COMSATS University Islamabad,
Abbottabad Campus

SOFTWARE DESIGN DESCRIPTION

(SDD DOCUMENT)

for

Interview Preparation Application

Version 1.0

By

Hozefa Hassan Rizvi	**CIIT/FA20-BSE-019/ATD**
Muhammad Hammad	**CIIT/ FA20-BSE-031/ATD**
Ashar Ali	**CIIT/ FA20-BSE-158/ATD**

Supervisor

Mam Sana Malik

Bachelor of Science in Computer Science (2020-2024)

Table of Contents

Revision History **3**
1. Introduction **5**
2. Design Methodology and software process model **5**
3. System Overview **5**
3.1 Architectural Design 6
3.2 Process Flow/Representation 9
4. Design Models [along with descriptions] **10**
5. Data Design **17**
6. Algorithm & Implementation **17**
7. Software Requirements Traceability Matrix **17**
8. Human Interface Design **18**

Revision History

Name	Date	Reason for changes	Version

Application Evaluation History

Comments (by committee) ***include the ones given at scope time both in doc and presentation**	**Action Taken**

Supervised by

Mam Sana Malik

Signature_____

Introduction

The project has focused on developing an **"Interview Preparation Application,"** which aims to assist software engineering graduates in excelling in their job interviews. The project encompasses various modules, including **user authentication and profile setup**, a comprehensive **question bank**, **community engagement features**, **expert interaction capabilities**, **mock interview**, and **administrative management functionalities**. These modules work cohesively to create an integrated platform that aids in interview preparation and skill enhancement.

Design methodology and software process model

Design methodology:

For the design methodology, the project has adopted a procedural approach to accommodate the specific requirements of the chosen technology stack. React Native, which utilizes JavaScript, for the most part operates with a procedural design approach, which emphasizes step-by-step execution and the use of functions for streamlined development. Although React Native incorporates certain object-oriented concepts, the project primarily aligns with a procedural design to ensure compatibility and optimal performance within the chosen framework.

Software Process Model:

Regarding the software process model, the project has followed the practices of the Agile methodology. Agile has been selected to facilitate iterative development, enhance flexibility, and allow for continuous user feedback. Given the dynamic nature of the project requirements and the necessity for adaptability in the development process, Agile stands out as the most suitable software process model. The emphasis on collaboration, adaptability, and user-centered development inherent in the Agile methodology has contributed significantly to the project's progress and success.

System overview

Purpose

The **Interview Preparation Application** aims to provide comprehensive support to software engineering graduates by facilitating effective interview preparation and skill enhancement.

Functionality:

- **Question Bank:** The application houses an extensive database of industry-specific interview questions and answers, allowing users to prepare for a diverse range of potential interview scenarios.
- **Expert Interaction:** Users can connect with experts who have given interview or have some knowledge about the interview and how to appear in an interview to seek guidance.
- **Mock Interview Analysis:** The application offers a sophisticated mock interview feature that provides real-time feedback on users' non-verbal communication, practical skills, and confidence levels, helping them identify areas for improvement.
- **Community Engagement:** Encouraging a collaborative environment, the application facilitates active participation within the software engineering community, fostering discussions, and the sharing of valuable interview experiences and tips among each other.
- **User Authentication and Profile Setup:** Users can easily create and manage their profiles, providing essential information related to their technical interests, academic background, and skillset, optimizing their interaction with the application's features.

Context:

Designed within the context of the fast-paced and competitive software engineering industry, the application emphasizes the importance of not only technical proficiency but also effective communication and practical skills during interviews.

Design:

With a user-centric and intuitive design, the application ensures a seamless and engaging experience for users, encouraging active participation and collaboration within the software engineering community.

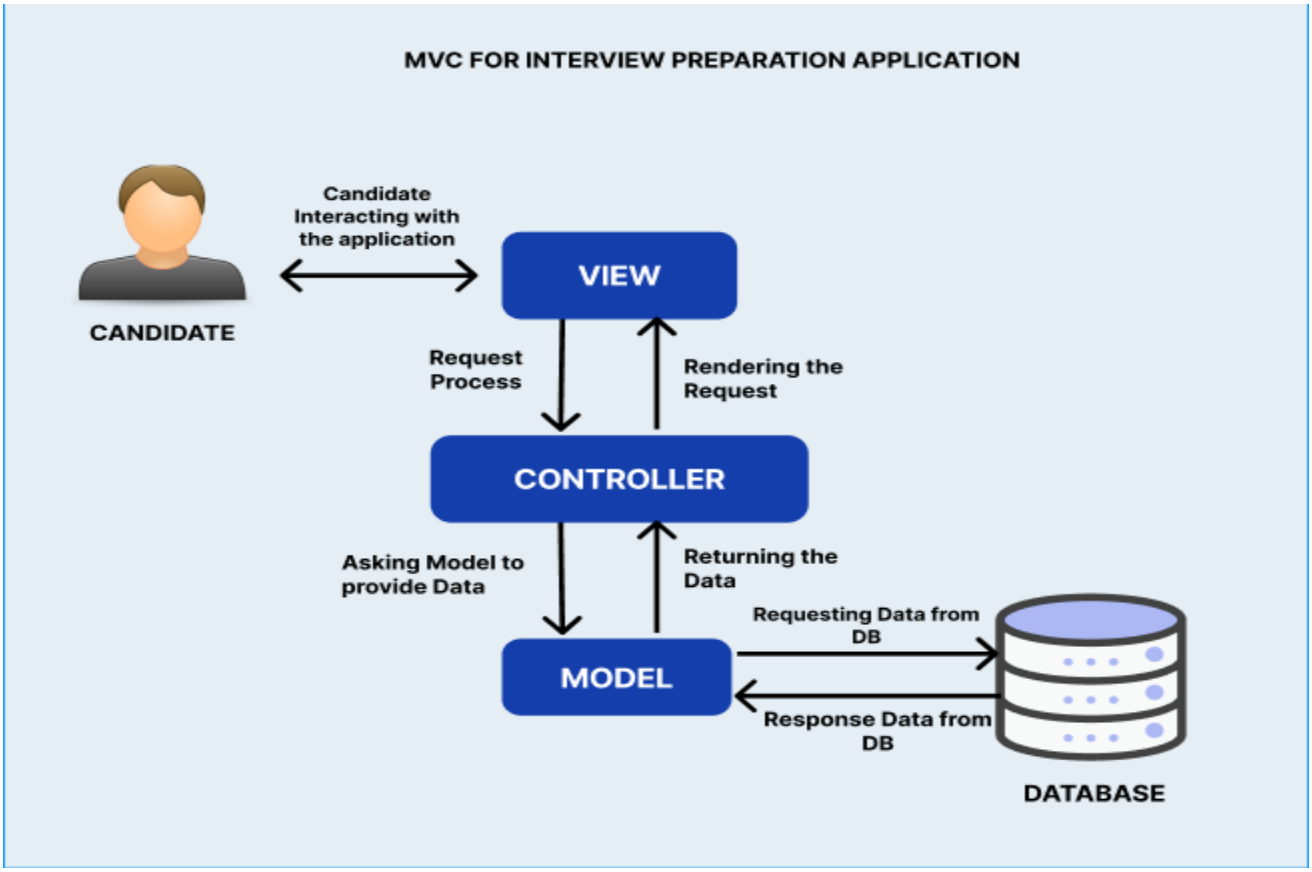
Background

The development of the Interview Preparation Application is driven by the need to bridge the gap between theoretical knowledge and the practical skills required for successful career placement in the software engineering domain.

Architectural design

For the Interview Preparation Application, a suitable architecture would be the Model-View-Controller (MVC) architecture. This choice is primarily based on the need for a structured and organized approach to handle the application's complex functionalities. MVC provides clear separation between the application's data, user interface, and business logic, promoting better code maintainability, scalability, and reusability.

Diagram:



From the above diagram:

Model:

The model represents the data and business logic of the application. In this case, the model would include things like:

- User profiles
- Interview questions and answers
- Expert profiles
- Mock interview results

View:

The view is the user interface of the application. It is responsible for displaying the data from the model to the user and taking input from the user. In this case, the view would include things like:

- Home page with a list of subfields related to software engineering.

