A

Major Project Report

on

COURSE PREREQUISITE ANALYSIS AND SKILL RECOMMENDATION SYSTEM

Submitted in Partial Fulfillment of
the Requirements for the Degree
of

Bachelor of Engineering

in

Computer Engineering

to

**Kavayitri Bahinabai Chaudhari
North Maharashtra University, Jalgaon**

Submitted by

**Kunal G. Patil
Bhavesh T. Devare
Himanshu V. Patil
Pranav S. Patil**

Under the Guidance of

Dr. Manoj E. Patil

DEPARTMENT OF COMPUTER ENGINEERING
SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY,
BAMBHORI, JALGAON - 425 001 (MS)
2023 - 2024

SSBT's COLLEGE OF ENGINEERING AND TECHNOLOGY, BAMBHORI, JALGAON - 425 001 (MS)

DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the major project entitled *Course Prerequisite Analysis and Skill Recommendation System* , submitted by

Kunal G. Patil
Bhavesh T. Devare
Himanshu V. Patil
Pranav S. Patil

in partial fulfillment of the degree of *Bachelor of Engineering* in *Computer Engineering* has been satisfactorily carried out under my guidance as per the requirement of Kavayitri Bahinabai Chaudhari North Maharashtra University, Jalgaon.

Date: 25 April 24
Place: Jalgaon

Dr. Manoj E. Patil
Guide

Dr. Manoj E. Patil
Head

Prof. Dr. G.K.Patnaik
Principal

Acknowledgements

We express our heartfelt gratitude to all who played a vital role in the successful completion of this project report. Our sincere thanks to the Almighty God for giving us the strength and resilience throughout this journey. Special appreciation to Principal Prof. Dr.Girish. K. Patnaik for providing the necessary resources for the project. We are thankful to all who contributed to this report. Special thanks to Prof. Dr. Manoj E. Patil, Head of the Department and Project Guide, for the support, insightful suggestions, and encouragement that were crucial in crafting this report. We also appreciate Prof. Dr. Manoj E. Patil's dedicated efforts in guiding us towards our goal and providing continuous encouragement to maintain progress. Our sincere thanks go to the Project In-charge for her valuable suggestions and guidance. We also acknowledge the contributions of other supervisors whose feedback enhanced our presentation skills. Finally, our profound thanks to our parents and friends, whose unwavering support was indispensable for the successful completion of this project.

Kunal G. Patil
Bhavesh T. Devare
Himanshu V. Patil
Pranav S. Patil

Contents

List of Figures

List of Tables

Abstract

In our rapidly changing world, where learning is increasingly personalized and easily accessible via the internet, there is a need for an AI NLP-based career guidance system. Traditional methods often fail to provide the personalized insights required to navigate the complexities of today's industries. This proposed system introduces a solution using AI and machine learning, offering a comprehensive approach to career guidance. It analyzes students' qualifications and technological knowledge, identifies individual skill gaps, and recommends relevant learning resources while considering industry trends for users' desired courses. By integrating these functionalities, this solution addresses the growing demand for precise, adaptable, and forward-looking career guidance.

Chapter 1

Introduction

In India, a major issue is the extensive and varied educational landscape, causing confusion and uncertainty among students about suitable courses and career paths. The sheer volume of syllabus content worsens this problem. However, by using Artificial Intelligence and Machine Learning, we can transform education. Our proposed system not only aims to simplify the maze of educational boards' syllabi but also suggests customized skills and courses, providing a personalized roadmap for each student. The proposed system addresses the challenges in Indian education and creates a more guided and empowered learning experience.

The organization of a chapter is as follows. Section 1.1 shows the Background of the project. The motivation of the project is represented in Section 1.2 and Scope is described in Section 1.3. Sections 1.4 and 1.5 describe the objective and Selection of the Life cycle mode respectively. Section 1.6 Show the Organization Of the Report. The Summary is mentioned in Section 1.7

1.1 Background

The new Course Prerequisite Analysis and Skill Recommendation System is a big change in education. It's different from the old way of picking courses because it uses technology like data analytics and machine learning. This system helps students choose the right courses by looking at their skills, academic history, and career goals. It gives personalized advice, making sure students pick courses that have the right prerequisites and match their needs. This system represents a major shift in education, focusing on personalized, data-driven choices for academic and professional success.

1.2 Motivation

The Course Prerequisite Analysis and Skill Recommendation System aims to simplify and improve education. In the ever-changing educational and professional landscape, students often struggle with many course options leading to limited choices and wasted resources. Educational institutions also face challenges in providing personalized guidance. This program uses data analytics and machine learning to give customized course recommendations based on individual skills and goals, enhancing student success and institutional efficiency, and revolutionizing the educational experience.

1.3 Problem Definition

In India, students often face confusion about which courses and careers to pursue due to the extensive educational options. The sheer volume of syllabus content adds to the complexity. Traditional methods lack personalization, leading to limited choices and inefficiencies. Educational institutions struggle to offer tailored guidance.

This project addresses the challenge by using Artificial Intelligence and Machine Learning. It aims to simplify the maze of educational boards' syllabi, offering personalized skills and course recommendations. The goal is to provide a clear path for each student, reducing confusion, and empowering better decision-making. The project seeks to make education more guided and accessible for students.

1.4 Scope

The Course Prerequisite Analysis and Skill Recommendation System covers a lot of ground, involving different aspects of education and career growth.

1.5 Objectives

- Improved Course Selection: Offers personalized course suggestions based on the unique skills and objectives of each student.
- Improve Graduation Rates: Ensure students meet required prerequisites to reduce dropouts and setbacks.
- Guide students to courses that optimize resource allocation and improve institutional outcomes.

- To provide the user with an analysis of their study to demonstrate their readiness for their goal

1.6 Selection of Life cycle model for development

Selecting the right life cycle model is crucial for the success of any project, and the choice of the Spiral Model for our project is both useful and correct. The Spiral Model is a flexible and iterative approach, perfectly suited for projects with evolving requirements and a need for constant risk assessment and management. Given the dynamic nature of our project, where requirements might change or new insights emerge, the Spiral Model allows us to accommodate these changes seamlessly. Its iterative cycles promote regular feedback, ensuring that our team can adapt to evolving needs and integrate improvements at every turn.

Our project involves many complex interactions and interdependencies that we may not fully understand at first. The Spiral Model's risk-driven approach helps us identify and address potential risks early in the development process, which is important for our project's success. By using this model, we aim to create a development process that can adapt to changes and prioritize risk management, ultimately leading to a strong and flexible solution.

