Chatter

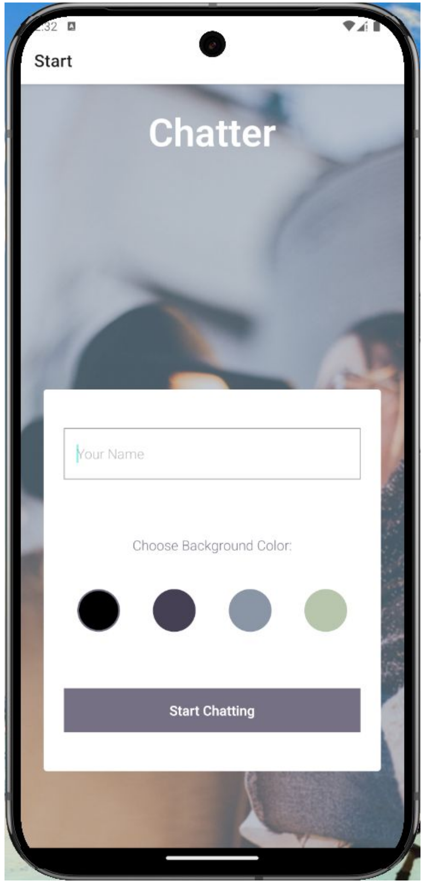
A React Native Case Study
Stephanie Leon

Overview

Chatter is a web application designed using React Native.

Users can join a chat room, choose the background color for the chat screen, and send and receive messages, images and their location.

Messages are able to be read offline and the chat app is compatible with a screen reader.



Purpose & Context

Chatter was created as part of a full stack web development course I completed through CareerFoundry. This was my first opportunity to create a phone application while using React Native and Expo.

Problem Definition

Objective:

The aim of this project was to build a chat app for seamless, real-time communication that supports customization, file sharing, and offline functionality.

Challenges:

- Enable efficient data storage and retrieval
- Ensure the app works across different devices with a consistent user experience
- Incorporate features such as custom actions, media sharing, and offline message handling

Research and Planning

Technology Stack Selection

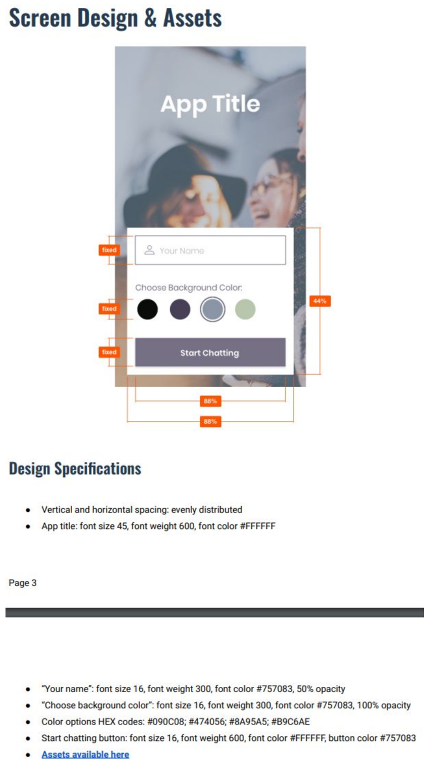
- React Native for cross-platform development
- Firebase for backend and real-time database capabilities
- Gifted Char for UI components

Design Considerations

- User-friendly interfaces
- Scalable architecture for future enhancements

Project Planning

- Define milestones: prototyping, feature development, testing, deployment



Screen Design & Assets

Design Specifications

- Vertical and horizontal spacing: evenly distributed
- App title: font size 45, font weight 600, font color #FFFFFF

