ELEC5620 Project Stage1

Intelligent Learning Platform

Siyu Chen 530666064

Sirui Li 530548597

Zhaoyu Wang 530454704

Zhehang Xu 540210303

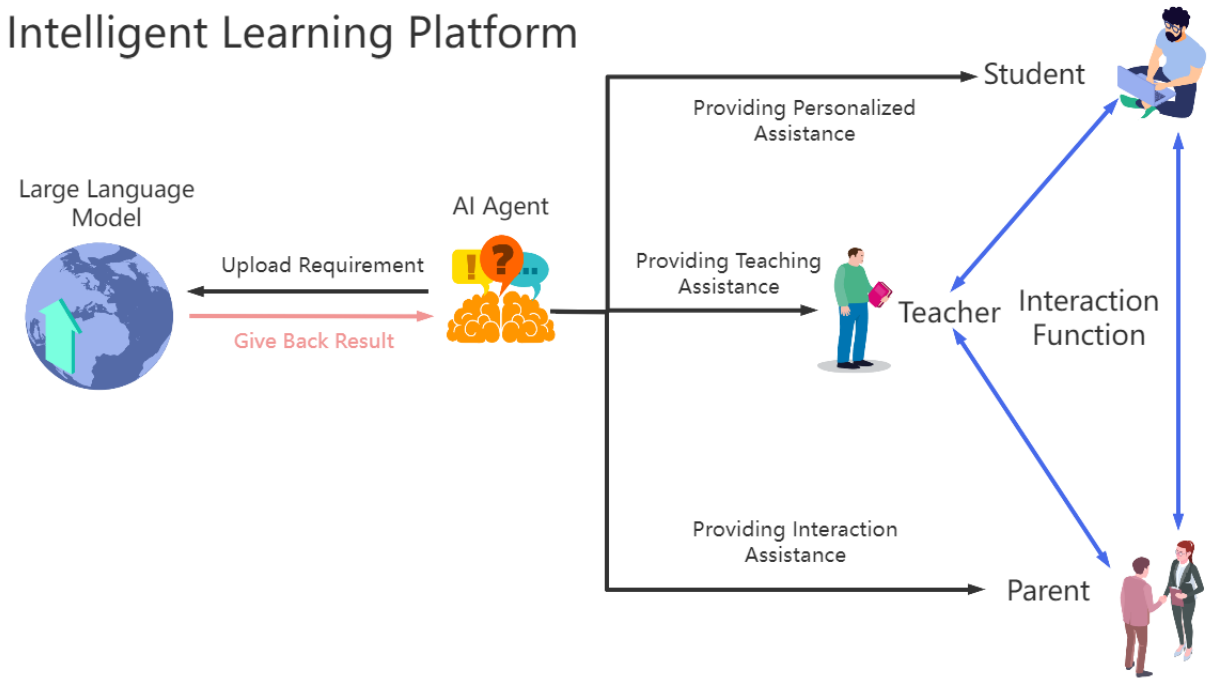
Wenxing Zhou 540347447

Table of Contents

Requirements and Specifications

Ad hoc Requirements (All Members)

In today's rapidly evolving educational environment, one of the key challenges is to provide personalized and adaptive learning experiences to keep up with the diverse and changing needs of students. And the need to strengthen the interaction among teachers, parents and students. Traditional educational methods often have difficulty in providing real-time support, customized learning paths, and tailored resources, resulting in gaps in individual learning outcomes. To address these challenges, our AI agent based intelligent learning platform, which provides dynamic on demand solutions by integrating LLM to meet the changing and specific needs of students, teachers, and parents. In this platform, students, teachers, and parents are end users.
In the next part, ad hoc diagrams and informal descriptions will be used to describe the ad hoc requirements of the system to meet dynamic and immediate needs such as personalized learning, tutoring, and resource recommendations, ensuring that the platform can effectively adapt to the unique needs of users. Here is the general ad hoc diagram for this intelligent learning platform, which consists of three main users and the core AI agent function with LLM.



The next informal description demonstrates the requirement that the platform can provide to the three users and describes the interactions function that the platform provides between the three users.

The platform can generate a highly customized learning plan for students based on their current knowledge level and recent performance metrics. Students can upload data such as their test scores, and recent learning objectives, the platform analyzes incorrect answers and weak areas. The system identifies the key concepts that students struggle with and generates a personalized learning schedule.

This outputted schedule outlines specific content to focus on at times, ensuring students can address their weak points efficiently. **(Zhaoyu Wang)**

Students can engage in real-time interactions with an AI agent to resolve academic queries quickly. The platform supports various input methods, including text, image uploads, and document sharing, enabling students to submit their questions in any format. The AI agent will process the inputs and return a detailed response, offering either direct answers or guiding problem-solving steps. Furthermore, the platform can log all tutoring sessions into a database, allowing students to revisit past questions and responses. The capability of this is crucial as it ensures that immediate student needs are met, providing prompt feedback based on the specific question. The platform can intelligently recommend study resources based on students' learning history and requests based on AI agent. As students' progress through the curriculum and face new challenges, the system can analyze their performance and goals to provide relevant materials. **(Zhehang Xu)**

The platform can provide an approach to personalized teaching by generating a tailored teaching plan for each student based on their background, current performance, and learning preferences. Teachers can input the student's learning data, and the agent will propose a plan that fits their specific learning pace and objectives. It ensures that teaching plans can be frequently updated to reflect real-time performance and changes in learning goals. **(Siyu Chen)**

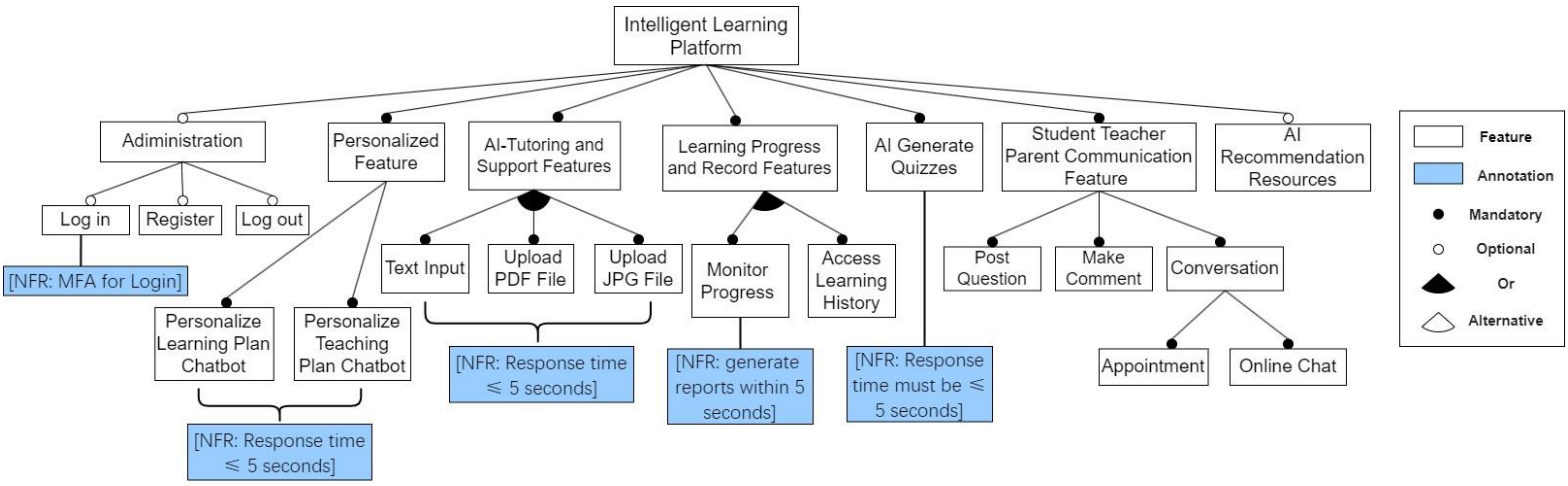
Teachers can create quizzes by specifying parameters such as question topics, types, and difficulty levels through LLM. The system can parse this information and generates corresponding questions. The generated quiz questions are output as text or documents, and teachers can choose their own use of the content. It can allow teachers to adjust in real-time based on student performance or further instruction requirements. It supports continuous customization of quizzes to align with current teaching goals and student competencies. **(Wenxing Zhou)**

The platform can allow students to schedule one-on-one appointments with teachers based on their immediate learning needs. This feature lies in its flexibility, enabling students to request consultations as issues arise, whether related to course content, exam preparation, or other academic concerns. The system can efficiently manage teacher availability and integrates with students' learning schedules to ensure that timely academic support is always accessible. **(Sirui Li)**

The platform can enable teachers and parents to monitor a student's progress through a detailed report system and history data. Teachers can upload exam scores and knowledge areas where the student is struggling. Parents can access their children's learning progress and information which teacher provided. The progress report will help parents and teachers develop more targeted educational strategies, providing an overview of both immediate and long-term academic needs. **(Sirui Li)**

Feature Diagram (Zhehang Xu, Wenxing Zhou)

The Intelligent Learning Platform is designed with a comprehensive set of features aimed at enhancing the learning experience and interaction for students, teachers, and parents. This feature diagram consists of 5 main mandatory features with 11 mandatory sub-features, and 4 optional features.



Non-Functional Requirements (NFRs) will also be shown below.

Personalized Feature:

- [NFR: Response time must be ⩽ 5 seconds after user input]

AI-Tutoring and Support Features:

- [NFR: Response time must be ⩽ 5 seconds after user input]

Learning Progress and Record Features:

- [NFR: The system should be able to generate progress reports within 5 seconds]

AI Generate Quizzes:

- [NFR: Response time must be ⩽ 5 seconds after user input]

Administration:

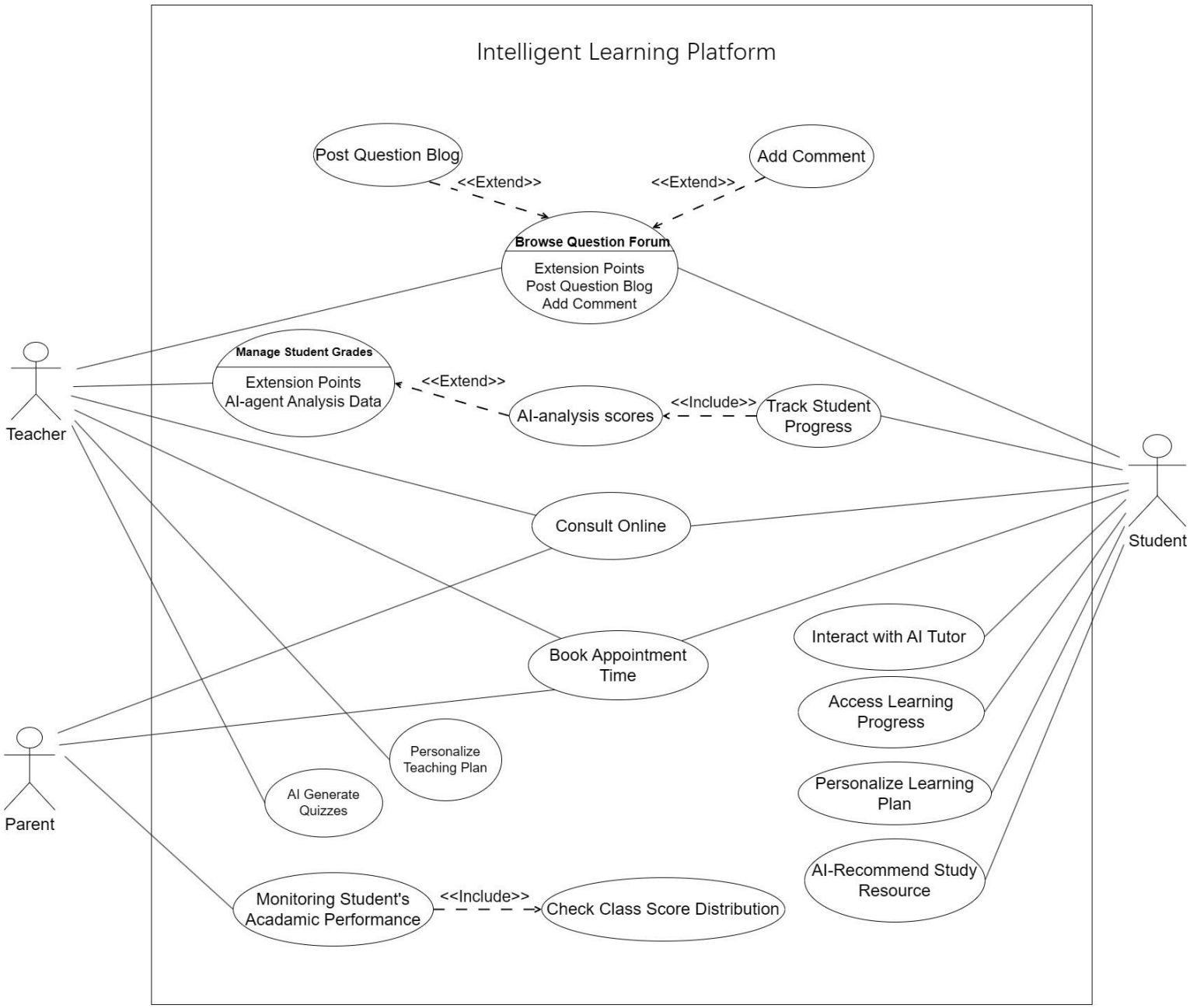
- [NFR: Multi-factor authentication (MFA) is required for Login]

All Data Transmission Related Feature and Login/Logout/Register Feature:

- [NFR: Data Transmission must be protected by asymmetric encryption]

Use Case Diagram (Zhaoyu Wang, Sirui Li, Siyu Chen)

This part demonstrates the comprehensive use case diagram that encompasses all architects and users.



Use Case Specification (All Members)

This part shows a total of 13 use case specifications, followed by a description of the use case, which is a collection of scenarios corresponding to successful runs through the use case. Also, for complex use cases, success and failure diagrams are included below.

Use Case 1 (Zhaoyu Wang)

Name:	**Personalize Learning Plan**
Goal:	Obtain learning plan
Level:	User Goal
Precondition:	Student input learning requirement to system
Success Condition:	Learning plan is generated successfully to user
Failure Condition:	System did not give any responses despite valid request
Trigger:	Student input request and click “submit”

1. **Student Submits Learning Requirements**
   The student inputs their learning needs or preferences into the system and clicks "submit."
2. **System Receives the Request**
   The system processes the student's learning requirements, analyzing the inputs such as subjects of interest, learning goals, and timeline.
3. **System Generates Personalized Learning Plan**
   Based on the student’s input, the system creates a personalized learning plan tailored to the student's requirements, including recommended courses, study resources, and a timeline for completion.
4. **Learning Plan Delivered to the Student**
   The generated learning plan is displayed to the student, either as a downloadable document or a viewable summary within the system.
5. **Notification of Success**
   The system notifies the student that their personalized learning plan has been successfully generated and is available for review.

