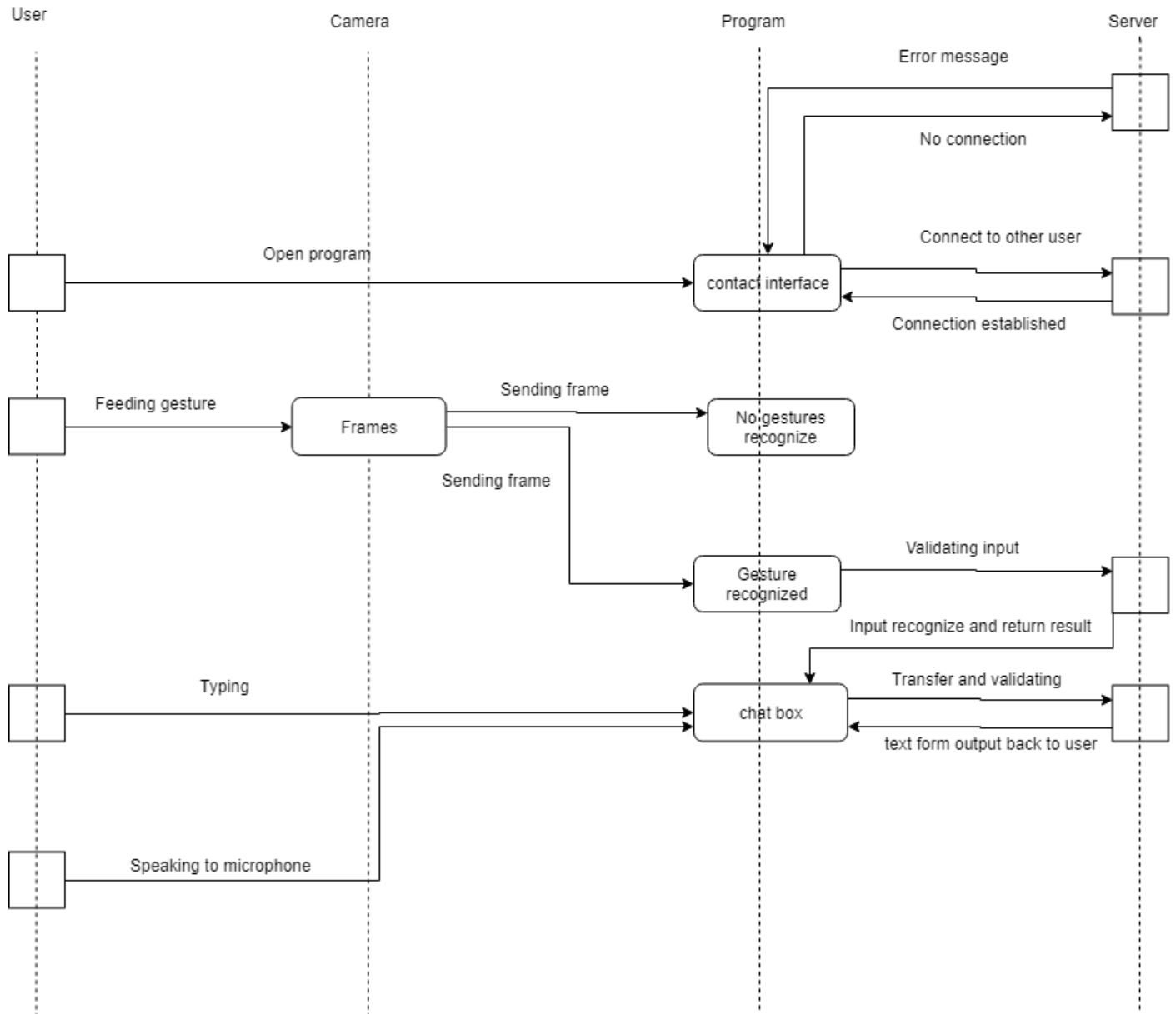
Purpose:

In this document our team stated the functional and non-functional specifications of our project. Each functional specification will go into detail about the operations our application will have to perform. It will discuss all of the required functions and any correlating information. Such information will include inputs, outputs, and any references each function will have to make. The non-functional specifications will discuss what our application will require. These requirements will include hardware and software specs. This document will also provide diagrams to show how our application will work.

