**EUROPEAN UNIVERSITY OF LEFKE**

FACULTY OF ENGINEERING

Graduation Project 2

Converting American Sign Language to words using Machine Learning

### Abdulganiy Lawal

### 20140002

**The project I created turns ASL (American Sign Language) into words. The main goal was to develop an app that can assist those who are hard of hearing or deaf in communicating with those who do not understand sign language.**

**A potential use case to this app is using it to communicate with hard of hearing individual in an health care system.**

**Supervisor**

Dr.Vesile Evrim

Publish Date

13/06/2023

**Table Of Contents**

[Your Name Surname i](#_Toc58713322)

[Your Student Number i](#_Toc58713323)

[Your Supervisor Name i](#_Toc58713324)

[Publish Date i](#_Toc58713325)

[1.Introduction 1](#_Toc58713326)

[1.1 Problem definition 1](#_Toc58713327)

[1.2 Goals 1](#_Toc58713328)

[2. Literature Survey 2](#_Toc58713329)

[3. Background Information 3](#_Toc58713330)

[3.1 Required & Used software 3](#_Toc58713331)

[3.2 Other software 3](#_Toc58713332)

[3.3 Hardware 3](#_Toc58713333)

[4. Design Documents 4](#_Toc58713334)

[4.1 Data flow diagram 4](#_Toc58713335)

[4.2 Your Context Diagram 4](#_Toc58713336)

[5. Methodology 5](#_Toc58713337)

[6. Conclusion 6](#_Toc58713338)

[6.1 Benefits 6](#_Toc58713339)

[a. Benefits to users : 6](#_Toc58713340)

[b. Benefits to me : 6](#_Toc58713341)

[6.2 Ethics 7](#_Toc58713342)

[6.3 Future Works 8](#_Toc58713343)

[7. References 9](#_Toc58713344)

# 

# 1.Introduction

## 1.1 Problem definition

**The purpose of this project is to develop an app that translates sign language to English words allowing deaf individual to communicate in a better way.**

**Some use cases of the app are:**

* **Using it in a hospital, where physicians and nurses might use the app to interact with deaf or hard of hearing individuals.**
* **Another use for my app is in the classroom where teachers can use the app to communicate with students that are hard of hearing.**

**It will be useful to facilitate communication and understanding with individual who have hearing disabilities.**

**The working logic of my app are:**

**Example-Problems :**

* **First the user will search for a friend and the add a friend**
* **The user will now click on any of his friend to create a chat session**
* **Now in the chat session, It is divided into the backt camera and a messaging view**
* **From the front camera the user can start signing**
* **The camera will then use an object detection model to identify the individual sign language and gestures being used in the video.**
* **Once the sign has been identified, the model will now send the sign word in English to the messaging view so that it can be sent to the friend users device.**

## Goals

* **To give people who are hard of hearing or deaf a way of communicating with those who do not understand asl.**
* **To promote inclusivity and acceptance among people with different level of communication ability**

# 2. Literature Survey

**They are a couple of application that are similar to mine but still different. Here are some of them:**

* **Hand Talk Translator**
* **Signily**

**Signily: This is a keyboard app that translate ASL into text as the user types. It uses machine learning algorithim to recognised to recognized asl signs and translate them into text[1]. It is available for ios and android devices. Here is a comparison between my app and signily.**

* **Supported Languages: Both apps will support American sign language**
* **Supported Devices: Signily works on ios and android while my app only works on ios**
* **Output mode: We both will use it in a message like setting, but signily is a keyboard app, while mine is a full fledge messaging app**
* **Input mode: Signily uses a keyboard to take input while my app uses the camera**

**Hand Talk Translator: The hand talk app automatically translates text and audio to American sign language(ASL) and Brazillian Sign Language (LIBRAS) through artificial intelligence[2]. Here is a comparison:**

* **Supported Language: My app supports only ASL while hand talks suppoth both ASL and LIBRAS**
* **Supported Devices: HandTalk works for both ios and android, my app will be specifically design for ios Device.**
* **Output Mode: My app output is a messenger while handtalks uses a 3d animated character that signs.**
* **Input mode: Handtalk uses the keyboard or audio has a way to collect information, my app uses computer vision to take signs from the user.**

