FIRST SEMESTER ACADEMIC YEAR 2024-2025

BACHELOR OR SCIENCE IN INFORMATION TECHNOLOGY

ASTREON: AI Study Assistant and Personalized Learning Support

A Software Engineering 1 Project
Presented to

Jerian R. Peren, LPT, PhD
Faculty Member

College of Engineering, Architecture and Computer Studies
Lyceum of the Philippines University Cavite

In Partial Fulfillment
Of the Subject Requirement in
CSCN10C-SOFTWARE ENGINEERING 1

By

Alhambra, Berna Marie
Bertoso, Hans Kyle
Sereno, Micah
Asinas, Alexander

Table of Contents

1 EXECUTIVE SUMMARY

In today's fast-paced digital learning environment, students face a myriad of challenges in managing their study schedules, retaining critical information, and receiving timely feedback. These hurdles can significantly impact their academic performance and overall learning experience. With this in mind, we set out to create a solution that addresses these struggles head-on and supports students in their journey to academic success.

Thus, *Astreon* was born – a comprehensive learning platform designed to help students overcome their challenges and reach their full potential.

History and Background

As online education continues to grow in popularity, the need for personalized, effective, and accessible learning tools becomes even more pressing. Students often find themselves overwhelmed by the sheer volume of material they must study, coupled with a lack of interactive support when they encounter difficult concepts. Astreon was conceived as a response to these pain points, aiming to provide a more tailored and engaging learning experience.

We identified that the most successful students are often those who can access the right resources at the right time, while also being able to choose study modes that match their individual learning styles. Additionally, students need real-time support, especially when they encounter difficult concepts or questions that traditional study resources can't answer. Astreon was designed to be the bridge that connects these needs.

Proposed Solution

Astreon is an AI-powered learning platform that offers students a range of study modes tailored to different learning styles. Whether a student prefers visual learning, active recall, or concept mapping, Astreon has the tools to support it. Through a combination of personalized study plans, AI-driven content delivery, and instant feedback, Astreon helps students manage their study schedules and understand complex topics.

The platform’s unique feature is its interactive AI-powered chatbox, which allows students to ask questions and get instant, detailed answers to even the most challenging queries. This real-time support aims to prevent frustration and help students move forward in their studies with confidence.

Project Vision

Our vision for Astreon is to create a platform that fosters academic excellence by providing students with the resources, tools, and support they need to thrive. Astreon will become a personalized learning assistant that adapts to each student’s preferences and learning style, helping them succeed not only in individual subjects but also in developing effective study habits for life.

Project Goals and Deliverables

1. **Personalized Study Modes**: Develop multiple study modes, including flashcards, quizzes, video explanations, and interactive simulations, to cater to various learning preferences.
2. **AI-Powered Assistance**: Build a robust AI system capable of answering a wide range of questions with accurate, context-aware responses, aiding students with real-time support.
3. **Study Schedule Management**: Implement a smart scheduling feature that helps students organize their study time efficiently and track their progress.
4. **Instant Feedback and Analytics**: Provide immediate feedback on quizzes, assignments, and study exercises, along with analytics to help students understand their strengths and weaknesses.
5. **Content Delivery**: Curate high-quality educational content across different subjects, ensuring it's up-to-date, engaging, and accessible.

Timeframe

- **Phase 1 (Concept Development & Design)**: 3 months
    - User research and defining key study modes
    - Initial design of the platform interface and user experience
    - Development of AI-powered chat functionality
- **Phase 2 (Platform Development & Testing)**: 6 months
    - Building the core platform with personalized learning features
    - Testing and refining AI responses
    - Early beta testing with real students for feedback
- **Phase 3 (Launch & Expansion)**: on-progress
    - Final refinements and optimizations based on user feedback
    - Public launch and marketing
    - Ongoing updates and content addition

Resources and Budget

- **Development Team**: A diverse team of software developers, AI specialists, UX/UI designers, and subject matter experts to ensure high-quality content and seamless platform functionality.
- **Budget Estimate**: PHP 1680
    - **Development**: PHP 600
    - **Tools and licensing**: PHP 300
    - **Marketing and Outreach**: 200
    - **Miscellaneous Costs**: 580

Success Criteria

1. **User Engagement**: Achieve a high level of engagement, with students actively using the platform for at least 15-20 hours per month.
2. **Improvement in Academic Performance**: Measure the academic improvement of students using Astreon through pre- and post-assessments, aiming for a 20-30% improvement in grades over a 3-month period.
3. **User Satisfaction**: Achieve a user satisfaction rate of 85% or higher in post-use surveys, with students rating the platform as effective, user-friendly, and supportive in their learning journey.
4. **Scalability**: Ensure that the platform can accommodate thousands of users simultaneously without any performance degradation, with the potential for expansion into other languages and educational systems.

Conclusion

With *Astreon*, we aim to revolutionize the way students approach learning by offering a platform that is both adaptable to their needs and responsive to their challenges. Through a combination of personalized study modes, AI-driven support, and real-time feedback, Astreon will empower students to reach their academic goals and, ultimately, their full potential.

1.1 SYSTEM OBJECTIVES

- **Personalized Learning Experience**
  - **Objective**: To offer a highly adaptable learning platform that tailors content, study methods, and schedules to individual students' needs, preferences, and learning styles.
  - **Key Features**:
    1. Customizable study modes (e.g., flashcards, quizzes, concept mapping).
    2. Adaptive learning algorithms that adjust difficulty levels based on student progress.
    3. Personalized study schedules and reminders to help students stay on track.
- **AI-Driven Question Support**
  - **Objective**: To integrate an AI-powered chatbox that can provide accurate, real-time assistance for student queries, helping them solve problems and understand complex concepts.
  - **Key Features**:
    1. AI chatbot capable of answering questions across multiple subjects.
    2. Contextual understanding for better, more relevant answers.

3. Continuous learning for the AI system to improve over time based on interactions.

- **Efficient Study Schedule Management**
  - **Objective**: To help students efficiently manage their study time with a smart scheduling feature that maximizes their productivity and ensures balanced learning across all subjects.
  - **Key Features**:
    1. A dynamic study planner that automatically adjusts based on assignments, exams, and personal study preferences.
    2. Reminders and notifications to keep students on track with their study goals.
    3. Integration with students' academic calendars for a streamlined planning experience.
- **Interactive Learning Tools**
  - **Objective**: To provide engaging, interactive learning tools that make studying more enjoyable and effective.
  - **Key Features**:
    1. Interactive quizzes, flashcards, and practice exams for active recall.
    2. Visual learning aids like diagrams, videos, and concept maps for better understanding.
    3. Gamification elements such as points, levels, and badges to motivate continuous learning.
- **Real-Time Feedback and Progress Tracking**
  - **Objective**: To offer immediate feedback on students' performance, allowing them to identify areas of improvement and track their learning progress over time.
  - **Key Features**:
    1. Instant feedback on quizzes and assignments to reinforce learning.
    2. Detailed performance analytics that highlight strengths and areas needing attention.
    3. Goal-setting tools that allow students to set targets and monitor their academic growth.
- **Scalable and Robust Platform**
  - **Objective**: To develop a platform that can scale to support a large number of users without compromising performance or user experience.
  - **Key Features**:
    1. Cloud-based infrastructure to handle growing user demands and data storage.

2. Scalable backend systems that can accommodate new features, additional subjects, and more users.
3. Cross-device compatibility, ensuring students can use Astreon on desktops, tablets, and smartphones.

- **User-Friendly Interface**
  - **Objective**: To ensure the platform is easy to navigate and intuitive for users of all tech proficiency levels, especially students and educators.
  - **Key Features**:
    1. Simple, clean, and visually appealing design that reduces cognitive load.
    2. Clear navigation paths for accessing study materials, schedules, and tools.
    3. Accessibility features such as text-to-speech, font adjustments, and color contrast for diverse user needs.
- **Collaborative Learning Opportunities**
  - **Objective**: To encourage peer learning by incorporating social and collaborative features where students can interact with each other, share knowledge, and solve problems together.
  - **Key Features**:
    1. Group study rooms for collaborative work on assignments or projects.
    2. Peer-to-peer tutoring features where students can assist each other.
    3. Discussion forums and communities to facilitate knowledge sharing and problem-solving.
- **Content Quality and Relevance**
  - **Objective**: To ensure the educational content available on the platform is accurate, up-to-date, and aligned with current academic curricula.
  - **Key Features**:
    1. Regular updates to the content to reflect new developments in various fields.
    2. Close collaboration with subject matter experts and educators to create or curate high-quality resources.
    3. Option for students to provide feedback on content quality to continuously improve materials.
- **Data Security and Privacy**
  - **Objective**: To prioritize the security and privacy of student data, ensuring the platform complies with local data protection laws (e.g., the Data Privacy Act of 2012 in the Philippines).

- **Key Features**:
    1. End-to-end encryption for sensitive data like personal information and study records.
    2. User authentication features such as two-factor authentication (2FA) for secure access.
    3. Transparent privacy policy and data protection practices to build trust among users.

1.2 BACKGROUND

Online education and digital learning tools are crucial for students' academic growth, but they often pose challenges in managing schedules, retaining information, and receiving feedback. These issues lead to frustration, disengagement, and decreased performance. Existing digital tools offer static content with limited personalization, and while some use AI, they lack real-time interaction or adapt to different learning styles.

The market gap presents a significant opportunity for a comprehensive platform that adapts to individual student needs, addressing the growing demand for personalized education experiences and effective time management tools.

Astreon is an adaptive and interactive platform designed to help students manage their academic journey with personalized learning paths, AI-driven assistance, and intuitive study tools. It provides immediate feedback, better study schedule management, and access to educational content in personalized formats.

Business Need and Stakeholder Objectives

Project Stakeholders:

• **Students**: Primary end-users needing personalized, interactive, efficient platforms.

• **Educators**: Teachers seeking tools to enhance learning experience and engagement.

• **Educational Institutions:** Schools, colleges, universities implementing innovative solutions.

Stakeholders aim to enhance academic outcomes, streamline learning, and meet demand for digital education tools.

Astreon Project Objectives:

• **Enhance Academic Performance**: Provide personalized study tools and real-time feedback for better understanding and higher grades.

• **Increase Engagement and Retention:** Cater to different learning styles and provide interactive, gamified learning experiences.

• **Enhance Educator Effectiveness**: Provide insights into student progress and challenges for targeted support and interventions.

• **Offer Scalable and Flexible Solutions**: Designed for accessibility to various educational institutions.

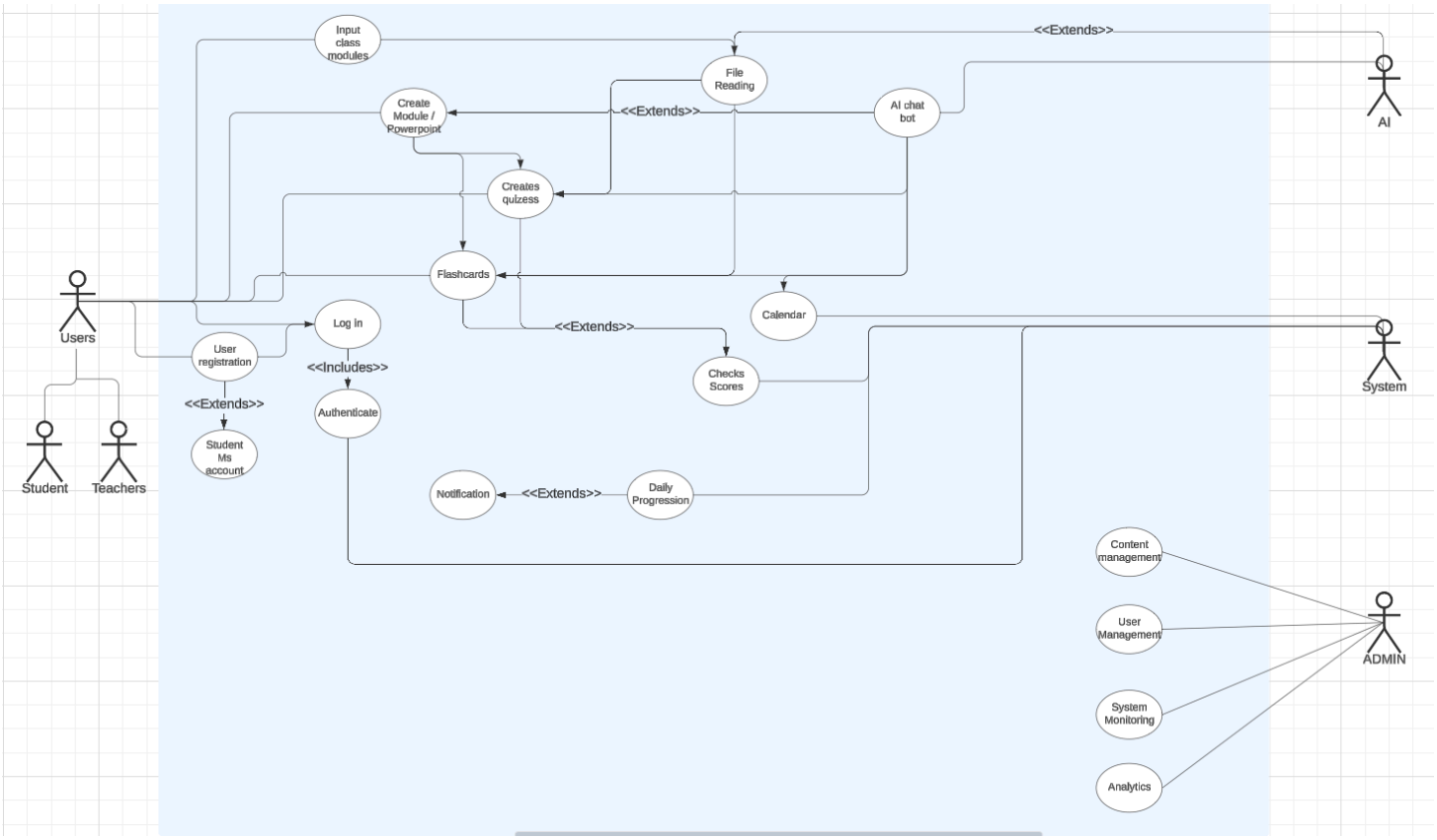
• **Support Lifelong Learning:** Foster self-directed learning and provide necessary tools for independent success.

1.2.1 Existing System Overview

The current version of *Astreon* offers a limited set of features designed to support students' learning needs, primarily through three study modes: flashcards, quizzes, and a chat box. These tools allow students to engage in active recall using flashcards, test their knowledge through quizzes, and receive AI-driven assistance for questions and challenges they may encounter during their studies via the chat box feature.

Additionally, users can upload various educational materials, such as PDF documents, PowerPoint presentations, and images, which can be incorporated into their study sessions. This feature allows students to study from a broader range of materials, making it easier to customize their learning experience with resources they already have.

While these features provide a basic yet functional structure for learning, the current system is still in the early stages of development. It lacks advanced capabilities like personalized learning paths, integrated study schedules, and real-time, interactive feedback mechanisms. Furthermore, the system does not yet offer a fully adaptive learning experience or extensive collaboration tools, which limits its potential to cater to diverse learning styles and provide more engaging, dynamic educational opportunities for students.



-Draw the existing system architecture or use case of the existing system

1.2.2 Existing System Business Process

The current business process revolves around educators and students using fragmented tools to facilitate learning and manage academic progression. Students and teachers rely heavily on traditional methods such as physical flashcards, printed quizzes, or standalone software for learning and evaluation. Scheduling study sessions, creating personalized learning schedules, and monitoring progress are largely manual processes, requiring significant effort from students. Teachers independently create and distribute materials like modules and quizzes, often without an integrated system for seamless sharing or tracking progress. This workflow lacks synchronization, collaboration, and centralization, which often leads to inefficiencies and difficulties in adapting to individual needs.

1.2.3 Existing System Problems and Issues

The current system is fraught with several inefficiencies and challenges. Firstly, the manual creation and distribution of learning materials can be time-consuming and prone to errors. Additionally, the absence of a centralized platform means that both students and teachers lack effective tools to monitor progress or receive actionable feedback. Students often struggle to stay consistent with their study plans due to a lack of tailored schedules, while teachers are unable to track individual student progress comprehensively. Furthermore, there is minimal automation to assist with repetitive tasks like generating quizzes or managing content, which limits productivity. These shortcomings result in reduced engagement, limited scalability, and an overall suboptimal learning experience.

1.3 PROPOSED SYSTEM

The proposed **Astreon Study Buddy** system is a comprehensive digital learning platform designed to streamline and enhance the educational experience for students and teachers. The system will include key functionalities such as the creation of flashcards and quizzes, automated progress tracking, and personalized study schedules tailored to individual needs. Teachers can upload learning modules, create quizzes, and monitor student performance with ease, saving time and increasing efficiency.

For students, the system provides adaptive tools to track daily progress, check scores, and receive feedback on their learning journey. The integrated AI chatbot offers real-time assistance and recommendations, making learning more engaging and accessible. Additionally, the system includes a calendar feature to help students manage their study schedules and deadlines effectively.

Administrators will benefit from tools for user management, content oversight, and system analytics, ensuring a secure and optimized experience for all users. By centralizing resources, automating repetitive tasks, and offering actionable insights, Astreon Study Buddy delivers significant value by improving productivity, engagement, and academic outcomes.

1.3.1 System Overview

The proposed *Astreon Study Buddy system* will enable students and teachers to interact with a centralized platform that simplifies the learning process. Users can log in securely, with students having the option to sign in using their Microsoft accounts. Teachers will be able to upload modules and PowerPoint presentations, which the system can process to generate flashcards and quizzes automatically. Students can access these learning materials, track their progress through a personalized dashboard, and receive tailored study schedules that adapt to their available time and goals.

An AI-powered chatbot will be integrated into the system, offering real-time assistance to answer questions and provide study guidance. A built-in calendar will help students organize their study sessions, and automated notifications will remind them of tasks and deadlines. Teachers can monitor student progress and performance through the platform, ensuring a more personalized learning experience. Additionally, administrators will have access to tools for user management, system monitoring, and generating analytics to enhance the overall functionality and reliability of the system. This solution aims to centralize and optimize the educational process, making it efficient and engaging for all users.

High level system overview:

Subsystem	Function
Authentication System	- User Registration - Log In/Out - Integration with Microsoft Accounts
Content Management	- Upload Modules and Presentations - Generate Flashcards - Create Quizzes
Study Progress Tracker	- Monitor Daily Progress - Check Scores - Generate Progress Reports
Scheduling System	- Create Study Schedules - Calendar Integration - Automated Notifications
AI Assistance	- AI Chatbot for Queries - Study Guidance - Learning Recommendations
Admin Dashboard	- User Management - System Monitoring - Analytics Reporting

1.3.2 Current System Hardware, Software and Network

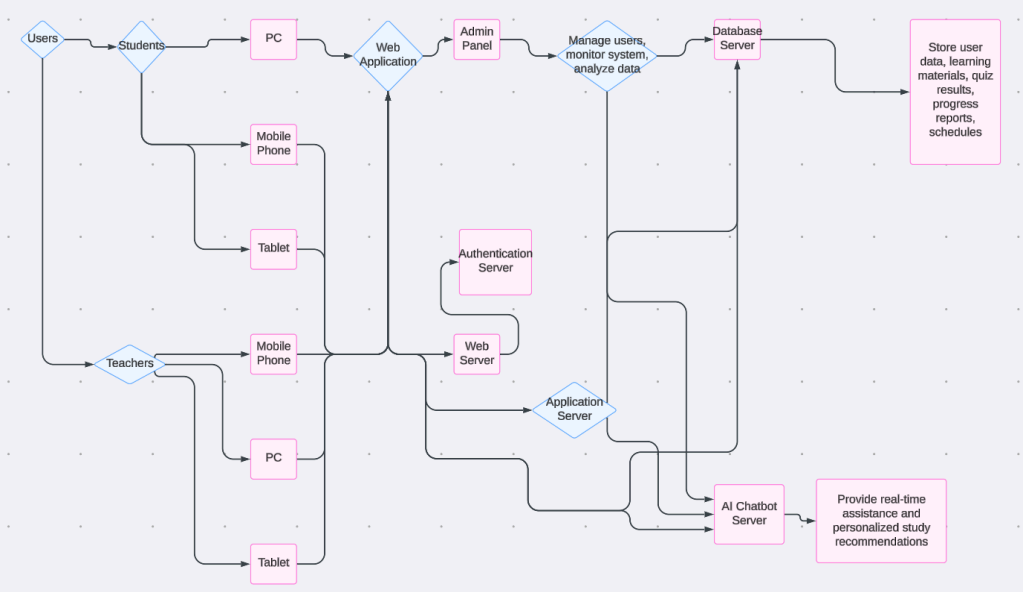
<Describe the current system's hardware and software configuration.>

System hardware:

Hardware	***Application Server***
Server Name	***APP_SVR_1***
CPU	***AMD Ryzen 5 5600H (4.2MHz Boost, 16MB L3 Cache)***
Memory	***8GB SO-DIMM DDR4-3200MHz***
Hard Disk Storage	***512GB NVMe PCIe 3.0 M.2 SSD***

System software:

Software	***Application Server***
Operating System	***Windows Server 2012 R2***
Application Server Software	***IBM WebSphere Application Server 8.5***
Backup Software	***CA Arcserve Backup V16***



Sample:

1.3.3 Volumes and Frequencies

<Summarise the data volumes and frequencies of processes of the current system.>

Data volume and frequencies:

Subsystem	Description	Volume and Frequencies
Authentication System	User logins and registrations	Approximately 1,500 logins per day
Content Management	Uploading modules, creating flashcards and quizzes	Around 500 new content uploads weekly
Progress Tracker	Monitoring daily study activities and scores	2,000 progress updates logged per day
Notification System	Sending reminders and updates	5,000 notifications sent daily

1.3.4 Interface with Other Systems

Interface	External Parties	Description
Microsoft Account API	*Microsoft*	*1. User authentication for secure login.* *2. Integration of user accounts with existing Microsoft services.*
Email Notification System	*Third-party email services*	*1. Sending automated reminders and notifications to users.*
Cloud Storage API	*Google Drive/OneDrive*	*1. Uploading and sharing learning materials.* *2. Storing study schedules and progress reports.*

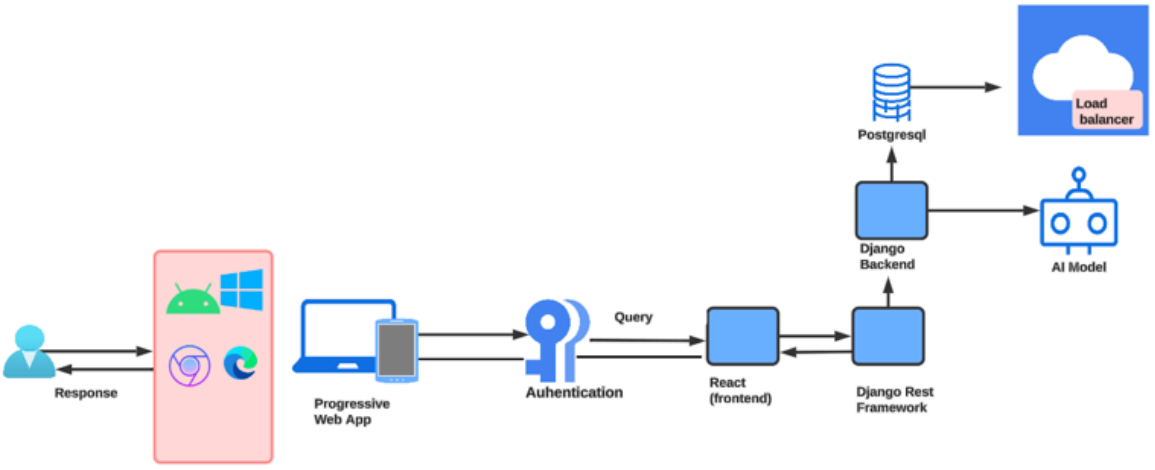
<Describe any current system interfaces.>

System interfaces:

1.3.5 System Functions

Item	System Functions
1	***Common Requirements for Astreon Study Buddy***
1.1	***General functional requirements for user registration, authentication, and secure login.***
1.2	***Integration with Microsoft accounts for seamless access.***
2	***Content Management System***
2.1	***Upload and processing of learning modules and PowerPoint presentations.***
2.2	***Automatic generation of flashcards and quizzes from uploaded content.***
2.3	***Organization and categorization of study materials for ease of access***
3	***Study Progress Tracking***
3.1	***Monitoring of individual learning progress and scores.***
3.2	***Display of daily progression reports for users.***
4	***Scheduling and Notification System***
4.1	***Creation of personalized study schedules based on user input.***
4.2	***Automated reminders and notifications for upcoming tasks and deadlines.***
5	***AI Assistance***
5.1	***AI chatbot for answering study-related queries and providing recommendations.***
5.2	***Personalized study guidance and tips based on user performance.***
6	***Administrative Functions***
6.1	***User and content management by administrators.***
6.2	***System monitoring and analytics for performance evaluation.***

1.3.6 Technical System Architecture



The **Astreon Study Buddy** system is built to operate within a Progressive Web Application (PWA) framework, ensuring accessibility across multiple platforms, including Android, Windows, and major web browsers like Chrome and Edge. The system begins with user interaction through the PWA, providing students and teachers a seamless experience to access features such as quizzes, flashcards, and progress tracking. To ensure security, users are authenticated through a dedicated **Authentication Service** before accessing the application's functionalities. Once authenticated, the **React frontend** serves as the main interface, sending user queries to the backend and dynamically rendering responses like quiz content, schedules, and feedback.

