**Software Requirements Specification**

**for**

**EduInsight**

**Student Performance Analysis & Learning Strategies Recommendation System**

**Prepared by**

**Aysha Sanam Ummer**

**Bettina Mariam Biju**

**Fathima Zanha K**

**Safuan P Anver**

**Batch 2 / Team 2**

**Guide : Dr. Cerene Mariam Abraham**

**26.02.2024**

**Table of Contents**

**1. Introduction 3**

1.1 Purpose 3 1.2 Intended Audience 3 1.3 Product Scope 4

1.4 References 4

**2. Overall Description 5**

2.1 Product Perspective 5

2.2 Product Functions 5

2.3 User Classes and Characteristics 5

2.4 Operating Environment 5

2.5 Design and Implementation Constraints 6

**3. External Interface Requirements 7** 3.1 User Interface 7

3.2 Hardware Interfaces 7

3.3 Software Interfaces 8

**4.System Features 8**

4.1 Comprehensive Student Profiling 9

4.2 Performance Analysis 9

4.3 Prediction and Recommendation 9

4.4 Was it helpful Feature 10

**5. Other Nonfunctional Requirements 10**

5.1 Performance Requirements 10

5.2 Safety Requirements 10

5.3 Security Requirements 11

5.4 Software Quality Attributes 11

5.5 Business Rules 11

**1. Introduction**

**1.1 Purpose**

The purpose of our software is to revolutionize the way student performance is analyzed and improved. By leveraging advanced data analytics and machine learning techniques, our platform aims to provide personalized recommendations tailored to each student's unique learning style and needs.

Through comprehensive performance analysis, our software will identify patterns and trends in students' academic progress. This includes factors such as strengths and weaknesses in different subjects, preferred learning modalities, and optimal study habits. With this information, our platform will generate targeted recommendations for optimizing study schedules, and employing effective study techniques.

By empowering students with actionable insights and personalized guidance, our software aims to enhance academic performance, increase motivation, and foster a culture of continuous improvement in learning outcomes. Ultimately, our goal is to support students in their mission to success and achievement.

**1.2 Intended Audience**

The intended audience for our software primarily consists of students across various educational levels, from primary school to higher education institutions. Specifically, the software is designed to cater to:

Students: The software serves as a valuable tool for students seeking to enhance their academic performance. By providing personalized recommendations on study techniques and time management strategies, students can optimize their learning experience and achieve better results in their studies.

Overall, the application has the potential to benefit a wide range of students by improving their performance

**1.3 Product Scope**

The product scope for our software, designed to enhance student performance through personalized recommendations and analysis, encompasses the following features and functionalities:

1. Performance Tracking and Analysis: The software will enable students to track their academic performance across various subjects and assessments. It will analyze performance data to identify trends, strengths, and areas for improvement.
2. Personalized Recommendations: Based on the analysis of performance data, the software will generate personalized recommendations for students. These recommendations may include suggested study techniques and time management strategies tailored to each student's learning style and needs.
3. Machine Learning Algorithms: The core functionality of the software will leverage machine learning algorithms to analyze the collected performance data. These algorithms will identify patterns, correlations, and trends in students' academic performance.
4. Predictive Analytics: Using machine learning models, the software will provide predictive analytics to forecast future academic performance based on historical data. This will enable early intervention and personalized support for students at risk of falling behind
5. User-Friendly Interface: The software will feature a user-friendly interface designed for easy navigation and accessibility. Students can easily input data, view recommendations, and track their progress without requiring extensive technical knowledge.

Overall, the product scope aims to empower students with personalized

recommendations and insights to optimize their learning experience,

improve academic performance, and achieve their educational goals.

**1.4 References**

1. Amrieh, E. A., Hamtini, T., & Aljarah, I. (2016). Mining educational data to predict student’s academic performance using ensemble methods. *International journal of database theory and application*, *9*(8)
2. Pashler, H., McDaniel, M., Rohrer, D., & Bjork, R. (2008). Learning styles: Concepts and evidence. *Psychological science in the public interest*,
3. van Sluijs, M., & Matzat, U. (2023). Predicting time‐management skills from learning analytics. *Journal of Computer Assisted Learning*.

