AI Proctor

Submitted By

Tasrif Nur Himel

221-35-1078

Supervised By

Dr. Imran Mahmud

Professor & Head

This project report has been submitted in fulfilment of the requirements for the degree of **Bachelor of Science in Software Engineering**

Department of Software Engineering

Faculty of Science and Information Technology

Supervisor Approval Form

Fall 2025	B.Sc. In SWE	Campus: DSC

Student Name	Student ID
Tasrif Nur Himel	**221-35-1078**

Project/Thesis Information	
Project/Thesis Title	AI Proctor – AI Powered Proctorial Service System
Type of work	Artificial Intelligence

Supervisor information	
Supervisor Name	**Dr. Imran Mahmud**
Supervisor Initial	**IM**
Completed Credit till now	139
How many credits in this semester	9
Amount (Due)	~~5000~~ 0.00
Supervisor Consent	☑ Yes ☐ No

Supervisor Signature

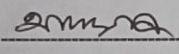
APPROVAL

This thesis titled on "**AI Proctor**", submitted by **Tasrif Nur Himel (ID: 221-35-1078)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

Dr. Imran Mahmud
Professor & Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman

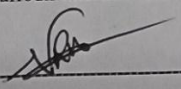


Afsana Begum
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

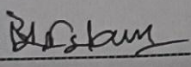
Md. Shohel Arman
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2



Nadira Islam
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 3



Md Manowarul Islam
Professor
Department of Computer Science and Engineering
Jagannath University, Bangladesh

External Examiner

Accounts Clearance

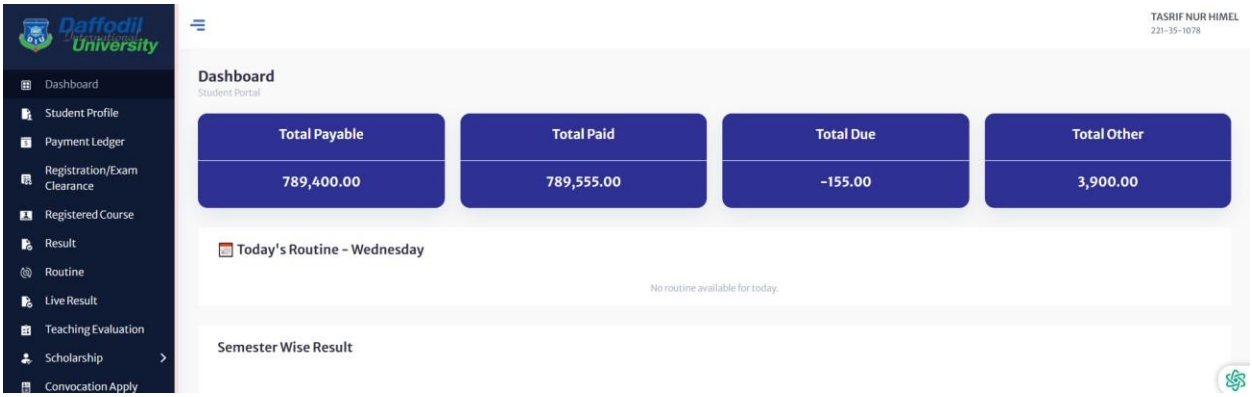


Table of Contents

Chapter 1 1

Introduction 1

Scenario Writing 2

Scenario-1: Student Reporting an Issue After Office Hours 2

Scenario-2: Admin Managing Policy Documents 3

Scenario-3: Admin Scheduling Meeting 3

Stakeholder 3

User Profile 4

User Profile-01: Student 4

User Profile-02: Proctor Authority 5

User Profile-03: Admin or IT 6

Tentative Elicitation Process 7

Scope 8

Project Planning and Initiation 9

Feasibility Study 10

1. Phase 1: Preliminary Analysis & Project Scope Definition 10

2. Phase 2: Market Feasibility Analysis (Market Research) 11

3. Phase 3: Technical Feasibility Analysis 12

4. Phase 4: Financial Feasibility Analysis 13

Chapter 2 16

Software Requirement Specification 16

SRS 16

Project Scheduling 21

User Case Diagram 22

User Case Description 23

Case Description-01: Registration 23

Case Description-02: Sign in 24

Case Description-03: Start Chat 25

Case Description-04: Start Voice 26

Case Description-05: Select Language 27

Case Description-06: Get Subscription 28

Case Description-07: Upload Document 29

Case Description-08: Schedule Meeting 30

Case Description-09: History 31

Case Description-10: Notification 32

Case Description-11: Logout 33
Activity Diagram 34
Activity Diagram-1: Registration 34
Activity Diagram-2: Login 35
Activity Diagram-3: Chat 36
Activity Diagram-4: Voice 37
Activity Diagram-5: Upload Document 38
Activity Diagram-6: Get subscription 39
Activity Diagram-7: Meeting Fix 40
Sequence Diagram 41
Sequence Diagram-1: Registration 41
Sequence diagram-2: Login 42
Sequence diagram-3: Chat 43
Sequence diagram-4: Voice 44
Sequence diagram-5: Upload Document 45
Sequence diagram-6: Get subscription 46
Sequence diagram-7: Meeting Fix 47
Sequence diagram-8: Logout 48
Chapter 3 49
Software Testing 49
Chapter 4 53
User Manual 53
Chapter 5 57
Project Summary 57
Reference 58
Achievement 59
Contract 60

CHAPTER -1
INTRODUCTION

So, let's try and explain what the AI Proctor is all about, shall we? Envision this, please: a virtual assistant speaking both English and Bangla, designed exclusively for Proctor Offices in universities, and the purpose of this AI Proctor is to ensure students have someone or something to fall back on regarding their security or university policies at any time of day or night. Need assistance at midnight? Don't worry, you're covered. The AI Proctor has capabilities for both text and voice assistance, assistance with locating official documents, and even allows students to schedule a meeting with the Proctor Office staff, even when the office is closed. Therefore, essentially, what this does is make the university a little bit safer and a whole lot more welcoming, considering assistance is always at your fingertips.

Why, though, is it important to establish this system? In other words, what is the large point of it? It is important to make sure students aren't left in limbo simply because it is after hours, For instance, maybe it's just a question about a guideline or maybe it's a meeting to talk to the staff member, the point is, people shouldn't need to wait until the next day to receive an answer. By answering all those guideline questions and preparing them to meet, it literally puts Proctor Offices into the twenty-first century. It is important to recognize, however, the impact it's had on campus security.

The features of the AI Proctor system that we would like to discuss are:

- It talks or chats with you/your choice and it supports both Bangla and English languages, at any time you want.
- Also, arranging a meeting with someone from the Proctor's office is possible directly through the chat function. You no longer need to wait for office hours.
- Say you are searching for information concealed within a school policy and say that this information can be retrieved directly from PDFs contained within because of a nifty thing called Retrieval-Augmented Generation (or RAG, but you don't have to remember this).
- In order to run everything smoothly in both languages, it keeps info in 'vector databases' for Bangla and English separately. It means you will receive better results, regardless of the language you prefer.
- Each institution is assigned its own space within the system, and the personnel and

students will only see what is relevant to their institution.

- The admin panel is there as well. Employees are allowed to upload and update official files in either language at any time.

- Logins are secure and protected by JWT, which is just a techie term for saying your information will stay protected whether you're a student or an administrator.

And all the things you ask and say in chat and voice? Just in case you want to go back and check, you and the admins can always look at it later. -One of the biggest problems that comes along with these types of systems is keeping it simple and natural. So the goal of this AI would be to make it seem more like a friend and less like a stiff computer program. If you ever need advice, the AI Proctor would love to help.

Scenario Writing

When it comes to figuring out how something actually works, it's simplest for me to think about how it could possibly be used by people. Therefore, let's think about a few different situations in which the AI Proctor system actually kicks in and makes a difference. In the instance of the "AI Proctor," let's create examples:

Scenario 1: Student Reporting a Problem Outside Office Hours

Description

- The student accesses the AI proctor system at 10 PM after being harassed on campus.
- The student chooses the language that is preferred, which is Bangla, and then communicates the problem through chat/voice interfaces.
- The use of the AI assistant facilitates the gathering of crucial information from the policies of the proctor office.
- The student asks for a meeting with the proctor office using the chat interface.

Resulting Action/Effect: The student gets immediate support and is sent an email confirming the meeting request. The admin will then check the request the next morning and set up the meeting.

Scenario 2: Admin Manages Policy Documents

Description

- A person with administrative privileges logs into the administrative interface and uploads a new PDF document consisting of up-to-date discipline policies in both Bangla and English.
- The system automatically updates the respective vector databases for each language.
- The admin looks into the conversation history of a particular user to monitor their recent conversations with the AI helper.

Resulting Action/Effect: The policies are now retrievable via the AI assistant so that the latest information is provided to the users.

Scenario 3: Admin schedules a meeting

Description

- A student submits a chat message requesting a meeting and throws in their preferred date and time.
- Admin receives a notification through email and then receives an email alert to which the admin reacts by jumping into the admin panel.

The admin might not be able to fix the time exactly, so they reschedule and send another email to the student meeting which is confirmed, with a new time.

Outcome: The student receives a confirmation email with the new details of the meeting, so there is seamless communication.

STAKEHOLDER

The major stakeholders in the AI Proctor system will include:

- **Students:** "They are at the center of what I do." They're the ones who come to me to report a problem or ask for advice or a meeting.
- **Proctor Office Authority:** they involve themselves in discipline, review requests for meetings, and ensure adherence to college rules and regulations.
- **Admin or IT Team:** They are operating in the background to make sure that everything is running smoothly on the platform while performing the tech heavy lifting that

universities require.

USER PROFILE

User Profile-01: Student

Characteristic	Notes on Characteristics	Requirement implied
Type of User	Visitor/ Registered Student	User interface, verification
Age Range	18-30 years	Age verification may be required depending on regulations.
Frequency of Use	Weekly or occasional	Performance, Operation, Acceptance
Mandatory	Yes	System should prioritize ease of use
Computer Experience	Not Mandatory	User interface, documentation training
Education	Varies (Undergraduate to Postgraduate)	Language simplification for clarity
Goals	Quick resolution of issues, access to policies	Focus on a smooth interaction
Language Skills	Bangla, English	Bilingual support
Number of Users	High Volume	System should be scalable and handle concurrent user load efficiently.
Training	Not required	System should be self-explanatory
Other System Used	May need flexible to integrate with university apps	Documentation, user interface.
Ways of Working	Anywhere	Acceptance, Safety, Security, Operation, Maintenance, Portability

User Profile-02: Proctor Office Authority

Characteristic	Notes on Characteristics	Requirement implied
Type of User	Proctor office Authorities and staff	Secure Admin Dashboard
Age Range	24-60 years	Accessibility for all age groups
Frequency of Use	Daily	Performance, Operation, Acceptance, Efficiency
Mandatory	Yes	Role based access control
Computer Experience	Moderate to High	Training materials for advance feature
Education	Entry is with a Bachelor's degree	Simplified but professional interface
Goals	Efficiently manage the System maintenance, policy updates.	Focus on administrative tools
Language Skills	Bangla, English	User interface in relevant language.
Number of Users	Low to moderate	Dedicated access for staff
Training	Require training for maintain system	Comprehensive training modules
Other System Used	University ERP systems	Integration capabilities
Ways of Working	Office-based	Acceptance, Safety, Security, Operation, Maintenance, Access Control

User Profile-03: Admin or IT Person

Characteristic	Notes on Characteristics	Requirement implied
Type of User	System Administrator	User interface, accessibility and security
Age Range	24-60 years	Age Verification as appropriate
Frequency of Use	As Needed	Performance, Reliability
Mandatory	Yes	Role based access control
Computer Experience	Advanced	Comfortable with system administration tools and technical concepts and advanced feature
Education	Technical background preferred	Knowledge of vector database management, LLM, and security principles beneficial.
Goals	Manage system settings, tenant control, access control	Focus on monitoring and maintenance
Language Skills	Bangla, English	User interface in relevant language.
Number of Users	Limited team	System should support efficient collaboration and role delegation if applicable and dedicated access for admins
Training	May require specialized training or documentation	Comprehensive documentation and user guides available for specific functionalities.
Other System Used	Maybe needed to be flexible to use the University ERP Systems	Integration Capabilities
Ways of Working	Office-based	Acceptance, Safety, Security, Operation, Maintenance, Access Control

Tentative Elicitation Process

If you want to create something that really makes a difference for others, you need to start by understanding exactly what they want and what irks them about the current state of affairs. Here's how we'll get into that information and get the story straight from the people who will use the system:

1. Surveys and Questionnaires

- We will send surveys to students, just to get an idea of how things aren't working and which languages they're most comfortable with and what they would love to see happen in the new system.
- Let's check in with folks in the proctor office too—we can ask them about headaches and find out what might really help their workload, such as ways to better handle meeting invites or document organization, and so on.
- The university admins will have a say too—we will ask for their input on what they require for the system to remain secure and scalable and integrate well into the other tech they have.

2. Interviews and Focus Groups

- Some are just going to require a little discussion, and we'll schedule one-on-one meetings with some of these important individuals, such as the proctors and university administration, on what their actual needs or pain points are.
- Then make sure that our assumptions are on the track and we will create groups of students for lively discussions. These focus groups help us to check our ideas and hear what is the really important thing straight from the source.

3. Watching How Things Work Right Now

- On occasion, it helps simply to observe how things work. We can examine how students interact with the proctoring office, in person or through a simple phone call, in order to identify problem points in the process.
- We will also spend some time with the proctors during their normal activities to observe what causes them to slow down or makes it harder than it needs to be.

4. Try It Out and Ask for Feedback

- After we have a prototype of our new AI Proctor, a few individuals will get to take it for a test run.
- Their input—what works, what doesn't, and what is confusing them—will help us make it even better before we launch it to the rest of the company.

5. Looking Through the Paperwork

- Peeking Behind the Documents to ensure we do not overlook anything of major significance, we will go through the existing university policies and the way things have been done up until now.
- Also, we will focus on ensuring that the system we develop will be compatible with existing systems used by the university.

Scope

Let's see what's covered in this project—and what each of us can do, and what the system can do for us.

Signing Up and Logging In

- Here, the objective is clear: there needs to be a simple means for students and employees to enroll and gain access to the system safely.
- So, we will have a registration form where we will enter the basics (stuff like name, email, and a password). Login will be secure, done through JWT (basically, a technology that will keep your account safe). Those who forget their password will be able to reset it through the mail.

Admin Panel for Handling Documents

- Admins will have their dashboard where they can either upload/download official policy files (like PDFs) in Bangla & English.
- The files are uploaded in a designated upload section, which allows both language systems to be updated automatically, with administrator access required for modifications.

Looking Up Information

- Overall concept: Answer every question promptly and in the correct language.
- The tech such as FAISS and RAG models is what assists behind the scenes to store information and provide spot-on and context-oriented responses in either Bangla or English based on what is preferred by the user.

Scheduling Meetings

- The setup of a meeting should be super simple. One can send a request via chat.

- After the request has been sent by the student, the appropriate people get an email, and then an admin approves and everyone is looped in.

Multi-University Support

- We want this system to work for more than one university in the future.
- This implies that each university has its own area with different domains and completely distinct storage for their data records. It has a subscription-based approach that allows new institutions to join as and when they are ready.

Security and Tracking

- Keeping things safe and traceable is a big deal.
- The system has secured user login processes and action tracking, as well as constant security audits.

Project Planning and Initiation

We began the development of the AI Proctor because the need for such a system became very apparent after realizing that there had been a major void in the provision of assistance for students beyond the usual hours that any educational organization operates, namely 8 in the morning to 4 in the afternoon. Many students end up having issues with safety and discipline that fall beyond those operating hours but almost always with no real system to offer them assistance.

Therefore, the purpose of this project has been crystal clear from the beginning, and that has been to make available assistance to students around the clock using a smart bilingual AI assistant. Whether through a late-night chat or voice message, the assistant is always there and ready, not simply spouting memorized responses but rather drawing from actual university policies and responding to requests using a safe and dynamic backend. The actual deployment of the project has, therefore, been founded on this purpose.

Feasibility Study

Phase 1: Preliminary Analysis & Project Scope Definition

- Firstly, let's discuss the importance of this project. The fact is that most of the university proctor offices are only operational within business hours. But sometimes students may find themselves in very grave circumstances at night or at the weekend when harassment or safety is a concern. Further to this point, to receive assistance or gather information may take hours or days. This is just not acceptable.

  So, we are going to solve all these issues with the help of an AI Proctor system. So, all said and done, it is all about providing students with instant support in Bangla or English, in both chat and voice modes, and whatever you need, whatever you require, it will all be available all day and night!

- **Scope:**

  - Develop an artificial intelligence assistant capable of comprehending Bangla and English languages, using intelligent models and vector databases.
  - Creation of an admin dashboard to upload files and manage meetings for authorized personnel.
  - It should be easy for different universities to be part of the system, so the system can grow.
  - Integration with whatever technology is in place at the universities to ensure seamless functioning.

  Stakeholders:

  - **Students:** These are the first people the system is used by, whether seeking assistance or filing complaints.
  - **Proctor Office Staff:** They are responsible for managing issues, meeting requests, and updating rules.
  - **University Admins:** These monitor the system and ensure it is protected and has the capacity to expand as required.

- **What's needs to Happen:**

  - Secure sign-ups and logins: This ensures that the data
  - Chat and voice support services in Bangla and English.
  - Easy ways to upload and manage documents.
  - Rapid setup of meetings and email notifications.
  - A design scalable for more schools.

- **Conclusion:** This scope encompasses all the trouble spots we have highlighted, namely better access, fewer waits, and greater responsiveness from the university's proctoring

office. It should serve all parties involved well, namely the students, employees, and administrators, making it a sound project to continue to analyze in the coming period.

Phase 2: Market Feasibility Analysis (Market Research)

- First, let's determine whether there's demand for what we are creating. What is this for, exactly? It's for the students and faculty who need the information quickly, or those who run the proctoring office.

- **Target Audience:**

  1. **Primary Users:** Students and university staff who need urgent help or information about policies.
  2. **Secondary Users:** Office staff at Proctor and administrators at the university dealing with cases.

- **Market Need:**

  - The majority of the universities in Bangladesh are still lackround-the-clock facilities.
  - Believe it or not, there are lots of students who take advantage of the Bangla language assistance, which is why ensuring bilingual
  - Tragically, cases such as harassment and cyberbullying are only on the rise and therefore a system of reporting is much needed.

- **Competitive Landscape (Bangladesh Perspective):**

  1. Currently, the proctor office of Bangladesh lacks anything like an AI
  2. The majority of countries still do this by hand, and it might be a little slow and hit or miss.

- **User Feedback:**

  1. We have asked around, and most students find that they would like an "always on" environment that is available in both languages.
  2. The major issues here are having a desire to keep their information private, trusting the AI's recommendations, and ease of use.

- **Will it be adopted by individuals?**

  - The short answer is: pretty likely. In fact, for colleges near major cities or larger towns, this flexible system means that even more students will benefit, and even more schools may participate without costing a fortune.

Conclusion: There is definitely a need, and no one is addressing it at this point. We have a good chance with this project if we do it correctly.

Phase 3: Technical Feasibility Analysis

- We want to ensure the system will mesh well within the current technical usage at universities.

