**NLP Chatbot Development using Dialogflow**

**Design Document**

**Version 1.0**



**Group Id: F24PROJECT9A4A9**

**Supervisor Name: Abdullah Qamar**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date (dd/mm/yyyy)** | **Version** | **Description** | **Author** |
| 20/02/2024 | 1.0 | Initial Design Document of NLP Chatbot Development using DialogFlow | BC200401263 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

1. [Introduction of Design Document](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#One) Page: 04
2. [Entity Relationship Diagram (ERD)](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#ERD) Page: 06
3. [Sequence Diagrams Page: 07](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#Six)
4. [Architecture Design Diagram](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#Seven) Page: 10
5. [Class Diagram Page: 12](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#class)
6. [Database Design](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#databasedesign) Page: 13
7. [Interface Design](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#interfacedesign) Page: 14
8. [Test Cases](file:///C:\Users\HP%20EliteBook\Desktop\1-DD-Spring%202024_CS619_9445_2.docx#testcases) Page: 15
9. **Introduction of Design Document**

The Design Document for the NLP Chatbot Development using DialogFlow for Training Company provides a detailed architectural blueprint of the system. It outlines the structural and functional aspects of the chatbot, ensuring a smooth transition from requirements analysis to implementation.

This document is crucial in software development as it serves as a guide for developers, designers, and testers, ensuring that the system meets business objectives and technical requirements. The document includes the following key sections:

1. **Entity Relationship Diagram (ERD):**

* Defines the database structure, showing relationships between various entities such as students, courses, schedules, and chat interactions.

1. **Sequence Diagrams:**

* Illustrates the interactions between different system components and actors (students, admin, and chatbot) during various processes like inquiries, course registration, and resource distribution.

1. **Architecture Design Diagram:**

* Provides an overview of the system’s high-level architecture, including backend components (DialogFlow, database, and APIs) and frontend components.

1. **Class Diagram:**

* Defines the object-oriented structure of the system, showcasing the classes, their attributes, methods, and relationships.

1. **Database Design:**

* A detailed schema of tables, attributes, relationships, primary keys, and foreign keys to store chatbot data, user queries, responses, and system configurations.

1. **Interface Design:**

* Visual representation of chatbot interfaces, including student and admin interaction screens, showcasing user-friendly navigation and design elements.

1. **Test Cases:**

* Provides various test cases to validate the chatbot’s functionalities, ensuring that it correctly handles user queries, delivers accurate responses, and efficiently manages course registrations and other features.

