

# SpeakEasy

*Integrating context-sensitive visual and aural language acquisition into your daily life.*

Gabriela Groth (Manager, Design), Eric Chew (Development), Tommy Truong (Documentation), Carlos Araujo (User testing)

## **Problem and Solution Overview**

Language students have a hard time learning a new language. They are often removed from real world situations while learning, and once they are on-the-ground trying to communicate, they lose the support they had in the classroom. SpeakEasy strives to overcome these limitations by creating a context-sensitive, on-the-go language learning environment. With instant contextual phrases suggestions, dynamic speaking support, and helpful growth tracking, SpeakEasy allows users to learn and master a language even while performing other activities.

## **Tasks**

In order to enable users to learn a new language easily and intuitively while having contextual support, we redefined our 3 representative tasks to more accurately guide our UI design

- 1) Understand and reproduce the correct pronunciation of words (simple)
- 2) Enhance real-life conversation through instant phrase suggestions (complex)
- 3) Integrate language learning into daily life (medium)

The meaning of our first task did not change much, but we reworded the task to make it more clear. Our previous task “intuitive pronunciation” was too broad and didn't clearly map to a scenario. We changed it to “understand and reproduce the correct pronunciation of words” to make it an actionable task a user would want to do.

Our second task also had a major change from our previous one: Contextually sensitive language support. Again, the previous the task was too broad and didn't help us designing our scenarios. During our low-fi prototype, we realized that many of the users not only had problems with figuring out what to say, they also had trouble understanding what was being said to them. Therefore we decided to include translations of what is being said to the user, in order to facilitate a real-time conversation. We also realized that it was important to emphasize the support and context part of this task. By support, we would like to provide the user with phrases suggestions and help understanding what was said. And by context, we would like to support users in a real-time conversation that they are struggling with. During real-life conversation with foreigners, students learning new languages struggle with what to say in response to an inquiry because they don't know or forget a specific word. This

limits the conversation smoothness and increases students' level of anxiety. Also, foreigners often speak in accents that you are not used to or faster than you can understand or use words or slangs that you haven't heard before.

Our last task didn't change too much from our previous one: Active language learning integration. We only dropped the active part of the task and emphasized the importance of daily life. During our low-fi prototype, we realized that the most important part of this task was not to be proactive to help the user learn the language, but to provide the data and tools for him to practice his speaking daily. This task has a medium complexity. Students often have a busy life, so it is important to provide data and progress information that they can review on their own time.

Our three tasks cover all important aspects of learning to speak a new language. They cover pronouncing words correctly, understanding what was said to enable the correct response, providing suggestion to enable smooth conversation and giving feedback so that student can learn and practice the new language without disrupting his or her daily routine.

### **Revised Interface Design**

In our new design, we made significant changes to both the Google Glass interface as well as the mobile application. On the Google Glass interface, we made three main changes: we added native translations next to each suggest phrase, included the ability to record and translate conversation currently happening, and made the gestures more intuitive by both adding a tutorial as well as including directive arrows on the screen to indicate various gestures. For the mobile application, we also made significant changes, including a new ability to view past conversation transcripts as well as sections to practice and see one's progress.

From our experimental results, it was clear that users with very little Spanish experience had difficulty understanding the conversation and phrases. During the simulation, they would ask us what the waiter was saying and what each phrase meant. Those who did utilize our translation gesture for phrases found that there was too much latency in having to use a gesture to translate each phrase and found it disruptive to the conversation. Many of the less experienced Spanish speakers were also still confused even after translating each phrase, as they could not understand the majority of what the waiter was asking them. Given this, we decided to improve the user's understanding by displaying translations of the phrases concurrently with the phrase themselves. We thought that this, combined with the added functionality of being able to record and translate conversation, would give the users a better idea of how to respond in different scenarios and eliminate this confusion that we saw in many of our tests.

The questions that users continued to ask us also pointed to the lack of understanding they had with our gestures. Many of them noted after the simulation that they had completely

forgotten each gesture after being overwhelmed by the waiter's Spanish and defaulted to asking our "wizard of oz" for help. In order to mediate this, we decided to add a short tutorial to the Google Glass application that would help teach users how to use the gestures rather than have them just dive in with little familiarity. Users in our experiment also expressed that they could not remember whether it was a swiping motion or a tapping motion that changed the phrases. Because of this, we also included arrows in our interface to indicate the ability to swipe left/right and up/down.