The Spiral Model of the Software Development Life Cycle (SDLC) typically involves the following phases:

- Planning
- Risk Analysis
- Building the system in small, incremental parts
- Assessing the results of each iteration
- Repeating the process through multiple cycles

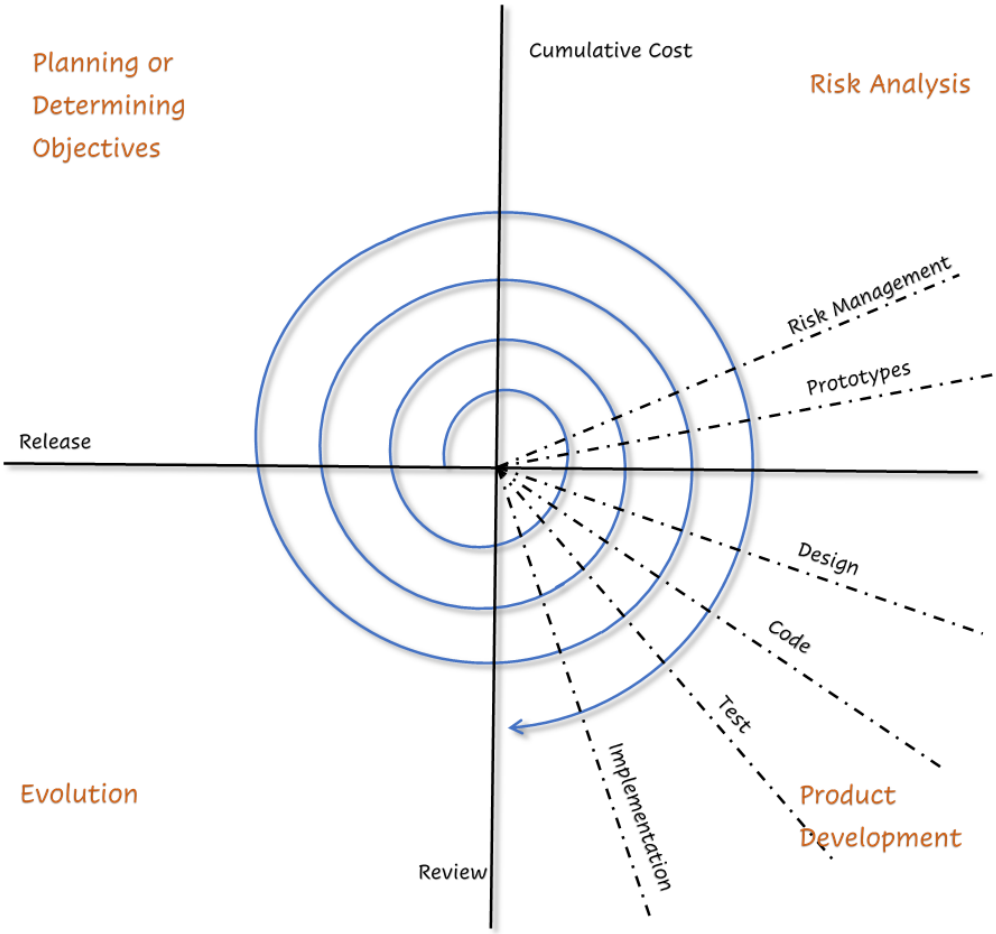


Figure 1.1: Spiral Model

1.7 Organization of Report

- **Chapter 1:** Titled "Introduction," this chapter provides details on the background, problem definition, scope, and objectives of the project. It also includes identifying the software development process model and organizing the report.

- **Chapter 2:** Project Planning and Management includes details on Feasibility Study, Risk Analysis, Project Scheduling, Effort Allocation, and Cost Estimation for the project.

- **Chapter 3:** Titled as Analysis, this chapter elaborates on Requirement Collection and Identification, H/w and S/w Requirements, Functional and Non-Functional Requirements, and a Software Requirements Specification (SRS).

- **Chapter 4**: This chapter covers system architecture designs and various UML diagrams for the project.

- **Chapter 5**: Focuses on coding/implementation details, algorithms, software, and hardware used in the development of modules in the Proposed System.

- **Chapter 6**: Discusses black box testing, identification and execution of test cases, and provides a summary of the testing process.

- **Chapter 7**: Presents the results of the project, initiates discussions on them, and highlights the advantages of the implemented system.

- **Chapter 8**: Concludes the project with a summary of findings and outlines future work that can be undertaken to enhance the system further.

1.8 Summary

In this chapter, an introduction is presented. In the next chapter project planning and management are described.

Chapter 2

Project Planning And Management

Scheduling in project management is the listing of activities, deliverables, and milestones within a project. A schedule also usually includes the planned start and finish date, duration, and resources assigned to each activity. Effective project scheduling is a critical component of successful time management.

This chapter is organized as follows.

Section 2.1 describes Feasibility Study of the project. Risk Analysis of the project selection is represented in Section 2.2. Section 2.3 represents Project Scheduling of the project. Effort Allocation of the project is described in Section 2.4. Section 2.5 describes Cost Estimation of the project. Finally, the Summary is described in last Section 2.6.

2.1 Feasibility Study

Feasibility study tells whether it is possible to develop project. Feasibility study is studied from various aspects like whether the system is feasibally, technically and economically stable. The proposed system should be accessed anywhere, anytime and should mobile. The system should be able to meet user needs and should be able to do use the resourses in the effective way.

The key consideration involved in the feasibilty study are
1. Economic Feasibility
2. Operational Feasibility
3. Technical Feasibility

2.1.1 Economical Feasibility

The Economical feasibility of Proposed System is described as follows.

1. Cost-Effective Technologies:The use of open-source and cost-effective technologies, such as Python and Visual Studio Code, helps in minimizing software development costs.

2. Return on Investment (ROI): The Course Prerequisite Analysis and Skill Recommendation System is a wise and promising investment in the field of educational technology.

2.1.2 Operational Feasibility

The proposed solution is expected to be operationally feasible and likely to improve user-system interaction. The system will support users with course recommendations and offer career guidance. It will also include all necessary functionalities to enhance operational feasibility.

2.1.3 Technical Feasibility

The technical feasibility of the Course Prerequisite Analysis and Skill Recommendation System is firmly rooted in its alignment with available technology and infrastructure. The system effectively utilizes existing hardware and software resources, reducing the need for extensive upgrades. The selected technologies, such as Artificial Intelligence and Machine Learning, are widely accessible and compatible with our project requirements. Our team has the necessary technical expertise, ensuring a smooth development process.

The system's modular architecture allows for flexibility and scalability, accommodating potential growth in users and data volume. With a focus on user-friendly interfaces and seamless integration, the system is designed to be technically feasible for both current and future needs. Overall, the technical feasibility of our system ensures a robust and sustainable solution that aligns with the technological landscape of the educational domain.

2.2 Risk Analysis

When considering risk feasibility, it's essential to address concerns related to the dataset, as it plays a crucial role in our Course Prerequisite Analysis and Skill Recommendation System. The risk of inaccurate or outdated data poses a potential challenge, emphasizing the need for a robust data verification and updating process. Ensuring the integrity of the dataset is vital to maintain the system's accuracy and reliability. Additionally, potential security risks related to the handling and storage of sensitive educational data should be addressed proactively to safeguard user privacy. By incorporating measures to verify, update, and secure the dataset, we enhance the overall risk feasibility of our system, reinforcing its capability to provide trustworthy recommendations and maintain data integrity.

2.3 Project Scheduling

Generally, project scheduling can be stated as the estimated time required for any project from its time of beginning to the end of the project. In detail, for every task, there is a dead-line because all the tasks for the completion of project are planned earlier. So that, each task is scheduled to certain time limit.Project Scheduling chart is mentioned in table 2.1.