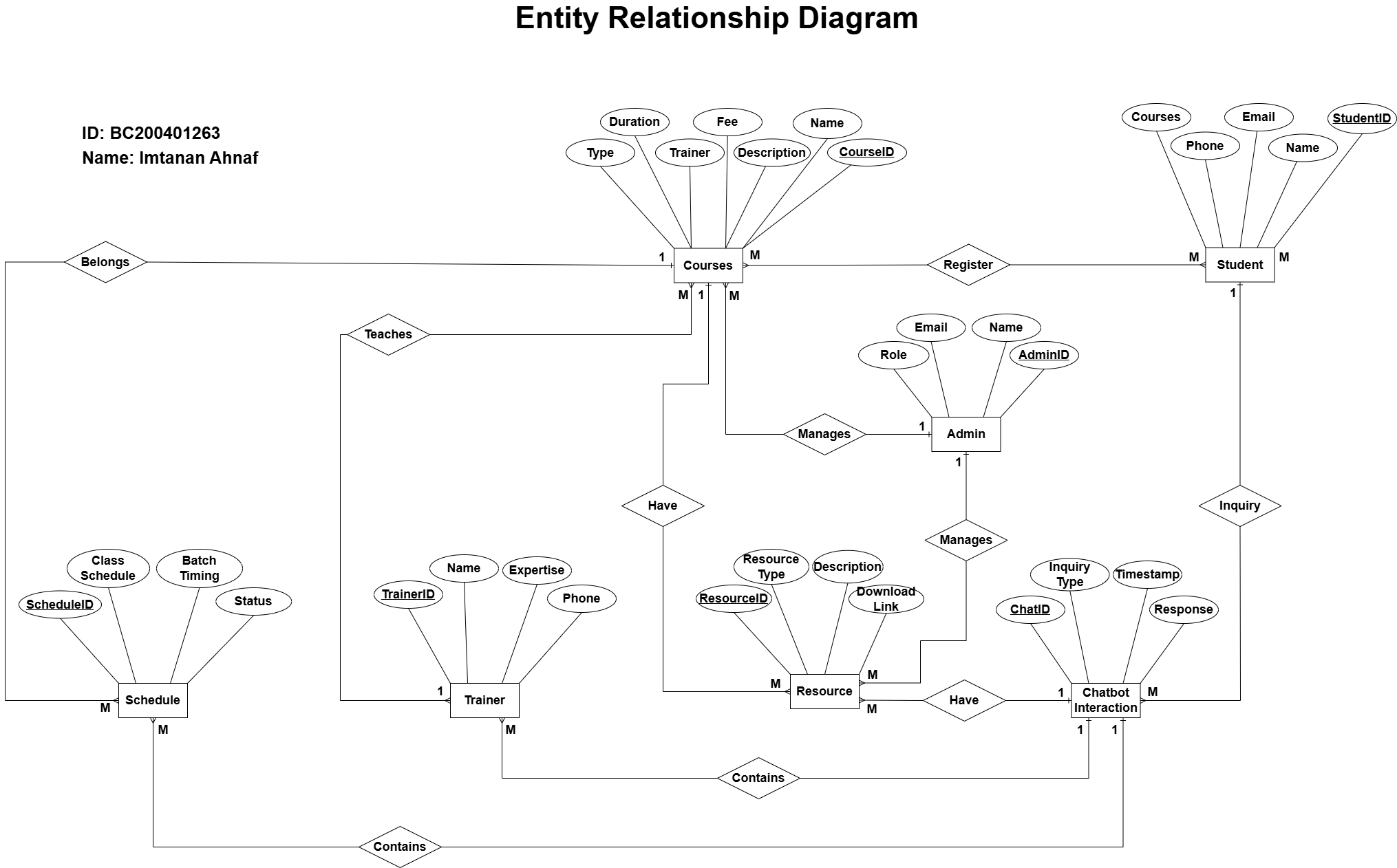
**Purpose and Benefits of the Design Phase:**

The design phase plays a critical role in software development as it bridges the gap between the Software Requirements Specification (SRS) and implementation. Key benefits include:

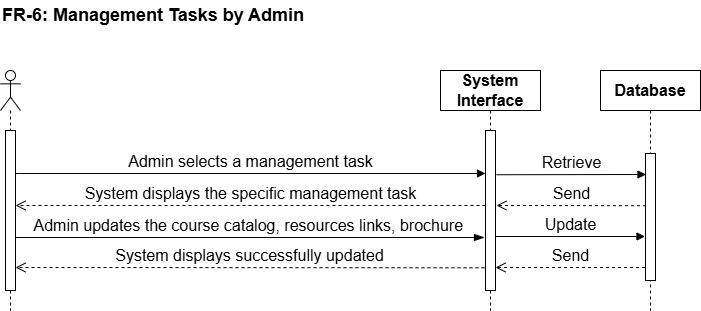
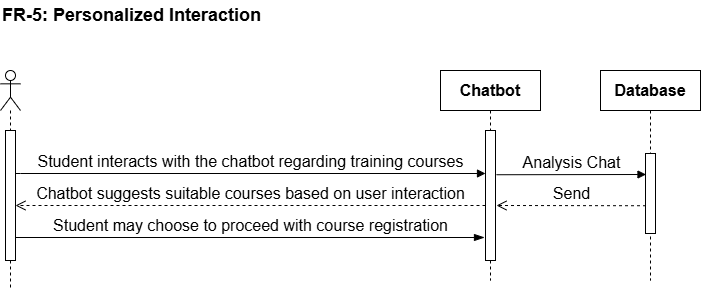
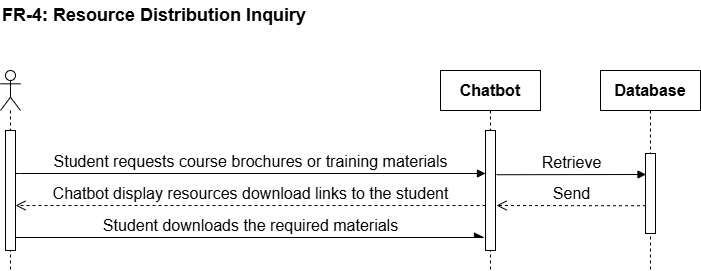
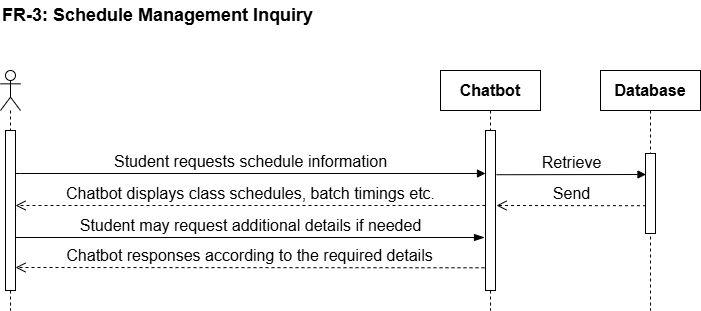
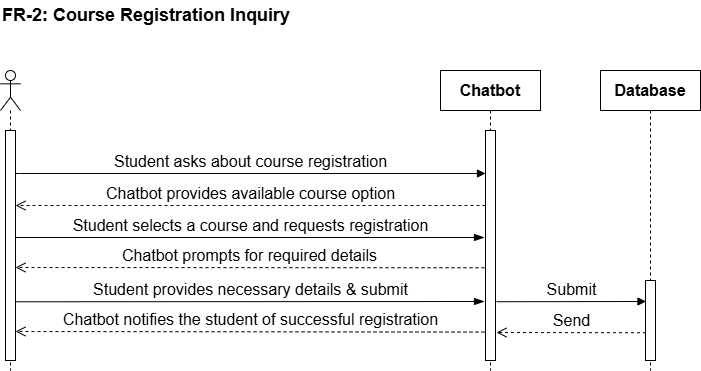
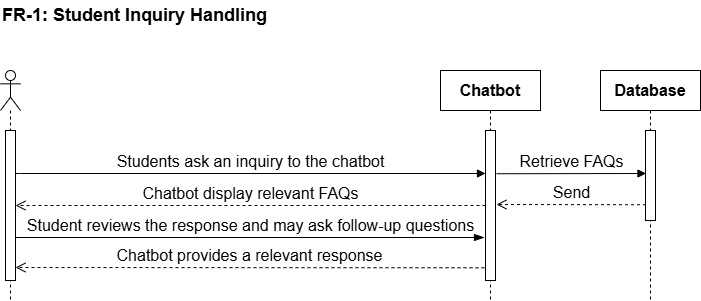
* **Clarity & Direction:** Establishes a clear development roadmap, ensuring that all stakeholders understand the system’s design and structure.
* **Error Prevention:** Helps identify potential issues early in the development cycle, reducing costs associated with rework.
* **Efficiency & Scalability:** Ensures an optimized and scalable architecture that supports future enhancements and integrations.
* **Improved Maintainability:** A well-documented design simplifies troubleshooting, debugging, and future modifications.
* **Seamless Collaboration:** Provides a reference document for developers, designers, and testers, ensuring consistency throughout the project.

This document serves as a foundation for the development team, ensuring that the chatbot system is built efficiently, meets business requirements, and delivers a seamless experience to both students and administrators.

1. **Entity Relationship Diagram (ERD)**



1. **Sequence Diagrams**



1. **Architecture Design Diagram**

NLP Chatbot Development using DialogFlow for Training Company, the project will be designed using a 3-Tier Architecture, which includes the Presentation Layer, Business Logic Layer, and Data Layer.

1. **Presentation Layer (Client-Side)**

* **Components:** Web Interface, Chatbot UI (Website, Messenger, etc.)
* **Technology:** HTML, CSS, JavaScript, Bootstrap (for Web), DialogFlow API integration.
* **Functionality:**
  + Provides a user-friendly interface for students and admin.
  + Receives user input (queries) and forwards them to the application layer.
  + Displays chatbot responses and other dynamic content.