Beyond just seeing mispronounced words on the mobile application, users also expressed a desire to see the context in which they mispronounced words, as well as a way to practice them and see their overall progress. Therefore, we decided to change the interface of our mobile application to allow users to review a transcript and recording of their conversations. These transcripts let users understand the context in which they missed words, and let them practice these words with contextual relevance. A progress screen was also included to give users an idea of overall improvement as well as gamify the learning process by providing various levels of achievement that users could obtain.

**Scenario 1:** Use the app to correct mispronunciations during conversation

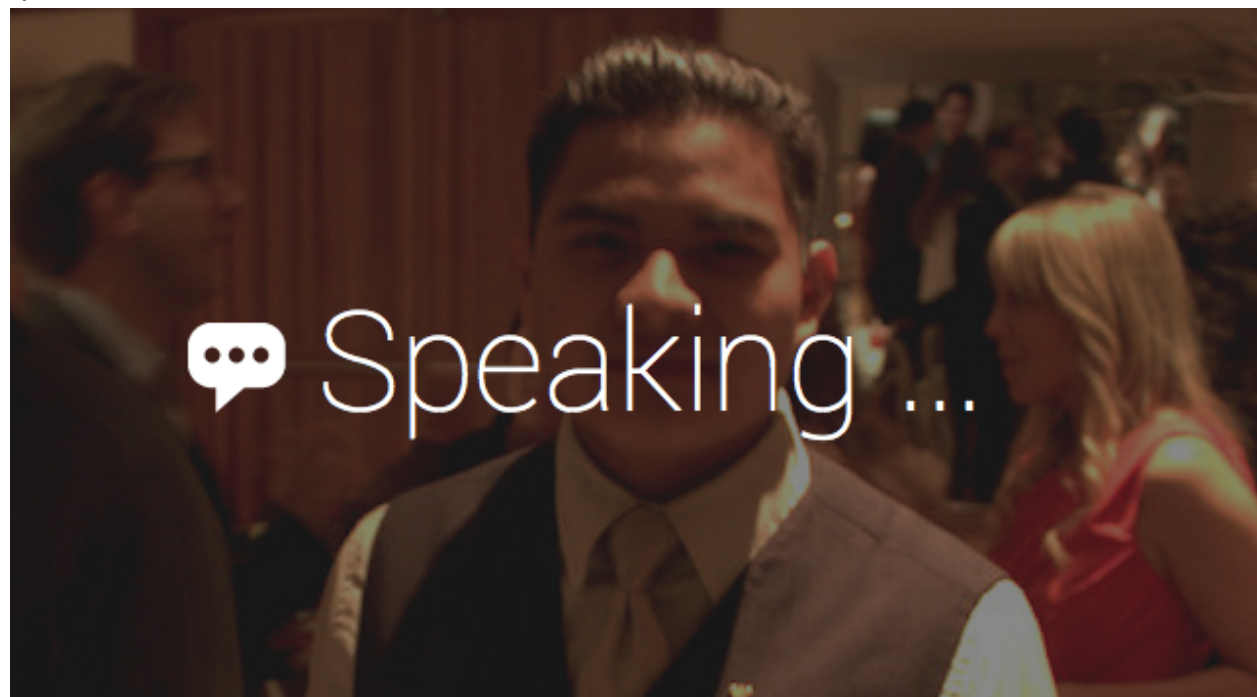
User wants to say a specific phrase.



