INFORMATICS INSTITUTE OF TECHNOLOGY

In Collaboration with

UNIVERSITY OF WESTMINSTER

EduGuide Sri Lanka

A Personalized A/L Stream Recommendation and University Guidance Chatbot

A dissertation by

Mr. Thenuka Dinujaya Ubayasena

w1840002 | 20201025

Supervised by

Mr. Devon Wijesinghe

Submitted in partial fulfilment of the requirements for the BEng (Hons) in Software Engineering degree at the University of Westminster.

April 2025

ABSTRACT

Problem: Sri Lankan students face significant challenges in selecting A/L streams and university programs due to a lack of structured guidance and accessible information. The current educational system prioritizes examination performance over aligning students' personal interests with career goals or offering viable options for those who do not meet Z-score thresholds, are given little to no attention. As a result, students often lack direction in choosing academic paths that suit their abilities and career aspirations, leading to uninformed decisions and missed opportunities.

Methodology: To address this issue, EduGuide Sri Lanka was developed as a personalized recommendation platform that assists students in selecting A/L streams and university programs. The system employs machine learning techniques, such as ensemble models, to predict suitable A/L streams based on a dataset of student preferences, academic performance, and career trends. Additionally, a Natural Language Processing (NLP) powered chatbot provides personalized university and career guidance by analyzing user queries and offering tailored recommendations. The platform is built with React.js for the front end and Node.js for the back end, ensuring a seamless and scalable user experience.

Initial Results: Preliminary evaluations demonstrate that the A/L stream recommendation model achieves an accuracy of 78%, effectively aligning student choices with their academic strengths and career aspirations. The NLP powered chatbot successfully answers user queries, providing insightful guidance on university programs and career options. Usability testing with some students revealed that the platform is intuitive and beneficial for their decision-making process.

Keywords: Machine Learning, Natural Language Processing (NLP), Educational Recommender System, Student Guidance, A/L Stream Selection, University Recommendation, Career Planning, Z-score Based Prediction, Personalized Learning

Subject Descriptors:
Information systems → Decision support systems → Educational recommender systems
Computing methodologies → Machine learning → Applications of machine learning

DECLARATION

I certify that this dissertation and its associated artifacts are entirely my own work. They have not been submitted previously, nor are they under consideration for any other degree program or qualification at another institution. All external sources referenced in this work have been properly cited.

Student Name: Thenuka Dinujaya Ubayasena

Registration Number: w1840002 | 20201025

Thenuka

Signature: …………………….   Date: 15th April 2025

ACKNOWLEDGEMENT

Doing my final year project was a challenging but rewarding experience. During the process, I gained much useful information, and I am sincerely grateful to those individuals who assisted me throughout.

Above all, I wish to convey my most sincere appreciation to my supervisor, Mr. Devon Wijesinghe, for his constant support, tolerance, and helpful suggestions. His professional expertise and guidance have been invaluable in defining my research and facilitating solutions to matters.

I am also thankful to the staff and the Informatics Institute of Technology (IIT) for their advice and for offering me the facilities and information necessary to accomplish this project. My sincere appreciation is extended to my friends and colleagues, who have been a source of encouragement and good discussions during this process.

Lastly, I thank my family, whose unwavering support and encouragement have been my greatest asset. Their belief in me has carried me through the most challenging moments.

This project has been a significant learning experience, and I hope it will prove useful to the practice of educational guidance.

Thank you all from the bottom of my heart,

Thenuka

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Acronym	Description
A/L	Advanced Level
AI	Artificial Intelligence
API	Application Programming Interface
DOM	Document Object Model
GCE	General Certificate of Education
GPA	Grade Point Average
ML	Machine Learning
NLP	Natural Language Processing
O/L	Ordinary Level
OOADM	Object-Oriented Analysis and Design Methodology
OOP	Object-Oriented Programming
RAM	Random Access Memory
SSADM	Structured Systems Analysis and Design Method
UAT	User Acceptance Testing
UGC	University Grant Commission
UI	User Interface

CHAPTER 01: INTRODUCTION

1.1 Chapter Overview

This chapter provides an overview of the EduGuide Sri Lanka platform, which aims to assist students in selecting suitable Advanced Level (A/L) streams and university pathways with the help of Machine Learning (ML) and a Natural Language Processing (NLP) based chatbot. The problem domain is defined in this chapter, explaining the problems Sri Lankan students face in making the right choices regarding A/L stream selection and higher education pathways. Moreover, it describes the technological approach, e.g., the introduction of ML for stream recommendations and NLP-based chatbot support for university guidance. The chapter also highlights the research's significance, e.g., its potential contribution to education and career guidance in Sri Lanka.

1.2 Problem Background / Domain

1.2.1 Lack of Guidance in A/L Stream Selection and Career Pathways

Choosing an A/L stream is an important choice that significantly determines a student's future academic and professional career direction. However, many Sri Lankan students are not exposed to properly organized guidance in choosing their streams, thereby making unprepared decisions that do not align with their interests, abilities, or future career paths (Adithya, 2024). The current education system is more exam-centric and provides students with less tailored guidance to follow future career paths according to their interests and abilities.

Although the Sri Lankan government provides basic recommendations on A/L streams, it fails to provide an interest-based, evidence-based system that considers personal skills and future career advancement (Development, Education and Learning in Sri Lanka, n.d.). Therefore, students are likely to choose streams based on peer pressure, parental desire, or misinformation, which further increases the likelihood of academic disappointment or job mismatches. Moreover, students who fail to enter university often struggle to access alternative education pathways and therefore miss out on opportunities to continue their education at the tertiary and vocational training levels (Fong and Biuk-Aghai, 2009).

Table 3 : G.C.E.(A.L.) Examination - 2023(2024)
Performance of Candidates - All Island

	School Candidates						All Candidates					
	Male		Female		Total		Male		Female		Total	
	No.	%	No.	%	No.	%	No.	%	No.	%	No.	%
No. Applied	115,742		166,017		281,759		142,521		204,881		347,402	
Number Sat	94,047	81.26	135,010	81.32	229,057	81.30	109,808	77.05	159,805	78.00	269,613	77.61
Eligible for University Entrance	54,038	57.46	97,305	72.07	151,343	66.07	61,500	56.01	111,944	70.05	173,444	64.33
Obtained 3A's	3,277	3.48	6,577	4.87	9,854	4.30	3,495	3.18	6,989	4.37	10,484	3.89
Passed in 2 Subjects	15,555	16.54	17,352	12.85	32,907	14.37	18,066	16.45	21,409	13.40	39,475	14.64
Passed in 1 Subject	12,448	13.24	10,653	7.89	23,101	10.09	14,747	13.43	13,805	8.64	28,552	10.59
Failed in All subjects	11,958	12.71	9,612	7.12	21,570	9.42	15,419	14.04	12,551	7.85	27,970	10.37

Percentages of the "Number Sat" are calculated according to the "Number Applied" in corresponding category
Percentages of "Eligible for University Entrance", "Obtained 3A's", "Passed in 2 Subjects", "Passed in 1 Subject", "Failed in All Subjects" are calculated according to the "Number Sat" in corresponding category

Figure 1: Performance of candidates 2023 – A/L Examination

Recent statistics highlight a notable percentage of students who have failed all three subjects. This shows the need for proper guidance when selecting a stream for Advanced Level studies. Therefore, considering the failures, it is good to provide better guidance for them to select their careers or higher studies according to their interests. The aim of this project is to reduce the number of students failing their A/L exams by developing a comprehensive support system (Aditya, 2024).

1.2.2 Challenges in University Admissions and Alternative Education Pathways

University admissions in Sri Lanka are largely based on Z-scores, which are calculated to determine admission to government universities. However, many students are uncertain about their options if they do not achieve the Z-scores for their chosen courses (Guide to Job Classifications and Position Management, 2023). Although there are private universities and other academic courses available, students are not aware of these options and struggle to find information about course availability, financial requirements, and career opportunities.

In addition, while specialized skills are increasingly sought after in emerging fields, non-traditional educational pathways often lack a readily available model to guide students who wish to pursue professional or non-traditional careers (STRATEGIES FOR ATTRACTING AND SUPPORTING NON- TRADITIONAL STUDENTS, n.d.). Lack of information equates to limited career mobility and missed opportunities in high-growth fields such as business, healthcare, and technology.

1.2.3 A Comprehensive Guidance System

To overcome them, this project EduGuide Sri Lanka proposes:

- **Advanced Level Stream Recommendation System:** Uses machine learning algorithms to suggest suitable Advanced Level streams based on student's interests, academic performance, and career aspirations.
- **University Guidance Chat:** Uses natural language processing (NLP) to provide personalized university suggestions, alternative study streams, and career guidance in an attractive and intuitive interface.

Using ML and NLP technologies, the proposed system will bridge the gap between education and career planning, enabling students to make critical career decisions guided by data, with personalized and contextual advice during decision-making situations.

1.3 Problem Definition

This study aims to bridge the gap in a comprehensive and user-friendly platform that guides Sri Lankan students in choosing their Advanced Level streams, suitable foundation courses, and eligible universities based on their interests and Advanced Level examination (AL) results (Fernando, Geethamali, & Kularathna, 2023). Currently, students and the public struggle to access well-structured, high-quality information that enables them to make good decisions for their studies and careers.

Sri Lankan students go through a significant learning transition when choosing their advanced-level streams, vocational courses, and university studies. However, the traditional school system is largely academically oriented and provides very little personalized feedback for long-term study and employment plans (Dhiman, kaur and Kumari, 2023). Therefore, due to poor career counseling and the absence of a properly designed career guidance structure, most students make decisions without guidance, leading to dissatisfaction, academic problems, and wasted career paths (Adithya, 2024).

Furthermore, university admission in Sri Lanka is largely based on Z-scores, which are used to gain admission to government universities (Fernando, Geethamali, & Kularathna, 2023). Students who fail to achieve the stated Z-score range are left uncertain about other options for higher education, such as private universities, foundation courses, or vocational training (Fong and Biuk-Aghai, 2009). Lack of awareness of these options leads to confusion and limited career mobility.

The existing educational guidance materials In Sri Lanka are often generic, and unstructured, and do not consider students' interests, abilities, and past academic performance when recommending A/L streams or courses in universities (Algolia.com, 2024). To date, there is no integrated, data-driven system that provides integrated and tailored recommendations, leading to suboptimal student decision-making.

1.3.1 Problem Statement

Despite the availability of extensive career guidance information, Sri Lankan students are unable to make rational decisions on A/L stream selection, university selection, and career paths. Current procedures cannot provide personalized data-driven guidance using students' academic results, interests, and career aspirations. Furthermore, the overreliance on Z-score-based university admissions leaves many students feeling like they have no other options, leading to confusion and missed opportunities for tertiary education.

This research aims to develop a machine learning-based recommendation system to provide students with personalized guidance on A/L subject selection, career choices, and university admissions. Using data-informed decision-making, the system is expected to reduce poor academic choices and improve student achievement by providing contextual, personalized suggestions based on student performance and interests.

1.4 Research Motivation

The purpose of this research is to provide Sri Lankan students with a data-driven, user-friendly platform to make intelligent decisions regarding higher-level subject selection, university courses, and career streams. Students often lose out due to a lack of structured guidance, peer pressure, or guesswork, and make decisions that are not aligned with their strengths or needs. With Sri Lanka's competitive education market, limited government university seats, and limited knowledge of alternative pathways such as vocational training or private universities, students themselves have considerable uncertainty about their future. As a Sri Lankan student, I can speak directly to these challenges, and I believe that NLP and machine learning (ML) can bridge this gap by providing personalized recommendations based on students' aspirations, academic performance, and career goals. The aim of this research is to develop EduGuide Sri Lanka, an artificial intelligence-based platform that provides higher-level syllabus, career guidance, and university admissions support. It also helps to make data-driven choices and reduces unnecessary academic stress and competition for limited university places.

1.5 Research Gap

1.5.1 Lack of personalized A/L stream selection advice

Most of the modern educational guidance frameworks in Sri Lanka provide general advice for selecting A/L streams, regardless of student interest, skill sets, and future aspirations. The existing advice is static, old-fashioned, and often based on external expectations rather than fact-based observations (wijesingha, 2021). Thus, the lack of personalization leads to uncertain choices by students, which in turn leads to academic dissatisfaction and career mismatches.

1.5.2 Lack of adequate understanding of alternatives to formal schooling

University admissions in Sri Lanka are also highly dependent on Z-scores, and those who do not achieve the appropriate levels have problems finding viable alternatives. Private universities, foundation courses, and vocational training schemes exist, but many candidates are unaware of or are not well-guided to explore these options. There is no holistic solution that includes alternative streams as part of the selection mechanism on current platforms (Colombo Telegraph, 2021).

1.5.3 Lack of AI-based, data-driven career and university recommendations

Most career guidance products in Sri Lanka do not use AI or machine learning to provide adaptive, data-driven suggestions. Current processes do not incorporate predictive analytics, failing to suggest career paths that match students' aptitude, performance and future job market needs (Sachintha, 2024). The lack of intelligent career path mapping leaves students uncertain about the future.

1.5.4 Disorganized and unorganized career guidance materials

Details on higher education streams, universities, and career choices are scattered across various government websites, university websites, and independent resources, making it difficult for students to get all the relevant information in one place (Print Edition – The Sunday Times, Sri

Lanka, n.d.). There is no single point that provides a one-stop solution to assist students in their education and career choice process without any hassle.

1.5.5 Rationale for the Research Gap

Students are struggling with educational and career decisions that impact their careers and lives due to the absence of a centralized, AI-based, and personalized counseling system. To fill these gaps, this research will introduce EduGuide Sri Lanka, an AI-based system that combines machine learning, natural language processing (NLP), and data-driven recommendations to help students choose higher education streams, identify career paths, and make informed university choices in a structured and user-friendly manner.

1.6 Contribution to the Body of Knowledge

1.6.1 Contribution to the Research Domain

This study contributes to the body of research by proposing a systematic framework that integrates academic performance indicators (e.g., Z-scores and subject strengths) with occupation-specific job market data, salary trends, and career training courses.

Traditional recommendation systems in education have largely ignored this holistic integration, typically addressing career or academic recommendations independently (Fong & Buik-Aghai, 2009). The research will provide meaningful insights into how AI-driven, customized educational guidance systems can bridge the gap between academic choices and genuine career opportunities. Furthermore, this study will contribute to the new field of educational data science by demonstrating that machine learning and natural language processing (NLP) can be used to improve student decision-making, reduce dropout rates, and enhance long-term career success (Algolia.com, 2024).

1.6.2 Contribution to the Problem Domains

EduGuide Sri Lanka portal will fill the lack of formal and personalized guidance among Sri Lankan school students in their educational decisions. By applying machine learning-based recommendations, the website will provide students with fact-based advice on A/L stream selection, foundation courses, and university choices based on their area of interest, strengths in the field of study, and career aspirations. This will reduce decision uncertainty, thereby allowing students to make interest-paired, fully informed decisions and, in doing so, improve their academic success and employability. In addition, the platform provides real-time job market trend analysis, allowing students to assess the long-term career implications of their own educational choices (Guide to Job Classifications and Position Management, 2023).

1.7 Research Challenge

1.7.1 Data Integration

Among the most important issues is the collection and aggregation of data from various sources such as public and private universities, vocational training institutes, and government schools. The sources provide data in different forms and structures, which are difficult to aggregate, sanitize, and generalize for use in recommended systems. In addition, there is limited access to up-to-date and reliable data, and extensive verification is required to ensure its accuracy.

1.7.2 Personalization

Another major challenge is to develop a highly personalized A/L stream and university recommendation system. The system must process many parameters such as students' interests, grades, and future career aspirations to provide personalized suggestions. Creating an effective machine learning model that can appropriately balance these parameters and generate useful recommendations is a sensitive task that requires careful feature engineering and model calibration.

1.7.3 User Engagement

Ensuring that students actively engage with the platform is critical to its success. The system should present recommendations simply and intuitively, encouraging students to explore different academic and career options. Poor UI/UX design, lack of interactive features, or information overload can discourage students from using the platform effectively. Therefore, striking a balance between complex guidance and an easy-to-use interface is one of the biggest challenges.

1.8 Research Questions

RQ1 – In what ways can an AI-driven digital platform assist Sri Lankan students in selecting A/L streams that match their unique interests, academic abilities, and career aspirations?

RQ2 – What are the key functionalities that should be included in a digital guidance portal to provide personalized and career-oriented recommendations for A/L streams, foundation courses, and university courses?

RQ3 – How can Z-score-based recommendations for government universities be meaningfully integrated to provide alternative pathways through private universities and vocational training for students who score below the cut-off scores?

RQ4 – How can motivational content such as success stories and expert advice support student engagement and decision-making within the educational guidance portal?

RQ5 – What are the key technical and user experience issues in developing a data-driven, personalized career guidance system for Sri Lankan students?

1.9 Research Aim

This research aims to design, develop, and evaluate a student-university linking portal that guides Sri Lankan students in selecting the most suitable A/L streams, foundation courses, and universities based on their interests, academic strengths, and career aspirations.

1.10 Research Objectives

Objectives	**Research Objective**	**Description**	**LOs Mapped**	**RQ Mapped**
Problem Identification	Identify the difficulties faced by Sri Lankan students in making well-informed decisions regarding A/L subject streams, foundation courses and university choices.	Examine existing problems in educational decision-making, e.g. lack of adequate guidance and lack of access to systematic career information.	LO1	RQ1, RQ5
Literature Review	Examine existing educational guidance	Critically evaluate existing tools and best practices in career and	LO2	RQ2

	systems at the national and global levels.	academic guidance to identify areas where the new system can be improved.		
Requirement Elicitation	Gather user needs and expectations for an effective student guidance system.	Conduct surveys and interviews with students, teachers and career counselors to determine key system features.	LO1, LO3	RQ2, RQ3
System Design	Develop a personalized recommendation system with A/L subject stream selection, career guidance and university recommendations.	Develop a system that uses machine learning to correlate student interest, strengths and academic performance with educational pathways.	LO2, LO4	RQ1, RQ2, RQ3
Implementation	Develop a data-driven recommendation system with Z-score-based university admissions and alternative pathways.	Design the system with features that suggest private universities and foundation courses if Z-score-based university admissions are not possible.	LO4, LO5	RQ3, RQ4
Testing & Evaluation	To evaluate the usability, accuracy, and efficiency of the platform.	Conduct user testing among students and instructors to test recommendation accuracy, user engagement, and system efficiency.	LO3, LO4, LO5	RQ4, RQ5

User Engagement & Motivation	To add student success stories and career mentoring to the platform.	Increase user motivation by providing real-life examples of students who made good educational choices.	L05, L06	RQ4
Documentation	To document the design, development, and evaluation process of the platform.	To prepare a comprehensive report on system development, testing, and results to ensure transparency and reproducibility.	LO6	RQ1, RQ2, RQ3
Publication & Dissemination	To share research findings and system design with academic and educational communities.	Publish findings in research journals or conferences and sell the platform to schools and universities.	LO7	RQ5

Table 1: Research Objectives

This structured list ensures that each objective aligns with the module's learning outcomes and research questions so that they are specific, measurable, achievable, relevant, and time-bound (SMART).

1.11 Chapter Summary

The current research study focuses on the design and evaluation of a student-university bridge portal to guide Sri Lankan students in making informed decisions regarding A/L stream selection, foundation courses, and university selection. Based on the application of machine learning and data-driven recommendations, the website will attempt to identify appropriate educational streams that match students' interests, academic strengths, and career aspirations. This study addresses existing educational guidance gaps by providing an integrated digital solution that enhances accessibility, personalization, and engagement in decision-making.

CHAPTER 02: LITERATURE REVIEW

2.1 Chapter Overview

This chapter presents a comprehensive literature review that discusses current studies, technological advances, and methodologies relevant to developing an A/L stream and university guidance system. The chapter is structured to explore significant issues that students face in selecting appropriate study paths, discussing relevant works and technological developments, and examine the performance and limitations of current systems. It also addresses gaps in the literature and how this project adds value to the discipline. The chapter concludes with a benchmarking and evaluation section, where a critical review of how the literature covered informs the design and development of the current system is provided.

2.2 Concept Map

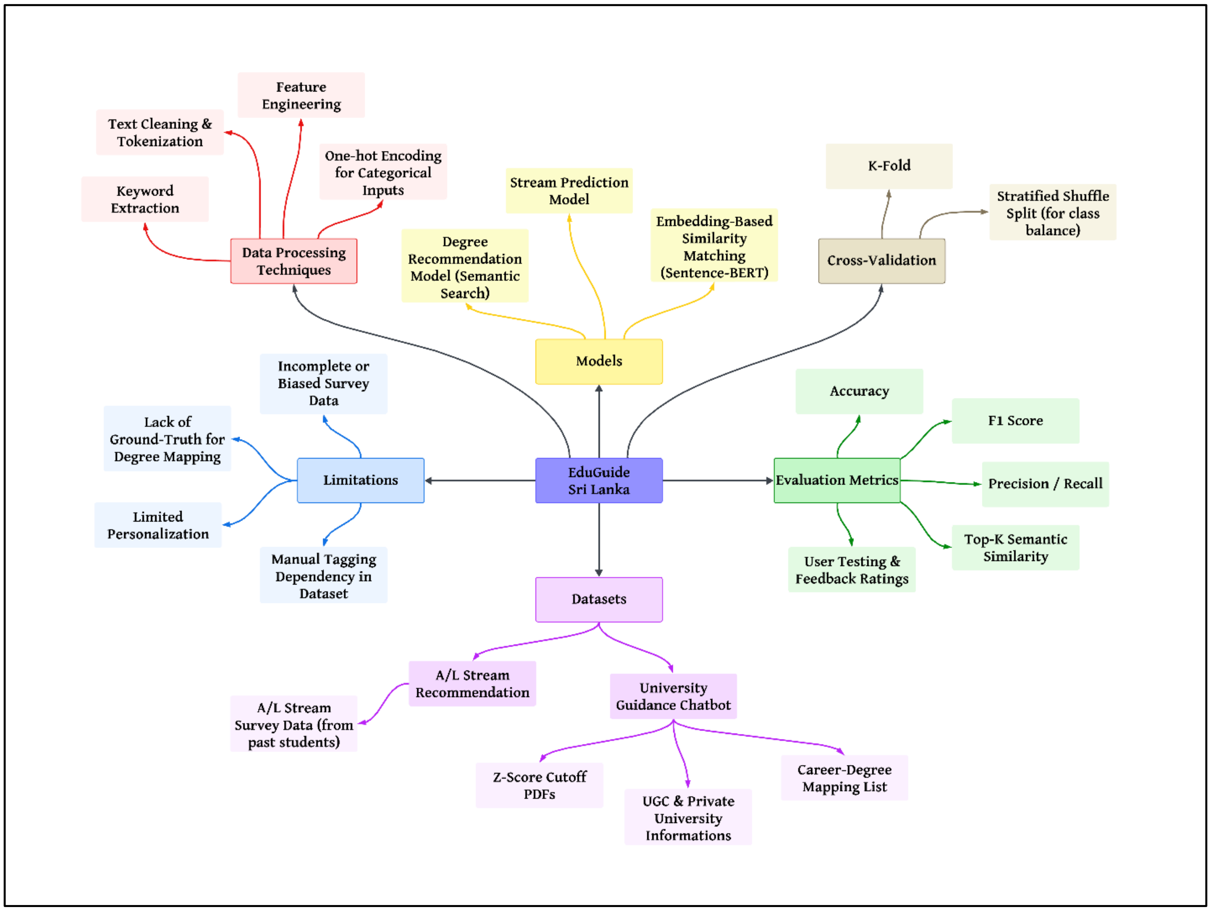


Figure 2: Concept Map (self-composed)

2.3 Problem Domain

2.3.1 Introduction to Recommendation Systems and Educational Recommendation Systems

Recommendation Systems (RS) are intelligent decision-support systems widely used in various industries and suggest options to users that best suit their needs and preferences. These systems use user data, preferences, behavior, and contextual data to make personalized recommendations (Shetty, 2019). Traditionally, they have been widely employed in e-commerce, streaming platforms, and social media sites. Recently, their adoption in the education sector has gained momentum (Dhahri and Khribi, n.d.).

Educational Recommendation Systems (ERS) are a subset of RS specifically tailored for learning settings. They suggest learning materials, courses, educational paths, and even peers. ERS has evolved with the increasing need to personalize education to suit the abilities, goals, and limitations of each learner. In the Sri Lankan education system, students must decide on an A/L stream (such as physical science, bioscience, commerce, arts, or technology) at a critical stage, and ERS can mitigate the risks of adverse choice (da Silva et al., 2022). Misguide choice can lead to low academic achievement, dissatisfaction, or limited access to education opportunities.

2.3.2 The Importance of Early Academic Stream Selection

The Sri Lankan General Certificate of Education Advanced Level (GCE A/L) is an important milestone that determines the future higher education and career paths of the students. Once a student has decided on a stream (Rusika, 2019), it usually limits or opens doors to certain university studies and careers. Early academic decision-making can close or open windows to future opportunities, especially in terms of selective university admissions.

Compared to more flexible education systems where course modules can be adjusted along the way, the Sri Lankan A/L system locks students into a stream for two years. The rigidity of this system creates a high risk of selection bias. Often, students are not aware of making decisions based on peer pressure, parental pressure, or the reputation of conceptual streams, rather than being consistent with their needs or abilities (Sabani, Jamaldeen and Muhammed, 2019). A recommendation system that considers personal interests, educational background, and career aspirations can help students make better stream choices.

2.3.3 Challenges in Choosing an Appropriate A/L Stream and University Program in Sri Lanka

Choosing an appropriate Advanced Level stream and a compatible university course in Sri Lanka is fraught with many interrelated difficulties. One of the most significant barriers is information asymmetry for students, especially in under-resourced or rural schools who lack access to accurate, comprehensive, and timely information on stream implications, university programs, career paths, and entry requirements (Jayasinghe et al., 2021). The lack of such information prevents students from making an informed decision at a crucial stage in their academic lives.