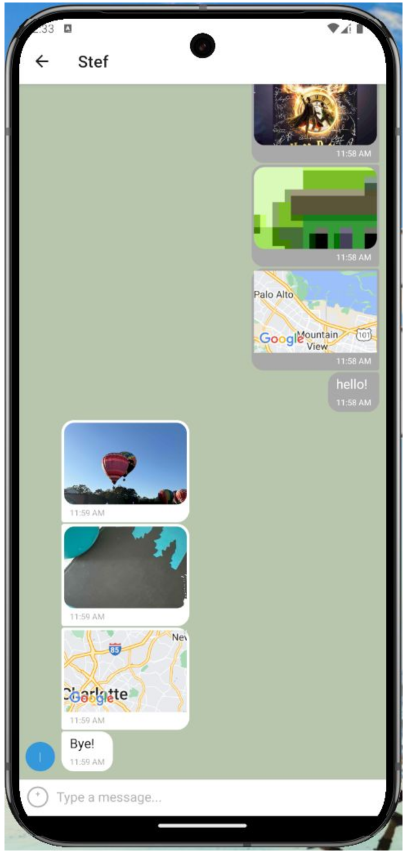
Page 3

- "Your name": font size 16, font weight 300, font color #757083, 50% opacity
- "Choose background color": font size 16, font weight 300, font color #757083, 100% opacity
- Color options HEX codes: #090C08; #474056; #8A95A5; #B9C6AE
- Start chatting button: font size 16, font weight 600, font color #FFFFFF, button color #757083
- Assets available here

Development Process

Prototyping:

- Basic React Native project setup
- Integration of navigation using React Navigation



Core Development:

- Components
- Features
- Backend Integration
- UI/UX

Development Process Cont.

Components:

- Developed key components (Chat.js, CustomActions.js)

Features:

- Custom actions for sharing and document uploads
- Offline functionality using AsyncStorage

Backend Integration:

- Connected Firebase for real-time messaging
- Defined secure authentication flows

UI/UX:

- Styled chat bubbles and messages lists for intuitive use

Configuration:

- Created configuration files (app.json, package.json)
- Ensured compatibility with various screen sizes and platforms

Testing and Debugging

Unit Testing:

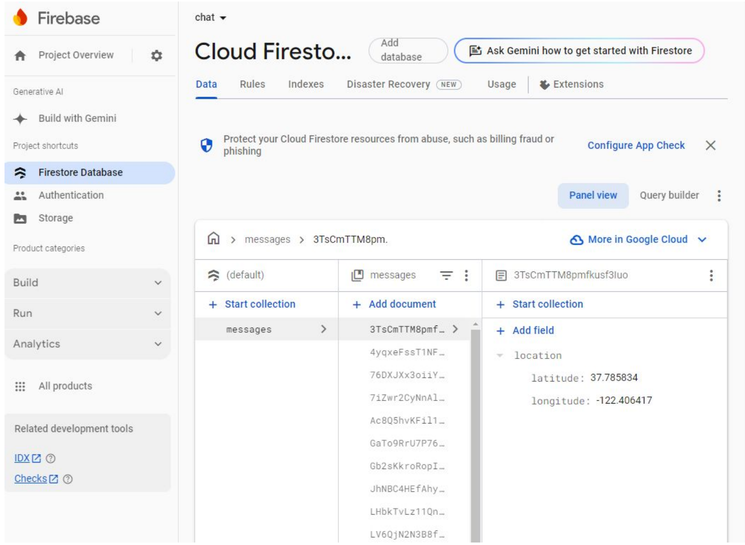
- Tested individual components for functionality
- Validated data flow between components

Integration Testing:

- Verified seamless integration with Firebase
- Ensured custom actions triggered correctly

Performance Testing:

- Assessed app performance under various network conditions
- Optimized for low-memory devices



Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing

messages > 3TsCmTTM8pm.

(default)
messages

messages
3TsCmTTM8pmf…
4yqxeFssT1NF…
76DXJXx3oiiY…
7iZwr2CyNnAl…
Ac8Q5hvKFil1…
GaTo9RrU7P76…
Gb2sKkroRopI…
JhNBC4HEfAhy…
LHbkTvLz11Qn…
LV6QjN2N3B8f…

