iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** V **Month of publication:** May 2023

DOI: https://doi.org/10.22214/ijraset.2023.53262

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

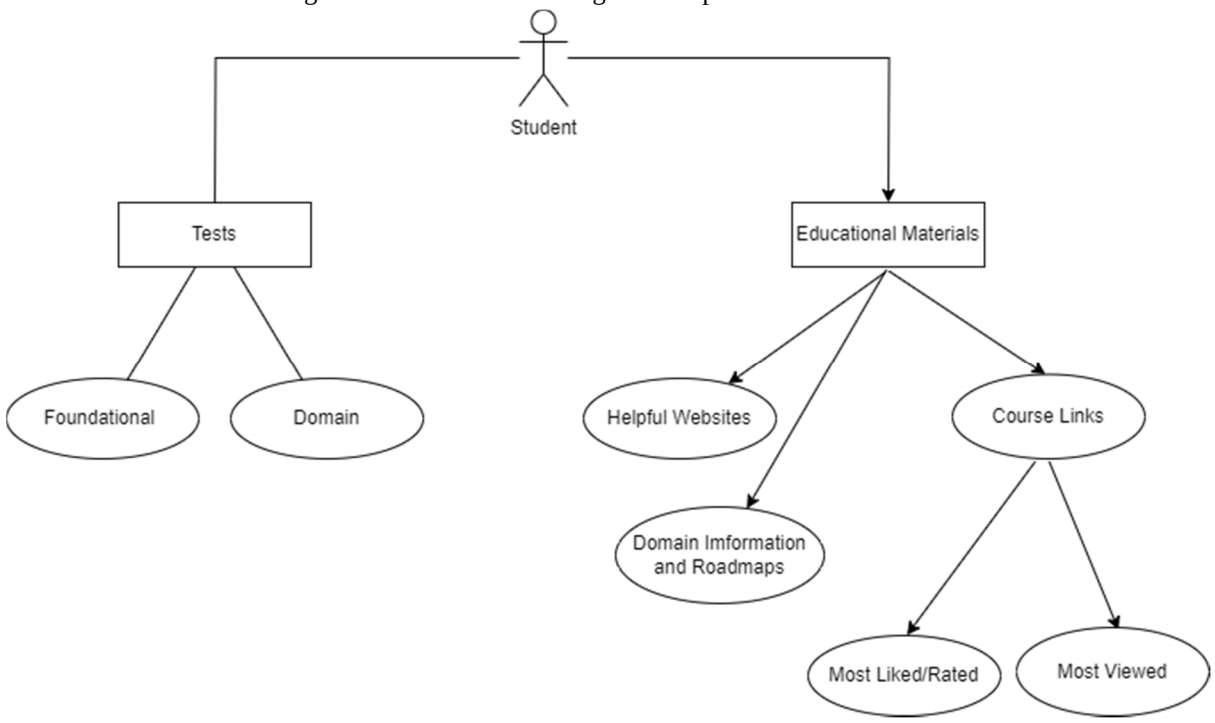
Implementation of a Computer Science Career Guidance Website that Makes Use of Machine Learning

Parth Lokhande[1], Yash Nayakwadi[2], Manali Umardand[3], Vedant Araj[4], Prof. Dheeraj Patil[5]
[1, 2, 3, 4, 5]*Department of Information and Technology (Sem II 2022-23) Nutan Maharashtra Institute of Engineering and Technology (An Institute affiliated to Savitribai Phule Pune University)*

Abstract: ***The difficulty that students face in the beginning of their computer science career such as finding a suitable sub-domain to learn and pursue, finding online resources to study stay prominent across the globe.***
Earlier research and development have provided aptitude test for suggesting students a job role in the industry, but there is a need for a platform that addresses the problem of computer science students to pursue a suitable domain like data science or cyber security etc. for them in the initial phases of their careers.
There is a need for a website to help students towards finding learning materials and their inclination towards various sub-domains in the industry.

I. INTRODUCTION

The implementation of a website that guides students with finding a career path with tests and educational materials



The website will offer a foundational aptitude test where students will login and start the test, this test tests the knowledge of the students and calculate scores for following basic subjects that are required for various fields in computer science: mathematics, operating systems, computer architecture, programming concepts, software engineering, computer networks, algorithms. Based on the scores of these subjects, the candidate will then be suggested with a suitable field.
The user can navigate to domain information pages on the website where he can learn more about the domain and look into roadmaps to pursue it, find various educational materials and take a test for that field to test their knowledge about that field.