1. **Application Layer (Business Logic)**

* **Components:** NLP Processing, DialogFlow, Backend Services (PHP/Python)
* **Technology:** DialogFlow (NLP Engine), PHP/Python (Backend), Flask/Django (for APIs)
* **Functionality:**
  + Processes user queries using DialogFlow's Natural Language Processing (NLP).
  + Determines intent and extracts relevant information from the query.
  + Calls backend services to fetch course details, schedules, FAQs, and registration forms.
  + Sends an appropriate response back to the Presentation Layer.

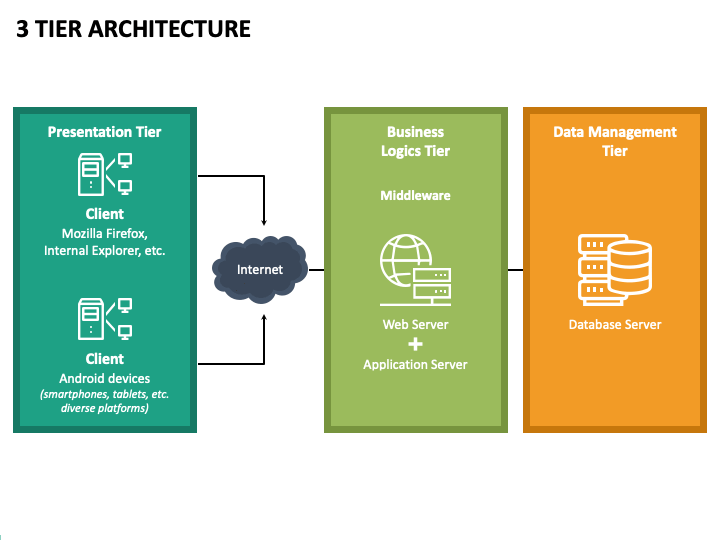
1. **Data Layer (Database and Storage)**

* **Components:** MySQL Database, Cloud Storage (for documents & resources)
* **Technology:** MySQL (Database), Firebase (for chatbot data storage), Cloud Storage (Google Drive/AWS)
* **Functionality:**
  + Stores chatbot training data (intents, responses, FAQs).
  + Manages student information, course details, schedules, and resource files.
  + Logs chatbot interactions for future improvements.

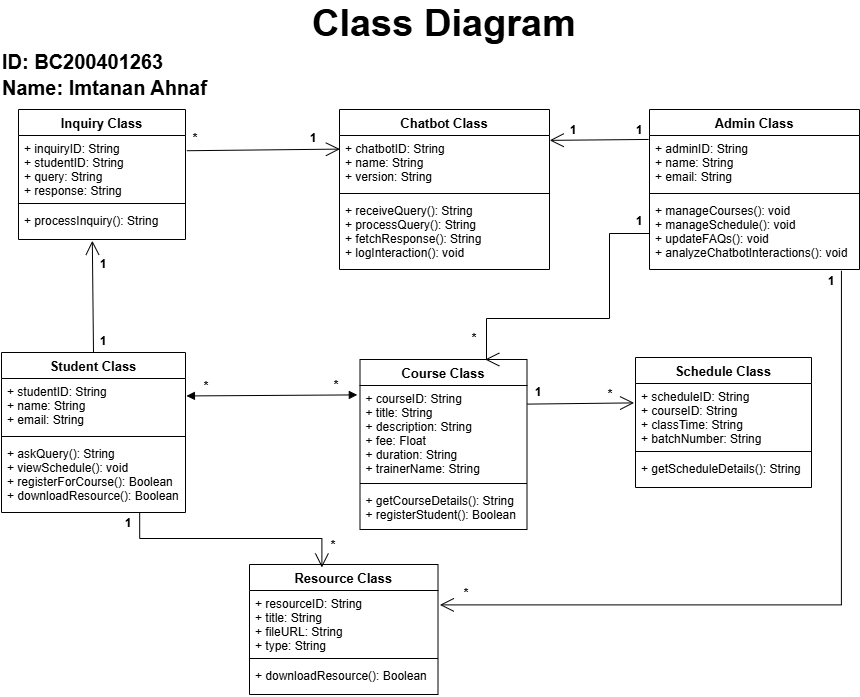
**Reason for choosing 3-Tier Architecture:**