3TsCmTTM8pmfkusf3luo

location
latitude: 37.785834
longitude: -122.406417

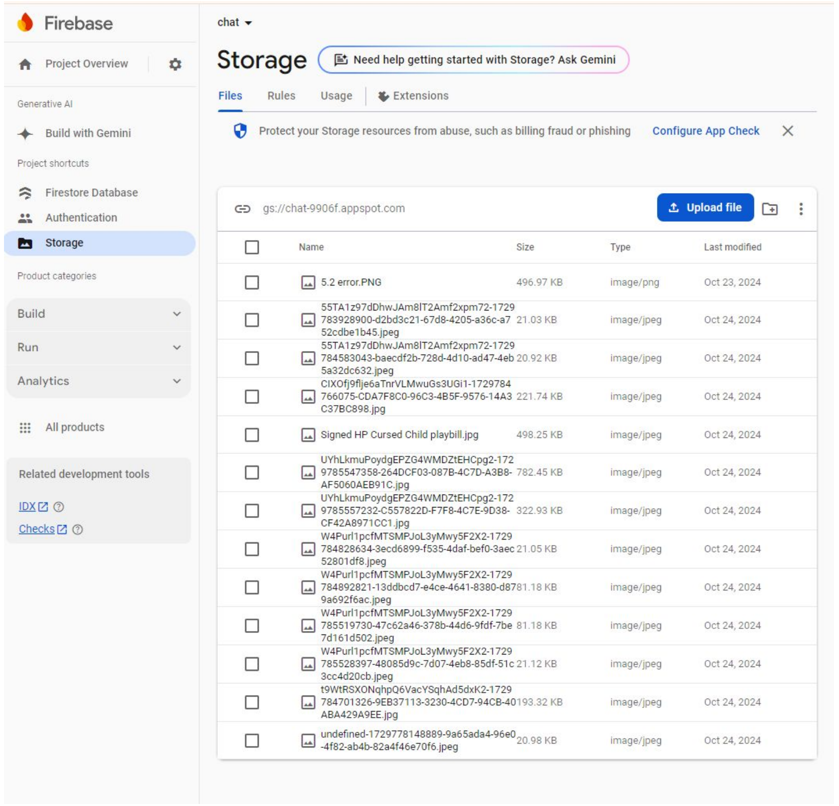
Deployment

Documentation

- Detailed README with instructions for setup, usage, and additional notes for reviewers

Feedback Loop

- Incorporated mentor feedback to improve code quality and user experience



Storage

Protect your Storage resources from abuse, such as billing fraud or phishing

gs://chat-9906f.appspot.com

Name	Size	Type	Last modified
5.2 error.PNG	496.97 KB	image/png	Oct 23, 2024
55TA1z97dDhwJAm8lT2Amf2xpm72-1729783928900-d2bd3c21-67d8-4205-a36c-a752cdbe1b45.jpeg	21.03 KB	image/jpeg	Oct 24, 2024
55TA1z97dDhwJAm8lT2Amf2xpm72-1729784583043-baecdf2b-728d-4d10-ad47-4eb5a32dc632.jpeg	20.92 KB	image/jpeg	Oct 24, 2024
CIXOfj9flje6aTnrVLMwuGs3UGi1-1729784766075-CDA7F8C0-96C3-4B5F-9576-14A3C37BC898.jpg	221.74 KB	image/jpeg	Oct 24, 2024
Signed HP Cursed Child playbill.jpg	498.25 KB	image/jpeg	Oct 24, 2024
UYhLkmuPoydgEPZG4WMDZtEHCpg2-1729785547358-264DCF03-087B-4C7D-A3B8-AF5060AEB91C.jpg	782.45 KB	image/jpeg	Oct 24, 2024
UYhLkmuPoydgEPZG4WMDZtEHCpg2-1729785557232-C557822D-F7F8-4C7E-9D38-CF42A8971CC1.jpg	322.93 KB	image/jpeg	Oct 24, 2024
W4Purl1pcfMTSMPJoL3yMwy5F2X2-1729784828634-3ecd6899-f535-4daf-bef0-3aec52801df8.jpeg	21.05 KB	image/jpeg	Oct 24, 2024
W4Purl1pcfMTSMPJoL3yMwy5F2X2-1729784892821-13ddbcd7-e4ce-4641-8380-d879a692f6ac.jpeg	81.18 KB	image/jpeg	Oct 24, 2024
W4Purl1pcfMTSMPJoL3yMwy5F2X2-1729785519730-47c62a46-378b-44d6-9fdf-7be7d161d502.jpeg	81.18 KB	image/jpeg	Oct 24, 2024
W4Purl1pcfMTSMPJoL3yMwy5F2X2-1729785528397-48085d9c-7d07-4eb8-85df-51c3cc4d20cb.jpeg	21.12 KB	image/jpeg	Oct 24, 2024
t9WtRSXONqhpQ6VacYSqhAd5dxK2-1729784701326-9EB37113-3230-4CD7-94CB-40ABA429A9EE.jpg	193.32 KB	image/jpeg	Oct 24, 2024
undefined-1729778148889-9a65ada4-96e0-4f82-ab4b-82a4f46e70f6.jpeg	20.98 KB	image/jpeg	Oct 24, 2024