# 3. Background Information

## 3.1 Required & Used software

* **Xcode :**

**A code editor to write swift and compile the code into a app**

* **Swift and Swift UI :**

**The programming language that will be use to develop the app**

* **Image Recognition library and compter vision libraries:**

**To capture and analyse frames of the user signing. The library that was use is the Vision Framework**

* **Create ML:**

**To train and deploy the asl recognition model**

* **StreamChat**

**This was use as a library for sending and receiving users messages**

* **Firebase**

**This BAAS (Backend as a service) was used to store users information and authentication**

## 3.2 Other software

* **BitBucket:**

**For design icons.**

## 3.3 **Hardware**

## IOS Device:

**To capture the user signs**

* **Computer**

**To train the machine learning algorithim**

# Methodology

**In order to accomplish this app, I used several technologies such as a Roboflow firebase, stream Chat SDK. Let’s go into their details**

**Authentication**

**So firebase was used for this part. Firebase is a Backend as a service SDK, I used firebase to authenticate the user so that it can be secured and I can have a record of all the usere who are using my app so that the user can add other user for communicating with them**

**Storage**

**The users has several information to store so for this I use firebase firestore. Cloud Firestore is a flexible, scalable database for mobile, web, and server development from Firebase and Google Cloud. Like Firebase Realtime Database, it keeps your data in sync across client apps through realtime listeners and offers offline support for mobile and web so you can build responsive apps that work regardless of network latency or Internet connectivity. Cloud Firestore also offers seamless integration with other Firebase and Google Cloud products, including Cloud Functions [4]. I am storing the user ‘s name, username , email , gender and their friends collection.**

**RoboFlow**

**Roboflow empowers developers to build their own computer vision applications, no matter their skillset or experience. We provide all of the tools needed to convert raw images into a custom trained computer vision model and deploy it for use in applications. Today, Roboflow supports object detection and classification models. I used it for annotation and labelling the image I used to train my machine learning model.[5]**

**CreateML**

**Experience an entirely new way of training machine learning models on your Mac. Create ML takes the complexity out of model training while producing powerful Core ML models. [6] I took the annoted file from roboflow and I used create ml to train my ml model.**