At its core, the **Django Rest Framework (DRF)** handles all incoming queries from the frontend, facilitating communication with the **Django backend**, which processes the application logic. The backend also integrates with an **AI Model** to generate dynamic and personalized content, such as customized quizzes, progress feedback, and study schedules based on user data. All relevant information, including user details, quiz results, and schedules, is stored in a **PostgreSQL database**, ensuring data consistency and reliability. A **Load Balancer** is deployed to manage database traffic, optimize performance, and ensure system scalability during high demand.

Once the backend processes the request, the responses are delivered back through the Django Rest Framework to the React frontend, where they are displayed to users in real-time. This architecture provides a robust, secure, and scalable system for Astreon Study Buddy, delivering an interactive and efficient learning experience for students and teachers alike.

1.4 ECONOMIC FEASIBILITY

1.5.1 Costs

- **Marketing and Outreach**: 200
- **Miscellaneous Costs**: 580

Development Cost	Cost ($)
Development	***600***
Tools and licensing	***300***
Marketing and outreach	***200***
Miscellaneous costs	***580***
Total	***1680***

1.5.2 Benefits

Tangible Benefits

The system is contributed to achieving the following benefits:

- *Saves time by automating study material creation*
- *improves learning outcomes with tailored study schedules and resources*
- *Supports both students and educators in achieving their goals*
- *provide flexibility in managing and adjusting study plans.*

Intangible Benefits

ASTREON includes an enhanced user experience through AI-powered features that provide personalized and efficient study tools, fostering greater user engagement and satisfaction. These features contribute to brand recognition and reputation in the education technology sector, helping establish a strong identity and positive word-of-mouth among users and educational professionals. The project also builds user trust and loyalty by offering reliable features like secure authentication and personalized content, leading to long-term retention and recommendations. Additionally, Astreon positively impacts education by integrating AI to improve students' learning experiences, while offering scalability for future enhancements. Finally, the development process facilitates knowledge transfer, boosting the team's expertise in cutting-edge technologies and preparing them for future innovations.

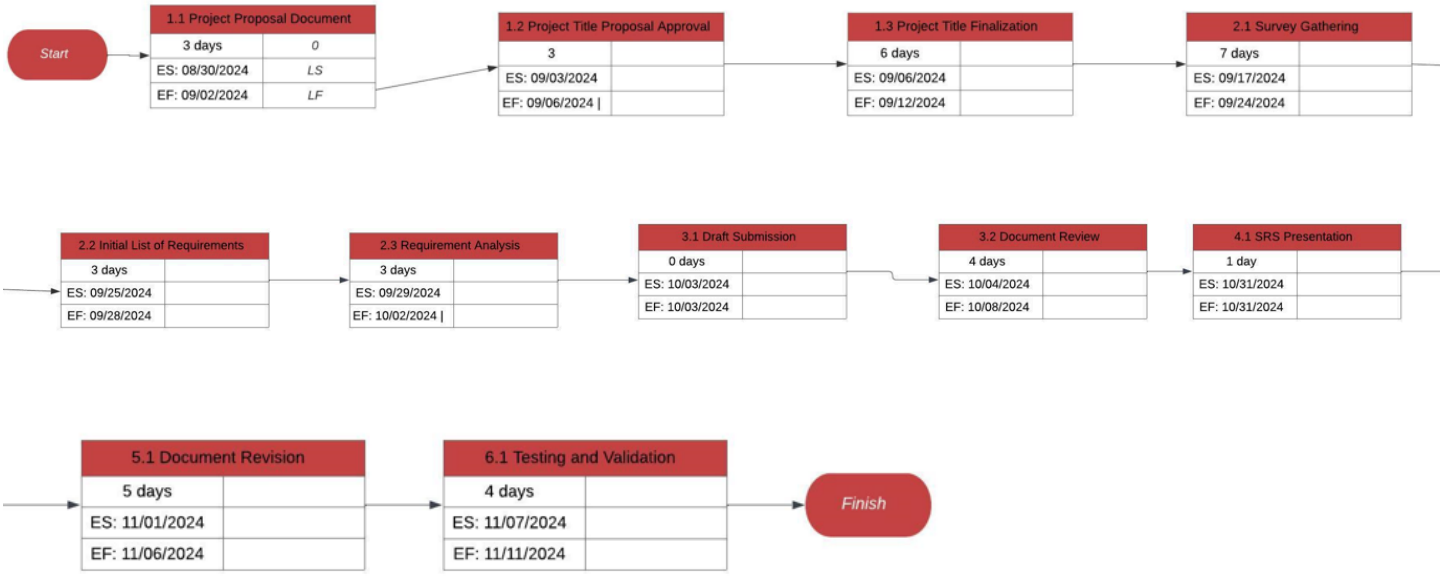
1.5.3 PROJECT COST AND BENEFIT ANALYSIS

Year	annual benefits	cumulative benefits	discount factor	present value
1	5000	5000	0.89	4464.285714
1	5000	10000	0.80	3985.969388
1	5000	15000	0.71	3558.901239
1	5000	20000	0.64	3177.590392
1	5000	25000	0.57	2837.134279

Total Present Value	18,024
Total Costs	1,680
Net Present Value	16334
Cost-benefit Ratio	2.98
Payback period	0.34
ROI	1388.10%

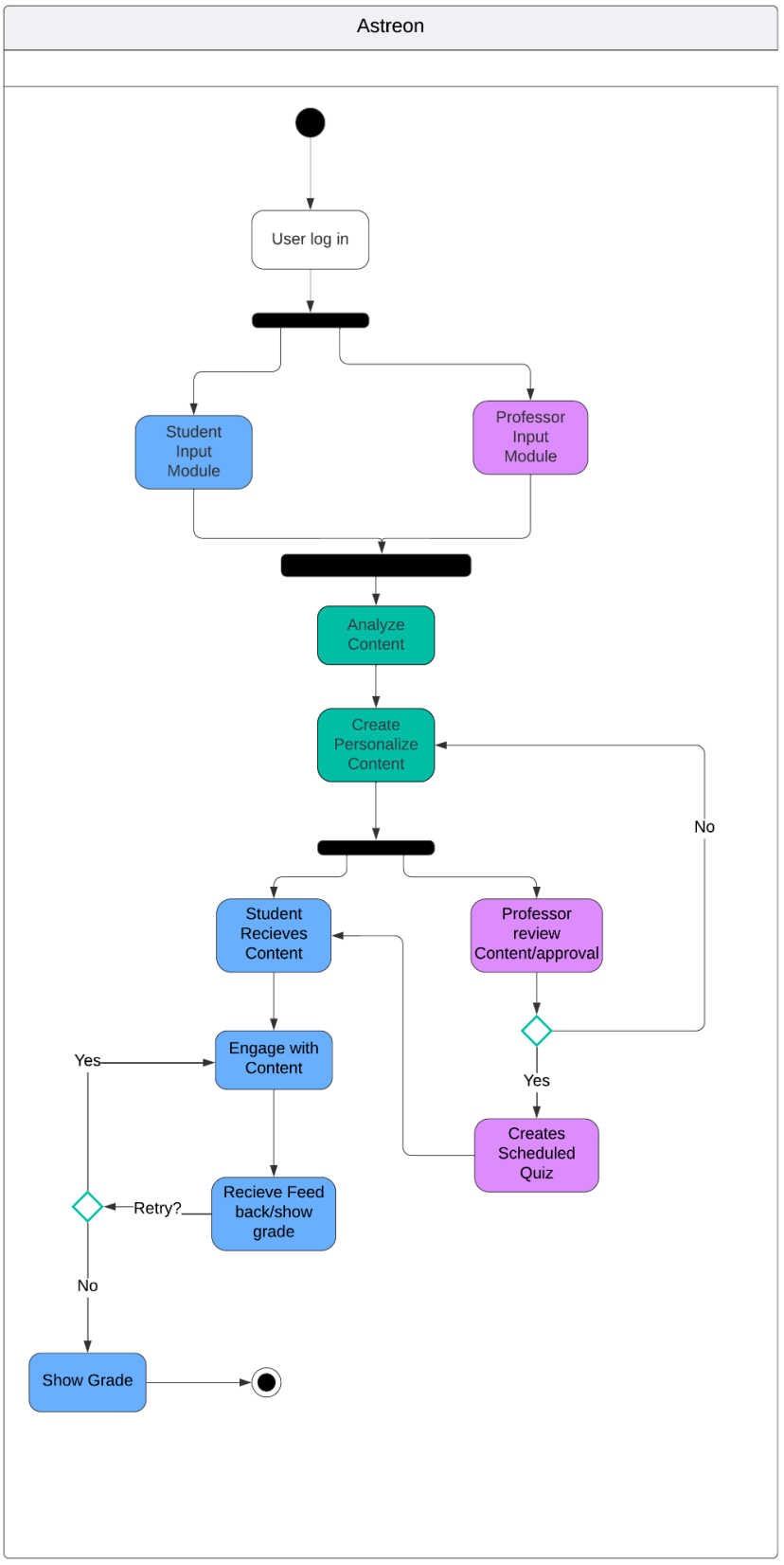
1.5 IMPLEMENTATION PLAN

Items	Activity	Start Date	End Date	Duration (Days)	Dependencies
1	***Survey Gathering***	*08/30/2024*	*09/02/2024*	*3*	*-*
1.2	***Initial List of Requirements***	*09/03/2024*	*09/06/2024*	*3*	*1.1*
1.3	***Requirement Analysis***	*09/06/2024*	*09/12/2024*	*6*	*1.2*
2.1	***Wireframing***	*09/12/2024*	*09/15/2024*	*3*	*1.3*
3.1	***API Development***	*09/15/2024*	*09/22/2024*	*7*	*1.3*
3.3	***Backend AUthentication***	*09/22/2024*	*09/25/2024*	*3*	*3.1*
5.3	***Frontend-Backend Integration***	*09/26/2024*	*09/30/2024*	*4*	*3.1*
6.2	***Integration Testing***	*09/30/2024*	*10/04/2024*	*4*	*5.3*
7.2	***Beta Deployment***	*10/10/2024*	*10/14/2024*	*4*	*5.3*
7.3	***Final Deployment***	*10/15/2024*	*10/19/2024*	*4*	*7.2*



1.3.7 Current System Overview

1.6 CURRENT BUSINESS PROCESS



1.7 CURRENT PROBLEMS AND ISSUES

1. Data Management Constraints

Storage and management of files, user data, and schedules can become increasingly complex as the system grows. Integrating and optimizing storage solutions (e.g., AWS) is a work in progress.

2. Limited AI Model Optimization

The AI model relies on pre-trained data without sufficient fine-tuning or optimization for specific use cases in education.

3. Lack of Personalization for Different Learning Styles

The current system does not dynamically adapt to the learning preferences of various users (e.g., visual learners, auditory learners, or kinesthetic learners).

4. Loading Time

Some of the features of astreon such as generate schedule, user prompts, quizzes, flashcards are not yet optimized so it takes quite a while to load the expected output

2 REQUIREMENTS SPECIFICATION

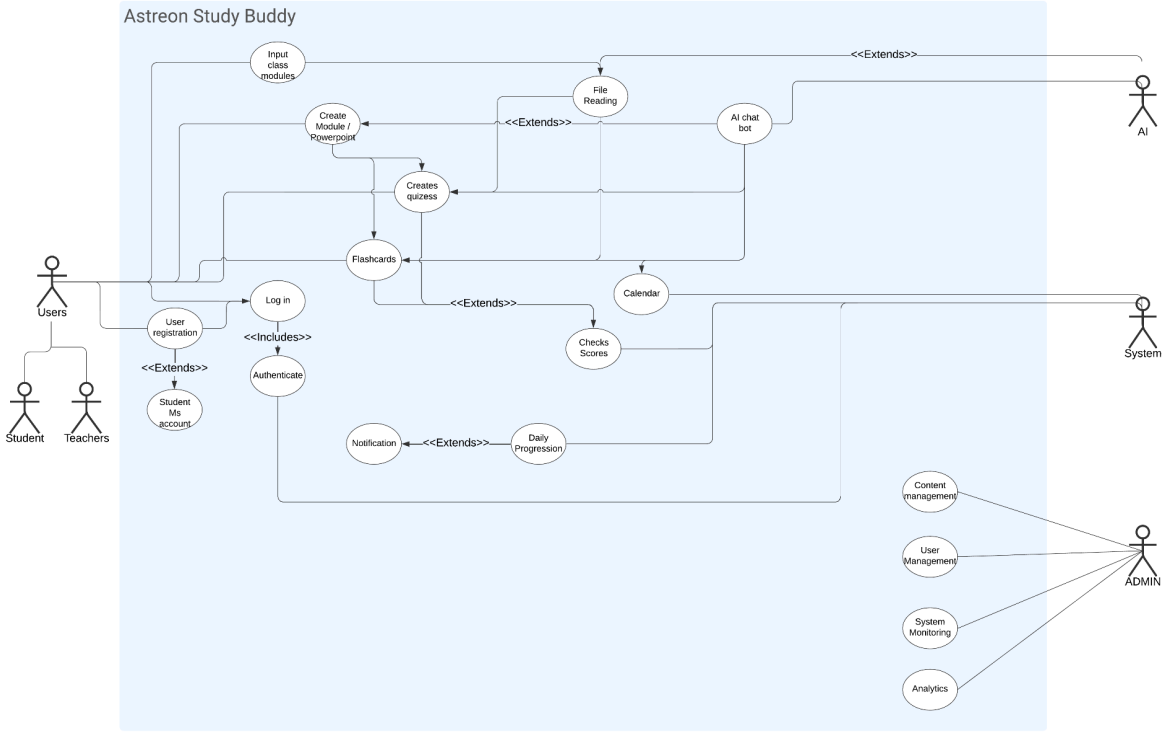
2.1 USER REQUIREMENTS DOCUMENT

2.1.1 Proposed System Overview

The proposed system, ASTREON: AI Study Assistant and Personalized Learning Support, is an advanced educational tool designed to streamline and enhance the learning experience for students and teachers. Utilizing artificial intelligence, ASTREON enables users to upload educational materials in various formats (PDF, DOCX, JPG, PNG) and generate personalized study aids such as quizzes, flashcards, notes, and study schedules.

The system features a user-friendly interface accessible through web and mobile browsers, allowing students and teachers to interact seamlessly with its tools. ASTREON also incorporates real-time progress tracking and AI-powered Q&A support to provide tailored recommendations and assistance.

2.1.2 Description of Proposed Astreon System



2.1.3 System User Profile

No.	User Role	Responsibilities	Branch/ Division/ Section/ Unit	Staff Post/Rank	Stakeholder Group
1	Student	Upload files, generate study aids, track progress, and access AI Q&A support.	Academic Users	Learner	End-users (Students)

2	Teacher	Provide educational materials, generate resources for students, and monitor student progress.	Faculty	Educator	End-users (Teachers)
3	Administrator	Manage user accounts, monitor system performance, and oversee analytics.	IT Operations	System Admin	System Maintainers

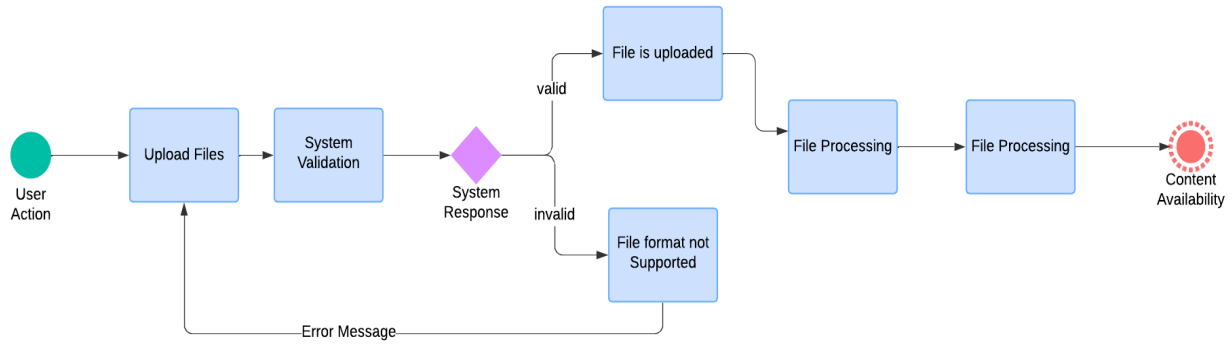
2.2 Future Business Process

2.2.1 List of Future Business Processes

Process ID	Business Process Title
BP-001	Upload Lesson Files
BP-002	Generate Quizzes
BP-003	Generate Flashcards
BP-004	Generate Notes
BP-005	Personalized Study Scheduling
BP-006	AI Question and Answer Assistance
BP-007	Track Learning Progress
BP-008	Manage User Accounts (Admin Function)

2.2.1.1 BP-001: Upload Lesson Files

The process enables students and teachers to upload lesson files in supported formats (e.g., PDF, DOCX, JPG, PNG). These files are processed by the system to extract relevant content, which can be used to generate study aids such as quizzes, flashcards, and notes.



BP-002: Generate Quizzes

The process allows users to generate quizzes based on uploaded lesson files. The system uses AI to analyze the file content and create customized quizzes.

Steps:

1. **User Request:** The user selects the uploaded file and initiates quiz generation.

2. **Quiz Parameters:** The system asks the user to define quiz parameters (e.g., number of questions).
3. **Quiz Generation:** The system generates the quiz and displays it to the user.
4. **Progress Tracking:** The quiz results are saved for tracking and analytics.

BP-005: Personalized Study Scheduling

This process allows students to create a personalized study schedule based on their availability and learning goals.

Steps:

1. **User Input:** The user provides their preferred study hours and days.
2. **Schedule Creation:** The system generates a schedule tailored to the user's input.
3. **Schedule Adjustment:** Users can modify the schedule anytime to fit their changing needs.

2.2.2 Functional Requirement

2.2.2.1 List of Functional Requirements

Req. ID	Requirement Title	Target Users	Priority
REQ-SYS-001	User Authentication	Students, Teachers, Admins	Must
REQ-UPL-001	Upload Lesson Files	Students, Teachers	Must
REQ-QUIZ-001	Generate Quizzes	Students	Must
REQ-FLASH-001	Generate Flashcards	Students	Must
REQ-NOTES-001	Generate Notes	Students	Must
REQ-AI-001	AI Question and Answer	Students	Must
REQ-PROG-001	Progress Tracking	Students	Should
REQ-SCHED-001	Personalized Study Schedule	Students	Should

2.2.2.2 REQ-SYS-001: User Authentication

Description	Frequency of Use	Acceptance Criteria
Secure login functionality for users (Students, Teachers, Admins).	Daily	• Login processing time must not exceed 3 seconds. • Incorrect login attempts return error messages, e.g., *"Invalid username or password."*

2.2.2.3 REQ-UPL-001: Upload Lesson Files

Description	Frequency of Use	Acceptance Criteria
Upload lesson files in formats like PDF, DOCX, JPG, and PNG for generating study aids.	Daily	• File upload supports up to 10MB. • System validates formats and sizes, providing clear error messages for invalid inputs.

2.2.2.4 QUIZ-001: Generate Quizzes

Description	Frequency of Use	Acceptance Criteria
Generate customizable quizzes from uploaded content.	Weekly	● Quiz generation completes within 20 seconds under normal load. ● Quizzes are aligned with parsed file content.

