Career Guidance Application

High National Diploma in Software Engineering

Final Project Report

2023.2F

Ekanayaka E. M. A. S. B.	COHNDSE232F-092
Chamikara P. A. G. P. J.	COHNDSE232F-035
P.G.Y.H. Madhusanka	COHNDSE232F-067
K. P. S. Nimsara	COHNDSE232F-039

School of Computing and Engineering

National Institute of Business Management

Colombo-7

Career Guidance Application

Authors:

Ekanayaka E. M. A. S. B.	COHNDSE232F-092
Chamikara P. A. G. P. J.	COHNDSE232F-035
P.G.Y.H. Madhusanka	COHNDSE232F-067
K. P. S. Nimsara	COHNDSE232F-039

Name of the Program: High National Diploma in Software Engineering 22.3F

Supervisor: Mr. Keerthi Kodithuwakku

Institution: School of Computing and Engineering, National Institute of Business Management

Date: 17/01/2025

"The project is submitted in partial fulfillment of the requirement of the Higher National Diploma in Software Engineering of National Institute of Business Management."

Declaration

“I certify that this project does not incorporate without acknowledgement any material previously submitted for a Higher National Diploma in any institution and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my project report, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and summary to be made available to outside organizations”

Preamble

Abstract

The career guidance application is a mobile application that will help students identify the best career opportunities in line with their educational background and personal interests. It will contain predictive algorithms that can make personalize recommendations on careers, bridging the gap between academic choices and career opportunities.

The app is built using Flutter for the frontend, Firebase for authentication, Spring Boot for backend services, and PostgreSQL for database management. It follows the Micro Service architecture, which is scalable and maintainable. Development processes will be covered by Agile methodology to make sure improvements continue from user feedback in the future.

Future enhancements will be made in integration with AI and machine learning for higher-level career predictions and partnering with universities and career counseling organizations.

Key Words

- Career Guidance,
- Predictive Algorithms,
- Android Studio,
- Firebase,
- PostgreSQL,
- Micro Service Architecture,
- Flutter
- Agile Methodology

List of Figures

List of Tables

List of Acronyms and Abbreviations

- MVC: Model-View-Controller
- ER: Entity-Relationship
- AI: Artificial Intelligence
- UI: User Interface

Acknowledgement

We clarify that, as far as We know and believe, this project is all original. We didn't copy anything from others without giving credit in the text. Also, we didn't use anything from a previous higher national diploma submission without acknowledging it. If our project report gets accepted, it's acceptable for it to be photocopied, shared between libraries, and have its title and summary shown to other groups.

Thank you!

Contents

1. Introduction

1.1. Introduction of the Application

The Career Guidance App is one of the creative, helpful apps for students to come over the difficult bridge between education and working. Using advanced algorithms and data analysis, the app gauges a user's academic achievements, skillsets, self-made extracurricular activities and personal interests. Then it produces accurate, tailored career suggestions, maximizing compatibility with each learner's individual features and goals. Further, the app features up and coming industry trends in real-time, providing students with the knowledge of how are the needs of the job market dynamic, and helping them with the decisions on their career path. It provides the tools and support needed for individuals to effectively reach their career aspirations while tackling a crucial gap in a structured approach to career pathways in the contemporary landscape.

1.2. Problem Definition

- **Uncertainty in Career Graph:** It is tough for a student to see how their academic performance and subject scores align with potential career trajectories. Students may feel unprepared for what lies ahead due to a lack of clarity in these questions.
- **Informal Resources Challenge:** The informal nature of some resources may lead to students relying on word-of-mouth recommendations that may not be suitable for their specific circumstances.
- **Missed Opportunities:** The misalignment between students' careers and their academic strengths leads many of them to miss out on opportunities that would have been a better fit for their skills and interests, resulting in lost potential.
- **Dynamic Industry Demands:** Industries are constantly redefining what is needed in the field, so students find themselves often behind the curve in terms of career expectations and skills necessary leading to a gap between education and industry.

1.3. Project Objectives

- **Data-driven personalized career recommendations:** Analyze academic qualifications and personal interests, and provide tailored career recommendations for each user.
- **Streamlined user experience:** Develop an easy-to-use application that allows students to be informed and involved in their career path.
- **Conform students to serve the market**: Create students who meet market demands by providing real-time updates on career trends.
- **Create a guided bridge between students education and their careers:** Guide them through the process of aligning education with career goals.
- **Data Privacy:** It means maintaining ethics by generating synthetic data whenever possible to avoid touching sensitive real-world data and respecting privacy when doing so.

1.4. Proposed Solution

The Career Guidance App is a mobile application which will help students by providing predictive algorithms and simple interface. It analyzes data including academic credentials, extracurricular accomplishments, and personal interests to offer personalized career paths. With ever-changing professional landscape, this app provides all its users a glimpse of upcoming career sectors, keeping them in pace with what the industries seek. Also, it is user-friendly, allowing for navigation that students can leverage to discover careers, plan ahead, link their current academic pursuits with future professional opportunities.

1.5. Chapter Summary

The Next Step App is A mobile platform Career Guidance App that helps students get career recommendations based on what they studied, skills, clubs they were in, things they are interested in, and combines this information to start transitioning students from education to the workforce. It solves the difficulties

with career mapping, lack of structured guidance, regular missed opportunities, and a rapidly changing industry needs. With its user-friendly interface and immediate access to ongoing career developments, the app allows students to make fact-based decisions about their future and ground their studies in their future work patterns.

2. Methodology

2.1. Introduction

To effectively tackle Career Guidance App the project will embrace the methodology. Agile is a collaborative approach that prioritizes customer development promoting flexibility and iterative processes. This makes it a suitable choice for developing software solutions that evolve over time based on user feedback and evolving requirements.

2.2. Data Collection Method(s)

Organizations were reluctant to share sensitive data they needed to train the AI model necessary for the Career Guidance App due to privacy issues. To overcome this shortcoming, we created a synthetic data set that replicates real-world conditions. Specific key attributes such as education, grades in various subjects, and interests were recorded along with that synthetic data. This included forming realistic pattern and relationship between the data points to ensure the AI model could still produce accurate and meaningful career suggestions. Using this method, we were able to achieve our AI model through an ethical process.

2.3. Software Process Model

We are not going to cover all of this in depth but rather it will be an overview, we will be taking the approach to implement Scrum which is one of the popular framework inside Agile. Furthermore, the project follows a microservices architecture where each module (like user authentication, data processing and career recommendation) can run independently of one another while offering scalability, flexibility and easier maintenance. In this method project is broken down into iterations called “sprints” where each sprint typically lasts 2 to 4 weeks. This framework also encourages opening up, being flexible, and delivering value. The project is set to be done in a series of sprints, delivering functionalities related to Career Guidance App on each sprint.

2.4. Software Development Tools

In Agile development we rely on a range of collaboration and project management tools to make communication, planning and tracking more efficient. For source management we'll be using tools, like Git & GitHub. For IDE ,we'll be using Visual studio code, Android Studio, IntelliJ Ultimate . For UML design tools Draw.io and for prototyping we'll use Figma. For database management PostgreSQL. Docker for containerization and orchestration.

As for communication, we'll have meetings (Physical and virtual). Use online platforms such as Google meet.

2.5. Testing Strategies

We hope to use Unit testing and Integration testing for validate that each of our software component performs as designed. Finally, we hope to use system testing to covers all the combined parts of our system.

Unit testing focuses on individual components in isolation, ensuring that each function or module operates correctly.

Integration testing verifies how different components interact and communicate with each other, ensuring seamless integration.

System testing encompasses the entire software system, evaluating its overall functionality, performance, and behavior under real-world scenarios.

Also, we will test this app on the live environment for better improvement for the users.

2.6. Implementation Plan

Objective: Implement a comprehensive career guidance application to enhance students career opportunities.

Scope: Select, install, configure, integrate, and train on a career guidance app solution that aligns with our application requirements and workflows.

Timeline:

1. Phase 1: Analysis (1 weeks)
2. Phase 2: Software Design and Prototyping (3 weeks)
3. Phase 3: Integration and Data Migration (1 weeks)
4. Phase 4: Testing (1 week)
5. Phase 5: Implementation (2 weeks)
6. Phase 6: Monitoring and Continuous Improvement (Ongoing)

Budget:

1. Server costs: $25 per month

Success Criteria:

- Successfully launch our career guidance application to the users.
- Seamless testing and monitoring for best user experience

2.7. Chapter Summery

This chapter outlines the methodology chosen for the project, including Agile principles, microservices architecture, data collection methods, Scrum as the software process model, software development tools, testing strategies, and an implementation plan. It serves as a foundational chapter in understanding how we aims to address career guidance based on educational background of the user challenges through a well-structured and collaborative approach.

