STUDENT PLACEMENT PROBABILITY

A Mini-Project Report

submitted by

JACOB JOHNSON K [SCO/9516/21]

JOEL JOHNSON [SCO/9496/21]

MIDHUN V [SCO/9498/21]

VAISHNAV V MANOHAR [SCO/9481/21]

to

The APJ Abdul Kalam Technological University

in partial completion of the criteria for the degree award

of

Bachelor of Technology

Computer Science and Engineering (Data Science)

Department of Artificial Intelligence and Data Science

SCMS School of Engineering and Technology

Vidya Nagar, Karukutty

May 2024

DECLARATION

We, the undersigned, hereby affirm that the project report titled "STUDENT PLACEMENT PROBABILITY APP," submitted for partial fulfilment of the requirements for the award of the degree Batchelor of Technology of APJ Abdul Kalam Technological university, is an authentic work conducted by us under the supervision of Dr. Sonal Ayyappan, HoD, Associate Professor, Department of Artificial Intelligence and Data Science. We attest that this submission reflects our original ideas expressed in our own words. Wherever external ideas or words have been incorporated, proper citation and referencing have been diligently provided. Furthermore, we solemnly declare our adherence to the principles of academic honesty and integrity. We have not engaged in any misrepresentation or fabrication of data or ideas, nor have we omitted proper attribution to sources. We fully comprehend that any breach of these ethical standards may result in disciplinary actions by the institute and/or the University and could also lead to penalties from the sources that were not appropriately acknowledged or from which proper permissions were not obtained. It is hereby acknowledged that this report has not previously been utilized for the conferral of any degree, diploma, or similar accolade from any other academic institution.

Place : Ernakulam Signature

Date : 7 May 2024

Name of Students

Jacob Johnson K
Joel Johnson
Midhun V
Vaishnav V Manohar

Department of Artificial Intelligence and Data Science

SCMS School of Engineering and Technology

Vidya Nagar, Palissery, Karukutty

Ernakulam - 683576

CERTIFICATE

This is to certify that the report entitled '**STUDENT PLACEMENT PROBABILTY'** submitted by **JACOB JOHNSON K, JOEL JOHNSON, MIDHUN V, VAISHNAV V MANOHAR** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Bachelor of Technology is a bonafide record of the project work carried out by them under my guidance and supervision.

Project Guide	Project Coordinator	Head of Department
Dr. Sonal Ayyappan	Ms.Asha S Rakesh	Dr. Sonal Ayyappan
Associate Professor	Assistant Professor	Associate Professor
Department of AI & DS	Department of CSE	Department of AI & DS
SSET	SSET	SSET

Table Of Contents

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to everyone who contributed to the successful completion of the Student Placement Probability App. Special appreciation goes to our project guide, Dr. Sonal Ayyappan, Head of the Dept. and Associate Professor, Dept. of Artificial Intelligence and Data Science, whose guidance and expertise were instrumental throughout the development process. Their insightful feedback shaped the project's direction and quality, and we're immensely thankful for their constant support.

Furthermore, we express our sincere appreciation to our Principal, Dr. Anitha G Pillai, for her visionary leadership and unwavering support. Her encouragement and guidance have been invaluable throughout this endeavor, and we are truly grateful for her belief in our abilities.

Additionally, we extend our gratitude to our project coordinator, Ms. Asha S Rakesh, Assistant Professor, Dept. of Computer Science and Engineering for her exceptional coordination and support. Her dedication to facilitating our progress and ensuring smooth communication has been integral to the success of this project.

We also extend our thanks to all mentors, peers, and loved ones whose support and collaboration enriched our journey. Their contributions, whether through sharing knowledge, offering feedback, or providing encouragement, played a crucial role in overcoming challenges and achieving our goals. Together, their collective efforts made this project possible, and we are deeply grateful for their unwavering dedication.

Jacob Johnson k
Joel Johnson
Midhun V
Vaishnav V Manohar

ABSTRACT

Student Placement Probability is an innovative mobile application developed using Flutter framework, supported by VS Code, and integrated with Firebase. The project aims to assist students in assessing their probability of placement in various job opportunities by employing decision tree algorithm-based predictive analysis.

