IBM-Project-50764-1660923555

UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

PROJECT DOCUMENTATION

Team ID: PNT2022TMID35187

DHIVYASHRI RAMESH 2019103015

ATHIRAI S 2019103006

KAVYA SRIDHAR 2019103027

SHRISHTI P 2019103060

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Project Overview

The University Admit Eligibility Predictor provides students with a web-based application that takes into account previous performances as well as current interests to determine an accurate list of universities and programs that would best suit the student. We do this with the help of technology such as HTML, CSS, JavaScript, IBM Cloud as well as Flask.

1.2 Purpose

The primary purpose of this application is to predict the chance of a student's admission to a particular university based on a number of factors with the help of logistic regression. The admission criteria differ from one university to another. The independent variables in this study will be measured statistically to predict graduate school admission. By analyzing previous performances as well as current interests the application allows predictive models to better prioritize the applicants, which by extension provides the admission to the right candidates.

2. LITERATURE SURVEY

2.1 Existing Problem

The number of students seeking admission for graduate studies is increasing year by year. As a result, there is a lot of competition and the chances of a student being admitted to a particular university is unpredictable. Students may also be unaware of the existence of certain programs that cater to their specific interests. An individual's choice of university as well as program is extremely consequential to their life trajectory. How a student chooses a university, and conversely how a university chooses a student, determines the student's future.

2.2 References

https://www.researchgate.net/publication/345391208_University_Admissions_Predictor
https://nevonprojects.com/college-admission-predictor-php/
https://www.eajournals.org/wp-content/uploads/Predicting-Student-University-Admission-Using-Logistic-Regression.pdf
https://medium.com/@jigar18011999/university-predictor-by-machine-learning-2d880e9f3a3

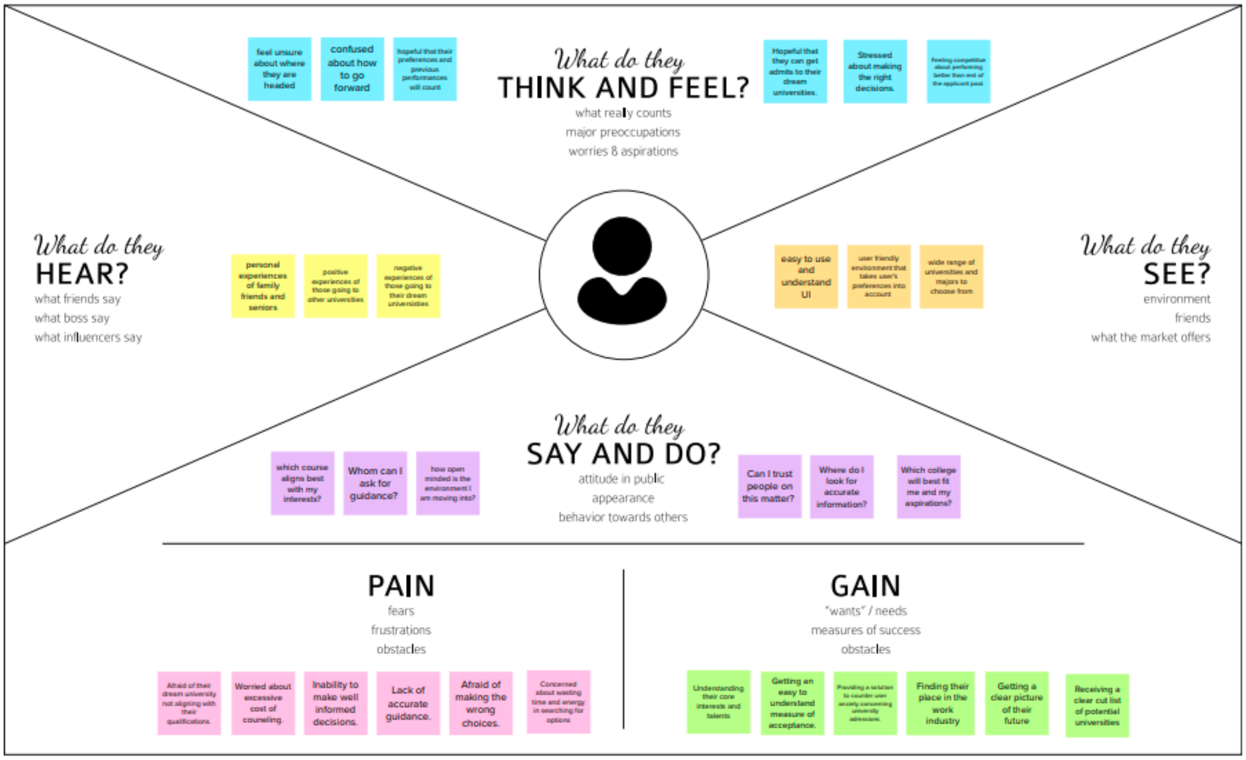
https://www.researchgate.net/publication/348433004_Graduate_Admission_Prediction_Using_Machine_Learning
https://www.eajournals.org/wp-content/uploads/Predicting-Student-University-Admission-Using-Logistic-Regression.pdf

2.3 Problem Statement Definition

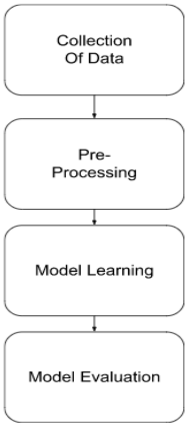
It's a well-known fact that education makes a man. For many students, choosing the right university and program is a challenge. Although many online resources and forums are available, their findings are not concrete, as most of them are based on mere assumptions of college rankings and not actual statistics. For a student, college education is more of an investment, the returns of which they receive in the form of employment. Hence, to guide the students in an efficient manner, the University Admit Eligibility Predictor is used, and it is based on the student's previous performances as well as current interests.

