# UNIVERSITY ADMIT ELIGIBILITY PREDICTION

## A MINI-PROJECT REPORT

TEAM ID:PNT2022TMID07958

***Submitted by***

| | |
|---|---|
| **ARCHANA JANANI S** | **(AC19UCS007)** |
| **DHARANESH RAJA S** | **(AC19UCS025)** |
| **JEEVA GANESAN T** | **(AC19UCS041)** |
| **KAMALI A S** | **(AC19UCS047)** |

***in partial fulfillment of the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**In**

**COMPUTER SCIENCE AND ENGINEERING**

**ADHIYAMAAN COLLEGE OF ENGINEERING(AUTONOMOUS)**

**DR.M.G.R NAGAR, HOSUR-635130**

**ANNA UNIVERSITY: CHENNAI 600025**

**NOVEMBER 2022**

# BONAFIDE CERTIFICATE

Certified that this mini project report **“**UNIVERSITY ADMIT ELIGIBILITY PREDICTION**”** is the bonafide work of **“ARCHANA JANANI S (AC19UCS007), DHARANESH RAJA S (AC19UCS025)** , **JEEVA GANESAN T(AC19UCS041)**, **KAMALI A S(AC19UCS047)”** who carried out the project under my supervision.

**SIGNATURE**

**Dr.G. FATHIMA, M.E., Ph.D.,**<br>
**HEAD OF THE DEPARTMENT**

Department of CSE,
Adhiyamaan College of Engineering,
(Autonomous)
Dr. M.G.R. Nagar,
Hosur – 635 130.

**SIGNATURE**

**Mrs. R.L. INDU LEKHA**

**ASSISTANT PROFESSOR/CSE**

Department of CSE,
Adhiyamaan College of Engineering,
(Autonomous)
Dr. M.G.R. Nagar,
Hosur – 635 130.

Submitted for the Mini project VIVA-VOCE Examination held on ________________ at **Adhiyamaan College of Engineering (Autonomous), Hosur-635 130.**

**INTERNAL EXAMINER** **EXTERNAL EXAMINER**

# ACKNOWLEDGEMENT

It is one of the most efficient tasks in life to choose the appropriate words to express one's gratitude to the beneficiaries. We are very much grateful to God who helped us all the way through the project and how moulded us into what we are today.

We are grateful to our beloved Principal **Dr. G. RANGANATH, M.E., Ph.D.,** Adhiyamaan College of Engineering (Autonomous), Hosur for providing the opportunity to do this work in premises.

We acknowledge our heartful gratitude to **Dr. G. FATHIMA, M.E., Ph.D.,** Professor and Head of the Department, Department of Computer Science and Engineering, Adhiyamaan College of Engineering (Autonomous), Hosur, for her guidance and valuable suggestions and encouragement throughout this project.

We are highly indebted to Professor, **Mrs. R.L. INDU LEKHA ,** Department of Computer Science and Engineering, Adhiyamaan College of Engineering (Autonomous), Hosur, whose immense support, encouragement and valuable guidance made us to complete this project successfully.

We also extent our thanks to Project Coordinator and all Staff Members for their support in complete this project successfully.

Finally, we would like to thank to our parents, without their motivational and support would not have been possible for us to complete this project successfully.

# UNIVERSITY ADMIT ELIGIBILITY PREDICTION

## CONTENTS

# CHAPTER 1

# INTRODUCTION

## ABSTRACT

This project aims to build a model that can help students to pick the right universities based on their profiles. Student admission problem is very important in educational institutions. The machine learning models to predict the chance of a student to be admitted to a master's program. This will assist students to know in advance if they have a chance to get accepted for the accurate predictions. we plan on training a machine learning model in order to provide results. The dataset contains information on the student profile and the university details with a field. The model performing the best is then used to evaluate the dependent variable i.e. The chances of admit to a university. We also aim to create a portal which filters and then provides a list of universities that fall into the profile's acceptance range. The machine learning models are multiple linear regression, k-nearest neighbor and random forest. Experiments show that the multiple linear regression model surpasses other models.

