

MAD ASSIGNMENT 6

DATE: 08-04-2023 (WEEK 6)

ASSIGNMENT DETAILS

1. There are 5 screens, UsersScreen, UserProfileScreen, AddPostScreen, PostDetailScreen & SettingsScreen.
2. Add a top bar on all screens (and appropriate navigation icons wherever required)
3. Add a Bottom bar containing icons (and navigation) for UsersScreen and SettingsScreen. (Hide the bottom bar except these 2 screens - only if you have enough time)
4. You can use icons as per your wish. Here's the sample site to download icons. (<https://www.flaticon.com/uicons/interface-icons>)
5. You must use Jetpack Compose to build the UI.

UsersScreen

1. Fetch the paginated list of users from given API and display them accordingly using Vertical Grid.
2. Upon clicking on the user, navigate to UserProfileScreen.

UserProfileScreen

1. Fetch the user details and posts from given API and display them accordingly.
2. Posts are non-paginated list (that is, you don't need pagination).
3. Upon clicking on post, navigate to PostDetailScreen.
4. Use a Floating Action Button (FAB) that navigates to AddPostScreen.

PostDetailScreen

1. Fetch the post detail and display it accordingly.
2. Also fetch the user of the post and show the name.
3. Upon clicking on the delete icon in the top bar, schedule the post deletion after 5 minutes, display a toast and navigate back.
4. For post deletion, send a DELETE request to server.

AddPostScreen

- Accept image url, caption, likesCount and commentsCount in TextField.
- When user clicks on **Create post** button, apply validation on these fields (such as string is not empty, number is not negative etc) and notify user if anything is wrong.
- If all validations pass, automatically generate the remaining fields (such as id, createdAt) and use the proper userId to create a post.
- Once post creation request is successful, go back to UserProfileScreen.

SettingsScreen

1. Allow users to select app language as English or Hindi (Those who don't know Hindi, can refer to Google Translate)
2. Upon selecting the language, it should persist even if user restarts the app.

API INFO

Below is the URL for each endpoint, along with the supported operations.

If you want to create any resource (POST request), please use the same number of parameters that you'll get in GET request. If it responds with error, do let me know.

For documentation, please refer <https://mockapi.io/>

Base URL

<https://64302d24b289b1dec4c30671.mockapi.io/api/v1/>

Users (GET/POST)

`users?page=1&limit=20`

User details (GET)

`users/{userId}`

User posts (GET/POST)

`users/{userId}/Post`

Post details (GET/DELETE)