Use Case 2 (Zhehang Xu)

Name	**Interact with AI tutor**
Goal	Obtain solution of the question
Level	User Goal
Precondition	Student submit the question to system
Success Condition	Solution to the question generated successfully

Failure Condition	System did not give any responses despite valid request
Trigger:	Student input question and click “submit”

1. **Student Submits a Question**
   The student enters a question into the system and clicks the "submit" button.
2. **System Receives the Question**
   The system receives the submitted question and begins processing it using the AI tutoring engine.
3. **AI Engine Generates a Solution**
   The AI engine analyzes the question and generates a solution based on its internal knowledge or algorithm.
4. **Solution Delivered to Student**
   Once the solution is generated, it is displayed to the student through the interface.
5. **Notification of Success**
   The system notifies the student that the solution to their question has been successfully generated and is ready for review.

Use Case 3 (Zhaoyu Wang)

Name	**Book Appointment Time**
Goal	Successfully schedule an onsite consultation with a teacher
Level	User Goal
Precondition	The teacher has an available slot at the time selected by the student for the appointment.
Success Condition	The student successfully schedules an onsite consultation with the teacher, receives a confirmation, and both the student and teacher are notified.
Failure Condition	The consultation time conflicts with the teacher's availability or a system error occurs, preventing the appointment from being scheduled.
Trigger	The student initiates an appointment request in the system.

1. **Student Views Teacher's Availability**
   The student logs into the system and checks the teacher's available time slots. The system displays the teacher’s schedule, allowing the student to choose a suitable time.
2. **Student Selects an Appointment Time**
   The student selects an available time based on their own schedule and submits the appointment request.
3. **System Verifies Teacher’s Availability**
   The system checks if the selected time conflicts with the teacher’s schedule. If there is no conflict, it proceeds to the next step. If a conflict exists, the student is prompted to select another time.
4. **System Generates Appointment Confirmation**
   If the time is valid, the system generates a confirmation for the appointment, which includes details such as the time and a meeting link.

5. **Notification Sent**
   The system sends a notification to both the student and the teacher, usually via email or an internal system notification, confirming the appointment.
6. **Appointment Information Logged**
   The system records the appointment details in the background, including the student's and teacher's IDs, appointment time, and appointment status, for future reference and management.

Use Case 4 (Siyu Chen)

Name	**AI-Recommend Study Resource**
Goal	Obtain study resource
Level	User Goal
Precondition	Student input request (learning background, current challenges, and learning goals) to system
Success Condition	Study resource is generated successfully to user
Failure Condition:	System did not give any responses despite valid request
Trigger:	Student input request and click "submit"

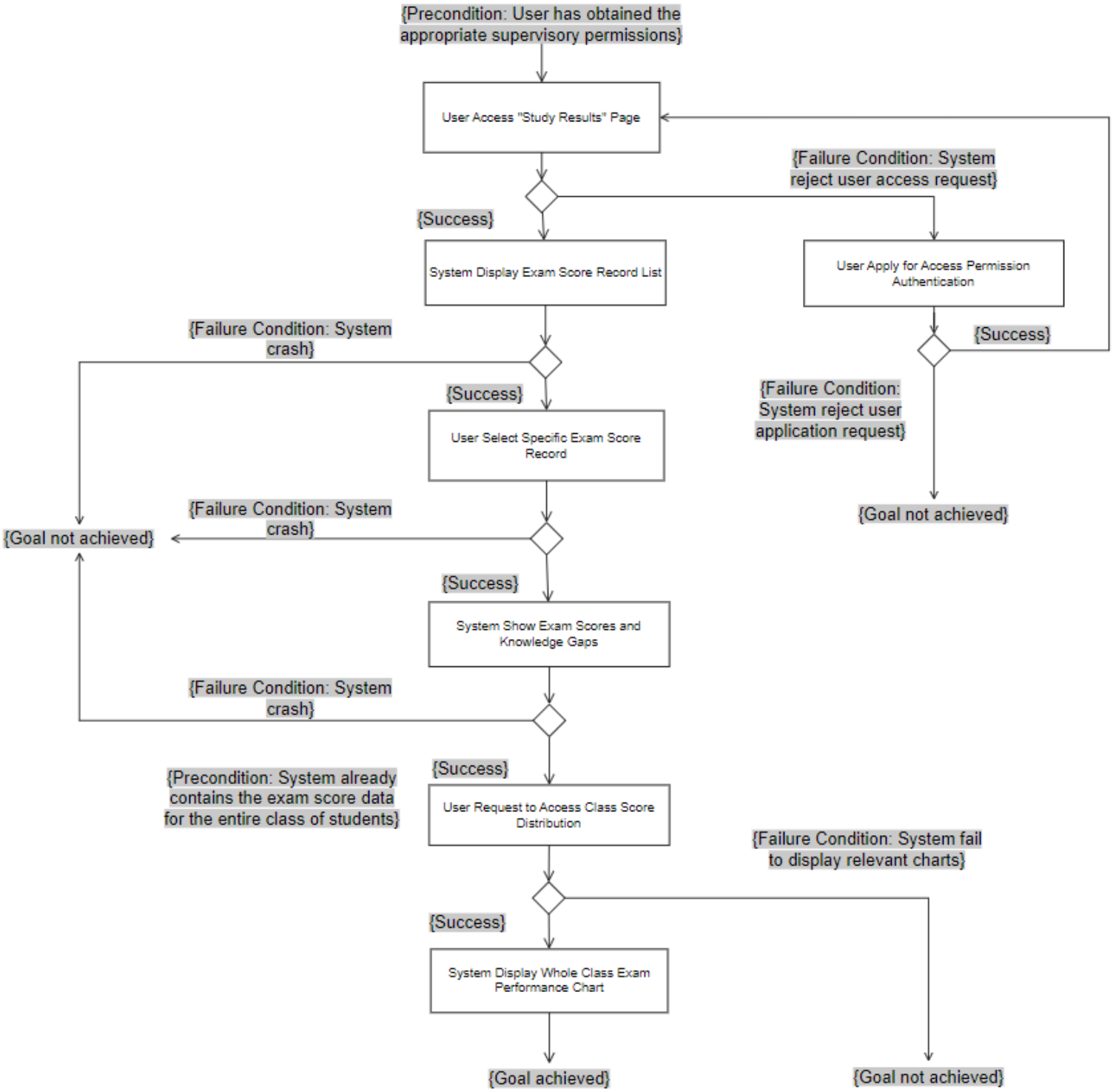
1. **Student Submits a Request**
   The student inputs their learning background, current challenges, and learning goals into the system and clicks "submit."
2. **System Receives and Processes the Request**
   The system analyzes the student's request based on the provided information, including the student's background and goals, to identify suitable study resources.
3. **System Generates Study Resource Recommendation**
   The system then generates a list of study resources tailored to the student's specific needs, such as articles, videos, or online courses.
4. **Resource Delivered to the Student**
   The recommended study resources are displayed to the student, either as a direct link or a downloadable document.
5. **Notification of Success**
   The system notifies the student that the study resource recommendation has been successfully generated and is available for review.

Use Case 5 (Siyu Chen)

Name	**Monitoring Student's Academic Performance**
Goal	supervise student's learning performance
Level	User Goal
Precondition	User has obtained the appropriate supervisory permissions.
Success Condition	User can successfully access and review the student's exam scores, gaining an accurate understanding of the student's exam performance.
Failure Condition	User is unable to access or review the student's exam scores.
Trigger	User clicks and enters the "Study Results" section from the dashboard

1. **Parent Access "Study Results" Page**
   User clicks "Study Results" section on the dashboard and will be redirected to the "Study Results" page.
2. **System Display Exam Score Record List**
   In the "Study Results" page, system displays a series of exam score records and each record has a "View" button behind it.
3. **Parent Select Specific Exam Score Record**
   User clicks the 'View' button for the exam score record they want to view.
4. **System Show Exam Scores and Knowledge Gaps**
   System displays the corresponding student's exam scores and a list of unmastered knowledge points.
5. **—include <Check Class Score Distribution Use Case>**

The Success and Failure diagram is shown in next page for explaining more details.


{Precondition: User has obtained the appropriate supervisory permissions}
User Access "Study Results" Page
{Failure Condition: System reject user access request}
{Success}
System Display Exam Score Record List
User Apply for Access Permission Authentication
{Success}
{Failure Condition: System crash}
{Success}
{Failure Condition: System reject user application request}
User Select Specific Exam Score Record
{Goal not achieved}
{Failure Condition: System crash}
{Goal not achieved}
{Success}
System Show Exam Scores and Knowledge Gaps
{Failure Condition: System crash}
{Precondition: System already contains the exam score data for the entire class of students}
{Success}
User Request to Access Class Score Distribution
{Failure Condition: System fail to display relevant charts}
{Success}
System Display Whole Class Exam Performance Chart
{Goal achieved}
{Goal not achieved}

Use Case 6 (Siyu Chen)

Name	**Check Class Score Distribution**
Goal	Understand the student's academic performance within the entire class
Level	Subfunction
Precondition	The system already contains the exam score data for the entire class of students
Success Condition	User can successfully view the whole class score distribution chart and the whole class unmastered knowledge points distribution chart.
Failure Condition	The system fails to display charts.
Trigger	User clicks "View Class Score Distribution" button

1. **User Requests to Access Class Score Distribution**
   User clicks "View Class Score Distribution" button below the student exam scores
2. **System Display Whole Class Exam Performance Chart**
   System displays the whole class score distribution chart and the whole class unmastered knowledge points distribution chart.

Use Case 7 (Zhehang Xu)

Name	**Consult Online**
Goal	facilitate effective communication with teachers through this channel
Level	User Goal
Precondition	User has successfully logged in
Success Condition	The student/parent successfully expressed their requests, and the teacher provided an appropriate response or feedback
Failure Condition	The relevant parties did not receive the expected information in a timely manner
Trigger	User clicks and enters the "Consult Online" section from the dashboard

1. **User Access "Consult Online" Page**
   User clicks "Consult Online" section from the dashboard
2. **System Display a list of available online Consultation Chat Rooms**
   Each online Consultation Chat Rooms represents communication with either teacher-with-students or teacher-with-parent.
3. **User Selects an online Consultation Chat Room**
   User double-clicks the online Consultation Chat Room they want to enter
4. **System Opens the online Consultation Chat Room**
   System displays the online chat room and its history messages
5. **User Type Message**
   User types their message to the input box of the online chat room

6. **User Send Message**
   User clicks the send button next to the input box

Use Case 8 (Wenxing Zhou)

Name	**AI Generate Quizzes**
Goal	generate quizzes questions by AI for users to practice, print papers or other usage
Level	User Goal
Precondition	User has successfully logged in
Success Condition	The user can describe the quizzes requirements for chatbot, and chatbot can generate suitable questions and arrange them into a pdf document for users to download.
Failure Condition	• User cannot send quizzes requirements to chatbot. • The AI cannot understand the requirements and generate suitable questions properly. • The document cannot be downloaded successfully.
Trigger:	User clicks the "Quizzes Generator" button on the dashboard.

1. **Teachers submit requirements**
   The teachers input the topic and difficulty requirements and knowledge should be involved into a chat box and send it to the system
2. **System receives the request**
   System receives the requirements, and delivers it to the AI generator
3. **AI generate quizzes questions and arrange them into a PDF document**
   AI receives the requirements of the system and generates corresponding questions and arranges the questions into a PDF document, and delivers the document to the system
4. **Questions delivered to students or teacher**
   System delivers the PDF document to teachers, and provides a download link for them

Use Case 9 (Zhaoyu Wang)

Name	**Personalize Teaching Plan**
Goal	Helping users to find gaps and customizing personalized teaching plans.
Level	User Goal
Precondition	Users has logged in the system successfully and obtained corresponding permission.
Success Condition	Users input the situation of the teaching process and system and recognize the requirements correctly and generate a personalized teaching plan for the teachers.

Failure Condition	• Users can not upload the requirement document successfully. • System rejects the access request of users. • System cannot understand the requirement right and deliver it to the AI model successfully. • The learning plan cannot be delivered to the users successfully.
Trigger	Users click the “Personalized Teaching Plan” button on the dashboard.

1.**Teachers submit Teaching Plan generation requirements**

The teachers input the topic and difficulty requirements and knowledge should be involved into a chat box and send it to the system

2.**System receives the request**

System receives the requirements, and delivers it to the AI generator

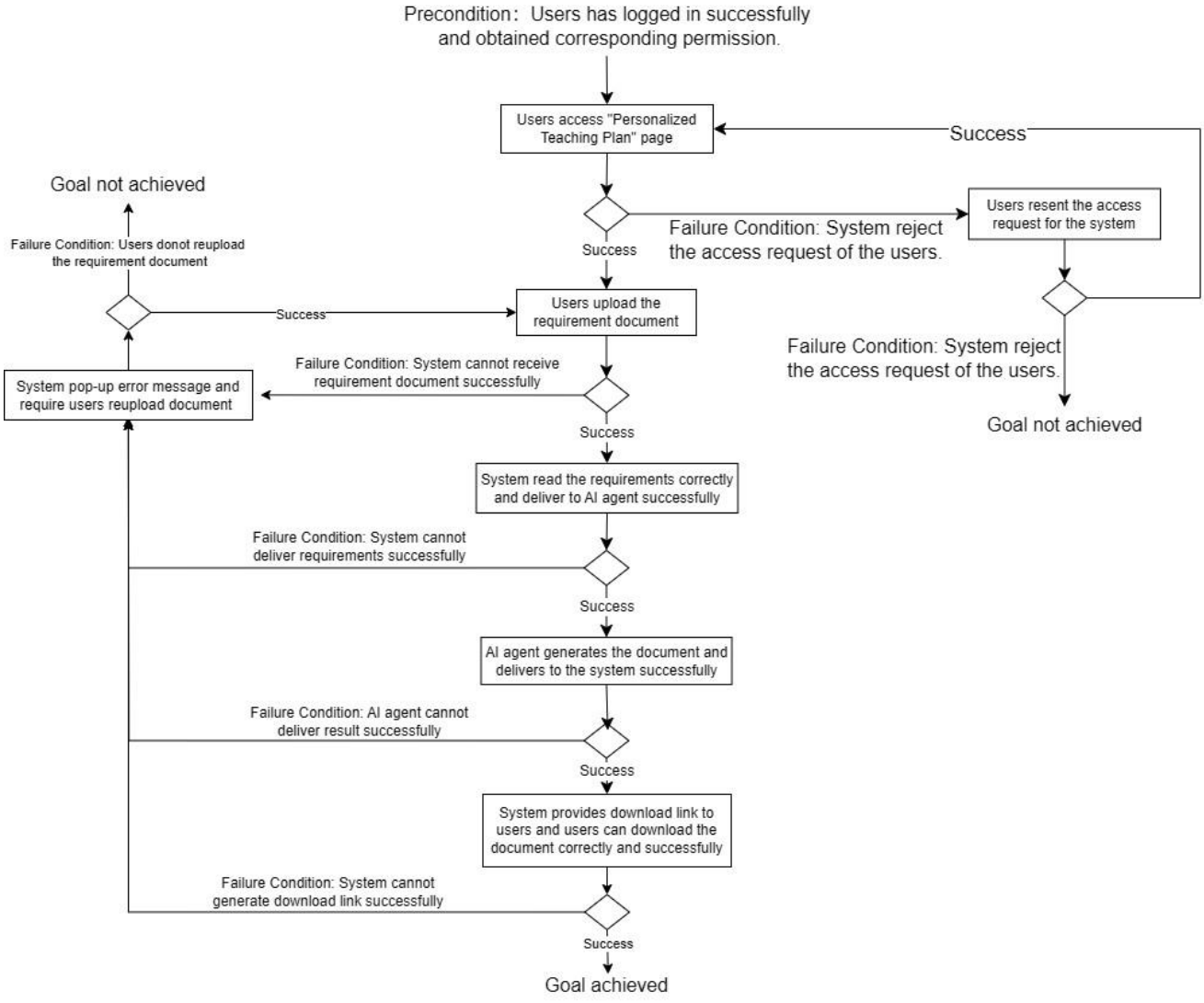
3.**AI generate Personalized Teaching Plans and arrange them into a PDF document**

AI receives the requirements of the system and generates personalized teaching plan and arranges the questions into a PDF document, and delivers the document to the system

4.**Questions delivered to teachers**

System delivers the PDF document to teachers, and provides a download link for them

The Success and Failure diagram is shown below for explaining more details.