**Keywords:** Machine learning algorithms, Students dataset, models, University ranking.

## PROJECT OVERVIEW

In the present schooling world there are numerous quantities of understudies who need to seek after Higher training in the wake of Engineering or any Graduate certification course. Advanced education in the sense, a few groups need to do MTech through GATE or through any Educational Institute Entrance Examination and a few groups need to do MBA through CAT or through any individual Educational Institute Entrance Examination and a few groups need to do Masters in abroad University. Understudy confirmation issue is vital in Educational Institutions. We are addressing ML models to anticipate the opportunity of an understudy to be conceded to a Master's program. This will help understudies to know ahead of time in the event that they get an opportunity to get acknowledged. The Machine learning models are  Decision tree regressor and Random Forest regressor. Investigations show that the Linear Regression model outperforms different models.

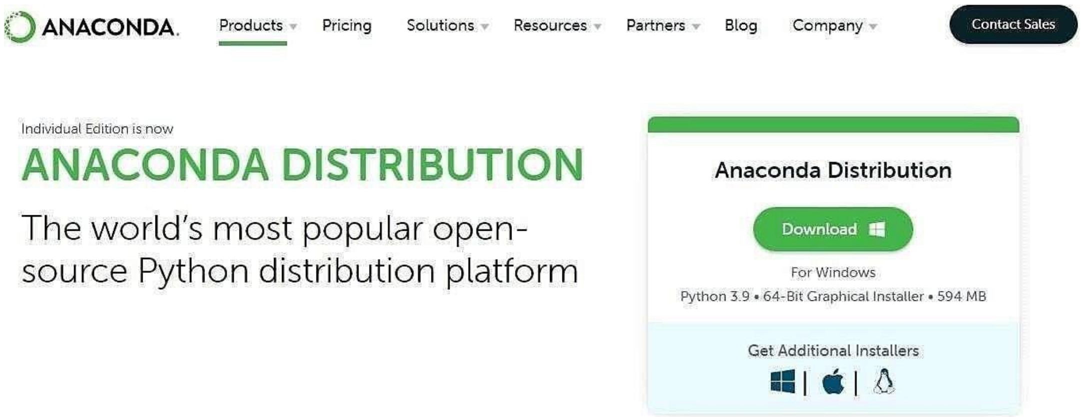
### PRE-REQUISTIES

### 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

**WAY TO INSTALL ANACONDA:**

**STEP 1:** Download and Anaconda



Individual Edition is now

ANACONDA DISTRIBUTION

The world's most popular open-source Python distribution platform

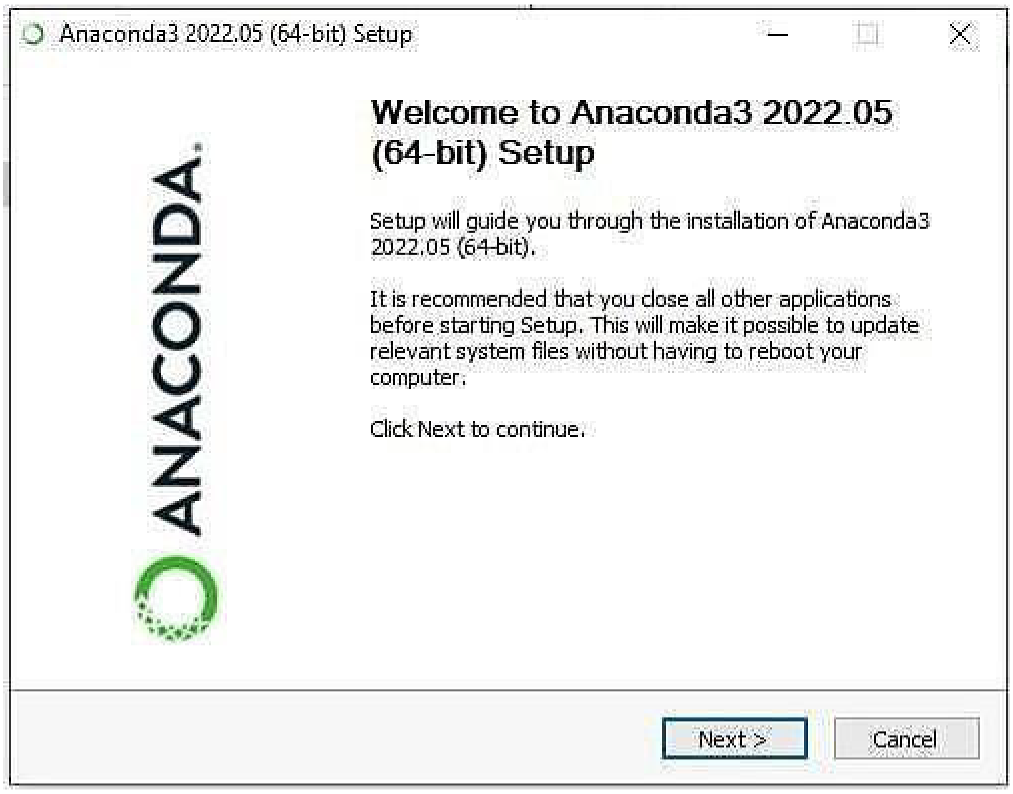
Anaconda Distribution

For Windows

Python 3.9 • 64-Bit Graphical Installer • 594 MB

Get Additional Installers

**STEP 2:** Install the Anaconda



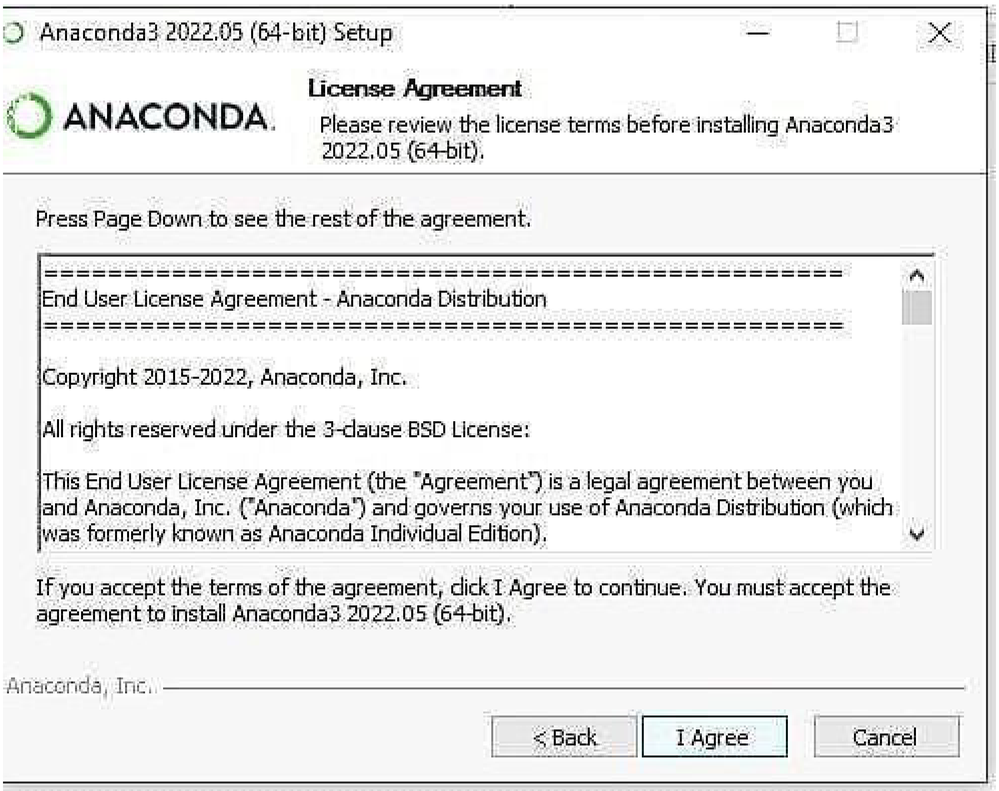
Welcome to Anaconda3 2022.05 (64-bit) Setup

Setup will guide you through the installation of Anaconda3 2022.05 (64-bit).