2.2.2.5 REQ-FLASH-001: Generate Flashcards

Description	Frequency of Use	Acceptance Criteria
Flashcards summarize key points from uploaded files.	Weekly	● Flashcards are accessible on mobile and web interfaces.

2.2.2.6 REQ-NOTES-001: Generate Notes

Description	Frequency of Use	Acceptance Criteria
Generate concise notes from uploaded content, including summaries or rewrites.	Weekly	● Notes generated are accurate and available for download.

2.2.2.7 REQ-AI-001: AI Question and Answer Assistance

Description	Frequency of Use	Acceptance Criteria
Users can ask subject-specific questions; AI answers based on referenced files.	Daily	● Response time does not exceed 10 seconds.

2.2.2.8 REQ-PROG-001: Progress Tracking

Description	Frequency of Use	Acceptance Criteria
Track progress, including quiz scores, study schedules, and flashcard usage.	Weekly	● Progress is visualized in graphs or dashboards.

2.2.2.9 REQ-SCHED-001: Personalized Study Schedule

Description	Frequency of Use	Acceptance Criteria
Create, customize, and adjust study schedules.	Bi-weekly	● Users receive reminders for upcoming sessions..

2.2.3 Non-functional Requirements

List of Non-functional Requirements

Req. ID	Category	Requirement Title	Target Users	Priority
REQ-PERF-001	Performance	Response Time	All Users	Must
REQ-PERF-002	Performance	System Throughput	All Users	Must
REQ-SEC-001	Security	Data Encryption	All Users	Must
REQ-SEC-002	Security	User Authentication	Admin, Students, Teachers	Must
REQ-BACK-001	Backup and Recovery	Data Backup and Recovery	Admin	Must
REQ-COMP-001	Compatibility	Cross-Browser Compatibility	All Users	Should
REQ-AVAIL-001	Availability	90% Uptime	All Users	Should
REQ-SCAL-001	Scalability	Support for 10 Concurrent Users	Admin, System	Must

2.2.3.1 REQ-PERF-001: Response Time

Description	Priority
The system responds to user actions within 20 seconds under normal load.	***Must***

2.2.3.2 REQ-PERF-002: System Throughput

Description	Priority
Support up to 10 concurrent users without noticeable performance degradation.	***Must***

2.2.3.3 REQ-SEC-001: Data Encryption

Description	Priority
Encrypt all data transmission using HTTPS and TLS protocols.	***Must***

2.2.3.4 REQ-SEC-002: User Authentication

Description	Priority
Multi-factor authentication (MFA) for admin-level users.	***Must***

2.2.3.5 REQ-BACK-001: Data Backup and Recovery

Description	Priority
Perform daily backups and ensure data restoration within 24 hours.	***Must***

2.2.3.6 REQ-COMP-001: Cross-Browser Compatibility

Description	Priority
Ensure consistent functionality across modern browsers like Chrome, Firefox, Edge, and Safari.	***Should***

2.2.3.7 REQ-AVAIL-001: 90% Uptime

Description	Priority
Maintain 90% availability for uninterrupted user access.	***Should***

2.2.3.8 REQ-SCAL-001: Scalability

Description	Priority
Support future scaling for up to 50 concurrent users.	***Must***

2.3 TECHNICAL REQUIREMENTS

2.3.1 List of Technical Requirements

Req. ID	Requirement Title	Priority	Category	Responsible Team Member(s)
TR-SBR-001	Server Housekeeping	Must	System Backup and Recovery	Systems Analyst(s) / Project Manager
TR-SBR-002	Backup and Recovery Procedures	Must	System Backup and Recovery	Systems Analyst(s) / IT Specialist
TR-SEC-001	Data Encryption Standards	Must	Privacy Requirements	Security Engineer / IT Specialist
TR-SEC-002	Multi-Factor Authentication	Should	Privacy Requirements	Security Engineer
TR-COMP-001	Cross-Browser Compatibility	Must	Interface Requirements	Frontend Developer
TR-SCAL-001	Scalability Requirements	Must	System Effectiveness and Scalability	Backend Developer
TR-INT-001	API Integration Standards	Must	Interface Requirements	Backend Developer / API Specialist
TR-PERF-001	Performance Monitoring	Should	Maintainability, Control, and Testing	Systems Analyst / QA Specialist
TR-DATA-001	Data Conversion Process	Must	Data Conversion	Database Administrator
TR-LOG-001	Logging and Diagnostics	Must	Maintainability and Control	DevOps Engineer
TR-TEST-001	Automated Testing Coverage	Must	Testing	QA Specialist / Developer
TR-UX-001	User Interface Accessibility	Should	User Experience	UX Designer / Frontend Developer

TR-SBR-001 Server House Keeping

Item	Description
Requirement ID	TR-SBR-001
Requirement title	Server Housekeeping
Priority	Must
Category	System Backup and Recovery
Technical requirement description	System logs must be archived weekly to backup storage to ensure data integrity and recoverability.

TR-SBR-002 Backup and Recovery Procedures

Item	Description
Requirement ID	TR-SBR-002
Requirement title	Backup and Recovery Procedures
Priority	Must
Category	System Backup and Recovery
Technical requirement description	Implement daily incremental backups and weekly full backups of the database and server files. Include off-site storage for critical data.
Responsible Team Member(s)	Systems Analyst(s) / IT Specialist

TR-SEC-001 Data Encryption Standards

Item	Description
Requirement ID	TR-SEC-001
Requirement title	Data Encryption Standards
Priority	Must
Category	Privacy Requirements
Technical requirement description	All data transmitted between clients and servers must be encrypted using HTTPS and TLS protocols. Data at rest must be encrypted using AES-256 standards.
Responsible Team Member(s)	Security Engineer / IT Specialist

TR-SEC-002 Multi-Factor Authentication

Item	Description
Requirement ID	TR-SEC-002
Requirement title	Backup and Recovery Procedures
Priority	Should
Category	Privacy Requirements
Technical requirement description	Enable MFA for admin accounts and optionally for regular users. Integration with email or app-based OTP (One-Time Password) systems required.
Responsible Team Member(s)	Security Engineer

TR-COMP-001 Cross-Browser Compatibility

Item	Description
Requirement ID	TR-COMP-001
Requirement title	Cross-Browser Compatibility
Priority	Must
Category	Interface Requirements
Technical requirement description	Ensure consistent performance and UI rendering across Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari. Perform cross-browser testing during QA.
Responsible Team Member(s)	Frontend Developer

TR-SCAL-001 Scalability Requirements

Item	Description
Requirement ID	TR-SCAL-001
Requirement title	Scalability Requirements
Priority	Must
Category	System Effectiveness and Scalability
Technical requirement description	System must support scaling for future demands, including handling up to 50 concurrent users in subsequent versions. Use a load balancer for traffic management.
Responsible Team Member(s)	Backend Developer

TR-INT-001 API Integration Standards

Item	Description
Requirement ID	TR-INT-001
Requirement title	API Integration Standards
Priority	Must
Category	Interface Requirements
Technical requirement description	Ensure RESTful APIs adhere to standard conventions (e.g., JSON responses, proper status codes). Document API endpoints for ease of integration.
Responsible Team Member(s)	Backend Developer / API Specialist

TR-PERF-001 Performance Monitoring

Item	Description
Requirement ID	TR-PERF-001
Requirement title	Performance Monitoring
Priority	Should
Category	Maintainability, Control, and Testing
Technical requirement description	Implement real-time performance monitoring using tools like Prometheus or New Relic. Provide alerts for server downtimes or unusual activity.
Responsible Team Member(s)	Systems Analyst / QA Specialist

TR-DATA-001 Data Conversion Process

Item	Description
Requirement ID	TR-DATA-001
Requirement title	Data Conversion Process
Priority	Must
Category	Data Conversion
Technical requirement description	Ensure proper migration of users and study data from the current database to the new PostgreSQL schema without data loss. Verify through automated testing.
Responsible Team Member(s)	Database Administrator

TR-TEST-001 Automated Testing Coverage

Item	Description
Requirement ID	TR-TEST-001
Requirement title	Automated Testing Coverage
Priority	Must
Category	Testing
Technical requirement description	Achieve 80% test coverage using unit tests, integration tests, and functional tests for critical features such as login and quiz generation.
Responsible Team Member(s)	QA Specialist / Developer

TR-UX-001 User Interface Accessibility

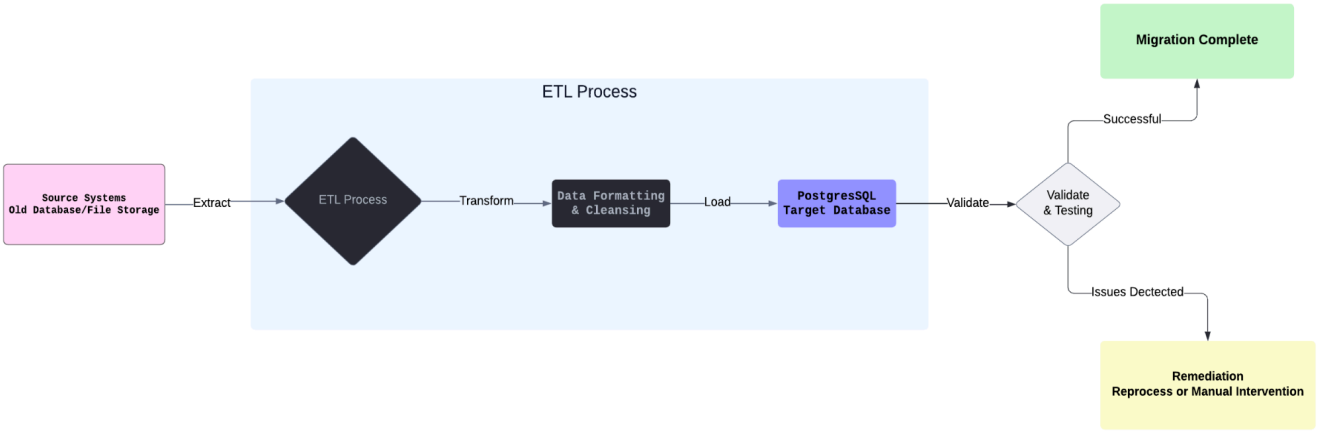
Item	Description
Requirement ID	TR-UX-001
Requirement title	User Interface Accessibility
Priority	Should
Category	User Experience
Technical requirement description	Follow WCAG 2.1 guidelines for web accessibility. Ensure keyboard navigation and screen reader compatibility for visually impaired users.
Responsible Team Member(s)	UX Designer / Frontend Developer

3 SYSTEM SPEC

The ASTREON system requires data migration to transition existing study data, user profiles, and associated materials to the new platform. This process involves extracting data from legacy systems, transforming it to align with the new database schema, and loading it into the PostgreSQL database used by ASTREON.

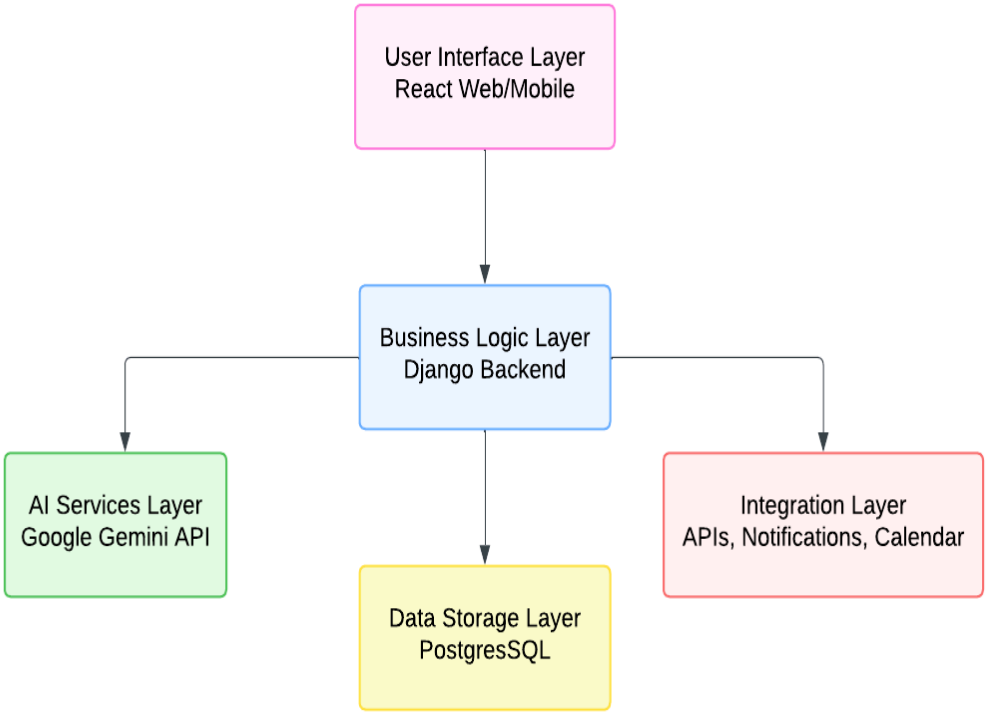
Steps in the Data Migration Process:

1. Extraction
   - Extract user data, uploaded materials, and progress tracking records from existing sources (e.g., old relational databases or file storage systems).
2. Transformation
   - Cleanse and format data to match the target schema of the PostgreSQL database.
   - Apply business rules, such as ensuring file paths and metadata align with ASTREON's structure.
3. Loading
   - Cleanse and format data to match the target schema of the PostgreSQL database.
   - Apply business rules, such as ensuring file paths and metadata align with ASTREON's structure.
4. Validation and Testing
   - Cleanse and format data to match the target schema of the PostgreSQL database.
   - Apply business rules, such as ensuring file paths and metadata align with ASTREON's structure.



3.1 SYSTEM DESIGN

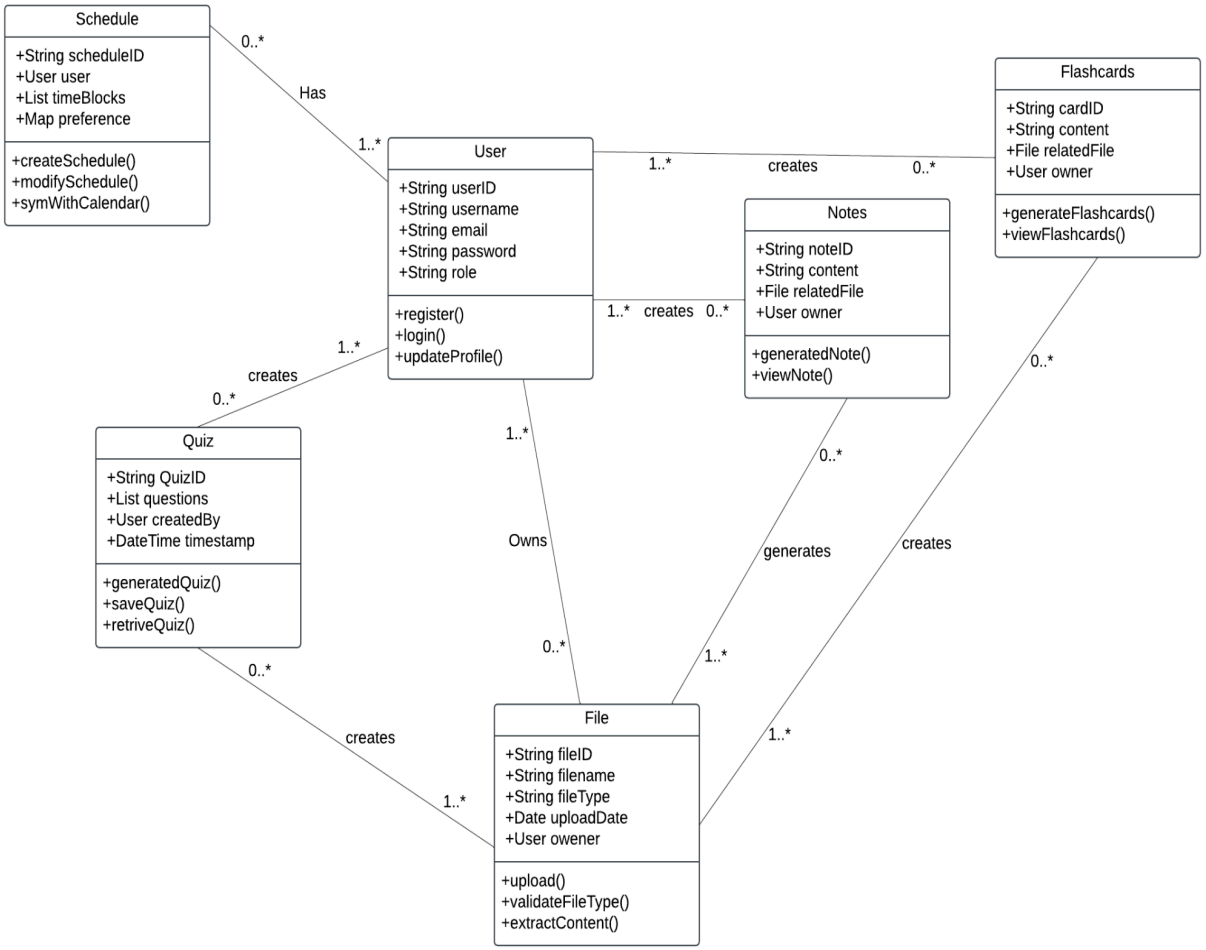
The ASTREON system is a cloud-based AI-powered platform designed to streamline the learning experience. It integrates various services, including user authentication, file upload, quiz generation, and personalized study schedules, to provide a comprehensive study assistant for students and teachers.



3.1.1 Application

The Application section provides a detailed breakdown of ASTREON's core components and interactions, with a focus on the Class Diagram to represent its structure. ASTREON's application is organized into modular layers, ensuring scalability and flexibility.

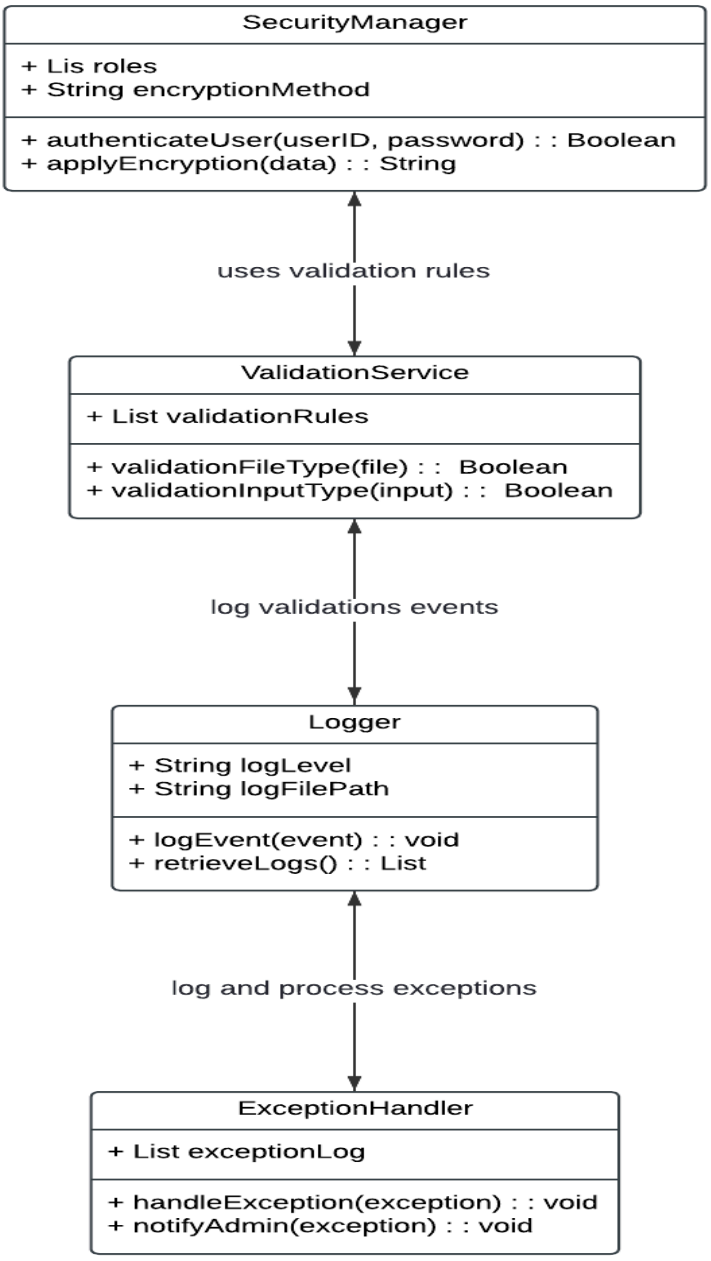
Class diagram


Schedule
+String scheduleID
+User user
+List timeBlocks
+Map preference
+createSchedule()
+modifySchedule()
+symWithCalendar()
0..*
Has
1..*
User
+String userID
+String username
+String email
+String password
+String role
+register()
+login()
+updateProfile()
1..*
creates
0..*
Flashcards
+String cardID
+String content
+File relatedFile
+User owner
+generateFlashcards()
+viewFlashcards()
Notes
+String noteID
+String content
+File relatedFile
+User owner
+generatedNote()
+viewNote()
1..*
creates
0..*
1..*
creates
0..*
Quiz
+String QuizID
+List questions
+User createdBy
+DateTime timestamp
+generatedQuiz()
+saveQuiz()
+retriveQuiz()
1..*
Owns
0..*
0..*
generates
1..*
0..*
creates
1..*
File
+String fileID
+String filename
+String fileType
+Date uploadDate
+User owener
+upload()
+validateFileType()
+extractContent()
1..*

3.1.2 Design Application

3.1.2.1 Describe Common Frameworks

Class diagram



Describe Common Frameworks

Security	● The system enforces role-based access control for Students, Teachers, and Admins. ● User authentication employs hashed passwords and MFA for Admins. ● TLS encryption ensures secure communication, and sensitive data (e.g., study progress) is encrypted at rest using AES-256. ● Regular audits and automated threat detection are implemented to monitor unauthorized access attempts.
Validation	● All critical business validations are located in the backend (business logic layer). ● Input validation ensures that uploaded files are in supported formats (e.g., PDF, DOCX, JPG). ● Study schedules are validated for time conflicts before being saved. ● Error messages are provided for invalid inputs at the presentation layer.
Transaction	● Operations like quiz generation or study material uploads are transactional, ensuring atomicity (either fully completed or rolled back). ● PostgreSQL's ACID properties guarantee data integrity across database operations.
Logging	● Detailed logs capture application activities, including login attempts, file uploads, and quiz completions. ● Logs exclude sensitive data and are time-stamped for traceability. ● Log retention is managed for 90 days and stored securely for auditing purposes.
Exception handling	● Exceptions are primarily handled at the backend, ensuring the user receives clear error messages for recoverable issues (e.g., *"File upload failed: Unsupported format"*). ● Unchecked exceptions are logged and monitored, triggering alerts for system administrators.
Reference table	● Static data such as quiz question types, supported file formats, and error codes are managed in database-backed reference tables. ● These tables can be updated without deploying new code.
Internationalisation	● Support for English as the primary language. Additional languages will be integrated in future versions. ● UTF-8 encoding ensures compatibility with multi-lingual data, especially in notes or flashcards.
Sample code artefact	● Implementation of a user authentication service using Django's built-in AbstractUser model, extended for role-based access control.