State Diagram



Sequence Diagram



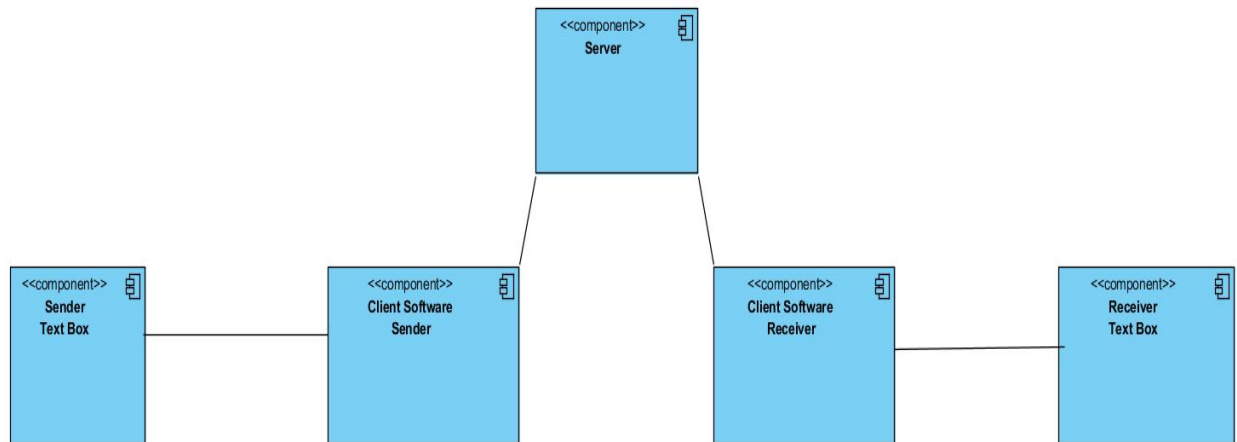
Functional Specification 1: Communicate by texting

1. User communicate with each other by texting

a. Process Description

- i. The user chooses another user to communicate with. The user types in their message in a message box. The receiving user gets the message by the sender as it is being typed in the box.

b. Diagram



c. Functional Definitions

i. Functional Definition 1

1. Name of Function: send_text
2. Component: Client Software Sender
3. Input: message/plaintext
4. Operation: This function will encrypt the string inputted by the user, then send the ciphertext to the server. This function will also communicate with the server using TCP.
5. Constraints:
 - a. If the user cannot connect to the server, the app will ask the user whether they want to try connecting again
6. Output: encrypted text
7. Data: String

ii. Functional Definition 2

1. Name of Function: receive_text
2. Component: Client Software Receiver
3. Input: ciphertext

4. Operation: This function will use TCP to connect to the server. Then it will receive messages, meant for the client, from the server. The function will then proceed to decrypt the message.
5. Constraints:
 - a. If the user cannot connect to the server, the app will ask the user whether they want to try connecting again
6. Output: Plaintext
7. Data: String
- iii. Functional Specification 3
 1. Name of Function: get_input
 2. Component: Sender Text Box
 3. Operation: This function will ask the user to type a message and save that message.
 4. Output: User inputted string
 5. Data: String
- iv. Functional Specification 4
 1. Name of Function: display_text
 2. Component: Receiver Text Box
 3. Input: Plaintext
 4. Operation: Display string to receiver
 5. Output: Display plaintext on the monitor
 6. Data: String
- d. References
 - i. None
- e. Data Dictionary
 - i. Text
 1. Data Type
 - a. String
- f. Test & Validation for each process
 - i. Tests
 1. Check if you can send texts to the server
 2. Check if you can receive texts from the server
 - ii. Validations
 1. Passes if you can send a text to the server
 2. Passes if you can receive a text from the server

