University Admit Eligibility Predictor

USING APPLIED DATA SCIENCE

A Project report submitted in partial fulfillment of 7th semester in degreeof

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Team ID: PNT2022TMID51972

Sairaj P	962319104076
Praveena Pothi R K	962319104070
Nivetha R	962319104063
Vishal R	962319104101

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AMRITA COLLEGE OF ENGINEERING AND TECHNOLOGY

ANNA UNIVERSITY: CHENNAI 600025

NOV-2022

AMRITA COLLEGE OF ENGINEERING AND TECHNOLOGY

(A Constituent College of Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"UNIVERSITY ADMIT ELIGIBILITY PREDICTOR"** is the bonafide record work done by **Mr SAIRAJ P** (**962319104076**)**, Ms PRAVEENA POTHI R K(962319104070), Ms NIVETHA R (962319104063)** and **Mr VISHAL R**(**962319104101**) for **IBM-NALAIYATHIRAN** in **VII** semester of **B.E.,** degree course **in Computer Science and Engineering** branch during the academic year of 2022 - 2023.

Staff-In charge	**Evaluator**
Mrs Karpagavalli C	**Mrs JothiLakshmi S L**

Head of the Department
Dr.Sivaraja P M

ACKNOWLEDGEMENT

We express our breathless thanks to our **Dr.T.Kannan, M.E., Ph.D.,** the principal, Amrita College of Engineering and Technology, Erachakulam, for giving constantmotivation in succeeding in our goal.

We acknowledge our sincere thanks to Head of the Department (i/c) **Dr. P M Siva Raja M.Tech., Ph.D.,** for giving us valuable suggestion and help towards us throughout thisProject.

We are highly grateful to thank our Project coordinator **Mrs Karpagavalli C**

and our Project Evaluator **Mrs JothiLakshmi S L** Department of Computer Science and Engineering, Amrita college of Engineering and Technology for the coordinating us throughout this Project.

We are very much indebted to thank all the faculty members of Department of Computer science and Engineering in our Institute, for their excellent moral supportand suggestions to complete our Project work successfully.

Finally our acknowledgment does our parents, sisters and friends those who had extended their excellent support and ideas to make our Project a pledge one.

Sairaj P

Praveena pothi R K

Vishal R

Nivetha R

CONTENTS

1. INTRODUCTION

The world's business sector is escalating and is constantly seeking information and experiences that are commonly beneficial to individuals. Young specialists who need to stay in their current positions are always looking for advanced degrees to help them address their skills and information. As such, the number of her sophomores applying for graduation exams has increased over the past decade. One of her main concerns is getting into fantasy her university. You can see that undergraduates are actually choosing to get their education at prestigious universities. Furthermore, when it comes to international alumni, the United States is the main trend for most of them.The most prestigious universities offer a wide range of courses accessible in any order, exceptionally accredited teaching and education programs, an international second Research scholarships for degrees are available.

According to Gauges, more than 4,444 of her 10 million international sophomores are enrolled in her 4,200+ colleges and universities, both private and public. In general, the number of undergraduates concentrated in America comes from Asian countries such as India, Pakistan, Sri Lanka, Japan and China. Select the United Kingdom, Germany, Italy, Australia, Canada as well as the United States. These countries are witnessing a rapid increase in the number of individuals seeking more advanced investigations. The basic reason why sophomores go on to master's programs in foreign graduate schools is that the number of vacancies is low and the number of people in these positions in each country is huge. This has led many professional undergraduates to pursue postgraduate studies. You can see that there are quite a few bachelor's degrees and master's degrees in computer science at US universities. The focus of this study applies to these undergraduate degrees. Many schools in the US follow comparative requirements for undergraduate accreditation. Schools consider several variables, including placement in fitness assessments and school performance ratings. English rankings are determined by exposure in English proficiency tests such as TOEFL and IELTS.

The University's Admissions Advisory Board makes decisions regarding the acceptance or rejection of specific young researchers based on the general profile of the applicant's application. Records recorded with this company are marked with informative areas. Acknowledgment is a 400-row data set containing seven different autonomic factors. ie

1. Graduate Record Examination 1 (GRE) score. The score consists of 340 foci.
2. English as a Foreign Language (TOEFL) test score. It consists of 120 priority areas.
3. Uni.Rating. Shows the position of colleges offering bachelor's degrees among various colleges.Your score will be out of 5.
4. Statement of Purpose (SOP), a record written to reveal the life, motivations and inspirations of a selected degree/college applicant. The score consists of five focal points.
5. The strength of a letter of recommendation (LOR) verifies the applicant's professional experience, falsifies validity, supports certainty, and guarantees your competence. The score consists of five focal points.
6. Undergraduate GPA (CGPA) from 10.

Research experience (either 0 or 1) that could support the application, such as distributing research papers at conferences or filling out as a right-hand exam for university faculty. One ward variable can be anticipated which is possibility of affirmation, that is as per the input given will be going from 0 to 1

1.1 PROJECT OVERVIEW

PRE REQUISTIES

Anaconda Installation:

Anaconda is a distribution of the Python and R programming languages for scientific computing that aims to simplify package management and deployment. The distribution includes data science packages suitable for Windows, Linux, and macOS. Developed and maintained by Anaconda. Founded in 2012 by Peter Wang and Travis Olyphant. As Anaconda, also known as Anaconda Distribution or Anaconda Individual Edition, the company's other products include his Anaconda Team Edition and Anaconda Enterprise Edition, neither of which are free.

PROJECT FLOW

You will go through all the steps mentioned below to complete the project.

1. User interacts with the UI (User Interface) to enter Data
2. The entered data is analyzed by the model which is integrated
3. Once model analyses the input the prediction is showcased on the UI

To accomplish this, we have to complete all the activities and tasks listed below

4. Data Collection.
    a. Collect the dataset or Create the dataset
5. Data Preprocessing.

    a. Import the Libraries.
    b. Importing the dataset.
    c. Checking for Null Values.
    d. Data Visualization.
    e. Taking care of Missing Data.
    f. Label encoding.
    g. One Hot Encoding.
    h. Feature Scaling.
    i. Splitting Data into Train and Test.

6. Model Building
    a. Training and testing the model
    b. Evaluation of Model
7. Application Building
    a. Create an HTML file
    b. Build a Python Code

PROJECT OBJECTIVES

1. To understand regression and classification problems.
2. To grab insights from data through visualization.
3. Applying different Machine Learning algorithms to determine the probability ofacceptance in a particular university.
4. Evaluation metrics build a web application using the Flask framework

DATA PRE-PROCESSING

Importing the Libraries:

It is important to import all the necessary libraries such as pandas, numpy, matplotlib.

1.Numpy-

It is an open-source numerical Python library. It contains a multi-dimensional array and matrix data structures. It can be used to perform mathematical operations on arrays such as trigonometric, statistical, and algebraic routines.

2.Pandas-

It is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language.

3.Seaborn-

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

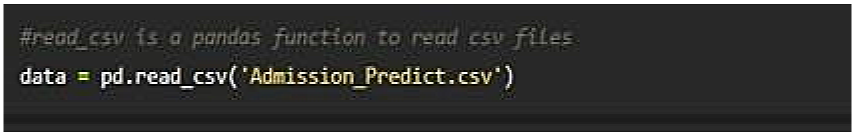
Matplotlib-

Visualisation with python. It is a comprehensive library for creating static, animated, and interactive visualizations in Python

Reading the Datasets:

You might have your data in .csv files, .excel files

Let’s load a .csv data file into pandas usingread_csv() function. We will need to locate the directory of the CSV file at first



```
#read_csv is a pandas function to read csv files
data = pd.read_csv('Admission_Predict.csv')
```

If your dataset is in some other location ,Then see below commandData= pd.read_csv(r”File_location/filename.csv”)

Note: r stands for "raw" and will cause backslashes in the string to be interpreted as actualbackslashes rather than special characters.

Our Dataset Admission_Predict contains following Columns1.Serial No.

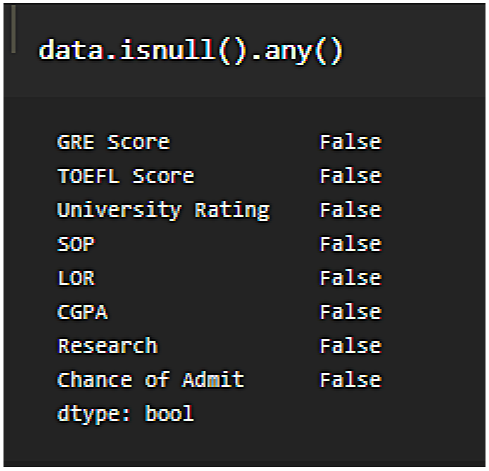
1. Serial No
2. GRE Score
3. TOEFL Score
4. University Rating
5. SOP
6. LOR
7. Chance of Admitd

Handling Missing Values:

After loading it is important to check the complete information of data as it can indicate many ofthe hidden information such as null values in a column or a row Check for the null values. if it is present then the following steps can be performed

a. Imputing data using the Imputation method in sklearn.

b. Filling NaNvalues with mean, median, and mode using fillna() method.You can check the null values with the function isnull().any()



```
data.isnull().any()

GRE Score            False
TOEFL Score          False
University Rating    False
SOP                  False
LOR                  False
CGPA                 False
Research             False
Chance of Admit      False
dtype: bool
```

a. If the dataset contains null values then the above functions return as true. But if you lookat the dataset you can observe that the dataset does not have any null values.

b. You can also check the number of null values present in the columns by the usingisnull().sum() function

As we don't have categorical data then we can skip the steps of label encoding and one-hot encoding

Data Visualization:

Data visualization is where a given dataset is presented in a graphical format. It helps the detection of patterns, trends and correlations that might go undetected in text-based data. Understanding your data and the relationship present within it is just as important as any algorithm used to train your machine learning model. Machine learning models will perform poorly on data that wasn't visualized and understood properly.

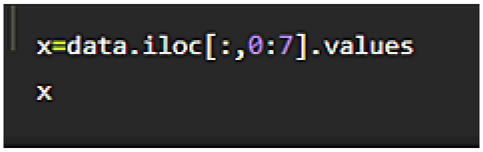
To visualize the dataset we need libraries called Matplotlib and Seaborn. The Matplotlib library is a Python 2D plotting library that allows you to generate plots, scatter plots, histograms, bar charts etc.

Splitting Dependent And Independent Columns:

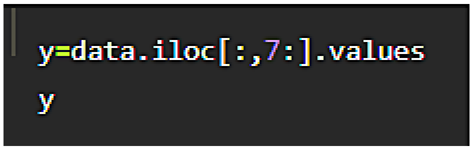
We need to split our dataset into the matrix of independent variables and the vector or dependent variable. Mathematically, Vector is defined as a matrix that has just one column.

a. To read the columns, we will useiloc of pandas (used to fix the indexes for selection)which takes two parameters — [row selection, column selection].

Let's split our dataset into independent and dependent variables.



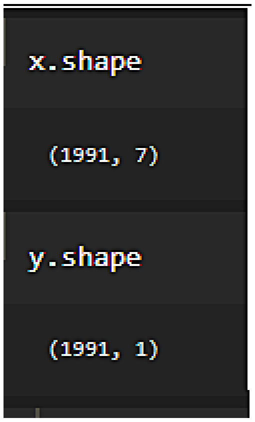
```
x=data.iloc[:,0:7].values
x
```



```
y=data.iloc[:,7:].values
y
```

From the above code “:” indicates that you are considering all the rows in the dataset and “0:7” indicates that you are considering columns 0 to 7 such as year, month, and day as input values and assigning them to variable x. In the same way in the second line “:” indicates you are considering all the rows and “7:” indicates that you are considering only the last column as output value and assigning them to variable y.

Let's Check the shape of x and Y



```
x.shape
```

```
(1991, 7)
```

```
y.shape
```

```
(1991, 1)
```

a. You can see in x we have 1991 rows with 7 columns and y has 1 column with the samenumber of rows

Splitting The Data Into Train And Test:

To train the model, first split the model into two segments: "training data" and "testing data". The classifier is trained using a 'training data set' and the performance of the classifier is tested on a non-fitting 'test data set'.

Training Set: The training Set is material for computers to learn how to process data. The AI uses computation to do the training part. The training dataset is used to learn and tune the classifier parameters.

Test set: A set of unseen data used solely to evaluate the performance of thefully displayed classifier.

When you are working on a model and you want to train it, you obviously have a dataset. But after training, we have to test the model on some test dataset. For this, you will need a dataset that is different from the training set you used earlier. But it might not always be possible to have so much data during the development phase. In such cases, the solution is to split the dataset into two sets, one for training and the other for testing.

To help us with this task, the Scikit library provides a tool, called the Model Selection library. There is a class in the library which is, ‘train_test_split.’ Using this we can easily split the datasetinto the training and the testing datasets in various proportions.

The train-test split is a technique for evaluating the performance of a machine learning algorithm.

b. Train Dataset:Used to fit the machine learning model.

Test Dataset:Used to evaluate the fit machine learning model.

In general, you can allocate 80% of the dataset to the training set and the remaining 20% to the test set. We will create 4 sets

a. x_train

b. x_test

c. y_train

d. y_test .

There are a few other parameters that we need to understand before we use the class:

e. test_size: this parameter decides the size of the data that has to be split as the test dataset. This is given as a fraction. For example, if you pass 0.5 as the value, the dataset will be split 50% as the test dataset and remaining a train dataset

f. random_state: here you pass an integer, which will act as the seed for the random number generator during the split. Or, you can also pass an instance of the Random_state class, which will become the number generator. If you don’t pass anything, the Random_state instance used by np.random will be used instead.

1.2 PURPOSE

This is the project for a new web-based University Admit Eligibility Predictor. Predictor is an ML based application that asks for the users to input their academic transcripts data and calculates their chances of admission into the University Tier that they selected. It also provides an analysis of the data and shows how chances of admissions can depend on various factors. This document describes the scope, objectives and goals of the system. In addition to describing the non-functional requirements, this document models the functional requirements with use cases, interaction diagrams and class models. This document is intended to direct the design andimplementation of the target system in an object-oriented language.

2. LITERATURE SURVEY

2.1 Existing Problem

It’s almost admission season and I’ve couple of friends who are in panic mode waiting for a callfrom the universities they’ve applied at.
This made me think — How can we predict whether a student will get an admit or not? What arethe parameters for selection? Can it be mathematically expressed?
All of these questions started popping up. This is the main existing problem.

2.2 References

- ➤ https://ieeexplore.ieee.org/document/9418279

Abstract:

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.
References: MS Acharya, A Armaan and AS Antony, "A comparison of regression models for prediction of graduate admissions", 2019.

- ➤ https://ieeexplore.ieee.org/document/9410717

Abstract:

Students applying for admissions to universities find it difficult to understand whether they have good chances of getting admission in a university or not. Keeping this in focus, we have used logistic regression techniques that have gained attention in software engineering field for its ability to be used for predictions. This is a novel work on a university admissions predictor using which students can evaluate their competitiveness for getting admission at a university.

References: M. Fatima and M. Pasha, "Survey of machine learning algorithms for disease diagnostic", *Journal of Intelligent Learning Systems and Applications*, vol. 9, no. 01, pp. 1, 2017.

