



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Experiment-3

**Student Name:** Rakshit Chauhan

**Branch:** CSE

**Semester:** 6th

**Subject Name:** System Design

**UID:** 23BCS12628

**Section/Group:** KRG-3B

**Date of Performance:** 28/01/26

**Subject Code:** 23CSH-314

1. **AIM :** To design a social media platform similar to Facebook or Instagram

2. **Objective:**

- To design and develop a **scalable social media application** that allows users to register, connect, and interact through posts.
- To provide core social networking features such as **posting, following, liking, commenting, and feed generation**.
- To ensure **high availability and low latency**, making the application accessible to a large number of users at all times.
- To design **RESTful APIs** that efficiently handle user onboarding, content management, and user interactions.

3. **Tools Required:**

- **Frontend:** HTML, CSS, JavaScript, React.js
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **API Testing:** Postman
- **Authentication:** JWT
- **Media Storage:** Cloudinary / AWS S3
- **Version Control:** Git, GitHub
- **Deployment:** AWS / Render / Vercel

4. **SYSTEM DESIGN / SYSTEM SPECIFICATION:**

4.1. **Functional Requirements:**

- Client should be able to register and login to the application.
- Client should be able to create post (text / image / videos)
- Client should be able to follow each other (or send friend requests).
- Client should be able to like or comment on the post.
- Client should be able to view the feed of post from users they follow.

4.2. **Non-functional Requirements:**

- **Scalability:** 500M daily active users.
- **Consistency & Availability:** For social media applications, **availability is more important than consistency**.

If an application is not available or working when users need it, then building such an

application has no real value.

For example, if Instagram is down for one hour, it becomes a **major problem** for users.

However, if Instagram is running and a post takes **500 ms longer** to reach followers, it is **not a big issue** when compared to downtime.

Therefore, in social media systems:

**Availability is prioritized over Consistency.**

- Latency: (Uploading speed of publishing post): 500ms to upload post

### 4.3. Core-Entites of the System:

1. User
2. Posts
3. Followers
4. Like and Comment
5. Feed

### 4.4. API Endpoints Creation:

#### A. User On-boarding API's

1. **User Registration: POST API CALL:** POST / api / users / register\_user
2. **User Login: POST API CALL:** POST / api / users / login
3. **User Data Display: GET API CALL:** GET / api / users / {user\_id} / profile
4. **User Data Update: PUT API CALL:** PUT / api / users / {user\_id} / profile