Use Case 10 (Sirui Li)

Name	**Browse Question Forum**
Goal	Providing a platform for students to browse the question blog list and comments on the forum page.
Level	User Goal
Precondition	Users have logged in successfully
Success Condition	Users can browse the question blogs and comments on the forum page correctly and successfully.
Failure Condition	• Users cannot view the question blogs and comments successfully. • Users see the wrong content of blogs or comments
Trigger	Users click the "Question Forum" button on the dashboard.

Users can browse the question blogs and comments on the forum page.
–Extension point1: Post Question Blog (use case 11)
–Extension point2: Add comment (use case 12)

Use Case 11 (Sirui Li)

Name	**Post Question Blog**
Goal	Users can post question blogs on the forum to ask questions and get answers.
Level	User Goal
Precondition	Users have logged in successfully
Success Condition	Users can open the editor, edit and post blogs successfully, and content of the posted blogs are displayed correctly.
Failure Condition	• Users cannot open the editor • Users cannot edit and post the question blogs successfully. • The blogs cannot be displayed on the forum page successfully. • The content of the posted blogs is wrong.
Trigger	Users click the "Post Question Blog" button on the forum page.

1. **User clicks the "Post Question Blog"**
   User clicks the "Post Question Blog" button and the editor page will be open, the user can edit the content of the question blog.
2. **User clicks the "Submit" button after editing**
   After editing the blog, users can click the "Submit" button to upload the question blog.
3. **System receives the request and data**
   System receives the submit request and the blog data, then display the blog on the forum page.

Use Case 12 (Sirui Li)

Name	**Add Comment**
Goal	Users can add comments to the question blogs which can help users to discuss the problems and get answers.
Level	User Goal
Precondition	Users have logged in successfully
Success Condition	Comments can be added to the question blogs successfully and displayed under the question blog correctly.
Failure Condition	Comments cannot be posted successfully.
Trigger	Users can input the comments in comment form and submit it to the blogs.

1. **User input the comment content**
   The user can input and edit the comment content into the input form.
2. **User clicks the "Submit" button**
   The user clicks the "Submit" button and can send the submit request to the system.
3. **System receives the request and display it on the page**
   System receives the request and display the comments under the blog.

Use Case 13 (Wenxing Zhou)

Name	**AI-Analysis Score**
Goal	AI analysis score of a certain student or score distribution of a group of students.
Level	Subfunction
Precondition	User has logged in successfully and obtained corresponding permission.
Success Condition	Users can choose the scope of students successfully, and the LLM can read the score correctly and generate score distribution diagrams and analysis successfully.
Failure Condition	The scope of students cannot be selected correctly and the diagram or analysis cannot be generated successfully.
Trigger	User clicks the "Score Analysis Generator" button, and selects the scope of the student and let LLM generate the score distribution diagrams and analysis.

1. **Select the scope of students**
   Teachers can decide the score of which students will be analyzed.
2. **System receives the generating request and deliver it to the LLM**
   System receives the request and delivers the request to the LLM and the model delivers the data back to the system after generating successfully.
3. **Distribution diagrams and analysis generated successfully**
   After the system receives the data, the diagrams and analysis will be demonstrated on the page.

Models and Specifications

Architecture Analysis and Design (All Members)

In the design of the Intelligent Learning Platform, our group adopt the Kruchten's 4+1 View Model to ensure that the system architecture is described from multiple perspectives, addressing the needs of various users. This model provides a comprehensive view of both the functional and non-functional requirements of the platform.

Use Case View

The use case demonstrates the system's functionality by modeling how it operates in real scenarios. It focuses on typical user interactions to showcase the system's ability to handle various requests and fulfill the real-time needs of its users in our intelligent learning platform. The use case view confirms that the platform performs as expected, ensuring it meets the practical requirements of diverse users. This view's detail has been shown in use case specification part.

Logical View

From the logical view, the system's core functional modules are need to be defined and organized to ensure that they work together effectively to meet the platform's functional requirements and deliver the results. Our platform includes essential functionalities such as personalized learning and teaching plans, AI tutoring, quiz generation, and progress monitoring. These features are abstracted into well-structured components, such as AI agents and user data storage, which handle specific tasks within the platform. This view provides a clear representation of the overall architecture, emphasizing the relationships and interactions between these modules. **Class diagrams** are used to model the system's static structure, showing how different classes are related, while **sequence diagrams** illustrate the dynamic interactions between modules during key operation.

Process View

Form process view, system focuses on the run-time behavior, in particular how concurrency is handled through tasks and processes to manage multiple user requests and real-time interactions. The platform must handle multiple requests at the same time, especially for AI-driven functions such as generating quizzes and personalised planning chatbots. The system allows concurrent processing of task. **Activity diagrams** are used to describe the sequence and concurrency of task execution, ensuring that the system allocates tasks efficiently, maintains high performance for users' operation.

Development View

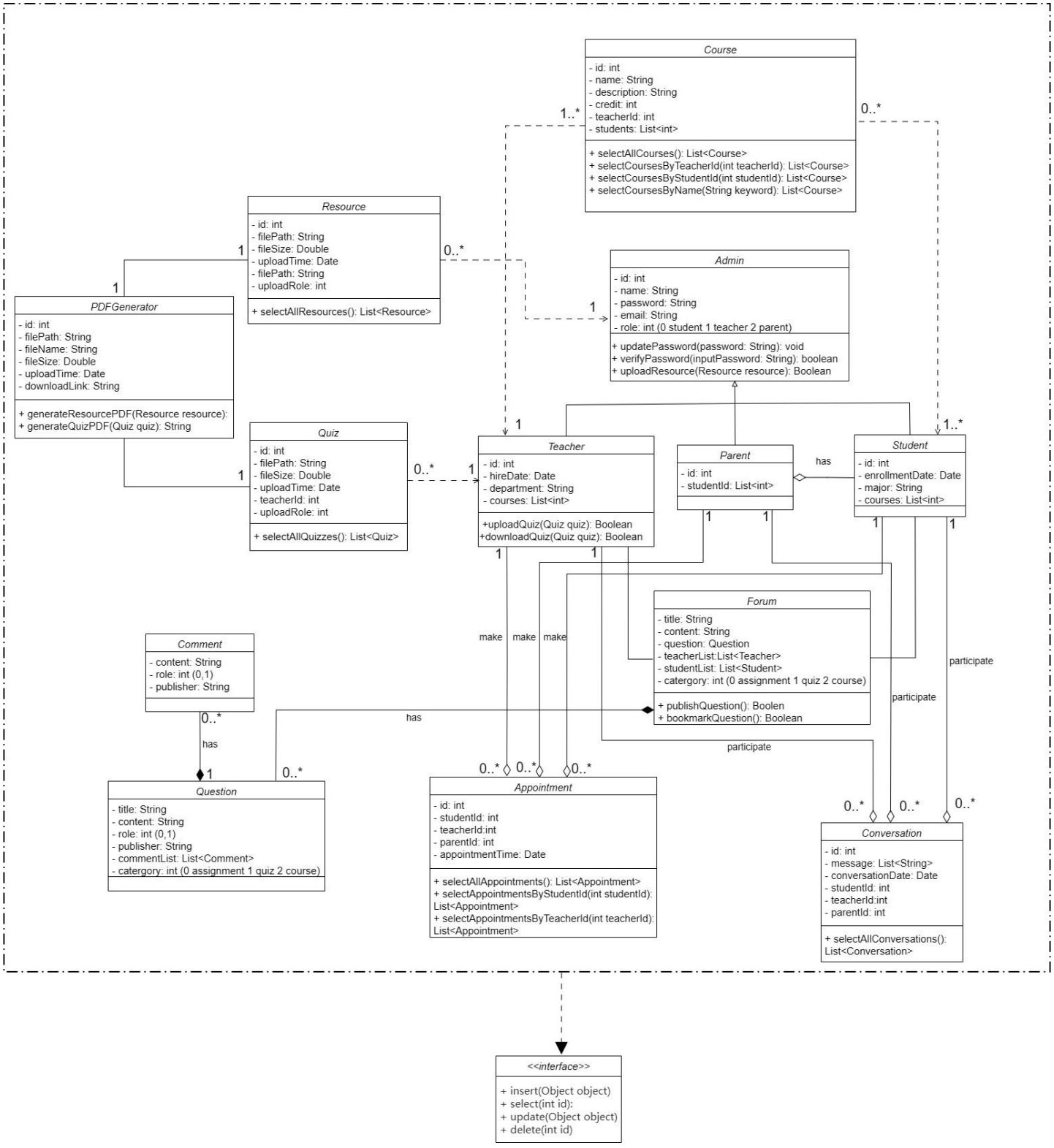
From the development view, system architecture is described from the perspective of the development stage, with a focus on breaking the system into independent modules. These include the backend services, AI model integration, and the database. This design improves the independence of each module, allowing them to be developed, tested, and deployed separately. The **package diagrams** are used to visually represent the organization of these modules, showing how different parts of the system are grouped together and how they interact. This modular development approach facilitates future functionality expansion or updates without affecting the existing system functions.

Physical View

From a physical view, system is structured with components such as AI services, user data storage, and backend services distributed across various servers to optimize performance for specific operational needs. This deployment architecture allows for secure handling of sensitive data, efficient workload distribution, and reliable performance. The **deployment diagram** illustrates how system components like the database server, web server, application server and end user are arranged within the infrastructure. It also highlights how the system to meets the necessary scalability, reliability, and security requirements.

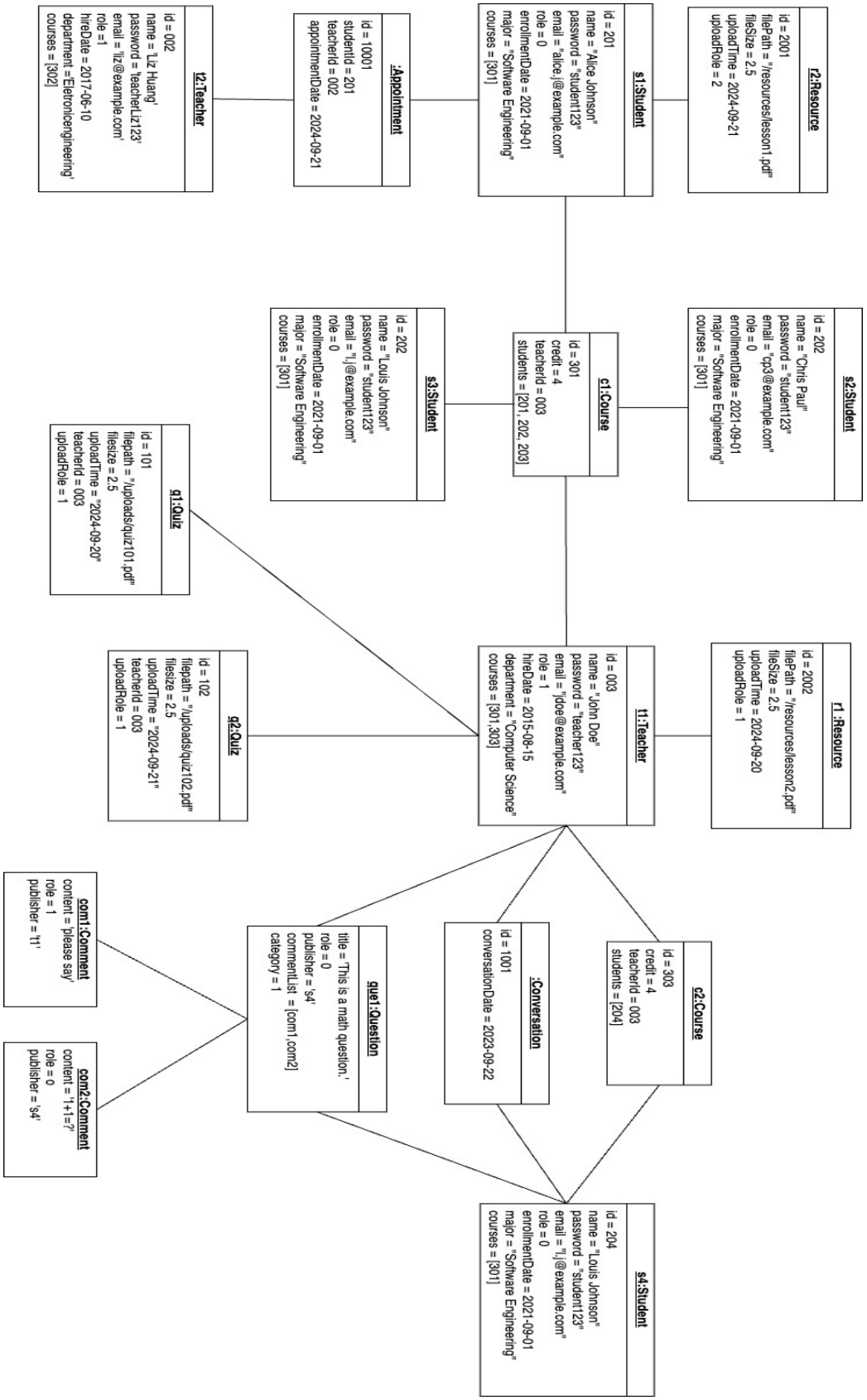
Logic View-Class Diagram (Zhaoyu Wang)

This class diagram shows the main entity classes of the Intelligent Learning Platform and their interrelationships, including Course, Resource, Quiz, PDFGenerator, Admin, Teacher, Parent, Student, Forum, Question, Comment, Appointment and Conversation classes and an insert, delete, select and update interface.

