PROJECT REPORT

UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

Domain : Applied Data Science
Team ID : PNT2022TMID11634
College Name : K.S. RANGASAMY COLLEGE OF TECHNOLOGY

Team Members

DHINESH KUMAR V [73771914111]
Department of Computer Science And Engineering

Guna M [73771914117]
Department of Computer Science And Engineering

Dhaneis G [73771914109]
Department of Computer Science And Engineering

Gokul Prasad R [73771914115]
Department of Computer Science And Engineering

PROJECT GUIDES

Industry Mentor: Sandesh P

Faculty Mentor: S VADIVEL

INDEX

1. INTRODUCTION

1.1 Project Overview:

The aim of this project is to help the students in shortlisting the universities and to help the students to save their time and money that they spend at the education consultancy firms. Also it will help them to limit the number of applications by providing them the right list of universities where they have the best chance of securing admission thus saving more money on the application fees. University and College research being one part of the university application process itself is an arduous and lengthy task. So, this predictor will help the students to choose the universities to apply based their profile. Hence, we have done this research project to solve that issue based on the cut off they have secured.

1.2 Purpose:

Learning data mining algorithms and implementing them in the real data sets. Building an efficient university research site for the students who have been planning to apply for master programs in various disciplines.Recommending best suitable universities to students based on their GRE, GPA and TOEFL scores and also predicting admission probability. It would definitely be easier for students if they get relief from step of selecting best suited universities and colleges for application. This project is to help the students to save their time and money that they have to spend at the education consultancy firms. And also it will help them to limit their number of application to a small number by proving them the suggestion of the universities where they have the best chance of securing admission thus saving more money on the application fees.The purpose of this analysis is to demonstrate the top contributing scores which helps the student to get the admission into the Master's degree program.

2. LITERATURE SURVEY

2.1 Existing problem:

The student details in separate records are tedious task,referring to all these record updating is needed .There is a chance for more manual errors.Previous research done in this area didn't consider all the factors which will contribute in the student admission process like TOEFL/IELTS, SOP, LOR and under graduate score. This model was developed to forecast the progress of students by comparing the score of students currently studying at university. The model thus predicts whether the student has chance of addmission to a university on the basis of various scores of students. The accuracy of the existing model is low.

2.2 References:

Paper-1

Title: University Admission Prediction using Google vertex AI.

Author: Kruthika, Apeksha ,Chinmaya , Madhumathi , Veena.

Year of publication: 2022

Methodology: The primary objective of this work is to make a Machine Learning model which could be utilized by understudies who need to seek after their Education. Many AI algorithms were used for this examination. Linear Regression model contrasted with different models gives the best outcome.At long last, understudies can have an open-source AI model which will assist the understudies with knowing their opportunity of entrance into a specific college with high exactness.

Advantage: At the point when the data to an algorithm is too enormous to ever be processed and it is suspected to be repetitive then it very well may be changed into a diminished arrangement of highlights. Deciding a subset of the initial features is called

feature selection.

Disadvantage: Particular student may or may not attend all of the exams that is given in the criteria.

Paper-2

Title: Personalized College Recommender and Cutoff Predictor for Direct Second Year Engineering.

Author: Abdul Ragab, Abdulfatah S Mashat, Ahmed Khedra.

Year of publication: 2022

Methodology: The system analyzes student academic merits, background, student records, and the college admission criteria. Then, it predicts the likelihood university college that a student may enter. In addition to the high prediction accuracy rate, flexibility is an advantage, as the system can predict suitable colleges that match the students' profiles and the suitable track channels through which the students are advised to enter. The system is adaptive, since it can be tuned up with other decision makers attributes performing trusted needed tasks faster and fairly.

Advantage: A prototype system is implemented and tested with live data available in the On Demand University Services (ODUS) database resources.

Disadvantage: It doesn’t work efficiently for smaller datasets while listing out the university.

Paper-3

Title: Prediction of the admission lines of college entrance examination based on machine learning.

Author: Zhenru Wang.

Year of publication:2016 2nd IEEE conference.

Methodology: The prediction of CEE scores is based on data statistics, probability

model and some weighted combination models. Since generating the model for predicting college admission lines uses too little reference factor, and the error is relatively large, so the reference value is very small. In this paper, machine learning methods are used to carry out the college admission lines of research and prediction. Specially, in this paper Adaboost algorithm is used to study and forecast, which belongs to ensemble learning. Finally, the result of this model is given, which is better than the current prediction method.

Advantage: Adam optimizer is that the learning rate does not need to be defined. The parameters had optimized depending on the number of weak learners and the learning rate value concerning the ensemble boosting classifier called Adaboost.

Disadvantage: Need more data in the future, we can also add university admission line Forecasts which are very significant work ,there are still a lot of things to be improved, and in the aspect of feature selection.

Paper-4

Title: Prediction Probability of Getting an Admission into a University using Machine Learning

Author: Sivasangari,Shivani,Bindhu,Deepa,Vignesh

Year of publication: 2021.

Methodology: In the present conditions, students regularly have difficulty finding a fitting institution to pursue higher studies based on their profile. There are some advisory administrations and online apps that recommend universities but they ask huge consultancy fees and online apps are not accurate. So, the aim of this research is to develop a model that predict the percentage of chances into the university accurately. This model provides also the analysis of scores versus chance of prediction based on historical data so that students can understand whether their profile is suitable or not. The proposed model uses linear regression and random forest algorithms but cat boost algorithm is giving highest accuracy.

Advantage: The proposed model gives a clear idea about the chances of eligibility of the student into the particular list of university.

Disadvantage: The eligibility criteria is mostly decided based on the CGPA, mark of the

student since it s considered more important.

Paper-5

Title: Prediction Probability of Getting an Admission into a University using Machine Learning

Author: Mukesh Kumar, A J Singh, Disha Handa

Year of publication: 2017

Methodology: College undergraduates frequently have an inclination to ponder over the chance that their profile suits the college requirements. Computer programs are exceptionally well trained and faster than humans in making decisions. Moreover, the cost of admission in a college is a lot, making it very crucial for a student that their profile gets shortlisted for a university admission.

Advantage: The proposed method considers diverse variables related to the student and his score in various tests. The dataset includes LOR, GRE score, CGPA, TOEFL score, University rating, SOP, etc. Based on all these criterias, the admission to a particular university of an undergraduate will be predicted.

Disadvantage: A particular student may or may not attend all of the exams that is given in the criteria

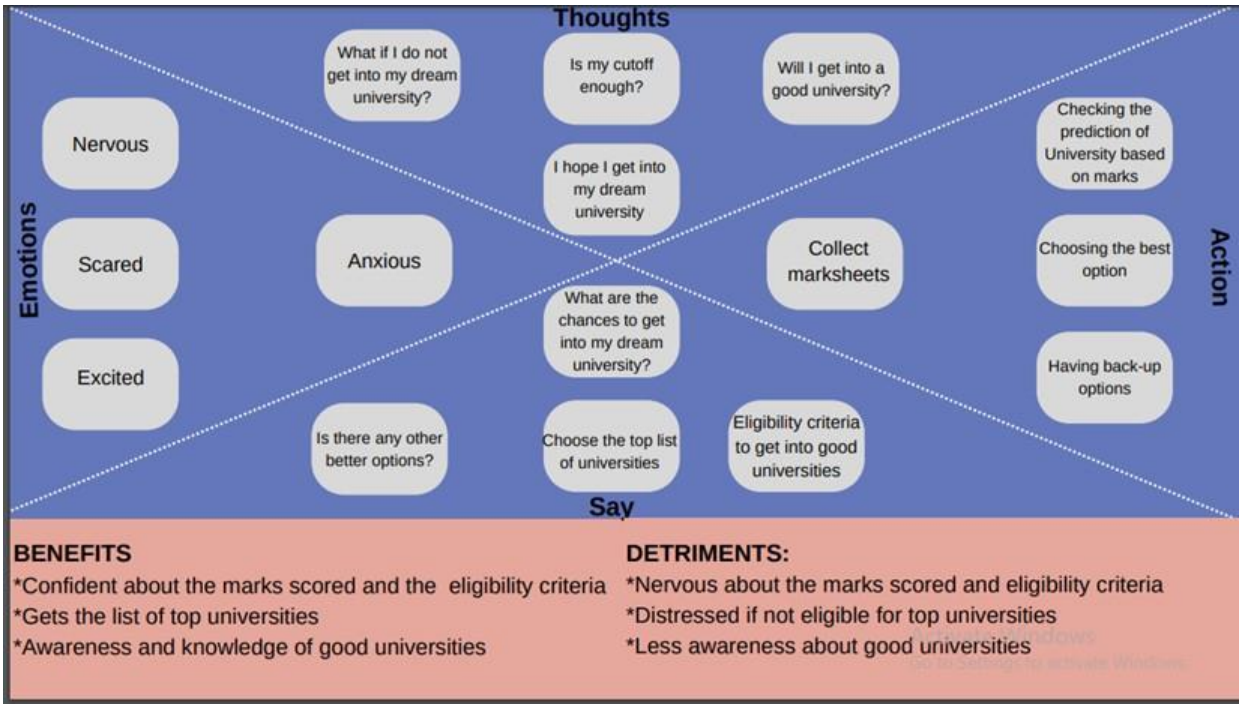
2.3 Problem Statement:

To create a web application that predicts a user's chances of admissions in the university of their choice for studying in abroad. The shortlisting of universities for a student can be done based on their profile details so that the students do not spend too much money in application fee charged by the university for the application process.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas:

