MEF University
Department of Computer Engineering
COMP 302 Software Engineering
Report #2

Group Members

Zeynep Yılmaz 042101088

Onat Sarıbıyık 042101097

Ömer Faruk Tüccar 042001032

Sude Göktaş 042101127

Zehra Mert 042201058

1. Introduction // ONAT
    - 1.1 Goals and objectives
    - 1.2 Statement of scope
    - 1.3 Software context
    - 1.4 Major constraints
2. Usage scenario Ömer
    - 2.1 User profiles
    - 2.2 Use-cases
        - 2.2.1 Use-Case Diagram
        - 2.2.2 Use-Case Descriptions
3. 4+1 View
    - 3.1 Logical View Zehra
        - Class Diagram
        - State Machine
        - Communication Diagram
    - 3.2 Process View Zeynep
        - Activity and Sequence Diagrams
    - 3.3 Development View Zeynep
        - Package Diagram or Component Diagram
    - 3.4 Physical View Sude
        - Architecture pattern & architectural design (Layered structure)
4. Restrictions, Limitations, and Constraints
    - All non-functional requirements
5. User Manual
6. Source codes

Report 2

- ✧ The remaining UML diagrams
    - Sequence
    - State machine
    - Package
    - Deployment
- ✧ Architecture Pattern

Report 2

- State Machine Diagrams for each class in class diagram.
- Sequence Diagrams (one diagram for each US)
- 1 Package diagram
- 1 Deployment diagram
- Architectural Design of your system.

1. Introduction

1.1 Goals and objectives

The primary goal of this project is to design and develop a professional networking platform inspired by LinkedIn, enabling users to connect, share content, and explore career opportunities. The objectives include building an intuitive interface, implementing reliable backend services, and delivering core features such as user authentication, profile management, connection building, content sharing, job listings, and admin moderation. The platform is aimed at enhancing professional interaction in a modern, secure, and user-friendly environment.

1.2 Statement of scope

This project covers the development of a web-based system that supports both individual users and corporate employers. It allows users to create accounts, manage professional profiles, engage in networking, post and interact with content, and apply for jobs. Employers can list job opportunities and manage applications. Additionally, system administrators have access to moderation tools. The platform will support essential functionalities while ensuring performance, scalability, and maintainability using **Python** with **FastAPI** connection, **SQLite**, and **HTML/CSS.**

1.3 Software context

The system will be a standalone web application with a backend developed in Python using FastAPI, and a frontend built with basic HTML and CSS for compatibility and ease of use. The SQLite database will serve as the data storage layer. The application will be run locally during development, and future deployment may involve cloud hosting solutions. External libraries may be integrated for user interface components, authentication, and database management.

1.4 Major constraints ??

- The project will use only open-source technologies to minimize cost.
- The frontend design is limited to basic HTML/CSS to keep it lightweight and accessible.
- The development must be completed within the academic semester timeline.
- The database is limited to SQLite, which may restrict concurrent user handling in a production environment.

