UNIVERSITY ADMIN ELIGIBILITY PREDICTOR

Domain : Applied Data Science

PROJECT REPORT

GOVERNMENT COLLEGE OF ENGINEERING, BARGUR	
TEAM ID: PNT2022TMID06653	
TAMARAI SELVAN R	2011909
VIGNESH R	1911151
MUKESH KUMAR K	1911126
SARVAN KUMAR M	2011907

INDEX

INTRODUCTION

1.1. PROJECT OVERVIEW

Students are often worried about their chances of admission to University. The aim of this project is to help students in shortlisting universities with their profiles. The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea.

1.2. PURPOSE

It helps student for making decision for choosing a right college. Here the chance of occurrence of error is less when compared with the existing system. It is fast, efficient and reliable. A voids data redundancy and inconsistency. Very user-friendly. Easy accessibility of data. It helps you to understand as to how your profile can be further improved to secure an admit in your target college. It can guide you whether you need to retake the GRE or not, in order to improve your chances of landing an admit in your preferred university. Students from rural background find it difficulty to do the necessary analysis and prepare a preference list. This idea will be beneficial for them.

Students who belong to multiple categories face difficulty in analyzing cut-offs in each of these categories and predict the best colleges they can get an admission in. What so ever is the student's rank, this application will aid them in finding the best branch and college for his/her rank. This accommodates the need of students to choose the best college and helps colleges too to recognize their stand in attracting students and finer prediction implies better results for the students.

LITERATURE SURVEY

2.1. EXISTING PROBLEM

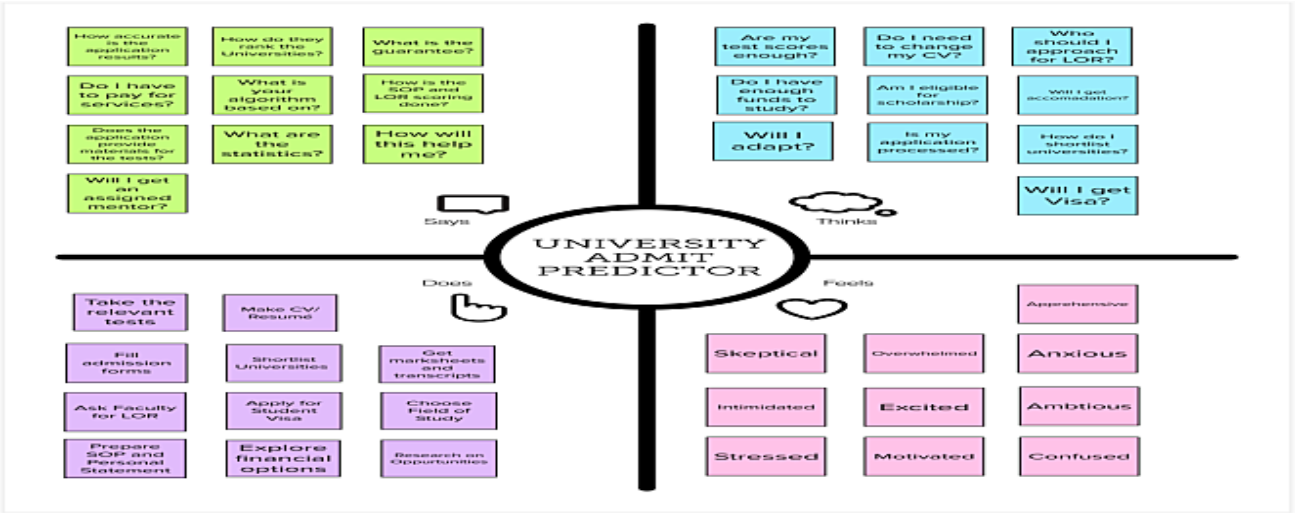
Previous research done in this area used Naive Bayes algorithm which will evaluate the success probability of student application into a respective university but the main drawback is they didn't consider all the factors which will contribute in the student admission process likeTOEFL/IELTS, SOP, LOR and under graduate score. Bayesian Networks Algorithm have been used to create a decision support network for evaluating the application submitted by foreign students of the university. This model was developed to forecast the progress of prospective students by comparing the score of students currently studying at university. The model thus predicted whether the aspiring student should be admitted to university based on various scores of students. Since the comparisons are made only with students who got admission into the universities but not with students who got their admission rejected so this method will not be that much accurate.

