GATEway to Success - Software Engineering Report

Problem-Statement:

Preparing for GATE is a pretty demanding process, often riddled with problems such as inefficient time management, no proper study plan, and a lack of access to the right resources. It becomes difficult for aspirants to balance their academic or professional commitments with staying on track in preparation.

GATEwayToSuccess addresses all such struggles by providing a simple yet effective web-based platform designed to streamline the preparation journey. With HTML, CSS, and JavaScript as core frontend technologies in construction, this application provides options such as customized study plans based on individual goals, a centralized resource hub containing study materials and mock tests, and an interactive quiz system to evaluate progress.A centralized resource hub consolidates study materials, including downloadable PDFs ,making it easier for users to access relevant content in one place. Additional features include countdown timers that help aspirants visualise their preparation timeline for more motivation.

The project faces challenges in producing a user-friendly interface suitable for all aspirants in diverse engineering backgrounds, with functionality that makes use of the dynamic possibilities of JavaScript. It has to ensure it is responsive enough for every device for accessibility purposes. GATEwayToSuccess empowers students with the structure and tools they need to succeed, showing how technology can effectively simplify and enhance the preparation process for high-stakes competitive exams.

Feasibility Study :

Existing Problem *:*

The process of preparing for GATE is challenging, but most of the existing platforms fail to cater to the diverse needs of aspirants. These websites provide generic content and static study materials without addressing individual learning goals, preparation timelines, or dynamic progress tracking. This lack of personalized guidance often leaves students overwhelmed and struggling to align their preparation with their specific objectives.

1. Technical Feasibility

- The platform will be developed using HTML, CSS, and JavaScript for the frontend, Node.js for backend logic, and MySQL for data management.
- It will support adaptive learning with personalized study plans and mock tests.
- Cloud-based deployment ensures scalability and accessibility.

2. Operational Feasibility

- User-friendly **UI/UX** to facilitate seamless navigation and engagement.
- Minimal training required for aspirants and educators.
- Features like progress tracking and study recommendations ensure better user adoption.

3. Resource Feasibility

- Requires skilled developers, database administrators, and subject matter experts.
- Utilizes **existing study materials, AI-based analytics, and real-time user feedback** to optimize learning.

REQUIREMENTS ENGINEERING

A.INCEPTION

Identification of Stakeholder(s)

1. Aspirants (GATE Students) :

- Primary beneficiaries of the platform, using it for structured GATE preparation with study plans, quizzes, and progress tracking.
- Will provide feedback on usability, content quality, and performance tracking effectiveness.

2. Subject Experts :

- Knowledge providers who contribute high-quality study materials, quizzes, and expert guidance to help aspirants prepare effectively for GATE.
- Ensure content accuracy by curating subject-specific resources, resolving doubts, and continuously updating study materials based on the latest exam trends.

Recognized Viewpoints:

- Aspirant (B.Tech, 1st Year, **Pardhu**) – Needs a structured roadmap for early preparation without disturbing regular academics.
- Aspirant (B.Tech, 3rd Year, **Chaitanya**) – Wants mock tests, last-minute revisions, and performance analytics to boost confidence.
- Working Professionals Preparing for GATE – Require a flexible, self-paced learning module that fits their work schedule.

- Non-Engineering Aspirants – Need personalized study materials and foundational concepts for a better understanding.

Working Towards Collaboration:

- Stakeholders expect a user-friendly and adaptive GATE preparation platform.
- They are willing to provide feedback on the system's effectiveness and suggest improvements.
- Collaboration will ensure continuous enhancements in study materials, test patterns.

Preparing First Few Questions for Requirement Elicitation

1. What challenges do you face while preparing for GATE?
   - Lack of personalized study plans that match individual strengths and weaknesses.
   - Limited free resources; high-quality materials are usually paid.
   - Difficulty in tracking progress and weak areas.
   - Overwhelming syllabus and time management issues.
2. How structured is your current GATE preparation?
   - "I follow a random sequence of topics without tracking progress."
   - "I have a goal but lack a step-by-step roadmap."
   - "I prepare with coaching classes but struggle with self-assessment."
3. Where do you refer for GATE study materials and updates?
   - "YouTube, Telegram groups, and random online sources."
   - "Reference books and coaching institutes."
   - "Asking seniors or professors for guidance."
4. How easily do you access free mock tests and quizzes?
   - "I struggle to find quality mock tests in one place."
   - "Many tests require paid access."
   - "No centralized platform for daily practice and instant feedback."
5. What is your ideal GATE preparation experience?
   - "I want a personalized, flexible study plan based on my schedule."
   - "A system that tracks my progress and suggests improvements."
   - "A platform that provides all necessary study materials, quizzes, and performance insights."
   - "A roadmap aligned with GATE syllabus and real-time performance analytics."