2. Usage scenario

2.1 User profiles

2.2 Use-cases

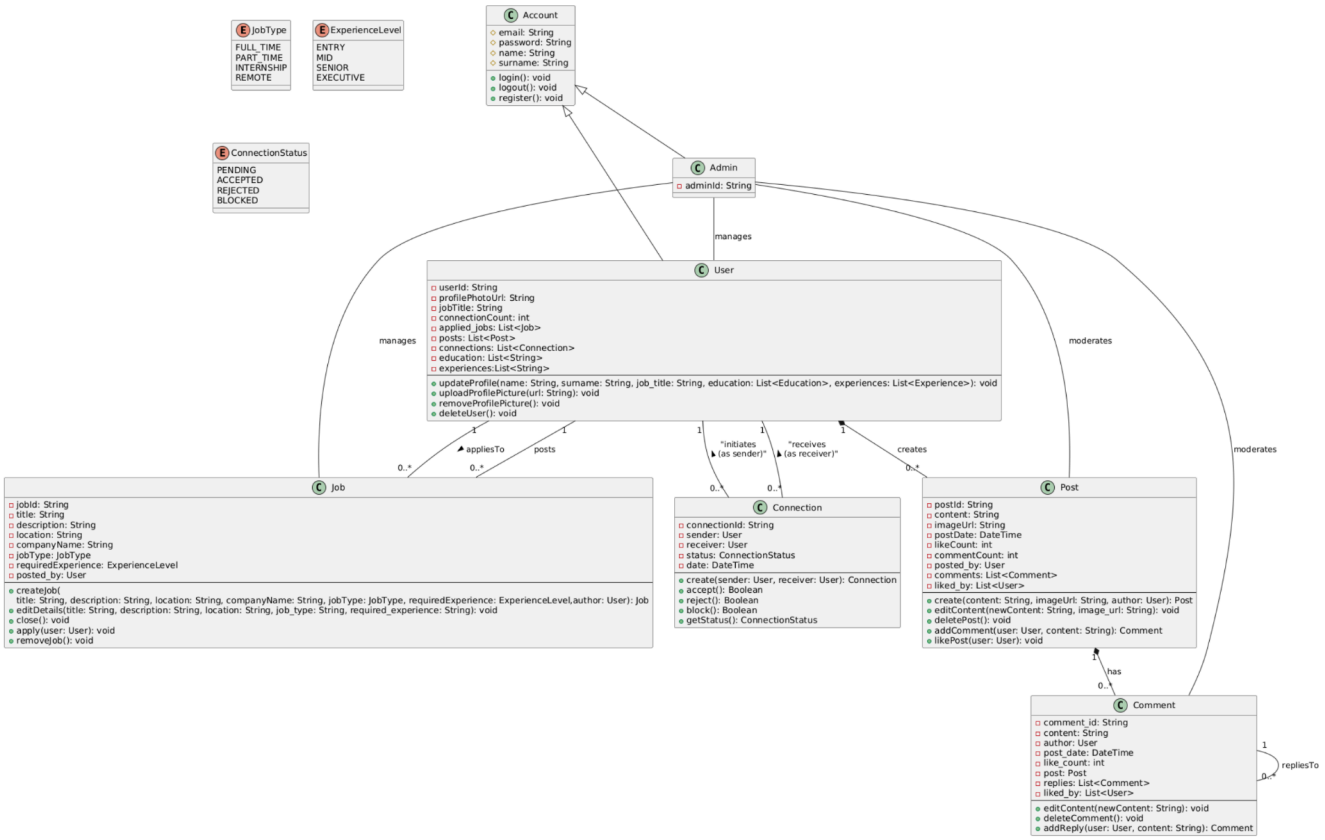
2.2.1 Use-Case Diagram

2.2.2 Use-Case Descriptions

3. 4+1 View

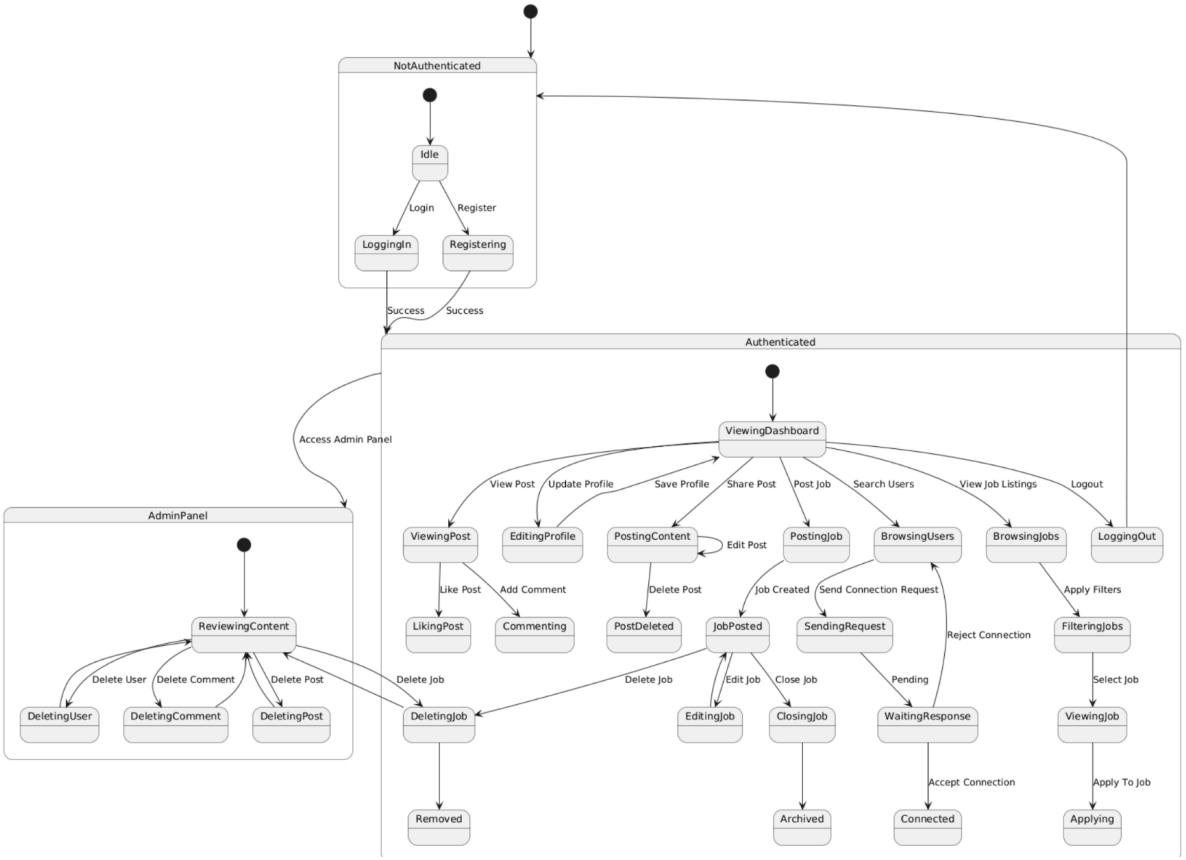
3.1 Logical View

Class Diagram



3.1.1 Class Diagram

State Machine Diagram



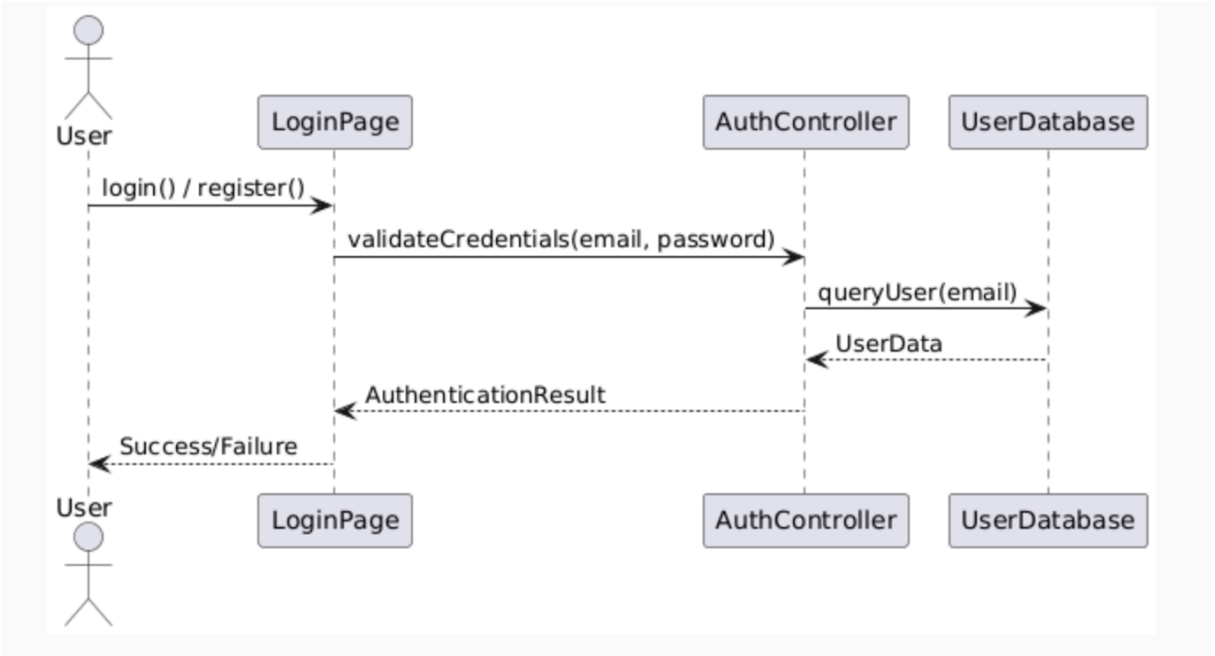
3.1.2 State Machine Diagram for the System

This diagram goes through the states for an User and admin's actions.It includes three main states: Unauthorized, Authorized and Admin Panel.

1. Initial state is unauthorized where users have restricted access to the dashboard. Once the user tries to log in or register, authentication is performed.
2. Once the user successfully logs in, a transition occurs to the Authenticated state where the user can perform several actions such as editing their profiles, connecting with peers, viewing and interacting with posts, sharing posts as well as searching jobs with filters and applying jobs.
3. If an Admin logs in, the system makes a transition to the Admin Panel, where they can apply moderating actions including managing users, posts, comments and jobs.

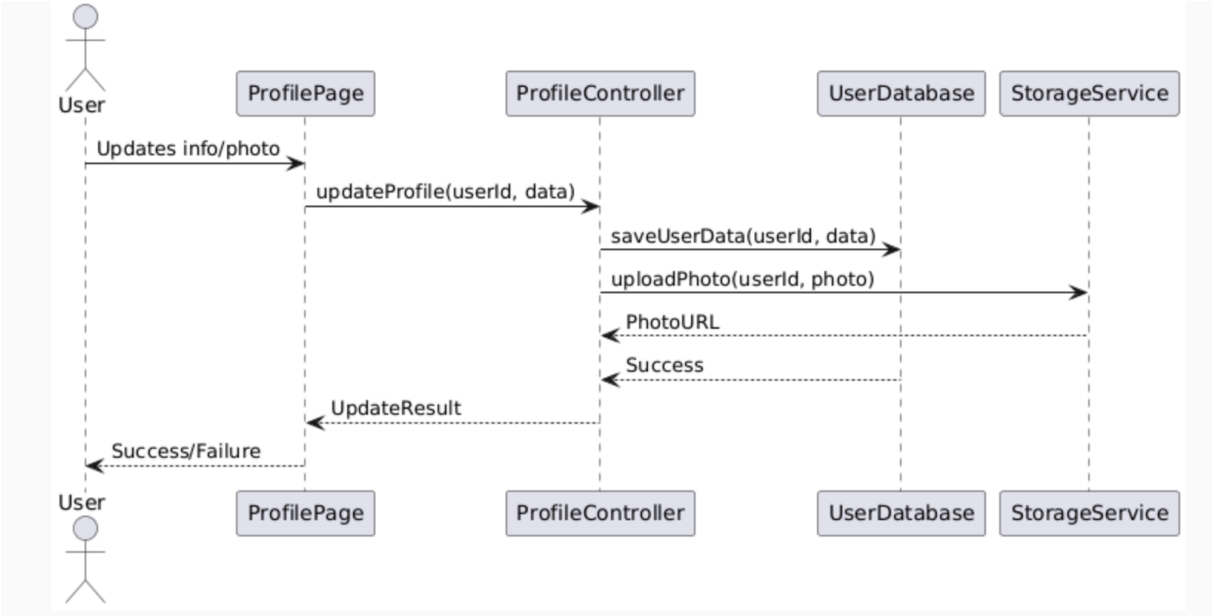
Communication Diagrams

- **User Account & Authentication**



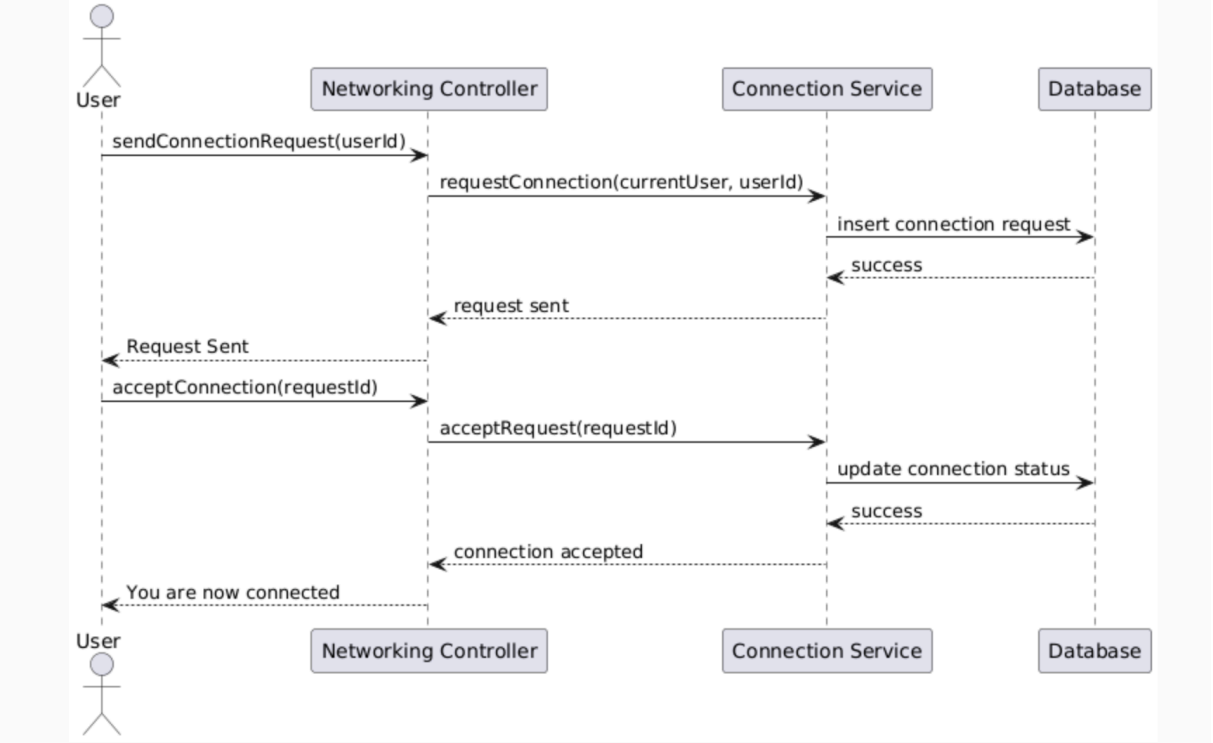
3.1.3 Communication Diagram for Login/Register