2.2. REFERENCES

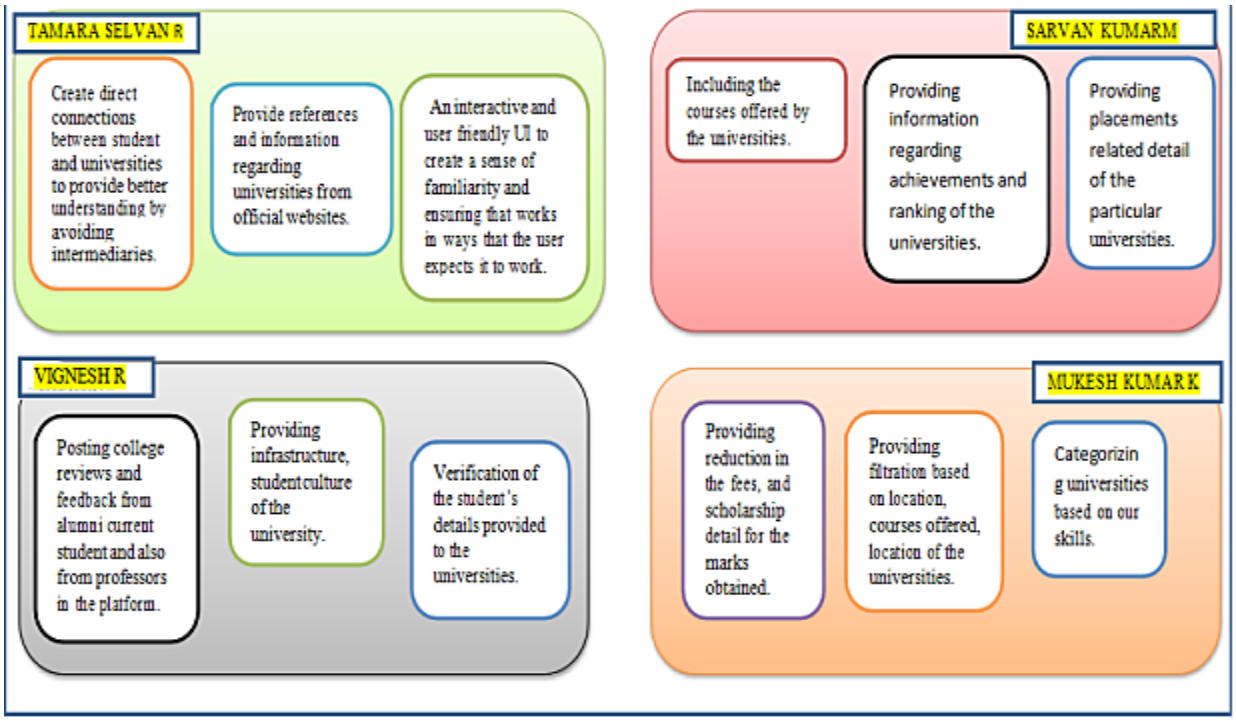
a. Predicting Undergraduate Admission, Year: 2017, Authors : Md. Protactinium, Urinal K anti Allowably, Algorithms :XGBoost,LightGBM,GBM.
b. Analysis & Prediction of American Graduate Admissions Process, Year: 2018, Author : Bhavya Ghai, Algorithms :Decision Tree,Random Forest,AdaBoost,Naive Baye's.
c. Applications of Supervised Learning Techniques on Undergraduate Admissions Data, Year: 2016, Author : Thomas Lux,Randall Pittman,Maya Shende,A nil Sheen, Algorithms :MultiLayer Perceptron,SVM Linear,SVM Poly's.
d. Name: Applications of Supervised Learning Techniques on Undergraduate Admissions Data, Year: 2016, Author : Thomas Lux,Randall Pittman,Maya Shende,A nil Sheen.
e. Graduate Admission Prediction Using Machine Learning, Year: 2020, Author : K. JeevanRatnakar, G. Koteswara Rao, B. DurgaPrasanth Kumar, G.prithvi, D.Venkata SaiEswar, Algorithms :Multiple Linear Regression,K-Nearest Neighbor, Random Forest, Multi layer Perceptron.
f. Name: Graduate Admission Prediction Using Machine Learning Techniques, Year: 2021, Author : Sara Alijasmi, Ali but Nassif, Ismail Shahin, Ashraf M Elanagar, Algorithms :Linear Regression.

IDEATION & PROPOSED SOLUTION

3.1. EMPATHY MAP CANVAS



3.2. IDEATION & BRAINSTORMIN



3.3. PROPOSED SOLUTION

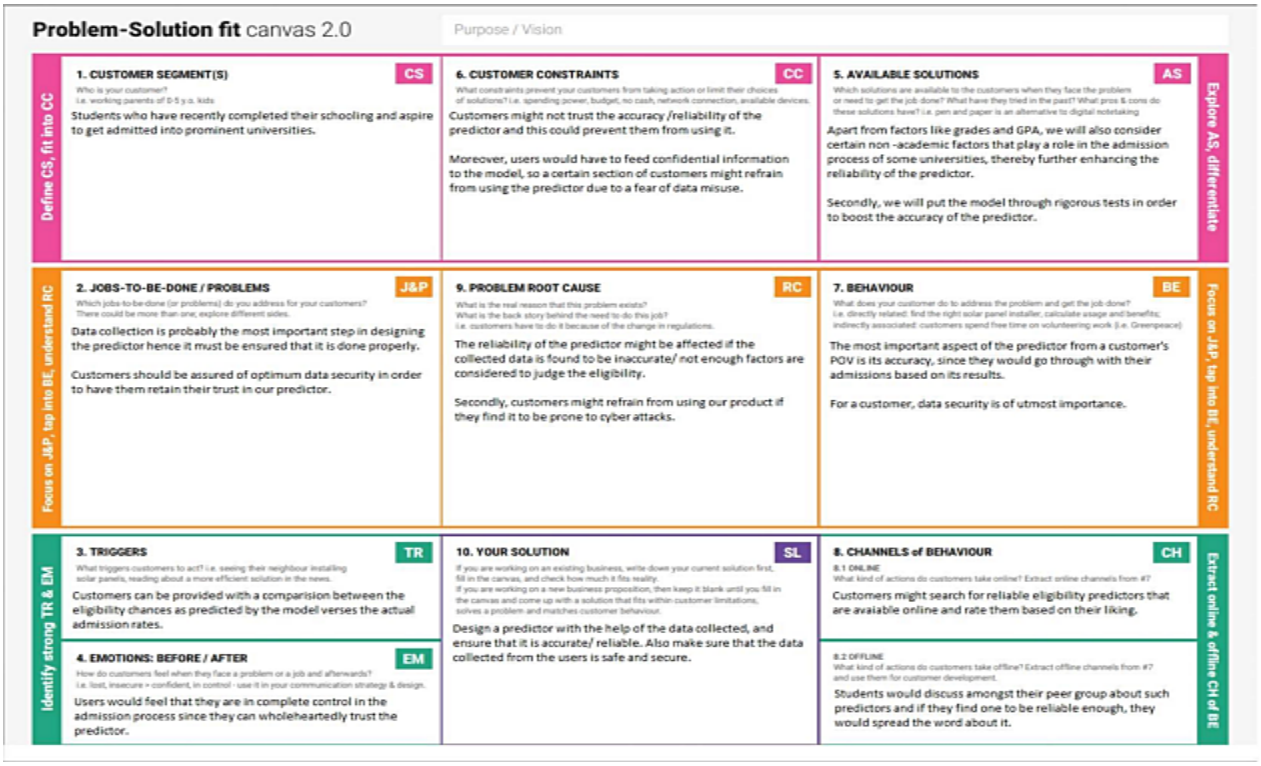
Project team shall fill the following information in proposed solution.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Students are often worried about their chances of admission to university. The aim of this project is to help students in shortlisting universities with their profiles.The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea. It also aims to make a direct connection between the students and the universities and avoid any intermediaries
2.	Idea / Solution description	This project intends to calculate the probability of acceptance in a particular grad-school after assessing the candidate's profile. The key attributes that will be considered for making the decisions are: i) GRE & TOEFL Scores ii) Undergrad CGPA iii) SOP & LOR iv) Corporate Work Experience / Research Experience v) Extra-curricular s For determining the % of acceptance, we will be using various ML models such as Logistic Regression, Multiple Linear Regression, Decision Tree & Random

		Forest and assess which model gives the highest accuracy with the help of performance metrics like accuracy score, precision and recall.
3.	Novelty / Uniqueness	The web-app will provide feedback on the parameters where the candidate is lacking so that he can improve on those areas.
4.	Social Impact / Customer Satisfaction	1. Students often feel difficult in shortlisting the universities to apply which they tend to wonder if their profile matches the requirement of a certain university. 2. Moreover, the cost of applying to a university is extremely high making it critical that students shortlist universities based on their profile. 3. A university admission prediction system is quite useful for students to determine their chances of acceptance to a specific university.