3. IDEATION & PROPOSED SOLUTION

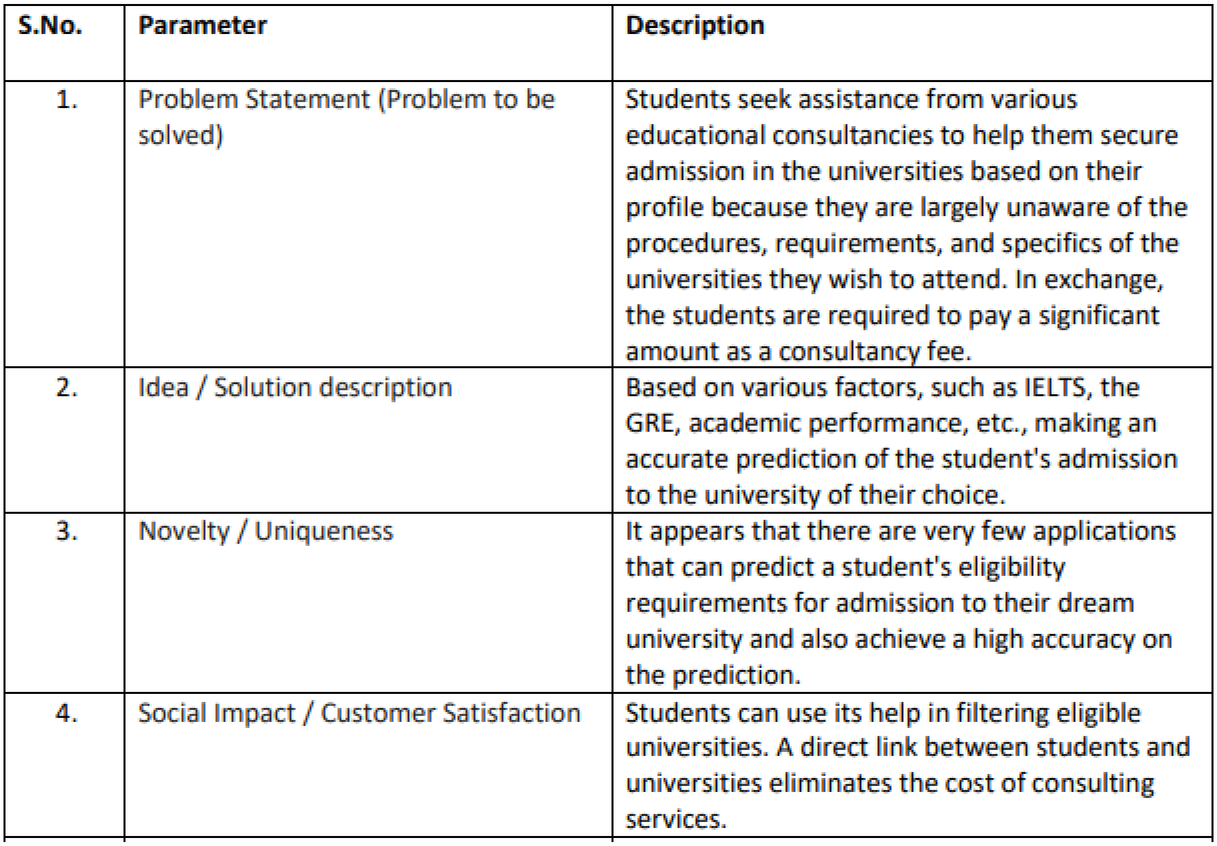
3.1 Empathy Map Canvas


feel unsure about where they are headed
confused about how to go forward
hopeful that their preferences and previous performances will count
What do they
THINK AND FEEL?
what really counts
major preoccupations
worries 8 aspirations
Hopeful that they can get admits to their dream universities.
Stressed about making the right decisions.
Feeling competitive about performing better than rest of the applicant pool.
What do they
HEAR?
what friends say
what boss say
what influencers say
personal experiences of family friends and seniors
positive experiences of those going to other universities
negative experiences of those going to their dream universities
easy to use and understand UI
user friendly environment that takes user's preferences into account
wide range of universities and majors to choose from
What do they
SEE?
environment
friends
what the market offers
which course aligns best with my interests?
Whom can I ask for guidance?
how open minded is the environment I am moving into?
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
Can I trust people on this matter?
Where do I look for accurate information?
Which college will best fit me and my aspirations?
PAIN
fears
frustrations
obstacles
Afraid of their dream university not aligning with their qualifications.
Worried about excessive cost of counseling.
Inability to make well informed decisions.
Lack of accurate guidance.
Afraid of making the wrong choices.
Concerned about wasting time and energy in searching for options.
GAIN
"wants" / needs
measures of success
obstacles
Understanding their core interests and talents
Getting an easy to understand measure of acceptance.
Providing a solution to counter user anxiety concerning university admissions.
Finding their place in the work industry
Getting a clear picture of their future
Receiving a clear cut list of potential universities

3.2 Ideation & Brainstorming



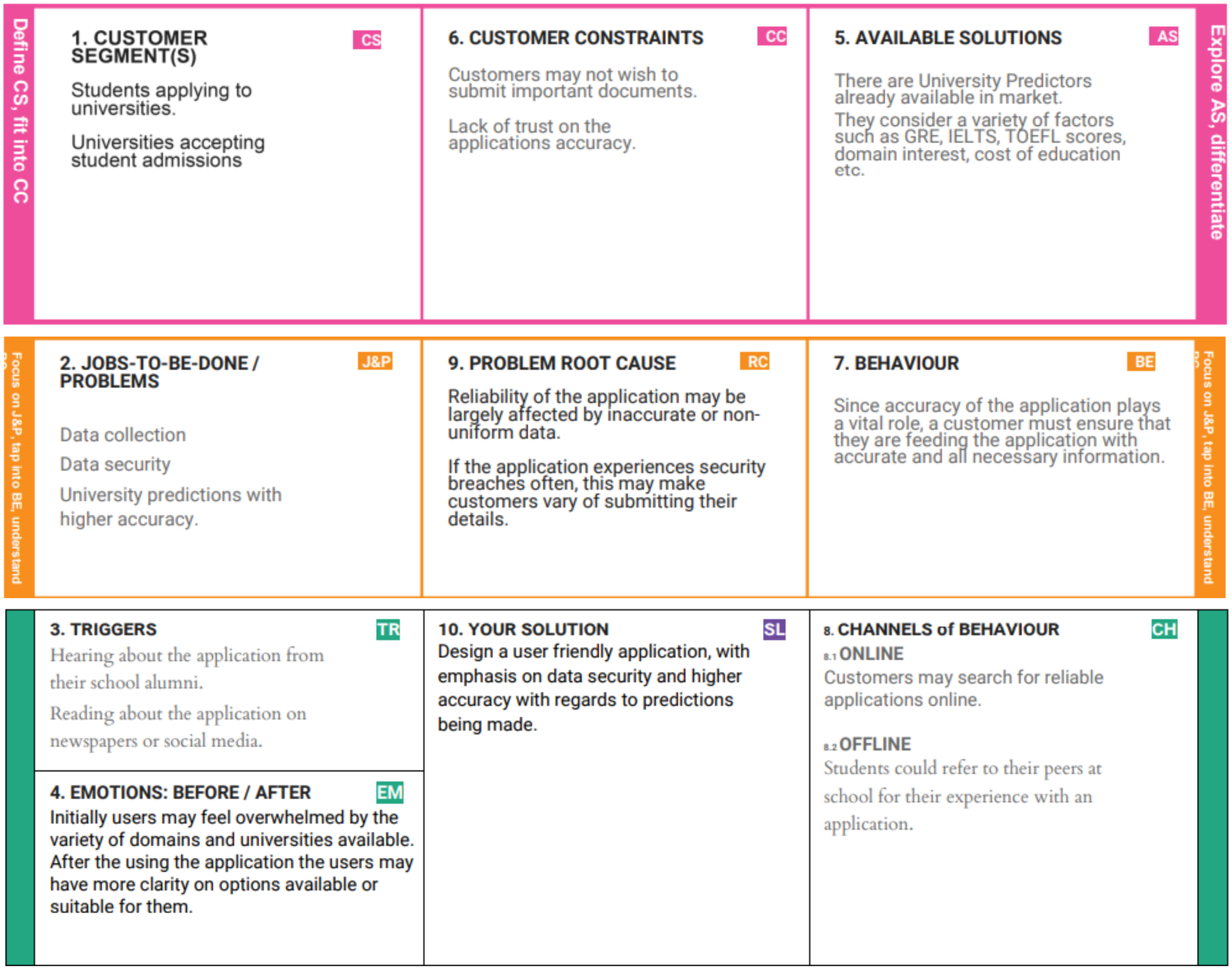
3.3 Proposed Solution



S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Students seek assistance from various educational consultancies to help them secure admission in the universities based on their profile because they are largely unaware of the procedures, requirements, and specifics of the universities they wish to attend. In exchange, the students are required to pay a significant amount as a consultancy fee.
2.	Idea / Solution description	Based on various factors, such as IELTS, the GRE, academic performance, etc., making an accurate prediction of the student's admission to the university of their choice.
3.	Novelty / Uniqueness	It appears that there are very few applications that can predict a student's eligibility requirements for admission to their dream university and also achieve a high accuracy on the prediction.
4.	Social Impact / Customer Satisfaction	Students can use its help in filtering eligible universities. A direct link between students and universities eliminates the cost of consulting services.