- **Hardware:**

    1. It's all web based, so no need for fancy gear
    2. Whether you are on a phone, laptop, or tablet, you are set.
    3. Using the cloud means we can grow and add more schools anytime.

- **Software:**

    1. **Frontend:** The frontend is designed by using React.js, which makes it look nice and run fast, whether it is for admins or regular users.
    2. **Backend:** The backend is built using FastAPI and Node.js to keep everything agile and nimble
    3. **Database:** Database used PostgreSQL (for basic details) and FAISS (for Vector DB as search is faster and smarter in this database)
    4. In direct integration with OpenAI GPT-4o and other APIs.

- **Technical Expert:**

    1. Our team is familiar with contemporary tech and AI, so implementation and maintenance should not be a problem.
    2. Neither will persons who are "un-techies" require much training on how to use the admin panel

- **Scalability:**

    1. The system is multi-tenant, which makes it easy to add new universities.
    2. Despite the prevalence of data, the system is not overwhelmed

- **Security:**

    1. Only people with the required credentials can log in because of JWT.
    2. Data will be encrypted, and we will conduct periodical security audits to identify any issues.

- **Conclusion:** Thus, from a technical point of view, the AI Proctor system is ready for deployment because it can effectively function with the type of infrastructure largely in place at many colleges. Thus, from a technical point of view, the AI Proctor system is

ready for deployment because it can effectively function with the type of infrastructure largely in place at many colleges.

Phase 4: Financial Feasibility Analysis

- The actual costs associated with building and sustaining a system like AI Proctor, and whether or not those costs are sensible given what we will be able to achieve with it.

- **Development & Operational Cost Breakdown (BDT):**

Expense	Responsibility	Estimated Cost (BDT)
AI Engineer	Building the brains; managing all the AI and smart search	50,000
Junior AI Engineer	APIs ticking and helping out in the backend	25,000
Frontend Developer	Making sure it all looks and feels right for users	20,000
Backend Developer	The server-side heavy lifting	25,000
UI/UX Designer	Designing experiences that are actually nice to use	20,000
OpenAI API (1st month testing)	1st month so we can road-test the AI	3,500
OpenAI API (production monthly)	The real deal, month-to-month, once we launch	7,000
Cloud Hosting	Servers and online space (Orangesoft BD – AWS Small Plan)	5,500
Domain & SSL	Web address + security, spread over each month	1,250
Total Initial Cost		**157,250 BDT**

- **Monthly Cost (BDT):**

Expense	Monthly Cost (BDT)
OpenAI API Usage	7,000
Cloud Hosting	5,500
Domain & SSL	250
Total Monthly	**12,750**

- **Domain and SSL**

  - **Domain Registration:** About Tk.1000 per year
  - **SSL Certificate:** About Tk.2000 per year
    - **Total Yearly Cost will be:** TK.3000

    - **Monthly Cost will be:** TK.3000 / 12 = TK.250 per month

- **Cloud Hosting**
  Given the system requirements, a good cloud hosting service will be required. Going by local companies:

  - **Provider:** Orangesoft BD
  - **Plan:** AWS SMALL
  - **Specifications:**
    - 4 Core Processor
    - 2 GB RAM
    - 20 GB SSD Disk Space
    - 4 GB Bandwidth

    Cost: TK.5500 per month

- **According to the projected usage:**

  - **Users:** 50 users per day
  - **Queries per User:** 15 queries (Prompt or question they asked for)
  - **Tokens per Query:** Estimated 300-500 tokens (including both input and output)

  Total Daily Tokens:

  - 50 users × 15 queries/user × 500 tokens/query = 375,000 tokens/day

  Total Monthly Tokens:

  - 375,000 tokens/day × 30 days = 11,250,000 tokens/month

  OpenAI GPT-4o Pricing:

  1. **Input Tokens:** $2.50 per 1 million tokens
  2. **Output Tokens:** $2.50 per 1 million tokens

- **Token Usage Estimate (OpenAI GPT-4o):**

  1. Users: 50/day × 15 queries = 750 queries/day
  2. Tokens: ~500 tokens/query = 375,000 tokens/day = 11.25M/month

3. Cost: Input + Output tokens @ $2.50/1M each ≈ $60 (≈ 7,000 BDT)

- **Benefits:**

1. Issues will be resolved much quicker for students and employees.
2. Students are never left waiting, help is always available.
3. Having multiple languages ensures that no one is left behind.
4. There is potential profit here with subscription sales to other universities.

- **What About the Return on Investment (ROI)?**

1. At this point, with these projections, we can anticipate covering costs and realizing a profit within 3 years.
2. Specific tangible benefits: reduced manual work, and more work is done faster.
3. Intangible: students trust their university more and feel happier about how promptly problems get resolved.

- **Conclusion:** Things are also positive financially. There is some initial outlay of cash, but it is clearly going to pay dividends with smooth-running universities and happy students. This has every chance of building itself into a bigger platform than may initially meet the eye and has been costed for local conditions for AI, servers, and staff.

Chapter – 2
Software Requirement and Specification

Functional SRS

SRS file - 01: User Registration

FR01	**Register / Login**
Description	**Anyone who are use this system should be able to sign up with an email or phone number, then log in securely with those details are no false or fake, no risk.**
Stakeholder	**Student, Admin, Proctor Authority**

SRS file - 02: User Login

FR02	**User Login**
Description	**After registration user able to login , it will allow to go homepage**
Stakeholder	**Student, Admin, Proctor Authority**

SRS file - 03: Start Chat

FR03	**Provide Chat**
Description	**Students should be able to chat with the assistant, either in Bangla or English. Ask questions, get help—whatever works for them.**
Stakeholder	**Student, Proctor Authority**

SRS file - 04: Chat Voice

FR04	**Provide Voice**
Description	**Users should be able to communicate with the assistant through voice in both Bangla and English language.**
Stakeholder	**Student, Proctor Authority**

SRS file - 05: Select Language

FR05	**Language Selection**
Description	**The system should allow users to select their preferred language (Bangla or English) before starting any chat or voice interaction. This setting should apply across the session.**
Stakeholder	**Student, Proctor Authority**

SRS file - 06: Get Subscription

FR06	**Admin Subscription**
Description	**University Admins must subscribe to activate and maintain access to the admin panel. Subscription should support monthly or yearly payments and generate invoice records.**
Stakeholder	**Admin**

SRS file - 07: Upload Document

FR07	**Admin Upload Document**
Description	**Admin should be able to upload Bangla and English PDFs for embedding into the system.**
Stakeholder	**Admin, IT Team**

SRS file - 08: Meeting Scheduling

FR08	**Schedule Meeting with Proctor Office**
Description	**Students can request meetings by providing details. The system generates and sends a meeting document to the Proctor Office.**
Stakeholder	**Student, Proctor Authority**

SRS file - 09: Chat and Voice History

FR09	**Save and Retrieve Chat and voice History in text format**
Description	**The system must save chat and voice history under selected topics for future reference.**
Stakeholder	**Student, Admin**

SRS file - 10: Receive Notification

FR10	**Notify Users of Important Events**
Description	**The system should notify users upon meeting confirmation, document submission, or emergency updates.**
Stakeholder	**Student and Admin**

SRS file - 11: Logout

FR11	**Logout**
Description	**Users can log out of this system by using this option. The login will be savedif the user wants to save their account.**
Stakeholder	**Student, Admin**

Non-Functional SRS

SRS file - 12: Performance

FR12	**Ensure High Performance**
Description	**The system should respond to user queries within 2 seconds and handle concurrent requests smoothly.**
Stakeholder	**Student, Admin, University IT Authority**

SRS file - 13: Usability

FR13	**Provide Intuitive User Experience**
Description	**The interface should be simple and intuitive, supporting both desktop and mobile browsers.**
Stakeholder	**Student, Admin**

SRS file - 14: Reliability

FR14	**Maintain System Reliability**
Description	**The system should have a minimum uptime of 99.5%, with automatic fallback in case of failures.**
Stakeholder	**Admin, IT Team**

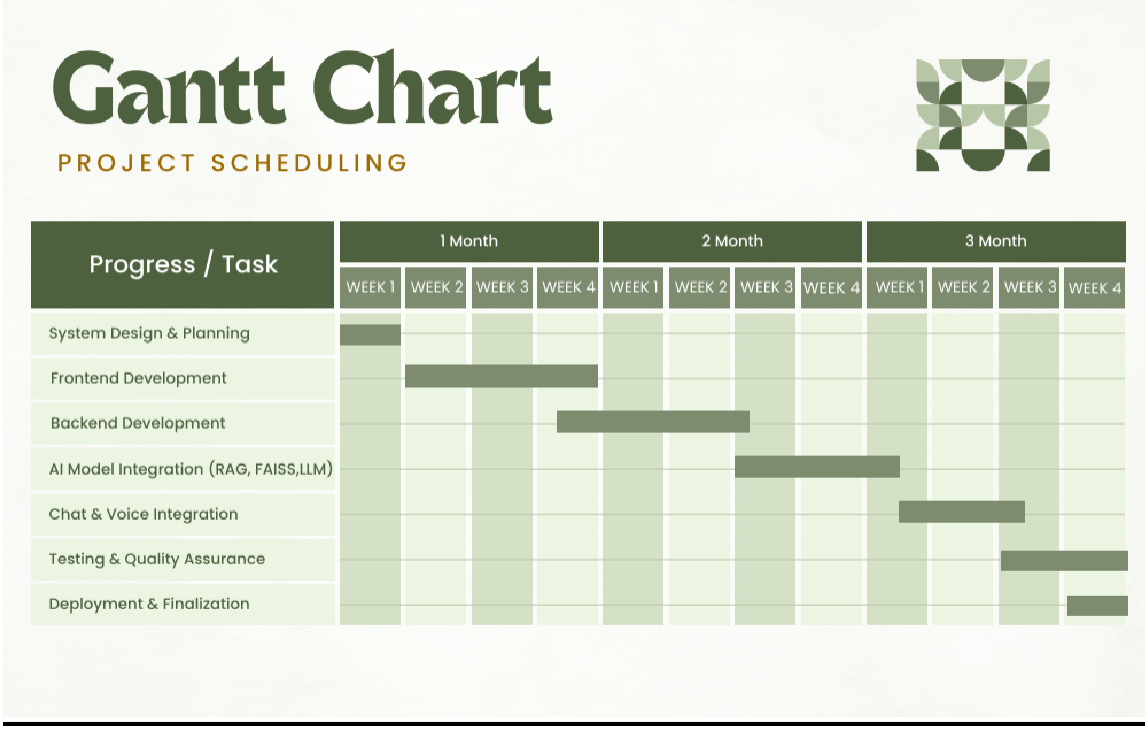
SRS file - 15: Security

FR15	**Secure User Data**
Description	**All user data must be encrypted and access-controlled. Meeting requests and documents must be securely transmitted.**
Stakeholder	**Student, Admin, University Authority**

SRS file - 16: Scalability

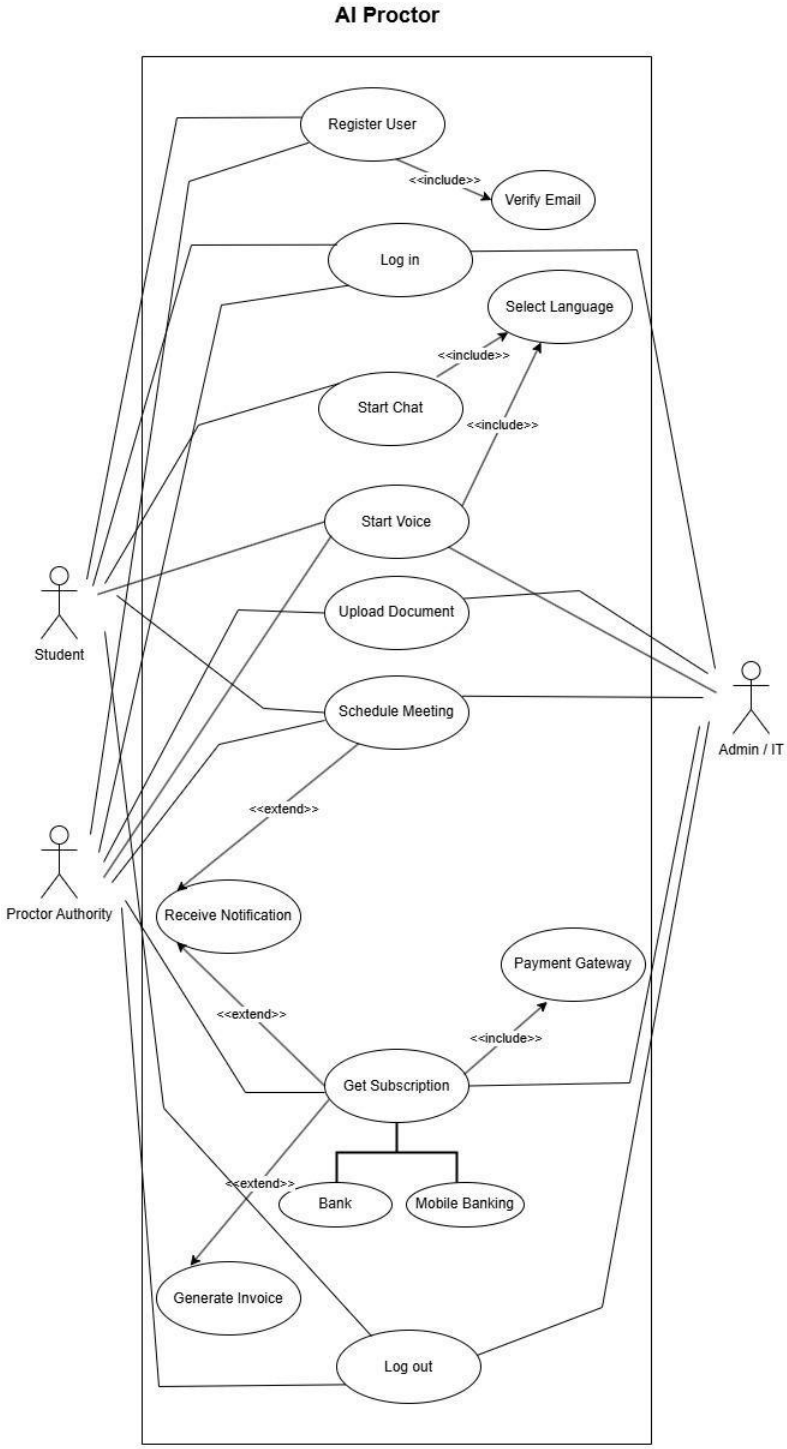
FR16	**Ensure Scalability**
Description	**Support 10,000+ concurrent users during peak hours**
Stakeholder	**Student, Admin, University Authority**

Project Scheduling



Task	Duration
System Design & Planning	1 weeks
Frontend Development	3 weeks
Backend Development	3 weeks
AI Model Integration (RAG, FAISS, LLM)	3 weeks
Voice Integration	2 weeks
Testing & Quality Assurance	2 weeks
Deployment & Finalization	1 week
Total Duration	3 months

User Case Diagram



User Case Description

Case Description-01: User Registration

<table>
<tr><td>Use Case</td><td>User Registration</td></tr>
<tr><td>Goal</td><td>Student can register to access the AI Proctor system.</td></tr>
<tr><td>Success End Condition</td><td>Notification: Successfully Complete Registration</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Submission Not Submitted”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student / Visitor

System Backend, Authentication Service</td></tr>
<tr><td>Trigger</td><td>User requests the registration form to create an account.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>User clicks “Sign Up” button.</td></tr>
<tr><td>2.</td><td>System provides the Sign Up form.</td></tr>
<tr><td>3.</td><td>User enters required information.</td></tr>
<tr><td>4.</td><td>User presses the “Submit” button.</td></tr>
<tr><td>5.</td><td>System validates and saves the information.</td></tr>
<tr><td>6.</td><td>System confirms: “Successfully Signed Up!”</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>2.1</td><td>Server Doesn’t Respond</td></tr>
<tr><td>2.1.a</td><td>Try Again Later!</td></tr>
<tr><td>3.1</td><td>User did not fill up required details</td></tr>
<tr><td>3.1.a</td><td>System checks & notifies: “Please! Fill Up the Box.”</td></tr>
<tr><td>4.1</td><td>System did not respond</td></tr>
<tr><td>4.1.a</td><td>Show Error Message</td></tr>
</table>
</td></tr>
</table>

Case Description-02: User Login

<table>
<tr><td>Use Case</td><td>User Login</td></tr>
<tr><td>Goal</td><td>Registered users can log in to access the AI Proctor system.</td></tr>
<tr><td>Success End Condition</td><td>Notification: “Login Successful”</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Invalid Credentials / Login Failed”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student, Admin, Staff

Authentication System</td></tr>
<tr><td>Trigger</td><td>User enters login credentials.</td></tr>
<tr><td>Description / Main Success Scenario</td><td><table>
<tr><td>1.</td><td>User clicks “Login” button.</td></tr>
<tr><td>2.</td><td>System provides Login form.</td></tr>
<tr><td>3.</td><td>User enters email & password.</td></tr>
<tr><td>4.</td><td>User clicks “Sign In”.</td></tr>
<tr><td>5.</td><td>System verifies credentials.</td></tr>
<tr><td>6.</td><td>System logs the user in and redirects to dashboard.</td></tr>
</table></td></tr>
<tr><td>Alternative Flows</td><td><table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>3.1</td><td>Incorrect Password</td></tr>
<tr><td>3.1.a</td><td>Notification: “Wrong Password”</td></tr>
<tr><td>3.2</td><td>Invalid Email</td></tr>
<tr><td>3.2.a</td><td>Notification: “Invalid Email Format”</td></tr>
<tr><td>5.1</td><td>Server not responding</td></tr>
<tr><td>5.1.a</td><td>Try Again Later</td></tr>
<tr><td>6.1</td><td>Session creation failed</td></tr>
<tr><td>6.1.a</td><td>Show Error Message</td></tr>
</table></td></tr>
</table>

Case Description-03: Start Chat

<table>
<tr><td>Use Case</td><td>Start Chat</td></tr>
<tr><td>Goal</td><td>User can communicate with the AI assistant using text chat.</td></tr>
<tr><td>Success End Condition</td><td>System shows AI-generated response.</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Chat Could Not Be Started”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student, Admin, Staff

Chat Engine, LLM Service</td></tr>
<tr><td>Trigger</td><td>User clicks “Chat” button.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>User selects “Chat” option.</td></tr>
<tr><td>2.</td><td>System loads chat interface.</td></tr>
<tr><td>3.</td><td>User types a message.</td></tr>
<tr><td>4.</td><td>User presses “Send”.</td></tr>
<tr><td>5.</td><td>System processes the message with the AI model.</td></tr>
<tr><td>6.</td><td>AI sends back the response and system displays it.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>2.1</td><td>Chat interface fails to load</td></tr>
<tr><td>2.1.a</td><td>Reload Suggestion</td></tr>
<tr><td>5.1</td><td>AI model not responding</td></tr>
<tr><td>5.1.a</td><td>Notification: “Service Unavailable”</td></tr>
<tr><td>6.1</td><td>Response cannot be displayed</td></tr>
<tr><td>6.1.a</td><td>Show Error Message</td></tr>
</table>
</td></tr>
</table>