This problem is compounded by the lack of systematic and personalized career guidance services. Although the Ministry of Education has endorsed the importance of career counseling in schools, its implementation across the country is inconsistent. Many schools operate without professional staff or use teachers with little or non-professional training to provide guidance and mentoring (Pereira and Pereira, 2020). Even in schools where they exist, they are often generic and offered temporarily, and are not adequately tailored to the specific needs, abilities, and long-term aspirations of students (Fernando et al., 2019; Rajapaksha, 2021). This lack of personalization often makes guidance ineffective or even misleading.

In addition, sociocultural variables are strongly influenced when decision-making. In many cases, students are pressured by parents or teachers to pursue traditional "prestige" fields such as medicine, law, or engineering, regardless of their actual interests or abilities (Gunawardena et al., 2018). Such social pressures, coupled with a lack of quality, personalized guidance, lead to misaligned academic and career decisions that are factors in student dissatisfaction, dropout, and underemployment.

The expansion of the higher education sector in Sri Lanka, particularly through the establishment of private universities, has contributed to the variety of options available to students. While this should improve accessibility, it also introduces another dimension of confusion. Students are now increasingly required to evaluate institutions and programs based on affordability, quality, recognition, industry relevance, and how well they fit into their A/L streams. This is also a challenging task when dealing with unstructured and non-comparable data (Wickramasinghe and Dayaratne, 2020).

Another critical gap is the lack of centralized portals that allow students to compare government and private universities, programs, scholarships, career opportunities, and admission criteria side by side. This complexity demonstrates the critical need for technology-driven solutions such as intelligent recommendation systems and AI-powered chatbots that can provide consistent, personalized, and scalable counseling across the country. A platform like EduGuide Sri Lanka which is developed by using machine learning, natural language processing, and semantic search can address critical gaps left by traditional counseling methods and allow students to make more informed, independent decisions about their academic futures (Hummel, 2024).

2.3.4 Student Preferences Play a Role in Academic Decision-Making

Incorporating student preferences and talents into academic guidance systems has been shown to increase satisfaction levels and measures of success. Research in educational psychology suggests that students perform better in their studies if they take courses that align with their intrinsic personality types and interests (Digital, Rowan and Works, 2013). Systems that incorporate such methods can guide students into streams and university programs that are appropriate not only for their academic abilities but also for their long-term interests and future career goals.

However, in Sri Lanka, there are some experimental attempts at such approaches, with promising results in increasing student satisfaction and reducing dropout rates (Murphy et al., 2013). By adding preference-aware filters to a recommender system, this project aims to fill an important gap in student-centric decision-making tools (Smith, 2023).

2.3.5 Personalization in Chatbot Responses for Educational Domains

Personalization is a critical feature that drives the efficiency and user satisfaction of educational recommendation systems. In chatbot applications, personalization refers to the adaptation of responses according to user-specific attributes such as background knowledge, interests, skills, career goals, and preferred language (DitchThatTxtbk, 2024).

In this project's University Counseling Chatbot, personalization is facilitated through user profiling and dynamic content filtering. By collecting information about the user's A/L stream, Z-score, career interests, and language choice when initiating the conversation, the chatbot can tailor responses and present university degree programs that are most relevant to the user's background and goals. This reduces information overload and improves decision-making for students who may otherwise be overwhelmed by the variety of options available to them (ChatBot Blog, n.d.).

Moreover, user-specific feedback increases the trust and loyalty of the user, especially when the system provides explanations for why it recommends a specific degree or institution (Pratap, 2023).

2.4 Existing Work

2.4.1 Overview of Previous Research

In recent years, an increasing number of research works have used artificial intelligence (AI), machine learning (ML), and natural language processing (NLP) to solve educational guidance and decision-making problems. The central idea behind such efforts is to provide students with personalized academic and career guidance, especially at crucial moments such as choosing a secondary education stream or enrolling in undergraduate programs.

Murtaza et al., (2022) explore the potential of AI-based personalized e-learning systems to enhance education by tailoring content and assessments according to the needs of individual learners. Unlike traditional e-learning, these systems adapt to a **learner's comprehension level** and **preferred learning modes**. The study identifies key requirements, challenges, and future research directions for implementing effective personalized e-learning systems.

Magar et al., (2023) developed a modern career guidance system using machine learning (ML) and natural language processing (NLP) to assist students in identifying appropriate higher education and career paths that match with their knowledge and skills. The system demonstrates the potential of artificial intelligence-based solutions for personalized guidance by processing user inputs and suggestions for relevant career options through **text** and **semantic analysis.**

Fong et al., (2010) propose a hybrid model combining neural networks and decision tree classifiers to predict university admissions. The system analyzes a **student's academic record**, **background**, and **university criteria using historical data**. The prototype, tested with data from Macau secondary schools, achieved high prediction accuracy and flexibility, matching students with suitable universities and advising them on appropriate admission channels. The hybrid model offers advantages such as generalizability and faster performance compared to using a neural network alone. It uses a neural network to select key features from student profiles and then applies the decision tree algorithm to generate admission rules. Experiments showed the hybrid classifier achieved a lower error rate compared to standalone neural networks and methods.

Dhar and Jodder, (2020) focused **prediction of appropriate educational pathways** by exploring students' prior academic performance using machine learning algorithms. The research puts into perspective the importance of data-driven decision-making in higher education, especially for students who lack access to professional career guidance services.

Fong and Biuk-Aghai, (2009) propose an automated university admission recommendation system (RSAU) for secondary school students in Macau. RSAU is based on a **hybrid model that combines neural networks and decision tree classifiers**. RSAU analyzes student data, including academic records and personal background, to predict appropriate university matches and admission strategies. The neural network identifies significant input attributes, and the decision tree generates admission rules. RSAU strives to improve the accuracy of predictions and flexibility of university recommendations and admission strategies in the face of the complex pre-tertiary education system in Macau. RSAU consists of a data analyzer, a classifier, and a visualization module, offering a user-friendly interface for teachers and school administrators.

Sah and Singh, (2022) conducted a comparative study of some machine learning models, namely **Decision Trees (DT)**, **Random Forests (RF)**, **Support Vector Machines (SVM)**, and **AdaBoost**, to predict students' intentions to pursue higher education. Their methodology, implemented in Python, demonstrates the place of algorithmic selection in maintaining the validity of academic guidance.

Munasinghe et al., (2025) present a **data mining-based stream recommendation system for Sri Lankan A/L students** to improve decision-making processes based on academic performance. The system obtains academic records of a sample group of 1,000 students and implements data preprocessing, balancing methods (Resample, SpreadSubSumple, and SMOTE), and classification algorithms (Random Forest, Decision Tree, and Naive Bayes). The study found that the random forest algorithm combined with the resample method achieved the highest accuracy (88.61% - 93.32%) in predicting appropriate subject streams, emphasizing the importance of science and mathematics performance. The algorithm selection, data balancing, and selection of relevant predictors for the recommender system are elaborated in the paper.

Finally, E. Padma et al., 2024 proposes a **ML Career Guidance System** and compared algorithms such as DT, SVM, and XG Boost to analyze student data, including academic records, technical skills, personal interests, and psychometric assessments, to provide personalized and unbiased career recommendations. Their study contributes to the field of career paths recommendation through supervised learning methods.

Although previous research has established a strong foundation for the application of intelligent systems in educational recommendations, particularly in A/L stream selection and localized university guidance, there is relatively little research focused on the Sri Lankan context. Moreover, most systems do not implement both ML-based prediction and conversational interfaces.

This identifies a primary research gap and the novelty of a system such as EduGuide Sri Lanka, which integrates stream recommendations and university guidance using an NLP-based chatbot.

2.4.2 Identification of Gaps in Recent Works

Although research on intelligent academic and career guidance systems is increasing, there are severe limitations and a set of research gaps, especially within the Sri Lankan education system.

Primarily, **most existing systems are designed for international or generic data sets**, making them less effective in local settings where education systems, subject combinations, university entry criteria, and career opportunities vary significantly. For example, studies by Magar et al., (2023), Sah and Singh, (2022), and Dhar and Jodder (2020) propose solutions based on data sets obtained from India or other regions, which are not compatible with the Sri Lankan Advanced Level education system and university admission process.

Second, **most studies focus on course recommendation, stream prediction, or university admissions**, rather than proposing a complete guidance system. Comprehensive platforms that guide students in choosing an A/L stream and at the same time guide them towards appropriate higher education pathways are rare, especially with the Sri Lankan context.

Furthermore, while many studies (e.g., Sah and Singh, 2022; Padma et al., 2024) have implemented machine learning techniques to predict academic pathways, there is a **significant lack of systems that integrate natural language processing (NLP) and conversational interfaces** that aim to provide personalized and engaging guidance. This shortcoming limits the functionality and accessibility of these systems for students who may prefer intuitive, chat-like interactions over traditional form-based approaches.

In addition, **there is no personalization of methods to students' interests, soft skills, or future career aspirations**. Most machine learning models are based solely on academic grades, and do not consider qualitative aspects such as personal aspirations, extracurricular strengths, and career motivations, essential ingredients to facilitate well-informed decision-making.

Another significant shortcoming is the **inadequate number of studies using data directly obtained from Sri Lankan students, either through surveys or institution-based reports**. Only a few studies, represented by Munasinghe et al., (2025), have attempted to use local datasets, which are crucial to ensure the relevance and accuracy of the models.

Finally, **system measurement and validation typically focus on algorithm performance measures (e.g., accuracy and RMSE)**, with minimal attention to practical usability testing, long-term use, or impact on students' decision-making behavior.

These shortcomings demonstrate the novelty and necessity of the proposed system, EduGuide Sri Lanka, which:

- Use both ML and NLP for both dual recommendations (University Pathway + A/L Stream),
- Incorporate university cut-offs based on local datasets and Z-scores,
- Provide a chat interface for better accessibility, and
- Tailor recommendations about individual needs, preferences, and goals.

2.5 Technological Review

2.5.1 Dataset

In machine learning applications, model performance is dictated by the relevance and quality of the datasets. To achieve the objectives of this project, multiple data sources were explored and compared to develop a dataset that would support the dual purposes of advising the A/L stream as well as supporting university guidance.

Unlike existing research, which is mainly based on publicly accessible or domain-based datasets provided through **Kaggle** (Magar et al., 2023) or proprietary institutional systems (Algarni & Sheldon, 2023), the present study used a **data collection mechanism** specifically designed for the

Sri Lankan educational context. A **customized dataset** was developed by conducting an extensive survey among school students across different regions. The survey collected key features including:

- Preferred streams and career aspirations
- Interests
- Performance in O/L subjects
- School type and district
- Awareness of university programs

The large dataset was built to ensure that the **cultural and contextual sensitivities** of Sri Lankan students are captured, which are excluded in international datasets, such as those used by Said et al. (2022) and Jiang et al. (2019).

Z-score cut-off data from the University Grants Commission (UGC) for the last years were tabulated to support the **University Guidance**. These were the lowest admission Z-scores by faculty for various districts and programs. This historical dataset helps to consider geographic and performance-based admission trends.

In addition, keyword links that are related to courses and careers were collected from **government university handbooks** and **private university web portals**, which complemented the chatbot's semantic search functionality. This information was organized to include course titles, admission criteria, known career paths, and related universities.

Although the dataset is **rich in qualitative data**, the limitations are:

- Small sample size for stream prediction currently (growth is underway)
- Reliance on student-provided information, which may be biased or variable
- Lack of proven results due to lack of long-term tracking

Despite these limitations, the dataset is **more locally relevant and adaptable** than any other general education data set available. Multi-source integration and survey design help to build a highly contextual, real-world relevant recommendation system for A/L students in Sri Lanka.

2.5.2 Preprocessing

Preprocessing is a critical step in machine learning processes, allowing the transformation of unprocessed data into a form that can be understood and exploited. In the context of this project, different datasets were used, each of which required unique preprocessing techniques to prepare the data for model training and semantic search.

For the **A/L Stream Recommendation**, raw survey data was collected from school students across a variety of educational backgrounds. This data was often unstructured or inconsistent, requiring a multi-stage preprocessing pipeline:

- **Handling Missing Values**: Missing survey responses were deleted or imputed with mode values for categorical variables (e.g. stream preference) and median values for numeric variables (e.g. academic scores).
- **Keyword matching and extraction**: Open-ended answers provided by students about their interests and career goals were examined using keyword matching techniques. A carefully selected list of subject-specific and career-related keywords was used to categorize the answers and infer potential study paths.
- **Text normalization**: All text, such as student interests, was cleaned using regular expressions. This included removing special characters, underlining, and removing irrelevant patterns.
- **Stopword removal and lemmatization**: To improve the text formatting and readability of the form input, stopwords were removed and lemmatization was applied via the NLTK library to reduce words to their basic form.
- **Categorical Variable Coding**: Categorical variables such as school type, preferred stream, and district were labeled for use in classification models.

For **University Guidance Chatbot**, program information was scraped or manually collected from private university websites and official Z-score cutoff data from the University Grants Commission (UGC). Preprocessing included:

- **Structured and semi-structured HTML/PDF content extraction**: Data related to university programs was extracted using BeautifulSoup as well as PDF parsing libraries and then structured into JSON or CSV formats.

- **Z-score normalization**: Previous years Z-score data was normalized to a standard form by standardizing column headings (e.g., district, course, faculty) and correcting inconsistencies observed across years.
- **Course and skill mapping**: Course titles and descriptions were preprocessed with sentence embedding techniques to power the semantic search model. This included data cleaning, removing excess white space, and vectorization with Sentence-BERT.

In contrast to previous studies that focused on simple imputation and feature encoding (Magar et al., 2023; Zayed et al., 2022; Xu et al., 2021), the current project used **domain-specific preprocessing** methods to deal with a variety of data types, including survey results, scraped university data, and historical government data. The customized preprocessing pipeline ensures that input data for the machine learning-based recommendation system and university guidance chatbot is clean, structured, and semantically meaningful.

2.5.3 Feature Engineering and Feature Extraction/Selection

Both feature engineering and feature selection are critical tasks in creating effective machine learning models, especially in educational recommendation systems, where input data can be user-generated, semi-structured, or user-generated.

In the scope of this project, feature engineering played a critical role in improving the predictive efficiency of the A/L stream recommendation model. Additional features were extracted from the survey dataset to capture the subtle nuances of student input. These are:

- **Keyword frequency scores**, derived from descriptions of student interest and aspiration, are specifically designed to measure the match between user input and predefined academic or career fields.
- **Stream relevance scores** are determined by aligning the keywords found with the subject streams through a specially designed stream-keyword matrix, which enables the model to numerically express the relevance of the streams.
- **Derived academic measures**, e.g., scores reflecting the adherence to mean grades or interests in relevant fields, were created with the aim of improving the input space of the classifier.

In relation to the university guidance component, semantic features were extracted from course descriptions using **Sentence-BERT embeddings**, facilitating a more fine-grained alignment between user-provided competencies and study programs. These vector models encapsulated the semantic correspondence between user submissions and program materials, going beyond mere surface-level keyword comparisons.

For feature selection, a filtering method was used to reduce dimensionality and improve generalizability. Features with low variance were removed, and the most relevant features, career interest scores, stream keywords, and basic demographic information—were retained for the advanced flow model. Then the feature importance was assessed using model-specific semantic methods, such as feature importance scores from random forests and SHAP values from XGBoost models.

This method differs from the research by Zayed, Salman, and Hasasneh, (2022), who applied the chi-squared test for feature selection, and Kumar et al., (2022), who used correlation-based feature selection (CFS) to evaluate correlations between features. In contrast to some research by Jiang, Pardos, and Wei, (2019), which relied on implicit embeddings (e.g., co-enrollment and course rating), this project emphasized explicit and interpretable features extracted from student feedback and survey structures. In addition, the semantic-based features used on the chatbot side mirrored Xu et al., (2021) approach of extracting latent learner and course features using knowledge graphs.

This combined feature engineering and selection pipeline ensured both model robustness and domain relevance and transferred traditional educational recommender principles to a local Sri Lankan context with a hybrid survey, Z-score, and scratch dataset.

2.5.4 Model Selection

Choosing an appropriate model contributes to the development of effective recommended systems. For A/L stream recommendations, some classification algorithms such as logistic regression, random forest, and XGBoost were compared to identify the best model in terms of performance measures such as accuracy and F1-score. The algorithms were chosen because they are interpretable, easy to use, and have been successful in previous research on educational recommenders. Sentence-BERT-based semantic similarity models were chosen for university guidance chatbot because they are particularly good at capturing the contextual meaning of student queries and aligning them with appropriate degree programs. Model selection was informed by a precedent for accuracy, interpretability, and computational simplicity.

2.5.5 Hyperparameter Tuning

Hyperparameter tuning was used to optimize model performance. Regarding the classical ML models used in advanced text streaming recommendation, techniques such as grid search and random search were used to optimize critical parameters such as the number of estimators in the random forest, maximum depth, and learning rate of XGBoost. In semantic search for chatbots, various pre-trained sentence-BERT models (e.g., all-MiniLM and paraphrase-MPNet) were evaluated for embedding quality and response relevance. Tuning was performed to improve prediction accuracy while maintaining generalization to different user inputs.

2.5.6 NLP and Chatbot Integration

Natural Language Processing (NLP) plays a crucial role in enabling chatbots to understand and respond appropriately to questions asked by users. In the case of the university guidance chatbot, **Sentence-BERT** (Reimers & Gurevych, 2019) was used to create semantic embeddings of user input, which were then compared with a manual knowledge base of university courses and career options. Through this method, the chatbot can retrieve the most appropriate responses to the context based on **cosine similarity-based semantic search**.

Preprocessing techniques, including **lowercasing, stopword removal, and keyword extraction**, were used to obtain a well-formatted and clean input. In contrast to traditional keyword-based systems, the use of **Transformer-based language models** allows for a deeper level of understanding of user intent, thereby improving recommendation quality.

The chatbot's back-end functionality is provided by **Flask** and communicates with a minimal front-end using a REST API. The interface ensures that context-sensitive, personalized feedback is presented based on students' interests, expertise, or Z-scores, making the system more interactive and user-friendly than typical drop-down-based tools.

2.6 Evaluation and Benchmarking

The effectiveness of the proposed A/L stream and university guidance system will be measured against a set of key performance indicators that reflect the technical robustness of the system and ease of use. These indicators will determine whether the system effectively enables students to make informed decisions regarding their A/L stream choices and future study streams.

One of the key evaluation measures will be **User Engagement**, which is measured by the number of recommendation paths completed. This includes the successful delivery of A/L subject recommendations as well as university or course advice. Various measures, including session completion rate, frequency of reuse and time spent engaging with key features (e.g. stream recommender or chatbot), will provide an insight into the relevance and usefulness of the platform for the students' needs.

The second key benchmark is **Recommendation Accuracy**. For the A/L stream recommender, this means verifying whether the ML model accurately aligns recommendations with student interest, career aspirations, and previous academic record. For the university guidance facility, recommendation accuracy is verified based on recommendations based on Z-scores and compared to official university enrollment statistics and last year's cut-off scores. Private university course recommendations are also tested by user ratings and how well the recommendations match the skills and interests provided by users.

User satisfaction and ease of use is another key evaluation pillar. The platform interface is piloted for ease of use, clarity of instructions, and accessibility. This is done through feedback from students, usability surveys, and interaction tracking. Key metrics are average time to reach a recommendation, bounce rate, and common navigation paths. Feedback allows us to determine pain points of friction or confusion that can be addressed in subsequent releases.

Finally, the system is continuously tested and iterated against metrics such as **user engagement**, **recommendation validity**, and **usability ratings**. Together, these features keep the system aligned with what students need and really help them make confident and well-informed decisions about both their A/L streams and university education paths.

2.7 Chapter Summary

This chapter presented the existing literature on A/L stream selection and university guidance systems, focusing on key domains such as recommender systems, NLP-based chatbots, data preprocessing, feature engineering, model selection, and evaluation methodologies. It highlighted the approaches, technologies, and limitations of the previous research, while providing the foundation and justification for the proposed ML-based recommendation platform for Sri Lankan students.

CHAPTER 03: METHODOLOGY

3.1 Chapter Overview

This chapter outlines the methodologies applied in the development of the student and university guidance portal. It describes steps involved in identifying the system requirements, design of the system, selecting suitable paradigms for the system and the process of constructing the system, as well as testing the constructed system. The chapter also covers the project management methodologies that have been utilized including defining the project scope and other managing approaches to ensure the portal effectively meets the needs of its targeted audience.

3.2 Research Methodology

The Saunders Research Onion Model was used to determine the research approach for this study. Each of the model layers is described below, along with the reasons for choosing the methodology.

Research Philosophy	For this study, **pragmatism** was chosen as the research philosophy. This philosophy is most appropriate as the proposed solution calls for evidence from both positivism and interpretivism. Since both quantitative and qualitative research will need to be conducted to evaluate the reliability and accuracy of the model, pragmatism offers the required flexibility.
Research Approach	A mixed method was used, which combines both **deductive and inductive** approaches. The deductive approach constructs hypotheses and tests them using collected data. The inductive approach starts with observations and derives a hypothesis from the findings. This combination allows for a well-rounded understanding of the research problem.
Research Strategy	The research strategy involves multiple data collection methods to ensure comprehensive results. The methods used includes **Questionnaires**, **Brainstorming** sessions, **Literature Review**, **Interviews** with industry and field experts, and **Surveys** among students and experts in the field.

Research Choice	A **mixed method** approach was chosen as the research method for the project. The reason is that the project required both qualitative and quantitative analysis.
Time Horizon	The duration of the project is determined by the time horizon. In this project, a **cross-sectional** time horizon has been used. Since the data collection is done over a specific time frame, this approach is appropriate for the study.
Analysis and Data Collection	**Questionnaires** and **interviews** will be used to obtain data for research. Experts in the field will be interviewed to gain sufficient understanding of the subject. A questionnaire will be given to the target group to fill out to gather information and understand the users' perspective. After proper analysis, the collected data will be used to establish the relevant dataset.

Table 2: Research Methodology

3.3 Development Methodology

Among the available development models, the **prototype** development model was chosen for this research project. The reason is that it is an iterative model that allows for continuous user participation and feedback. This allows stakeholders to test the system at various stages so that changes and improvements can be made. The prototype model copes well with change and therefore is an ideal choice to go through the system in iterations to perfect it for the desired purposes.

3.4 Project Management Methodology

The project management approach for this project is **Agile PRINCE2**. This approach combines the responsiveness and flexibility of Agile with the formal governance of PRINCE2. By combining these approaches, the project captures the iterative development and rapid response to change of Agile, while providing proper planning, risk management, and formal decision-making through PRINCE2. This combination allows the project to effectively handle changing requirements while meeting key deliverables within the timelines.

3.4.1 Schedule

3.4.1.1 Gantt Chart

The Gantt Chart is placed under **APPENDIX A**.

3.4.1.2 Deliverables and Dates

Deliverable	Dates
Initial Project Idea and Supervisor Selection - The first step in the project involved selecting a research topic and securing a supervisor for guidance.	1st July 2024
Initial Project Proposal - A proposal outlining the project objectives, scope, and expected outcomes was submitted for approval.	1st August 2024
Literature Review - A comprehensive study of existing research and technologies related to the project was conducted to establish a strong foundation.	2nd September 2024
Software Requirement Specification (SRS) - A document detailing the functional and non-functional requirements of the system was prepared.	3rd October 2024
Project Proposal and Requirement Specification (PPRS)- The refined project proposal and requirement specification.	18th November 2024
Proof of Concept - Initial prototype were presented to validate the feasibility of the concept.	20th December 2024
Interim Progress Demonstration (IPD) - A progress review showcasing the system's development, features, and functionality up to that point.	3rd February 2025
Final Project Report (FPR) and Demo (FPD) - The final report, along with a live demonstration of the fully developed system, was presented for evaluation.	15th April 2025

Table 3: Project Deliverables

3.5 Resources

3.5.1 Hardware Resources

Hardware Requirements	Justification

Intel Core i5/i7 processor	To provide the power needed for processing.
Minimum 12GB RAM	For large datasets management and otherwise.
20GB Disk Space or above	To store datasets, documents and developments of the prototype.
GPU	A powerful and efficient GPU is required to train ML models.
External Hard Drive/Cloud Storage	For backup and version control.

Table 4: Hardware Requirements

3.5.2 Software Resources

Software Requirements	**Justification**
Operating System – Windows 10 or above	To run required development software and handle backend and frontend development.
Google Colab	For training and evaluating machine learning models related to A/L stream recommendations and University guidance.
Python Programming Language	For implementing the backend, machine learning algorithms, and recommendation systems.
ReactJS Framework	For developing an interactive and user-friendly frontend interface.
LocalHost - Flask	To run the application locally. Flask will be used to serve ML models locally.
MongoDB	To manage and store users' data.
Microsoft Office / Google Docs	For preparing reports, documentation, and presentations.
Visual Studio Code / Jupyter Notebook	For ML models, backend and frontend development.
Postman	To test and debug backend API endpoints.
Github	As a version control system for managing the project's source code.
ClickUp	For project management and task tracking.
Zotero	To manage references for literature review and documentation.

Draw.io / Figma / Lucid Chart / Miro	To create diagrams and designs for documentation.

Table 5: Software Requirements

3.5.3 Technical Skills

The following new technical skills were learned during the project:

- Machine Learning: Using ML techniques to apply models for advanced stream selection and university guidance.
- React.js Development: Building an interactive and responsive UI.
- Backend development and API integration.
- UI components for designing our own theme.
- Knowledge of the Sri Lankan education system and university admission process.
- Experience in building user-centric platforms for educational guidance.
- Database Management: Creating and managing relational databases in MongoDB.

3.5.4 Data Requirements

The project relies on organized datasets to make accurate recommendations:

- A/L stream recommendation: Student preference data collected through surveys and school records.
- University and degree guidance: University course descriptions, entry requirements, and career paths.
- Cut-off score dataset: Z-score cut-off scores from the previous years from the Ministry of Education website.

This combination of hardware, software, skills, and data assets ensures the successful development and launch of the platform.

