EduBot: Deliverable 2

November 2024

Nour Hany - ID: 22-101068
Ahmed Gamal - ID: 22-101186
Shady Ali - ID: 22-101195
Ahmed Sameh - ID: 22-101198

Domain Classes: Noun Technique

November 2024

Case Study

EduBot is an AI-powered academic assistance system designed to support users including students, professors, and academic advisors in various academic activities. The system leverages advanced machine learning models and large language models (LLMs) to provide personalized support, including score predictions, academic recommendations, and assignment help for students. Professors and academic advisors can access class statistics, review performance insights, and assist students based on real-time and historical data.

Students interact with EduBot for their academic needs by providing their information, such as study habits, previous scores, and academic goals. The system predicts potential exam scores, offers tailored recommendations, and assists with assignment solutions. Academic advisors and professors can view aggregated data on student performance and offer guidance accordingly. The system also includes regular optimizations, retraining of models, and fine-tuning of data.

Tabular Representaion

Table 1: Noun Analysis for EduBot System

Identified Noun	**Notes on Including Noun as a Thing to Store**	**Decision**
Bot	Core Communication point for EduBot.	Include
Users	General entity covering all individual types (students, advisors) interacting with EduBot.	Include
Students	Primary users receiving academic support. Key for interactions and data tracking.	Include
Professors	Users who provide academic oversight, view statistics, and support students. (Same as advisors)	Exclude
Advisors	Users who monitor and assist with student performance through insights and guidance.	Include
Academic Activities	Broad term encompassing academic tasks. Might not need individual storage.	Exclude
Machine Learning Models	Technology component of EduBot, not directly stored as a system entity.	Exclude
Large Language Models (LLMs)	Part of the system's underlying functionality, rather than a stored entity.	Exclude
Score Predictions	Important output feature for student performance; may be generated dynamically.	Exclude
Academic Recommendations	Core feature, output of user interaction with Bot, possibly stored if personalized recommendations are kept for future reference.	Exclude
Assignment Help	Service provided to students, likely temporary as part of interactions.	Exclude
Class Statistics	Aggregated data for professors and advisors, useful for academic insights.	Exclude
Performance Insights	Analysis data that helps professors and advisors provide guidance.	Exclude
Historical Data	Stored information to support future predictions and recommendations.	Include
Information	Broad term; may be overly general and cover various specific data points in EduBot.	Exclude
Study Habits	Specific data on student routines, essential for personalized recommendations.	Exclude (part of historical data)
Academic Goals	Relevant for personalizing student recommendations; could be stored.	Exclude (part of historical data)
Exam Scores	Core data point for tracking academic progress and performance.	Exclude (part of historical data)
Assignment Solutions	Likely a temporary part of interaction, may not need long-term storage.	Exclude
Optimizations	Process-oriented function; part of backend operations, not stored as an entity.	Exclude
Fine-tuning	Related to model maintenance; backend process, not a stored entity.	Exclude
Message	Represents the text transferred between the student and the Bot	Include
Course	Each Student SCore record has a course tied to it	Include

UML Diagram

The following is the UML diagram for the EduBot system.