**2. Overall Description**

**2.1 Product Perspective**

The proposed system is inspired from International Journal of Database Theory and Application Vol 9 - Mining Educational Data to Predict Student’s Academic Performance using Ensemble Methods. The project focuses on adding a few additional features to the same, to make it more functional and efficient for the users.

**2.2 Product Functions**

The core functions of the proposed upgrade includes :

* **Performance Classification :** Classifies students into different performance categories (e.g., high achievers, average performers, at-risk students) based on predictive models and predefined criteria.
* **Personalized Recommendation Engine :** Recommends personalized learning resources, study materials, and extracurricular activities tailored to individual student needs and learning styles using collaborative filtering and content-based recommendation systems.

**2.3 User Classes and Characteristics**

The proposed system provides services to students who are the primary user.

* **User module:** In the user module the student can give their input from which personalized recommendations will be provided.

**2.4 Operating Environment**

The software requirements for the proposed system are :

* **Operating System:**  Windows
* **Web Server:** Apache HTTP Server, Nginx, or Microsoft Internet Information Services.
* **Programming Language :** Backend: Python , Frontend: HTML, CSS, JavaScript

Data Analysis: Python (with libraries like Pandas, NumPy, SciPy)

* **Machine Learning Libraries:** TensorFlow, PyTorch, Scikit-learn (for building and training machine learning models). Pandas, NumPy (for data manipulation and preprocessing)
* **Development Frameworks:** Django, Flask (for backend development in Python).

**2.5 Design and Implementation Constraints**

**Design constraints:**

* The interface should be intuitive, easy to navigate, and user-friendly to accommodate users with varying levels of technical proficiency and educational backgrounds.
* The predictive models should achieve satisfactory performance metrics such as accuracy, precision, recall, and F1-score to provide reliable predictions and recommendations.

**Implementation constraints:**

* The effectiveness of the system relies on the availability and quality of student performance data. Inconsistent data collection methods, incomplete records, and data silos can hinder the accuracy of predictions and recommendations.
* As the number of users and data volume grows, the system should be able to scale efficiently to handle increased loads without sacrificing performance.
* Ensuring user acceptance and adoption of the system among students. user-friendly interfaces, clear documentation, and training sessions can facilitate the adoption process.
* The system requires specific software infrastructure, including databases, analytics tools, and machine learning libraries. Ensuring compatibility, reliability, and performance of the infrastructure is essential for system functionality.

**3.1 User Interfaces:**

**Web Interface:**

* The primary user interface for EduInsight will be a dynamic web application.
* Developed using Django, HTML, CSS, and JavaScript.
* Intuitive and user-friendly design for easy navigation.
* Dashboard displaying individual student profiles, academic insights, and recommendations.

**API Interface:**

* An API created using Flask to facilitate communication between different components.
* Well-documented API endpoints for external integration if required.

**User Authentication:**

* Secure login and registration system to ensure data privacy.
* Access control to distinguish between student, teacher, and administrator roles.

**Feedback Mechanism:**

* A feedback system allowing students to provide insights on the effectiveness of recommendations.
* Interface for administrators to review and analyze feedback.

**3.2 Hardware Interfaces:**

**Server Requirements:**

* Specification for the server hardware where the machine learning models are deployed.
* Consideration for server capacity and scalability to handle increasing data and user load.

**Client Devices:**

* Compatibility requirements for various client devices such as laptops, tablets, and smartphones.
* Browser compatibility specifications for optimal performance.

**3.3 Software Interfaces:**

**API Communication:**

**Endpoint Specification:**

* Define API endpoints, methods, and response formats.
* Include details for retrieving specific resource information and searching resources.