➤ https://ieeexplore.ieee.org/document/6416521

Abstract:

This paper presents a new college admission system using hybrid recommender based on data mining techniques and knowledge discovery rules, for tackling college admissions prediction problems. This is due to the huge numbers of students required to attend university colleges every year. The proposed HRSPCA system consists of two cascaded hybrid recommenders working together with the help of college predictor, for achieving high performance.

References: G. Ganapathy, and K. Arunesh, "Models for Recommender Systems in Web Usage Mining Based on User Ratings" Proceedings of the World Congress on Engineering, Vol. I WCE2011.

➤ https://dl.acm.org/doi/10.1145/3388818.3393716

Abstract:

With the increase in the number of graduates who wish to pursue their education, it becomes more challenging to get admission to the students' dream university. Newly graduate students usually are not knowledgeable of the requirements and the procedures of the postgraduate admission and might spent a considerable amount of money to get advice from consultancy organizations to help them identify their admission chances.

References: E. Roberts, "using machine learning and predictive modeling to assess admission policies and standards," 2013.

- https://medium.com/@jigar18011999/university-predictor-by-machine-learning-2d880e9f3a3

Abstract:

This article describes the architecture and algorithms of the proposed system. ANN, decision trees, and logistic regression were used to find admissions for a particular student. ML models take into account various parameters such as GRE and TOEFL scores, SOP, and LOR. Finally, after evaluation, the authors state that decision trees are the most accurate among the tree algorithms used.

- https://github.com/satwik2663/Machine-Learning-Graduate-Studuent-Admission-Predictor

Abstract:

Today, there are many students who travel to USA to pursue higher education. It is necessary for the students to know what are their chances of getting an admit in the universities. Also, universities manually check and count the total number of applicants who could get an admit into university. These methods are slow and certainly not very consistent for students and universities to get an actual result. This method is also prone to human error and thus accounts for some inaccuracies. Since the frequency of students studying abroad has increased, there is a need to employ more efficient systems which handle the admission process accurately from both perspectives.

➤ https://github.com/anjanatiha/University-Admission-Match-Predictor

Abstract:

i. Analyzed university admission statistics.

ii. Developed tools for matching university (in percentile) using CGPA, GRE (Verbal,Quantitative, Analytical Writing) scores.

2.3 Problem Statement Definition

Problem Statement(PS)	**I am (Customer)**	**I'm trying to**	**But**	**Because**	**Which makes mefeel**
PS-1	Student	I am looking university forlong time	Still I am facing difficulties to select the university that is having good environment	University is in Chennai but I want in my Home town	Difficult tofind the university.
PS-2	Student	I am Searching University on my phone	It is difficultto find and it takes long time	It is difficult tosearch in offline	Exhausted

➤ https://github.com/karanwadhwa/dd-admission-predictor

Abstract:

This system was originally developed only for Engineering College Admissions in Maharashtra,India but can essentially be adapted for other streams too. The purpose of it is to build a system to predict the users chances for getting into a certain college.

3. IDEATION AND PROPOSED SOLUTION

Ideation is the process where 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 process. In thisproject the ideation phase consist of,

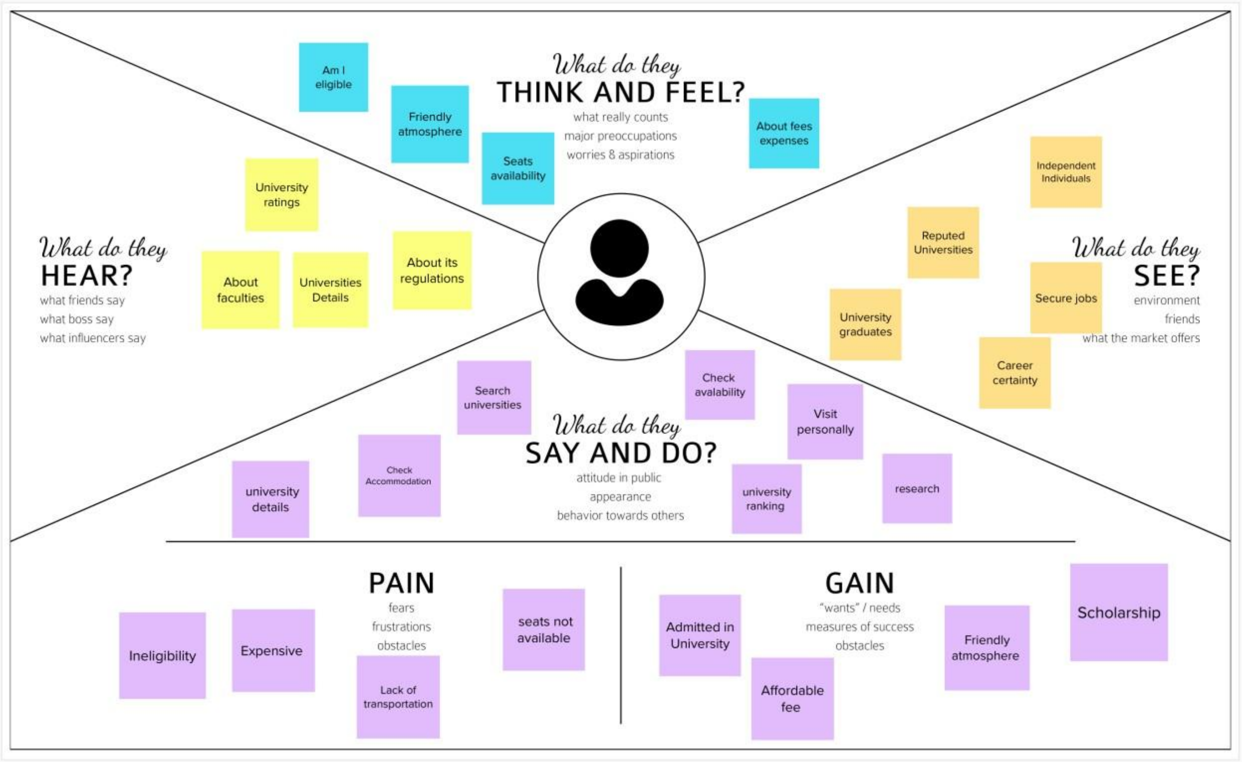
1. Empathy Map

2.Brainstorming

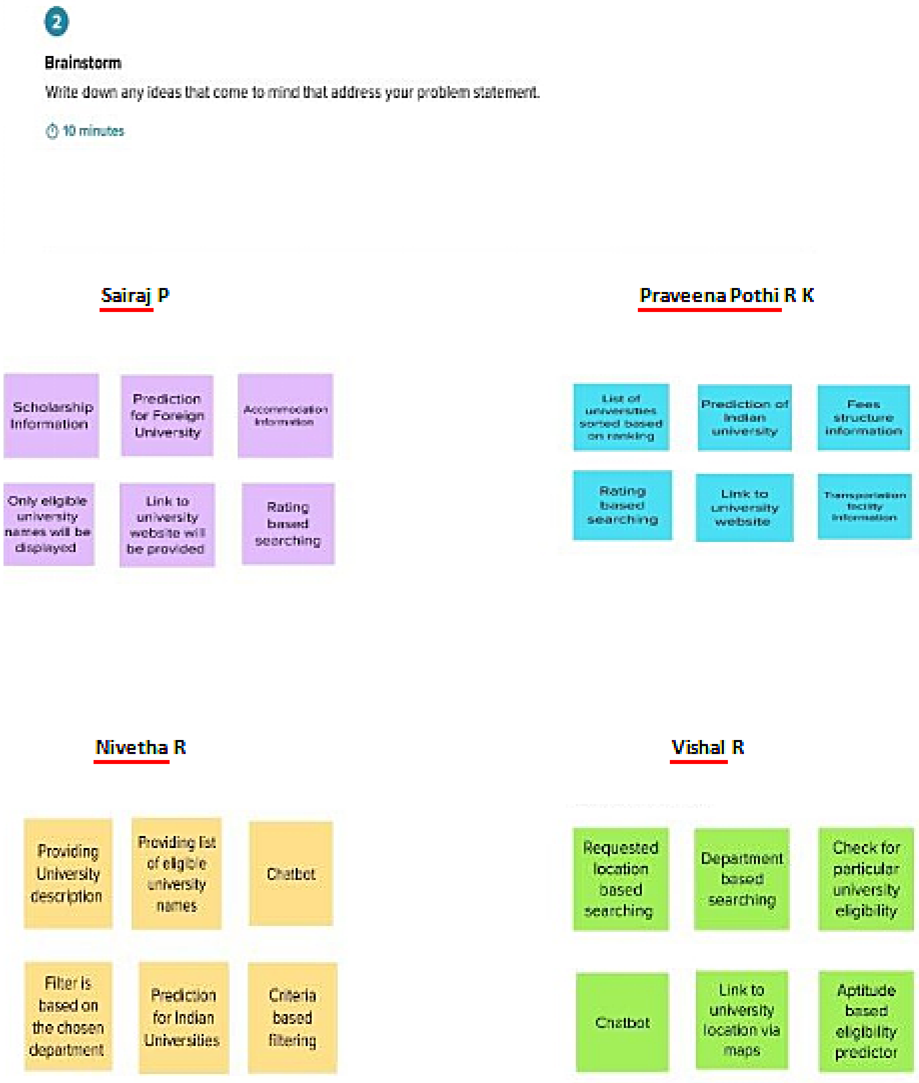
3.Proposed Solution

4.Problem Solution Fit

3.1 EMPATHY MAP



.3.2 BrainStorming & Ideation



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Sairaj P

Scholarship Information

Prediction for Foreign University

Accommodation Information

Only eligible university names will be displayed

Link to university website will be provided

Rating based searching

Praveena Pothi R K

List of universities sorted based on ranking

Prediction of Indian university

Fees structure information

Rating based searching

Link to university website

Transportation facility information

Nivetha R

Providing University description

Providing list of eligible university names

Chatbot

Filter is based on the chosen department

Prediction for Indian Universities

Criteria based filtering

Vishal R

Requested location based searching

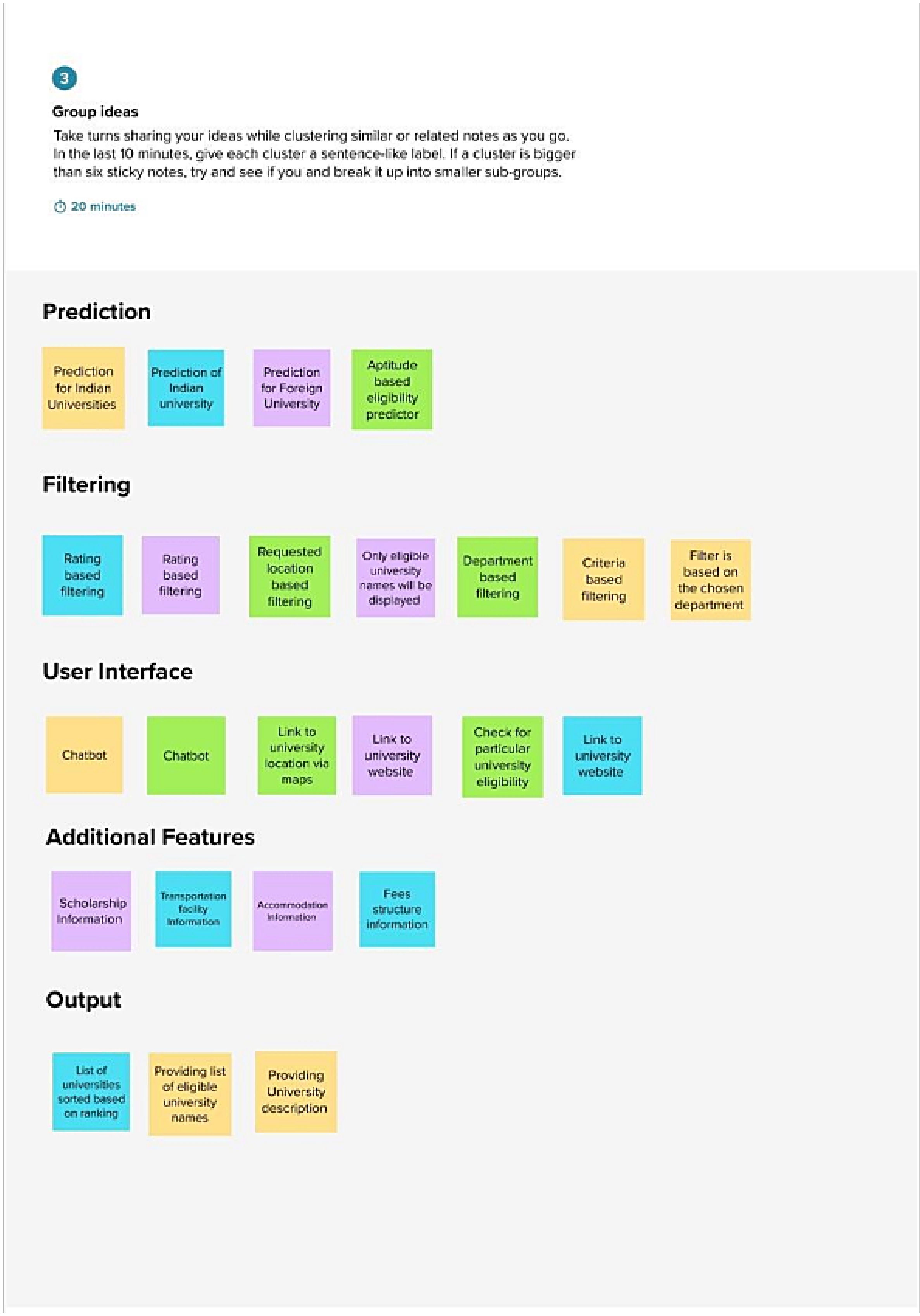
Department based searching

Check for particular university eligibility

Chatbot

Link to university location via maps

Aptitude based eligibility predictor



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Prediction

- Prediction for Indian Universities
- Prediction of Indian university
- Prediction for Foreign University
- Aptitude based eligibility predictor

Filtering

- Rating based filtering
- Rating based filtering
- Requested location based filtering
- Only eligible university names will be displayed
- Department based filtering
- Criteria based filtering
- Filter is based on the chosen department

User Interface

- Chatbot
- Chatbot
- Link to university location via maps
- Link to university website
- Check for particular university eligibility
- Link to university website

Additional Features

- Scholarship Information
- Transportation facility information
- Accommodation information
- Fees structure information