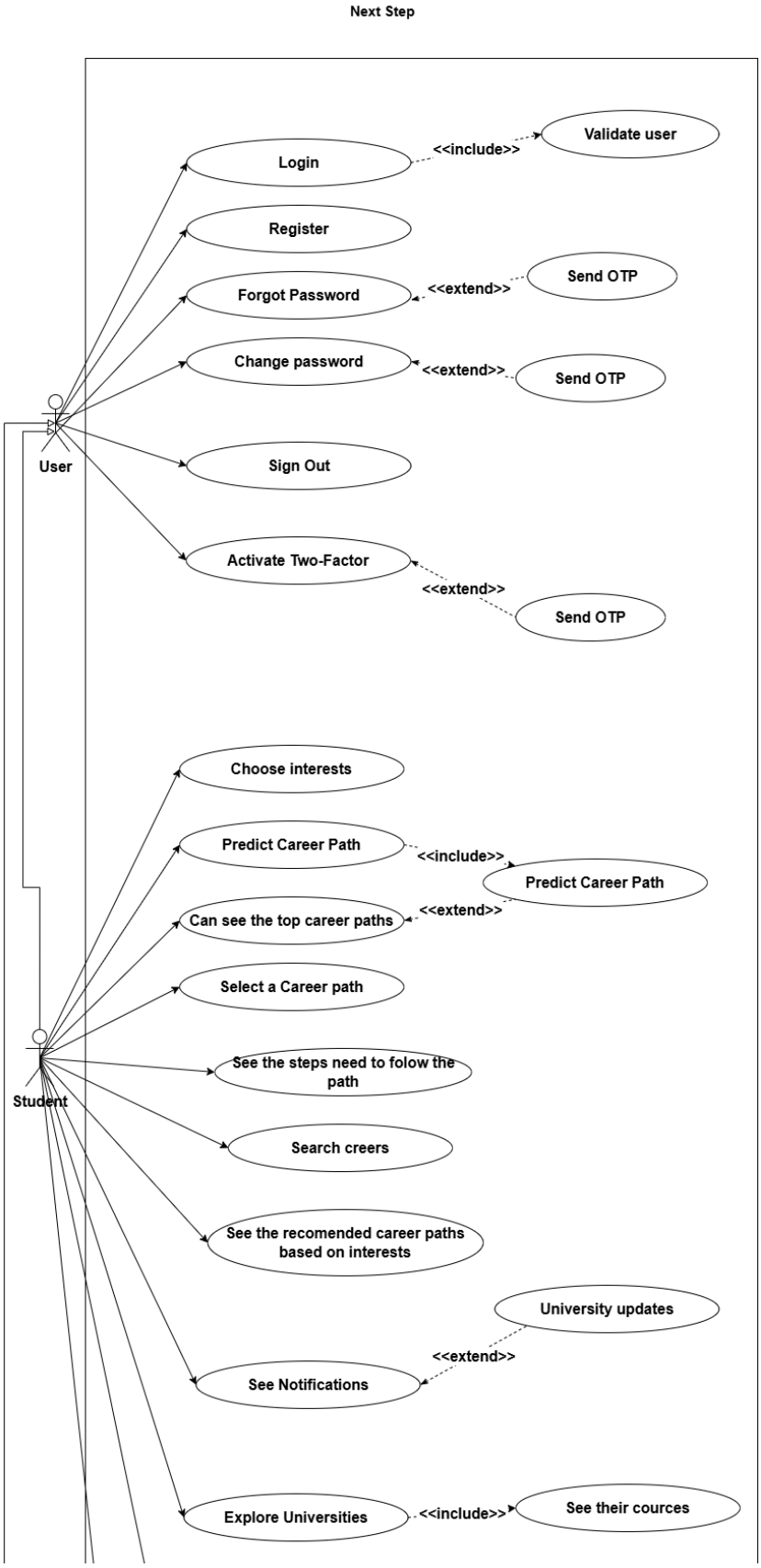
3. Analysis

3.1. Introduction

This section covered the way of analyzing the problem. Additionally, it covers Class diagram of proposed system, Sequence Diagrams (Each Use case) for Proposed System and ER Diagram of the Proposed System.