3.6 Risks and Mitigation

Risk	Severity (Low, Medium, High)	Frequency (Rare, Likely, Frequent)	Mitigation Strategy

Low user engagement or adoption	Medium	Likely	Design with feedback from the users right from the onset, conduct user evaluation to the wrong and enhance the use.
Stakeholder disagreement on features	Medium	Rare	Sustain communication and always call stakeholders' meetings to discuss goals.
Technical bugs or errors post-launch	Medium	Frequent	Perform all forms of testing, they need to adopt Bug-Reporting system, and they should form a quick-reaction maintenance crew.
Inability to integrate Z-score-based recommendations accurately	High	Likely	Carry out intensive testing of the integration algorithm of the Z-score and give the available fallbacks.

Table 6: Risks & Mitigation

3.7 Chapter Summary

Specifically, this chapter describes the approaches that were employed to design the portal for linking students and universities: Saunders' Research Onion for collection of data through surveying and the Objectives, Attributes, Actors, Data, and Method for a modular approach to the work. The system uses Object-Oriented Programming with Structured Programming for the management of complicated interfaces and simple procedures correspondingly. The major processes that are included in this context are the data gathering process, recommendation system designing and evaluation process. The project scope focuses on providing personalized educational guidance for Sri Lankan students, ensuring accurate recommendations through machine learning-based predictions and structured decision models.

CHAPTER 04: SOFTWARE REQUIREMENTS SPECIFICATION

4.1 Chapter Overview

In the SRS chapter, various requirements of the system are discussed in predetermined functional and non-functional categories. This chapter describes how the requirements were elicited from stakeholders, as well as their perceptions of the requirements. The social specification known as MoSCoW is used to prioritize the requirements according to the acronyms Must Have, Should Have, Could Have, and Will Not Have. To better illustrate how the system will engage users, this chapter also presents several diagrams, including the Rich Picture Diagram, Stakeholder Onion Model, Context Diagram, and Use Case Diagram.

4.2 Rich Picture Diagram

The Rich Picture Diagram (RPD) demonstrates the setting of the system and the particulars of its interactions with other actors and procedures. It offers an overview of the overall system and specifies the important areas of interaction.

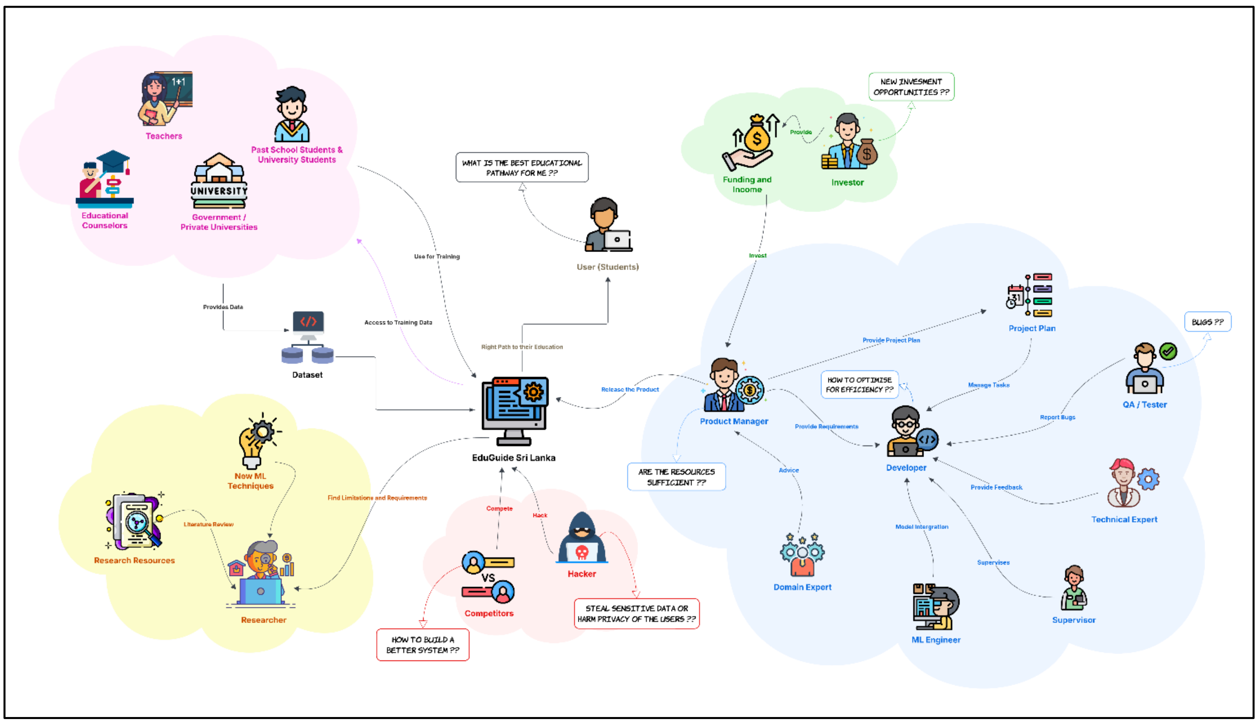


Figure 3: Rich Picture Diagram (self-composed)

4.3 Stakeholder Analysis

4.3.1 Stakeholder Onion Model

Stakeholder Onion Model is a useful tool that depicts stakeholders using rings to show how involved they are in a particular system. Hence the core of the model revolves around system users while other layers encompass other neutral stakeholders in the system. Holding this model makes it easier to understand which of the participants make critical system decisions and which can be considered as external reviewers (Sharp, Finkelstein and Galal, 1999).

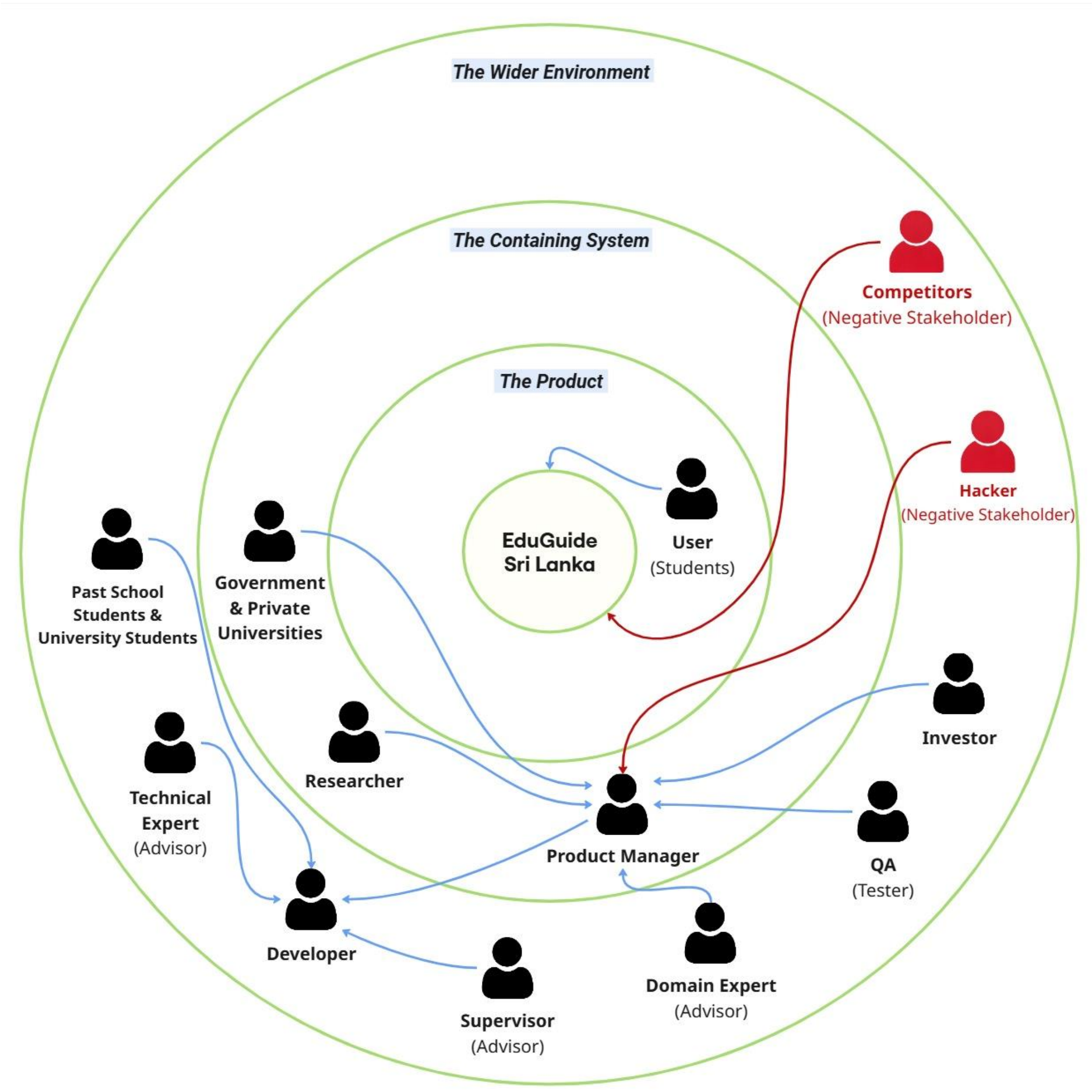


Figure 4: Stakeholder Onion Model (self-composed)

4.3.2 Stakeholder Viewpoints

Stakeholder	Role	Description
System Stakeholders		
User (Sri Lankan Students)	Normal Operator	Need a platform that provides accurate recommendations on A/L Stream, universities and courses.
Containing System Stakeholders		
Product Manager	Operational Beneficiary	Manage the project's overall plan, required goals and features according to users' needs and stakeholder's feedback.
Government & Private Universities	Functional Beneficiary	A user-friendly integration that transparently displays program offerings and enrolls students based on their Z-scores and skill sets is needed to establish trust and engagement. Require a system that accurately reflects their course offerings and entry criteria ensuring they attract qualified students.
Researcher	Functional Beneficiary	Conducts research to identify educational trends, student needs, and recommendations for improvements to the system.
Wider Environment Stakeholders		
Developer	System Developer	Developing the system in accordance with specifications, making sure it satisfies security, scalability, and performance requirements.
QA (Tester)	Software Testers	Testing the usability, performance, security and functionality to make sure the system performs as expected.

Past School Students & University Students	Outer Beneficiary	Contribute to the system through their experiences and expertise on how they chose their educational path.
Supervisor	Academic Guide	Give guidance and feedback on reaching 1st step to last step in the project. Mainly in implementation step.
Technical Expert	System Architect	Provides technical guidance on integrating AI/ML and other technologies.
Domain Expert	Knowledge Contributor	Provides insights into Sri Lankan education policies, university admission processes, and courses guidance.
Investor	Financial Beneficiary	Provides funding for research, development, and deployment of the platform.
Competitors	Negative Stakeholder (Market Rivals)	Competitors try to provide special features or better user experiences to attract students to their educational systems.
Hackers	Negative Stakeholder (Cyber Threat)	Threat to the system's data security, including Z-score data and private student information etc.

Table 7: Stakeholder Viewpoints

4.4 Selection of Requirement Elicitation Methodologies

To ensure that the system can meet the needs of its users and provide accurate recommendations, multiple requirement elicitation methodologies were selected. The selected methodologies were evaluated based on their ability to collect reliable information on students' decision-making, university entry trends, and career guidance requests.

Method 01: Literature Review
Literature review is a crucial activity in understanding the limitations of existing educational guidance systems and the gaps in A/L stream and University systems. At the beginning of this project, an extensive survey of the literature was conducted to explore the existing systems used for student guidance, their effectiveness and their areas of improvement. This assessment process enabled an in-depth examination of current technologies, best practices in recommendation systems and new ideas in career guidance. These findings guided the research direction and ensured that the proposed system addresses the most important issues within the Sri Lankan education system while utilizing current knowledge.
Method 02: Survey
Surveys were used as the main method to obtain quantitative data from students regarding their A/L stream, university choice, and future career decision-making process. Surveys are suitable when collecting information from a broad sample to determine representative diversity in student preferences, concerns, and expectations. Compared to interviews or focus groups, surveys provide statistically analyzable information and are therefore a more scalable and objective method of extraction (Oates, 2006).
Method 03: Document Analysis
To ensure the accuracy and timeliness of the recommendation system, document analysis was used to draw conclusions from official UGC admission trends, private university course designs and Z-score cut-off scores. Although stakeholder interviews are based on subjective responses, document analysis provides verified and authoritative data and reduces the likelihood of introducing bias. This method ensures that the system is calibrated against actual admission criteria and program offerings, making it reliable and accurate.
Method 04: Interview
Insights from experts in the students, field (domain experts), educators and technical specialists are crucial for further refining the system and meeting the needs of students. Interviews with education experts, career counselors, and college admissions staff provided insightful observations about student decision-making processes, the role of Z-score-based university admissions, and non-traditional pathways. In addition, interviews with technical experts helped determine the most effective recommendation algorithms and personalized techniques. The interviews were crucial to validating the research process, finding solutions to the projected

problems, and ensuring that the platform accurately guides students toward well-informed academic and career decisions.

Table 8: Requirement Elicitation Methodologies

Justification for not choosing other methods:

- **Observations:** It is not possible to observe students choosing A/L streams or universities and do not provide detailed information on decision-making factors.
- **Workshops:** Since the project deals with individual students, workshops are not the best way to gather diverse and bulk inputs.

By applying surveys to document analysis for general student feedback and confirmed enrollment details, the requirements elicitation process ensures the depth and accuracy of the system development.

4.5 Findings

4.5.1 Literature Review

Finding	Citation
Personalized recommendation systems increase student satisfaction by pairing educational choices with career goals through AI-based learning models.	(Adil Ellikkal and S. Rajamohan, 2024)
A personalized e-learning system adapts content and tests to individual learners through AI-based methods, improving learning over traditional systems. This paper discusses the key factors, AI advantages, and research gaps in personalized education, and proposes a five-module framework for adaptive learning. It also points out future research directions and challenges in deploying personalized e-learning solutions.	(Murtaza et al., 2022)
Virtual professional networking sites, such as LinkedIn, can provide effective ways for young people to find, retain,	(Johnson et al., 2020)

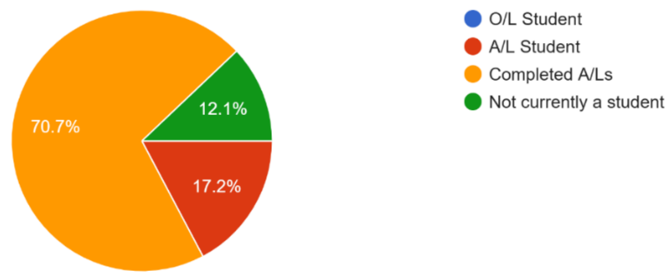
and advance their careers, addressing the shortcomings of traditional job search methods.	
Poor career guidance in developing countries leads to uninformed student decisions, thus creating the need for virtual mentoring sites.	(OECD, 2025)
Research emphasizes the complicated, various types of character of university admissions, wherein other factors beyond academic achievement influence a student's qualifications for higher education. Conventional admission procedures that depend exclusively on standardized results may neglect the unique requirements and potential of individual students.	(Fong and Biuk-Aghai, 2009)
A hybrid recommendation model, exemplified by the RSAU (Recommender System of Admission to University), uses neural networks and decision tree algorithms to improve decision-making accuracy, tackling challenges faced in non-standardized educational environments, such as those present in Macau.	(Fong, Si and Biuk-Aghai, 2010)

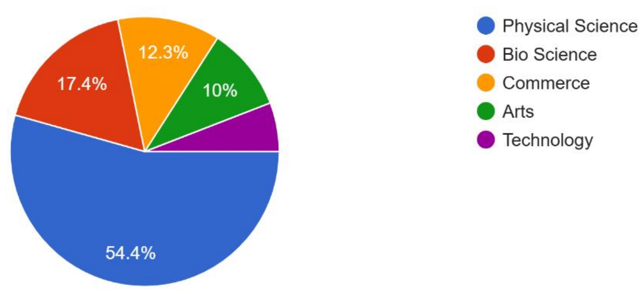
Table 9: Findings from Literature Review

4.5.2 Survey

Questionnaires carried out with the students demonstrated that many of them lack sufficient understanding with regards to the options opened to them after their A/L or O/L and such things as private universities and other diverse study opportunities.

Question	What is your current academic status?
Aim	To check whether the feedback is based on the real experience of the respondent who is currently going through or has passed out from the A/L stream selection process. This ensures that the collected data is relevant and reliable for providing future students.
Findings	

<table>
<tr><td></td><td>


The above statistics indicate that most respondents fall under the completed A/Ls category. 17.2% of respondents are currently students and the other 12.1% are not students. Therefore, feedback is important to understand the common challenges and helpful for the development of better stream and university guidance tool.
</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Question</td><td>Which A/L stream did you choose?</td></tr>
<tr><td>Aim</td><td>To understand the trends when selecting A/L streams and to identify how the stream selection has been distributed among the students. With the use of this data, the guidance tool can be tailored to assist the future students according to their needs and interests.</td></tr>
<tr><td>Findings</td><td>


The above statistics indicate that the majority (54.4%) of respondents' choice is the Physical Science stream. Respondents who selected Bio Stream have a percentage of 17.4. The commerce stream has been selected by 12.3%. With the percentage of 10 Arts selected for their A/L streams and the others have selected Technology. This feedback shows the high tendency towards the science related streams, maybe due to the job opportunities or personal interests. This information
</td></tr>
</table>

	should be considered when developing the education, career and the university guidance system to ensure balanced information and support to every stream.

Table 10: Findings from Questionnaire (Survey)

The other survey findings can be found in APPENDIX B.

4.5.3 Document Analysis

The document analysis involved using up-to-date educational guidelines, university entry requirements and A/L stream selection frameworks provided by the Ministry of Education and other standard sources. By doing this, key functional and non-functional requirements were identified, so that the system is consistent with the realities of academic demands and policy standards.

4.5.4 Interviews

Semi-structured interviews were conducted with educational advisors, teachers and students to gather information on the issues faced in selecting A/L streams and choosing university courses. The interviews provided valuable qualitative information, which helped to identify user expectations and key system features.

Interview results can be viewed here.

The details of interviewees are included in the APPENDIX B.

Participant Role	Key Insight	Suggested Feature
A/L Students	Difficulty choosing stream due to lack of proper guidance.	Stream recommender system
Teachers	Students often follow peer pressure rather than interests.	Interest-based career suggestion
Parents	Wants a platform that explains career paths after A/L clearly.	Clear roadmap from stream to career

Figure 5: Findings from Interviews

4.6 Summary of Findings

The results of the different possibility elicitation techniques affirmed the requirements for an extensive system that covers stream selection, proposed universities, as well as career advice. The platform also requires constant updates in order that it can continue to hold interest among the users.

Findings	Literature Review	Survey	Interview
Many educational platforms to enhance to student's education level.	√		√
Help students to search for universities, courses, and appropriate careers.	√	√	√
Provides information about subject areas and courses.	√	√	
Over 50% of students are not confident that the A/L stream they've selected aligns with their career goals.		√	
The potential job roles after A/Ls are the most important feature preferred by most of the students.	√	√	√
The tool that recommends government universities based on their Z-score has many "YES" responses in the survey.		√	
According to student feedback, multiple students requested to add "Scholarship Finder" to the system.		√	√

Table 11: Summary of Findings

4.7 Context Diagram

A context diagram is a high-level graphical representation of how a system interacts with its environment and external entities. It is a simple diagram used to capture the scope, boundaries, and data flows of the system.

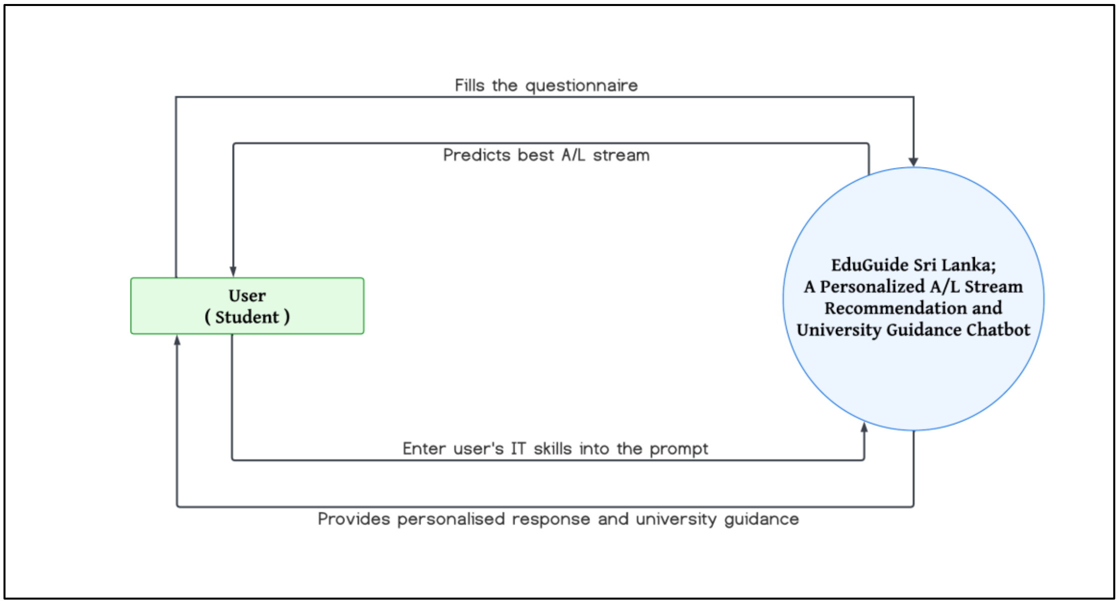


Figure 6: Context Diagram (self-composed)

4.8 Use Case Diagram

A use case diagram, one of the key elements of the Unified Modeling Language (UML), is a graphical representation of how users (actors) use a system to achieve specific goals.

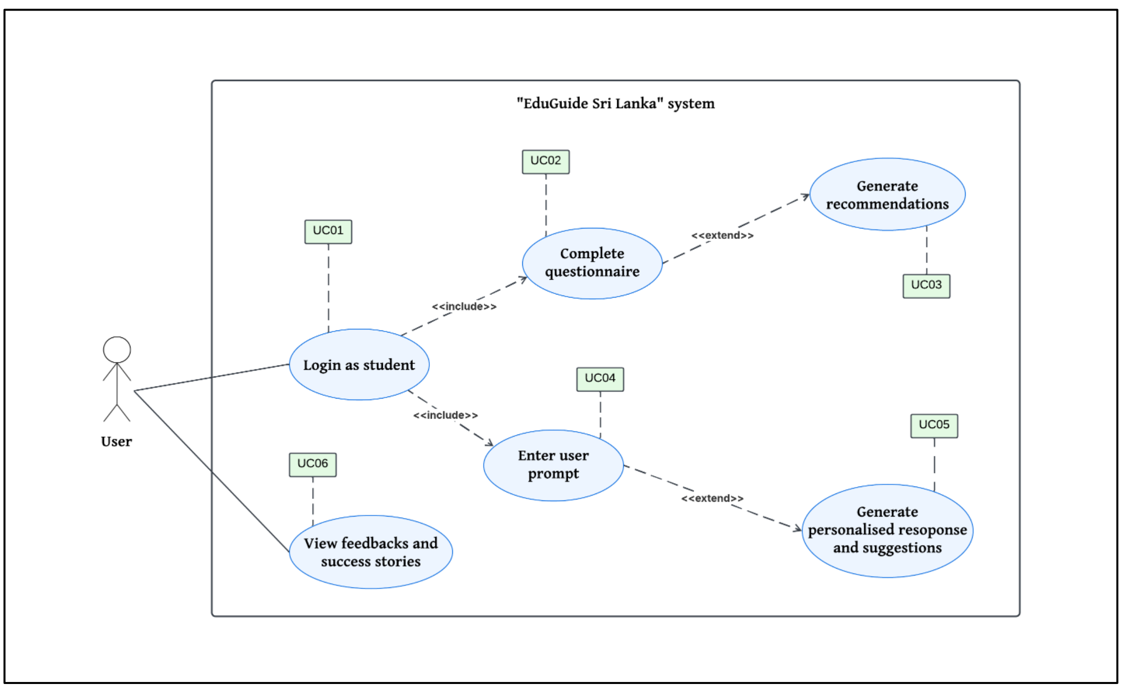


Figure 7: Use Case Diagram (self-composed)

4.9 Use Case Description

A use case description is typically a text document that describes the required functionality of the use case.

Use Case Name	Complete questionnaire	
Use Case ID	UC02	
Description	The student completes a questionnaire to provide information for personalized recommendations.	
Participating Actors	User (Student)	
Pre-Conditions	The user must be logged in.	
Extended Use Cases	Generate recommendations	
Included Use Cases	Login as student	
Main Flow	Actor	System
	1. The user provides responses based on their interests, grades, and preferences.	1. The system presents a series of questions to the user. 3. The system validates the responses. 4. The questionnaire data is saved in the system.
Exceptional Flows	Failure in submitting the questions would result in receiving no recommendation.	
Post Conditions	The generated recommendation results will be demonstrated to the user.	

Table 12: Use Case Description for "Complete Questionnaire"

Use Case Name	Enter user prompt
Use Case ID	UC04
Description	The student can enter a text-based query to get guidance from the chatbot.
Participating Actors	User (Student)
Pre-Conditions	The user must be logged in.

Extended Use Cases	Generate Personalized Response and Suggestions	
Included Use Cases	Login as student	
Main Flow	Actor	System
	1. The user navigates to the chatbot interface. 2. The user enters a prompt (e.g., “Enter the skill area you are interested in (e.g., 'data analytics')”).	3. The system processes the input and retrieves relevant responses.
Exceptional Flows	If the system encounters an error while submitting the prompt, an error message is displayed, and the user is prompted to retry.	
Post Conditions	The system processes the prompt and generates a response.	

Table 13: Use Case Description for "Enter User Prompt"

The remaining Use Case Descriptions are available in APPENDIX B.

4.10 Requirements

The requirements for the "EduGuide Sri Lanka" system are categorized using the MoSCoW principle, which ensures a clear prioritization of features and functionalities.

