Basic Instagram Pages in Android Studio

Makhatbek Iliyas
School of Information Technology and Engineering
Kazakh-British Technical University
Almaty, Kazakhstan
Email: il_makhatbek@kbtu.kz

18 October 2024

Contents

1 Introduction **2**

2 Project Setup **2**

3 Page Design **2**

3.1 Home Feed 2

3.2 Profile Page 3

3.3 Search Page 4

3.4 Add Post Page 4

3.5 Notifications Page 5

4 Navigation **5**

5 User Interaction **6**

6 Challenges and Solutions **6**

7 Conclusion **7**

8 References **7**

1 Introduction

The purpose of this assignment was to create a simplified version of Instagram using Android Studio. The main focus was on implementing the core pages of Instagram, such as the Home Feed, Profile, Search, Add Post, and Notifications pages. This project helped me understand how to design layouts, handle navigation, and simulate user interaction within an app.

2 Project Setup

I started by creating a new Android project in Android Studio using the Empty Activity template. I chose Kotlin as the programming language for the project and set the minimum API level to 21 (Android 5.0).

To ensure the project could function smoothly, I added the necessary dependencies, such as: Material Components for UI elements, RecyclerView for displaying lists, Navigation Component to manage fragment navigation.

Here's an example of my `build.gradle` file:

```
dependencies {
    implementation 'com.google.android.material:material:1.9.0'
    implementation 'androidx.recyclerview:recyclerview:1.2.1'
    implementation
    ↪  'androidx.navigation:navigation-fragment-ktx:2.3.5'
    implementation 'androidx.navigation:navigation-ui-ktx:2.3.5'
}
```

3 Page Design

3.1 Home Feed

The Home Feed page displays a list of posts using a RecyclerView. Each post includes an image, username, caption, and a like button. To create this, I designed an XML layout for individual posts and then used a RecyclerView adapter to populate the feed.

For the layout, here is an example of the `item_post.xml` file:

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="8dp">
    <ImageView
        android:id="@+id/post_image"
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:scaleType="centerCrop" />

    <TextView
        android:id="@+id/post_username"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Username" />

    <TextView
        android:id="@+id/post_caption"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Caption" />
</LinearLayout>
```

This layout is used in the RecyclerView to show each post in the feed.

3.2 Profile Page

The Profile page shows the user's profile picture, username, bio, and a grid of their posts. I used a grid-style RecyclerView to display the posts in a similar way to Instagram's profile page.

In `fragment_profile.xml`, the structure looks like this:

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <ImageView
        android:id="@+id/profile_picture"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:src="@drawable/ic_profile" />

    <TextView
        android:id="@+id/username"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Username" />

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/profile_posts"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:numColumns="3" />
</LinearLayout>
```

3.3 Search Page

For the Search Page, I included an EditText as a search bar and used a RecyclerView to display user profiles in the search results. Each result shows the user's profile picture and username.

3.4 Add Post Page

The Add Post Page allows the user to select an image and add a caption to their post. There's an ImageView that shows a preview of the post and an EditText where the user can write a caption. A button is provided to simulate posting the content.

Here is a simplified version of the layout:

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <ImageView
        android:id="@+id/post_image_preview"
        android:layout_width="match_parent"
        android:layout_height="200dp"
        android:src="@drawable/ic_add_photo" />

    <EditText
        android:id="@+id/post_caption"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Write a caption..." />
    <Button
        android:id="@+id/btn_upload_post"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Upload" />
</LinearLayout>
```

3.5 Notifications Page

The Notifications Page lists notifications such as likes and comments using a RecyclerView. Each notification is represented by a simple text that describes the action.

4 Navigation

To navigate between the pages, I used the BottomNavigationView along with the Navigation Component. The bottom navigation allows users to easily switch between the Home Feed, Search, Add Post, Profile, and Notifications pages.

Here's the bottom_nav_menu.xml file that defines the navigation items:

```
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/homeFragment"
     ↪  android:icon="@drawable/ic_home" android:title="Home" />
    <item android:id="@+id/searchFragment"
     ↪  android:icon="@drawable/ic_search" android:title="Search"
     ↪  />
    <item android:id="@+id/addPostFragment"
     ↪  android:icon="@drawable/ic_add" android:title="Add Post"
     ↪  />
    <item android:id="@+id/profileFragment"
     ↪  android:icon="@drawable/ic_profile"
     ↪  android:title="Profile" />
    <item android:id="@+id/notificationsFragment"
     ↪  android:icon="@drawable/ic_notifications"
     ↪  android:title="Notifications" />
</menu>
```

In the MainActivity.kt, I connected the bottom navigation with the navigation controller like this:

```
val navController = findNavController(R.id.nav_host_fragment)
val bottomNav =
↪  findViewById<BottomNavigationView>(R.id.bottom_navigation)
bottomNav.setupWithNavController(navController)
```

5 User Interaction

To handle user interactions, such as clicking on a post or liking a post, I used click listeners. For example, in the Home Feed, when a username is clicked, the app navigates to the Profile Page of that user.

Here's an example of setting a click listener in `HomeFragment.kt`:

```
recyclerViewAdapter.setOnItemClickListener { post ->
    val action =
    ↪  HomeFragmentDirections.actionHomeToProfile(post.userId)
    findNavController().navigate(action)
}
```

6 Challenges and Solutions

One challenge I faced was implementing the navigation between fragments. Initially, the app would crash when navigating between certain pages because I didn't properly set up the navigation graph. I fixed this by carefully organizing the navigation actions and ensuring all the fragment IDs were correctly referenced in the navigation graph.

Another issue was managing the RecyclerView for the Home Feed. At first, the posts wouldn't display correctly because I hadn't set the layout manager. After setting it to `LinearLayoutManager`, the problem was resolved.

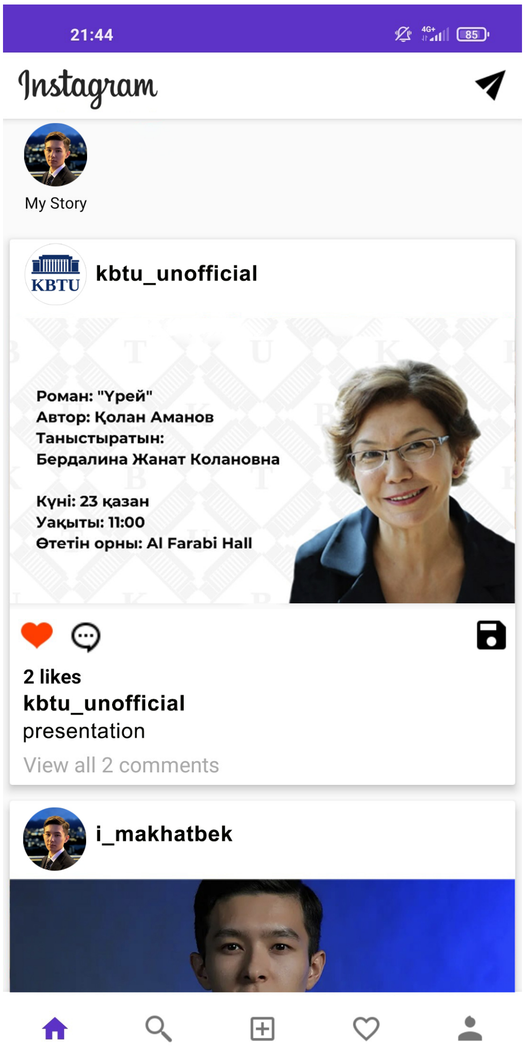
Additionally, I encountered challenges while implementing the image upload feature on the Add Post Page. Integrating image selection from the device's gallery required understanding how to use Intents and handle the results properly. I learned to implement an `onActivityResult` method to receive the selected image and display it in the ImageView.

7 Conclusion