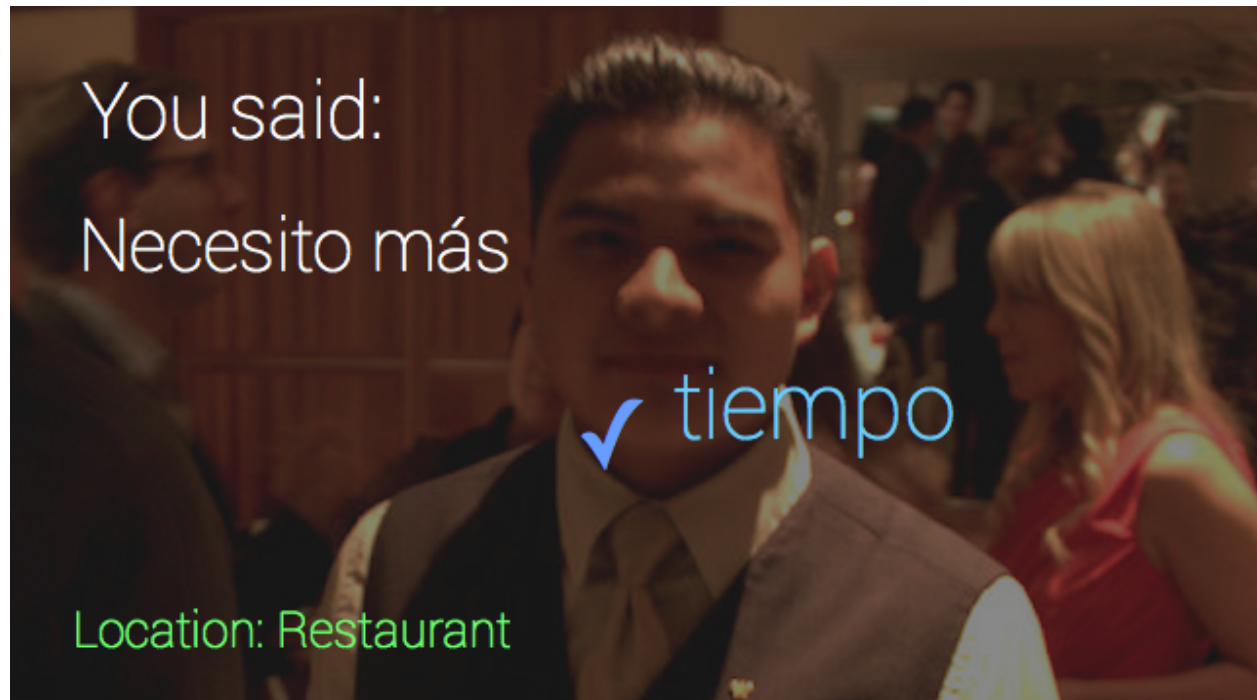
User then speaks the phrase, but makes a mistake. The app will show his error:



The user wants to hear the correct pronunciation, so he taps on the Glass device to hear it spoken:

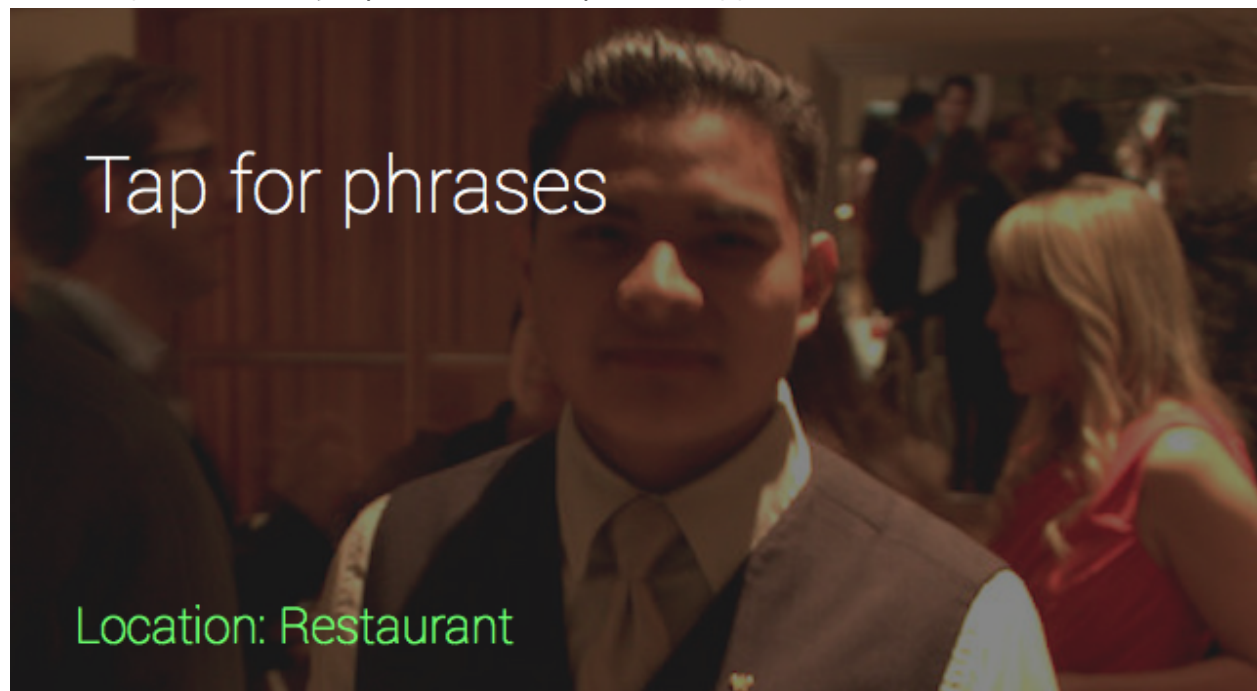


The user replicates the pronunciation correctly, so the app briefly shows a checkmark next to the word before returning to the previous screen.