5.	Business Model (Revenue Model)	The enrolment process is often complex and varies based on each university and its location. Predictive models help ease the process and increase efficiency.
6.	Scalability of the Solution	Universities can be encouraged to maintain database of incoming students' eligibility. This would in turn help the applications in making better predictions.

3.4 Problem Solution Fit



Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand
Focus on J&P, tap into BE, understand

1. CUSTOMER SEGMENT(S) CS

Students applying to universities.

Universities accepting student admissions

6. CUSTOMER CONSTRAINTS CC

Customers may not wish to submit important documents.

Lack of trust on the applications accuracy.

5. AVAILABLE SOLUTIONS AS

There are University Predictors already available in market.

They consider a variety of factors such as GRE, IELTS, TOEFL scores, domain interest, cost of education etc.

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Data collection

Data security

University predictions with higher accuracy.

9. PROBLEM ROOT CAUSE RC

Reliability of the application may be largely affected by inaccurate or non-uniform data.

If the application experiences security breaches often, this may make customers vary of submitting their details.

7. BEHAVIOUR BE

Since accuracy of the application plays a vital role, a customer must ensure that they are feeding the application with accurate and all necessary information.

3. TRIGGERS TR

Hearing about the application from their school alumni.

Reading about the application on newspapers or social media.

4. EMOTIONS: BEFORE / AFTER EM

Initially users may feel overwhelmed by the variety of domains and universities available. After the using the application the users may have more clarity on options available or suitable for them.

10. YOUR SOLUTION SL

Design a user friendly application, with emphasis on data security and higher accuracy with regards to predictions being made.

8. CHANNELS of BEHAVIOUR CH

8.1 ONLINE

Customers may search for reliable applications online.

8.2 OFFLINE

Students could refer to their peers at school for their experience with an application.

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Requirements	All the needed files are been asked to feed in the website. By having the file, it will do all the pre-processing and shows all the required information to the student(user). The information includes the list of all the possible universities and streams.
FR-4	User Details	Has to feed some documents 1. Score Sheets 2. Letter of Recommendation (LOR) 3. Statement of Purpose (SOP) 4. Curriculum Vitae (CV)

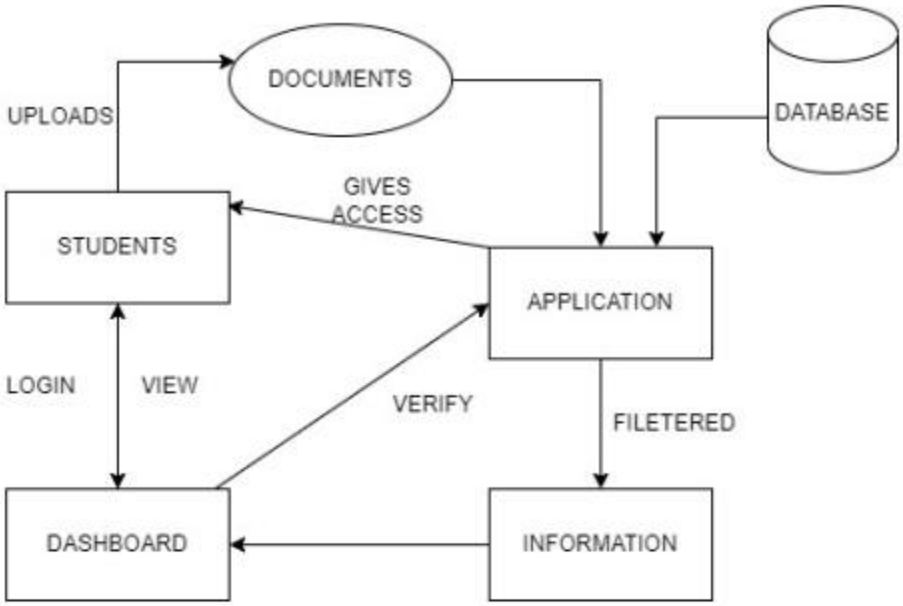
4.2 Non-Functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	• Our website is very user friendly so even the layman can able to access our website. • There is no need for any pre requisite technical skill in order to access our website. • Each and every content of the page will be in synchronous way. Thus, it will not take much time to refresh or reload.
NFR-2	**Security**	• The user who is having the valid credentials can able to access our site. • Data they are feeding into our website will not be accessed by any one of them.
NFR-3	**Reliability**	• Our website is more reliable. Since nobody can able to see the data fed by the user. • The user can get the result with higher percent of accuracy.
NFR-4	**Performance**	• User can able to handle the process in our website even by having internet connection

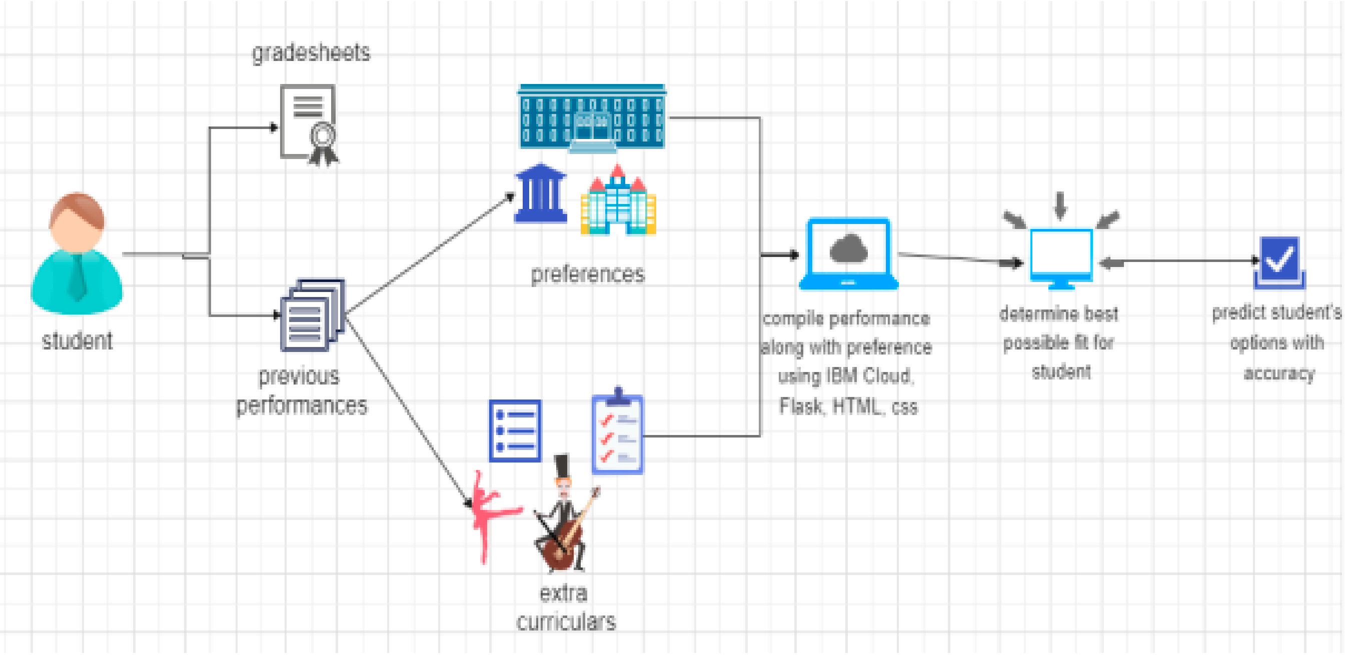
		with normal speed. There is no need of high-speed internet connection. • Traffics can be handled effiectively.
NFR-5	**Availability**	• Students can avail our website from any of the browser in faster and efficient way.
NFR-6	**Scalability**	• Our website will be easily scalable in the case of getting increasing number of users data from our website. • If needed we do scale up the CPU or Processer in order to speed up the processing capability of our website. • There by it reduces the downtime of our website.