3.2. UML Diagram.

3.2.1. Use Case Diagram of Proposed System.



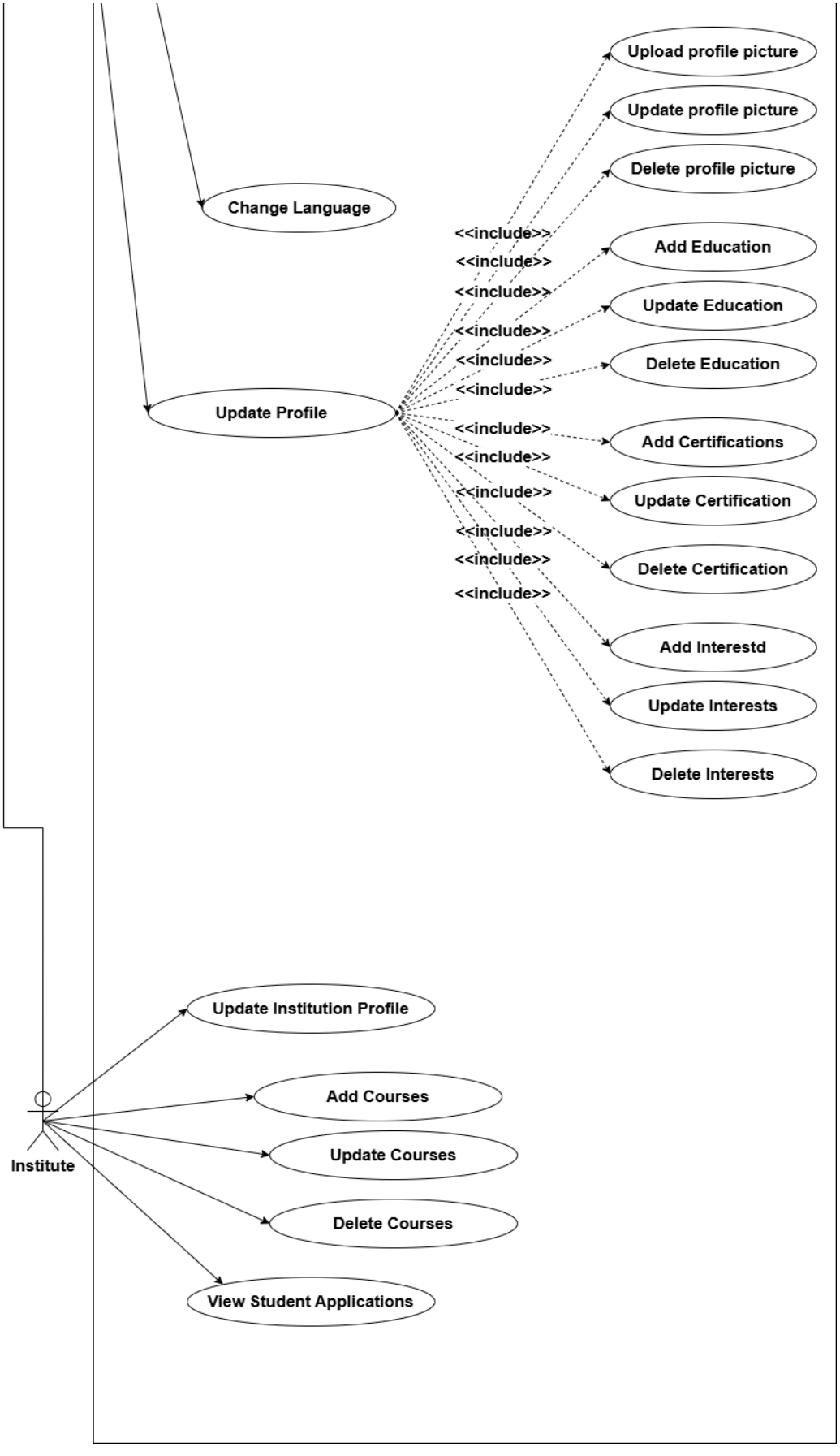


Figure 1: Use case Diagram

3.2.2. Class Diagram of Proposed System

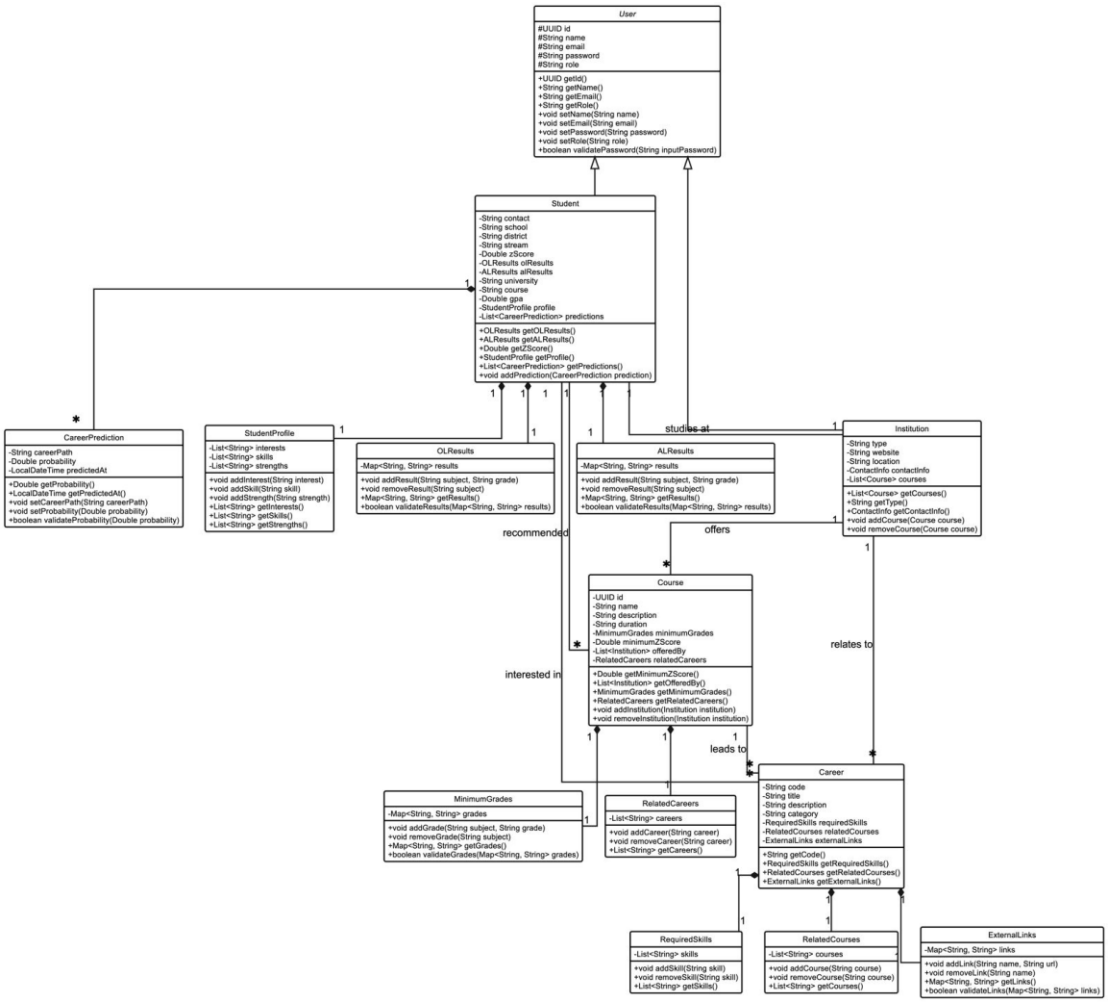


Figure 2: Class Diagram

3.2.3. Sequence Diagrams (Each Use case) for Proposed System

User Register Sequence

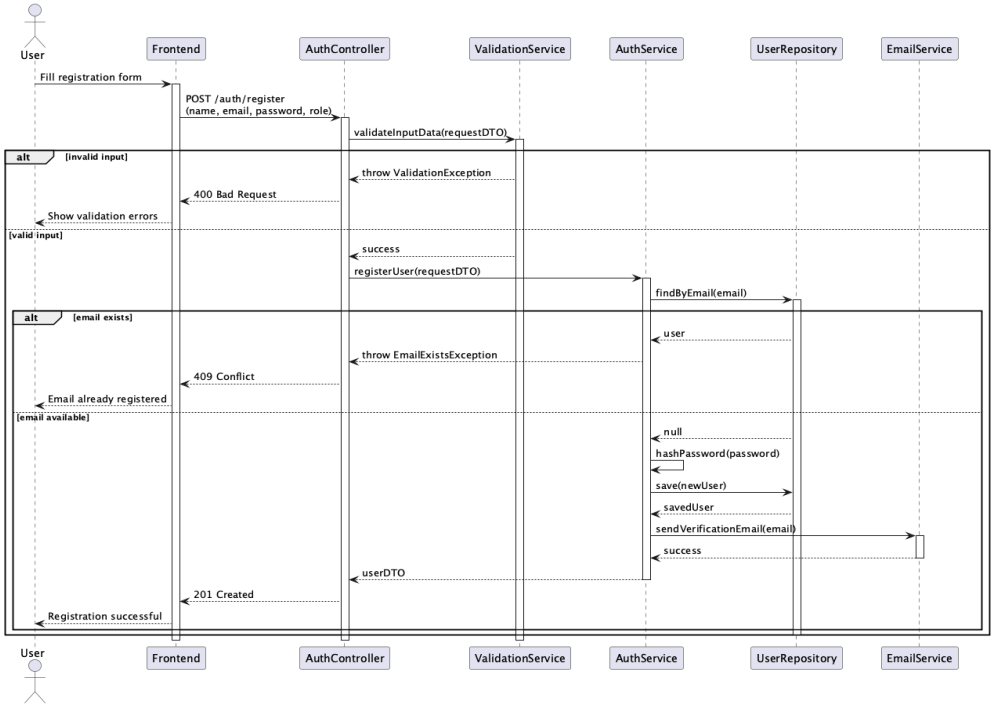


Figure 3: User Register Sequence

User Login Sequence

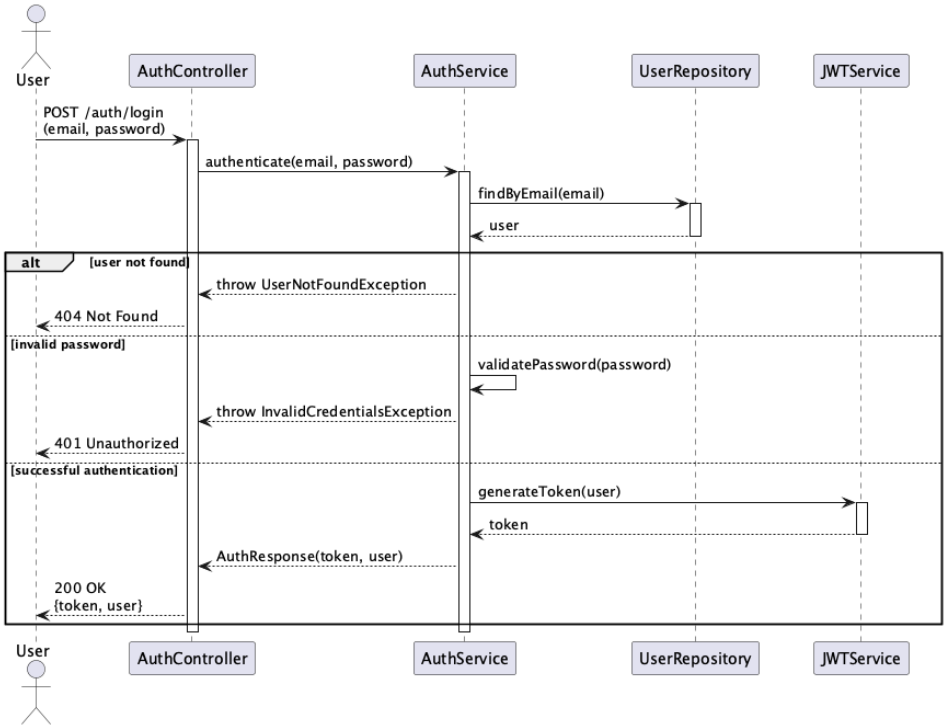


Figure 4: User Login Sequence

Profile Management Sequence

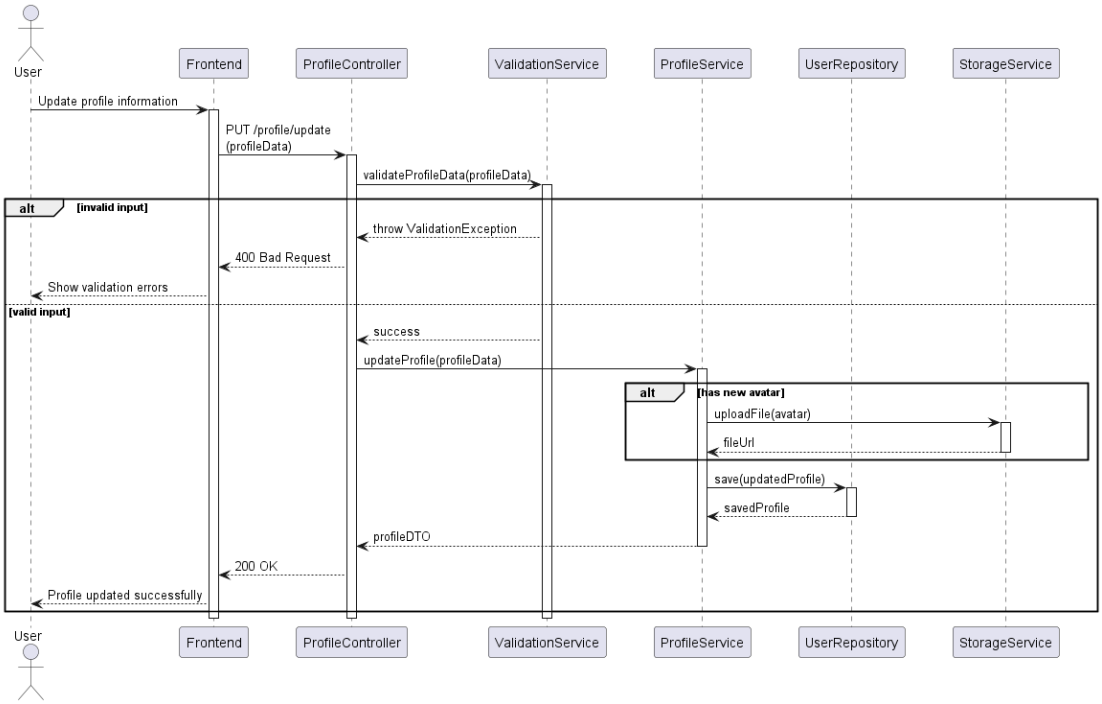


Figure 5: Profile Management Sequence

Password Reset sequence

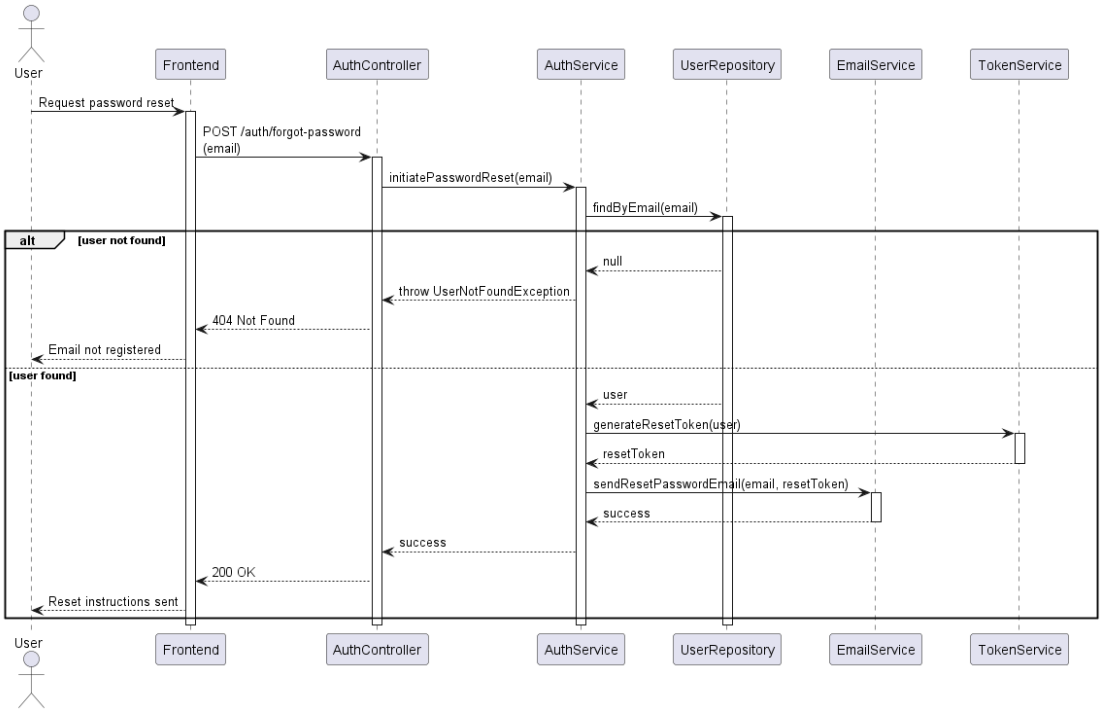


Figure 6: Password Reset sequence

