

TONGUES: MEDIUM-FIDELITY PROTOTYPING

Alex Wu, Anna Yelizarova, Ishita Prasad

Prototype Link: <https://tongues.proto.io/player/?id=4d2b83c5-b109-4b11-bd36-94474e4ccdd0>

PROBLEM & SOLUTION

Tongues is a crowdsourcing application that helps people find real time, accurate and colloquial translations, and have fun with the quirks of different languages. When automatic translators just can't get it right, Tongues enables users to ask the people around them what the real way to say something is. This project is being developed by Alex Wu (Team Manager and User Testing), Ishita Prasad (Visual/Interaction Designer and Documentation Coordinator), and Anna Yelizarova (Developer and Visual/Interaction Designer).

This week, the development plan for Tongues involved creating a medium-fidelity prototype for the application. We previously had a lo-fi prototype that we conducting user tests on, and we hoped to reconstruct our application with the results of this feedback. The medium-fidelity prototype aims to give users the experience of actually using and interfacing with the Tongues application on a mobile device.

TASKS

The three tasks that our prototype enabled ranged from simple to difficult, as indicated below:

1. Ask for a translation: a user could request a translation for a phrase into the language of the place they are currently in.
2. Answer translations: a user can participate in an upvoting-style game by upvoting or writing their own translation for requested translations. This process is intended to be a humorous insight into their culture or locality's quirks and phrases. In response to feedback, we changed our initial point system to one in which users could "level up" by upvoting or writing more translations, moving from, say, "Novice" to "Apprentice" and other fun levels.
3. Change location: a user can change their location to request or answer translations for languages spoken in other regions by changing their location.

REVISED INTERFACE DESIGN

Based on the feedback we received from our previous prototype, we made 2 major changes to our UI. First, we changed our transitions on the translator side. In our low-fi prototype, we had a feedback page after each upvote. However, according to our participants, this was slow and confusing to them. Instead we opted to have the feedback and rewards appear on the same page as an animation. We haven't figured out the details of how we want this to occur since it was a technically difficult to create in our prototype. For the time being we have a star appear next to the saying if the top pick was chosen. This ties into our next major change - better gamification. Instead of having points pop up like in our old prototype we decided to have an icon replace it. Users were confused by what the points meant so instead we want to have something more intuitive like thresholds and badges. Ultimately we would want to keep track of individual's skill and participation so we could gift them rewards such as airline discounts or trips to the destination that

they are using the app for.

PROTOTYPE OVERVIEW

To construct our medium-fidelity prototype, we used the tool *Proto.io*. We surveyed a number of prototyping tools, including Marvel and InVision. Ultimately, we chose Proto.io because we felt that it had a stronger capability to create forked storyboards based on different user inputs, and it also had the feature of letting the user type - a crucial piece of interaction for our application. Additionally, we used Photoshop to create the graphics. Although Photoshop has a heavy functionality, we chose to use it as it gave us more freedom to create a unique design, and our team knew the application well.

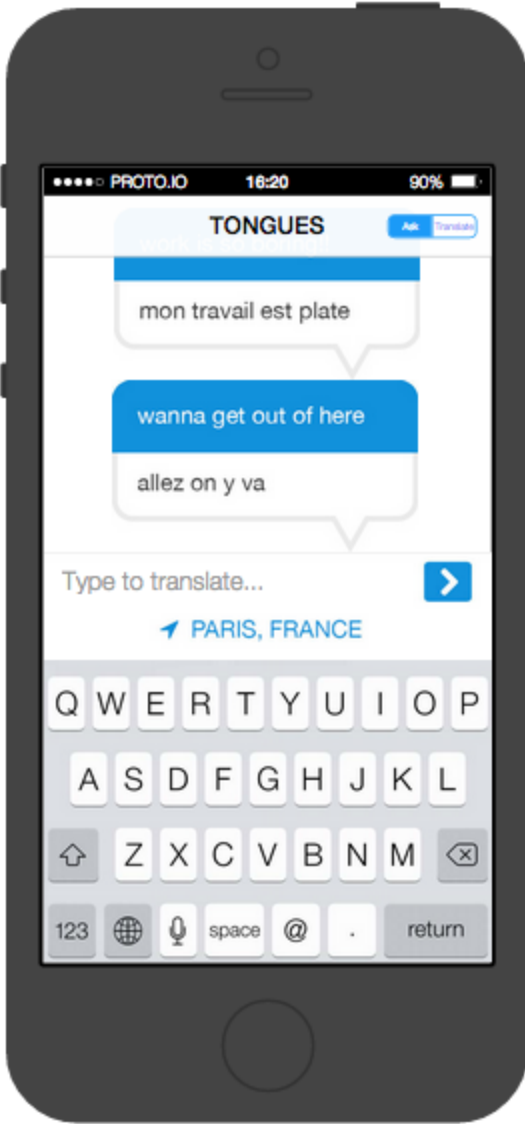
The initial plan we had included many subtleties (in terms of graphics) to help make the app feel “real” (eg. bubbles that faded in, spinning wheels while an answer was “retrieved”) but increasing the interactivity and number of subtle graphic points exponentially increased the complexity of the prototype in Proto.io.