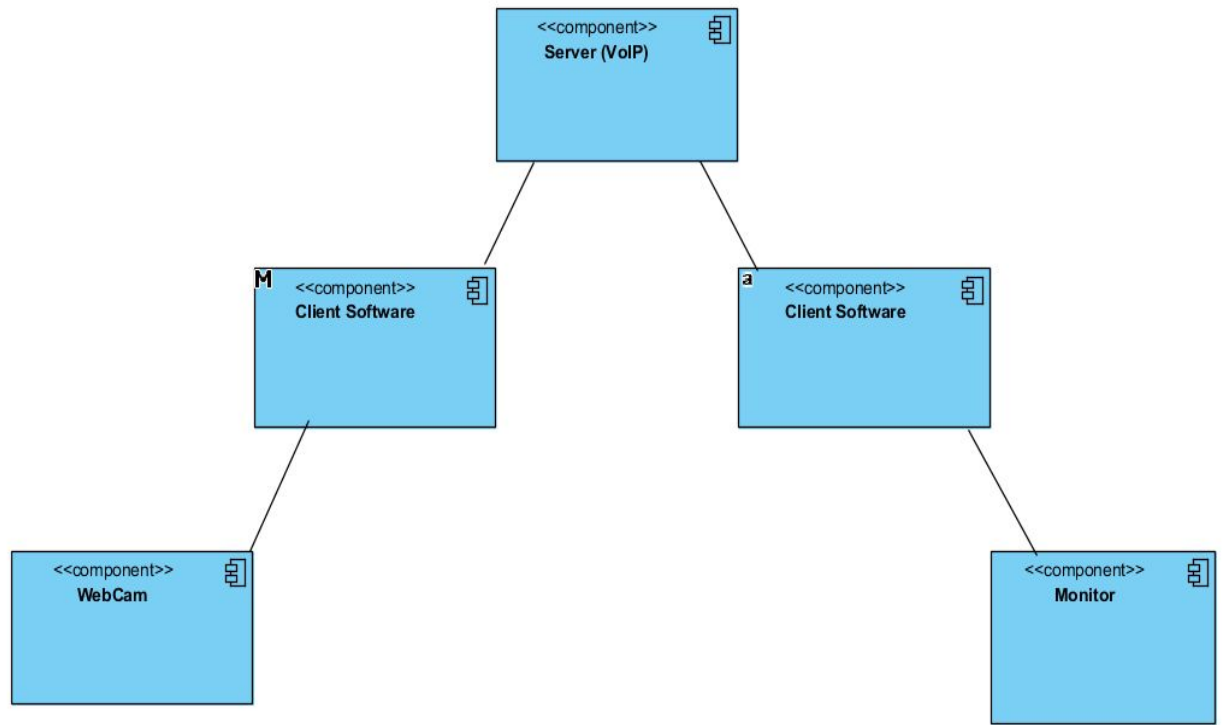
Functional Specification 2: Video Chat

1. Users view, hear, speak to each other in real time.

a. Process Description

- i. In real time user can hear and see each other. If a user talks the user on the opposite end should hear what he or she said without delay. Also any movement done by the user should be clearly visible to the receiving user.

b. Diagram



c. Functional Definitions

i. Functional Definition 1

1. Name of Function: get_video_data
2. Component: Webcam
3. Operation: Constantly gets source data from webcam and encodes it into a bitstream until video ends
4. Output: bitstream
5. Data: jpeg, mp3

ii. Functional Definition 2

1. Name of Function: send_video_data
2. Component: Client Software
3. Input: video bitstream

4. Operation: This function will connect the client to the server using TCP. It will then continue sending video data to server until the session terminates
 5. Constraints:
 - a. If the user cannot connect to the server, the app will ask the user whether they want to try connecting again
 6. Output: video bitstream
 7. Data: jpeg, mp3
- iii. Functional Definition 3
1. Name of Function: receive_video_data
 2. Component: Client Software
 3. Operation: This function will connect the client to the server using TCP. It will then receive video datastreams from the server that are meant for the client.
 4. Constraints:
 - a. If the user cannot connect to the server, the app will ask the user whether they want to try connecting again
 5. Output: video bitstream
 6. Data: jpeg, mp3
- iv. Functional Definition 4
1. Name of Function: display_video
 2. Component: Client Software
 3. Operation: Displays video content on screen
 4. Output: video
 5. Data: jpeg, mp3
- d. References
- i. ASL Dataset
 - ii. TensorFlow API and OpenCV API
- e. Data Dictionary
- i. Picture
 1. Data type
 - a. Jpeg
 - b. label/string
 - c. Rgb
 - ii. Audio
 1. Data Type
 - a. mp3