Institute Management Sequence

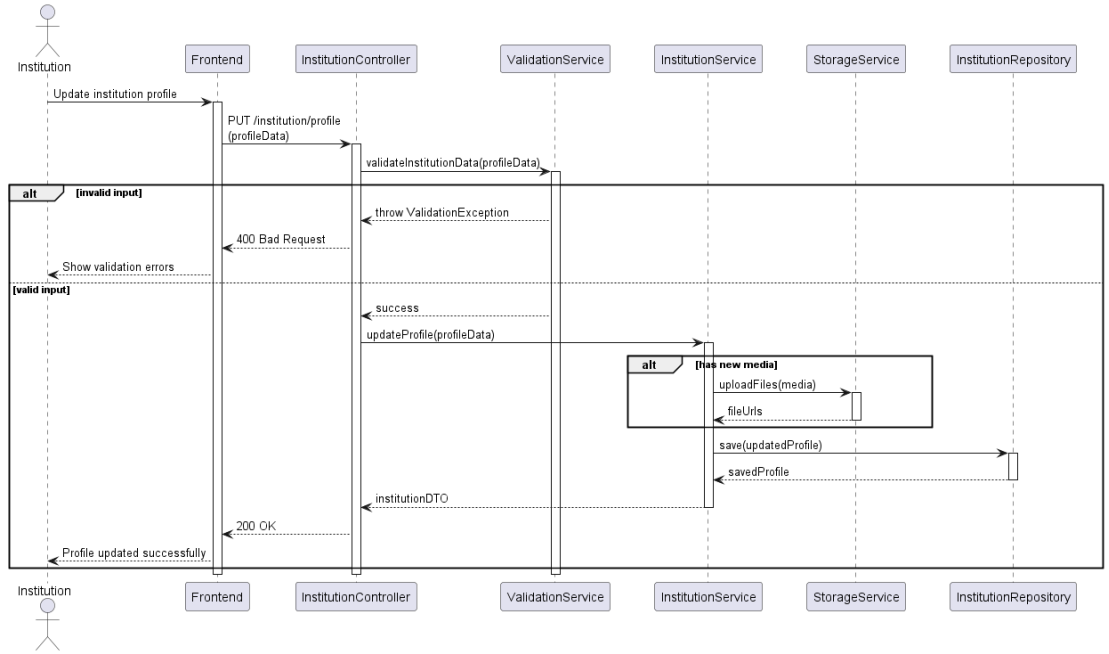


Figure 7: Institute Management Sequence

Career Recommendation Sequence

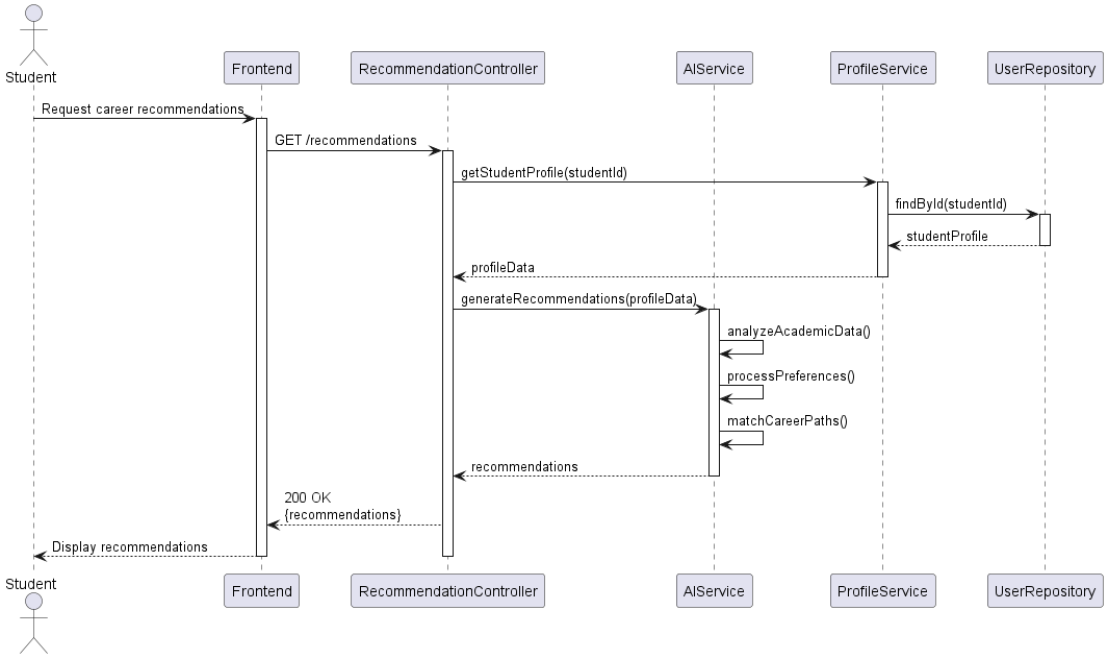


Figure 8: Career Recommendation Sequence

Course Exploration Sequence

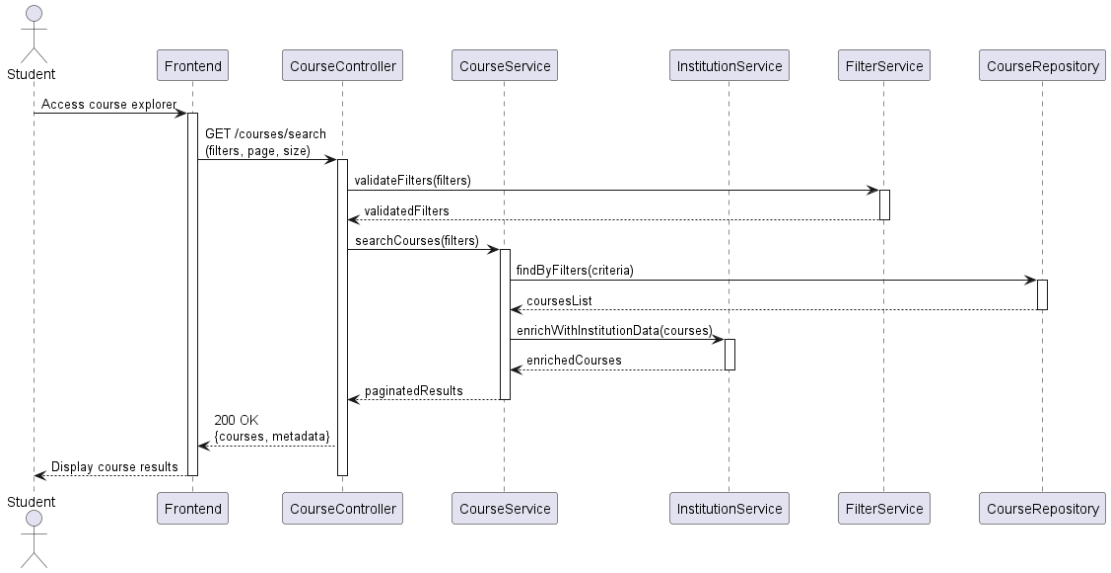


Figure 9: Course Exploration Sequence

Program Tracking Sequence

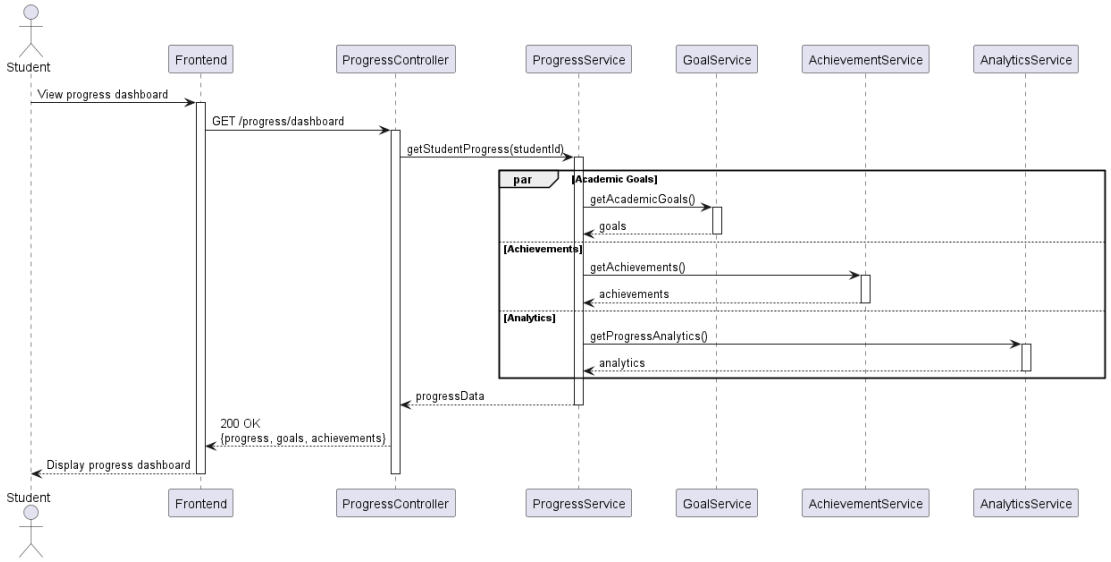


Figure 10: Program Tracking Sequence

AI Configuration Sequence

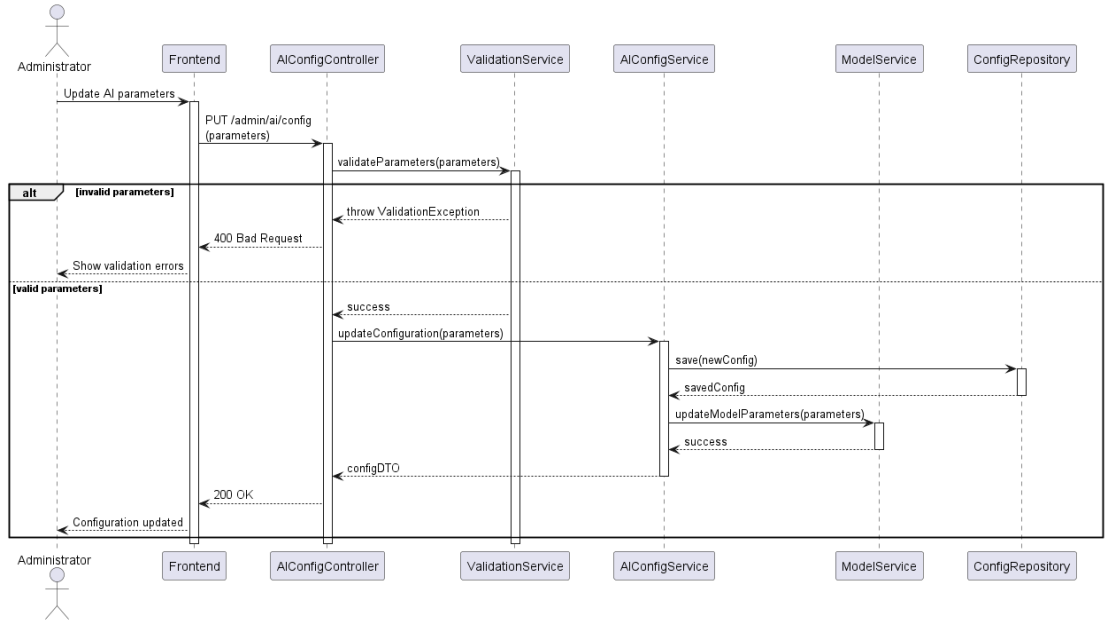


Figure 11: AI Configuration Sequence

Admission Management Sequence

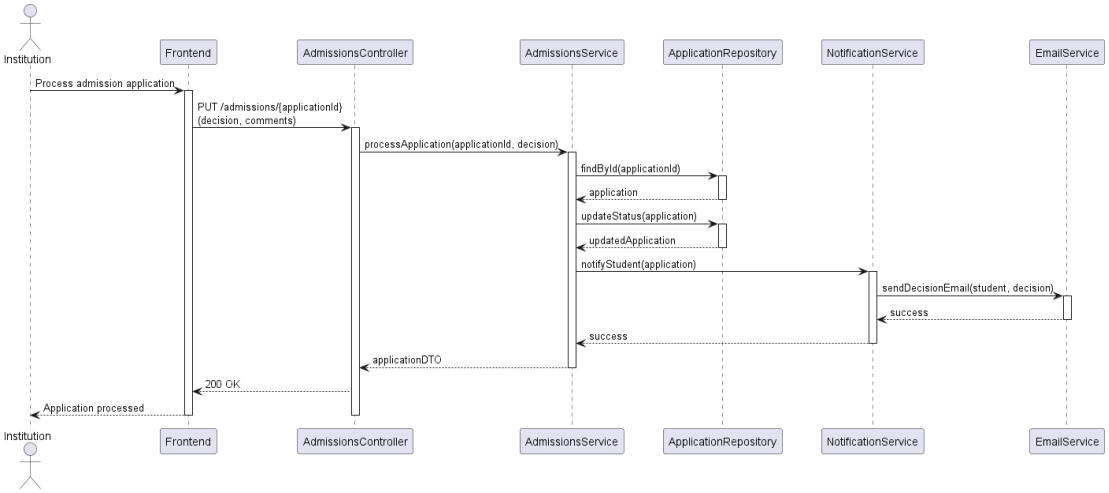


Figure 12: Admission Management Sequence

Admin Monitoring Sequence

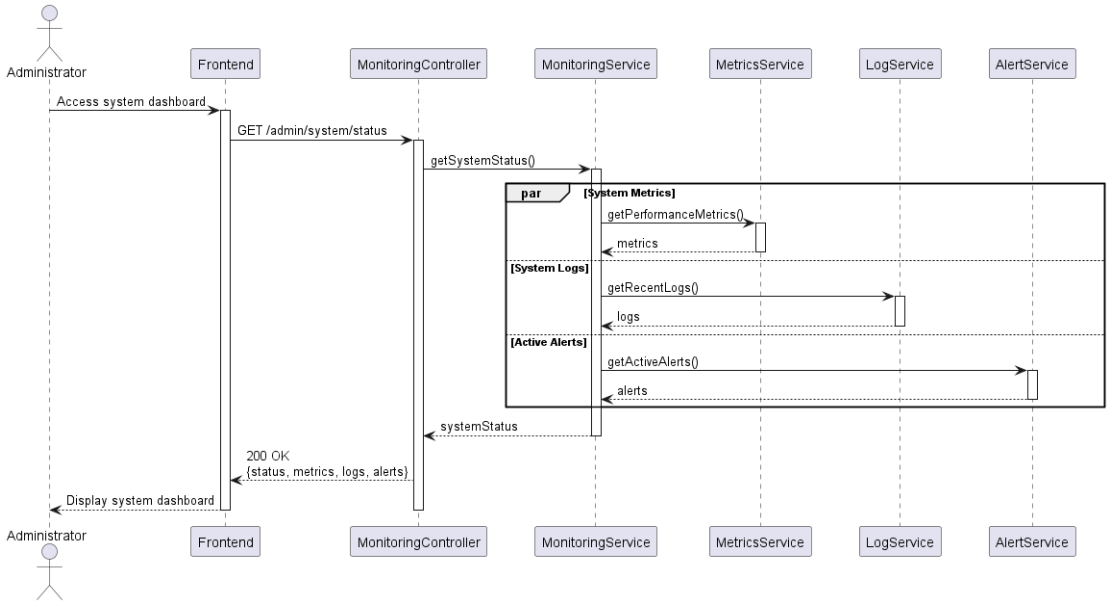


Figure 13: Admin Monitoring Sequence

3.3. Software Architecture.

It uses MVC microservice architecture to separate concerns and ensure scalability. Moreover, the system follows a microservices architecture where each functionality of the application like user interfaces, career recommendation and prediction model, data repository, etc., runs as an independent service. This helps in creating a loose Based on the scenario discussed above, development of apps become easy and you can maintain it with no problems as each module is loosely coupled and work independently. A career recommendation service can evolve completely separate, which allows flexibility and adaptability as the app itself grows.