* **Separation of Concerns** – Each layer has its own responsibility.
* **Scalability** – The system can easily be extended by upgrading individual layers.
* **Security** – The database is not directly exposed to the user.
* **Flexibility** – The chatbot can integrate with multiple platforms (Web, Messenger etc.).

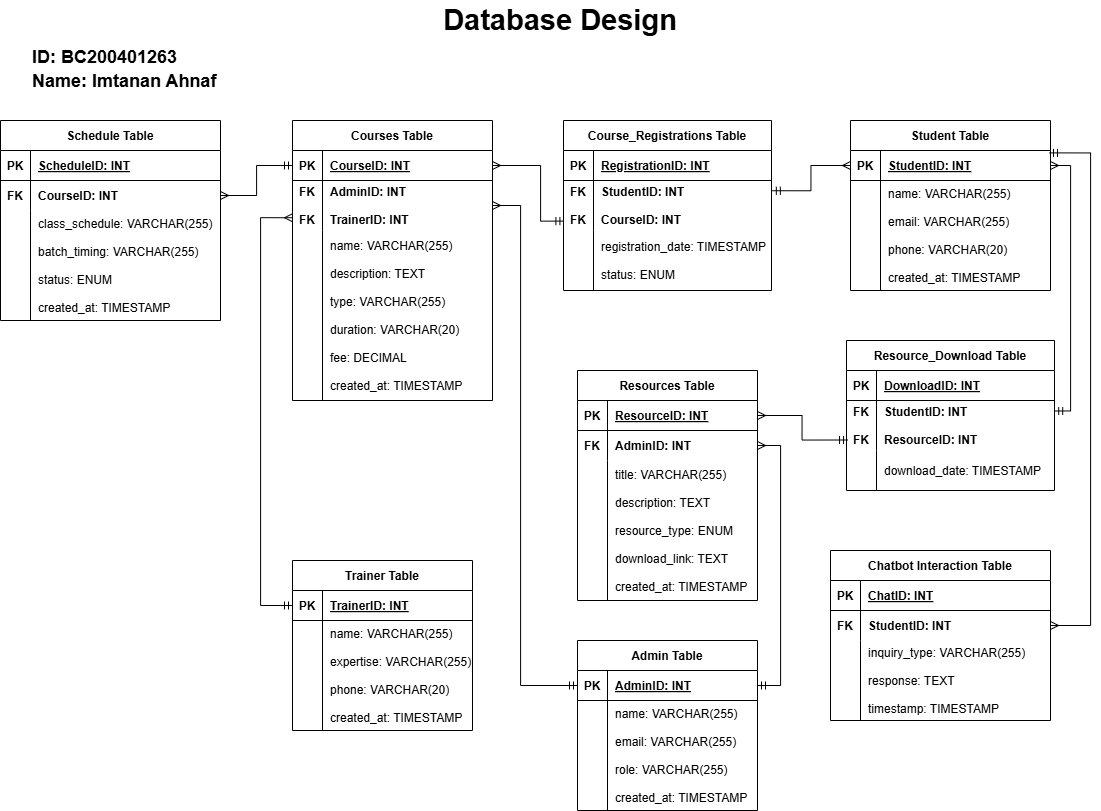
**3-Teir Architecture Diagram:**



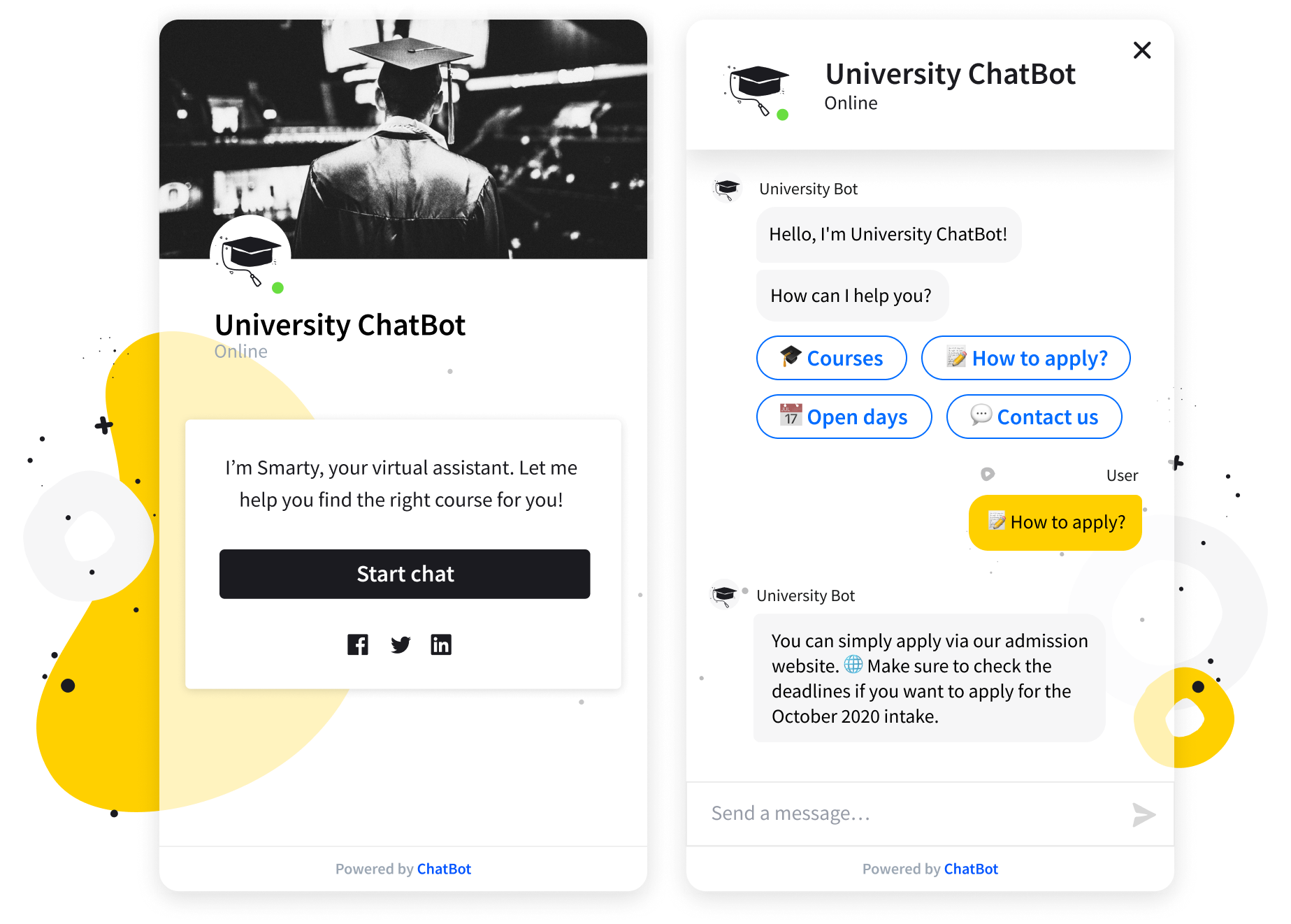
1. **Class Diagram**



1. **Database Design**



1. **Interface Design**



1. **Test Cases**

|  |  |
| --- | --- |
| **FR ID:** | **FR-01** |
| **Test Case Title:** | Student Inquiry Handling |
| **Test Case Id:** | TC-01 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User asks an inquiry. 2. Chatbot retrieves FAQs. 3. Chatbot provides a relevant response. |
| **Description:** | The chatbot answers FAQs related to courses, fees, schedules, trainers, content, and certification requirements. |
| **Alternative Paths:** | User may refine or rephrase their query if the chatbot does not initially find a match. |
| **Pre-Conditions:** | User has access to the chatbot interface. |
| **Post Conditions:** | User receives an accurate response. |
| **Exception:** | Chatbot fails to find an answer due to incomplete FAQs or incorrect query interpretation. |
| **Tested By:** | BC200401263 |
| **Result:** | Pass/Fail. |