- **Profile Management**



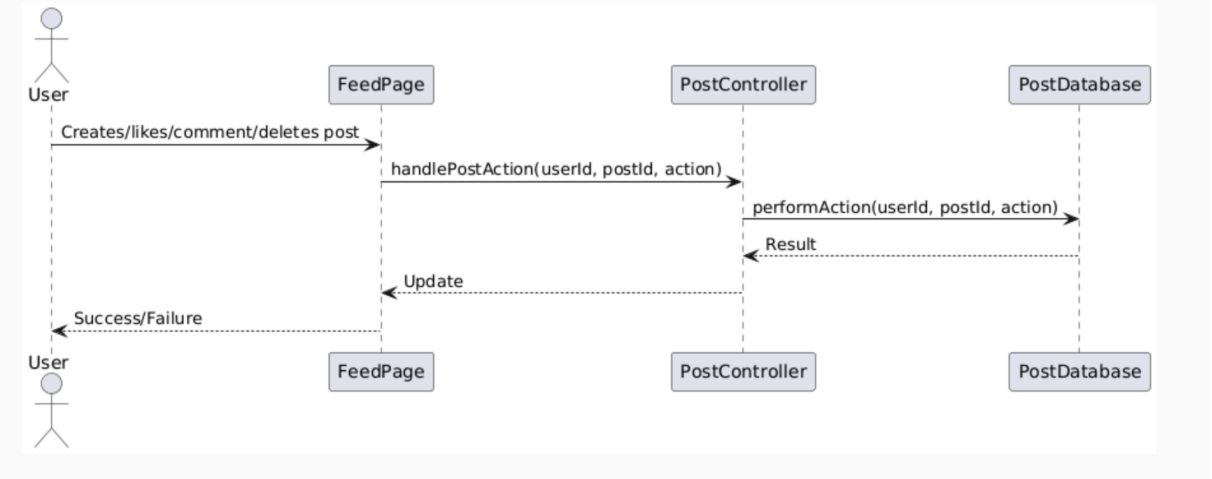
3.1.4 Communication Diagram for Profile Management

- **Connections**



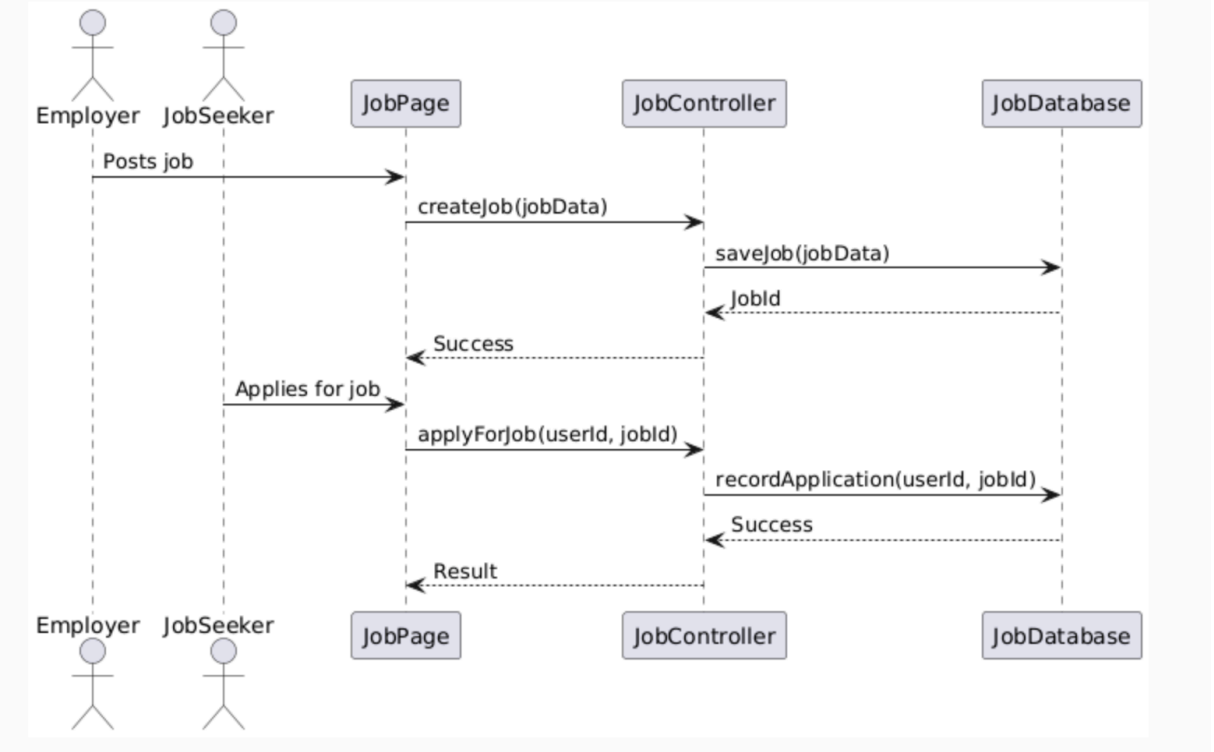
3.1.5 Communication Diagram for Connections

- **News Feed**



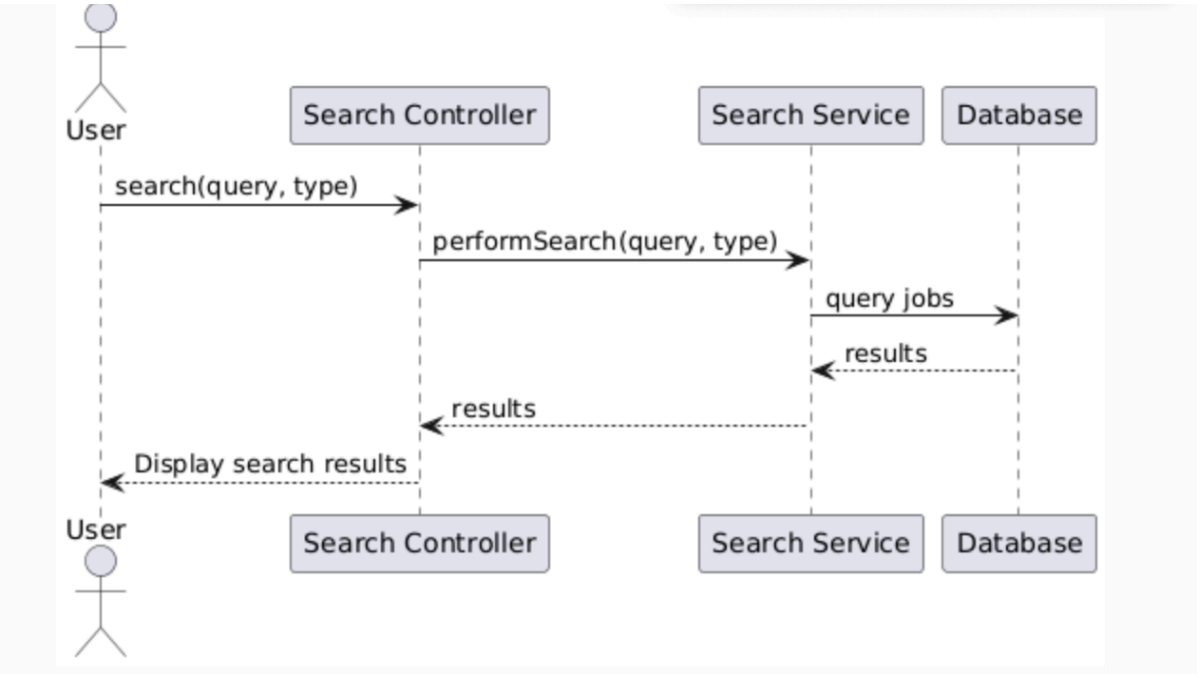
3.1.6 Communication Diagram for News Feed and Posts

- **Job System**



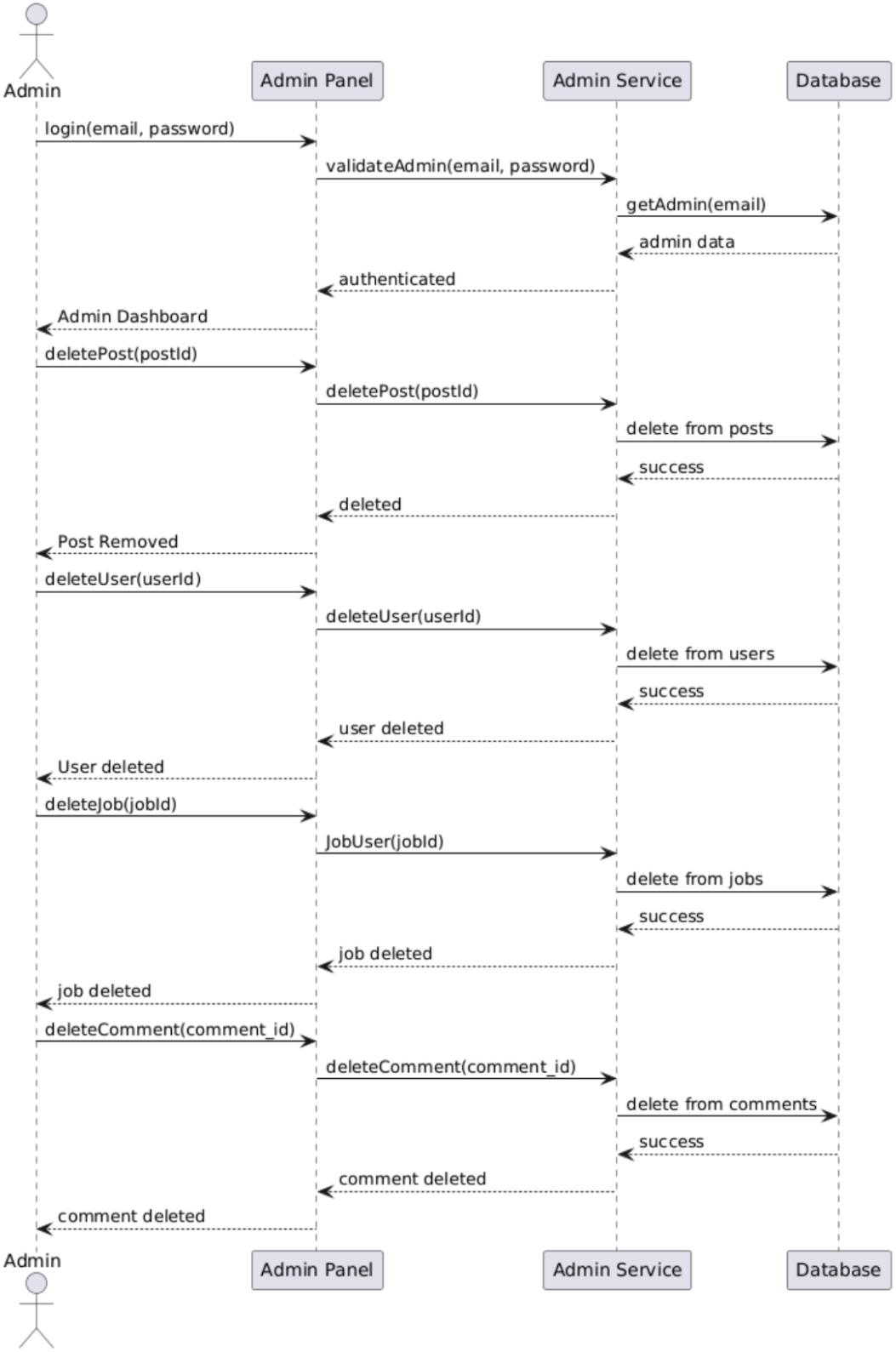
3.1.7 Communication Diagram for the Job System