3.4. ER Diagram of the Proposed System.

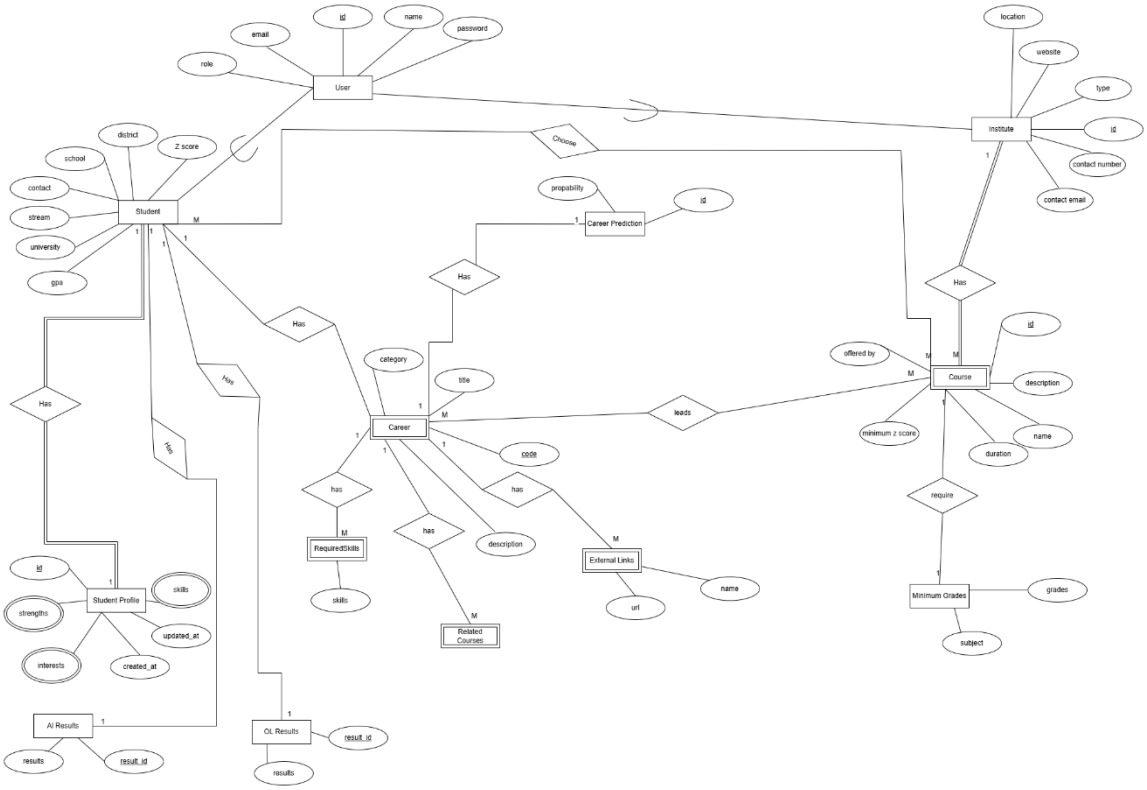


Figure 14: ER Diagram

3.5. Chapter Summary

This chapter displays key diagrams that give the overall structure and functionality of our career guidance application. The use case diagram highlights the interactions between users and the system, highlighting essential features such as finding a career path, and finding a best course. The class diagram gives a detailed view of the system's architecture, showing the relationships between different classes and their attributes. The entity-relationship (ER) diagram shows the database structure, depicting entities, relationships, and data flow within the system.

4. Solution Design

4.1. Introduction

This section covered the way of Design the problem. It covers User Interfaces, Database design, and Business viability of the Application.