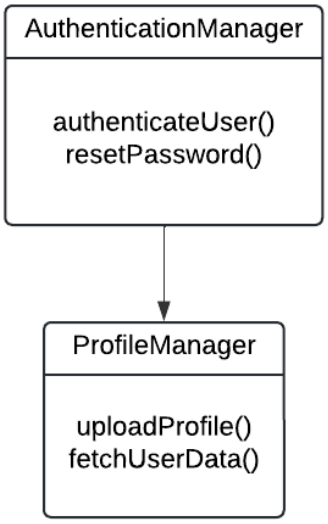
Describe each subsystem into components

Class Diagram

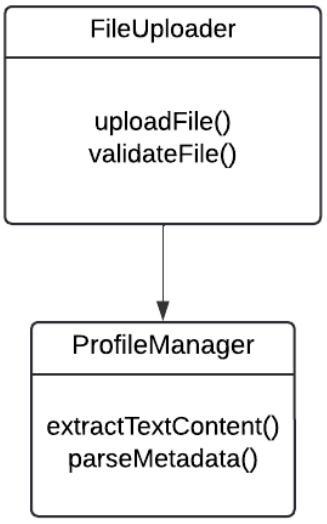
1. User Management Subsystem

Components:

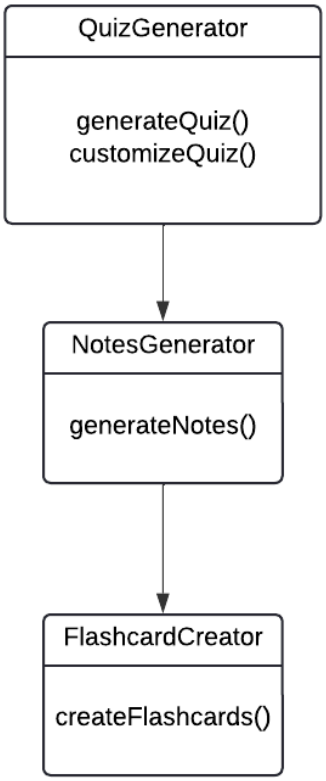
- AuthenticationManager: Manages login and role-based access.
- ProfileManager: Handles user profile updates.



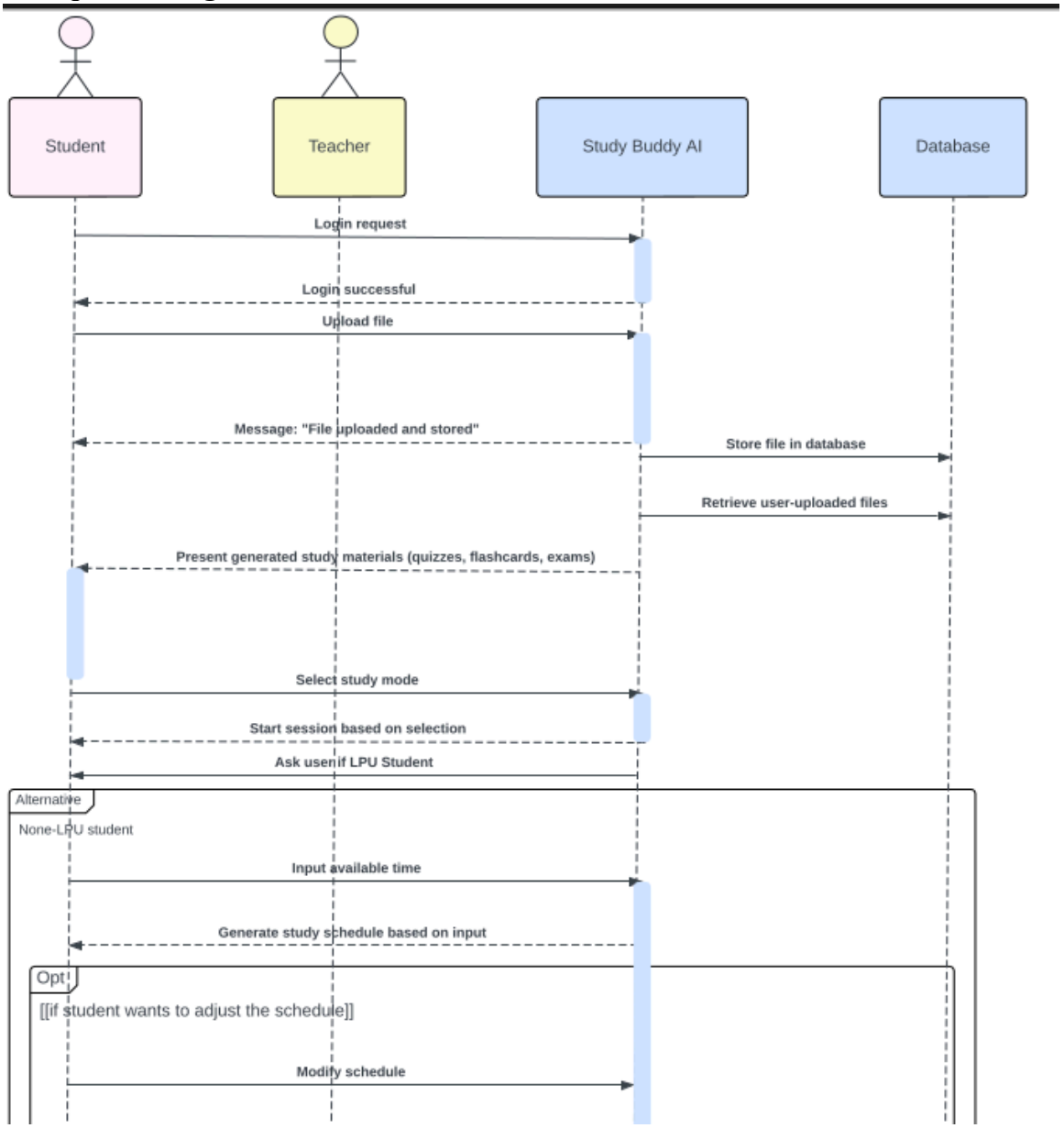
2. **File Processing Subsystem**
   Components:
   - FileUploader: Handles file uploads and validations.
   - ContentExtractor: Extracts text and metadata from uploaded files.

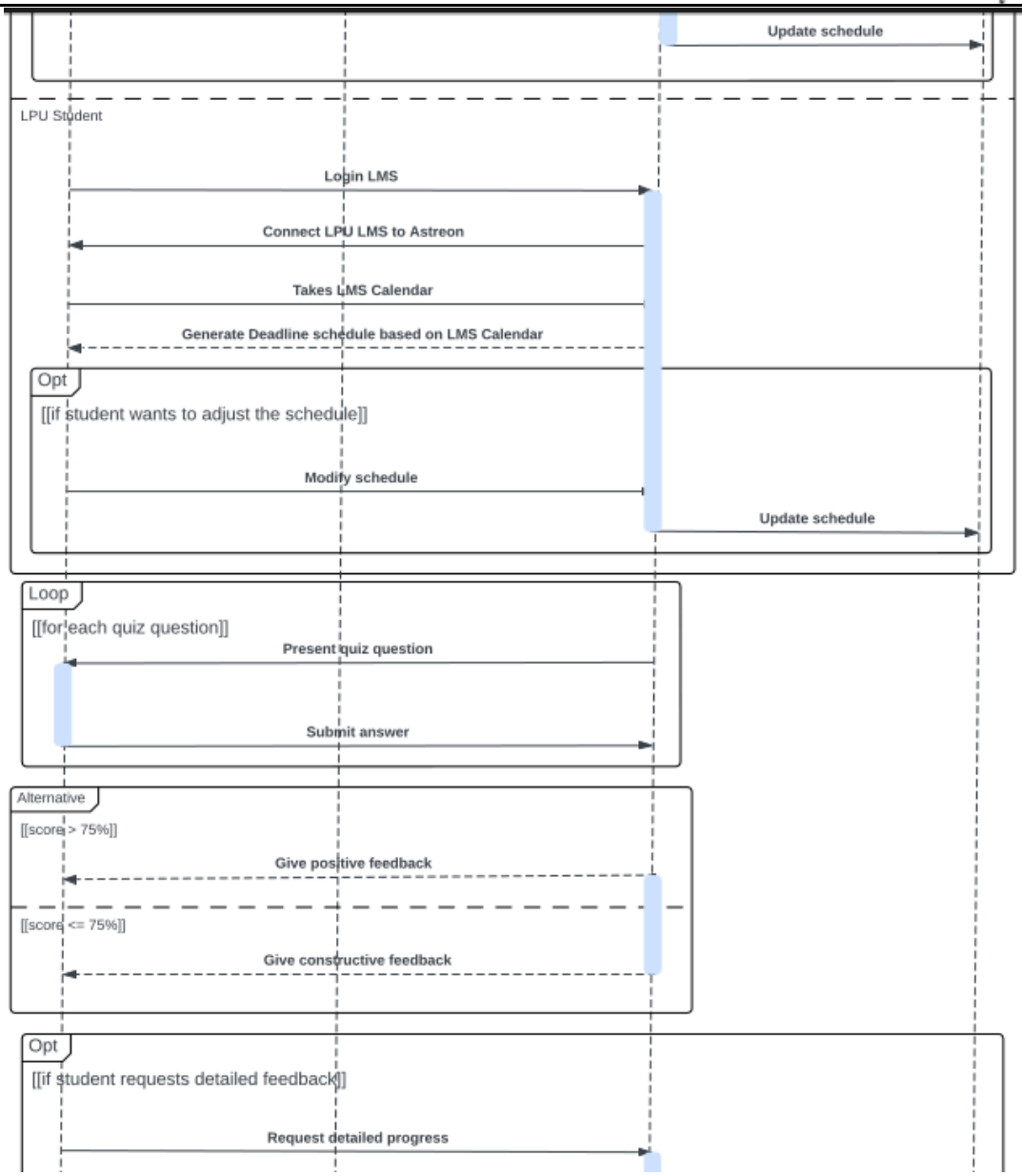


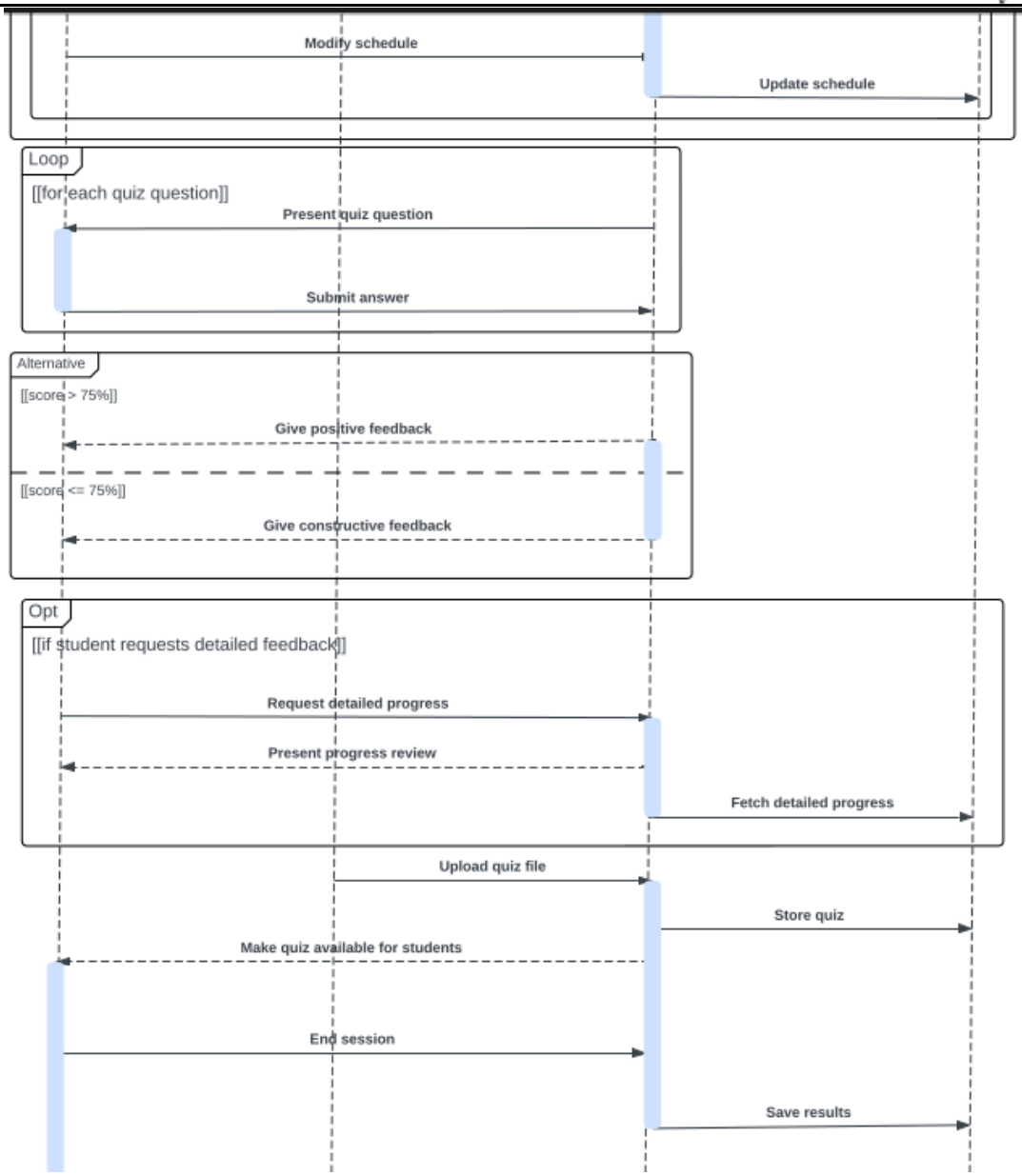
3. **AI Services Subsystem**
   Components:
   - QuizGenerator: Creates quizzes based on extracted content.
   - NotesGenerator: Summarizes uploaded materials.
   - FlashcardCreator: Builds flashcards from key points.


QuizGenerator
generateQuiz()
customizeQuiz()
NotesGenerator
generateNotes()
FlashcardCreator
createFlashcards()

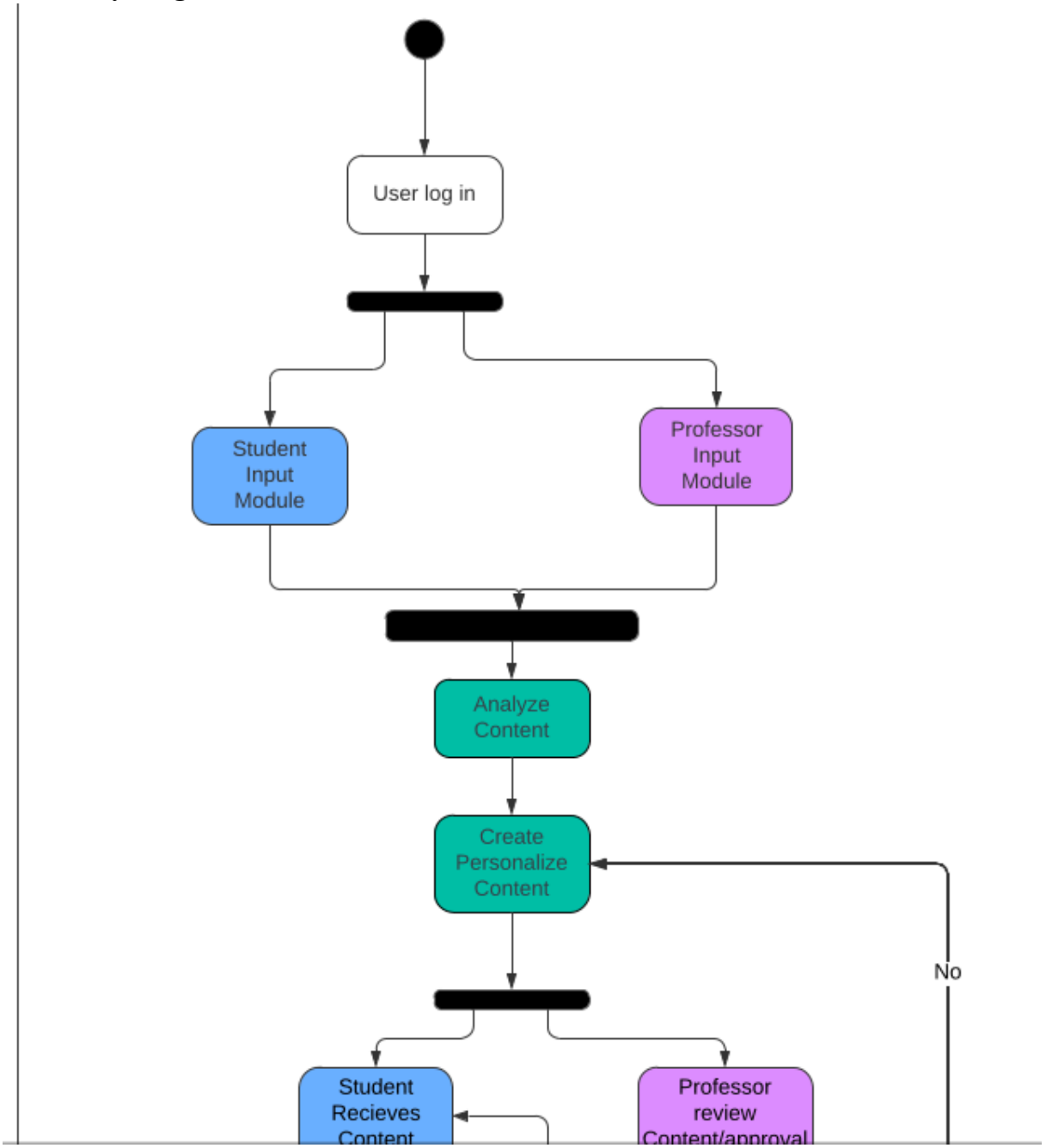
Sequence Diagram

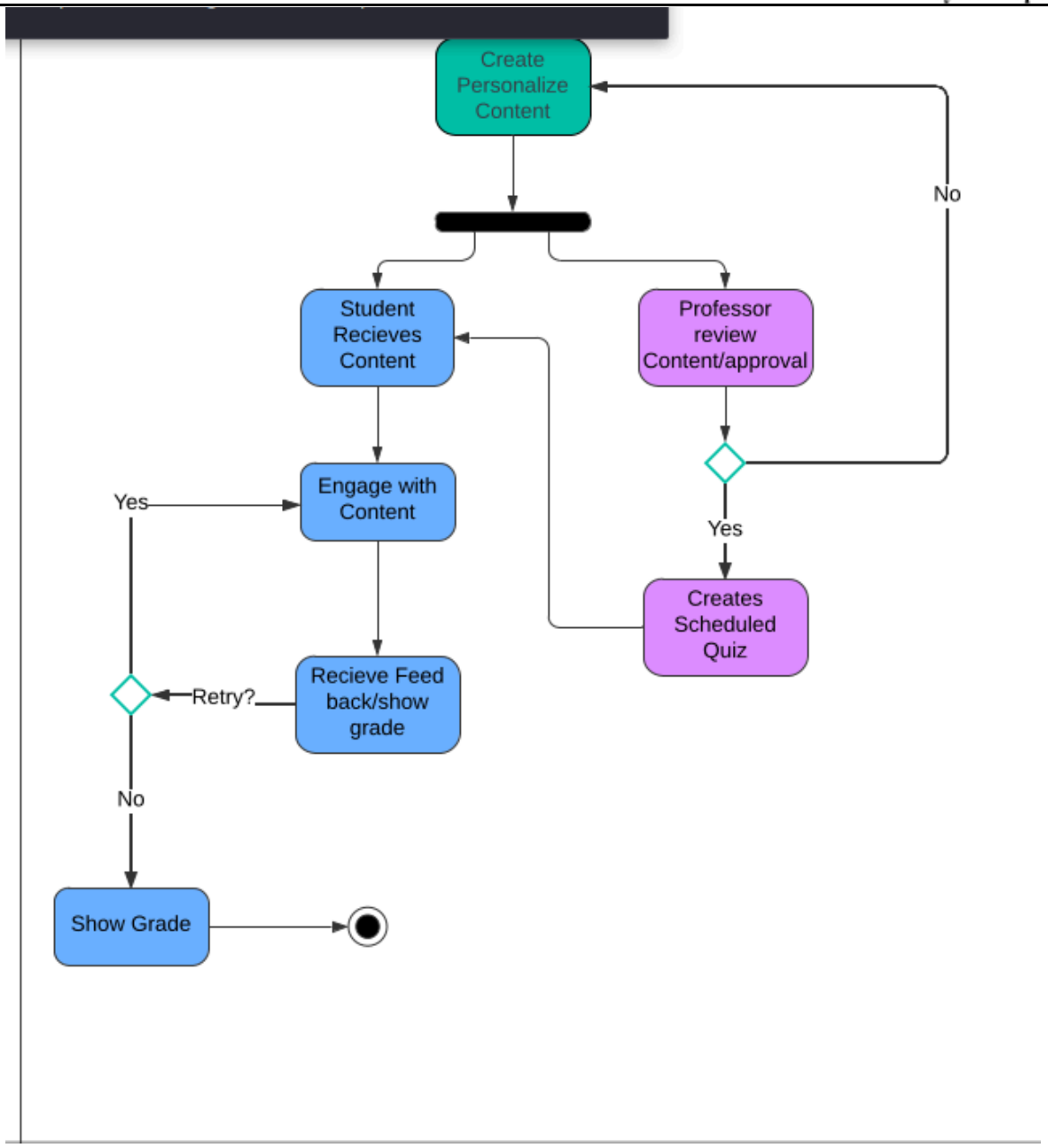



Update schedule
LPU Student
Login LMS
Connect LPU LMS to Astreon
Takes LMS Calendar
Generate Deadline schedule based on LMS Calendar
Opt
[[if student wants to adjust the schedule]]
Modify schedule
Update schedule
Loop
[[for each quiz question]]
Present quiz question
Submit answer
Alternative
[[score > 75%]]
Give positive feedback
[[score <= 75%]]
Give constructive feedback
Opt
[[if student requests detailed feedback]]
Request detailed progress