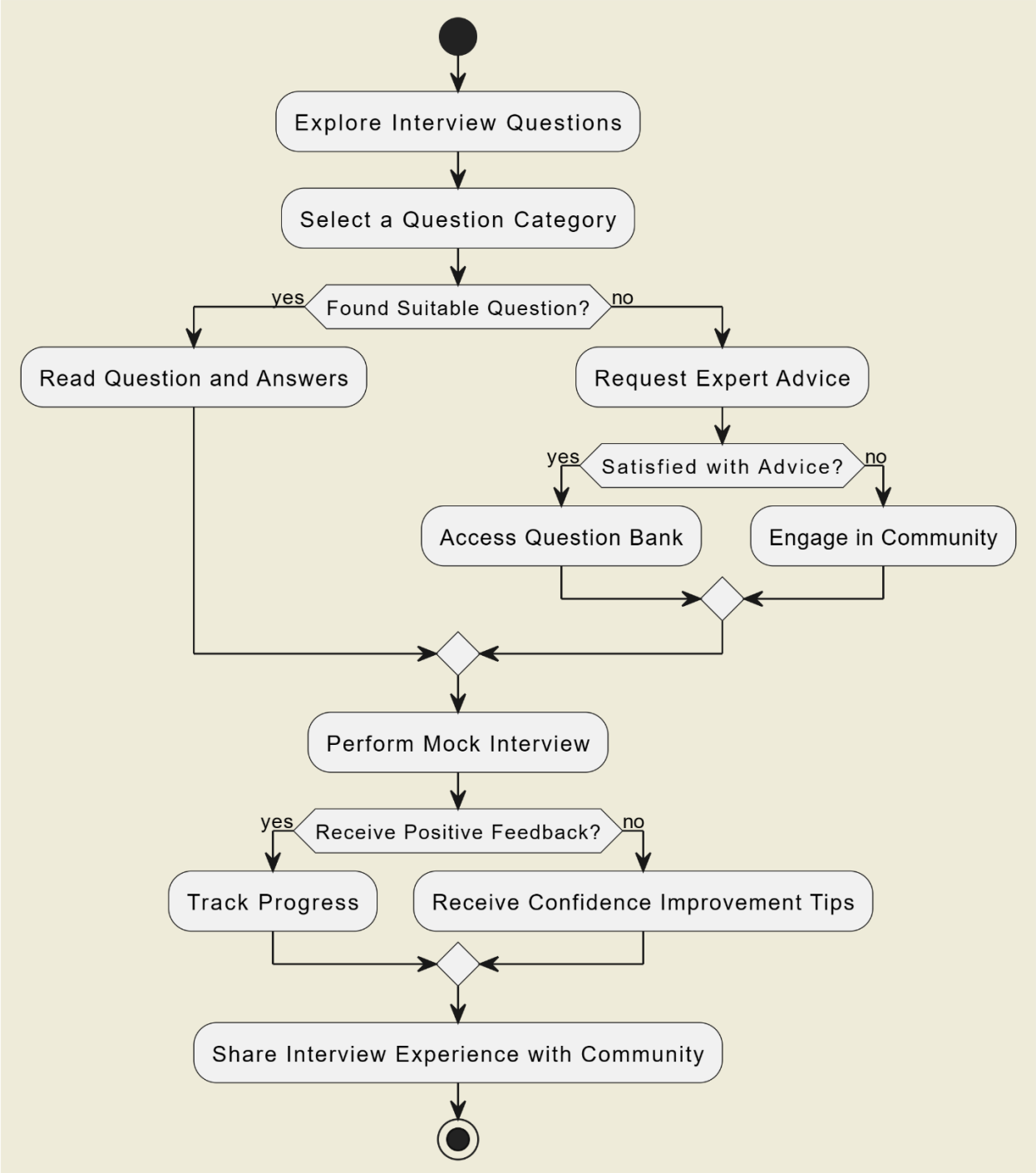
- Question Bank page with a list of interview questions
- Expert Panel page with a list of experts
- Mock Interview page with an interface for the user to practice answering interview questions.
- Community engagement where users can ask questions and share insights.

Controller:

The controller acts as an intermediary between the model and the view. It is responsible for handling user requests, updating the model, and returning the updated view to the user. In this case, the controller would handle things like:

- Retrieving a list of interview questions for a given subfield
- Saving the user's answers to a mock interview
- Forwarding a question to the Expert Panel
- Posting a message to the discussion forum