Output

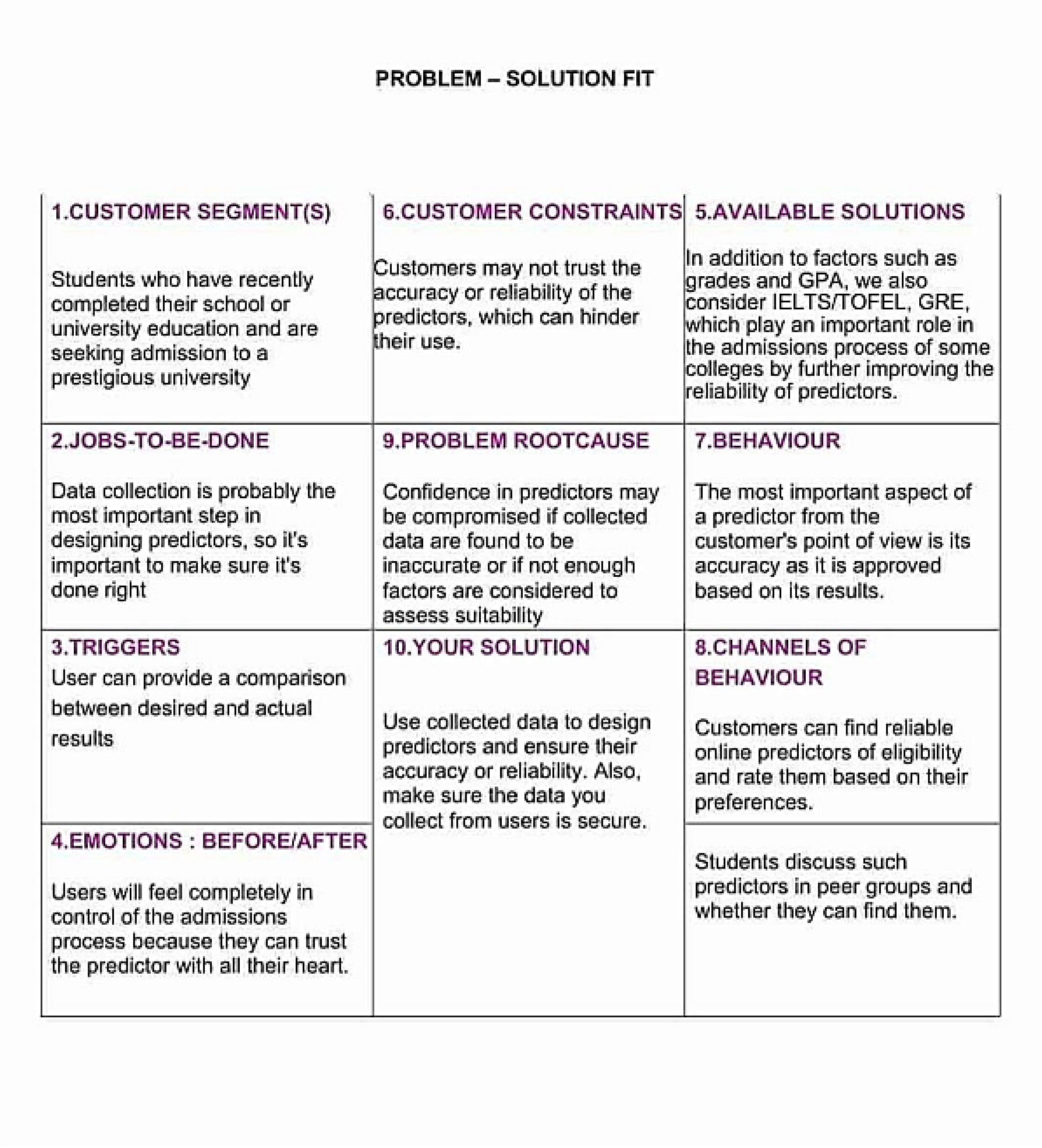
- List of universities sorted based on ranking
- Providing list of eligible university names
- Providing University description

3.3 Proposed Solution

1.	Problem Statement (Problemto besolved)	Students often worry about their chances of getting into college. The goal of this project is to help profile shortlisted college students. Predicted results give them a good ideaof their likelihood of getting into a particular college. This analysis is also useful for students preparing or planning to prepare for a betterimage must. It also aimsto connect students and universities directly, withoutintermediaries.
2.	Idea/ Solution description	This project aims to calculate the likelihood of admission to aparticular graduate schoolafter evaluating a candidate'sprofile. The main attributes considered in decision makingare: 1. GRE &TOEFL Scores 2. Undergraduate CGPA

			3. SOP &LOR 4. CorporateWorkExperience/Research ExperienceExtracurricular Activities 5. Extra-curriculars Determine Acceptance Rate, Logistic Regression, Multi linear, Usea variety of ML models such as regression, decision trees,and random forests, and use performance metrics such as

3.4 Problem Solution fit



PROBLEM – SOLUTION FIT

1.CUSTOMER SEGMENT(S) Students who have recently completed their school or university education and are seeking admission to a prestigious university	**6.CUSTOMER CONSTRAINTS** Customers may not trust the accuracy or reliability of the predictors, which can hinder their use.	**5.AVAILABLE SOLUTIONS** In addition to factors such as grades and GPA, we also consider IELTS/TOFEL, GRE, which play an important role in the admissions process of some colleges by further improving the reliability of predictors.
2.JOBS-TO-BE-DONE Data collection is probably the most important step in designing predictors, so it's important to make sure it's done right	**9.PROBLEM ROOTCAUSE** Confidence in predictors may be compromised if collected data are found to be inaccurate or if not enough factors are considered to assess suitability	**7.BEHAVIOUR** The most important aspect of a predictor from the customer's point of view is its accuracy as it is approved based on its results.
3.TRIGGERS User can provide a comparison between desired and actual results	**10.YOUR SOLUTION** Use collected data to design predictors and ensure their accuracy or reliability. Also, make sure the data you collect from users is secure.	**8.CHANNELS OF BEHAVIOUR** Customers can find reliable online predictors of eligibility and rate them based on their preferences.
4.EMOTIONS : BEFORE/AFTER Users will feel completely in control of the admissions process because they can trust the predictor with all their heart.		Students discuss such predictors in peer groups and whether they can find them.

4.Requirment Analysis

4.1 Functional Requirment

FR No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	User Registration	Registration throughForm Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login through username and passwordLogin through GmailLogin through LinkedIN
FR-4	Administration work	Check qualified candidate detail Make allotment
FR-5	Admission Details	Check seat availabilityCheck college infrastructure Check fees details
FR-6	Local counsellor	Issue the final allotment order

4.2 Non Functional Requirment

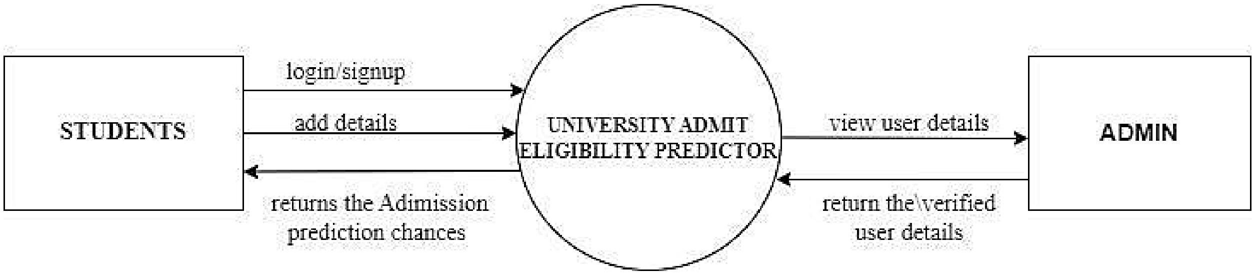
FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	a. A logical interface is essential to makeeasy use ofsystem, speeding up common tasks. b. The product could be used by two categories of people mainly administrator category andotherusers.

NFR-2	**Security**	Some of the factors that are identified toprotectthe software from accidental or malicious access,use, modification, destruction, or disclosure are described below: a. Keep specific log or history data sets. b. Utilize certain cryptographic techniques. c. Restrict the no of systems that can access the online admission system site. This could be done only by registering the systems physical addresses before using them for online admissionprocess. a. Check data integrity for criticalvariables. b. Every user should be licensed to usethe systemunder any of the four categories provided i.e. either verifier or advisor or local counsellor or administrator. c. Communication needs to be restrictedwhen the application is validating the user or license.

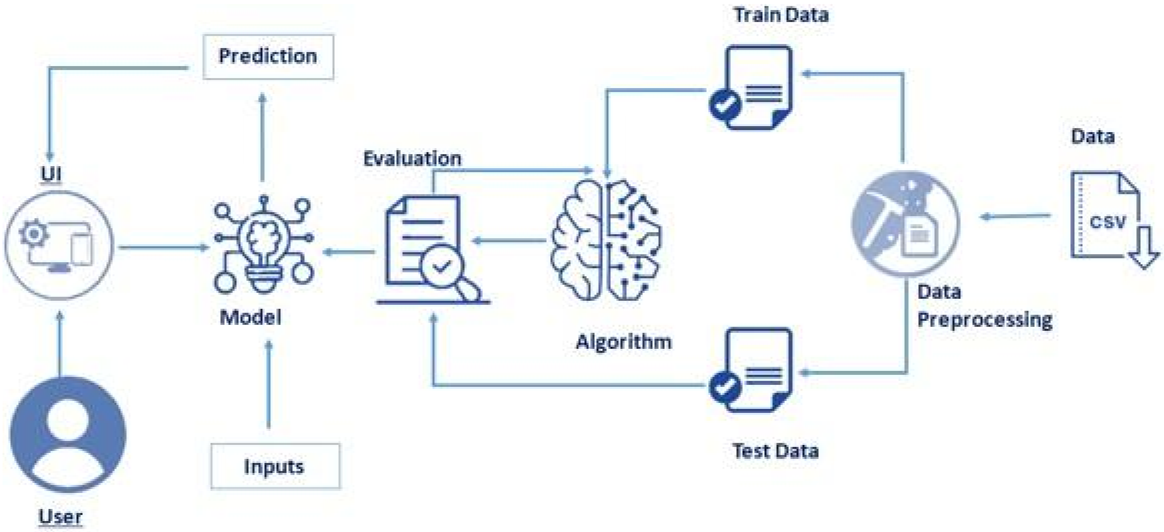
NFR-3	**Reliability**	a. All data storage for user variables will be committed to the database atthe time of entry. b. Data corruption is prevented byapplying thepossible backup procedures and techniques.
NFR-4	**Performance**	a. The database should be able toaccommodate aminimum of 10,000 records of students. b. At any instant the system should support use ofmultiple users at a time. c. Availability results of the requestedcollege should be presented to the student in max of twoseconds, so retrieving of data should be reliable. d. As each student will be given a maximum timeof 10min, accessing from the database

5.PROJECT DESIGN

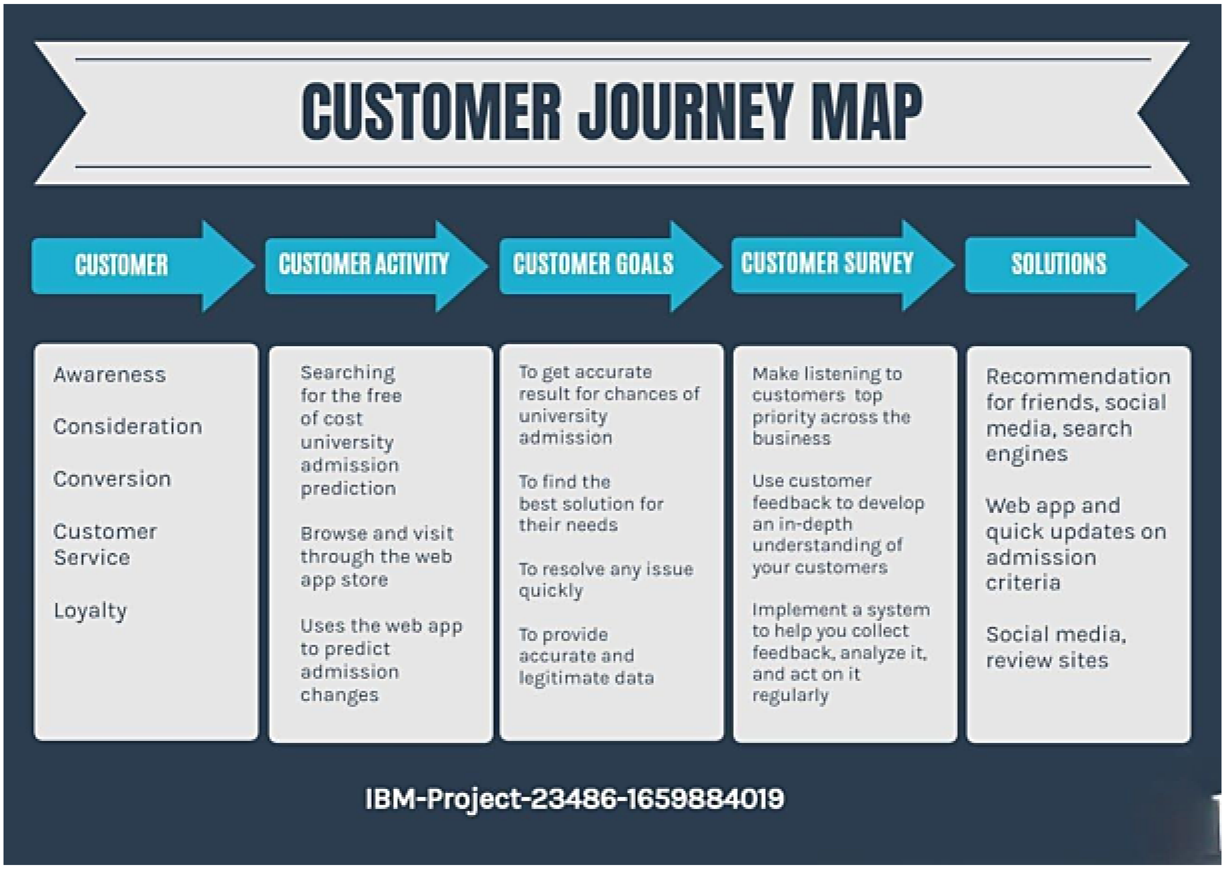
5.1 Data Flow Diagram


STUDENTS
login/signup
add details
returns the Adimission prediction chances
UNIVERSITY ADMIT ELIGIBILITY PREDICTOR
view user details
return the\verified user details
ADMIN

