Team 15

**Team members:** Anjali Roy, Divya Shakamuri, Shayen Patel, Morgan Marino

**Project name:** iMessages++

**Project Synopsis**: An application for iOS devices that allows you to write and test code over messages.

**Project Description:** The purpose of iMessages++ is to communicate code easily through messaging. This software will be designed for iPhones and an extension will be required to implement the features. The application will appear on the gray task bar when texting another individual. The icon will be like VS Code or XCode and reside besides the other icons such as game pigeon. Once the button image is clicked, it will display a few options of what language you wish to type in as well as the distinguished color scheme for syntax. A drop-down option will appear like online GDB where the list of languages will be displayed. Once the option is selected the messages will prepopulate a template of code to get started. Auto tab and auto parse will be installed into our design. This feature will allow for easy usability and efficient typing of the program.

A main component to our software includes color coding for different functions. For instance, object may appear blue while strings can appear yellow. This feature will allow the user to quickly identify the main elements in their code. However, an important distinction is this color scheme will vary based on the language (Haskell, Python, C++, JS, etc.) The idea is to implement bright and bold colors throughout the languages. Additionally, we need to ensure that it is compatible with the iPhone keyboard. Doing so will guarantee high performance and good user experience. Making sure that the comments symbol and indentation button are consistently on the keyboard will push for better written code. Another element to consider in the software is an easy shortcut for longer lines of code. This challenge we face is the iPhone screen being significantly shorter than a laptop or desktop. Curating easy readability with long lines of code is something we will have to explore.

Creating a low-level compiler to test the code is an additional component we wish to develop. Figuring out how this design will work while displaying syntactical errors will be a major improvement to the original idea. However, this is not as simple as it may seem because the iOS needs to support this high-level program. We would like for this application to be downloaded off the App Store and throw out regular updates. Following the theme of iPhones, we hope to include change in highlight style (dark mode, light mode, etc.). This way it will be consistent for the user throughout their device. This continuity though small will result in good user experience.

Diagram

Description automatically generatedDiagram

Description automatically generated