5.	Business Model (Revenue Model)	1.Advertisements of different universities could be placed in the web-app to generate revenue through ads. 2.A separate premium plan could be created where the students can directly interact with the professors and alumni of the university through video calls.
6.	Scalability of the Solution	A future update could have chat space where candidates, faculties, current students of the university and alumni can interact and candidates can get their doubts resolved instantly.

3.4 PROBLEM SOLUTION FIT



REQUIREMENT ANALYSIS

4.1. FUNCTIONAL REQUIREMENT

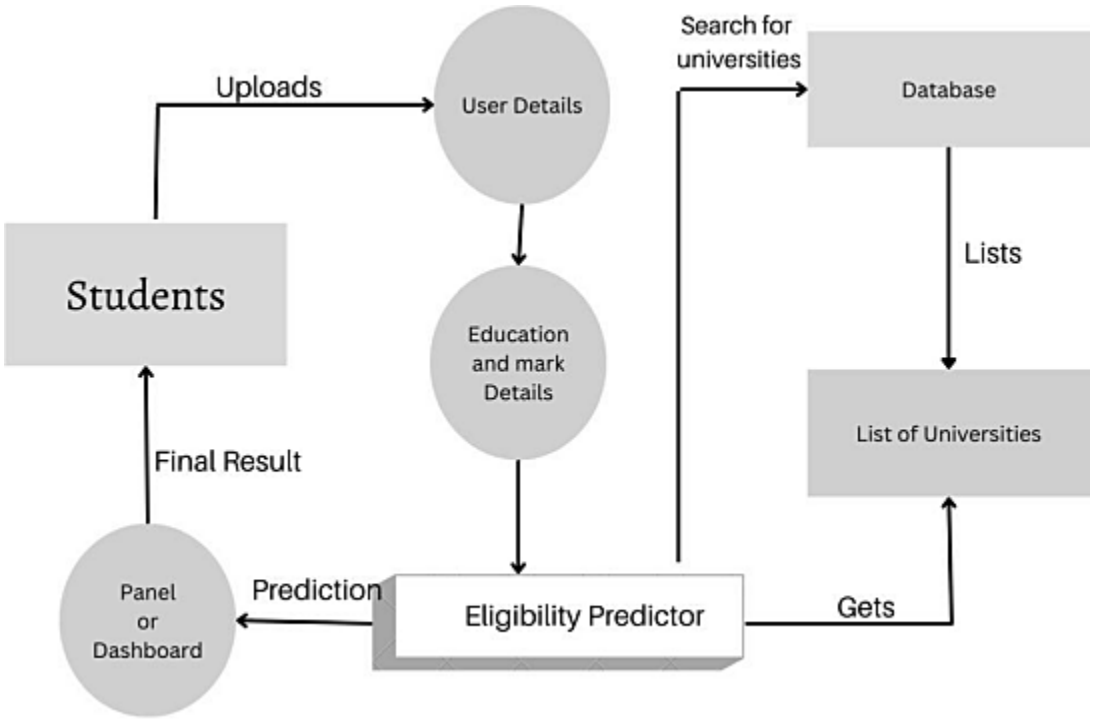
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Collecting the details of the student	Collecting TOFEL , Gre , University Rating, or ,sop and research
FR-2	Launch the website	launch the website in the browser and enter the collected values and click submit
FR-3	View the result	Then the user can able to view the eligibility
FR-4	Taking decision	By the result user can knew whether he is eligible or not

4.2. NON-FUNCTIONAL REQUIREMENT

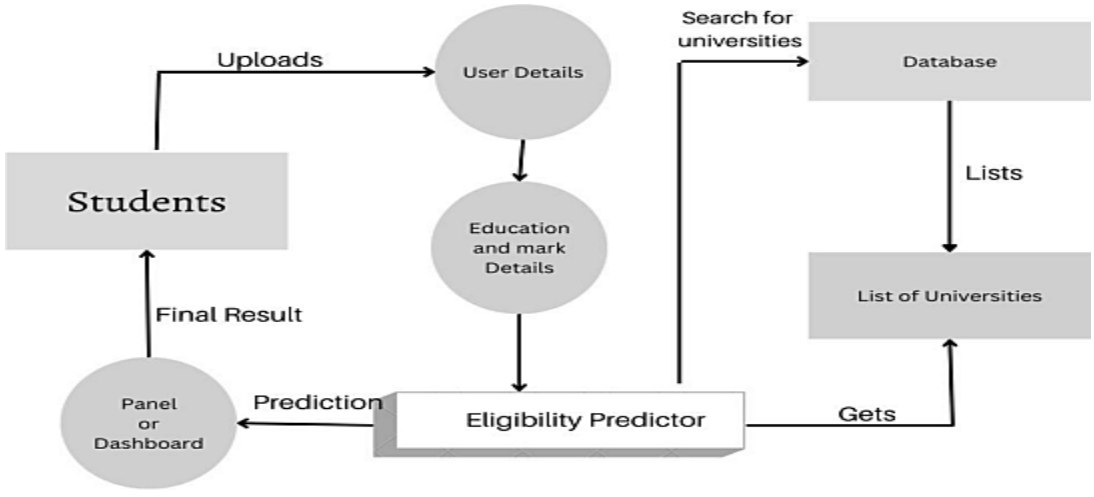
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It is hard for the people does not know about thee eligibility criteria. It should be easy to understand the website
NFR-2	**Security**	User data is completely secure. It will not get stored in the server Https provides good encryption on to the data
NFR-3	**Reliability**	The website is reliable. It has to be more accuracy
NFR-4	**Performance**	The page load me, and the ML model predicting time should be with in 5sec
NFR-5	**Availability**	The user can able to use the website 24hours
NFR-6	**Scalability**	Can be scalable up to 2,00,000 ML product request predicting at a second Can be scaled to used database to track the previous previous statistics