Priority Level	Description
Must have (M)	Essential requirements that are necessary for the final product (Essential Requirements)
Should have (S)	Essential requirements that make a significant contribution to the final product (Important but not Critical)
Could have (C)	Requirements that can be included if necessary do not make a major difference to the final product (Desirable features)
Won’t have (W)	Requirements that are not considered or will not be included in the final product (Out of Scope)

Table 14: MoSCoW Method

4.10.1 Functional Requirements

FR ID	Requirement	Priority Level	Use Case
FR1	The system must allow students to register and log in securely.	M	UC01
FR2	The system must provide an interactive questionnaire for A/L stream recommendation.	M	UC02
FR3	The system must generate A/L stream recommendations based on user input and historical data.	M	UC03
FR4	The system must allow users to enter queries to the university guidance chatbot.	M	UC04
FR5	The chatbot must generate personalized responses based on the user's prompt and provide suggestions.	M	UC05
FR6	The system must store and manage user data securely.	S	
FR7	The system must allow users to view testimonials and success stories.	S	UC06
FR8	The user could receive an error message if the input data is in the wrong format.	C	

Table 15: Functional Requirements

4.10.2 Non-Functional Requirements

NFR ID	Requirement	Description	Priority Level
NFR1	Performance	The responsiveness of the system - its ability to process information and provide relevant suggestions within a reasonable time frame - is crucial. It must be able to handle large amounts of data without failing the quality of suggestions and system performance.	M
NFR2	Accuracy	The final output of the system should be of high accuracy and reliability.	M

NFR3	Maintainability	Best practices and coding standards should guide the system's design and deploy.	S
NFR4	Scalability	The system must be able to support the development of the platform with consistent performance and efficiency.	C
NFR5	Extensibility	The system should allow developers to easily add or remove system features.	C
NFR6	Usability	Users should be able to operate the EduGuide system and use it easily.	C

Table 16: Non-Functional Requirements

4.11 Chapter Summary

The chapter on the Requirements has discussed the functional and non-functional requirements of the system and roughly how those requirements were collected and prioritized. To explore the overall interaction of the system, a Rich Picture Diagram was used, to identify and understand the stakeholders, a Stakeholder Onion Model was used, to represent the interaction of the business with the system, a Use Case Diagram was used and to Prioritize requirements MoSCoW principle was used. This means while developing the system, special emphasis is to be paid to the “Must Have” and other features that the solution believes should be additionally incorporated may be implemented as extensions in the future.

CHAPTER 05: SOCIAL, LEGAL, ETHICAL AND PROFESSIONAL ISSUES (SLEP)

5.1 Chapter Overview

This chapter discusses the social, legal, ethical, and professional (SLEP) issues faced in the development of the EduGuide Sri Lanka system. The system that provides Sri Lankan students with advanced-level stream and university guidance involves the processing, storage, and presentation of user information, and is required to comply with ethical and legal standards. The BCS Code of Conduct has been reviewed for compliance with professional requirements, and mitigating measures have been implemented to respond to any issues that arise.

5.2 SLEP Issues and Mitigation

5.2.1 Social Issues

Student preferences and academic information are gathered to generate recommendations. In the wrong hands, the data can be misused or biased recommendations can be generated. There is an open data handling policy that informs users about how their data will be used. The system is free from gender, racial, and socio-economic bias.

5.2.2 Legal Issues

Handling student data involves compliance with privacy and data protection laws (e.g. Sri Lanka Personal Data Protection Act). Everything collected as personal data is anonymized whenever possible, and active consent is required for data collection before use. Any data shared from external sources respects copyright and fair use guidelines.

5.2.3 Ethical Issues

The advice provided by the system can have a profound impact on an individual's career choice. Bias in the dataset or algorithm can lead to incorrect advice. Machine learning algorithms are trained on a variety of datasets to minimize bias. Additionally, users receive clear disclaimers that the recommendations should be taken as advice and not as final choices.

5.2.4 Professional Issues

As the system involves interactions with academic professionals and institutions, professional communication and data representation are of utmost importance. The principles of the BCS Code of Conduct, such as integrity, competence, and accountability, are strictly followed. Written consent was required to ensure that professional standards were maintained for the interviews and questionnaires used in data collection.

5.3 Chapter Summary

This chapter examined the social, legal, ethical, and professional issues of the EduGuide Sri Lanka system. The project complies with data privacy legislation, ethical guidelines for AI and proposals for bias, professional codes of conduct to address threats to data privacy, and professional ethics. By ensuring transparency, consent, and fairness, the system provides a trusted and accountable guidance platform for students.

CHAPTER 06: DESIGN

6.1 Chapter Overview

This chapter describes the design phase of the A/L stream, career, and university guidance recommendation system, detailing the transition from conceptualization to system architecture and interface design. It outlines the architectural choices and how the programming approach was chosen for the development of the platform.

Emphasis therefore falls on developing a structured blueprint that will integrate data-driven recommendations, user interactions, and system functionalities flawlessly. The section first provides a high-level system architecture, followed by in-depth design components inclusive of user interface prototypes, system workflows, and essential design diagrams. This design then forms the basis for implementing the system in a scalable, efficient, user-friendly way to set up and deploy the platform.

6.2 Design Goals

The A/L Stream Recommendation and University Guidance system is to be effective, scalable, user-friendly, and a web-based application to provide correct recommendations based on the student's skills, interests, Z-scores, and career goals. The following are the key design objectives in developing the design:

Design Goal	Description
Usability	The primary objective of this project is to assist students in selecting appropriate A/L stream, career, and university options based on their skills and preferences. The system should be user-friendly and simple to use in a way that is easily accessible to students and parents without technical difficulties, as required.
Performance	The system should effectively respond to the predictive performance of the system, which includes multiple recommendation models such as A/L stream selection, career guidance, and university matching. The system should provide quick and accurate recommendations to ensure seamless user experience.

Reliability	The system should be as minimal errors and bugs as possible to maintain the mindset. Careful testing should be done to ensure the accuracy and reliability of the results.
Maintainability	Since updates to datasets, machine learning models, and programming logic, the system should be developed and updated in a way that makes it easy to change over time.
Explainability	The recommendation should be explained so that users can understand why the A/L stream or university courses for which the recommendation is made are recommended. Provides explanations about satisfaction.
Scalability	The system should be able to accommodate many users at a time, such as students, teachers, and parents. The system architecture should be built to accommodate future expansion, such as increasing the recommendation categories or accommodating additional universities.

Table 17: Design Goals

This systematic approach has enabled the creation of a highly flexible recommendation system that fills the knowledge gap in the educational guidance of Sri Lankan students.

6.3 System Architecture Design

6.3.1 Architecture Diagram

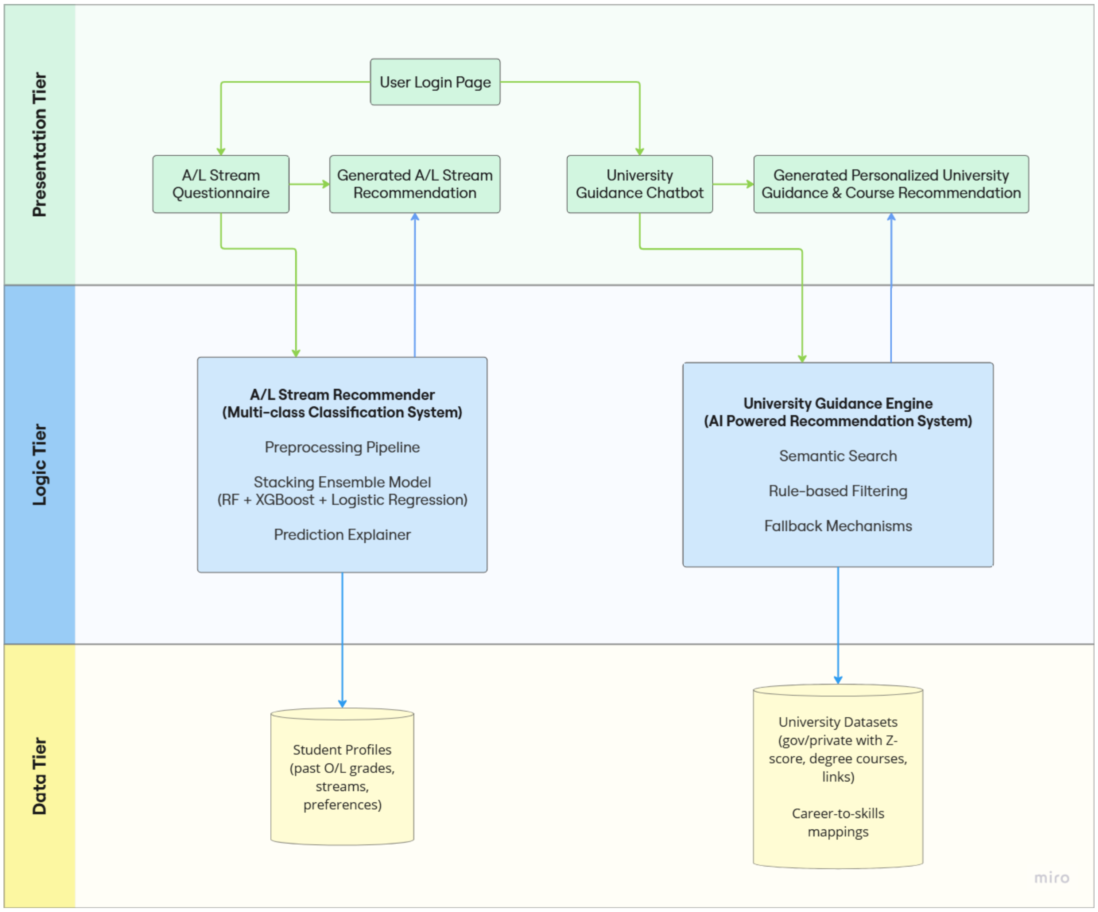


Figure 8: Architecture Diagram

6.3.2 Discussion of Tiers of the Architecture

Data Tier

The core of the system contains all the datasets needed and user information. This will include,

- Student data includes student input such as past O/L grades, streams, and preferences.
- Eligibility is predicted using historical Z-score cutoffs and university admission trends over the past few years.
- The system maps career options to the required skills, helping students understand which university courses align with their abilities and aspirations.

- Machine Learning Training Dataset: This dataset consists of historical university admission trends and student profiles to improve recommendation accuracy.

This tier allows for the easy retrieval of data, training of machine learning models, and the persistence of relevant data for further analysis.

Logic Tier

All inputs are processed by the logic tier, which also verifies them and uses machine-learning models to produce suggestions. There are two main parts to it:

A/L Stream Recommender (Multi-class Classification System):

- Preprocesses student input, handling missing values and formatting inconsistencies.
- Analyzes student performance, interests, and academic preferences using feature engineering.
- The system employs a stacking ensemble model (Random Forest, XGBoost, and Logistic Regression) to enhance prediction accuracy.
- Suggest the best A/L stream and prediction explainer.

University Guidance Engine:

- Utilizes past Z-score cutoffs and program admission trends for university recommendations.
- Recommends universities based on students' scores and preferences using a hybrid approach combining semantic search, rule-based filtering, and fallback mechanisms.

At this tier, the decision-making will be made and checked whether the recommendations are relevant and evidenced.

Presentation Tier

The presentation tier is the top-most layer where the users interact with the system. Its components allow students to input their academic interests, previous O/L grades, and career aspirations.

Users communicate with:

- Students provide their academic background and interests through the A/L Stream Selection Form to receive tailored stream recommendations.

- Students enter their Z-score and preferred career-related skills into the University Guidance Chatbot for personalized courses and university recommendations.
- The system generates personalized A/L streams and university recommendations based on students' academic performance, interests, and career goals.

The presentation tier connects the user and the system by forwarding inputs from the user to the logic tier and presenting results processed by the backend.

6.4 Detailed Design

6.4.1 Choice of Design Paradigm

There are two major design paradigms in software engineering: namely, Object-Oriented Analysis and Design Methodology (OOADM) and the Structured Systems Analysis and Design Method (SSADM). SSADM is the most suitable design paradigm for this research topic.

SSADM was selected because:

- It is centered on a data-driven approach, and it is typical of this project to use structured data such as student academic records, student preferences, Z-scores, and university cut-off scores.
- The use of data flow diagrams (DFDs) and entity-relationship diagrams (ERDs) lends itself to a structured approach and makes it easier to visually understand the flow of data through the system.
- SSADM provides systematic, logical systems analysis and design methodology that is well suited for projects involving complex data transformation and decision-making processes, such as A/Ls stream selection and university recommendation.
- The system requires proper separation of concerns between data collection, processing (reasoning), and output (recommendation), which is inherent in the layered documentation and modeling techniques of SSADM.

Therefore, SSADM is an appropriate and sequential fit for this method of designing the core processes of this recommendation system.

6.4.2 Data Flow Diagrams

6.4.2.1 Level 01 Data Flow Diagram

At Level 01 DFD captures the high-level flow of information processed by the system. This shall include users' subject preferences as well as A/L results, which will be used to generate the suggestions. It extends further by detailing the interactions among the database, the recommendation engine, and the user interface.

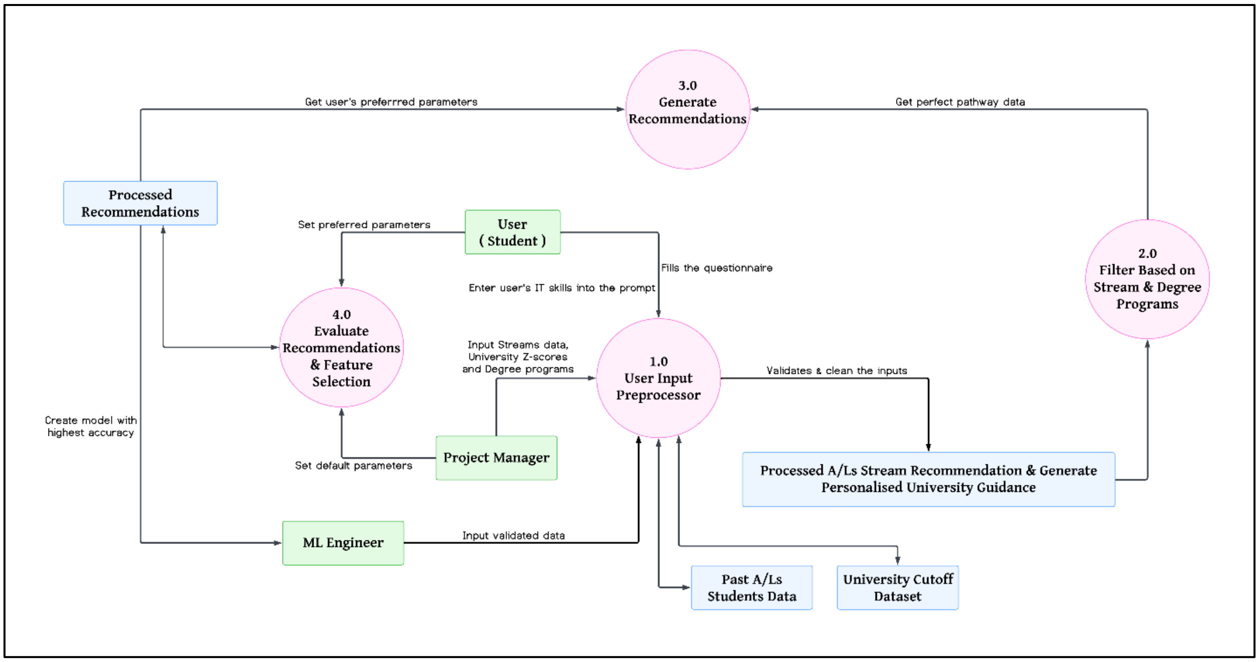


Figure 9: Level 01 Data Flow Diagram (self-composed)

6.4.2.2 Level 02 Data Flow Diagram

Level 02 DFD elaborates on the recommendation process, focusing on data processing done at the logic tier. It includes:

- Procedures for validation to ensure accurate and correct user input presentation for validation of user input data like test scores and subject preferences.
- The recommendation module based on machine learning now processes the user data and, along with the learned models, makes suggestions on universities and A/L streams.
- Database operations include saving student preferences and accessing historical Z-score data.

This systematic deconstruction helps us understand how different elements of the system work together to produce customized recommendations for students.

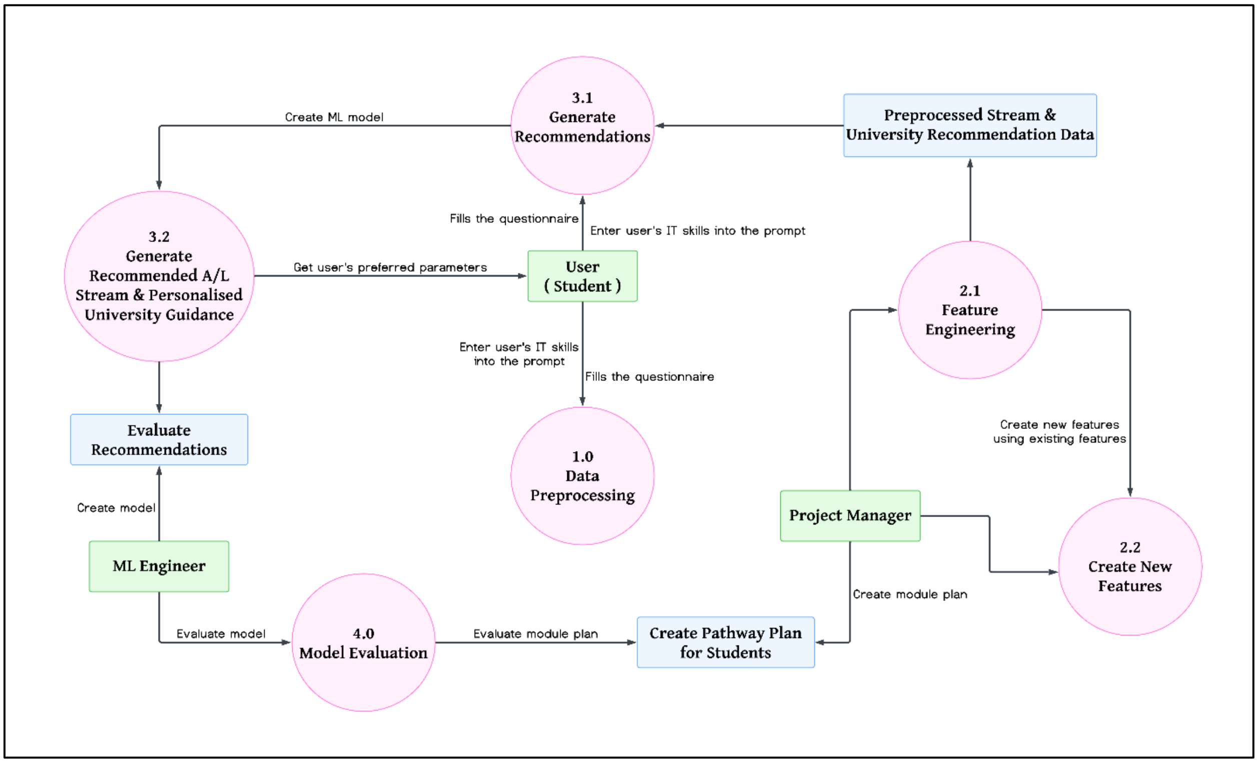


Figure 10: Level 02 Data Flow Diagram (self-composed)

6.5 Algorithm Design

The EduGuide Sri Lanka system is designed to provide data-driven, personalized suggestions for A/L stream selection and university guidance based on the academic performance, individual needs, and historical educational trends of Sri Lankan students. The system uses a hybrid algorithm approach that combines both rule-based reasoning and machine learning models to preserve accuracy, explainability, and scalability.

A/L Stream Recommendation Algorithm

The A/L Stream Recommender uses a Stacking Ensemble model to leverage the strengths of multiple classifiers to improve prediction performance. The following base learners used are:

- Random Forest Classifier
- XGBoost Classifier
- Logistic Regression (Meta-Learner)

This model architecture was used because it has:

- Strong performance for small to medium-sized datasets.
- Ability to learn nonlinear relationships.
- Improved generalization capability through ensemble voting.

The most critical steps:

- Data Preprocessing: Handles missing values, encoded categorical attributes, and normalized student input (A/L results, interests, etc.).
- Feature Engineering: Draws additional features such as subject preference scores, interest vectors, and skill alignment.
- Model Training: Train the base models with historical student data and the provided A/L stream decisions. Logistic regression model as meta-classification in the stack.
- Hyperparameter Tuning: Fine-tune each base model by grid search with cross-validation to achieve maximum accuracy with minimal overfitting.
- Prediction: The most appropriate A/L stream is returned with confidence and explanation (for explainability) via SHAP values.

University Recommendation Algorithm

To guide universities, a hybrid AI-driven recommendation system is used, which combines:

- Semantic search using embeddings (for preference matching)
- Rules-based filtering (based on historical Z-score cutoffs and qualifications)
- Fallback Logic (for returning alternatives on non-specific matches)

The most critical steps:

- Input Parsing: The student provides his Z-score, and preferred IT skills.
- Semantic Matching: The student input is matched with university programs based on vector similarity (via sentence embedding).
- Rules-based Qualification Filtering: Excludes courses based on historical past Z-score cutoffs.
- Final Ranking: Compares semantic matching scores with the qualification scores to rank the programs.

Recommendation Output: Provides the degree course ranking list and the corresponding university name, link, and justification.

6.6 UI Design

The innovative and user-friendly UI design ensures a smooth experience for students seeking university guidance and A/L stream suggestions. Because of the responsiveness and accessibility inherent in the design and simplicity of the design, users can seamlessly input their preferences and obtain their customized recommendations.

6.6.1 Low Fidelity Wireframes

Low-fidelity wireframes provide a very basic view for visualization of the system's user interface.

Figure 11: Low Fidelity Wireframe - Usability and accessibility (self-composed)

Figure 12: Low Fidelity Wireframe - A/L Stream Recommendation Page (self-composed)

The remaining low-fidelity wireframes can be viewed in the APPENDIX.

6.6.2 High Fidelity Prototype

High-fidelity wireframes, which include interactive features, colors and icons, provide a more refined and detailed representation of the user interface using Figma. Because these wireframes closely resemble the finished design, usability and accessibility issues are successfully taken care of before development starts.

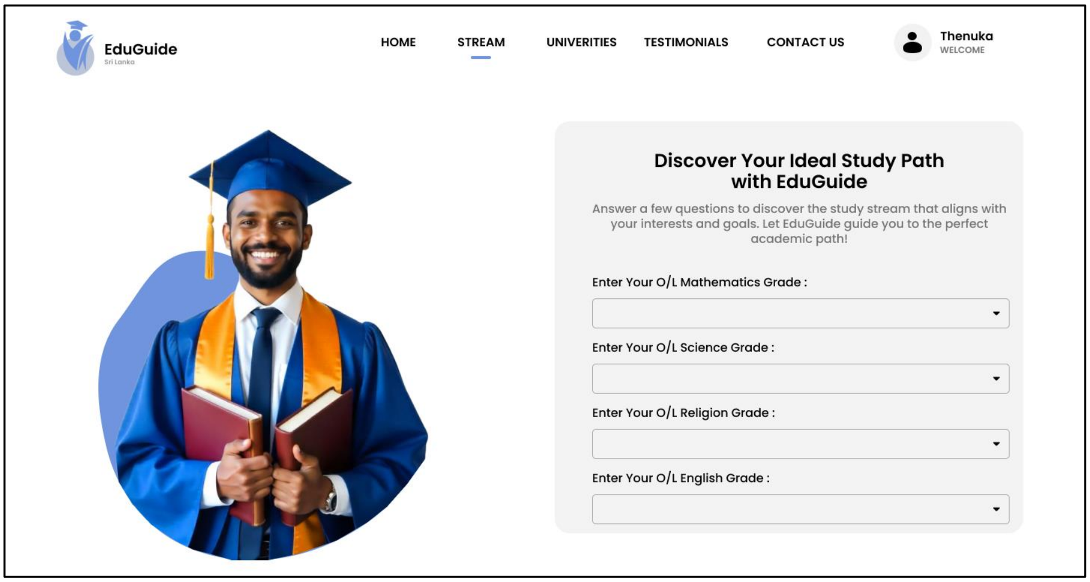


Figure 13: High Fidelity Wireframe - A/L Stream Recommendation Page (self-composed)

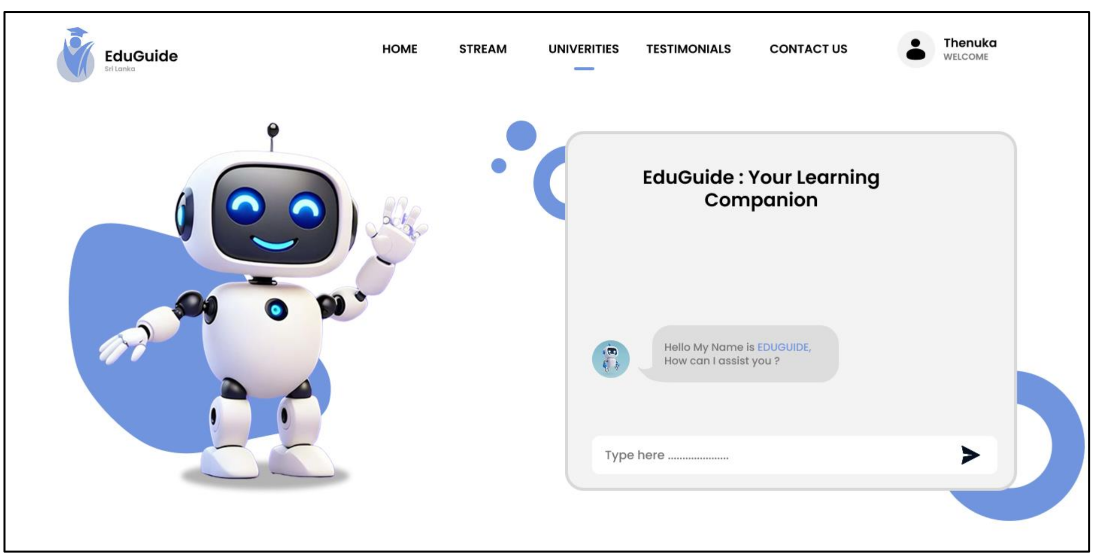


Figure 14: High Fidelity Wireframe - University Guidance Chatbot (self-composed)

The remaining high-fidelity wireframes can be viewed in the APPENDIX C.