©IJRASET: All Rights are Reserved

II. LITERATURE REVIEW

A. *Research Papers*

1) The research paper titled “Domain-specific NLP system to support learning path and curriculum design at tech universities” proposes an intelligence system using NLP to bridge the gap between rapidly evolving tech industry demands and university courses in CS/IT. The system includes a Named Entity Recognition (NER) model, CSITNER, that accurately extracts tech-related skills. It also features a hybrid model, hybrid CSIT-CRS, which recommends personalized courses, career paths, and online courses. The system received positive feedback from a survey with students and faculty members from tech universities. It benefits stakeholders in tech-related industries by aiding decision-making in course selection and career paths.

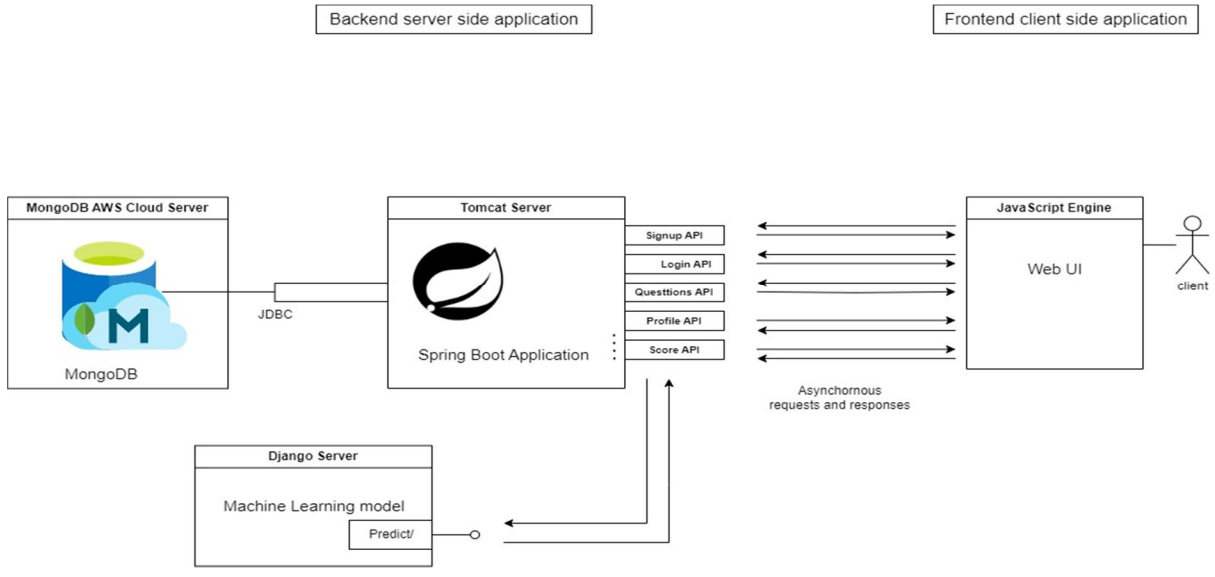
2) The research paper titled “Recommendation of Branch of Engineering using machine learning” highlights the challenges faced by students and parents in choosing the right college due to the abundance of information available online. It emphasizes the need for intelligent recommendation systems to help with the college admission process and reduce the workload on counselors. The proposed system aims to assist students in selecting the appropriate branch of engineering based on their marks. It utilizes techniques like K-nearest neighbors and collaborative filtering to recommend branches and colleges respectively. The system aims to automate and simplify the recommendation process, providing students with a list of eligible branches and colleges based on their scores.

3) In the research paper titled “Development of a hybrid students’ career path recommender system using machine learning techniques” the information calls attention to the challenges faced by students in choosing a career path in the rapidly advancing and diverse field of computer science. Traditional methods may be time-consuming, while machine learning techniques offer potential solutions. A career recommender system using machine learning is proposed to assist computer science undergraduates in selecting suitable career paths based on their academic records, talents, and interests. The system aims to help students make knowledgeable decisions early in their university studies, leading to improved academic performance and future productivity. The system would recommend one of six domains: Project Manager, Database Administrator, Software Developer, Business Intelligence Analyst, Security Administrator, or Technical Support. The proposed system can benefit students, job seekers, and employers by providing guidance and relevant information in the IT industry.