5.2 Solution & Technical Architecture



5.3 User Stories



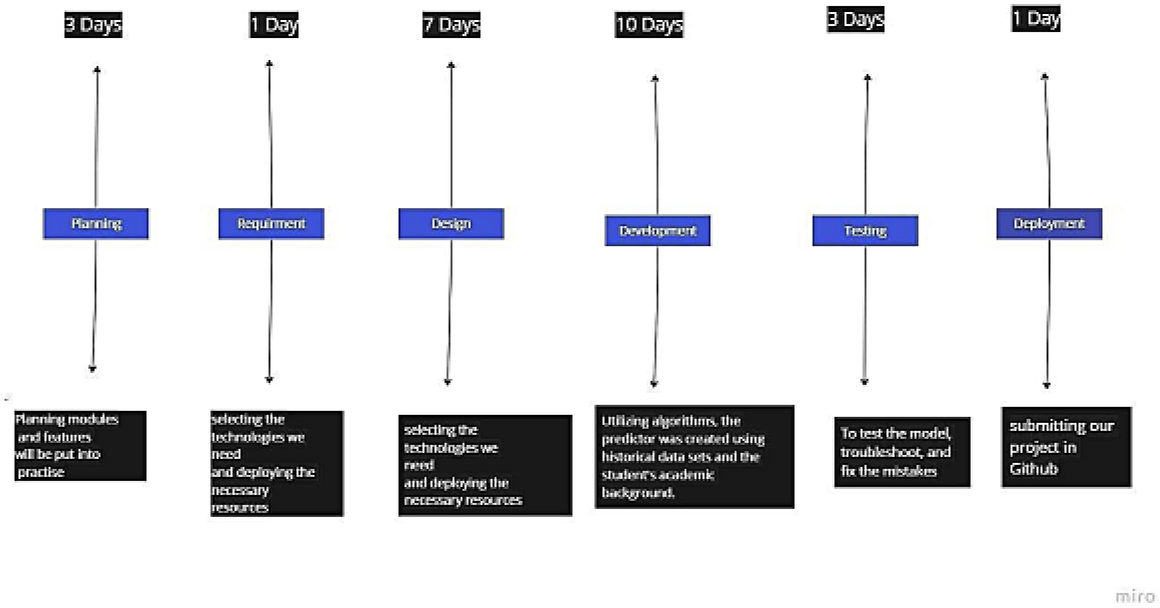
CUSTOMER JOURNEY MAP

CUSTOMER	CUSTOMER ACTIVITY	CUSTOMER GOALS	CUSTOMER SURVEY	SOLUTIONS
Awareness Consideration Conversion Customer Service Loyalty	Searching for the free of cost university admission prediction Browse and visit through the web app store Uses the web app to predict admission changes	To get accurate result for chances of university admission To find the best solution for their needs To resolve any issue quickly To provide accurate and legitimate data	Make listening to customers top priority across the business Use customer feedback to develop an in-depth understanding of your customers Implement a system to help you collect feedback, analyze it, and act on it regularly	Recommendation for friends, social media, search engines Web app and quick updates on admission criteria Social media, review sites

IBM-Project-23486-1659884019

6.PROJECT PLANNING AND SCHEDULING

6.1 Sprint planning and estimation


3 Days
1 Day
7 Days
10 Days
3 Days
1 Day
Planning
Requirment
Design
Development
Testing
Deployment
Planning modules and features will be put into practise
selecting the technologies we need and deploying the necessary resources
selecting the technologies we need and deploying the necessary resources
Utilizing algorithms, the predictor was created using historical data sets and the student's academic background.
To test the model, troubleshoot, and fix the mistakes
submitting our project in Github
miro

6.2 Sprint delivery schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and	10	High	All

			confirming my password			
Sprint-1	Gmail notification	USN-2	As a user, I will receive confirmation email once I have registered for the application	5	Low	All
Sprint-2	Login	USN-3	As a user, I can login with my credential and geta personalized dashboard	5	Low	All
Sprint-2	View institutes	USN-4	As a user, I can view list of institutes	10	Low	All
Sprint-3	prediction	USN-5	As a user, I can provide my scores for prediction	10	High	All
Sprint-4	preferences	USN-6	As a user I can select and filter results based on my preferences	20	High	All

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

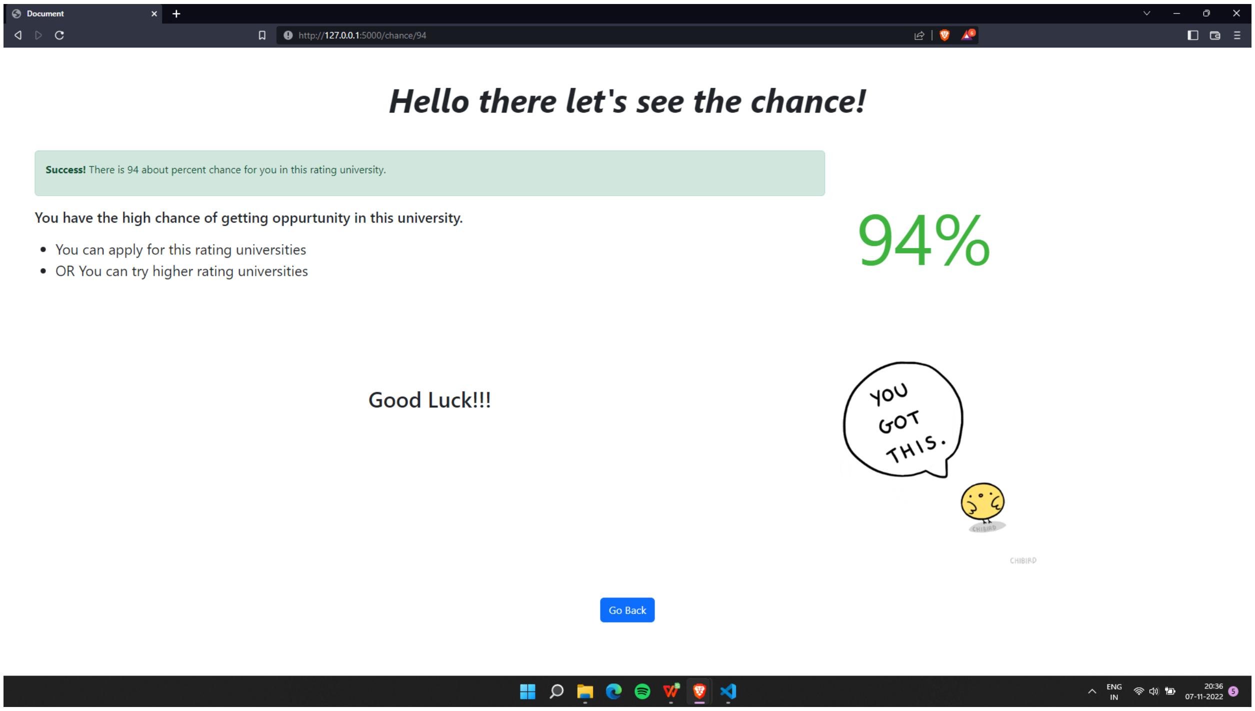
Av=60 story points/24 days=2.5

Burn down Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

7.CODING AND SOLUTIONING

7.1 Feature 1



Hello there let's see the chance!

Success! There is 94 about percent chance for you in this rating university.

You have the high chance of getting oppurtunity in this university.

- You can apply for this rating universities
- OR You can try higher rating universities

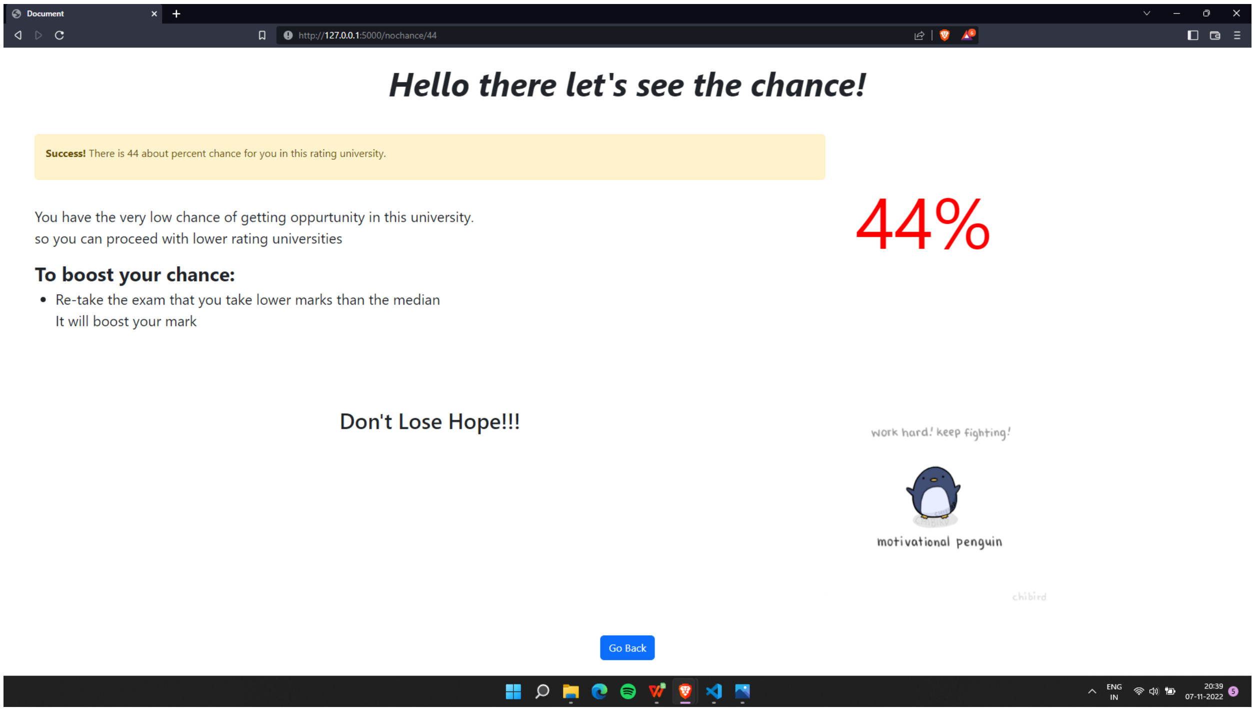
94%

Good Luck!!!

Source code:

```
{% block body%}
 <body>

    <h1 class="text-center" id="title" ><strong>Hello there let's see the chance!</strong></h1>
    <div class="m-5 d-flex flex-row ">
        <div class="col-sm-8">
            <div class="alert alert-success" style="height:70px;">
                <strong>Success!</strong> There is {{content[0]}} about percent chance for you in this rating university.
              </div>
              <p style="font-size:22px; font-weight: 500;">You have the high chance of getting oppurtunity in this university.<ul style="font-size:22px;"><li> You can apply for this rating universities</li><li>OR You can try higher rating universities</li></ul></p>
        </div>
       <div class="m-5" id="mark1"><span>{{content}}%</span></div>
       </div>
       <div class="m-5 d-flex flex-row ">
        <div class="col-sm-8"><p class="text-center pt-5" style="font-size:33px; font-weight: 500;">Good Luck!!!</p></div>
        <div class="">
            <img id="image"src="/static/success1.gif" alt="success">
           </div>
       </div>

       <div class="d-flex justify-content-center">
        <button class="btn btn-primary " style="margin-top: 0;"><a href="/" style="color:white; text-decoration : none" >Go Back</a></button></div></body>
{%end block body%}
```

7.2 Feature 2



Hello there let's see the chance!

Success! There is 44 about percent chance for you in this rating university.

44%

You have the very low chance of getting oppurtunity in this university.
so you can proceed with lower rating universities

To boost your chance:

- Re-take the exam that you take lower marks than the median
  It will boost your mark

Don't Lose Hope!!!

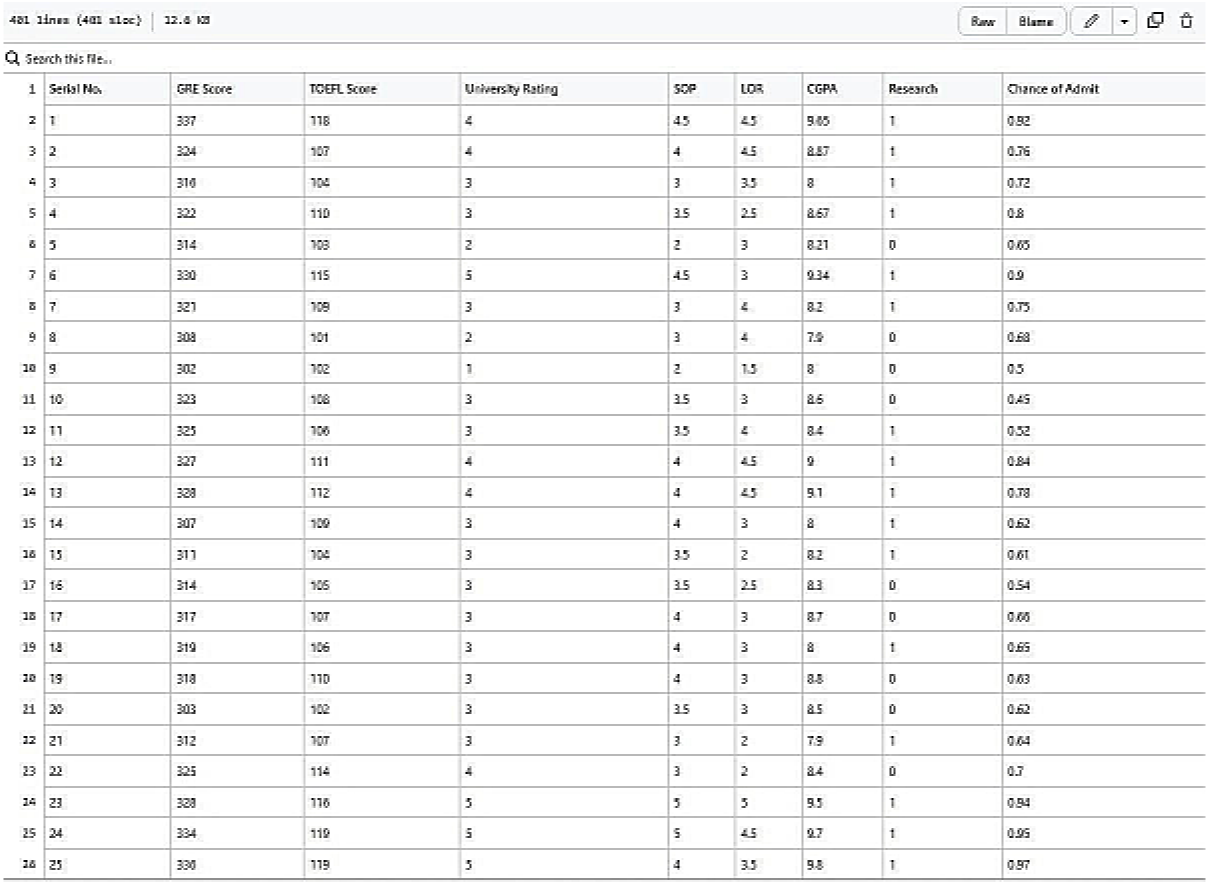
Source code:

```
{% block body %}

<body>

    <h1 class="text-center" style="margin-top:25px; font-style: italic; font-weight:400; font-size:
50px;"><strong>Hello there let's see the chance!</strong></h1>
    <div class="m-5 d-flex flex-row ">
        <div class="col-sm-8">
            <div class="alert alert-warning" style="height:70px;">
                <strong>Success!</strong> There is {{content[0]}} about percent chance for you in this
```

```
rating university.
            </div><br>
            <p style="font-size:22px; font-weight: 400;">You have the very low chance of getting
oppurtunity in this university.<br>so you can proceed with lower rating universities</p>
            <strong style="font-size:30px;">To boost your chance:</strong>
            <ul style="font-size:22px;">
              <li>Re-take the exam that you take lower marks than the median <br>
                 It will boost your mark</li>
            </ul>
       </div>
      <div class="m-5" id="mark2"><span>{content[0]}}%</span></div>
      </div>
      <div class="m-5 d-flex flex-row ">
       <div class="col-sm-8"><p class="text-center pt-5" style="font-size:33px; font-weight: 500;">Don't
Lose Hope!!!</p></div>
       <div class="">
          <img id="image" src="/static/motivation.gif" alt="motivation">
         </div>
      </div>

      <div class="d-flex justify-content-center">
       <button class="btn btn-primary " style="margin-top: 0;"><a href="/" style="color:white; text-
decoration : none" >Go Back</a></button>
      </div>
</body>
```

7.3 Database Schema

The database used here in this project was Admission_Predict.csv. The sample screenshot of thedatabase are,



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
24	334	119	5	5	4.5	9.7	1	0.95
25	336	119	5	4	3.5	9.8	1	0.97

8 TESTING

8.1 Test Cases

Test Case Analysis

This report shows the number of test cases that have passed , failed , and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3

Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.2 User Acceptance Testing

1.Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the University Admit Eligibility Predictor project at the time of the release to User Acceptance Testing (UAT).