6.7 System Process Workflow

The diagram below covers the processing of data within the system and user interactions. It starts when the user feeds his information via the interface: academic information, and personal preferences. The backend is used for the processing of the information where the machine learning models analyze and make recommendations. This program would go on to analyze user preference and academic achievement and suggest the stream selection in A/L accordingly. The university advising module recommends appropriate universities and degree programs to the user based on the user's choices, cutoff scores, and historical university admission data. The finished product is then presented to the user through an easy-to-use interface to ensure a seamless and smooth experience.

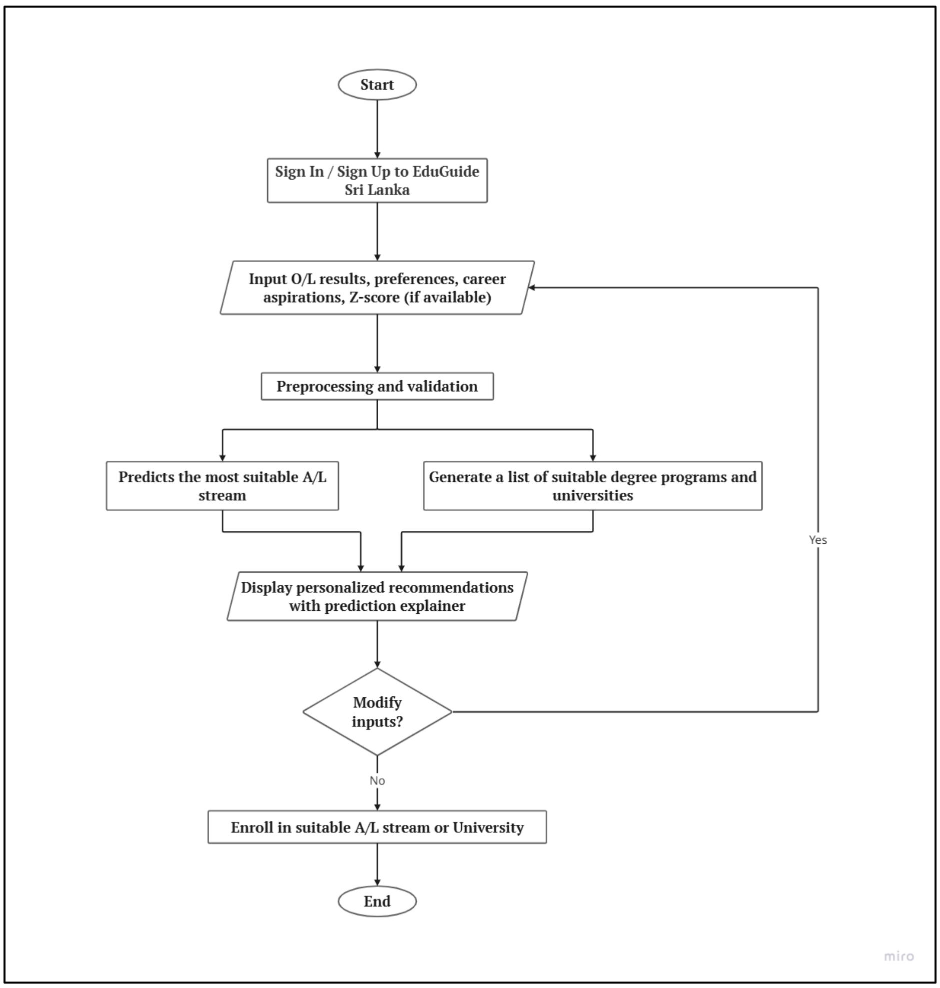


Figure 15: System Process Workflow (self-composed)

6.8 Chapter Summary

This chapter covered key aspects of the system, including algorithm design, user interface wireframes, and system workflows, which gave a comprehensive review of how the system works and how it was designed. The debate highlighted how the different layers of the system work together to process data efficiently. Wireframes also showed the intended user interface, which ensures usability and accessibility. The system workflow diagram outlined the step-by-step process of how data would be input, processed, and then presented as output. The implementation phase, covering the technical aspects of the system and the development process, will be discussed in detail in the next chapter.

CHAPTER 07: IMPLEMENTATION

7.1 Chapter Overview

This chapter outlines the deployment of the EduGuide system, focusing on translating the design into functional components. It touches on setting up the development environment, integrating the front-end and back-end modules, and implementing the machine learning model and chatbot. The section also discusses the adoption of the testing framework used to test the accuracy, usability, and performance of the system.

7.2 Technology Stack

7.2.1 Technology Stack

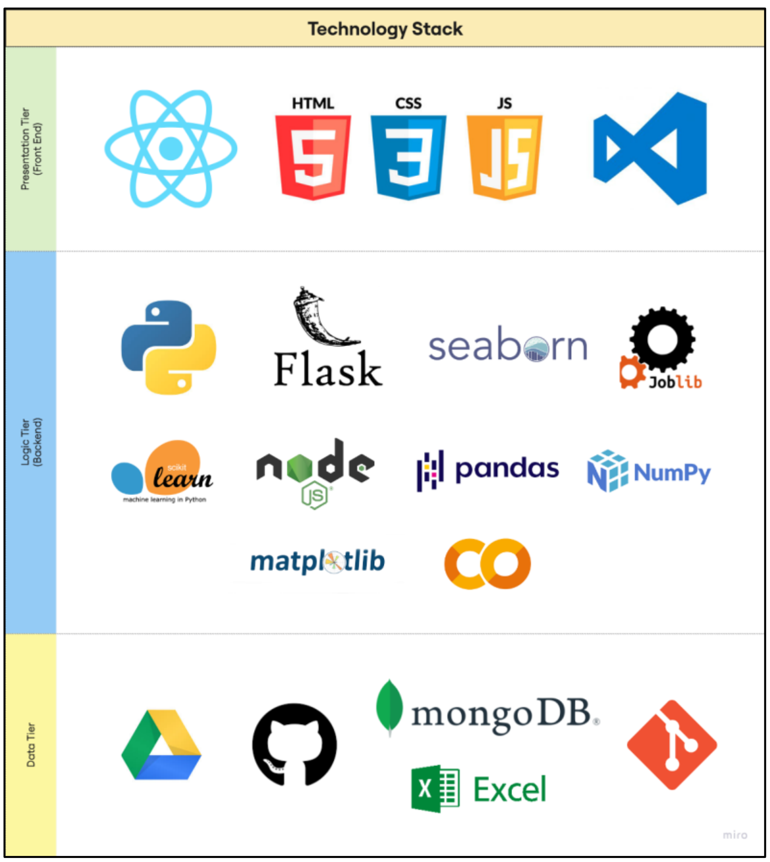


Figure 16: Technology Stack

7.2.2 Dataset Selection

The A/L stream selection component is built on survey data collected from over 500 past school students from various schools in Sri Lanka. The survey elicited responses on their academic backgrounds, fields of interests and post A/L destinations, providing a strong foundation for determining why students choose streams based on performance and interest.

For the Career and Skills Mapping component, a skills dataset was obtained from the publicly available source repository of Mendeley Data (https://data.mendeley.com/datasets/4spj4mbpjr/2). The dataset links various career options with the required skills and work-related knowledge, helping the system map user needs with appropriate academic and professional careers.

The University Guide Recommendation Factor uses a hand-collected dataset that includes both government and private universities. The dataset includes past Z-score cut-off scores (last year) obtained from the Ministry of Education of Sri Lanka and other university-specific data such as degrees awarded, admission criteria, and location.

Where necessary, synthetic data was used to supplement and validate the predictions, especially when real-world data was lacking or unavailable.

7.2.3 Programming Languages

Programming Language	Justification
Node.js	Used for backend development, mainly handles user authentications such as login and registration. Its non-blocking behavior prevents multi-request congestion.
Python	Used in developing machine learning models used in A/L stream and university guidance recommendations. It offers an extensive ML library for effective data processing and predictive analysis.
React (JavaScript)	Used for frontend development, it provides a responsive and interactive user interface to enable seamless communication with backend and ML services.

HTML / CSS	Used in conjunction with React to style and render the web interface, providing a clean, user-friendly and visually pleasing layout.

Table 18: Programming Languages

7.2.4 Libraries

Library	Justification
Pandas	Used for data manipulation, cleaning and preprocessing in the ML pipeline.
NumPy	Enables numerical operations and efficient data manipulation.
Matplotlib	Supports data visualization for understanding trends and distributions in datasets, aiding analysis and reporting.
Scikit-Learn	Scikit-learn is a very popular Python machine-learning package that offers a wide range of classification, regression, and clustering techniques. It is ideal for putting predictive models into practice because of its simplicity and well-documented API. It also provides model evaluation and preprocessing methods that improve the accuracy and dependability of such models.
Seaborn	Based on Matplotlib, it provides beautiful and improved statistical plots.
Joblib	It is used to efficiently save and load trained machine learning models, including efficient serialization of large NumPy arrays and ML pipelines.

Table 19: Libraries

7.2.5 Development Frameworks

Framework	Purpose	Justification
Node.js	Backend development	Node.js is a lightweight and efficient JavaScript runtime for building scalable backend services. It was used to implement user authentication (registration, login) and API integrations with

		non-blocking I/O operations for real-time communication.
React	Frontend development	React is a widely used JavaScript library embracing component-based architecture for maintainable and reusable code. Its Virtual DOM creates optimized rendering for efficiency, hence a fast and responsive user experience. Besides, React makes it easier to create beautiful applications with the allowance of integration of UI frameworks like Material UI and Tailwind CSS.
Flask	ML model integration (Python APIs)	Flask is a lightweight Python web framework that is used to build RESTful APIs rapidly. It is highly adaptable and very easy to integrate with any machine learning model. Its simple architecture and handling of integrated requests make Flask a very good choice for implementing machine learning models as web services.

Table 20: Development Frameworks

7.2.6 IDEs

IDE	Justification
VS Code	Used for front-end (React.js) and back-end (Node.js) development. Its rich extension ecosystem, built-in terminal, and Git integration make it ideal for full-stack web development.
Google Colab	Used for developing, testing, and visualizing machine learning models. It provides GPU acceleration and native Python library integration, making it a perfect choice for rapid prototyping in the ML pipeline.

Table 21: IDEs

7.2.7 Summary of Technology Selection

Component	Tools

Development Frameworks	React, Flask, Node.js
Programming Languages	JavaScript, Python
Libraries	Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn, Joblib
IDEs	VS Code, Google Colab
Database	MongoDB
Version Control	Git, GitHub

Table 22: Summary of Technology Selection

7.3 Implementation of Core Functionalities

7.3.1 A/L Stream Recommendation

7.3.1.1 Data Preparation and Keyword Mapping

Appendix D provides data preparation and keyword mapping with supporting code snippets.

7.3.1.2 A/L Stream Prediction Model (Ensemble ML Approach)

This section explains how to train and build a core machine learning model that predicts the most suitable A/L stream based on a student's O/Ls results and their interests.

O/L Grade Mapping

Subject scores in subjects such as math, science, and history convert letter grades to numerical scores (A=5, B=4, etc.) so that they do not lose their ordinal property. This transformation is essential for training algorithms that require numeric inputs.

```
# ========================== 3. DEFINE GRADE MAPPING ===========================
# O/L Grades often have an *ordinal* relationship: A > B > C > S > W
grade_mapping = {
    'A': 5,
    'B': 4,
    'C': 3,
    'S': 2,
    'W': 1,
}
```

Figure 17: O/L Grade Mapping

This method is beneficial because many machine learning algorithms and computer programs need numerical values to analyze information and determine patterns. With numerical grading, the system can easily analyze students' performances in streams.

Feature Identification

This is one of the essential steps towards preparing the dataset for machine learning. We split the dataset into features (X) and targets (y). Features are input variables that help the model to understand patterns. The target label is the output that we need from the model to predict. When the datasets are split into two sets, 80% of the data is for the training model and the remaining 20% is used to test how the model performs on new and unseen data. The numerical columns (subject grades) and categorical inputs (e.g., career interest, favorite subject) are calculated for transformation in the pipeline.

```
# For classification, we do a stratified split (helps preserve class proportions)
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y)
```

Figure 18: Split into two sets the dataset

Preprocessing Pipelines

This refers to a sequence of operations used to clean and process the data before training a machine learning model. The importance of this process is that the real-world data is always messy, and it may have missing values or it may be in a format that the model cannot read. To overcome this, a modular preprocessing strategy was used. Therefore, within the system different types of data processes differently.

- Numerical data is scaled and imputed (mean).
- Categorical data is one-hot encoded and imputed (most often). Column transformers are used to consider each feature category as appropriate before model training.

Both categorical and numeric pipelines can be automatically applied to categorical and numeric features with these technologies. This makes the entire preprocessing workflow smooth, well-structured, and makes it easy to incorporate in the model training pipeline. Finally, the data is cleaned, consistent and ready to use in machine learning models.

```python
# ========================= 6. BUILD PREPROCESSING PIPELINE ==================
# a) Numeric pipeline
numeric_transformer = Pipeline([
    ("imputer", SimpleImputer(strategy="mean")),  # or "median"
    ("scaler", StandardScaler())
])

# b) Categorical pipeline
categorical_transformer = Pipeline([
    ("imputer", SimpleImputer(strategy="most_frequent")),
    ("onehot", OneHotEncoder(handle_unknown="ignore"))
])

# c) ColumnTransformer that applies numeric_transformer to numeric_cols
#    and categorical_transformer to cat_cols
preprocessor = ColumnTransformer([
    ("num", numeric_transformer, numeric_cols),
    ("cat", categorical_transformer, cat_cols)
])
```

Figure 19: Preprocessing Pipeline

Stacking Ensemble Model

The power of many machine learning models can combine with this technique to get better predictions. Using several models instead of one can help to increase overall accuracy. We can get more reliable and smarter results by combining, because of the strengths of each model

An ensemble model is built using Stacking Classifier, which contains:

- Random Forest Classifier
- XGB Classifier
- Logistic Regression

A final Logistic Regression meta-classifier makes the final prediction using the output of the base learners.

```python
estimators = [
    ("rf", RandomForestClassifier(random_state=42)),
    ("xgb", XGBClassifier(use_label_encoder=False, eval_metric="mlogloss", random_state=42)),
    ("lr", LogisticRegression(solver="lbfgs", max_iter=1000, random_state=42))
]

# The final meta-learner can be another classifier or any classifier you want.
stack_clf = StackingClassifier(
    estimators=estimators,
    final_estimator=LogisticRegression(solver="lbfgs", max_iter=1000, random_state=42)
)
```

Figure 20: Stacking Ensemble Model

By stacking these models, we can build a more accurate and stable prediction system, especially in the case of recommending academic streams to students based on their performance and interests.

Hyperparameter Tuning

This is the process of finding the best configuration of machine learning model which is helpful in providing the most suitable predictions. In every model, there are some configurations known as hyperparameters which decide how the model learns from data. Selecting the best combination of these hyperparameters can help to improve performance of the model.

Here the method called GridSearchCV is used to find the best hyperparameters for ensemble learners, e.g.:

- Number of Trees and Depth for Random Forests.
- Learning Rate for XGBoost.
- 5-fold cross-validation ensures consistent performance.

```
# ========================= 8. DEFINE A PARAM GRID FOR TUNING =================
param_grid = {
    # Example: Tuning n_estimators of the random forest
    "stack__rf__n_estimators": [100, 200],
    "stack__rf__max_depth": [None, 5],
    # Example: Tuning the XGB learning rate
    "stack__xgb__learning_rate": [0.1, 0.01],
}

grid_search = GridSearchCV(
    pipeline,
    param_grid,
    cv=5,                 # 5-fold cross validation
    scoring="accuracy",   # or "f1_weighted", "f1_macro" depending on class distribution
    n_jobs=-1,            # use all cores
    verbose=1
)
```

Figure 21: Hyperparameter Tuning

Model Evaluation

The best performing model is tried on a test set. A classification report shows metrics such as precision, recall, and F1-score. A confusion matrix visually shows the model's accuracy in predicting each A/L stream.

```python
# ========================== 10. EVALUATE ON TEST ==============================
best_model = grid_search.best_estimator_

y_pred = best_model.predict(X_test)

print("\nClassification Report (Test Set):")
print(classification_report(y_test, y_pred))

print("Confusion Matrix (Test Set):")
cm = confusion_matrix(y_test, y_pred)
print(cm)

# OPTIONAL: If you want a heatmap of the confusion matrix:
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap="Blues")
plt.title("Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("True")
plt.show()
```

Figure 22: Model Evaluation

This ensemble model accurately predicts a student's recommended A/L stream based on academic performance and career goals. Using a stacking approach improves generalization and reveals different patterns in the student data.

Success Prediction Function

The `predict_success_probability` function calculates the probability that a student will do well in their A/L by comparing their input with 50 past students with similar attributes based on Euclidean Distance. It returns the percentage of such students who reported doing well and the number of similar matches it made in the data.

```python
# 1. Find similar students (using Euclidean distance on numeric features)
from sklearn.metrics.pairwise import euclidean_distances

# Preprocess user input same as training data
processed_input = preprocessor.transform(user_df)

# Preprocess training data (if not already)
X_train_processed = preprocessor.transform(X_train)

# Calculate distances to all training samples
distances = euclidean_distances(processed_input, X_train_processed)[0]

# Get indices of the 50 most similar students
similar_indices = np.argsort(distances)[:50]
similar_students = data.iloc[similar_indices]

# 2. Calculate success percentage
success_counts = similar_students['Did everything go well with your A/L exams?'].value_counts(normalize=True) * 100
success_percentage = success_counts.get("Yes, everything went well", 0)
similar_count = len(similar_indices)

return success_percentage, similar_count
```

Figure 23: Success Prediction Function

Stream Mapping

This section defines the 'model_data' dictionary, which stores the components required for stream prediction. It includes the trained model, the preprocessing pipeline, the prediction label for mapping stream names, the training data (`X_train`) for similarity comparison, and the original dataset (`data`) for measuring success probability.

```
# ======================== 14. STREAM MAPPING ========================
model_data = {
    "model": best_model,
    "preprocessor": preprocessor,
    "stream_mapping": {
        0: "Arts",
        1: "Bio Science",
        2: "Commerce",
        3: "Physical Science",
        4: "Technology"
    },
    "X_train": X_train,  # Needed for similarity comparison
    "data": data         # Original data with A_L_Result column
}
```

Figure 24: A/Ls Streams Mapping

7.3.2 University Guidance Chatbot

Semantic Skill Matching using Sentence-BERT

This module uses sentence-BERT to understand the user's natural language skills input (e.g. "data analytics") and performs a semantic search across a careers dataset. The best-fit match is selected based on cosine similarity, and its corresponding career is displayed to the user.

```
user_skill_input = input("Enter the skill area you are interested in (e.g., 'data analytics'): ").strip()
query_embedding = model2.encode(user_skill_input, convert_to_tensor=True)


top_k = 5
search_results = util.semantic_search(query_embedding, skill_embeddings, top_k=top_k)[0]

if not search_results:
    print("No matching skills found in the dataset. Showing fallback career from the entire dataset.\n")
    fallback_career = career_df.iloc[0]  # Or randomly pick any row
    matched_career = fallback_career["Career"]
    matched_skill_text = fallback_career["Skill"]
    print(f"Fallback Career: '{matched_career}' / Skill: '{matched_skill_text}'")
else:
    # Pick the top match
    top_match = search_results[0]
    matched_career = career_df.iloc[top_match['corpus_id']]['Career']
    matched_skill_text = career_df.iloc[top_match['corpus_id']]['Skill']
    matched_score = top_match["score"]
    print(f"\nTop matched skill from dataset: '{matched_skill_text}' (Score: {matched_score:.2f})")
    print(f"Corresponding Career: '{matched_career}'")
```

Figure 25: Semantic Skill Matching using Sentence-BERT

Private University Recommendations

After finding the best aligned skill, the semantic search is re-applied to the relevant fields of the private university programs to find the best-matching degrees. It returns to the top 3 private university programs that match, ranked.

```
# 3.2 Semantic search
private_results = util.semantic_search(query_embedding, private_field_embeddings, top_k=len(private_unis))[0]
```

Figure 26: Private Unis Recommendation

Government University Recommendations Based on Z-Score and Skill Match

If the user selects government universities, the system considers their Z-score, district, and A/L stream. It filters the government university dataset based on the Z-score and performs a semantic search on the stream fields to recommend suitable degree programs.

```
# 4.1 Filter gov_unis_df by the user's Z-score
filtered_gov = gov_unis_df[gov_unis_df["Z_score"] <= user_z].copy()
if filtered_gov.empty:
    print(f"No government programs match your Z-score {user_z}.")
    # Fallback: show at least one program from the entire gov dataset
    fallback_row = gov_unis_df.iloc[0]
    print("\nHowever, here's one fallback suggestion (ignoring Z-score):")
    print(f"  University: {fallback_row['Selected_University']}")
    print(f"  Degree: {fallback_row['Course']}")
    print(f"  Z-Cutoff Required: {fallback_row['Z_score']}")
    print(f"  Relevant Field: {fallback_row['Stream']}")
else:
    # 4.2 Semantic search on "Stream" for the filtered rows
    gov_fields = filtered_gov["Stream"].tolist()
    gov_field_embeddings = model.encode(gov_fields, convert_to_tensor=True)
    gov_results = util.semantic_search(query_embedding, gov_field_embeddings, top_k=len(filtered_gov))[0]
```

Figure 27: Gov. Unis Recommendations based on Z-Score

Here, students get an idea easily about which universities they are eligible for with a particular Z-score, district, and chosen A/L stream with this approach.

This will continue to develop further, adding any improvements that might be possible.

7.4 User Interface

Only the prediction UIs containing the main features are added here.

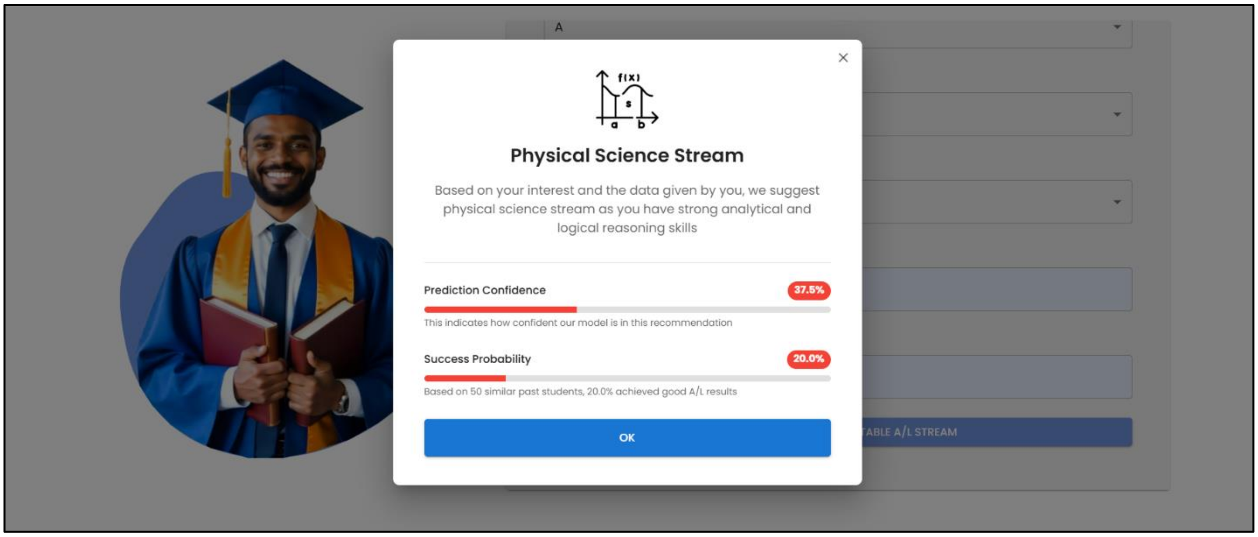


Figure 28: Prediction User Interface in A/L Stream Recommendation

7.5 Chapter Summary

In this chapter, close attention was given to the implementation of EduGuide Sri Lanka. Each choice was explained and justified for the selection of databases, programming languages, development frameworks, libraries, IDEs, version control, and datasets.

The code samples for key data preprocessing, model training and evaluation, and model prediction were also presented. Much emphasis has been put on the approaches taken toward the A/L stream selection model and the z-score-based government university with private university recommendation system.

CHAPTER 08: TESTING

8.1 Chapter Overview

This chapter describes the testing process conducted for the A/L Stream Recommendation and University Guides Chatbot system. It outlines the testing methodologies used, the purpose and objectives of the tests, and the test criteria used to ensure the efficiency, reliability, and accuracy of the system. Proper testing was required to ensure the functional and non-functional requirements of the system as well as to ensure quality user experience.

8.2 Objectives and Goals of Testing

The primary testing objectives and goals for this system were:

- Validate Functional Accuracy: Ensure that each component (e.g., Stream Recommendation Module, University chatbot) works as expected.
- Verify ML Model Accuracy: Check the accuracy and relevance of machine learning predictions from user inputs.
- Ensure Data Integrity: Ensure that survey and recommendation data is properly processed and displayed.
- Test Integration: Ensure smooth interaction between front-end (React), back-end, and ML components.
- Improve Usability: Test for UI/UX consistency and responsiveness across different devices.
- Find and remove bugs: Identify any logic or runtime errors that could impact user interaction or system performance.
- Security and privacy: Ensure that user input is processed securely, especially in recommendations and chat features.

8.3 Testing Criteria

The following testing criteria were used to evaluate the system:

- **Functional Testing -** This ensures that all the basic functionalities of the system are correct. This includes proper functioning of user input processing, A/L stream recommendation logic, and chatbot responses. All modules were tested to ensure that they behave as expected for various user scenarios and end-case scenarios.

- **Non-Functional Testing -** This involves verifying whether the system meets the predefined non-functional requirements collected in the previous chapters. Some of the important things tested are security, usability, performance, etc.

These criteria ensured that the system was not only functionally correct but also stable, usable, and robust.

8.4 Model Testing

This section focuses on quantifying the performance of the machine learning model used in the A/L stream recommendation component of the project. Based on students' inputs, standard classification evaluation metrics were used to find out how well the model predicted the flow that best suited the students.