PROJECT DESIGN

5.1 Data Flow Diagrams



5.2 Solution & Technical Architecture



S.No	Component	Description	Technology
1.	User Interface	How user interacts with applicationе.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson ASSISTANT
4.	Database	Data Type, Configurations etc.	csv
5.	External API	Purpose of External API used in theapplication	List of eligible Universities
6.	Machine Learning Model	To predict whether a student is eligibleto get admitted in a university	Prediction Model
7.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry,

5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Student)	Dashboard	USN-1	As a user, I can view the cut off marks of previous years in my dashboard	I can access and download the files	High	Sprint-1
		USN-2	As a user, I can view university details and their rankings	I can only view(read-only)	Medium	Sprint-1

		USN-3	As a user, I can review the experience of the students in the university	I can access the review sections	Medium	Sprint-2
		USN-4	As a user, I can upload my documents	I have read and writeaccess to upload files	High	Sprint-1
		USN-5	As a user, I can fill out the general and educational details in the form	I have read and write access to the forms filled	High	Sprint-2
	Predictor	USN-6	I can view the list of universities in which I ameligible to get an admission	I can receive the final result as whether eligibleor not	High	Sprint-2
		USN-7	I can view the list of universities I am eligiblewith the same cut-off but in previous years	I can access the files with read-only permission	Medium	Sprint-2
Administrator	Dashboard	USN-8	As an administrator, I can have access to update the latest updates of the universities	I can have access to readand write the university information in the dashboard	High	Sprint-3
		USN-9	As an administrator, I can access any resources available in the page	I can access the resourcesthat are available	Medium	Sprint-3
		USN-10	As an administrator, I can have a track on theuniversities the student is eligible to get admission at	I can access the list of the universities obtained as result	High	Sprint-3

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

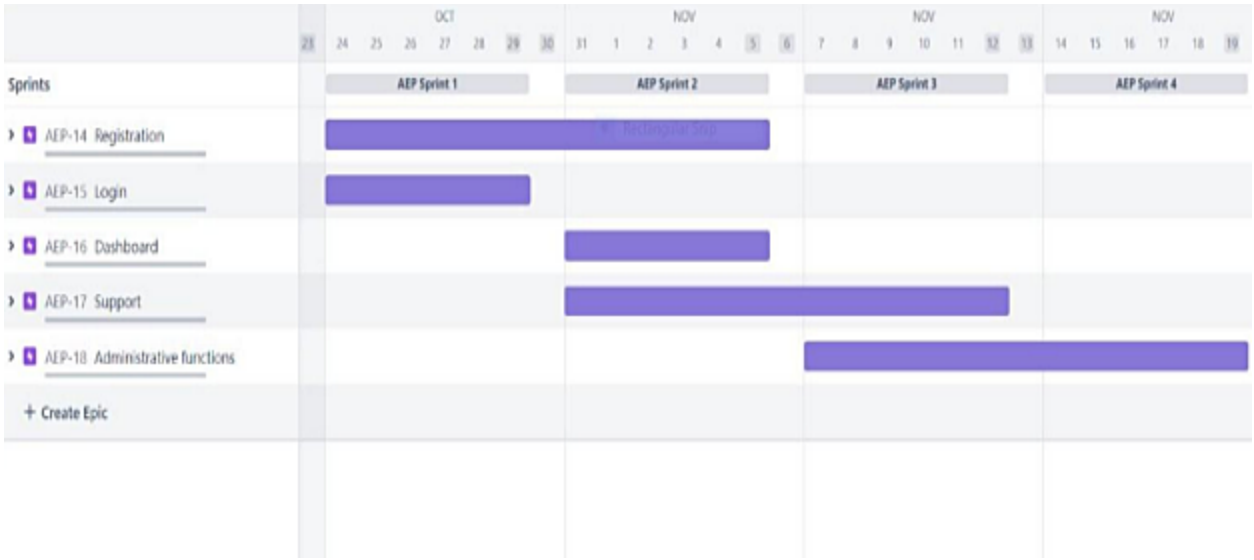
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Model Prediction	USN-1	As a user, I will be doubtful is the predicted value is right or wrong	1	Low	Tamaraiselvan R
Sprint-2		USN-2	As a user,I need a perfect value for the given data . so pre-process the data well	2	High	Mukesh Kumar K
Sprint-2		USN-3	As a user, I can register for the application through Gmail.	2	High	Vignesh R

Sprint-3	FrontEnd development	USN-4	As a user, I need the smooth functioning of the website and need a user friendly UI	1	Medium	Sarvankumar M
Sprint-4	Cloud Deployment	USN-5	As a user, I can access it from anywhere in the world	1	`Low	Tamaraiselvan R
Sprint-4	Dashboard	USN-6	Homepage for user to know about the webpage	1	Low	Sarvankumar M

6.2 SPRINT DELIVERY SCHEDULE

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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 REPORTS FROM JIRA



CODING & SOLUTIONING

7.1 FEATURE

- We have updated the website image which can change the visual appearance of website that can be very effective way to refresh the above fold content .
- It help to attract attention and to guide the user 's line of sight.
- Clear and consistent navigation is a basic requirement for the user interface.
- If it turns out to be difficult, most users will not want to waste their time to figure it out Easy to Understand
- Well designed and functional
- Site visitors are always in a hurry. Don't make them work for information. User Experience plays a key role in helping visitors use, understand and stay on your website.
- Optimized for Search and the Social Web
- Images enables the identification of features that may not be as easily detected in the spatial domain.