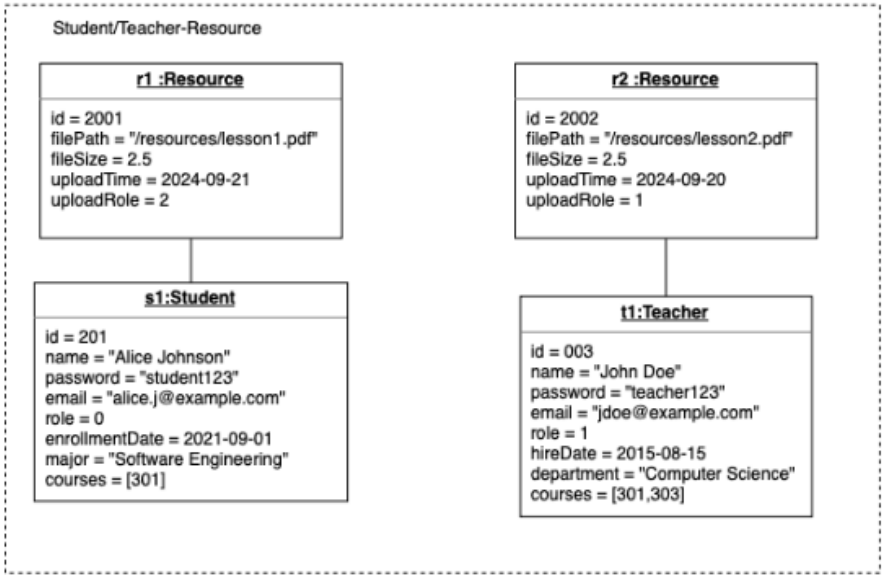


Object Diagram (Wenxing Zhou)

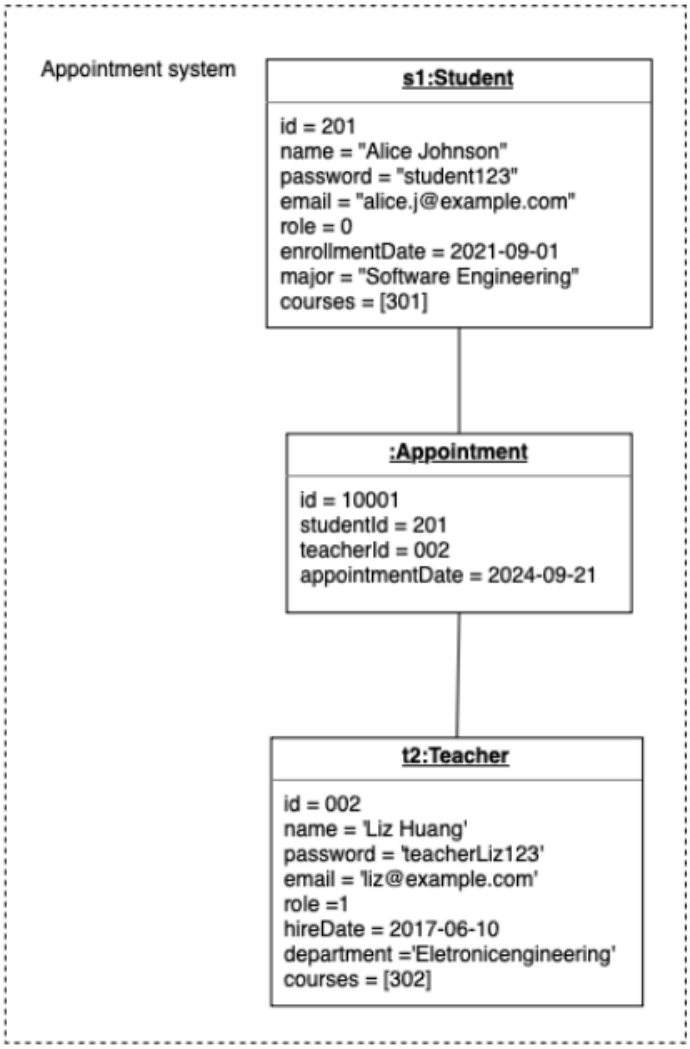
This overall object diagram shows the objects and their interrelationships in the Intelligent Learning Platform. And below it is split into individual object diagrams based on different scenarios.