|  |  |
| --- | --- |
| **FR ID:** | **FR-02** |
| **Test Case Title:** | Course Registration Inquiry |
| **Test Case Id:** | TC-02 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User views course catalog. 2. User selects a course. 3. Chatbot provides a registration form. |
| **Description:** | Allows students to register for available courses, completing the process with email confirmation. |
| **Alternative Paths:** | User may choose another course after viewing the catalog. |
| **Pre-Conditions:** | Courses are available and updated in the catalog. |
| **Post Conditions:** | User successfully registers for a course and receives a confirmation. |
| **Exception:** | Registration fails if required fields are incomplete or if there is a technical issue. |
| **Tested By:** | BC200401263 |
| **Result:** | Pass/Fail. |

|  |  |
| --- | --- |
| **FR ID:** | **FR-03** |
| **Test Case Title:** | Schedule Management Inquiry |
| **Test Case Id:** | TC-03 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User asks about schedule. 2. Chatbot provides class details, timings, or batch statuses. |
| **Description:** | Students can inquire about ongoing/upcoming class schedules, batch timings, or statuses. |
| **Alternative Paths:** | User can request details of a different class or batch if needed. |
| **Pre-Conditions:** | Schedule information is available in the system. |
| **Post Conditions:** | User gets accurate schedule details. |
| **Exception:** | Details cannot be retrieved due to missing or outdated schedule data. |
| **Tested By:** | BC200401263 |
| **Result:** | Pass/Fail. |

|  |  |
| --- | --- |
| **FR ID:** | **FR-04** |
| **Test Case Title:** | Resource Distribution Inquiry |
| **Test Case Id:** | TC-04 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User requests materials. 2. Chatbot provides links for download. |
| **Description:** | The chatbot enables students to download course brochures or training materials. |
| **Alternative Paths:** | User can try downloading alternate resources if initial selection fails. |
| **Pre-Conditions:** | Resources are uploaded and available in the system. |
| **Post Conditions:** | User successfully downloads requested resources. |
| **Exception:** | Resources unavailable or downloads fail due to server or file issues. |
| **Tested By:** | BC200401263 |
| **Result:** | Pass/Fail. |

|  |  |
| --- | --- |
| **FR ID:** | **FR-05** |
| **Test Case Title:** | Personalized Interaction |
| **Test Case Id:** | TC-05 |
| **Actor:** | Student (Customer) |
| **Actions:** | 1. User interacts with chatbot. 2. Chatbot analyses interaction. 3. Chatbot suggests courses. |
| **Description:** | The chatbot recommends suitable courses based on interaction history or specific inquiries. |
| **Alternative Paths:** | User may ignore the chatbot's recommendation and explore the course catalog manually. |
| **Pre-Conditions:** | User provides enough interaction data for analysis. |
| **Post Conditions:** | User receives personalized course suggestions. |
| **Exception:** | Recommendation fails if user data is insufficient or interaction history is incomplete. |
| **Tested By:** | BC200401263 |
| **Result:** | Pass/Fail. |

|  |  |
| --- | --- |
| **FR ID:** | **FR-06** |
| **Test Case Title:** | Management Tasks |
| **Test Case Id:** | TC-06 |
| **Actor:** | Admin |
| **Actions:** | 1. Admin logs in. 2. Admin performs tasks like managing course catalog, FAQs, or resources. |
| **Description:** | The chatbot answers FAQs related to courses, fees, schedules, trainers, content, and certification requirements. |
| **Alternative Paths:** | Admin can manage the course catalog, upload/download training materials, and update FAQs for improved system responses. |
| **Pre-Conditions:** | Admin may choose to delegate specific tasks (e.g., uploading brochures) to a subordinate. |
| **Post Conditions:** | System is updated with new courses, FAQs, and resources. |
| **Exception:** | Changes fail to save due to incorrect inputs or system errors (e.g., server downtime). |
| **Tested By:** | BC200401263 |
| **Result:** | Pass/Fail. |