5. PROJECT DESIGN

5.1 Data Flow Diagram



5.2 Solution & Technical Architecture

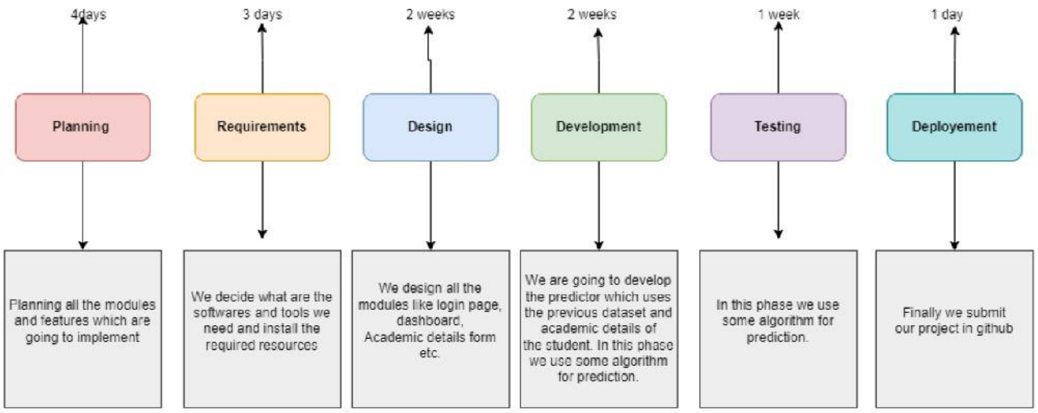

gradesheets
student
preferences
previous performances
compile performance along with preference using IBM Cloud, Flask, HTML, css
determine best possible fit for student
predict student's options with accuracy
extra curriculars

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 university details and their rankings.	Read Only	Medium	Sprint-1
		USN-2	As a user, I edit the information I have entered.	I can read and write files.	High	Sprint-1
		USN-3	As a user I can upload documents.	I can read and write files.	High	Sprint-1
		USN-4	As a user, I can register for the application through Gmail	Write only.	Low	Sprint-2
		USN-5	As a user, I can log into the application by entering user id & password	Write only.	Medium	Sprint-1
	Predictor	USN-6	As a user I can view all available universities.	Read Only	Medium	Sprint-1
		USN-7	As a user I can filter universities based on criteria.	Read Only	Medium	Sprint-1
Administrator	Dashboard	USN-8	As an administrator I have access to all files.	Read Only	Medium	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation



Planning all the modules and features which are going to implement

We decide what are the softwares and tools we need and install the required resources

We design all the modules like login page, dashboard, Academic details form etc.

We are going to develop the predictor which uses the previous dataset and academic details of the student. In this phase we use some algorithm for prediction.

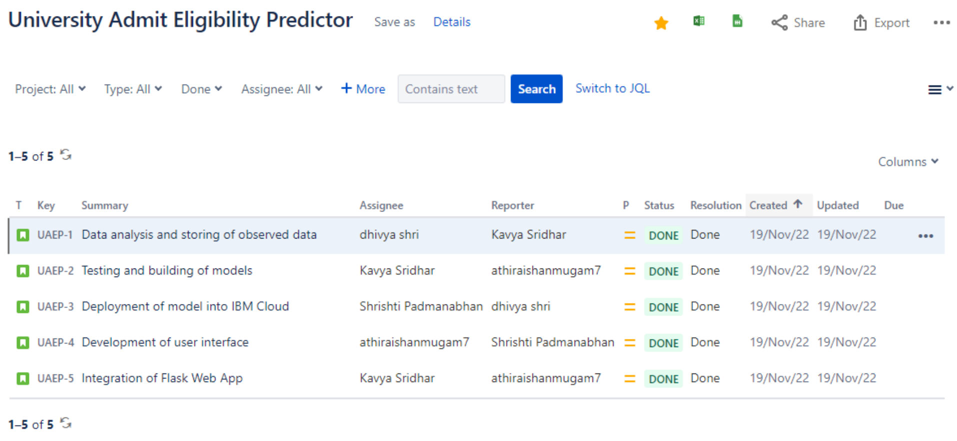
In this phase we use some algorithm for prediction.

Finally we submit our project in github

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	5	6 Days	24 Oct 2022	29 Oct 2022	5	1/Nov/2022
Sprint-2	6	6 Days	31 Oct 2022	05 Nov 2022	6	8/Nov/2022
Sprint-3	6	6 Days	07 Nov 2022	12 Nov 2022	6	16/Nov/2022
Sprint-4	5	6 Days	14 Nov 2022	19 Nov 2022	5	23/Nov/2022

6.3 Reports from JIRA



University Admit Eligibility Predictor

Project: All   Type: All   Done   Assignee: All

1–5 of 5

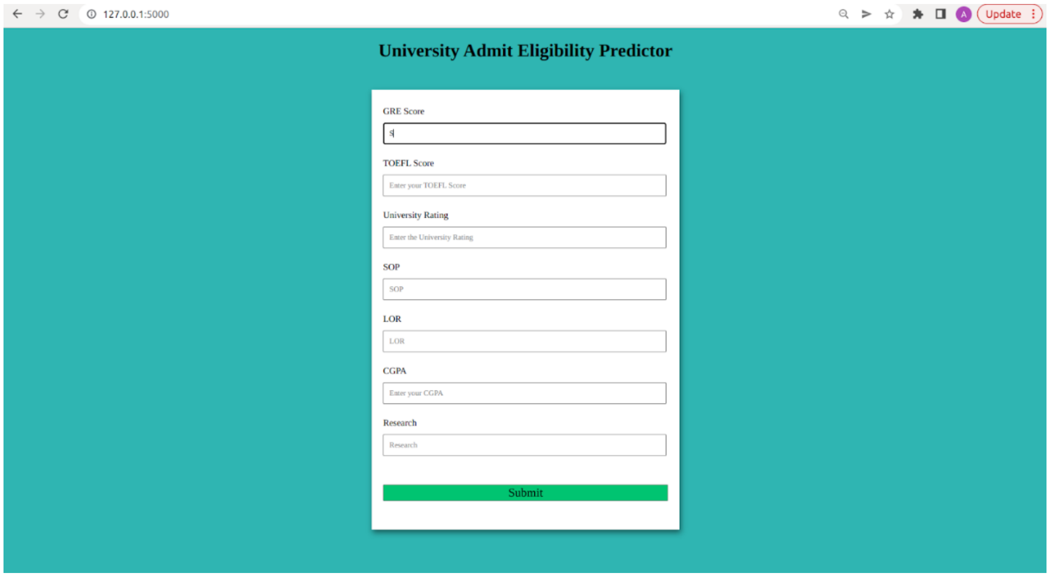
T	Key	Summary	Assignee	Reporter	P	Status	Resolution	Created	Updated	Due
	UAEP-1	Data analysis and storing of observed data	dhivya shri	Kavya Sridhar	=	DONE	Done	19/Nov/22	19/Nov/22	
	UAEP-2	Testing and building of models	Kavya Sridhar	athiraishanmugam7	=	DONE	Done	19/Nov/22	19/Nov/22	
	UAEP-3	Deployment of model into IBM Cloud	Shrishti Padmanabhan	dhivya shri	=	DONE	Done	19/Nov/22	19/Nov/22	
	UAEP-4	Development of user interface	athiraishanmugam7	Shrishti Padmanabhan	=	DONE	Done	19/Nov/22	19/Nov/22	
	UAEP-5	Integration of Flask Web App	Kavya Sridhar	athiraishanmugam7	=	DONE	Done	19/Nov/22	19/Nov/22	