B.ELICITATION :

1. Interview

- One-on-one discussions with aspirants and subject experts.
- Gather personalized requirements regarding learning difficulties, content preferences, and feature expectations.

2. Brainstorming

- Conducted team sessions to analyze user feedback.
- Identified core challenges and potential solutions for personalized learning.

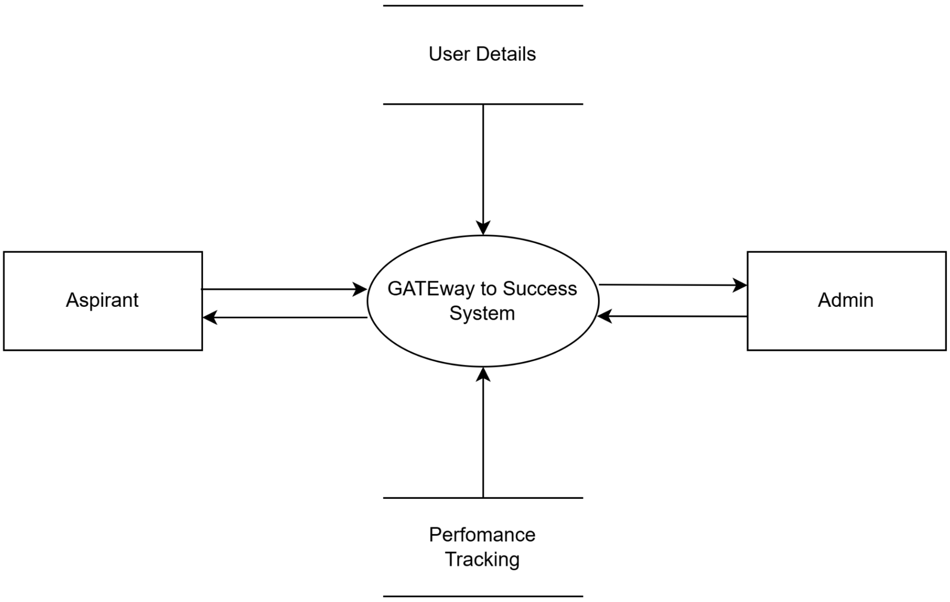
3. Quality Function Deployment(QFD) :

Quality Function Deployment (QFD) is a systematic process used to translate user needs into technical requirements for the system. It ensures that the platform is user-centric, efficient, and aligned with aspirants' expectations.

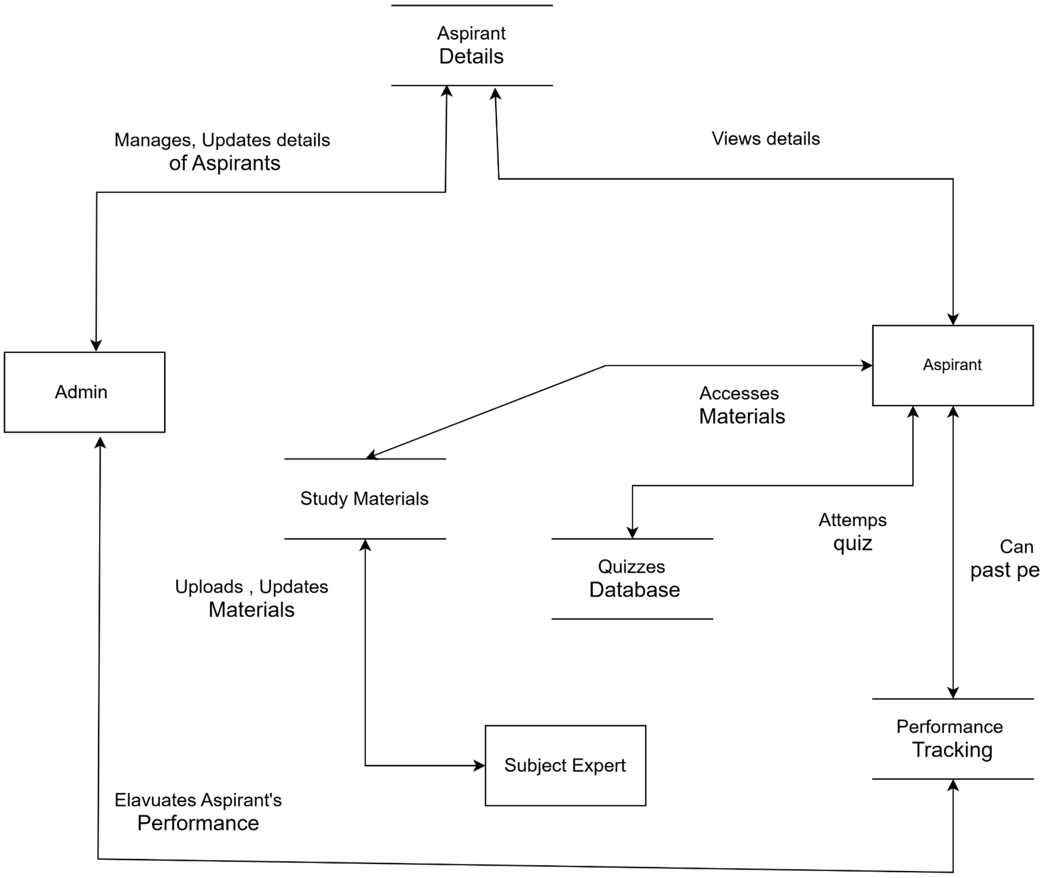
REQUIREMENT ANALYSIS :