Table 2.1: Project Scheduling Table

Task	Start Date	End Date
Selection of Project Title	08/08/23	10/08/23
Gathering Information	14/08/23	22/08/23
Analyze the Information	22/08/23	08/09/23
Made the problem Statement	26/08/23	29/08/23
Discussion with Guide	29/08/23	29/08/23
Check Scope	01/09/24	03/09/23
Check Feasibility	04/09/23	08/09/23
Design UML Diagrams	11/09/23	29/09/23
Present to Guide(Stage 1)	07/10/23	07/10/23
Discuss on Report (Stage 1)	10/10/23	30/10/23
Implement Model	27/01/24	23/02/24
Implement Test Model	26/02/24	08/03/24
Project Demo	11/03/24	11/03/24
Preparation of Report (Stage 2)	18/03/24	30/03/24
Final Presentation	16/04/24	16/04/24
Final Project Presentation and Submission	25/04/24	25/04/24

2.4 Effort Allocation

Project means teamwork, Project is developed by a combination of efforts of team. So the whole project is divided into modules and the number of modules is allotted to team members.

The modules like planning, coding, gathering information, designing, implementing, combining, etc. all the modules are distributed to each member of the group itself, and the implementation and coding of the project is done by all the members together.

Table 2.2: Effort Allocation

	Kunal Patil	Bhavesh Devare	Himanshu Patil	Pranav Patil
Topic Finalization	✓	✓	✓	✓
Existing System Analysis	✓	✓	✓	✓
Requirements Gathering	✓	✓	✓	✓
Checking Scope	✓	✓		
Checking Feasibility			✓	✓
Dataset Building	✓	✓	✓	✓
PPT Creation				✓
UML Diagram Designing		✓	✓	✓
Prototype Building	✓		✓	✓
Report Writing	✓	✓	✓	✓

2.5 Cost Estimation

The basic COCOMO estimation model is given by the following expressions:

Effort , E = a1 x $KLOC^{a2}$ PM .... Eq.(1)

Devlopment Time , Tdev = b1 x $EFFORT^{b2}$ Months .... Eq.(2)

Productivity ,P = $\frac{KLOC}{PM}$.....Eq.(3)

Where, KLOC is the estimated size of the software product indicated in a1,a2,b1, and b2 are constants for each group of software products, T dev is the estimated time to develop the software expressed in months, Effort is the total effort required to develop the software product, expressed in person-months (PMs) in table 2.3.
From eq.1 calculate the Effort required for software production.

E = a1 x $KLOC^{a2}$ PM

= 2.5 x $1323^{1.05}$

= 4737.81 PM

From eq.2 Calculate the estimated time to develop Software.

Tdev = b1 x $Effort^{b2}$PM

= 2.5 x $4737.81^{0.38}$

= 62.33 Months

From eq.3 Calculate the Productivity of the Project.

Productivity = $\frac{KLOC}{PM}$

P = 0.2793 KLOC/PM

Table 2.3: Cost Estimation

1	Total No of persons working	4 Person
2	Time taken (in months)	7 Months
3	Total Time Allocated Per Day	6 hrs
4	Actual Working Hour	1260 hrs
5	Cost Per Hour	Rs 20
6	Total Estimated Project Cost For a Person	Rs 25,200
7	Total Estimated Project Cost For 4 Person	Rs 1,00,800
8	Total Estimated Project Cost For Total Project	Rs 1,10,000

2.6 Summary

In this chapter, the project Planning and Management of the project is described. In next chapter, the project analysis is described.

Chapter 3

Analysis

This chapter is about the detailed examination of the project. Analysis simplifies the project by dividing it into different parts, which is a crucial process for development. It encompasses in-depth assessments of technical, financial, logistical, and operational aspects to ensure that the project is well-conceived and can be successfully executed.

The organization of the Chapter is as follows. Section 3.1 represents Requirement Collection and Identification. Software Requirements and Specification are described in the Section 3.2. Section 3.3 describes a summary of the chapter

3.1 Requirement Collection and Identification

In the crucial phase of Requirement Collection and Identification for the Course Prerequisite Analysis and Skill Recommendation System, our focus extends to various aspects. We actively engage with stakeholders to ensure a comprehensive understanding of user needs. Through interviews and surveys, we assess what users require from the system. Additionally, we delve into the functional requirements, detailing the essential functions and interfaces the system must have. Non-functional requirements take center stage, addressing critical aspects such as performance, security, and usability.

Moreover, we pay special attention to data requirements, specifying the sources, quality standards, and integration needs for syllabus and user data. This involves understanding the intricacies of syllabus data, ensuring its accuracy, and recognizing the significance of user data in shaping our recommendations. Technical and compliance requirements are meticulously outlined, considering infrastructure and legal regulations. The following are the crucial things we need in our proposed system.

- Syllabus data
- User data

3.2 Software Requirements Specifications (SRS)

The Software Requirements Specification (SRS) serves as a detailed blueprint, outlining user needs, functional features, and technical parameters for the Proposed system. It provides a clear roadmap for development, ensuring alignment with stakeholder expectations.

3.2.1 Product Feature

The proposed system features are mentioned below:

- The web application is designed to be fast, secure, and user-friendly..
- The user interface of a web application is more relevant.
- The website's content has been thoroughly verified and is free of plagiarism.
- The Prerequisite Analysis section assists users in receiving recommendations for skills and courses.

3.2.2 Operating Environment

The web application will operate within the following environment:

- Operating System: Windows 10 Pro/linux/Mac OS or Later.
- Python must be Installed.
- Any system with at least 8GB RAM.
- System With Minimum i3 10th Gen Processor

3.2.3 Assumption

- The system relies on the availability of the dataset so it must be accurate, reliable.
- User must input the correct lump feature for accurate prediction.
- The model's performance is expected to be consistent.
- It is assumed that the user will have a working internet connection with sufficient internet speed.

3.2.4 Functional Requirements

Functional requirements for the Proposed System include core capabilities such as user authentication, syllabus data parsing, and personalized course recommendations guides developers in implementing essential features that contribute to the system's usability and effectiveness.

- Users should be able to create accounts, log in, and access personalized recommendations based on their profiles.
- The system should efficiently extract, validate, and organize syllabus data from various sources to ensure accurate course information.
- The system should analyze user profiles and syllabus data to recommend suitable courses, considering prerequisites and individual skill gaps.
- Accuracy of the recommendation should be higher.

3.2.5 Non-Functional Requirements

Non-functional requirements for the proposed system involve aspects beyond specific functionalities. They encompass elements such as system performance, security, and user experience.

- Response Time:
  The system should respond to user requests within 3 seconds to ensure a smooth and efficient user experience.
- Availability:
  Internet service is required for this application.
- Responsibility:
  The system should have an intuitive user interface with clear navigation to facilitate ease of use for individuals with varying levels of technical expertise.
- Security:
  All user data, including syllabus information, should be encrypted and stored securely to meet privacy standards and protect sensitive information.

3.2.6 External Interfaces

- Hardware interface

  The hardware requirement includes a system with the following configurations:

  - CPU: Intel Core i5 or i7 processors from the 11th generation or newer, or AMD Ryzen 5 or 7
  - RAM: 8 GB or More
  - Keyboard and Mouse, or Pen and Touchscreen for touch-enabled interfaces
  - High-resolution monitor (4K recommended) with support for multiple displays
  - Network: Stable internet access

- Software interface

  This project is a web-based application, so users only need a browser with an internet connection. It will utilize various software components for its web-based functionality. Web servers are required to host the web application from the developers' standpoint.