#### B. User Post's

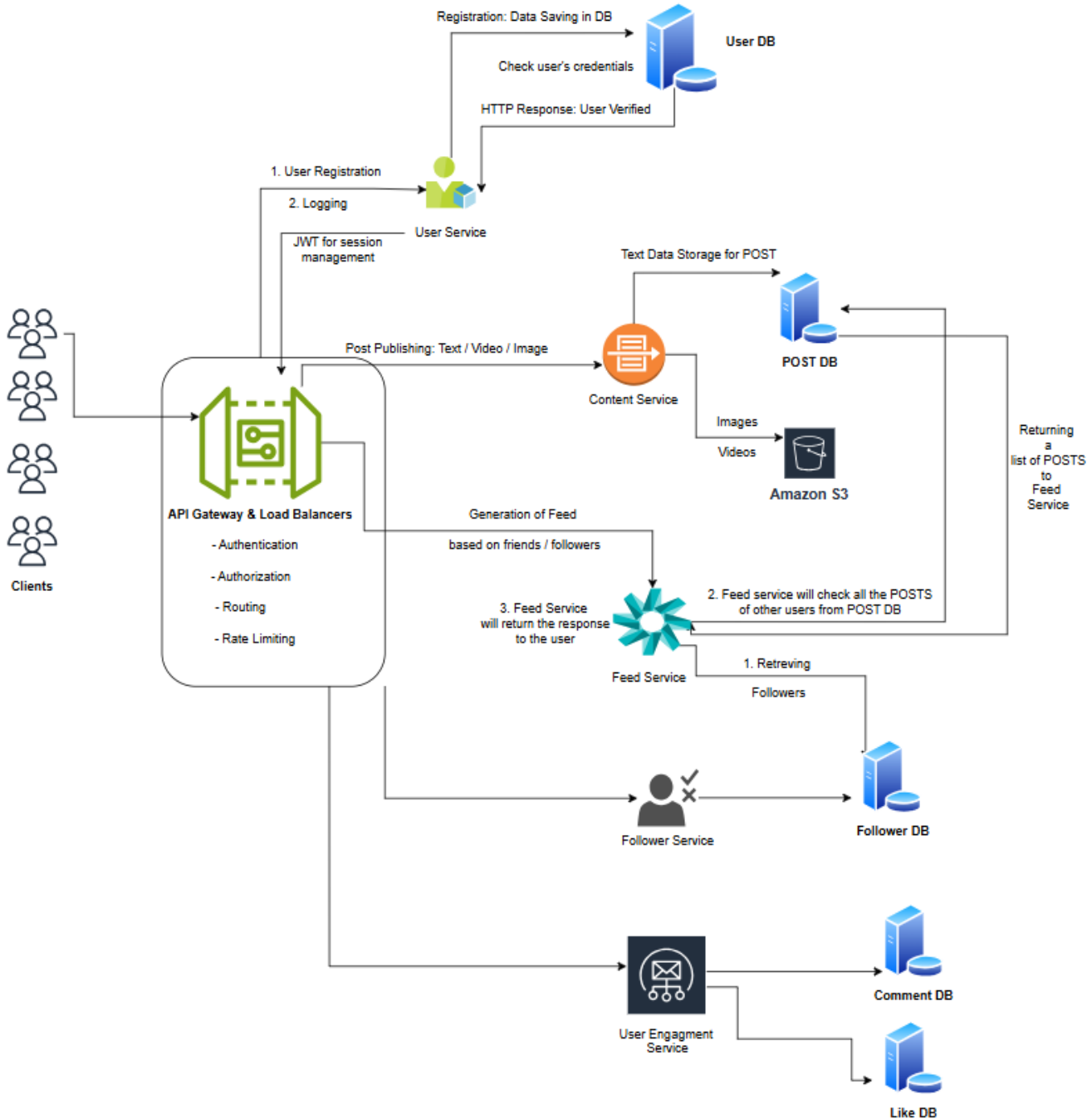
1. POST / api / user\_id / posts
2. GET / api / posts / {post\_id}
3. PUT / api / posts / {post\_id}
4. DELETE / api / posts / {post\_id}
5. GET / api / posts / feed / limit = {limit} & offset = {offset} : PAGINATION
6. GET / api / users / {user\_id} / posts: PAGINATION