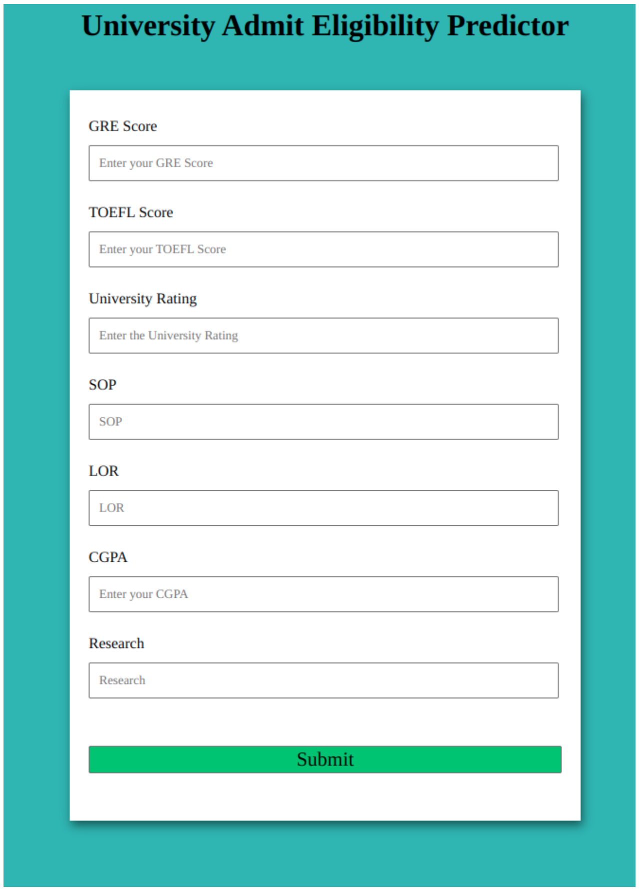
1–5 of 5

7. CODING & SOLUTIONING

7.1 Features

The scores are collected in the forms and the values are passed as parameters in the API**.**

