Software Design Specification (SDS)

Project Name: Basmagly
Prepared By: Mohamed Hatem 202001235 - Kareem Elozeiri 202000896
Date: 8/11/2024

---

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) outlines the requirements for the Student Study Assistant Website “Basmagly” (Release 1.0). The product provides a platform for students to enhance their study experience by uploading documents and leveraging features like question-answer generation, quizzes, summaries, and context-aware chats.

1.2 Scope

The product provides a platform for students to enhance their study experience. The full scope of the system, including both frontend (React-based) and backend (Django-based) components, as well as the integration of key technologies like Hugging Face models and LangChain for natural language processing and orchestration. The document addresses all functional and non-functional requirements necessary for implementing and deploying the complete system.
. The software will perform the following major tasks:

- uploading documents
- leveraging features like question-answer generation and summaries
- generating quizzes
- Chatting for clarification

---

2. System Overview

The system consists of the following components:

- **Frontend**: React.js
- **Backend**: Django
- **Database**: PostgreSQL

---

3. System Architecture

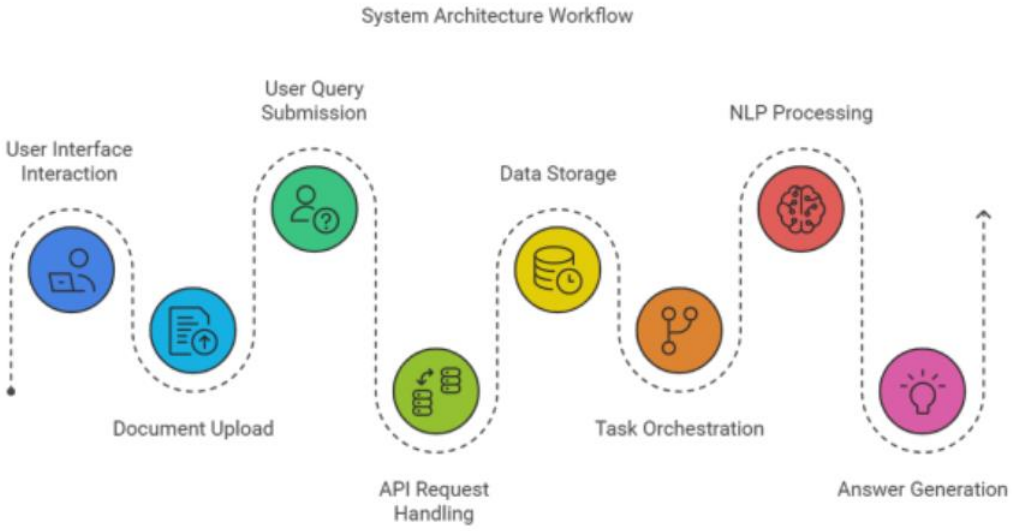
3.1 Architectural Design

This project follows the client-server, microservices architecture, where:

- **Frontend** communicates with the backend using API.
- **Backend** interacts with the database to manage and retrieve data.

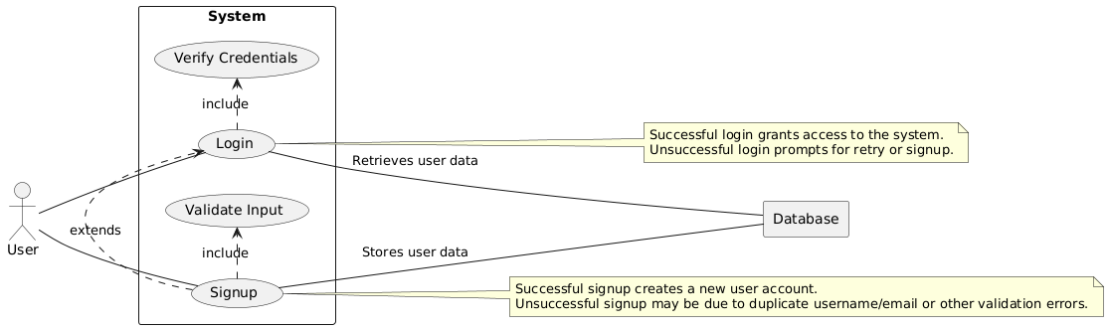
3.2 Data Flow

1. **User Interaction**: The user interacts with the UI to perform an action (e.g., create a new entry, update an existing record).
2. **Request Processing**: The frontend sends an API request to the backend server.
3. **Data Handling**: The backend processes the request, interacts with the database, and fetches or updates the necessary data.
4. **Response**: The backend sends the response back to the frontend, updating the UI.