It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

**STEP 3:** Click I Agree



**License Agreement**

Please review the license terms before installing Anaconda3 2022.05 (64-bit).

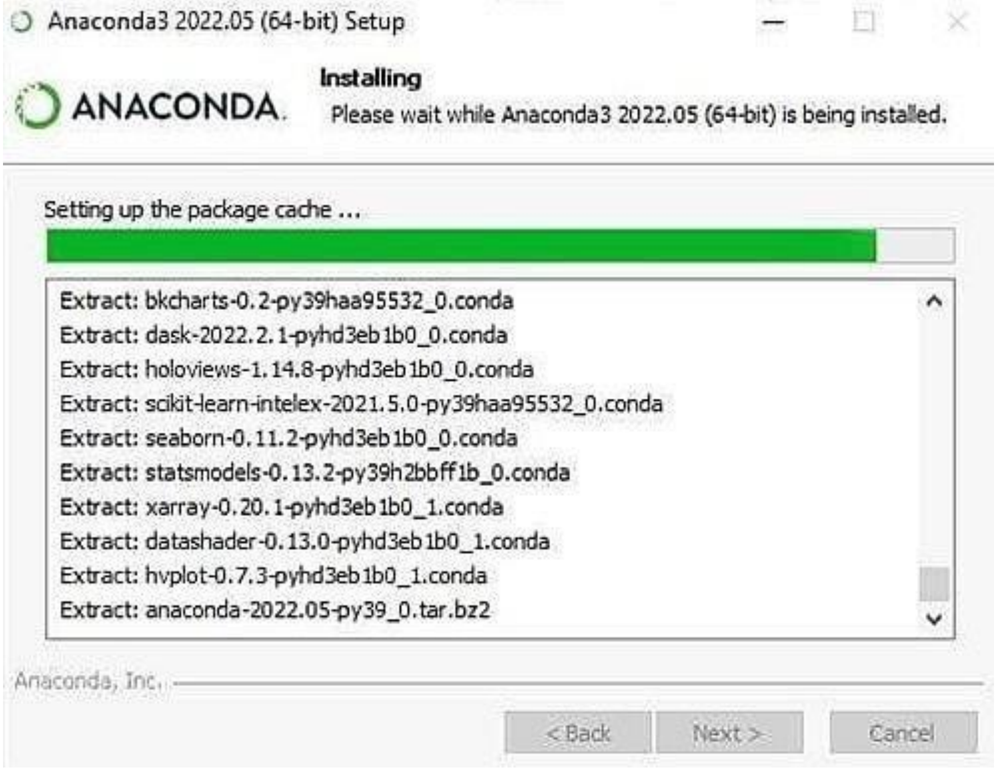
Press Page Down to see the rest of the agreement.

```
===================================================
End User License Agreement - Anaconda Distribution
===================================================

Copyright 2015-2022, Anaconda, Inc.

All rights reserved under the 3-clause BSD License:

This End User License Agreement (the "Agreement") is a legal agreement between you
and Anaconda, Inc. ("Anaconda") and governs your use of Anaconda Distribution (which
was formerly known as Anaconda Individual Edition).
```

If you accept the terms of the agreement, click I Agree to continue. You must accept the agreement to install Anaconda3 2022.05 (64-bit).

Anaconda, Inc.