4) The research paper titled “A Web-Based Platform for Predicting Student Careers using Machine Learning with Learning Resources for Computer Science Domains” presents a web-based application that uses advanced machine learning techniques to help students and enthusiasts learn computer science fundamentals and choose suitable career paths. The application offers skill assessment tests, personalized career suggestions, curated learning resources, and career roadmaps. By analyzing test results, the system provides personalized recommendations based on users' strengths and interests. The application serves as a valuable tool for individuals navigating the fast-paced digital world, enabling them to make informed career decisions and pursue their preferred domains in software and computer science.

III. METHODOLOGY

A. *Application Architecture*



ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

The framework which is used to develop the client-side application is ReactJS, this stands as the front-end of the web-application. It has a home page, domain page, test pages etc.

The business logic layer is built using the Spring-Boot framework which will act as a server-side application, all data processing and validation along with API logic is written in this layer, it uses JPA connection to connect to the cloud database.

Below is a list of APIs it offers with their description

1) Sign UP API

Endpoint: POST / API /SignUp

Summary: Post SignUp API

Description:

This endpoint is used to save user details during the sign-up process. It expects a JSON object in the request body containing the following properties:

- contact: An integer representing the user's contact information.
- email: A string representing the user's email address.
- first Name: A string representing the user's first name.
- last Name: A string representing the user's last name.
- password: A string representing the user's password.
- profession: A string representing the user's profession.
- subject Details: An object containing additional details about the user's subjects (referenced from the "Subject Details" definition).
- username: A string representing the user's chosen username.

Responses:

- 200: OK. Indicates a successful operation with a response of type string.

Overall, this endpoint allows users to provide their details for signing up, and upon successful execution, it returns a “Signup successful!!!” response and if username already exists it returns “Username already exists!” response.

2) Login API

Endpoint: POST /API/login

Summary: Post Login API

Description: This endpoint is used to authenticate users and perform a login operation. It expects a JSON object in the request body containing the following properties:

password: A string representing the user's password.

username: A string representing the user's username.

Responses:

200: OK. Indicates a successful login operation with a response of type string.

Overall, this endpoint allows users to provide their login credentials (username and password), and upon successful authentication, it returns an "Logged In successfully" response and if invalid returns an “Invalid Username or password” response.

3) Get All Questions API

Endpoint: GET /API/questions

Summary: Get all questions

Description: This endpoint is used to fetch all the questions available in the application. It does not require any request body parameters.

Responses:

200: OK. Indicates a successful operation with a response of type string.

Overall, this endpoint allows users to retrieve all the questions in the application, and upon successful execution, it returns a "Questions array as Response body (JSON array)" response with the list of questions as a string.