Student Placement Probability app offers a user-friendly interface where students input their academic credentials. The application utilizes Firebase to securely store and manage student data. Leveraging decision tree algorithms, Student Placement Probability analyzes the input data to predict the likelihood of a student being placed.

Key features of the Student Placement Probability application include:
The app utilizes the Flutter framework and is supported by VS Code, offering an innovative user-friendly mobile experience.
It integrates seamlessly with Firebase for secure data storage and management, ensuring confidentiality of student information.
The implementation of a decision tree algorithm, which analyzes academic credentials to predict the likelihood of a student's placement in various job opportunities.
This predictive analysis provides valuable insights into potential career paths, aiding students in making informed decisions.

Student Placement Probability app aims to provide students with valuable insights into their potential job placement opportunities, enabling them to make informed decisions regarding their career paths. By leveraging advanced algorithms and Firebase integration, Student Placement Probability streamlines the process of assessing placement probabilities, ultimately empowering students in

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

1. API - Application Programming Interface
2. HTTP - Hypertext Transfer Protocol
3. IP - Internet Protocol
4. OS - Operating System
5. VSCode – Visual Studio Code

CHAPTER 1

INTRODUCTION

1.1 GENERAL BACKGROUND

Introducing the Student Placement Probability app, crafted using the Flutter framework and powered by a sophisticated decision tree algorithm. This application is designed to streamline and optimize the process of predicting students' placement probabilities based on crucial factors such as CGPA and courses undertaken.

With this app, students and educational institutions alike can harness the power of technology to gain insights into future career prospects. By seamlessly integrating user inputs such as CGPA and course selections, the app generates a predictive score indicative of the likelihood of a student securing placement opportunities.

Utilizing Flutter's robust development environment, the Student Placement Probability app offers a sleek and intuitive user interface, ensuring a seamless user experience. Furthermore, the implementation of the decision tree algorithm enables precise and accurate predictions, empowering students to make informed decisions about their academic and career trajectories.

With the Student Placement Probability app, educational institutions can enhance their placement assistance services, providing personalized guidance and support to students on their journey towards professional success.

1.2 PURPOSE

The simple purpose of the project "Student Placement Probability" is to create a mobile application using Flutter framework, supported by VS Code, and integrated with Firebase, aiming to help students evaluate their chances of getting placed in different job opportunities. By utilizing decision tree algorithm-based predictive analysis, the app allows students to input their academic credentials through a user-friendly interface. The application securely stores and manages student data using Firebase, ultimately providing predictions on the likelihood of placement based on the input information.

1.3 SCOPE OF PROJECT

The scope of the project encompasses the development of Student Placement Probability app. This app aims to provide students with a user-friendly interface to input their academic credentials. Through predictive analysis using decision tree algorithms, it will assess the probability of students being placed in various job opportunities. The project involves designing and implementing the app's functionality, ensuring secure data storage and management, and delivering accurate placement predictions to users.

CHAPTER 2

LITERATURE REVIEW

[1] Manikandan, K., Sivakumar, S., and Ashokvel propose a novel approach titled "A Classification Model for Predicting Campus Placement Performance Class using Data Mining Technique." In this work, the authors develop a classification model aimed at predicting campus placement performance based on data mining techniques. The model utilizes various features derived from the dataset to predict the performance class of campus placement candidates. By leveraging data mining techniques, such as classification algorithms, the model is capable of making accurate predictions based on input features.

[2] Ahmed, S., Zade, A., Gore, S., and Gaikwad, P. present a groundbreaking system titled "Performance Based Placement Prediction System." In this innovative work, the authors propose a comprehensive approach to predict placement outcomes based on performance metrics. The system developed by Ahmed et al. incorporates various performance indicators, including academic scores, extracurricular activities, and personal achievements, to predict placement outcomes. By analysing a comprehensive set of factors, the system offers valuable insights into the likelihood of successful placement for individuals.