f. Test and Validation

i. Tests

1. Check if webcam is recording
2. Check if program can access webcam recording
3. Check if program can pass data to server
4. Check if program can receive data from server
5. Check if program can display the received data

ii. Validations

1. Passes if we can prove webcam is recording data
2. Passes if program can access webcam data with our program
3. Passes if we see the data being stored on our server
4. Passes if program can receive data from server
5. Passes if program can output the data on the monitor

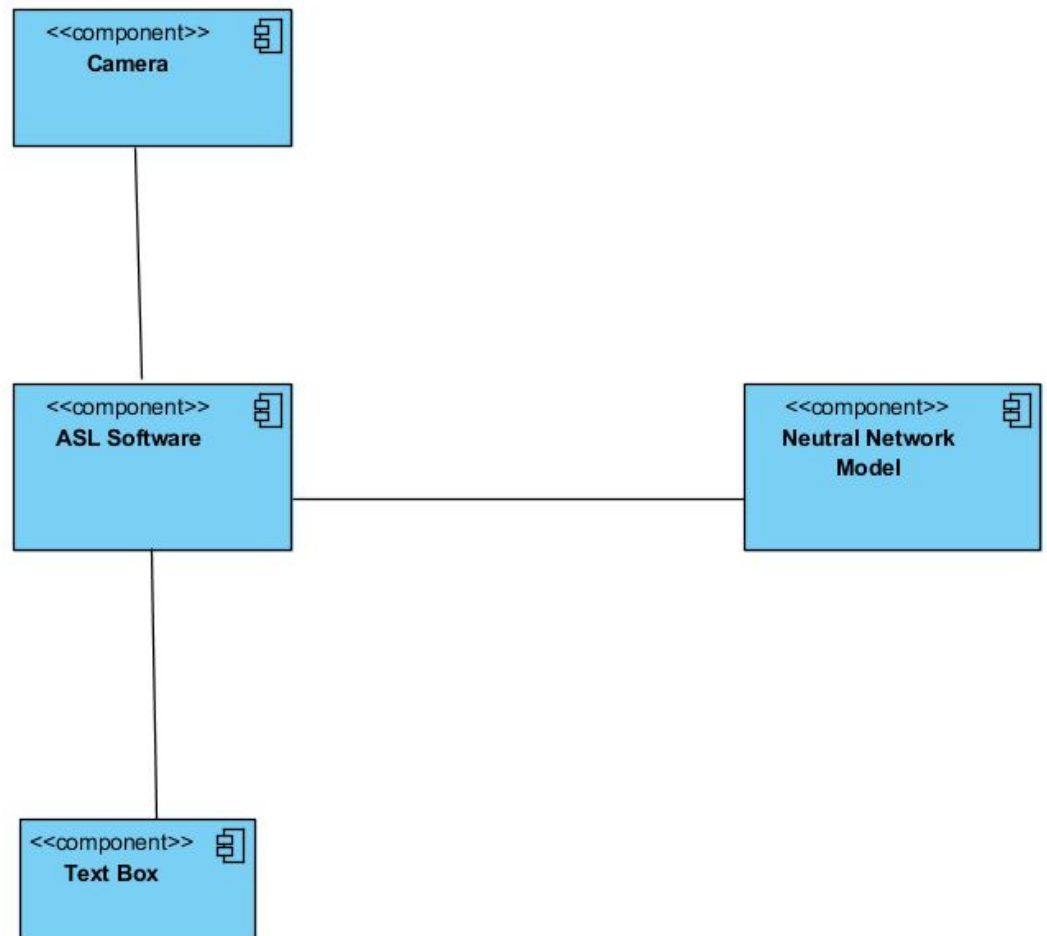
Functional Specification 3: ASL to Text

1. Take in ASL input and outputs it as english in text

a. Process Description

- i. The user presents ASL hand gestures. The webcam on a computer scans the ASL hand gestures and provides it as an input to the software. The software will use the input and compare it to our ASL dataset. The result of this mapping will be text displayed for the ordinary person to read.