- **Search & Filtering**



3.1.8 Communication Diagram for the Search and Filtering

- **Admin Panel**

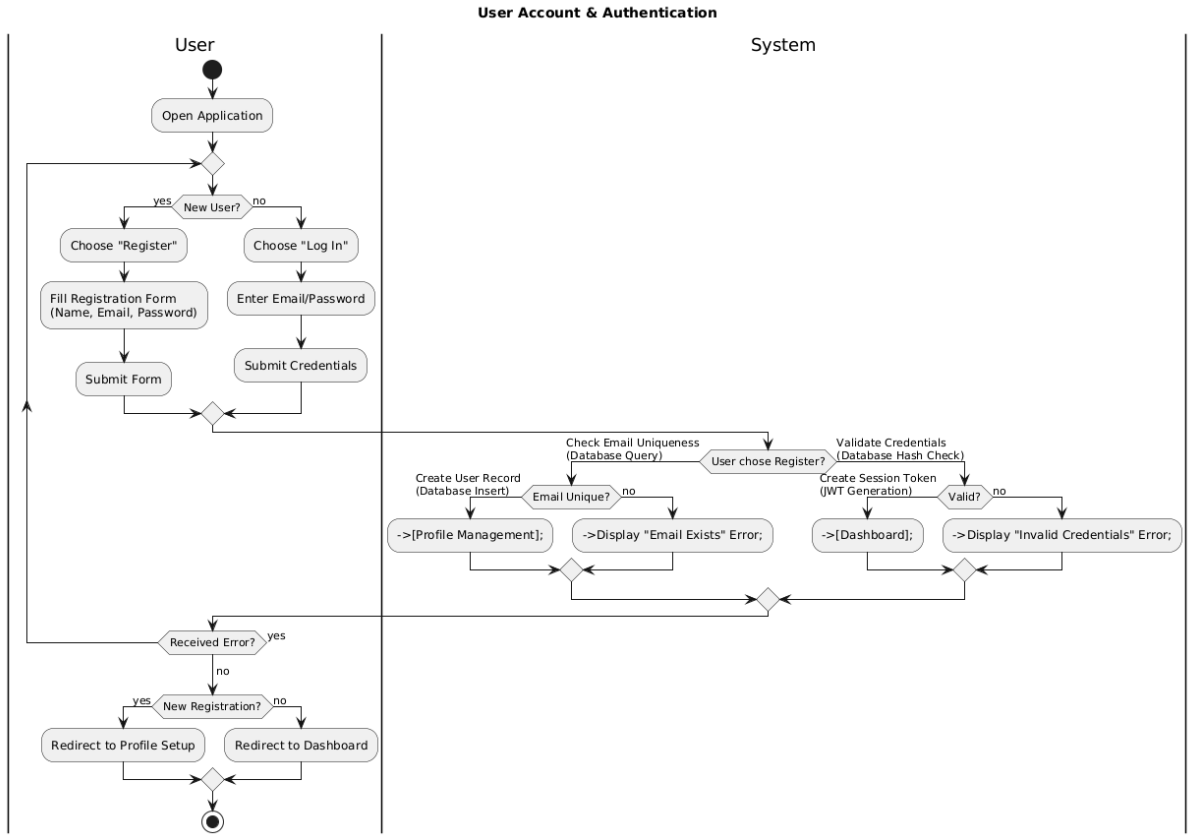


3.1.9 Communication Diagram for the Admin Panel

3.2 Process View

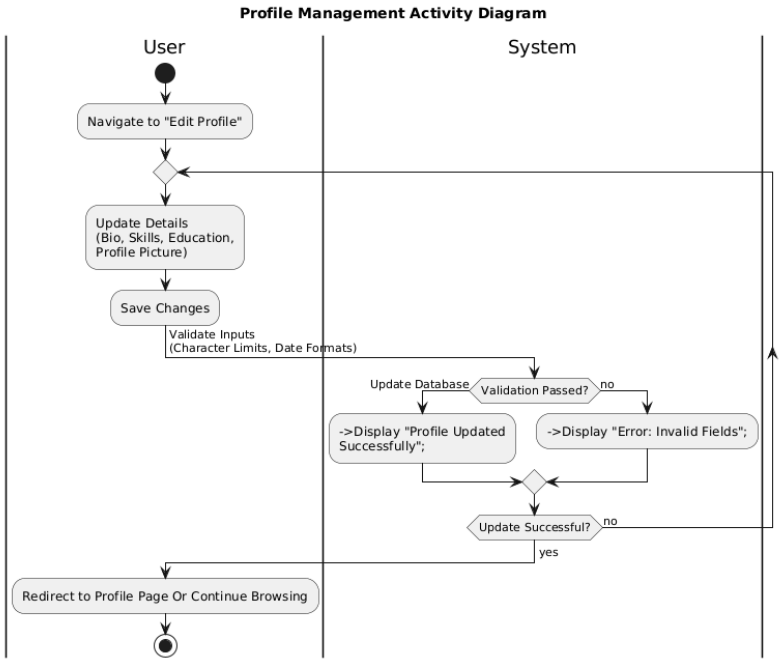
Activity Diagrams

Users register/login via email/password. The system checks email uniqueness (registration) or credentials (login), redirecting to profiles/dashboards if valid or showing errors if invalid.



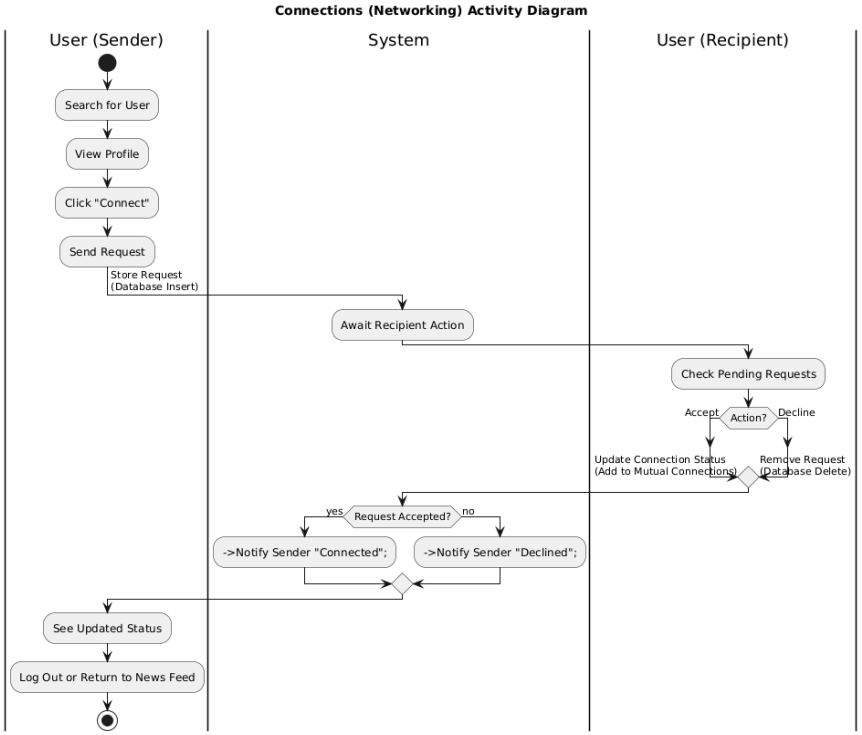
3.2.1 : User Account & Authentication Activity Diagram

Users edit profile details (bio, skills, education). The system validates inputs (format, limits) and updates the database, confirming success or flagging errors.