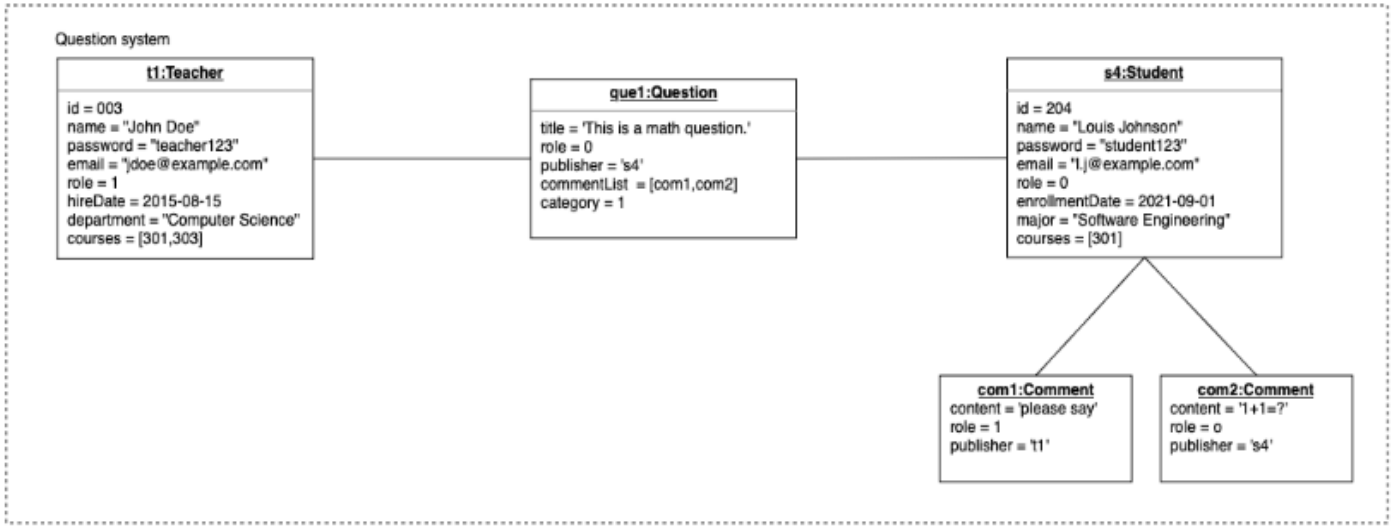
Teacher Student AI Resource



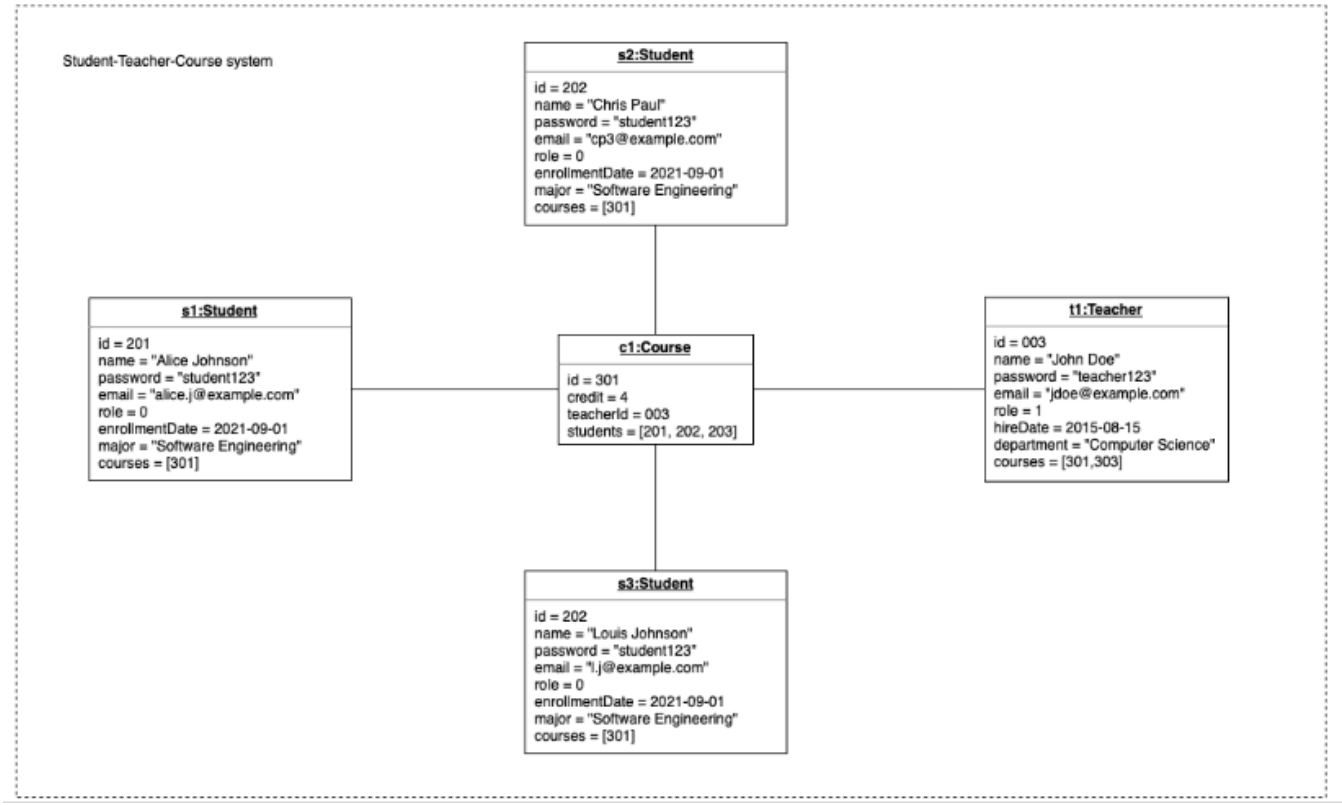
Appointment



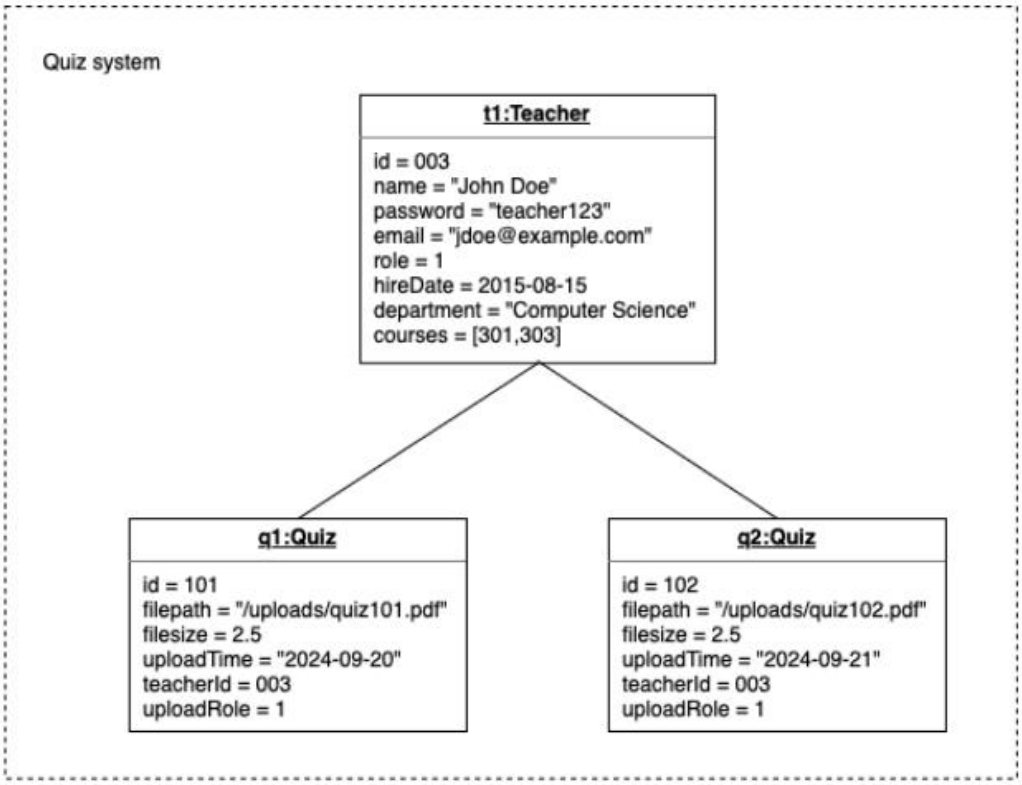
Forum Post Question and Comment



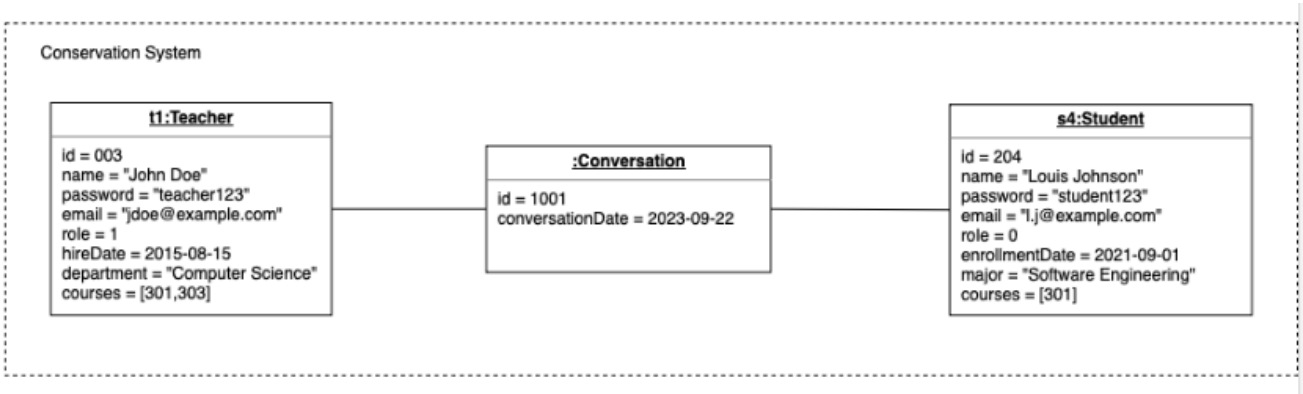
Teacher Student Course Record



Teacher Generating Quiz



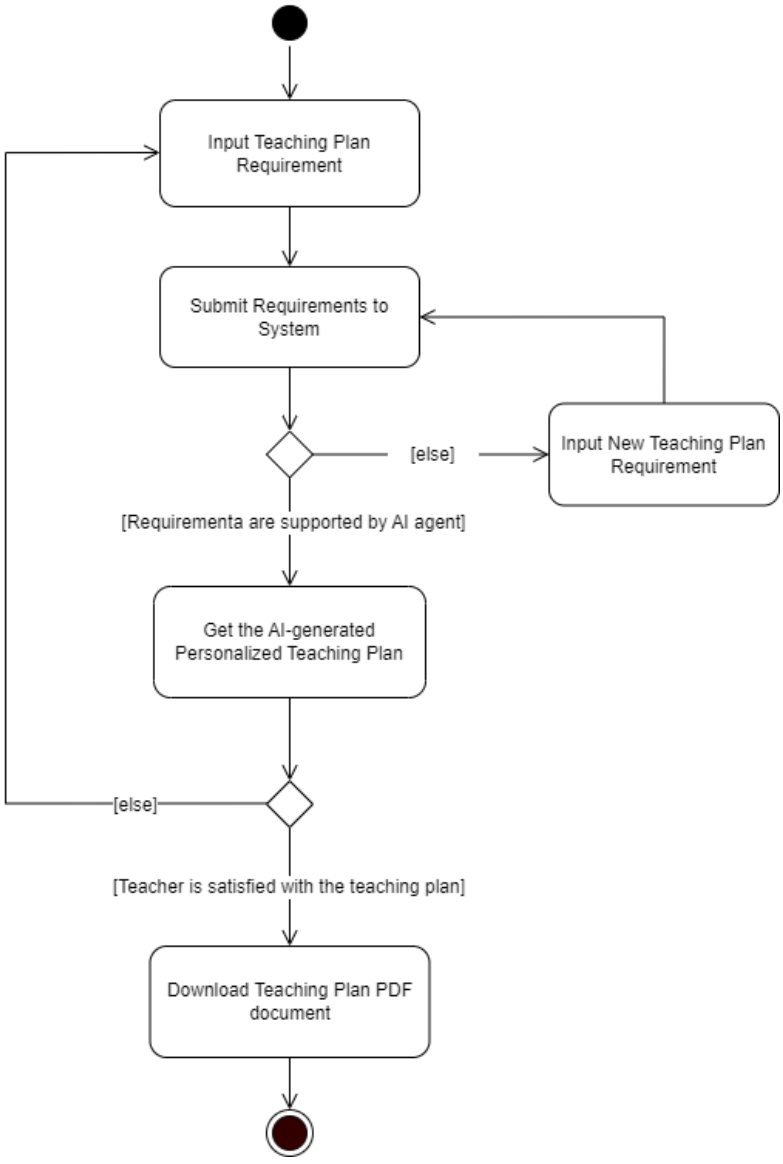
Teacher Student Conversation



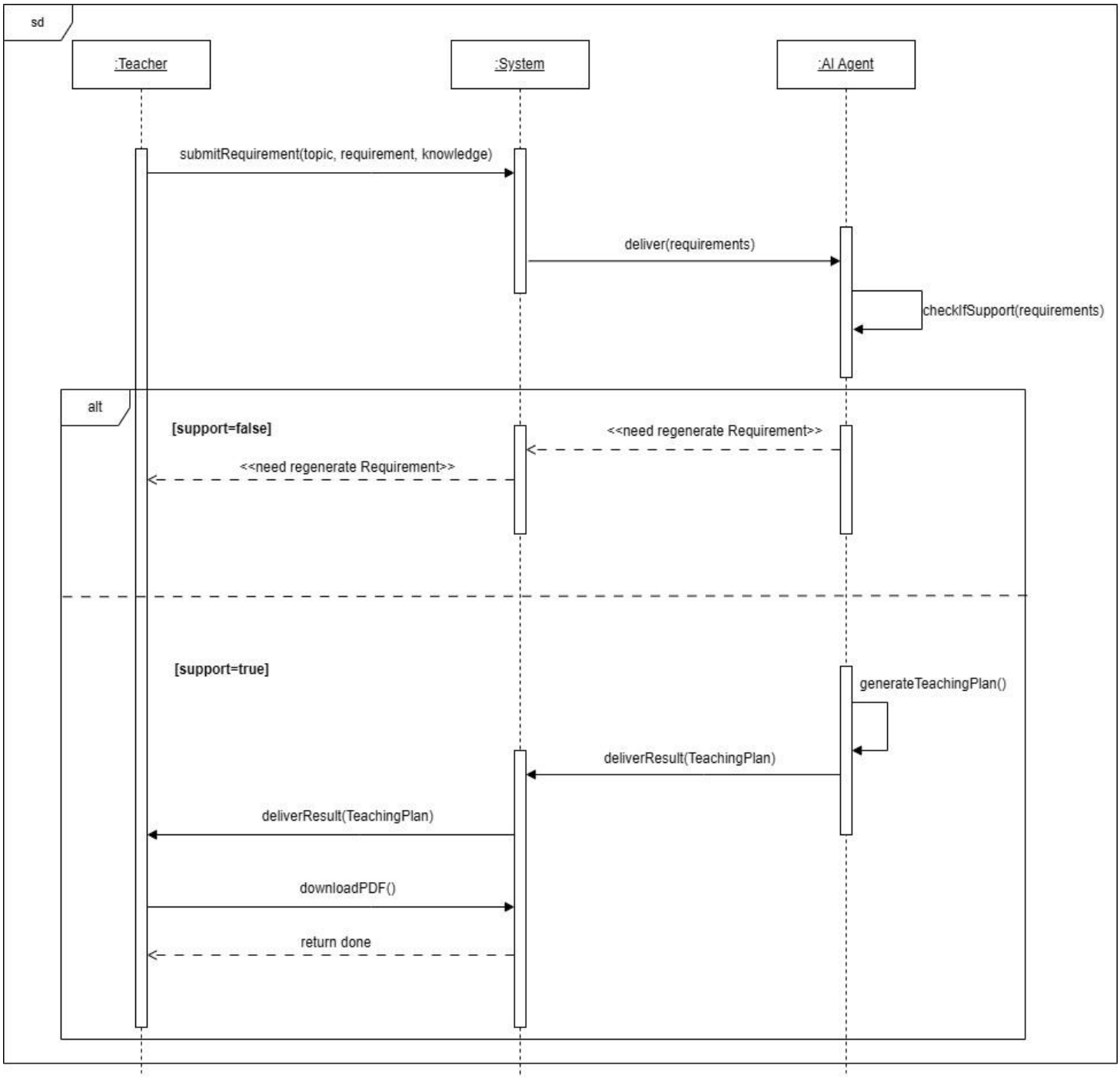
Scenario1 (Process View Activity Diagram, Logic View Sequence Diagram, State Machine Diagram) (Siyu Chen)

This scenario is **Personalized Teaching Plan for Teachers**, teachers enter teaching plan requirements and submit them to the system, which sends the requirements to an AI agent for a supportability check. If the requirement is supported by the AI, the system generates a personalized lesson plan and the teacher can download the corresponding PDF document. If the requirement is not supported, the teacher can re-enter a new requirement. Throughout the process, requirements are passed and results are generated between the system and the AI agent, which ultimately satisfies the teacher's instructional needs. The Activity diagram, Interaction Sequence diagram and State Machine Diagram will be shown below.

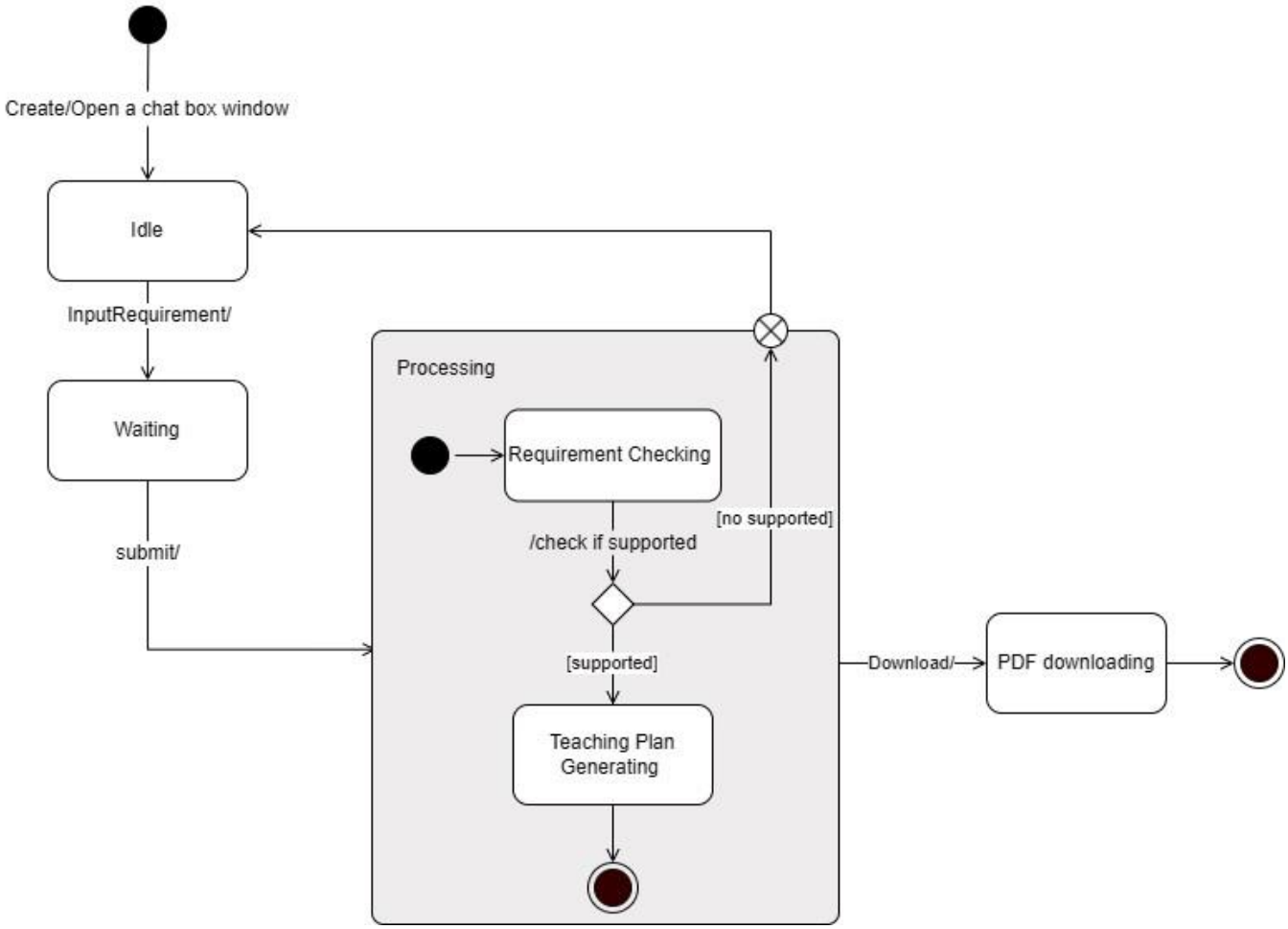
Activity Diagram Personalized Teaching Plan



Sequence Diagram Personalized Teaching Plan



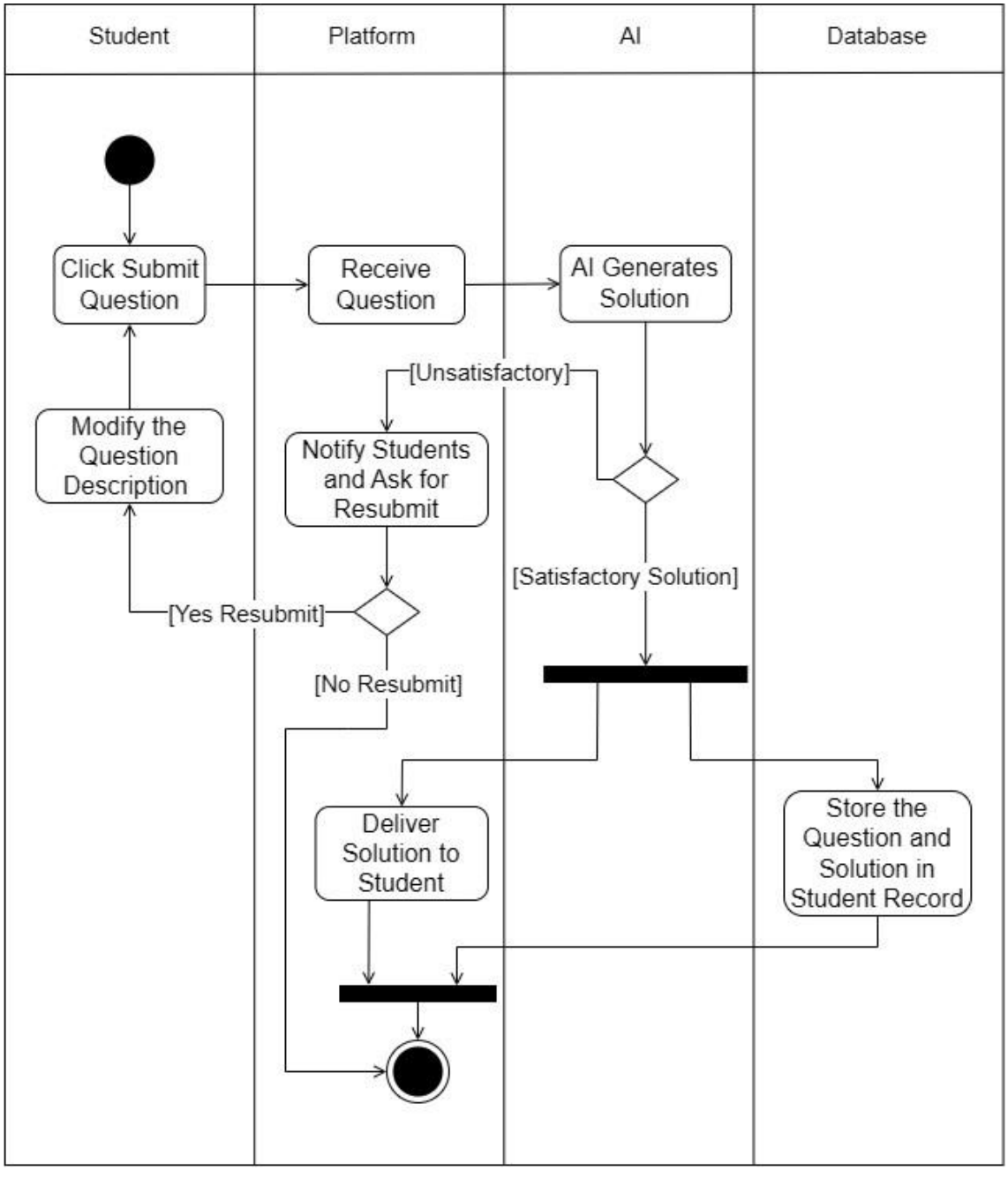
State Machine Diagram Personalized Teaching Plan