b. Diagram



c. Functional Definitions

- i. Functional Definition 1
 1. Name of Function: `get_current_frame`
 2. Component: Camera
 3. Input: Light

4. Operation: The camera takes video and the frames will be pass to ASL Software to be process
 5. Constraint:
 - a. The video will take in video in frames
 - b. The frames will be converted into matrix where each elements will be the value of the pixel
 - c. Each frame will be pass individually
 6. Output: Image
 7. Data: Picture(pixel)
- ii. Functional Definition 2
1. Name of Function: process_image
 2. Component: ASL Software
 3. Input: Picture(frame) in form of matrix
 4. Operation:
 - a. The ASL will process the image in a way that could be compared with the Neural Network Model
 - b. After it is done being compared with the neural network, it will receive a character and it will write that character in the text box
 5. Constraint:
 - a. The ASL software will process the matrix into a fully connected network(data) that could be compared to the neural network model
 - b. The ASL software will received a character of which the fully connected network is closely similar to
 - c. If the neural network return a NULL, then no character shall be written in the text box
 6. Output: Text
 7. Data: Text (String), fully connected network(matrix)
- iii. Functional Definition 3
1. Name of Function: compare_with_model
 2. Component: Neural Network Model
 3. Input: fully connected network
 4. Operation: The fully connected network will be compared to all the model that is in the neural network model, and it will return a character that is closely similar to in the model
 5. Constraint:

- a. The model has to have 80% and above a character. Otherwise it will return a Null
 - 6. Output: Character/alphabet
 - 7. Data: fully connected network(matrix), Character/alphabet (string)
- iv. Functional Definition 4
 - 1. Name of Function: write_text
 - 2. Component: Text
 - 3. Input: Text
 - 4. Operation: The text is passed by ASL model after comparing with the neural network model. The text will be written in the text box of the user
 - 5. Constraint:
 - a. The user should have the option to modify the text in the text box
 - 6. Output: Text
 - 7. Data: Text(string)
- d. References
 - i. ASL Dataset
 - ii. TensorFlow API and OpenCV API
- e. Data Dictionary
 - i. Picture
 - 1. Data type:
 - a. Matrix of pixel
 - ii. Text
 - 1. String
- f. Test and Validation
 - i. Tests
 - 1. Check if it recognizes our hands
 - 2. Check if it recognizes ASL hand gestures
 - 3. Check if it translates ASL properly into English text
 - 4. Check if it properly outputs the translation as text
 - ii. Validations
 - 1. Passes if it recognizes our hands
 - 2. Passes if it recognizes ASL hand gestures
 - 3. Passes if it translates ASL properly into English text
 - 4. Passes if it properly outputs the translation as text