Final Solution

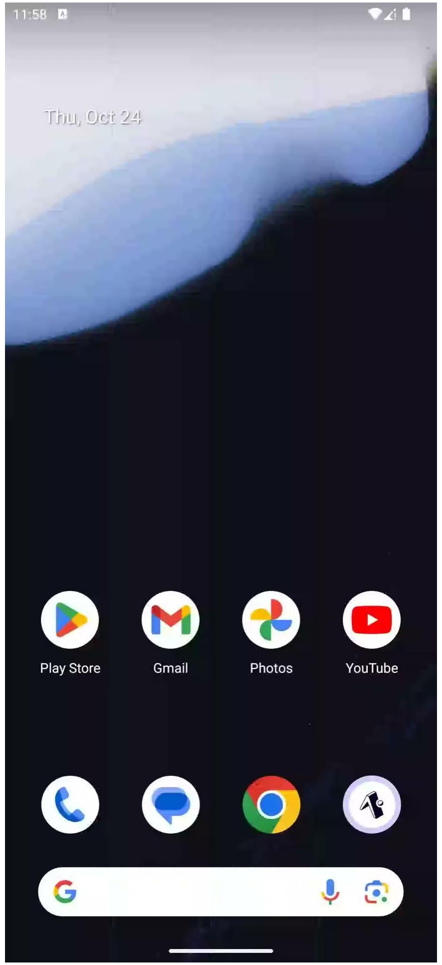
Deliverable:

- A fully functional React Native chat app with the following capabilities:
  - Real-time messaging powered by Firebase
  - Media sharing via custom actions
  - Offline support for seamless user experience
- Scalable architecture for adding new features

Outcome:

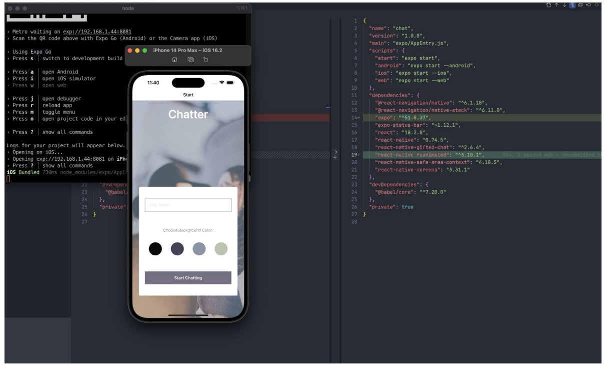
- Successfully launched the app, meeting the defined objectives

Click to see a video of the app and it's features! →



Lessons Learned

- Make sure the versions of each package being used are compatible:
  - I ran into issues early in the project with Expo saying it needed to be updated and the fixes provided in the console did not work. I needed to run npm i expo@latest and npm install react-native-reanimated@3.10.1 to fix the issues I was having.
- Using branches in GitHub is helpful when problem solving.
  - I was able to ask for feedback from my tutor, make a new branch and continue working on other features while they reviewed my work.



```
› Metro waiting on exp://192.168.1.44:8081
› Scan the QR code above with Expo Go (Android) or the Camera app (iOS)

› Using Expo Go
› Press s │ switch to development build

› Press a │ open Android
› Press i │ open iOS simulator
› Press w │ open web

› Press j │ open debugger
› Press r │ reload app
› Press m │ toggle menu
› Press o │ open project code in your ed

› Press ? │ show all commands

Logs for your project will appear below.
› Opening on iOS...
› Opening exp://192.168.1.44:8081 on iPh
› Press ? │ show all commands
iOS Bundled 730ms node_modules/expo/AppEn
```

```
{
  "name": "chat",
  "version": "1.0.0",
  "main": "expo/AppEntry.js",
  "scripts": {
    "start": "expo start",
    "android": "expo start --android",
    "ios": "expo start --ios",
    "web": "expo start --web"
  },
  "dependencies": {
    "@react-navigation/native": "^6.1.18",
    "@react-navigation/native-stack": "^6.11.0",
    "expo": "^51.0.37",
    "expo-status-bar": "~1.12.1",
    "react": "18.2.0",
    "react-native": "0.74.5",
    "react-native-gifted-chat": "^2.6.4",
    "react-native-reanimated": "^3.10.1",      You, 1 second ago • Uncommitted c
    "react-native-safe-area-context": "4.10.5",
    "react-native-screens": "3.31.1"
  },
  "devDependencies": {
    "@babel/core": "^7.20.0"
  },
  "private": true
}
```

Future Iterations

- Additional features like recording and playing sounds, uploading an audio file and sending an audio file
- Users able to have multiple chats
- The ability to have a group chat between users

Duration:

I completed the Chat app in half a month.

Credits:

Lead Developer: Stephanie Leon
Tutor: Jesus Diaz
Mentor: Shreyansh Kumar