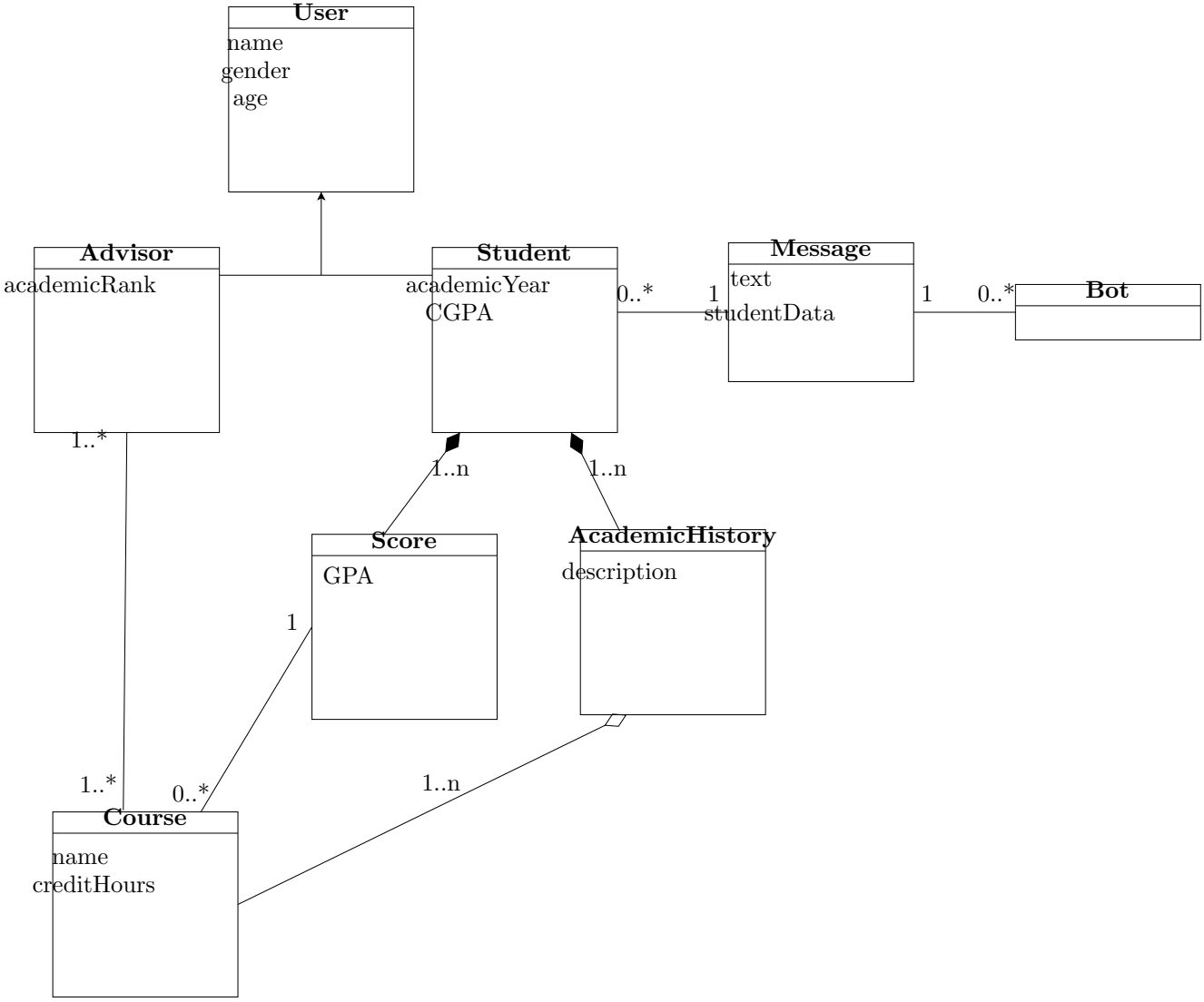

Figure 1: EduBot UML Class Diagram

CRUD Technique

Table 2: CRUD Verification for EduBot use cases

Domain Class	**CRUD**	**Verified Use Case**
User	C	Create User
	R	Get User Info
	U	Update User Info
	D	Delete User (left institution)
Student	C	Create Student
	R	Read Student Info
	U	Update Student Info
	D	Delete Student
Advisor	C	Create Advisor
	R	Read Advisor Info
	U	Update Advisor Info
	D	Delete Advisor
Course	C	Create New Course
	R	Read Course Info
	U	Update Course Info
	D	Delete Course
Message	C	Send Message to Bot
	R	Bot Reads Message
Academic History	C	Add To Academic History
	R	Fetch Academic history for bot/advisor
	U	Add To Academic History
	D	Delete Record from Academic History
Score	C	Add A Course Score
	R	Read Course Score
	U	Add A Course Score
	D	Delete Course Score

CRUD Technique

Table 3: CRUD Analysis Result: Use Case / Domain Class Matrix

Use Case vs. Domain Class	**User**	**Student**	**Advisor**	**Bot**
Storing Students Information	CU	CU		
Recommendation /Prediction	R	R		
Display Student Information	R	R		
Display Course Statistics	R	R		

Use Case vs. Domain Class	**Score**	**Academic History**	**Course**	**Message**
Storing Students Information	CU	CU	R	R
Recommendation /Prediction	R	R	R	R
Display Student Information	R	R	R	
Display Course Statistics	R	R	R	

Updated UML Use Case Diagram

The following is the Updated UML Use Case diagram for the EduBot system.