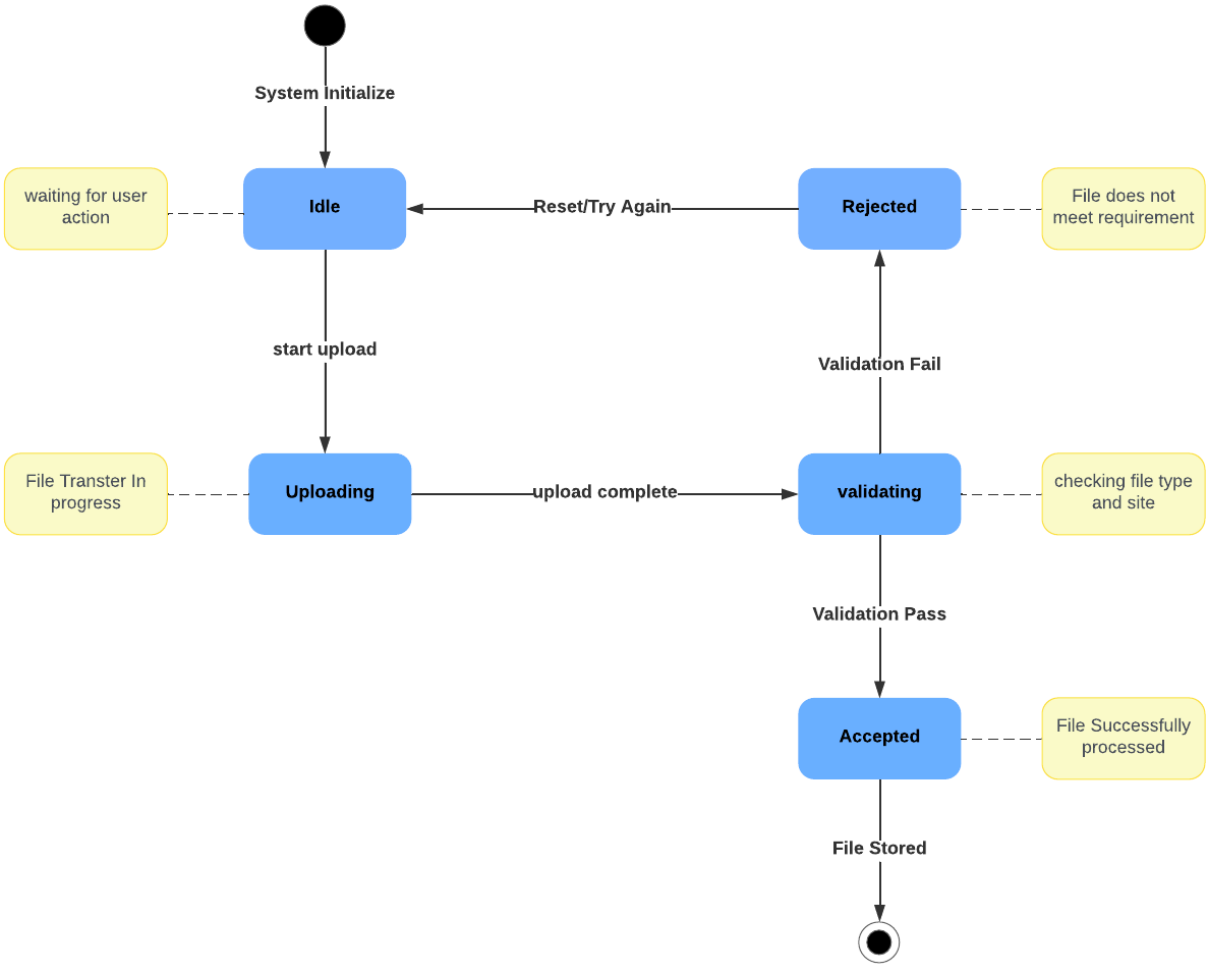

Modify schedule
Update schedule
Loop
[[for each quiz question]]
Present quiz question
Submit answer
Alternative
[[score > 75%]]
Give positive feedback
[[score <= 75%]]
Give constructive feedback
Opt
[[if student requests detailed feedback]]
Request detailed progress
Present progress review
Fetch detailed progress
Upload quiz file
Store quiz
Make quiz available for students
End session
Save results

Activity Diagram




Create Personalize Content
No
Student Recieves Content
Professor review Content/approval
Engage with Content
Yes
Yes
Creates Scheduled Quiz
Recieve Feed back/show grade
Retry?
No
Show Grade

State Diagram



Rule#	Rule	Rule Attributes	Rule Conditions	Rule Actions	Rule Priority	Rule Validity	Dependency
1	File Validation	File type, size, and format	File must be less than 10MB and in supported formats (PDF, DOCX, JPG).	Reject invalid files with an error message.	Must	Valid for all file uploads	None
2	User Authenticatio n	User role and credentials	Users must have valid credentials.	Allow access or reject login attempts.	Must	Valid for every login request	None
3	Quiz Generation	Content extracted from files	File content must be readable by AI models.	Generate quiz or prompt user for corrections.	Must	Valid when quiz generation is requested	Depends on File Validation
4	Study Schedule	User preferences and availability	Time blocks must not overlap.	Notify user if schedule conflicts exist.	Should	Valid during schedule creation	None

5	Progress Tracking	User quiz and flashcard usage	Must track only completed sessions.	Save progress data to database.	Must	Valid when user completes a session	Depends on Quiz Generation and Flashcards

Business Rules Description:

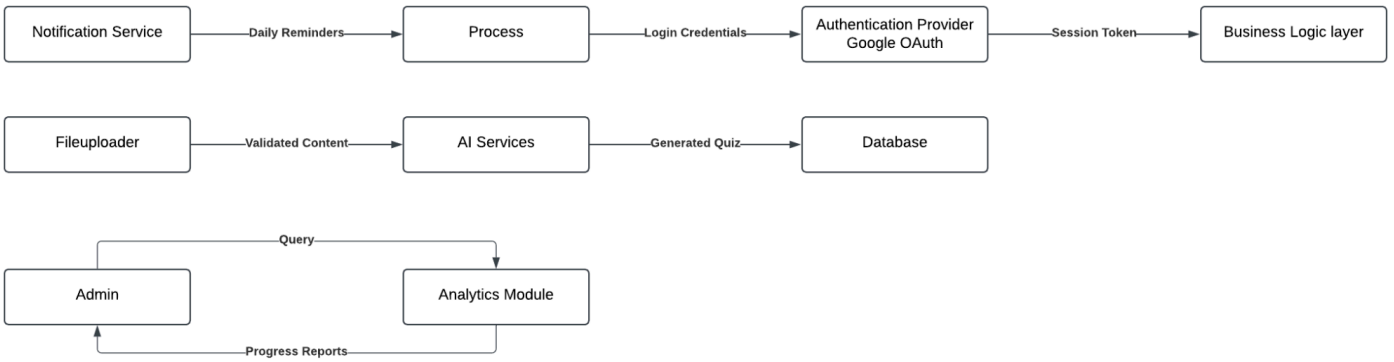
- **File Validation**: Ensures files are acceptable for processing.
- **User Authentication**: Validates user identity and roles.
- **Quiz Generation**: Creates quizzes only if file content is valid and readable.
- **Study Schedule**: Prevents overlapping schedules and conflicts.
- **Progress Tracking**: Tracks and saves user progress after valid study sessions.

3.1.3Design Integration

3.1.3.1 Integration Design Overview

Integration Flows:

1. Authentication
    - Interfaces with external authentication providers to validate user credentials and roles (e.g., Google OAuth).
2. AI Services
    - Sends user-uploaded file content to AI APIs for quiz generation, flashcards, and notes.
3. Notifications
    - Pushes reminders or updates to users via email or in-app notifications.
4. Progress Analytics
    - Pulls user activity data to generate analytics for performance tracking.



Interface Name	Interface Frequency Type	Actors Involved	Context goal	Preconditions	Post conditions
User Authentication	On-demand (each login request)	User, Authentication Provider (e.g., Google OAuth)	Validate user credentials and roles.	User provides valid credentials (email, password).	User is logged in, session token is generated.
Quiz Generation API	On-demand (triggered by user)	FileUploader, AI Services	Generate quizzes from uploaded content.	File content is successfully extracted and validated.	Generated quiz is stored in the database.
Notification Service	Periodic (daily for reminders)	User, Notification Provider	Notify users of pending study schedules or updates.	User has a valid study schedule or new update.	Notifications are successfully sent via email or app.
Progress Analytics	On-demand (triggered by admin query)	Admin, Analytics Module	Retrieve user activity data for progress analysis.	User activity data is available in the database.	Analytics reports are generated and displayed.

3.1.3.2 Data Mapping and Transformation Rules

Data control description:

<Source System> Data Element	Required (Y/N)	*<Target System>* Data Element	Required (Y/N)	Mapping Logic
Legacy Database userID	Yes	PostgreSQL userID	Yes	Direct mapping; lowercase field name conversion.
Legacy File System filePath	Yes	PostgreSQL file_path	Yes	Transform directory structure into new path format.
Legacy Database quizQuestions	Yes	AI Services API questions	Yes	Parse content, format as JSON for API compatibility.
AI Services Response generatedNotes	v	PostgreSQL notes	Yes	Extract key notes and save in the database field.
Notification Service emailAddress	Yes	PostgreSQL user_email	Yes	Validate email format before storing in the database.

3.1.3.3 Integration Sub-System Design

Details on Sub-Processes

Sub-Process Name	Input Fields	*Output Fields*	Data Types
Authenticate User	email, password	session_token, error	String, String
Upload File	fileName, fileType	fileID, filePath	String, String
Generate Quiz	fileContent, userID	quizID, questions	String, List
Send Notifications	emailAddress, message	deliveryStatus	String, Boolean

Exception Handling

- Authentication Failures
  - Invalid credentials trigger an error message ("Invalid login details.").
- File Upload Errors:
  - Unsupported file types or exceeding size limits trigger exceptions ("Unsupported file format.").
- Integration Failures:
  - API timeouts or incorrect responses are logged and retried automatically.

Logging and Security

- Logging Framework:
  - Logs capture critical events (e.g., failed logins, file upload errors) using Python's logging module or similar frameworks.
  - Logs are stored securely, with access restricted to system admins.
- Security Framework:
  - All sensitive data (e.g., user credentials) is encrypted using AES-256.
  - Global variables like API keys are securely stored in environment variables.

API and Library Details

API Name	**Purpose**	*Field Types*
Google OAuth	User authentication and role verification	email, password
Gemini AI Services	Quiz generation and content extraction	fileContent, JSON
SendGrid API	Email notifications	emailAddress, message

3.1.3.4 Design Data Conversion

Source data entity description:

Source	Source Data Entity	Destination	Target Data Entity	Transformation/ Cleansing Rules	Notes
Legacy Database	userID	PostgreSQL	user_id	Transform field name to lowercase; remove duplicates.	Ensure userID values are unique.
Legacy File Storage	filePath	PostgreSQL	file_path	Standardize directory structure for compatibility.	File paths must adhere to the new schema.
AI Service Responses	generatedNotes	PostgreSQL	notes	Extract key notes and format as plain text for storage	Ensure valid response structure from API.
Notification Logs	email	PostgreSQL	user_email	Validate email address format before storage.	Ensure no malformed email entries.

Expected Results of Data Conversion Process

1. Data Integrity:
   - All source data is accurately transformed and stored in the target database without loss.
2. Consistency:
   - Field names and formats in the target database are standardized.
3. Error-Free Conversion:
   - Invalid or incomplete records are flagged for review.

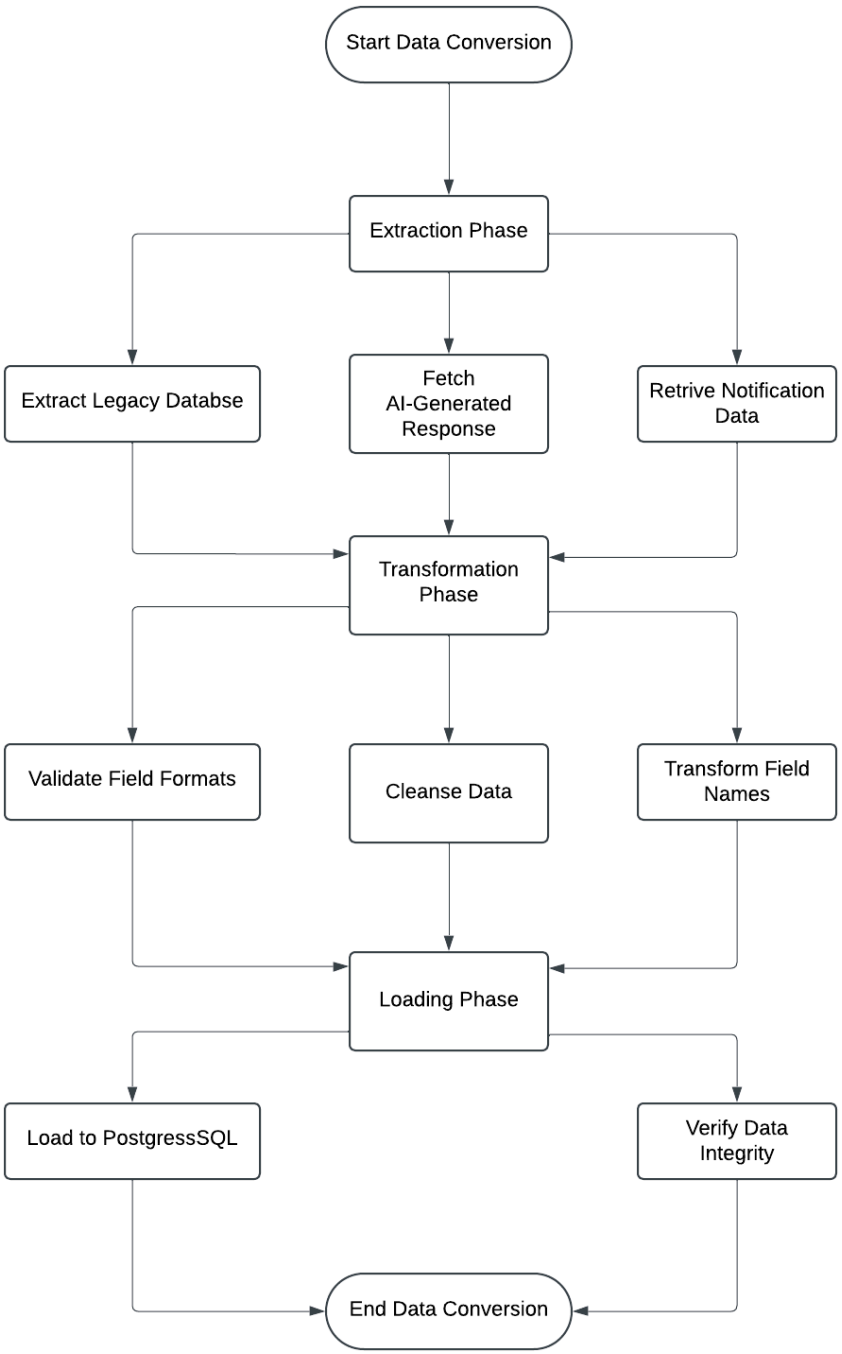
Business Rules for Data Conversion Process

- Data Integrity:
  - All source data is accurately transformed and stored in the target database without loss.
- Consistency:
  - Field names and formats in the target database are standardized.
- Error-Free Conversion:
  - Invalid or incomplete records are flagged for review.

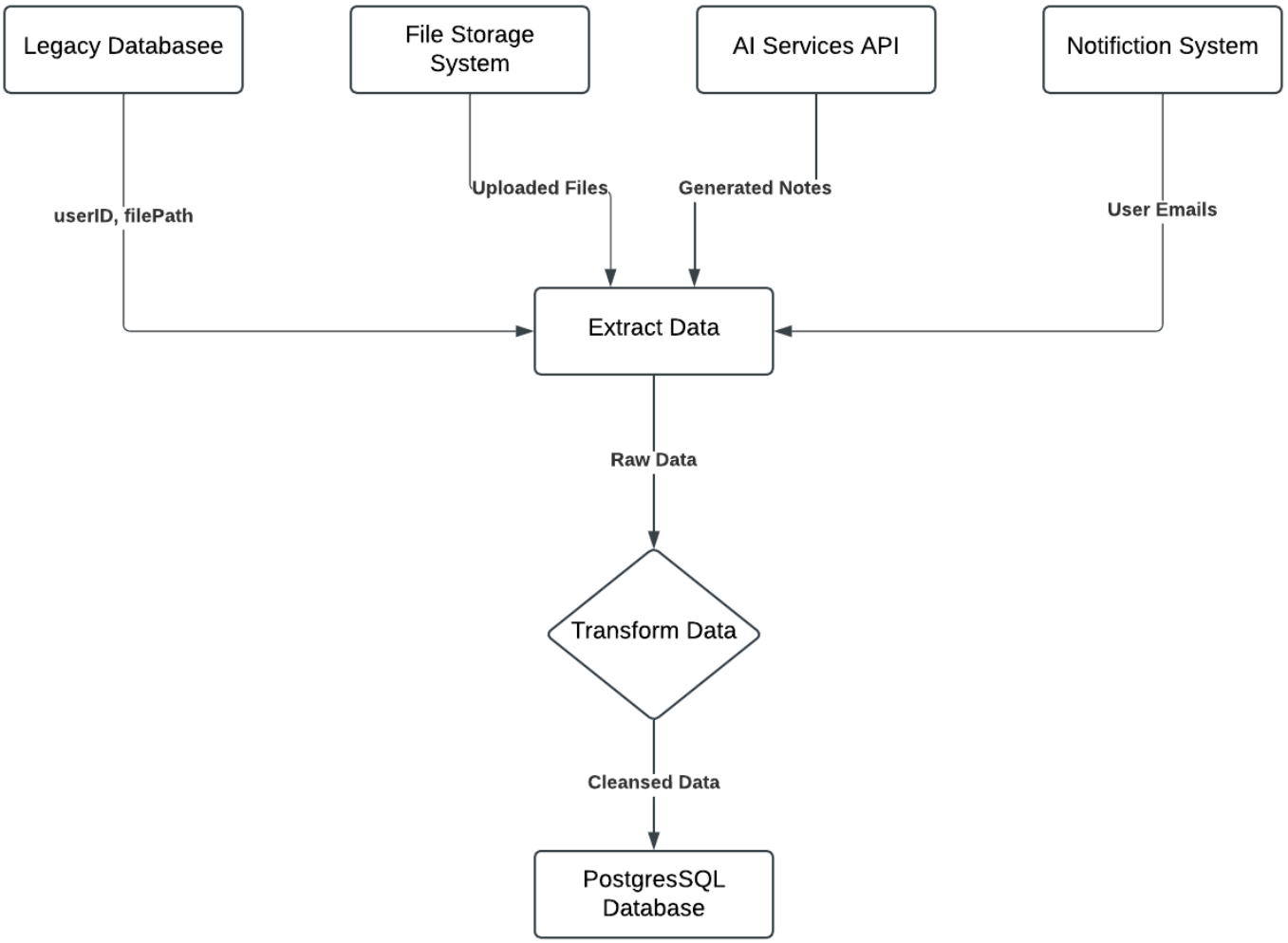
Data conversion results:

Source					**Target**				
Data Entity	**Description**	**Allowable Values**	**Field Type**	**Field Length**	**Cardinality**	**Mapping Rule**	**Data Entity**	**Field Type**	**Field Length**
userID	Unique identifier for users	Numeric, non-negative	Integer	10	One-to-One	Direct mapping; lowercase transformation required	user_id	Integer	10
filePath	Path to user-uploaded files	Valid file system paths	String	255	One-to-Many	Directory restructuring required	file_path	String	255
generated Notes	Notes extracted from AI services	Text data	Text	Unlimited	One-to-One	Extract key sections from AI response JSON	notes	Text	Unlimited
email	User email addresses	Valid email format	String	100	One-to-One	Validate format before storage	user_email	String	100

Process Flow Diagram:



Data Flow Diagram:



Error Handling Requirements

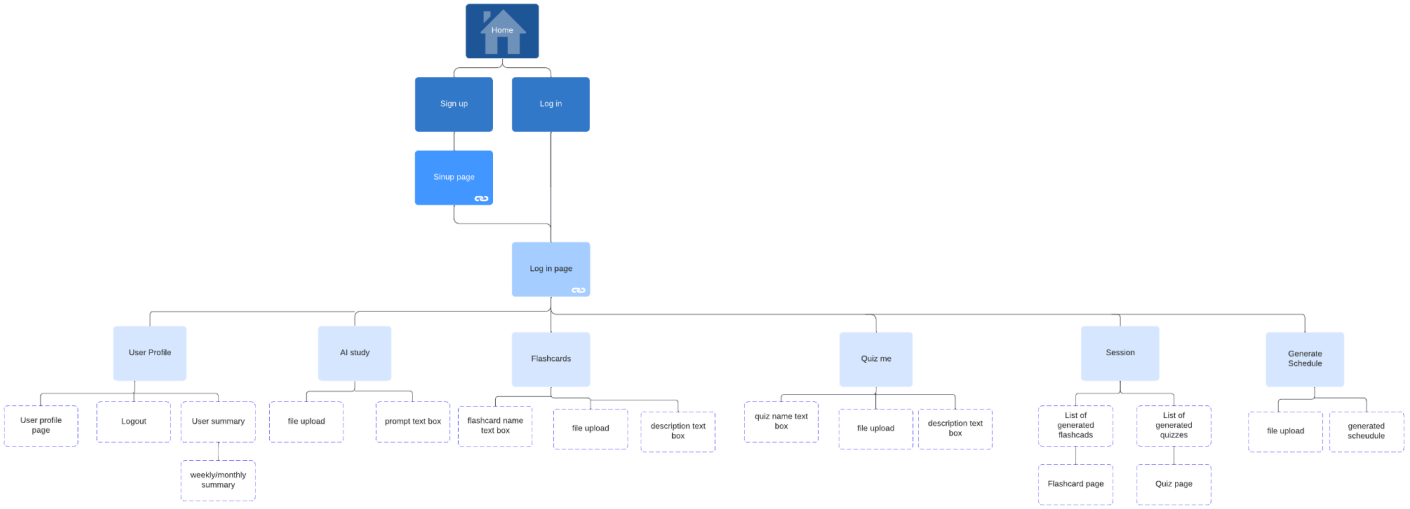
- Invalid Data::
  - Log and flag records with missing or malformed fields for review.
- Duplicate Records:
  - Remove duplicate entries during transformation.
- API Errors:
  - Retry failed API calls up to three times; log persistent failures.
  -

<Describe design of extraction programs, and conversion, cleansing, and loading programs.>

Step	**Details**
Extraction	Scripts to fetch data from legacy systems and API endpoints, using Python or ETL tools.
Transformation	Apply cleansing rules in Python or SQL scripts; validate field formats and constraints.
Loading	Use PostgreSQL COPY or INSERT commands to load data; ensure transaction safety.