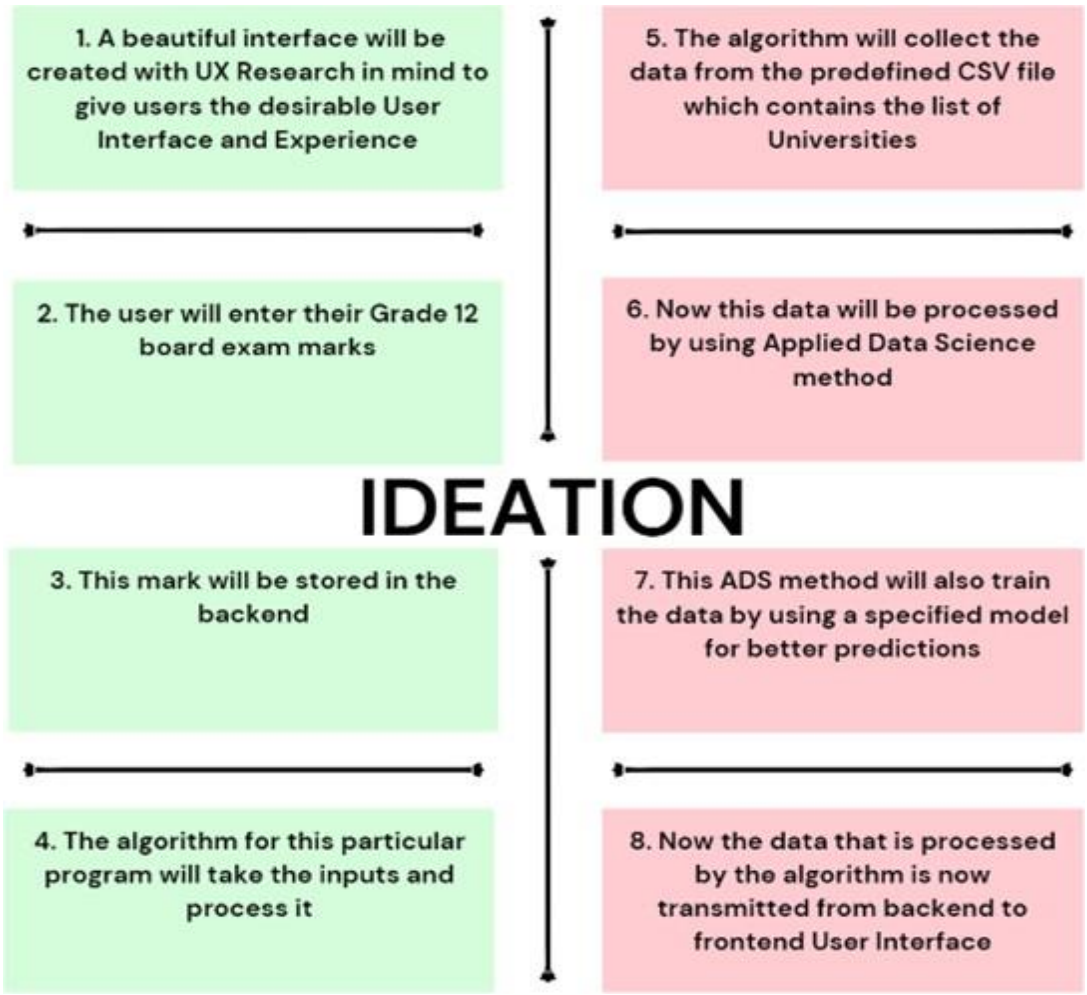
An empathy map canvas is a more in-depth version of the original empathy map, which helps identify and describe the user's needs and pain points. And this is valuable information for improving the user experience. An empathy map canvas serves as a foundation for outstanding user experiences, which focus on providing the experience customers want rather than forcing design teams to rely on guesswork.



3.2 Ideation & Brainstorming:

Ideation is often closely related to the practice of brainstorming a specific technique that is utilized to general new ideas. A principal different between ideation and brainstorming is that ideation is commonly more thought of as begin an individual pursuit, while brainstorming is almost always a group activity.

Brainstorming is usually conducted by getting a group of people together to come up with either general new ideas or ideas for solving a specific problem or dealing with a specific situation.



3.3 Proposed Solution:

S.No	Parameter	Description
1.	Problem Statement	Students are constantly worried about getting admission into their dream university, so inorder to overcome this we are implementing this project to fulfill these objectives. This project will help students to shortlist the universities they are eligible based on their profile. They will also have a clear idea about which universities they can aid after successful filteration. The project also creates direct contact between the students and the universities to avoid intermediates.
2.	Idea description	This project plans to compute the chances of acceptance into a university based on the student's profile. The key aspects considered for shortlisting are: i) CGPA of undergraduate degree ii) GRE & TOEFL/IELTS scores iii) SOP & LOR iv) Related work or research experience v) Extra-curricular activities. Inorder to find the chances of acceptance, various ML models such as Logistic Regression, Multiple Linear Regression, Decision Tree and Random Forest will be used. Analysis of the model along with parameters such as precision, accuracy and recall should also be provided.

3.	Novelty / Uniqueness	The intention of this project is to develop a hybrid model based on deep learning and also need to provide better accuracy than the previous existing models.The web application will also provide a feedback report to the students about the areas the student should show improvement.
4.	Customer Satisfaction and Social Impact	Students find it very difficult to shortlist the universities because of the different requirements and pre-requisites of a university. The application cost for applying to a particular university is also different which also is a tricky part. A university eligibility and admission predictor will be very helpful for the students to determine the probability of acceptance into that particular university. This system also does not charge any payment of fees for the profile analysis of the student and to give the shortlist of universities.

5.	Business Model	By placing advertisements of different universities in the web app, we can generate revenue from the application through advertisements. In future, a separate premium plan can be created where the students can make video calls through online platform and interact with the professors and alumni of the university.

6.	Scalability of the Solution	In future we can also have a chat space where the students can clarify their doubts with the professors, alumni and current students instantly.In order to deal with huge volumes of data in the future, cloud based storages(IBM Cloud, AWS, Microsoft Azure) and NoSQL databases(MongoDB, Redis) could be used instead of the traditional RDBMS storage. Distributed big data processing techniques could be initiated if the number of users using the website increase exponentially in the future.

3.4 Problem Solution Fit:

<table>
<tr><td>1.CUSTOMER SEGMENTS
Candidates who have recently completed Their school or university education and those seeking admission into a reputed institution</td><td>6.CUSTOMER CONSTRAINTS
Customers may not trust the accuracy and legitimacy of predictor and hence hesitate to use. Since users provide their confidential data ,they expect that it does not get misused.</td><td>5.AVAILABLE SOLUTIONS
in additions to candidates grade and GPA ,we also consider IELTS/TOEFL,GRE which also plays an important role in the admission process</td></tr>
<tr><td>2. JOBS-TO-BE-DONE/PROBLEMS
Collection of proper data in designing predictors is the most important step. The data of the user should be secure</td><td>9.PROBLEM ROOT CAUSE
If the collected data is inaccurate t6hen it will create an impact on the reliability of predictor. If the customers find our predictor unsafe or at risk, then they will hesitate to use it</td><td>7.BEHAVIOUR
The foremost aspect of a predictor from a customer point of view is the accuracy as it is approved based on its results</td></tr>
<tr><td>3.TRIGGERS
Comparison between the actual and desired results can be provided by the user</td><td rowspan="2">10.YOUR SOLUTION
Using the collected data ,a predictor needs to be designed with proper accuracy and reliability. Checking whether the data is secured is also important.</td><td rowspan="2">8.CHANNELS OF BEHAVIOUR
Customers can look for reliable predictors online and rate them based on their experience.These predictors can be analysed by students among their peer group and if they like it ,they might also recommend it from others.</td></tr>
<tr><td>4.Emotions: Before/After
Users will feel completely confident about the admission process because of the security and trust provided by the predictor</td></tr>
</table>

4. REQUIREMENT ANALYSIS

4.1 Functional Requirements:

Functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR 1	User Registration	Registration through Form Registration through mail Registration through LinkedIn
FR 2	User Confirmation	Confirmation via Email Confirmation via OTP
FR 3	User Login	Login with username and password Login through Gmail Login through LinkedIn
FR 4	Administration work	Check certified candidate's details Make allotment

FR 6	Local counsellor	Issue the final allotment order

4.2 Non-functional Requirements:

Non-functional requirements of the proposed solution.