4) Get Questions by Domain Id API

Endpoint: GET / API /questions/{domain}

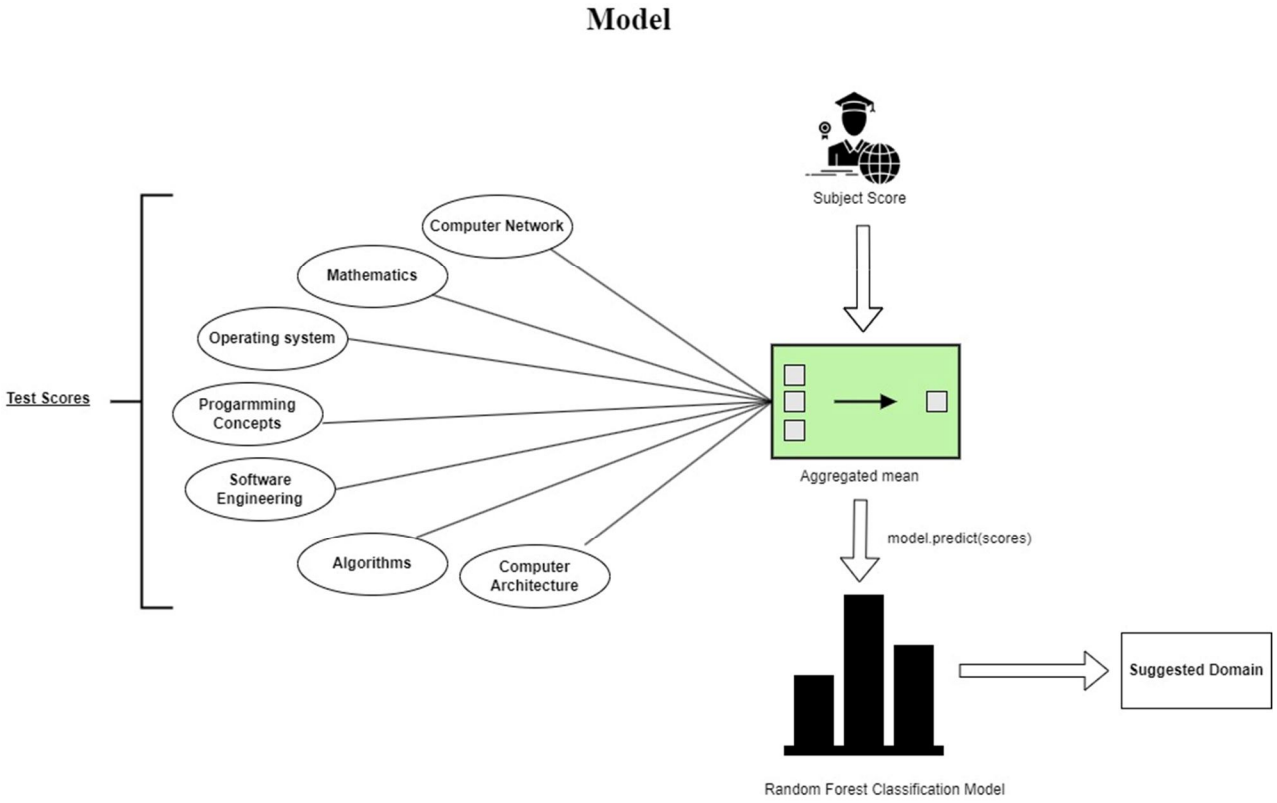
©IJRASET: All Rights are Reserved

Summary: Get questions by domain
Description: This endpoint is used to fetch questions associated with a specific domain in the application. It expects a path parameter called "domain" which represents the domain ID.
Parameters:
domain (path parameter): A string representing the domain ID for which the questions should be retrieved. This parameter is required.
Responses:
200: OK. Indicates a successful operation with a response of type string.
Overall, this endpoint allows users to retrieve questions based on a specific domain ID, and upon successful execution, it returns an ""Questions array as Response body (JSON array " response with the list of questions as a string.

5) Profile API
Endpoint: GET / API /profile/{username}
Summary: Get Profile by username
Description: This endpoint is used to fetch a user's profile information based on their username in the application. It expects a path parameter called "username" which represents the user's username.
Parameters:
username (path parameter): A string representing the username for which the profile should be retrieved. This parameter is required.
Responses:
200: OK. Indicates a successful operation with a response of type string.
Overall, this endpoint allows users to retrieve a user's profile information based on their username, and upon successful execution, it returns an "Profile object (JSON object)" response with the profile information as a string.

6) Score API
Endpoint: POST / API /scores
Summary: Post Score
Description:
This endpoint is used to save scores or performance data for different subjects. It expects a JSON object in the request body containing the following properties:
-email: A string representing the email/username for which the average score should be calculated.
-computerNetwork: A floating-point number representing the score in the "Computer network" subject.
- Computer Architecture: A floating-point number representing the score in the "computerArchitecture" subject.
- Algorithms: A floating-point number representing the score in the " Algorithms " subject.
- id: A string representing an identifier for the score data.
- mathematics: A floating-point number representing the score in the "Mathematics" subject.
- operatingSystem: A floating-point number representing the score in the "Operating system" subject.
-Programming: A floating-point number representing the score in the "Programming Concepts" subject.
-softwareEngineering: A floating-point number representing the score in the "Software Engineering" subject
The random forest classification model is hosted on a Django server on the same server machine. It offers the ‘/predict’ API to where post requests are received from the Spring-Boot application which contains the test scores, and it will return a response containing the suggested domain in JSON format.
The backend database is a MongoDB database which is a NoSQL database that will store all the application data in JSON format. This database is hosted on MongoDB’s cloud AWS server.

B. Prediction Model



Since we must classify the student according to its course in a particular subdomain this becomes a classification problem the algorithm that you use to solve this classification problem is the random forest algorithm. We have used data from previous research in the paper 'Student Career Prediction Using Advanced Machine Learning Techniques by K. Sripath Roy, K.Roopkanth, V.Uday Teja, V.Bhavana, J.Priyanka' to which we added our own data which contains data collected through LinkedIn API, student surveys and some random data. This data had a column which was the suggested job role depending on various factors such as subject marks, interests, completed courses etc. These job roles have been classified into main domains which is the output parameter for our classification model and the input parameters are scores of foundational subjects.