**Integration Guidelines:**

* Provide guidelines for seamless third-party integration.
* Cover authentication, authorization, error handling, pagination, rate limiting, data formats, CORS, versioning, documentation, testing environment, feedback, and security measures.

**URL Support:**

**Resource URL Structure:** Specify the URL structure for managing resources. Include path parameters and HTTP methods.

**Integration Considerations:** Discuss considerations related to URL support.

**4. System Features**

This section provides a comprehensive overview of the specific functional requirements designed to enhance the user experience within the existing EduInsight system to improve user interaction by seamlessly integrating the following features into the system.

**4.1 Comprehensive student profiling**

* **Description:**The system constructs comprehensive profiles for each student based on the processed data. These profiles provide a holistic view of individual learning behaviors, enabling personalized insights and recommendations.
* **Stimulus/Response:**The student enters the data and these sources are processed by EduInsight. This includes academic grades, attendance records, participation levels of each student, extracurricular activities, assessment scores, and feedback from the students.The response is the comprehensive profile generated by EduInsight in a holistic view manner.
* **Functional Requirements:**The system must collect and integrate relevant data and develop algorithms to generate comprehensive student profiles.

**4.2 Performance Analysis**

* **Description:** EduInsight utilizes machine learning algorithms to analyze diverse datasets, such as academic grades, attendance records, and participation levels. This allows for the identification of patterns, correlations, and trends within the data.
* **Stimulus/Response**:When the student enters the data, machine learning algorithms identify patterns in academic data, recognizing trends and correlations.The response involves generating insights into individual and collective learning behaviors based on the analyzed datasets.
* **Functional requirements:**Implement machine learning algorithms capable of handling various types of educational data.Also,develop mechanisms to identify and select relevant features for analysis within the datasets.

**4.3 Prediction and Recommendation**

* **Description:**The recommendation engine translates the analysis into actionable steps, offering tailored suggestions for study techniques and time management strategies. This feature aims to support students in optimizing their learning approach.
* **Stimulus/Response:**The recommendation engine generates personalized suggestions for study techniques and time management strategies from the analyzed data that the student can take to optimize their learning approach.
* **Functional Requirements:**Develop algorithms capable of translating complex analysis into clear and actionable recommendations.The recommendation engine must seamlessly integrate with the machine learning analysis to utilize the insights generated.

**4.4 Was it helpful Feature**

* **Description:**The system gives an additional feature where it takes feedback from the user asking if the profile generated and the techniques recommended are to their liking and if they are useful.
* **Stimulus/Response**:Once the system displays the output the feedback form will pop and the user can input necessary information.
* **Functional Requirements:**The system must be robust enough to incorporate any modifications required.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

The performance requirements for this system necessitate efficient processing of diverse datasets, rapid execution of machine learning algorithms for real-time pattern recognition, and seamless integration of the recommendation engine into the dynamic web interface. Additionally, the system should exhibit low-latency response times, ensuring timely delivery of personalized study suggestions and adaptability to evolving student performance data.

**5.2 Safety Requirements**

The system prioritizes student data security and privacy. Regular security audits and updates are conducted to address vulnerabilities and maintain system integrity.

**5.3 Security Requirements**

Security measures for the system include robust data encryption for secure transmission and storage of sensitive student information. Conduct regular security audits and updates to address emerging threats and vulnerabilities, ensuring continuous resilience against potential cyber attacks.

**5.4 Software Quality Attributes**

The need of these attributes are a must:

* Ensures reliable performance through machine learning algorithms.
* Showcase scalability with advanced data analytics, machine learning, and a dynamic web interface.

**5.5 Business Rules**

* **Users of the system:** The system supports distinct user roles for students.
* **Privacy:** The system keeps student information private and secure by restricting access and using strong data encryption.
* **Service Availability:** The system is always available, providing secure and uninterrupted access for students.
* **User Training:** Learn how to use the system by understanding how to enter data, interpret personalized suggestions, and navigate the user-friendly interface for a better educational experience.