NFR No.	Non Functional Requirement	Sub Requirement (Story / Sub-Task)
NFR 1	Usability	i)A logical interface is essential for the ease of the system to speed up all the common tasks ii) The product can be used mainly by administrators and other users.
NFR 2	Security	Some of the factors that are identified to protect the software from malicious access, destruction are described below: i)Keep specific history data sets. ii)Utilize certain cryptographic techniques iii) Check data integrity for critical variables iv) Communication needs to be restricted when the application is validating the user v) Every user should be licensed to use the system vi) Restrict the number of systems that can access the online admission site

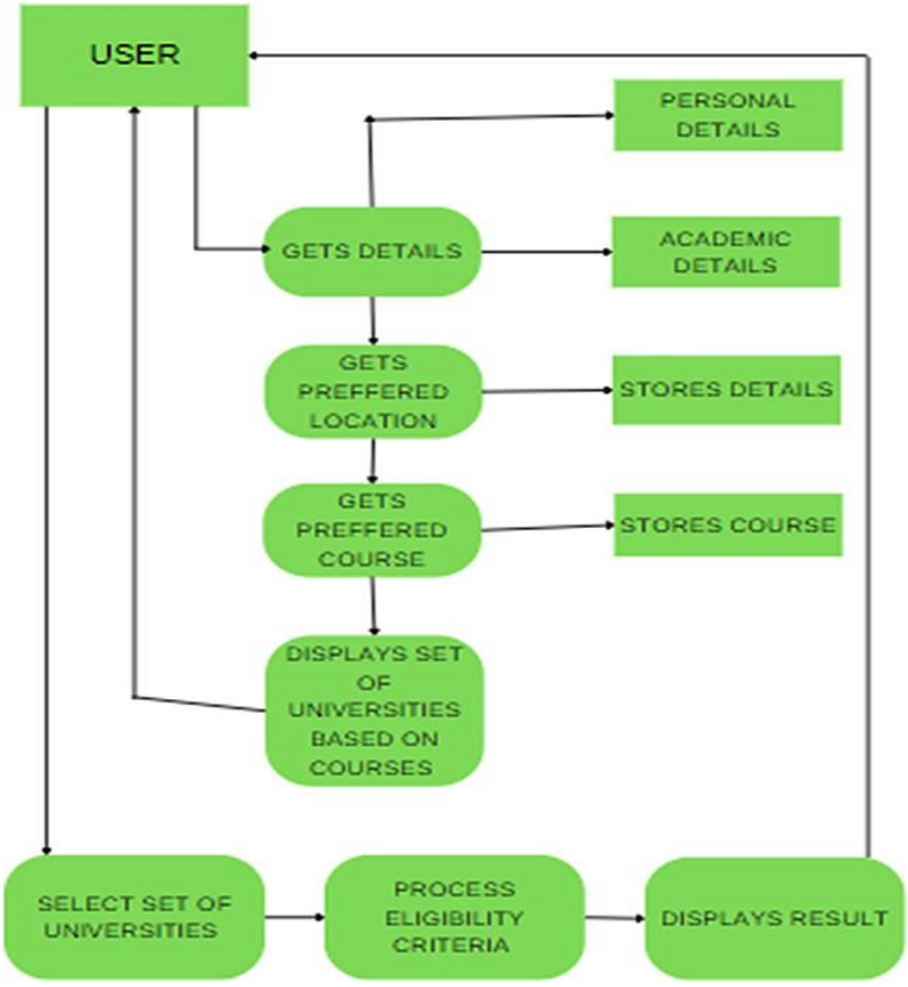
NFR 3	Reliability	i) All data storage for user variables will be committed to the database at the time of entry. ii) Data corruption is prevented by applying the possible backup procedures and techniques.
NFR 4	Performance	i) The database should be able to accommodate a minimum of 10,000 profile records of students. ii) At any instant the system should support use of multiple users at a time. iii) Availability of the requested university should be presented to the student in maximum of two seconds, so retrieving of data should be reliable. iv) As each student is given a maximum time of 10 minutes, accessing from the database should be done at relevant speed.
NFR 5	Availability	The system should be designed in such a way that it can be accessed by the user anytime from any place. In case of increase of hardware and database failure, a replacement page will be shown. Database can be retrieved from the data folder.

NFR 6	Scalability	Assesses the highest workloads where the performance of the system should not be compromised. It deals with the system's response time under different load condition requirements. Example: The system must be scalable enough to support 1,000,000 visits at the same time while maintaining accuracy and optimal performance.

5. PROJECT DESIGN

5.1 Data Flow Diagrams:

A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method (SSADM).



A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method (SSADM).

5.2 SOLUTION AND TECHNOLOGY ARCHITECTURE

SUMMARY:

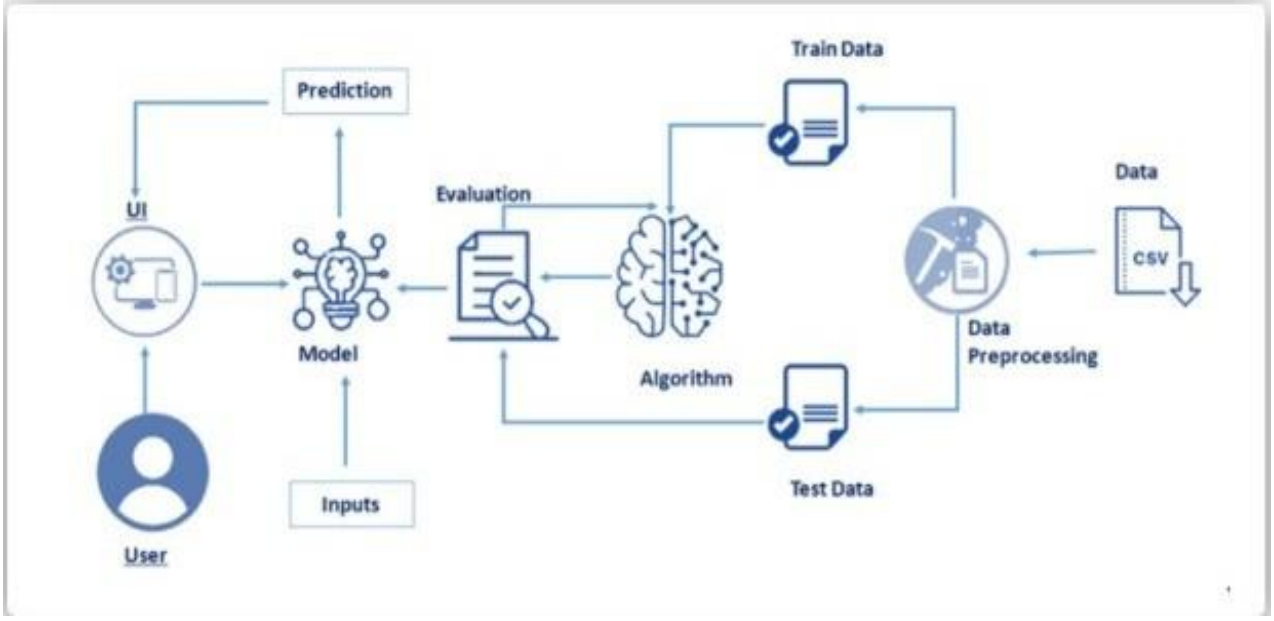
The university admit eligibility predictor is find the applicant's ability to admit to their desired university.

Components and technology:

S.No	Component	Description	Technology
1.	User Interface	Front end part of the application for accepting the user's data	Flask, Streamlit
2.	Dataset pre-processing	Removing unpredictability in the dataset	Pandas, Numpy, Python
3.	Application Logic	Main business logic for the application	Python
4.	Database	Used for storing data about the student and universities	MySQL, IBM DB2, IBMCloudant, etc.
5.	Data Visualization	Graphical visualization of student data , university's previous acceptance rate, heat map representing the correlation of different attributes	Matplotlib, Seaborn,Plotly
6.	File Storage	Used for storing SOPs, LORs, grades and other documents uploaded by	IBM Cloud File Storage

		the user	
7.	ML Model	Models used for prediction - Logistic Regression, DTree, Random Forest and Hybrid Deep Learning based model	Scikit-Learn

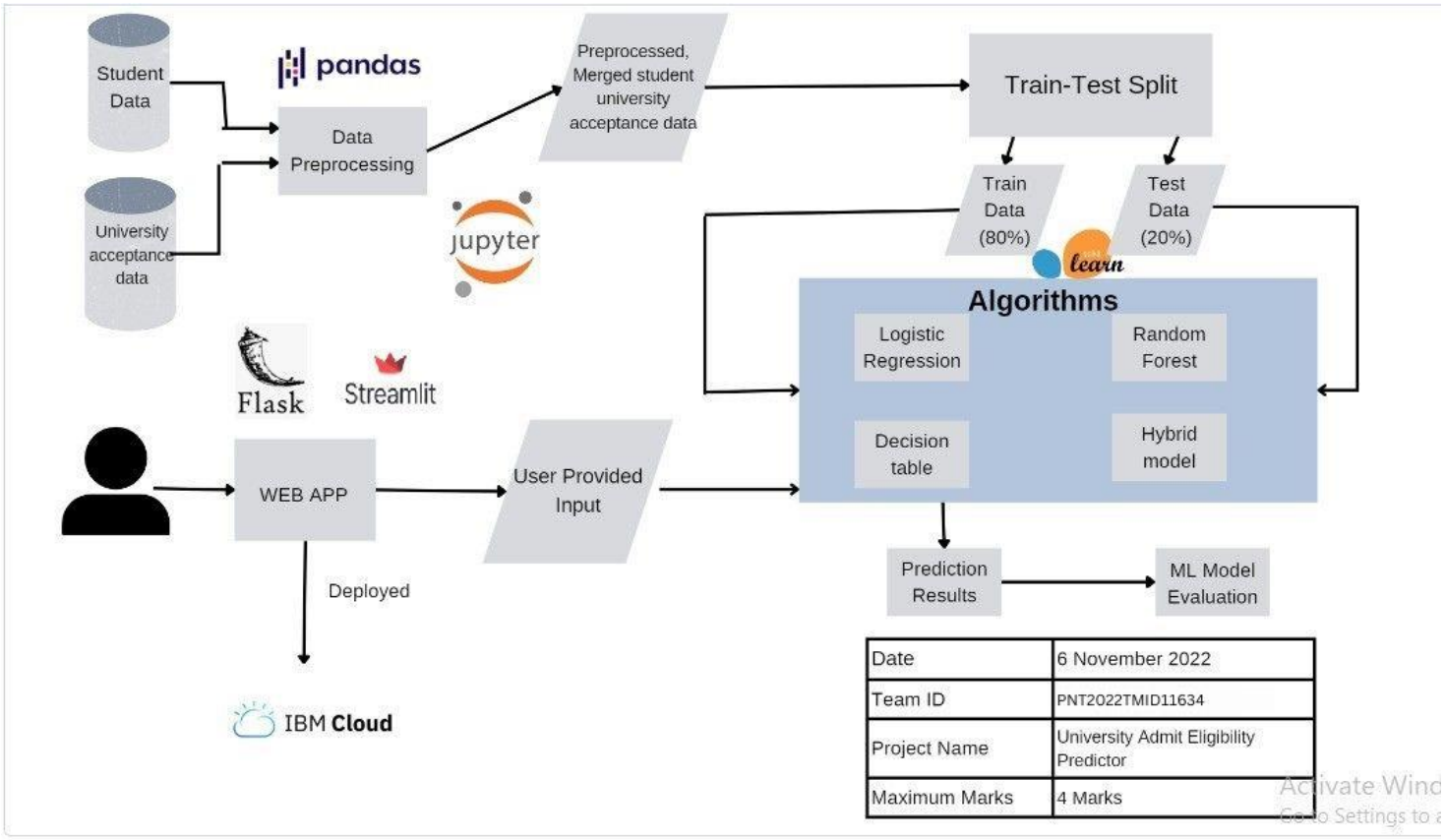
8.	Performance Metrics	Accuracy of the ML model on the trained and tested data	Root Mean SquaredLogarithmic Error(RMSLE), MeanSquared Error (MSE)
9.	Infrastructure	Cloud Server Configuration for hosting the web app	IBM Cloud Hosting



