**Team USA Project Proposal**

Team Members:

Chloe Andresol

Amir Binaiekondolojy

Kelly Fernandez

Rachel Gillespie

Vehbi Karaagac

Huda Khalid

Andrew Kim

Isaac Richards

**CodeLingual** (Title work-in-progress)

1. **Introduction**
   * A web application whose purpose is to help individuals quickly learn the basics of new coding languages by translating functions from one language to another.
   * When beginner programmers are assigned tasks in both school and the workforce, they are often expected to code in unfamiliar languages with little-to-no time allotted learning time. The process of relearning the foundations and syntax of new languages when others are already known can be tedious and frustrating, especially when traditional internet searches often result in confounding or outdated answers. CodeLingual will help to cross this bridge with relative ease by providing a programmer-approved, one-stop-shop for function translation between two (or more) languages.
2. **Project Details/Specifications**
   * A user would type in a standard or common function in a coding language that they know, and then the application will display the equivalent function in the language that they are curious about learning.
     1. For example, if a user typed in toUpperCase() and selected Java as their source language, the application will return upper() if Python was selected as their target language.
   * The user will also be able to submit feedback if they type in a function and no results are returned, or if the application displays the wrong answer.
   * If the user happens to be experienced in another language, they can also contribute by submitting new translations for review. When this occurs, the application’s administrators may double-check the correctness of the new translation and add it to the underlying function database.
   * Stretch goals/low-priority features:
     1. A web scraping-powered translation feature for results that come up empty because they haven’t been added manually. These web-scraped translations will also be sent to the app admins for review.
     2. A log-in page and separate UI for application administrators that allows them to review submitted translations from the application.
     3. A database-scheme that scans through translations based on how frequently they’re searched for (to promote time-efficiency).
   * We will be leveraging the MERN technology stack to create this application.
     1. A MongoDB database will be used to store translation information and metadata.
     2. ExpressJS and NodeJS will be used for back-end work, which will include inserting, modifying, removing and retrieving information from the database. (These frameworks will also power the web-scraping component of our application if we have time for it.)
     3. ReactJS will be used to create the front-end interface.
3. **Motivation**
   * We, as students, take time to simply *start* learning new languages and finding the right resources to do so. Our team came up with this idea to fulfill the need for a universal starting point and give students like us that extra boost.
   * Having an application that gives computer science students a clearer idea of what we’re getting into when learning new programming languages will most certainly help motivate us into picking them up faster.
4. **Possible Complications**
   * Due to the scale of the idea, we are most likely starting off with languages we are most comfortable with. As there are several dozen coding languages out in the web. We had the idea of web scraping, but including this will take much longer than the allotted semester work period.