For more accuracy, if the student is already pursuing a degree or course in computer science and knows some of the subject marks that the student has scored he can give it as an input on the profile page so that when the test is taken the scores of test subjects get added with the corresponding degree/course marks and a mean of these 2 scores is taken as an input to the predict function.

The libraries used are- the Pandas library to handle data frames, the Sklearn library to use random forest model and train test split to split the data into training and testing data and pickle library to save the model to a file.

Reasons for using Random Forest model:

Paste decision tree diagram here

a) Robustness: Random Forests are less susceptible to overfitting and noisy data compared to individual decision trees.

b) Feature Importance: The algorithm provides a measure of feature importance, allowing insights into which features are most influential for classification.

c) Scalability: Random Forests can handle large datasets with numerous input features efficiently.

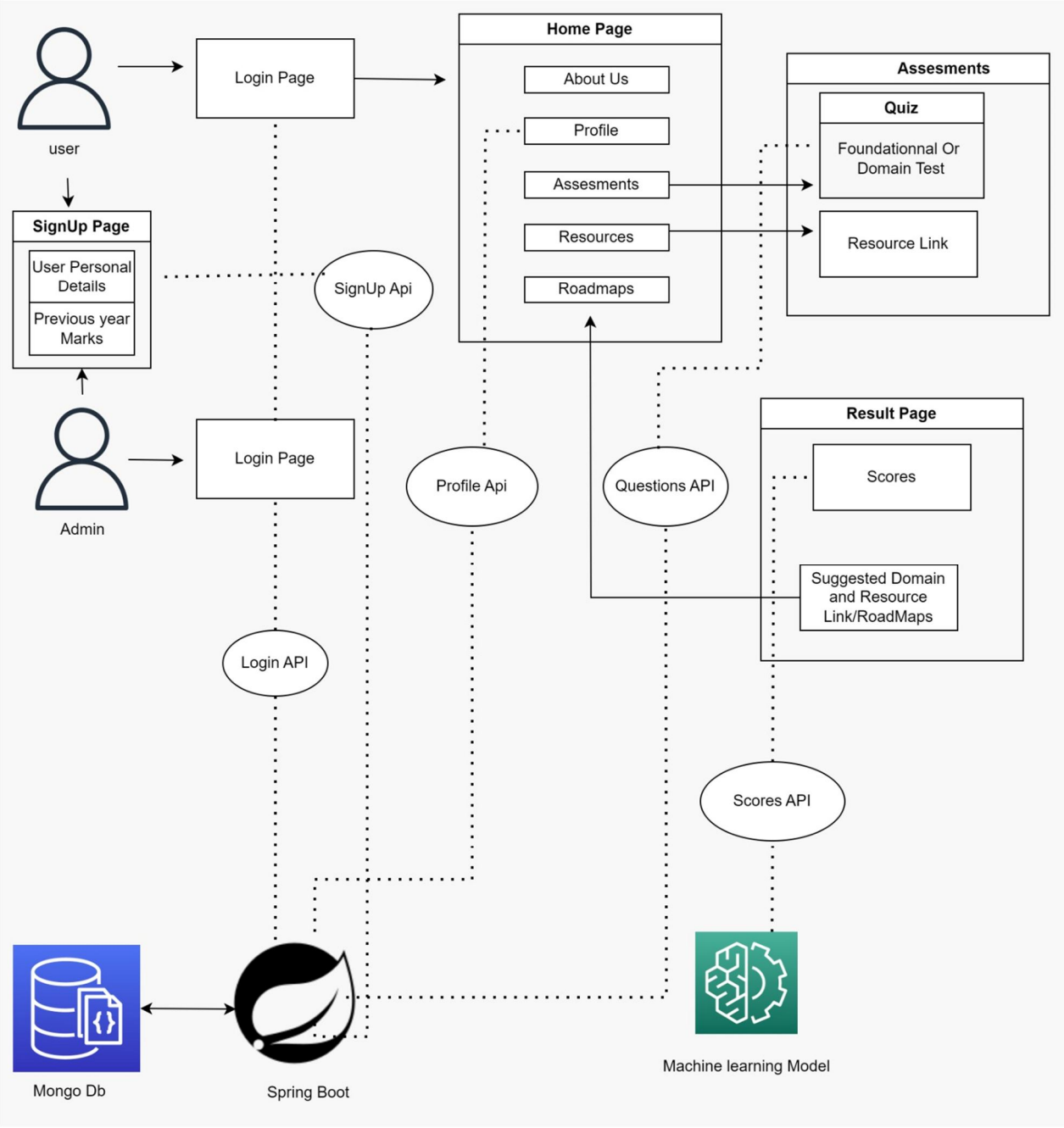
d) Versatility: It can handle both categorical and numerical features without requiring extensive data pre-processing.