Application Characteristics:

S.No	Characteristics	Description	Technologies Used
1.	Security Implementiation	Authentication of the user is crucial before making predictions	Cloud authenticationserviceswith modern,secure encryptionschemes like SHA256
2,	Availability	As the web app is hosted on cloud, it is accessible and supported by any device from anywhere. Load balancing is also implemented using IBM cloud services to distribute the load across multiple servers.	IBM Cloud Hosting,IBM Load Balancer
3.	Performance	Four different ML models need to be implemented - Logistic Regression, Decision Tree, Random Forest and Hybrid model. The highest accurate model is obtained after comparing the model accuracy and recall values.	Scikit-Learn, RootMean SquaredLogarithmic Error(RMSLE), MeanSquared Error(MSE)

Architecture Diagram:



Date	6 November 2022
Team ID	PNT2022TMID11634
Project Name	University Admit Eligibility Predictor
Maximum Marks	4 Marks

5.3 User Stories

Use the below template to list all the user stories for the product.

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, we can register for the application by using our email and password credentials	2	High	2
Sprint-1		USN-2	As a user, the confirmation mail will be received to the registered email id through which the registration gets completed	1	Medium	1
Sprint-1	User login and authentication	USN-3	As a user, we will be able to login to the application using the same login credentials.	4	High	2
Sprint-2	Update information	USN-4	As a user, I have to update my user profile by entering the necessary details. As an admin, I can check the details	5	High	3

			entered by the user.			
Sprint-3	View courses	USN-5	As a user, I will be able to view the list of courses eligible based on my profile details	5	Medium	4
Sprint-3	Predictor	USN-6	As an admin, I can build and train the model using the details provided by the user and the ML algorithms	5	High	4
Sprint-4	Verification process	USN-7	As a user, I can view the details about the university's entry procedure, venue of counselling and verification	2	Low	2
Sprint-4	Confirmation	USN-8	As an admin, I can transfer the confirmation of the user for prediction into the database	3	High	3

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING & SCHEDULING:

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	The primary objective of this work is to make a Machine Learning model which could be utilized by understudies who need to seek after their Education. Many AI algorithms were used for this examination. Linear Regression model contrasted with different models gives the best outcome.At long last, understudies can have an open-source AI model which will assist the understudies with knowing their opportunity of entrance into a specific college with high exactness.	14 NOV 2022
Empathy Map	An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community	7 NOV 2022
Ideation	Ideation is the process where	6 NOV 2022

	you generate ideas and solutions through sessions such as Sketching, Prototyping, Brainstorming, Brainwriting, Worst Possible Idea, and a wealth of other ideation techniques. Ideation is also the third stage in the Design Thinking proces	
Proposed Solution	This project will help students to shortlist the universities they are eligible based on their profile. They will also have a clear idea about which universities they can aid after successful filteration	15 NOV 2022
Problem Solution Fit	This project plans to compute the chances of acceptance into a university based on the student's profile. The key aspects considered for shortlisting	9 NOV 2022

6.2 SPRINT DELIVERY SCHEDULE

Product Backlog, Sprint Schedule, and Estimation

Project Tracker, Velocity & Burndown Charts

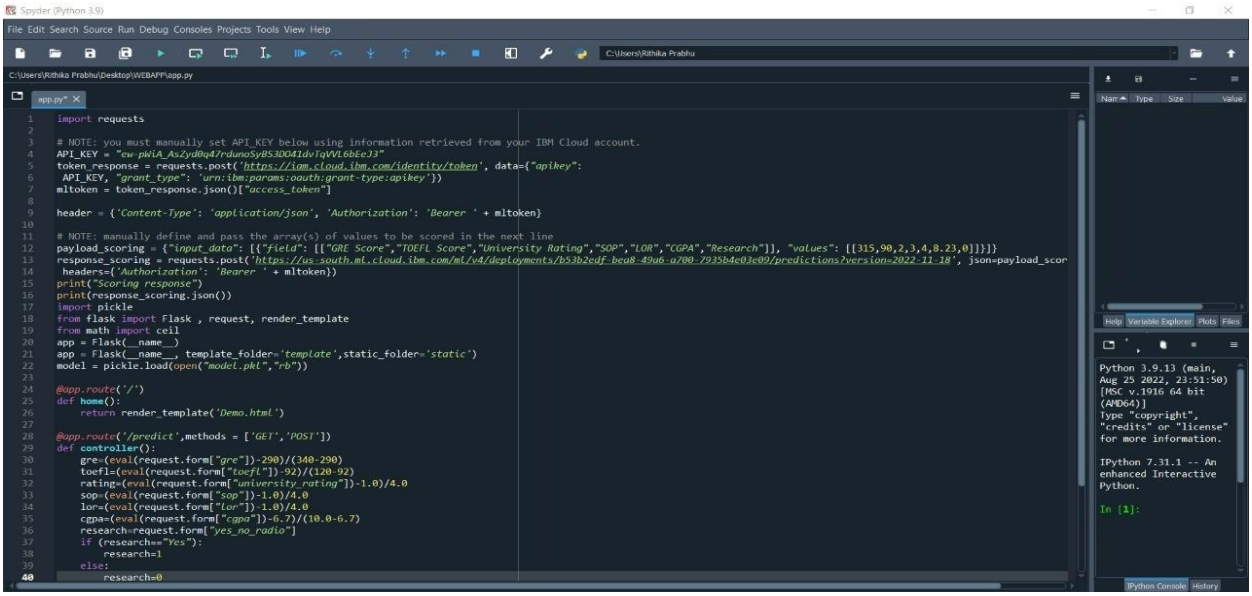
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	**7**	**6 Days**	4 November 2022	11 November 2022	**7**	11 November 2022
Sprint-2	**5**	**6 Days**	6 November 2022	14 November 2022	**5**	11 November 2022
Sprint-3	**10**	**6 Days**	9 November 2022	15 November 2022	**10**	14 November 2022
Sprint-4	**5**	**6 Days**	10 November 2022	15 November 2022	**3**	15 November 2022