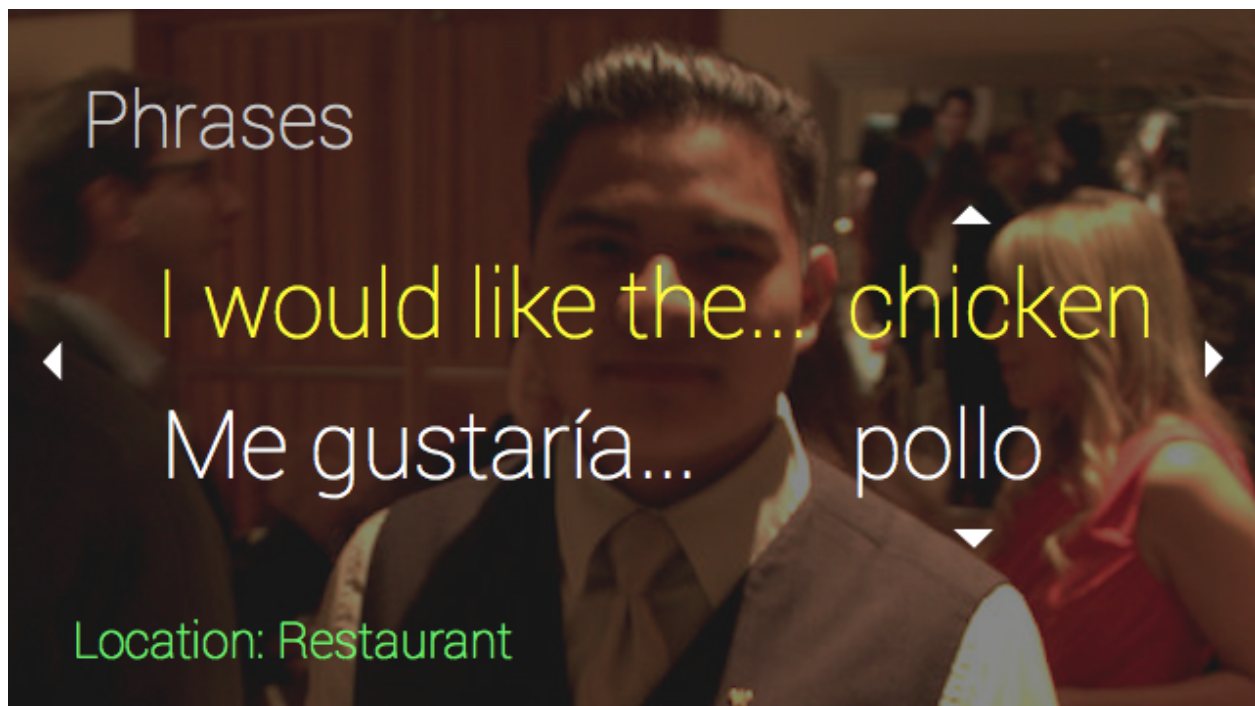
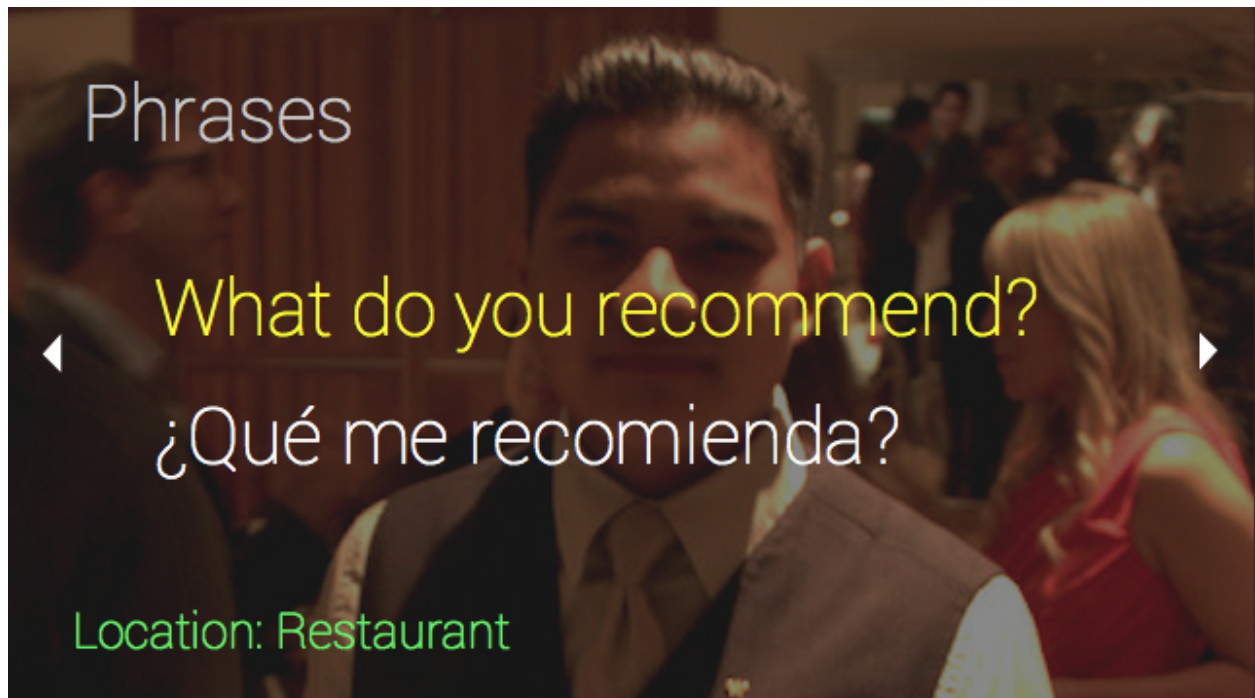
**Scenario 2:** Get helpful phrases specific to where you are (in this example, a restaurant)

The user is approached by a waiter who asks if he/she is ready to order. The user needs help answering back, so they tap the device for phrase suggestions.