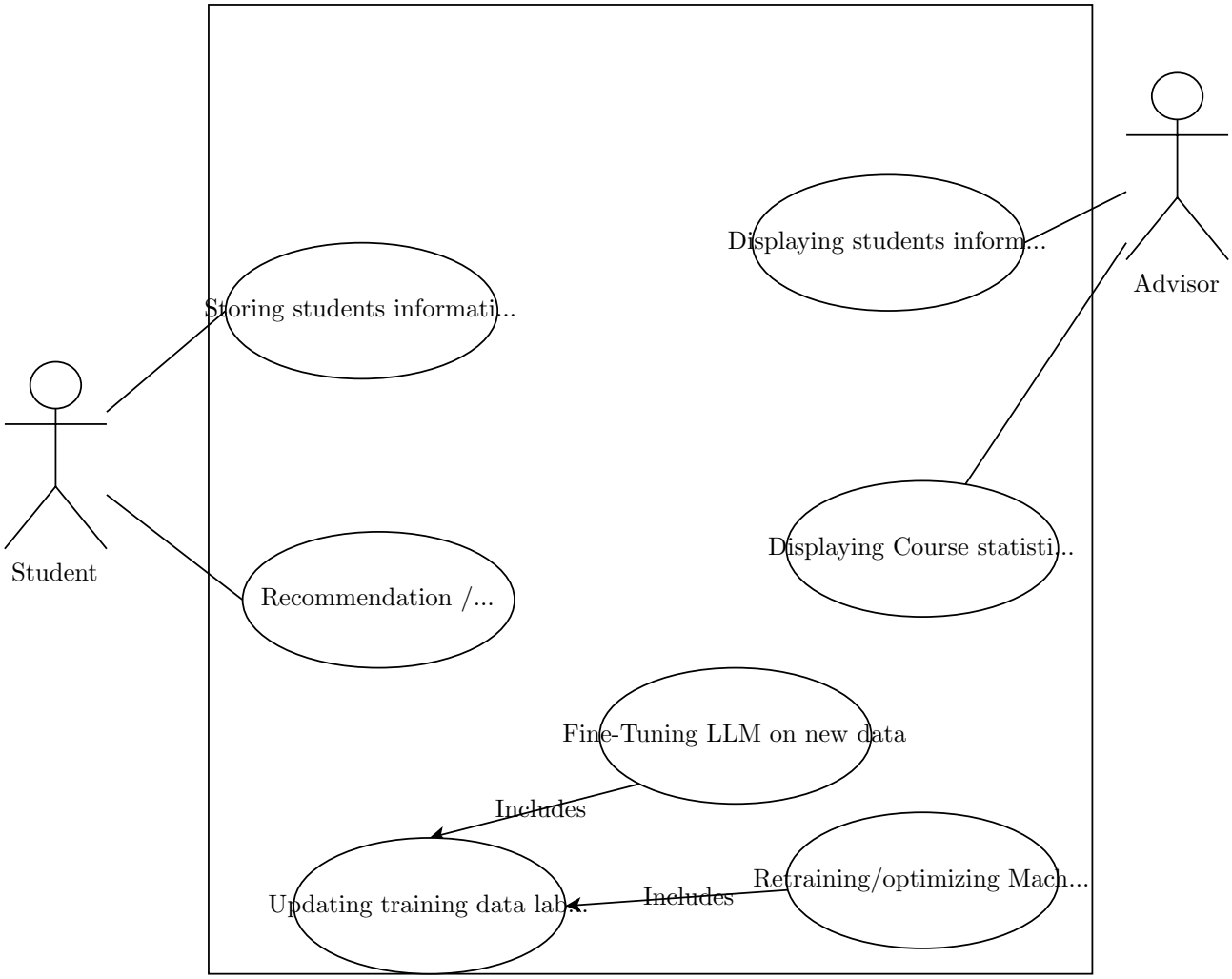

Figure 2: EduBot UML Use Case Diagram