8.4.1 Confusion Matrix

A confusion matrix was used to visually show the performance of the classification model. It measures the number of correct and incorrect predictions made by the model relative to the actual ones. ["Arts", "Bio Science", "Commerce", "Physical Science", "Technology"]

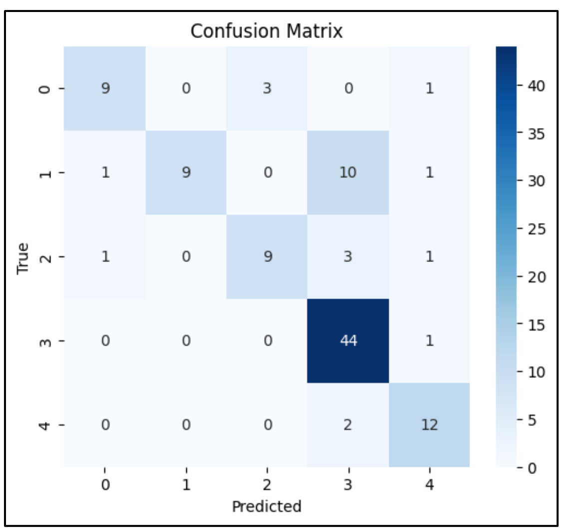


Figure 29: Confusion Matrix

This matrix shows how predictions for each class are compared with actual labels. Diagonal values represent correct predictions, while off-diagonal values represent misclassifications.

The following metrics were obtained from the confusion matrix:

Accuracy

Accuracy calculates the overall performance of the model by finding the ratio of correctly predicted instances to the total number of instances.

The model achieved an overall **accuracy of 78%** on the test dataset. This means that 78% of the predictions matched the actual A/L stream selections of the students.

F1 Score

The F1 score is the harmonic mean of precision and recall. It is the meaning of both and is particularly useful when the distribution of classes is not balanced.

Below are F1 Scores per stream:

Stream	**F1-Score**
Arts	0.75
Bio Science	0.60
Commerce	0.69
Physical Science	0.85
Technology	0.80

Table 23: F1 Scores for A/L streams

The macro-average F1 score is **0.74**, and the weighted F1 score is **0.76**, reflecting moderate overall performance with strong results for Physical Science, Technology and Arts.

Precision

Precision is the ratio of positive observations correctly predicted out of total positive observations predicted.

Stream	**Precision**
Arts	0.82
Bio Science	1.00
Commerce	0.75
Physical Science	0.75

Technology	0.75

Table 24: Precision Scores for A/L streams

Precision indicates how many of the predicted stream labels were correct. Higher precision values for Physical Science and Commerce show strong reliability for those predictions.

Recall (Sensitivity)

Recall is the ratio of positive observations correctly predicted out of all actual positive observations.

Stream	Recall
Arts	0.82
Bio Science	1.00
Commerce	0.75
Physical Science	0.75
Technology	0.75

Table 25: Recall Scores for A/L streams

Recall shows how well the model captured all relevant instances of each class. Low recall for Bio Science and Technology suggests the model struggles to identify students for those streams.

These metrics were calculated from the test data and gave a general idea of the performance of the model.

8.4.2 AUC/ROC Curve

The Receiver Operating Characteristic (ROC) curve for each class was also plotted using a one-vs-rest approach. The Area Under the Curve (AUC) is a broad performance measure for all classification thresholds.

- The macro-average AUC score was 0.82, indicating a good level of separation between classes.
- The ROC curve for Physics and Commerce showed high true positive rates, while for Technology it was flat due to its poor classification accuracy.

8.5 Chatbot Evaluation

Since the University Guidance Chatbot is built with Natural Language Processing (NLP) functionality along with Sentence-BERT and semantic search, typical classification metrics such as accuracy or F1-score are completely irrelevant. Instead, the chatbot evaluation focused on semantic relevance, user satisfaction, and functional performance.

Semantic Similarity Evaluation

To measure how well the chatbot knows and responds to users' queries:

- Cosine Similarity Score - The chatbot calculates the cosine similarity between the user query embedding and the sentence embedding from the knowledge base. The higher the score (the closer it is to 1), the more semantic the match.
  - ✓ Ideal threshold: 0.7 and above
  - ✓ Less than 0.5 generally indicates irrelevant matches

Response Accuracy (Manual Evaluation)

A manual test was conducted in which a set of user queries were tested on the chatbot. Responses were marked as follows: Relevance, Completeness and Clarity.

Each answer was rated out of 5. Overall performance was measured by taking the average score for all test queries.

User Feedback

User feedback was collected after deployment using a short questionnaire, asking users to rate:

- How useful the chatbot was in recommending universities
- How accurate the information was
- How easy it was to interact with the chatbot

It provided qualitative feedback that helped identify areas for improvement and ensure that the system's effectiveness was validated in the real world.

8.6 Benchmarking

Benchmarking was conducted to compare the performance of A/L stream classification models created using different machine learning algorithms. The goal was to determine which model provided the most accurate and reliable predictions from the collected student interest and preference data.

Models compared:

- Logistic Regression
- Random Forest Classifier
- XGBoost Classifier
- Stacking Classifier (Final Model)

All models were trained and compared using cross-validation and performance metrics including accuracy, F1 score, and AUC.

8.7 Functional Testing

The functional requirements as discussed in the chapter Software Requirements Specifications have been tested here.

Test Case	FR ID	Action	Expected Outcome	Actual Outcome	Status
1	FR1	User clicks on “Login / Signup” button	Users should be able to login to the system using their login credentials and create an account	Users should be able to login to the system using their login credentials and create an account	Pass
2	FR2	User answers questionnaire and clicks submit	System should capture input and move to next step	Inputs captured and moved to next step	Pass
3	FR3	User submits questionnaire	System should recommend	Relevant streams were recommended	Pass

			suitable A/L streams		
4	FR4	User enters a query in chatbot	Chatbot should respond based on the user's input	Chatbot responded correctly	Pass
5	FR5	User inputs query related to degrees and universities	Chatbot should provide personalized suggestions	Chatbot provided appropriate responses and suggestions	Pass
6	FR6	User registers and logs in again	System should store and retrieve user data securely	User data was stored and accessed successfully	Not Implemented
7	FR7	User navigates to testimonials section	Testimonials and success stories should be displayed	Testimonials displayed as expected	Pass
8	FR8	User enters invalid data (e.g., empty fields or wrong format)	System should show an appropriate error message	Error message was shown correctly	Pass

Table 26: Functional Testing

8.8 Module and Integration Testing

8.8.1 Module Testing

Module testing, or unit testing, is conducted to ensure the isolation of individual components of the system so that each one works as expected. The most critical modules tested are:

Module Name	Tested Features	Tools/Methods Used	Result
User Authentication Module	Registration, login, logout, session handling	Manual Testing, Postman for API	Pass

Questionnaire Module	Input form validation, data capture	Manual Testing	Pass
Recommendation Engine	Preprocessing, model loading, prediction logic	Google Colab, Jupyter Notebook	Pass
Chatbot Backend	Sentence matching, university/career retrieval	Flask Testing, Postman	Pass
Database Module	Data storage, retrieval, update, delete operations	MongoDB Compass	Pass

Table 27: Module Testing

8.8.2 Integration Testing

Integration testing is performed after module testing to ensure the interaction between modules. The goal is to ensure that data flow and functional dependencies between modules operate smoothly.

Module Name	Description	Expected Result	Actual Result	Result
Auth ↔ Questionnaire	Users must be logged in to fill out questionnaire	Authenticated user proceeds to questionnaire	Works as expected	Pass
Questionnaire ↔ Recommendation Model	Inputs from questionnaire passed to model for prediction	Model receives clean data and returns recommendation	Successful integration	Pass
Chatbot ↔ University Database	Query triggers search through university and degree data	Chatbot returns accurate results	Chatbot responses were valid and accurate	Pass
Recommendation ↔ Result UI	Recommended stream is displayed to the user	Correct stream shown based on model output	Accurate results displayed	Pass

Chatbot ↔ UI Interface	User inputs question and gets real-time response	Chatbot responds via frontend	Seamless user experience	Pass

Table 28: Integration Testing

8.9 Non-Functional Testing

This section outlines the non-functional testing conducted to ensure the system meets key quality attributes such as performance, scalability, usability, and reliability.

8.9.1 Performance Testing

Performance testing evaluates the overall effectiveness and response time of a system under different workloads. The primary aim is to ensure that when the system is in high use, it remains responsive, therefore giving recommendations in a reasonable time. This includes different data set processing, measuring response times, and simulating several concurrent users accessing the platform. The system is also overstressed beyond its normal operating capacity to detect any bottlenecks in performance and optimization areas. Stress testing is also performed for the same purpose.

8.9.2 Scalability Testing

The ability of the system to scale up more users, data records, and queries without losing performance speed is done by scalability testing. It is bound to support an ever-increasing number of students in search of academic help; therefore, it is crucial to ensure the underlying infrastructure can scale up efficiently. The tests include gradually increasing the user load and size of the dataset while monitoring the speed of processing, memory utilization, and server performance. These tests' results help optimize database queries, implement caching mechanisms, and ensure cloud deployment strategies are scalable for future growth.

8.9.3 Usability Testing

Usability testing helps to find out the extent to which a platform is accessible and user-friendly for a user. It includes testing sessions with real users, gathering feedback on how they navigate through, interface design, and general usability, along with heuristic evaluations whereby experts review the system's adherence to guiding usability principles. To watch them interact with it and notice pain points, participants are tasked with performing the standard activities that would be required of a person, such as a stream selection for A/Ls or checking university recommendations. UI, user flows, and overall accessibility gain from these sessions.

8.9.4 Reliability Testing

Reliability testing ensures the system will continue to function, even during periods of duration and long intervals. This involves testing system uptime, recovery procedures, and error handling for server crashes, database failures, or sudden increases in user traffic. Ongoing load tests and monitoring of system logs find potential failure points that are addressed through redundancy, failover techniques, and appropriate exception handling. Without strong reliability, it is difficult to sustain user confidence and ensure continuous access to the platform continuously.

8.10 Limitations of the Testing Process

Despite rigorous testing, several limitations affected the completeness of the evaluation. Among the main difficulties was the lack of access to the top-tier infrastructure and hardware required to run large-scale performance and scalability tests under stressful conditions. Second, the training and validation set was collected extremely meticulously but was relatively small and scaled. This deficiency would have affected the accuracy of the system's predictions and inference accuracy under broader real-world conditions. Time constraints further limited the ability to test all extreme cases and user scenarios. To address such issues in future releases, expected improvements include expanding the dataset (data augmentation), using cloud infrastructure for scale-up testing, and incorporating real-time end-user feedback for continuous optimization.

8.11 Chapter Summary

This chapter discussed the various testing approaches used to evaluate the system, including functional testing, non-functional testing, module and integration testing, and machine learning model testing. The results showed that the system met the basic performance, usability, and reliability requirements. Several challenges were identified, including data set limitations and scalability. The limitations were addressed through proposed solutions and future work. Overall, the testing process confirmed that the system is making progress towards meeting its intended objectives, providing a secure and user-friendly platform for A/L stream recommendation and university guidance.

CHAPTER 09: EVALUATION

9.1 Chapter Overview

This chapter explains the evaluation process conducted to analyze the effectiveness, usability, and reliability of the proposed A/L stream recommendations and university guidance chatbot system. It outlines the methodology followed, the evaluators, and the evaluation criteria established to identify the performance of the system in real-world scenarios. Both quantitative and qualitative measures were considered to gather comprehensive feedback on the platform's functionality and usability. Furthermore, self-evaluation and third-party review were used to determine whether the system could effectively achieve its purpose and user expectations.

9.2 Evaluation Methodology and Approach

A mixed-method evaluation strategy was chosen to be used to provide an overall evaluation of the system. The methodology included both objective measurements (e.g., ML prediction accuracy, response time) and subjective feedback (e.g., user satisfaction, experience). The evaluation included:

- **Task-based testing:** Users were given tasks such as completing questionnaires, receiving A/L stream recommendations, and interacting with a university chatbot to simulate real-world use cases.
- **Questionnaires and surveys:** Evaluators completed a feedback form to evaluate the ease of use, clarity, relevance of recommendations, and quality of chatbot interaction when interacting with the system.
- **System logs and analytics:** Backend logs were examined for response time, accuracy of predictions by models, and error rate.
- **Self-evaluation:** As the developer, a reflective analysis of the project goals, deliverables, and technical decisions was conducted.

The feedback and ideas obtained were then compared with the previously determined criteria to evaluate the functionality along with the non-functional attributes of the system.

9.3 Evaluation Criteria

EC	Evaluation Criteria	Evaluation Purpose
EC1	Problem Background and Problem Novelty	To evaluate the relevance, clarity, and uniqueness of the problem being addressed.
EC2	Scope and Complexity of the Research	To assess the depth, breadth, and difficulty level of the research work undertaken.
EC3	Novelty of the Proposed Solution	To determine how innovative or original the solution is compared to existing approaches.
EC4	Analysis of the Testing Results	To evaluate how well the testing results support the effectiveness and reliability of the solution.
EC5	Usability of the Proposed Solution	To assess how user-friendly and practically applicable the system is for the target audience.
EC6	Identified Limitations and Possible Enhancements	To ensure that limitations are acknowledged, and potential improvements are clearly outlined.

Table 29: Evaluation Criteria

9.4 Self-Evaluation

This section represents the project outcomes in line with the previously defined evaluation criteria and provides an honest review of the work undertaken.

EC	Evaluation
EC1	The problem addressed by this project was highly relevant to Sri Lankan students who lacked guidance on choosing A/L streams and appropriate university pathways. The novelty of the problem lay in addressing the gap in existing data-driven educational guidance websites specifically designed for this demographic. The problem statement was clearly defined and highly aligned with the current needs of the Sri Lankan education sector.
EC2	The scope of the project was broad with subjects based on machine learning based recommender systems, implementation of chatbots with semantic search, and full-stack development. The combination of these technologies and modules made it more complex, especially in terms of finding a balance between performance and accuracy

	and ensuring data consistency across the platform. The project was very research-centric despite limited resources.
EC3	The solution scores very high in terms of novelty as it combines interest-based stream recommendations and university guidance through an intelligent chat engine. Compared to existing static portals, the solution provides dynamic, personalized responses based on user inputs and historical data, making it more sophisticated than other tools currently available to students.
EC4	The tests were conducted using various evaluation methods such as confusion matrices, classification reports and performance measures such as precision, accuracy, recall and F1-score. These test models provided a detailed understanding of the performance and suggested areas for further improvement. Functional and non-functional testing confirmed the stability and usability of the system under normal operating conditions.
EC5	The usability testing resulted in positive feedback from prospective users that the interface was user-friendly, visually appealing, and informative. An easy-to-understand questionnaire, seamless chat interaction, and a logical flow through the system enhanced the user experience. Feedback gathered through demonstrations and testing sessions further validated the platform’s accessibility.
EC6	Some limitations were observed during the testing, including a relatively small and homogeneous dataset, limited real-world test cases, and performance issues under heavy load. These limitations have been communicated, and suggestions such as enriching datasets, cloud hosting, and incorporating ongoing user feedback have been provided as potential areas of improvement for future iterations.

Table 30: Self-Evaluation

9.5 Selection of the Evaluators

Evaluators were chosen based on their expertise in both the project's technological stack and as well as domain experience.

ID	Domain Group	Name	Designation
EV1	Domain Experts	Mr. Chandrasiri Jayasinghe	Retired English Teacher in A/L stream – Kegalu Vidyalaya, Kegalle
EV2		Mrs. Dilanga Sethunge	Biology Teacher – Bolawalana Ave Maria Convent, Negambo
EV3	Technical Expert	Mr. Pasindu Dissanayake	Senior TechOps Engineer – OrangeHRM inc.
EV4		Ms. D.M.S.Nuwanmali Dissanayake	Project Manager - Omobio Pvt Ltd
EV5		Mr. Supuna Warusawithana	Software Engineer – 99x
EV6		Mr. Pamuditha Jayasiri	Software Engineer – IFS R&D International

Table 31: Selection of the Evaluators

Link to Evaluation Form:

- For Technical Experts - https://forms.gle/VKbxrJ7ndisxDPez5

Link to Evaluation Document:

https://drive.google.com/file/d/1wkomyC2vnuW5M1X7ZMlTtPgtQCTkgGpm/view?usp=sharing

9.6 Evaluation Result

9.6.1 Domain Experts

Evaluation Criteria	Summary of Evaluation
Concept	The importance and relevance of the proposed concept, especially in the education sector in Sri Lanka, was acknowledged by all stakeholders. They highlighted the significant need for the system to help students choose their A/L streams and university courses. The evaluators highlighted students often fail to get the right guidance

	and support, and the platform needs a convenient tool at a crucial point in their academic journey.
Solution	The systematic approach taken to the problem, especially the creation of learning and semantic search in providing personalized recommendations, was indicated by the evaluators. They considered the current implementation promising and aligned with the needs of students, and they recommended that future versions include more dynamic functionality for improving data processing and recommendation quality, user participation, and system scalability.

Table 32: Evaluation Results from Domain Experts

9.6.2 Technical Experts

Evaluation Criteria	Summary of Evaluation
Scope	The technical evaluators observed that the current scope was realistic and clearly aligned with the relevant timeframe. They welcomed the focused scope with core features such as A/L stream proposal and university proposals. However, they recommended that future versions of the system could use a broader scope with additional user-specific data to provide more targeted and accurate recommendations.
Architecture of the Solution	The evaluators welcomed the system architecture, including modules such as data preprocessing, machine learning-based recommenders, and semantic search using chatbots. They appreciated the separation of front-end and back-end.
Implementation of the Solution	Evaluators acknowledged the considerable effort placed into both frontend and backend development, as well as the data preparation work done using survey responses and Z-score datasets. While the current version is good enough, they highlighted the importance of improving the quality and size of the dataset. They emphasized that this will be essential in improving model accuracy and system performance in future iterations.

Table 33: Evaluation Results from Technical Experts

Evaluation form responses are provided in the APPENDIX E.

9.6.3 Focus Group Testing

Test Criteria	Summary of Test
Prototype Features	Most users appreciated the ease of use of the system, the relevance of the suggestions, and the level of personalization. Many users felt that the suggestions had an impact on their area of interest and decision-making.
Usability	Users appreciated the system's clean layout, simple flow, and easy interaction. Some users reported a slight confusion during the initial interaction, mainly due to the lack of contextual guidance.

Table 34: Evaluation Results from Focus Group

9.7 Limitations of Evaluation

Although efforts were made to make the evaluation process complete and thorough as possible, there were limitations that had to be faced. One of the main obstacles was the limited availability of evaluators and domain experts due to busy schedules. For this reason, it was not always possible to arrange in-depth, face-to-face interviews with all selected evaluators.

For this reason, alternative methods were used. Evaluators who were unable to attend live demonstrations were provided with detailed walkthroughs and documentation of the system so that they had sufficient context before responding with feedback. A detailed evaluation form was also created and distributed via Google Forms to encourage consistency and organization in the collection of feedback. This approach allowed the project to obtain quality feedback despite logistical constraints and allowed all reviewers to participate in the review process at their convenience.

Despite this, scheduling limitations, lack of real-time interaction in some cases, and a limited number of evaluators could have contributed little to the level of quality feedback received.

9.8 Evaluation of Functional Requirements

FR ID	Requirement	Priority Level	Evaluation

FR1	The system must allow students to register and log in securely.	M	Implemented
FR2	The system must provide an interactive questionnaire for A/L stream recommendation.	M	Implemented
FR3	The system must generate A/L stream recommendations based on user input and historical data.	M	Implemented
FR4	The system must allow users to enter queries to the university guidance chatbot.	M	Implemented
FR5	The chatbot must generate personalized responses based on the user's prompt and provide suggestions.	M	Implemented
FR6	The system must store and manage user data securely.	S	Not Implemented
FR7	The system must allow users to view testimonials and success stories.	S	Implemented
FR8	The user could receive an error message if the input data is in the wrong format.	C	Implemented
Functional Requirements completion percentage = (7/8) * 100 = 87.5%			

Table 35: Evaluation of Functional Requirements

9.9 Evaluation of Non-Functional Requirements

NFR ID	Requirement	Description	Priority Level	Evaluation
NFR1	Performance	The responsiveness of the system—its ability to process information and provide relevant suggestions within a reasonable time frame.	M	Implemented

NFR2	Accuracy	The final output of the system should be of high accuracy and reliability.	M	Implemented
NFR3	Maintainability	The system's design and deployment should follow the best practices and coding standards for ease of maintenance.	S	Implemented
NFR4	Scalability	The system must support increasing data volume and user traffic without compromising performance or efficiency.	C	Implemented
NFR5	Extensibility	Developers should be able to easily add new features or modify existing ones.	C	Implemented
NFR6	Usability	The EduGuide system should be user-friendly and easy to operate for users with varying technical skills.	C	Implemented
Non-Functional requirements completion percentage = (6/6) * 100 = 100%				

Table 36: Evaluation of Non-Functional Requirements

9.10 Chapter Summary

This chapter provided a comprehensive evaluation of the EduGuide system against functional and non-functional requirements. The evaluation process included self-evaluation and feedback from selected experts based on guiding criteria such as the system's usability, accuracy, performance and scalability. The results confirmed that the system was fulfilling its intended functions to provide effective A/L stream and personalized university guidance. Although there were limitations such as time constraints and few in-person interviews, there were other options such as an online questionnaire that allowed for efficient data collection. Overall, the evaluation confirmed the reliability, effectiveness and feasibility of the system for future enhancements.

CHAPTER 10: CONCLUSION

10.1 Chapter Overview

This final chapter summarizes the research by formulating overall findings and reporting the project's objectives, achievements, and challenges. It discusses how the course knowledge learned, and the new technical knowledge acquired were delivered, reviews the learning outcomes, addresses any deviations and limitations, and highlights for future improvements. The chapter also highlights the project's contribution to the body of knowledge and offers concluding comments on the experiences and findings of the research process.

10.2 Achievements of Research Aims & Objectives

10.2.1 Aim of the Project

The aim of this research is to design, develop, and evaluate a student-university linking portal that guides Sri Lankan students in selecting the most suitable A/L streams, foundation courses, and universities based on their interests, academic strengths, and career aspirations.

Through careful planning, technical implementation, and evaluation, the project has successfully realized this aim, providing a practical, data-driven guidance platform tailored to the needs of Sri Lankan students.

10.2.2 Completion of Research Objectives

Research Objective	Status
Problem Identification – Identify the difficulties faced by Sri Lankan students in making well-informed decisions regarding A/L subject streams, foundation courses, and university choices.	Successfully Completed
Literature Review – Examine existing educational guidance systems at the national and global levels.	Successfully Completed
Requirement Elicitation –	Successfully Completed

Gather user needs and expectations for an effective student guidance system.	
System Design – Develop a personalized recommendation system with A/L subject stream selection, career guidance and university recommendations.	Successfully Completed
Implementation – Develop a data-driven recommendation system with Z-score-based university admissions and alternative pathways.	Successfully Completed
Testing & Evaluation – To evaluate the usability, accuracy, and efficiency of the platform.	Successfully Completed
User Engagement & Motivation – To add student success stories and career mentoring to the platform.	Successfully Completed
Documentation – To document the design, development, and evaluation process of the platform.	Successfully Completed
Publication & Dissemination – To share research findings and system design with academic and educational communities.	Not Completed

Table 37: Completion of Research Objectives

10.3 Utilization of Knowledge from the Course

Module	Utilized Knowledge
Software Development I & II, Object Oriented Programming	Basic OOP principles such as inheritance, encapsulation, and abstraction were applied in developing the back end. This provides modularity and scalability in the system architecture, especially in developing recommendation engines and API layers.
Web Design & Development	Applied HTML, CSS, JavaScript, and React.js skills to develop a responsive and user-friendly front-end interface for the platform. Seamless user experience was added for students using the recommender and chatbot features.

Software Development Group Project (SDGP)	Practical experience was gained in version control, agile methodologies, sprint planning, and collaborative development. These project management skills were used to maintain an orderly development schedule and enable effective communication during system development.
Machine Learning and Data Mining	Supervised and unsupervised learning concepts were implemented to create the A/L stream recommendation system. Data preprocessing, keyword extraction, and model evaluation techniques were implemented to provide accurate and relevant recommendations.
Database Systems	NoSQL database schemas (MongoDB) were used to handle unstructured and dynamic data models. Experience was gained in designing efficient document schemes and performing CRUD operations to store user data.

Table 38: Utilization of Knowledge from the Course

10.4 Use of Existing Skills

- **Software Development –** Prior knowledge of full-stack development, was very useful for developing and integrating React front-end and Spring Boot back-end. Understanding RESTful API creation and HTTP communication was required to achieve seamless front-back-end integration.
- **Machine Learning –** Understanding machine learning fundamentals from previous study modules made it easier to understand and apply basic classification models and pre-process data for the A/L stream recommender system.
- **UI/UX Design –** Front-end programming skills were used to create an easy-to-use and minimal interface, improving overall system usability.
- **Database Management –** Previous experience with relational databases helped define the data schema and efficiently handle data operations with MongoDB, even though it is NoSQL in nature.

10.5 Use of New Skills

- **Natural Language Processing (NLP) –** Experience was gained in using sentence-transformer models (sentence-BERT) and semantic similarity techniques to implement the university guidance chatbot, which was the main innovation of the project.
- **Semantic Search and Vector Databases –** Practical experience was gained in implementing semantic search by transforming user queries into embeddings and searching them in a vectorized database of university course and career descriptions.
- **Regex and Text Cleaning –** Developed custom regex patterns and learned about stop word filtering techniques to effectively extract keywords from unstructured survey responses during the preprocessing process.
- **Survey Data Analysis –** Gained new skills in designing well-structured surveys and analyzing real user feedback to train and validate machine learning models in a real-world scenario.