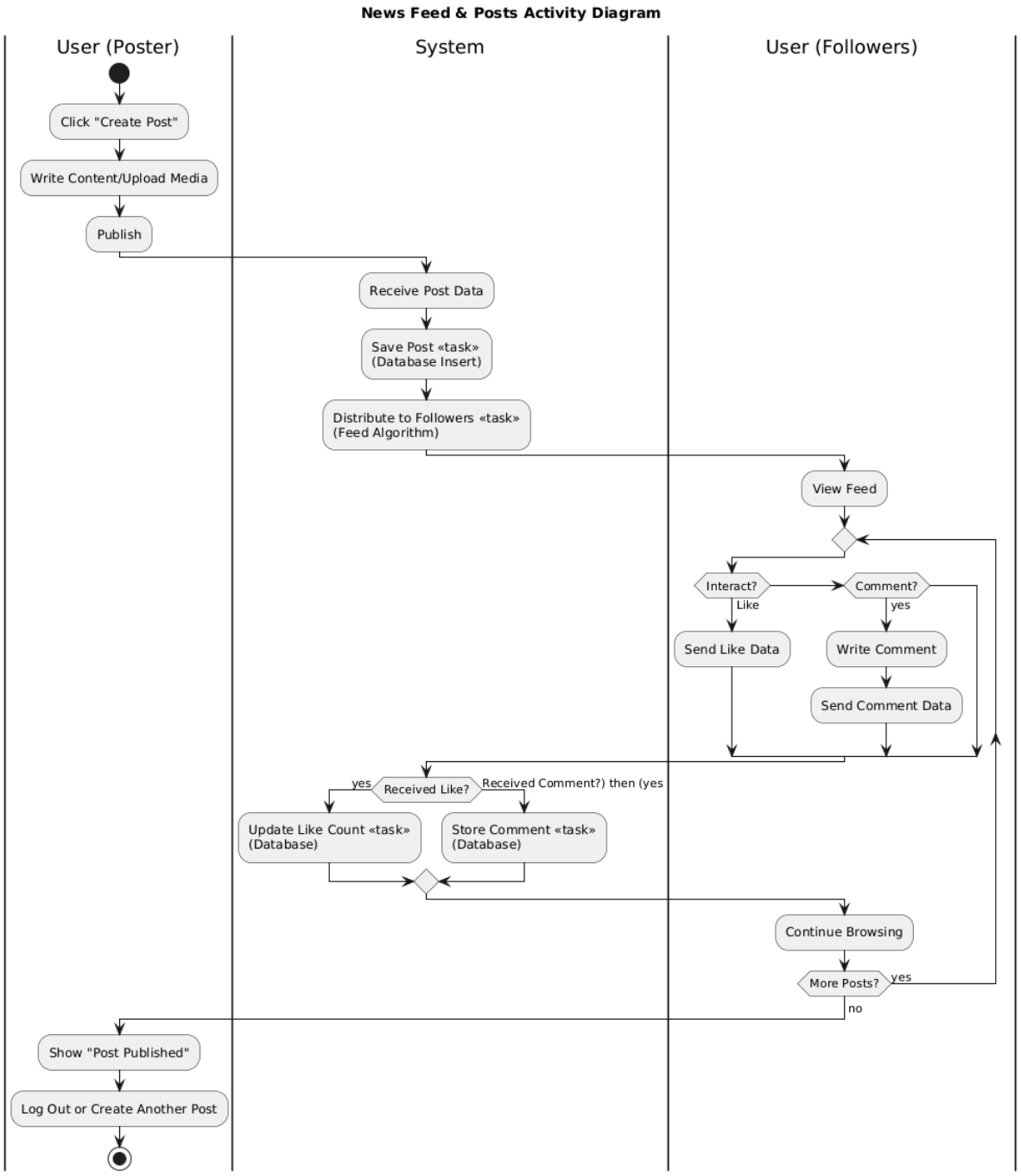
3.2.2 : Profile Management Activity Diagram

Users send connection requests, stored as "pending" in the database. Recipients accept/decline, updating mutual connections or notifying the sender.



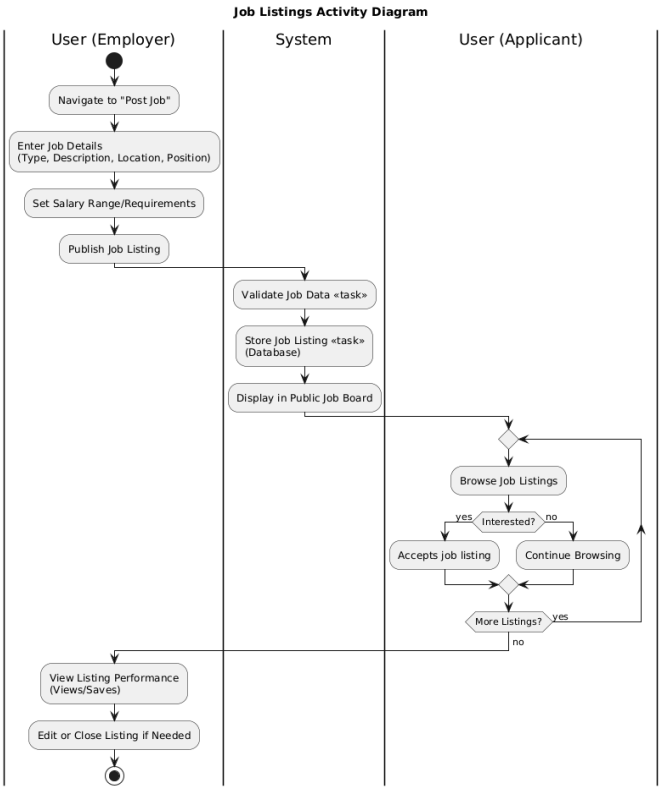
3.2.3 : Connections (Networking) Activity Diagram

Users create posts (text/media), validated and saved to the database. Followers see posts in feeds and engage via likes/comments, triggering real-time updates.



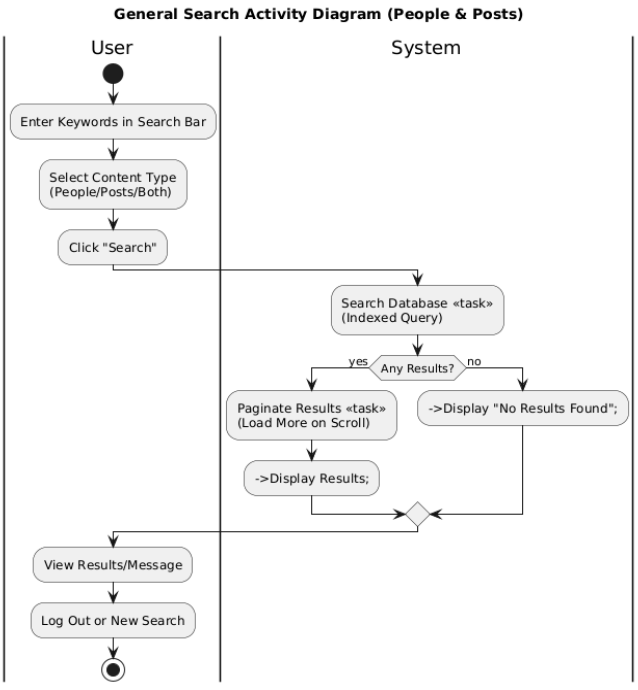
3.2.3 : News Feed & Posts Activity Diagram

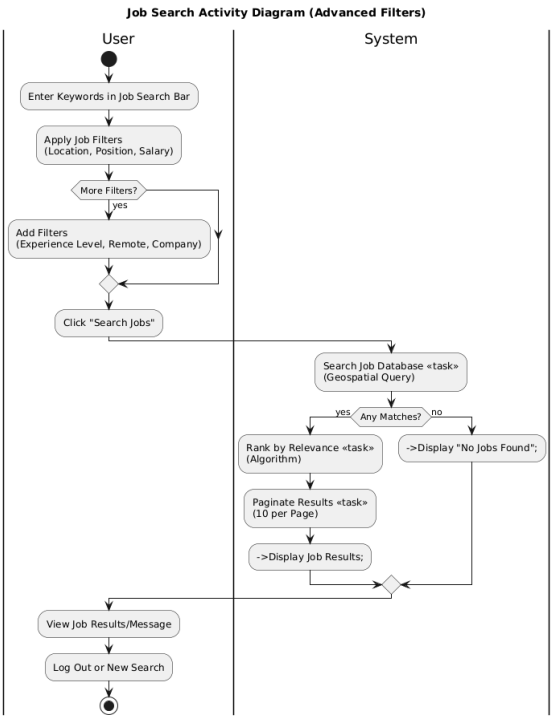
Employers post jobs (validated details), added to listings. Candidates filter/search jobs, apply with resumes, and employers receive notifications.



3.2.4 : Job Listings Activity Diagram

General search finds users/posts via keywords/filters; job search uses advanced filters (location, remote). Results show ranked matches or "no results" messages.

