IceBreaker: Catch Up w/ Friends!

Elias Taylor, Fabio Tran

Advisor: Prof. Aspen Olmsted

Wentworth Institute of Technology School of Computing and Data Science

Abstract

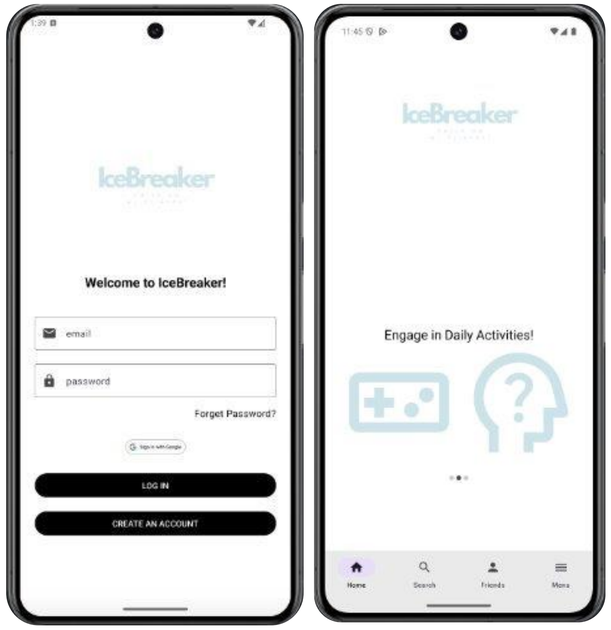
IceBreaker is a social networking mobile app that will encourage users to participate in a brief activity with their friends once a day through a randomized notification alert. The first goal of the app is to help people maintain relationships with friends and loved ones whom we may forget to reach out to during our busy lives. Furthermore, the app will facilitate the growth of potential new relationships as it aids in "breaking the ice" where it might have been traditionally difficult for certain individuals.

Introduction

Developing a mobile application with social networking features from the ground up will have us working with programming languages such as Java or Kotlin, frameworks such as Jetpack Compose, and external cloud services such as Firebase, all while abiding by best practices of software engineering principles. At the end of the semester, our team was able to create a high-quality Android application while delivering upon every major component and almost every feature brainstormed in our initial proposal

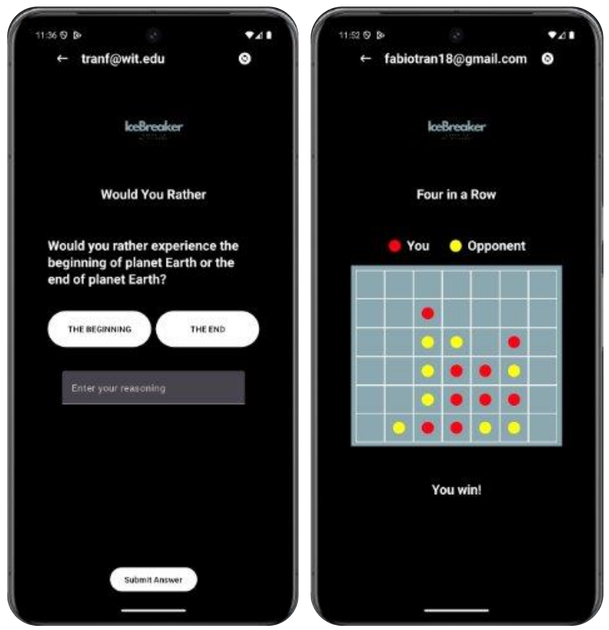
Design

We decided given the scope of the project to program our application for Android devices using Java and Kotlin. From here we looked at Firebase as a resource that could store app data and support our intended functionalities. With our software and resources, we could then move to implementation



Implementation

The first sprint of our project was spent implementing the basic components of the app so that we could add the core functionalities afterward. First, we created an Android Studio project and connected it to a GitHub repository for version control and code sharing. From here we implemented the base UI and navigation between pages, utilizing Jetpack Compose for the login aspects. A large component of this step included utilizing Firebase's Authentication and Firestore Database to register and store user data for the social networking features. With user creation possible we then made sure users can search, add, remove, message, and most importantly play each other in daily activities.



Our second large sprint was focused on the core concept around our application, users participating in daily activities. To make sure that users can participate we had to finalize the friend system so users can add each other via email, search for their friends, chat, and view all the friends they have made.

Our final large sprint was spent on the activities themselves. We developed two types for users to participate in, question-based and game-based, each with its unique features. In total there is tic-tac-toe, four in a row, would you rather, and hypotheticals. After getting the notification system implemented all we had for development was to fine-tune and debug any lingering issues.

Further Development

Given more time, our team would have liked to undergo the process of app submission and approval to a proven Android application marketplace such as Google Play (Store). From there we can have more users access our application and review our work. We also hope to further develop the application to support more messaging/profile customization and the group/organization functionality we had to sacrifice due to time constraints.

Conclusion

We believe our final product fulfills its original purpose of getting friends to engage with each other daily through fun games and stimulating questions. We learned a lot from working with familiar resources from our classes as well as unfamiliar ones. At every step of the process, our team had to lean into research to learn about best practices, optimal solutions, and industry standards to make not only an appealing app from the front but also a clean and manageable product in the back.
We consider the project an overall success with plenty of room for progression in future iterations of the application.

Bibliography

Android Studio: "Android Studio." *Android Developers*, Google, https://developer.android.com/studio. Accessed 29 July 2024.
GitHub: "GitHub." *GitHub*, Microsoft, https://github.com/. Accessed 29 July 2024.
Firebase: "Firebase." *Firebase*, Google, https://firebase.google.com/. Accessed 29 July 2024.
Java: "Java." *Java*, Oracle, https://www.java.com/en/. Accessed 29 July 2024.