Logo

Description automatically generatedA picture containing diagram

Description automatically generated

*Kingdom of Saudi Arabia*

*Ministry of Education*

*King Faisal University*

***College of Computer Sciences & Information Technology***

**[An AI-Driven Major Decision Support Tool for Students]**

*A project submitted as a requirement for the course “Artificial Intelligence-CS-411”*

By:

**Section 2**

Talal Al-Qahtani - 218036092

Ahmed Al-Nuwaihel - 221425928

To Laboratory Instructor:

**Mr. Marwan El-Haj**

**Course Instructor**

**Prof. Alaa Alsagheer**

Submission Date: 14/12/2021

Table of Contents

1. [Abstract 3](#_Toc184577459)
2. [Methodology 3](#_Toc184577460)
3. [Implementation Details and Result 5](#_Toc184577461)

3.1 [Diagram to show implementation overview 4](#_Toc184577462)

[3.2 Implementation Discussion (Processes and Operations) 5](#_Toc184577463)

1. [User Interface 7](#_Toc184577464)
2. [Conclusion 8](#_Toc184577465)
3. [References 9](#_Toc184577466)
4. Abstract

The project takes us a step closer to designing an AI-Driven Major Decision Support Tool which will aid students in selecting an academic major most appropriate to them. The respondents’ individual answers to this set of questions are used as a basis for generating suggestions geared towards one’s interests, high school GPA and entrance exam scores.

Uncertainty and diversity of options are two of the most hindering complexity students face while choosing their majors. This could result in frustration, a lot of changes in majors or poor career fit. A device facilitating this decision making process is much needed.

The tool aims at pupils who require assistance in choosing a career path which is suitable for them. It provides specific options but is meant to support and not substitute academic advising.

1. Methodology

The **frontend** of the system was developed using HTML, CSS, and JavaScript, creating a user-friendly interface for students to input their responses. The design ensures simplicity and accessibility, making it easy for users to interact with the tool and receive recommendations.

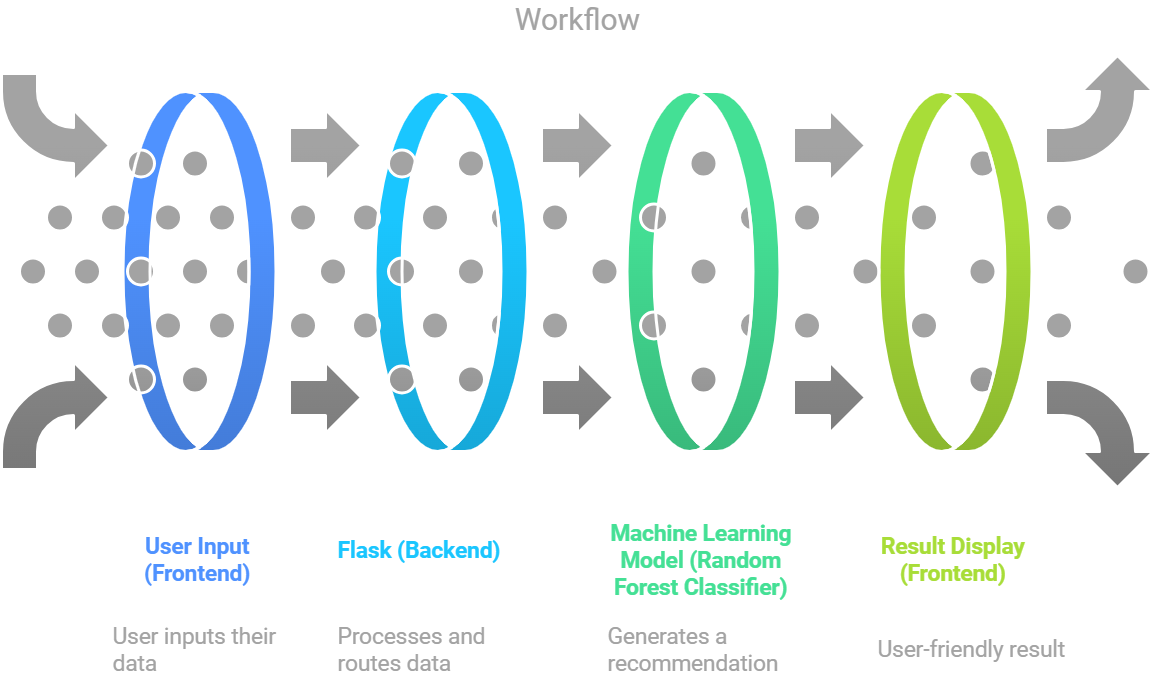
The **backend** was built using Flask, which serves as the communication bridge between the user interface and the AI model. Flask processes the inputs from the frontend, passes them to the AI model for prediction, and returns the output, ensuring smooth functionality across the system.

At the core of the system is a **machine learning model** implemented using a **RandomForestClassifier** from Scikit-learn. The model processes data after feature scaling to standardize numerical values and label encoding to transform categorical features like gender and learning style into numerical formats. These steps ensure the model can handle diverse input data effectively.