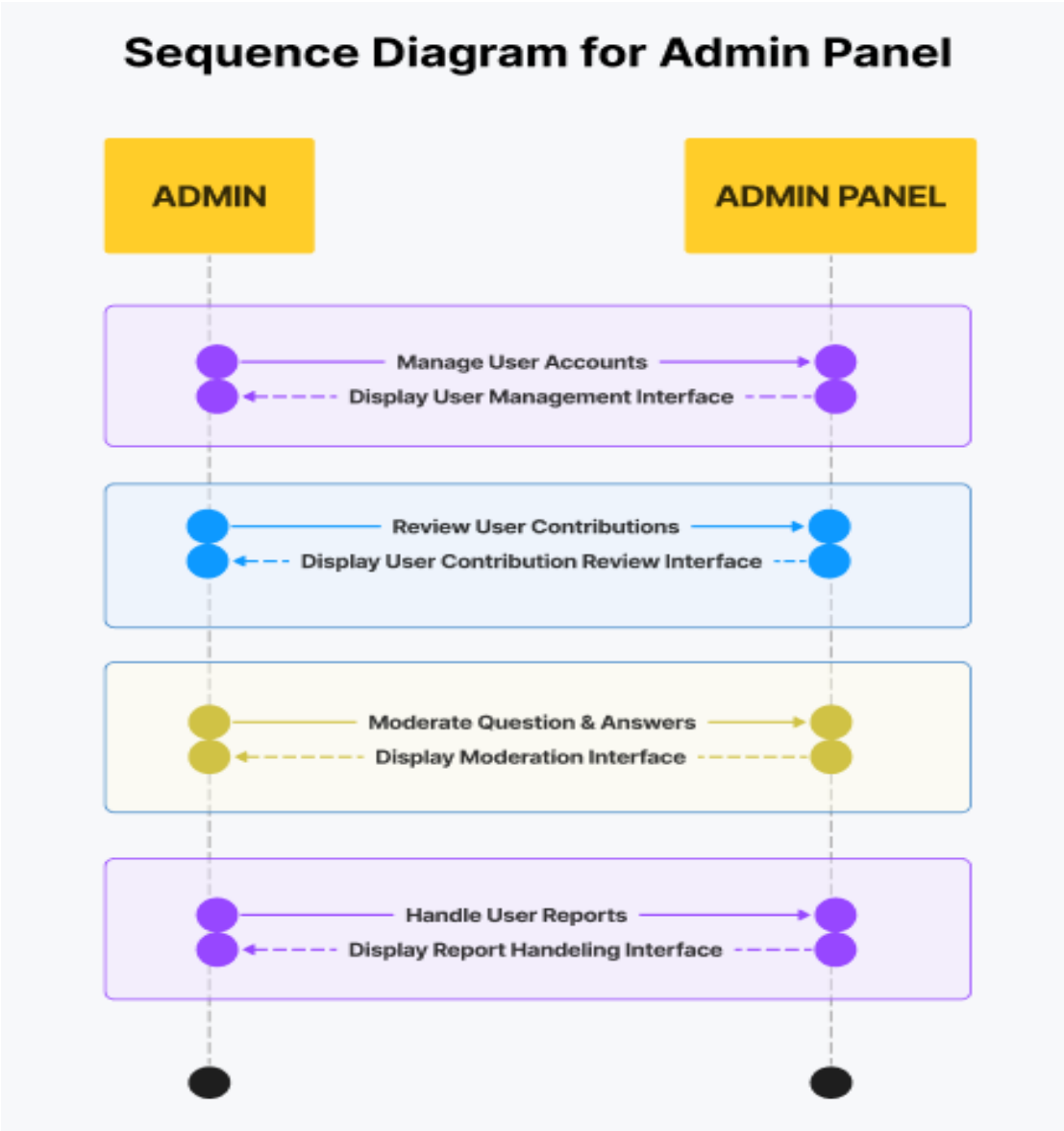
Process flow/Representation



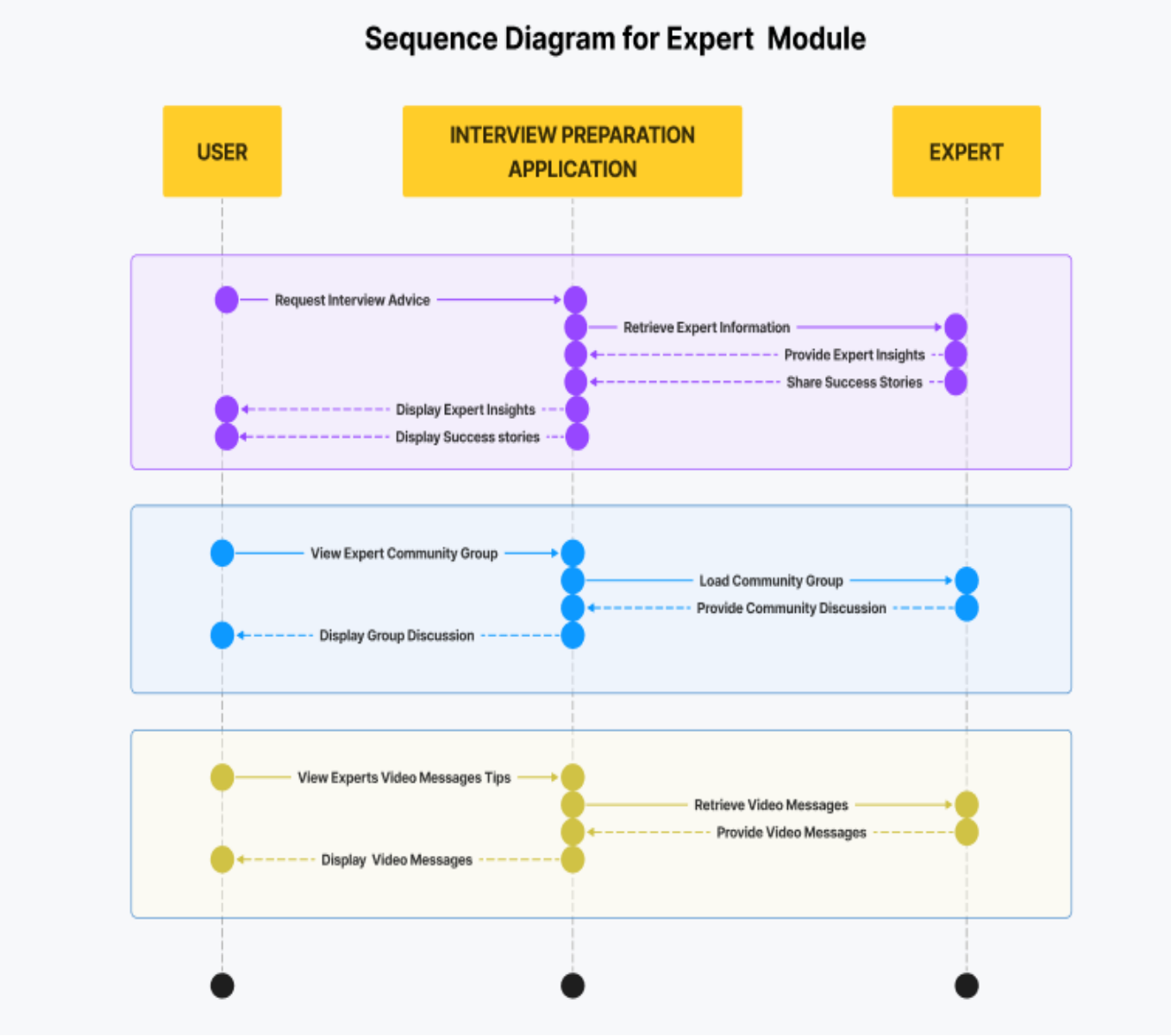
Design models:

Sequence Diagrams:

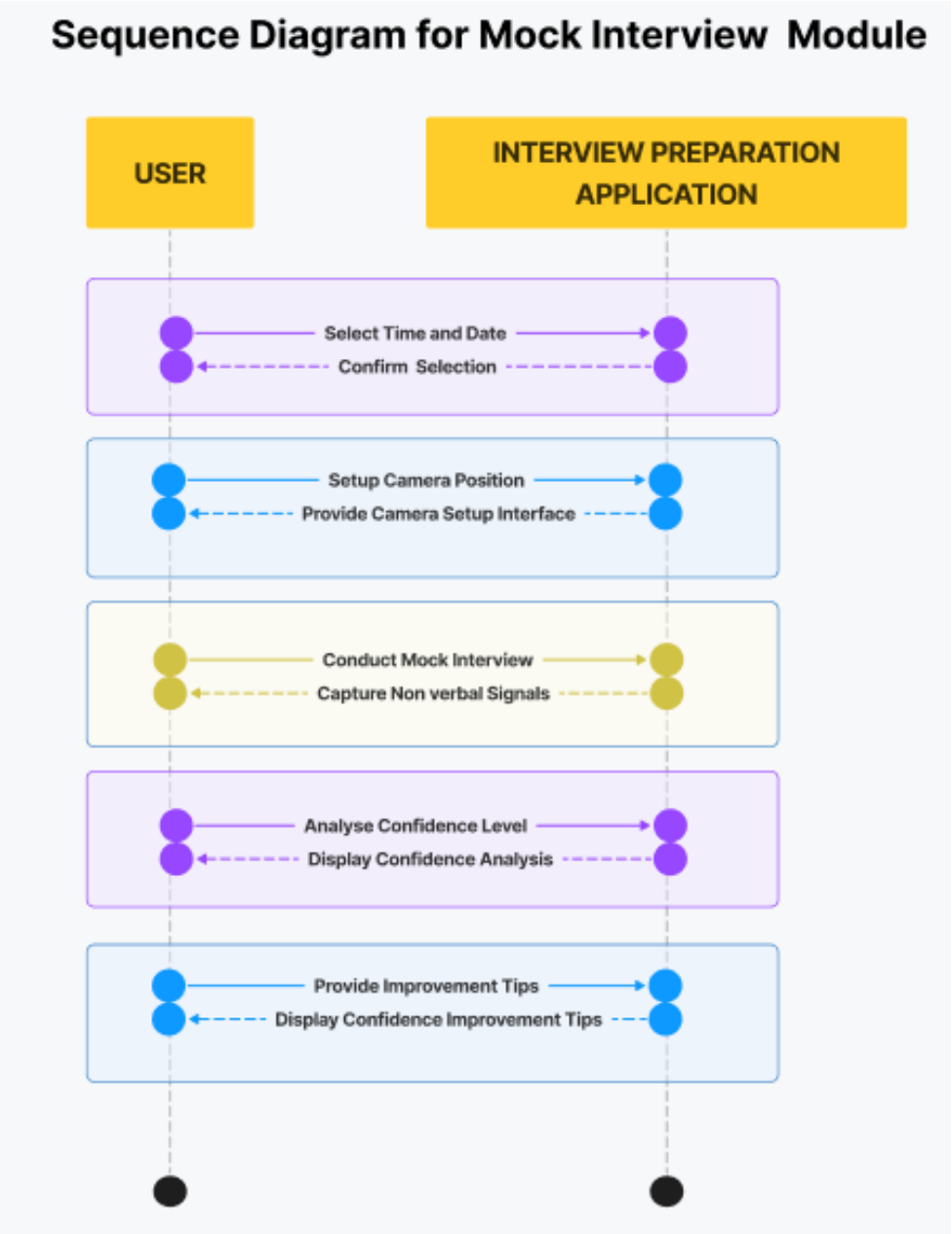
1. **Sequence Diagram (Admin Module):**



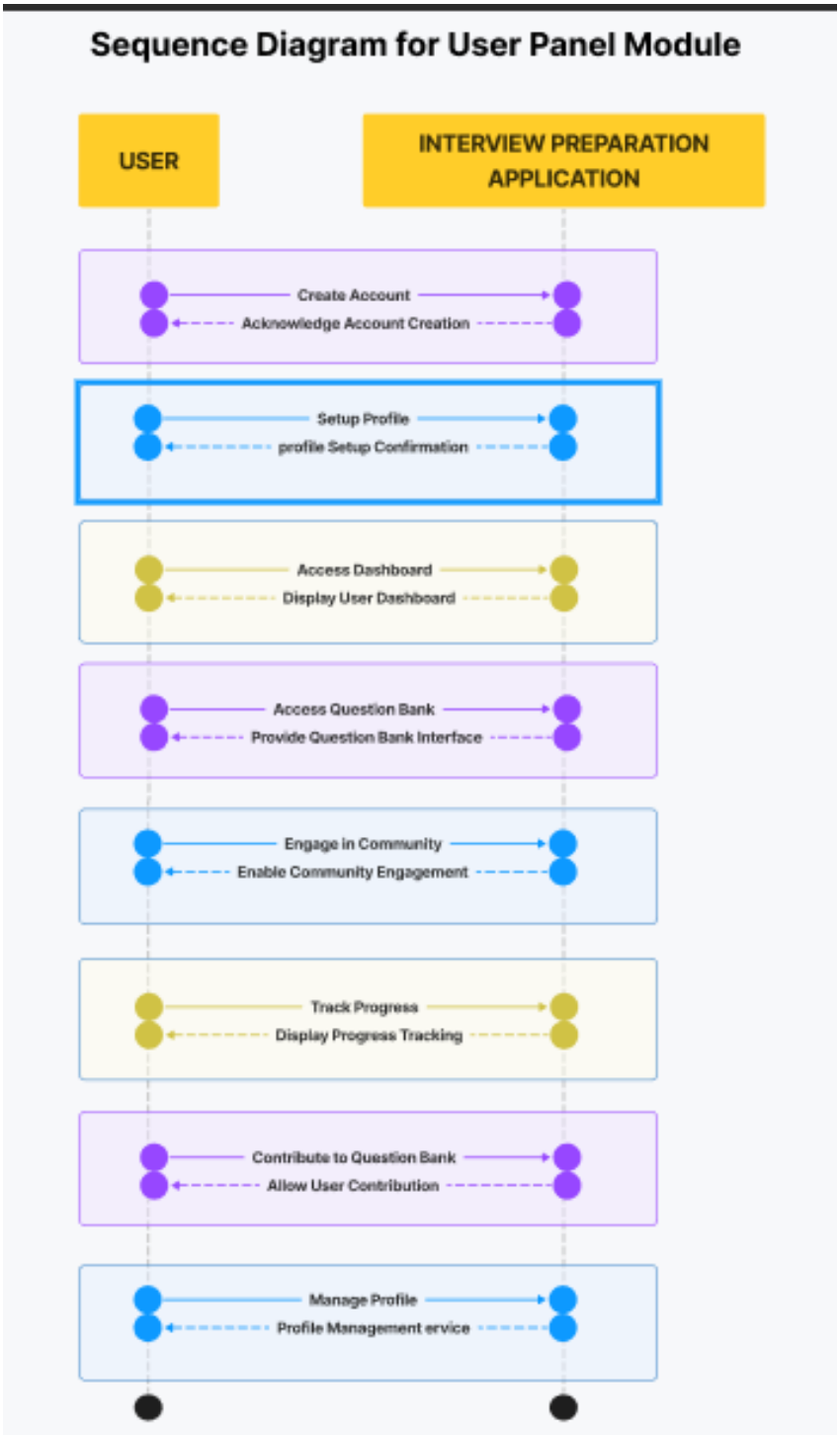
2. Sequence Diagram (Expert Module):