Data Flow Diagrams (DFD) :

DFD LEVEL-0 : Level - 0 depicts the entire system as a Single Process. The Aspirant and Admin interacts with GATEway to success system making changes to User Details and Performance Tracking Databases.



DFD-LEVEL-1: Level-1 DFD depicts the main system into multiple sub-processes . Here, Aspirant access study materials, view details, attempt quiz, track performance whereas Admin manages aspirant details, evaluates aspirant's performance. The open rectangle represents the databases and the closed rectangle represents the actors.



REQUIREMENT TYPES:

Functional Requirements :

1. Syllabus :
   - The entire syllabus for different subjects or courses will be displayed when logging into the interface.
   - It is possible to download or print the Syllabus for offline purposes.
2. Materials
   - All material to be studied is published like in PDF, video, or slides.
   - Organized Categorically (e.g., by subject, topic, or difficulty level).
3. Quizzes :
   - Practice quizzes.
   - Features might include multiple choice, timer, or adaptive difficulty type questions.
4. Roadmap for Exams :
   - A simple, step-by-step guide towards preparing for an exam.
   - It includes some milestones, dates, and access to resources in one place.

5. Progress (or Dashboard) :
   - A graphically illustrative view of user performance.
   - Scores, completion percentage, time elapsed, strengths, and weaknesses
6. Timer :
   - Quizzes, tests, and even other time-bound exercises
   - Low time alerts
7. Mock Tests
   - Mock versions of the entire test that would simulate the real exams.
   - Immediate feedback of results in terms of the score and areas of improvement.

Non-Functional Requirements :

1. Database Maintenance:
   - Database updated and optimized for performance.
   - Regular backups to prevent data loss.
   - Monitoring of database health in order to find and rectify deadlocks or slow queries.
2. Timer and Other Calculations :
   - Effective time management of quizzes, mock tests, and scheduling in real time.
   - Calculation of support for grading, tracking the users progress or statistics.
3. Display announcements :
   - This is the display of major updates or notices where it will definitely be very visible. The significant announcements also guarantee that they are displayed in any device and easily accessible.
4. Authentication :
   - This involves safe log-in mechanisms for authenticating the use.
   - Role-based access control, (e.g., admin, student) limits the features based on user roles
   - Safe storage and retrieval mechanisms for passwords.
5. Quizzes and Other Things Scheduling :
   - Users, either admins or teachers, can schedule quizzes or exams.
   - Notifications or reminders for scheduled events.
   - Rescheduling is possible.