Case Description-4: Start Voice Chat

<table>
<tr><td>Use Case</td><td>Start Voice Chat</td></tr>
<tr><td>Goal</td><td>User can speak with the AI assistant using voice input.</td></tr>
<tr><td>Success End Condition</td><td>System plays AI-generated voice response.</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Voice Chat Failed”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student, Admin, Staff

Speech-to-Text (STT), Text-to-Speech (TTS), LLM Engine</td></tr>
<tr><td>Trigger</td><td>User clicks “Voice Chat” button.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>User clicks “Start Voice Chat”.</td></tr>
<tr><td>2.</td><td>System activates microphone permission.</td></tr>
<tr><td>3.</td><td>User speaks.</td></tr>
<tr><td>4.</td><td>STT engine converts speech to text.</td></tr>
<tr><td>5.</td><td>AI processes the text and generates answer.</td></tr>
<tr><td>6.</td><td>TTS engine converts text to audio and System plays the AI’s voice response.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>Microphone not permitted</td></tr>
<tr><td>1.1.a</td><td>Notify: “Enable Microphone Access”</td></tr>
<tr><td>2.1</td><td>STT Engine Error</td></tr>
<tr><td>2.1.a</td><td>Try Again</td></tr>
<tr><td>4.1</td><td>Noise detected / Low clarity</td></tr>
<tr><td>5.1</td><td>AI fails to respond</td></tr>
<tr><td>5.1.a</td><td>Notification: “Service Unavailable”</td></tr>
<tr><td>6.1</td><td>Audio cannot be played</td></tr>
<tr><td>6.1.a</td><td>Show Error Message</td></tr>
</table>
</td></tr>
</table>

Case Description-05: Select Language

<table>
<tr><td>Use Case</td><td>Select Language</td></tr>
<tr><td>Goal</td><td>User can change system language (Bangla/English).</td></tr>
<tr><td>Success End Condition</td><td>System reloads UI & interactions in chosen language.</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Language Change Failed”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student</td></tr>
<tr><td>Trigger</td><td>User selects a language from the menu.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>User opens Homepage</td></tr>
<tr><td>2.</td><td>System displays available languages.</td></tr>
<tr><td>3.</td><td>User selects preferred language.</td></tr>
<tr><td>4.</td><td>System updates interface language.</td></tr>
<tr><td>5.</td><td>System reloads content in new language.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System cannot load languages</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>3.1</td><td>Unsupported Language Selected</td></tr>
<tr><td>3.1.a</td><td>Show Warning Message</td></tr>
<tr><td>4.1</td><td>Translation Module Error</td></tr>
<tr><td>4.1.a</td><td>Notification: “Cannot Change Now”</td></tr>
<tr><td>5.1</td><td>UI fails to refresh</td></tr>
<tr><td>5.1.a</td><td>Reload Application</td></tr>
</table>
</td></tr>
</table>

Case Description-06: Admin Subscription

<table>
<tr><td>Use Case</td><td>Admin Subscription</td></tr>
<tr><td>Goal</td><td>Admin can purchase a subscription to activate and maintain access to the Admin Panel.</td></tr>
<tr><td>Success End Condition</td><td>Notification: “Subscription Activated Successfully”</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Subscription Failed / Payment Unsuccessful”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Admin

Payment Gateway, Bank/Mobile Banking</td></tr>
<tr><td>Trigger</td><td>Admin selects subscription plan (monthly/yearly).</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>Admin selects “Get Subscription”.</td></tr>
<tr><td>2.</td><td>System displays available plans.</td></tr>
<tr><td>3.</td><td>Admin selects monthly or yearly subscription.</td></tr>
<tr><td>4.</td><td>System redirects to payment gateway.</td></tr>
<tr><td>5.</td><td>Admin completes payment via bank/mobile banking.</td></tr>
<tr><td>6.</td><td>System verifies payment & updates subscription status and confirm“Subscription Activated Successfully”.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>3.1</td><td>Plan not selected</td></tr>
<tr><td>3.1.a</td><td>System notifies: “Please Select a Plan”</td></tr>
<tr><td>4.1</td><td>Payment gateway not responding</td></tr>
<tr><td>4.1.a</td><td>Try Again Later</td></tr>
<tr><td>5.1</td><td>Payment failed</td></tr>
<tr><td>5.1.a</td><td>Notification: “Payment Unsuccessful”</td></tr>
<tr><td>6.1</td><td>System cannot update subscription status and Show Error Message</td></tr>
</table>
</td></tr>
</table>

Case Description-07: Upload Document

<table>
<tr><td>Use Case</td><td>Upload Document</td></tr>
<tr><td>Goal</td><td>Admin/IT Team uploads Bangla or English PDF files for embedding.</td></tr>
<tr><td>Success End Condition</td><td>Notification: “Document Uploaded Successfully”</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Upload Failed”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Admin, IT Team

File Storage System</td></tr>
<tr><td>Trigger</td><td>TriggerUser clicks “Upload Document” button.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>Admin opens the document upload interface.</td></tr>
<tr><td>2.</td><td>System displays upload options.</td></tr>
<tr><td>3.</td><td>Admin selects PDF file.</td></tr>
<tr><td>4.</td><td>Admin presses “Upload”.</td></tr>
<tr><td>5.</td><td>System validates file type and size.</td></tr>
<tr><td>6.</td><td>System uploads document to storage and processes embedding and “Document Uploaded Successfully”.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>3.1</td><td>Wrong file type selected</td></tr>
<tr><td>3.1.a</td><td>Notification: “Only PDF Allowed”</td></tr>
<tr><td>5.1</td><td>File size too large</td></tr>
<tr><td>5.1.a</td><td>Notification: “File Size Exceeded”</td></tr>
<tr><td>6.1</td><td>Upload failed due to network/storage error</td></tr>
<tr><td>6.1.a</td><td>Show Error Message</td></tr>
</table>
</td></tr>
</table>

Case Description-08: Schedule Meeting

<table>
<tr><td>Use Case</td><td>Schedule Meeting</td></tr>
<tr><td>Goal</td><td>Student can request a meeting with the Proctor Office.</td></tr>
<tr><td>Success End Condition</td><td>Notification: “Meeting Request Sent Successfully”</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Meeting Request Failed”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student

Proctor Authority, Notification Module</td></tr>
<tr><td>Trigger</td><td>Student selects “Schedule Meeting”.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>Student clicks “Schedule Meeting”.</td></tr>
<tr><td>2.</td><td>System displays meeting request form.</td></tr>
<tr><td>3.</td><td>Student fills in details (reason, date, time).</td></tr>
<tr><td>4.</td><td>Student presses “Submit”.</td></tr>
<tr><td>5.</td><td>System generates meeting request document.</td></tr>
<tr><td>6.</td><td>System sends request to Proctor Office and confirm “Meeting Request Sent Successfully”.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>3.1</td><td>Student missed required details</td></tr>
<tr><td>3.1.a</td><td>System notifies: “Please Fill All Required Fields”</td></tr>
<tr><td>5.1</td><td>Document generation failed</td></tr>
<tr><td>5.1.a</td><td>Show Error Message</td></tr>
<tr><td>6.1</td><td>System fails to send request</td></tr>
<tr><td>6.1.a</td><td>Notification: “Request Not Sent”</td></tr>
</table>
</td></tr>
</table>

Case Description-09: Chat & Voice History

<table>
<tr><td>Use Case</td><td>Chat & Voice History</td></tr>
<tr><td>Goal</td><td>System saves and retrieves chat/voice session history for users.</td></tr>
<tr><td>Success End Condition</td><td>History displayed successfully.</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Unable to Load History”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student, Admin

Storage System</td></tr>
<tr><td>Trigger</td><td>User opens “History” section.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>User selects “Chat & Voice History”.</td></tr>
<tr><td>2.</td><td>System fetches saved text-based history.</td></tr>
<tr><td>3.</td><td>System loads voice-to-text converted history.</td></tr>
<tr><td>4.</td><td>System organizes data under topics.</td></tr>
<tr><td>5.</td><td>History displayed to the user.</td></tr>
<tr><td>6.</td><td>User reviews previous conversations.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System unavailable</td></tr>
<tr><td>1.1.a</td><td>Try Again!!</td></tr>
<tr><td>2.1</td><td>Storage response delay</td></tr>
<tr><td>2.1.a</td><td>Notification: “Loading...”</td></tr>
<tr><td>4.1</td><td>Sorting or formatting issue</td></tr>
<tr><td>4.1.a</td><td>System retries formatting</td></tr>
<tr><td>5.1</td><td>History cannot be displayed</td></tr>
<tr><td>5.1.a</td><td>Show Error Message</td></tr>
</table>
</td></tr>
</table>

Case Description-10: Notification

<table>
<tr><td>Use Case</td><td>Notifications</td></tr>
<tr><td>Goal</td><td>Users receive system alerts for meetings, updates, submissions, etc.</td></tr>
<tr><td>Success End Condition</td><td>Notification delivered successfully.</td></tr>
<tr><td>Failed End Condition</td><td>Notification: “Failed to Send Notification”</td></tr>
<tr><td>Primary Actors:

Secondary Actors:</td><td>Student, Admin</td></tr>
<tr><td>Trigger</td><td>System triggers an event requiring user alert.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>An event occurs (meeting confirmation, document approval, updates).</td></tr>
<tr><td>2.</td><td>System detects the event.</td></tr>
<tr><td>3.</td><td>Notification module prepares alert message.</td></tr>
<tr><td>4.</td><td>System sends notification to the user.</td></tr>
<tr><td>5.</td><td>User receives alert in the app.</td></tr>
<tr><td>6.</td><td>System confirms delivery.</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>Notification module error</td></tr>
<tr><td>1.1.a</td><td>System retries</td></tr>
<tr><td>4.1</td><td>User device offline</td></tr>
<tr><td>4.1.a</td><td>Notification queued</td></tr>
<tr><td>5.1</td><td>Delivery failed</td></tr>
<tr><td>5.1.a</td><td>Show Error Message</td></tr>
<tr><td>6.1</td><td>Confirmation not received</td></tr>
<tr><td>6.1.a</td><td>System logs failure</td></tr>
</table>
</td></tr>
</table>

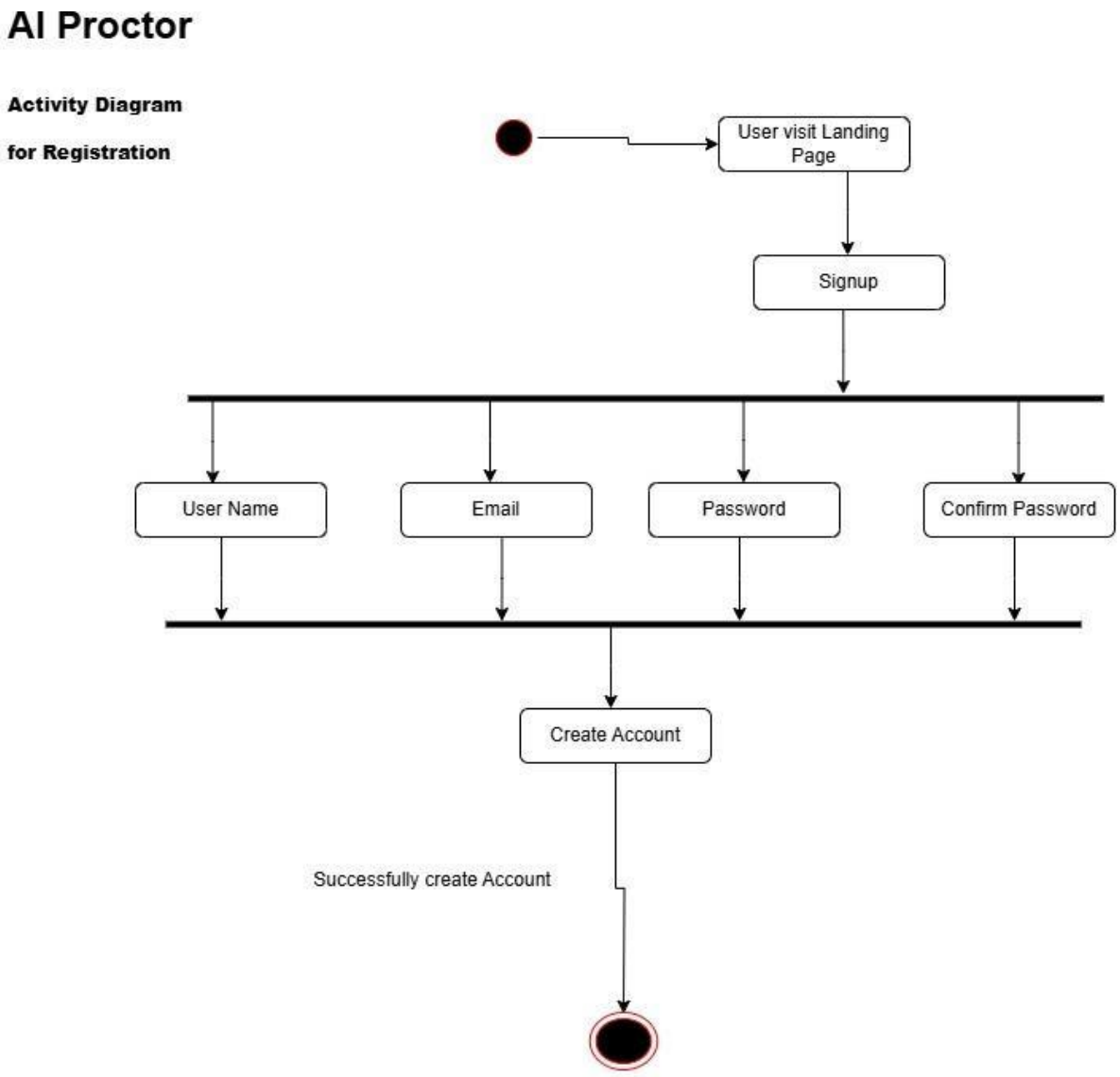
Case Description-11: Logout

<table>
<tr><td>Use Case</td><td>Logout</td></tr>
<tr><td>Goal</td><td>User can safely log out of the AI Proctor System.</td></tr>
<tr><td>Success End Condition</td><td>Notification: Successfully Logged Out</td></tr>
<tr><td>Failed End Condition</td><td>Notification: Logout Failed</td></tr>
<tr><td>Primary Actors:

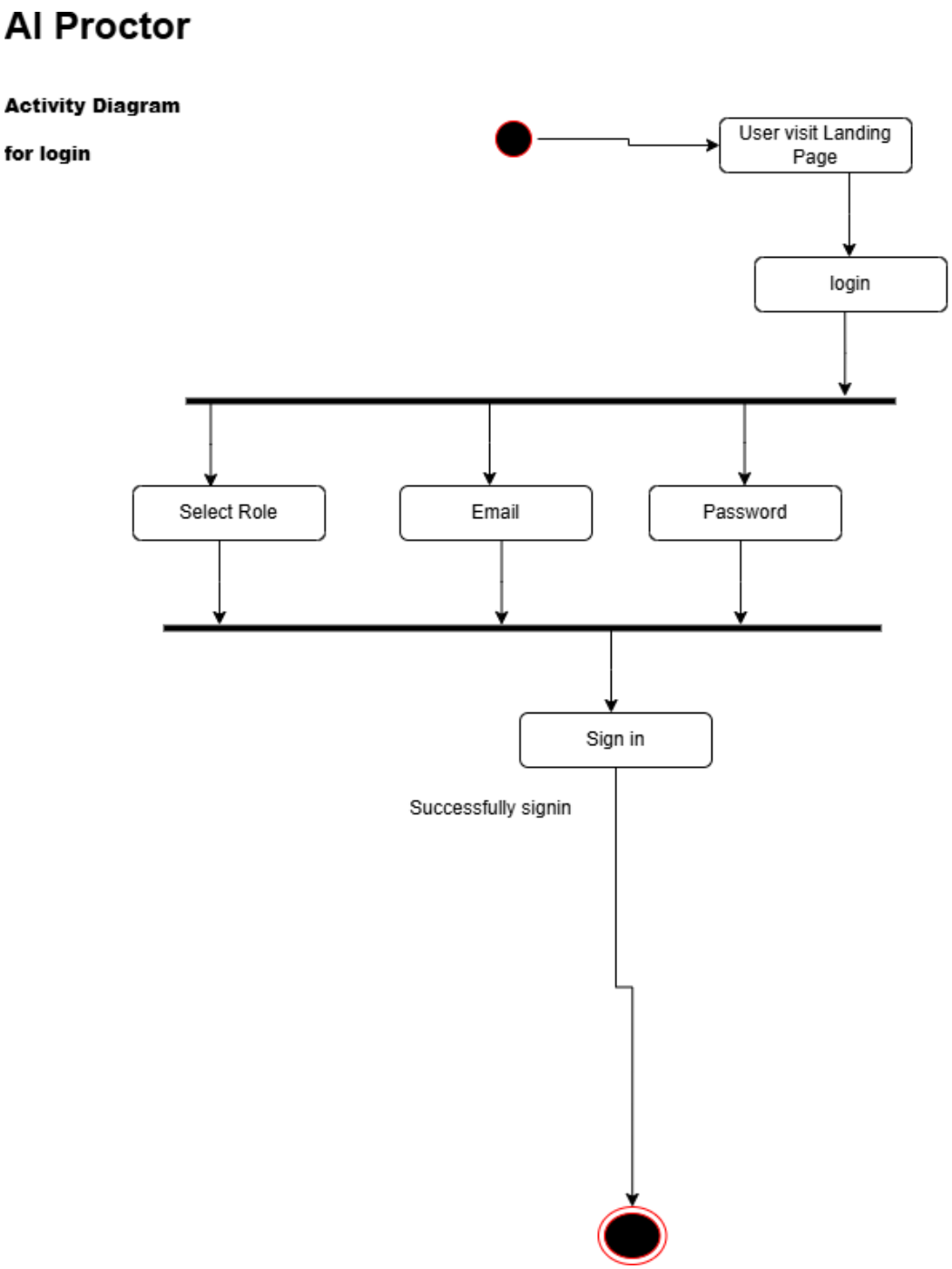
Secondary Actors:</td><td>Student, Admin, Staff</td></tr>
<tr><td>Trigger</td><td>User clicks the Logout button to exit the system.</td></tr>
<tr><td>Description / Main Success Scenario</td><td>
<table>
<tr><td>1.</td><td>User clicks on the Logout button.</td></tr>
<tr><td>2.</td><td>System verifies the current active session.</td></tr>
<tr><td>3.</td><td>System terminates the session.</td></tr>
<tr><td>4.</td><td>User is redirected to the login/landing page.</td></tr>
<tr><td>5.</td><td>System displays notification: Successfully Logged Out</td></tr>
</table>
</td></tr>
<tr><td>Alternative Flows</td><td>
<table>
<tr><td>1.1</td><td>System Error</td></tr>
<tr><td>1.1.a</td><td>Notify user: System Error! Please Try Again!!</td></tr>
<tr><td>2.1</td><td>Session Not Found / Session Expired</td></tr>
<tr><td>2.1.a</td><td>Notify user: Session Already Expired!</td></tr>
<tr><td>3.1</td><td>System failed to terminate session</td></tr>
<tr><td>3.1.a</td><td>Notification: Logout Failed! Please Try Again!</td></tr>
<tr><td>4.1</td><td>Redirection not working</td></tr>
<tr><td>4.1.a</td><td>Show Error Message: Unable to Redirect</td></tr>
</table>
</td></tr>
</table>