**Vision Framework**

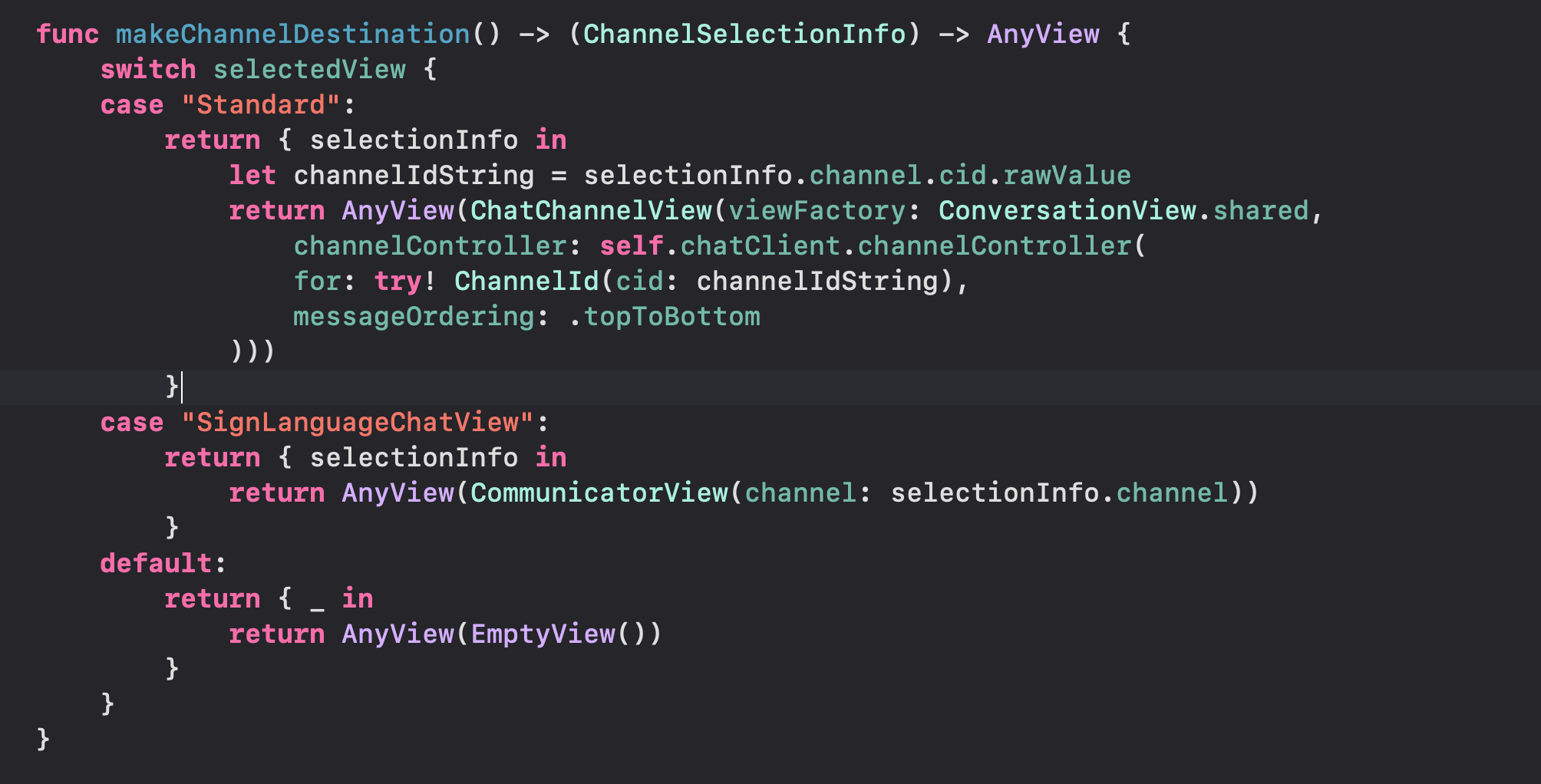
**The Vision framework performs face and face landmark detection, text detection, barcode recognition, image registration, and general feature tracking. Vision also allows the use of custom Core ML models for tasks like classification or object detection[7]. I used this package to create the camera view and I passed each frame of the back camera to the ml model gotten from create ml to recognize the hand signs.**

**StreamChat**

**Stream chat is a platform that allows developers to build apps with messaging capabilities without any problems. I used Stream chat to handle the messaging aspect of my app.**