The tool was trained on the **Australian Student Performance Data (ASPD24)** dataset sourced from Kaggle. This dataset contains features such as high school GPA, entrance exam scores, gender, and learning style, which provide valuable insights into a student’s academic profile. These inputs allow the system to generate tailored major recommendations based on a comprehensive understanding of each user’s strengths and preferences.

1. Implementation Details and Result

## **3.1 Diagram to show implementation overview**



## **Implementation Discussion (Processes and Operations)**

The development of the **AI-Driven Major Decision Support Tool** was carried out in a structured manner, combining frontend and backend components with a machine learning model to create a seamless user experience. The following is a breakdown of the processes and operations involved:

**Frontend Development:**

The user interface (UI) was designed with **HTML**, **CSS**, and **JavaScript** to ensure it is both user-friendly and responsive. The frontend collects essential information from the user, such as **High School GPA**, **Entrance Exam Scores**, **Gender**, **Learning Style**, and **Study Performance**. These inputs are a mix of numerical and categorical data.

**Backend Development (Flask API):**

The **Flask framework** was chosen to build the backend API, acting as the intermediary between the frontend and the machine learning model. Once the backend receives the user inputs, it processes them to prepare for the prediction. This involves **feature scaling** for numerical data (like GPA and exam scores) and **label encoding** for categorical data (such as gender and learning style), converting them into formats that the machine learning model can understand.

After the data is processed, the backend sends it to the **Random Forest Classifier** model. Upon receiving the model's output, the backend formats the prediction into a user-friendly result and sends it back to the frontend for display.

**Machine Learning Model (Random Forest Classifier):**

The system employs the **Random Forest Classifier**, a machine learning algorithm known for its effectiveness in handling both numerical and categorical data. This algorithm operates through a structured, step-by-step process to build the model and make predictions:

1. **Construction of Decision Trees**: During the training phase, multiple decision trees are constructed, each trained on a different subset of the data.
2. **Bootstrap Aggregation (Bagging)**: Random sampling is used to create these subsets, ensuring diversity among the trees.
3. **Feature Randomization**: At each split in a tree, a random subset of features is considered, improving the model’s robustness and reducing overfitting.
4. **Prediction Aggregation**: For classification tasks, the outputs of all trees are combined using majority voting to produce the final prediction.

The model was trained on the **Australian Student Performance Data (ASPD24)** dataset, which includes features such as **High School GPA**, **Entrance Exam Scores**, and **Learning Style**. These inputs provide a comprehensive representation of a student's academic and personal profile.

**Data Cleaning Process:** To ensure a clean and reliable dataset, the following steps were taken:

**1. Handling Missing Values:**

* For numerical features like **High School GPA** and **Entrance Exam Scores**, missing values were replaced with the column mean to retain the dataset's distribution.
* For categorical features such as **Gender** and **Learning Style**, missing values were filled with the most frequent value (mode) to maintain consistency.

**2.Removing Duplicates:**  
Duplicate rows were identified and removed to ensure each record in the dataset was unique. This step prevented data bias and optimized the dataset for training.