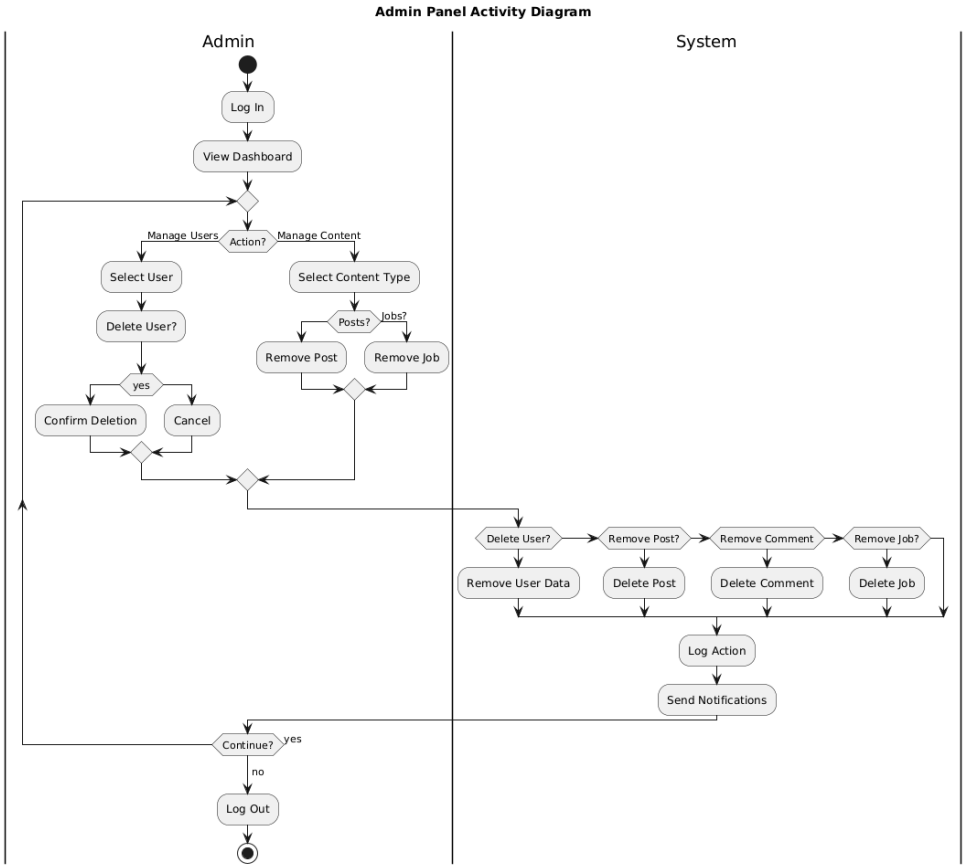




3.2.5 : Job Listings Activity Diagram

3.2.6: Job Search Activity Diagram

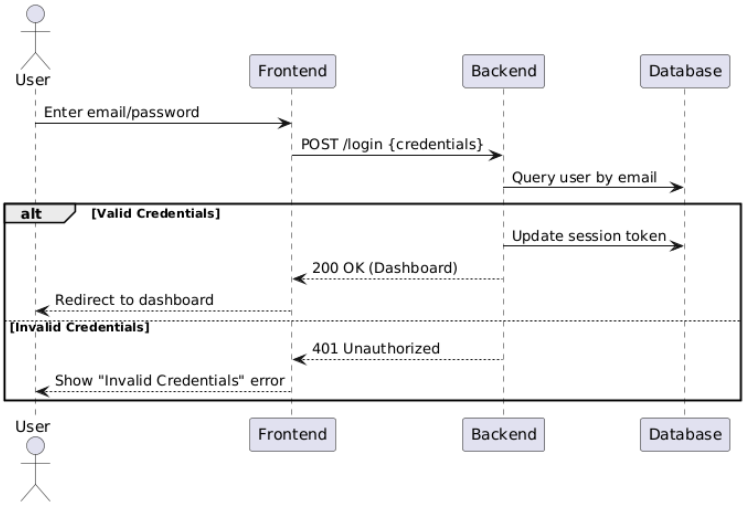
Admins log in securely, manage users/content (delete accounts/posts), and confirm actions to enforce data integrity.



3.2.7: Admin Panel Activity Diagram

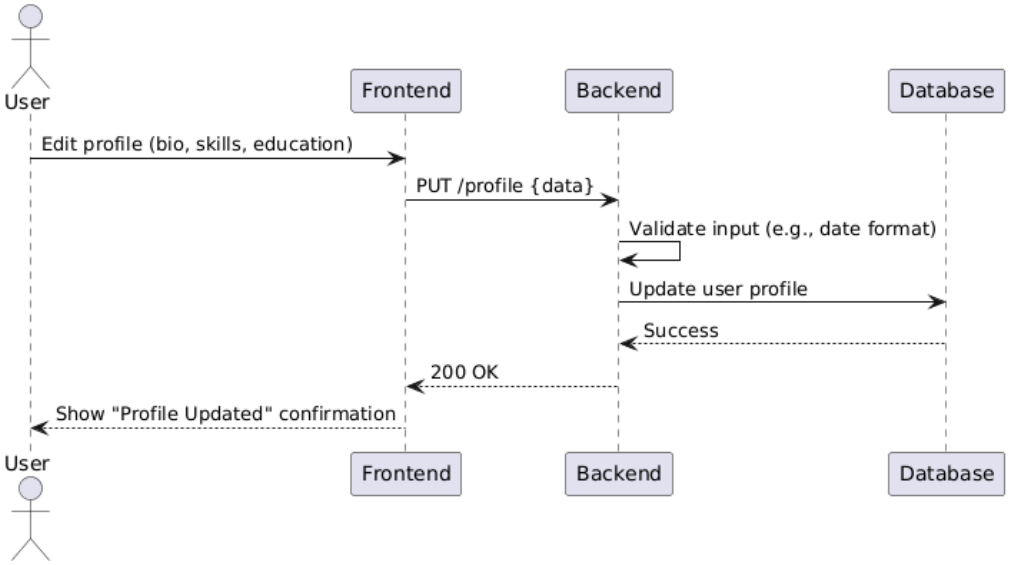
Sequence Diagrams

User submits email/password; backend checks uniqueness (registration) or credentials (login). If valid, account is created/session starts; else, errors are displayed.



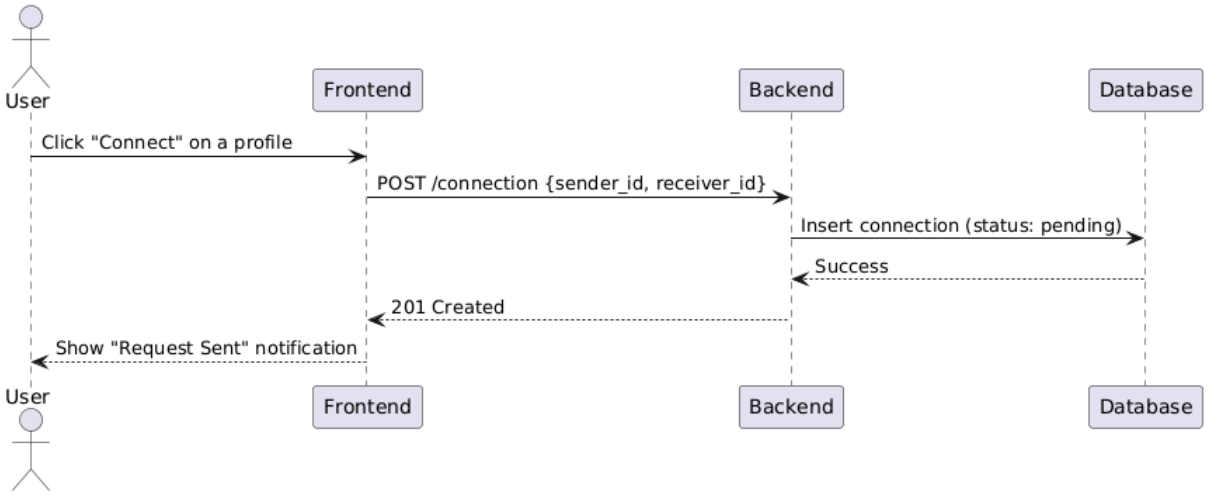
3.2.8 : User Authentication & Login Sequence Diagram

User edits profile data; backend validates inputs (format, file size). Valid data updates the database; invalid fields trigger error prompts.



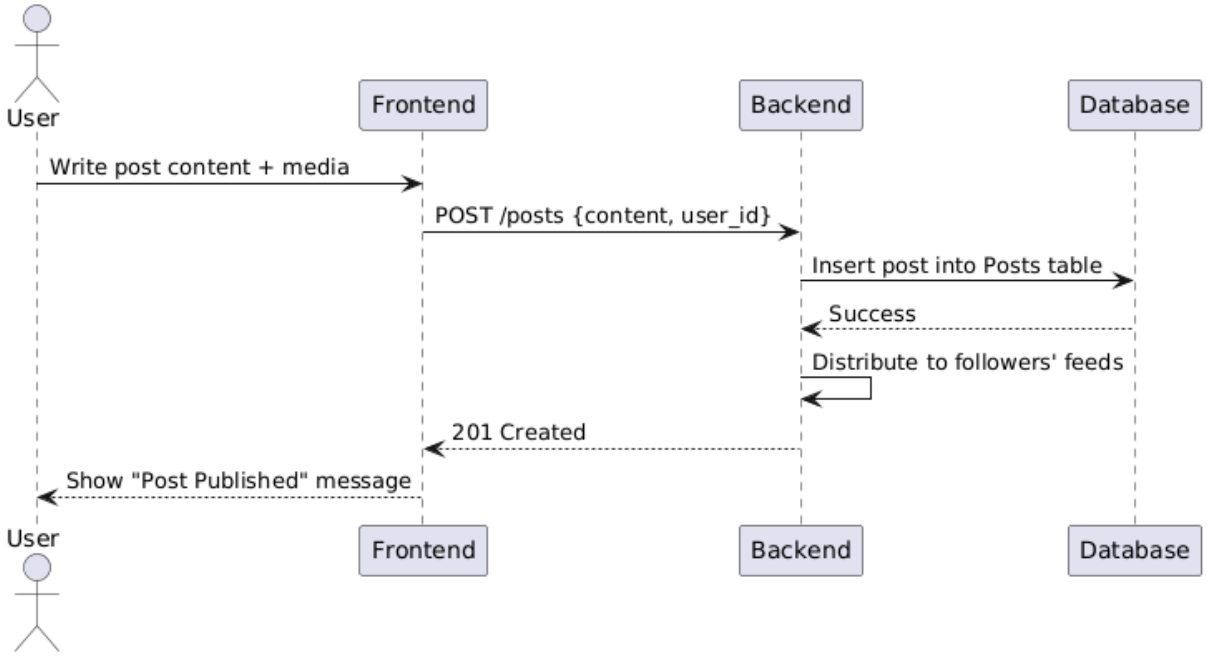
3.2.8 : Profile Management Sequence Diagram

User sends connection request; backend stores it as "pending." Recipient accepts/declines, updating both profiles and notifying the sender



