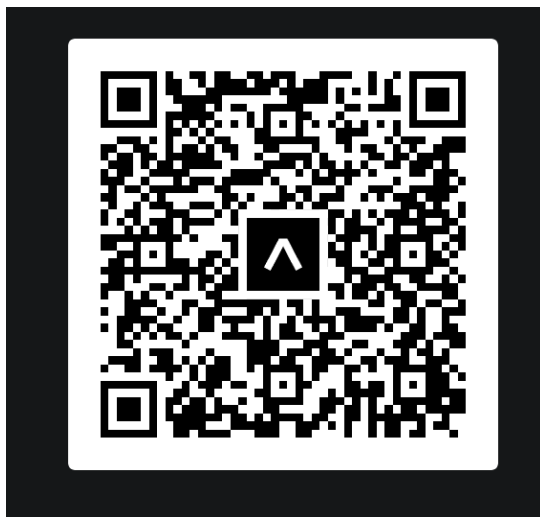




TrainGone README

CS 147: Autumn 2023

Jennifer Fung, Dria Lee, Arjun Sharma, Daniel Trejo



Final Prototype is linked above. See instructions below.

ABOUT

This platform is designed to bridge the gap between ASL learners (who may struggle to recognize new signs) and members of the Deaf community (whose primary language is ASL and may not necessarily know the English translation for a sign).

While current ASL dictionaries assume that the user already knows an English word and is seeking the corresponding ASL sign, there is no intuitive platform that allows a user to identify the English word for an ASL sign based on its visual parameters (handshape, body location, palm orientation, and movement).

TrainGone functions both as a “reverse ASL dictionary” and a social educational platform where one can contribute to an ASL sign’s dictionary entry by uploading a video of themselves using the key ASL sign in an example story. This allows ASL learners to gain contextual exposure to unfamiliar signs by seeing crowdsourced examples, and fluent ASL signers to engage in storytelling, the foundation of Deaf culture and community.

“Train gone” is an ASL idiom akin to the English idiom, “that ship has sailed”, meaning that the conversation subject has come and gone or that the speaker doesn’t wish to repeat themselves. *TrainGone* is designed for those one and done situations where you didn’t quite catch a sign or you want to share a story once and have it viewed over and over without repeating yourself.

DESIGN TOOLS

React Native, **Expo**, and **GitHub** were used in the making of this prototype. **React Native** was the framework used for the app. **Expo** was mostly used to test the interface of the app, but we downloaded a few of their libraries in order to create video components for the app, use the camera functions, and to download free icons for the tab navigation bar. **GitHub** was used to facilitate collaboration between the software developers.



OPERATING INSTRUCTIONS

Since we only had around two weeks to create the final prototype, we had to prioritize the functionality of the three tasks, while other planned functionalities would be inaccessible. The limitations to this prototype will be described below.

To access the prototype, download **Expo Go** on your mobile device and then scan the QR code above. This will open the app, where you will be able to access the prototype. You may need to create an Expo account.

Home Page (1st Task Flow):

- ❖ This is the first page that appears when you open the app. From here, there are several places you can go.
- ❖ **Search by Word:** This search function allows you to type an English word, and you will be taken to the dictionary entry of that word. Due to our constraints, the dictionary entries that we have available are: Milk, Water, Coffee, Baseball, Soccer, Football, Hello, What's up, and Good to see you.
- ❖ **Search by Sign:** This is our first task flow, and is the actual reverse ASL dictionary. If you click it you will see images of various hand gestures (note that the list is not complete), and you can click on them to look for a word again, this search is also limited. Below we have listed the words that you can look up
 - Milk: C and S hand shapes, palm facing non-dominant side, chest area, palm back and forth movement.
 - Water: W/6 hand shape, palm facing non dominant side, chin location, back and forth movement
 - Coffee: S hand shape, palms facing each other, chest location, back and forth motion and circular motion
- ❖ **Search by Category:** This will take you to a video feed, in which each video is part of the category you selected. The only categories implemented are Sports, Drinks, and Greetings.

Feed Page (2nd Task Flow)

- ❖ Selecting **Feed** in the bottom tab will take you to this section. This page is a video feed of dictionary entries that people have posted. You can save them by clicking the heart icon in order to look at later and learn. (N.B. while you can like the video, it will not show up in a saved folder for you. We did not have enough time to implement this and was not an impactful part of our second task flow).
- ❖ Each word has a name and a category. Clicking on the category will take you to a secondary feed in which all of the videos are from that category alone.

Create Video Page (3rd Task Flow)

- ❖ Now it's time to show off what you have learned in the first two tasks! Clicking on the **Create Video** tab at the button will ask you for camera and microphone permissions. The camera will open up and you can begin recording.
- ❖ If you don't want to record a video from the app, you can also upload videos from your camera roll.
- ❖ Once you have a video that you want to post, you can add the important information, such as the name of the word, definition, and all of the hand signs you used to create

the video. That way when other people utilize the **Search by Sign** feature, they can easily find your word.

- ❖ After posting, you can find your new word in the **Definition** section of your profile

Profile Page

- ❖ Clicking on the **Profile** tab takes you to your profile page. Here you can see both your **Definitions**, which are the words you have uploaded, and your personal **Dictionary**, which are words you have liked and want to practice later
- ❖ There are videos in both sections, but since they weren't a major part of our tasks, we did not have enough time to implement functionality.



WIZARD OF OZ

Since we do not have an existing network of users, it would be difficult and counter-productive to create a large amount of videos and then an algorithm to display them on our prototype. Since the specifications of the algorithm do not affect the overall user interface and experience with a prototype, we decided to imitate the video feed for the user. The handful of videos that were created were displayed on the feed, so that when the user swiped, it would appear as if there were an infinite amount of videos, similar to other social media platforms. In reality, had the user scrolled down far enough, he/she would arrive at the end.

HARD CODED ITEMS

The complex task for our project - creating a video - was fully implemented in our prototype. However the other two tasks, to look up a word and to browse other users' videos, were limited by the fact that we did not have an existing user base. This meant that there were no entries of words that the user could look up, nor could the user look at other people's videos. Therefore, we had to hard code some data in order to complete the other two task flows. We created nine dictionary entries, each belonging to one of three categories:

- Drinks: Water, Coffee, Milk
- Sports: Baseball, Soccer, Football
- Greetings: Hello, Good to see you, What's up

In addition, we added thumbnails of videos in the home screen and profile screens. This was all done to give the appearance of an existing, fully functioning app.