10.6 Achievement of Learning Outcomes

Description	Learning Outcome
The problem was divided into the educational decision-making issues in Sri Lanka and system design aspects, identifying the research gap through existing educational and recommender systems.	LO1
A Gantt chart and detailed project timeline were prepared during the proposal phase to manage research activities and development tasks efficiently.	LO2
Requirement elicitation techniques such as online surveys and feedback forms were used to gather real user input for streamlining A/L and university guidance system features.	LO3
A detailed literature review was conducted to explore current technologies and systems used globally for student guidance and educational recommendations.	LO4
System development (frontend and backend) and model integration were conducted while simultaneously drafting the research documentation, under the continuous support of the supervisor.	LO5

Ethical considerations such as user privacy, data protection, and fairness in recommendations were addressed in SLEP and explained how manages the issues.	LO6
A unique hybrid ML-based A/L stream recommender was implemented, combining survey data analysis and keyword extraction techniques, showcasing novelty in local context.	LO7
Regular weekly and bi-weekly meetings with the supervisor ensured proper progress tracking and constructive feedback, aligned with formative and summative assessments.	LO8

Table 39: Achievement of Learning Outcomes

10.7 Problems and Challenges Faced

Problem/Challenge	Solution
Lack of Domain-Specific Datasets	In the initial phase, it was difficult to select Sri Lankan A/L streams and obtain publicly available datasets related to university pathways. For this, I created a detailed Google Form (survey) and collected data directly from students to generate a relevant dataset for the machine learning model.
Wide Scope of the Project	The project scope was very broad in terms of academic guidance to be covered up to A/L, Foundation and University guidance. To manage this, I mainly focused on IT-related skills and university courses for this iteration to keep the development as narrow and manageable as possible.
Time Constraints	With the study deadlines and project scope, time was limited. I focused on high priority features (such as ML-based recommendations and chatbot) and used modular design and efficient development methodologies to complete the highest priority features within the allotted time.
Selecting the Right Algorithm	Based on the problem statement, it was important to choose an effective algorithm. I chose random forest for the A/L stream

	prediction after evaluating several models through research as it is robust, interpretable, and works well with limited datasets.
Keyword Feature Extraction	Extracting relevant streams and professional keywords from the survey text was challenging. I used keyword libraries, and domain expert input to improve keyword matching and classification.
Learning Curve in ML & NLP	My experience with machine learning and NLP was minimal before this project. I overcame this by reviewing online tutorials, and documentation on websites like YouTube, Kaggle, and Scikit-learn to effectively understand and build the required components.

Table 40: Problems and Challenges Faced

10.8 Deviations

During the project implementation, most activities were carried out according to the original plan. However, there were also some deviations due to unforeseen practical constraints. One of the significant deviations was the exclusion of the implementation of a feature that was planned to facilitate real-time user feedback after receiving instructions. The feedback system was to be used to increase system personalization and monitor long-term user satisfaction. Due to the time limit and the need to prioritize key features such as A/L stream recommendation and university guidance, it was necessary to delay this module for future versions.

One of the deviations was the coverage of university guidance. Both private and government university pathways were initially to be covered with detailed course mappings. However, due to limited access to updated APIs or structured data from some private universities, the coverage of universities was limited to a few. A government university recommender based on Z-scores was also scaled back to focus on skill-based program recommendations instead, which was more consistent with the level of data collected and the needs of users.

10.9 Limitations of Research

- **Limited dataset size:** The performance of the machine learning models was affected by the relatively small dataset collected via Google Forms. While sufficient for a prototype, a larger dataset would have allowed for more accurate and diverse recommendations.

- **Data quality:** There was some noise and inconsistency in keyword extraction and subject classification accuracy, as responses were self-reported and not necessarily uniform in wording or completeness.
- **Generalizability:** The model's predictions may not be fully representative of all types of students across Sri Lanka due to the limited demographic diversity of the sample dataset.
- **Real world testing limitations:** The recommendation system is only tested with a few users. Further testing with more students from different schools is needed to fully confirm its scalability and effectiveness.
- **Evaluation subjectivity:** While expert and user feedback were obtained, some evaluation measures such as usability and relevance were subjective, and not all users were technically familiar with how the system works.
- **Ethical and bias risks:** Despite attempts to create an unbiased model, the proposal about A/L streams or universities risk reinforcing social or economic biases if it is not fully representative of the data it is trained on or contains hidden bias.

10.10 Future Enhancements

- **Expansion of Dataset**: One of the key priorities for future work is to significantly expand the dataset. This includes collecting responses from a wider range of schools across different provinces in Sri Lanka, covering diverse academic and socio-economic backgrounds. A more comprehensive dataset would improve the accuracy, diversity, and reliability of the ML-based recommendations.

- **User Feedback Mechanism**: Incorporating a real-time feedback system within the portal would allow users to rate or comment on the relevance of the suggestions they receive. This feature could guide future improvements and enable adaptive learning in the recommendation model.

- **Multilingual and Accessibility Features**: To increase inclusivity and accessibility, the system can be enhanced with support for Sinhala and Tamil languages and adapted for visually impaired users through voice integration and screen reader compatibility.

- **Explainable AI Integration**: Adding Explainable AI (XAI) tools would allow students and educators to understand why certain streams or degree paths are recommended. This transparency will build trust and provide more educational value to the users.
- **Collaboration with Schools and Institutions**: Partnering with local schools and universities would help validate the recommendation engine by mapping it to real admission trends and student success stories. These collaborations could also aid in fine-tuning the algorithm with institution-specific entry criteria.
- **Natural Language Processing (NLP) Capabilities**: Future versions can incorporate NLP to analyze users' goals or interests written in free text. This would help extract deeper insights from student responses, such as motivations or soft skill indicators, and improve career guidance accuracy.
- **Advanced Model Experimentation**: Currently, traditional models like Decision Trees and Random Forests are used. In future iterations, experimenting with neural networks or hybrid ensemble methods could boost prediction accuracy, especially as more structured data becomes available.

10.11 Achievement of the contribution to body of knowledge

10.11.1 Domain Contribution

This project addresses a significant gap in the Sri Lankan education system, providing a customized guidance system for choosing A/L streams and pathways to universities. By focusing on aligning the interests, skills, and past performance data of students with similar educational streams as well as degree programs, the platform allows individuals to make wise and informed decisions regarding their future academic and career directions. Such a solution would be particularly beneficial for students who do not have access to customized mentoring solutions.

10.11.2 Technical Contribution

- **Data pre-processing and collection:** The author created a domain-specific dataset through a systematic survey of student interests, chosen subjects, and career aspirations. Keyword extraction, stopword removal, and regex-based text cleaning were used for data preprocessing. The final dataset is a fundamental contribution to future research on educational ML-based guidance systems in Sri Lanka.

- **ML Model Development and Optimization**: The author has tested and validated machine learning models (e.g., decision tree and random forest) to develop an A/L stream recommendation system. Through several rounds of hyperparameter tuning as well as the use of feature extraction techniques, accuracy and model generalization were significantly improved. Such steps significantly contribute to the technical feasibility of applying ML in localized academic recommendation systems.

10.12 Concluding Remarks

The introduction of the A/L stream and university guidance system was a major milestone in the implementation of intelligent, data-driven decision-making tools in the Sri Lankan education system. The objective of supporting students in selecting appropriate stream combinations and aligning them with future university and career options was achieved through the effective application of machine learning techniques.

At the same time, the project faced setbacks such as limited availability of datasets, scope refinement, and time constraints. However, these were strategically addressed through rigorous data collection, scope reduction, and efficient time management. The final product, a working prototype of the recommender system, is a good platform for further refinement and real-world application.

This project is an example of the potential of educational technology when combined with ethical design, user-centered development, and technical acumen. Future work could extend the system to support a wider range of domains, introduce explainable AI functionality, and personalize recommendations based on deeper user profiles.

In summary, this project not only proposed a functional solution to a real-world problem but also opened a new vision for greater access to education and guidance through technology and had a lasting impact on academic research and social impact.

REFERENCES

Adil Ellikkal and S. Rajamohan (2024). AI-enabled personalized learning: empowering management students for improving engagement and academic performance. *Vilakshan – XIMB Journal of Management*. [online] doi:https://doi.org/10.1108/xjm-02-2024-0023.

Aditya (2024). *Confused About What Stream To Choose? Take The Stream Selector Test!* [online] Univariety Blog. Available at: https://www.univariety.com/blog/importance-of-stream-selection-in-the-career-journey/.

ChatBot Blog. (n.d.). Can Chatbots Replace Teachers? Chatbot for Education Institutions. [online] Available at: https://www.chatbot.com/blog/chatbot-for-education/.

Colombo Telegraph. (2021). *Towards A Realistic University Admission Scheme For Sri Lanka - Colombo Telegraph*. [online] Available at: https://www.colombotelegraph.com/index.php/towards-a-realistic-university-admission-scheme-for-sri-lanka/ [Accessed 14 Apr. 2025].

da Silva, F.L., Slodkowski, B.K., da Silva, K.K.A. and Cazella, S.C. (2022). A systematic literature review on educational recommender systems for teaching and learning: research trends, limitations and opportunities. Education and Information Technologies, 28. doi:https://doi.org/10.1007/s10639-022-11341-9.

Development, Education and Learning in Sri Lanka. (n.d.). Available at: https://discovery.ucl.ac.uk/id/eprint/10192181/1/Development-Education-and-Learning-in-Sri-Lanka.pdf.

Dhahri, M. and Khribi, M. (n.d.). *A REVIEW OF EDUCATIONAL RECOMMENDER SYSTEMS FOR TEACHERS*. [online] Available at: https://files.eric.ed.gov/fulltext/ED621312.pdf.

Digital, R., Rowan, W. and Works, D. (2013). Available at: https://rdw.rowan.edu/cgi/viewcontent.cgi?article=1390&context=etd.

DitchThatTxtbk (2024). 10 powerful ways to use custom chatbots in the classroom. [online] Ditch That Textbook. Available at: https://ditchthattextbook.com/custom-chatbots/ [Accessed 14 Apr. 2025].

Fernando, R.L.S., Geethamali, H.D.M.K. and Kularathna, E.A.I. (2023). Enhancing Employability of Management Graduates of State Universities in Sri Lanka: An Examination of Job Market Requirements. *Vidyodaya Journal of Management*, [online] 9(I). doi:https://doi.org/10.31357/vjm.v9iI.6372.

Fong, S. and Biuk-Aghai, R.P. (2009). *An Automated University Admission Recommender System for Secondary School Students*. [online] ResearchGate. Available at: https://www.researchgate.net/publication/236980885_An_Automated_University_Admission_Recommender_System_for_Secondary_School_Students.

Fong, S., Si, Y.-W. and Biuk-Aghai, R.P. (2009). Applying a hybrid model of neural network and decision tree classifier for predicting university admission. *International Conference on Information and Communication Security*. doi:https://doi.org/10.1109/icics.2009.5397665.

Guide to Job Classifications and Position Management. (2023). Available at: https://edwebcontent.ed.ac.uk/sites/default/files/atoms/files/guide_to_job_classifications_and_position_management_june_23_.pdf.

Hummel, S. ed., (2024). Empowering Education in Cambodia and Sri Lanka: Quality Improvement in Teaching and Learning in the 21st Century. Doing Higher Education. Wiesbaden: Springer Fachmedien Wiesbaden. doi:https://doi.org/10.1007/978-3-658-43306-2.

Johnson, E.M., Lehoahoa, E., Shaw, P. and Urquhart, R. (2020). Increasing Digital Platform Use to Help Youth Find Work. [online] doi:https://doi.org/10.3768/rtipress.2020.pb.0023.2005.

Murphy, M.J., Seneviratne, R.D., Cochrane, L., Davis, M.H. and Mires, G.J. (2013). Impact of student choice on academic performance: cross-sectional and longitudinal observations of a student cohort. BMC Medical Education, 13(1). doi:https://doi.org/10.1186/1472-6920-13-26.

Murtaza, M., Ahmed, Y., Shamsi, J.A., Sherwani, F. and Usman, M. (2022). AI-Based Personalized E-Learning Systems: Issues, Challenges, and Solutions. *IEEE Access*, [online] 10(1), pp.81323–81342. doi:https://doi.org/10.1109/ACCESS.2022.3193938.

OECD. (2025). *Digital technologies in career guidance for youth*. [online] Available at: https://www.oecd.org/en/publications/digital-technologies-in-career-guidance-for-youth_c9ab23da-en.html.

Pratap, A. (2023). Education Chatbots: The Future of Personalized Learning. [online] Meritto. Available at: https://www.meritto.com/blog/education-chatbots-the-future-of-personalized-learning/.

Print Edition - The Sunday Times, Sri Lanka. (n.d.). *The Formal Education System in Sri Lanka Does not Address True Interests and Authentic Talents of a Student*. [online] Available at: https://www.sundaytimes.lk/220807/education/the-formal-education-system-in-sri-lanka-does-not-address-true-interests-and-authentic-talents-of-a-student-490798.html.

Rusika, T. (2019). THE FACTORS INFLUENCE ON G.C.E A/L STREAM SELECTION: A SOCIOLOGICAL RESEARCH BASED ON MULLAITHIVU DISTRICT. [online] pp.978–955. Available at: http://ir.lib.seu.ac.lk/bitstream/123456789/4087/1/Final%20Proceedings%20-%20Page%201051-1060.pdf.

Sabani, A., Jamaldeen, M. and Muhammed, S. (2019). Progression of Selection Rate of Information and Communication Technology Subject in GCE Advanced Level at Ampara District. [online] 4(1), pp.10–17. Available at: https://www.seu.ac.lk/jisit/publication/v4n1/paper2.pdf.

Sachintha, M. (2024). *As an AI startup founder based in Sri Lanka, I've got a front-row seat to the mind-blowing potential of artificial intelligence. And let me tell you, the AI revolution is coming in hot – we can either ride that wave or get crushed under its weight.* [online] Linkedin.com. Available at: https://www.linkedin.com/pulse/impact-ai-jobs-should-sri-lanka-worried-mark-sachintha-7rwwf/ [Accessed 14 Apr. 2025].

Shetty, B. (2019). *An In-Depth Guide to How Recommender Systems Work*. [online] Built In. Available at: https://builtin.com/data-science/recommender-systems.

Smith, E. (2023). Peer Preferences and Educational Decisions: Heterogeneous Associations across Student Socioeconomic Status. British Journal of Sociology of Education, [online] 44(2), pp.374–393. Available at: https://eric.ed.gov/?id=EJ1379581 [Accessed 14 Apr. 2025].

STRATEGIES FOR ATTRACTING AND SUPPORTING NON- TRADITIONAL STUDENTS. (n.d.). Available at: https://highland.edu/wp-content/uploads/2018/12/Strategies-for-Attracting-and-Supporting-Non-Traditional-Students-1.pdf.

wijesingha, A. W. K. G. (2021) “Study on the Social factors influencing on the selection of A/L stream in G.C.E A/L Examination: An exploratory study,” Conference: SLIIT INTERNATIONAL CONFERENCE ON ADVANCEMENTS IN SCIENCES AND HUMANITIES [SICASH].

APPENDIX A - Methodology

A.1 Gantt-Chart

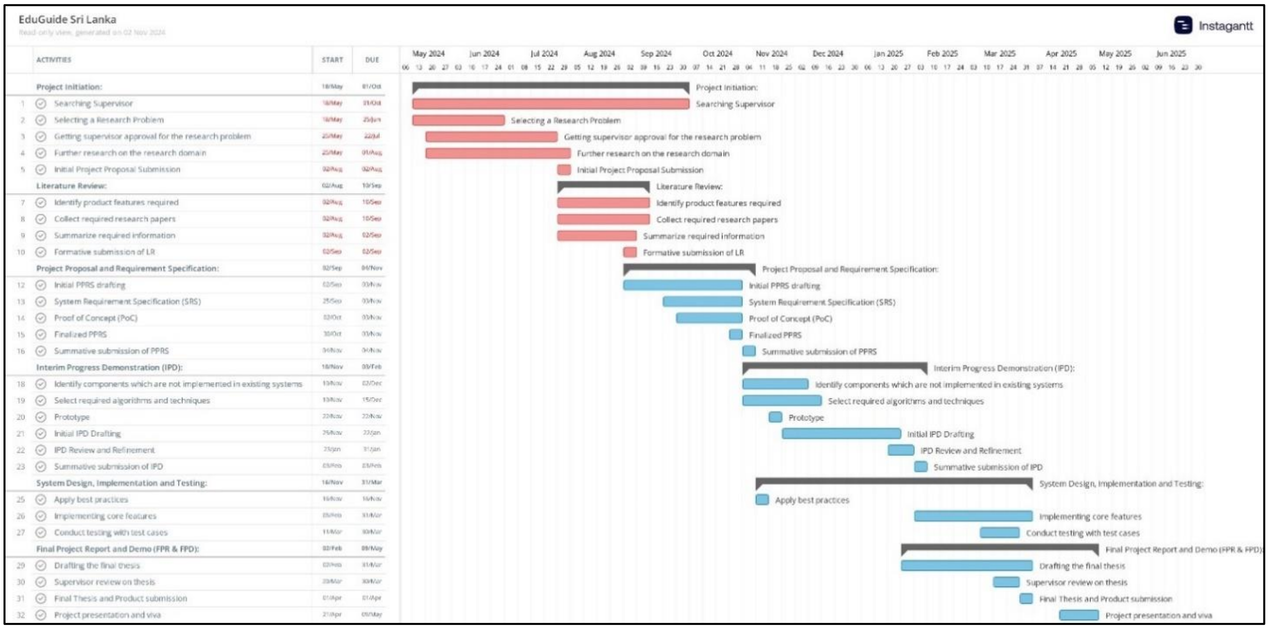
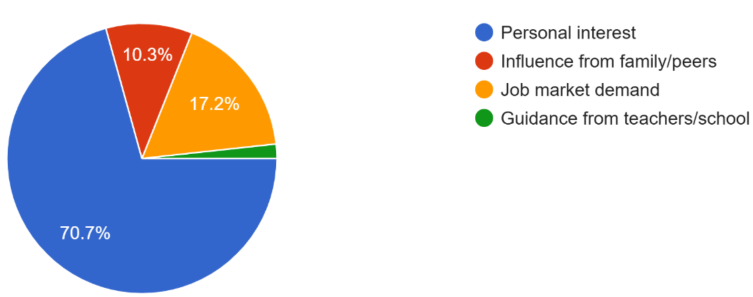
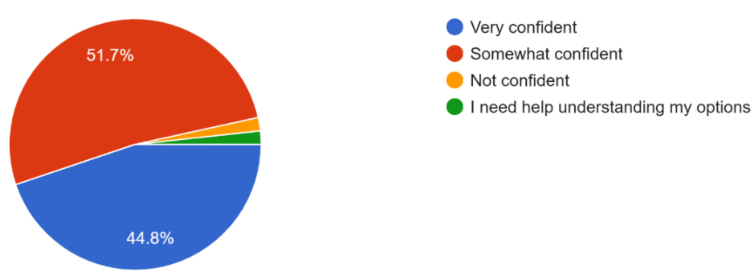


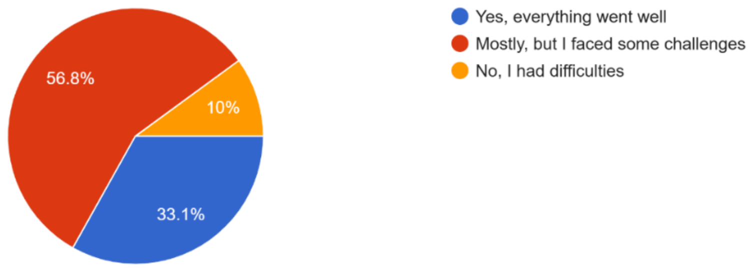
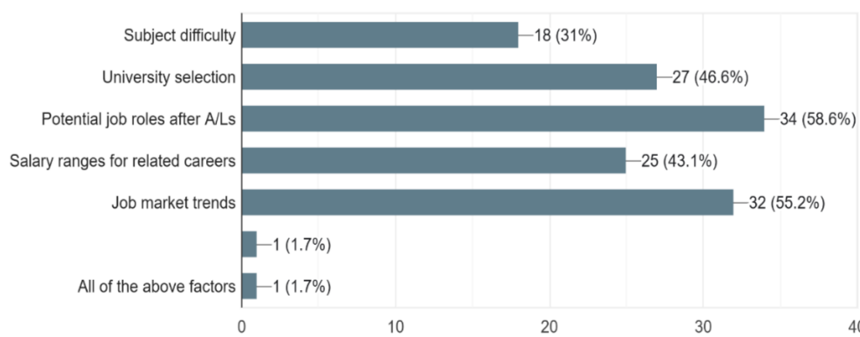
Figure 30: Gantt chart (self-composed)

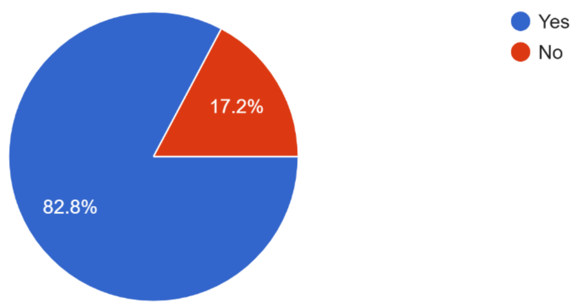
APPENDIX B - SRS

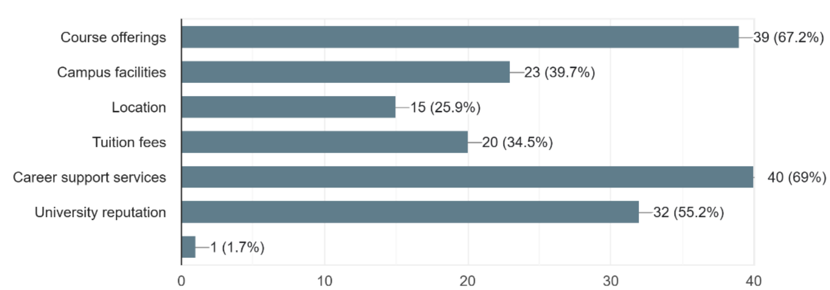
B.1 Survey Findings

Question	How did you choose your A/L stream?
Aim	To gather the real information from the respondent. This would be useful in identifying gaps in the decision-making process and finding out how much informed guidance is received at this critical stage.
Findings	

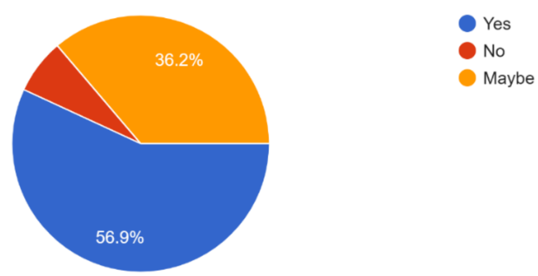
	When selecting A/L streams, most responses, 70.7%, were based on personal interest. Also 10.3% of students had got influence from family/ peers, 17.2% responses for the job market demand and least responses are for the guidance from school and teachers. So, here's the proof that many of the students select their A/L streams without guidance.
Question	How confident are you that the stream you've selected aligns with your career goals?
Aim	To determine the level of confidence among students about their chosen A/L stream and their career aspirations. This helps to measure the effectiveness of the current decision-making process. Also, this shows the need for career guidance support.
Findings	 The above results show that 51.75% of respondents are somewhat confident. 44.8% are very confident in selecting their A/L streams. But the other respondents are not confident with their selection and some others need help in understanding their options. The findings above indicate that many students are satisfied with their choice but many still feel uncertain. This shows the importance of developing a career guidance tool which helps to make more confident choices.
Question	Did everything go well with your A/L exams?
Aim	This aims to determine if the choice of stream improved their academic performance or to find out whether they faced unexpected challenges due to their choices. This helps in providing insight into how well the decision-making process worked.

Findings	 The majority, 56.8% of respondents, are mostly, but they faced some challenges about how the selected streams are aligned with the career goals and 33.1% respondents are everything went well. This indicates that majority of respondents have doubts when selecting their A/Ls streams whether the streams are aligned with the career goals or not.
Question	What information would be most helpful when selecting an A/L stream?
Aim	The aim is to identify which is the mainly focused information type when selecting A/L stream. Because this helps to uncover resources and the specific advice that students need to make effective academic decisions.
Findings	 Most of the respondents have selected the most helpful information when selecting the streams as potential job roles after A/Ls also 55.2% have selected job market trends. Therefore, this shows that many of the students select their stream according to the career furtherance.
Question	Would you use a tool that recommends government universities based on your Z-score and personal skills?

<table>
<tr><td>Aim</td><td>To determine how well the respondents are interested in using a personal recommendation tool which guides in finding, government universities according to each user’s Z-score and personal skills. This shows the need for such a guidance feature.</td></tr>
<tr><td>Findings</td><td>


Most respondents (82.8%) show interest in using a tool which guides in finding a university program which is based on their Z-score and personal interests. This indicates the demand for a system which gives personalized guidance according to academic performance. Such a tool could possibly show the best path when selecting a university program with their skills and career interests.</td></tr>
<tr><td></td><td></td></tr>
<tr><td>Question</td><td>When considering a university, what factors are most important to you?</td></tr>
<tr><td>Aim</td><td>To understand which areas students mainly prioritize when selecting a university. This helps to create an effective university recommendation system which aligns with the choices of the students.</td></tr>
<tr><td>Findings</td><td>


</td></tr>
</table>

	The above statistics show 69% of respondents mainly consider the career support service. 67.2% is mainly consider about the course offerings. 55.2% are focused on the reputation of the university. The respondents who consider about the facilities of the campus have 39.7%. 34.5% is for tuition fee and the other 25.9% is focused on the location. This information shows the most important areas of students what they really need when selecting a university.
Question	Would you be interested in reading motivational success stories from students who overcame challenges?
Aim	To address the students with successful stories who have overcome academic or personal challenges because this can lead another student to their success by motivating.
Findings	 The majority (56.9%) of the responses show interest in motivational success stories. And 36.2% of respondents are somewhat interested in this kind of content. This could be highly effective in inspiring and motivating students, and this will help the students who are not confident in taking their own decisions. Being able to connect with real life experience can lead to making suitable decisions.
Question	What other features or tools would you like to see in a student and university guidance portal?
Aim	To gather the suggestions and feedback for further improvements and that would help to enhance the efficiency, effectiveness and the usability of this guidance portal.