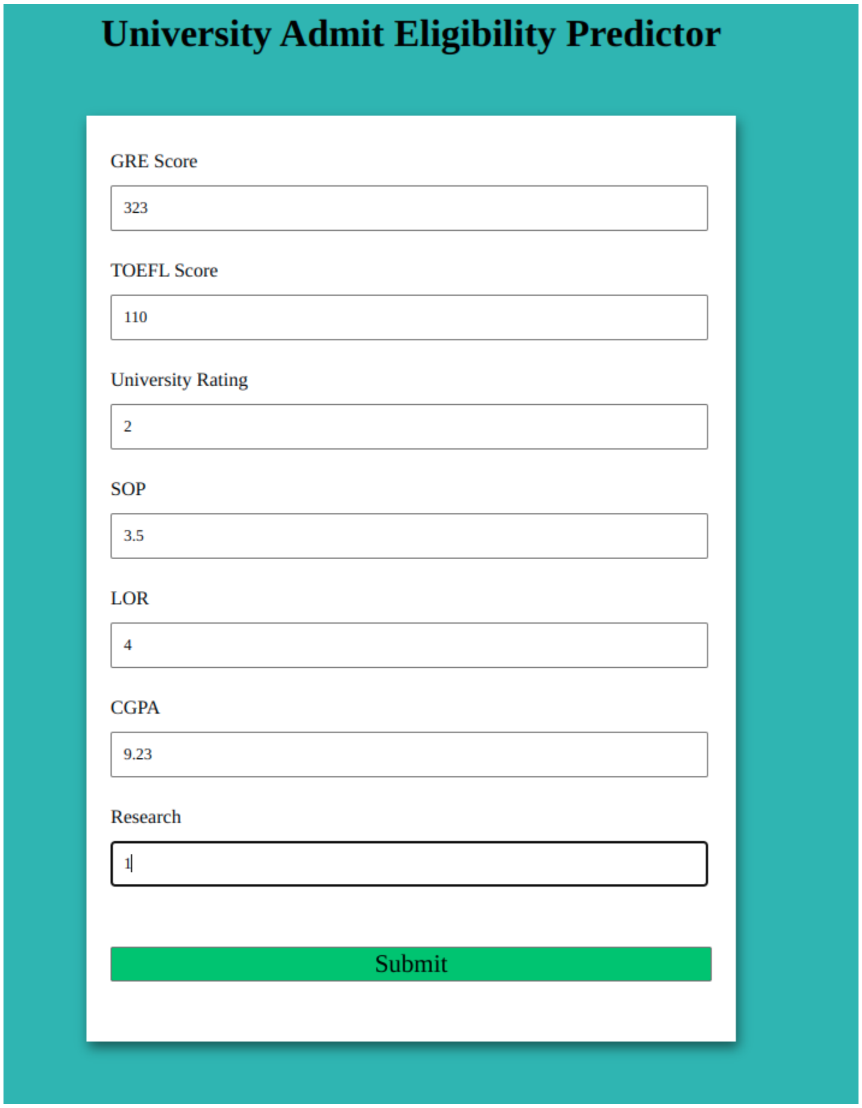




8. TESTING

8.1 Test Cases

TEST CASE 1



University Admit Eligibility Predictor

GRE Score

323

TOEFL Score

110

University Rating

2

SOP

3.5

LOR

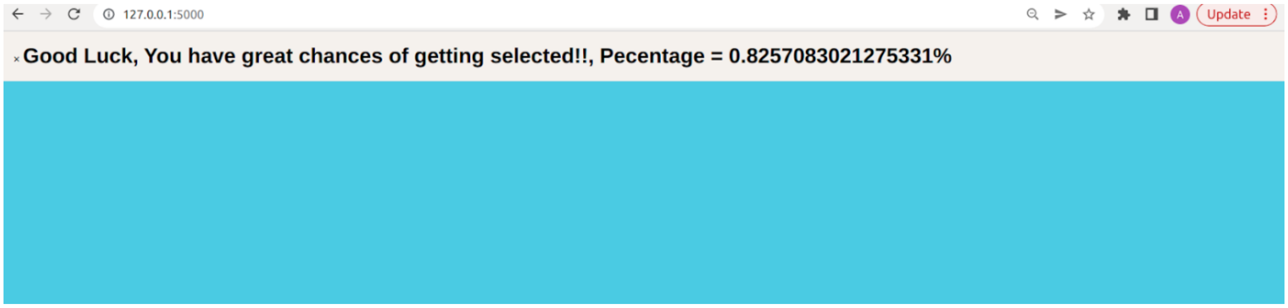
4

CGPA

9.23

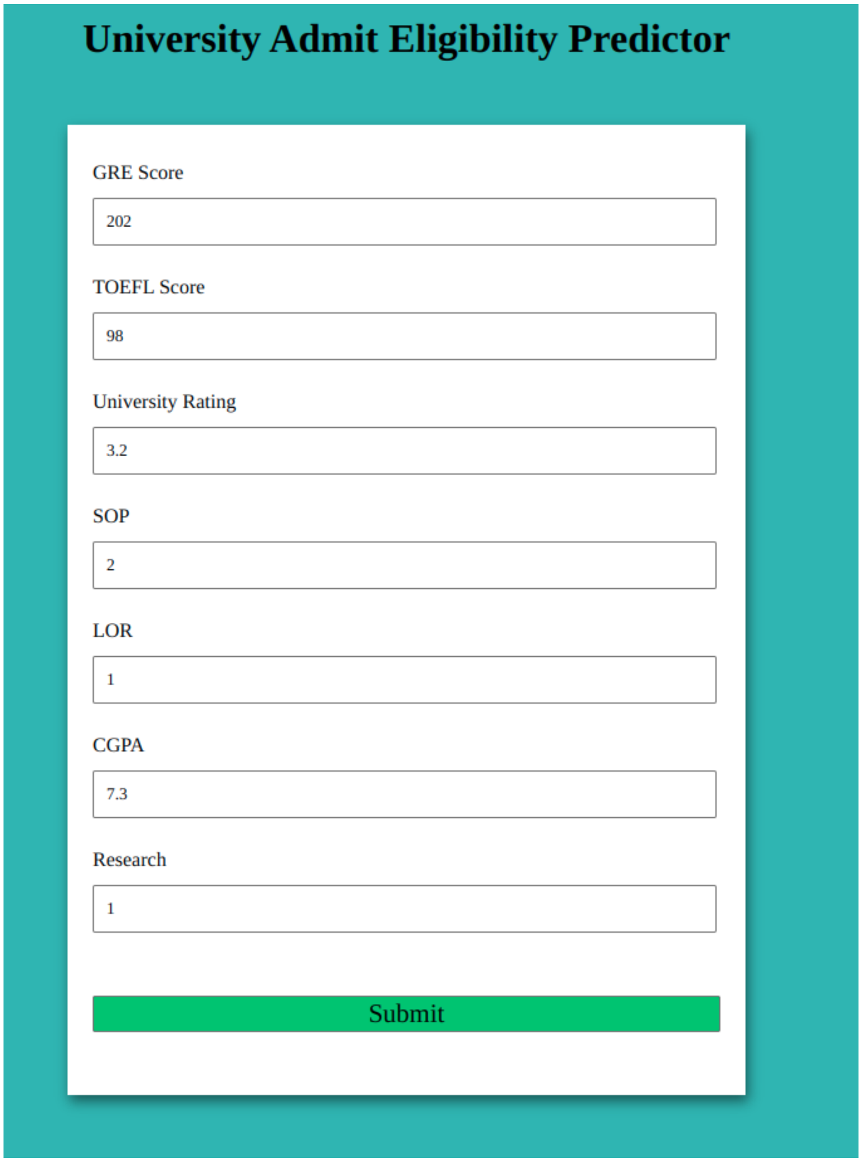
Research

1



Good Luck, You have great chances of getting selected!!, Pecentage = 0.8257083021275331%

TEST CASE 2



University Admit Eligibility Predictor

GRE Score

202

TOEFL Score

98

University Rating

3.2

SOP

2

LOR

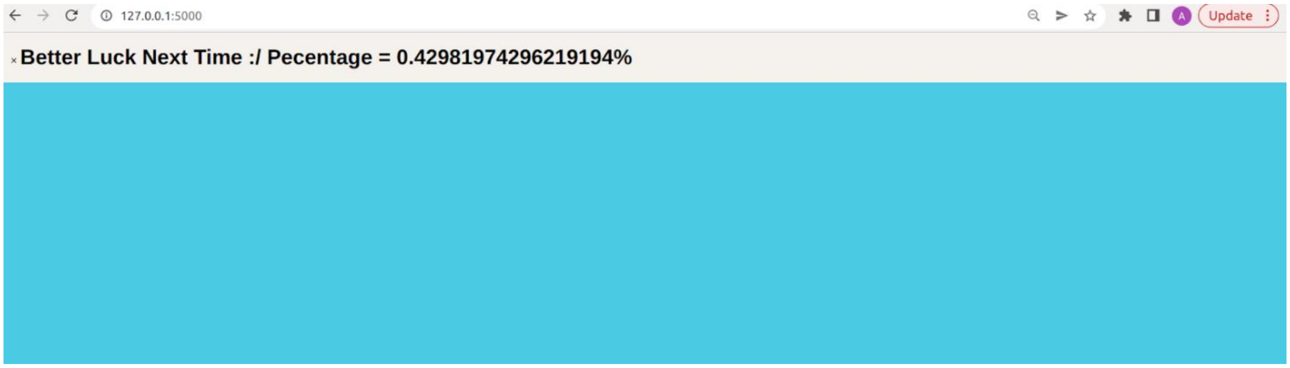
1

CGPA

7.3

Research

1



Better Luck Next Time :/ Pecentage = 0.42981974296219194%

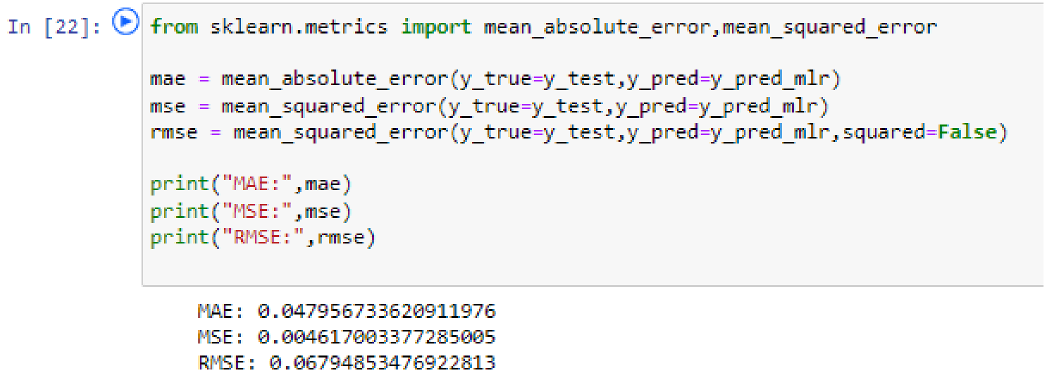
9. RESULTS

9.1 Performance Metrics

MAE
The Mean absolute error represents the average of the absolute difference between the actual and predicted values in the dataset. It measures the average of the residuals in the dataset

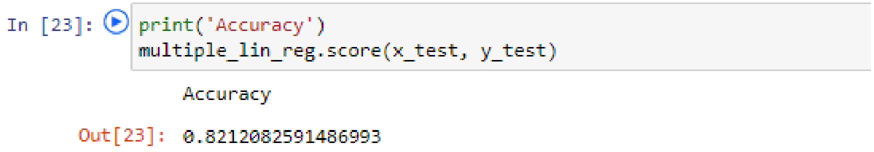
MSE
Mean Squared Error represents the average of the squared difference between the original and predicted values in the data set. It measures the variance of the residuals.