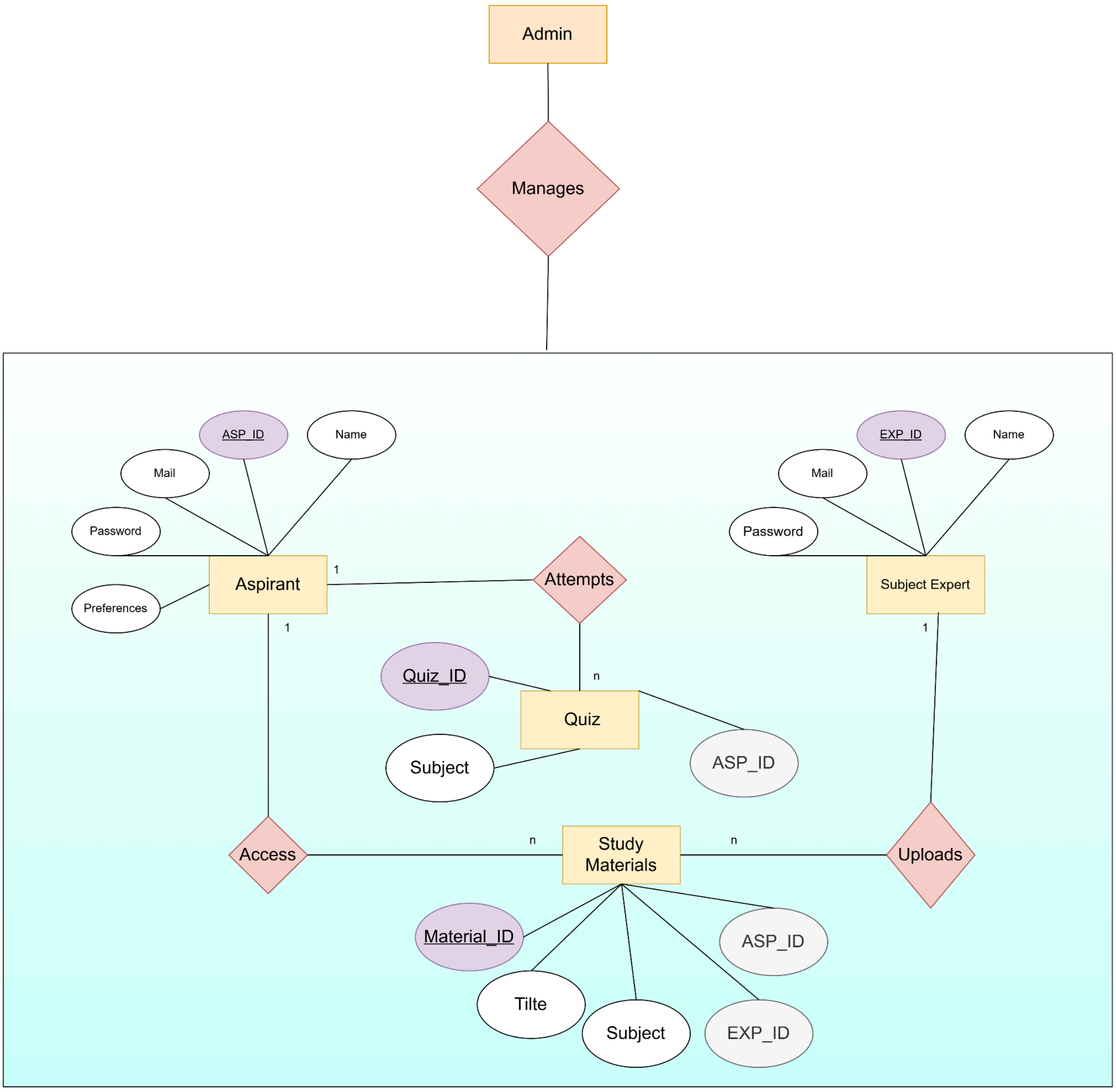
System Requirements :

1. Web Browser (Chrome) :
   - Normal running with compatibility of using other web browsers such as Google Chrome.
   - Use in other significant browsers (not necessary).
2. Internet Connection (at least 30 Mbps) :
   - A stable internet connection at 30 Mbps to run all materials, quizzes, and videos without glitches.
3. Hardware Device (PC or Smartphone or Laptop) :
   - Desktop, laptop, tablet, smartphone responsive on all screen sizes and resolutions.

User Requirements :

1. Language Support (Only English) :
    - It should be in English only for now.
    - Use of good and professional language.
2. Simple and Easy-to-Understand Interface :
    - This will be a user-friendly design for all the users.
    - Ease of navigation with a little learning curve.
3. Dark and Light Theme :
    - Change between the light mode, which will default, and the dark mode that will be very user-friendly during night time
    - Color schema customization
4. Implement as Website:
    - A website that sits on top of a browser so doesn't require an installation.