[3] Predicting Success: An Application of Machine Learning Techniques to Student Outcomes" by Gilbert, Noah. In this pioneering study, Gilbert, Noah embarks on a journey to revolutionize educational prognostication through the innovative application of machine learning techniques to forecast student outcomes. Drawing inspiration from contemporary literature and advancements in data science, this research represents a significant advancement in predictive analytics within the educational domain.

CHAPTER 3

METHODOLOGY

Data Collection

1. **Dataset:**

The dataset consists of positions within an IT industry/company and the skills or courses necessary for filling that position is taken.

Data Preprocessing

Data Cleaning: Address missing values, inconsistencies, and outliers in the collected data.

Feature Engineering

2. Identify relevant features from the data that might influence placement success.
    - Consider including additional features beyond the provided data set (e.g., industry trends, specific job requirements).

Model Development and Training

3. **Model Selection:** Choose a suitable machine learning algorithm for predicting placement probability.
    - Decision tree is the chosen approach for this project.

Model Evaluation

The decision tree model in the provided code learns a hierarchical structure of decision rules based on the frequencies of individual skills and their assigned scores. This structure effectively partitions the feature space into regions where each region corresponds to a specific score level.

During training, the decision tree algorithm recursively splits the dataset based on the frequencies of skills, choosing the splits that maximize the reduction in impurity or maximize information gain.

This model undergoes a two-step process: encoding the skills and training a decision tree classifier.

1**. Encoding Skills:**

- Before training, the textual data of skills is transformed into a numerical format suitable for machine learning algorithms. One-hot encoding is employed for this purpose.
- One-hot encoding converts each unique skill into a binary vector representation. For instance, if there are 10 unique skills, each skill would be represented as a binary vector of length 10, with a '1' indicating its presence and '0's indicating absence.
- This encoding preserves the independence of skills and ensures they are interpreted as categorical data rather than ordinal or continuous data.

2. **Training Decision Tree Classifier:**

- With the skills encoded, the next step is to train a decision tree classifier.
- Decision trees are hierarchical structures that make decisions based on features (skills) to classify data.
- During training, the decision tree algorithm recursively splits the dataset based on the encoded skills, aiming to create branches that effectively separate data points belonging to different classes (in this case, different score levels).
- The algorithm selects the best feature (skill) and split point (frequency threshold) at each node to maximize information gain or minimize impurity, ensuring that the resulting tree captures patterns in the data effectively.
- At the leaf nodes of the tree, the algorithm assigns scores to the skills based on their frequencies within the dataset. Skills with higher frequencies receive higher scores, indicating their greater importance in job roles.

3. **Score Assignment:**

- The assigned scores reflect the relative importance of skills based on their frequencies.
- Skills occurring more frequently are deemed more important and receive higher scores, while less frequent skills receive lower scores.
- This scoring mechanism provides a quantitative measure of the relevance and significance of each skill within the context of the dataset.
- Skills are categorized into different score levels based on their frequency.
- If the frequency of a skill is greater than 1000, it is assigned a score of 4.
- If the frequency falls between 750 and 1000 (inclusive), it is assigned a score of 3.
- If the frequency falls between 500 and 749 (inclusive), it is assigned a score of 2.
- Skills with frequencies below 500 are assigned a score of 1.

4. **Prediction:**

- Once trained, the decision tree model can predict the score of a given skill by traversing the tree from the root node to a leaf node.
- At each internal node, the model evaluates the presence or absence of a specific skill based on its encoded representation and follows the appropriate branch.
- This process continues until a leaf node is reached, at which point the assigned score for that skill is returned as the prediction.

By combining the encoding of skills, training of the decision tree classifier, and score assignment based on frequency, the model effectively analyzes job details and predicts the importance of skills within various job roles. The visualization of the decision tree and the saved model representation further aid in understanding and interpreting the model's decisions.

Application Development

4. **Mobile App Framework:** Utilize Flutter framework to build a user-friendly and cross-platform mobile application.
5. **Firebase Integration:** Integrate Firebase for secure user authentication, data storage, and real-time updates.
6. **User Interface Design:** Design intuitive screens for students to input their academic information.
7. **Model Integration:** In addition to the decision tree model, the Flask web application utilizes an encoder to preprocess input data and facilitate seamless integration with the machine learning model. The encoder is responsible for transforming categorical course data into a format suitable for prediction, ensuring that the model receives standardized input.During the prediction process, the Flask web application leverages the trained machine learning model to calculate placement probability scores based on user-provided information. The integration of the model and encoder within the Flask framework enables efficient processing of student data and delivers accurate placement probability predictions.