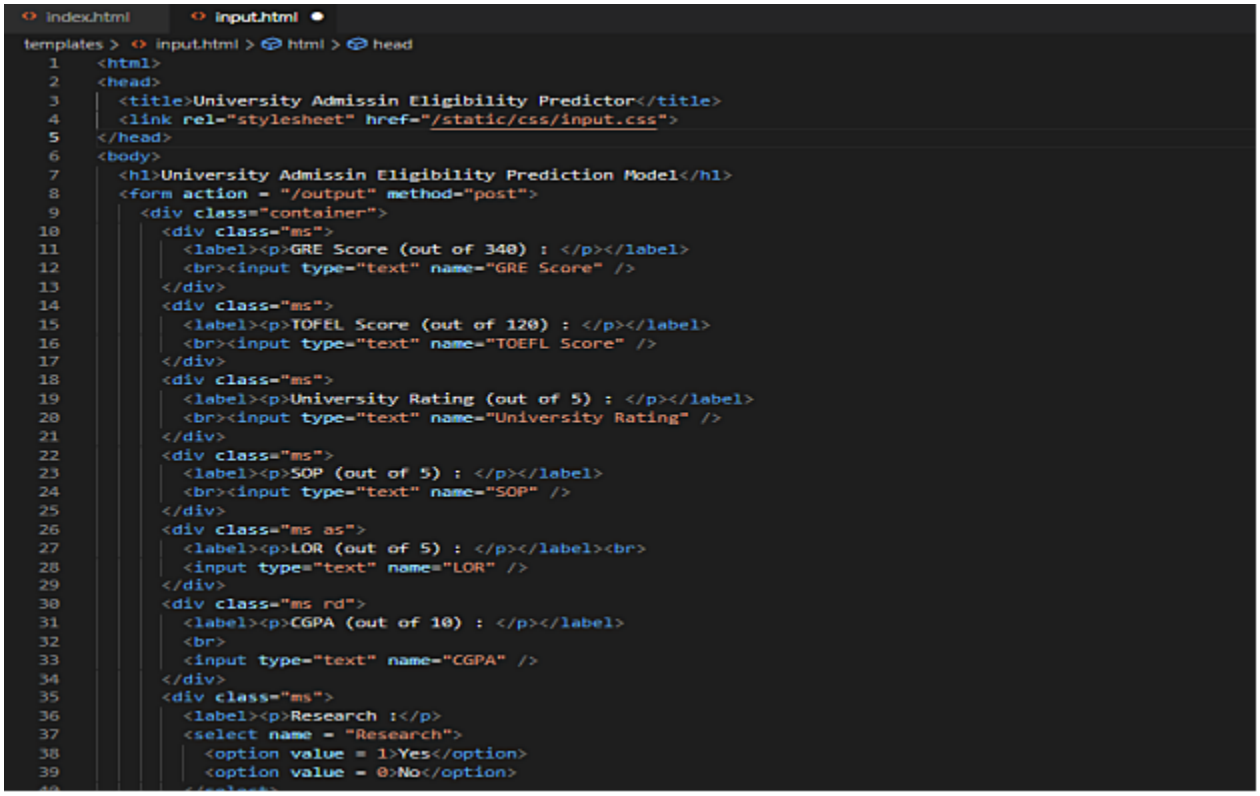
7.2 FEATURE 2

Index.html

templates > index.html > html > head > title

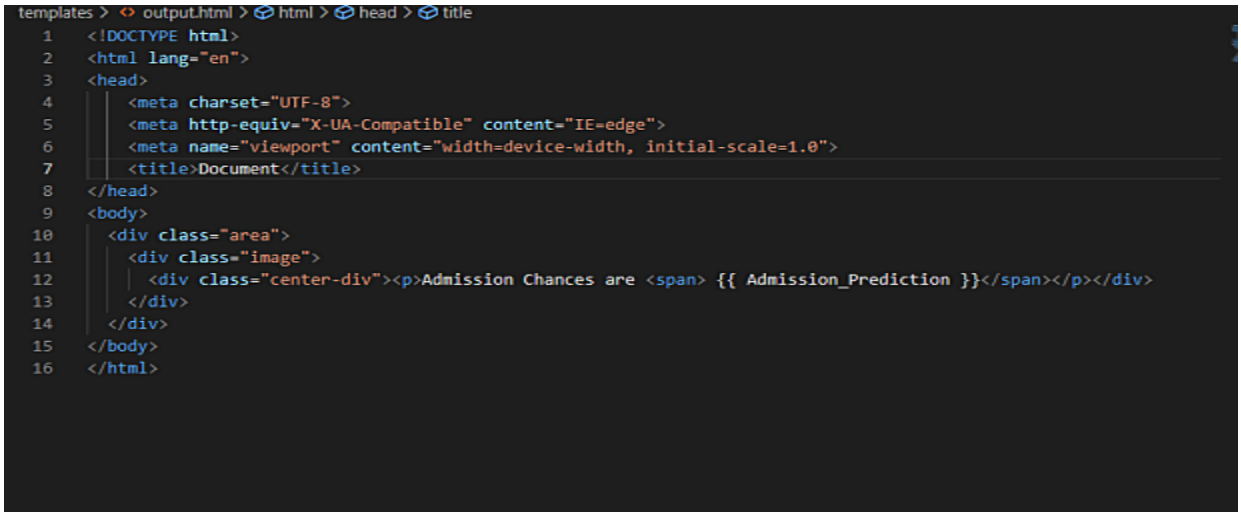
```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="/static/css/index.css">
    <title>University Admission Eligibility Predictor</title>
</head>

<body>
    <form action="/input" method="post">
        <fieldset>
            <legend>University Admission Eligibility Predictor</legend>
            <textarea cols="30" rows="10">Students are often worried about their chances of admission to Univers
        </textarea>
            <p>Click the below button to Go for model Prediction
                <br>
                <a href="input.html"><button>Model Prediction</button></a>
            </p>
        </fieldset>
    </form>
</body>

</html>
```