Entity-Relationship Diagram :



REQUIREMENT DOCUMENTATION

DEFINING USE CASES AND SCENARIOS :

1. Use Case: User Login

Actors: Aspirant (primary), System
Goal: Allow aspirants to securely log in to their account.

Preconditions:

✔ The user must have a valid registered account.
✔ The system should store encrypted credentials securely.

Main Flow:

1. The aspirant enters their email and password.
2. The system validates credentials from the database.
3. If valid, the user is redirected to their dashboard.

Alternate Flow:

- If credentials are invalid, the system displays an error message.

Postconditions:

✔ The user is logged in and can access study plans, quizzes, and progress reports.

2. Use Case: Generate Personalized Study Plan

Actors: Aspirant (primary), System
Goal: Provide a customized study plan based on the aspirant's target exam date and weak areas.

Preconditions:

✔ The aspirant must complete their profile with target exam date and subjects.

Main Flow:

1. The user selects subjects and inputs exam date.
2. The system analyzes syllabus coverage, weak areas, and available time.
3. The system generates a personalized study plan.

Alternate Flow:

- If no weak areas are identified, the system generates a default study plan.

Postconditions:

✔ The aspirant receives a structured day-wise study plan.

3. Use Case: Attempt Mock Test

Actors: Aspirant (primary), System
Goal: Allow aspirants to practice with mock tests and evaluate their performance.

Preconditions:

✔ The user must be logged in.
✔ The system should have mock tests stored in the database.

Main Flow:

1. The user selects a mock test.
2. The system retrieves the test from the database.
3. The user answers questions.
4. The system automatically evaluates the test and provides a score and analysis.

Alternate Flow:

- If the test is interrupted, the system saves progress and allows the user to resume later.

Postconditions:

✔ The user receives feedback on strengths and weaknesses.

4. Use Case: Logout

Actors: Aspirant (primary), Admin (primary), System
Goal: Allow users to log out securely from their accounts.

Preconditions:

✔ The user must be logged in.

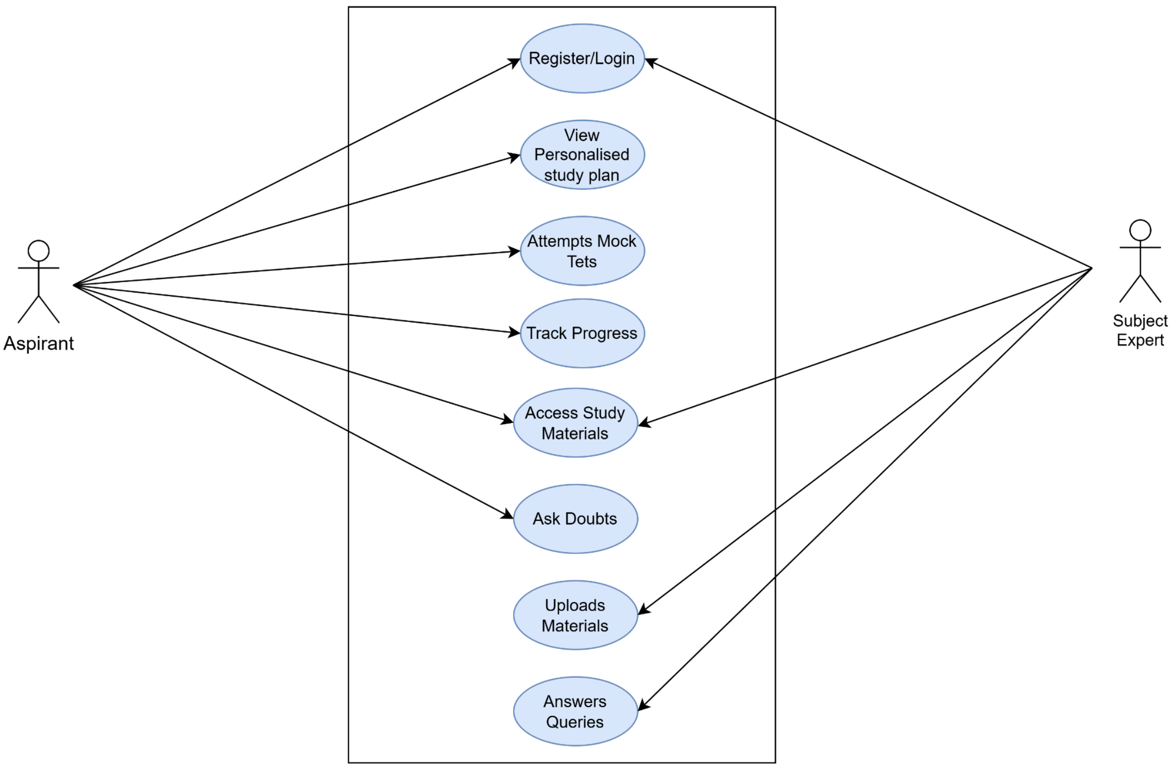
Main Flow:

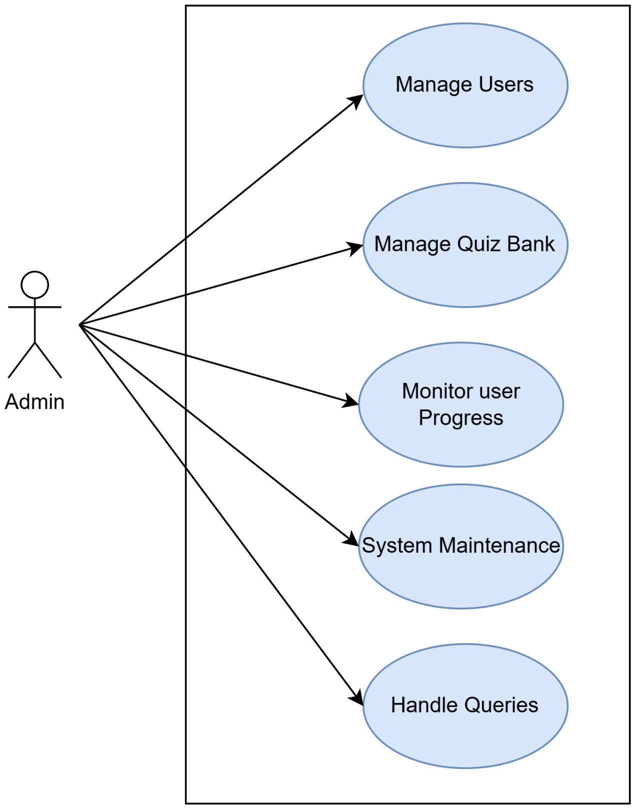
1. The user clicks "Logout".
2. The system clears session data.
3. The user is redirected to the login page.

Postconditions:

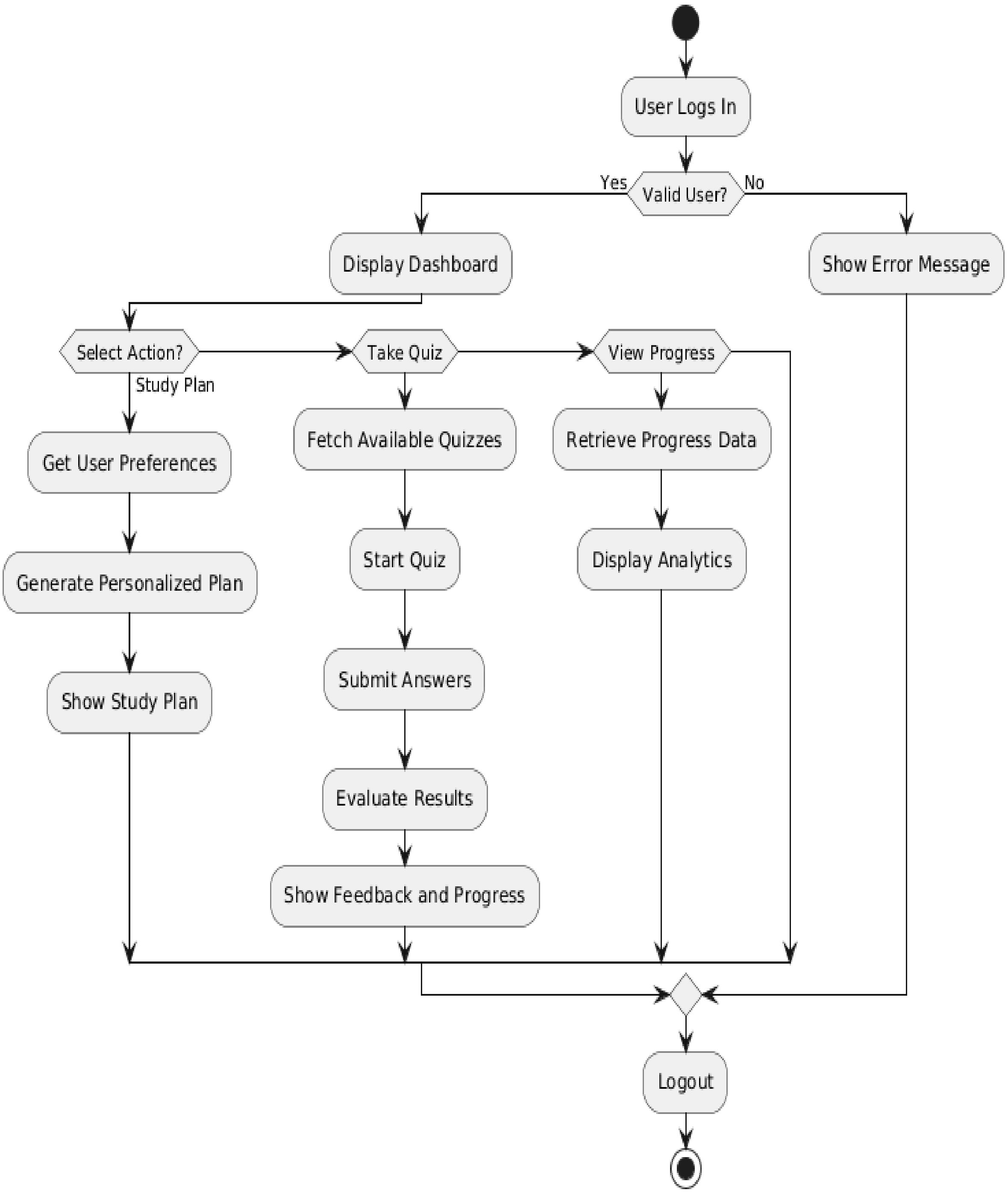
✔ The user session is terminated securely.