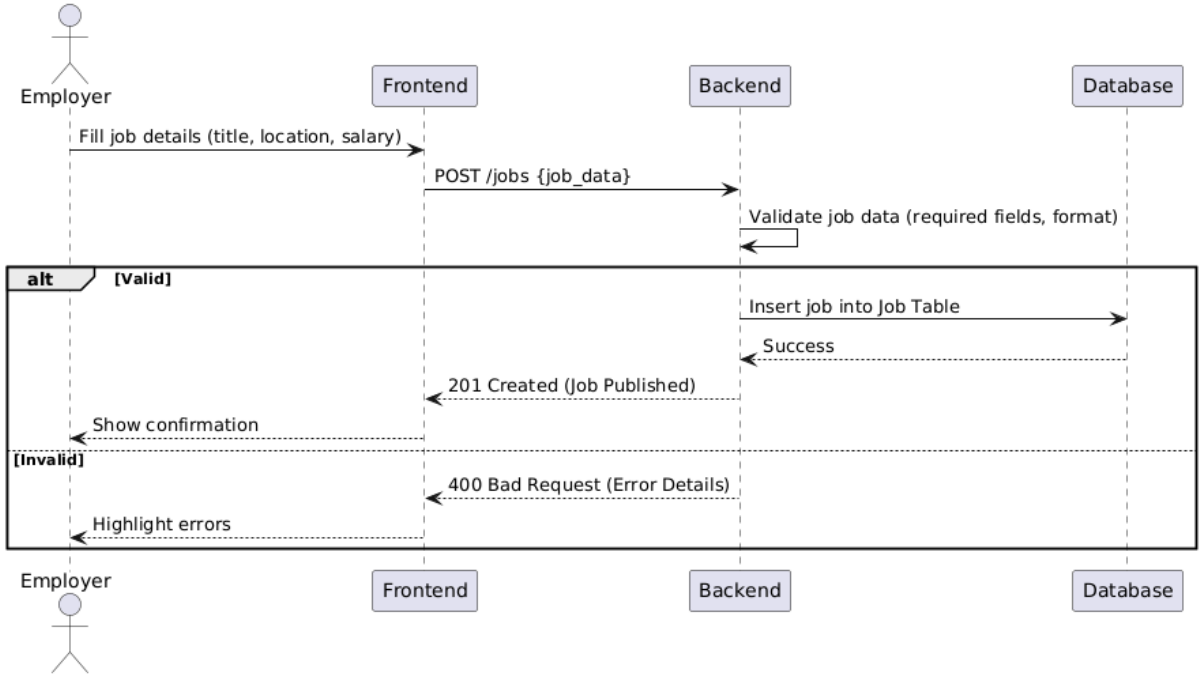
3.2.9 :Connections (Networking) Sequence Diagram

User creates a post; backend validates and saves it. Followers receive the post in feeds; likes/comments update counters in real-time.

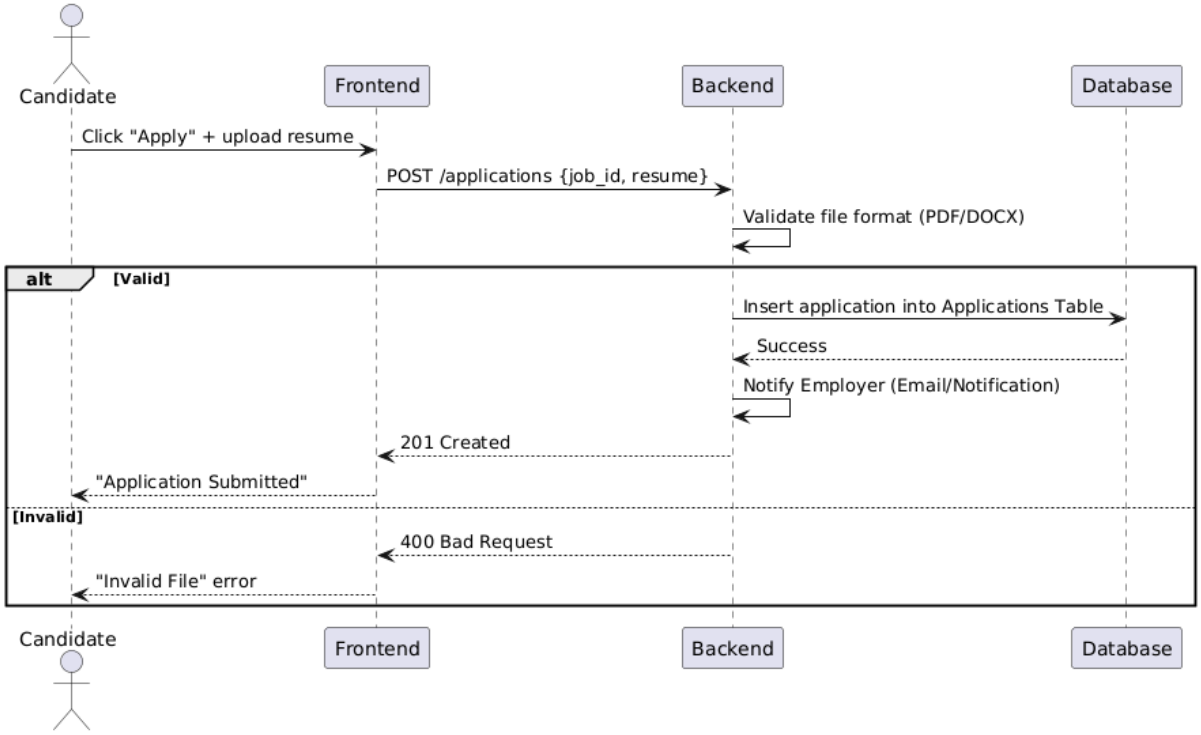


3.2.10 : News Feed & Posts Sequence Diagram

Employer posts job details; backend validates and publishes it. Candidates apply via resumes; backend checks file formats and alerts employers.