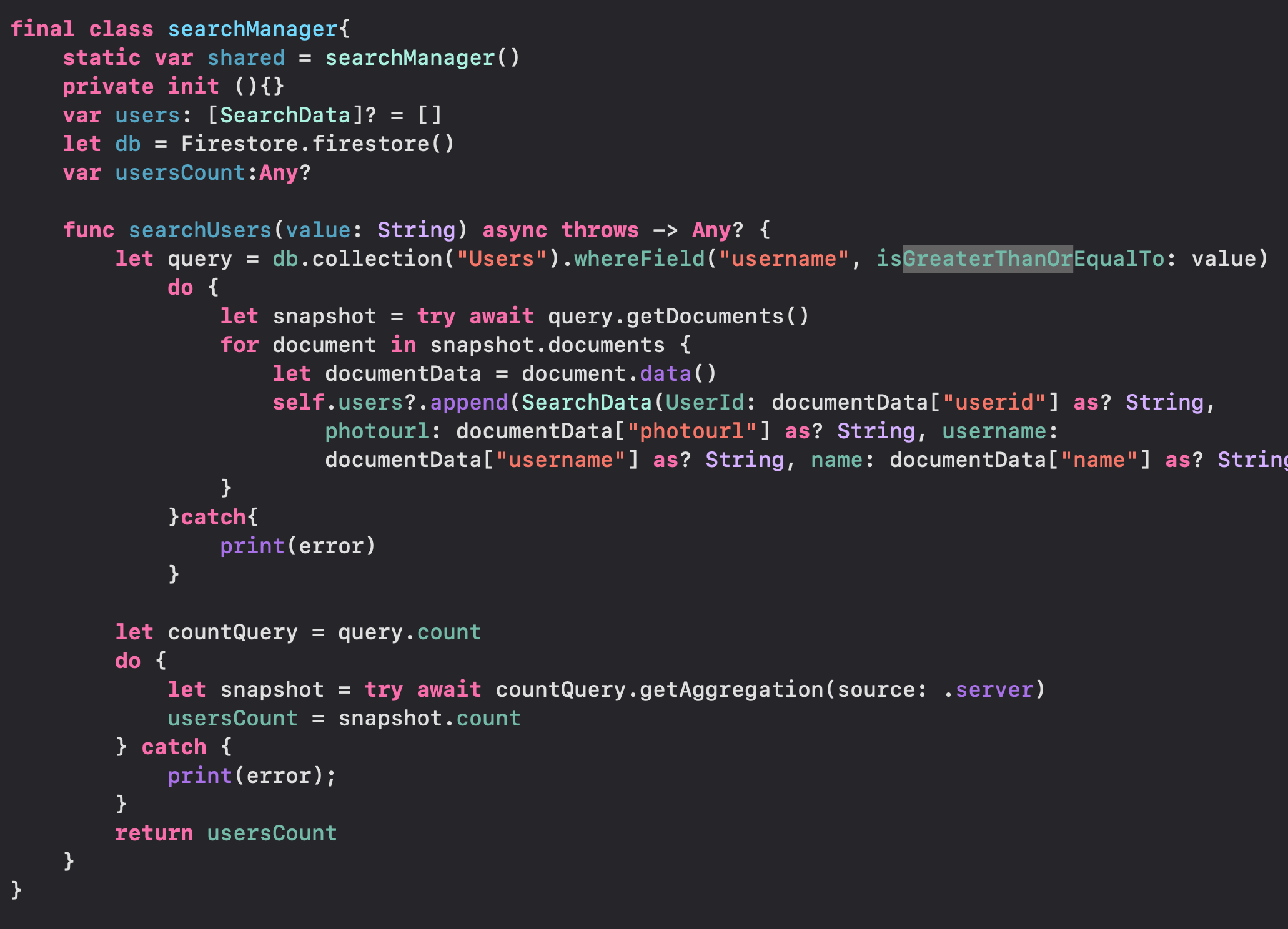
**Some Aspect of my code**

**This code is responsible for sending the user to a message view. I spent a lot of time trying to figure out how to segue the user from the conversation view to my own custom view. The code has two cases the first is responsible for taking the user to the default chat view that came with the stream chat sdk, while the second one is responsible for taking the user to my own implementation of the chat view, which consist of the camera and a message view**

****

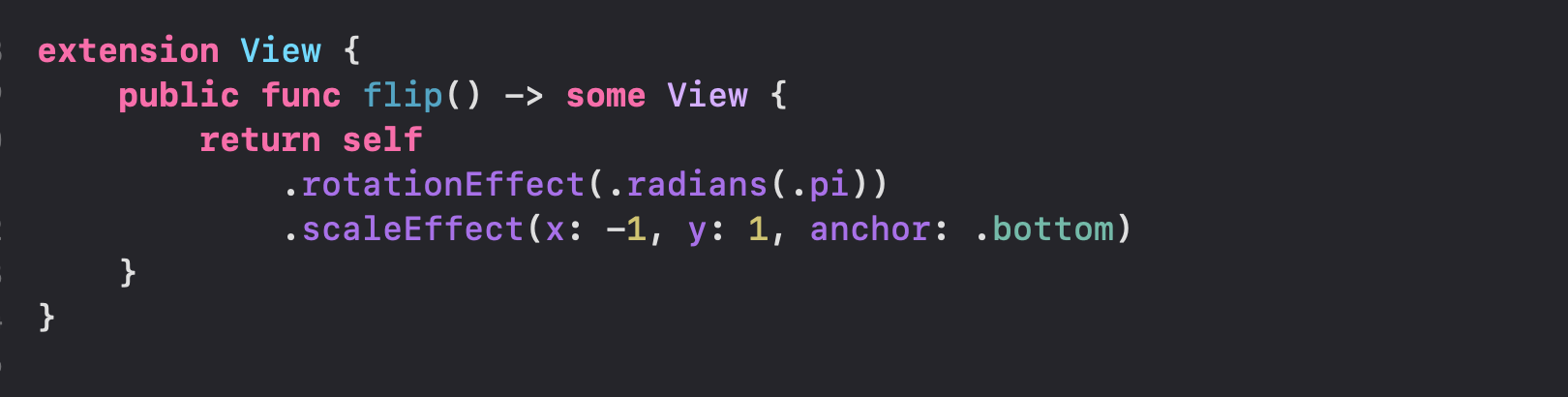
**This three code is responsible for showing the friend view from the database. The reason this was included is because it took some time for me to figure how to display the friend view.**