The pickle library is used for storing the model into a file and then using it later. This file is loaded into the predict function into the Django server.

The Django server receives JSON input on its '/predict' API which contains the scores a foundational subject. These scores are given as an input to the model and the model gives a prediction of the suitable domain. This suitable domain is returned in the response body.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

C. Activity Diagram



The activity diagram represents the flow of activities in a Career Guidance System that helps individuals discover suitable career paths based on a test. Let's go through the main elements and their flow:

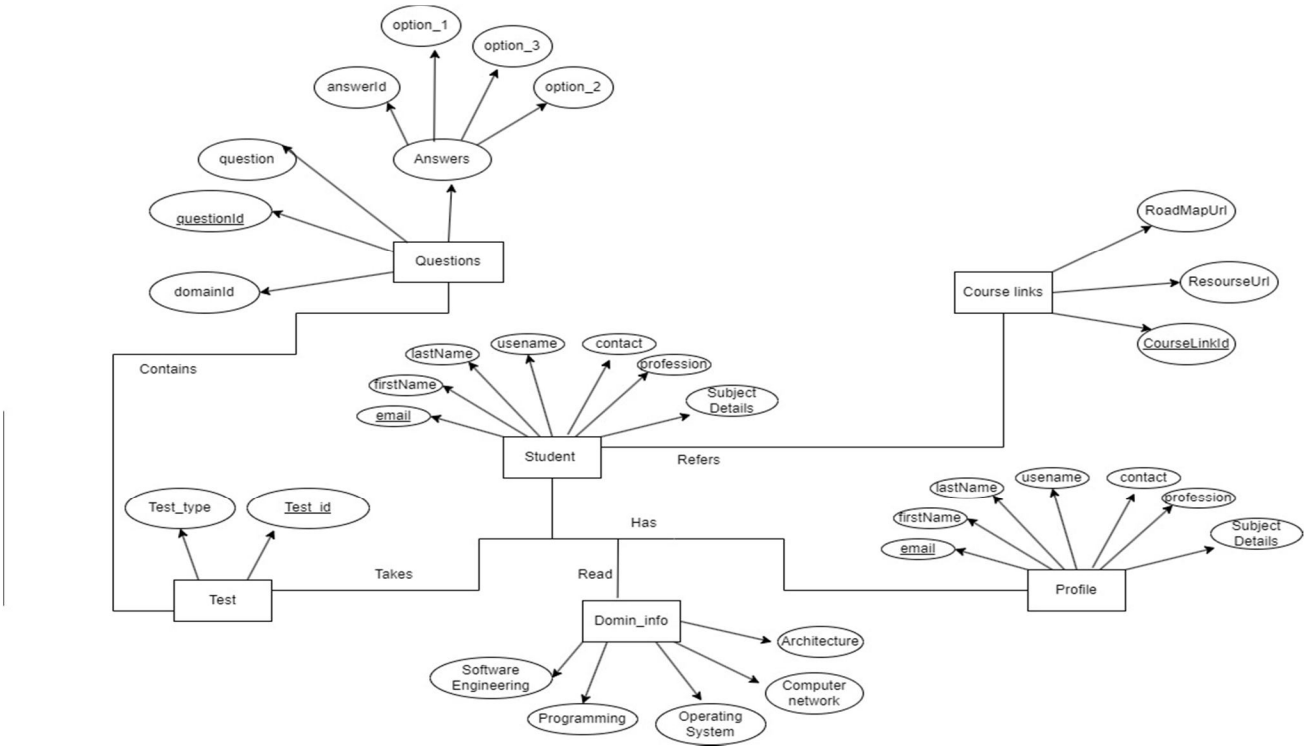
1) User Signup: The first activity is "User Signup/Login." This activity involves users creating an account or providing their basic information to access the career guidance system. Signup/Login API is used in this activity.
2) Home: Once the user is logged in it can choose any options from home page to get started, update information, taking tests or for getting guidance.
3) Take Assessment: After signup, users proceed to the "Take Assessment" activity. Here, they are presented with a career assessment test comprising various questions related to their interests, skills, and preferences. Questions API is used in this activity

4) Result: Once the user completes the test, the system moves to the "Result" activity. This activity involves analysing the user's responses and determining their strengths, interests, and potential career matches and suggest a suitable domain for career. Score API is used to get suggested domain from predict API which is the classification result of ML Classification Model

5) Resources/Roadmaps: Once the user gets the suitable domain the system moves to “Roadmap/Resources” activity where user gets all the information and guidance related about domain suitable for them.