RMSE
Root Mean Squared Error is the square root of Mean Squared error. It measures the standard deviation of residuals.



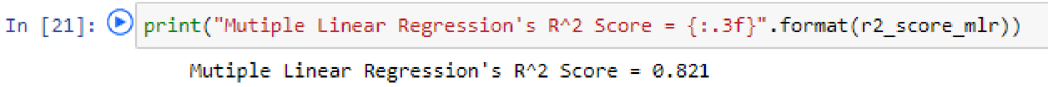
```
In [22]: from sklearn.metrics import mean_absolute_error,mean_squared_error

         mae = mean_absolute_error(y_true=y_test,y_pred=y_pred_mlr)
         mse = mean_squared_error(y_true=y_test,y_pred=y_pred_mlr)
         rmse = mean_squared_error(y_true=y_test,y_pred=y_pred_mlr,squared=False)

         print("MAE:",mae)
         print("MSE:",mse)
         print("RMSE:",rmse)

             MAE: 0.047956733620911976
             MSE: 0.004617003377285005
             RMSE: 0.06794853476922813
```

ACCURACY
The accuracy of a machine learning classification algorithm is one way to measure how often the algorithm classifies a data point correctly. Accuracy is the number of correctly predicted data points out of all the data points.



```
In [23]: print('Accuracy')
         multiple_lin_reg.score(x_test, y_test)

         Accuracy

Out[23]: 0.8212082591486993
```

R2 SCORE
The coefficient of determination or R-squared represents the proportion of the variance in the dependent variable which is explained by the linear regression model. It is a scale-free score i.e., irrespective of the values being small or large, the value of R square will be less than one.



```
In [21]: print("Mutiple Linear Regression's R^2 Score = {:.3f}".format(r2_score_mlr))

         Mutiple Linear Regression's R^2 Score = 0.821
```

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

There is an endless number of advantages of ML. We can take a look at the one. which are really helpful. The advantages of Machine Learning tell us how using ML would benefit us. So, let's have a look at the advantages of Machine Learning:

- Automation of Everything
- Wide Range of Applications
- Scope of Improvement
- Efficient Handling of Data
- Best for Education

DISADVANTAGES

There are significant disadvantages for the application and usage of machine learning in general. The main disadvantages to be kept in mind are:

- The dataset could become outdated.
- The model accuracy may not be sufficient.
- There might be many exceptions to all the observations.
- There are many factors that cannot be put into a machine learning model.

11. CONCLUSION

The subject of this examination was to determine if the below variables contribute to the admission of student to Master's degree program:

- GRE Score
- TOEFL Score
- University Rating
- SOP
- LOR
- CGPA

The results of this examination appear to indicate that it greatly contributes to the response variable 'Chance of Admit'. Higher the GRE, TOEFL score then higher the admit chances. The model predicts 80.7% accuracy and can be used for predicting the admit chances based on the above factors. This model will be helpful for the universities to predict the admission and ease their process of selection and timelines.

As part of the hypothesis, the model proved that admission to a Master's degree program is dependent on GRE, TOEFL and other scores.

This model would likely be greatly improved by the gathering of additional data of students from different universities which have similar selection criteria to choose the candidates for Master's program.

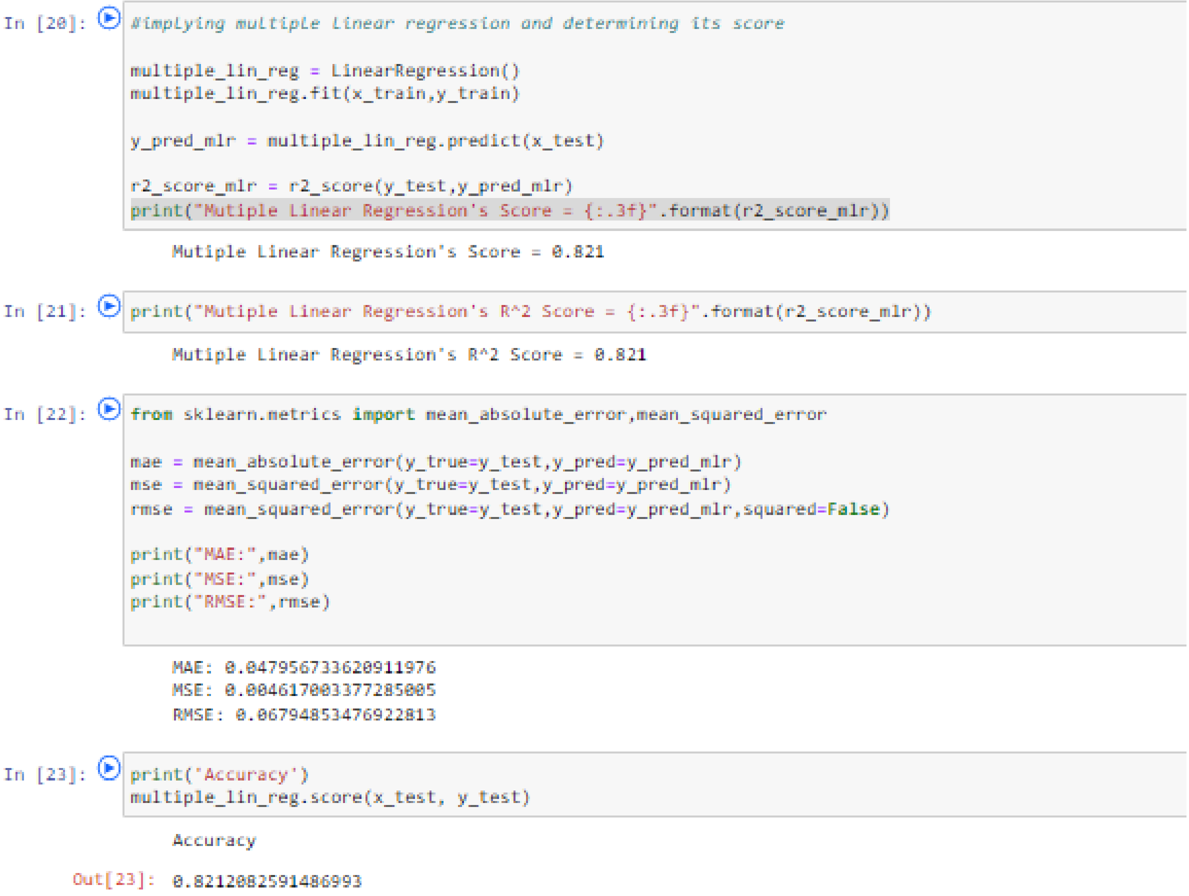
12. FUTURE SCOPE

A major future scope is to increase the accuracy of the model. University Admission depends on many factors, among them GPA, GRE and TOEFL are most important. We have used these three features in this project but more features can be implemented to get more accurate results. Other features could be Statement of Purpose (SOP), Letter of Recommendation (LOR) subjective evaluation, industry experience, internship experience, papers published, journals published etc. Also, as an extension to this work, recommendation of university with respect to research interest can be made with further study. A more semantic approach can be taken to the problem and a more meaningful diagnosis can be provided with actions to be taken instead of just a result. These are very feasible to work and to scale. This could be also used to provide resources to prepare for the admission process. The application can also be made more accommodating to processes in other countries.

