ENGR 390

Software Design Application - Sign Sound See
Scrum Report V.1

Submitted to :

Dr. Kharma

October 15th, 2017

Team Name: The Best

By:
Cameron Hall - 40004827
Yazan Odeh - 26556191
Yousef Abo El Foul - 27199120
Nicholas Gattuso - 40007087

Concordia University

ABSTRACT

This report discusses the ENGR 390 course project to designing a glove that translate sign language via an Android application. The report follows the construction of the glove and application from initial design and research, to components and material selection, to construction and prototyping. It includes calculations on our chosen design using quantitative and qualitative analysis such as pros and cons, and a decision matrix.

TABLE OF CONTENTS

1. INTRODUCTION

American Sign Language is a natural language and is the main sign language of the Deaf and impaired communities in the United States and most of Canada. English words are often borrowed through fingerspelling, which is the representation of the letters of a writing system using only the hands. This alphabet has often been used in deaf education, mainly that of young children.

Figure 1 illustrates what is known as the American Manual Alphabet.

The purpose of this project is to design a glove that will be able to translate fingerspelling via an Android application with the goal of being used as a tool for deaf education.

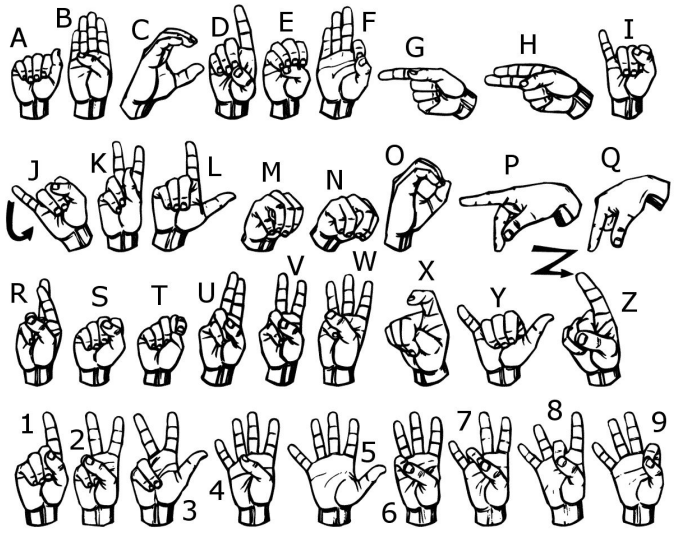

Figure 1: American Manual Alphabet [1]

2. Prescrum Assignment

2.1 Brief Project Description

We have decided to structure our requirements into two sections based on the two main components of our project (the glove and the application). This will enable us to better identify the project's requirements as our product involves many moving parts.

Glove requirements:

Our requirements for the glove consist of having it recognize gestures and hand movements. The glove should also be able calibrate itself for each user. The glove should have the ability to recognize the bending of each finger in relation to the other fingers in order to then associate the series of bent fingers to a standard american sign language letter. Finally, the glove should be able to transmit this information to the app.

Application requirements:

The application should prompt the user to undergo a calibration setup for the glove. The app will store the calibration information for each individual user. The app will receive and store the letters transmitted by the glove. From there, the application should output the letters by displaying them as text and voicing them as sound. The app will include a tutorial section, which will act as a reference to teach the user the American Sign Language alphabet. Finally, the app will have a settings section that will enable the user to recalibrate the glove in the case where it was originally miscalibrated. The settings will also enable the user to choose an output preference (voice or text).

