

<School of Computing & Information Sciences>

Summer 2024 Capstone II Project

<Blog Post>

Student: Josue Arreaga

Product Owner: Alvaro Licea

Instructor/Faculty: Dr. Masoud Sadjadi, Florida International University

PROBLEM

Within a communication app, the crux of the functionality lies within the ability to communicate with users effectively. A message channel should be created to communicate with as many users as one wants to communicate ideas to. If the number of users increases, the communications stream needs to be scalable to support the increasing conversations. This facilitates the need for the development of a communication protocol.

Communication among users should allow users to manage the messages they say or highlight messages, so messages are not lost across the communication history. The messages should also be preserved over time so users can reference previous conversations as needed.

Users should be able to communicate utilizing different mediums to each other, because text communications can be limiting. Messaging should support sending multiple media types, such as images to allow users to convey greater thought.

SYSTEM DESIGN

Supabase was utilized to manage the backend. It supports the creation of users which serve as the basis for conversation.

React Native was utilized to implement the front end and manage the user interface for the messaging platform.

Stream was utilized to facilitate user communications, maintain record of message history, and implement post management.

APP DESIGN

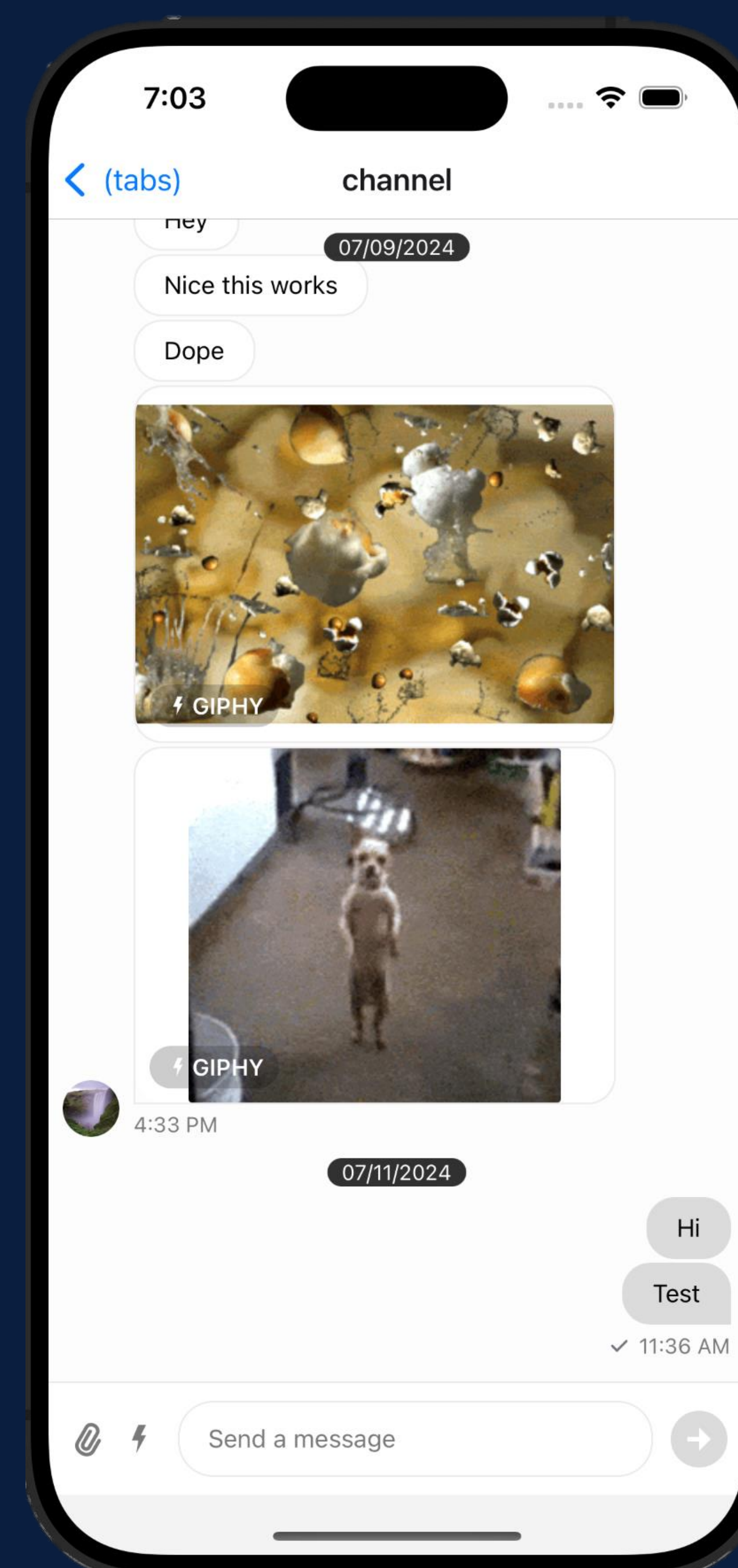


Figure 1: Chat channel between two users. Supports image and file sharing functionality.