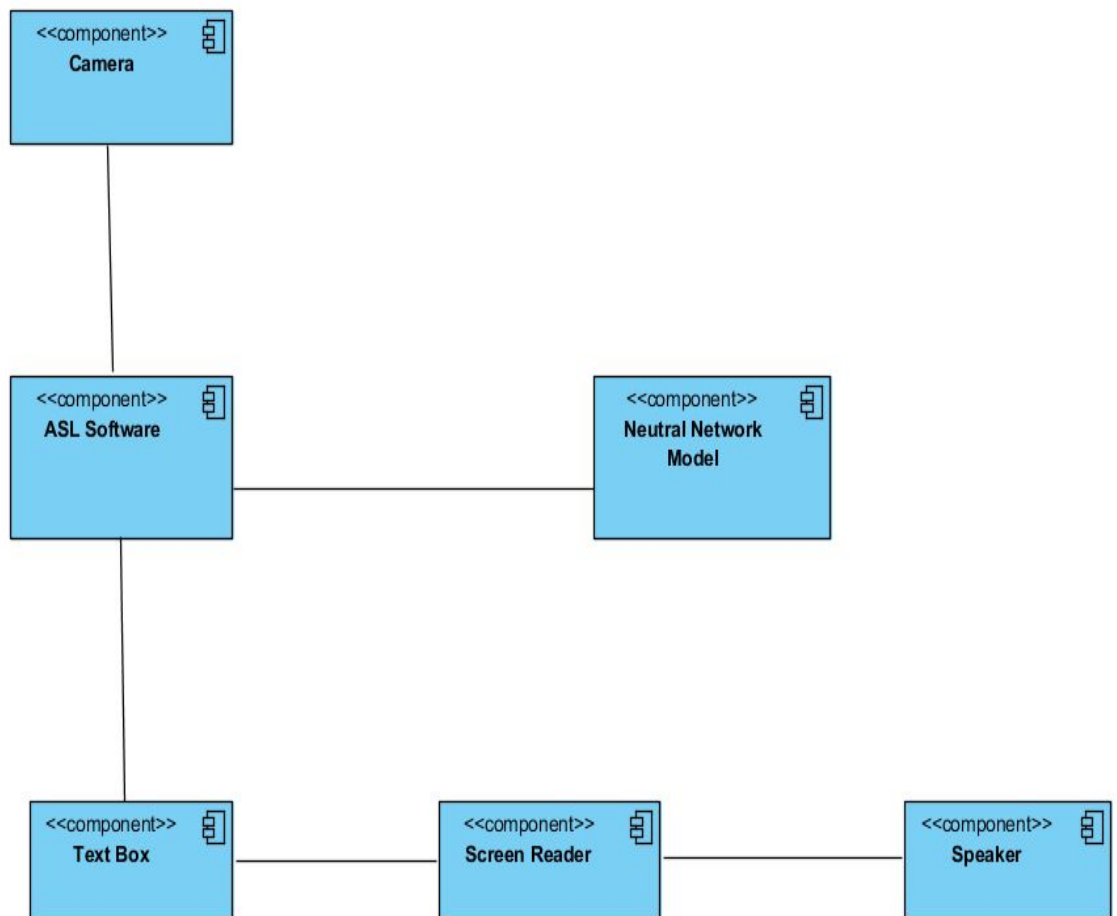
Functional Specification 4: ASL to Audio

1. Take in ASL input and outputs it as english audio output

a. Process Description

- i. The user presents ASL hand gestures. The webcam on a computer scans the ASL hand gestures and provides them as input to the software. The software will use the input and compare it to our ASL dataset. The result of this mapping will be text displayed, our computer will then read out loud the text displayed for the user to hear.

b. Diagram



c. Functional Definitions

i. Functional Definition 1

1. Name of Function: get_current_frame
2. Component: Camera
3. Input: Light
4. Operation: The camera takes video and the frames will be pass to ASL Software to be process

5. Constraint:
 - a. The video will take in video in frames
 - b. The frames will be converted into matrix where each elements will be the value of the pixel
 - c. Each frame will be pass individually
6. Output: Image
7. Data: Picture(pixel)

ii. Functional Definition 2

1. Name of Function: process_image
2. Component: ASL Software
3. Input: Picture(frame) in form of matrix
4. Operation:
 - a. The ASL will process the image in a way that could be compared with the Neural Network Model
 - b. After it is done being compared with the neural network, it will receive a character and it will write that character in the text box
5. Constraint:
 - a. The ASL software will process the matrix into a fully connected network(data) that could be compared to the neural network model
 - b. The ASL software will received a character of which the fully connected network is closely similar to
 - c. If the neural network return a NULL, then no character shall be written in the text box
6. Output: Text
7. Data: Text (String), fully connected network(matrix)

iii. Functional Definition 3

1. Name of Function: compare_with_model
2. Component: Neural Network Model
3. Input: fully connected network
4. Operation: The fully connected network will be compared to all the model that is in the neural network model, and it will return a character that is closely similar to in the model
5. Constraint:
 - a. The model has to have 80% and above a character. Otherwise it will return a Null
6. Output: Character/alphabet