Activity Diagram

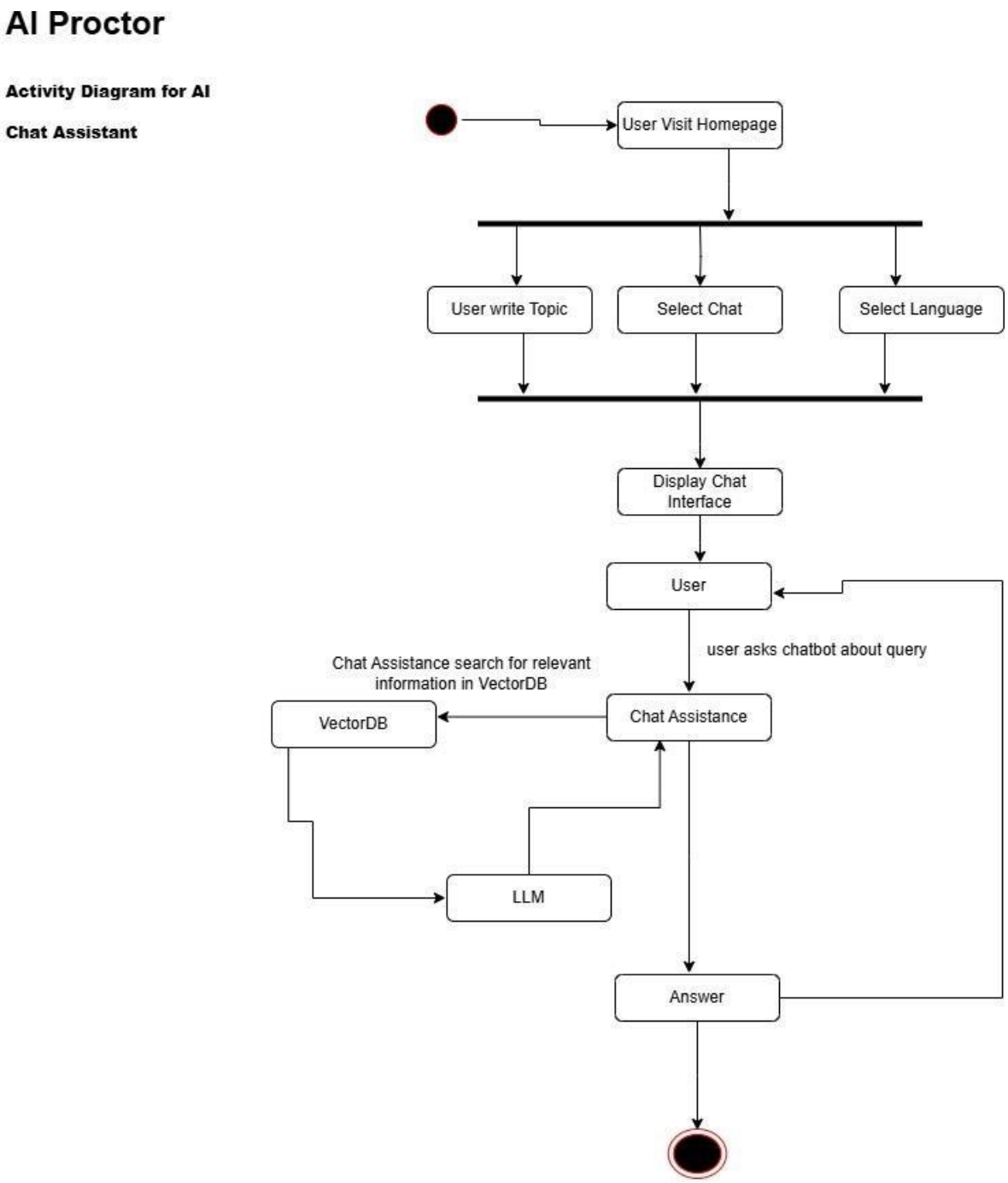
Activity Diagram-1: Registration



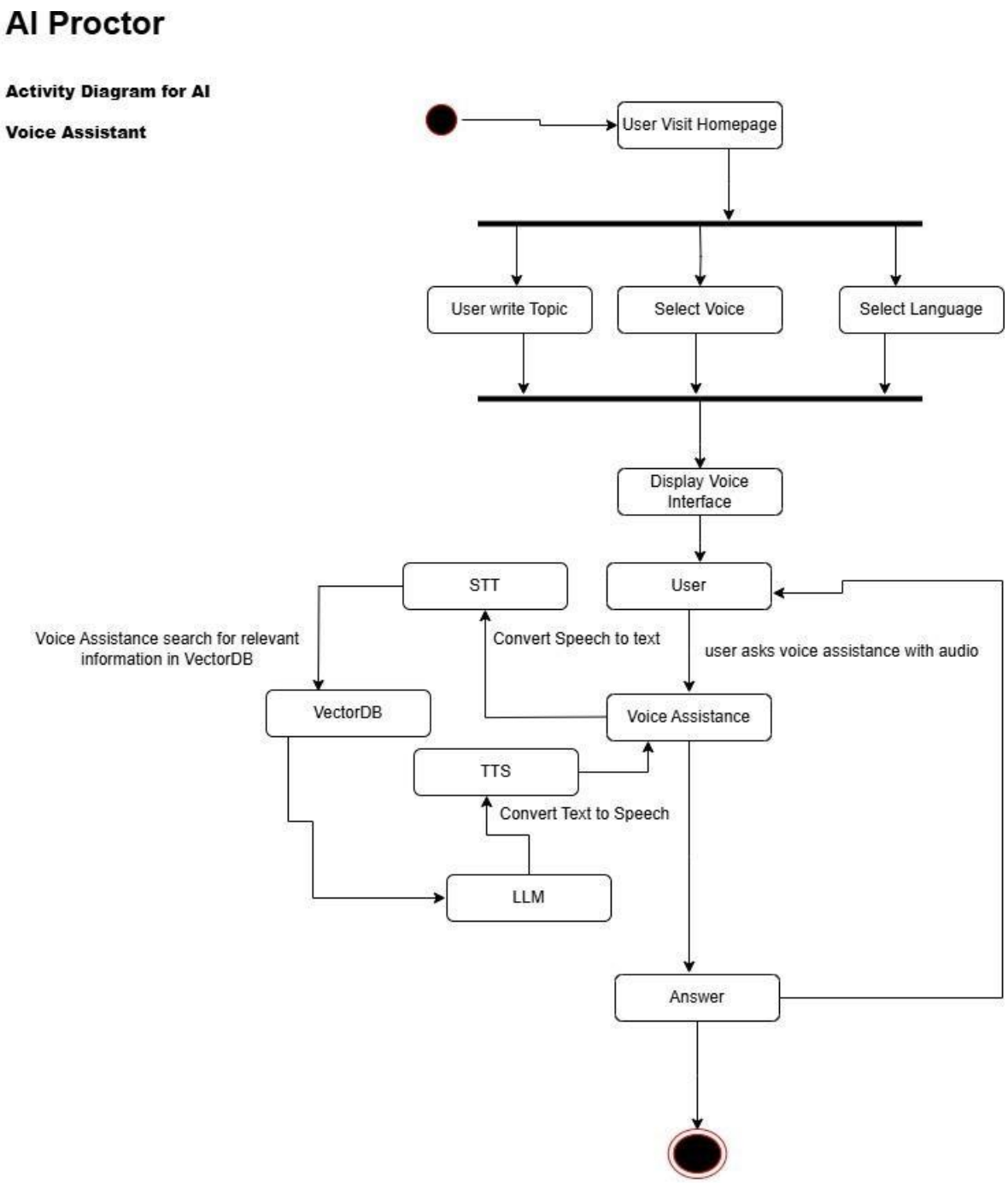
Activity Diagram-2: Login



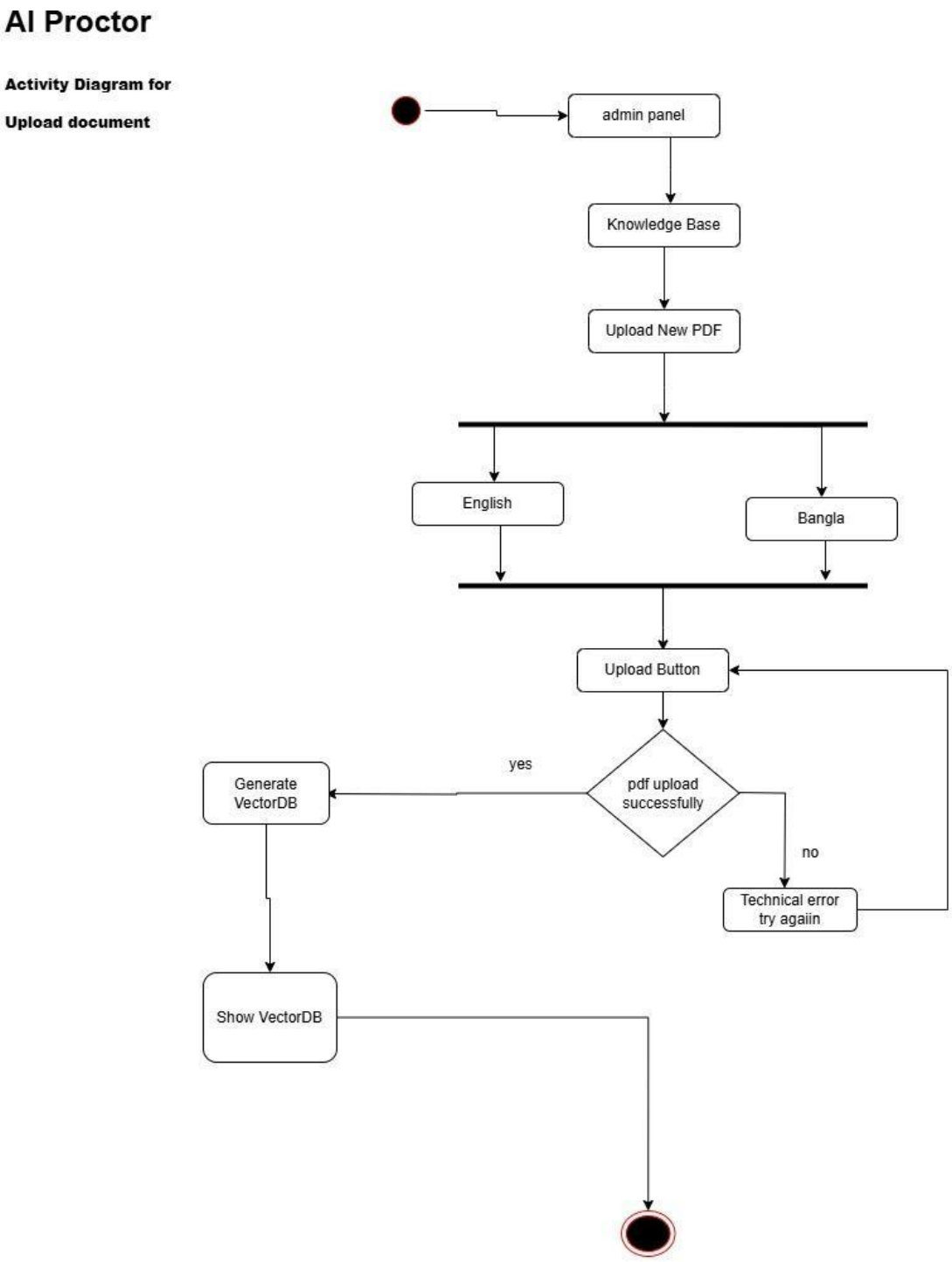
Activity Diagram-3: Chat



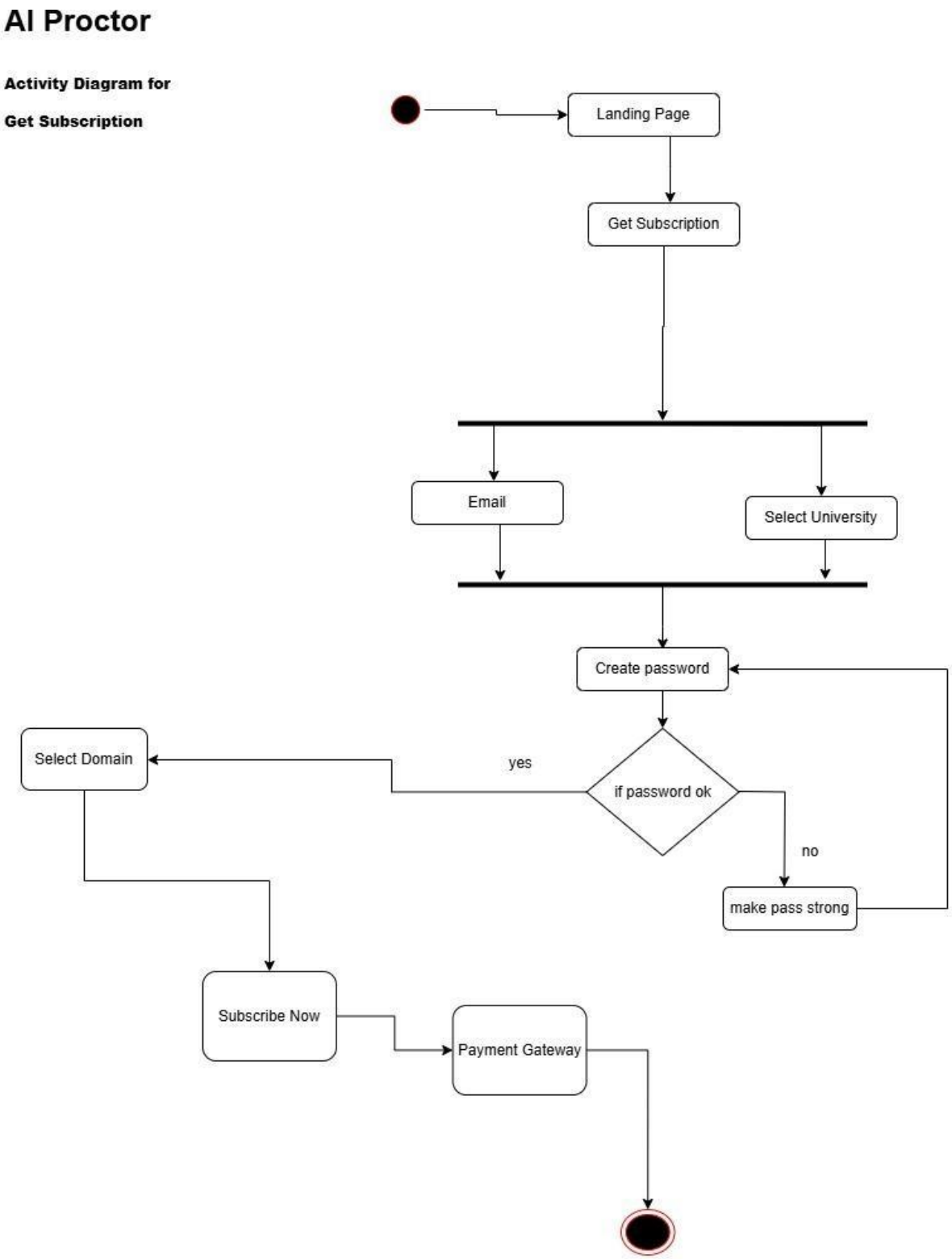
Activity Diagram-4: Voice Chat