`users/{userId}/Post/{postId}`

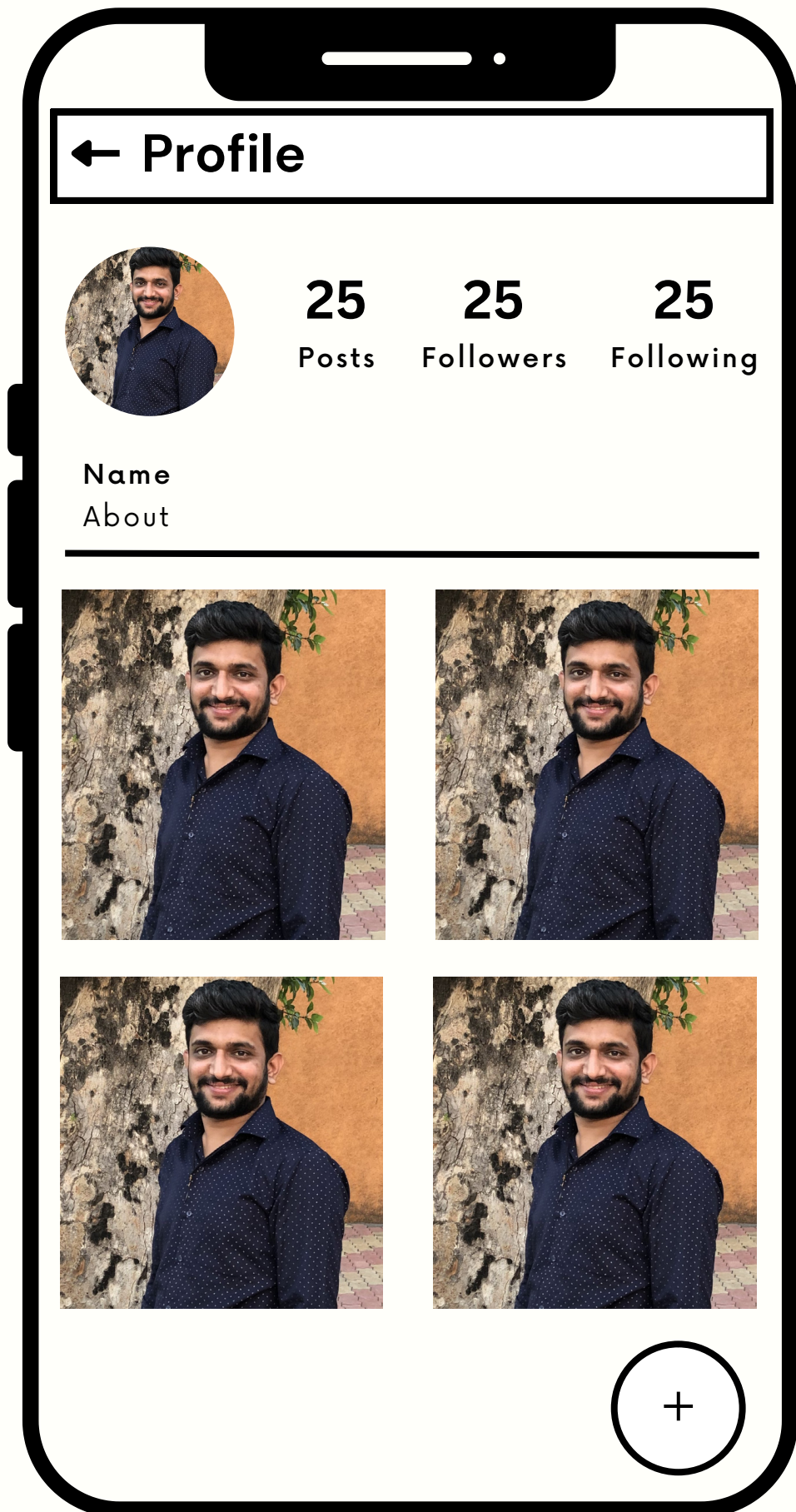
SUBMISSION

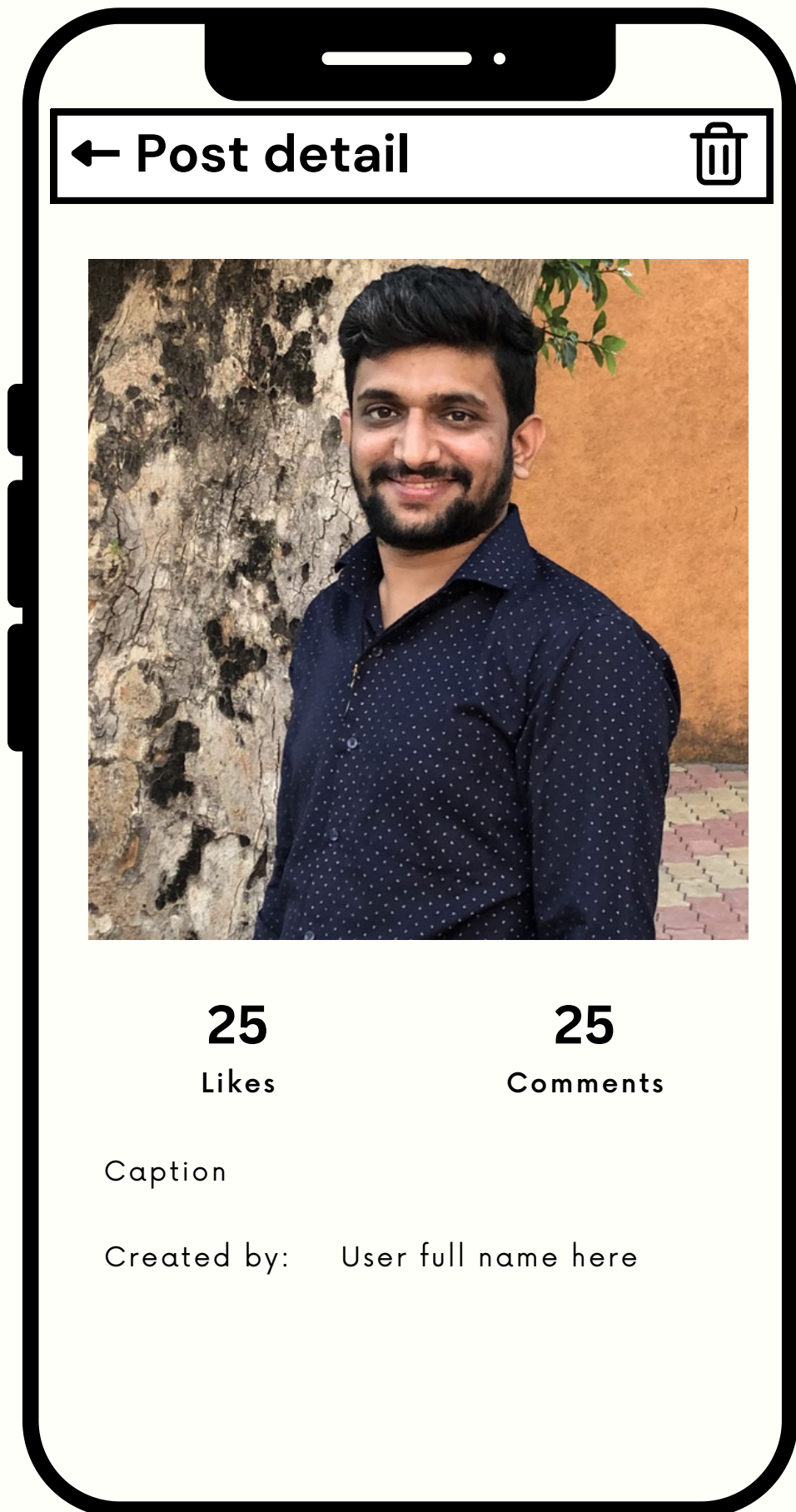
1. Upload your project on GitHub. You **must have to use a private repository** that is only accessible to you. (Skip if already done in previous assignment)
2. When you finish your project, assign access to me so I can review the code. This is a one-time only procedure. (Skip if already done in previous assignment)
3. Write a LinkedIn post about the assignments and tag me (<https://www.linkedin.com/in/hadiyarajesh>). Create a demo video of your app's behavior and upload it with the post. This is optional but **recommended**.
4. Submit your assignment by 16 April, 2023 EOD.

REVIEW

1. I'll review the code and assign you a score. (Private to me)
2. Based on the ranking, I'll announce top 3 learners and initiate full refund to them.







← Add post

URL

Caption

Likes

Comments

Create post