2.Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, andhow they were resolved

Resolution	**Severity 1**	**Severity 2**	**Severity 3**	**Severity 4**	**Subtotal**
By Design	10	4	2	3	19
Duplicate	0	0	0	0	0
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reprodu	0	0	0	0	0

ced					
Skipped	0	0	1	1	2
Won't Fix	0	0	0	0	0
Totals	24	14	13	26	64

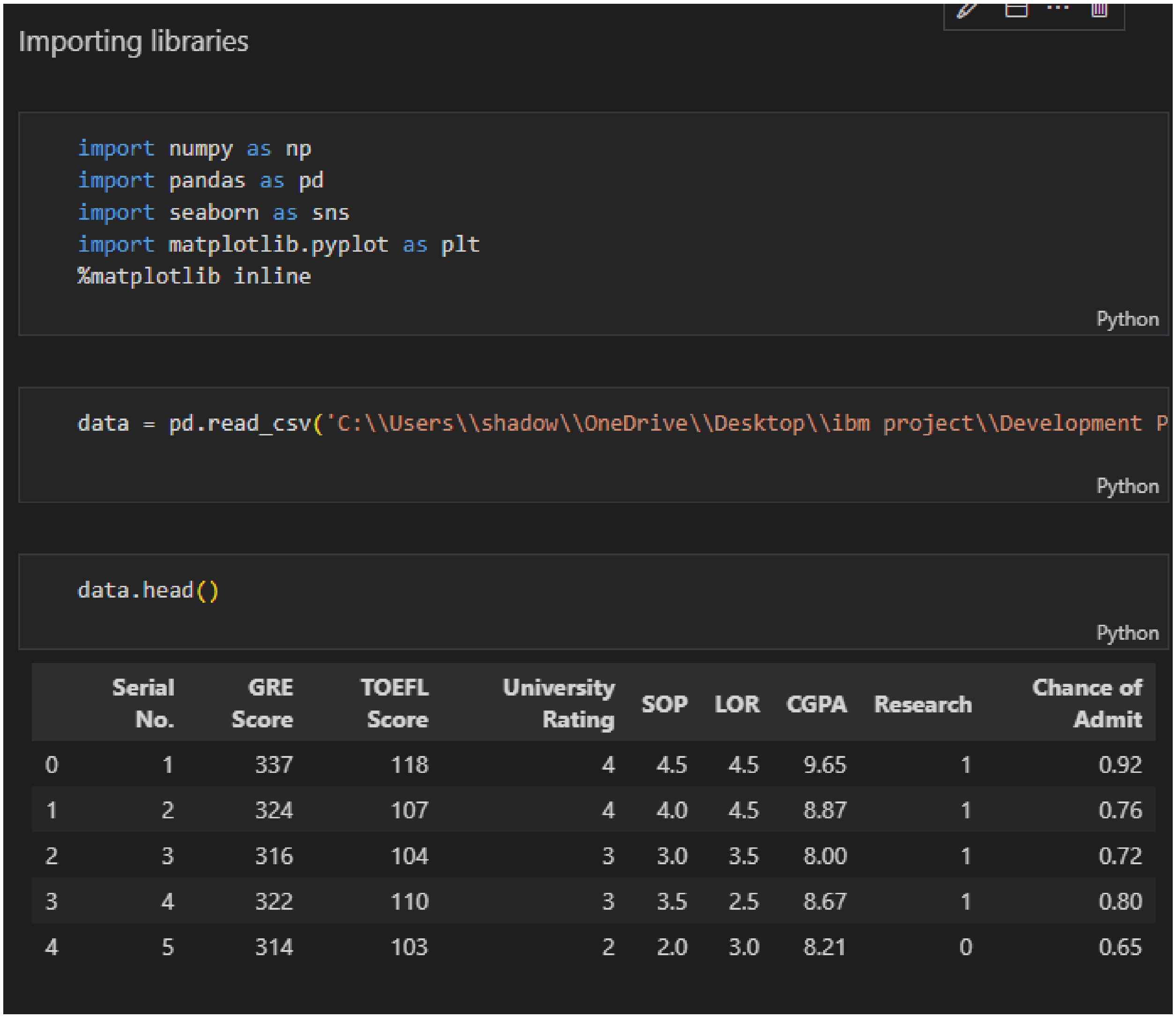
Test case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9 RESULT ANALYSIS

9.1 Perfomance metrics



Importing libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

Python

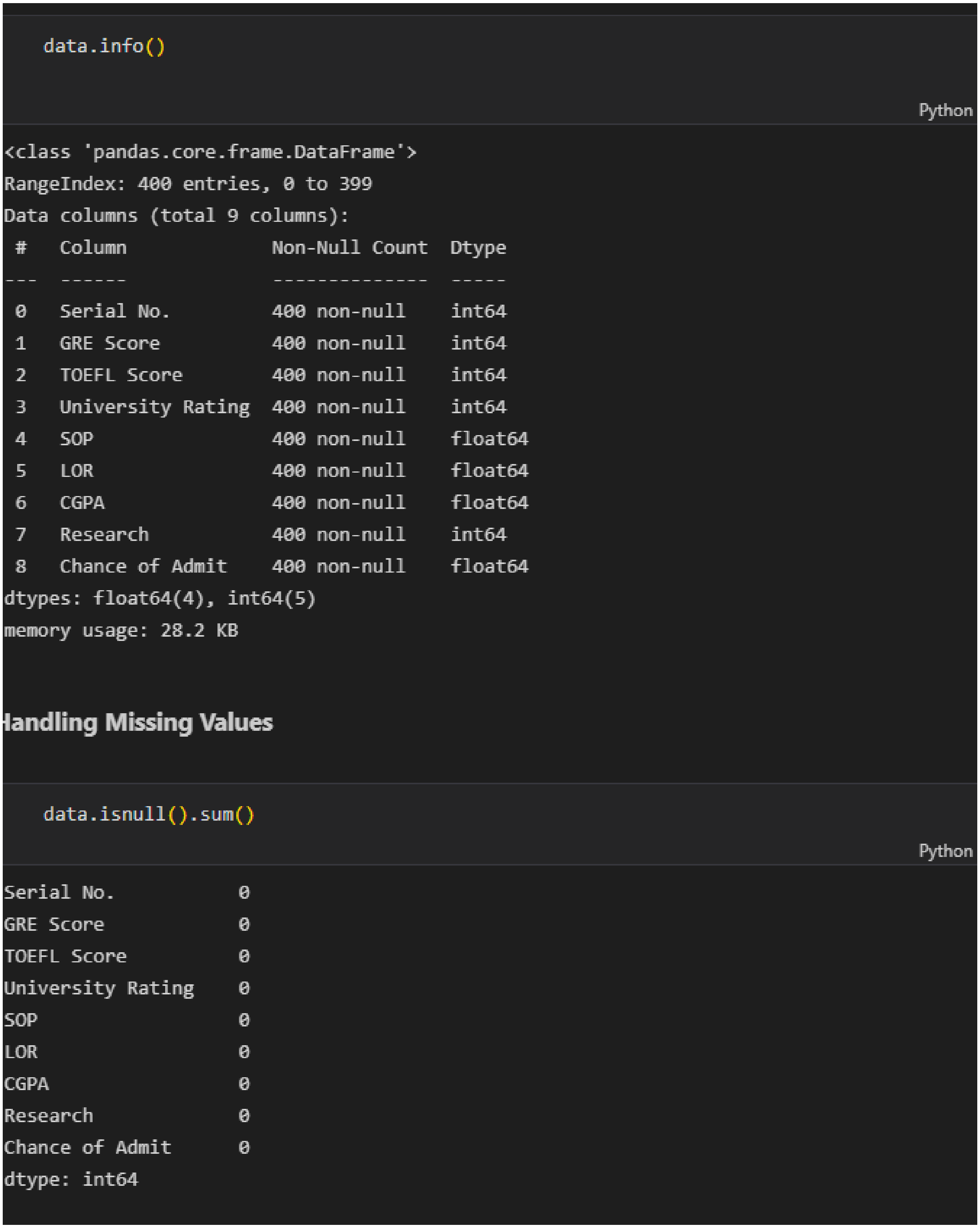
```
data = pd.read_csv('C:\\Users\\shadow\\OneDrive\\Desktop\\ibm project\\Development P
```

Python

```
data.head()
```

Python

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65



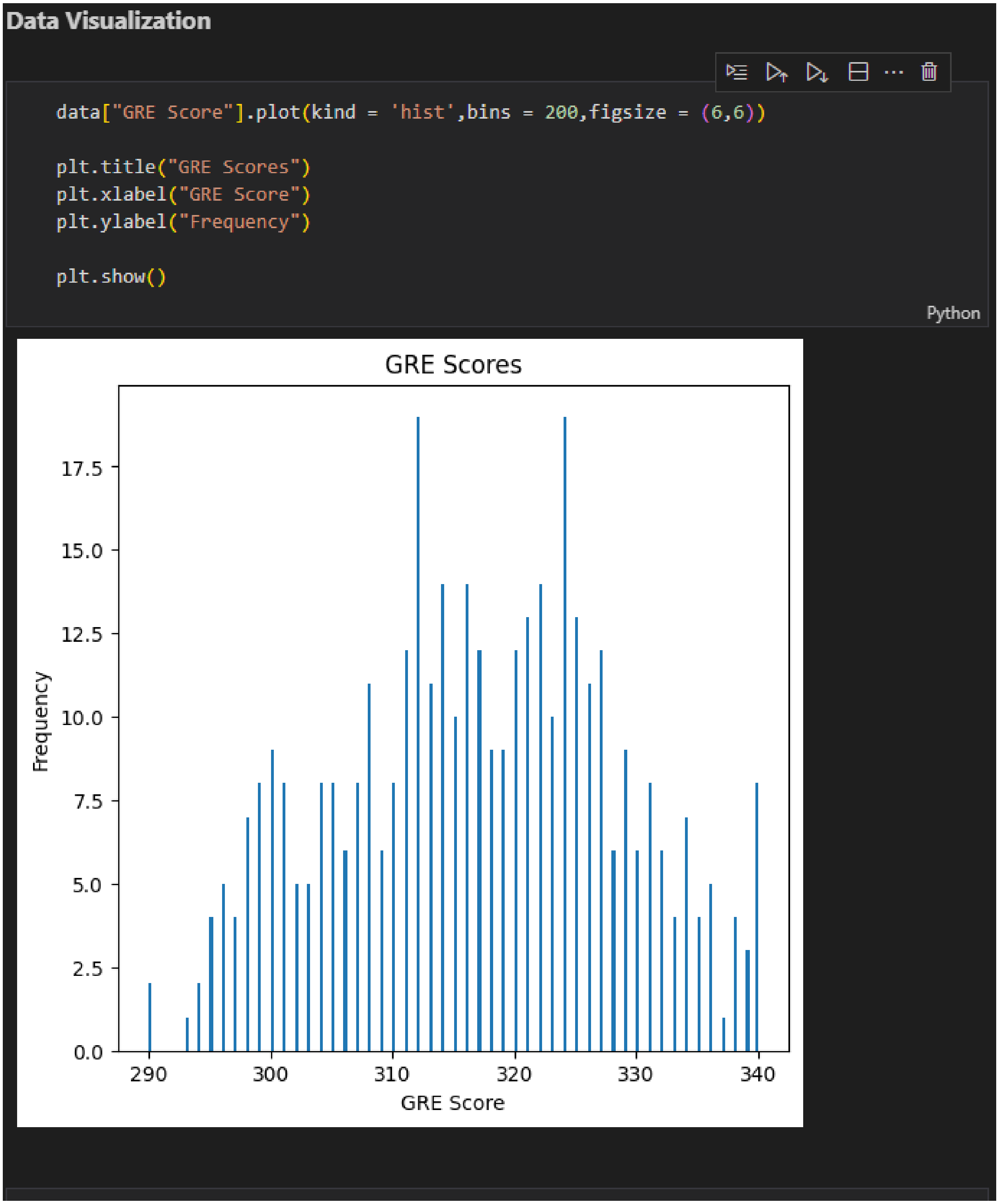
```python
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 400 entries, 0 to 399
Data columns (total 9 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Serial No.         400 non-null    int64  
 1   GRE Score          400 non-null    int64  
 2   TOEFL Score        400 non-null    int64  
 3   University Rating  400 non-null    int64  
 4   SOP                400 non-null    float64
 5   LOR                400 non-null    float64
 6   CGPA               400 non-null    float64
 7   Research           400 non-null    int64  
 8   Chance of Admit    400 non-null    float64
dtypes: float64(4), int64(5)
memory usage: 28.2 KB
```