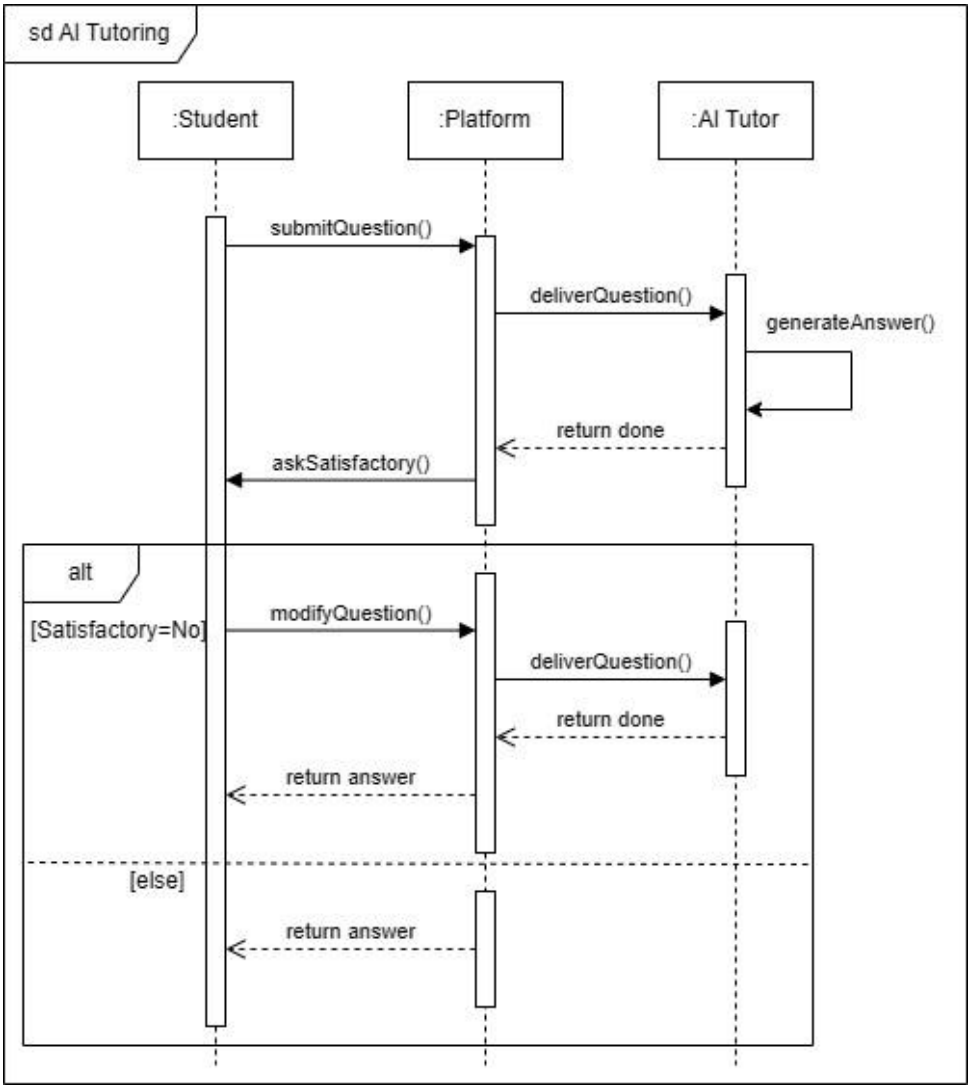
Scenario2 (Process View Activity Diagram, Logic View Sequence Diagram, State Machine Diagram) (Zhehang Xu)

This scenario is **AI Tutoring for Students**, and student first submits a question through the platform, which receives the question and sends it to the AI for processing. If the answer generated by the AI does not meet the student's requirements, the system notifies the student to revise and resubmit the question. If the answer meets the requirements, the system delivers the answer to the student and stores the question and answer in the database at the same time. The Activity diagram, Interaction Sequence diagram and State Machine Diagram will be shown below.

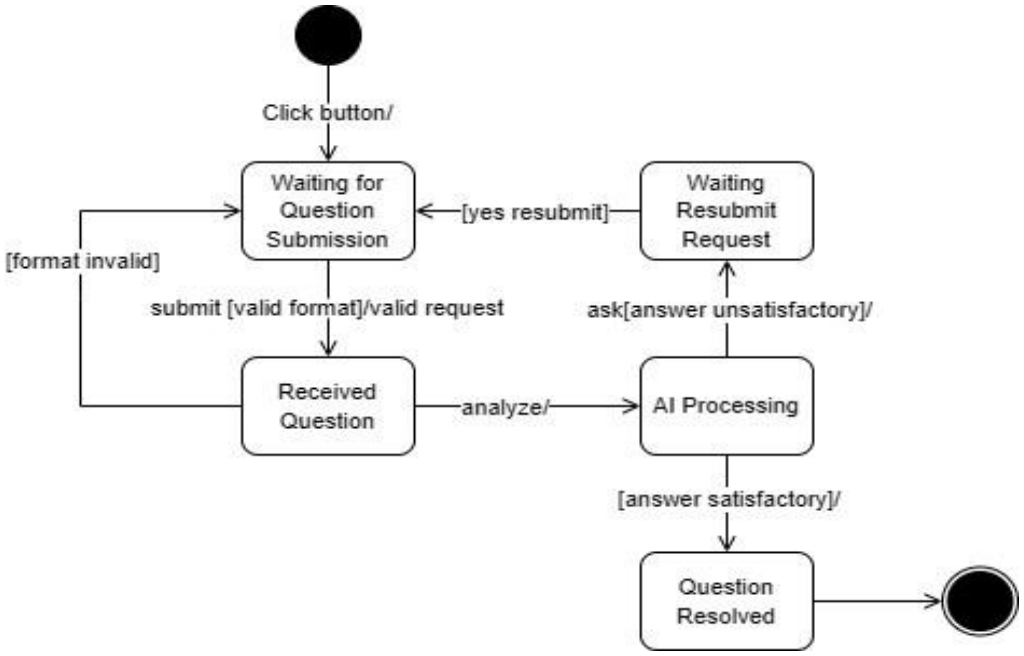
Activity Diagram AI Tutoring



Sequence Diagram AI Tutoring



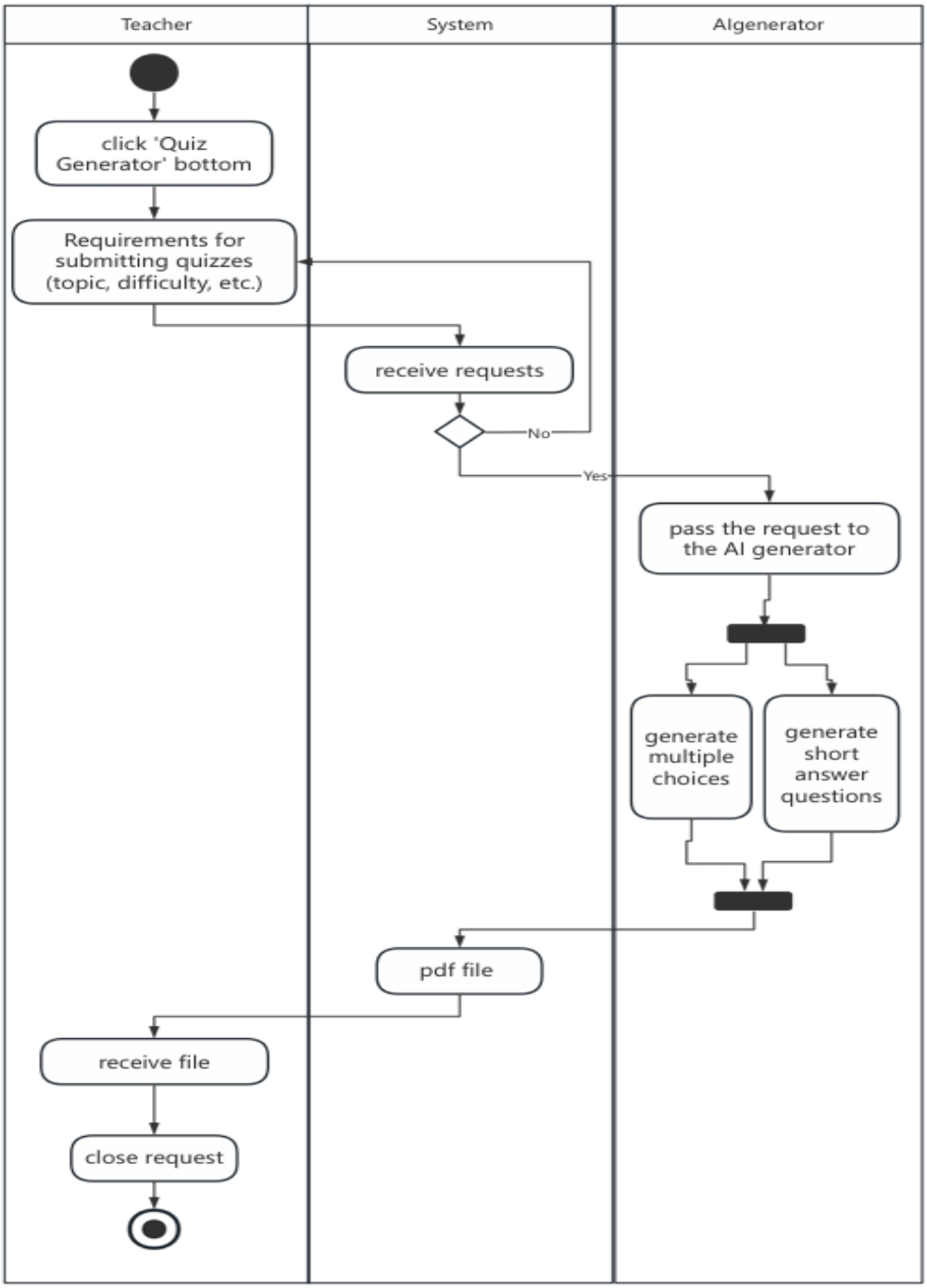
State Machine Diagram AI Tutoring



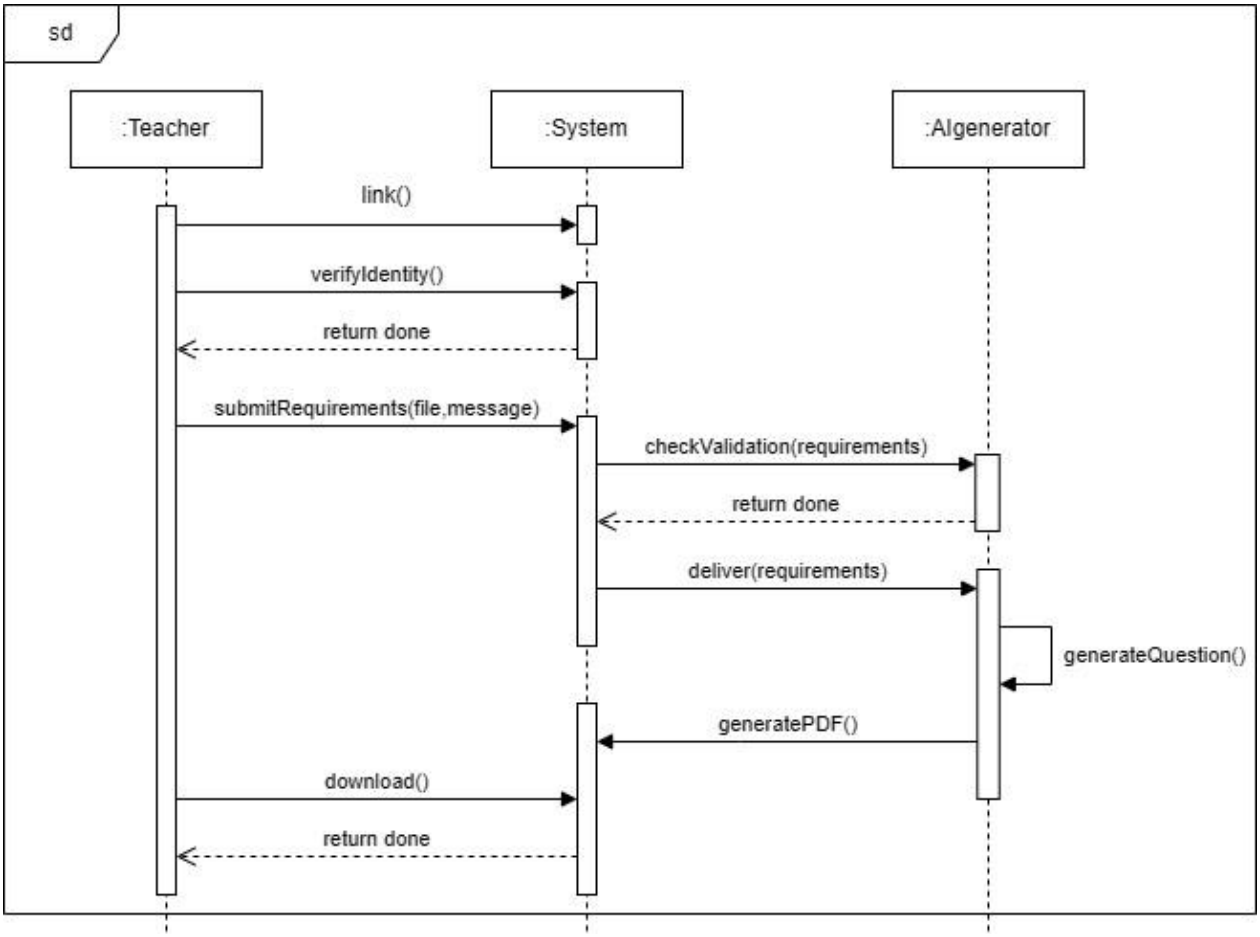
Scenario3 (Process View Activity Diagram, Logic View Sequence Diagram, State Machine Diagram) (Wenxing Zhou)

This scenario is **AI Generating Quiz for Teachers**, teachers click on the "Quiz Generator" button and submits a request for a quiz. The system receives the request and passes it to the AI Generator, which generates multiple-choice and short-answer questions based on the requirements, and returns the quiz to the teacher as a PDF file. The Activity diagram, Interaction Sequence diagram and State Machine Diagram will be shown below.