3. Sequence Diagram (Mock Interview Module):

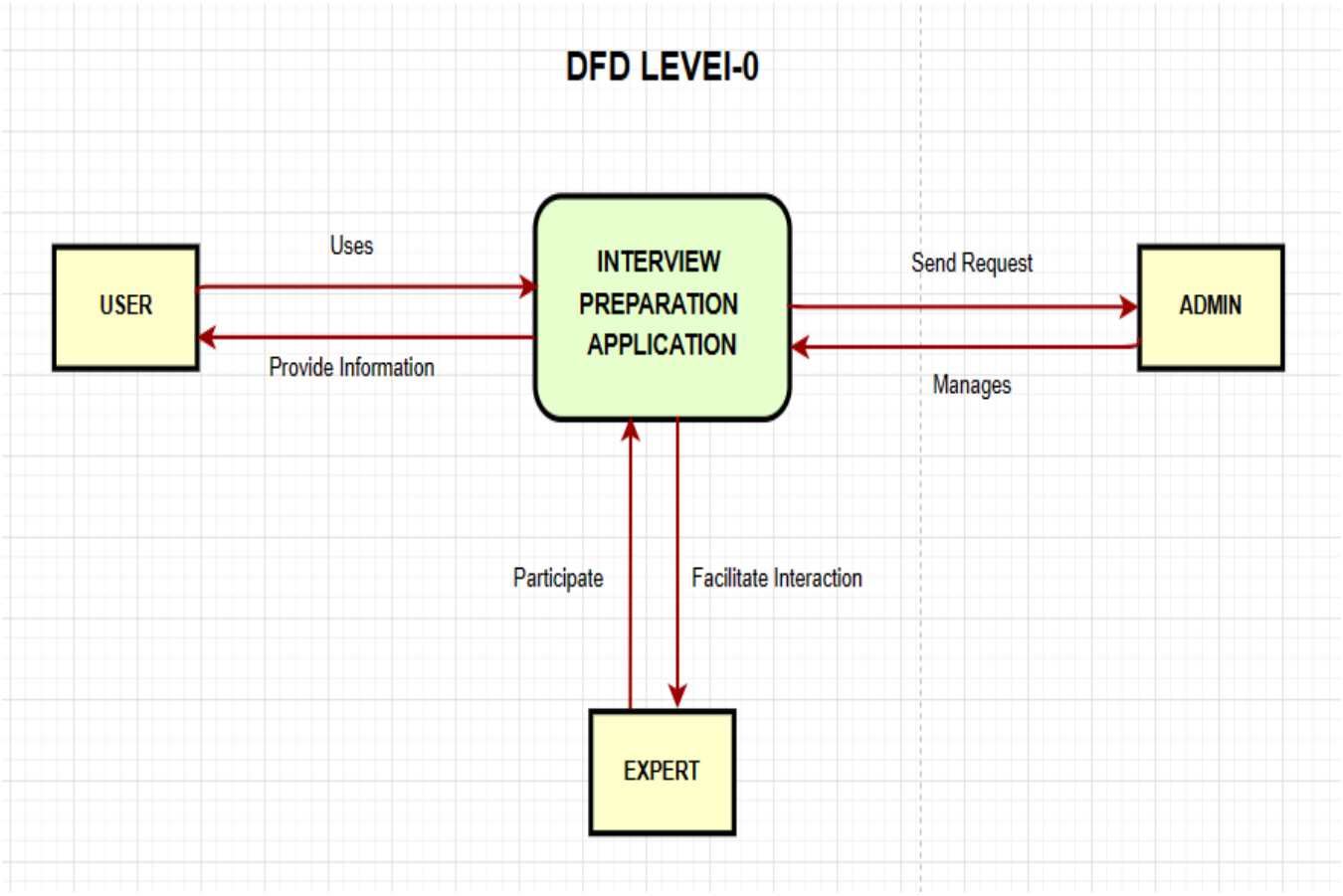


4. **Sequence Diagram (User Module):**

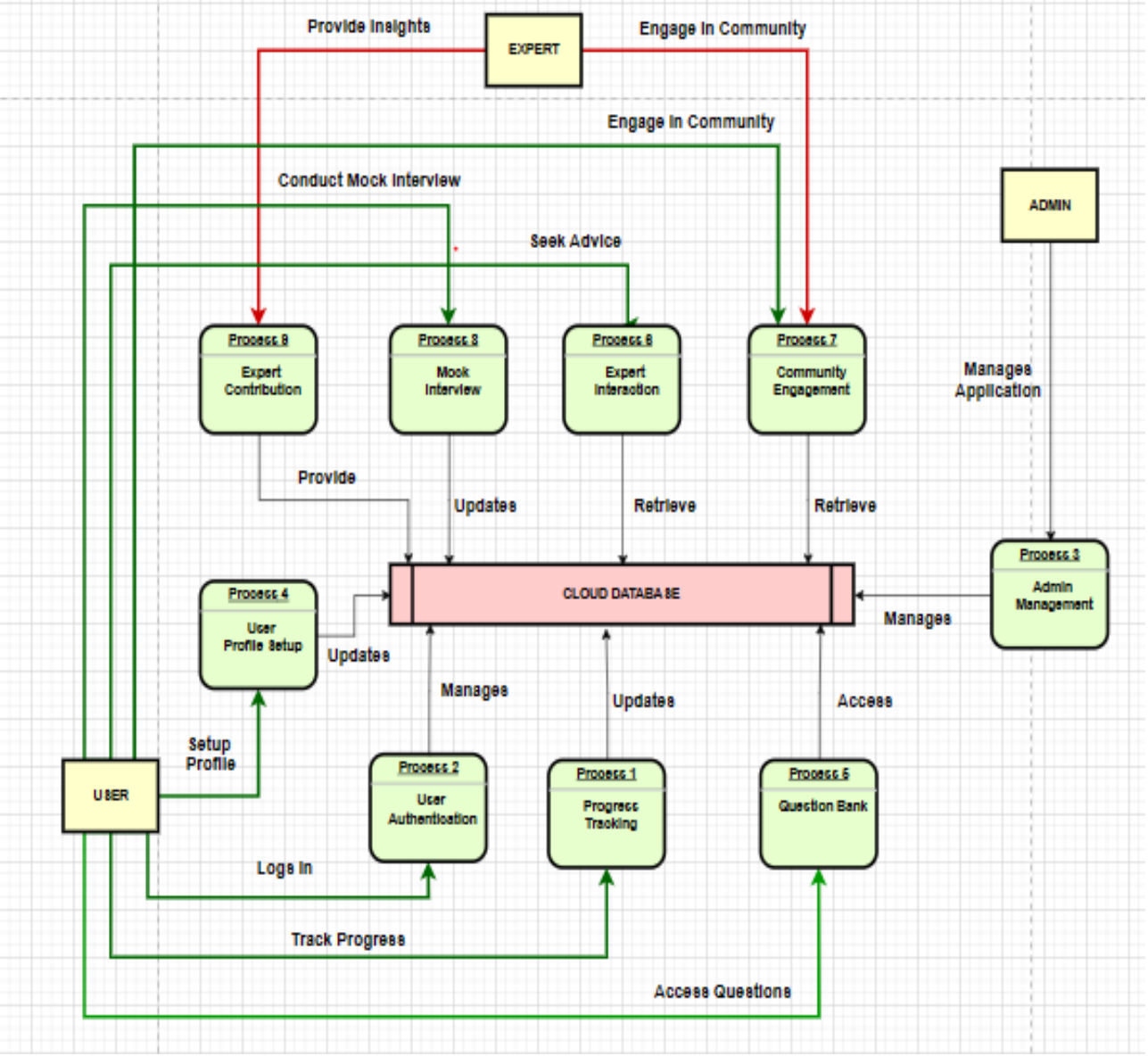


Data Flow Diagram

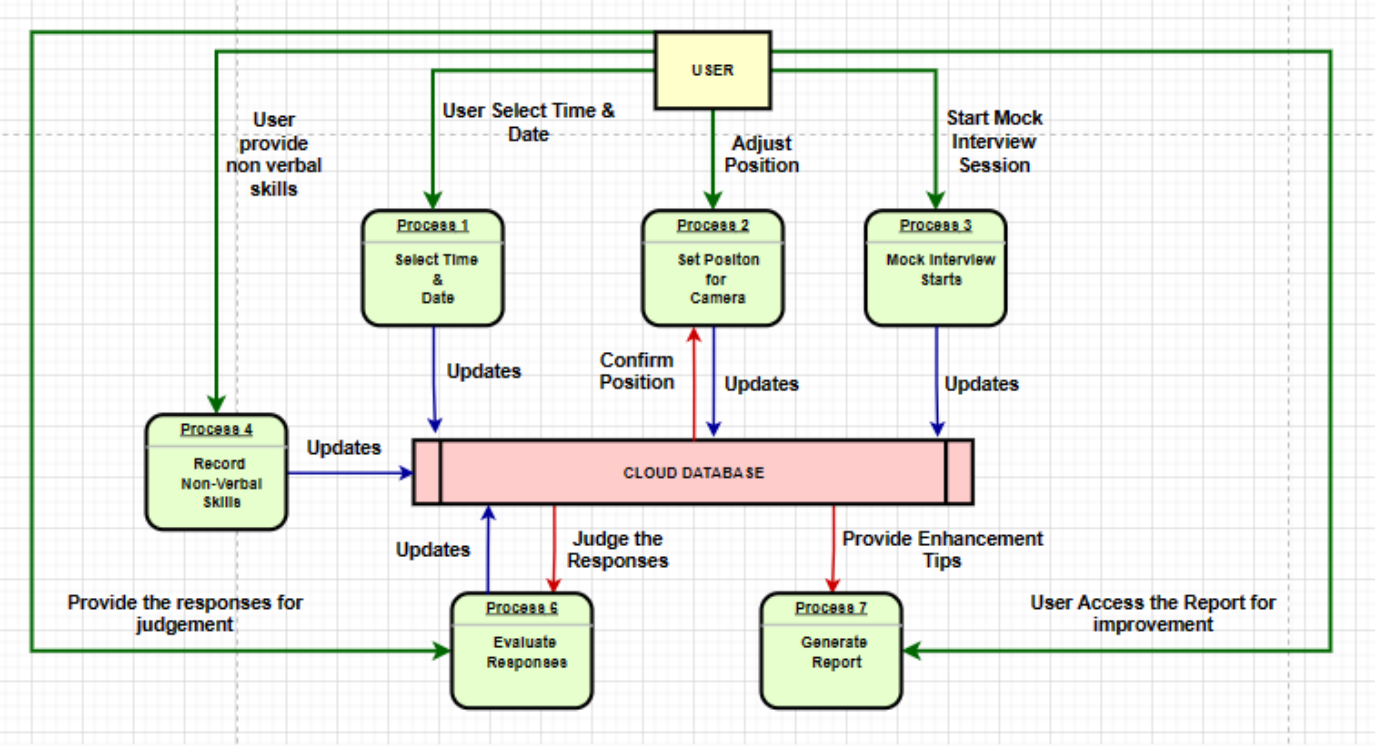
DFD Level-0:


DFD LEVEI-0
USER
Uses
Provide Information
INTERVIEW
PREPARATION
APPLICATION
Send Request
Manages
ADMIN
Participate
Facilitate Interaction
EXPERT

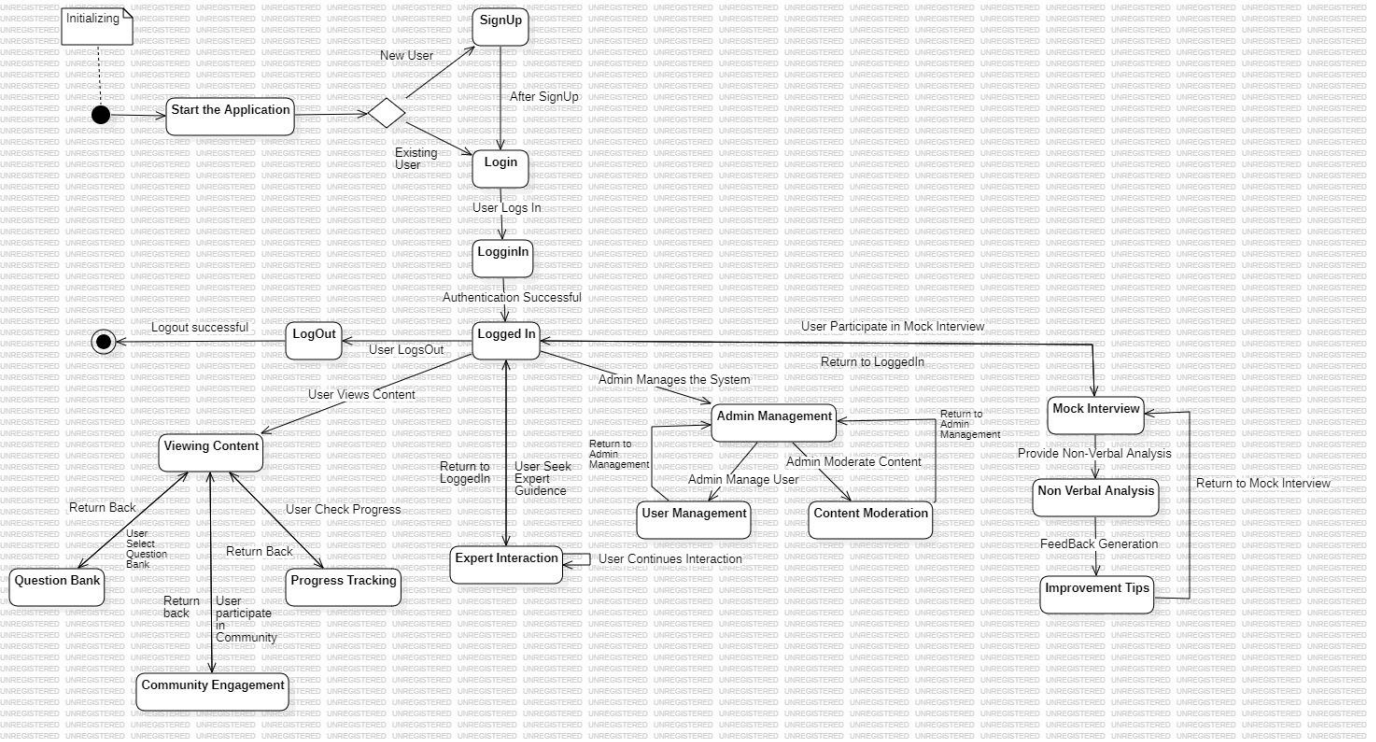
DFD Level-1:



DFD Level-2(Mock Interview Process):



State Transition Diagram:



Data design

We're using the Firebase Database management system by google. At the run time the data will be retrieved from the cloud database

Algorithm & Implementation

- **Mock Interview:** Prompt users to select a time slot, analyze real-time video data for non-verbal cues and confidence levels, and provide personalized improvement tips based on the analysis results.
- **Expert Panel:** Facilitate real-time communication, manage forum discussions, and deliver personalized expert advice and success stories based on user preferences and interests.
- **Admin Panel:** Manage users, moderate content, and handle reports using data validation.

Software requirements traceability matrix

Req. Number	Ref. Item	Design Component	Component Items
FR01	DFD	Level-0	Overall system data flow
FR02	DFD	Level-1	Detailed process data flow
FR03	DFD	Level-2	Detailed data flow for Mock Interview Process

Human interface design

User Panel Module:

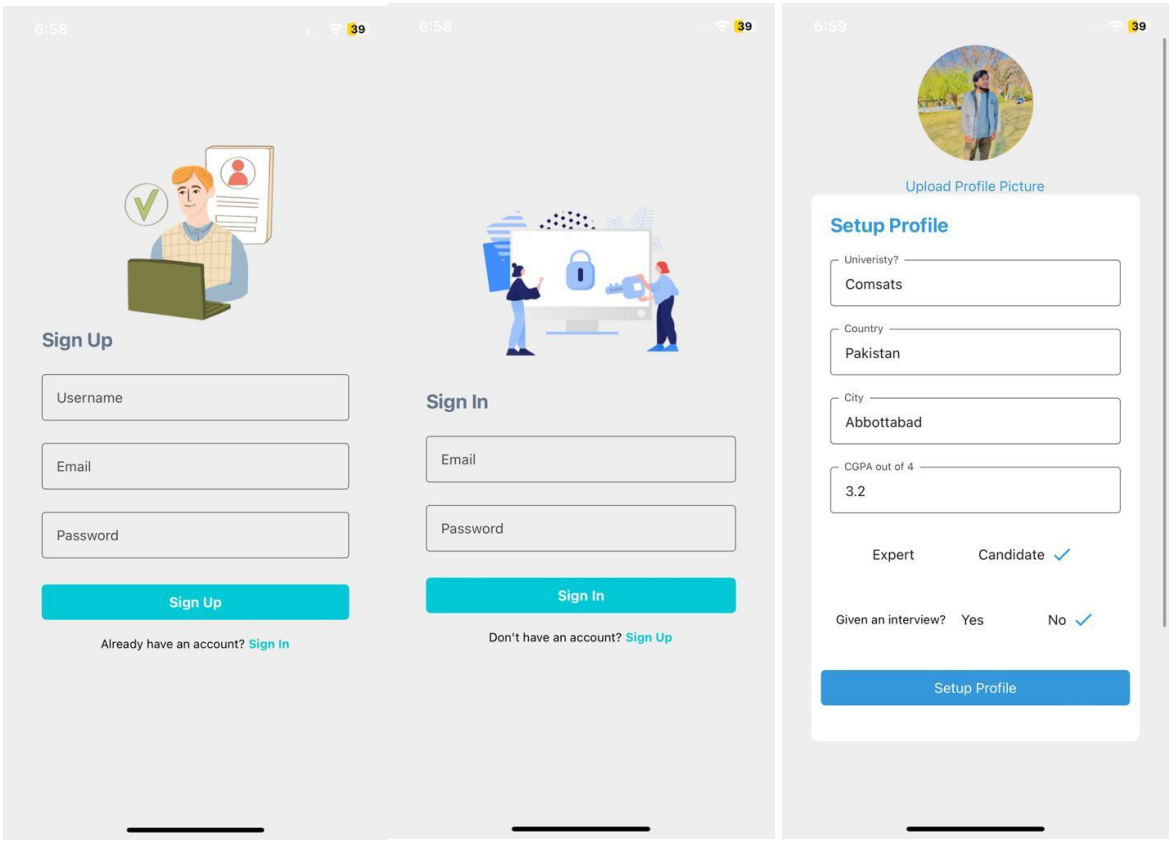
Splash Screen Page:

39

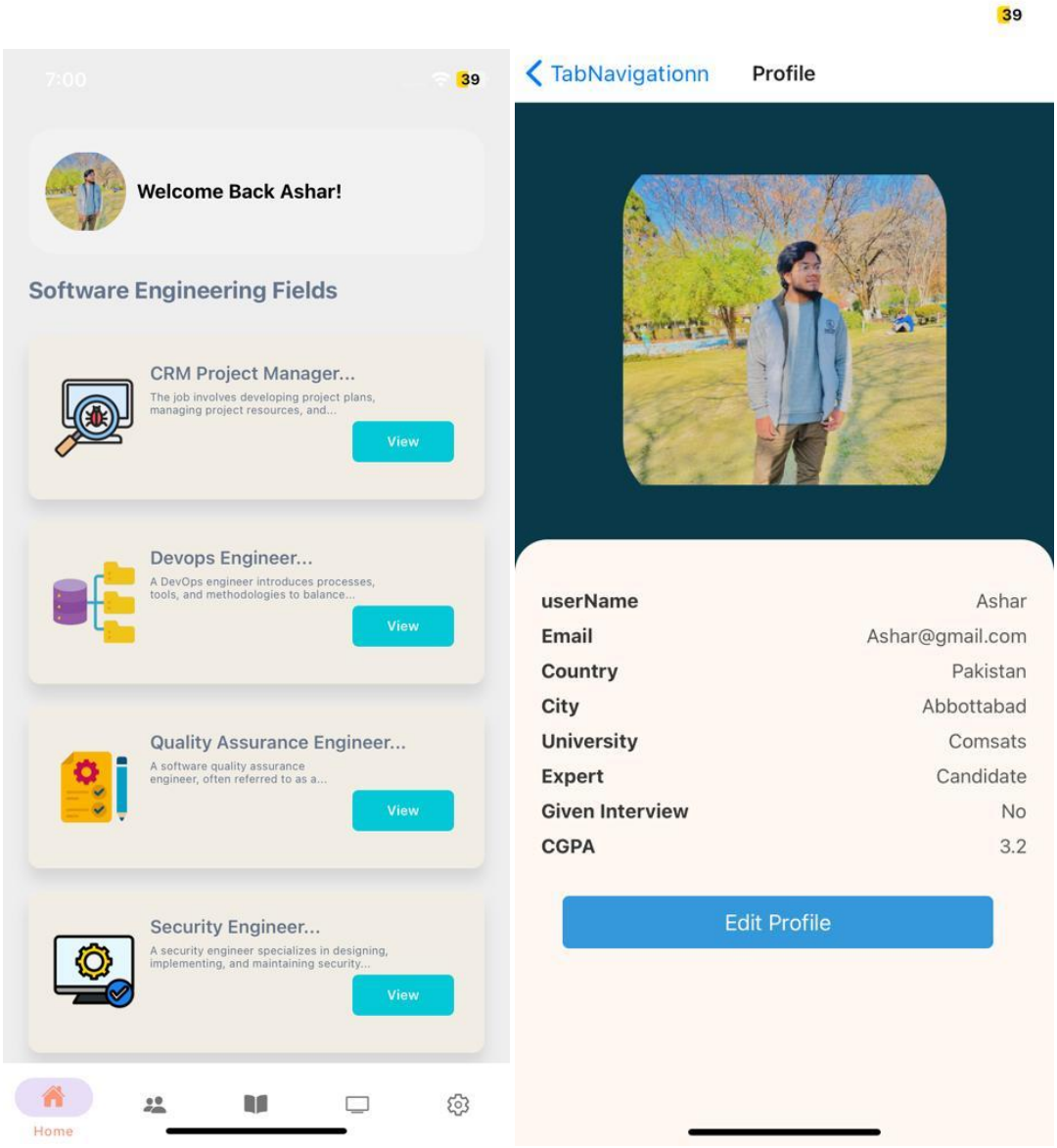
Interview Prep App

(For Software Engineering Graduates)