Handling Missing Values

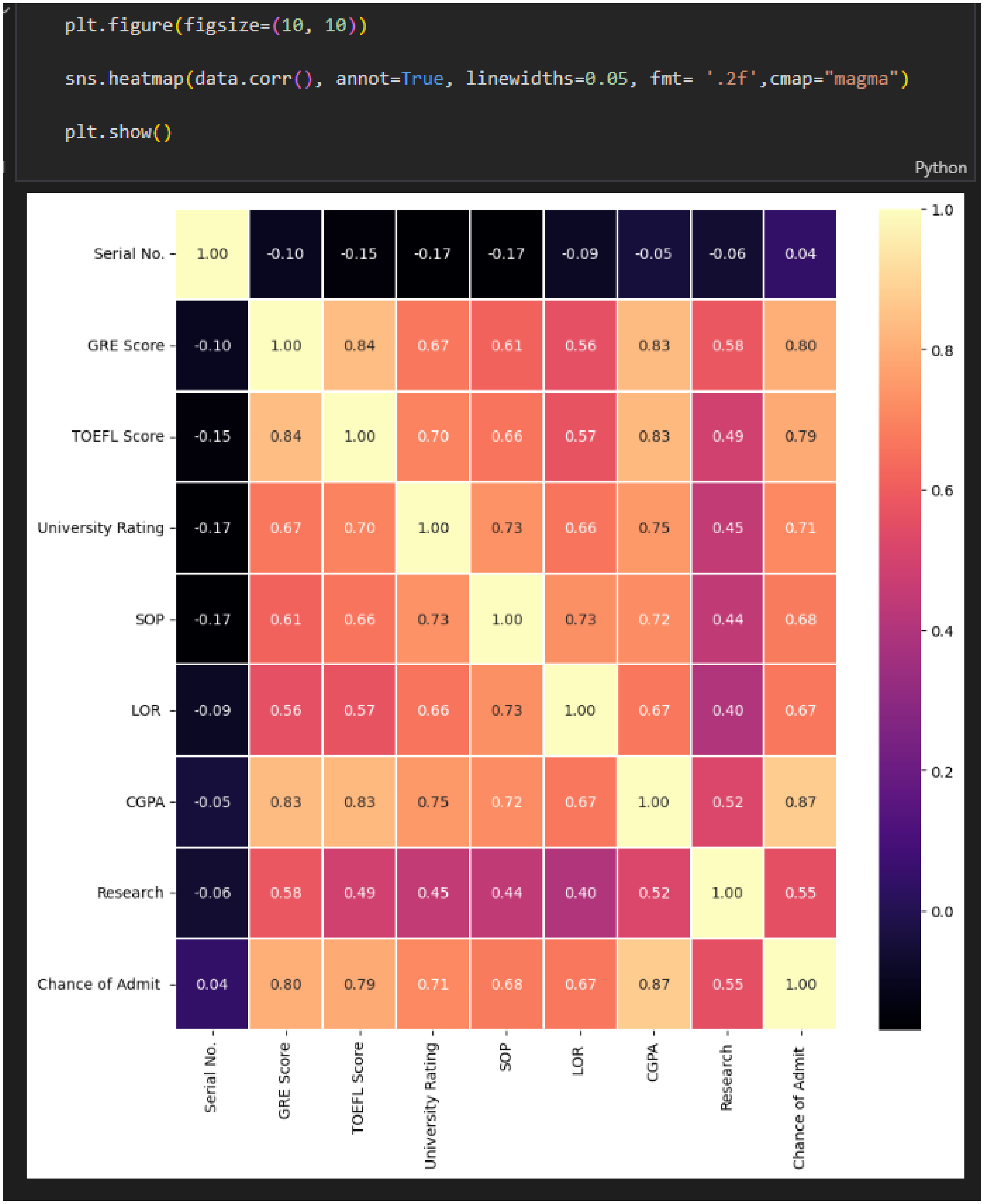
```python
data.isnull().sum()
```

```
Serial No.           0
GRE Score            0
TOEFL Score          0
University Rating    0
SOP                  0
LOR                  0
CGPA                 0
Research             0
Chance of Admit      0
dtype: int64
```



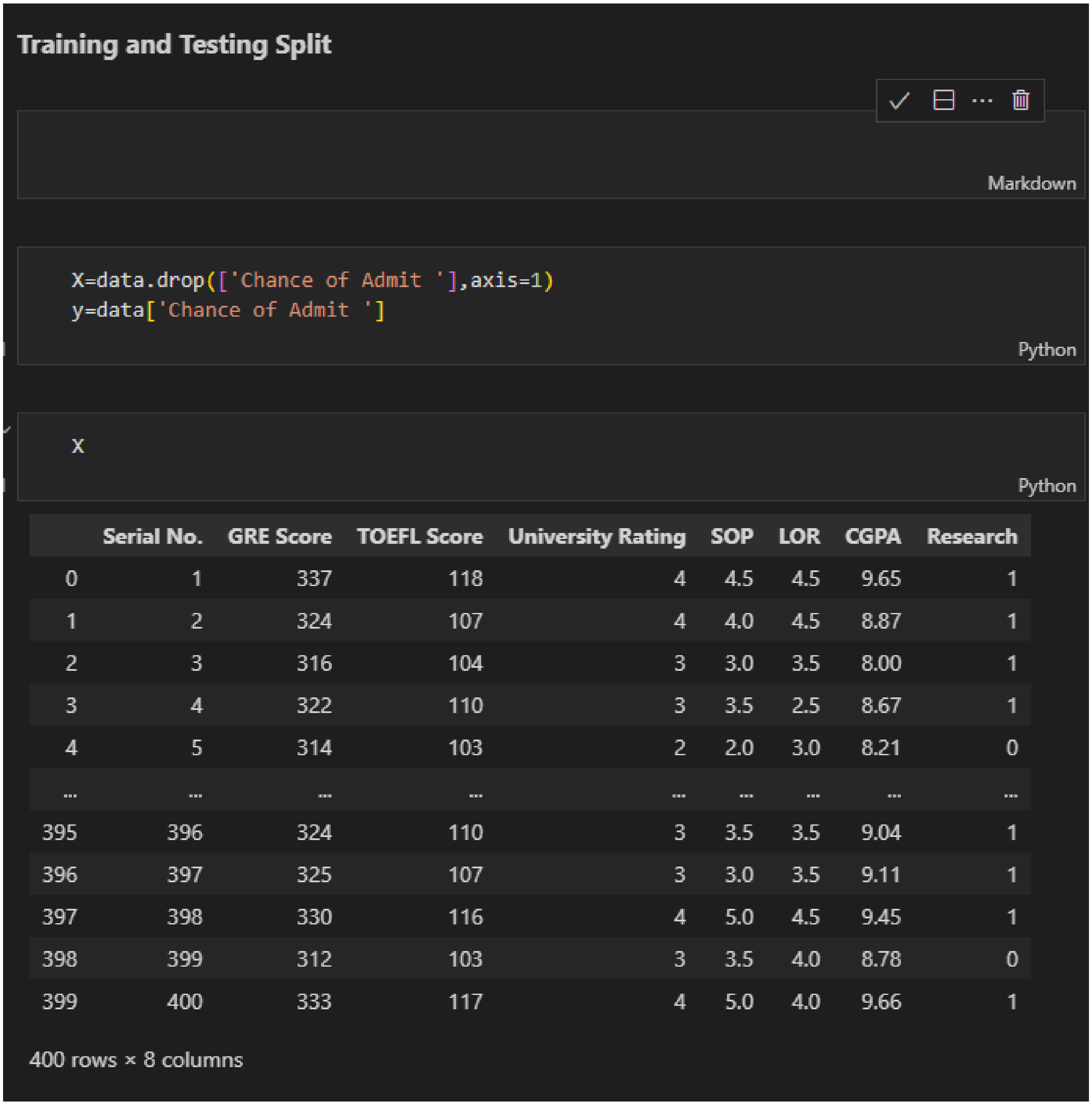
Data Visualization

```python
data["GRE Score"].plot(kind = 'hist',bins = 200,figsize = (6,6))

plt.title("GRE Scores")
plt.xlabel("GRE Score")
plt.ylabel("Frequency")

plt.show()
```



```python
plt.figure(figsize=(10, 10))

sns.heatmap(data.corr(), annot=True, linewidths=0.05, fmt= '.2f',cmap="magma")

plt.show()
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
Serial No.	1.00	-0.10	-0.15	-0.17	-0.17	-0.09	-0.05	-0.06	0.04
GRE Score	-0.10	1.00	0.84	0.67	0.61	0.56	0.83	0.58	0.80
TOEFL Score	-0.15	0.84	1.00	0.70	0.66	0.57	0.83	0.49	0.79
University Rating	-0.17	0.67	0.70	1.00	0.73	0.66	0.75	0.45	0.71
SOP	-0.17	0.61	0.66	0.73	1.00	0.73	0.72	0.44	0.68
LOR	-0.09	0.56	0.57	0.66	0.73	1.00	0.67	0.40	0.67
CGPA	-0.05	0.83	0.83	0.75	0.72	0.67	1.00	0.52	0.87
Research	-0.06	0.58	0.49	0.45	0.44	0.40	0.52	1.00	0.55
Chance of Admit	0.04	0.80	0.79	0.71	0.68	0.67	0.87	0.55	1.00



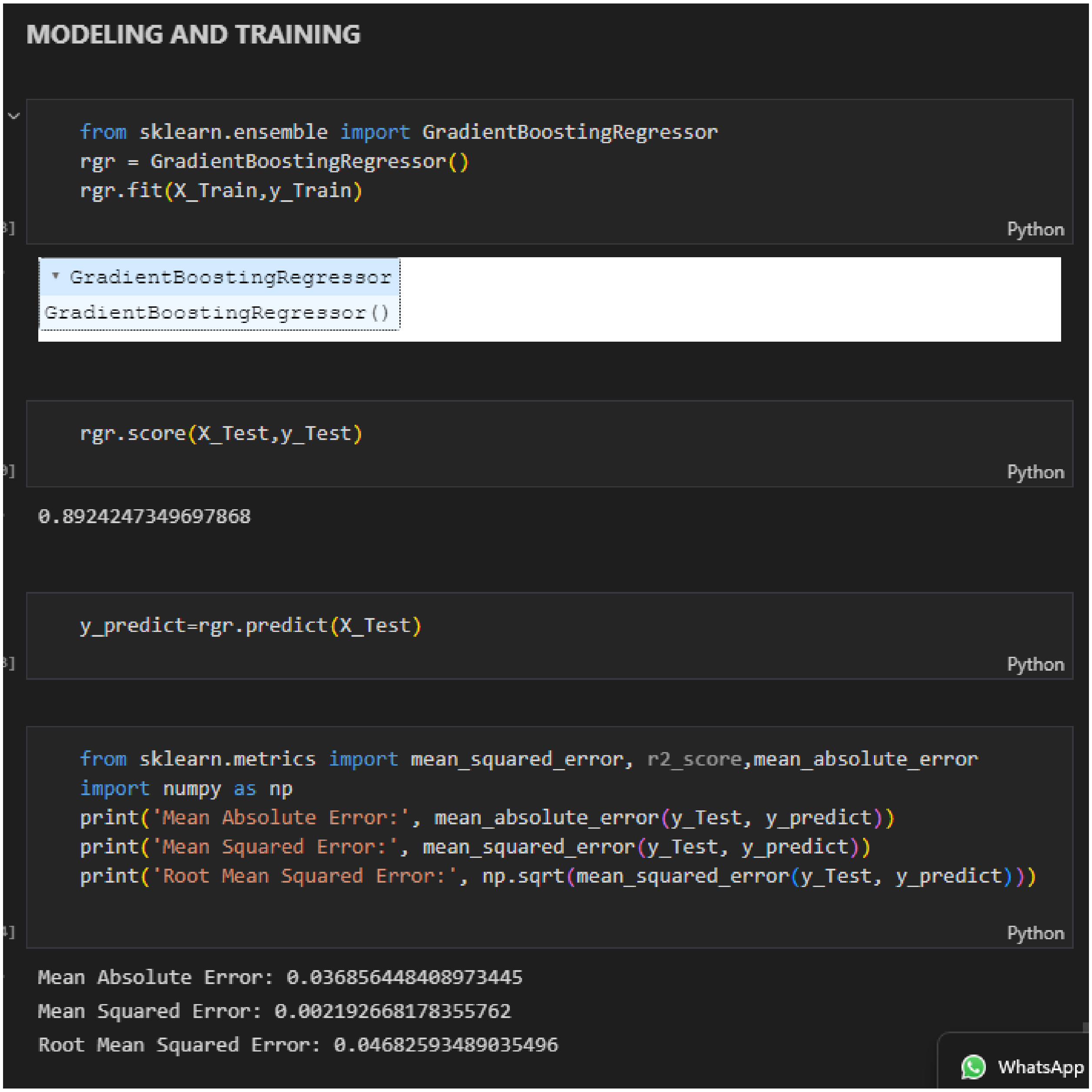
Training and Testing Split

```python
X=data.drop(['Chance of Admit '],axis=1)
y=data['Chance of Admit ']
```

```python
X
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research
0	1	337	118	4	4.5	4.5	9.65	1
1	2	324	107	4	4.0	4.5	8.87	1
2	3	316	104	3	3.0	3.5	8.00	1
3	4	322	110	3	3.5	2.5	8.67	1
4	5	314	103	2	2.0	3.0	8.21	0
...	...	...	...	...	...	...	...	...
395	396	324	110	3	3.5	3.5	9.04	1
396	397	325	107	3	3.0	3.5	9.11	1
397	398	330	116	4	5.0	4.5	9.45	1
398	399	312	103	3	3.5	4.0	8.78	0
399	400	333	117	4	5.0	4.0	9.66	1

400 rows × 8 columns



MODELING AND TRAINING

```python
from sklearn.ensemble import GradientBoostingRegressor
rgr = GradientBoostingRegressor()
rgr.fit(X_Train,y_Train)
```

```
▾ GradientBoostingRegressor
GradientBoostingRegressor()
```

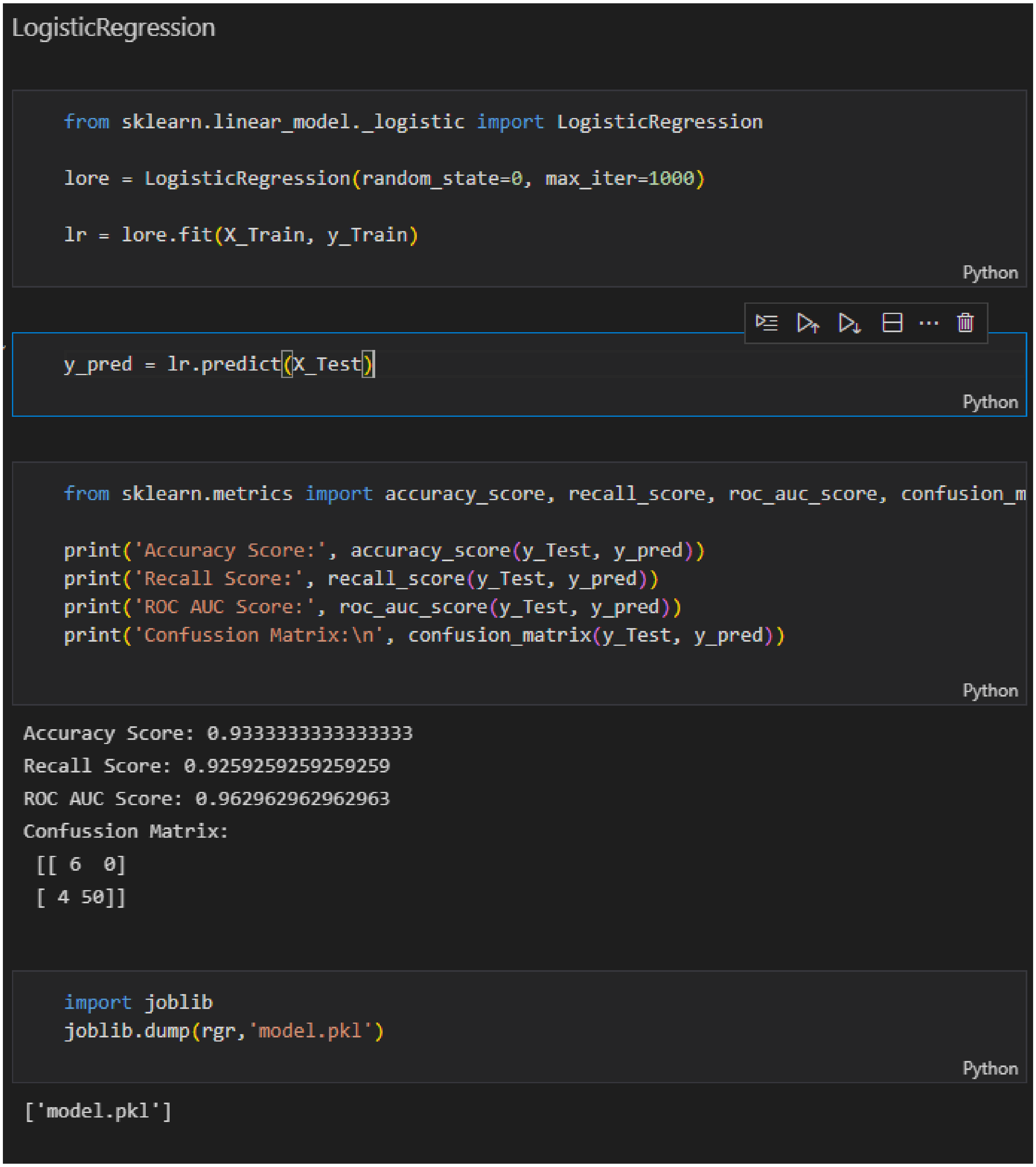
```python
rgr.score(X_Test,y_Test)
```

```
0.8924247349697868
```

```python
y_predict=rgr.predict(X_Test)
```

```python
from sklearn.metrics import mean_squared_error, r2_score,mean_absolute_error
import numpy as np
print('Mean Absolute Error:', mean_absolute_error(y_Test, y_predict))
print('Mean Squared Error:', mean_squared_error(y_Test, y_predict))
print('Root Mean Squared Error:', np.sqrt(mean_squared_error(y_Test, y_predict)))
```

```
Mean Absolute Error: 0.036856448408973445
Mean Squared Error: 0.002192668178355762
Root Mean Squared Error: 0.04682593489035496
```