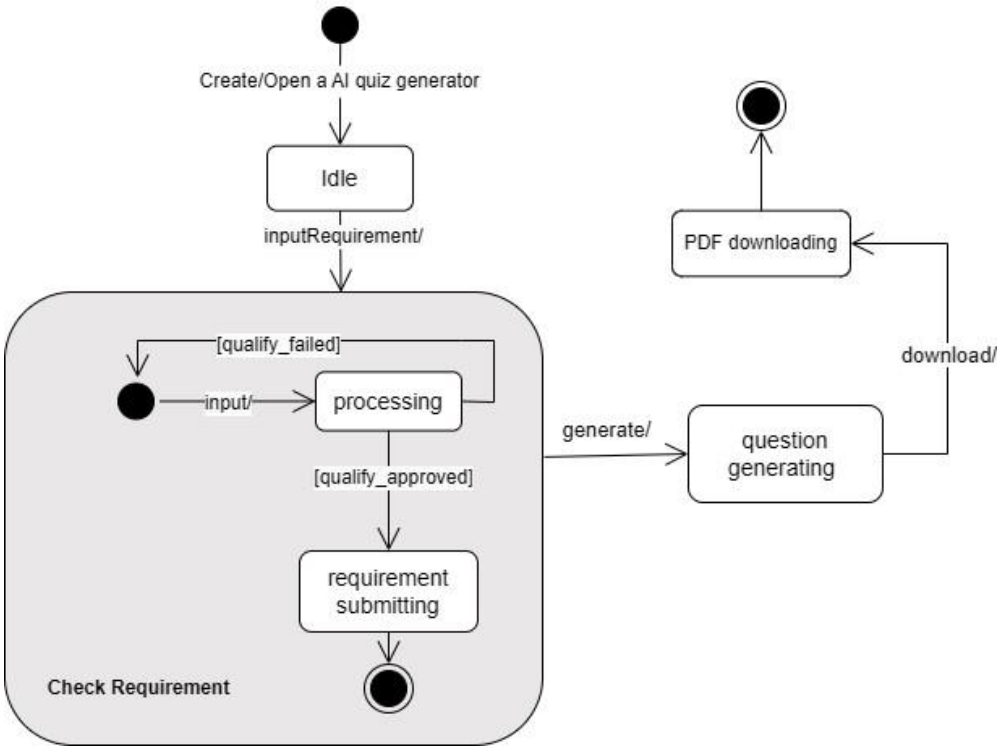
Activity Diagram AI Generating Quiz



Sequence Diagram AI Generating Quiz



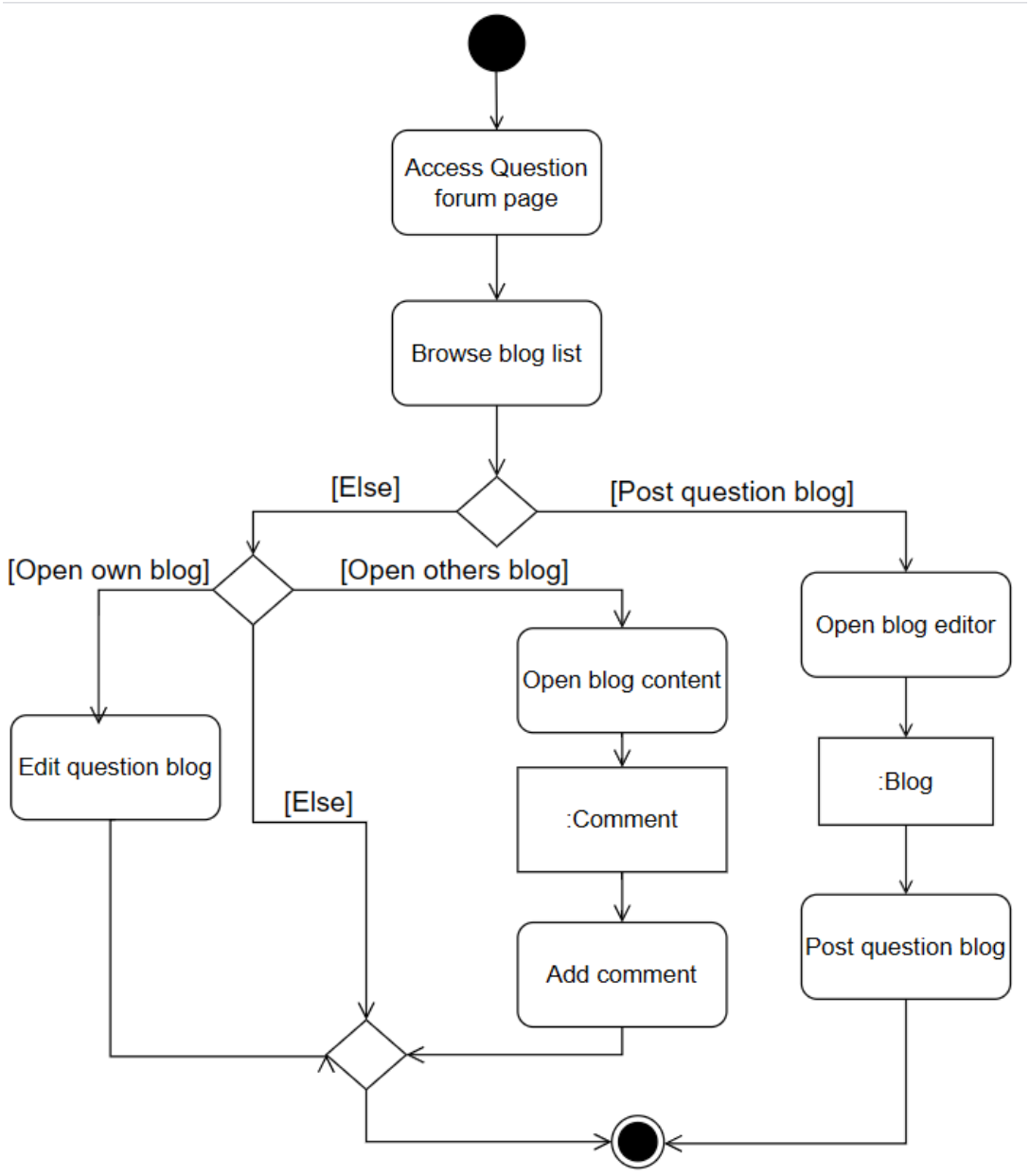
State Machine Diagram AI Generating Quiz



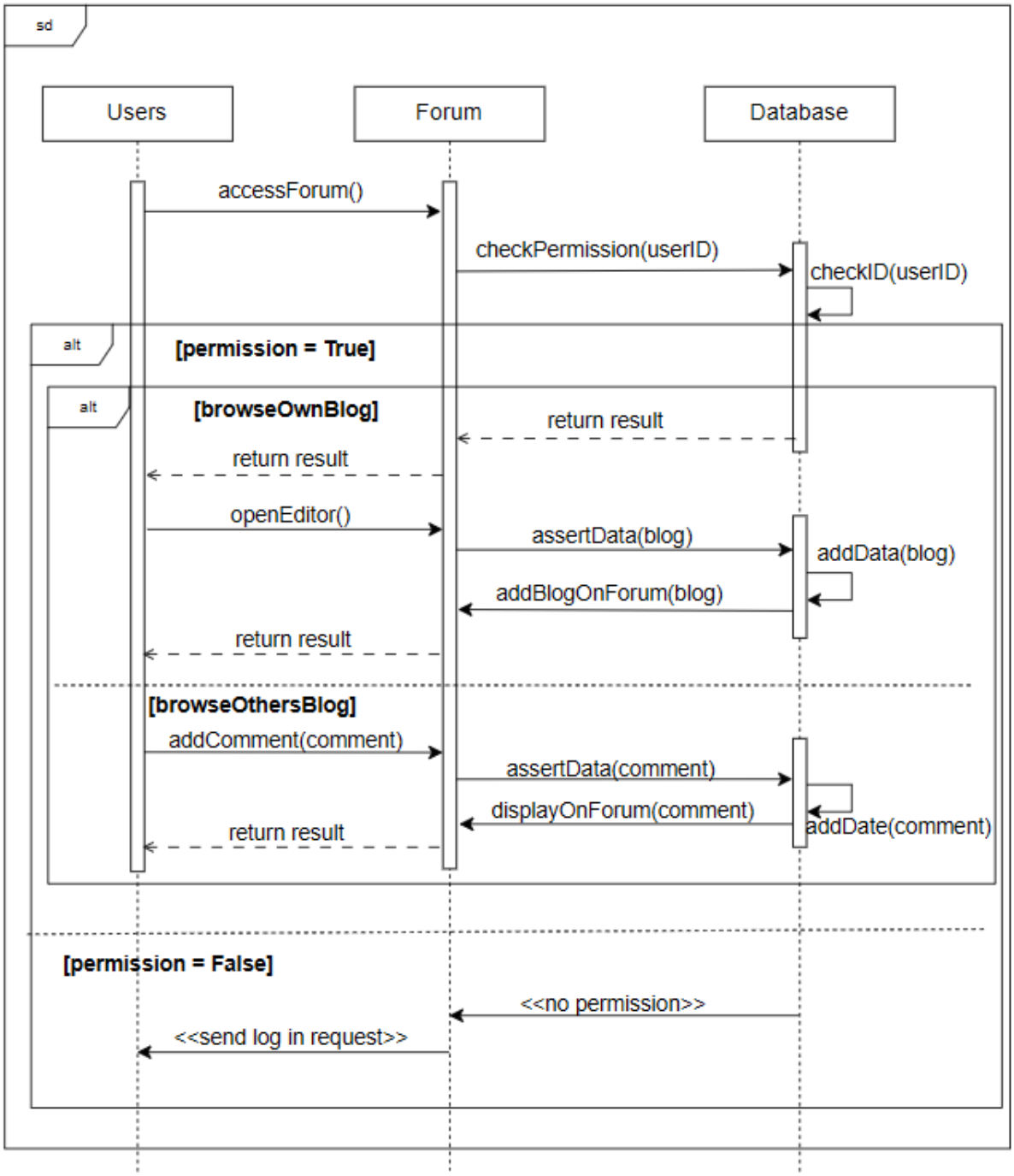
Scenario4 (Process View Activity Diagram, Logic View Sequence Diagram, State Machine Diagram) (Sirui Li)

This scenario is **Forum Operation for Students**, the students visit the forum page and browse the list of blogs, and then have the option of opening their own blog to edit it, or opening someone else's blog to view the content and add a comment. If the student chooses to post a new question blog, they are taken to the blog editor to edit and post the blog. The Activity diagram, Interaction Sequence diagram and State Machine Diagram will be shown below.