4.2. Interface Design

Interface 01: Login UI

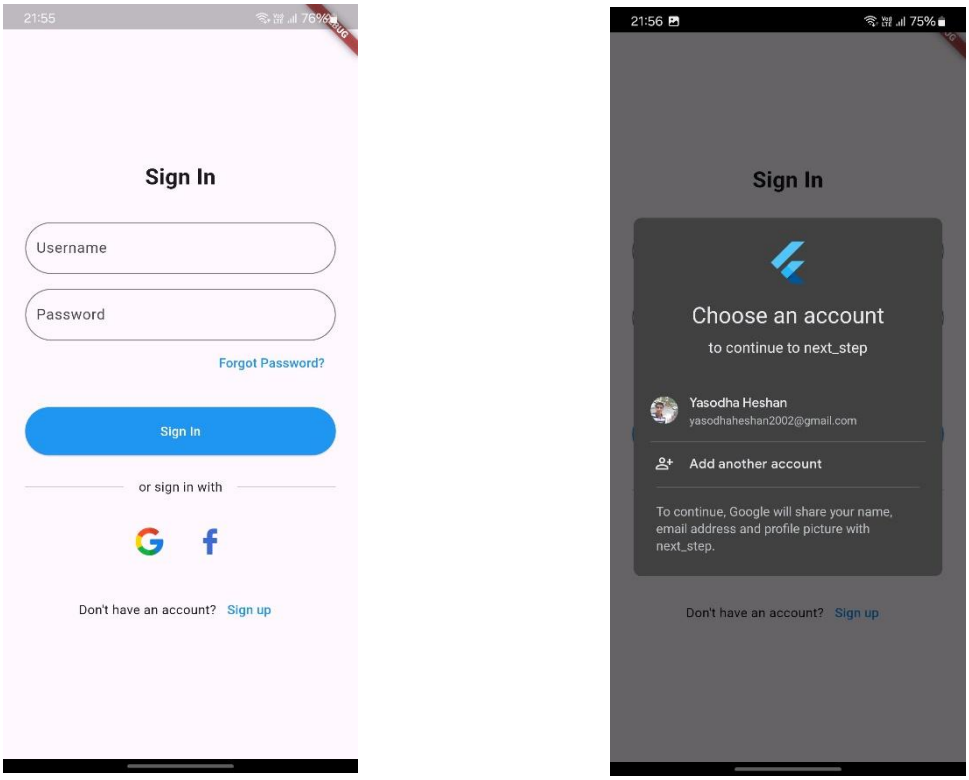


Figure 15: Login UI

Interface 02: Forgot Password UI

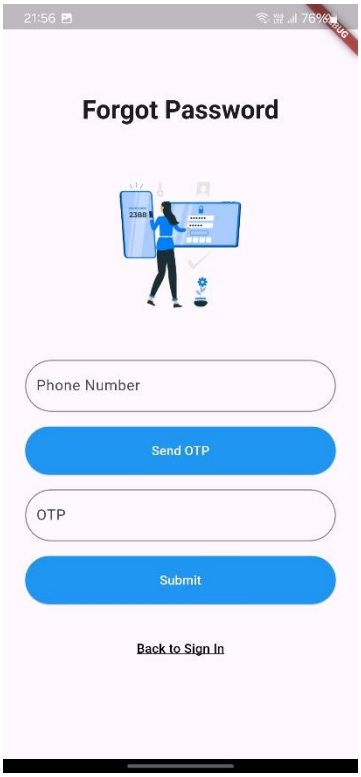


Figure 16: Forgot Password UI

Interface 03: Sign Up UI

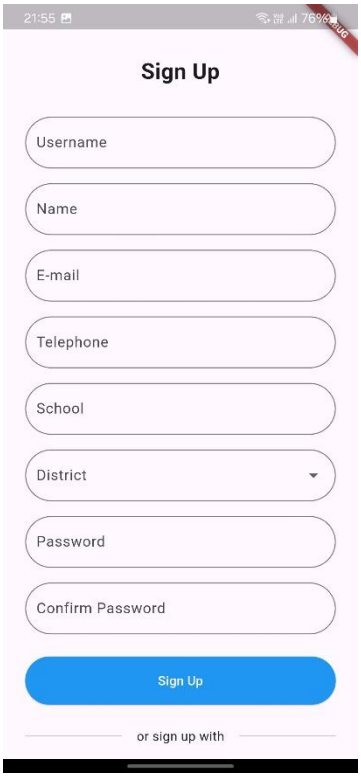


Figure 17: Sign UP UI

Interface 04: Home UI

Figure 18: Home UI

Interface 05: Recommend UI

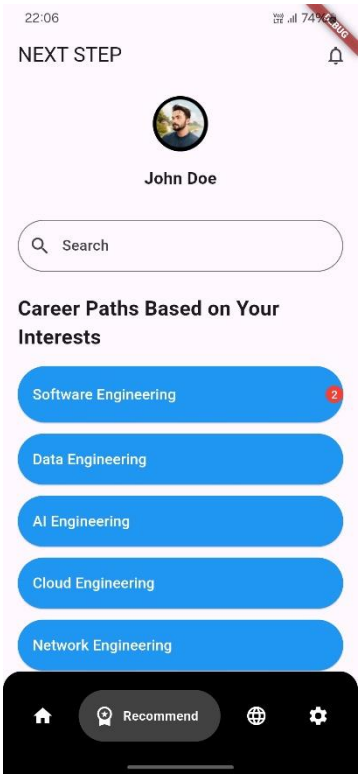


Figure 19: Recommend UI

Interface 06: Detailed Recommend UI

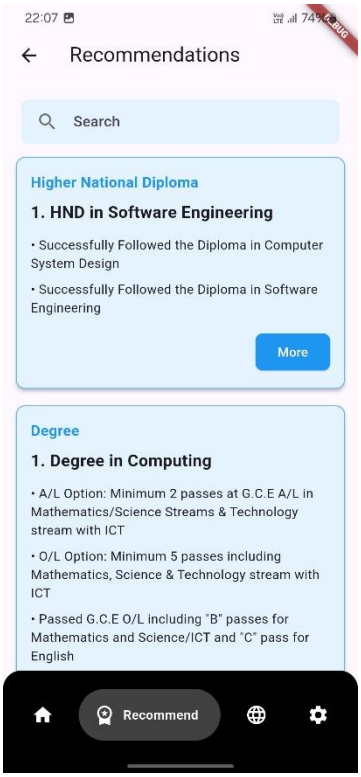


Figure 20: Detailed Recommend UI

Interface 07: Notification UI

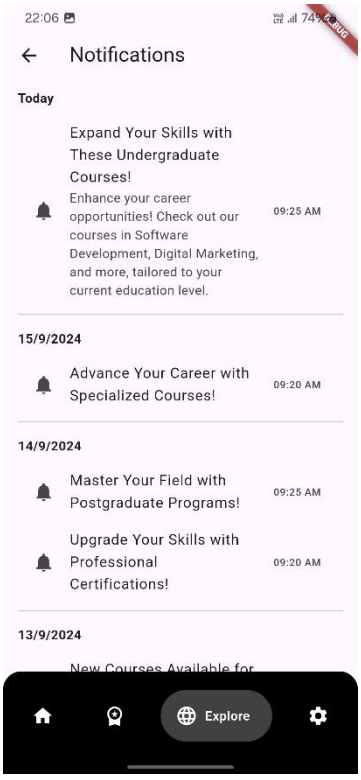


Figure 21: Notification UI

Interface 08: Explore UI

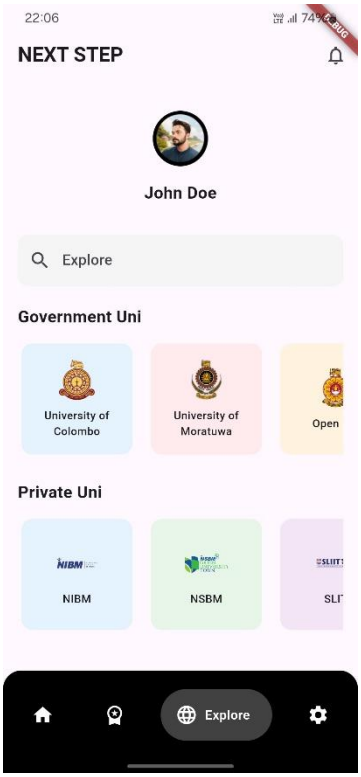


Figure 22: Explore UI

Interface 09: Detailed Explore UI

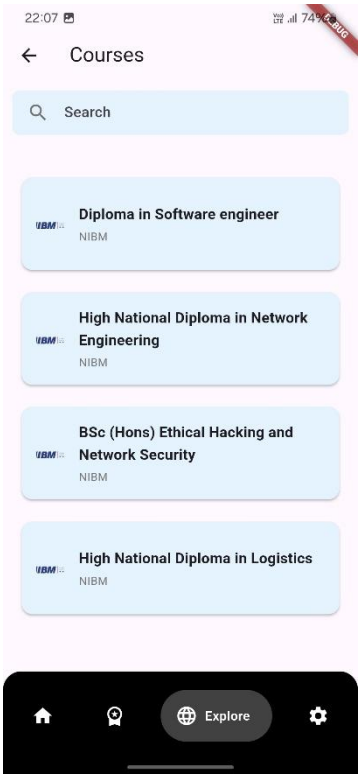


Figure 23: Detailed Explore UI

Interface 10: Profile UI

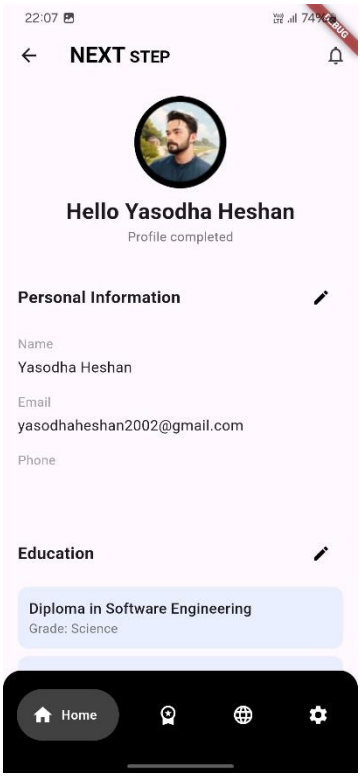


Figure 24: Profile UI

Interface 11: Settings UI

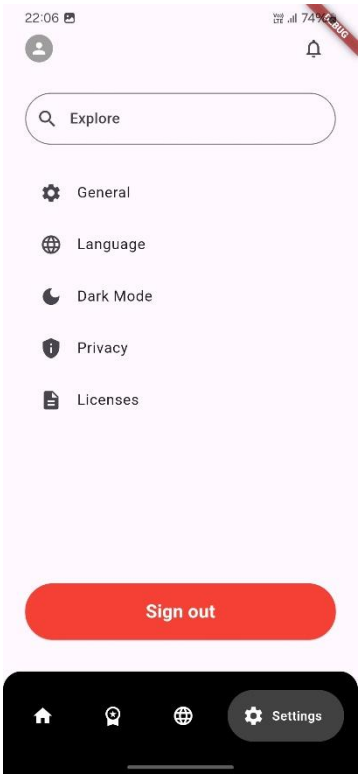


Figure 25: Settings UI

Interface 12: Setting General UI

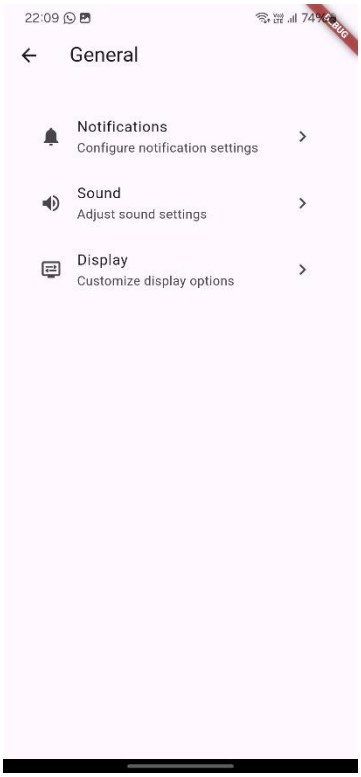


Figure 26: Setting General UI

Interface 13: Edit Profile UI

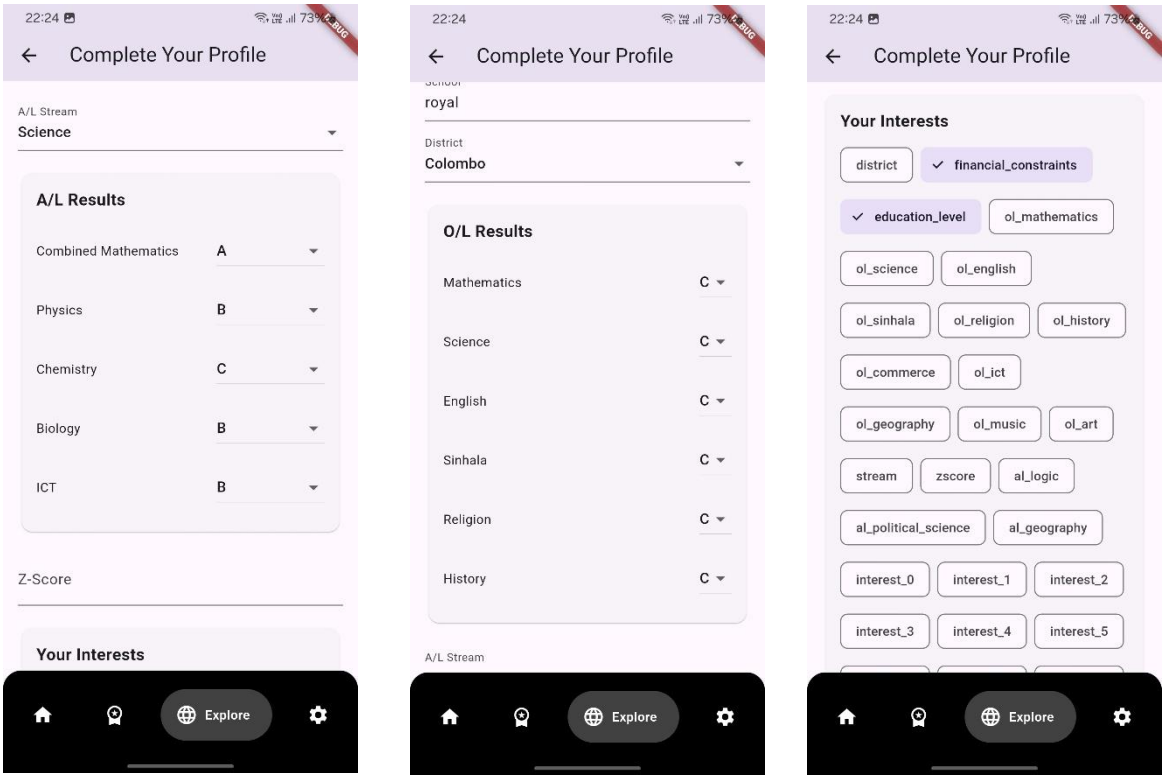


Figure 27: Edit Profile UI

4.3. Database Design

1. Users

Column Name	Data Type	Size	Constraints	Description
id	UUID	-	PRIMARY KEY	Unique identifier
name	VARCHAR	100	NOT NULL	User's full name
email	VARCHAR	100	NOT NULL, UNIQUE	User's email address
password	VARCHAR	255	NOT NULL	Hashed password
role	VARCHAR	20	NOT NULL	User role (student/institution)
created_at	TIMESTAMP	-	NOT NULL	Record creation timestamp
updated_at	TIMESTAMP	-	NOT NULL	Record update timestamp

Table 1: Users

2. Students

Column Name	Data Type	Size	Constraints	Description
id	UUID	-	PRIMARY KEY, FK(users.id)	Student ID
contact	VARCHAR	20	NOT NULL	Contact number
school	VARCHAR	100	NOT NULL	School name
district	VARCHAR	50	NOT NULL	District
stream	VARCHAR	50	NOT NULL	Study stream
z_score	DECIMAL	(4,2)	-	Z-Score
university	VARCHAR	100	-	Current university
course	VARCHAR	100	-	Current course
gpa	DECIMAL	(3,2)	-	Current GPA
institution_id	UUID	-	FK(institutions.id)	Current institution

Table 2: Students

3. Student_Profiles

Column Name	Data Type	Size	Constraints	Description
id	UUID	-	PRIMARY KEY	Profile ID
student_id	UUID	-	FK(students.id)	Reference to student
created_at	TIMESTAMP	-	NOT NULL	Record creation timestamp
updated_at	TIMESTAMP	-	NOT NULL	Record update timestamp

Table 3: Student Profile

4. Profile_Interests

Column Name	Data Type	Size	Constraints	Description
profile_id	UUID	-	FK(student_profiles.id)	Reference to profile
interest	VARCHAR	100	NOT NULL	Interest description
PRIMARY KEY	-	-	(profile_id, interest)	Composite key

Table 4: Profile Interests

5. Profile_Skills

Column Name	Data Type	Size	Constraints	Description
profile_id	UUID	-	FK(student_profiles.id)	Reference to profile
skill	VARCHAR	100	NOT NULL	Skill description
PRIMARY KEY	-	-	(profile_id, skill)	Composite key