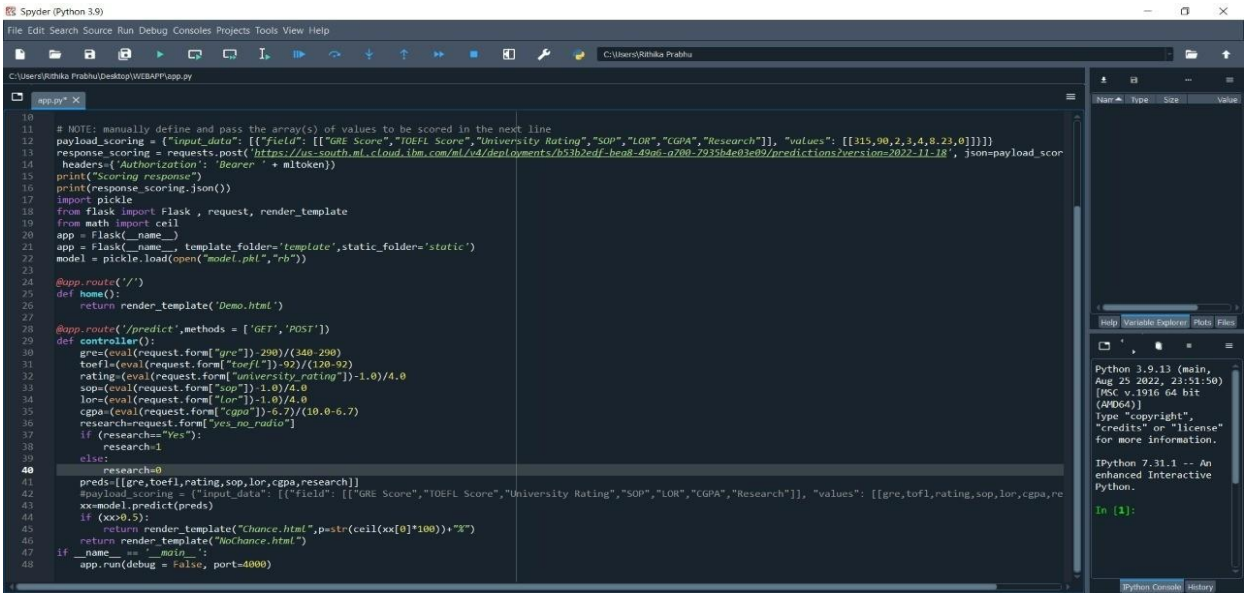
-

7. CODING AND SOLUTIONING

7.1 Feature1:

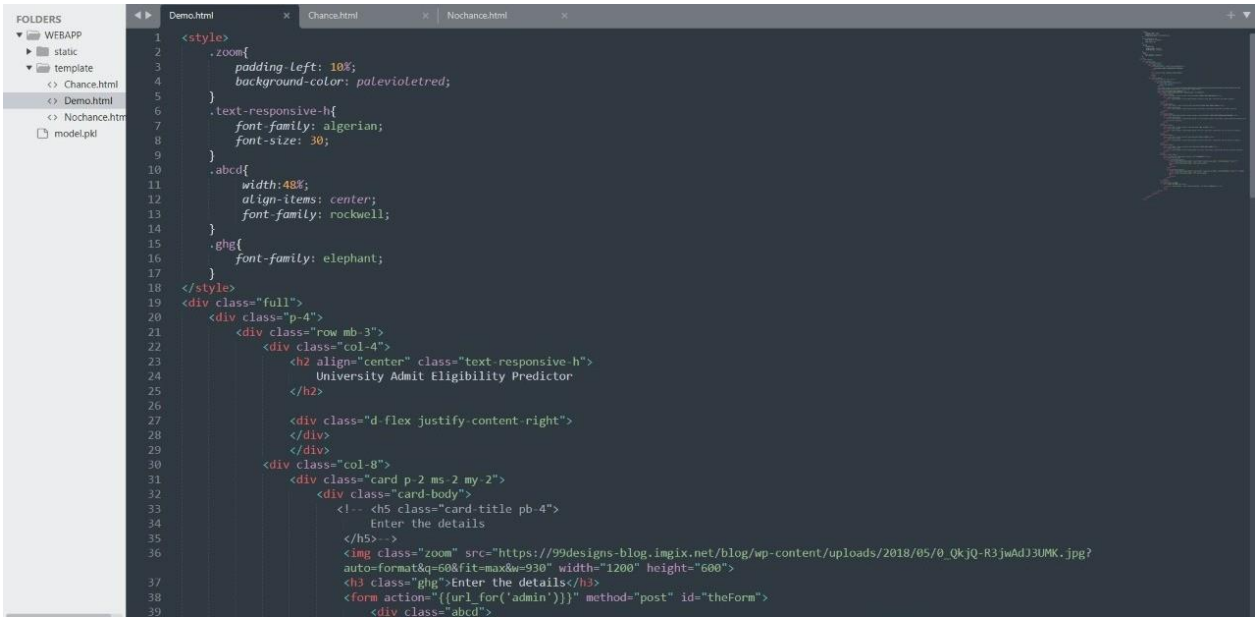
i. Analyzed university admission statistics.
ii. Developed tools for matching university (in percentile) using CGPA, GRE,(Verbal, Quantitative,Analytical Writing), TOEFL and IELTSscores.
iii. Languages : Python.
iv. Tools/IDE : Anaconda.
v. Libraries : Streamlit, Matplotlib, numpy, pandas, scikit-learn, seaborn





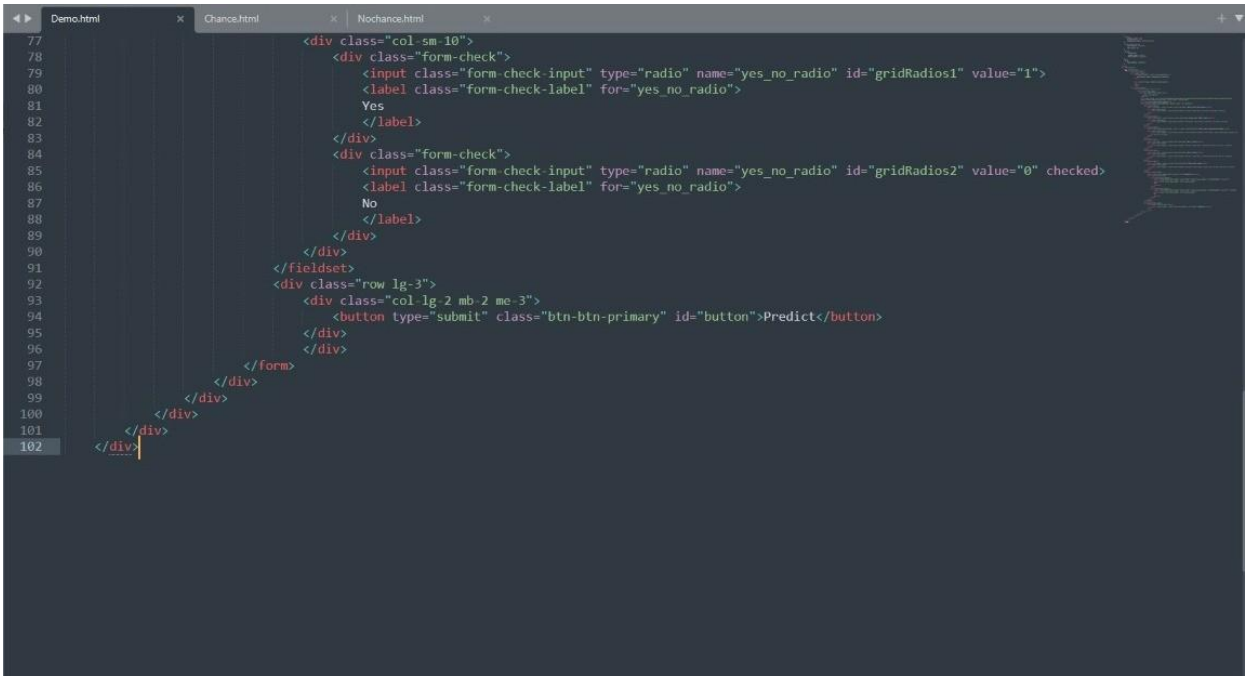
7.2 Feature2:

Demo.html



```
<style>
    .zoom{
        padding-left: 10%;
        background-color: palevioletred;
    }
    .text-responsive-h{
        font-family: algerian;
        font-size: 30;
    }
    .abcd{
         width:48%;
         align-items: center;
         font-family: rockwell;
    }
    .ghg{
        font-family: elephant;
    }
</style>
<div class="full">
    <div class="p-4">
        <div class="row mb-3">
            <div class="col-4">
                <h2 align="center" class="text-responsive-h">
                    University Admit Eligibility Predictor
                </h2>

                <div class="d-flex justify-content-right">
                </div>
                </div>
            <div class="col-8">
                <div class="card p-2 ms-2 my-2">
                    <div class="card-body">
                       <!-- <h5 class="card-title pb-4">
                            Enter the details
                        </h5>-->
                        <img class="zoom" src="https://99designs-blog.imgix.net/blog/wp-content/uploads/2018/05/0_QkjQ-R3jwAdJ3UMK.jpg?
                        auto=format&q=60&fit=max&w=930" width="1200" height="600">
                        <h3 class="ghg">Enter the details</h3>
                        <form action="{{url_for('admin')}}" method="post" id="theForm">
                            <div class="abcd">
```

```
                            <div class="abcd">
                                <label for="gre" class="col-lg-2 col-form-label">Enter your GRE Score:</label>
                                <div class="col-lg-10">
                                    <input type="number" class="form-control" id="gre" name="gre" min="260" max="340" required>
                                </div>
                            </div>
                            <div class="abcd">
                                <label for="toefl" class="col-lg-2 col-form-label">Enter your TOEFL Score:</label>
                                <div class="col-lg-10">
                                    <input type="number" class="form-control" id="toefl" name="toefl" min="50" max="120" required>
                                </div>
                            </div>
                            <div class="abcd">
                                <label for="university_rating" class="col-lg-2 col-form-label">Enter your University Rating:</label>
                                <div class="col-lg-10">
                                    <input type="number" class="form-control" id="university_rating" step="0.01" name="university_rating" min=
                                    "1" max="5" required>
                                </div>
                            </div>
                            <div class="abcd">
                                <label for="sop" class="col-lg-2 col-form-label">No. of SOP:</label>
                                <div class="col-lg-10">
                                    <input type="number" class="form-control" id="sop" name="sop" step="0.01" min="1" max="4" required>
                                </div>
                            </div>
                            <div class="abcd">
                                <label for="lor" class="col-lg-2 col-form-label">No. of LOR:</label>
                                <div class="col-lg-10">
                                    <input type="number" class="form-control" id="lor" name="lor" step="0.01" min="1" max="4" required>
                                </div>
                            </div>
                            <div class="abcd">
                                <label for="cgpa" class="col-lg-2 col-form-label">Enter your CGPA:</label>
                                <div class="col-lg-10">
                                    <input type="number" class="form-control" id="cgpa" name="cgpa" step="0.01" min="5" max="10" required>
                                </div>
                            </div>
                            <fieldset class="abcd">
                                <legend class="col-form-label col-sm-2 pt-0">Research:</legend>
                                <div class="col-sm-10">
```


Demo.html
Chance.html
Nochance.html

<input class="form-check-input" type="radio" name="yes_no_radio" id="gridRadios1" value="1">
<label class="form-check-label" for="yes_no_radio">
Yes
</label>