****

****

****

**This code is responsible for displaying the chat bubbles on the users screen**

****

# Conclusion

## Benefits

**Benefits to users :**

* + - **Improved Communication: The app can help user communication more effectively and accurately with people who do not understand it**
    - **This can help users participate in social activities.**
    - **It helps users to communicate without the need of an interpreter.**

### Benefits to me :

* **The oppurtinity to develop a useful tool for deaf or hard of hearing people to improve their communication**
* **A chance to learn how to train a machine learning model**
* **The chance to make a significant contribution to the deaf and hard of hearing population.**

## Ethics

**The use of computer vision has a lot of ethical issues**

* + - **Legal Consent Violations: Facial Recognition has been used to collect personal data without consent, resulting in violations of privacy laws uch as Illinois’ Biometric Information Privacy Act (BIPA) and the California Consumer Privacy Act (CCPA), prompting a multitude of class action lawsuits. Due to a public outcry against privacy violations, Apple delayed the launch of its controversial CV-driven software designed to detect and report Child Sexual Abuse Material (CSAM) found on personal devices. Users feared that their personal images could be misused for government surveillance or false prosecution. [3]**
    - **Ethical Consent Violations**
    - **Facial Bias**

**Why did I choose this project?**

**The reason why I chose this project was because I watched a google presentation on a smart glass they created that can display a conversion between different users speaking in different language and it displays it using augmented reality. So I decide to do something similar but for ASL instead.**

## Future Works

**Yes the improvements that I will be doing are:**

* **The possibility of the users to train their own sign language variants which can be used by others**
* **A learning component, in which a non-deaf person can learn sign language through the use of augmented reality**
* **Push Notifications**
* **Adding another output apart from messaging. i.e Augmented reality**
* **Make an android variant of the app**
* **Adding more sign language support**

# References

**[1] *Signily: American Sign Language Keyboard App*. (n.d.). Signily: American Sign Language Keyboard App. Retrieved December 27, 2022, from** [**https://www.corada.com/products/signily-american-sign-language-keyboard-app#:~:text=Signily%20is%20a%20sign%20language,American%20Sign%20Language%20(ASL)**](https://www.corada.com/products/signily-american-sign-language-keyboard-app#:~:text=Signily%20is%20a%20sign%20language,American%20Sign%20Language%20(ASL))**.**

**[2] Hand Talk: your website accessible in ASL. (n.d.). Hand Talk - Learn ASL Today. Retrieved December 27, 2022, from https://www.handtalk.me/en/**

**[3] Ethical Issues in Computer Vision and Strategies for Success — Innodata. (2021, November 9). Innodata Inc. Retrieved December 27, 2022, from** [**https://innodata.com/ethical-issues-in-computer-vision-and-strategies-for-success/**](https://innodata.com/ethical-issues-in-computer-vision-and-strategies-for-success/)

**[4]Firestore  |  Firebase. (n.d.). Firebase https://firebase.google.com/docs/firestore/**

**[5] Overview - Roboflow. (n.d.). https://docs.roboflow.com/**

**[6] Apple Inc. (n.d.). Create ML - Machine Learning - Apple Developer. Apple Developer. https://developer.apple.com/machine-learning/create-ml/**

**[7] Vision | Apple Developer Documentation. (n.d.). Apple Developer Documentation. https://developer.apple.com/documentation/vision**