USE CASE DIAGRAMS :

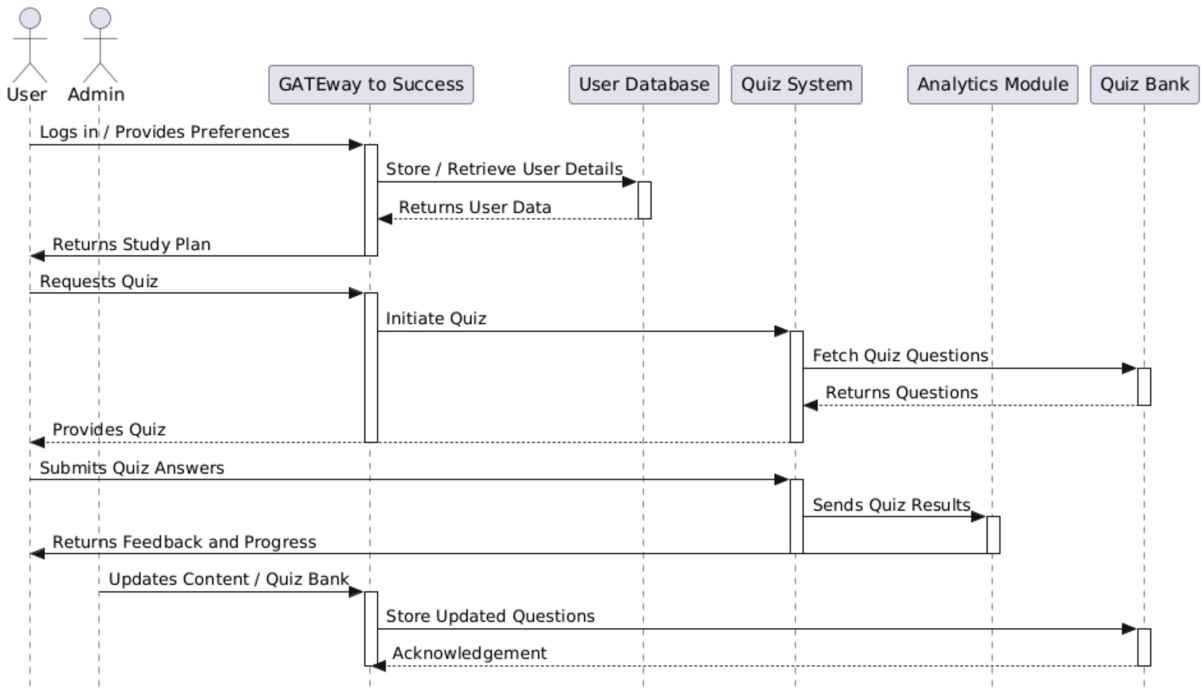




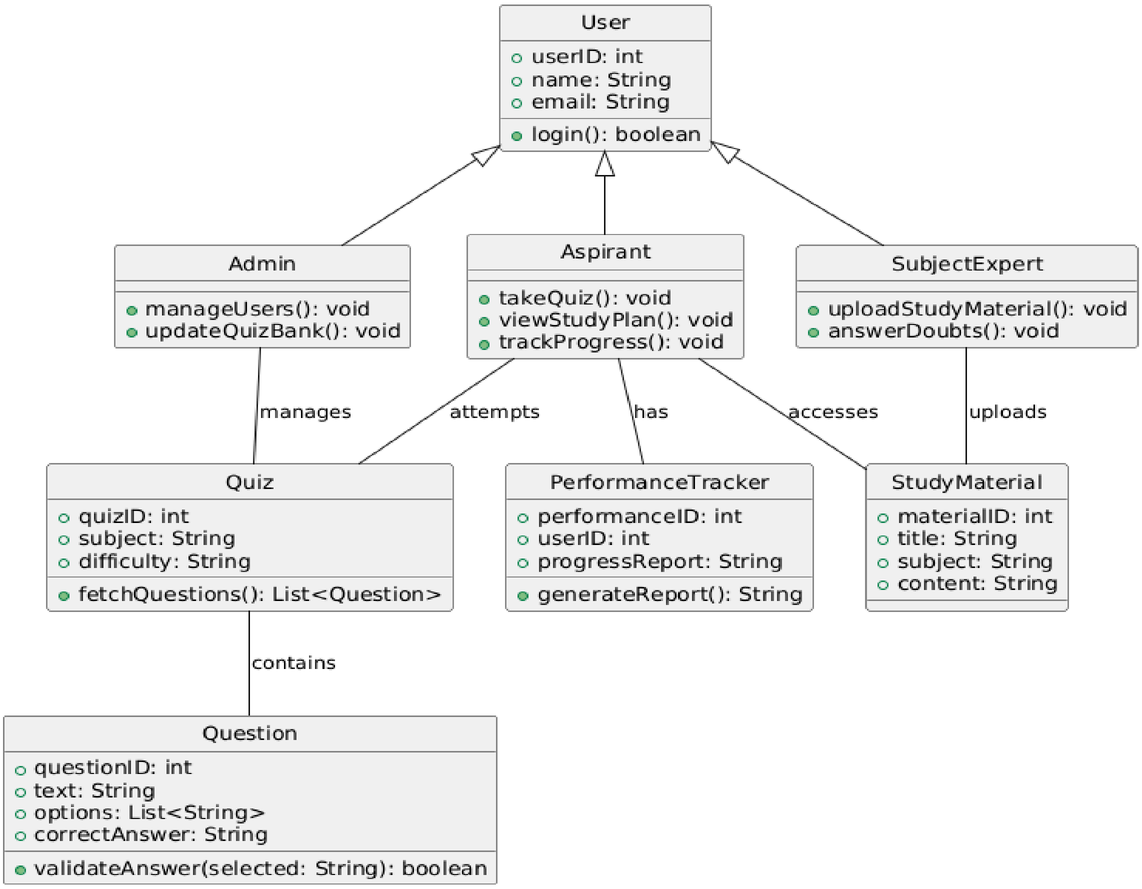
ACTIVITY DIAGRAM :



SEQUENCE DIAGRAM :



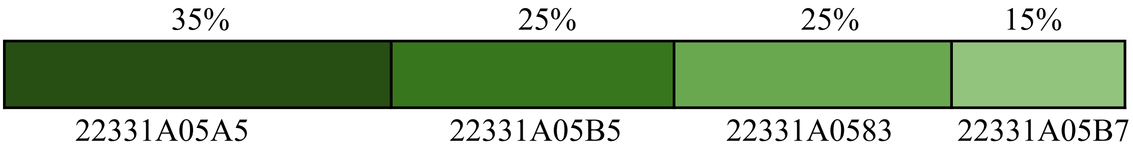
CLASS DIAGRAM :



Presented By :

MUDUNOORI SRI CHAITANYA VARMA	22331A05A5
NEKKALA GOWTHAM	22331A05B5
KOTTANA HEMANTH	22331A0583
NELLURI ENOSH RAJU	22331A05B7

Gantt Chart representing involvement of team mates :


35%
25%
25%
15%
22331A05A5
22331A05B5
22331A0583
22331A05B7