LogisticRegression

```python
from sklearn.linear_model._logistic import LogisticRegression

lore = LogisticRegression(random_state=0, max_iter=1000)

lr = lore.fit(X_Train, y_Train)
```

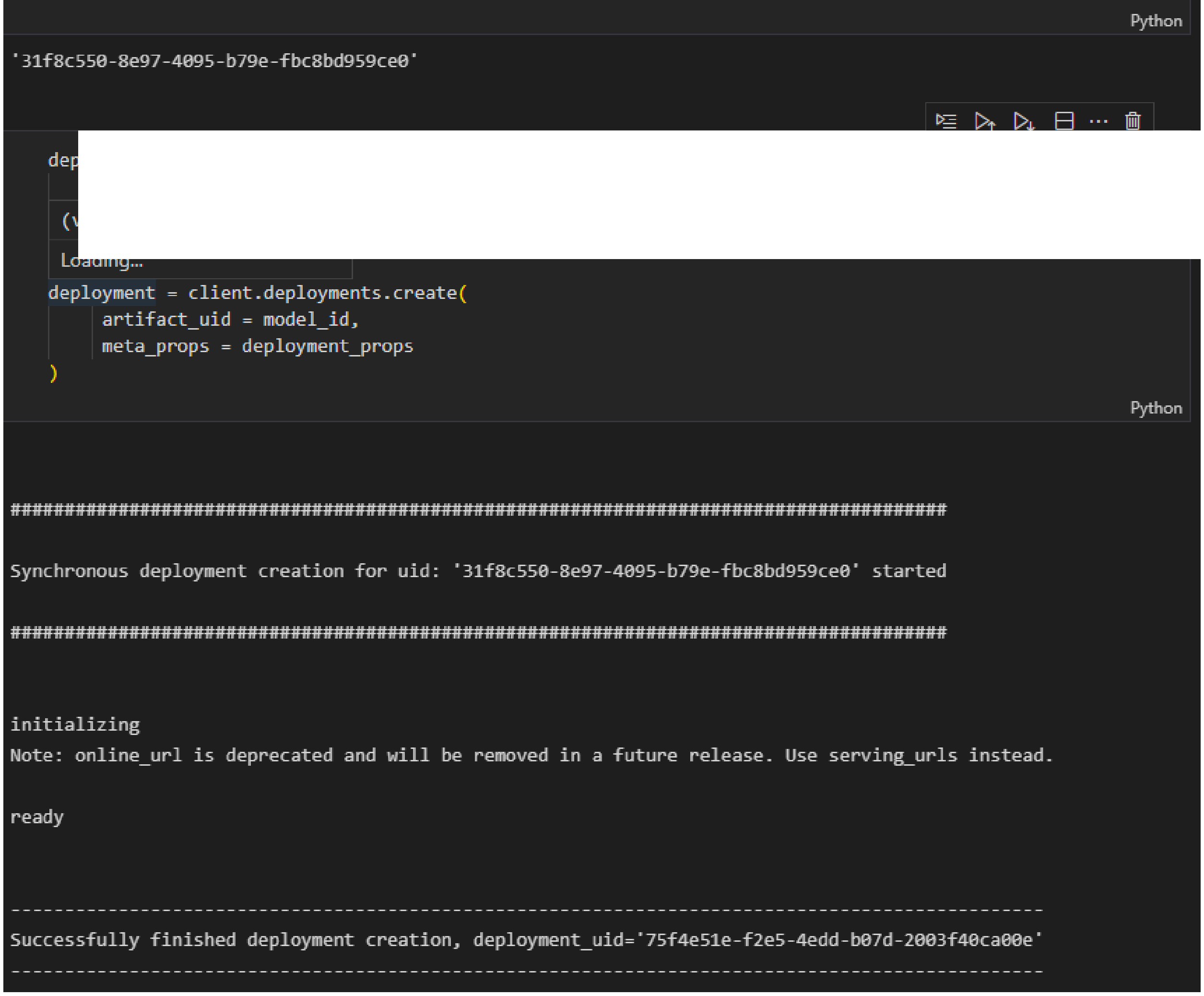
```python
y_pred = lr.predict(X_Test)
```

```python
from sklearn.metrics import accuracy_score, recall_score, roc_auc_score, confusion_m

print('Accuracy Score:', accuracy_score(y_Test, y_pred))
print('Recall Score:', recall_score(y_Test, y_pred))
print('ROC AUC Score:', roc_auc_score(y_Test, y_pred))
print('Confussion Matrix:\n', confusion_matrix(y_Test, y_pred))
```

```
Accuracy Score: 0.9333333333333333
Recall Score: 0.9259259259259259
ROC AUC Score: 0.962962962962963
Confussion Matrix:
 [[ 6  0]
 [ 4 50]]
```

```python
import joblib
joblib.dump(rgr,'model.pkl')
```

```
['model.pkl']
```



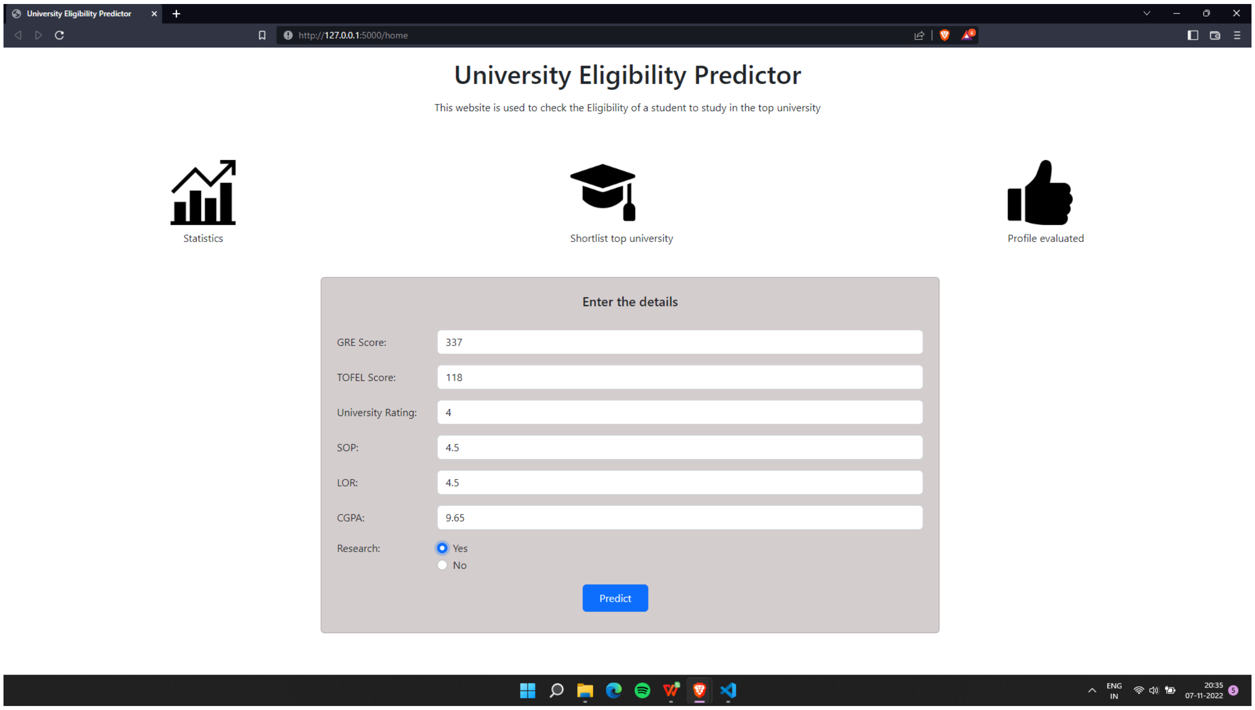
```
'31f8c550-8e97-4095-b79e-fbc8bd959ce0'
```

```
dep
(v
Loading...
deployment = client.deployments.create(
    artifact_uid = model_id,
    meta_props = deployment_props
)
```

```
#######################################################################################

Synchronous deployment creation for uid: '31f8c550-8e97-4095-b79e-fbc8bd959ce0' started

#######################################################################################


initializing
Note: online_url is deprecated and will be removed in a future release. Use serving_urls instead.

ready


------------------------------------------------------------------------------------------------
Successfully finished deployment creation, deployment_uid='75f4e51e-f2e5-4edd-b07d-2003f40ca00e'
------------------------------------------------------------------------------------------------
```

OUTPUT IMAGES

1 Chance Input



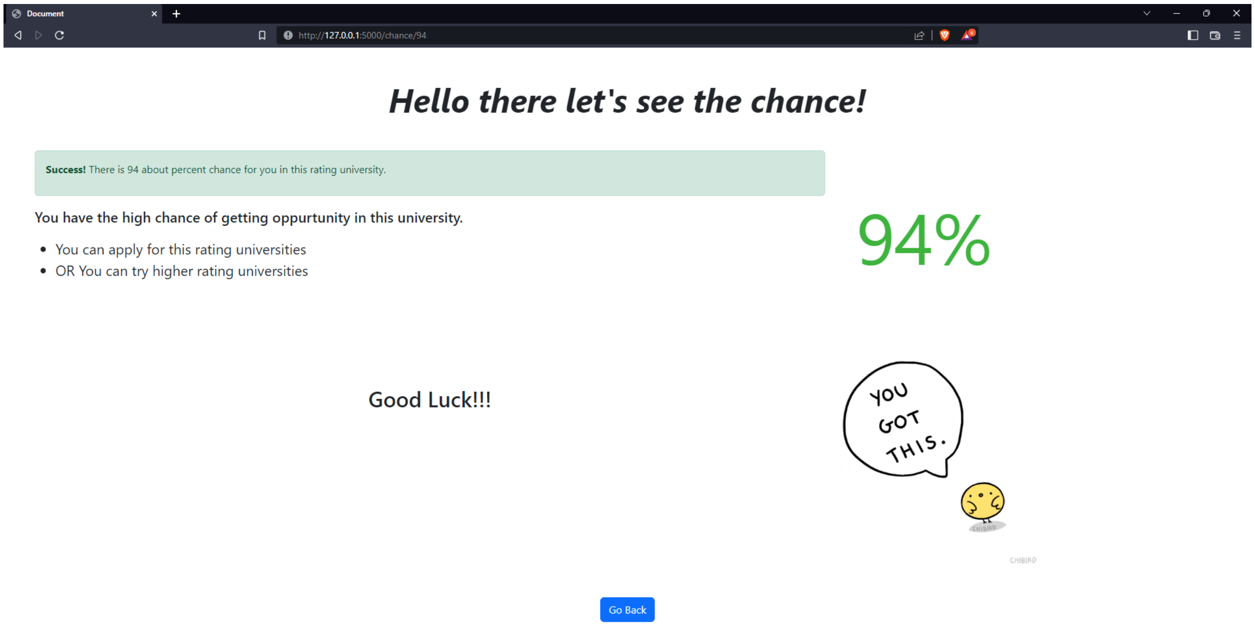
University Eligibility Predictor

This website is used to check the Eligibility of a student to study in the top university

Enter the details

GRE Score:	337
TOFEL Score:	118
University Rating:	4
SOP:	4.5
LOR:	4.5
CGPA:	9.65
Research:	● Yes ○ No

2.Chance Output



Hello there let's see the chance!

Success! There is 94 about percent chance for you in this rating university.

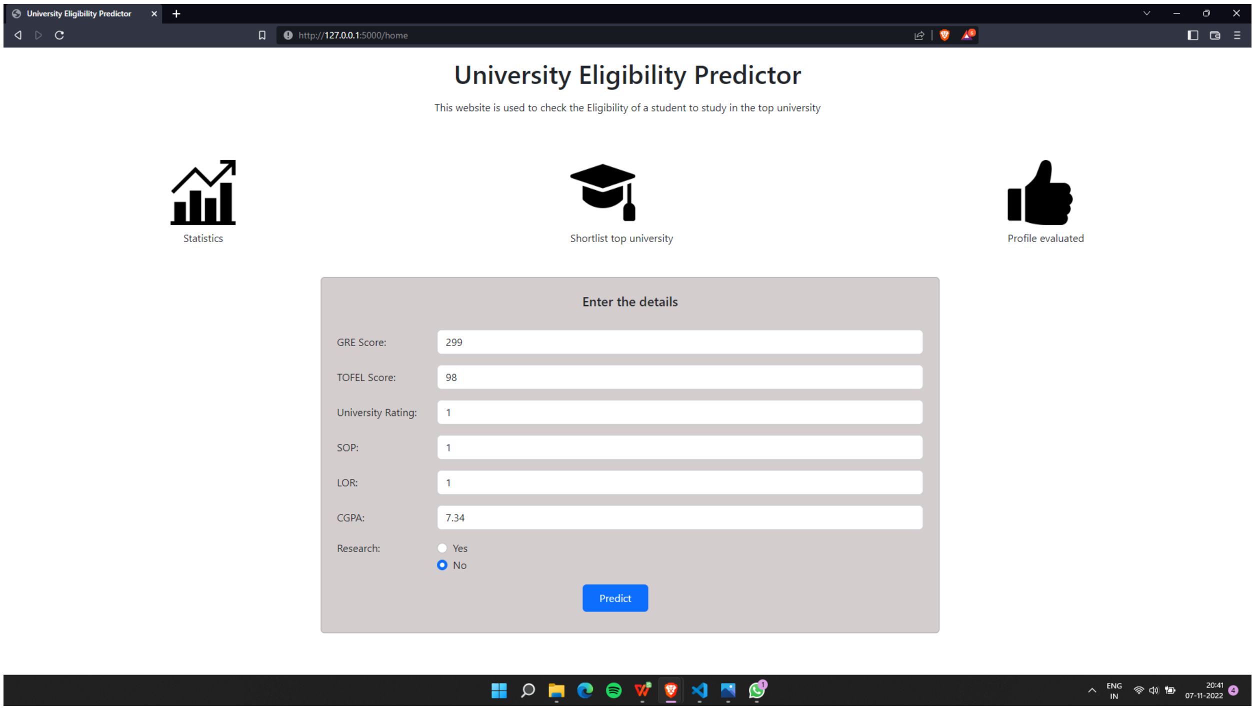
You have the high chance of getting oppurtunity in this university.