Input.html



```
templates > input.html > html > head
<html>
<head>
  <title>University Admissin Eligibility Predictor</title>
  <link rel="stylesheet" href="/static/css/input.css">
</head>
<body>
  <h1>University Admissin Eligibility Prediction Model</h1>
  <form action = "/output" method="post">
    <div class="container">
      <div class="ms">
        <label><p>GRE Score (out of 340) : </p></label>
        <br><input type="text" name="GRE Score" />
      </div>
      <div class="ms">
        <label><p>TOFEL Score (out of 120) : </p></label>
        <br><input type="text" name="TOEFL Score" />
      </div>
      <div class="ms">
        <label><p>University Rating (out of 5) : </p></label>
        <br><input type="text" name="University Rating" />
      </div>
      <div class="ms">
        <label><p>SOP (out of 5) : </p></label>
        <br><input type="text" name="SOP" />
      </div>
      <div class="ms as">
        <label><p>LOR (out of 5) : </p></label><br>
        <input type="text" name="LOR" />
      </div>
      <div class="ms rd">
        <label><p>CGPA (out of 10) : </p></label>
        <br>
        <input type="text" name="CGPA" />
      </div>
      <div class="ms">
        <label><p>Research :</p>
        <select name = "Research">
          <option value = 1>Yes</option>
          <option value = 0>No</option>
```

Output.html



```
templates > output.html > html > head > title
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
  <div class="area">
    <div class="image">
      <div class="center-div"><p>Admission Chances are <span> {{ Admission_Prediction }}</span></p></div>
    </div>
  </div>
</body>
</html>
```

TESTING

8.1 TEST CASES

Trained input

```
In [30]: x_train
```

Out[30]:

	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research
93	301	97	2	3.0	3.0	7.88	1
23	334	119	5	5.0	4.5	9.70	1
299	305	112	3	3.0	3.5	8.65	0
13	307	109	3	4.0	3.0	8.00	1
90	318	106	2	4.0	4.0	7.92	1
...	...	...	...	...	...	...	...
255	307	110	4	4.0	4.5	8.37	0
72	321	111	5	5.0	5.0	9.45	1
396	325	107	3	3.0	3.5	9.11	1
235	326	111	5	4.5	4.0	9.23	1
37	300	105	1	1.0	2.0	7.80	0

320 rows × 7 columns

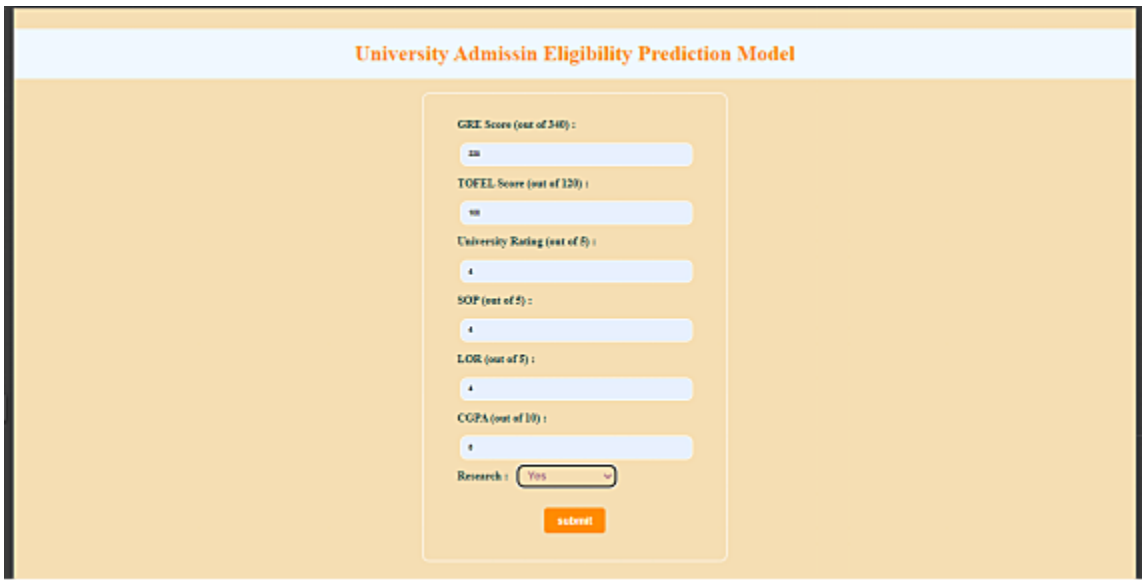
Predicted output

```
In [29]: y_pred_mlr
```

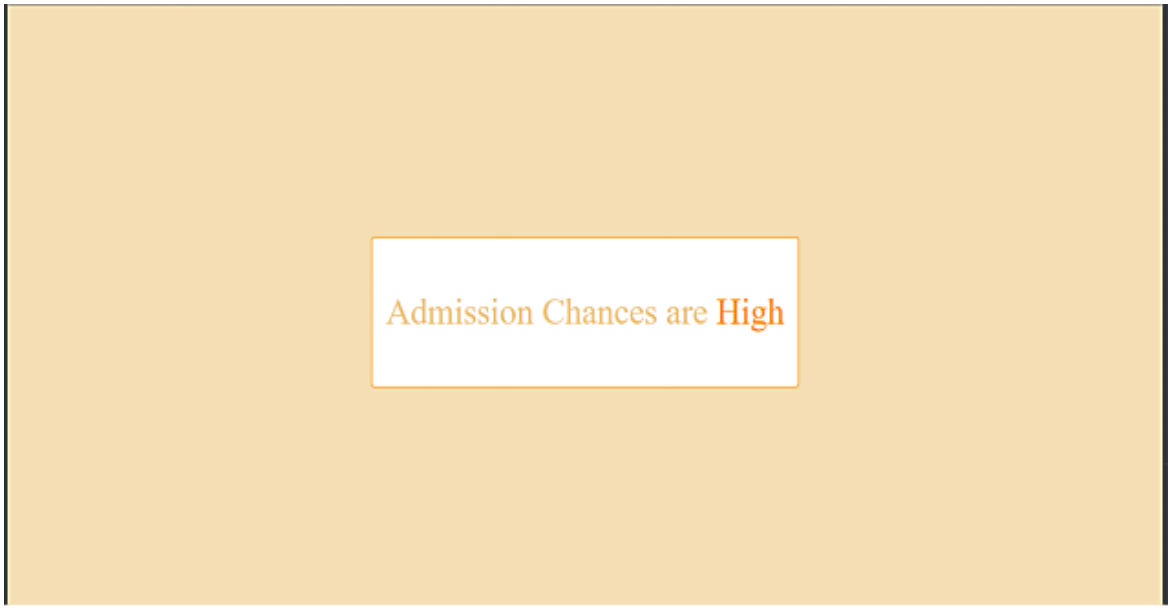
```
Out[29]: array([0.7226944 , 0.69413229, 0.78864499, 0.78216045, 0.85343383,
                0.64491036, 0.63638313, 0.55607164, 0.49601148, 0.93154092,
                0.81489263, 0.93486376, 0.88212268, 0.64871558, 0.72652977,
                0.68591465, 0.81796893, 0.86744325, 0.50758006, 0.6920451 ,
                0.667625  , 0.78064666, 0.83421581, 0.91967463, 0.65430912,
                0.56652414, 0.7241488 , 0.70847631, 0.90514058, 0.65497936,
                0.94911478, 0.63848498, 0.79257757, 0.78496168, 0.71720411,
                0.65418614, 0.43262218, 0.65210164, 0.90479229, 0.76249685,
                0.84265544, 0.68014996, 0.88404992, 0.64896391, 0.97727002,
                0.7111686 , 0.73740191, 0.82754289, 0.63802488, 0.6669232 ,
                0.78417803, 0.56425656, 0.80796338, 0.73154013, 0.70870657,
                0.89037172, 0.47935505, 0.52449211, 0.78599642, 0.78868035,
                0.92965198, 0.73217127, 0.87659205, 0.72122612, 0.53196654,
                0.77603428, 0.85923897, 0.75137956, 0.65711657, 0.79869589,
                0.81106646, 0.57971478, 0.70359047, 0.63237323, 0.83828148,
                0.44112986, 0.65906072, 0.61447964, 0.7047058 , 0.60611081])
```