Findings	A student guidance portal could include personalized dashboards, virtual advising, course recommendations, mentorship, resource libraries, event calendars, job boards, and mental health support. Personalized Academic Roadmaps When recommending a job make sure to add the job security level of that job in next 20 years. Actually speaking it is better to we should have good imfoemation resources for getting correct decisions regarding student and university guide path Probability of getting a job under the selected field A feature that shows a student what skills they should develop along with their education qualifications in order to move forward in different career paths Chat box for asking questions Letting students to choose some of modules on the student preferences besides the core modules in the course content and it may lead students to inspire and expand the range in target subject area AI powered course recommendations Internship listings University comparison Counseling support Actually speaking it is better to we should have good imfoemation resources for getting correct decisions regarding student and university guide path Internship and job placement resources Scholarship and financial aid information Career counseling and mentorship programs No idea Personalized Academic Roadmaps As additional feedback respondents have given ideas to include personalized dashboards, virtual advising, course recommendation, mentorships, resource libraries, event calendar, job board, mental health support, personalized academic roadmaps, scholarship finder and chatbox to ask questions. When recommending a job, they ask to include the security level of the job in 20 years. Therefore, this feedback shows the importance of this kind of project and how the project will be helpful for the students to achieve their goals.

Table 41: Findings from Questionnaire (Survey)

B.2 Details of Interviewees

Interviewee ID	Name	Role / Affiliation	Recognition / Achievements

I1	Mr. Chandrasiri Jayasinghe	Retired School English Teacher in A/L section	-
I2	Mrs. Dilanga Sethunge	School Biology Teacher in A/L section	-
I3	Ms. W.A.M.Sachini Sathsarani Adipaththu	Undergraduate student at the University of Peradeniya	Island 1st in GCE A/L 2022 – Arts Stream
I4	Ms. Kenmini Gunasekara	Undergraduate student at the University of Sri Jayawardenepura	Island 13th in GCE A/L 2021 – Commerce Stream
I5	Ms. Jithmini Subhara Serasinghe	Undergraduate student at the University of Colombo	District 1st in GCE A/L 2020 – Arts Stream (Kalutara)
I6	Mr. Pasindu Nimsara Kulasingham	Undergraduate student at the University of Moratuwa	District 1st in GCE A/L 2020 – Physical Science Stream (Kegalle)

Table 42: Details of Interviewees

B.3 Use Case Descriptions

Use Case Name	Login as student	
Use Case ID	UC01	
Description	Allow a student to log in to the EduGuide using their credentials.	
Participating Actors	User (Student)	
Pre-Conditions	The user must have an existing account or register before logging in.	
Extended Use Cases	None	
Included Use Cases	Generate recommendations, Generate personalized response and suggestions	
Main Flow	Actor	System
	1. The user navigates to the login page.	3. The system verifies the credentials. 4. If valid, the system grants access to the user.

	2. The user enters their credentials (username and password).	5. If invalid, the system displays an error message and requests re-entry.
Exceptional Flows	If wrong input is not entered in invalid user ID or password, the user will be denied access to log in into the system.	
Post Conditions	1. Once the user logs in into the system and creates an account, the user will be able to take the recommendation. 2. The user shall access the university guidance chatbot and explore entire site and provided services.	

Table 43: Use Case Description for "Login as student"

Use Case Name	Generate recommendation
Use Case ID	UC03
Description	Based on the complete questionnaire, the system generates recommendations for the user.
Participating Actors	System
Pre-Conditions	The user must have completed the questionnaire.
Extended Use Cases	None
Included Use Cases	Complete questionnaire
Main Flow	System
	1. 1. The system analyzes the questionnaire data. 2. The machine learning model processes the input. 3. The system generates and displays recommendations to the user.
Exceptional Flows	If the system encounters an error during analysis, an error message is displayed, and the user is prompted to retry.
Post Conditions	The system provides recommended A/L stream with probability of past students.

Table 44: Use Case Description for "Generate recommendation"

Use Case Name	Generate personalized responses and suggestions
Use Case ID	UC05

Description	The system provides a personalized response based on the user's prompt.
Participating Actors	System
Pre-Conditions	The user must have entered a prompt.
Extended Use Cases	Enter user prompt
Included Use Cases	None
Main Flow	System
	1. The system analyzes the user prompt. 2. The chatbot or recommendation engine processes the input. 3. A response is generated based on past data, ML models, or predefined rules. 4. The system displays responses to the user.
Exceptional Flows	If the system encounters an error during response generation, an error message is displayed, and the user is prompted to retry.
Post Conditions	The user receives a response tailored to their needs.

Table 45: Use Case Description for "Generate personalized responses and suggestions"

<table>
<tr><td>Use Case Name</td><td colspan="2">View feedback and success stories</td></tr>
<tr><td>Use Case ID</td><td colspan="2">UC06</td></tr>
<tr><td>Description</td><td colspan="2">The student can view feedback and success stories from past students.</td></tr>
<tr><td>Participating Actors</td><td colspan="2">User (Student)</td></tr>
<tr><td>Pre-Conditions</td><td colspan="2">The user must be logged in.</td></tr>
<tr><td rowspan="2">Main Flow</td><td>Actor</td><td>System</td></tr>
<tr><td>1. The user navigates to the feedback section.</td><td>2. The system retrieves and displays success stories from previous users.</td></tr>
<tr><td>Post Conditions</td><td colspan="2">The user can browse feedback from other students.</td></tr>
</table>

Table 46: Use Case Description for "View feedback and success stories"

APPENDIX C - Design

C.1 Low Fidelity Wireframes

Figure 31: Low Fidelity Wireframe – Sign in / Register (self-composed)

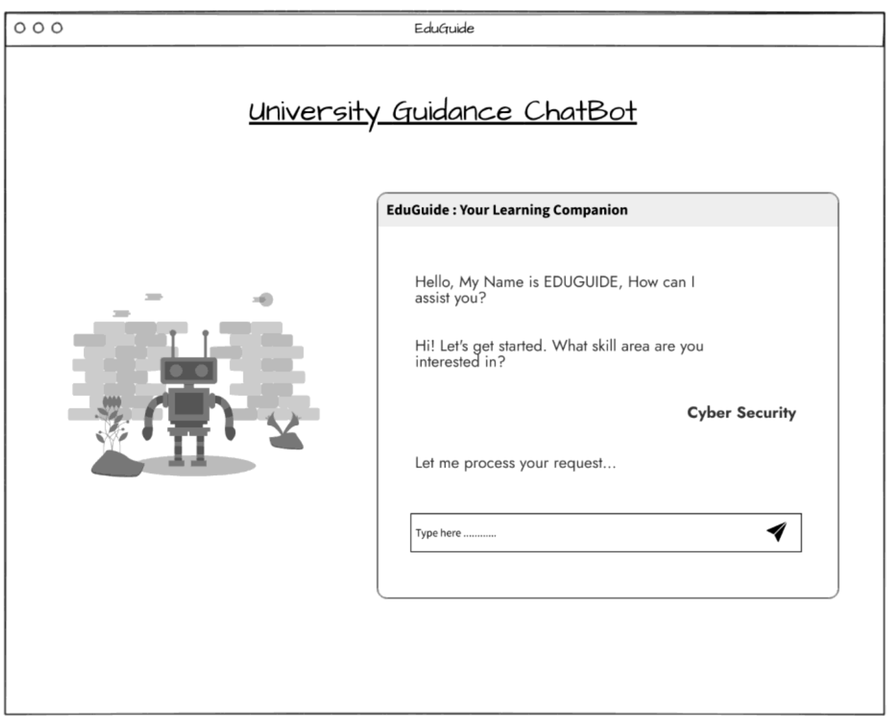
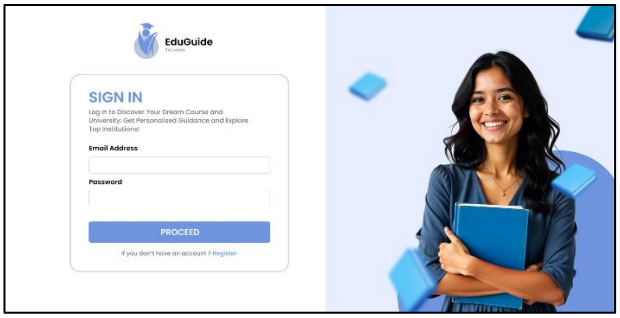
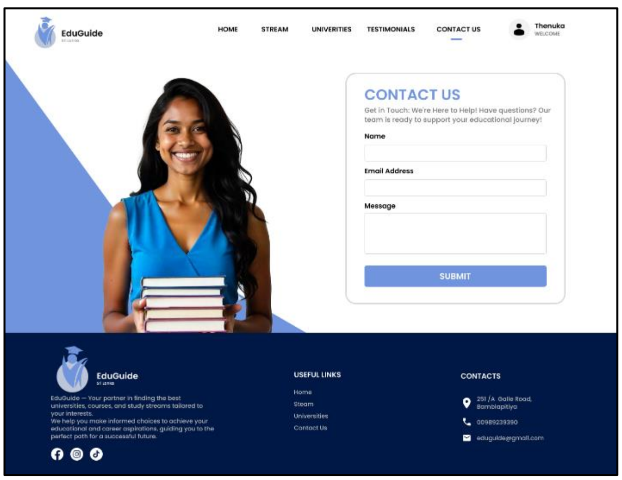
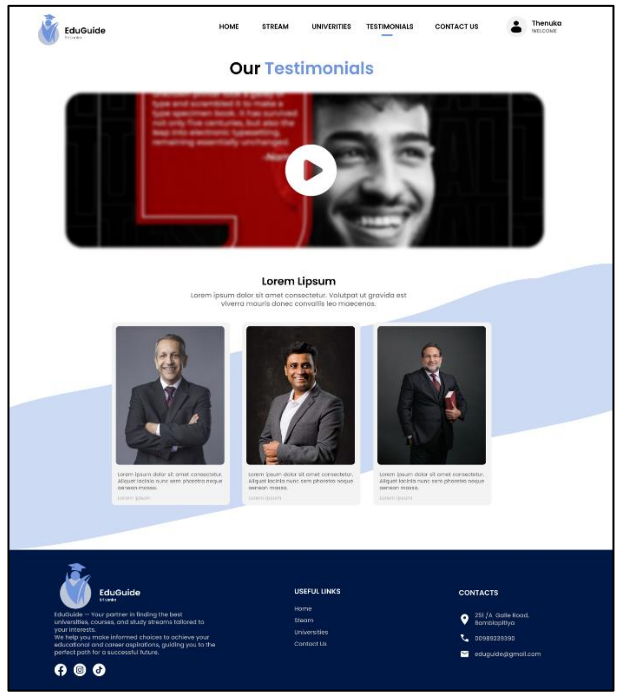


Figure 32: Low Fidelity Wireframe - University Guidance Chatbot (self-composed)

C.2 High Fidelity Prototype







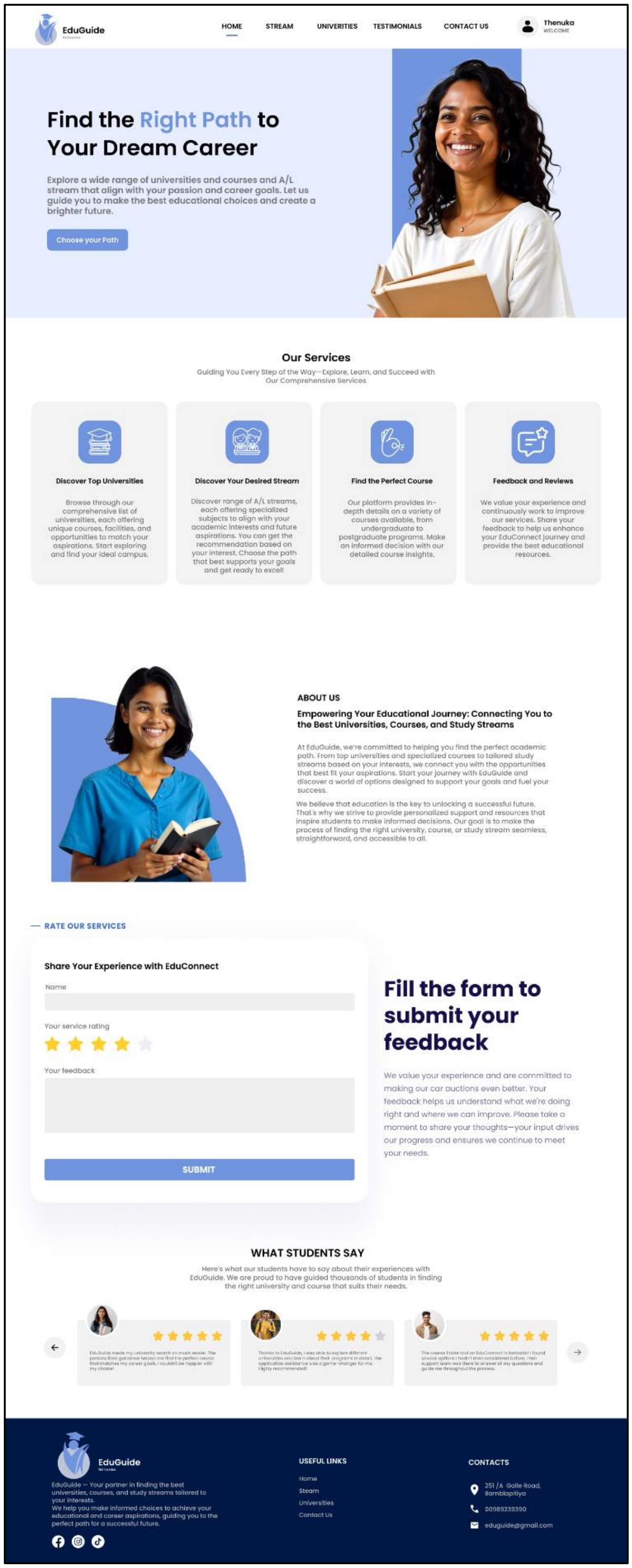


Figure 33: High Fidelity Wireframes (Register, Home, Contact Us, Feedback pages)

APPENDIX D – Implementation

Data Preparation and Keyword Mapping

Favorite Subject Keyword Grouping

I created a keyword dictionary that includes categorized subject-related terms. The code scans through each student's favorite subject and:

- Identifies which subject category it belongs to (e.g. 'Math', 'Science', 'ICT').
- When it finds a matched one, then the system determines under which category the keyword should fall.
- The corresponding category is moved to a new column: Favorite Subject. This structured format simplifies model training.

```python
# Function to check for the presence of keyword groups
def check_keyword_group_presence(favorite_subject, keyword_group):
    for keyword in keyword_group:
        # Check if the keyword is present in the description
        if keyword in favorite_subject:
            return True
    return False
```

Figure 34: Subject Keywords Grouping - Data Preprocessing

```python
# List of keyword groups
keywords = {
    'maths': ['maths', 'mathematics', 'combined', 'math', 'pure'],
    'science': ['science', 'bio'],
    'ict': ['ict', 'computer', 'information', 'it'],
    'commerce': ['commerce', 'economics', 'accounting', 'econ', 'business', 'accounts', 'economy', 'buisness', 'account'],
    'english': ['english'],
    'history': ['history'],
    'geography': ['geography'],
    'french': ['french'],
    'agri': ['agri'],
    'dancing': ['dancing'],
    'literature': ['literature'],
    'sinhala': ['sinhala'],
    'art': ['art'],
    'japanese': ['japanese'],
    'music': ['music'],
    'tamil': ['tamil'],
    'german': ['german'],
    'technology': ['technology']
}
```

Figure 35: List of Subject Keywords - Data Preprocessing

```python
# Create a column for each keyword group to mark its presence
for feature_name, keyword_group in keywords.items():
    # The lambda function checks if any of the keywords are present in the description
    data[feature_name] = data['Favorite Subject'].apply(
    lambda desc: check_keyword_group_presence(str(desc), keyword_group)
    )
```

Figure 36: Keyword group moving to 'Favorite Subject' - Data Preprocessing

The categorization process, which is based on keywords, enables a standard format for recording student interest in subjects and makes it easier to clean and preprocess data before applying machine learning models. Similar words are grouped under one subject to increase efficiency and this helps the system to prevent confusion. The favorite subject column becomes an important part when identifying the most suitable academic stream for each student based on their interest and performance.

Career Interest Classification

As well as subjects, this section categorizes career interests into broad professional job categories with keyword groups. Careers like "Software Developer", "Data Analyst", etc. are all categorized under IT/Software Developer.

This helps the system in understanding and comparing different career paths even when students use different words to refer to them. Through these categories, the system can easily identify patterns and provide more effective recommendations on which academic stream or profession may be suitable for a student.

```python
def check_keyword_group_presence(career, keyword_group):
    """
    Check if any keyword from the keyword group is present in the career string.
    The check is case-insensitive.
    """
    career = career.lower()  # Convert career description to lowercase
    for keyword in keyword_group:
        if keyword.lower() in career:  # Convert keyword to lowercase for case-insensitive matching
            return True
    return False
```

Figure 37: Career Keywords Grouping - Data Preprocessing

```python
# List of keyword groups
keywords = {
    'IT/ Software Developer': ['it', 'computer', 's.e.', 'computing', 'ict', 'web', 'programmer', 'software', 'software engineering', 'software engineer', 'data science', 'i
    'Engineer': ['engineering', 'civil engineering', 'electronic engineering', 'mechanical engineer', 'robotics', 'engineer', 'to be a engineer', 'construction'],
    'Designer': ['fashion designer', 'uiux development', 'designing', 'photographergraphic designer', 'fashion design', 'software ui designer', 'ui designing', 'software eng
    'Lecturer/ Professor': ['maths', 'academia', 'a professor', 'a lecturer or an entrepreneur', 'lecturer', 'lecture', 'lecture engineer', 'university lecture', 'lecturing'
    'Teacher': ['educator', 'english', 'language specialist', 'teacher', 'pre school teacher', 'teaching', 'teaching tutoring', 'teaching or hygiene officer'],
    'Businessman': ['business', 'ceo', 'buisness', 'businessman', 'my business', 'a businessmen', 'starting my own business', 'having my own business', 'owner of a business'
    'Business Analysis/ Business Manager/ BIS': ['management', 'management side', 'business analysis', 'business analyst', 'business manager', 'business information systems'
    'Accountant': ['accountant', 'accountancy side', 'accounting', 'acconting'],
    'Medical Industry': ['bio', 'technical', 'doctor', 'doctor medicine', 'medical doctor', 'medical', 'clinician in the medical field', 'medical industry', 'dental surgeon'
    'Chemist': ['chemist', 'chemical field'],
    'Banker': ['to be banker', 'banking', 'banker', 'banking side'],
    'Aviation': ['aviation', 'food qualityhygiene control business management and aviation', 'aviation field', 'pilot'],
    'Finance': ['finance', 'financial', 'finance sector', 'financial analyst'],
    'Manager': ['manager', 'managing', 'program manager', 'project management', 'project manager', 'quality or laboratory manager', 'as a manager', 'project manager', 'sport
    'Lawyer': ['lawyer', 'law'],
    'Hotel Industry': ['hotel industry', 'hotel field', 'hotelier'],
    'Food Scientist': ['food scientist', 'scientist', 'in food science field', 'food technologists', 'food science', 'food industry', 'food technologist', 'food production',
    'Sportsman': ['being a sportsman cricketer', 'sports'],
    'Tourism': ['tourism', 'tourism and hospitality', 'travel guide'],
    'Agriculture Industry': ['agriculture', 'agronomist', 'agricultural economist'],
    'Laboratary Side': ['laboratary side', 'laboratory scientists', 'in a laboratory'],
    'Microbiologist' : ['microbiology', 'microbiologist'],
    'HR': ['human resources', 'hr side'],
    'Music Industry': ['music industry'],
    'Photographer': ['photographer'],
    'Other': ['actuary', 'dancing', 'motor mechanic', 'machinist', 'electrician', 'farming', 'sl army', 'tea taster', 'physiologist', 'dairy industry', 'marine biologist', '
}
```

Figure 38: List of Career Keywords - Data Preprocessing

```python
# Create a column for each keyword group to mark its presence
for feature_name, keyword_group in keywords.items():
    # The lambda function checks if any of the keywords are present in the description
    data[feature_name] = data['Career'].apply(
    lambda desc: check_keyword_group_presence(str(desc), keyword_group)
    )
```

Figure 39: Keyword group moving to 'Carrer' - Data Preprocessing

APPENDIX E – Evaluation

E.1 Evaluation Form Responses from Technical Experts

What are your impressions about the problem that the author is trying to address?

4 responses

This is a very valuable topic nowadays, especially for a country like Sri Lanka, as it helps students choose the right A/L stream and university path based on their interests and goals.

This is a common issue faced by students who have just completed their O/L studies. I believe this will serve as an excellent guiding tool to help them choose the right path for their higher education.

Thenuka is trying to solve a common problem faced by many students in Sri Lanka. After completing their O/L exams, students often don't know which A/L subject stream to choose. They don't always get proper advice that matches their interests, strengths, and future goals. This can lead to choosing the wrong subjects and struggling later on. The project plans to create a chatbot that gives personal recommendations for A/L subject streams and also helps students understand which university paths are suitable for them. It's a helpful idea that can guide students to make better choices for their future.

The problem the author is addressing is both timely and relevant, particularly within the Sri Lankan educational context. Many students face uncertainty and confusion when selecting their A/L subject streams due to a lack of structured, personalized guidance. This decision significantly impacts their future academic and career paths. By identifying this gap, the author aims to provide a much-needed solution that blends technology with personalized education planning. The focus on aligning subject streams with student interests, preferences, and long-term goals shows a deep understanding of the root cause of poor

Figure 40: Evaluation Form Responses 1

Suggestions for improving the implementation from a technical perspective?

4 responses

Consider using a more comprehensive and high quality dataset to enhance the accuracy and reliability of the recommendations

Since you are handling personal information, it's crucial to address privacy concerns—implementing data encryption at rest and in transit is a solid starting point for securing user data.

Add Data Privacy and Security Features: Ensure that user data is protected by adding proper authentication, encryption, and privacy policies.

Data Collection & Quality: Ensure that student profiling data (interests, goals, academic performance, etc.) is gathered through well-structured, validated questionnaires to improve recommendation accuracy. Integrating psychometric or interest-based tests (like Holland Codes) could also provide deeper insights.

Recommendation Engine: Consider implementing a hybrid recommendation system that combines rule-based logic with machine learning (e.g., decision trees, clustering, or collaborative filtering). This will allow the system to adapt over time and improve with user feedback.

Figure 41: Evaluation Form Responses 2

Any general feedback about the project?

3 responses

Overall, the concept of this project will be beneficial for students seeking the most suitable A/L stream that aligns with their strengths and future goals.

Overall, this is a very thoughtful and well-planned project. It focuses on a real problem that many students face , not knowing which A/L subjects to choose or how to plan for university. The idea of using a chatbot to give personal advice makes the process easier and more accessible for students.

The system already covers the most important parts, and the future improvements show that the Thenuka is thinking ahead. Features like multi-language support and visual tools will make it even more helpful for students.

It's a smart use of technology to support education, and with some more development, this project could make a big difference in the lives of many students. Well done.

Overall, this is a well-conceived and socially impactful project that addresses a significant gap in Sri Lanka's education system. The concept of using AI or smart decision-support systems to guide students in making one of their earliest and most critical academic choices is commendable. It reflects a deep understanding of local challenges faced by students and the potential for technology to bridge those gaps.

Figure 42: Evaluation Form Responses 3

APPENDIX F – Dataset Publication

The author published the dataset on kaggle.com but it's reviewing still.

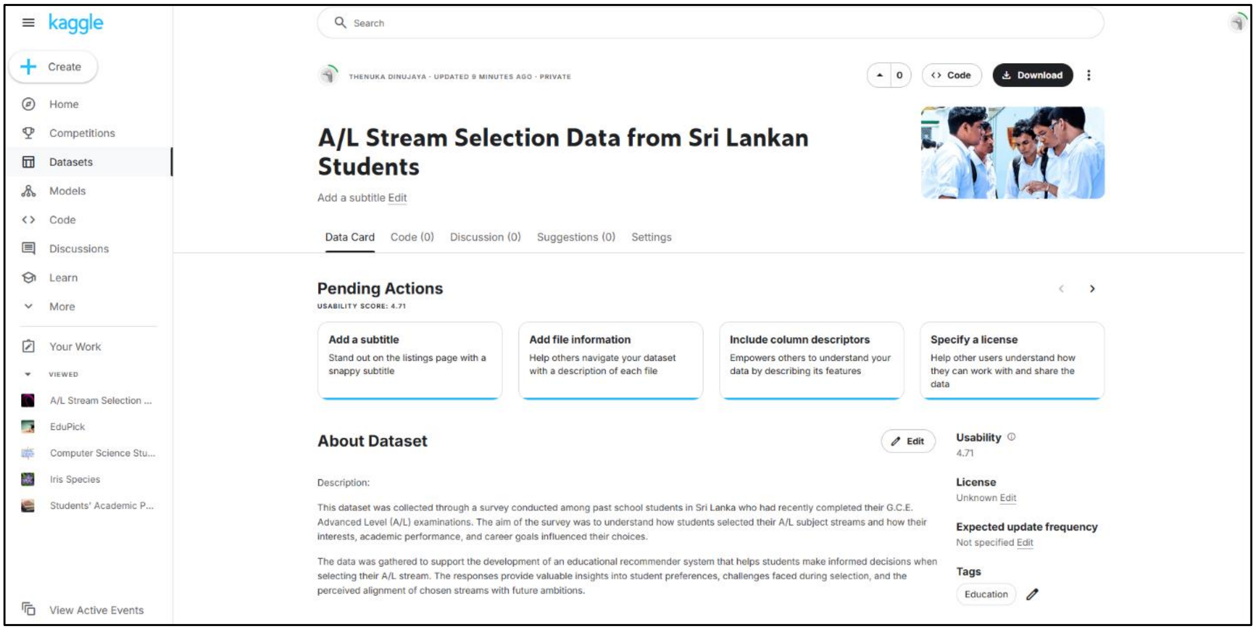


Figure 43: Dataset Publication