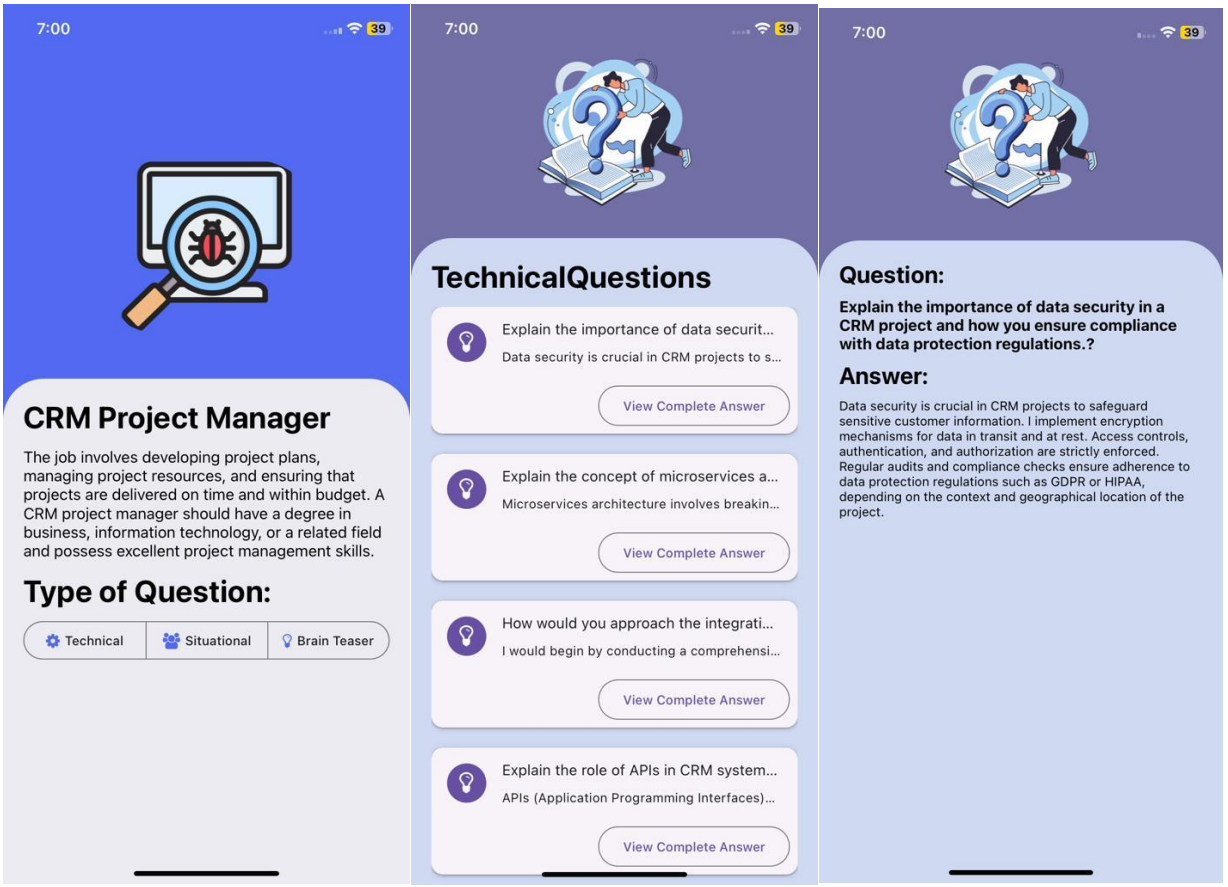
Signup & Sign in Page & Setup Profile Screen:



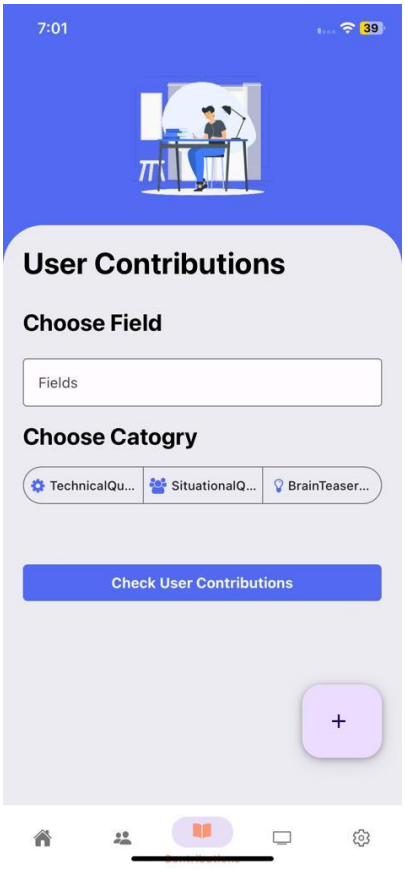
Home Screen & Profile Screen:

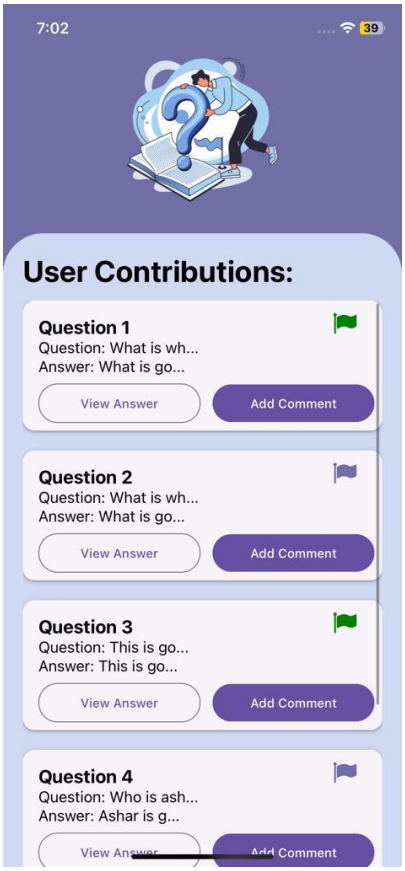


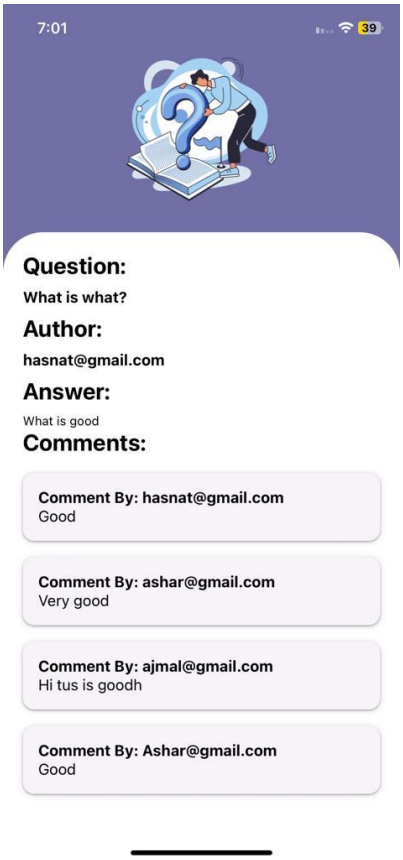
Field Screens:

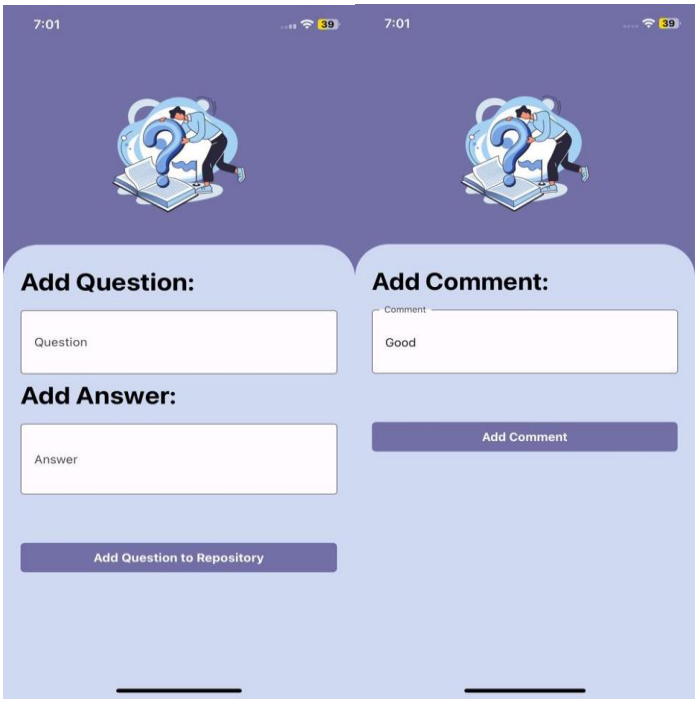


User Contributions Screen:

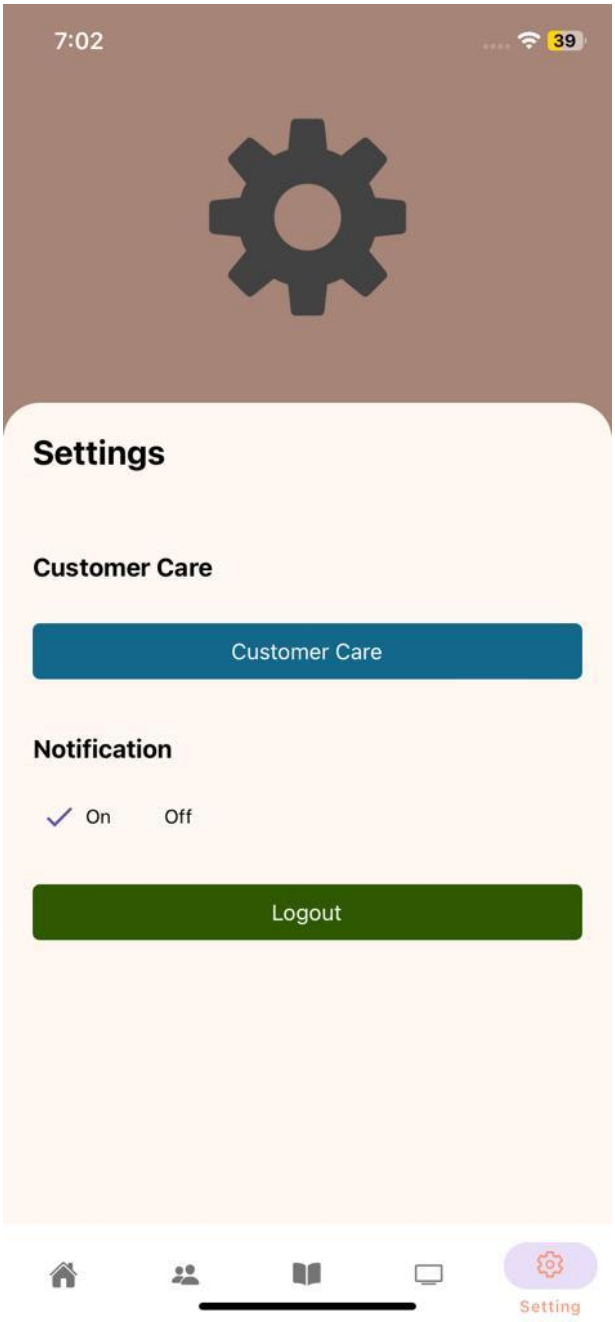




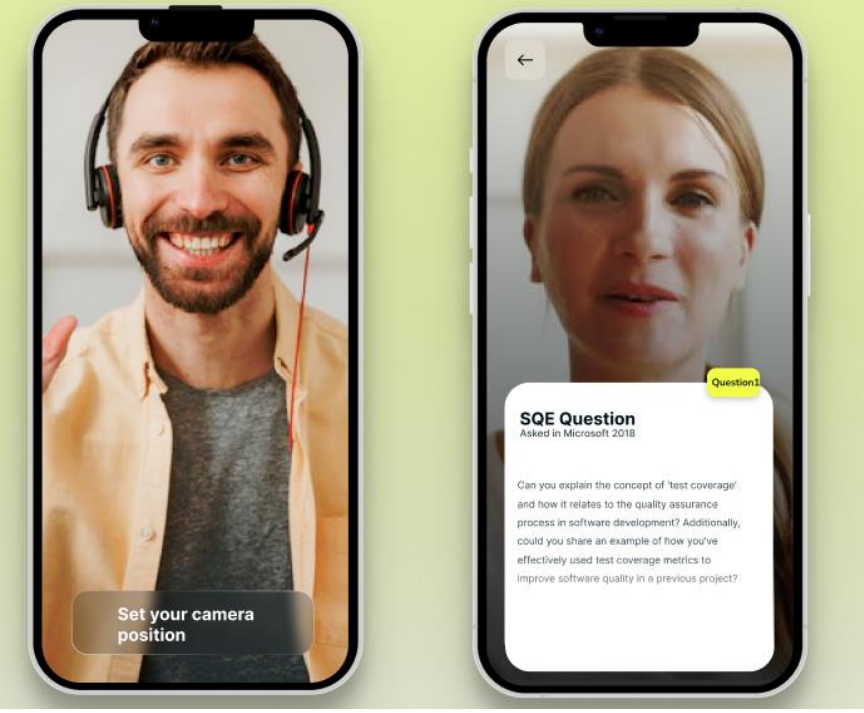
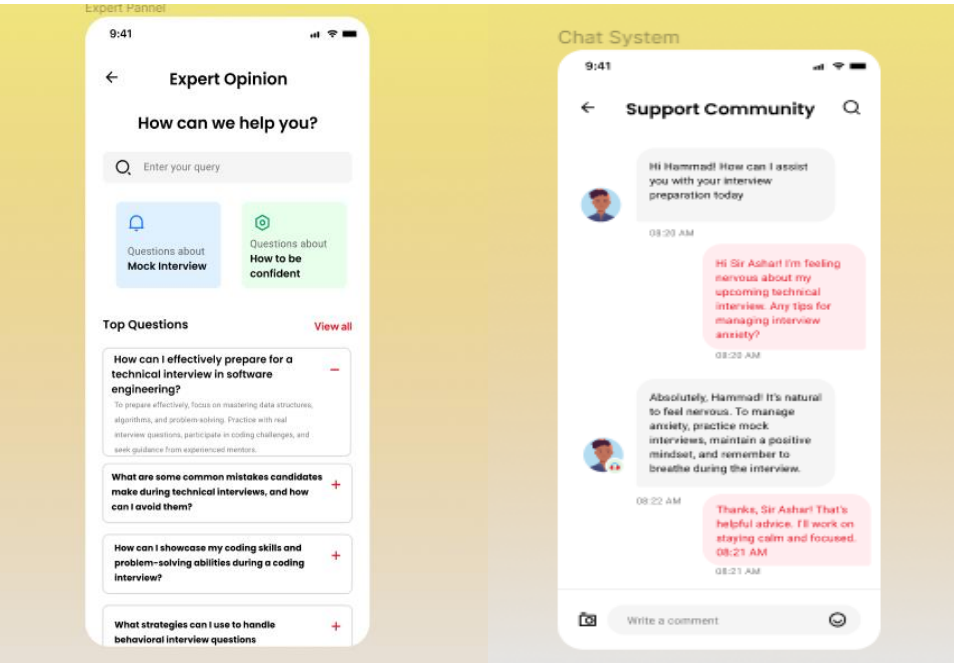




Setting Screen:



Mock Interview Module:



Expert Panel & Chat System: