ARCHITECTURE DOCUMENT

Product/Project Title: Qurb App

Project Team Members:

1. Mohammad Affan Ullah Habib (mh06358)
2. Syed Muhammad Daniyal Murtaza Zaidi (sz06880)
3. Syed Muhammad Hussain (sh06892)

INTRODUCTION:

Our project is an application offering hyperlocal social networking services for neighborhoods.

Qurb (Urdu for Proximity) provides neighbors with an exclusive platform to connect, share information and befriend those in close proximity. The product will have all your neighborhood information in one place. No need to add neighbors on personal apps like Facebook or WhatsApp.

We intend to employ Firebase as our backend technology and Flutter (Dart) as our frontend technology. We have settled on this stack. These two technologies are ideally suited for our product-based solution. We chose Firebase as our backend and database mostly because it provides a straightforward yet powerful user management and authentication mechanism, which was what we were looking for. Also, Firebase does a great job of handling storage. These are all essential elements of a fantastic yet straightforward social media platform. Flutter enables us to create incredibly precise and aesthetically beautiful user interfaces, which we want to include in our application.

FUNCTIONAL & NON-FUNCTIONAL REQUIREMENTS:

Functional Requirements:

- Sign Up/Registration
- Login
- Comment
- Posting (images/text)
- Reactions
- Displaying Profile Data
- Searching and Following Different Users

Non-Functional Requirements:

- User Authentication
  Authentication is a key aspect of our social media app.
- Password Encryption
  Mainly to avoid security breaches/privacy invasions.
- Usability/Reliability
  The social media app should be easy to use and navigate, with intuitive user interfaces that allow users to quickly and easily perform the desired actions. The social media app should be reliable and available 24/7 to users.
- Scalability
  The social media app should be designed to handle a large number of users and their data without any degradation in performance.
- Compatibility
  The social media app should be compatible with various devices, platforms, and operating systems to provide users with a consistent experience.
- Accessibility
  The social media app should be accessible to all users, including those with disabilities.
- Maintainability

The social media app should be easy to maintain and update, with well-documented code and a modular architecture that allows for easy changes.

USE CASES:

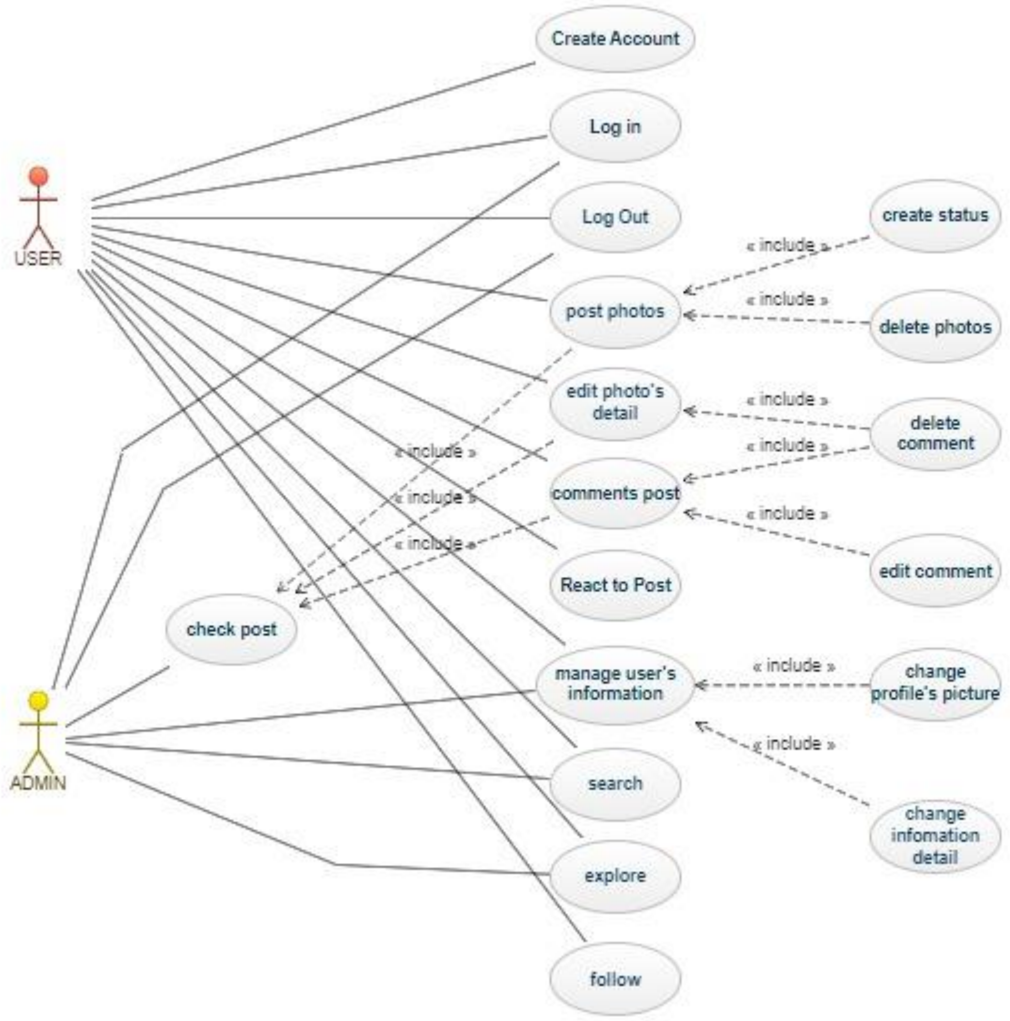
There are 2 actors: each one with different sets of privileges/tasks.

- **User:** Users are basically those who have signed up on the app. A regular user can login, update their profile, create post, comment/post and log out.

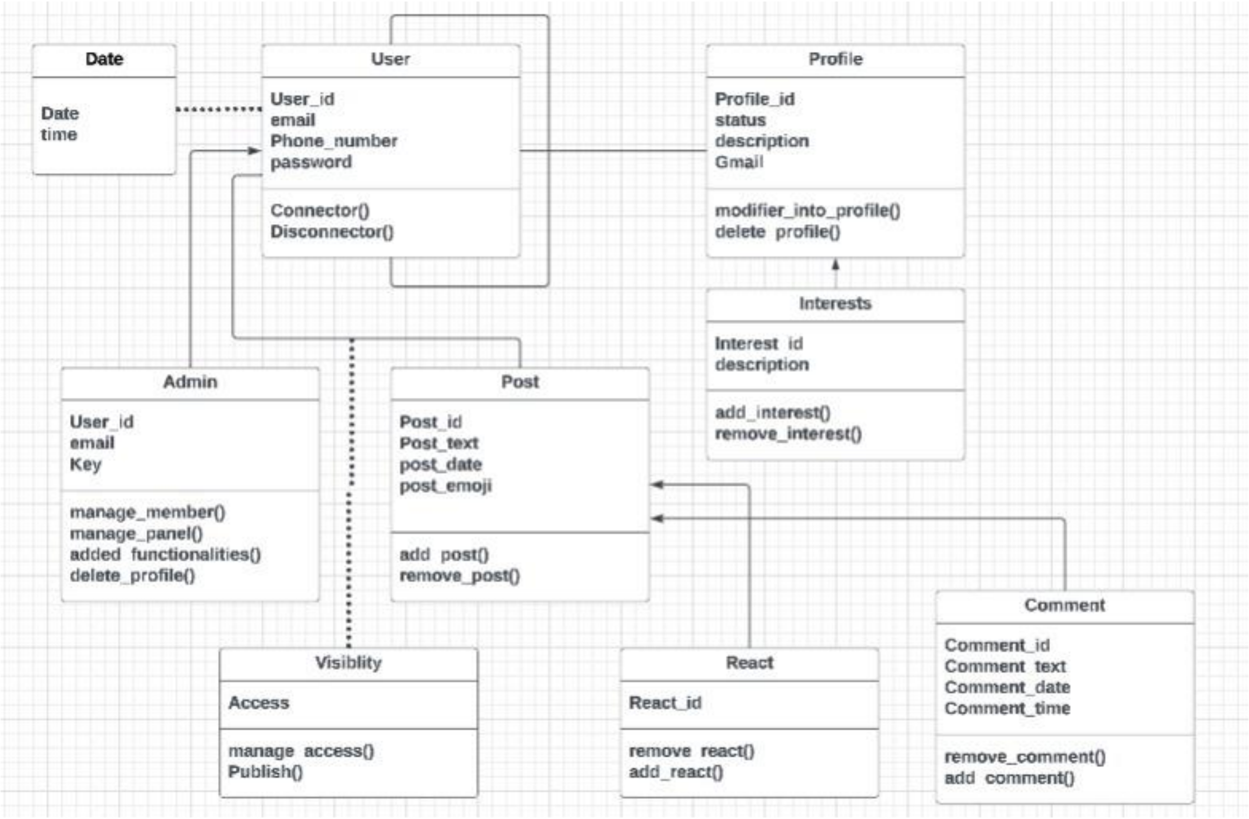
Use-case	Description
Login	User can login if they have already registered before.
CreateAccount/ Register	User can register using their HU ID.
Search users	User will be able to search other users on the app
Comment	A user can comment on any post.
Post a Photo	A user can post images
Follow users	User can follow different users on the app
React to a post	User can react to a post.
Manage user info	A user can manage his/her profile info
Log out	They can log out.