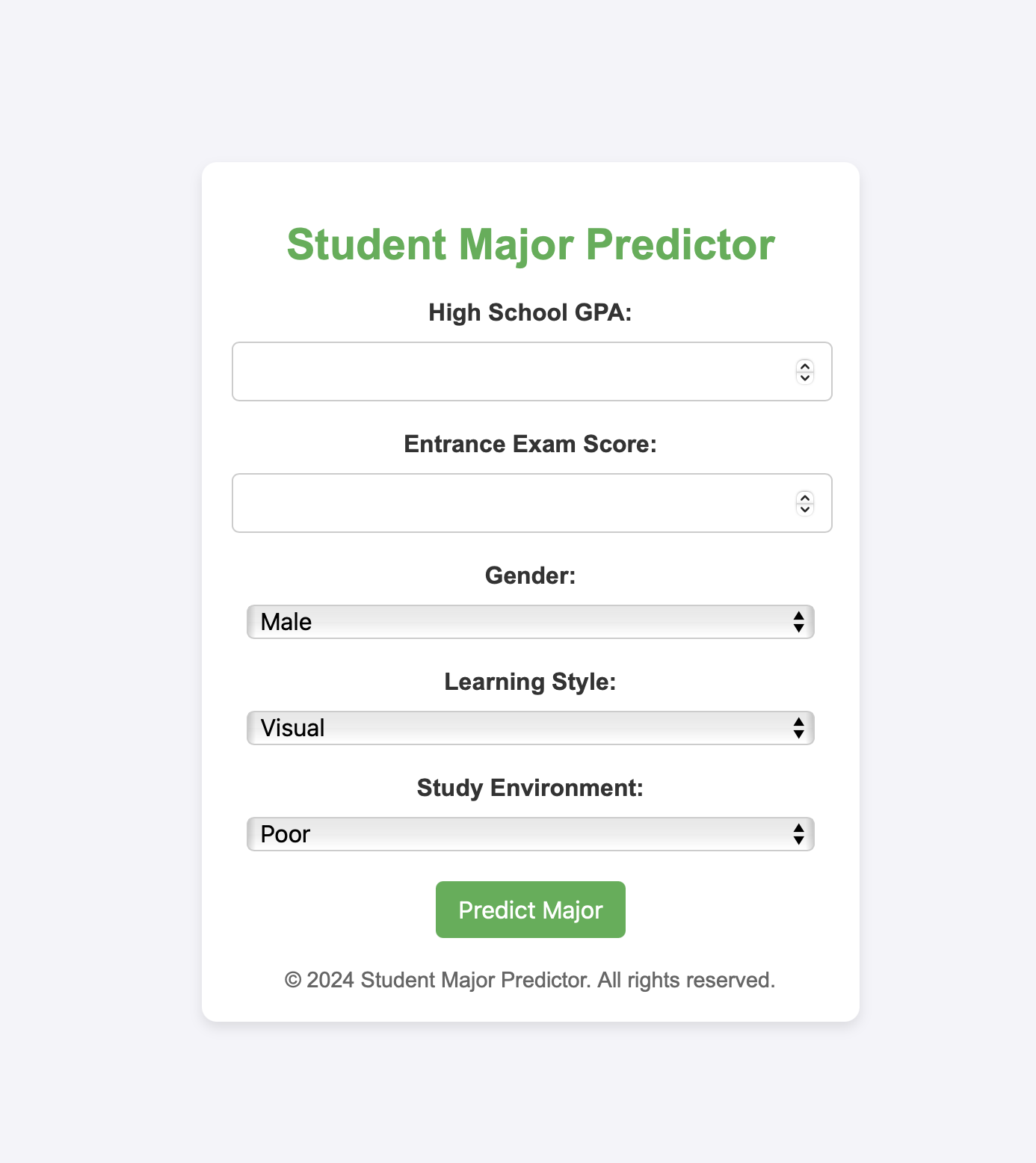
**Training process:** the Random Forest algorithm was optimized to generalize well to new data while minimizing the risk of overfitting. After training, the model was evaluated using a **20% hold-out test set** that was not part of the training data. Performance metrics such as **accuracy** and **precision** were calculated to assess the model's ability to make reliable predictions on unseen data, ensuring the system delivers accurate and personalized recommendations.

**Results Display:**

Once the model generates a recommendation, the backend sends the prediction back to the frontend, where it is displayed in an easily interpretable format. The recommendation provides the most suitable academic major based on the user's input, helping guide their decision-making process.

This streamlined process ensures that the system can efficiently process user input, make accurate predictions, and present tailored suggestions to students, facilitating their major selection process.

1. User Interface

The user interface (UI) is designed to provide a simple and intuitive experience for students, allowing them to input their relevant details to receive personalized major recommendations. Built with **HTML, CSS, and JavaScript**, the interface ensures accessibility, ease of use, and visual clarity.

The UI consists of several input fields where students can provide the following information:

* **High School GPA**: A numerical input field where students enter their GPA, allowing the system to factor in their academic performance.
* **Entrance Exam Scores**: Another numerical field for students to input their scores from any entrance exams they have taken, such as SAT, ACT, or other standardized tests.
* **Gender**: A dropdown or radio button selection that allows students to choose their gender, which helps the model understand demographic patterns that may influence academic preferences.
* **Learning Style**: A selection input where students can choose their preferred learning style (e.g., visual, auditory, kinesthetic). This helps the system align the recommended majors with how the student learns best.
* **Study Performance**: A field where students can provide feedback on their study habits or performance (“Poor,” “Average,” “Good,” “Excellent”). This additional input can further refine the recommendations, helping the system suggest majors that align with the student's learning capacity and motivation.

Once all the inputs are provided, the system processes the information and generates a recommended major based on the student’s profile. The interface is designed to be user-friendly, with clear instructions and validation to ensure all inputs are correct before submission. The results are displayed on the same page, providing the student with a tailored suggestion based on their academic background and preferences.

By focusing on simplicity and clarity, the user interface aims to make the process of major selection easier and more personalized for each student.

1. Conclusion

The **AI-Driven Major Decision Support Tool** leverages artificial intelligence to assist students in selecting academic majors aligned with their strengths, preferences, and academic profiles. By combining a user-friendly interface, a robust backend, and a machine learning model trained on relevant data, the system provides reliable and personalized recommendations. While it is designed to simplify the decision-making process, the tool complements rather than replaces traditional academic advising, empowering students to make informed choices about their future.

1. References

|  |  |
| --- | --- |
| 1 | Ayub, N. (2024). *Australian student performance data (ASPD24)* [Data set]. Kaggle. <https://www.kaggle.com/datasets/nasirayub2/australian-student-performancedata-aspd24> |
| 2 | OpenAI. (2024). *ChatGPT* (December 2024 version) [Large language model]. Retrieved from <https://openai.com/chatgpt> |