2.2 Alternative Approaches

The first approach we are considering for the glove is for it to be able to recognize gestures and hand movements with the use of flex sensors. The flex sensors will be attached to each finger and will be connected to an Arduino. The resistance in flex sensors change based on the degree at which they are bent. This information can then be sent to the Arduino where it will be used to encode the bending of each finger and translate it into a letter. The second approach we are considering for the glove gesture recognition is to have sensor points positioned at the tip of each finger. The sensor points will be tracked by a Kinect camera (motion sensing input device) connected to an Arduino. The camera will send this information to the Arduino. where it will once again be used to encode the position of each finger and translate it into a letter. To transmit the lettering information to the app, we are considering using a bluetooth module , which can be attached to the Arduino as an add-on. Another method to transfer information is to use an Android USB cable that will link the Arduino directly to the user's Android phone. Upon startup, the app should prompt the user to undergo a calibration setup for the glove. For this setup, we want to test and record the range of the user's hand flexibility. The user will undergo a specific set of gestures, which will in turn, be recorded as parameters within the Arduino's algorithm. The calibration results could either be saved in the app's database within a user's profile or it could simply store the calibration results locally within the app. Once the calibration has been completed, the app will be able to receive the letters transmitted by the glove. It will then, display and/or voice the letters. We can either voice the letters after every gesture or after a set of letters are written with the glove in which case the output would be a word.

2.3 Evaluation of Alternatives

2.3.1 Quantitative Pros and Cons

Feature A.1: Kinect Camera

Pros	Cons
Simple implementation	Low accuracy
	Low mobility

Feature A.2: Flex Sensors

Pros	Cons
High accuracy	Complex implementation
High mobility	

Feature B.1: Bluetooth connectivity

Pros	Cons
High mobility	Requires Arduino integration

Feature B.2: USB connectivity

Pros	Cons
No hardware integration required	Low mobility

Feature C.1: Local Calibration

Pros	Cons
Database not required	Calibration required on every startup

Feature C.2: Calibration Database

Pros	Cons
Calibration once per user profile	Requires the implementation of a database

Feature C.1: Word Voicing

Pros	Cons
Will output complete words	Cannot correct mistakes
	Requires the implementation of a database

Feature C.2: Letter Voicing

Pros	Cons
Database not required	Will only output letters
Corrections not needed	

2.3.2 Decision Matrix

Primary design choices predetermined based off of preferences. This will reduce the size of the decision matrix by eliminating the weaker design combinations.

Designs:

Design 1: Flex sensors, bluetooth, local calibration, letter voicing

Design 2: Flex sensors, bluetooth, local calibration, word voicing

Design 3: Flex sensors, bluetooth, calibration database, word voicing

Comparison Criteria	**Weight**	**Design 1**		**Design 2**		**Design 3**	
		Score	**Result**	**Score**	**Result**	**Score**	**Result**
Usability	**4**	**2**	**2 /4**	**3**	**3/4**	**4**	**4/4**
Speed	**3**	**3**	**3/3**	**3**	**3/3**	**3**	**3/3**
Simplicity of design	**2**	**2**	**2/2**	**1**	**1/2**	**0**	**0/2**
Total	**10**	**7**	**7/10**	**7**	**7/10**	**7**	**7/10**

As can be seen by the decision matrix, these primary design choices scored equally across all comparison criteria. This is caused by the relationship between user usability and simplicity of design. In other words, as the usability increases, the simplicity of the design equally diminishes, and vice-versa.

2.4 Recommendation

The main issue we encountered is the trade-off between the usability of our product and the simplicity of its design. Although usability is our personally prefered criteria, the more the design becomes more complex. A complex design requires more time, which in turn, increases the risk of not meeting the product's completion deadline. This is the reason why it was scored equally within our comparisons, the higher the risk the greater the reward. With all that said, we have decided that we will attempt to implement the more complex design (design 3). In the worst case scenario, where we believe that we will not make the product's deadline, we have the option to pivot to one of the simpler designs as there designs are similar to Design 3.

3. CONCLUSION

In conclusion, we have decided on the third design to implement for the sign language application and glove. We detailed our project requirements and features. We have also, explored various potential design features. We have evaluated the primary design selections through analysis techniques such as a quantization of pros and cons as well as a decision matrix.

4. REFERENCES

[1]Bill, "Sign Language", Lifeprint.com, 2017. [Online]. Available: http://lifeprint.com/asl101/topics/wallpaper1.htm. [Accessed: 15- Oct- 2017].