  - Programming Language: Python 3.10 or higher
  - Framework: FastAPI (Python 3.10+), Django 5.0.1
  - Database: Sqlite3
  - Development Tools: Visual Studio Code, PyCharm
  - Packages: spaCy, NLTK
  - Graphics Card: DirectX 11 or later with WDDM 1.3 driver

3.3 Summary

In this chapter, analysis is presented. In the next chapter, the design is described.

Chapter 4

Design

Design phase is the activity of designing and modeling the various components of a software system. The system design provides the understanding and procedural details necessary for implementing the system. Design is helpful for a better understanding of the project. It contains UML diagrams and data flow diagrams. UML is a modeling language that is used to document object-oriented analysis and design.

The organization of the Chapter is as follows. Section 4.1 describes the system architecture of the project. Section 4.2 represents UML Diagrams (Use case Diagram, Sequence Diagram, Deployment Diagram, etc.) of the project. Finally, the Summary is described in the last Section 4.3.

4.1 System Architecture

The proposed system's architecture is based on a client-server model. A user interface allows for secure interaction with the system, requiring user profile details to generate the output analysis. The server side handles the analysis and output, while the dataset includes educational board-wise syllabus data and the source code used for output generation.

To ensure data accuracy, User profile received on backend will be cleaned using data pre-processing techniques. The required syllabus is then extracted from the dataset. They both will be tokenized and vectorized using suitable parameters and model. This will help us to produce a detailed comparative analysis with recommendations.

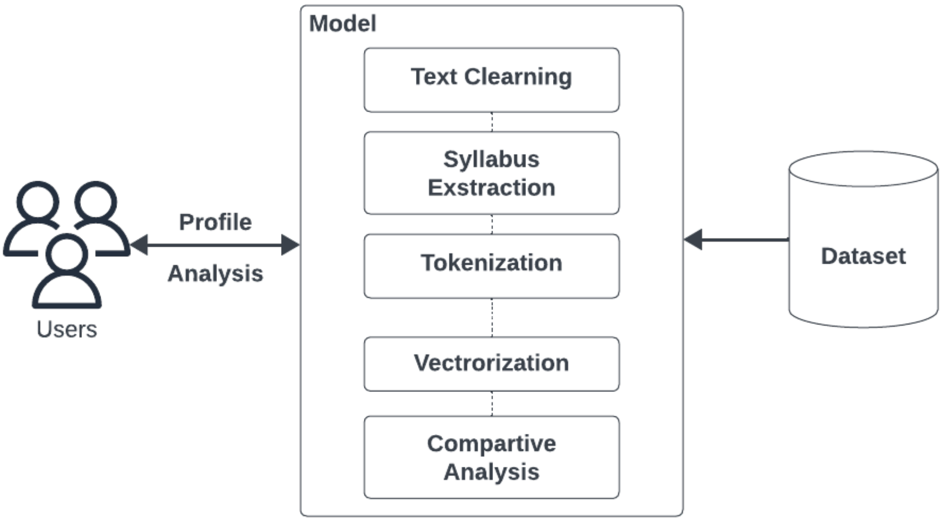


Figure 4.1: System Architecture

4.2 UML Diagrams

The UML is a language for:

- Visualizing: The structures which are transient can be represented using the UML
- Specifying: The UML addresses the specification of all the important analysis, design and implementation decisions that must be made in developing and deploying a software-intensive system
- Constructing: The UML is not a visual programming language, but its models can be directly connected to a variety of programming languages
- Documenting: The UML addresses the documentation of a system's architecture and all of its details

4.2.1 Use Case Diagram

In Figure 4.2, We have shown 2 actors Admin and User. The Admin is responsible for managing the system, including updating the dataset, authenticating users, enabling permissions, and viewing user feedback. The Admin will continuously manage and update the dataset and system to ensure smooth performance.

Users can register and log in. By completing their profiles, they gain profile access, which helps the system better understand them. User profile data will assist the system in generating better analyses. After navigating to the Analyze option, users can receive detailed recommendations based on their profiles.

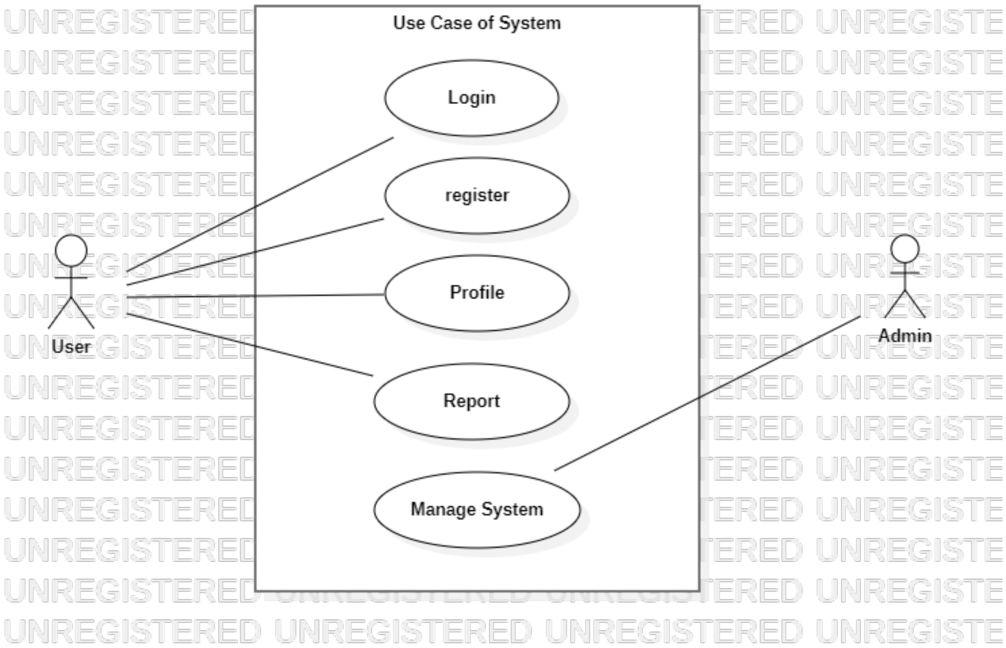


Figure 4.2: Use Case Diagram

4.2.2 Sequence Diagram

Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lifelines, different processes or objects that live simultaneously and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. figure 4.3 shows a sequence diagram for Use cases.

The program includes Interface, System, Model, and Report classes. It demonstrates the process of generating a report from the interface section. First, user inputs are received from the interface, then the system requests data from the backend. On the model side, it extracts and processes the required data, generating the analysis that will be stored in the report object. The final result is displayed on the interface for the user.