- You can apply for this rating universities
- OR You can try higher rating universities

94%

Good Luck!!!

3.NoChance Input



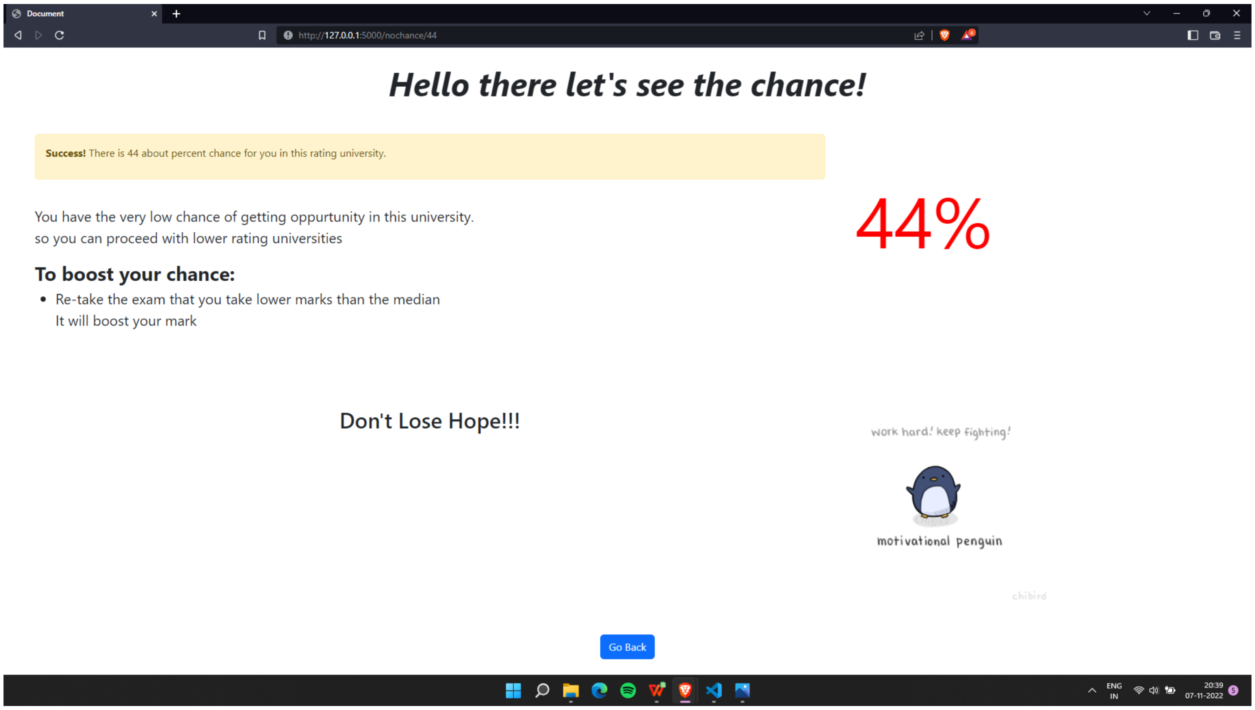
University Eligibility Predictor

This website is used to check the Eligibility of a student to study in the top university

Enter the details

GRE Score:	299
TOFEL Score:	98
University Rating:	1
SOP:	1
LOR:	1
CGPA:	7.34
Research:	○ Yes ● No

4.NoChance Output



Hello there let's see the chance!

Success! There is 44 about percent chance for you in this rating university.

44%

You have the very low chance of getting oppurtunity in this university.
so you can proceed with lower rating universities

To boost your chance:

- Re-take the exam that you take lower marks than the median
  It will boost your mark

Don't Lose Hope!!!

10. ADVANTAGES & DISADVANTAGES

a.Advantages

1. It helps student for making decision for choosing a right college.
2. Here the chance of occurrence of error is less when compared with the existing system.
3. It is fast, efficient and reliable.
4. Avoids data redundancy and inconsistency.
5. Very user-friendly.
6. Easy accessibility of data.

b.Dis-Advantages

1. Required active internet connection.
2. System will provide inaccurate results if data entered incorrectly.

11. CONCLUSION

This system ,being the first we have created in Python using ML algorithms and other front endlanguages such as html, css, java script , 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 wereadded, ideas for additional features or methods to improve the usability of the system madethemselves apparent. Furthermore, adding one feature meant that another required feature was now possible, and balancing completing these required features with the ideas for improvementas well as remembering everything that had to be done was a project in itself. Debugging cansometimes be a relatively straight forward process, or rather rather finding out what you mustdebug can be. Since so many parts of the admissions system are integrated into one another, if anerror occurs on one page, it may be a display error, for example; 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 processand can be frustrating if the apparent cause of a problem is not obvious at first. Language usedmust be simple and easy to understand and compatibility is paramount. If this system were not designed as an entirely web based application, it would not have been possible to recreate its current state of portability. Overall, the system performs well,and while it does not include all ofthe features that may have been desired, it lives up to initial expectations. The majority of features that are included work flawlessly and the errors that do exist are minor or graphical.

12. FUTURE SCOPE

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

-

1. This can be accessed anytime anywhere, since it is a web application provided only aninternet connection.
2. The user had not need to travel a long distance for the admission and his/her time isalso saved as a result of this automated system.

13.1 SOURCE CODE:

HTML CODES

Uploading Html codes

```
{% block code%}
<body>
    <h1 class="text-center mt-3">University Eligibility Predictor</h1>


    <p  class="text-center mt-3 font-weight-normal" >This website is used to check the
Eligibility of a student to study in the top university</p>
    <div class="heading">
        <div class="ml-1 p-4" >
            <img src="/static/trend.png" alt="like">
            <p class="mt-2 text-center">Statistics</p>
        </div>
        <div class="ml-1  p-4">
            <img  src="/static/mortarboard.png" alt="university">
            <p class="mt-2 text-center">Shortlist top university</p>
        </div>
           <div class="ml-2 p-4">
            <img src="/static/like.png" alt="like">
            <p class="mt-2 text-center">Profile evaluated</p>
        </div>
    </div>
<div class="col-6 " id="main" >
        <div class="card p-2 ms-2 my-2" style="background-color: rgb(212, 205, 205);">
            <div class="card-body" >
        <h5 class="card-title pb-4 text-center">
          Enter the details
        </h5>
        <form action="/" method="post"  style="background-color:rgb(212, 205, 205) ;"
id="theForm">
          <div class="row mb-3">
            <label for="gre" class="col-lg-2 col-form-label">GRE Score:</label>
                <div class="col-lg-10">
              <input type="number" class="form-control" id="gre" name="gre"
min="250" max="340" placeholder="250 to 340" required>
            </div>
 </div>
          <div class="row mb-3">
            <label for="tofel" class="col-lg-2 col-form-label">TOFEL Score:</label>
            <div class="col-lg-10">
              <input type="number" class="form-control" id="tofel" name="tofel"
min="50" max="120" placeholder="50 to 120" required>
            </div>
          </div>
          <div class="row mb-3">
            <label for="university_rating" class="col-lg-2 col-form-label">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" placeholder="1 to 5" required>
            </div>
          </div>
          <div class="row mb-3">
            <label for="sop" class="col-lg-2 col-form-label">SOP:</label>
            <div class="col-lg-10">
              <input type="number" class="form-control" id="sop" name="sop"
step="0.01" min="1" max="5" placeholder="1 to 5" required>
            </div>
          </div>
          <div class="row mb-3">
                <label for="lor" class="col-lg-2 col-form-label">LOR:</label>
            <div class="col-lg-10">
              <input type="number" class="form-control" id="lor" name="lor"
step="0.01" min="1" max="5" placeholder="1 to 5" required>
                </div>
          </div>
          <div class="row mb-3">
            <label for="cgpa" class="col-lg-2 col-form-label">CGPA:</label>
            <div class="col-lg-10">
              <input type="number" class="form-control" id="cgpa" name="cgpa"
step="0.01" min="5" max="10" placeholder="5 to 10" required>
            </div>
          </div>
          <fieldset class="row mb-3">
            <legend class="col-form-label col-sm-2 pt-0">Research:</legend>
                <div class="col-sm-10">
              <div class="form-check">
                <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>
              </div>
              <div class="form-check">
                <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>
              </div>
                </div>
          </fieldset>

          <div class="row lg-3 justify-content-center">
            <div class="col-lg-2 mb-2 me-3">
              <button type="submit" class="btn btn-primary  m-auto" style="padding:8px
25px; "  id="button">Predict</button>
            </div>

        </form>
        </div>
      </div>
```

STYLE.CSS

```
.heading{
    display:flex;
     justify-content: space-around;
      margin-top: 45px;
}
#title{
    margin-top:50px;
    font-style: italic;
    font-weight:400;
    font-size: 50px;
}
#image{
    height: 335px;
    width: 335px;
}
img{
    width:100px;
    height: 100px;
}

#main{
    margin:auto;
}

#mark1
{
    font-size:110px;
    color: rgb(62, 180, 62)
    }
     </div>


    </body>


    {%end block%}

#mark2{
    font-size:110px;
    color: red;
}
```

PYTHON CODE

```
from flask import Flask, render_template, redirect, url_for, request
import requests

app = Flask(__name__)
@app.route("/", methods = ['POST', 'GET'])
def index():
    if request.method == 'POST':
        arr = []
        for i in request.form:
            val = request.form[i]
            if val == ":
                return redirect(url_for("index"))
            arr.append(float(val))
        # NOTE: you must manually set API_KEY below using information retrieved from your IBM
Cloud account.
        API_KEY = "kLmQAEgx0aXP45J2FG060MUCY8Iqe5Uj1rI_Zzj78db-"
        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}

        payload_scoring = {
            "input_data": [{"fields":[  'GRE Score',
                                        'TOEFL Score',
                                        'University Rating',
                                        'SOP',
                                        'LOR ',
                                        'CGPA',
                                        'Research'],
                            "values": [arr]
                            }]
                        }

        response_scoring = requests.post(
            'https://us-south.ml.cloud.ibm.com/ml/v4/deployments/75f4e51e-f2e5-4edd-b07d-
2003f40ca00e/predictions?version=2022-11-05',
            json=payload_scoring,
            headers=header
        ).json()

        result = response_scoring['predictions'][0]['values']

        if result[0][0] > 0.5:
            return redirect(url_for('chance', percent=result[0][0]*100))
        else:
            return redirect(url_for('no_chance', percent=result[0][0]*100))
    else:
        return redirect(url_for("demo"))
@app.route("/home")
def demo():
    return render_template("index.html")

@app.route("/chance/<percent>")
def chance(percent):
    return render_template("chance.html", content=[percent])

@app.route("/nochance/<percent>")
def no_chance(percent):
    return render_template("noChance.html", content=[percent])

@app.route('/<path:path>')
```

```
def catch_all():
    return redirect(url_for("index"))

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

13.2 GITHUB LINK:

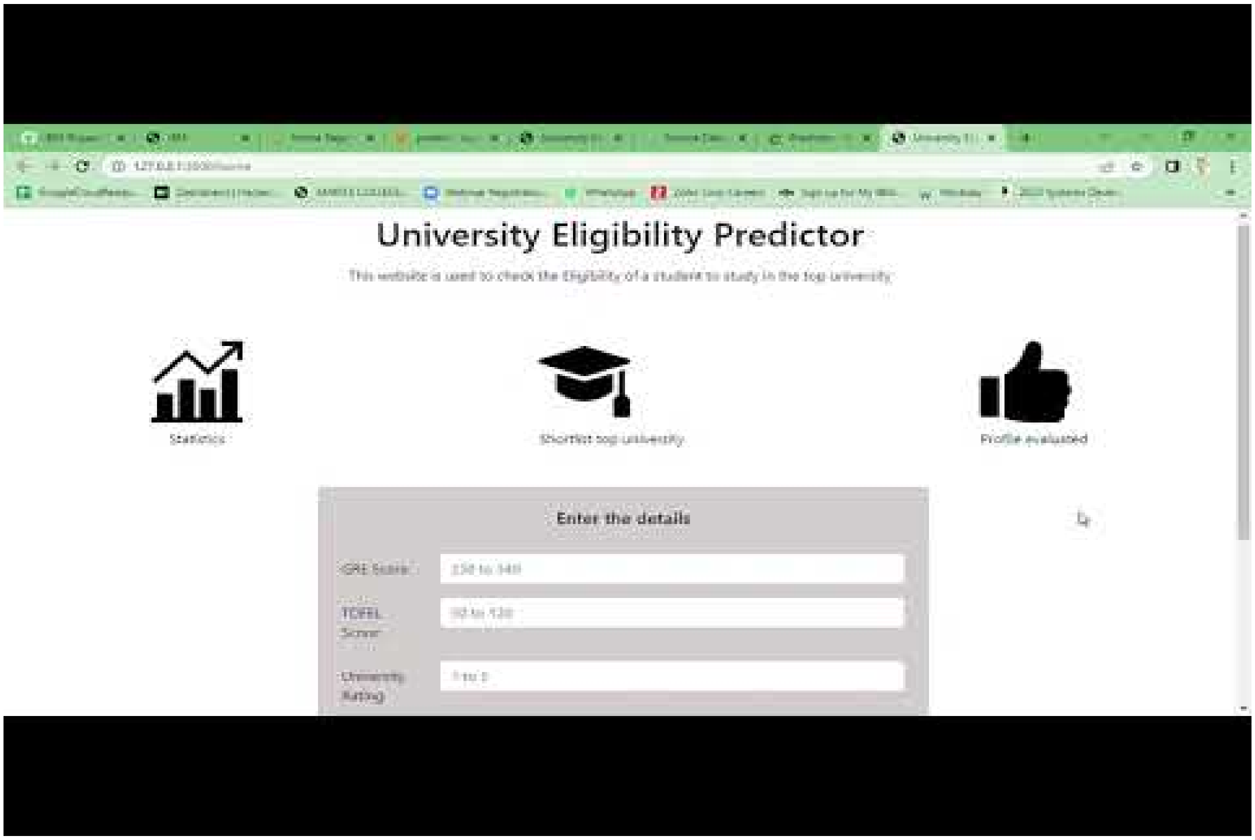
https://github.com/IBM-EPBL/IBM-Project-39153-1660398138

13.3 . DEMO LINK:

Drive link

https://drive.google.com/file/d/1xNA8yJ_3og73dS2QGy1BybneAh9yevy_/view

Youtube link



University Eligibility Predictor

This website is used to check the Eligibility of a student to study in the top university

Enter the details

GRE Score : 230 to 340

TOEFL Score : 90 to 120

University Rating : 1 to 5