- **Admin:** Admins have some extra privileges in addition to all privileges a regular user has. An admin can delete user, comment, and post. This will ensure a peaceful and secure environment is created on the app.

Use-case	Description
Login	Admin can login if they have already registered before.
Register	Admin can register
Comment	An admin can manage comments on any post.
Manage pots	Admin can manage posts of other users.
Search/Explore	An admin can visit profiles of other users.



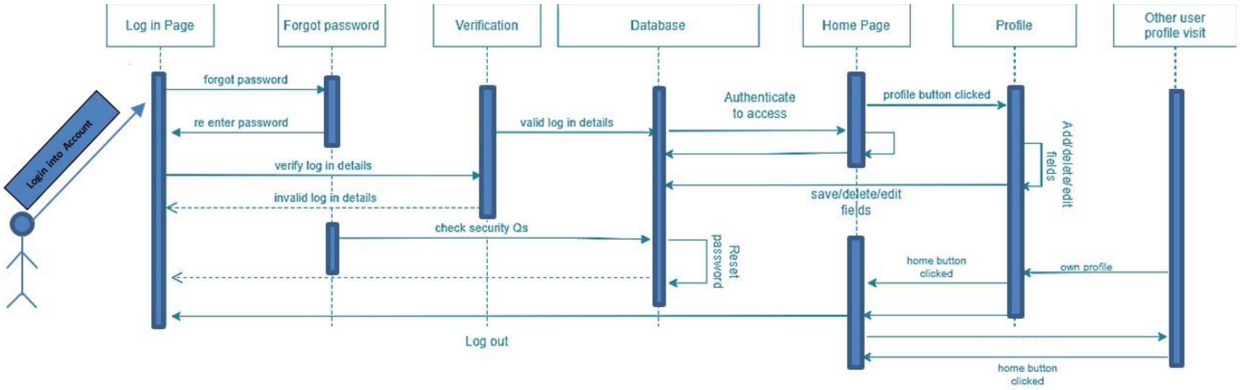
CLASS DIAGRAM:



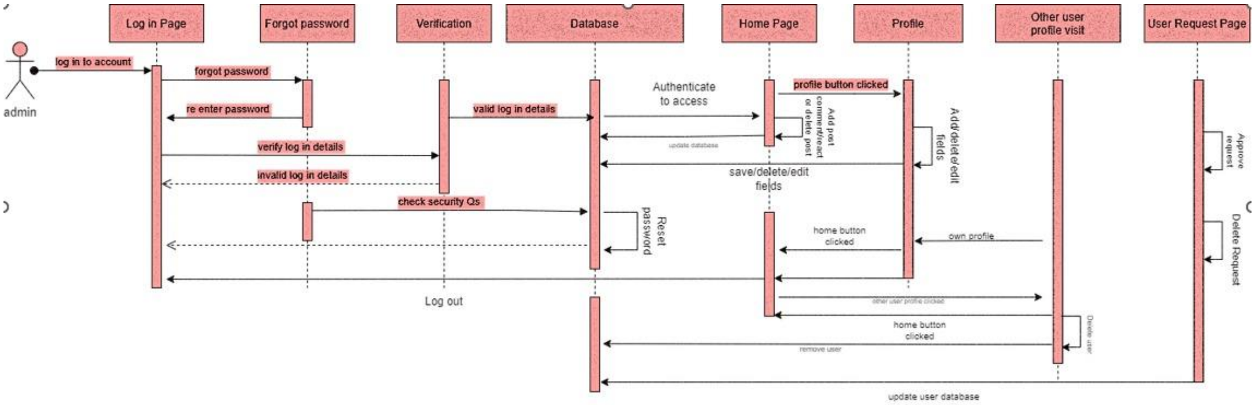
SEQUENCE DIAGRAM:

User and Admin have roughly the same sequence diagram apart from a few additional step(s), attached below.

User:


Log in Page
Forgot password
Verification
Database
Home Page
Profile
Other user profile visit
Login into Account
forgot password
re enter password
valid log in details
Authenticate to access
profile button clicked
verify log in details
invalid log in details
Add/delete/edit fields
save/delete/edit fields
check security Qs
Reset password
home button clicked
own profile
Log out
home button clicked

Admin:


Log in Page
Forgot password
Verification
Database
Home Page
Profile
Other user profile visit
User Request Page
admin
log in to account
forgot password
re enter password
valid log in details
Authenticate to access
profile button clicked
Add post comment/react or delete post
update database
Add/delete/edit fields
Approve request
verify log in details
save/delete/edit fields
invalid log in details
check security Qs
Reset password
Delete Request
home button clicked
own profile
Log out
other user profile clicked
home button clicked
Delete user
remove user
update user database