13. APPENDIX

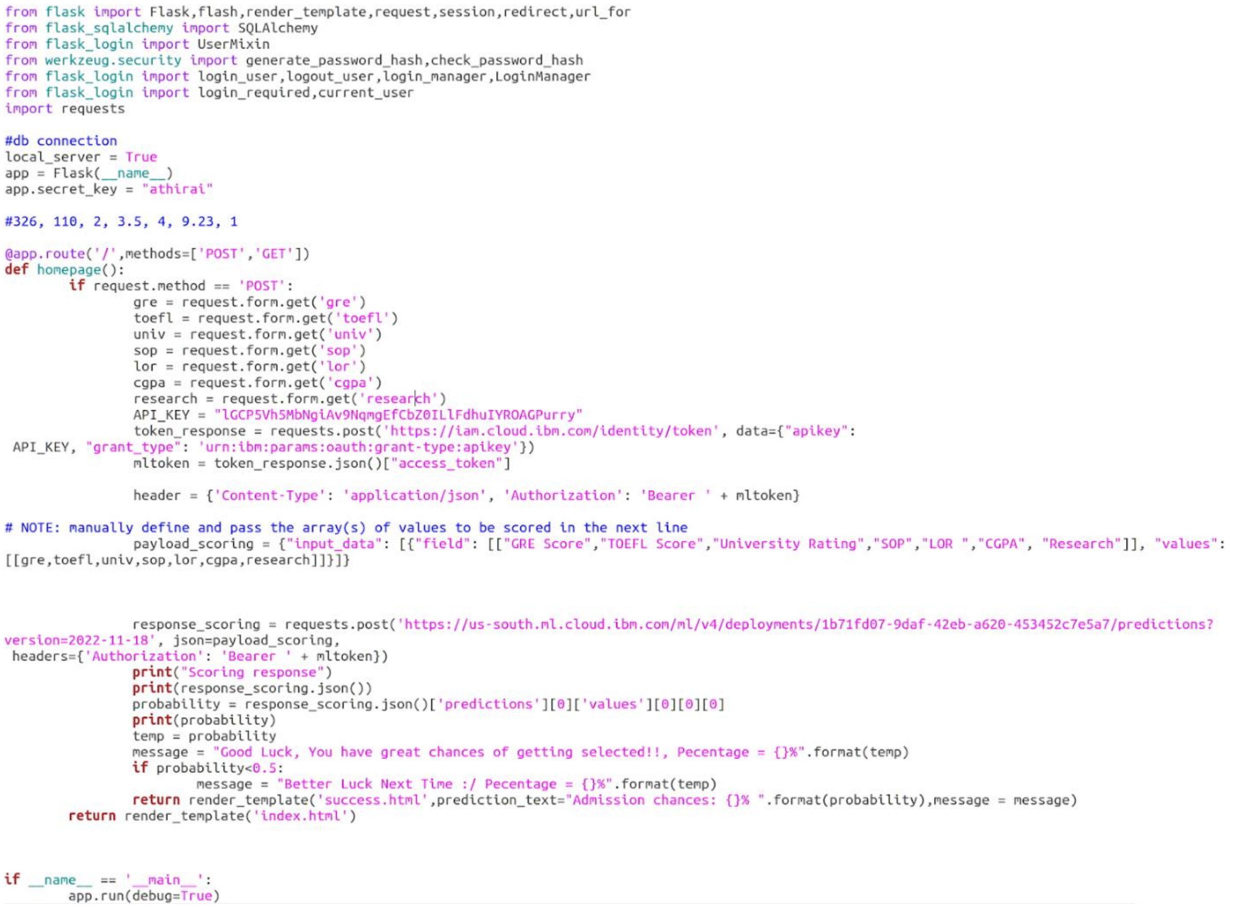
13.1 Source Code

MULTIPLE LINEAR REGRESSION



```
In [20]: #implying multiple linear regression and determining its score

         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("Mutiple Linear Regression's Score = {:.3f}".format(r2_score_mlr))

            Mutiple Linear Regression's Score = 0.821

In [21]: print("Mutiple Linear Regression's R^2 Score = {:.3f}".format(r2_score_mlr))

            Mutiple Linear Regression's R^2 Score = 0.821

In [22]: from sklearn.metrics import mean_absolute_error,mean_squared_error

         mae = mean_absolute_error(y_true=y_test,y_pred=y_pred_mlr)
         mse = mean_squared_error(y_true=y_test,y_pred=y_pred_mlr)
         rmse = mean_squared_error(y_true=y_test,y_pred=y_pred_mlr,squared=False)

         print("MAE:",mae)
         print("MSE:",mse)
         print("RMSE:",rmse)

            MAE: 0.047956733620911976
            MSE: 0.004617003377285005
            RMSE: 0.06794853476922813

In [23]: print('Accuracy')
         multiple_lin_reg.score(x_test, y_test)

            Accuracy

 Out[23]:   0.8212082591486993
```

MAIN.PY



```
from flask import Flask,flash,render_template,request,session,redirect,url_for
from flask_sqlalchemy import SQLAlchemy
from flask_login import UserMixin
from werkzeug.security import generate_password_hash,check_password_hash
from flask_login import login_user,logout_user,login_manager,LoginManager
from flask_login import login_required,current_user
import requests

#db connection
local_server = True
app = Flask(__name__)
app.secret_key = "athirai"

#326, 110, 2, 3.5, 4, 9.23, 1

@app.route('/',methods=['POST','GET'])
def homepage():
        if request.method == 'POST':
                gre = request.form.get('gre')
                toefl = request.form.get('toefl')
                univ = request.form.get('univ')
                sop = request.form.get('sop')
                lor = request.form.get('lor')
                cgpa = request.form.get('cgpa')
                research = request.form.get('research')
                API_KEY = "lGCP5Vh5MbNgiAv9NqmgEfCbZ0ILlFdhuIYROAGPurry"
                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":
[[gre,toefl,univ,sop,lor,cgpa,research]]}]}



                response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/1b71fd07-9daf-42eb-a620-453452c7e5a7/predictions?
version=2022-11-18', json=payload_scoring,
 headers={'Authorization': 'Bearer ' + mltoken})
                print("Scoring response")
                print(response_scoring.json())
                probability = response_scoring.json()['predictions'][0]['values'][0][0][0]
                print(probability)
                temp = probability
                message = "Good Luck, You have great chances of getting selected!!, Pecentage = {}%".format(temp)
                if probability<0.5:
                        message = "Better Luck Next Time :/ Pecentage = {}%".format(temp)
                return render_template('success.html',prediction_text="Admission chances: {}% ".format(probability),message = message)
        return render_template('index.html')


if __name__ == '__main__':
        app.run(debug=True)
```

13.2 GitHub & Project Demo Link

GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-50764-1660923555

DEMO VIDEO LINK

https://drive.google.com/file/d/1lIGmWleEVCf6ryBXMGOAVYleRx3AIySU/view?usp=sharing