**STEP 4:** Installing the Requiring packages FRGT



**Installing**

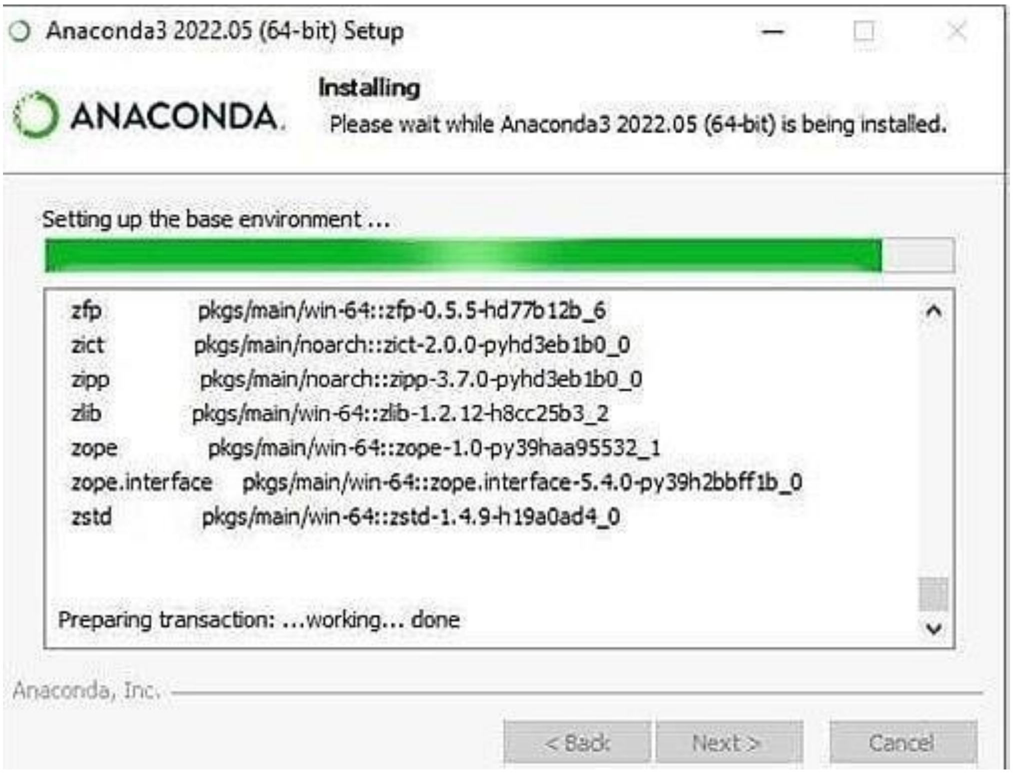
Please wait while Anaconda3 2022.05 (64-bit) is being installed.

Setting up the package cache ...

```
Extract: bkcharts-0.2-py39haa95532_0.conda
Extract: dask-2022.2.1-pyhd3eb1b0_0.conda
Extract: holoviews-1.14.8-pyhd3eb1b0_0.conda
Extract: scikit-learn-intelex-2021.5.0-py39haa95532_0.conda
Extract: seaborn-0.11.2-pyhd3eb1b0_0.conda
Extract: statsmodels-0.13.2-py39h2bbff1b_0.conda
Extract: xarray-0.20.1-pyhd3eb1b0_1.conda
Extract: datashader-0.13.0-pyhd3eb1b0_1.conda
Extract: hvplot-0.7.3-pyhd3eb1b0_1.conda
Extract: anaconda-2022.05-py39_0.tar.bz2
```

Anaconda, Inc.