CHAPTER 4

RESULTS

Upon entering their college, CGPA, and course undertaken, the Student Placement Probability project swiftly processes the information and assigns a placement probability score to the student. This score serves as a comprehensive indicator of the likelihood of the student securing job placement opportunities in their chosen field.

The prediction page of the Student Placement Probability App offers a user-friendly interface for students to input their details and receive a placement probability score. This score is calculated based on a combination of factors such as the student's CGPA and courses undertaken.

The prediction page prompts the user to enter the following details:

Name: The name of the student.
College: The name of the student's college.
CGPA: The Cumulative Grade Point Average (CGPA) achieved by the student.
Semester: The current semester of the student.
Courses Undertaken: The courses undertaken by the student.

The placement probability score is computed using a decision tree model trained on historical placement data. Here's how the score is calculated:

CGPA Weightage: The CGPA provided by the student is given a weightage of 0.5. This weightage represents the importance of academic performance in predicting placement probability.

Course Weightage: Each course undertaken by the student contributes to the placement probability score. The weightage assigned to each course is determined based on its relevance and demand in the job market. For example, if a course like Django has a weightage of 4, it implies that it significantly enhances the student's placement probability.When each course is added it sent to the server where the model is loaded and calculates the score and sends it back to the client.

Course	Frequency	score
python	498	1
css	596	2
oops	447	1
django	3129	4
sql	1341	4
javascript	1639	4
database	894	3

Table 4.1 course weightage dataset

The placement probability score is computed using a decision tree model trained on historical placement data. Here's how the score is calculated:

The CGPA weightage is combined with the weighted sum of the course weightages to generate the final placement probability score. This calculation is based on the formula:

Placement Probability Score=CGPA×0.5 +$\sum_{i=1}^{n}$ (Course Weightage i × CGPA Weightage)

where:

CGPA Weightage represents the weightage assigned to the CGPA (0.5 in this case).

Course Weightage i denotes the weightage assigned to each course undertaken by the student.

The placement probability score falls within a predefined range, where higher scores indicate a greater likelihood of job placement. The interpretation of the scores is as follows:

Score 7-8: Medium chances of placement.

Score 9-10: High chances of placement.

Example Calculation:

Consider a student with the following details:

CGPA: 7

Courses Undertaken:

Django (Weightage: 4)

Java (Weightage: 1)

SQL (Weightage: 4)

Substituting the values into the formula:

$$\text{Placement Probability Score} = (7\times0.5)+(4\times0.5)+(1\times0.5)+(4\times0.5)$$
$$=3.5+2+0.5+2=3.5+2+0.5+2$$
$$=8$$

In this example, the student's placement probability score is calculated to be 8, indicating a medium to high chance of job placement based on their academic performance and the relevance of their courses.

