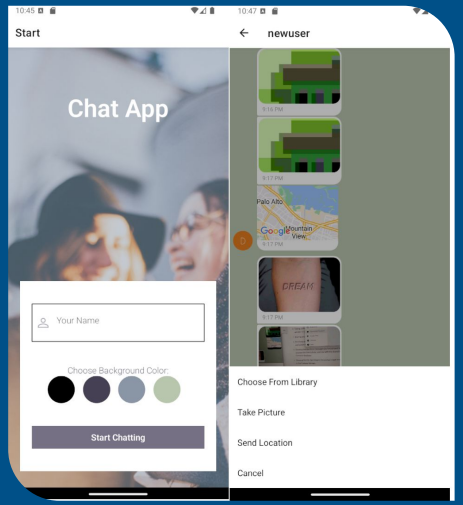
React Native Mobile Chat App



Project Overview:

Role: Lead Developer Mentors: Dean Gilewicz & Neal Peters
Tools, Skills, & Methodologies: React Native Expo & ExpoGo App Android Studio Google Firestore Database Firebase Cloud Storage

My chat app is a native chat application built with React Native and optimized for both Android and iOS devices. The goal of the app is to demonstrate my knowledge of JavaScript mobile development and to provide users with a chat interface with options to share images and their location.

Key Features:

- Customizable chat screen background
- Additional Communication Features: Image & Location sharing
- Offline message reading
- Screen reader compatibility

Purpose & Context:

This was a personal project I built as part of my web development course at CareerFoundry to demonstrate my understanding of React Native, Expo, Google Firestore/Firebase.

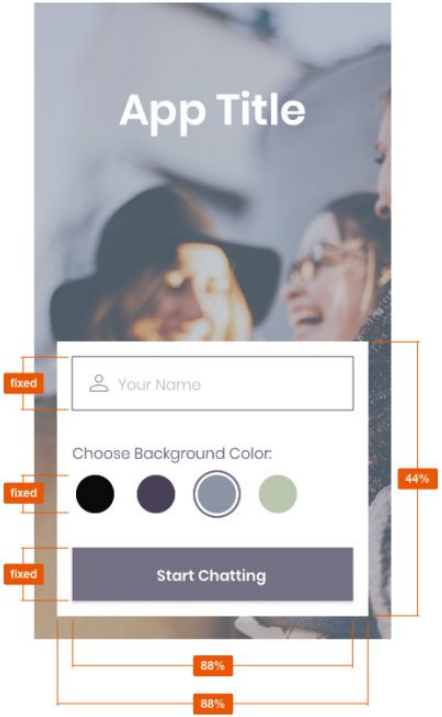
Objective:

- To create a chat application for mobile devices that I could add to my professional portfolio
- Problem to be solved:
    - to build an application that provides users with a chat interface and options to share images and their location, while utilizing tools like Android Studio and the Gifted Chat library

Duration:

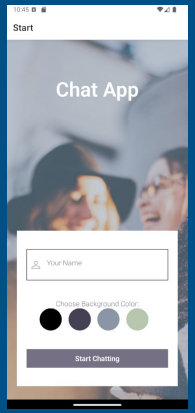
Compared to other JavaScript projects I've completed, I was able to complete this project much quicker than other projects as this was a much simpler project. Additionally, I was provided with precise design specifications for the application.

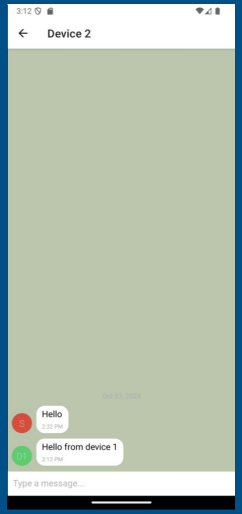
Screen Design & Assets



STEPS

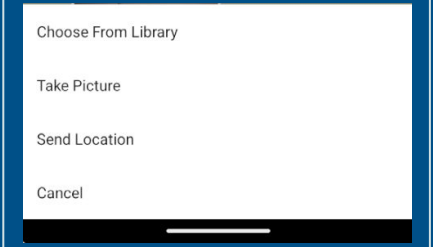
The approach started with learning about the different kinds of mobile web applications, the importance of learning about native mobile app development, and the basics of React Native. With this information, I set up my React Native development environment and the basic chat app structure, like a start page that allows the users to select the background color of the chat screen and a chat page.





I was given a project brief with a set of technical requirements and application features. Using the project brief as a reference, I used React Native's Gifted Chat library to implement an accessible chat UI on the chat page and implemented keyboard adjustments for different devices, like iOS and Android devices.

Next, I initialized Firebase Cloud Storage and Google Firestore Database in my application to enable an anonymous sign-in method for users and to allow users to read and send messages in real time. Then, I implemented client-side data storage using React Native's AsyncStorage to store chats locally, so they're available to users when their devices are offline, and disallowed the creation of new messages when users are offline.



The final features I implemented were additional communication features, such as the ability to share images and location data. In order to implement such features, I added functions to my app that firstly asked the user for permission to access the device's camera and geolocation and gave the app the capability to access said camera and geolocation. Additionally, I created storage in Firebase Cloud Storage for the user's media files, so that other chat participants can view images when they are sent.

- FINAL THOUGHTS -

Takeaways & App Improvements:	What didn't go well?	What went well?
One of the requirements of the project was that the app must authenticate users anonymously via Google Firebase authentication. I feel that this feature is limiting to that app and the user's capabilities for personalization (i.e. personalizing a profile on the app, etc.). In the future, I plan on updating the application with a new create user feature, a new login page, and a profile page. I want these features to include a username, password, email, user image, and potentially other features. I think this will advance the application from being just a simple chat app to a more user-friendly, fun, and useful application.	The most difficult challenge I experienced with this project was working with Expo and the ExpoGo app. Initially when I started working with this project, I was using an older laptop that didn't have a large amount of storage available to utilize, so I was struggling to successfully run the Android Emulator from the ExpoGo app. Fortunately, I was able to get a new laptop that had a much larger storage capacity, so I was able to easily run the Android Emulator and able to complete the project.	This project was one of the final major JavaScript projects, so at this point in the course I was much more familiar with JavaScript. I believe this allowed me to complete this with greater ease than I did with previous projects, even though I had to get a new laptop to complete the project. Additionally, this project rounds out my professional portfolio as this is the only project that demonstrates my knowledge of JavaScript mobile development.