

Mobile Application Developer Assignment

GIPHY Viewer

Name: Vishal Venkatraman

Phone: +852 67692044

Email: vishal.venkatraman97@gmail.com

Address: 26G, Tower 5, Rambler Crest, 1 Tsing Yi Road, Tsing Yi, New Territories, Hong Kong

Introduction

GIPHY Viewer is an Android application that displays various GIFs based on the user's search requests. This is accomplished by consuming the Search API provided by GIPHY. For its implementation, React Native was chosen for two reasons. Firstly, its hybrid nature could be useful in the future should the application be ported to iOS. Secondly, React Native is part of a rapidly growing ecosystem of programming resources, and it was thus chosen over the other options.

Design

In keeping with the latest paradigm in React, the entire application was implemented with Hooks: not only did this choice result in a more concise codebase, but it also affords the potential for future growth, what with the widespread adoption of Hook-based architecture in the JavaScript community.

Given the straightforward nature of the project, the only external library used was NativeBase: it is the most widely adopted UI components library available to React Native, and its battle-tested nature made it the most plausible choice. The logical portions of the codebase were implemented using vanilla React Native resources.

Approach

The UI was conceived first, as having an image of the final product aids with the decomposition of the design logic into manageable sub-portions. Indeed, the draft of the UI was followed by its rough implementation. Following this, the GIPHY API was integrated into the application, as it would not be reasonable to implement the required functionality without the data to be consumed.

Finally, the required functionality was implemented and tested manually, owing to the lack of time for rigorous and automated testing.

Business Outcome

GIPHY Viewer, though limited in its current form, can be improved to offer community-building and social networking. Short-form multimedia (such as meme videos, and GIFs) is popular among the growing demographic of children and young adults, and this application can be adapted to serve such content as that found on TikTok, Instagram, and similar platforms.

Go Live

The API key used in this development instance of the GIPHY Viewer limits the number of results that can be derived from the API per instance, so it must be upgraded to a production-level key. Additionally, given less that one work

day to complete the application, it was not possible to conduct extensive testing on the finalised product. Indeed, the following items must be addressed prior to an official launch.

1. The deletion of saved GIFs is not working as required: instead of removing the chosen GIF, everything other item is purged.
2. It was not possible to implement the scroll-and-load functionality that was requested.
3. A more robust implementation of the application would have included Redux, which provides tools for state management. Indeed, with the data intensive nature of the API, it would have been better to use Redux instead of relying on the barebones state tools provided by React.

Source Code

The following is a GitHub link with the required source code.

<https://github.com/TheConMiz/gifhyViewer>

Should the .apk file provided in the repository not work on test device, consider cloning the repository and running it on an Android Emulator provided by Android Studio.

Additionally, the following is a link to a video demonstrating the application's features and UI/UX.

<https://youtu.be/BYRyu5z9L84>