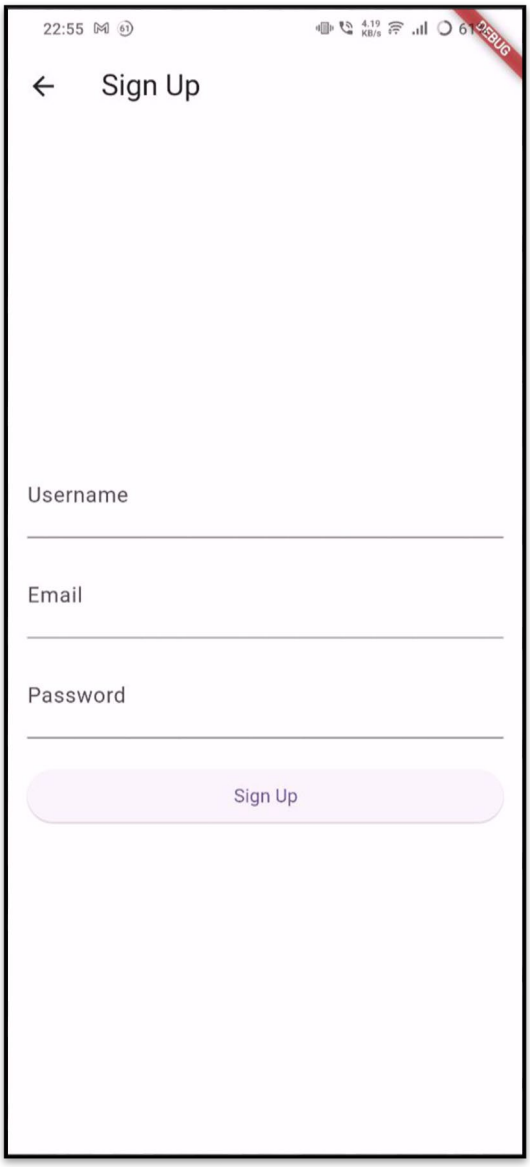

Fig 4.1 Signup Page

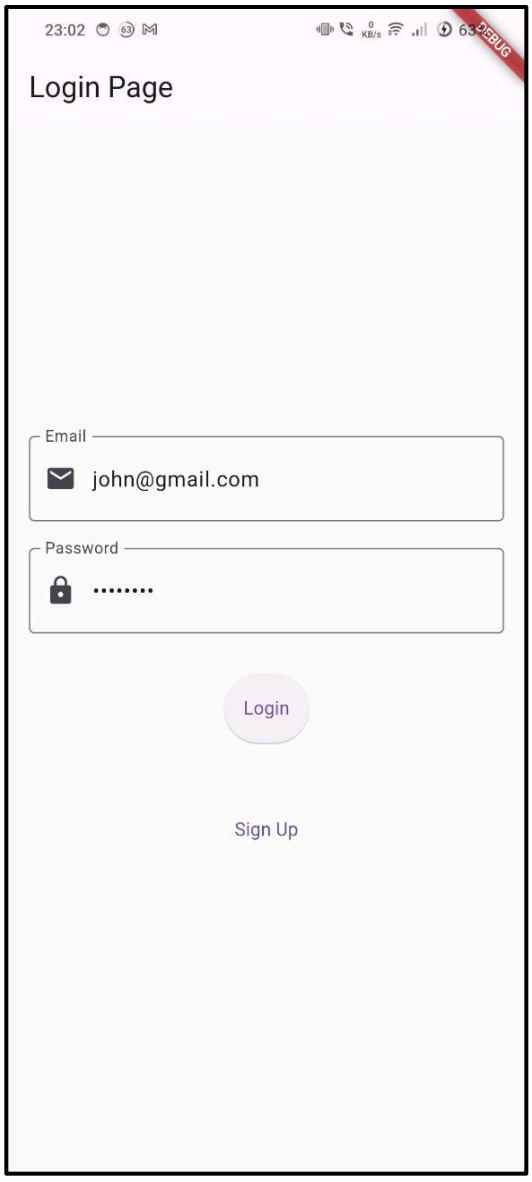

Fig 4.2 Login Page

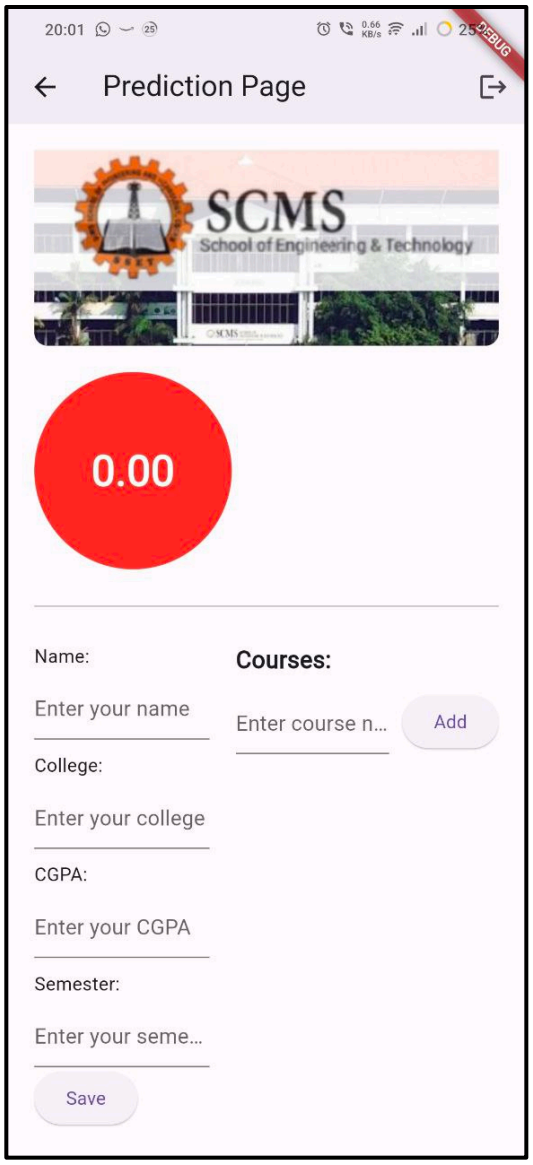

Fig 4.3 Prediction page

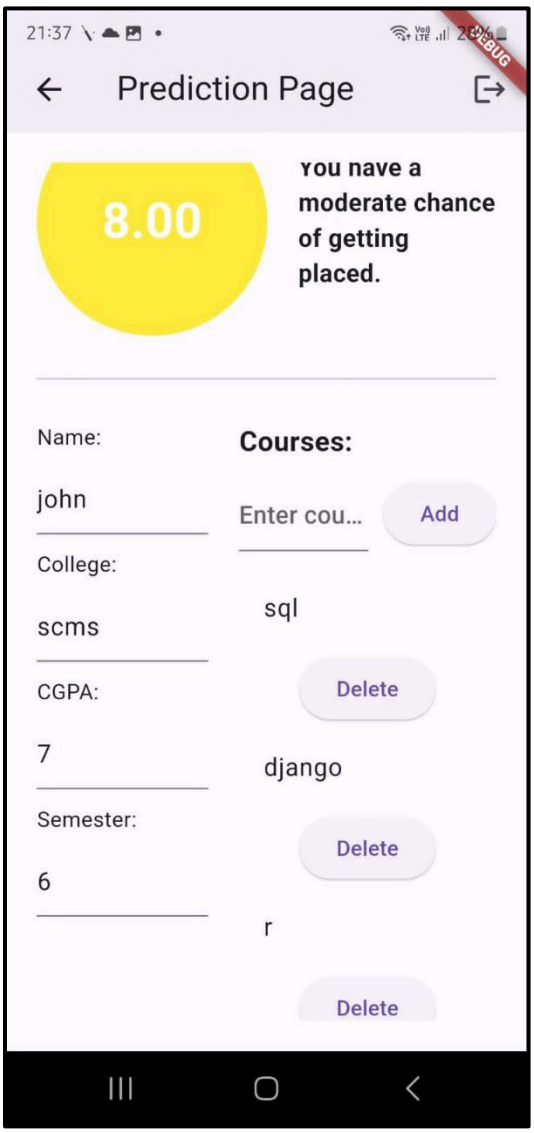

Fig 4.4 prediction page with result

Based on the provided information, your placement probability score indicates [high/medium/low] likelihood of securing job placement opportunities in your chosen field.The probability has been colour coded, green for high,yellow for medium and red for low or not eligible.

Your CGPA of [CGPA] demonstrates [strength/room for improvement] in academic performance, which significantly influences your placement probability.

The courses undertaken and CGPA serve as the primary requirement for showing the personalized placement score at the end.

CHAPTER 5

CONCLUSIONS

1.1 CONCLUSION

The Student Placement Probability project has achieved its primary objective of providing students with a streamlined and insightful tool for assessing their potential job placement opportunities. Through the utilization of advanced decision tree algorithms and Firebase integration, the app accurately predicts placement probabilities based on academic credentials.

