Assignment 2, Mobile development

Rafael Baimurzin

18.10.2024

Table of content

Introduction

This project involved developing a simplified version of Instagram using Android Studio with Jetpack Compose. The objective was to create a functional mobile app that mimics the core features of Instagram, including a Home Feed, Profile Page, Search Page, Add Post Page, and Notifications Page. The purpose was to learn about modern Android development practices, such as working with Compose, implementing user interactions, and handling navigation in a mobile app.

Project Setup

The initial setup of the project began by creating a new Android Studio project using the "Empty Compose Activity" template. Material components and jetpack compose were added to the build.gradle file. Additionally, internet permissions were declared in the AndroidManifest.xml to load external images for the app.

Page design

The Home Feed page displays a list of posts using a LazyColumn. Each post includes an image, the username of the user who posted it, a caption, and the number of likes. Data for each post is managed using a Post data class, and interactions like liking a post are handled using mutable states.

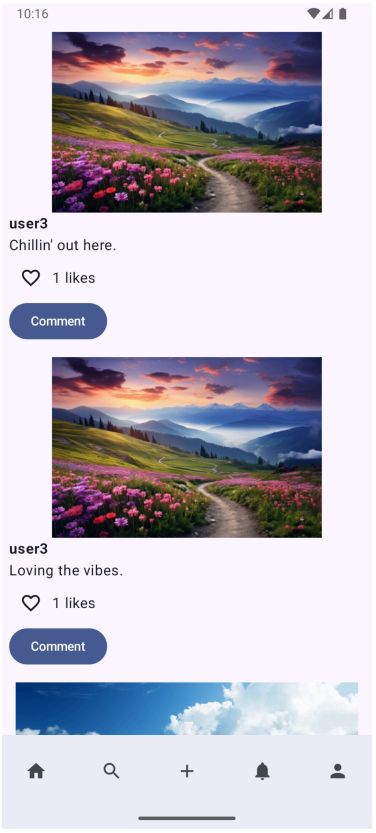

Figure 1. Home screen

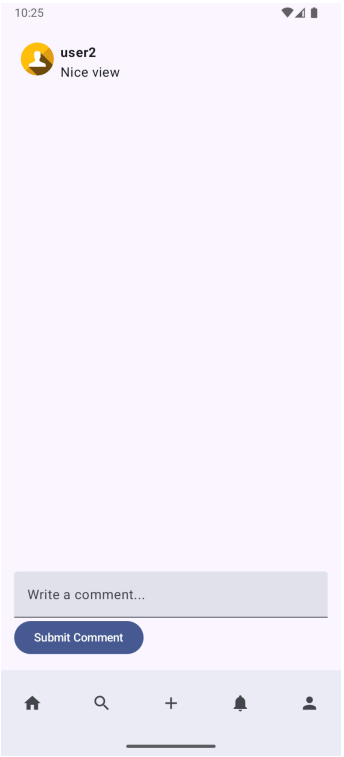

Figure 2. Comment section

The Profile Page shows user-specific information such as their profile picture, bio, and posts count. The user's posts are displayed in a grid layout using a `LazyVerticalGrid`. The `User` data class was used to manage the user details.

Figure 3. User profile

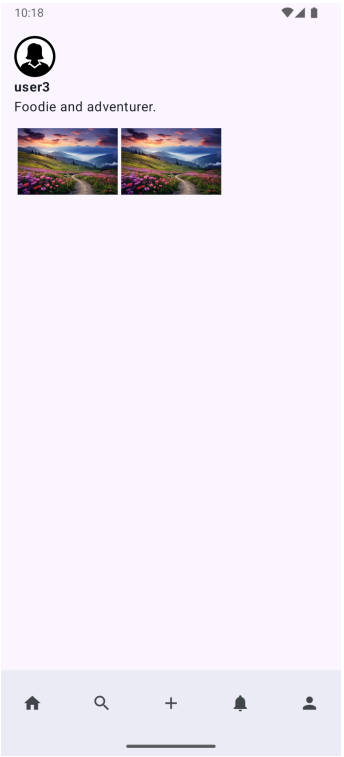

Figure 4. Other user profile

The Search Page contains a search bar implemented with a `TextField` for users to enter queries. Below the search bar, search results are displayed using a `LazyColumn`. Users can navigate to individual profiles by clicking on the search results.