8.2 USER ACCEPTANCE TESTING

INPUT



OUTPUT



RESULTS

9.1 PERFORMANCE METRICS

In [49]:
```
multiple_lin_reg = LinearRegression()
multiple_lin_reg.fit(x_train,y_train)

y_pred_mlr = multiple_lin_reg.predict(x_test)

r2_score_mlr = r2_score(y_test,y_pred_mlr)
print("Multiple Linear Regression's Score = {:.3f}".format(r2_score_mlr*100))
```

```
Multiple Linear Regression's Score = 80.790
```

ADVANTAGES

● It helps student for making decision for choosing a right college.
● Here the chance of occurrence of error is less when compared with existing system.
● It is fast, efficient and reliable.
● Avoids data redundancy and inconsistency.
● Very user-friendly.
● Easy accessibility of data.
● It would be the easiest mode to predict the university/colleges person is applicable for as well as it would unbiased and totally transparent.
● Individually would no more need to depend upon the consultancies who may be slightly deviated towards the list of colleges/university that may be having contract with them.
● Moreover applying to only that colleges/university where the student has genuine chance would even reduce application process.
● Additionally living expense of the area where colleges/university is located would also be
provided on website.

DISADVANTAGES

● Required active internet connection.
● System will provide inaccurate results if data entered incorrectly.
● Other factors such as changes in policies by the university or by the country can also affect
chances of admissions in a way that is beyond the scope of this project.
● Admissions also depend on the individual university's policy regarding the intake of foreign
students and is not modeled by our system.25

CONCLUSION

Student admission problem is very important in educational institutions. In this project addresses machine learning models to predict the chance of a student to be admitted. This will assist students to know in advance if they have a chance to get accepted. In this paper, machine learning models were performed to predict the opportunity of a student to get admitted to a master's program. The machine learning models included are multiple linear regression, k nearest neighbor, random forest, and Multi layer Perceptron. Experiments show that the Multi layer Perceptron model surpasses other models. As for the future work, more models can be conducted on more datasets to learn the model that gives the best performance

FUTURE SCOPE

The future scope of this project is very broad. Few of them are:

This can be implemented in less time for proper admission process.

1. This can be accessed anytime anywhere, since it is a web application provided only an internet connection.
2. The user had not need to travel a long distance for the admission and his/her time is also saved as a result of this automated system.
3. 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.
4. It also provides an analysis based on the data set used that shows how the different affect chances of admissions.
5. 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.
6. Future work in the project could include weighing in the features that have been ignored yet like percentage seats for Foreign Students.
7. Other criterion's like Co-curricular achievements, Leadership positions held, job experience etc can also be included as metrics for the model.

APPENDIX

SOURCE CODE

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8">
 <meta http-equiv="X-UA-Compatible" content="IE=edge">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <link rel="stylesheet" href="/static/css/index.css">
 <title>University Admission Eligibility Predictor</title>
</head>
<style>
      body{
      font-family: Arial, Helvetian, sans-serif;
      background-color: #EEE;
  }
  field set{
      margin: 100px auto;
      background-color: white;
      border: none;
      padding: 20px 0;
      width: 600px;
      filter: drop-shadow(10px 10px 80px rgba(0,0,0,.18));
  }
  field set legend{
      width: 100%;
      padding: 10px 0;
      text-align: center;
      background-color: orange;
      color: white;
```

```
    margin: 0 auto;
    transform: scale(1.1);
    border-radius: 100px;
    font-weight: bold;
    letter-spacing: 1px;
    font-size: larger;
}
field set p{
    padding:20px 10px;
    background: wheat;
    color: rgb(175, 96, 0);
    font-size: large;
    text-align: center;
}
field set p > b{
    text-transform: uppercase;
    letter-spacing: 1;
}
field set textarea{
    position: relative;
    display: block;
    width: 90%;
    height: 150px;
    border:2px solid #eee;
    padding: 10px;
    resize: none;
    margin: 10px auto;
    font-size: large;
}
button{
    background: wheat;
    color: dark orange;
    border-color: dark-orange;
    border-radius: 10px;
    min-width: 150px;
    height: 30px;
```

```
            margin-top: 10px;
        }
        button:hover {
            background: dark-orange;
            color: wheat;
            border-color: wheat;
            border-radius: 10px;
            min-width: 150px;
            height: 30px;
        }
    </style>
    <body>
        <form action="/input" method="post">
        <field set>
            <legend>University Admission Eligibility Predictor</legend>
            <textarea cols="30" rows="10">Students are often worried about their chances
of admission to University. The aim of this project is to help students in shortlisting
universities with their profiles. The predicted output gives them a fair idea about their
admission chances in a particular university. This analysis should also help students who
are currently preparing or will be preparing to get a better idea.
            </textarea>
            <!-- <textarea cols="30" rows="10">Lorem ipsum dolor sit met conjecture
dissociating
                elite. Omis aperitif alias sapper mode temporal at cum-que deficiencies
                gusto incident necessitate met provident Tempora, quit veritable
                 cum tenet dolorous aspersions. Place at?</textarea> -->
            <p>Click the below button to Go for model Prediction
              <be>
              <a href="input.html"><button>Model Prediction</button></a>
            </p>
        </fieldset>
    </form>
    </body>
    </html>
```

Input.html