**STEP 5:**Setting up the base environment

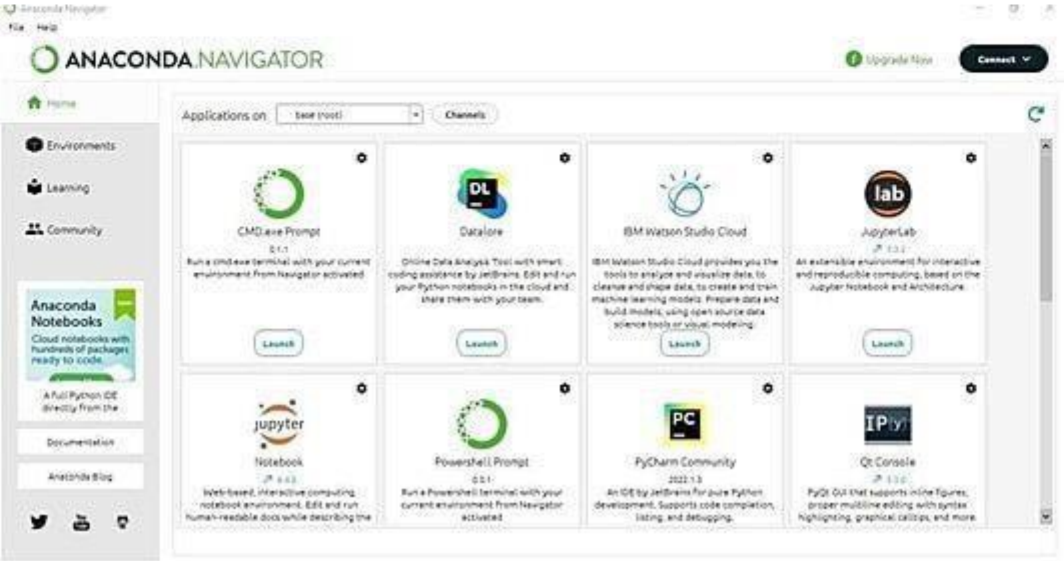


Please wait while Anaconda3 2022.05 (64-bit) is being installed.

Setting up the base environment ...

```
zfp          pkgs/main/win-64::zfp-0.5.5-hd77b12b_6
zict         pkgs/main/noarch::zict-2.0.0-pyhd3eb1b0_0
zipp         pkgs/main/noarch::zipp-3.7.0-pyhd3eb1b0_0
zlib         pkgs/main/win-64::zlib-1.2.12-h8cc25b3_2
zope         pkgs/main/win-64::zope-1.0-py39haa95532_1
zope.interface   pkgs/main/win-64::zope.interface-5.4.0-py39h2bbff1b_0
zstd         pkgs/main/win-64::zstd-1.4.9-h19a0ad4_0

Preparing transaction: ...working... done
```

**STEP 6:** Successfully Installed and check the Anaconda Navigator working or not



CMD.exe Prompt 0.1.1 — Run a cmd.exe terminal with your current environment from Navigator activated

Datalore — Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

IBM Watson Studio Cloud — IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

JupyterLab 3.3.2 — An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook 6.4.8 — Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the

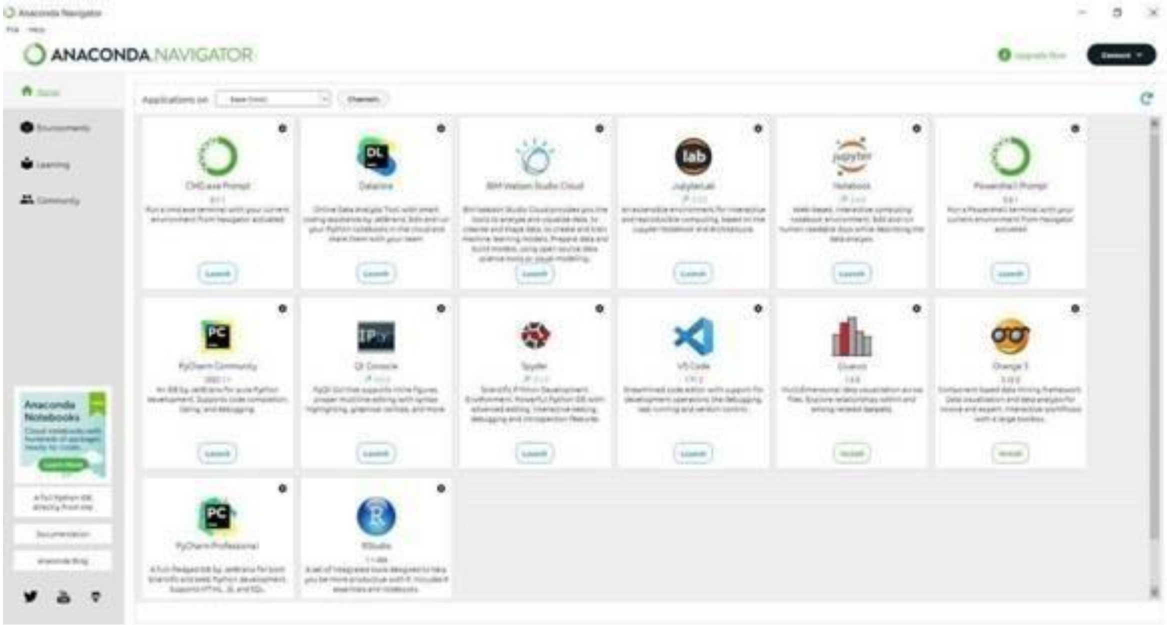
Powershell Prompt 0.0.1 — Run a Powershell terminal with your current environment from Navigator activated