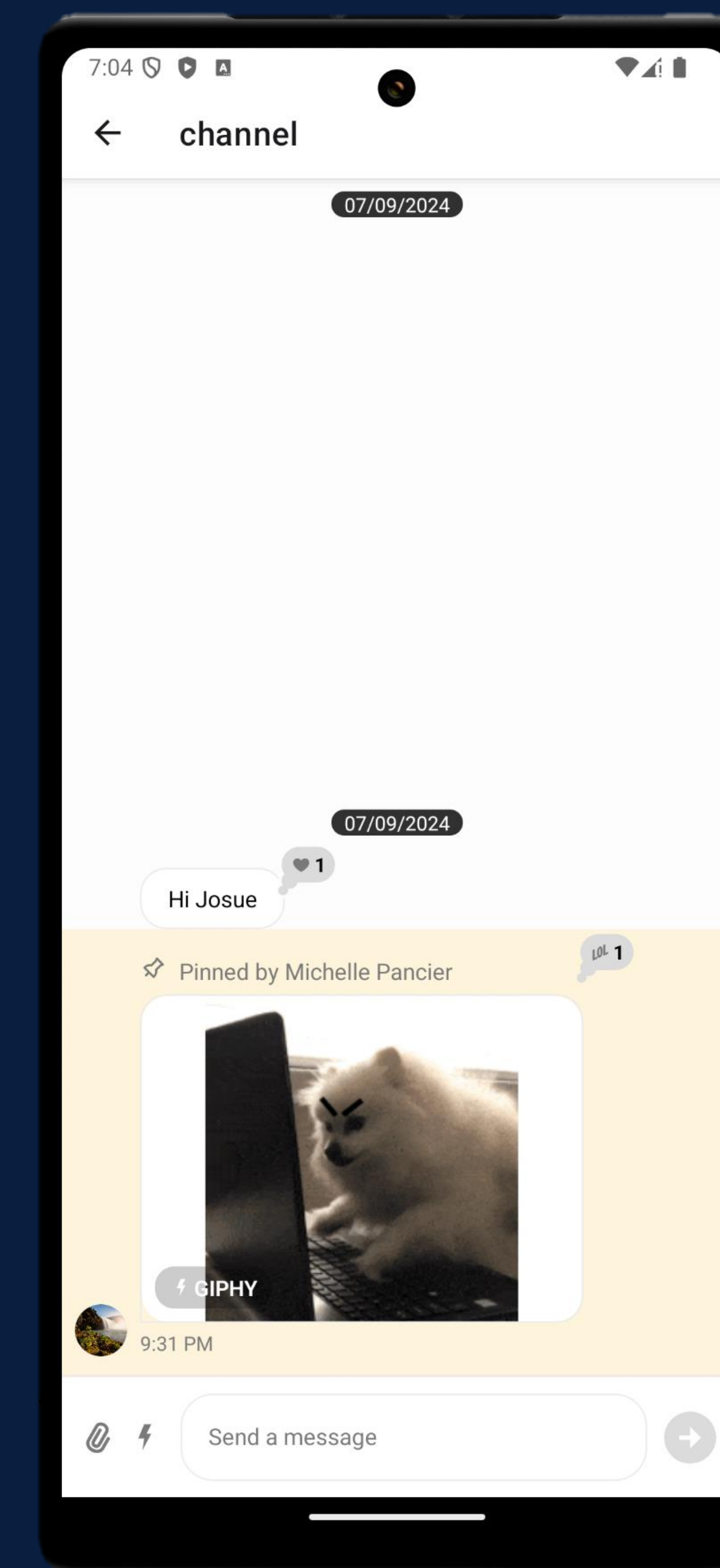


Figure 2: Users reacting and bookmarking messages they deem as important.

REQUIREMENTS

To run the application, ensure the following programs are installed:

- VS Code
- Node.js
- React Native With Expo Go
- Android or iOS emulators
- Physical iOS or Android device with latest updates
- Brew
- Watchman

IMPLEMENTATION

- Created the user messaging system. This involved the implementation of the Stream interface.
- Facilitated the communication of text components, images, and gifs to other users on top of the Stream interface.
- Implemented post management functionality. This allows users to react and highlight messages they deem as important and delete messages that they did not intend to send. This was implemented atop of the Stream interface as a base.
- Collaborated with other team members to refine the front end and user interface of the application, alongside helping to implement cross platform support.

SUMMARY

In summary, the objective of this application is to facilitate communications across multiple users with a robust and accessible platform. User communications across is implemented alongside the ability to manage messages and engage with messages through multiple means. The messaging system implemented can be further enhanced to facilitate forum like communication of likeminded individuals. Additional work towards the application can be allocated to extending the communication channels not only across users but also around common communities.



ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by Masoud Sadjadi. I thank Masoud Sadjadi for his assistance and mentorship that I received throughout the Capstone II project.