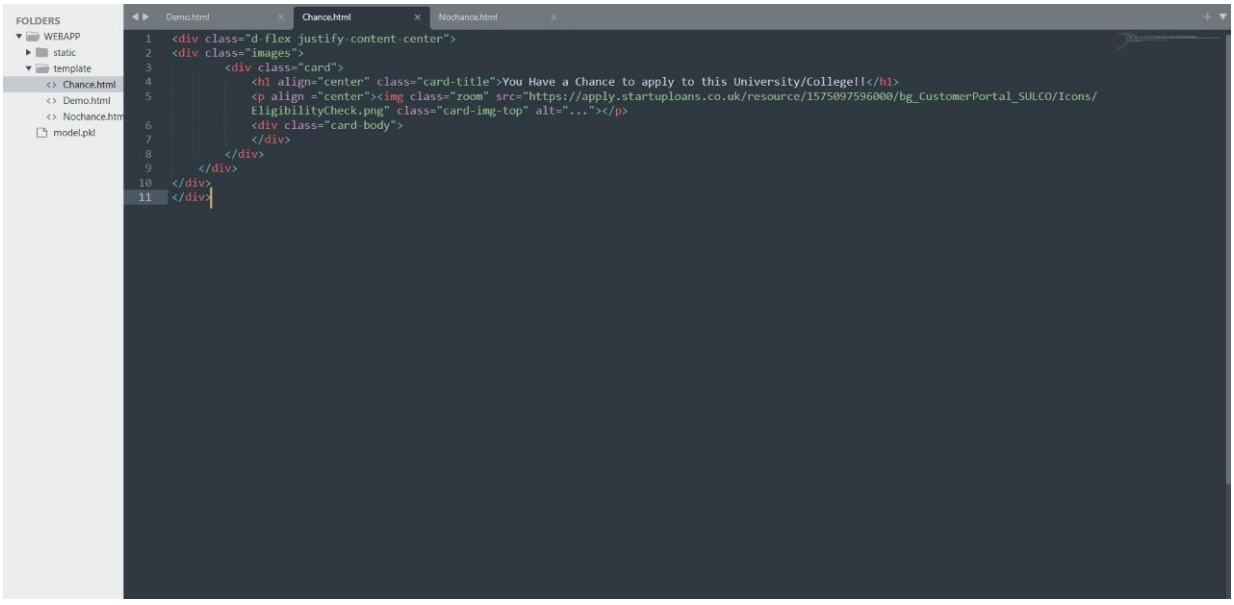
<input class="form-check-input" type="radio" name="yes_no_radio" id="gridRadios2" value="0" checked>
<label class="form-check-label" for="yes_no_radio">
No
</label>

</fieldset>

<button type="submit" class="btn-btn-primary" id="button">Predict</button>

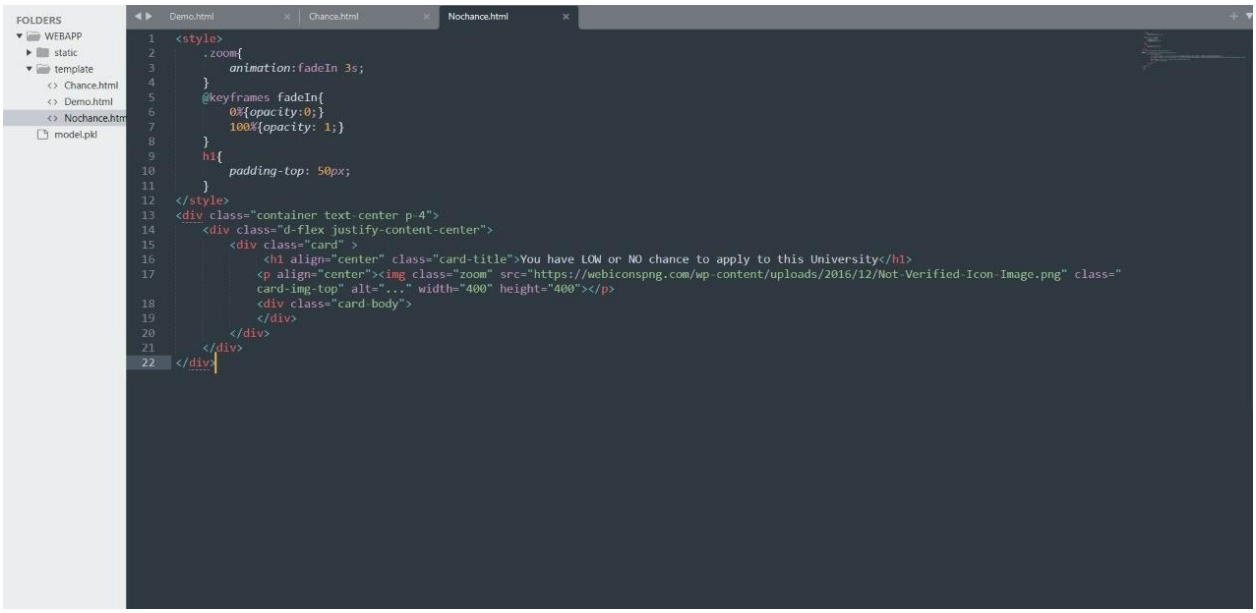
</form>

Chance.html


FOLDERS
WEBAPP
static
template
Chance.html
Demo.html
Nochance.html
model.pkl
Demo.html
Chance.html
Nochance.html

<h1 align="center" class="card-title">You Have a Chance to apply to this University/College!!</h1>
<img class="zoom" src="https://apply.startuploans.co.uk/resource/1575097596000/bg_CustomerPortal_SULCO/Icons/
EligibilityCheck.png" class="card-img-top" alt="...">

NoChance.html



7.3 Database Scheme:

Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
1	337	118	4	4.5	4.5	9.65	1	0.92
2	324	107	4	4	4.5	8.87	1	0.76
3	316	104	3	3	3.5	8	1	0.72
4	322	110	3	3.5	2.5	8.67	1	0.8
5	314	103	2	2	3	8.21	0	0.65
6	330	115	5	4.5	3	9.34	1	0.9
7	321	109	3	3	4	8.2	1	0.75
8	308	101	2	3	4	7.9	0	0.68
9	302	102	1	2	1.5	8	0	0.5
10	323	108	3	3.5	3	8.6	0	0.45
11	325	106	3	3.5	4	8.4	1	0.52
12	327	111	4	4	4.5	9	1	0.84
13	328	112	4	4	4.5	9.1	1	0.78
14	307	109	3	4	3	8	1	0.62
15	311	104	3	3.5	2	8.2	1	0.61
16	314	105	3	3.5	2.5	8.3	0	0.54
17	317	107	3	4	3	8.7	0	0.66
18	319	106	3	4	3	8	1	0.65
19	318	110	3	4	3	8.8	0	0.63
20	303	102	3	3.5	3	8.5	0	0.62
21	312	107	3	3	2	7.9	1	0.64
22	325	114	4	3	2	8.4	0	0.7
23	328	116	5	5	5	9.5	1	0.94

8.TESTING

In [ ]:
```
x_train
```

Out[ ]:

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research
3	322	110	3	3.5	2.5	8.67	1
18	318	110	3	4.0	3.0	8.80	0
202	340	120	5	4.5	4.5	9.91	1
250	320	104	3	3.0	2.5	8.57	1
274	315	100	1	2.0	2.5	7.95	0
...	...	...	...	...	...	...	...
71	336	112	5	5.0	5.0	9.76	1
106	329	111	4	4.5	4.5	9.18	1
270	306	105	2	2.5	3.0	8.22	1
348	302	99	1	2.0	2.0	7.25	0
102	314	106	2	4.0	3.5	8.25	0

320 rows × 7 columns

In [ ]:
```
multiple_lin_reg.predict(x_train)
```

Out[ ]:
```
array([[0.74604196],
       [0.73901061],
       [0.99499641],
       [0.71465428],
       [0.58666289],
       [0.70570088],
       [0.79635618],
       [0.72390065],
       [0.91430317],
       [0.65782685],
       [0.60293691],
       [0.84357917],
       [0.49457813],
       [0.72066904],
       [0.61590143],
       [0.63303675],
       [0.73458027],
       [0.77284896],
       [0.54224384],
       [0.7740777 ],
       [0.67388358],
       [0.68770413],
       [0.71751013],
       [0.79218791],
       [0.74689676],
       [0.60135562],
       [0.6550291 ],
       [0.65024192],
       [0.64937148],
       [0.61847543],
       [0.71837168],
       [0.62572932],
       [0.56180503],
       [0.58932197],
       [0.82917531],
       [0.60765804],
       [0.45309083],
       [0.88506537],
       [0.772133  ],
       [0.81857796],
       [0.62930747],
       [0.705755  ],
```

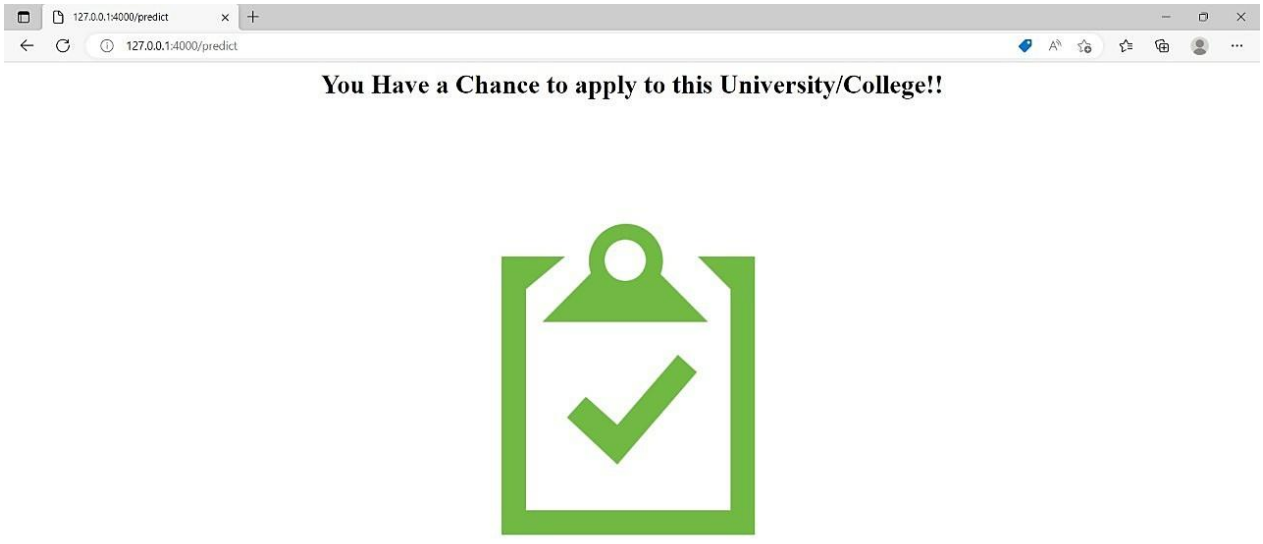
In [ ]:

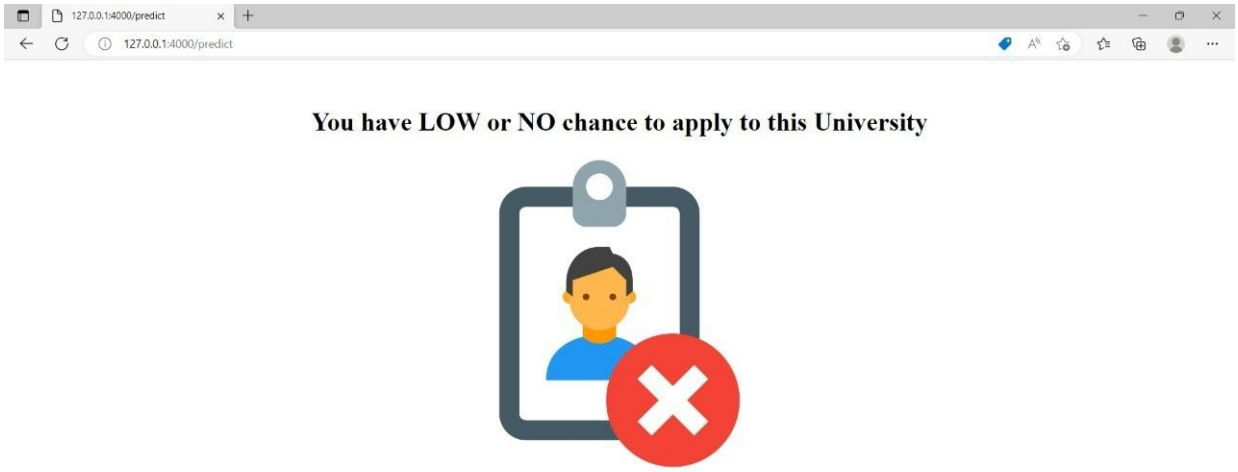
```
import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = ""
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

# NOTE: manually define and pass the array(s) of values to be scored in the next line
payload_scoring = {"input_data": [{"field": [["GRE Score","TOEFL Score","University Rating","SOP","LOR","CGPA","Research"]], "values": [[315,90,2,3,4,

response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/b53b2edf-bea8-49a6-a700-7935b4e03e09/predictions?version=2022-11
 headers={'Authorization': 'Bearer ' + mltoken})
print("Scoring response")
print(response_scoring.json())
```

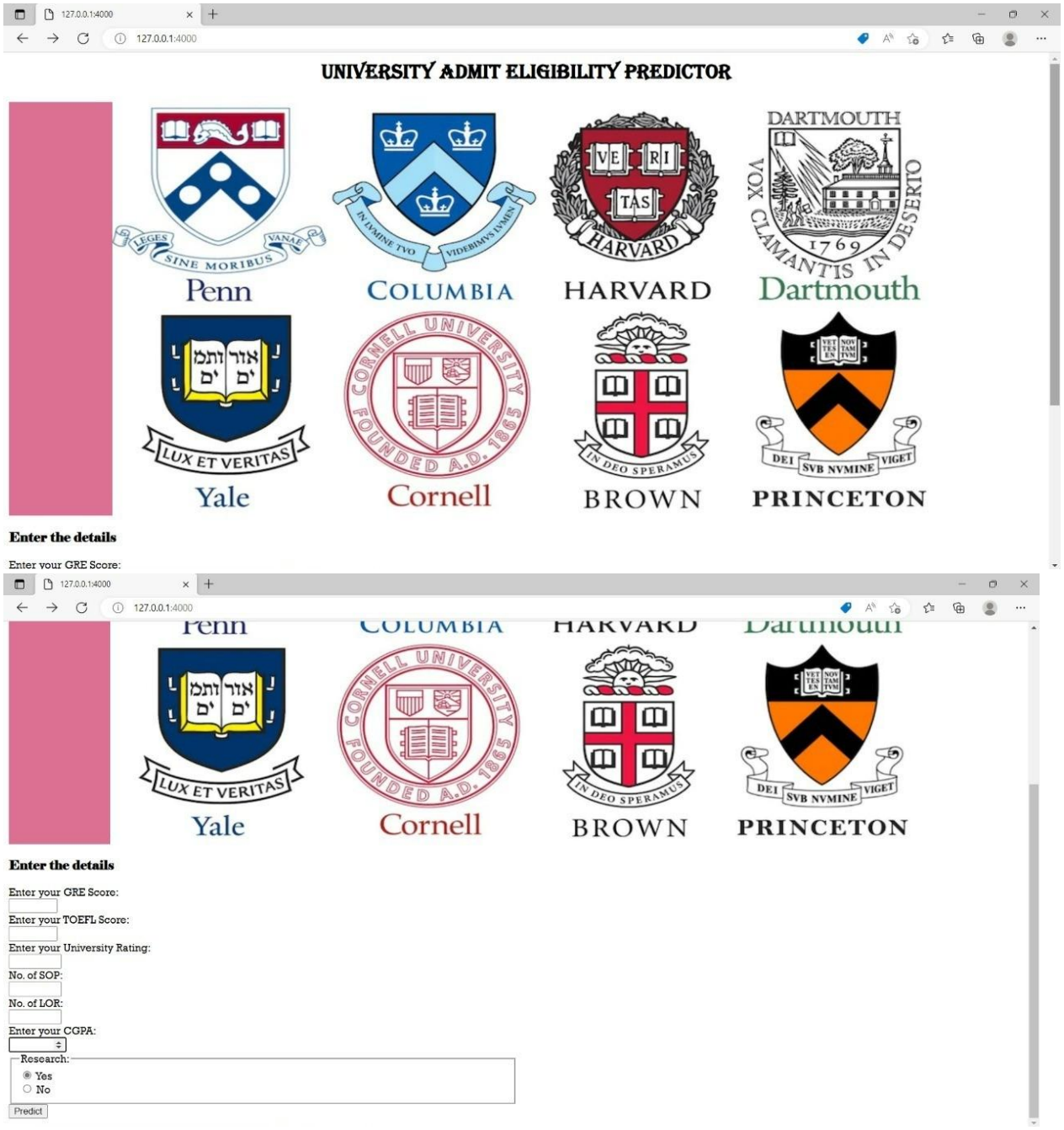
```
Scoring response
{'predictions': [{'fields': ['prediction'], 'values': [[[0.6222734570109485]]]]}]}
```

9. RESULTS

9.1 Performance Metrics:






127.0.0.1:4000
127.0.0.1:4000
UNIVERSITY ADMIT ELIGIBILITY PREDICTOR
Penn
COLUMBIA
HARVARD
Dartmouth
Yale
Cornell
BROWN
PRINCETON
Enter the details
Enter your GRE Score:
127.0.0.1:4000
127.0.0.1:4000
Penn
COLUMBIA
HARVARD
Dartmouth
Yale
Cornell
BROWN
PRINCETON
Enter the details
Enter your GRE Score:
Enter your TOEFL Score:
Enter your University Rating:
No. of SOP:
No. of LOR:
Enter your CGPA:
Research:
Yes
No
Predict

10. ADVANTAGES:

- Reach to geographically scattered student. One of the important objectives of the admission system is communicate with all the students scattered geographically.
- Reducing time in activities: Reduce the time taken process the applications of students ,admitting a student, conducting the online examination, verify student marks, and send call letters to selected students.
- Centralized data handling: Transfer the data smoothly to all the departments involved and handle the data centralized way.
- Paperless admission with reduced man power : Reduce the manpower needed to perform all the admission and administration task by reducing the paper works Cost cutting. Reduce the cost involved in the admission process
- Operational efficiency: Improve the operational efficiency by improving the quality of the process.
- Manage large number of student details.
- Manage all details of student who registered for the course.

DISADVANTAGES:

- Require much man power i.e. much efforts, much cost and hard to operate and maintain.
- Since, all the work is done in papers so it is very hard to locate a particular student record when it is required.
- Admissions also depend on the individual university’s policy regarding the intake offoreign studentsand is not modeled by our system.
- Required active internet connection.