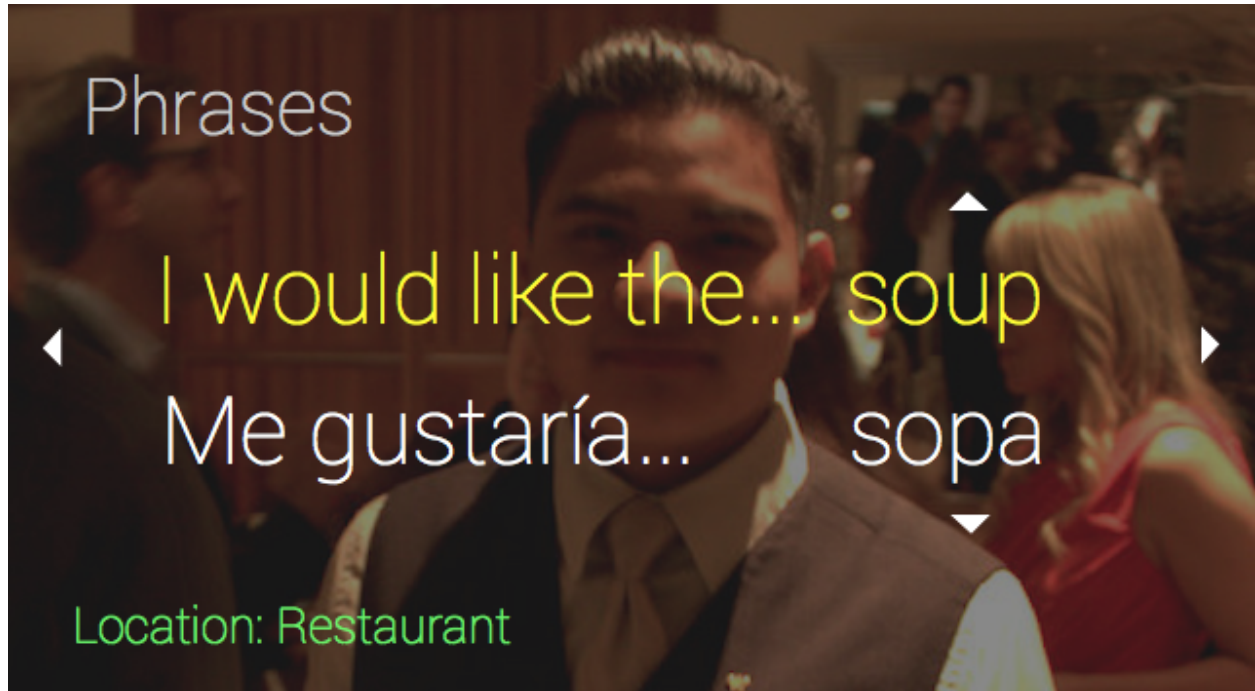




They then swipe forward and backward to cycle to get to the phrase they want.

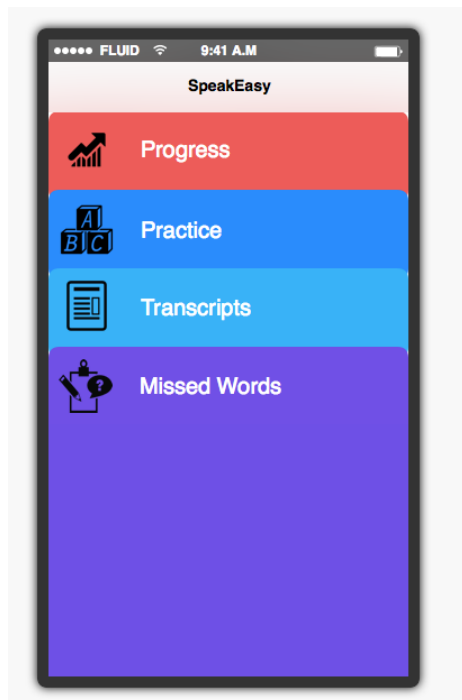


They can now swipe up and down to go through possible completions for the phrase.



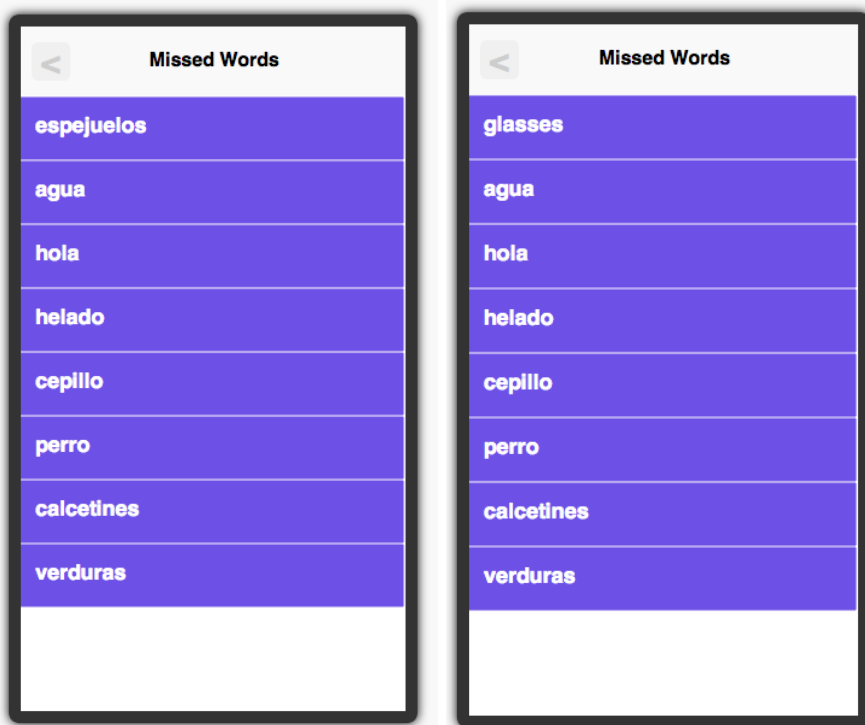
**Scenario 3:** Use app to get feedback on what you've learned and where you need to improve.