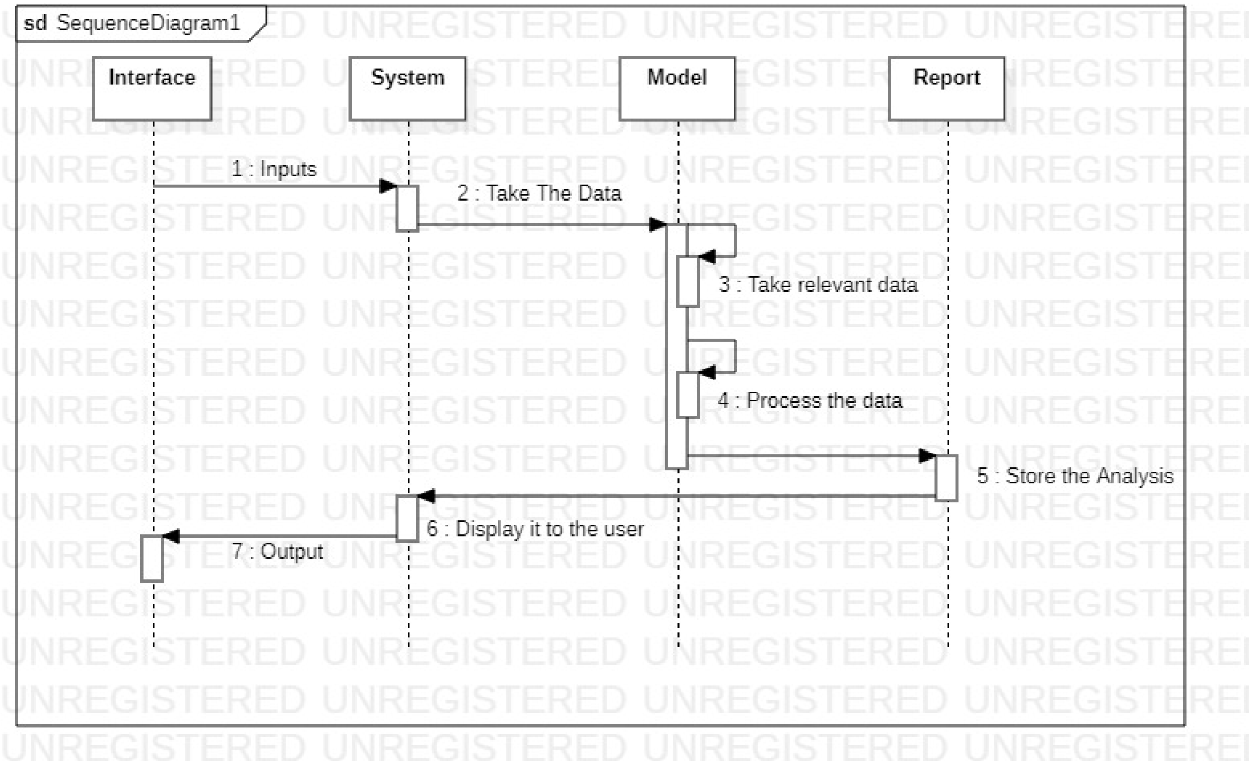


Figure 4.3: Sequence Diagram

4.2.3 Deployment Diagram

In figure 4.4, Deployment diagram model the physical architecture of a system. The below Deployment diagram shows the relationships between the software and hardware components in the system and the physical distribution of the processing. User and Admin will be connected to the main server. The model will be deployed on api server. There will be active request response cycle between api server and web server.

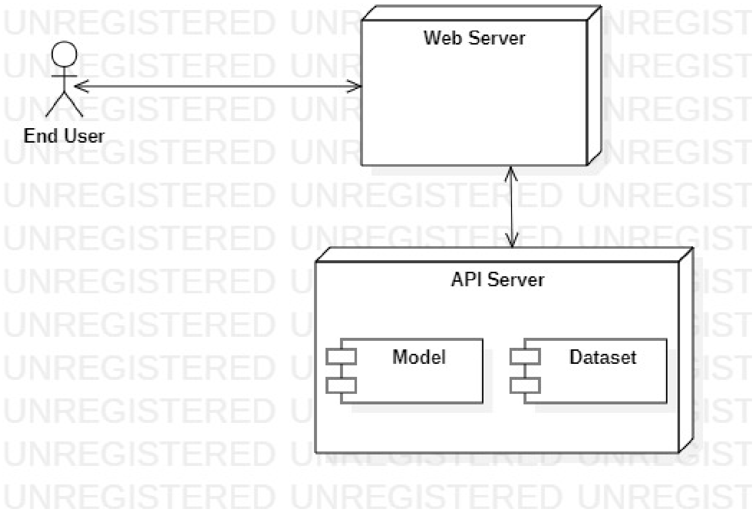


Figure 4.4: Deployment Diagram

4.3 Summary

In this chapter, system design is discussed. In the next chapter, the conclusion is presented.

Chapter 5

Coding/Implementation

The implementation phase is the lengthiest and most crucial stage in software development. Once the software design is finalized, a team of developers begins coding the design using a programming language. The software interface and its internal operations, as per the design phase, are executed in the implementation phase.

This chapter mainly contains the following sections: In Section 5.1 Algorithm is described. Software and Hardware for development in detail are described in 5.2. Section 5.3 contains a summary.

5.1 Algorithm

An algorithm in the implementation phase demonstrates the critical steps of our project's operation. It is a step-by-step procedure showing how we apply different steps to the data in order to obtain accurate output.

The following steps are applied to preprocess the data and obtain the final output.

- **Data Preprocessing:**
  - Tokenization: Splitting the input texts into individual words or tokens. This helps in to prepare the text for further processing.
  - Lowercasing: Converting all the text into lowercase. This ensures that words are treated the same regardless of their casing.
  - Removing Special Stop Words: We are removing out the null values or the values that we do not required for further processing from the input data.
  - Named Entity Recognition (NER): This step categorizes skill names or topic names under the same category for those with naming conventions like HTML, HTML5. This will be under the same category.

- Stemming or Lemmatization: This involves reducing words to their root form to improve text normalization and reduce vocabulary size.
- Vectorization: The process of converting text data into numerical vectors that machine learning models can understand and it helps in calculations.

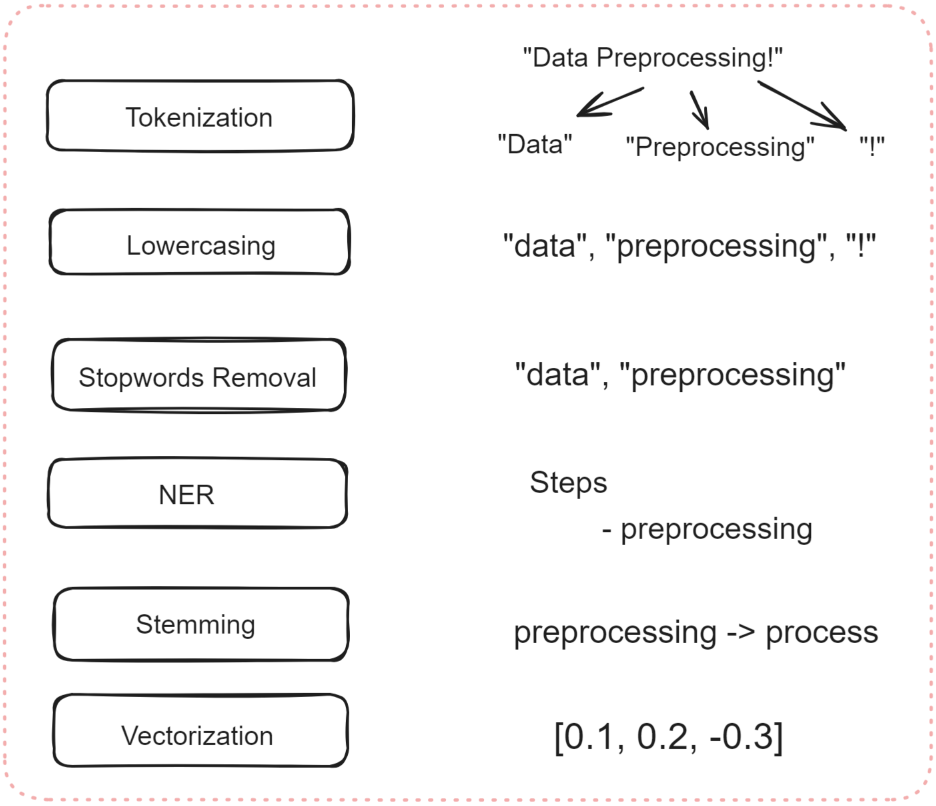


Figure 5.1: NLP Steps

■ Overall Working of Proposed System:

- Profile Completion: This is the initial step and crucial for generating further analysis. Users should first complete their profile data, including background information, education details, and their areas of interest.
- We have incorporated 2 features into our project. The first one is Prerequisite Analysis. It considers the profile data, processes it, and generates an analysis of the highest class the user is completing their prerequisites for preferred courses.
- The other one is a comparative analysis. This feature compares users' resume data with the job description to show how well the resume fits the job description.
- Finally, generate a report containing all the details analysis of the user. This analysis takes care of considering all the required factors that are essential for recommendations of skills or courses to the user based on their education journey.

■ Cosine Similarity Measure:

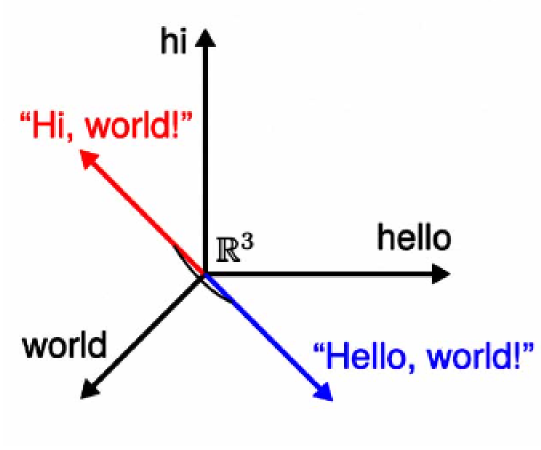


Figure 5.2: Cosine Similarity

- A Cosine Similarity Measure is a metric we are using to measure the similarity between two non-zero vectors in an inner product space. It calculates the cosine of the angle between the vectors, indicating how closely related they are.
- Each educational board or institution typically has a curriculum that specifies the skills or subjects taught at each level (e.g., class or grade). By representing these skills as vectors, where each dimension corresponds to a specific skill, you can compare the student's completed skills vector with the prerequisite skills vectors for different classes using Cosine Similarity.
- The class with the highest similarity score indicates that the level at which the students completed the skills is most similar to the prerequisites skills. This level is considered the highest class completed by the student.
- Once the highest class completed is identified, we recommend the additional skills required for a specific course or career path based on the prerequisite skills for that class. These suggested skills are those that the student has not yet acquired but are essential for advancing in their chosen field.

5.2 Software and Hardware for Development in Detail

Software:

- Frontend: HTML5, CSS3, JavaScript (ES6+)
- Backend: FastAPI (Python 3.10+), Django 5.0.1
- Database: Sqlite3

- Version Control: Git
- Development Environment: VS Code, PyCharm

Hardware:

- Processor: Intel Core i5 or i7 processors from the 11th generation or newer, or AMD Ryzen 5 or 7 processors
- RAM: 8 GB or more
- Storage: SSD (Solid State Drive) for faster read/write speeds
- Input Device: Keyboard and Mouse, or Pen and Touchscreen for touch-enabled interfaces
- Output Device: High-resolution monitor (4K recommended) with support for multiple displays
- Network: Stable internet connection for cloud-based development and collaboration

5.2.1 Modules in Proposed System

Profiles:
User profiles contain personal information, educational data, career preferences, and achievements. This data is useful for analysis and recommendations.

Report:
The report contains data generated using the dataset and user profile data. This module assists in managing analyzed data and presenting it to users.

5.3 Summary

In this chapter, the topic of implementation is discussed. The following chapter will focus on testing.

Chapter 6

Testing

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. In simple words, testing is executing a system in order to identify any gaps, errors, or missing requirements contrary to the actual requirements. In general, testing is finding out how well something works. In terms of human beings, testing tells what level of knowledge or skill has been acquired. In computer hardware and software development, testing is used at key checkpoints in the overall process to determine whether objectives are being met.

The organization of this Chapter is as follows. Section 6.1 Describes Black Box Testing. Description of the testing in Section 6.2. Section 6.3 Provides a short summary.

6.1 Black Box Testing

Black box testing has aided in identifying functionality, usability, and other feature gaps. It offers a comprehensive view of software performance and outcomes, enhancing software quality and expediting time to market. This testing approach helps in minimizing the risk of software failures for end users.

It is a Software Testing method that analyzes the functionality of a software/application without knowing much about the internal structure/design of the item that is being tested and compares the input value with the output value. The main focus of Black Box Testing is on the functionality of the system as a whole.

Advantages:

- Well-suited and efficient for large code segments.
- Code access is not required.

- Clearly separates the user's perspective from the developer's perspective through visibly defined roles.

6.2 Test Cases Identification and Execution

Test case 1:	
Purpose:	User Profile Completion
Pre-requisite:	Internet and User's Data are needed
Test data:	Personal information like name, educational background, and career preferences
Expected result:	User should receive a status of 200 and the data should be successfully saved.
Test Case Result:	PASS

Table 6.1: Test case 1

Test case 2:	
Purpose:	Pre-Requisite Analysis
Pre-requisite:	User must complete the profile setup step.
Test data:	Preferred Course and Profile Data
Expected result:	Formatted analysis should be generated.
Test Case Result:	PASS

Table 6.2: Test case 2

Test case 3:	
Purpose:	Comparative Analysis
Pre-requisite:	Resume in PDF format
Test data:	Skills Data extracted from a PDF
Expected result:	Resume Score should get generated
Test Case Result:	PASS

Table 6.3: Test case 3

6.3 Summary

In this chapter, the system testing details and testing environment are described. The following chapter will present the results and discussion.