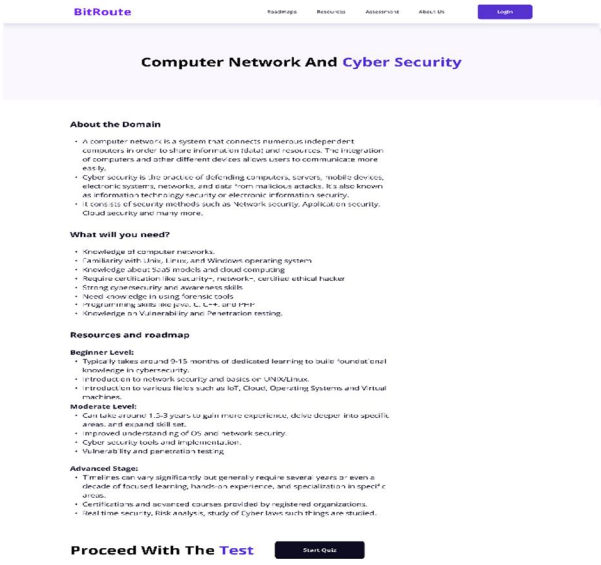
All the activities Communicate to MongoDB database with the help of Spring Boot API’s.

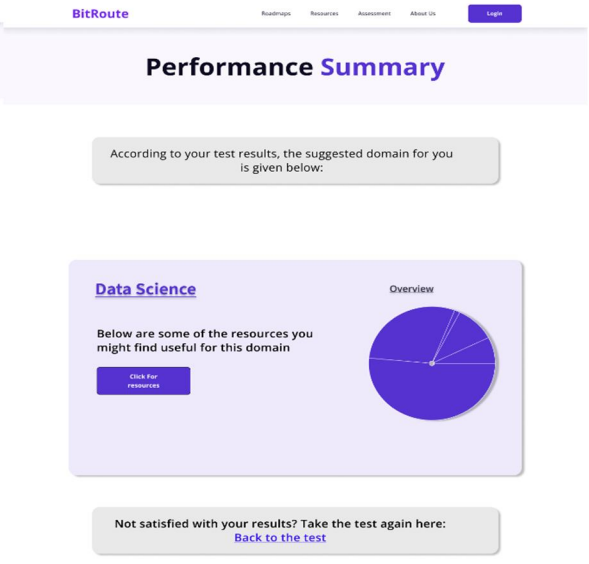
D. ER Diagram



IV. RESULTS

Below are some generated images that represent the user interface of the web application.





V. CONCLUSIONS

In conclusion, this research paper demonstrates the effectiveness of a web-based platform integrated with a machine learning model-based aptitude test. By offering personalized guidance and learning resources, the platform successfully assists students in pursuing diverse domains. The findings emphasize its potential in helping students identify strengths, explore different areas, and make informed career choices. This integration of machine learning models within a web-based platform holds promise for enhancing career guidance and educational experiences. Further research can optimize and expand upon this implementation to maximize its impact.

VI. FUTURE SCOPE

1) Personal progress of individuals
2) Games and tournaments/contests
3) Other domains and branches