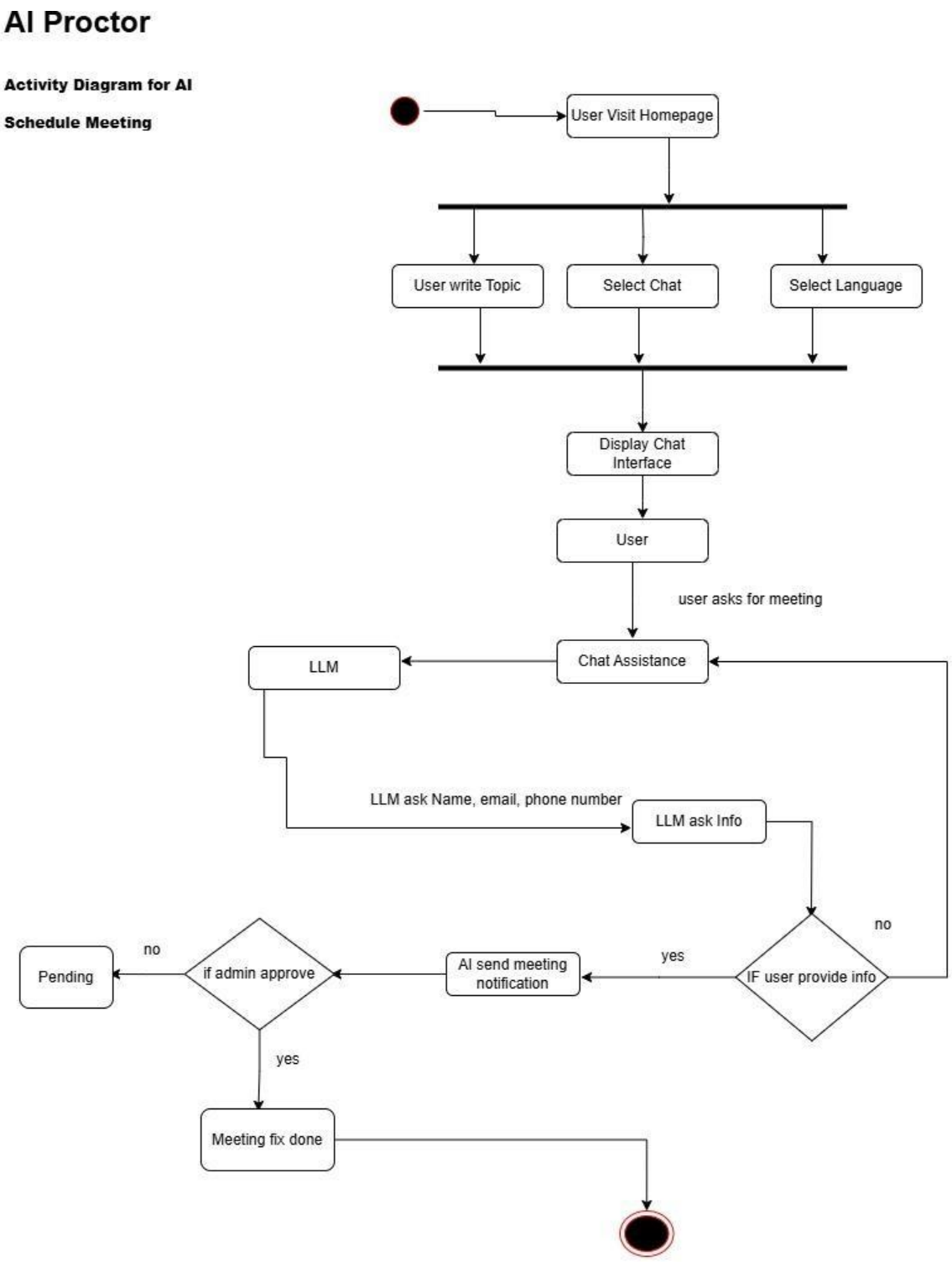
Activity Diagram-5: Upload document



Activity Diagram-6: Get Subscription

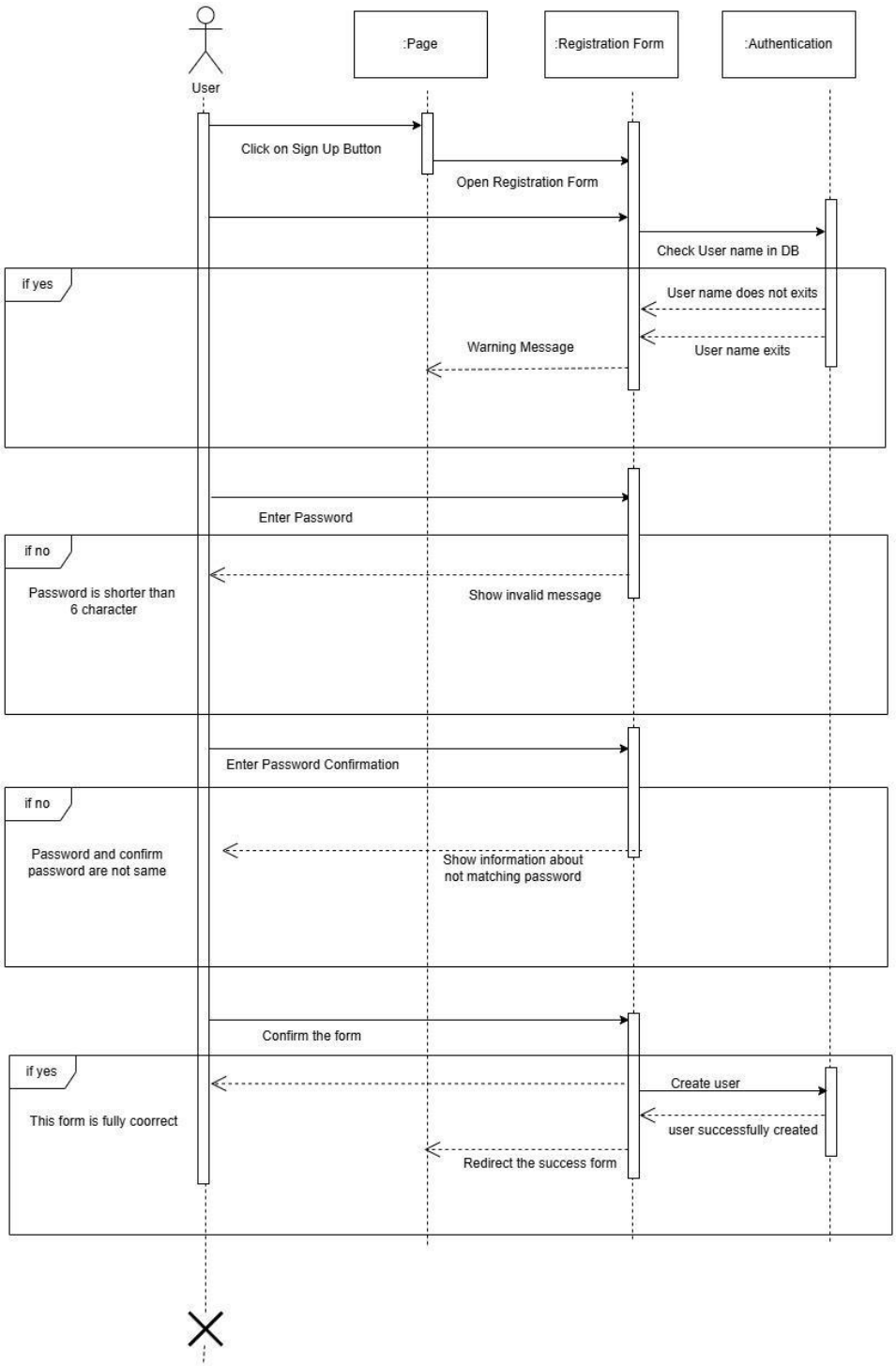


Activity Diagram-7: Meeting Fix

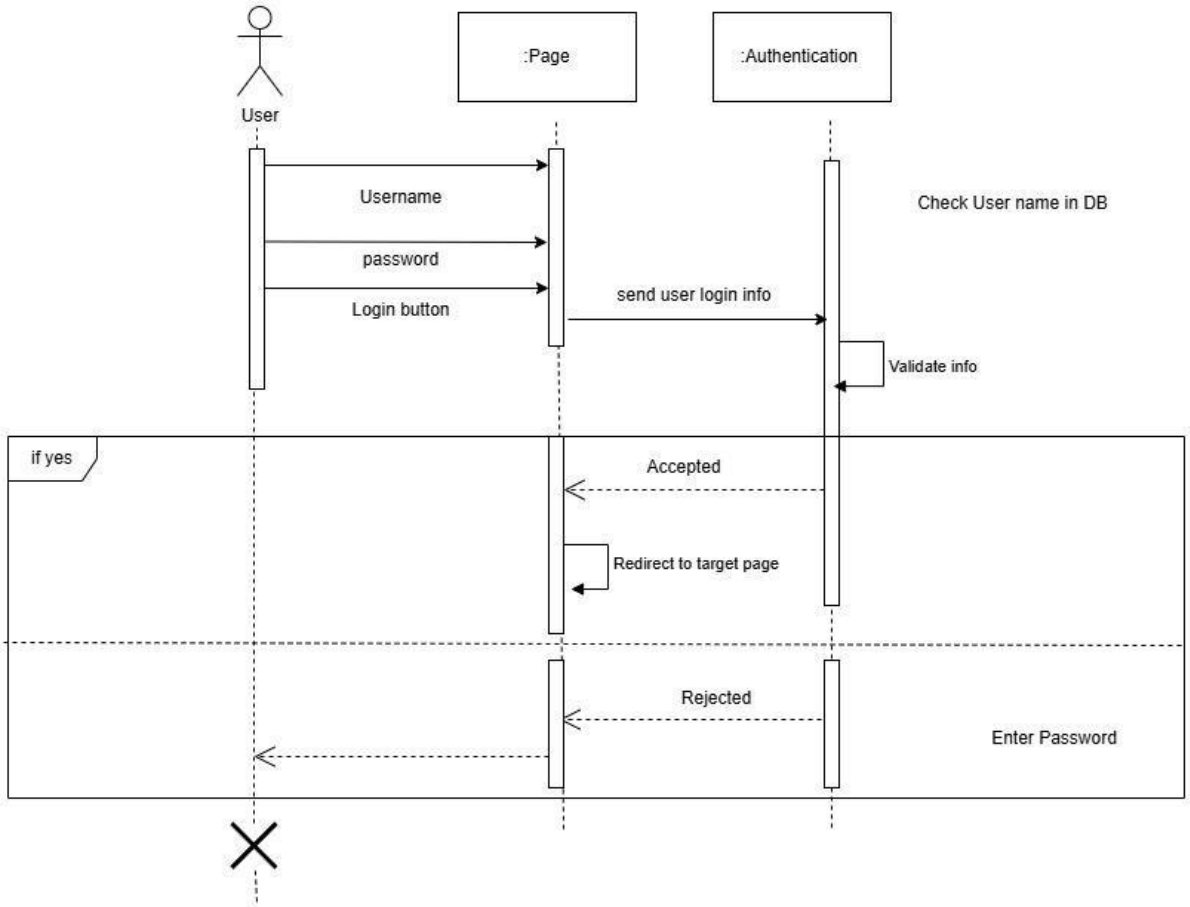


Sequence Diagram

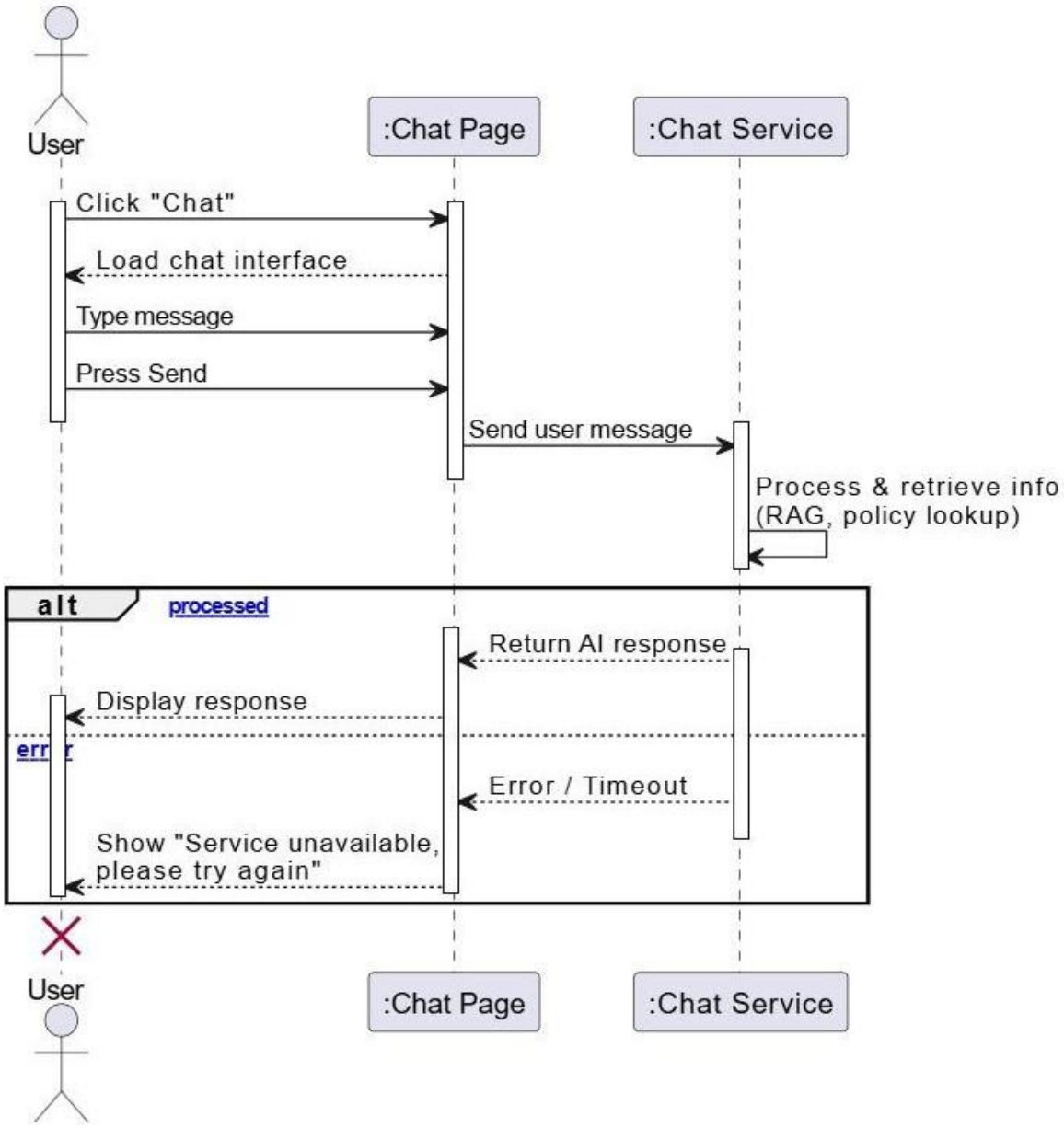
Sequence Diagram - 01: User Registration