PyCharm Community 2022.1.3 — An IDE by JetBrains for pure Python development. Supports code completion, listing, and debugging.

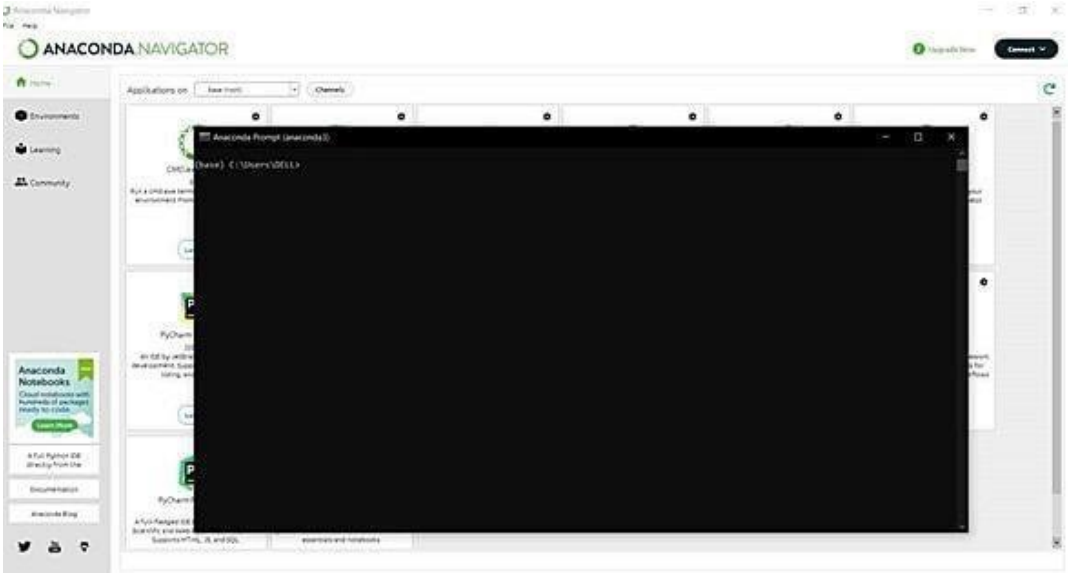
Qt Console 5.3.0 — PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