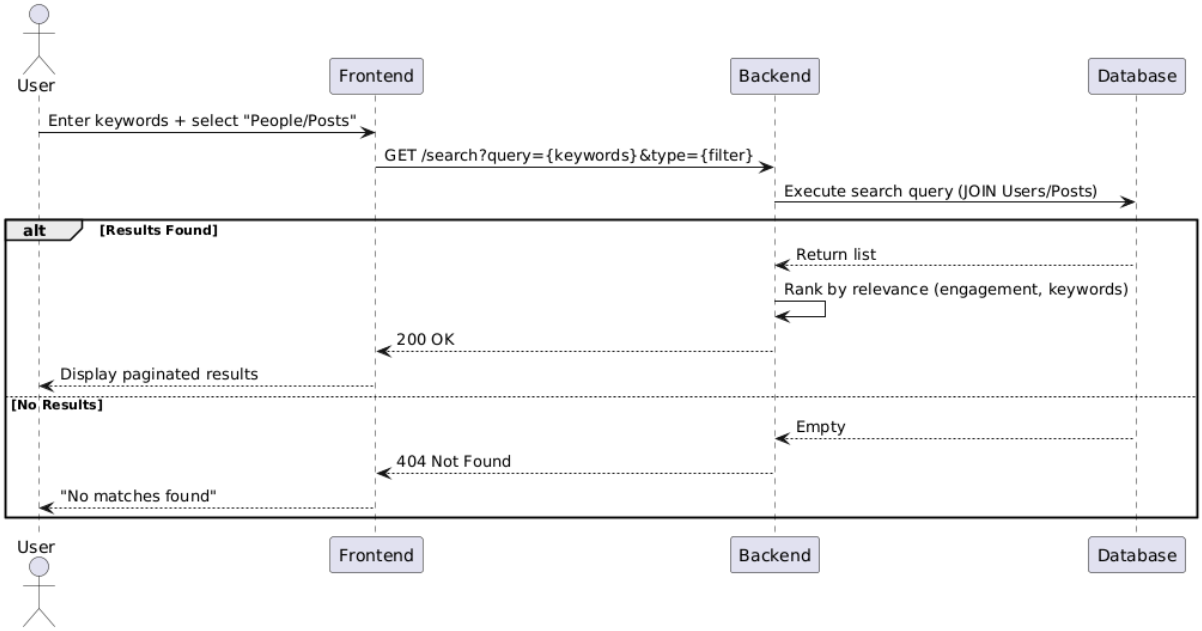


3.2.11 : Employer Job Posting Sequence Diagram

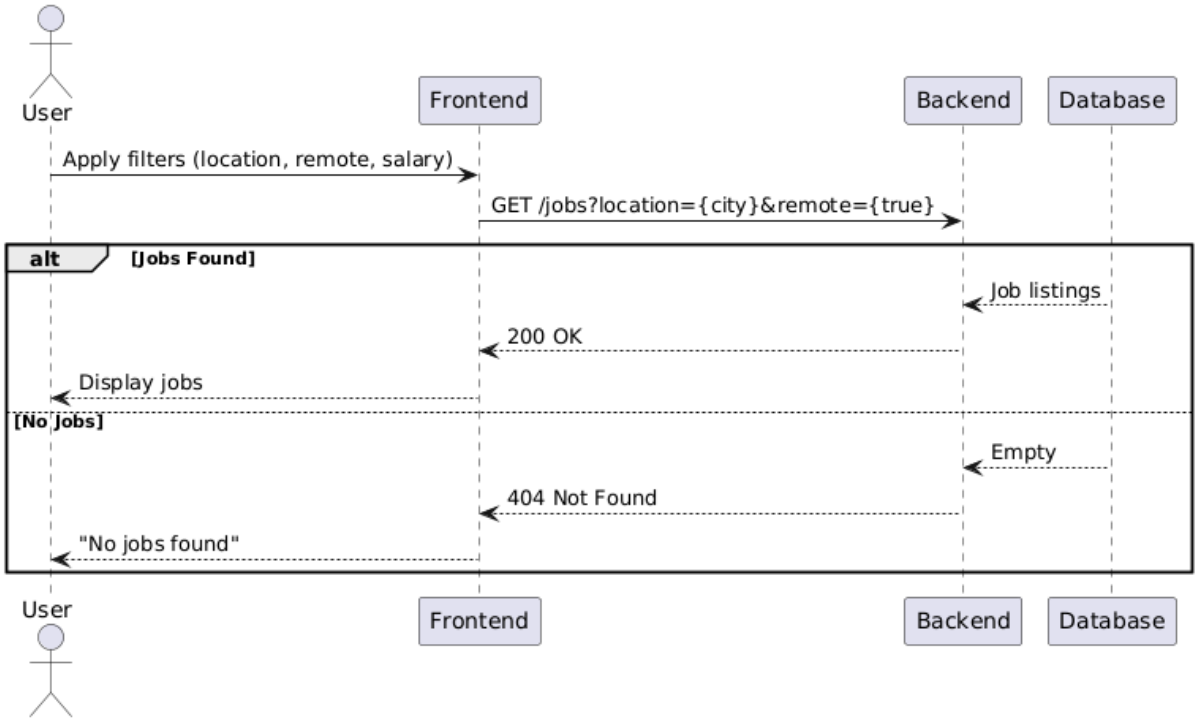


3.2.12 : Candidate Job Application Sequence Diagram

User enters keywords/filters; backend queries the database. Results are ranked and paginated, or "no results" messages appear.

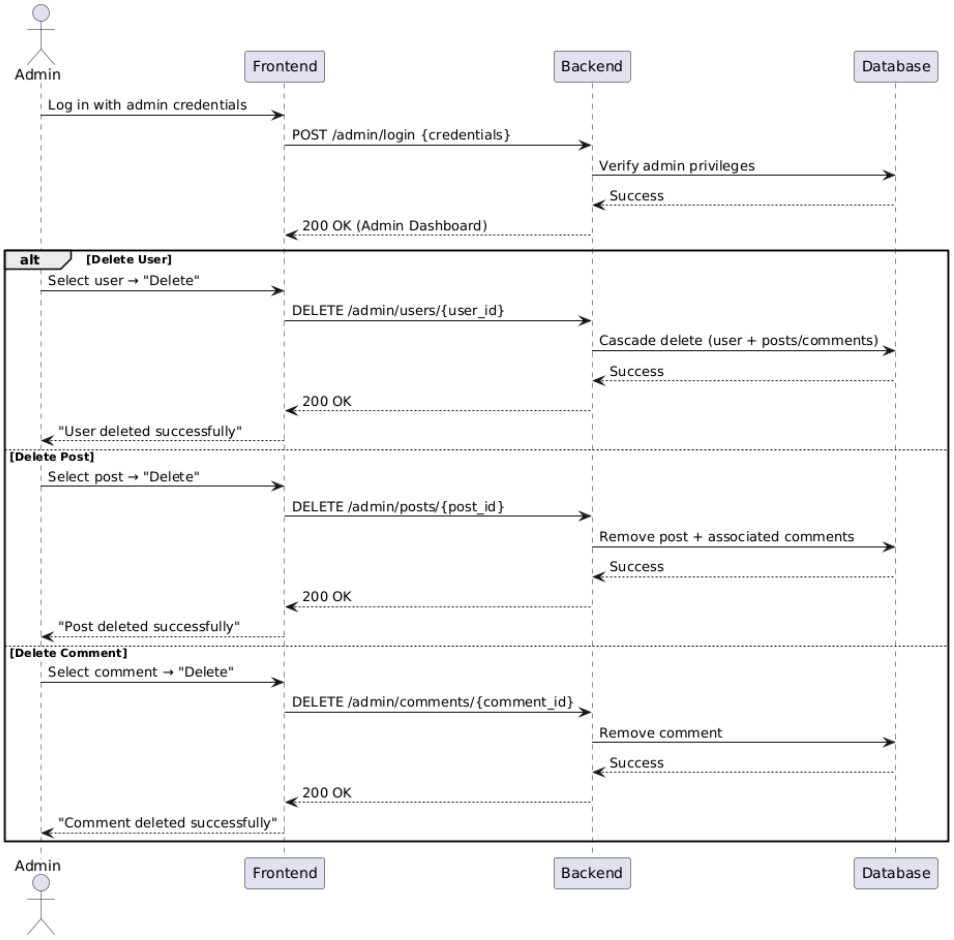


3.2.13 : General Search Sequence Diagram



3.2.13 : Job Search Sequence Diagram

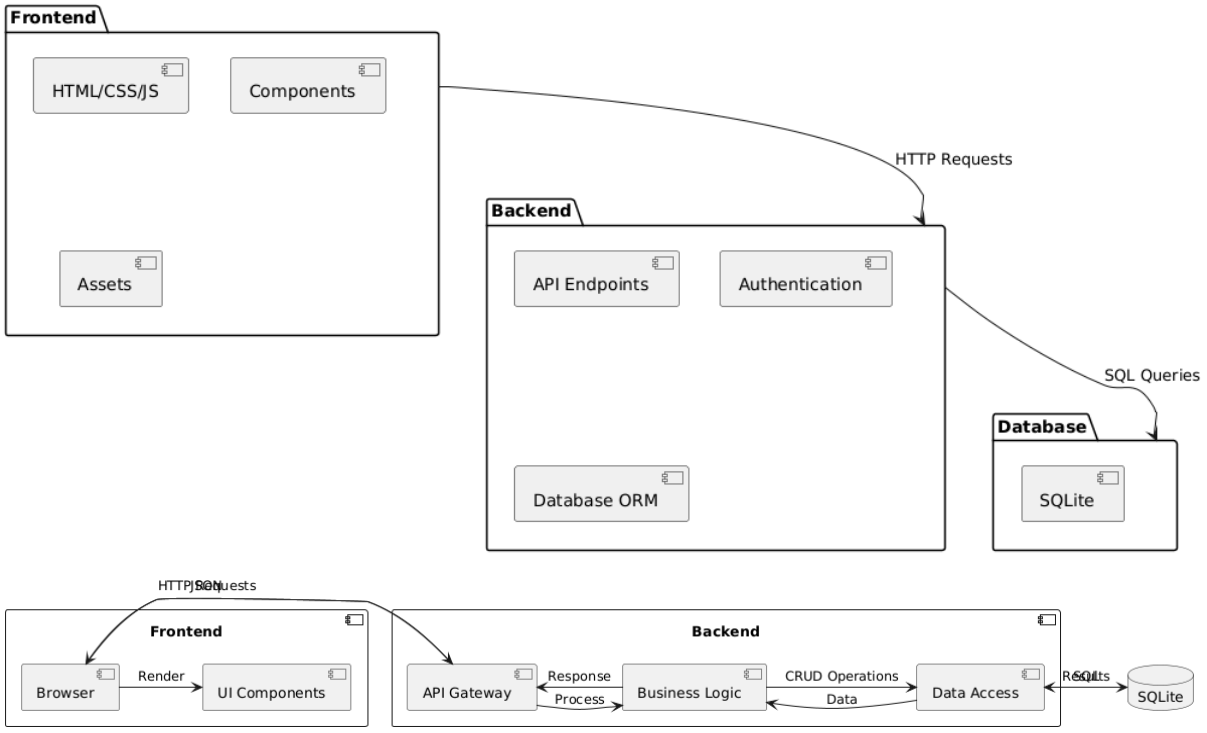
Admin deletes users/posts; backend verifies permissions and cascades deletions. Database updates, and confirmation is sent to the admin.



3.2.11 : Admin Panel Sequence Diagram

3.3 Development View

Package Diagram or Component Diagram

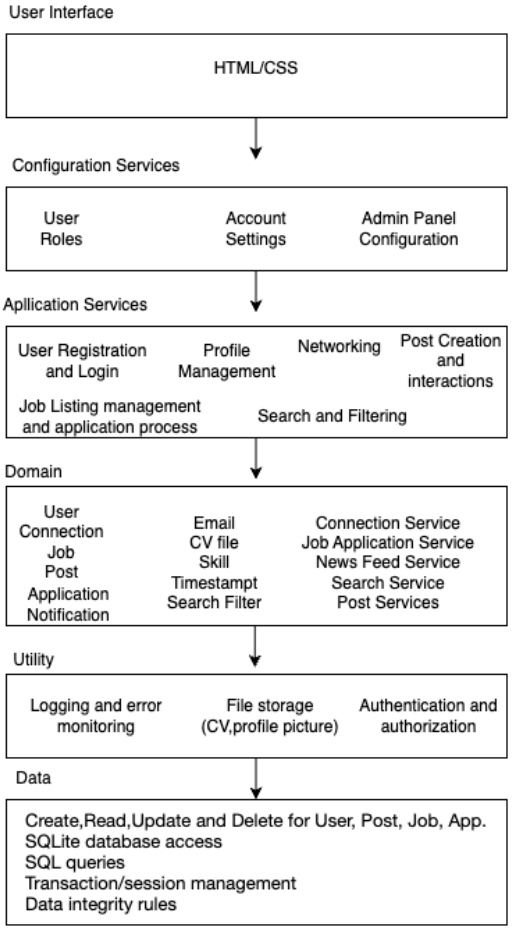


Düzenlenecek.

3.4 Physical View

Architecture pattern & architectural design

(Layered structure)



4. Restrictions, Limitations, and Constraints

All non-functional requirements

5. User Manual

6. Source codes