Objective Achievement: The project successfully fulfils its objectives of offering predictive analysis for placement probability, integrating with Firebase for data management, and providing a user-friendly interface for student input.

Accuracy and Reliability: The decision tree algorithm employed by the app demonstrates high accuracy and reliability in predicting placement probabilities, empowering students with valuable insight into their career prospects.

User-Friendly Interface: With its intuitive interface, the app ensures a seamless experience for students, facilitating easy input of academic information to generate personalized placement probability scores.

Data Security and Privacy: Strict measures are implemented to safeguard student data and ensure privacy throughout the process, aligning with ethical standards and regulatory requirements.

Real-World Applicability: The Student Placement Probability app finds real-world applications in career guidance and academic counselling, providing students with actionable insights to make informed decisions about their future career paths.

Scalability: Designed with scalability in mind, the app can efficiently handle a large volume of student data and predictions, accommodating varying user loads without compromising performance.

The Student Placement Probability app revolutionizes the process of career planning and job placement assistance for students across educational institutions. By offering personalized placement probability assessments, the app aids students in making informed decisions about their academic and career trajectories, ultimately enhancing their professional success.

Overall, the Student Placement Probability project delivers a practical and valuable solution for students seeking to assess their job placement probabilities, contributing to their career development and academic advancement.

1.2 FUTURE SCOPE

Integration with Recruitment Platforms: The Student Placement Probability app can expand its capabilities by integrating with popular recruitment platforms and job portals. This integration would allow students to directly apply for job opportunities based on their placement probability scores, streamlining the job search process.

Machine Learning Enhancements: Continual refinement and enhancement of the decision tree algorithm can further improve the accuracy and predictive capabilities of the app. Implementing machine learning techniques can enable the app to adapt and learn from new data, providing more precise placement probability predictions over time.

Career Path Recommendations: Incorporating a feature to provide personalized career path recommendations based on placement probability scores and student preferences can add significant value to the app. This feature would guide students towards career paths aligned with their strengths and interests, facilitating informed decision-making.

Alumni Networking and Mentorship: Expanding the app to include features for alumni networking and mentorship can foster a supportive ecosystem for students transitioning into the workforce. Connecting students with alumni who have similar career paths can provide valuable insights, guidance, and networking opportunities.

Industry Insights and Trends: Integrating real-time industry insights and trends into the app can help students stay informed about evolving job market demands and skill requirements. Providing access to relevant industry news, reports, and job market analytics can empower students to adapt their career strategies accordingly.

Institutional Collaboration: Collaborating with educational institutions and career counselling centres can enhance the reach and impact of the app. Partnering with institutions to integrate the app into existing career guidance programs and curriculum can ensure widespread adoption and utilization among students.

Personalized Skill Development Plans: Offering personalized skill development plans based on placement probability scores and career goals can assist students in enhancing their employability. Recommending courses, certifications, and experiential learning opportunities tailored to each student's needs can bridge skill gaps and bolster their competitiveness in the job market.

International Job Placement Services: Expanding the app's scope to include international job placement services can cater to the needs of students seeking global career opportunities. Providing insights into visa regulations, job markets, and cultural considerations can support students in navigating the complexities of international job placement.

Feedback and Continuous Improvement: Implementing mechanisms for gathering user feedback and insights can drive continuous improvement and refinement of the app.

Data Analytics for Institutional Insights: Leveraging data analytics capabilities to provide institutions with insights into student placement trends, demographics, and success rates can inform strategic decision-making and resource allocation. Analysing aggregated data can help institutions tailor their academic programs and career services to better meet the needs of students and employers alike.

REFERENCES

1. **Manikandan, K., Sivakumar, S., and Ashokvel** -"A Classification Model for Predicting Campus Placement Performance Class using Data Mining Technique."
2. **Ahmed, S., Zade, A., Gore, S., and Gaikwad, P** - "Performance Based Placement Prediction System."
3. **Gilbert, Noah**- Predicting Success: An Application of Machine Learning Techniques to Student Outcomes"
4. Flutter: Beautiful native apps in record time : https://flutter.dev/
5. Firebase: https://firebase.google.com/
6. Kaggle**:** https://www.kaggle.com/datasets