Chapter 7

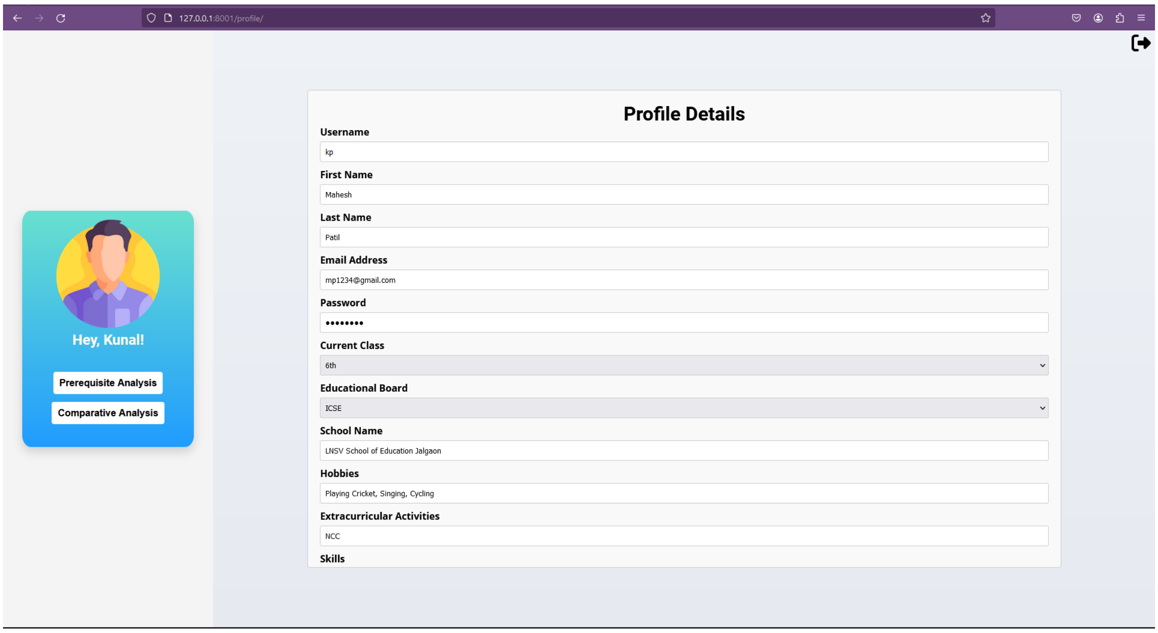
Result and Discussion

The results section is a section containing a description of the main findings of a research, whereas the discussion section interprets the results for readers and provides the significance of the finding. Writing the results and discussion as separate sections allows you to focus first on what results you obtained and set out clearly what happened in your experiments and investigations without worrying about their implication.

In this chapter, there are 3 Sections in which Section 7.1 describes the result. Discussion is presented in Section 7.2. Section 7.3 contains a summary.

7.1 Result

These are the results after the completion of the proposed system.



Profile Details

Username: kp

First Name: Mahesh

Last Name: Patil

Email Address: mp1234@gmail.com

Password: ••••••••

Current Class: 6th

Educational Board: ICSE

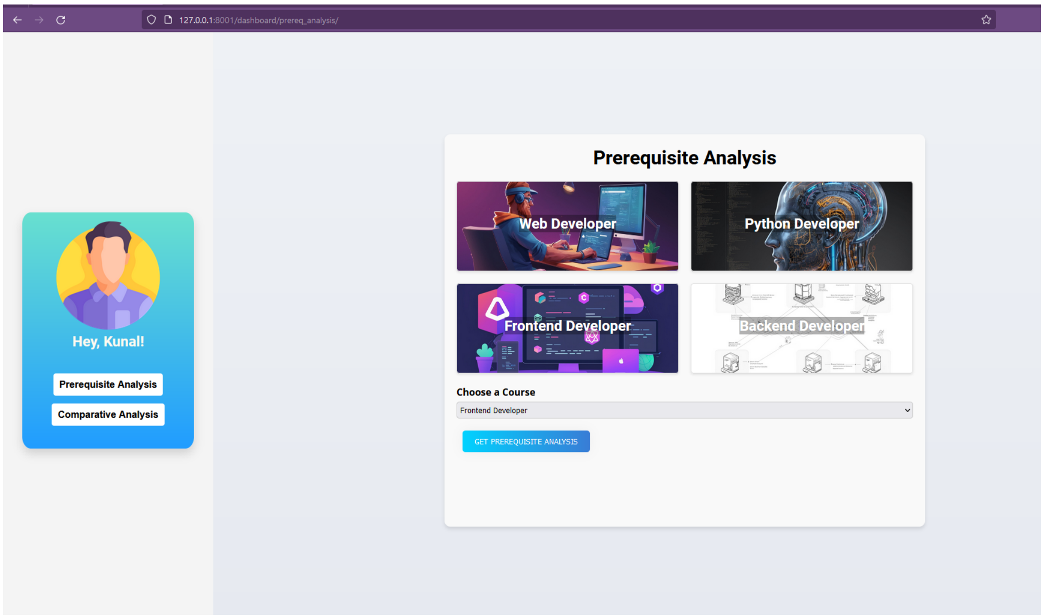
School Name: LNSV School of Education Jalgaon

Hobbies: Playing Cricket, Singing, Cycling

Extracurricular Activities: NCC

Skills

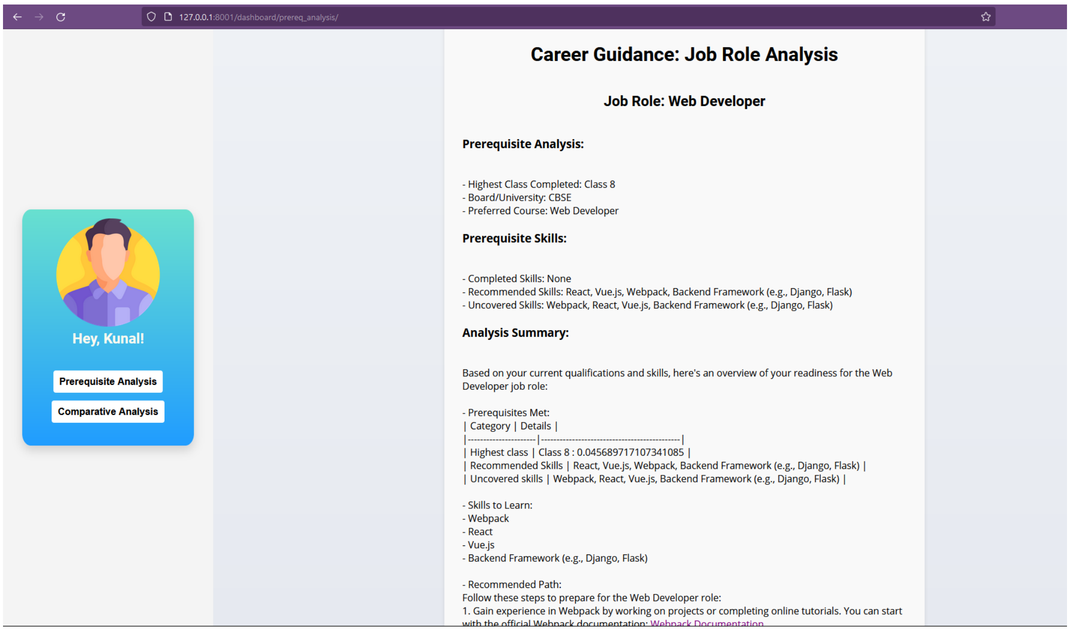
Figure 7.1: Students Profile Page



Prerequisite Analysis

Choose a Course: Frontend Developer

Figure 7.2: Prerequisite Analysis



Career Guidance: Job Role Analysis

Job Role: Web Developer

Prerequisite Analysis:

- Highest Class Completed: Class 8
- Board/University: CBSE
- Preferred Course: Web Developer

Prerequisite Skills:

- Completed Skills: None
- Recommended Skills: React, Vue.js, Webpack, Backend Framework (e.g., Django, Flask)
- Uncovered Skills: Webpack, React, Vue.js, Backend Framework (e.g., Django, Flask)

Analysis Summary:

Based on your current qualifications and skills, here's an overview of your readiness for the Web Developer job role:

```
- Prerequisites Met:
| Category | Details |
|--------------------|------------------------------------------|
| Highest class | Class 8 : 0.045689717107341085 |
| Recommended Skills | React, Vue.js, Webpack, Backend Framework (e.g., Django, Flask) |
| Uncovered skills | Webpack, React, Vue.js, Backend Framework (e.g., Django, Flask) |
```

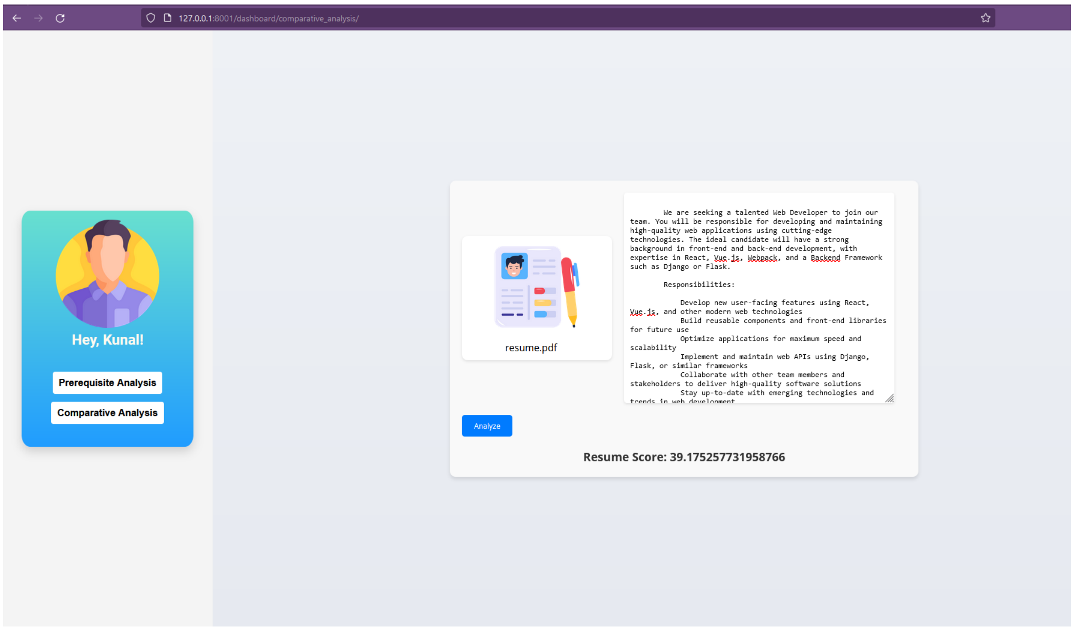
- Skills to Learn:
- Webpack
- React
- Vue.js
- Backend Framework (e.g., Django, Flask)

- Recommended Path:

Follow these steps to prepare for the Web Developer role:

1. Gain experience in Webpack by working on projects or completing online tutorials. You can start with the official Webpack documentation: Webpack Documentation

Figure 7.3: Prerequsite Analysis Report



We are seeking a talented Web Developer to join our team. You will be responsible for developing and maintaining high-quality web applications using cutting-edge technologies. The ideal candidate will have a strong background in front-end and back-end development, with expertise in React, Vue.js, Webpack, and a Backend Framework such as Django or Flask.

Responsibilities:

Develop new user-facing features using React, Vue.js, and other modern web technologies
Build reusable components and front-end libraries for future use
Optimize applications for maximum speed and scalability
Implement and maintain web APIs using Django, Flask, or similar frameworks
Collaborate with other team members and stakeholders to deliver high-quality software solutions
Stay up-to-date with emerging technologies and trends in web development

Resume Score: 39.175257731958766

Figure 7.4: Comparative Analysis

7.2 Discussion

The proposed solution provides easy-to-understand analysis for students, helping them plan their academic journey effectively. It offers a user-friendly interface for a positive experience. By using Natural Language Processing techniques like TF-IDF vectors and cosine similarity, the system compares course requirements and student skills, suggesting a roadmap for primary and secondary students. These machine learning techniques enable personalized recommendations, bridging the gap for students and guiding them towards informed decisions.

7.2.1 Advantages

- Personalized Learning Path : The system can propose a personalized learning path to enhance the student's knowledge and skills. This can aid students in advancing more efficiently and effectively in their academic or career endeavors.

- Early Identification of Potential: By analyzing a student's learning patterns and academic performance, the system can identify potential talents or areas of interest that the student may not have recognized. This early identification can guide the student towards opportunities that align with their strengths.

- Improved Academic and Career Planning: The system can help students plan their academic and career paths more strategically by analyzing their past learnings and academic subjects to suggest courses, internships, or extracurricular activities that align with their goals.

- Enhanced Motivation and Engagement: Personalized recommendations can enhance student motivation and engagement by making their learning experiences more relevant and meaningful. When students recognize the direct link between their past accomplishments and future prospects, they are more likely to remain motivated and focused.

Chapter 8

Conclusion and Future Work

This chapter presents the conclusion and future work for the proposed system.

The organization of this chapter is as follows. Section 8.1 represents the Conclusion of the proposed system, Section 8.2 discusses the future work for the system.

8.1 Conclusion

In conclusion, the proposed system offers a comprehensive solution to the course selection process by leveraging technology and data analysis to recommend essential skills for students. By streamlining course selection and aligning skills with course requirements, this system not only simplifies registration but also enhances student success. As technology continues to advance, the integration of artificial intelligence and machine learning will further enhance the precision and customization of the system, providing students with a personalized and adaptable learning experience. Implementing this proposed system represents a significant step towards a more tailored and effective educational approach.

8.2 Future Work

To improve the system, we can work on making the skill suggestions more accurate by refining our algorithms. We'll also gather feedback from students and educators to make sure the suggestions are helpful. Adding more data sources and courses will give a broader range of skills to match, making the suggestions even more useful. Finally, we can integrate real-time job market data to help students see which skills are in demand, making their education more relevant to today's job market.

Bibliography

[1] Derick Prince B, Madhan K, Vishwa K, Yamunathangam D. *Job and Course Recommendation System using Collaborative Filtering and Naive Bayes algorithms.* In: *2023 2nd International Conference on Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA).* 2023. Available at: `https://ieeexplore.ieee.org/document/9586143`

[2] Susan Sajadi, Olivia Ryan, Lisa Schibelius, Mark Huerta. *Using Generative AI to Assist in Individual Performance Feedback for Engineering Student Teams.* In: *2023 IEEE Frontiers in Education Conference (FIE).* 2023. Available at: `https://ieeexplore.ieee.org/document/9528851`

[3] *Data gathering from various sites.* In: *Byju's, Vedantu, CBSE Board.* 2023.

[4] Software Engineering: A Practitioner's Approach by Roger S. Pressman, McGraw Hill, *7th Edition.* 2010. Available at: `https://www.amazon.com/Software-Engineering-Practitioners-Roger-Pressman/dp/0078022126`

[5] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software.* Addison-Wesley, 1994. Available at: `https://www.amazon.com/Design-Patterns-Object-Oriented-Addison-Wesley-Professional-ebook-dp-B000SEIBB8/dp/B000SEIBB8/`

[6] Mike Cohn. *Agile Estimating and Planning.* Prentice Hall, 2005. Available at: `https://www.amazon.com/Agile-Estimating-Planning-Mike-Cohn/dp/0131479415`

Index