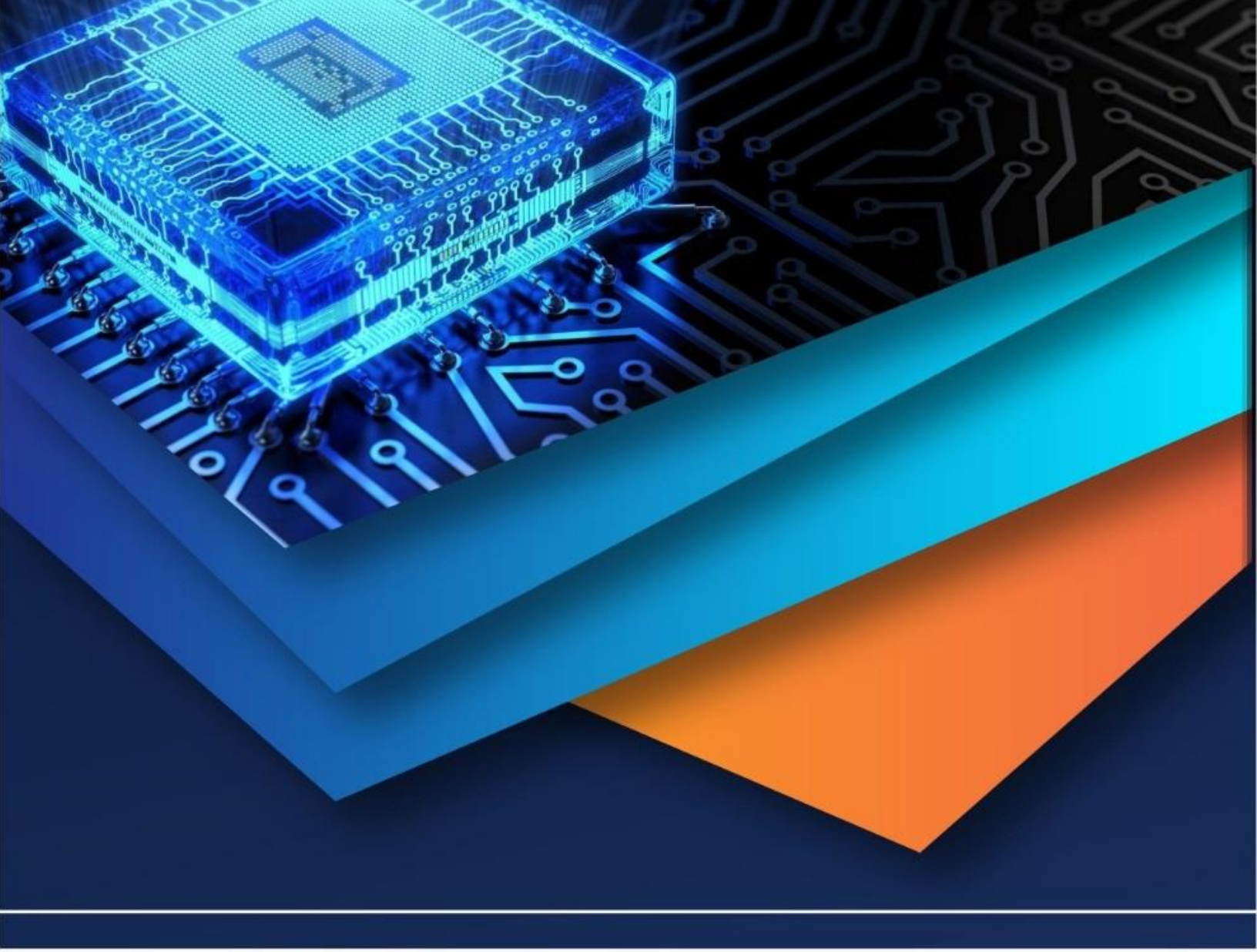
VII. ACKNOWLEDGEMENT

We gratefully acknowledge the invaluable contributions of various individuals and organizations that made this research project possible. Their support and collaboration have been instrumental in the successful execution of this endeavor. We would like to express our sincere gratitude to the following: Primarily, we would like to express our deepest gratitude to our research supervisor, Prof Dheeraj Patil, for his patience and guidance throughout this project. We'd like to acknowledge the Department of Information Technology of Nutan Maharashtra Institute of Information Technology for providing and supporting this research project. Lastly, we are also thankful to our colleagues, who provided help for this research project. We appreciate the knowledge and help they brought to this project. We would like to say thank to everyone who has been a part of this project.

REFERENCES

[1] Idakwo, J., Babatunde, A. J., & Kolajo, T. Development of a hybrid students' career path recommender system using machine learning techniques. Journal of Career Guidance and Development, 15(3), 123-145. 2022
[2] Vo, N. N. Y., Vu, Q. T., Vu, N. H., Vu, T. A., Mach, B. D., & Xu, G. Domain-specific NLP system to support learning path and curriculum design at tech universities. Journal of Educational Technology and Innovation, 10(3), 123-145 .2021
[3] Bhanuse, R. S., & Yenurkar, G. (2020). Recommendation of Branch of Engineering using machine learning. International Research Journal of Engineering and Technology (IRJET), 07(03), 5189. 2020
[4] Lokhande, P., Nayakwadi, Y., Umardand, M., Araj, V., & Patil, D. A Web-Based Platform for Predicting Student Careers using Machine Learning with Learning Resources for Computer Science Domains. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 11(3), 2243-2249. 2023

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)