11. CONCLUSION

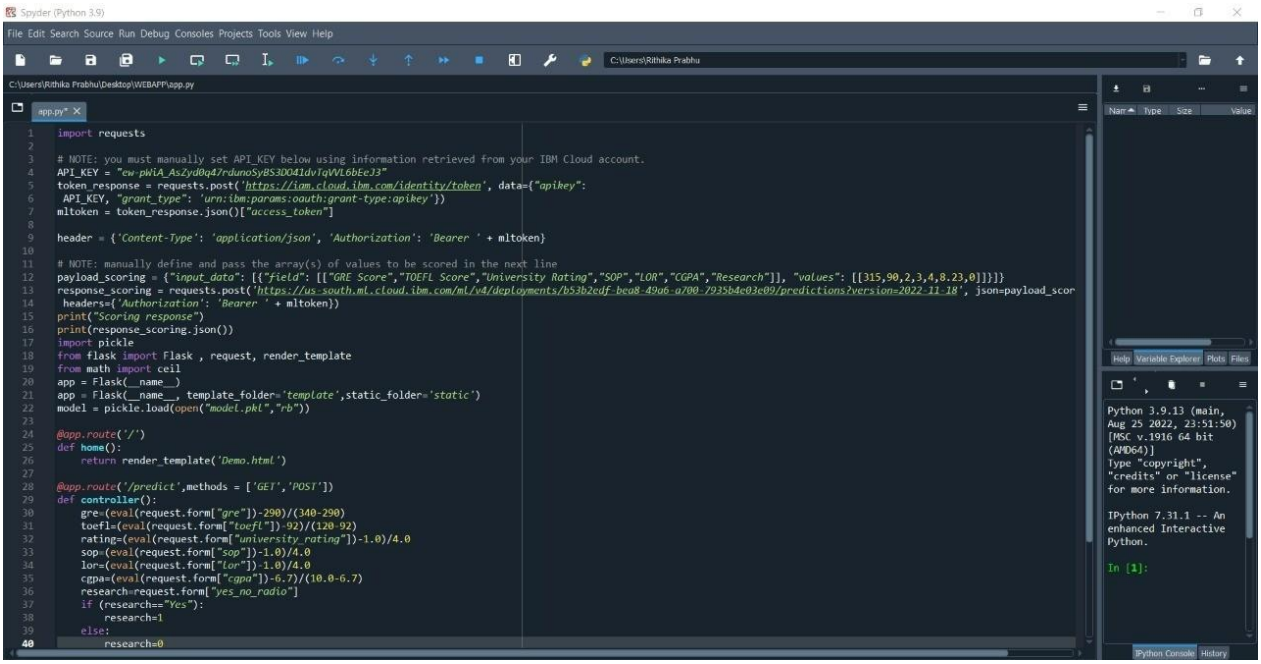
This system, being the first we have created in PHP ,has proven more difficult than originally imagined. While it may sound simple to fill out a few forms and process the information, much more is involved in the selection of applicants than this. Every time progress was made and features were added, ideas for additional features or methods to improve the usability of the system made themselves apparent. Furthermore, adding one feature meant that another required feature was now possible, and balancing completing these required features with the ideas for improvement as well as remembering everything that had to be done was a project in itself. Debugging can sometimes be a relatively straight forward process, or rather finding out what you must debug can be. Since so many parts of the admissions system are integrated into one another if an errors occurs on one page it may display error.it may be the information is not correctly read from the database; or even that the information is not correctly stored in the database initially, and all three must be checked on each occasion. This slows down the process and can be frustrating if the apparent cause of a problem is not obvious at first.

12. FUTURESCOPE

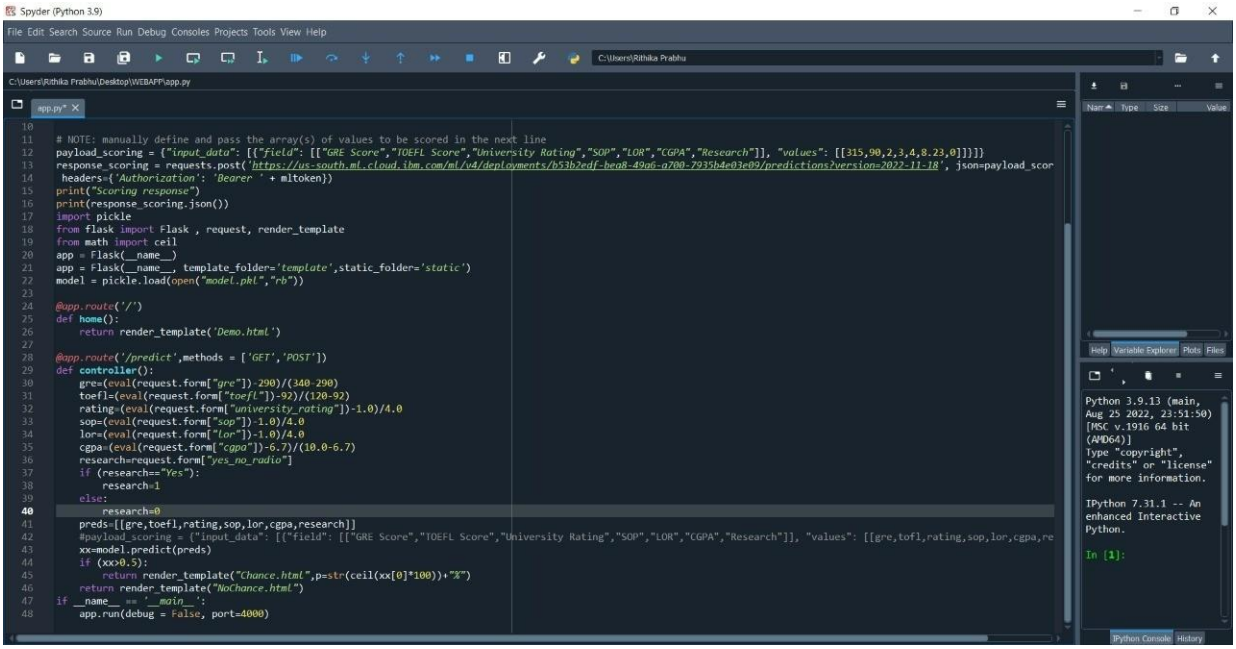
The scope of this project is a web application that allows users to enter their academic data and get predictions of their chances of admissions in the university tier of their choosing. It also provides them answers to the most common FAQ's that arise when thinking of admissions abroad for Post Graduate studies. It also provides an analysis based on the data set used that shows how the different parameters affect chances of admissions. A Database will also be implemented for the system so that students can save their data and review and edit it as they progress with the most recent predictions being saved with their profile. Issues of web security other than password protection within the website are not part of this project.

13. APPENDIX

13.1 Source Code:



```
import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "ew-pWiA_AsZyd0q47rdunoSyBS3DO41dvTqVVL6bEeJ3"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

# NOTE: manually define and pass the array(s) of values to be scored in the next line
payload_scoring = {"input_data": [{"field": [["GRE Score","TOEFL Score","University Rating","SOP","LOR","CGPA","Research"]], "values": [[315,90,2,3,4,8.23,0]]}]}
response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/b53b2edf-bea8-49a6-a700-7935b4e03e09/predictions?version=2022-11-18', json=payload_scor
 headers={'Authorization': 'Bearer ' + mltoken})
print("Scoring response")
print(response_scoring.json())
import pickle
from flask import Flask , request, render_template
from math import ceil
app = Flask(__name__)
app = Flask(__name__, template_folder='template',static_folder='static')
model = pickle.load(open("model.pkl","rb"))

@app.route('/')
def home():
    return render_template('Demo.html')

@app.route('/predict',methods = ['GET','POST'])
def controller():
    gre=(eval(request.form["gre"])-290)/(340-290)
    toefl=(eval(request.form["toefl"])-92)/(120-92)
    rating=(eval(request.form["university_rating"])-1.0)/4.0
    sop=(eval(request.form["sop"])-1.0)/4.0
    lor=(eval(request.form["lor"])-1.0)/4.0
    cgpa=(eval(request.form["cgpa"])-6.7)/(10.0-6.7)
    research=request.form["yes_no_radio"]
    if (research=="Yes"):
        research=1
    else:
        research=0
```



```
# NOTE: manually define and pass the array(s) of values to be scored in the next line
payload_scoring = {"input_data": [{"field": [["GRE Score","TOEFL Score","University Rating","SOP","LOR","CGPA","Research"]], "values": [[315,90,2,3,4,8.23,0]]}]}
response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/b53b2edf-bea8-49a6-a700-7935b4e03e09/predictions?version=2022-11-18', json=payload_scor
 headers={'Authorization': 'Bearer ' + mltoken})
print("Scoring response")
print(response_scoring.json())
import pickle
from flask import Flask , request, render_template
from math import ceil
app = Flask(__name__)
app = Flask(__name__, template_folder='template',static_folder='static')
model = pickle.load(open("model.pkl","rb"))

@app.route('/')
def home():
    return render_template('Demo.html')

@app.route('/predict',methods = ['GET','POST'])
def controller():
    gre=(eval(request.form["gre"])-290)/(340-290)
    toefl=(eval(request.form["toefl"])-92)/(120-92)
    rating=(eval(request.form["university_rating"])-1.0)/4.0
    sop=(eval(request.form["sop"])-1.0)/4.0
    lor=(eval(request.form["lor"])-1.0)/4.0
    cgpa=(eval(request.form["cgpa"])-6.7)/(10.0-6.7)
    research=request.form["yes_no_radio"]
    if (research=="Yes"):
        research=1
    else:
        research=0
    preds=[[gre,toefl,rating,sop,lor,cgpa,research]]
    #payload_scoring = {"input_data": [{"field": [["GRE Score","TOEFL Score","University Rating","SOP","LOR","CGPA","Research"]], "values": [[gre,tofl,rating,sop,lor,cgpa,re
    xx=model.predict(preds)
    if (xx>0.5):
        return render_template("Chance.html",p=str(ceil(xx[0]*100))+"%")
    return render_template("NoChance.html")
if __name__ == '__main__':
    app.run(debug = False, port=4000)
```

13.2 Github Link :

Github Link: https://github.com/IBM-EPBL/IBM-Project-31381-1660199942