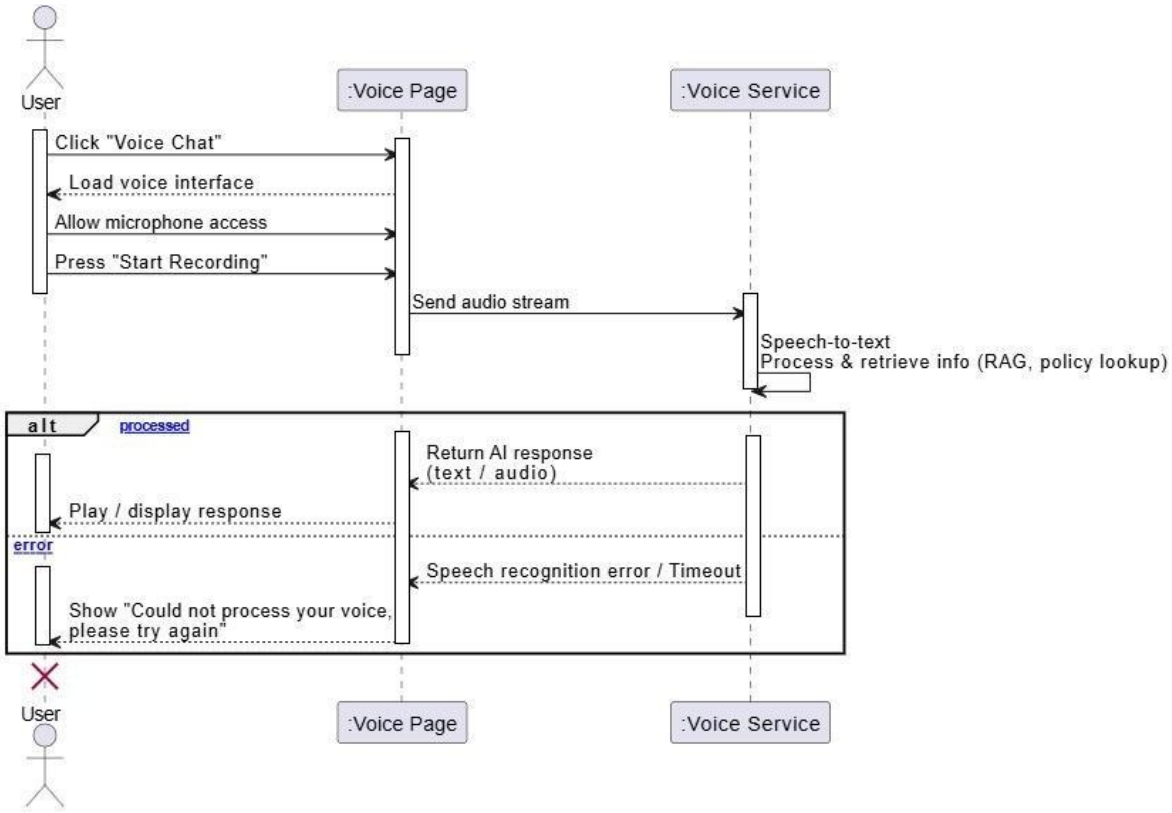
Sequence Diagram - 02: User Login



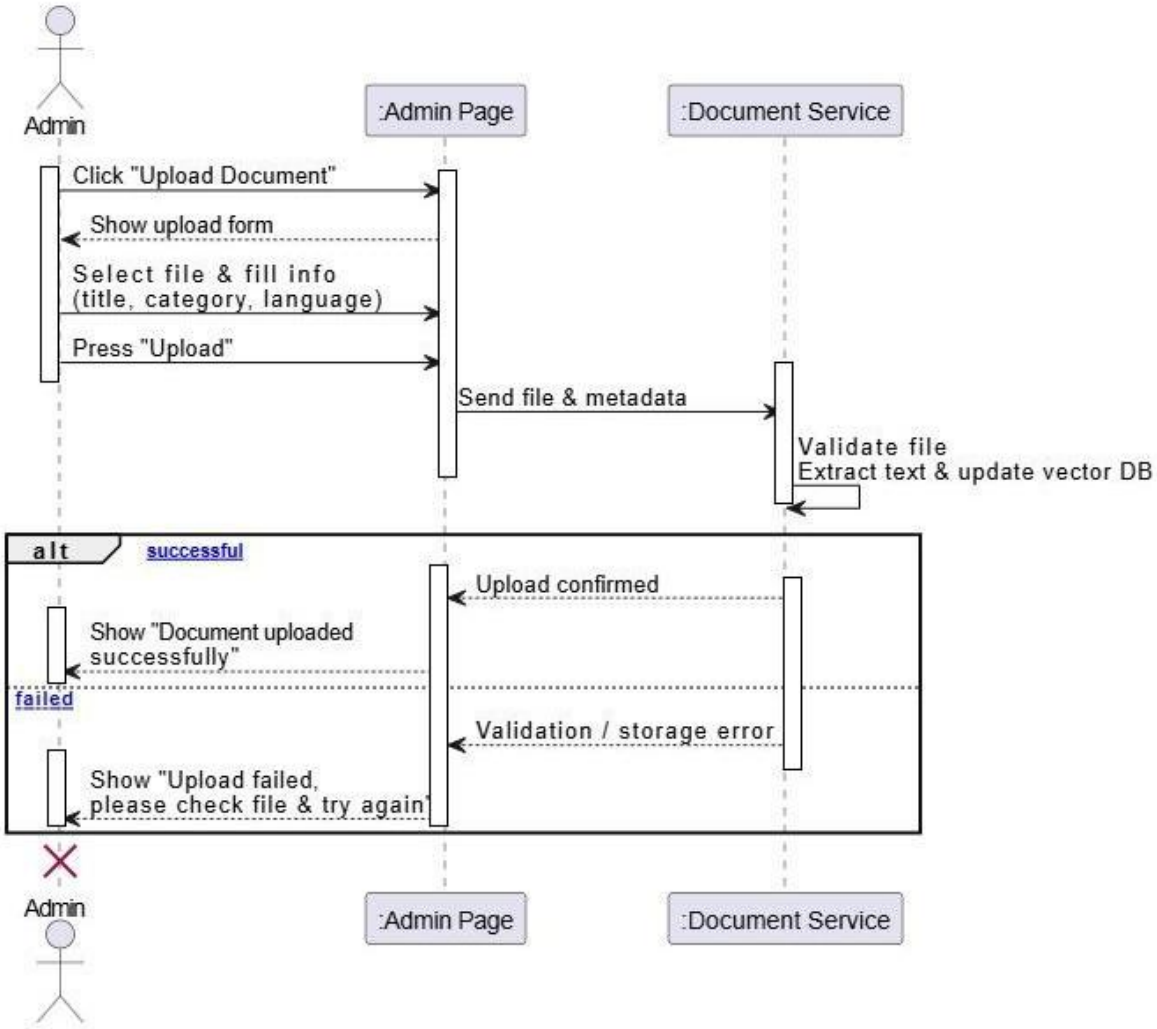
Sequence Diagram - 03: User Chat



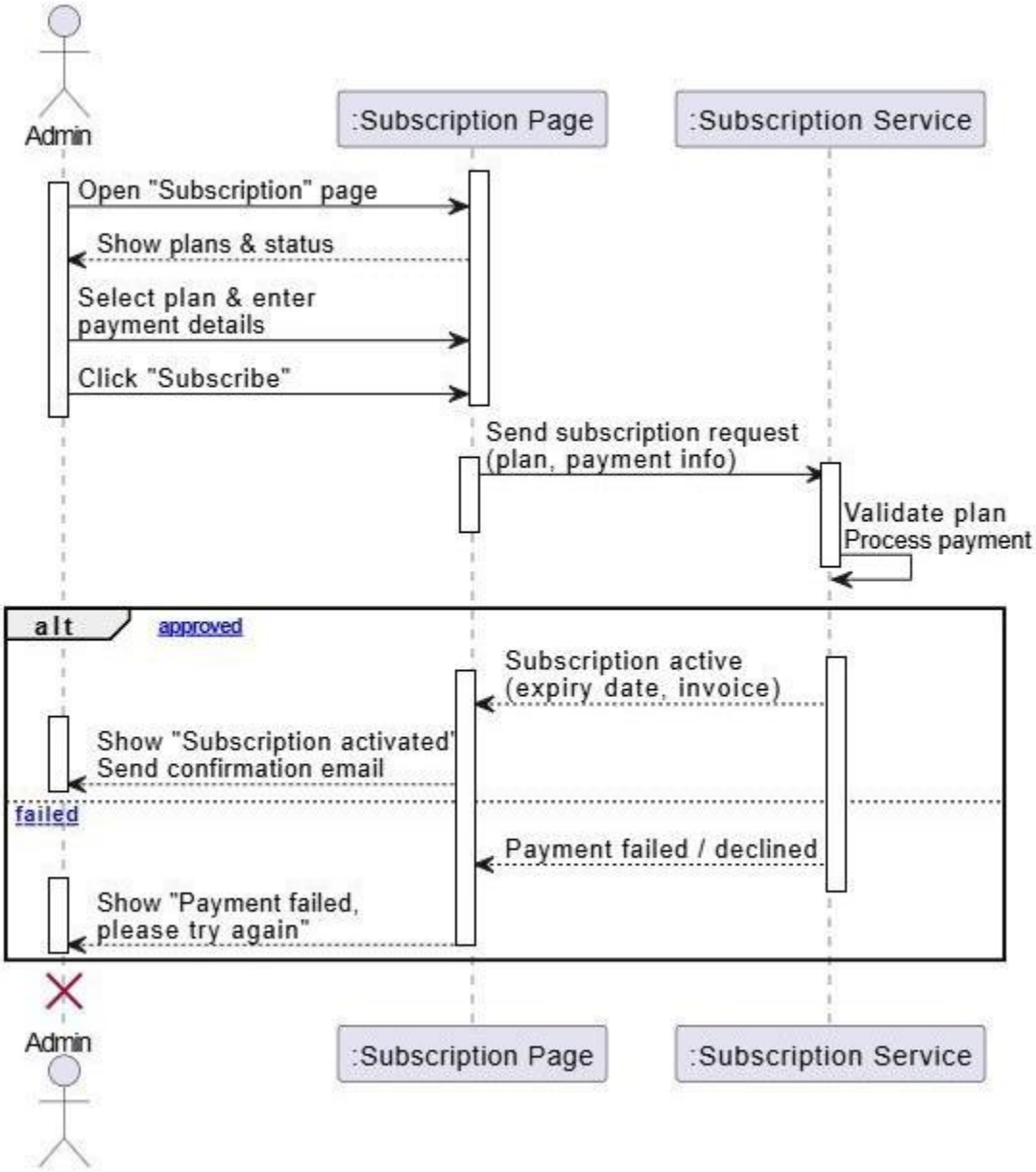
Sequence Diagram - 04: User Voice Chat



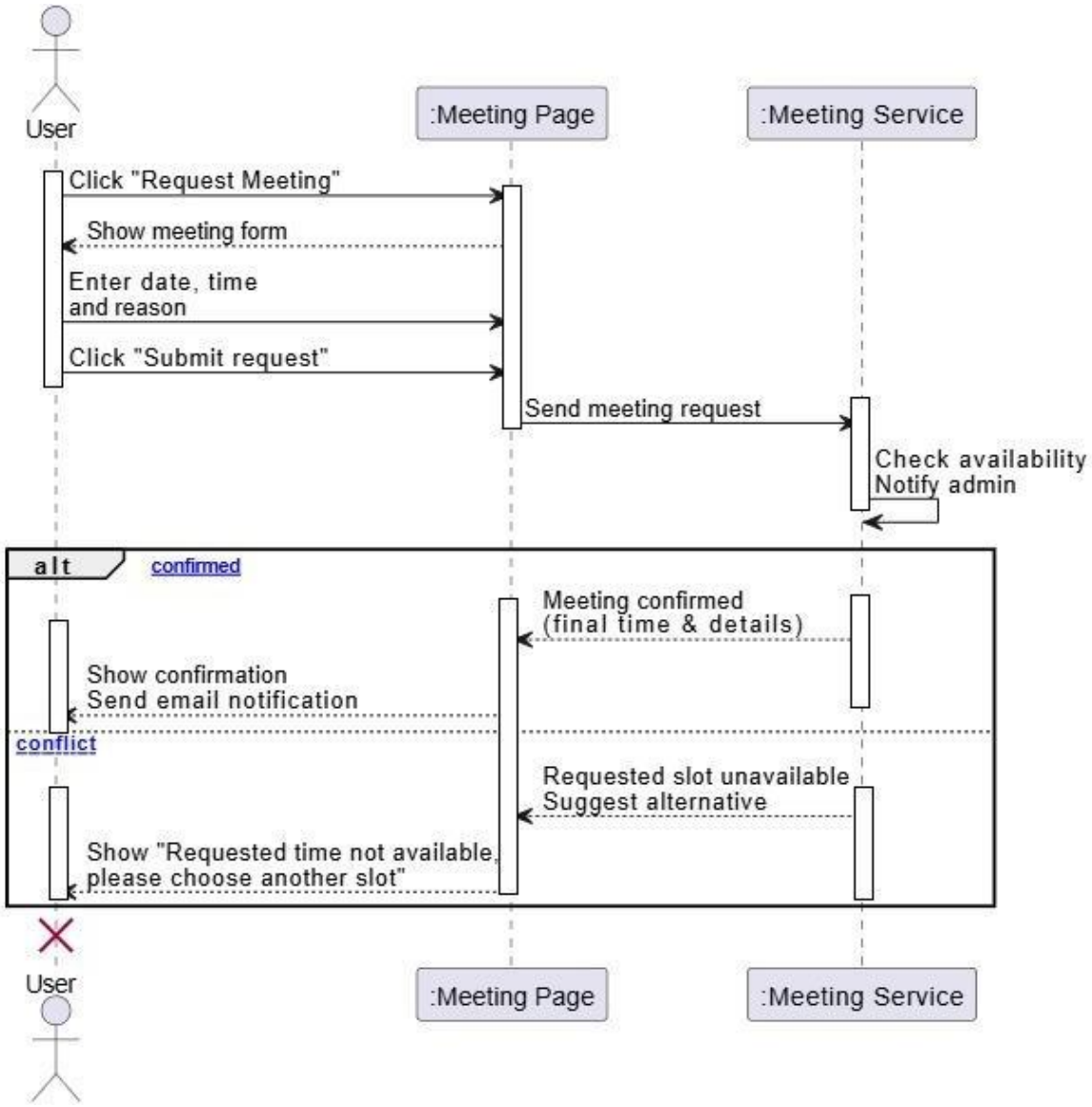
Sequence Diagram - 05: Upload document



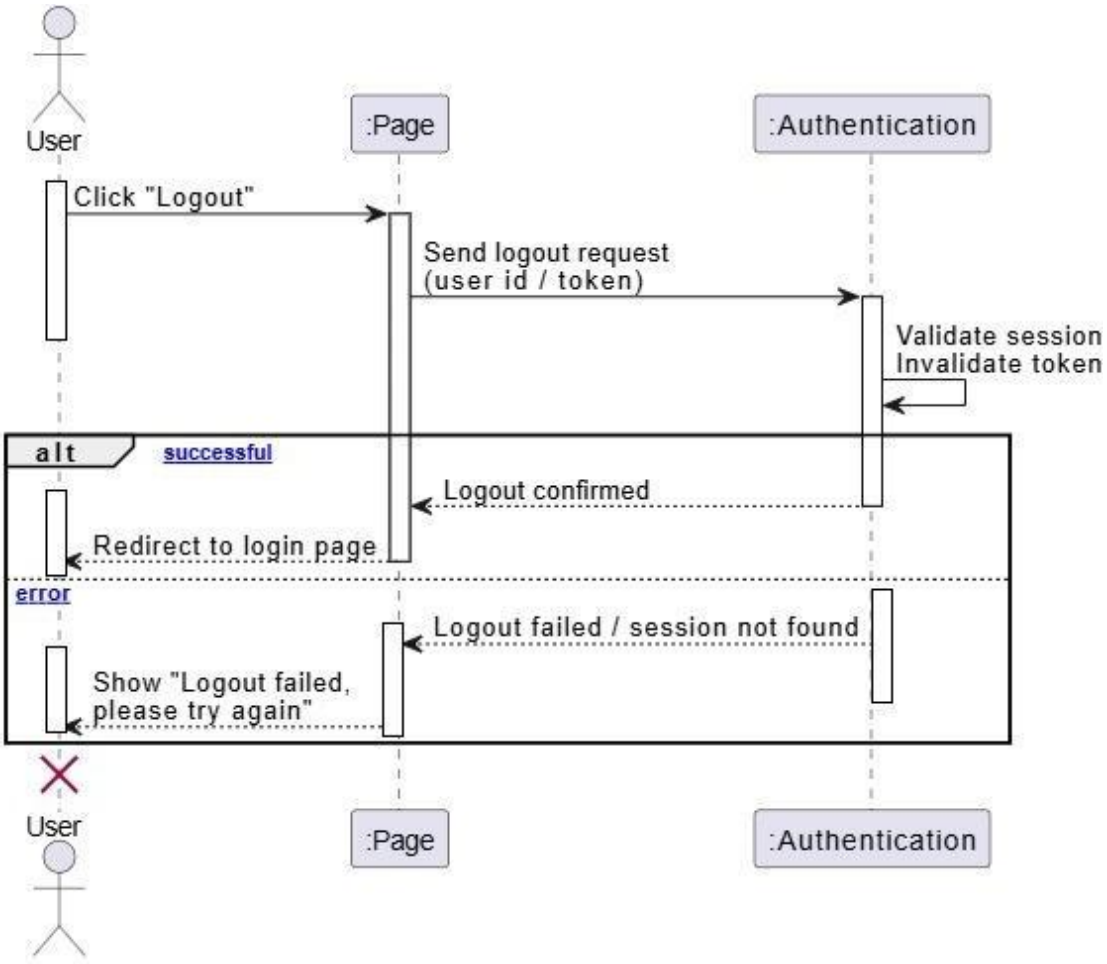
Sequence Diagram - 06: Get Subscription



Sequence Diagram - 07: Meeting Fix



Sequence Diagram - 08: Logout

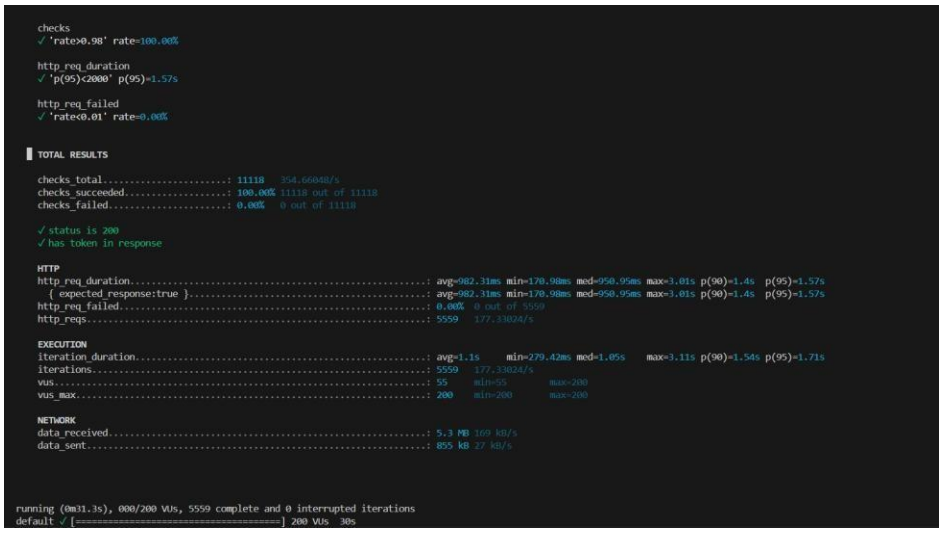


Chapter - 3
Software Testing

Login Route

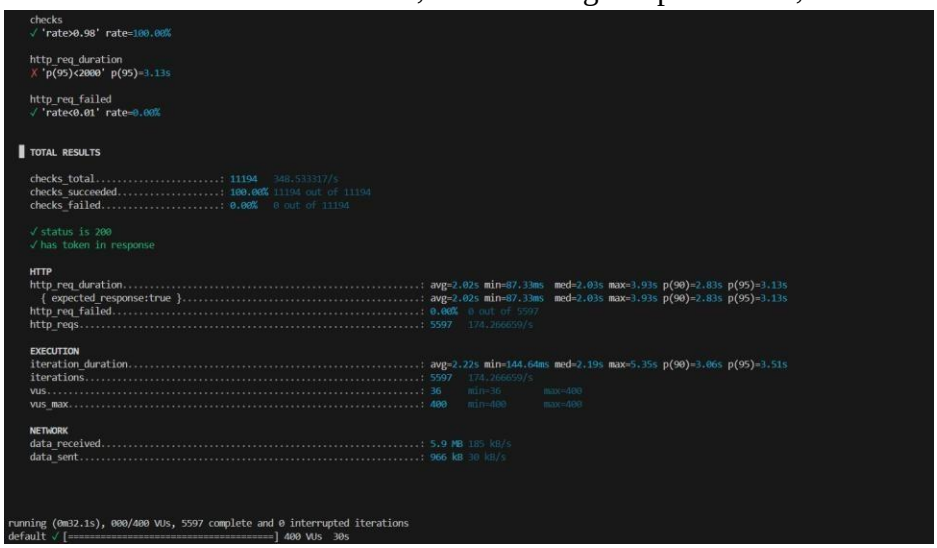
200 User

Load: 200 VUs for 31.3 seconds
Performance: **100**% success rate, **982ms** average response time, **1.57**s 95th percentile



400 Virtual Users

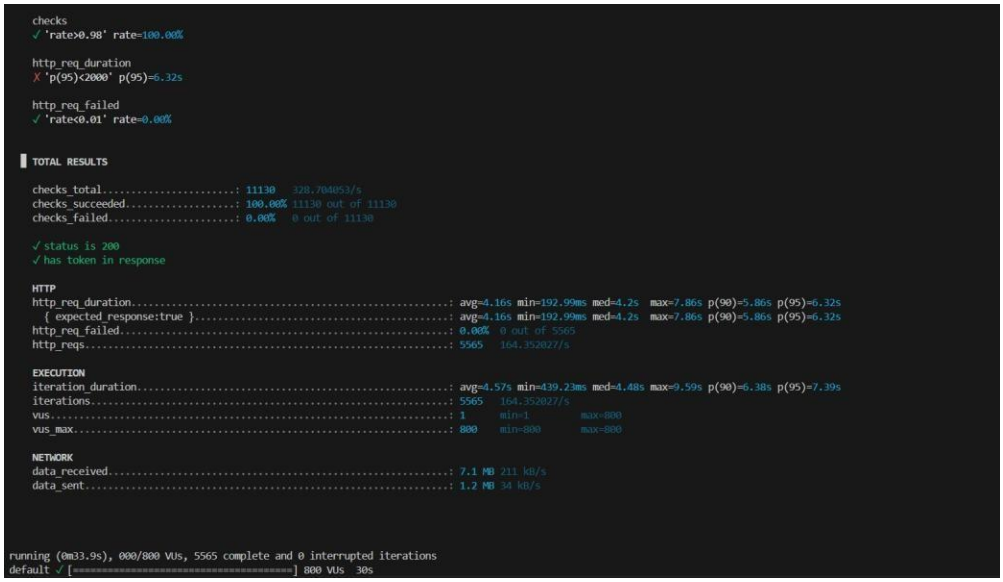
Load: **400** VUs for 32.1 seconds
Performance: **100**% success rate, **2.02**s average response time, **3.13**s 95th percentile



800 Virtual Users

Load: **800** VUs for **33.9** seconds

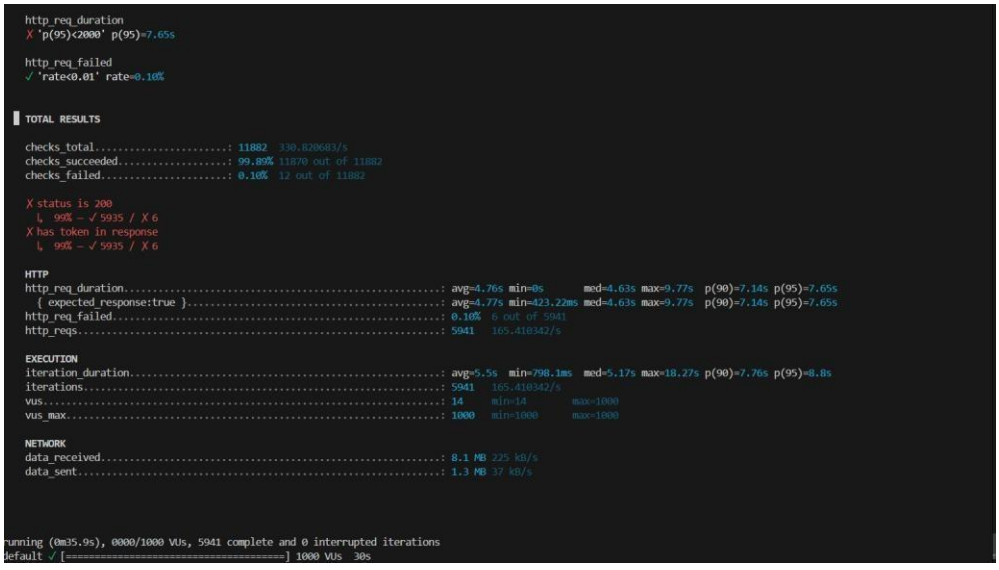
Performance: **100**% success rate, **4.16**s average response time, **6.32**s 95th percentile



1000 Virtual Users

Load: **1000** VUs for **35.95** seconds

Performance: **99.89**% success rate, **4.76**s average response time, **7.65**s 95th percentile