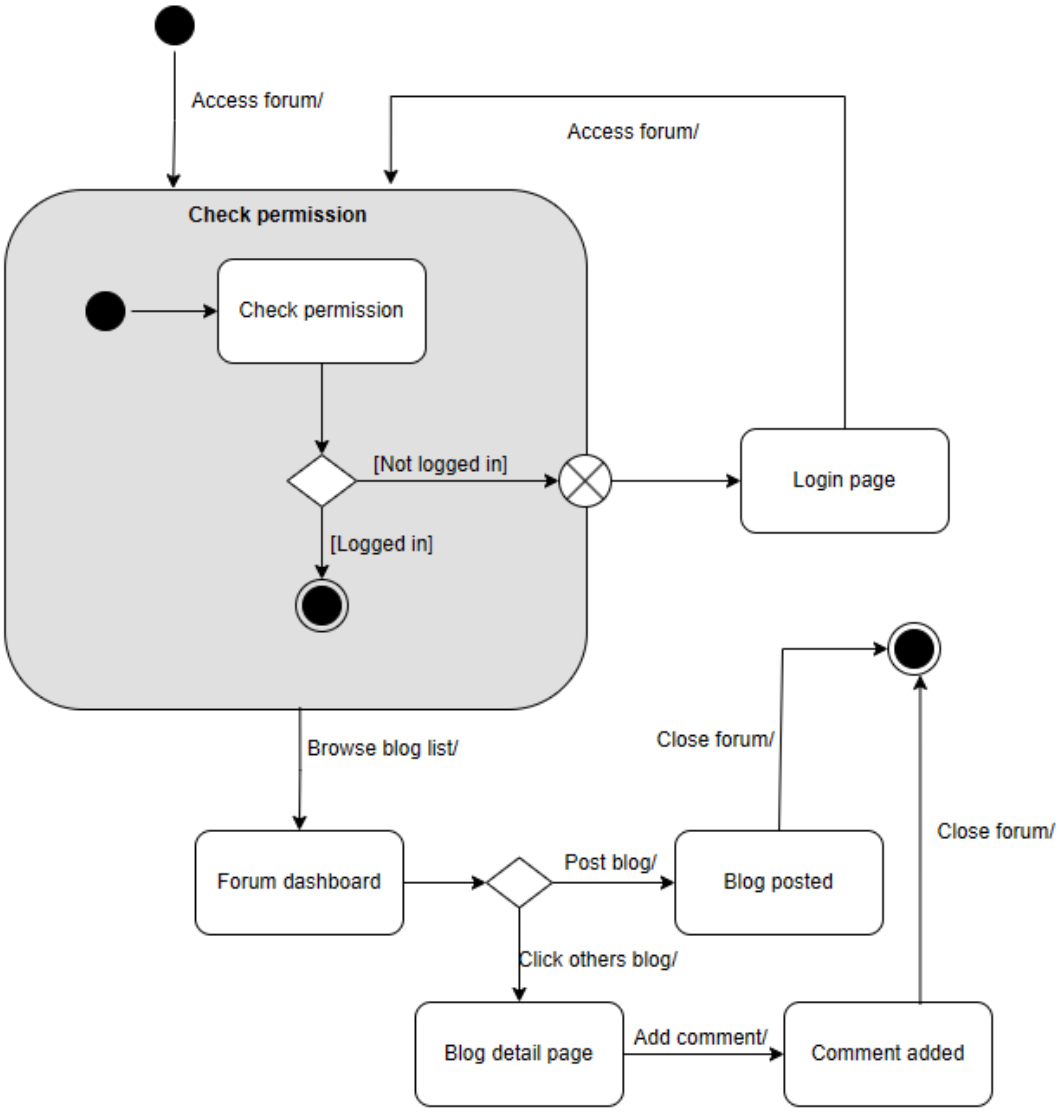
Activity Diagram Forum Operation



Sequence Diagram Forum Operation



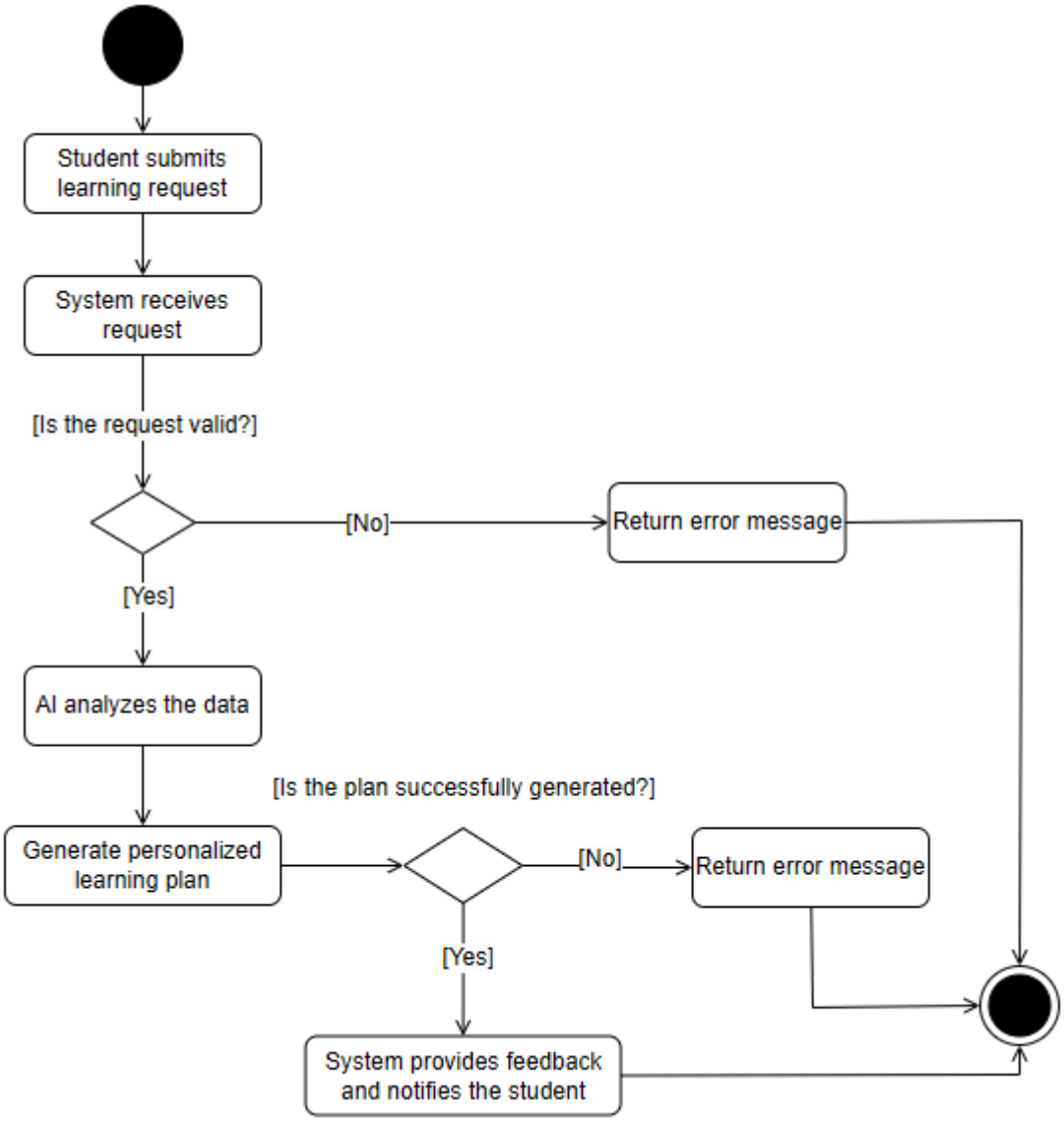
State Machine Diagram Forum Operation



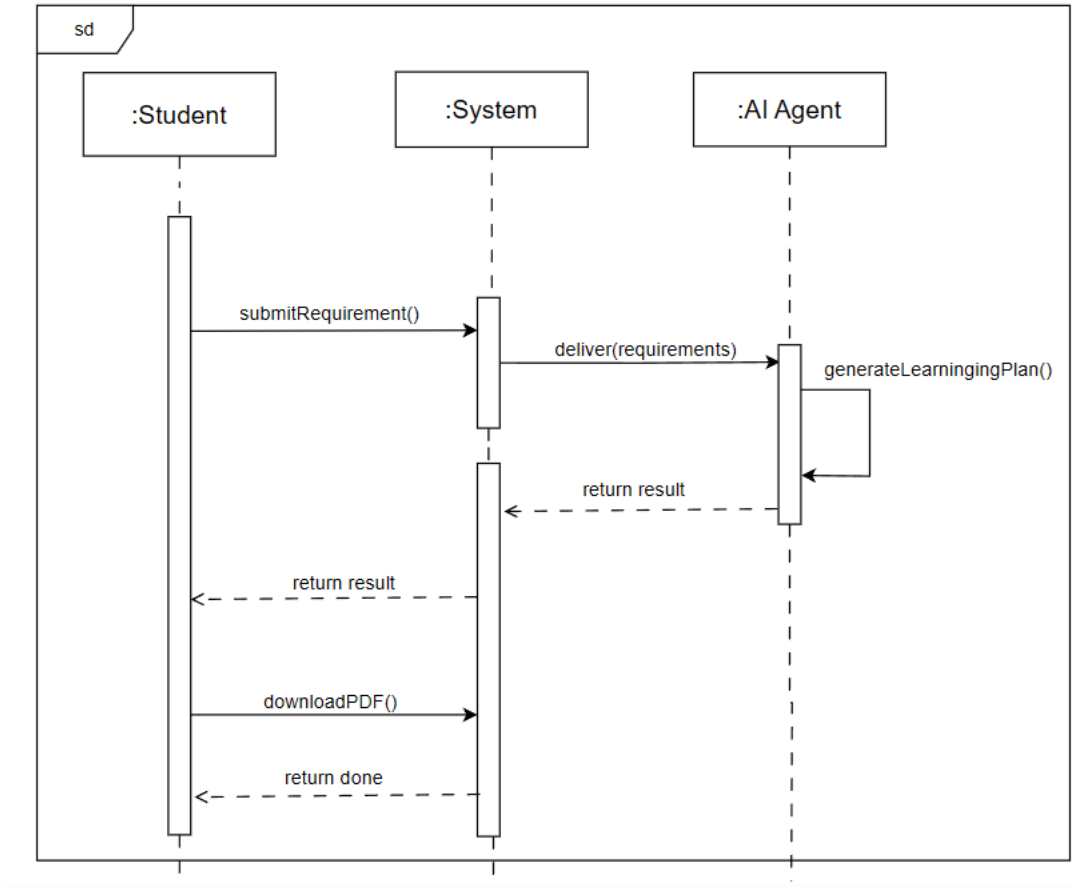
Scenario5 (Process View Activity Diagram, Logic View Sequence Diagram, State Machine Diagram) (Zhaoyu Wang)

This scenario is **Personalized Learning Plan for Students**, the student submits a request to generate a personalized learning plan, and the system receives and verifies the validity of the request. If the request is invalid, the system returns an error message; if valid, the AI analyzes the data and generates a personalized study plan. If the generation is successful, the system gives feedback to the student and provides the option to download the PDF. The Activity diagram, Interaction Sequence diagram and State Machine Diagram will be shown below.

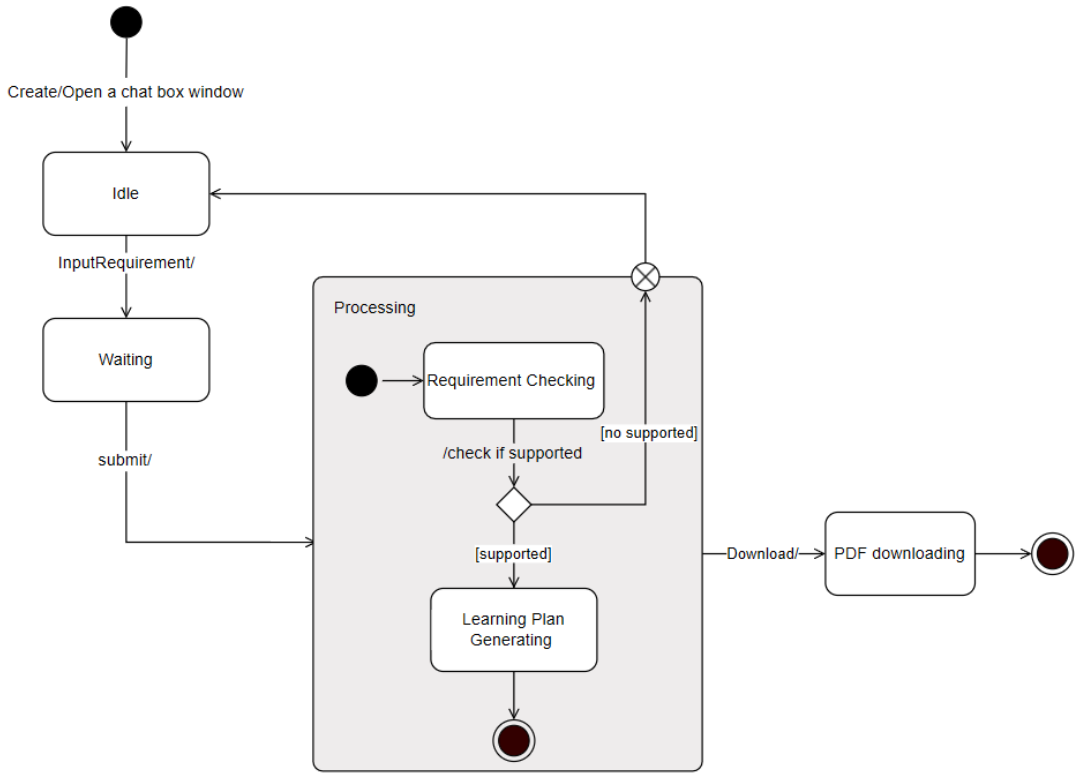
Activity Diagram Personalized Learning Plan



Sequence Diagram Personalized Learning Plan

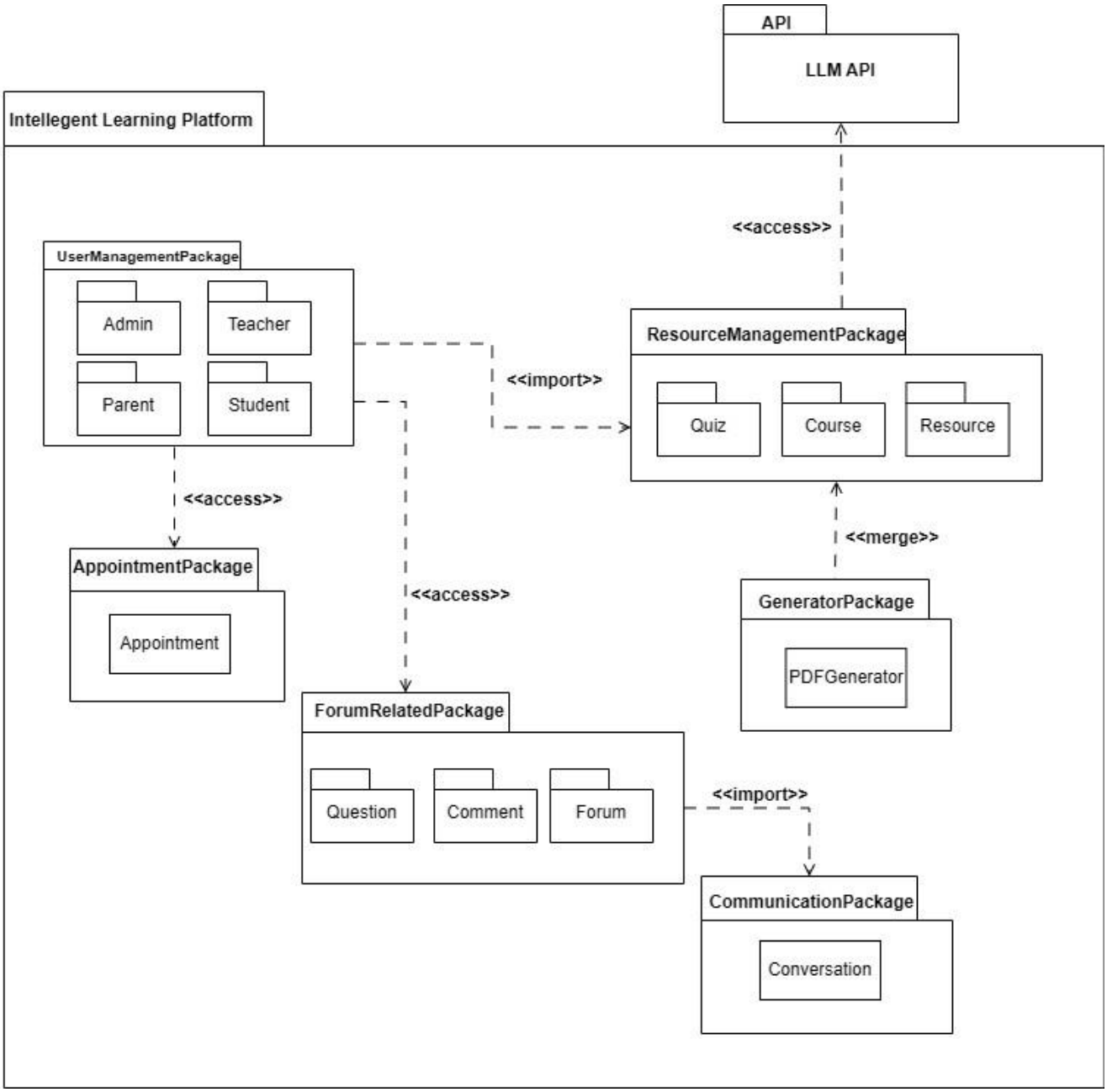


State Machine Diagram Personalized Learning Plan



Development View-Package Diagram (Zhaoyu Wang)

This package diagram shows the main modules of the Intelligent Learning Platform and the relationships between them, with individual packages interacting through relationships such as import and access. The system consists of several functional packages, including UserManagement, Generator, Forum, Appointment, Communication and ResourceManagement, which handles the management of quizzes, courses and resources, and is able to access external LLM APIs for intelligent support.



Physical View-Deployment Diagram (Zhehang Xu)

This deployment diagram shows a SpringBoot based web application architecture where the client interacts with the web server through a browser (BrowserClient) and the web server hosts the SpringBoot application and communicates with the application server through the REST/HTTP protocol. The application server is hosted on Tomcat and runs in the Spring Boot container and is responsible for handling business logic. The application server interacts with MySQL and Oracle databases through the MyBatis framework. In addition, the system integrates calls to external APIs (Open AI 4o) to support AI agent functionality.