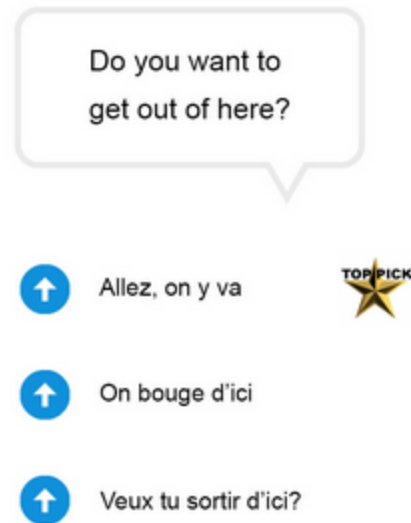
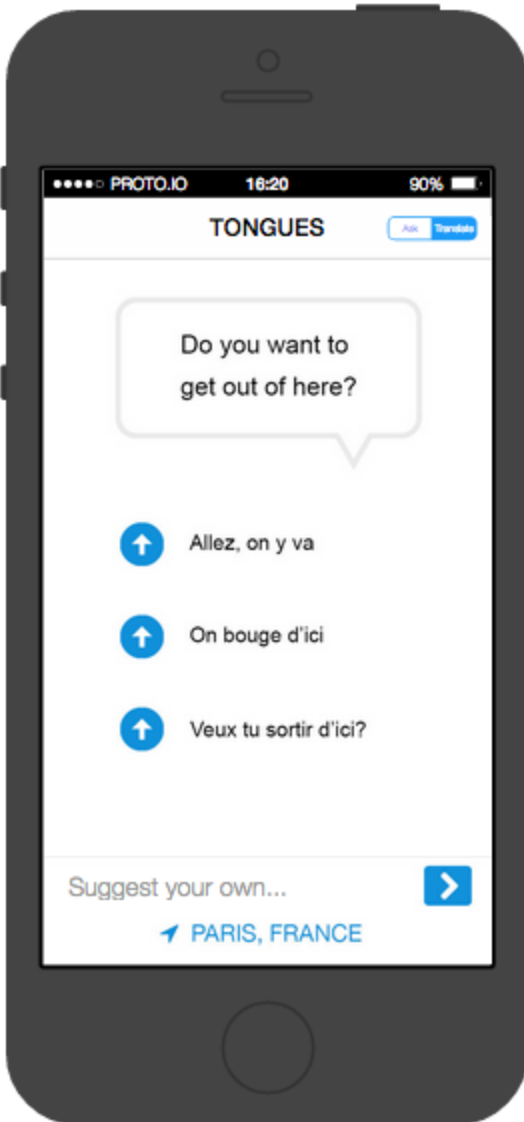
Limitations: Our prototype was limited in a number of ways. Firstly, we chose not to implement several secondary features such creating an account, viewing scores/levels, changing settings, etc. We initially intended to have a slide-out sidebar menu (eg. the Gmail or Venmo apps) to support these features, but did not implement it because it was not crucial and did not directly support the 3 task interaction.

Our second major limitation was that no matter what the user typed in the “asker” task, the application would always assume that the phrase was one of three “preprogrammed” or “hard-coded” ones. We would also not return crowdsourced answers, but answers that we had written beforehand. In a similar vein, in the “translator” task, we only presented 3 phrases to translate, although in reality we hope that the translator will have a continuous stream of phrases. Additionally, we could not effectively score the upvoting with the limitations of our application, so that was hardcoded into the application as well. Additionally, we only supported two regions, Paris and Berlin, as opposed to regions all around the world.

Finally, our third major limitation was in the graphics and complexity of our application, as discussed previously. Because of the complexity of Proto.io and its function as a simple prototyping app, we were not able to effectively prototype some of the interactions and animations that we believe would be crucial for a final application. For example, we hoped to have both the “asker” and “translator” side available for German and French, and smooth translations between all 4 “sides” but the prototype’s complexity was doubled and we were unable to do this.

PROTOTYPE SCREENSHOTS





Medium-Fidelity Prototype Readme

Our prototype is created with Proto.io. Because of the medium-fi prototype nature of it, there are some restrictions in regard to its functionality and interactivity. This document contains instructions required to operate it.

The path we recommend through our application is the following:

1. Play around by typing on the “ask” screen (initial screen) (task 1)
2. Tap the switcher at the top to switch modes and play around (task 2)
3. Tap the location at the bottom and play around (task 3)

“ASK” SCREEN

On the ask screen, you can type whatever you like to be translated. However, our prototype will assume that you have typed one of the following 3 sentences:

“Wanna get out of here?”

“It’s about time”

“Work is so boring”

And accordingly retrieve a translation. However, you can imagine that the phrase you actually typed in will be the one translated. We thought it was important to have the ability to type; however, because this is a prototype, we could not allow for arbitrary sentences and had to hardcode it.

“TRANSLATE” SCREEN

On the translate screen, you are presented a sentence to translate. This happens 3 times. However, we request that you imagine a continuous stream of translations presented to you, not just 3.

SWITCHING BETWEEN SCREENS

When on the “ask” screen, you can switch to the “translate” screen by tapping the switcher on the top navigation bar. Our initial plan had a slide-out side menu to allow users to toggle between these modes, along with information about account, settings, etc. but our prototype became too complicated with that.

LOCATION SWITCHER

You may tap on the location at the bottom of the screen to change your region (and thus language) by finding the region on the map that appears. Unfortunately, our prototype only supports two hard-coded locations: Paris, France and Berlin, Germany. Additionally, you will not be able to switch locations effectively on the “translator” side of the prototype.