```
<html>
<head>
 <title>University Admissin Eligibility Predictor</title>
 <link rel="stylesheet" href="/static/css/input.css">
</head>
<style>
 body{
  text-align:center;
  background-color:wheat;
  margin: 0;
 }
 h1{
  color:dark-orange;
  font-size:32px;
  padding:20px;
  margin-top:30px;
  background-color: alienable;
 }
 .ms p{
  color:#0F3D3E;
  font-size:17px;
  font-weight:bold;
  display:inline-block;
  max-width:250px;
 }
 .ms input{
  width:350px;
  height:35px;
  border-radius:10px;
  border:2px solid white;
  background-color:wheat;
  color:#AC7088;
  font-weight:bold;
```

```
  font-size:10px;
  padding:2px 10px;
  display:inline-block;
  margin-left:5px;
}
.ms select{
  width:150px;
  height:35px;
  border-radius:10px;
  border:2px solid white;
  background-color:wheat;
  color:#AC7088;
  font-weight:bold;
  font-size:16px;
  padding:2px 10px;
  display:inline-block;
  margin-left:10px;
}
.container{
  display: block;
  margin: 0 auto;
  background-color: wheat;
  border:2px solid white;
  text-align: left;
  padding: 20px 50px;
  width: max-content;
  border-radius: 10px;
}
.ms{
  display: block;
  width: max-content;
}
input[type="submit"]{
  background-color:dark-orange !important;
  color:white;
  border:none;
```

```
    border-radius:5px;
    padding:10px 20px;
    font-size:16px;
    font-weight:bold;
    display:block;
    margin:20px auto;
   }
</style>
<body>
  <h1>University Admission Eligibility Prediction Model</h1>
  <form action = "/output" method="post">
    <div class="container">
      <div class="ms">
        <label><p>GRE Score (out of 340) : </p></label>
        <br><input type="text" name="GRE Score" />
      </div>
      <div class="ms">
        <label><p>TOFEL Score (out of 120) : </p></label>
        <br><input type="text" name="TOEFL Score" />
      </div>
      <div class="ms">
        <label><p>University Rating (out of 5) : </p></label>
        <br><input type="text" name="University Rating" />
      </div>
      <div class="ms">
        <label><p>SOP (out of 5) : </p></label>
        <br><input type="text" name="SOP" />
      </div>
      <div class="ms as">
        <label><p>LOR (out of 5) : </p></label><br>
        <input type="text" name="LOR" />
      </div>
      <div class="ms rd">
        <label><p>CGPA (out of 10) : </p></label>
        <be>
        <input type="text" name="CGPA" />
```

```
    </div>
    <div class="ms">
      <label><p>Research :</p>
      <select name = "Research">
        <option value = 1>Yes</option>
        <option value = 0>No</option>
      </select>
    </div>
    <input type="submit" value="submit"/>
  </div>
 </form>
 <!-- <b>{{y}}</b> -->
</body>
</html>
```

Output.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
    body{
      margin: 0;
    }
 .area
 {
 margin: 0;
 height: 100vh;
 display: flex;
 flex-direction: column;
 justify-content: center;
```

```
            align-items: center;
            background-color: wheat;
            }
            .center-div{
              font-size: 50px;
              border:2px solid dark orange;
              padding: 25px 20px;
              border-radius: 5px;
              background-color: white;
            }
            p{
              color: rgb(238, 198, 123);
            }
            span{
              color: dark orange;
            }
              </style>
          </head>
          <body>
            <div class="area">
              <div class="image">
                <div class="center-div"><p>Admission Chances are <span> {{
Admission_Prediction }}</span></p></div>
              </div>
            </div>
          </body>
          </html>
```

app_ibm.py

```
import pandas as pd
from flask import Flask, request, personify, render_template
import json
import requests
# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "HSH_DnKXsrObyBjFChwLYk3pLkmNwlBnM2cPrXsCLV9b"
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"]
print("mltoken",mltoken)
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + ml token}
app = Flask(__name__)
# model = pickle.load(open('model.pkl', 'rb'))
@app.route('/')
def hello world():
   return render_template("index.html")
@app.route('/input', methods=['GET','post'])
def hello():
   return render_template("input.html")
@app.route('/output', methods=['GET','post'])
def predict():
      GRE_Score = int(request.form['GRE Score'])
   TOEFL_Score = int(request.form['TOEFL Score'])
   University_Rating = int(request.form['University Rating'])
   SOP = float(request.form['SOP'])
   LOR = float(request.form['LOR'])
   CGPA = float(request.form['CGPA'])
   Research = int(request.form['Research'])
   payload_scoring = {"input_data": [{"fields": [["GRE Score","TOEFL Score","University Rating","SOP","LOR","CGPA","Research"]], "values":
```

```
[[GRE_Score,TOEFL_Score,University_Rating,SOP,LOR,CGPA,Research]]}]}
        response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/20de2495-5ae1-4f85-ba14-
000badf504be/predictions?version=2022-11-11', json=payload_scoring,
        headers={'Authorization': 'Bearer ' + ml token})
        print("Scoring response")
        predictions=response_scoring.json()
        print(predictions)
        pre = predictions['predictions'][0]['values'][0][0]
        output=pred*100
        if(output>50):
          return render_template('output.html', Admission_Prediction='High')
        else:
          return render_template('output.html', Admission_Prediction='Low ')
        # return render_template('output.html', Admission_Prediction=output)
    if __name__ == "__main__":
        app.run(debug=True)
```

GITHUB & PROJECT DEMO LINK

GITHUB

IBM-EPBL/IBM-Project-33912-1660228832: University Admit Eligibility Predictor (github.com)

PROJECT DEMO LINK

https://drive.google.com/file/d/1O1HfuVTJfjuHv4a2dl-pT5OCnwzqH48e/view?usp=share_link