Table 5: Profile Skills

6. Profile_Strengths

Column Name	Data Type	Size	Constraints	Description
profile_id	UUID	-	FK(student_profiles.id)	Reference to profile
strength	VARCHAR	100	NOT NULL	Strength description
PRIMARY KEY	-	-	(profile_id, strength)	Composite key

Table 6: Profile Strengths

7. OL_Results

Column Name	Data Type	Size	Constraints	Description
student_id	UUID	-	PRIMARY KEY, FK(students.id)	Reference to student
subject	VARCHAR	50	NOT NULL	Subject name
grade	VARCHAR	2	NOT NULL	Grade achieved
PRIMARY KEY	-	-	(student_id, subject)	Composite key

Table 7: OL Results

8. AL_Results

Column Name	Data Type	Size	Constraints	Description
student_id	UUID	-	PRIMARY KEY, FK(students.id)	Reference to student
subject	VARCHAR	50	NOT NULL	Subject name
grade	VARCHAR	2	NOT NULL	Grade achieved
PRIMARY KEY	-	-	(student_id, subject)	Composite key

Table 8: AL Results

9. Institutions

Column Name	Data Type	Size	Constraints	Description
id	UUID	-	PRIMARY KEY, FK(users.id)	Institution ID
type	VARCHAR	50	NOT NULL	Institution type
website	VARCHAR	255	-	Website URL
location	VARCHAR	255	NOT NULL	Physical location
contact_phone	VARCHAR	20	NOT NULL	Contact number
contact_email	VARCHAR	100	NOT NULL	Contact email

Table 9: Institution

10. Courses

Column Name	Data Type	Size	Constraints	Description
id	UUID	-	PRIMARY KEY	Course ID
name	VARCHAR	200	NOT NULL	Course name
description	TEXT	-	NOT NULL	Course description
duration	VARCHAR	50	NOT NULL	Course duration
minimum_z_score	DECIMAL	(4,2)	-	Minimum Z-Score required

Table 10: Courses

11. Course_Minimum_Grades

Column Name	Data Type	Size	Constraints	Description
course_id	UUID	-	FK(courses.id)	Reference to course
subject	VARCHAR	50	NOT NULL	Subject name
minimum_grade	VARCHAR	2	NOT NULL	Minimum grade required
PRIMARY KEY	-	-	(course_id, subject)	Composite key

Table 11: Coarse Minimum Grades

12. Institution_Courses

Column Name	Data Type	Size	Constraints	Description
institution_id	UUID	-	FK(institutions.id)	Reference to institution
course_id	UUID	-	FK(courses.id)	Reference to course
PRIMARY KEY	-	-	(institution_id, course_id)	Composite key

Table 12: Institution Courses

13. Careers

Column Name	Data Type	Size	Constraints	Description
code	VARCHAR	20	PRIMARY KEY	Career code
title	VARCHAR	100	NOT NULL	Career title
description	TEXT	-	NOT NULL	Career description
category	VARCHAR	50	NOT NULL	Career category

Table 13: Careers

14. Career_Skills

Column Name	Data Type	Size	Constraints	Description
career_code	VARCHAR	20	FK(careers.code)	Reference to career
skill	VARCHAR	100	NOT NULL	Required skill
PRIMARY KEY	-	-	(career_code, skill)	Composite key

Table 14: Career Skills

15. Career_Courses

Column Name	Data Type	Size	Constraints	Description
career_code	VARCHAR	20	FK(careers.code)	Reference to career
course_id	UUID	-	FK(courses.id)	Reference to course
PRIMARY KEY	-	-	(career_code, course_id)	Composite key

Table 15: Career Courses

16. Career_External_Links

Column Name	Data Type	Size	Constraints	Description
career_code	VARCHAR	20	FK(careers.code)	Reference to career
name	VARCHAR	100	NOT NULL	Link name/description
url	VARCHAR	255	NOT NULL	External URL
PRIMARY KEY	-	-	(career_code, name)	Composite key

Table 16: Career External Links

17. Career_Predictions

Column Name	Data Type	Size	Constraints	Description
id	UUID	-	PRIMARY KEY	Prediction ID
student_id	UUID	-	FK(students.id)	Reference to student
career_code	VARCHAR	20	FK(careers.code)	Reference to career
probability	DECIMAL	(5,4)	NOT NULL	Prediction probability
predicted_at	TIMESTAMP	-	NOT NULL	Prediction timestamp

Table 17: Career Predictions

4.4. Business Viability

It fills a crucial gap in career guidance and planning, which highlights its strong business viability. Partnership opportunities: Collaborations with universities, career counseling services, and job placement agencies could significantly expand its reach and credibility. The app can make its way into universities as part of their student services and help them to better prepare their students for the job that is waiting out there for them. Career coaching companies can use the app's predicting models for customized suggestions and recruiting companies can benefit from its analysis models to find better candidates for different positions. Furthermore, the app can have a subscription-based model for premium features, including personalized professional mentoring and sophisticated market analysis. It can also make money by partnering with schools and advertisers who would target students.

5. Conclusion

The Career Guidance App properly offers personalized career recommendations. Although the app meets its objectives, future improvements could include AI integration for better predictions. This enhancement would enable more precise and adaptive career suggestions by incorporating real time data and advanced machine learning techniques. Furthermore, the expansion of the app to include partnerships with educational institutions and career counseling organizations could greatly increase its availability and influence. In the course of further developing its features, the app may realize its potential to be a primary resource in helping to close the education and career planning gap.

6. References

Webb, P., & Syer, D. (2016). Spring Boot: Simplifying the development of production-ready applications. Spring Framework Journal, 5(1), 45-60. https://spring.io/guides/gs/spring-boot/

Firebase. (n.d.). Authenticate using Firebase with Google sign-in. Firebase. https://firebase.google.com/docs/auth/web/google-signin

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://scikit-learn.org/stable/

Biørn-Hansen, A., Majchrzak, T. A., & Grønli, T. M. (2017). Progressive web apps: The possible web-native unifier for mobile development. In International Conference on Web Information Systems and Technologies (pp. 344-351). SCITEPRESS SciTePress - Publication Details

Merkel, D. (2014). Docker: Lightweight Linux containers for consistent development and deployment. Linux Journal, 2014(239), Article2. https://www.linuxjournal.com/content/docker-lightweight-linux-containers-consistent-development-and-deployment

7. Appendices

AI model training matrix

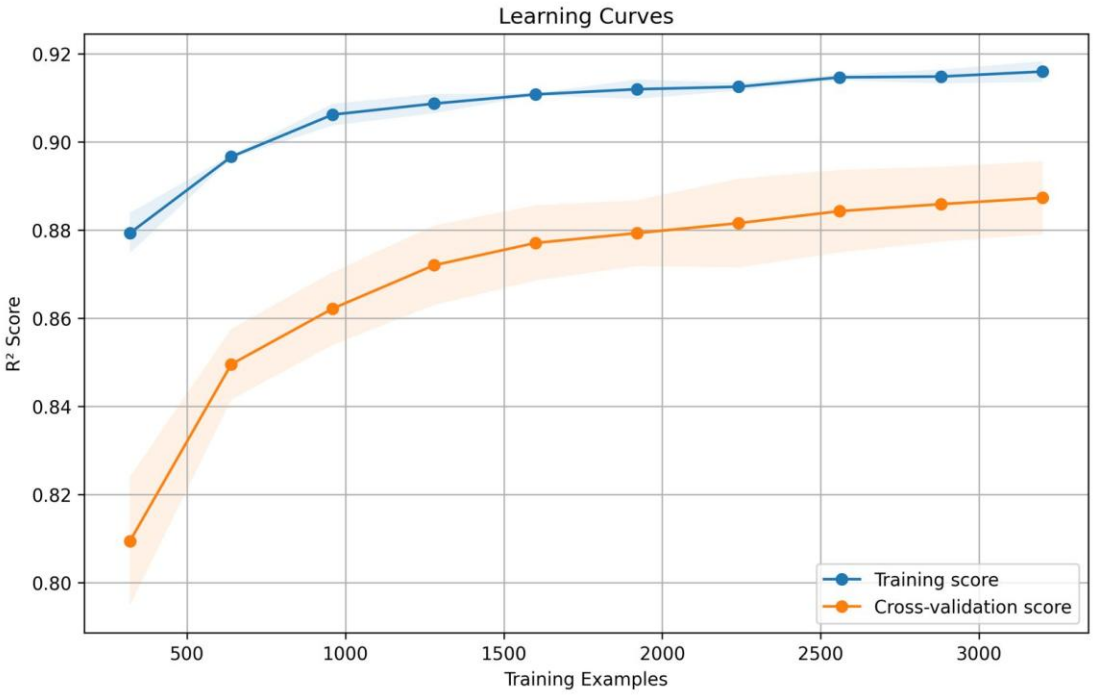


Figure 28: AI Model Training Matrix

- Next Step main source code repository
  https://github.com/AdithyaSean/Next-Step
- Next Step prediction model training
  https://github.com/AdithyaSean/next-step-ai
- Next Step frontend – microservice
  https://github.com/AdithyaSean/next-step-flutter (github.com)
- Next Step users – microservice
  https://github.com/AdithyaSean/next-step-users (github.com)
- Next Step recommendation – microservice
  https://github.com/AdithyaSean/next-step-recommendations (github.com)
- Next Step education – microservice
  https://github.com/AdithyaSean/next-step-education (github.com)
- Next Step gateway
  https://github.com/AdithyaSean/next-step-gateway (github.com)