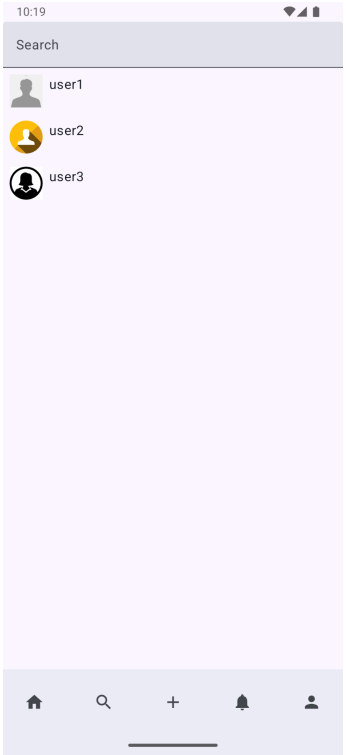

Figure 5. Search screen

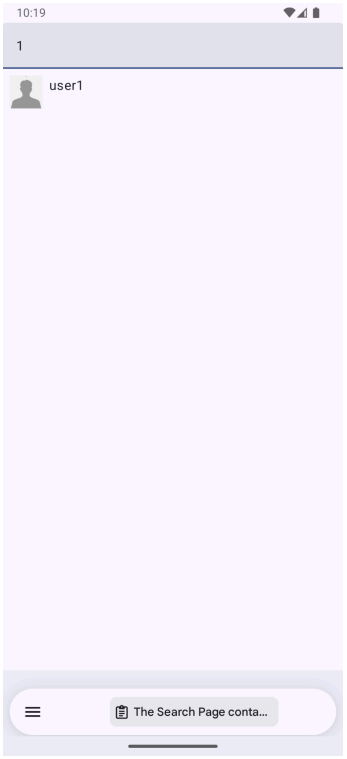

Figure 6. Search result

The Add Post Page allows users to upload a new post by entering an image URL and a caption. It contains two input fields, both implemented using `TextField` components. After entering the details, users can submit the post, which is then added to the list of existing posts.

Figure 7. Create Post Screen

The Notifications Page displays a list of notifications (e.g., likes and comments on posts). The notifications are displayed in a simple `LazyColumn` with each notification represented as a text item.

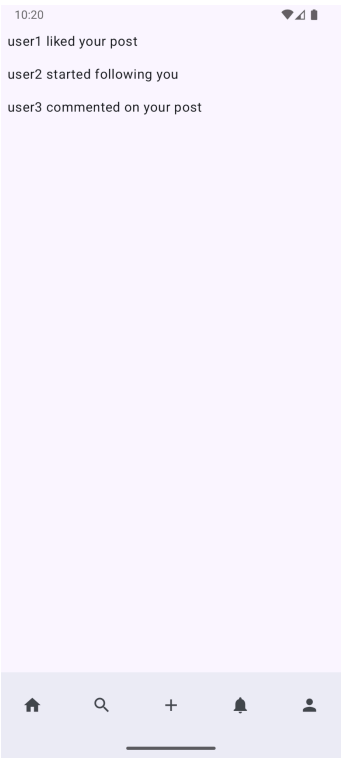

Figure 8. Notification Screen

Navigation

Navigation between pages is implemented using the Jetpack Navigation Component. A `NavController` is used to manage navigation actions throughout the app, and the bottom navigation bar allows users to switch between the main sections of the app, including Home Feed, Search, Add Post, Notifications, and Profile. The `NavHost` is used to define the various routes, and navigation actions are triggered through `NavController` commands. This helped maintain a structured navigation flow, making the app intuitive for users.

User Interaction

User interactions are handled using Compose's `clickable` and `IconButton` modifiers. Users can like a post by clicking on the like button. A mutable state (`isLiked`) tracks whether the post has been liked. Users can click on the comment button, which navigates to a dedicated comment page where they can add comments. Users can click on a username to navigate to that user's profile page.

Challenges and Solutions

Initially, implementing navigation between different pages was challenging due to unfamiliarity with Jetpack Navigation. This was resolved by reading the documentation and experimenting with the `NavController` and `NavHost` components. Keeping track of states for features like liking a post or submitting a comment required careful consideration. This challenge was addressed by using Compose's `remember` and `mutableStateOf` to properly handle UI state changes. Passing data between composables, such as user information or posts, was challenging at first. Breaking down the composables into smaller reusable functions and managing state at the higher levels helped ensure data was properly propagated throughout the UI.

Conclusion

This assignment provided valuable experience in building a feature-rich mobile application using Jetpack Compose. Key learnings include the importance of managing navigation, state, and UI components effectively to deliver a seamless user experience. The project emphasized the significance of UI/UX in mobile development, as intuitive design and smooth interactions greatly contribute to the overall appeal and usability of an app.