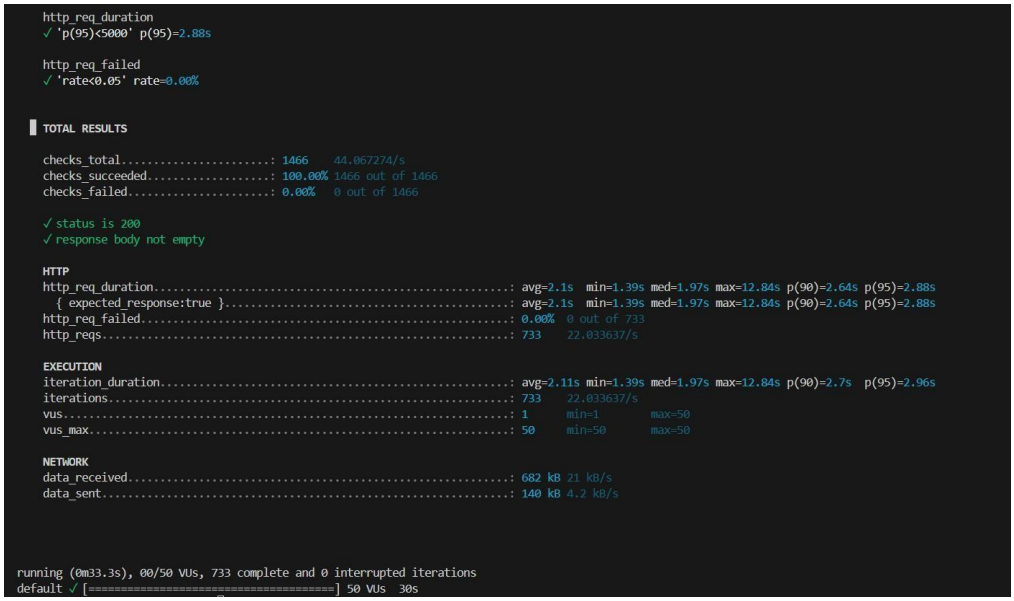


Chat Route

50 Virtual Users

Load: **50** VUs for **33.3** seconds

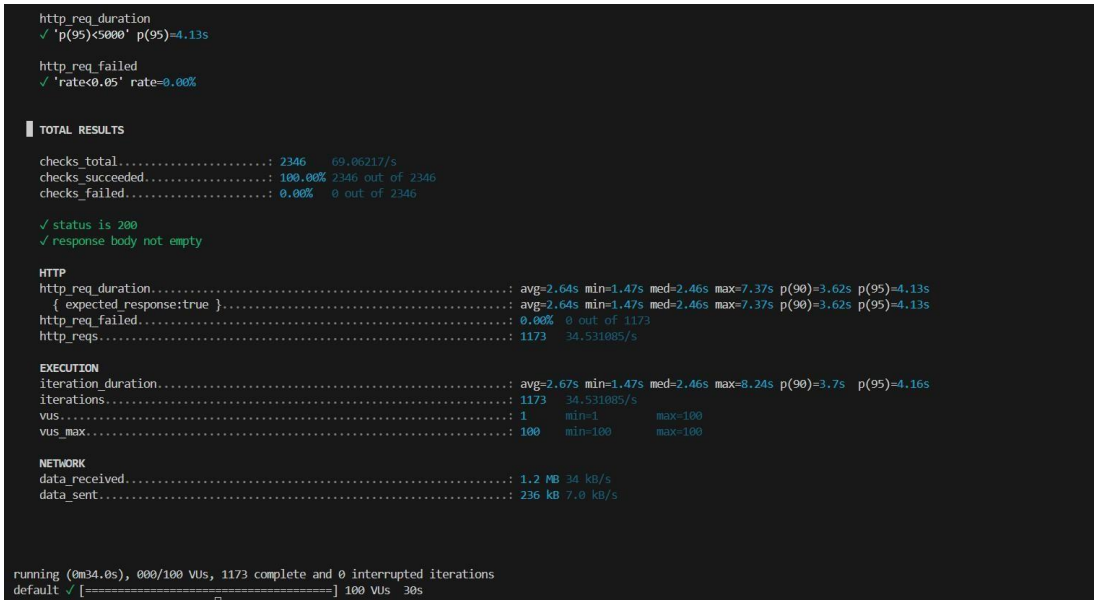
Performance: **100**% success rate, **2.1**s average response time, **2.88**s 95th percentile



100 Virtual Users

Load: **100** VUs for **34.0** seconds

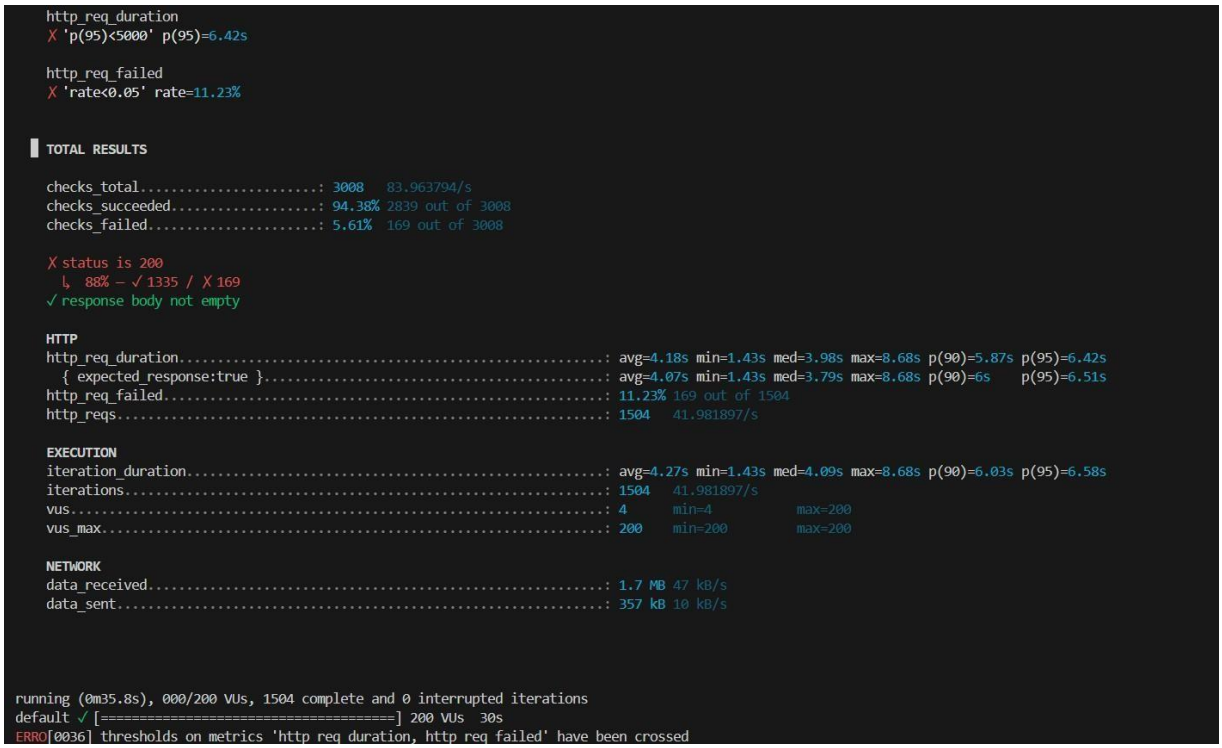
Performance: **100**% success rate, **2.64**s average response time, **4.13**s 95th percentile



200 Virtual Users

Load: **200** VUs for 35.8 seconds

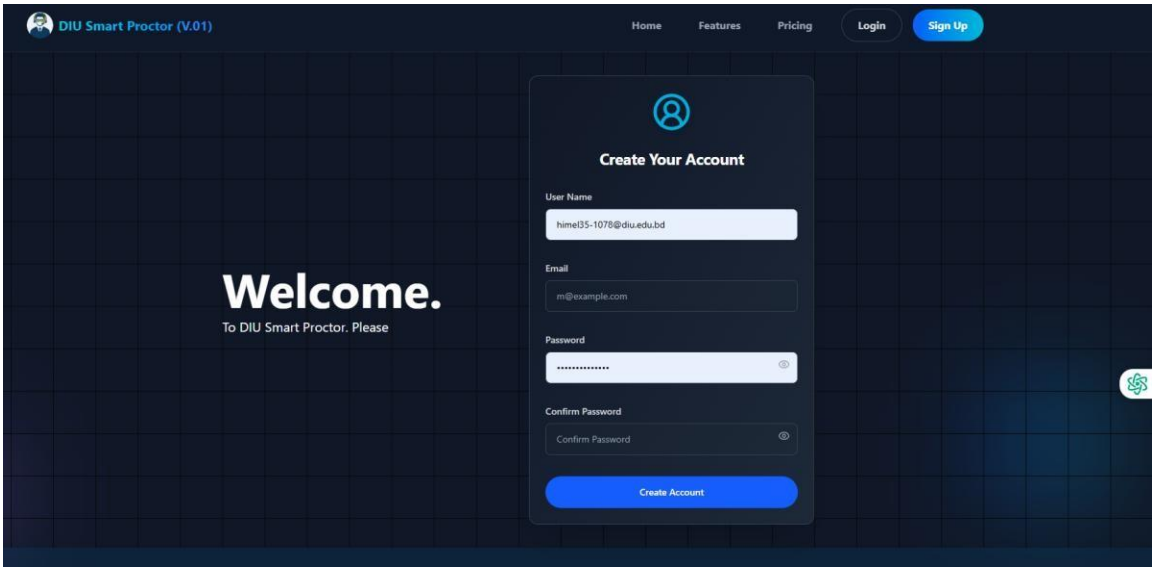
Performance: **94.38**% success rate, **4.18**s average response time, **6.42**s 95th percentile



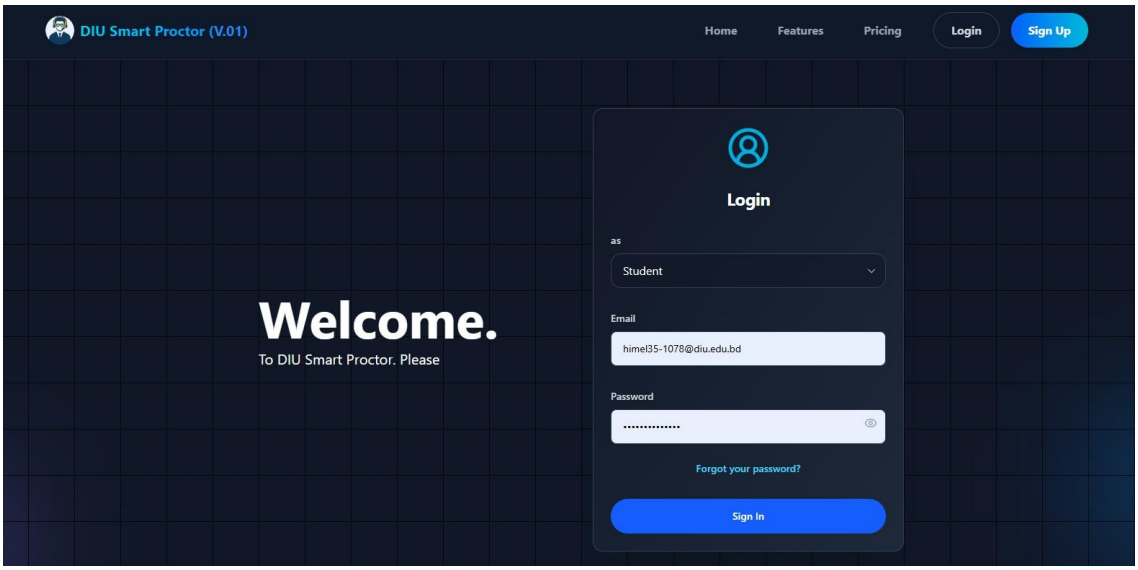
Chapter – 4

User Manual

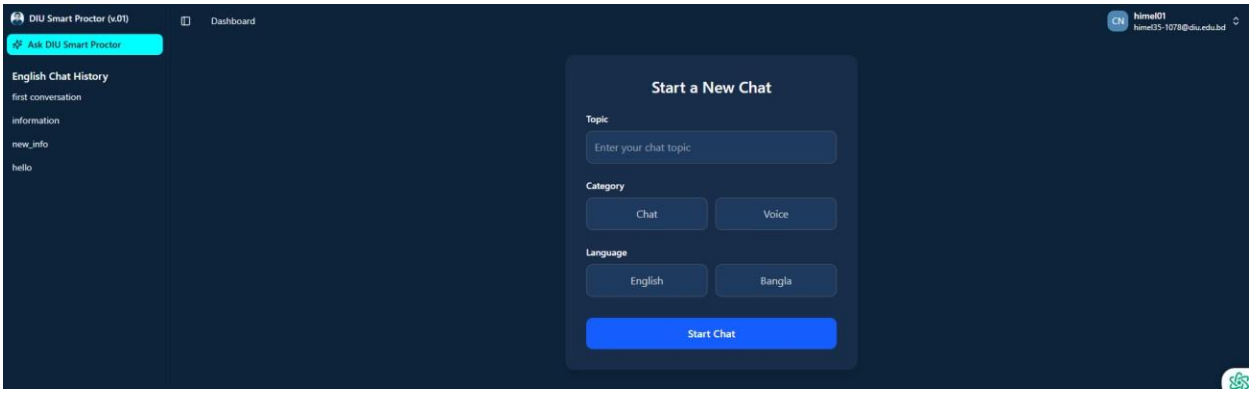
User Registration:



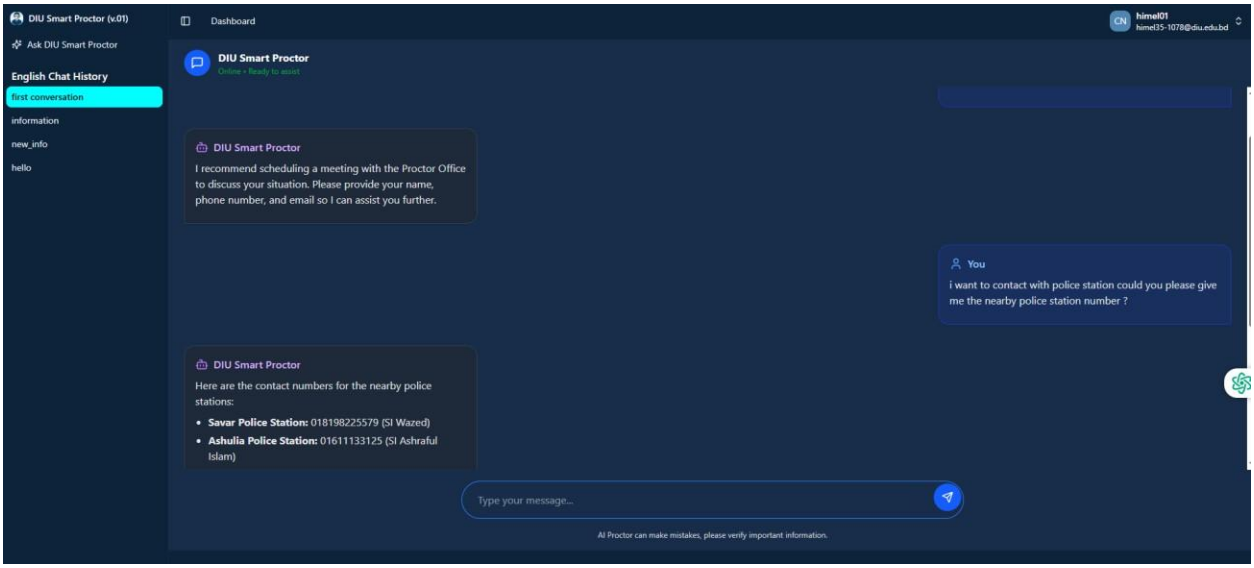
User Login:



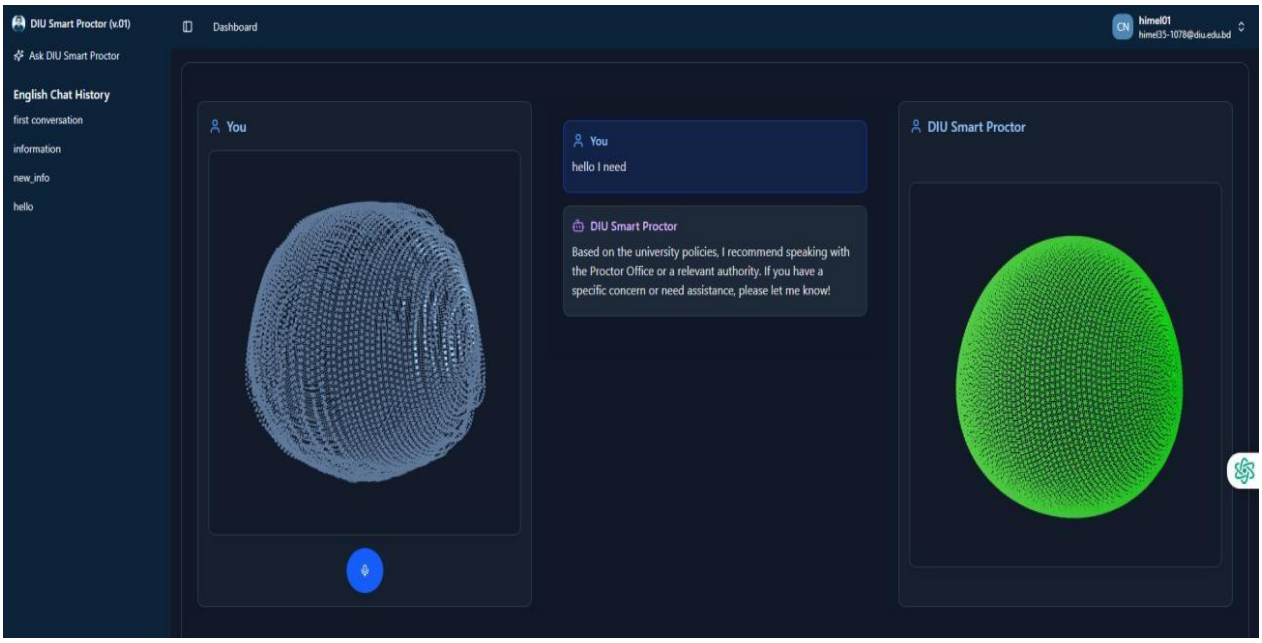
Homepage:



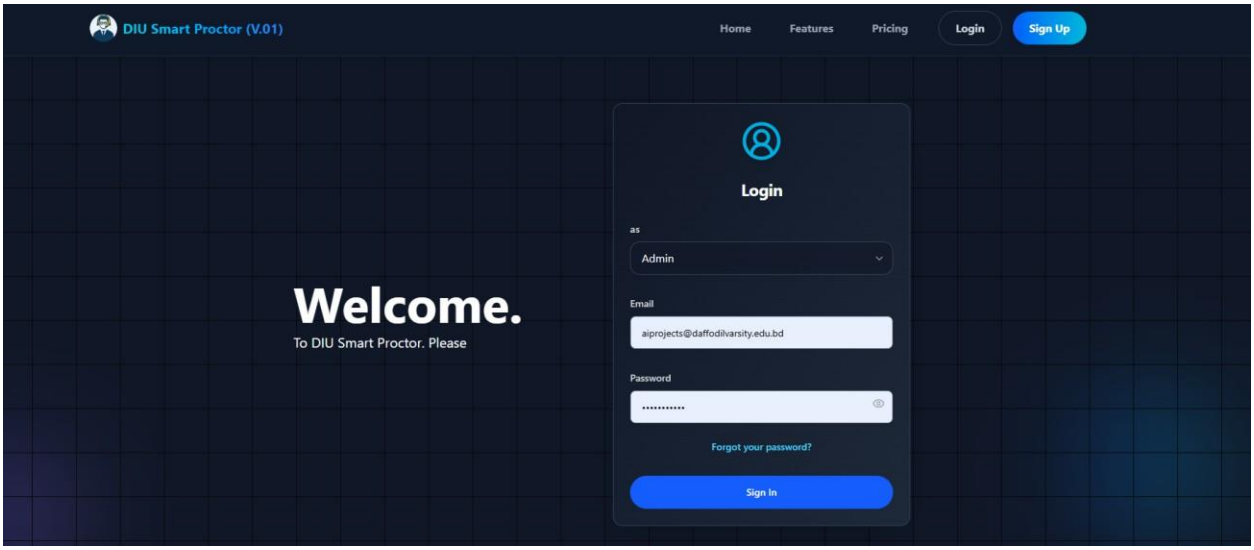
Chat:



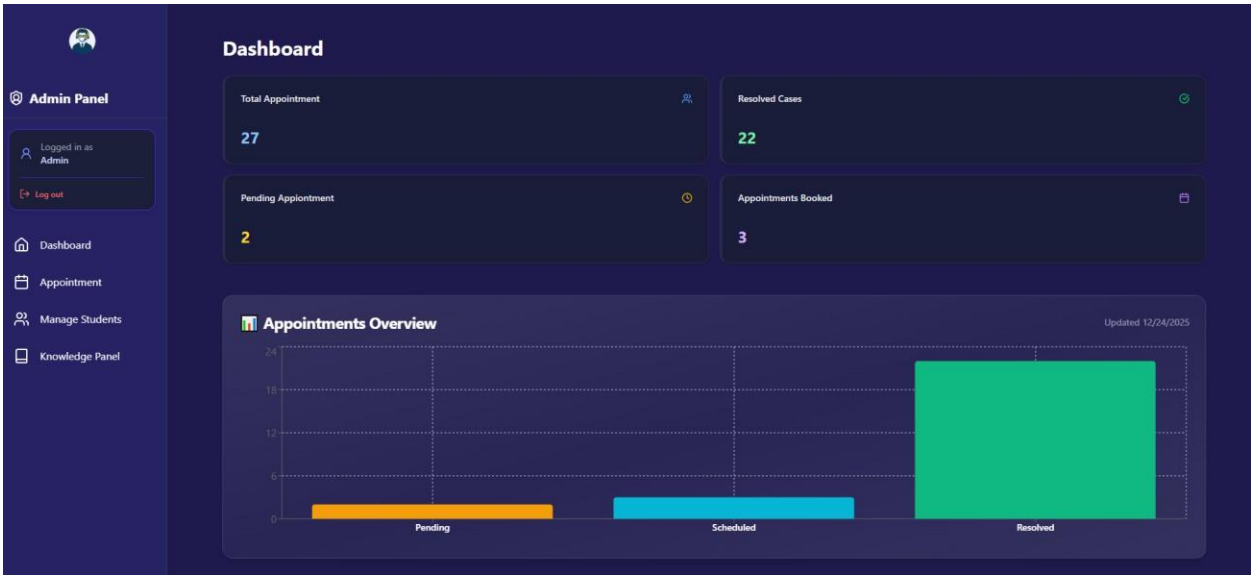
Voice:



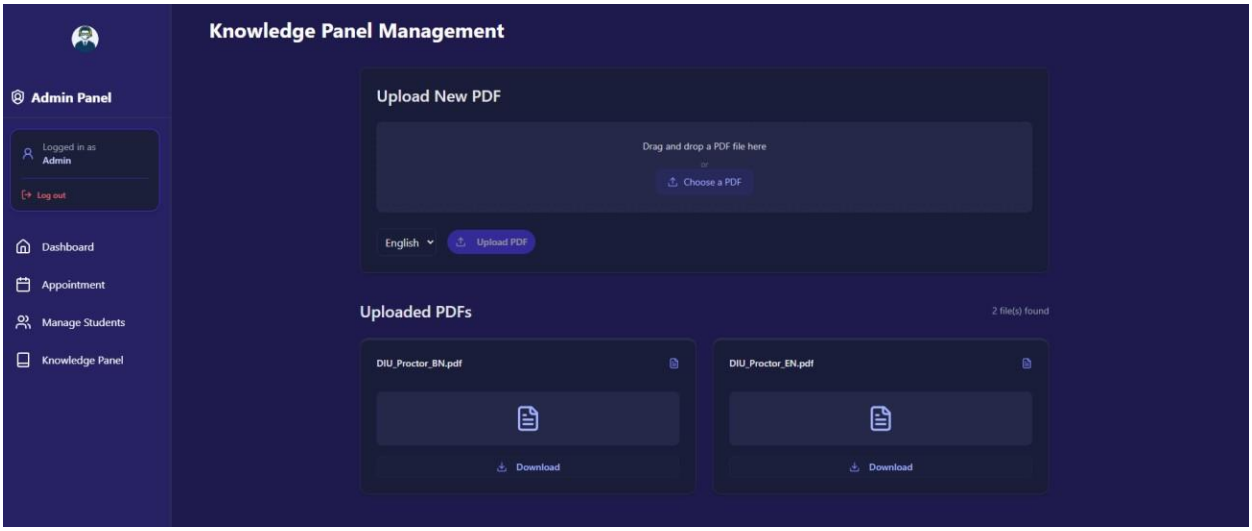
Admin Login:



Admin Panel:



Knowledge Base:



Chapter – 5

Project Summary

Introduction:

This is about the design of an AI-based proctor support system that assists university students any time even after office hours. It supports Bangla as well as English languages, and it allows students to question, inquire about rules, and also request a meeting with the proctor office through chat as well as voice communication.

Project Limitation:

Based on the limited timeframe, budget, and the limited personnel involved, not all the ideas that came up were included during the development of this version of the system. Some additional sophisticated functionalities, such as integration with all the softwares available within the university, detailed dashboards, etc., have not yet been implemented.

Scope

At present, this project involves handling user registration and login functionality, bi-lingual chat and voice services, search functionality for policy documents, meeting scheduling functionality, notification functionality, and multi-tenant functionality for various universities. It has not yet involved mobile apps, AI-based call handling functionality, or complex reporting tools for admins

Future Work

In the future, an AI call agent can be incorporated so that students can make a normal call and receive assistance from AI without even using a browser. The speech support of the Bangla language can be enhanced so that it can understand any accent, slang, and speech pattern of Bangladesh

Conclusion

This project has successfully produced a functional AI Proctor system for students, providing them with quick assistance at all times and also helping staff members manage cases more seamlessly. In the process, they proved the value of clear requirements, simple designs, and robust security for university software.

Reference

1. NVIDIA. (n.d.). *Building AI-based Intelligent Applications*. NVIDIA Deep Learning Institute. Retrieved May 24, 2025, from https://learn.nvidia.com/courses/course-detail?course_id=course-v1:DLI+S-FX-15+V1
2. OpenAI. (n.d.). *Embeddings*. OpenAI Platform Documentation. Retrieved May 24, 2025, from https://platform.openai.com/docs/guides/embeddings
3. LangChain. (n.d.). *Introduction to LangChain*. LangChain Documentation. Retrieved May 24, 2025, from https://python.langchain.com/docs/introduction/
4. Phidata. (n.d.). *Email Tool*. PhiData Documentation. Retrieved May 24, 2025, from https://docs.phidata.com/tools/email
5. Hugging Face. (n.d.). *Introduction to Agents*. Hugging Face Learning Center. Retrieved May 24, 2025, from https://huggingface.co/learn/agents-course/en/unit0/introduction
6. The Data Quarry. (n.d.). *Vector Databases and RAG: A Comprehensive Guide*. The Data Quarry. Retrieved May 24, 2025, from https://thedataquarry.com/blog/vector-db-2

Achievement

Daffodil International University

Home Events Academic Campus & Community Media Corner Search...

Home › News › DIU Agentic AI Excellence Awards 2025 distribution ceremony held

DIU Agentic AI Excellence Awards 2025 distribution ceremony held

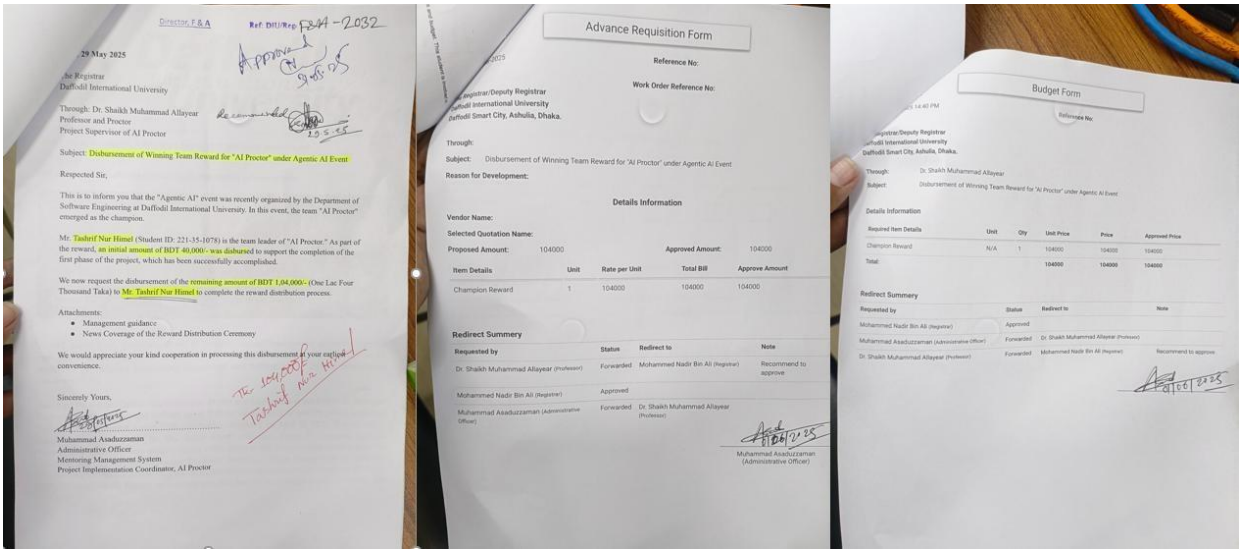
Dr. Md. Sabur Khan, Chairman, Board of Trustees, Daffodil International University handing over the Champion Award to the winners of DIU Agentic Excellence Award 2025 jointly organized by Software Engineering Department and Daffodil HR of Daffodil International University.

Based on the theme -Celebrating innovation, impact and intelligence in the Age of AI, Software Engineering Department and Daffodil HR of Daffodil International University jointly organized DIU Agentic Excellence Award 2025 distribution ceremony on April 12, 2025 at Kazi Nazrul Eduplex of the university. Dr. Md. Sabur Khan, Chairman, Board of Trustees, Daffodil International University was present as the chief guest at the program and handed over the prizes among the winners. Professor Dr. Imran Mahamud, Head, Department of Software Engineering presided over the program. The program was also attended by Professor Dr. Syed Akter Hossain, Dean, Faculty of Science and Information Technology (FSIT), Professor Mr Kabir of Civil Engineering Department and Md. Shohel Arman, Assistant Professor & Director, Data Science Lab of the Software Engineering Department.

In the Competition at total of 17good quality project was submitted by the students of which 'AI Proctor' became Champion, 'Libra AI' became 1ST Runner Up and 'Wisdomc AI' became 2nd Runner Up.

Contract

1.



Director, F & A

Ref. DIU/Reg-FAM-2032

29 May 2025

The Registrar
Daffodil International University

Through: Dr. Shaikh Muhammad Allayear
Professor and Proctor
Project Supervisor of AI Proctor

Subject: Disbursement of Winning Team Reward for "AI Proctor" under Agentic AI Event

Respected Sir,

This is to inform you that the "Agentic AI" event was recently organized by the Department of Software Engineering at Daffodil International University. In this event, the team "AI Proctor" emerged as the champion.

Mr. Tashrif Nur Himel (Student ID: 221-35-1078) is the team leader of "AI Proctor." As part of the reward, an initial amount of BDT 40,000/- was disbursed to support the completion of the first phase of the project, which has been successfully accomplished.

We now request the disbursement of the remaining amount of BDT 1,04,000/- (One Lac Four Thousand Taka) to Mr. Tashrif Nur Himel to complete the reward distribution process.

Attachments:
- Management guidance
- News Coverage of the Reward Distribution Ceremony

We would appreciate your kind cooperation in processing this disbursement at your earliest convenience.

Sincerely Yours,

Muhammad Asaduzzaman
Administrative Officer
Mentoring Management System
Project Implementation Coordinator, AI Proctor

Tk. 104,000/-
Tashrif Nur Himel

Advance Requisition Form

Reference No:

Work Order Reference No:

Registrar/Deputy Registrar
Daffodil International University
Daffodil Smart City, Ashulia, Dhaka.

Through:

Subject: Disbursement of Winning Team Reward for "AI Proctor" under Agentic AI Event

Reason for Development:

Details Information

Vendor Name:

Selected Quotation Name:

Proposed Amount: 104000 Approved Amount: 104000

Item Details	Unit	Rate per Unit	Total Bill	Approve Amount
Champion Reward	1	104000	104000	104000

Redirect Summery

Requested by	Status	Redirect to	Note
Dr. Shaikh Muhammad Allayear (Professor)	Forwarded	Mohammed Nadir Bin Ali (Registrar)	Recommend to approve
Mohammed Nadir Bin Ali (Registrar)	Approved		
Muhammad Asaduzzaman (Administrative Officer)	Forwarded	Dr. Shaikh Muhammad Allayear (Professor)	

Muhammad Asaduzzaman
(Administrative Officer)

Budget Form

Registrar/Deputy Registrar
Daffodil International University
Daffodil Smart City, Ashulia, Dhaka.

Through: Dr. Shaikh Muhammad Allayear

Subject: Disbursement of Winning Team Reward for "AI Proctor" under Agentic AI Event

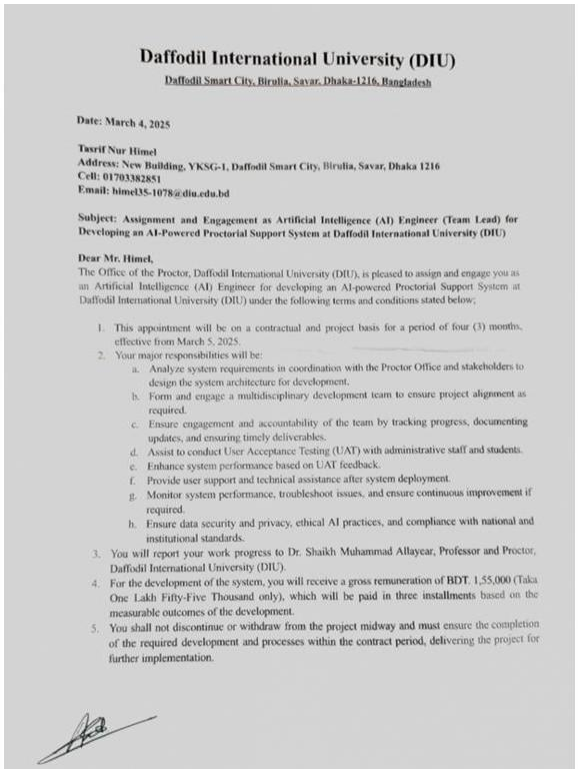
Details Information

Required Item Details	Unit	Qty	Unit Price	Price	Approved Price
Champion Reward	N/A	1	104000	104000	104000
Total:			104000	104000	104000

Redirect Summery

Requested by	Status	Redirect to	Note
Mohammed Nadir Bin Ali (Registrar)	Approved		
Muhammad Asaduzzaman (Administrative Officer)	Forwarded	Dr. Shaikh Muhammad Allayear (Professor)	
Dr. Shaikh Muhammad Allayear (Professor)	Forwarded	Mohammed Nadir Bin Ali (Registrar)	Recommend to approve

2.



Daffodil International University (DIU)
Daffodil Smart City, Birulia, Savar, Dhaka-1216, Bangladesh

Date: March 4, 2025

Tasrif Nur Himel
Address: New Building, YKSG-1, Daffodil Smart City, Birulia, Savar, Dhaka 1216
Cell: 01703382851
Email: himel35-1078@diu.edu.bd

Subject: Assignment and Engagement as Artificial Intelligence (AI) Engineer (Team Lead) for Developing an AI-Powered Proctorial Support System at Daffodil International University (DIU)

Dear Mr. Himel,
The Office of the Proctor, Daffodil International University (DIU), is pleased to assign and engage you as an Artificial Intelligence (AI) Engineer for developing an AI-powered Proctorial Support System at Daffodil International University (DIU) under the following terms and conditions stated below;

1. This appointment will be on a contractual and project basis for a period of four (3) months, effective from March 5, 2025.
2. Your major responsibilities will be:
   a. Analyze system requirements in coordination with the Proctor Office and stakeholders to design the system architecture for development.
   b. Form and engage a multidisciplinary development team to ensure project alignment as required.
   c. Ensure engagement and accountability of the team by tracking progress, documenting updates, and ensuring timely deliverables.
   d. Assist to conduct User Acceptance Testing (UAT) with administrative staff and students.
   e. Enhance system performance based on UAT feedback.
   f. Provide user support and technical assistance after system deployment.
   g. Monitor system performance, troubleshoot issues, and ensure continuous improvement if required.
   h. Ensure data security and privacy, ethical AI practices, and compliance with national and institutional standards.
3. You will report your work progress to Dr. Shaikh Muhammad Allayear, Professor and Proctor, Daffodil International University (DIU).
4. For the development of the system, you will receive a gross remuneration of BDT. 1,55,000 (Taka One Lakh Fifty-Five Thousand only), which will be paid in three installments based on the measurable outcomes of the development.
5. You shall not discontinue or withdraw from the project midway and must ensure the completion of the required development and processes within the contract period, delivering the project for further implementation.

Thanking You,

Muhammad Asaduzzaman
Administrative Officer
Office of the Proctor
Daffodil International University

Acknowledgment and Acceptance:

I, Tasrif Nur Himel, hereby acknowledge and accept the offer of appointment as Artificial Intelligence (AI) Engineer for the development of the AI-Powered Proctorial Support System at Daffodil International University (DIU), along with the terms and conditions outlined above.

Signature: Himel

Date: 04/03/2025

221-35-1078

ORIGINALITY REPORT

7%	2%	2%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	2%
2	Submitted to Daffodil International University Student Paper	2%
3	Submitted to Domain Academy Student Paper	<1%
4	Submitted to Gulf College Oman Student Paper	<1%
5	Submitted to University College London Student Paper	<1%
6	Submitted to University of Belgrade, Faculty of Organizational Sciences Student Paper	<1%
7	Submitted to University of New England Student Paper	<1%
8	devby.io Internet Source	<1%
9	Submitted to University of Bedfordshire Student Paper	<1%
10	Submitted to University of Derby Student Paper	<1%

11	Submitted to University of Westminster Student Paper	< 1 %
12	Submitted to National Institute of Business Management Sri Lanka Student Paper	< 1 %
13	www.igtr-aur.org Internet Source	< 1 %
14	Submitted to Staffordshire University Student Paper	< 1 %
15	Submitted to University of Technology, Sydney Student Paper	< 1 %
16	Submitted to Southern New Hampshire University – Continuing Education Student Paper	< 1 %
17	Submitted to Higher Education Commission Pakistan Student Paper	< 1 %
18	Submitted to Strategy First Institute Student Paper	< 1 %
19	Submitted to University of Melbourne Student Paper	< 1 %
20	toppodcast.com Internet Source	< 1 %
21	Submitted to Swinburne University of Technology Student Paper	< 1 %
22	dspace.daffodilvarsity.edu.bd:8080	

Internet Source

<1%

23 ojs.aaai.org
Internet Source

<1%

24 AlSammarraie, AlHasan. "Development and Evaluation of an Agentic LLM-Based RAG-Enabled Framework for Evidence-Based Patient Education.", Hamad Bin Khalifa University (Qatar)
Publication

<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		