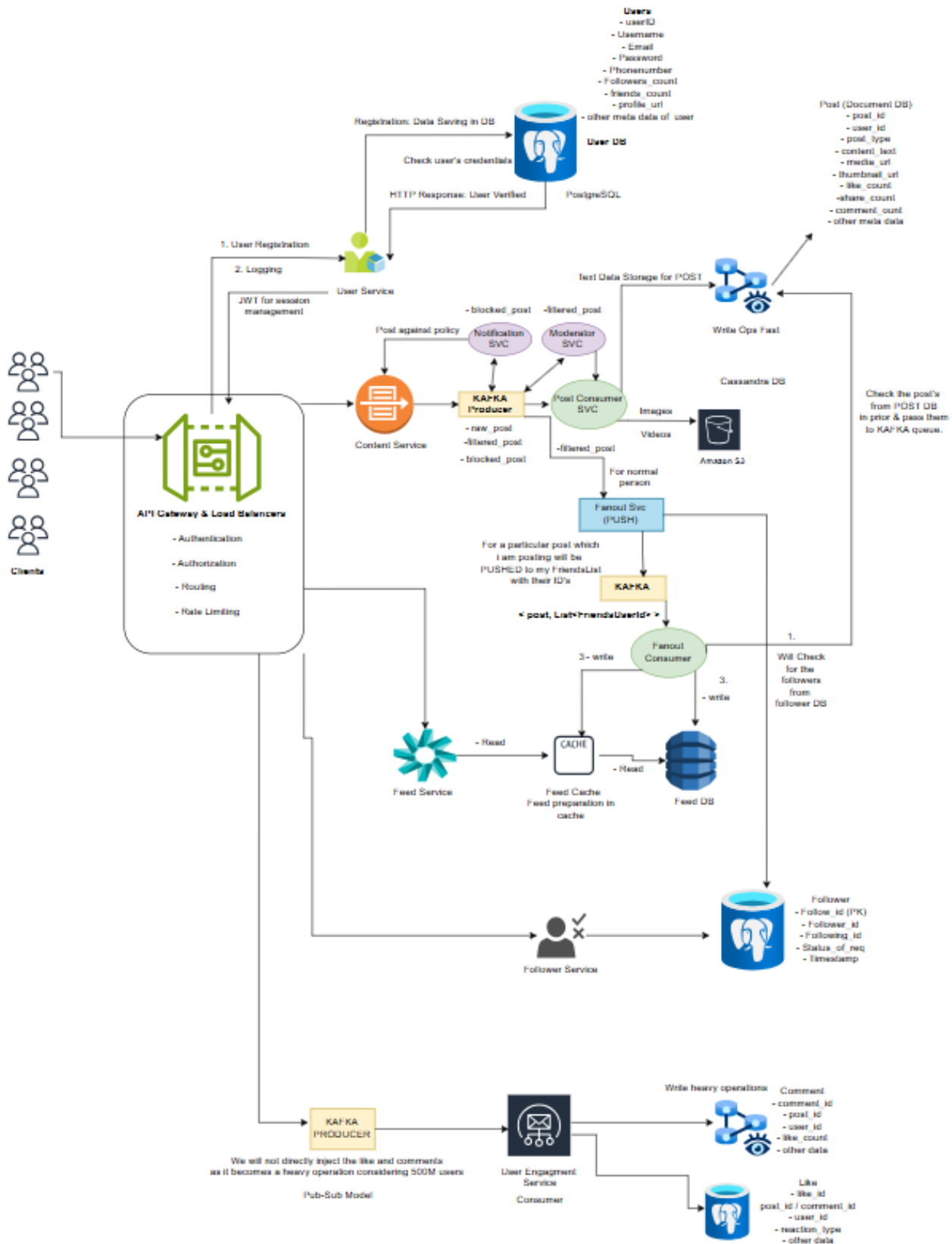
#### C. User Interactions

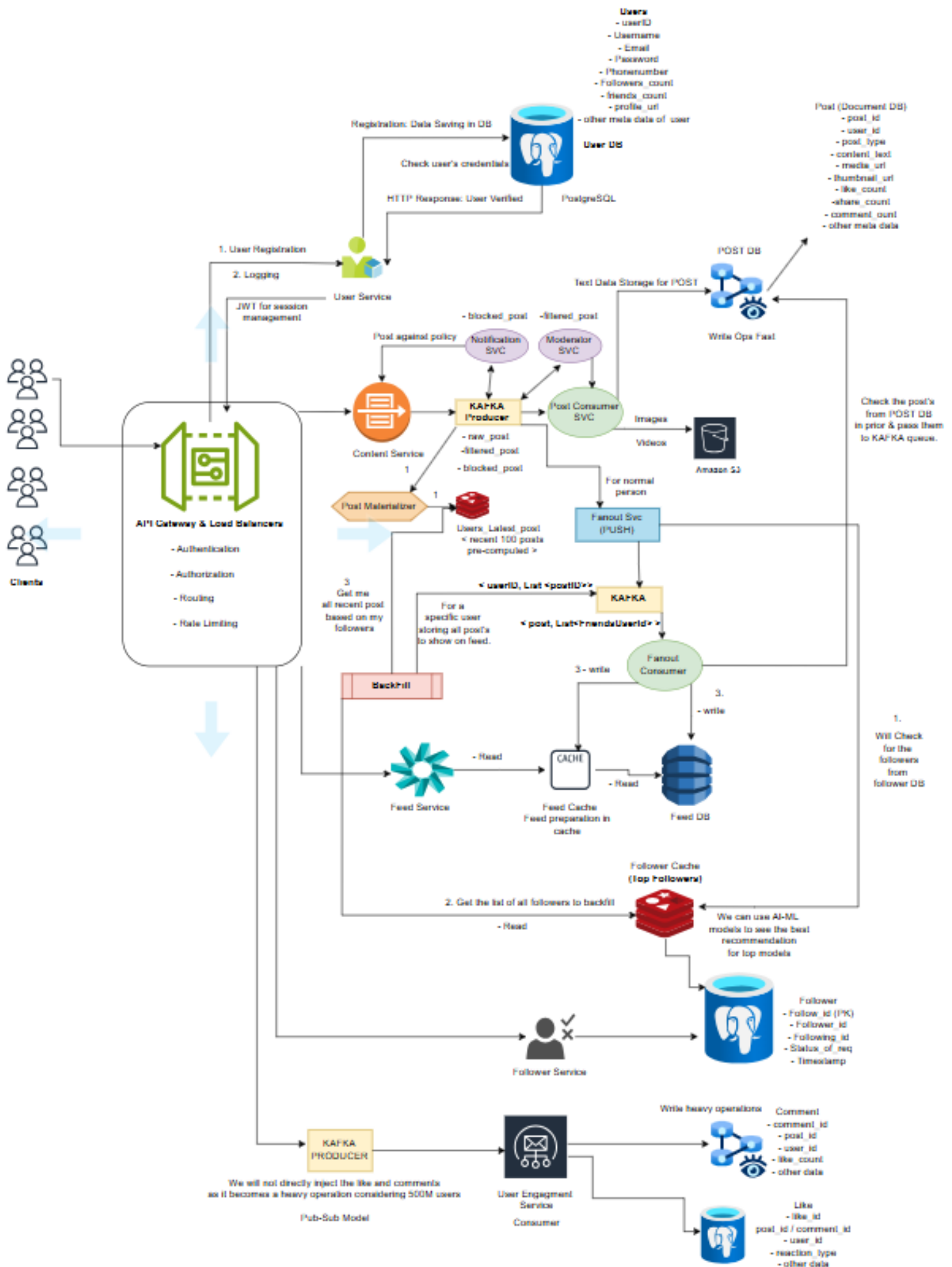
1. POST / api / posts / {post\_id} / like
2. DELETE / api / posts / {post\_id} / unlike
3. POST / api / posts / {post\_id} / comments
4. GET / api / posts / {post\_id} / comments
5. PUT / api / {post\_id} / comments / {comment\_id}
6. DELETE / api / {post\_id} / comments / {comment\_id}
7. POST / api / users / {user\_id} / follow
8. DELETE / api / users / {user\_id} / unfollow

## 5. HLD(High Level Design):

We have to follow a distributed / micro-services approach not the monolithic one.







## 7. Learning Outcomes

- Understand and define **functional and non-functional requirements** of a social media system.
- Design a system that prioritizes **availability over consistency** as per real-world social media use cases.
- Identify and model **core entities** such as Users, Posts, Followers, Likes, Comments, and Feed.
- Design **RESTful API endpoints** for user management, post handling, and social interactions.
- Apply concepts of **scalability, latency, and pagination** in system design.
- Gain practical exposure to **real-world system design considerations** used in large-scale applications like Instagram or Facebook.