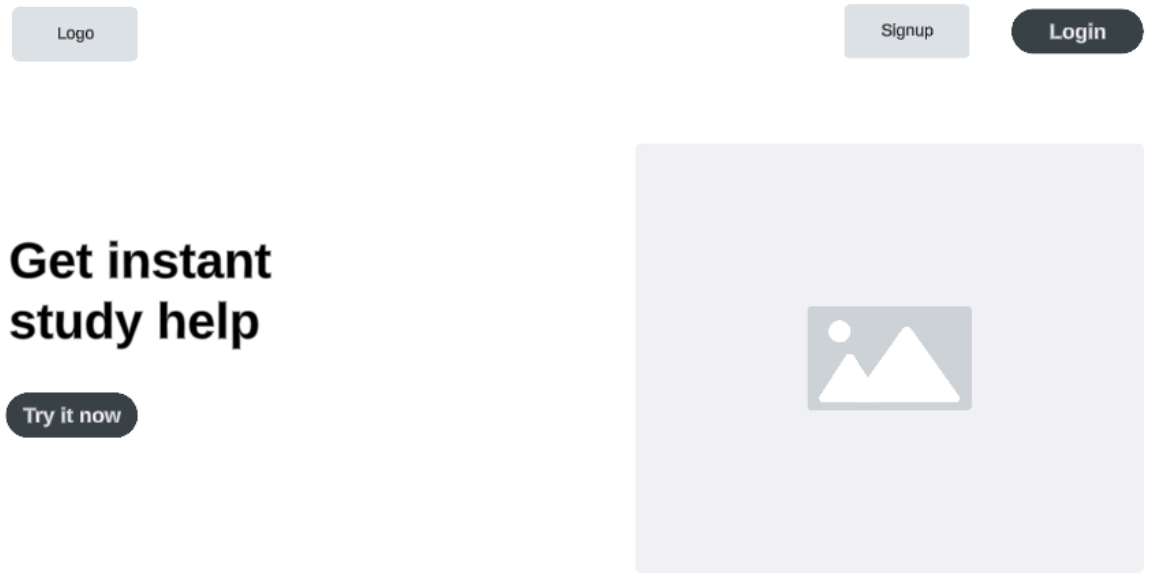
3.1.4 User Experience Design

(Site Map)



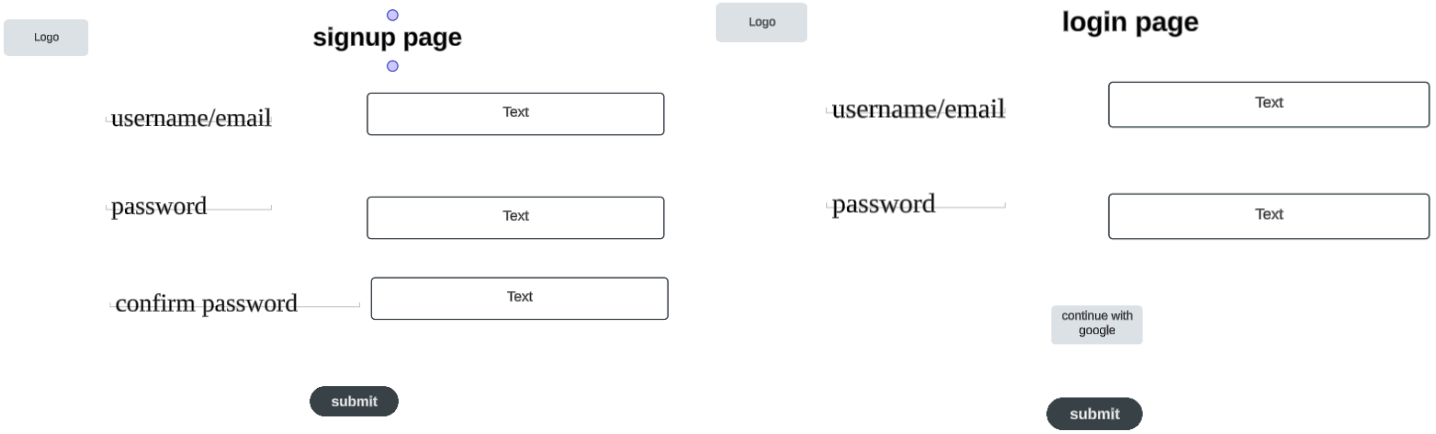
Screen Descriptions and Relationships

Screen/Page	**Description**	***Linked Screens***
Home Page	Entry point for all users. Provides login, registration, and basic app overview.	Login Page, Registration Page, About Page, Dashboard
Login Page	Allows existing users to authenticate and access their dashboard.	Home Page, Dashboard
Registration Page	Allows new users to create an account.	Home Page
Dashboard	Central hub for users to access all features and monitor their progress.	File Upload Page, AI Tools, Study Schedule, Notifications, Settings
Chatbot/File Upload Page	Allows users to send prompts and upload lesson materials for AI processing.	Dashboard, AI Tools
AI Tools	Provides access to quiz generation, flashcards, and notes.	Quizzes Page, Flashcards Page, Notes Page, Generate Schedules
Study Schedule	Allows users to view, create, and manage their study plans.	Dashboard, Notifications
Notifications	Displays reminders and updates for study activities.	Dashboard, Study Schedule
Settings	Allows users to manage account details and app preferences	Dashboard



Screen actions (if applicable):

Type	Label	Action	Comments
button	***signup***	***brings to the signup page***	***N/A***
button	***login***	***brings to the login page***	***N/A***
button	***try it now***	***brings to the signup page***	***N/A***



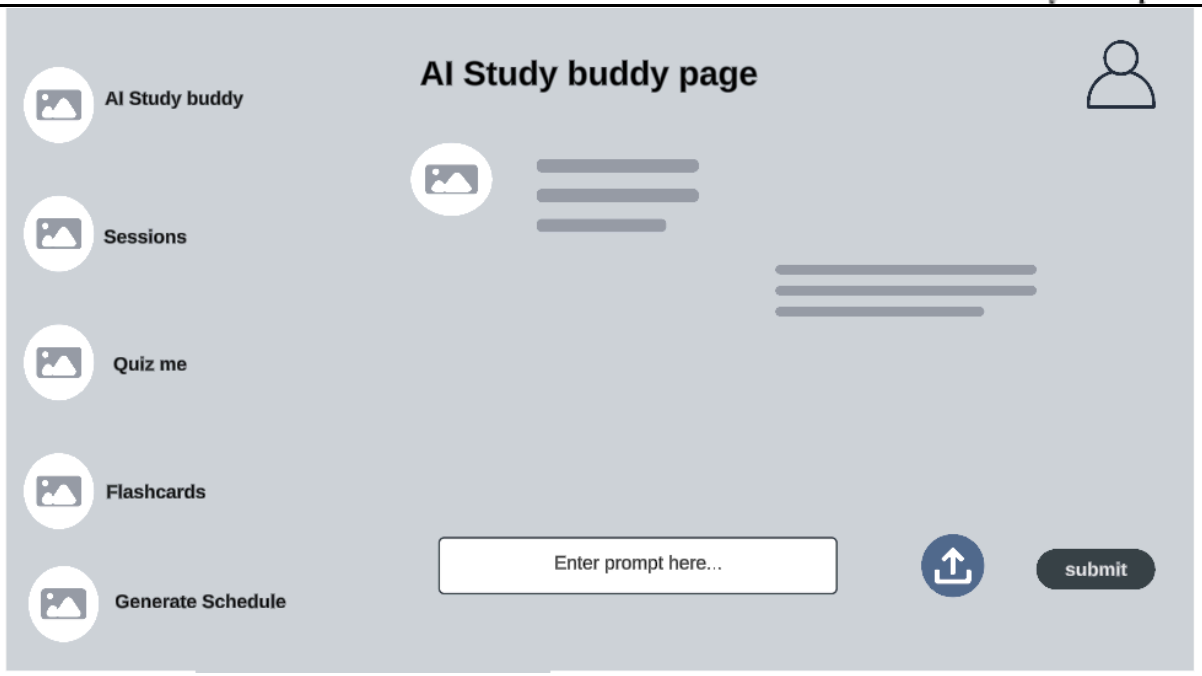
Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
Singup	*user_username* *user_email* *user_password*	*VARCHAR*	*should have combination of special characters, numbers*	***N/A***
Login	*user_username* *user_email* *user_password*	***VARCHAR***	*should have combination of special characters, numbers*	***N/A***

Screen actions (if applicable):

Type	Label	Action	Comments
Input	*username* *password* *confirm password*	*<Input>*	***N/A***
Button	***submit***	***submit the credentials inputted to be verifies***	***N/A***

Messages (if applicable):

Message ID	Description	Type	Triggering Event
MSG101	***Display a message to the user that they cannot use the username anymore***	***Error message***	***Submit***
MSG102	***Validation for strong password***	***Error message***	***Submit***
MSG103	***Display warning when the credentials during signup doesn't match***	***Warning***	***Submit***



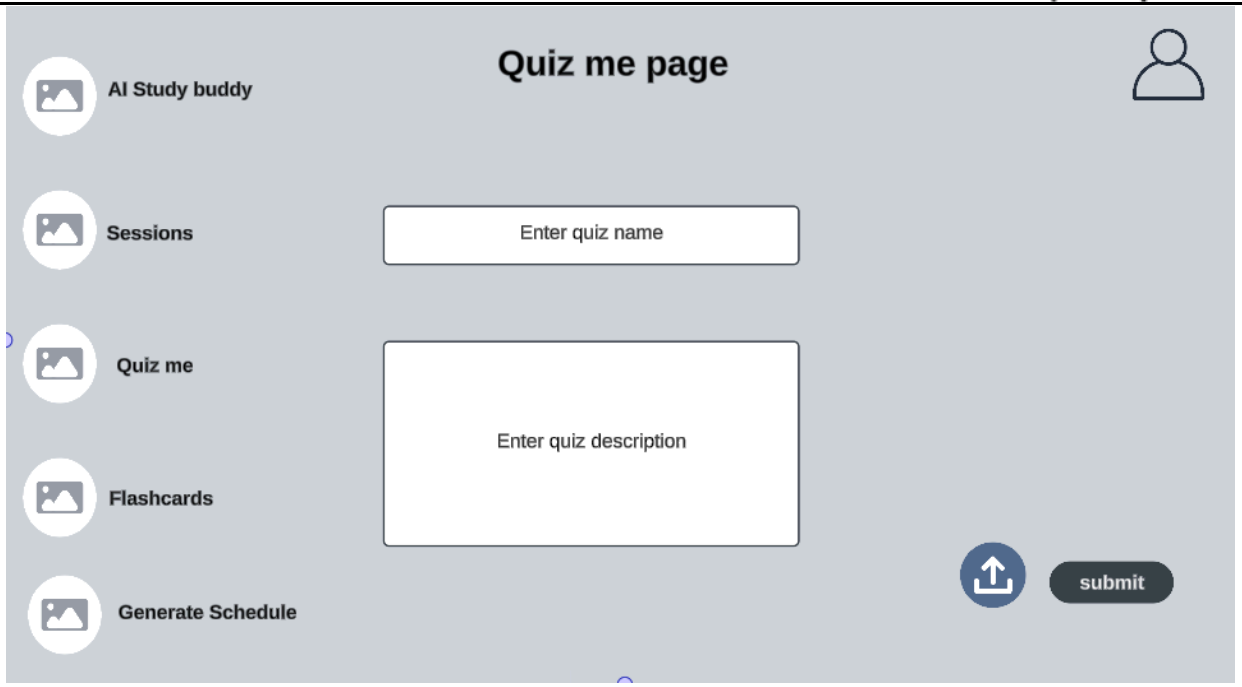
Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
AI chatbot	*user_prompt chatbot_prompt*	*VARCHAR*		***N/A***

Screen actions (if applicable):

Type	Label	Action	Comments
input	*enter your prompt here*	*user will input a query in the textbox*	***N/A***
button	*button*	*submit the user query to be analyzed by the ai*	***N/A***
button	*upload file icon*	*user query attachment files*	***N/A***

Messages (if applicable):

Message ID	Description	Type	Triggering Event
MSG104	***Displays a message that the file size is too big***	***Error message***	***Submit***



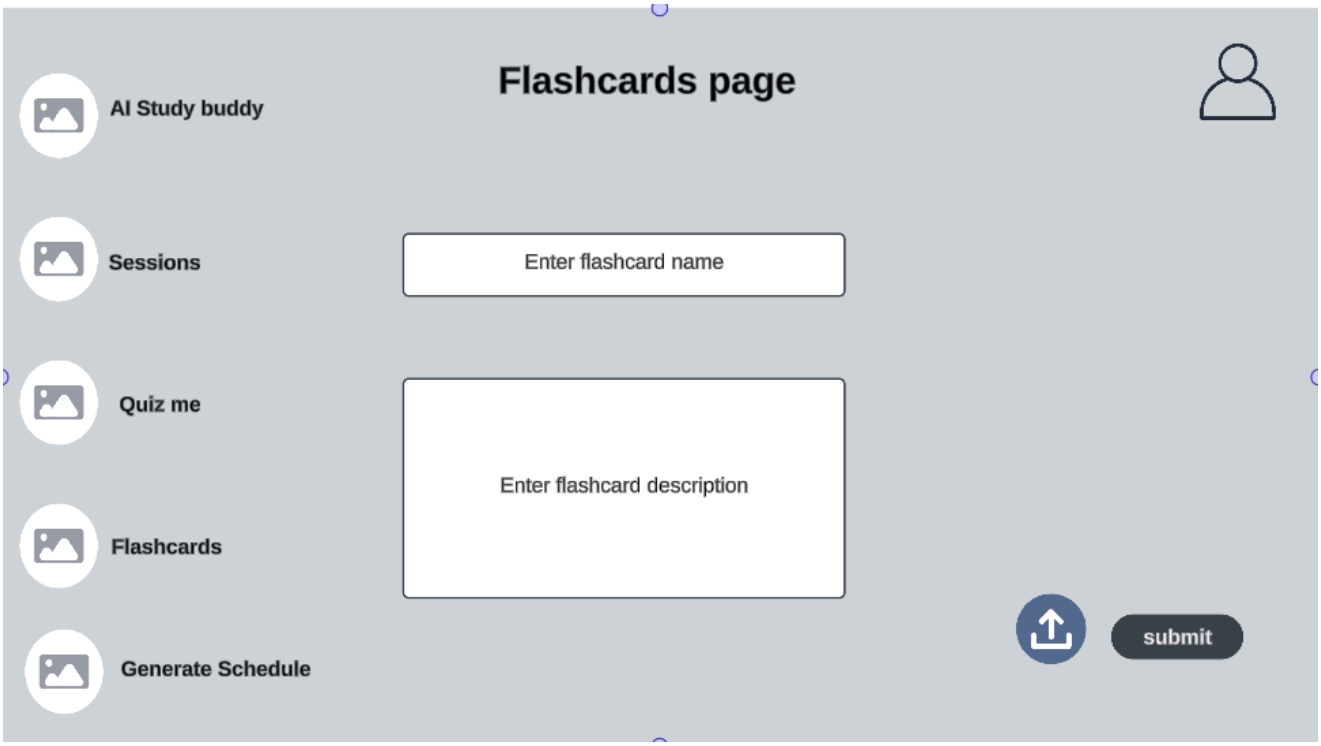
Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
Quiz me	*user_quiz* *user_quiz_description*	*VARCHAR*		***N/A***
Quiz me	*user_files*	***to be stored in AWS***	*should be static files*	***N/A***

Screen actions (if applicable):

Type	Label	Action	Comments
Input	*quiz name* *quiz description*	*Allow the user to input the quiz name and description*	*N/A*
Button	***submit***	***submit and generates the quiz***	***N/A***
Button	***upload file***	***upload files for the quiz to be generated on***	***N/A***

Messages (if applicable):

Message ID	Description	Type	Triggering Event
MSG104	***Displays a message that the file size is too big***	***Error message***	***Submit***

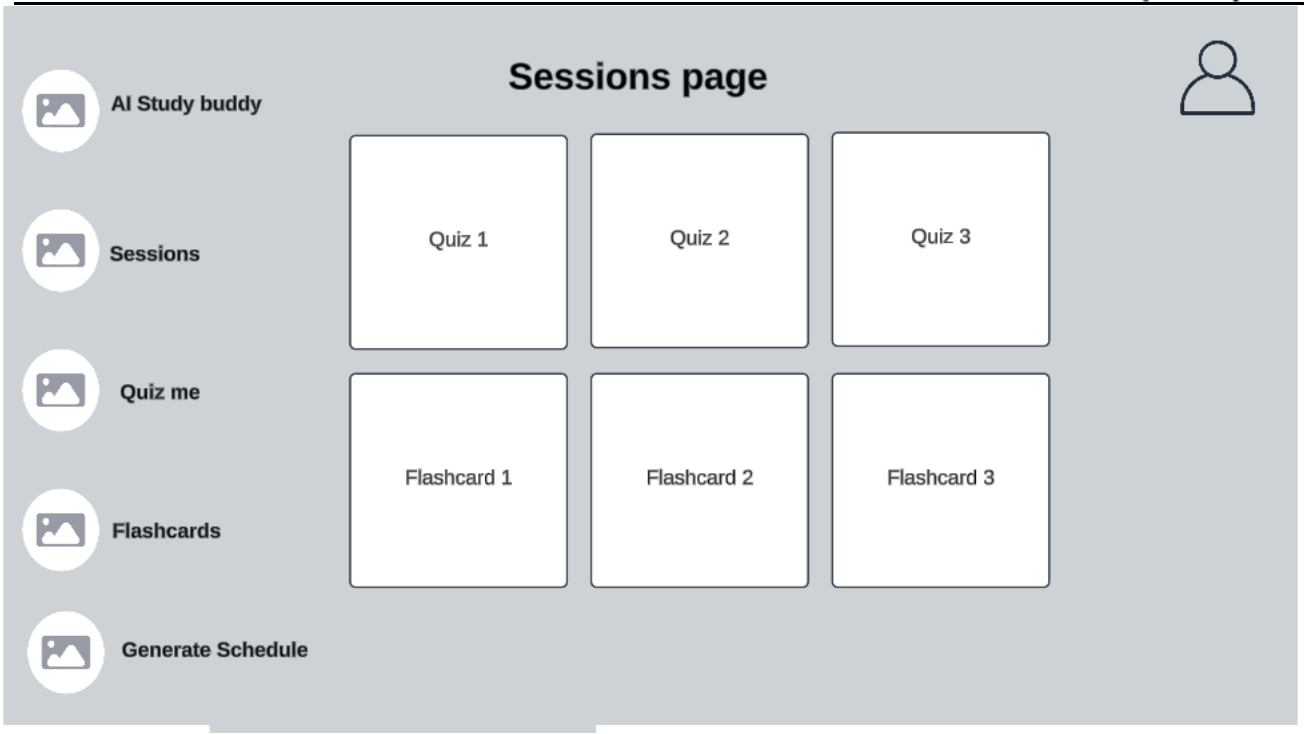


Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
Quiz me	*user_quiz* *user_quiz_description*	*VARCHAR*		***N/A***
Quiz me	*user_files*	***to be stored in AWS***	*should be static files*	***N/A***

Screen actions (if applicable):

Type	Label	Action	Comments
Input	*quiz name* *quiz description*	*Allow the user to input the flashcards name and description*	*N/A*
Button	***submit***	***submit and generates the flashcards***	***N/A***
Button	***upload file***	***upload files for the flashcards to be generated on***	***N/A***

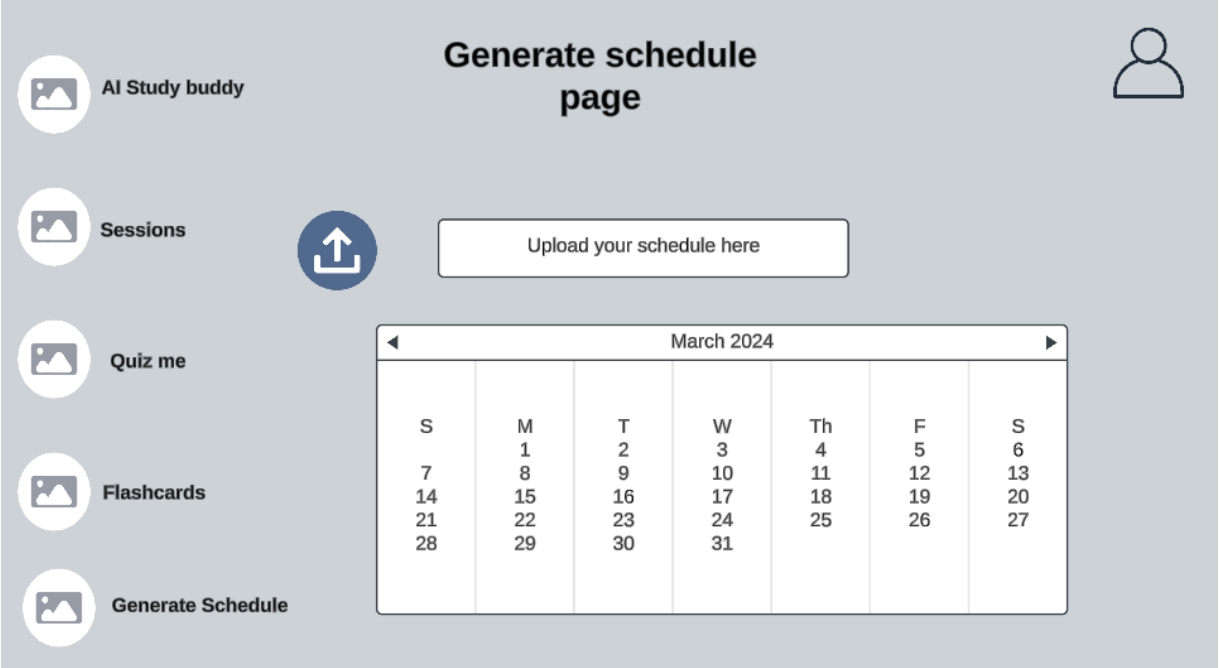
Message ID	Description	Type	Triggering Event
MSG104	***Displays a message that the file size is too big***	***Error message***	***Submit***



Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
Sessions	*user_session*	*VARCHAR*	*token to store for the user session*	***N/A***
Sessions	*user_quiz*	***INT(PK)***	*should be user quiz primary key, can be null*	***N/A***
Sessions	*user_flashcard*	***INT(PK)***	*should be user flashcard primary key, can be null*	

Screen actions (if applicable):

Type	Label	Action	Comments
button	*quiz* *flashcard*	*The designated button will go to either flashcard or quiz page*	*N/A*



Elements presented (if applicable):

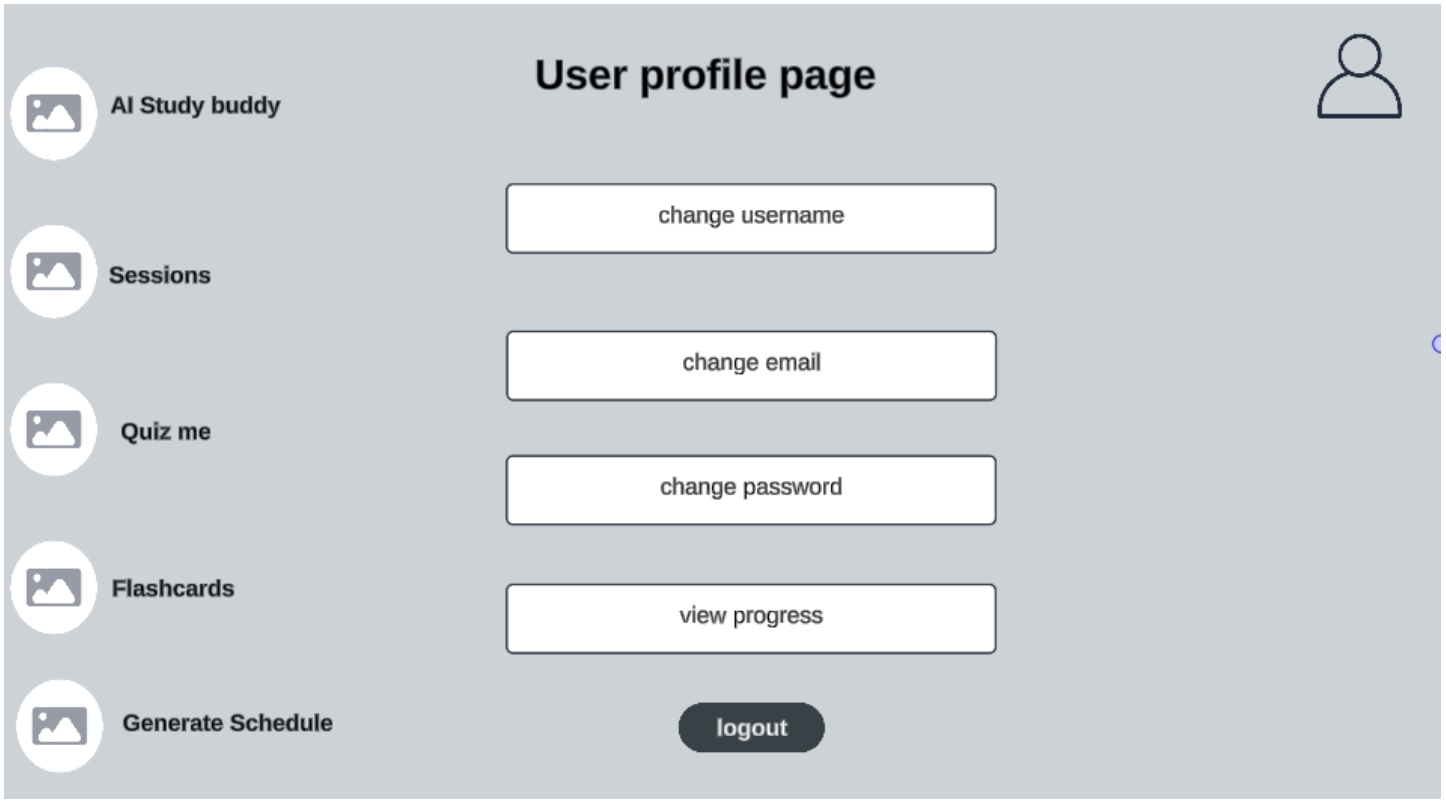
Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
Study Start Time	Schedule, start_time	Time	Valid time format (HH:MM AM/PM)	Required input field
Study End Time	Schedule, end_time	Time	Valid time format (HH:MM AM/PM)	Must be after start time
Preferences	Schedule, preferences	Text	Cannot be empty	User-defined input
Days of Week	Schedule, days_of_week	Char(20)	Select multiple valid days.	Checkbox inputs for days

Screen actions (if applicable):

Type	Label	Action	Comments
Button	"Generate Schedule"	Creates and saves the schedule.	Ensures no time conflicts.
Button	"Clear Form"	Clears all input fields	Resets all fields to blank
Link	"Back to Dashboard"	Navigates to the dashboard	Returns user to main hub.

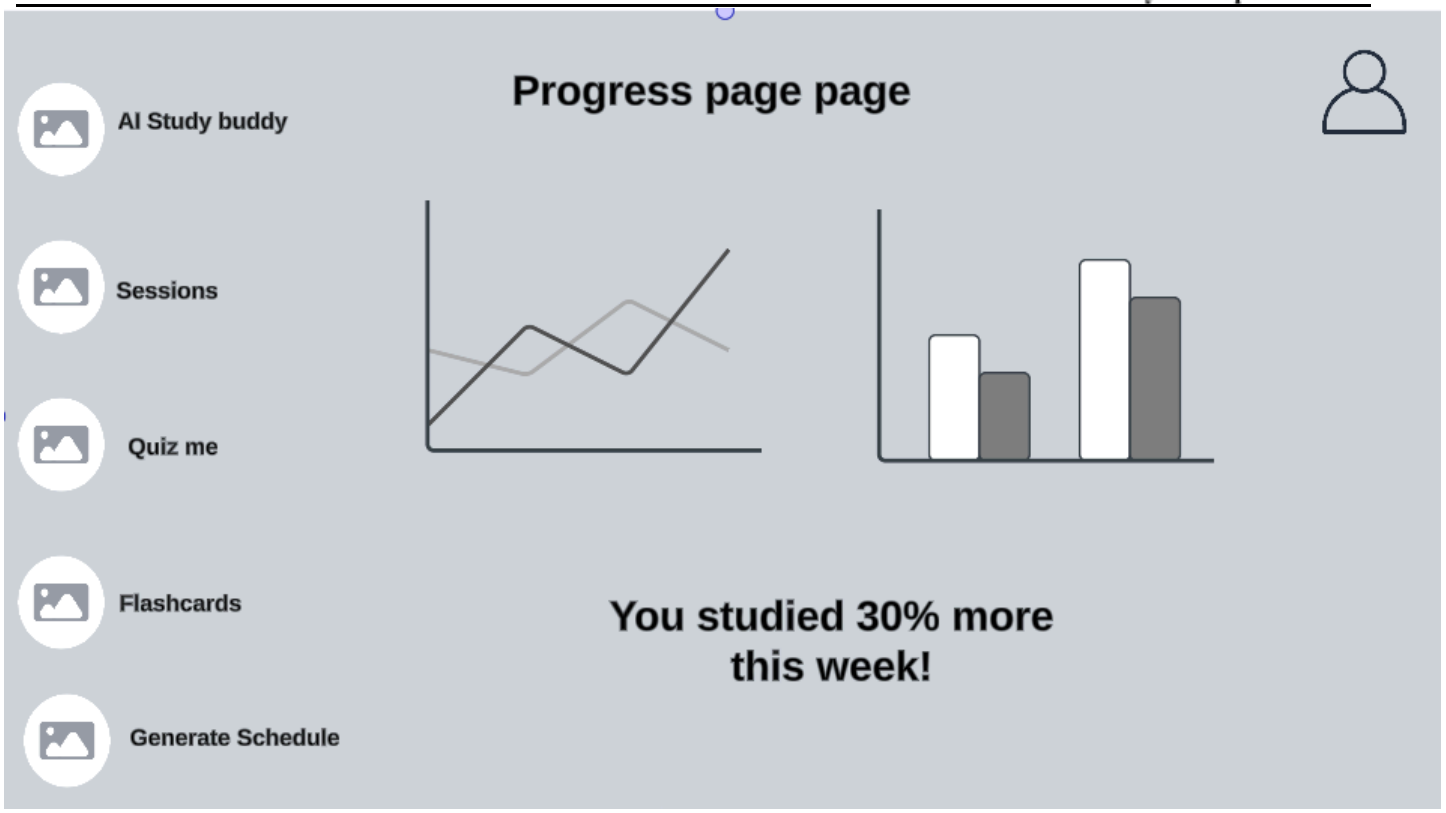
Messages (if applicable):

Message ID	Description	Type	Triggering Event
MSG 104	***Displays a message that the file size is too big***	Error	upload button
MSG701	"Schedule created successfully.”	Success	After clicking "Generate Schedule."
MSG702	"Error: Time blocks overlap."	Error	When time conflicts are detected
MSG703	"Please complete all required fields."	Error	When mandatory fields are empty



Screen actions (if applicable):

Type	Label	Action	Comments
input	*change username* *change email* *change password*	*allows the user to modify their credentials*	*N/A*
button	*view progress*	*opens progress page*	*N/A*
button	*logout*	*logouts the user*	*N/A*
button	*submit*	*saves the user modified credential*	*N/A*



Elements presented (if applicable):

Field Name in Mock-up	Table Name, Column Name	Data Type	Validation	Comments
Progress page	*session_start_time* *session_end_time*	*DATETIME*		

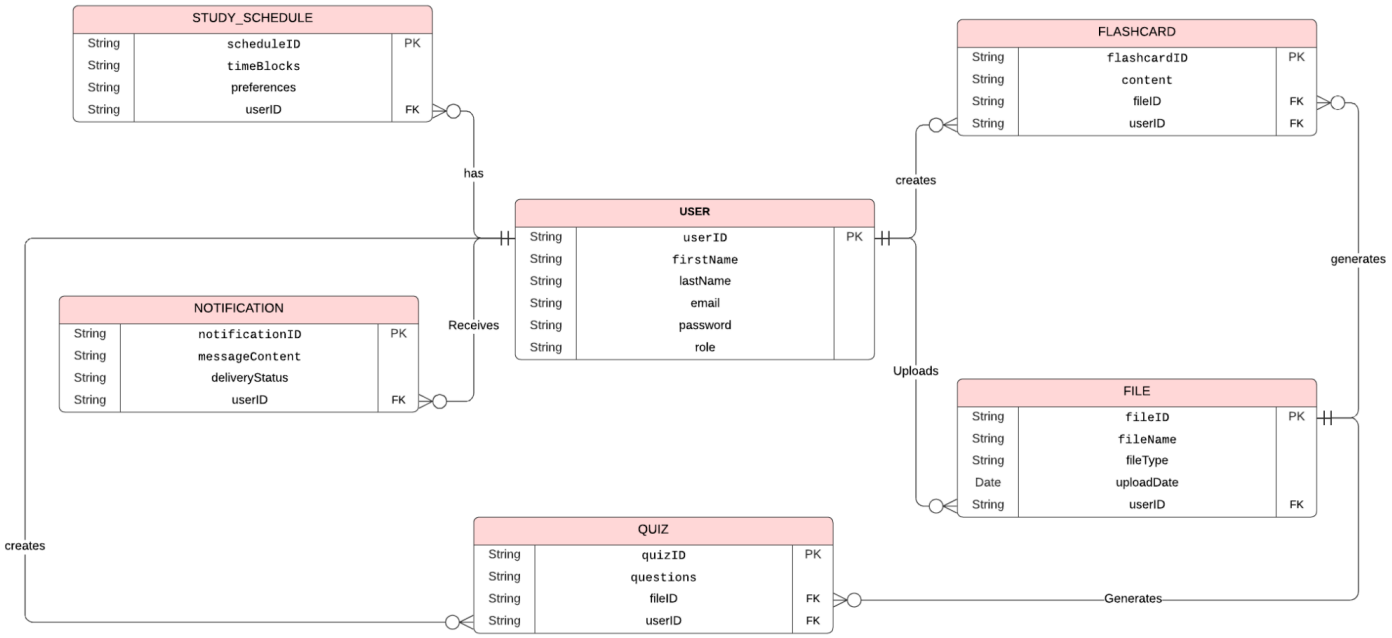
3.1.5 Data Model

The Data Model section defines the organization of data within ASTREON, including its logical representation. It provides a high-level structure of how data is grouped, its attributes, and the relationships between data entities. This ensures the system's database supports its functionalities effectively, such as managing user profiles, file uploads, quiz generation, and study schedules.

3.1.5.1 Logical Data Model

Logical data entity description:

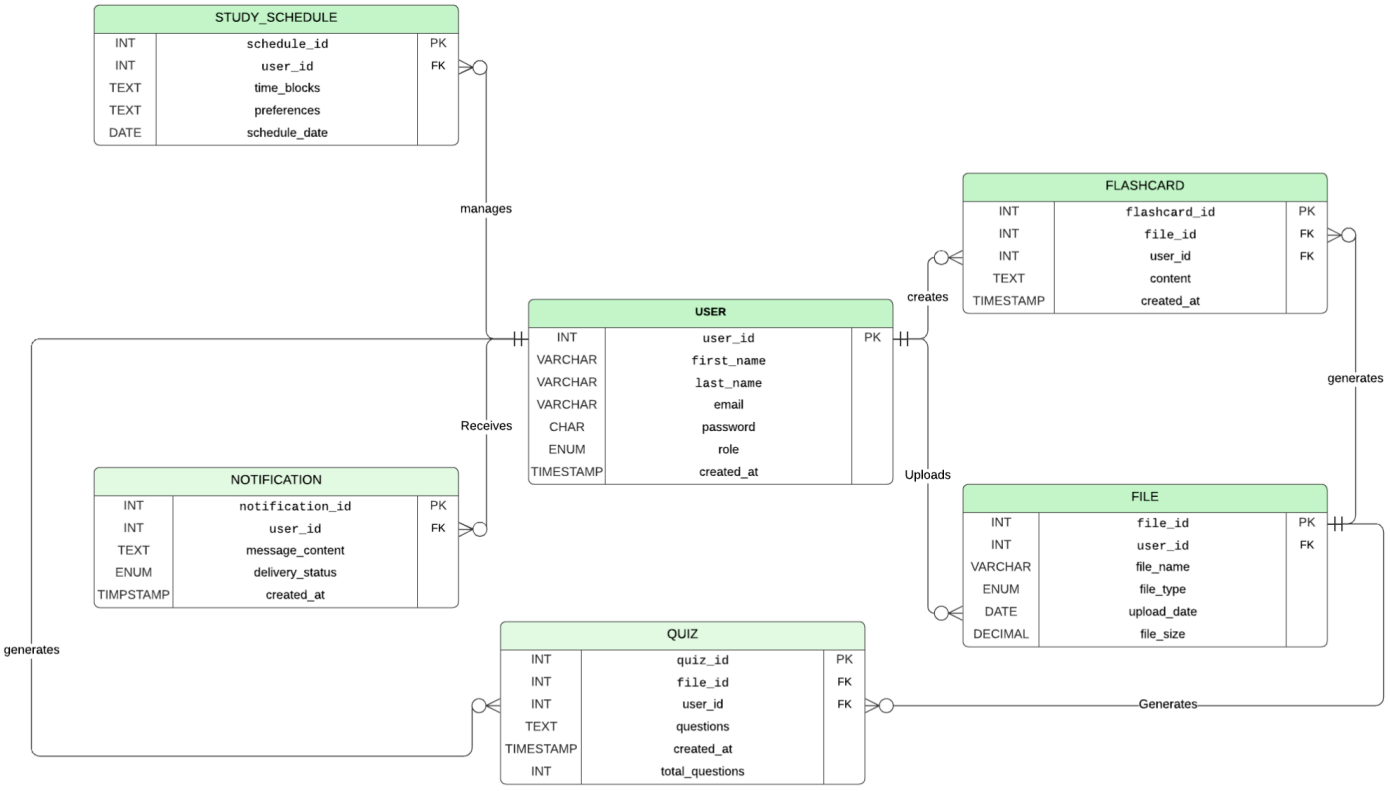
Logical Data Entity	Logical Data Entity Description
User	Represents a registered user of the system, including attributes like first name, last name, email, role, and password
File	Represents a file uploaded by a user, including attributes like filename, file type, upload date, and associated user ID
Quiz	Represents a generated quiz, including attributes like quiz ID, questions, user ID, and associated file ID
Flashcard	Represents flashcards generated from files, including attributes like flashcard ID, content, and associated file ID.
Study Schedule	Represents a personalized study schedule, including attributes like schedule ID, user ID, time blocks, and preferences.
Notification	Represents notifications sent to users, including attributes like notification ID, message content, delivery status, and user ID.



3.1.6 Physical Data Model

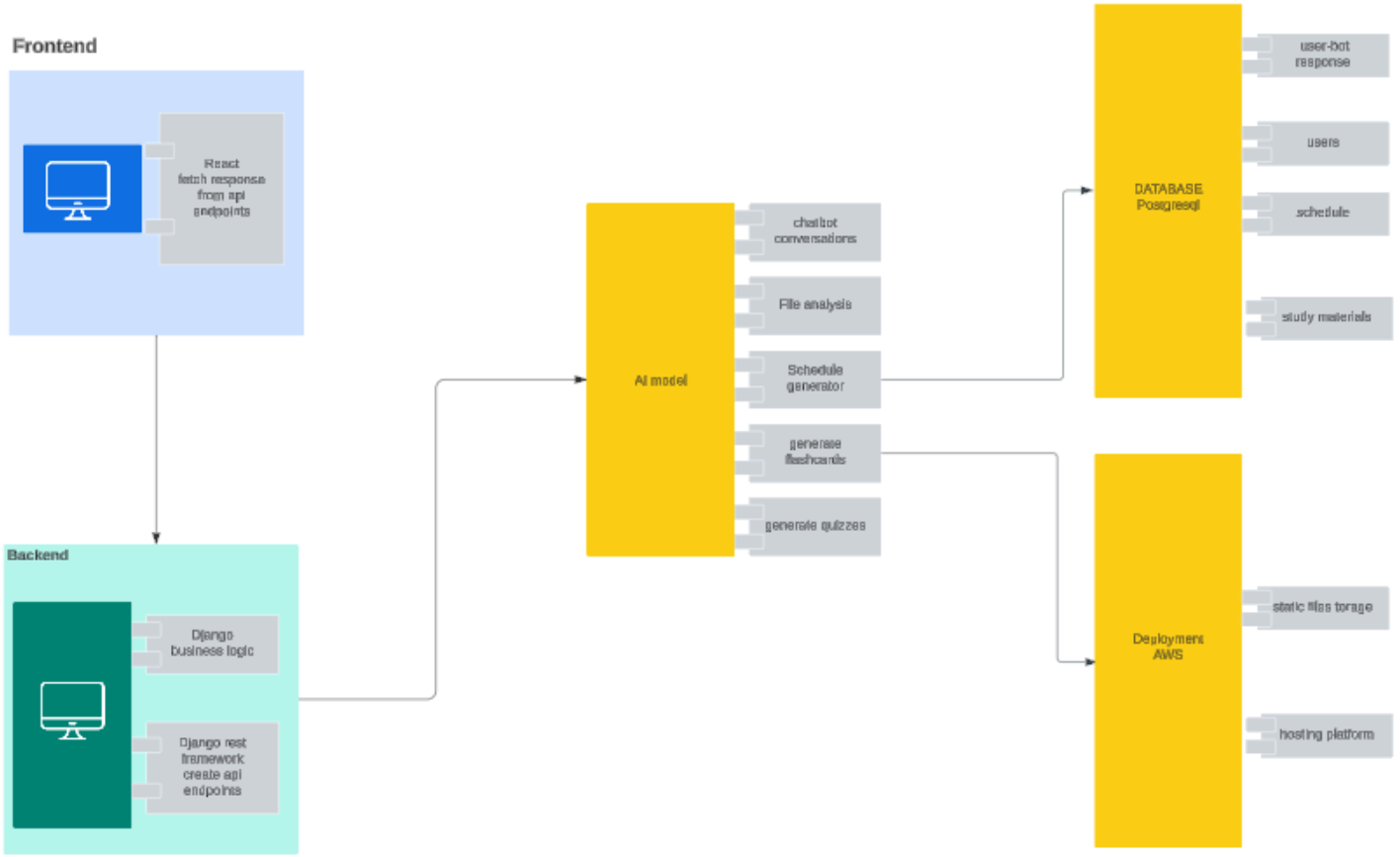
Table Name	Field Name	Field Format	Field Length	Description	Mandatory	Primary Key	Foreign Key
User	user_id	Integer	10	Unique identifier for the user	Yes	Yes	None
	first_name	Char	50	User's first name	Yes	No	None
	last_name	Char	50	User's last name	Yes	No	None
	last_name	Char	100	User's email address	Yes	No	None
	password	Char	255	Hashed password for authentication	Yes	No	None
	role	Char	20	Role of the user (e.g., Admin, Student)	Yes	No	None
File	file_id	Integer	10	Unique identifier for uploaded file	Yes	Yes	None
	file_name	Char	255	Name of the uploaded file	Yes	No	None
	file_type	Char	20	Type of the uploaded file (e.g., PDF)	Yes	No	None
	upload_date	Date	-	Date the file was uploaded	Yes	No	None
	user_id	Integer	10	ID of the user who uploaded the file	Yes	No	None
Quiz	quiz_id	Integer	10	Unique identifier for the quiz.	Yes	Yes	None
	questions	Text	Unlimited	List of questions in the quiz.	Yes	No	None
	file_id	Integer	10	ID of the file the quiz is based on.	Yes	No	file.file_id
	user_id	Integer	10	ID of the user who generated the quiz.	Yes	No	user.user_id
Flashcard	flashcard_id	Integer	10	Unique identifier for the flashcard	Yes	Yes	None
	content	Text	Unlimited	Content of the flashcard	Yes	No	None
	file_id	Integer	10	Content of the flashcard	Yes	No	file.file_id
	user_id	Integer	10	ID of the user who owns the flashcard	Yes	No	user.user_id
Study Schedule	schedule_id	Integer	10	Unique identifier for the schedule	Yes	Yes	None

	time_blocks	Text	Unlimited	Time blocks allocated for studying	Yes	No	None
	preferences	Text	Unlimited	User preferences for study schedule	No	No	None
	user_id	Integer	10	ID of the user the schedule belongs to	Yes	No	user.user_id
Notification	notification_id	Integer	10	Unique identifier for the notification	Yes	Yes	None
	message_content	Text	Unlimited	Content of the notification	Yes	No	None
	delivery_status	Char	20	Status of the notification delivery	Yes	No	None
	user_id	Integer	10	ID of the user receiving the notification	Yes	No	user.user_id