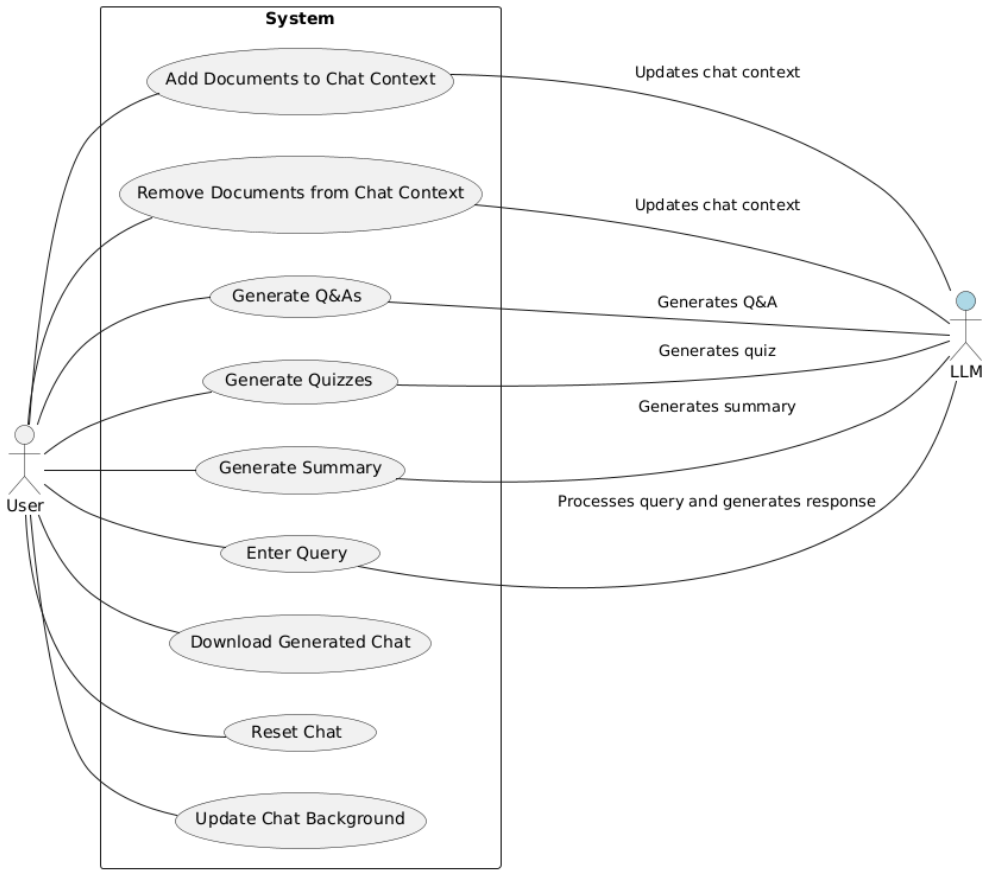


3.3 Use case diagram:

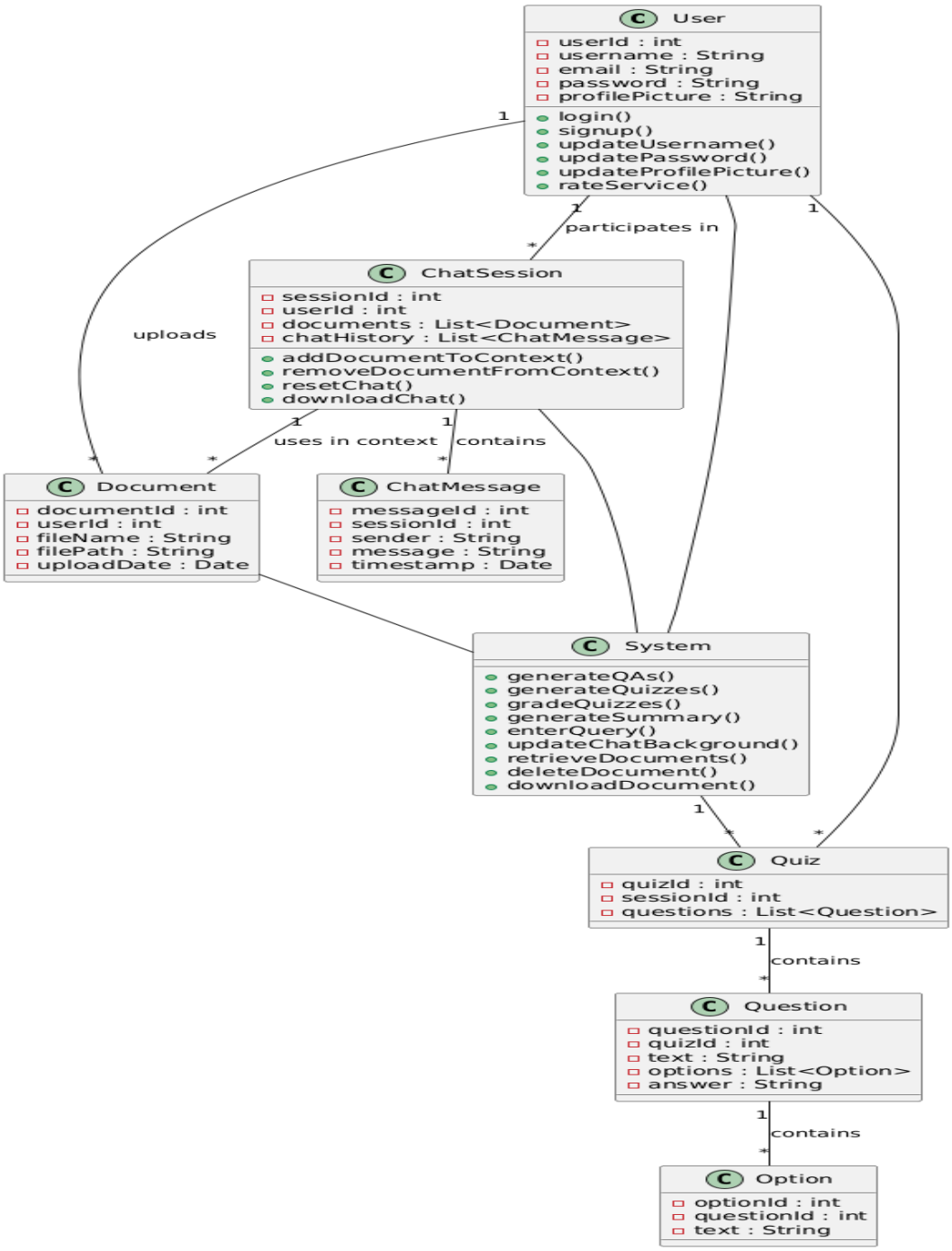
Sign in FR:



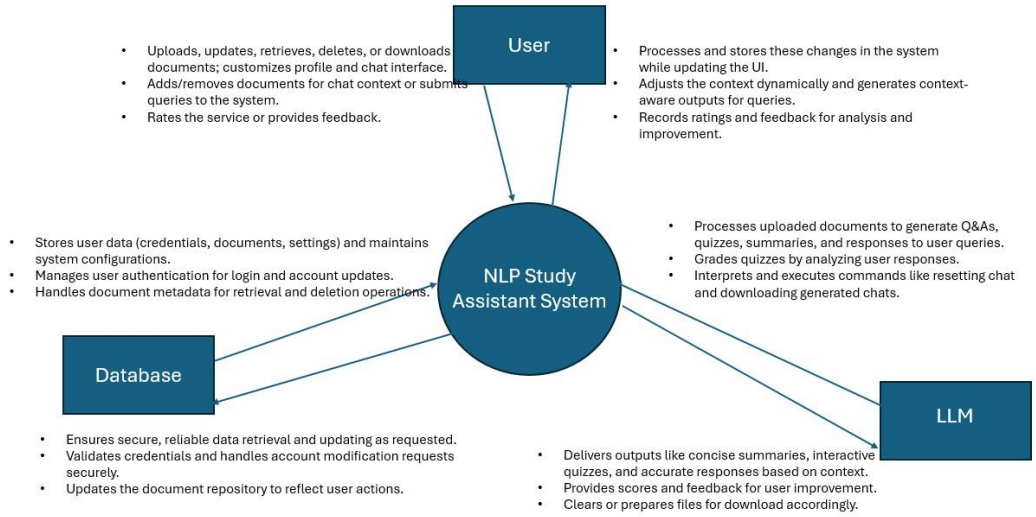
Chatting FR:


System
Add Documents to Chat Context
Remove Documents from Chat Context
Generate Q&As
Generate Quizzes
Generate Summary
Enter Query
Download Generated Chat
Reset Chat
Update Chat Background
User
LLM
Updates chat context
Updates chat context
Generates Q&A
Generates quiz
Generates summary
Processes query and generates response

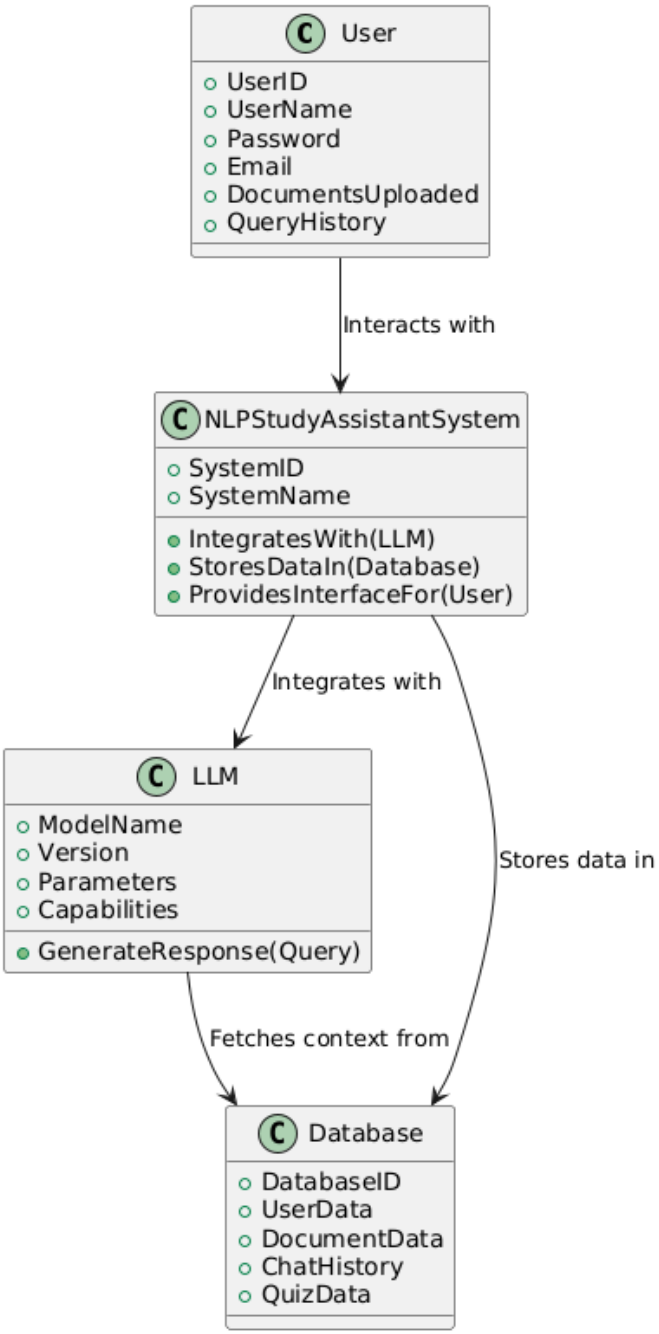
3.4 Class Diagram:



3.5 Context diagram:

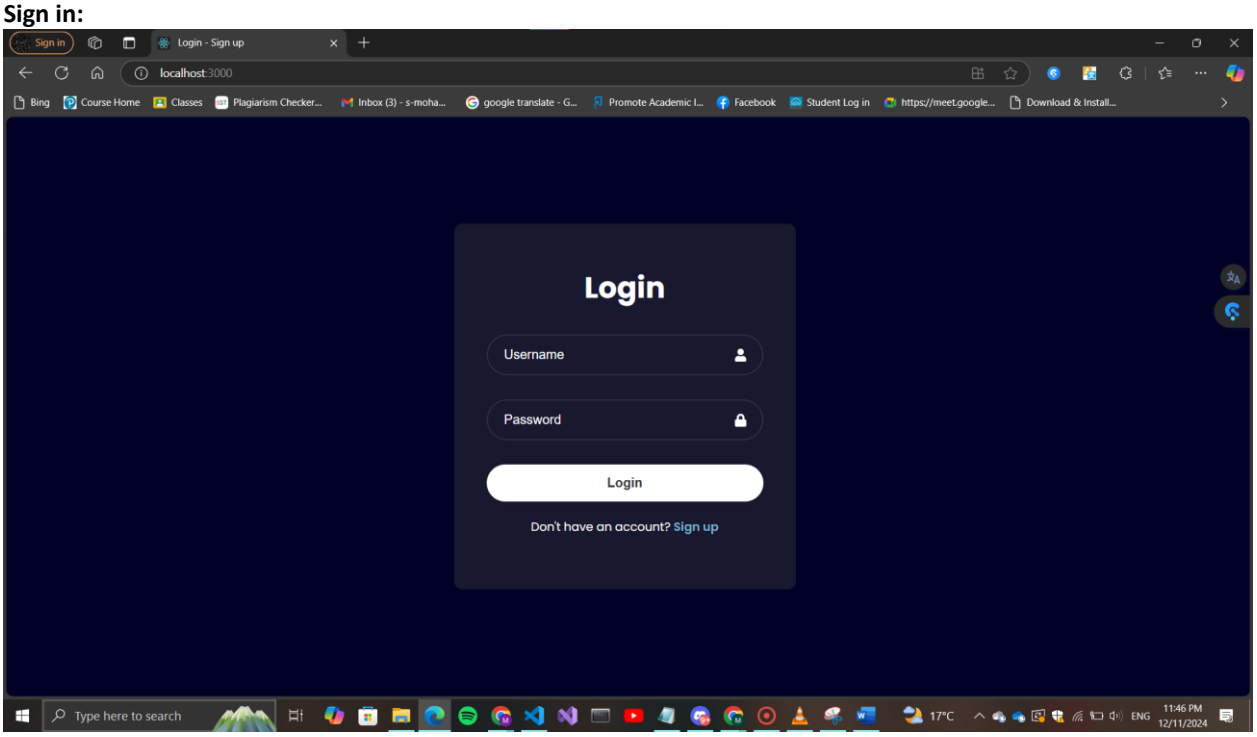


3.6 Concept Diagram:

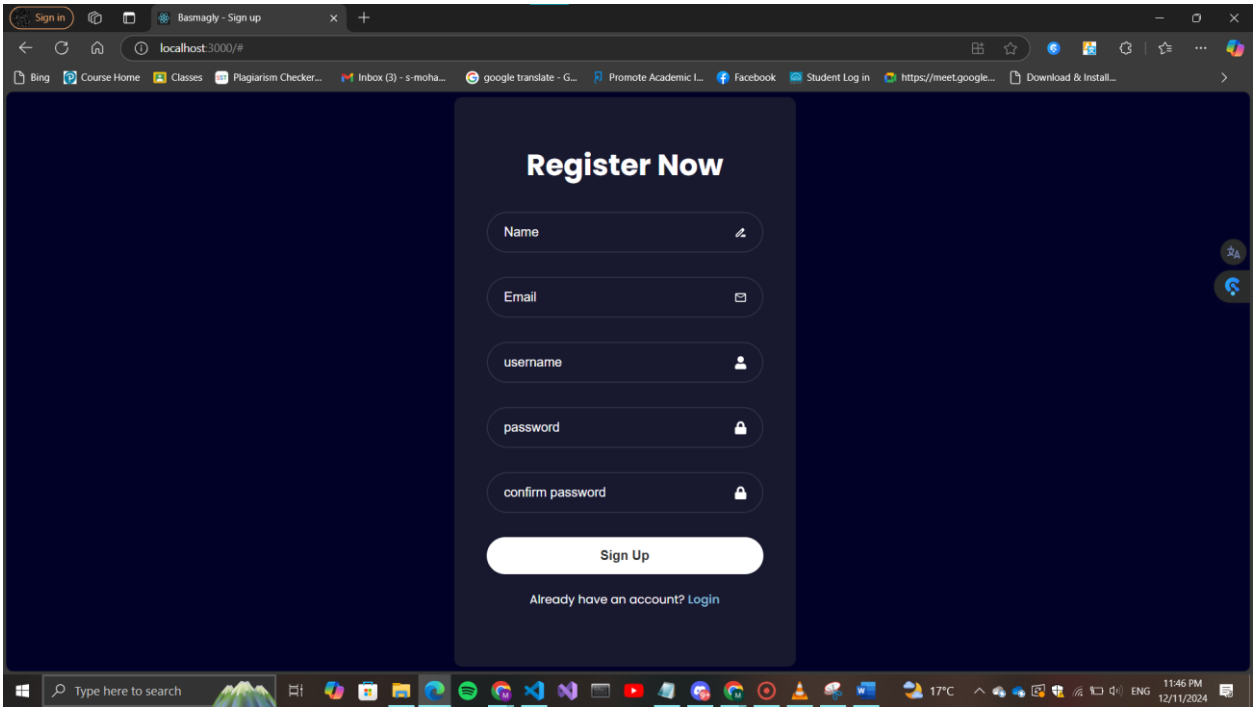


3.7 GUI Design:

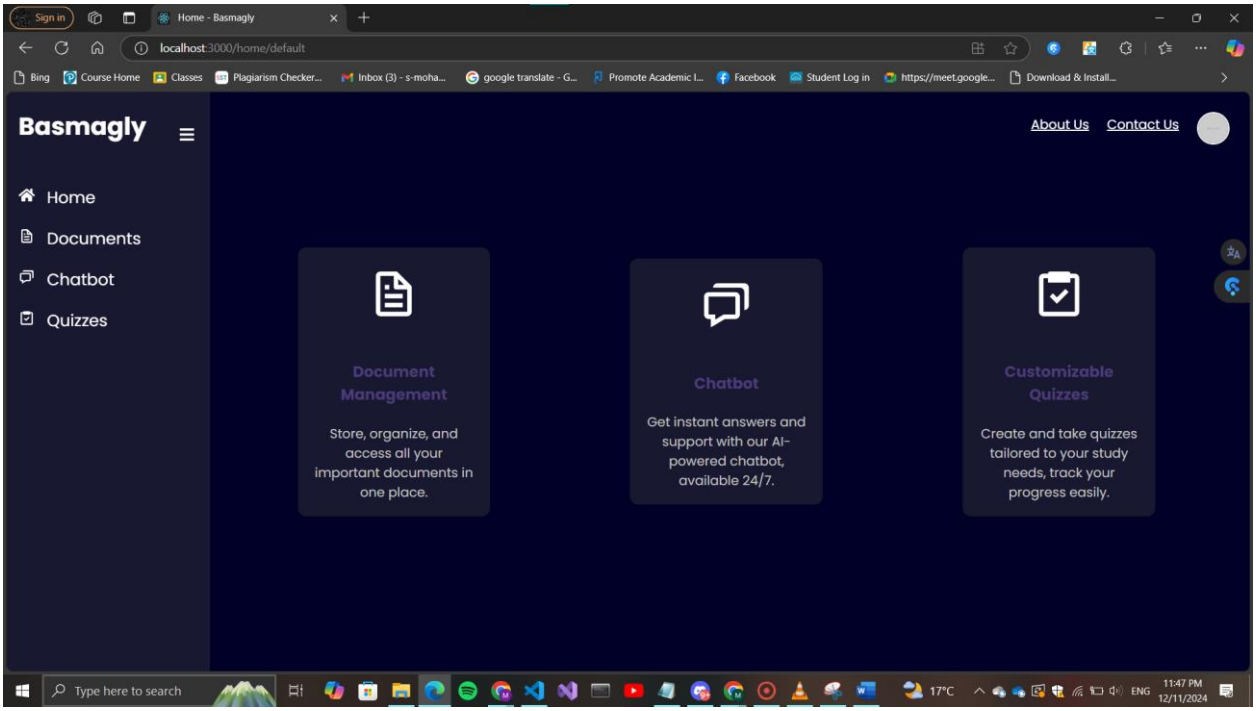
Sign in:



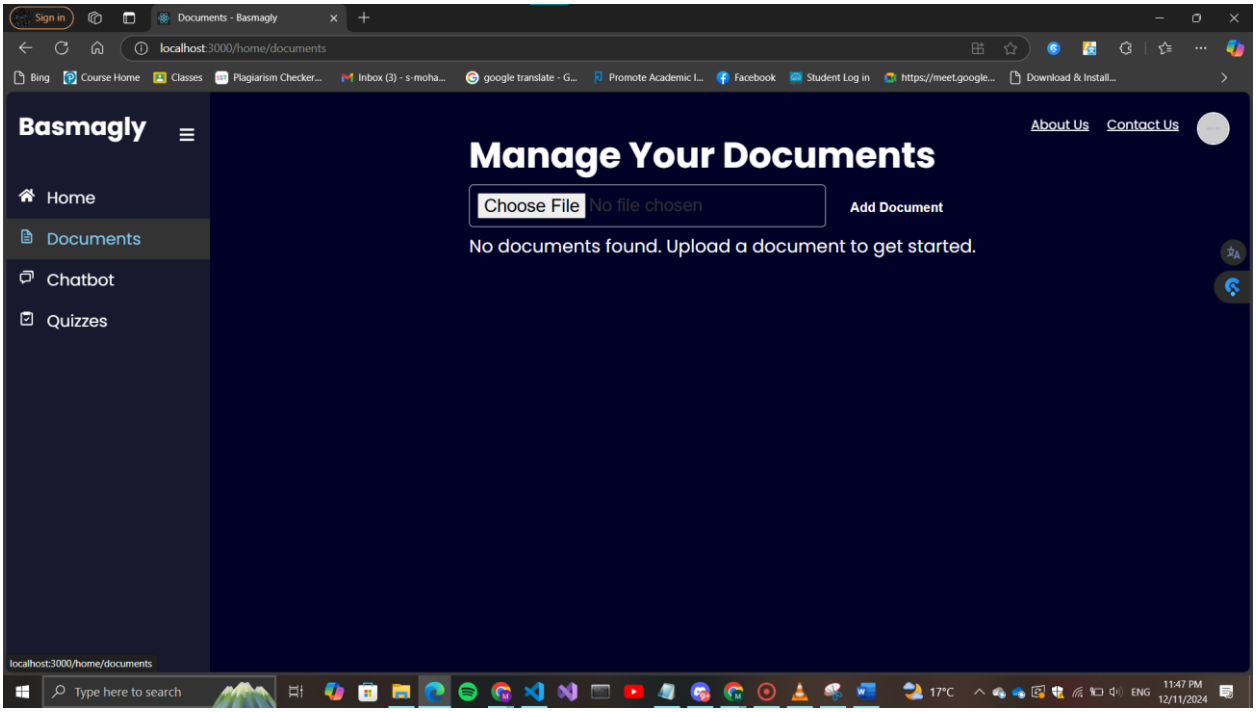
Sign up:



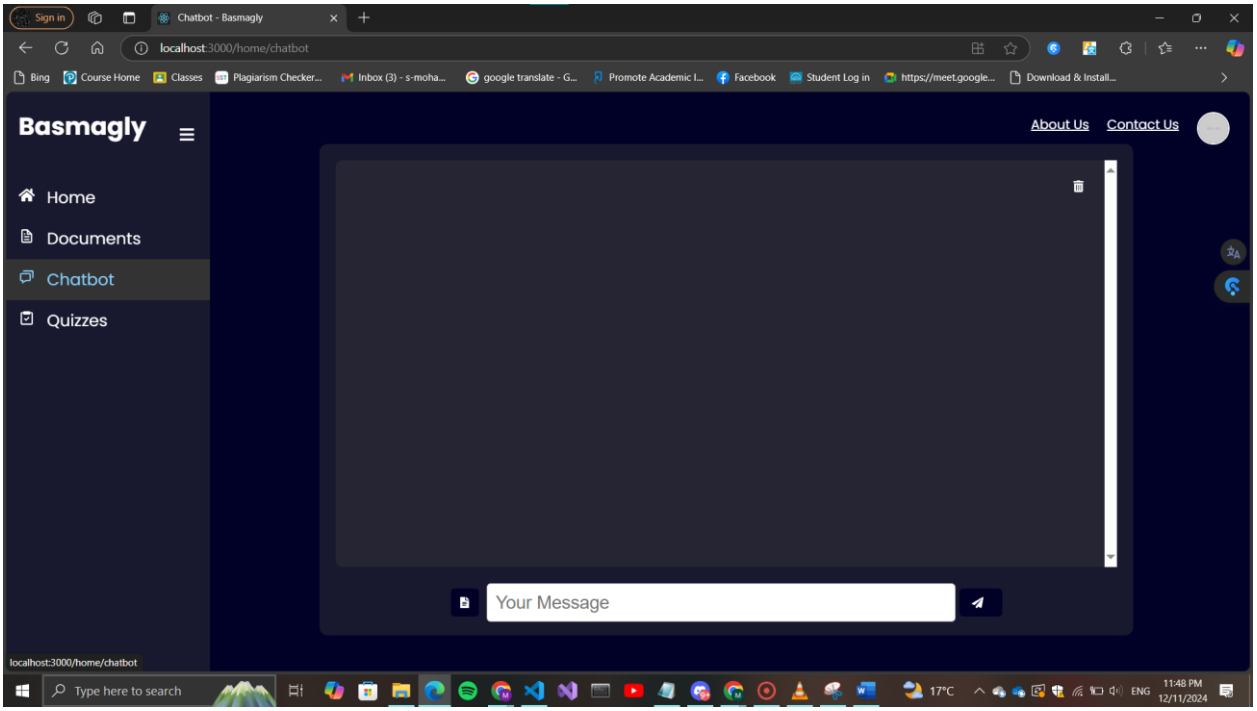
Home Page:



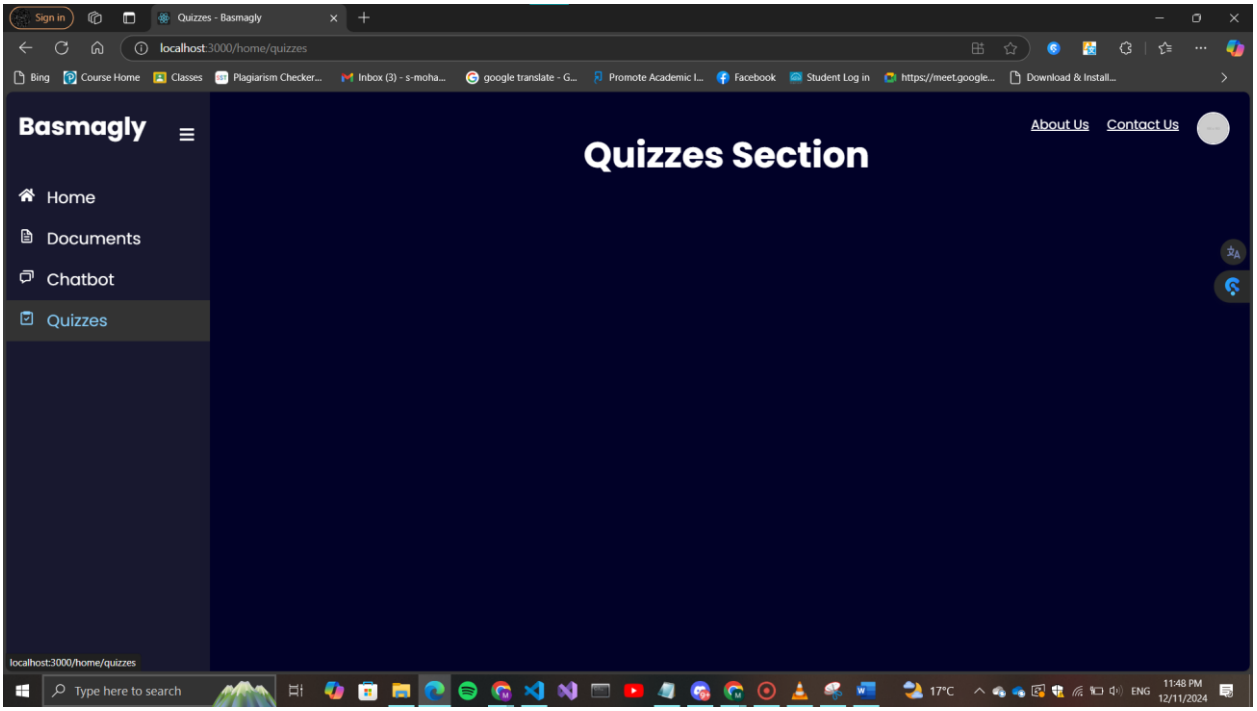
Document Management Page:



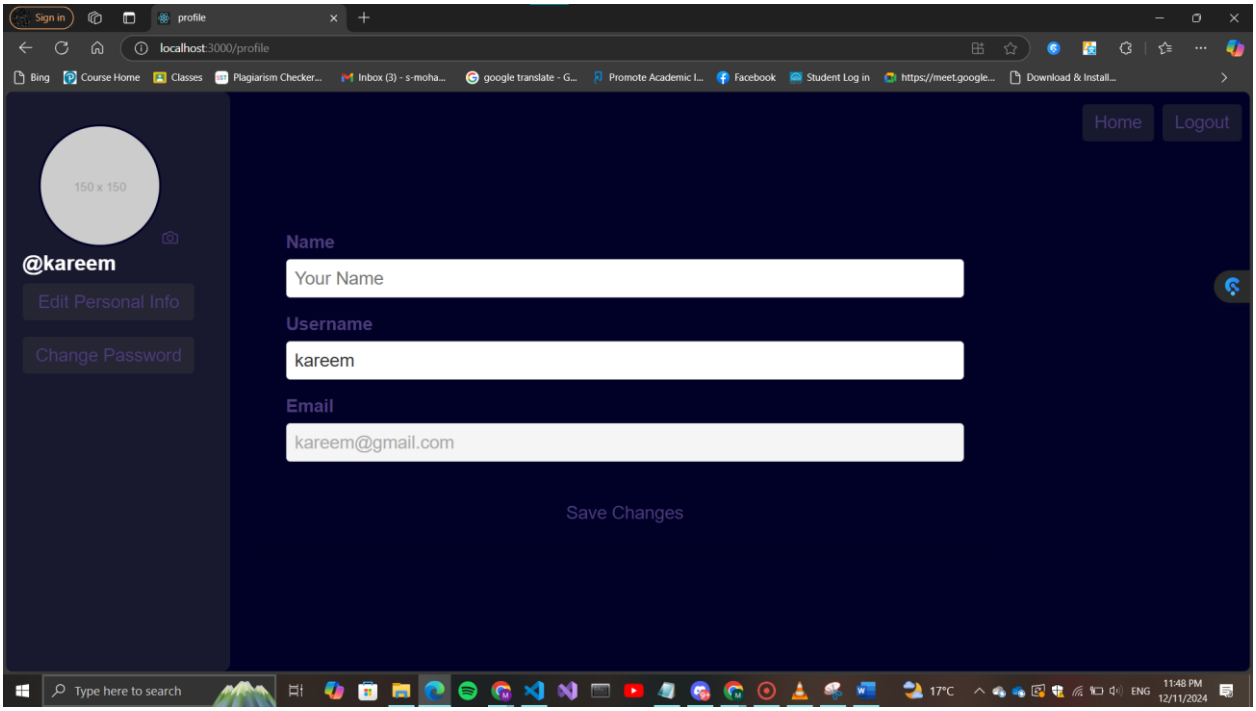
Chatbot Page:



Quizzes Page:



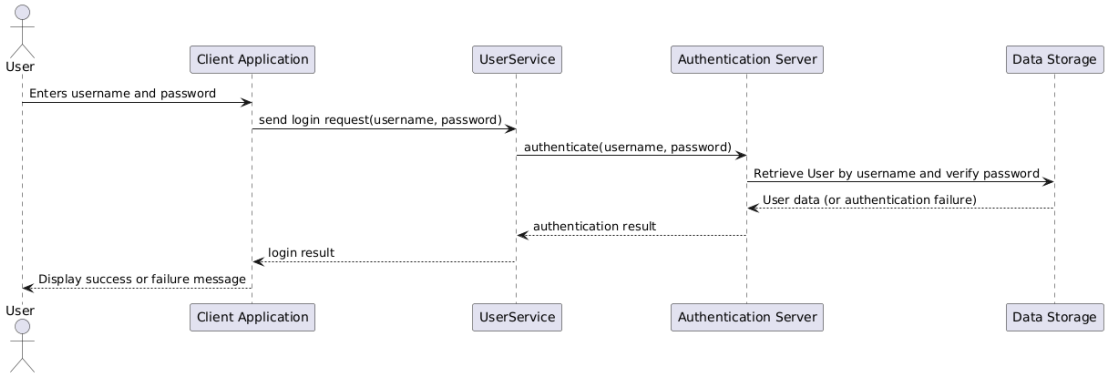
Profile Page:



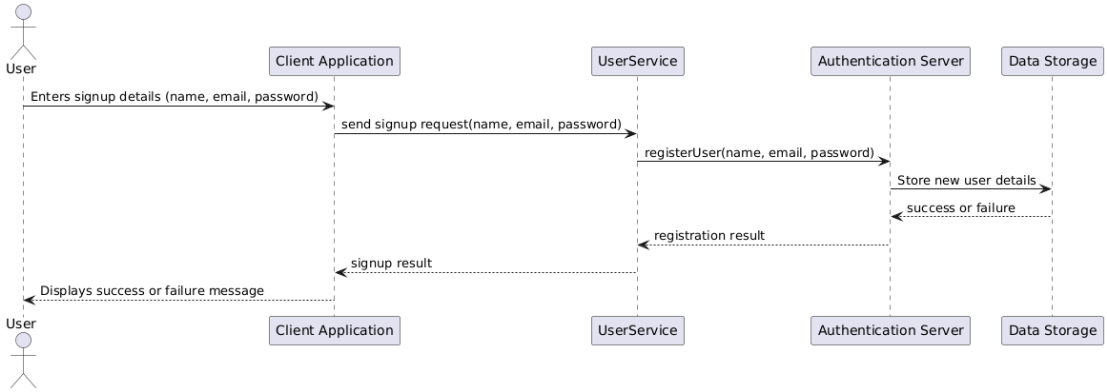
---

5. Sequence Diagrams:

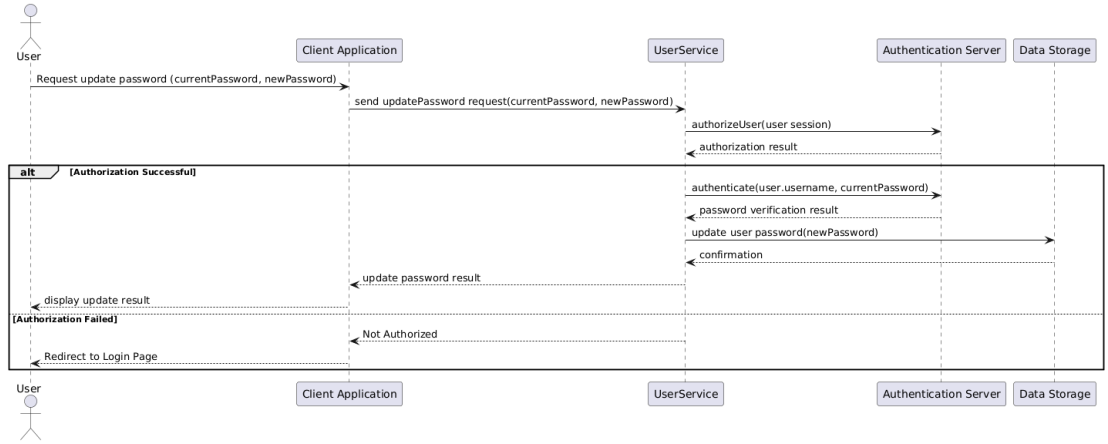
- **Login:**



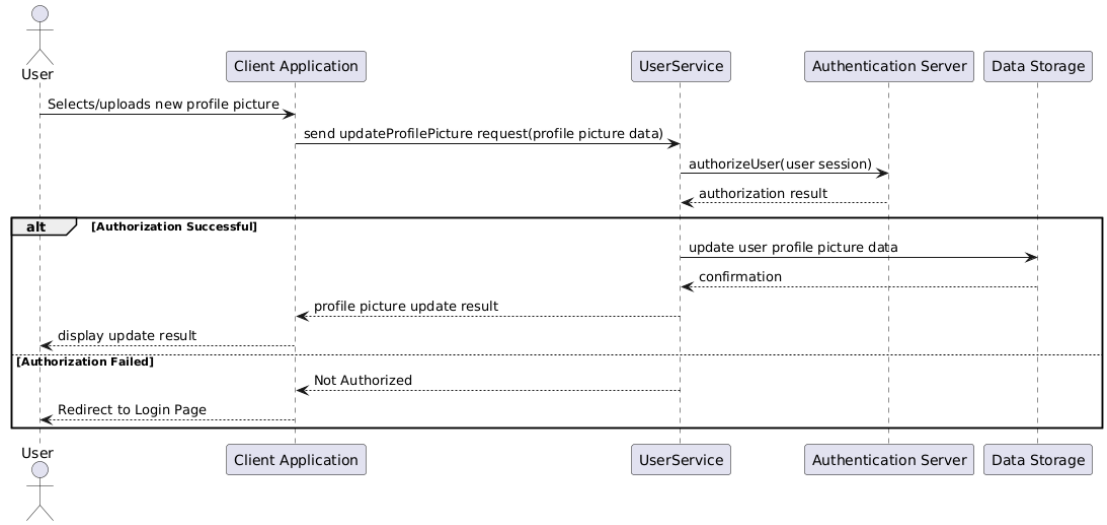
- **Sign up:**



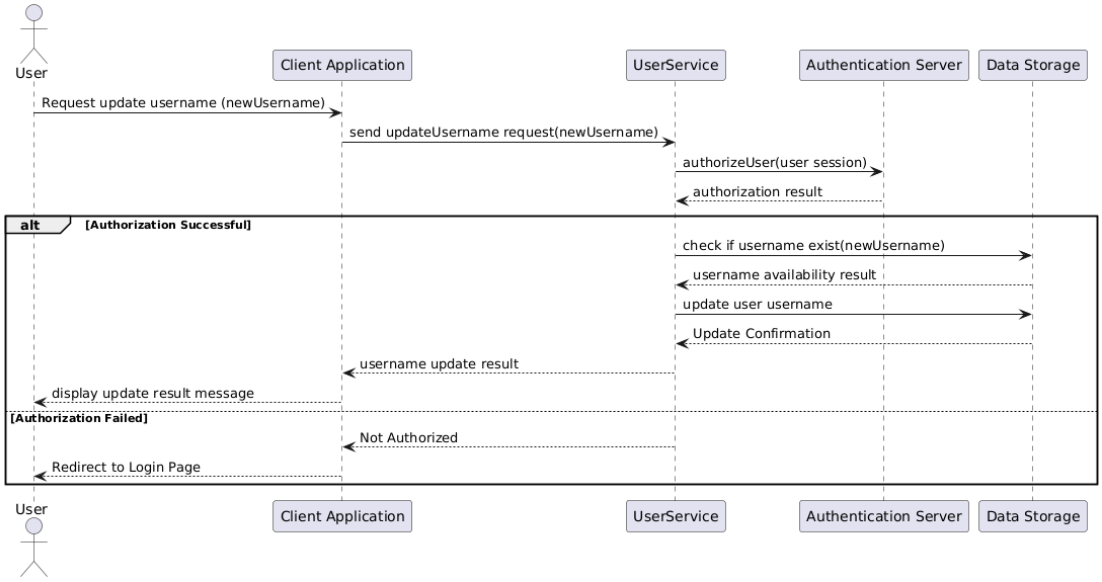
- **Change Password:**



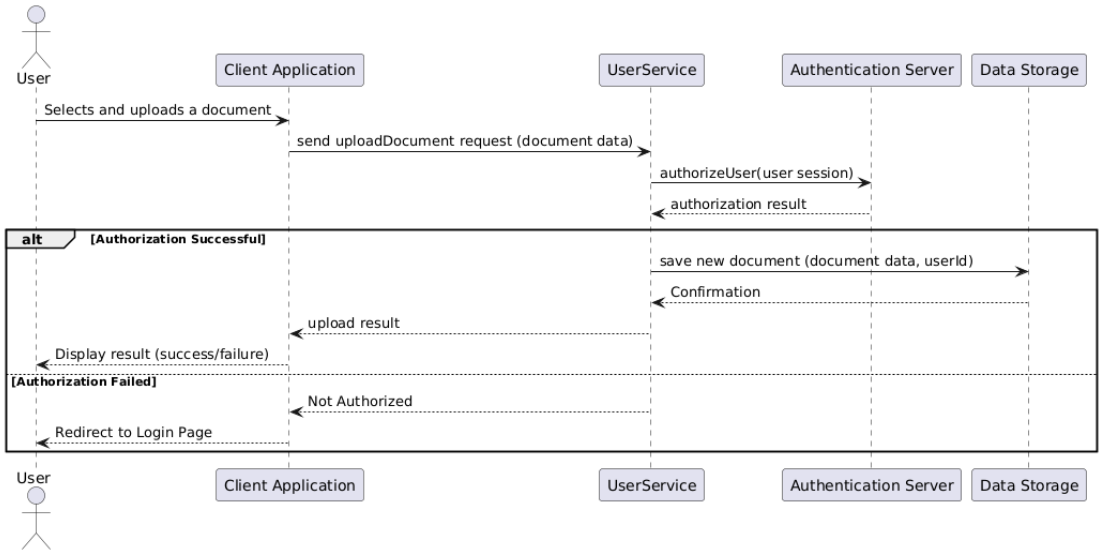
- **Update profile photo:**



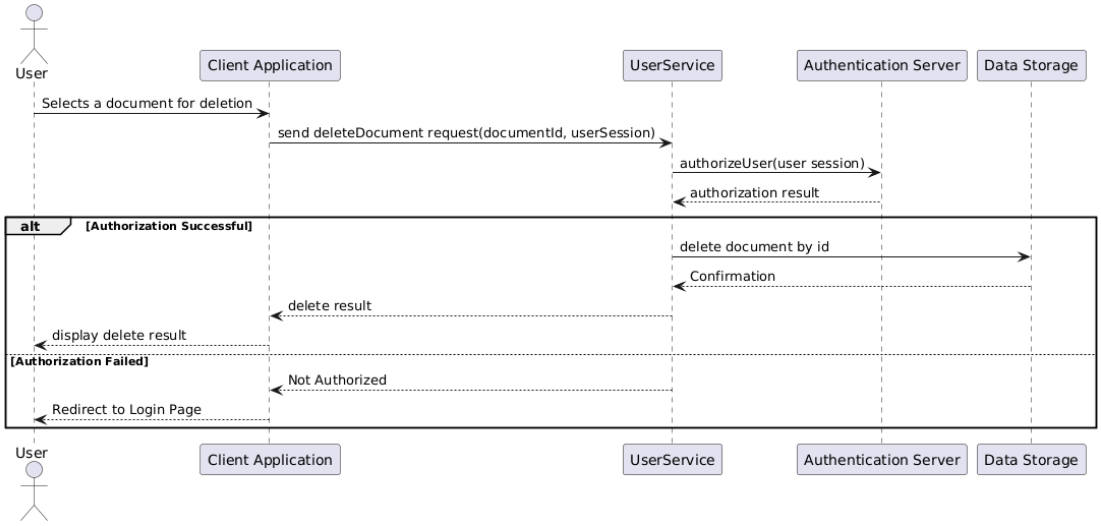
- **Update Username:**



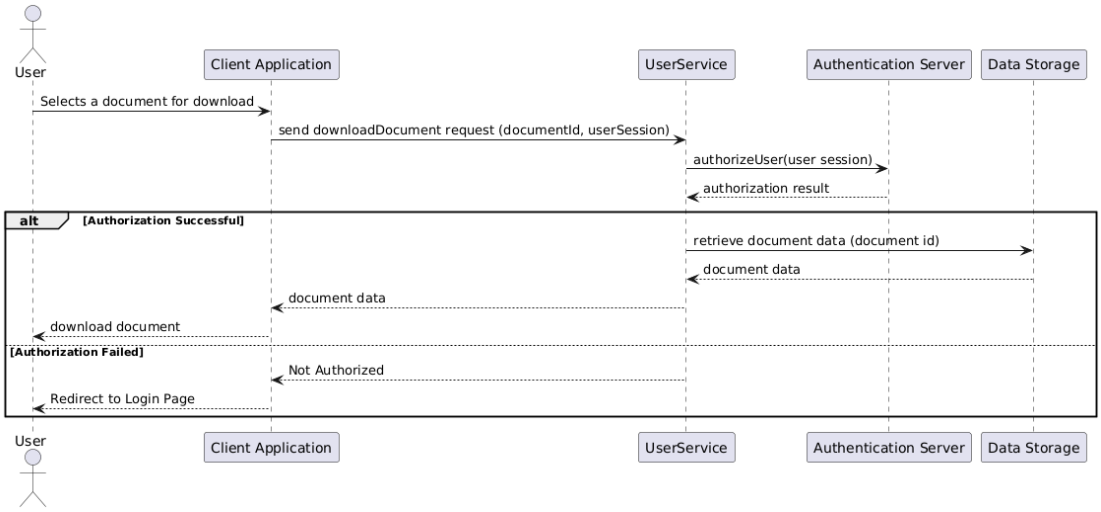
- **Upload Document:**



- **Delete Document:**



- **Download Document:**



5. Technology Stack

- **Frontend**: React.js
- **Backend**: Django
- **Database**: PostgreSQL
- **Hosting**: AWS or Google Cloud

---

7. Conclusion

The Basmagly is designed to fulfill the specified functional and non-functional requirements as described in this SDS. The design outlined here will ensure that the system is robust, scalable, and user-friendly, providing the intended value to its users.