7. Data: fully connected network(matrix), Character/alphabet (string)

iv. Functional Definition 4

1. Name of Function: write_text
2. Component: Text
3. Input: Text
4. Operation: The text is passed by ASL model after comparing with the neural network model. The text will be written in the text box of the user
5. Constraint:
 - a. The user should have the option to modify the text in the text box
6. Output: Text
7. Data: Text(string)

v. Functional Definition 5

1. Name of Function: process_text_to_audio
2. Component: Screen reader software
3. Input: Text
4. Operation: Get text from the text box, process the text, and output an audio data that is passed on to the output_audio
5. Constraint:
 - a. The user have the option to read message out loud or not.
 - b. The text will be processed as a whole and the output will be a sequence of frequency of the whole text in the text box
 - c. The digital audio will be pass as a whole to the speaker
6. Output: Digital audio (frequency)
7. Data: Text (String), digital audio(frequency)

vi. Functional Definition 6

1. Name of Function: output_audio
2. Component: Speaker
3. Input: digital audio (frequency)
4. Operation: Get an audio data from process_text_to_audio and output a sound through the speaker
5. Constraint:
 - a. The speaker will not spell out each character but it will be able to spell out each word
6. Output: Audio (soundwave)
7. Data: Audio (soundwave)

d. References

- i. ASL Dataset
 - ii. TensorFlow API and OpenCV API
- e. Data Dictionary
 - i. Picture
 - 1. Data type
 - a. Rgb
 - b. Pixel (matrix)
 - ii. Text
 - 1. Data Type
 - a. String
 - iii. Audio
 - 1. Data Type
 - a. Frequency
 - b. Soundwave
- f. Test and Validation
 - i. Tests
 - 1. Check if it recognizes our hands
 - 2. Check if it recognizes ASL hand gestures
 - 3. Check if it translates ASL properly into English text
 - 4. Check if it properly reads the text aloud
 - ii. Validations
 - 1. Passes if it recognizes our hands
 - 2. Passes if it recognizes ASL hand gestures
 - 3. Passes if it translates ASL properly into English text
 - 4. Passes if it properly reads the text aloud

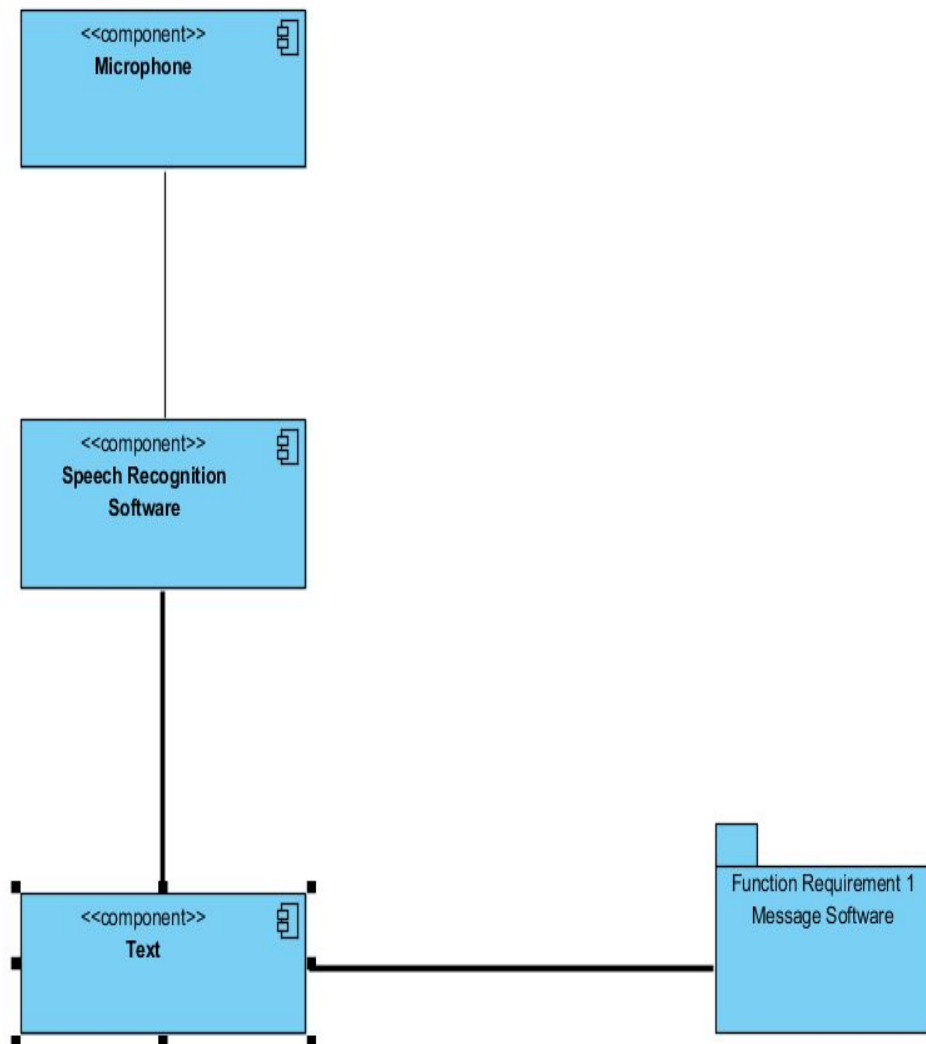
Functional Specification 5: Audio to Text