4 TECHNICAL SYSTEM OPTION

4.1 TECHNICAL SYSTEM ARCHITECTURE

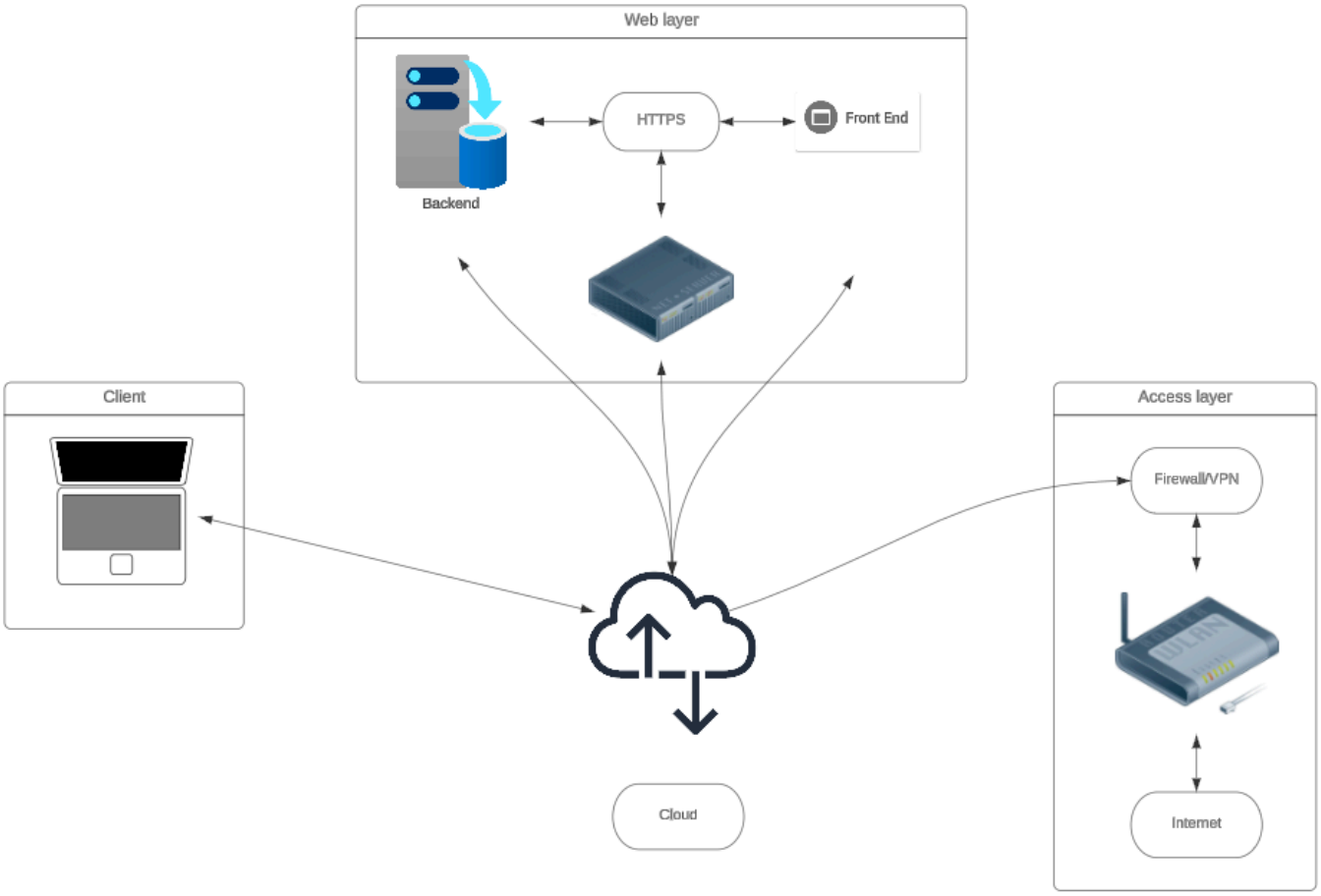


The diagram represents the technical architecture for Astreon and shows the interaction between Frontend, Backend, the AI Model, the Database, and the Deployment infrastructure. It follows a clear flow of information and responsibilities across the system.

The frontend section uses react framework in which it fetches the response from the backend api points

The backend uses django and django which handles the business logic while the django rest framework creates RESTful API endpoints to be fetched by the frontend. The AI model functionalities are included in the diagram

The database and deployment section depicts the database that is being used and what are the data to be stored. For the deployment, AWS platform will be used to deploy the application and store the static files.



4.1.1 Network Architecture

The diagram represents a three-layered architecture that includes the Client Layer, Web Layer, and Access Layer. It illustrates how data flows between the different components of your system, with communication happening securely over the cloud.

Client Layer

-Represents the end user devices (e.g laptop, computers)

Web Layer

- Core of the system where the application resides (frontend, backend, api communication)

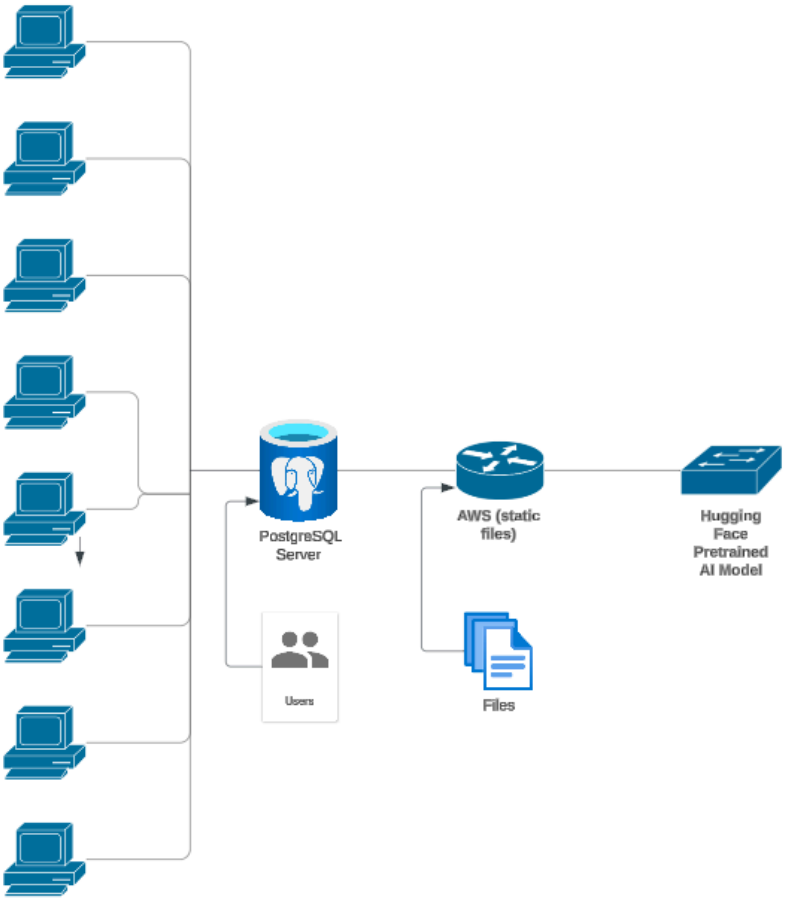
Access Layer

-This is the layer that handles the network security and the connection to the internet

-Firewall/VPN: Protects the network by filtering incoming and outgoing traffic

-Internet: Represents the broader network through which the client connects to the application.

4.1.2 Storage Architecture



When a client sends a request, such as a query or an uploaded file, the system processes it and stores relevant details in the PostgreSQL database to ensure that the session is persisted. This allows the system to maintain continuity, ensuring the client's request is not lost and can be referenced later if needed. The database stores session information, such as a unique session ID, user data, and metadata related to the query. On the other hand, files (like documents or images) that the client uploads are stored separately in AWS, which helps optimize the database by preventing large file storage directly in the database. Storing files in AWS also makes the retrieval process more efficient and scalable. Finally, the AI model, which is pretrained and can be fine-tuned, is accessed via Hugging Face. This enables the system to provide personalized responses and improvements over time by continuously fine-tuning the model based on user interactions and queries, ensuring better performance and a more accurate, responsive system.

4.2 IMPACT ANALYSIS

Current State :

The current state of Astreon provides a personalized AI study assistant and personalized support, however these are only limited features as for the mvp. These features include the ai chatbot , generation of quizzes, flashcards and the generation of schedule with a user summary to track their progress overtime

Future Enhancements :
The team are planning to cater different types of learners by providing a much more dynamic content based on the type of learner they are. Along with that, the learning paths for each user will be implemented to further provide a clear roadmap to help them optimize their learning

Effects on organization and staffing levels:

- *Improved user engagement:*

  The enhanced system will lead to enhanced user engagement since this will be focused on user input and cooperation to provide them a better experience

- *Market position:*

  Since astreon is planning to add a feature enhancement to cater different type of learners.Hence, it will be able to attract market lead and more users

Staffing Requirements:

Additional technical staff:

To implement new features effectively, we need to have more staff with the expertise that they are able to implement the said future enhancement

Significant Changes in User Operating Procedures:

Enhanced personalization:

With the introduction of dynamic based content based on the learner type, user will be able to get much more tailored content, in which this would require them to interact with the system differently

User onboarding:

Since the team need to assess which type of learner the user will be the user onboarding will change and we might need to have to conduct a survey to further meet the objective of the system

Potential Challenge

Learning curve:

With every additional features that are being implemented with astreon the learning curve therefore becomes more steeper in which this will become a great challenger for the developer to which the team will need to cater the user needs to build an app that will be intuitive

Implementation consideration:

Training Needs users:

A detailed user guide or tutorial will be necessary to assist users in understanding and making full use of the dynamic content and learning paths.

Savings on Replaced Equipment and Associated Costs:

- *AI pretrained model:*

  Utilizing the use of an API or an AI pretrained model will help the team to save on lots of resources, since we don't have to start from scratch to train a model in which it will cost a lot of money and computing resources.

- *Cloud Infrastructure:*

  Migrating static files to AWS storage will reduce the need for local server resources, resulting in savings on hardware and maintenance costs.

Risk Analysis:

- *Delays in Feature Implementation:*

  There may be delays in rolling out the enhanced features, especially if additional resources or more complex integrations are needed.

- *Scope:*

  As new ideas are introduced (e.g., personalized content or learning paths), there could be an expanding scope of the project, leading to delays and resource strain.

- *Data Privacy and Security:*

  As new user data (e.g., personalized learning paths) is collected and stored, there may be increased concerns about data privacy and security.

- *System Downtime: Integrating new features like dynamic content and AI fine-tuning could lead to system downtime if not properly managed and tested during implementation.*

4.3 IMPLEMENTATION PLAN

1. ***Requirements Gathering*** *(2 Weeks)*

- *Description:*

  Identify specific functional and non-functional requirements for dynamic content delivery, learning paths, and progress tracking. Conduct interviews with stakeholders (students, faculty, and admin) to clarify needs and expectations.
- *Contingency Plan: If requirements are unclear or delayed, iterative refinements and prototype reviews will ensure alignment.*

2. ***Analysis and Design*** *(3 Weeks)*

- *Description: Design system architecture, data models, and API interactions for dynamic content, learning paths, and AI model integration. Focus on optimizing storage through AWS and database interaction.*
- *Contingency Plan: Create mockups and prototypes to validate design decisions with stakeholders.*

3. ***Development*** *(12 Weeks)*

- *Description:*
- *Develop modules for dynamic content delivery tailored to learner types.*

  Implement AI fine-tuning using pre trained Hugging Face models/gemini api.
- *Create user-specific learning path algorithms.*
- *Store static files on AWS and optimize PostgreSQL database interactions.*
- *Contingency Plan: Development will be divided into sprints to identify and resolve blockers early. Critical features will be prioritized to ensure minimum viable product (MVP) completion.*

4. ***Testing*** *(3 Weeks)*

   Description: Conduct unit testing, integration testing, and user acceptance testing (UAT) to ensure system performance and reliability. Engage selected users to validate new features.
- *Contingency Plan: Buffer time allocated for resolving critical bugs or functionality gaps.*

5. ***Deployment*** *(1 Week)*

- *Description: Incrementally deploy features to minimize downtime. Conduct monitoring to ensure a smooth transition.*
- *Contingency Plan: Rollback strategy in place for unexpected system failures or user issues.*

6. ***Documentation*** *(2 Weeks)*

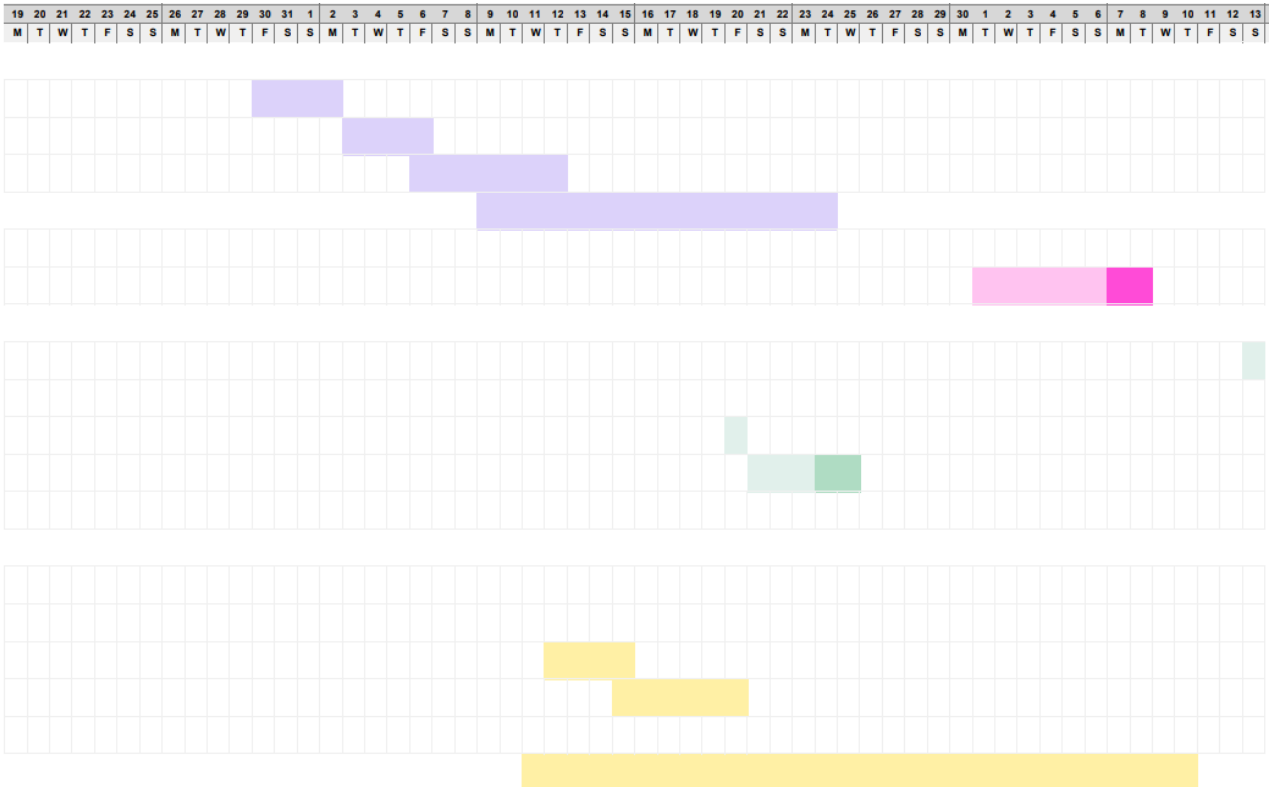
- *Description: Prepare comprehensive documentation, including system manuals, user guides, and training materials for staff and end users.*
- *Contingency Plan: Use feedback from user training sessions to refine documentation.*

7. ***Meetings and Review*** *(Ongoing)*

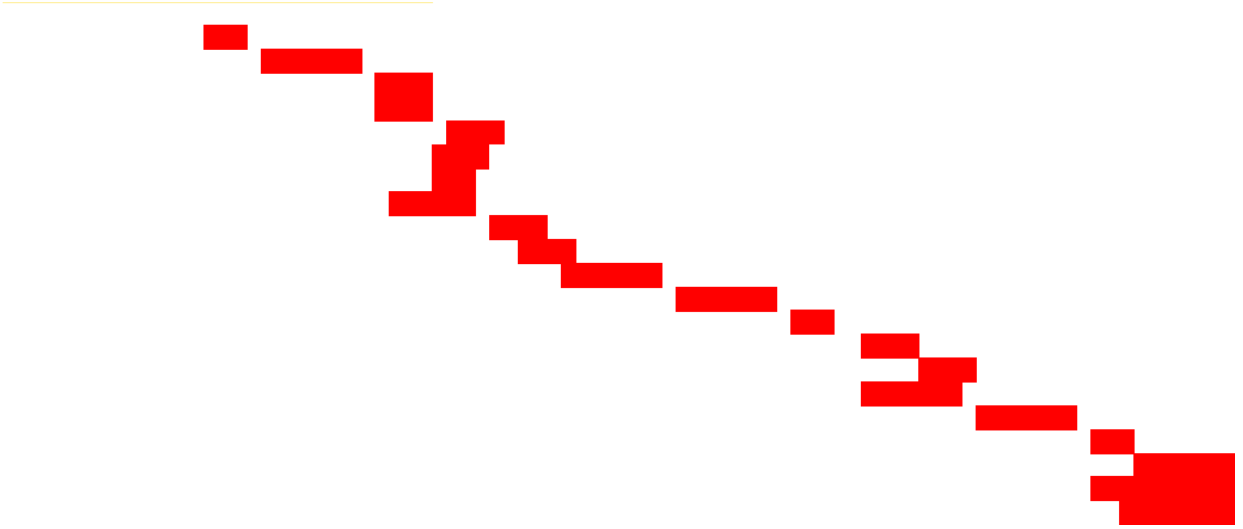
- *Description: Conduct weekly progress meetings to assess status, identify challenges, and ensure alignment with goals. Post-deployment review to collect user feedback and optimize system performance.*

<Gantt Chart Diagram.>

TASK		PROGRESS	START	END
Project Proposal Phase				
Project Proposal Document		100%	8/30/24	9/2/24
Project Title Proposal Approval		100%	9/3/24	9/6/24
Project Title Finalization		100%	9/6/24	9/12/24
Request Letter Creation and Approval		100%	9/17/24	10/1/24
Requirement Gathering/Analysis Phase				
Survey Gathering		75%	10/1/24	10/8/24
Initial List of Requirements		100%	10/9/24	10/12/24
Requirement analysis		100%	10/13/24	10/16/24
SRS Development Phase				
Draft Submission		100%	9/20/24	9/20/24
Document Review		60%	9/21/24	9/25/24
SRS Presentation		50%	10/31/24	10/31/24
Document Revision		25%	11/15/24	11/20/24
Testing and validation		25%	10/31/24	11/4/24
Design Phase				
Wireframing	Hans	100%	9/12/24	9/15/24
UI/UX Design Inital Phase	Hans	100%	9/15/24	9/20/24
ERD Designing	Hans	100%	11/10/24	12/13/24
UI/UX Enhancement	Hans	100%	9/20/24	11/17/24



Development Phase				
Project environment setup	All	100%	9/25/24	9/28/24
Implement user auth backend (django only)	Berna	100%	9/29/24	10/6/24
UI login/signup	Micah	100%	10/7/24	10/10/24
Implement url endpoint	Berna	100%	10/7/24	10/10/24
UI login code revisions	Micah	100%	10/12/24	10/15/24
Backend user auth code revisions	Berna	100%	10/8/24	10/11/24
Integrate frontend with frontend	Berna	100%	10/11/24	10/13/24
Set up docker	Hans, Micah, Be	100%	10/8/24	10/13/24
UI login code revisions	Micah	100%	10/15/24	10/18/24
Backend login revisions	Berna	100%	10/17/24	10/20/24
UI for chatbot page	Micah	100%	10/20/24	10/26/24
Intergrate UI with React	Micah	100%	10/28/24	11/3/24
Add email confirmation to auth	Hans	100%	11/5/24	11/7/24
Models for custom user, file/image uplaods,	Hans	100%	11/9/24	11/12/24
Implement restful apis for user auth	Hans	100%	11/13/24	11/16/24
Integrate frontend with backend with email v	Hans	100%	11/9/24	11/15/24
Create sidebar and header components	Micah	100%	11/17/24	11/23/24
Create models for session, quiz, flashcards	Berna	100%	11/25/24	11/27/24
Create rest api endpoints for session, quiz fl	Micah	100%	11/28/24	12/5/24
Create ui for session, quiz, flashcards,calend	Micah	100%	11/25/24	12/5/24
Integrate backend with frontend	Berna	100%	11/27/24	12/5/24



Key

AAA	***System Name***
SRAA	***Security Risk Assessment and Audit***
PIA	***Privacy Impact Assessment***
HW	***Hardware***
SW	***Software***
DR	***Disaster Recovery***