The user goes to the mobile app, which starts off on this home page:



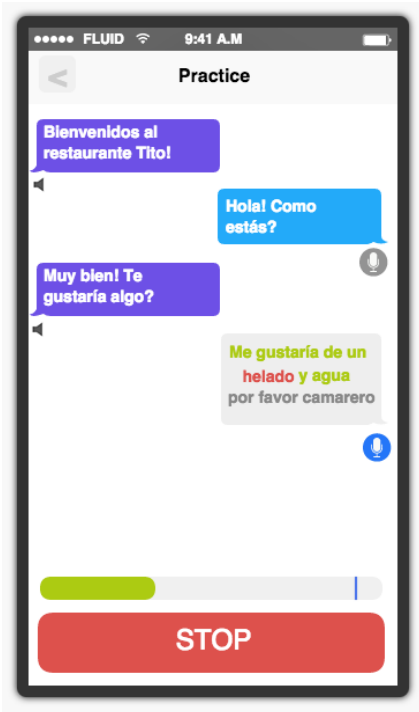


If the user wants to see what words they mispronounced or did not know throughout the day, they can go to the Missed Words tab.

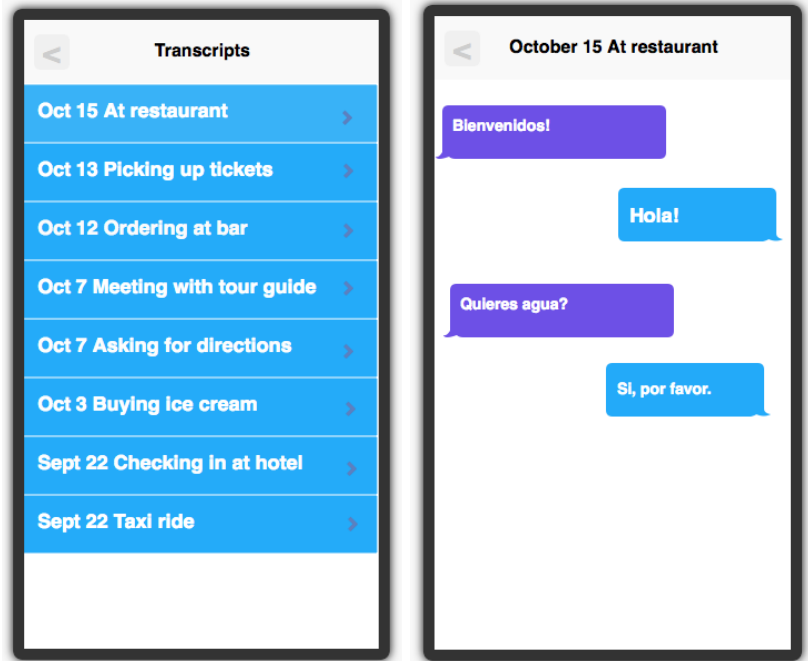


When the user swipes on a particular word (the first one in this case) the translation of the word comes up. When they tap it, the pronunciation will be said (although this has not been implemented yet).