1. Take audio input and outputs text

a. Process Description

- i. The user talks to the computer. The mic on a computer interprets the audio input and uses the data as an input to voice recognition software. The voice recognition software produces the text version of the audio input.

b. Diagram



c. Functional Definitions

- i. Functional Definition 1
 1. Name of Function: get_audio
 2. Component: Microphone

3. Input: soundwave
4. Operation: Get an audio input from the user, convert it into digital data, and passed it to the Speech Recognition software
5. Constraint:
 - a. The digital will be in the form of frequency
 - b. The digital data will be pass in frames of millisecond
 - c. The user will have the option to turn on microphone to able to use this functionality
6. Output: audio
7. Data: Audio (soundwave), Digitize audio (frequency)

ii. Functional Definition 2

1. Name of Function: process_audio
2. Component: Speech Recognition Software
3. input: Digitize audio(frequency)
4. Operation: Process the audio and translate it into text to the text box
5. Constraint :
 - a. The audio will be concatenated together
 - b. Once the process able to recognize a word or character, it will pass that output to the textbox
6. Output: text
7. Data: Audio(mp3)

iii. Functional Definition 3

1. Name of Function: send_message
2. Component: Function Requirement 1
3. Input: Text
4. Operation: Send encrypted message to the server
5. Constraint:
 - a. The message will have to go through an encryption process
 - b. The user will have a button that they could click on to send the message
 - c. User are able to edit the message in the box before they click the button to send
6. Output: text(encrypted)
7. Data: Text (string),

d. References

- i. None

e. Data Dictionaries

i. Audio

1. Data Type

- a. Soundwave
- b. Frequency

ii. Text

1. Data Type

- a. String

f. Test and Validation

i. Tests

- 1. Check if the application can pick up audio
- 2. Check if it can recognize English speech
- 3. Check if it can translate English speech to text

ii. Validation

- 1. Passes if the application can pick up audio
- 2. Passes if it can recognize English Speech from audio input
- 3. Passes if it can translate English speech to text

Non-Functional Specifications

1. Operating System

- a. Windows 10
- b. Latest mac OS

2. Camera

a. Light exposure

- i. The background on which the camera is facing should not face a strong source of light while detecting the object.

b. Min Requirements

- i. 720p HD resolution
- ii. Video rate of 30 FPS
- iii. Standard Lens

The camera lens have to be a standard lens if any non-standard lens is used, then it will disrupt the image processing task of the machine.

3. CPU

a. Processor

Min. Requirements 2 GHZ

4. RAM

a. Min. Requirements 4GB

5. Graphics Card

a. Min. Requirements

Intel HD Graphics 4000 graphics with 384 MB of shared DDR3 memory. This will not affect the performance of the machine but it will affect the when training the machine.