Through this assignment, I learned a lot about Android development, particularly about creating user interfaces and managing fragment navigation. It also gave me insights into the importance of UI/UX design in mobile applications. Building the basic Instagram pages helped me understand how to work with RecyclerView and manage interactions effectively.

8 References

- Material Components Documentation
- RecyclerView Tutorial
- Android Navigation Component Guide

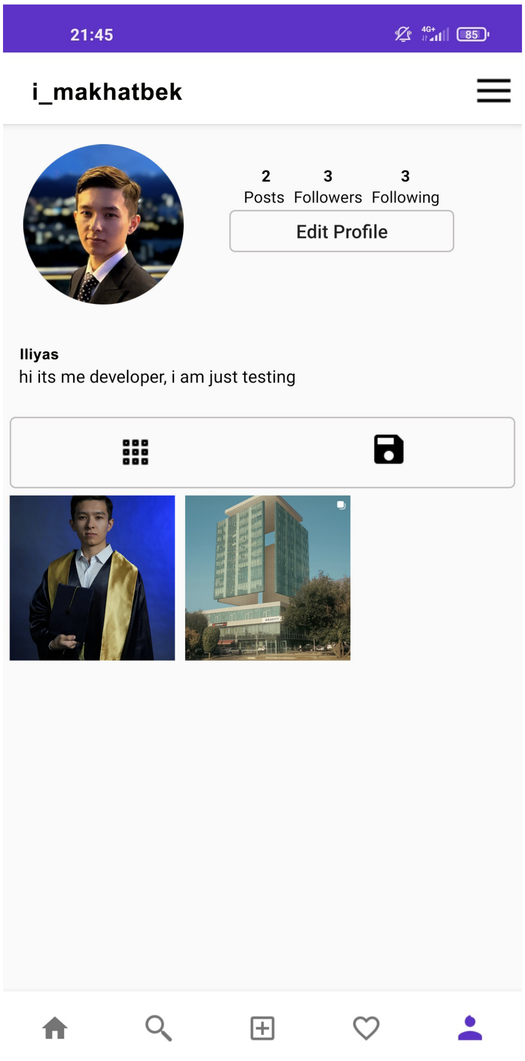


Роман: "Үрей"
Автор: Қолан Аманов
Таныстыратын:
Бердалина Жанат Колановна

Күні: 23 қазан
Уақыты: 11:00
Өтетін орны: Al Farabi Hall

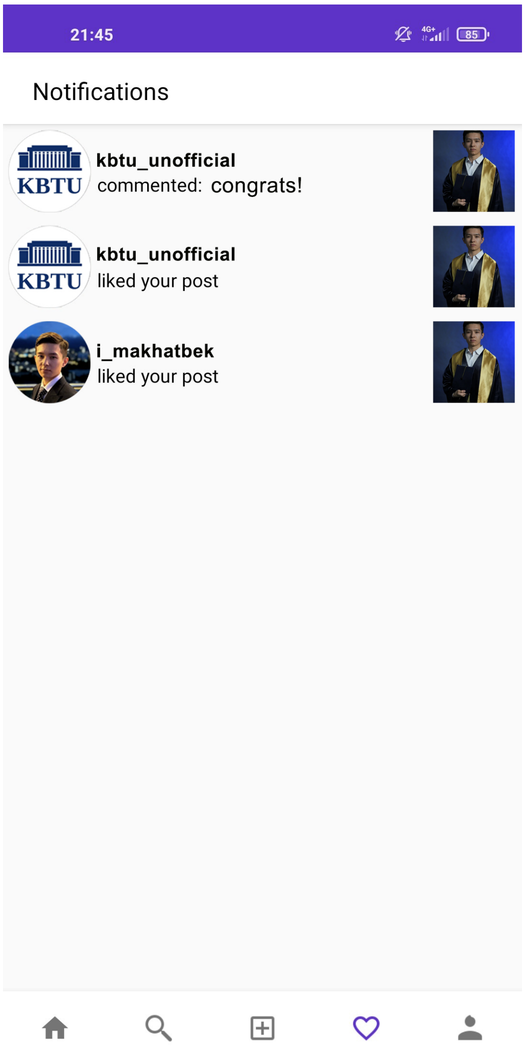
2 likes
kbtu_unofficial
presentation

Figure 1: Home Feed Screenshot



Iliyas
hi its me developer, i am just testing

Figure 2: Profile Page Screenshot



21:45

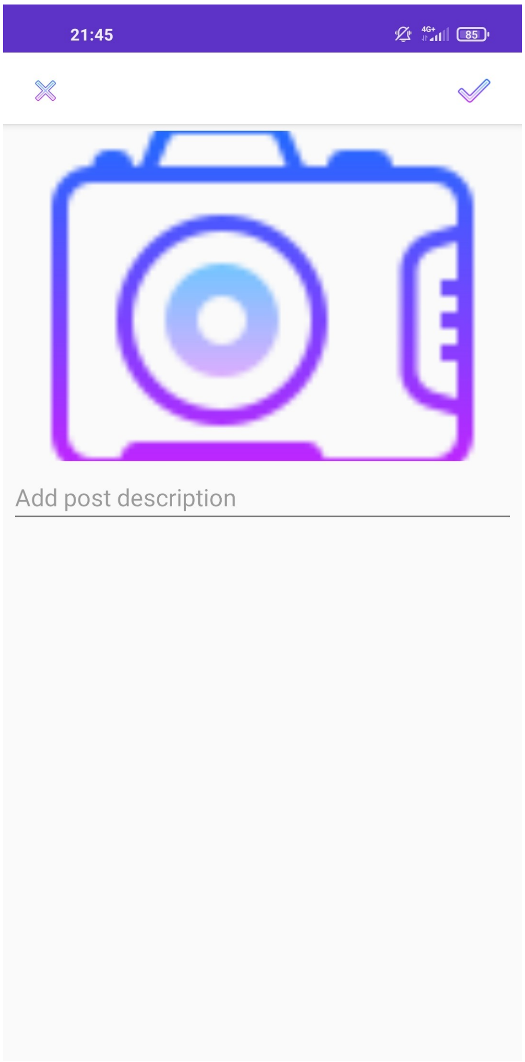
Notifications

kbtu_unofficial
commented: congrats!

kbtu_unofficial
liked your post

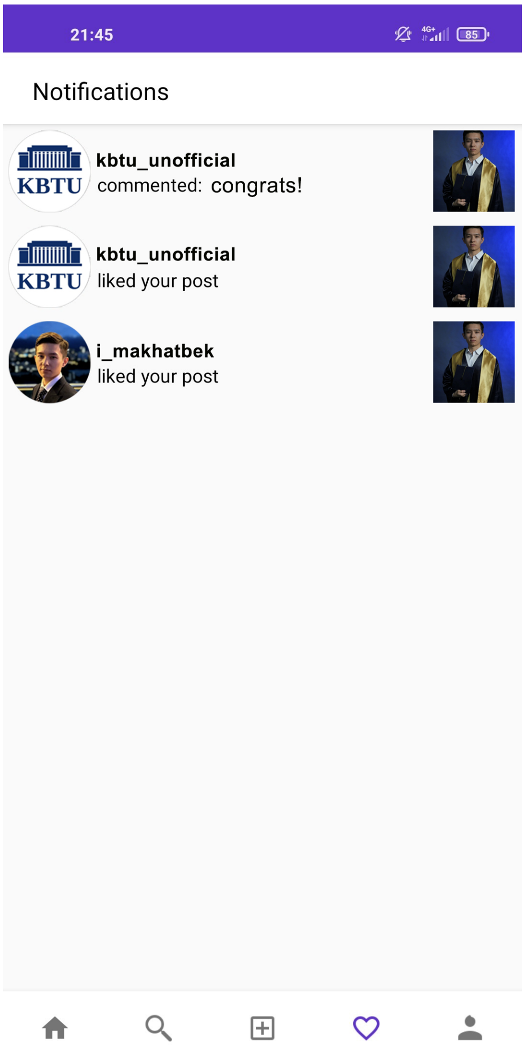
i_makhatbek
liked your post

Figure 3: Search Page Screenshot



Add post description

Figure 4: Add Post Page Screenshot



21:45

Notifications

kbtu_unofficial
commented: congrats!

kbtu_unofficial
liked your post

i_makhatbek
liked your post

Figure 5: Notifications Page Screenshot