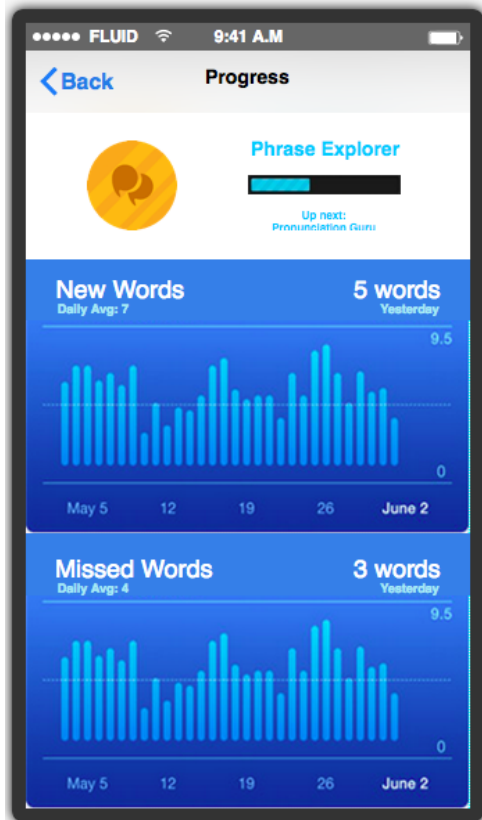
The user can then go back to the home screen and go to the Practice tab to work on their pronunciation with guided practice sessions.



Users can then go back to the home screen and go to Transcripts which will show them transcripts of the conversations they recorded throughout the day.



Finally, the user can go back to the home screen and go to Progress. There they will see some summary statistics of how their language skills have been changing over time.



## PROTOTYPE OVERVIEW

### Google Glass portion:

Link: <https://www.justinmind.com/usernote/tests/13029184/13029208/13039767/index.html>

To prototype for the Glass portion of the app, we used Justinmind Prototyper, because it has support for Google Glass. The tool worked really well in showing what screens the user will see in their line of vision, and it had many useful templates specific for Glass that helped us understand a little more about the Glass interface. Overall, the tool was pretty easy to use and we learned how to create screens and connect them without too much difficulty, and we were able to include examples of all the major screens and functions that we wanted to show.

However, it was difficult to develop a prototype for Glass without actually having a Glass device. Thus, our prototype must be viewed in a web browser, and viewers must click buttons that represent the actual gestures that one would use on a Glass device. Justinmind Prototyper also claimed to allow collaborative and shared editing of the same prototype, but this functionality appears to be broken so we had to develop on just one machine.

We didn't use any wizard of oz techniques. The prototype is all hard-coded - the prototyping tool doesn't allow us to gather actual location information to provide appropriate phrases. Thus, the prototype as it is now only shows usage in a specific restaurant scenario. In addition, we do not have any translation/phrases database set up, so we just added a couple example phrases with their translations directly into the prototype.

**Mobile portion:**

Link:

[https://www.fluidui.com/editor/live/preview/p\\_WA6PIId5uRoSaCNqym8d8ZBBBrFe6mtTdl.1414726288453](https://www.fluidui.com/editor/live/preview/p_WA6PIId5uRoSaCNqym8d8ZBBBrFe6mtTdl.1414726288453)

To prototype the mobile part of the app we used FluidUI. We chose this tool because it had a very easy learning curve, and we could quickly build a prototype of a multi-screen application. The downside to the fast and intuitive prototyping was that the tool did not have as much functionality as we would have liked. It does not have support for swiping, just tapping. The prototype is all hard-coded because many of our functionalities require listening to a user's conversation and transcribing it. For this reason, we populated the prototype with examples data of what a user's application might look like during use.

Another downside of the prototyping tool was that it did not allow for real-time collaboration. This slowed down our design process and made it difficult to really collaborate on each part of the design.

**README:**

Link:

[https://docs.google.com/a/stanford.edu/document/d/1b2nhxAA3evk-v6RnPO\\_Vz1aHAh7PUSjBkmgovzQjrk8/edit?usp=sharing](https://docs.google.com/a/stanford.edu/document/d/1b2nhxAA3evk-v6RnPO_Vz1aHAh7PUSjBkmgovzQjrk8/edit?usp=sharing)