**Python packages installation:**

**Step 1:** Open the anaconda navigator in the start menu



**Step 2:** Open the CMD.exe prompt



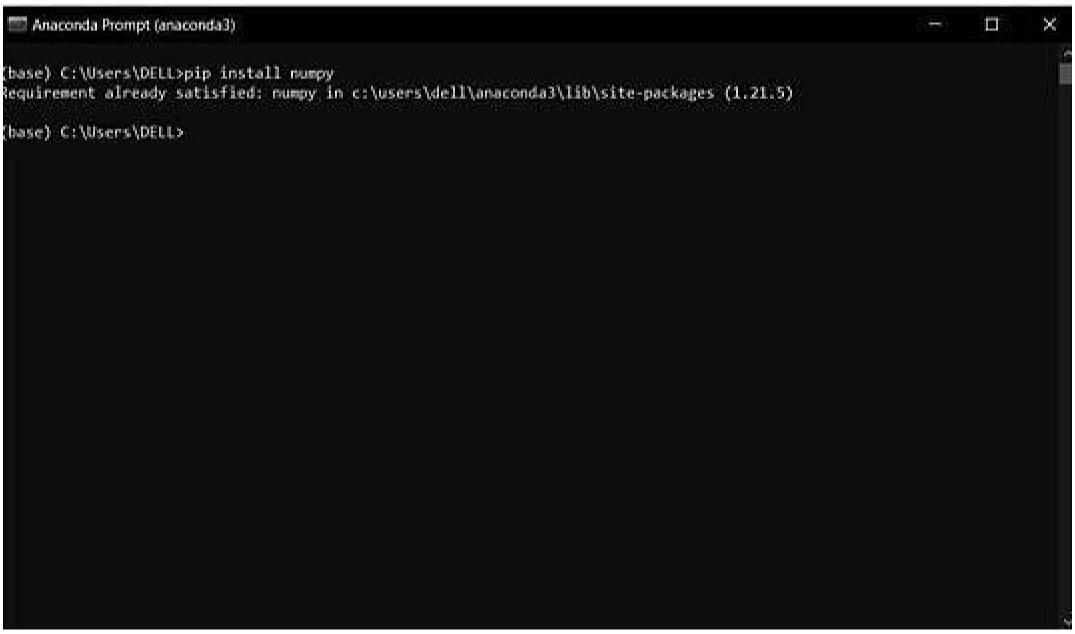
```
(base) C:\Users\DELL>
```

**Step 3:** Install the NUMPY package

To enter the numpy package enter the command in the CMD.exe Command: **Pip Install numpy**

**Numpy:**

This package is used to perform numerical computations. This package comes pre-installed with Anaconda. NumPy is used for manipulating arrays. NumPy stands for Numerical Python.



Anaconda Prompt (anaconda3)

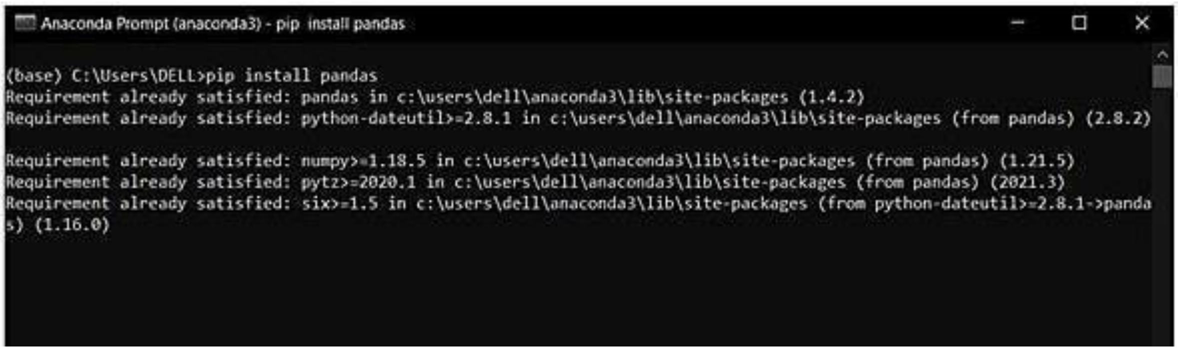
```
(base) C:\Users\DELL>pip install numpy
Requirement already satisfied: numpy in c:\users\dell\anaconda3\lib\site-packages (1.21.5)

(base) C:\Users\DELL>
```

**Step 4:** Install the pandas package.

To enter the pandas package enter the command in the CMD.exe Command: **Pip install pandas**

**Pandas:**

Pandas is one of the most widely used Python libraries for data science. It provides powerful and easy-to-use structure and data analysis tools. This package comes pre-installed with Anaconda. An open source library built on top of the NumPy library. A Python package that provides various data structures and operations for working with numerical data and time series. Mainly, it's common for data to be imported and analyzed much easier. Pandas is fast, providing users with high performance and productivity.



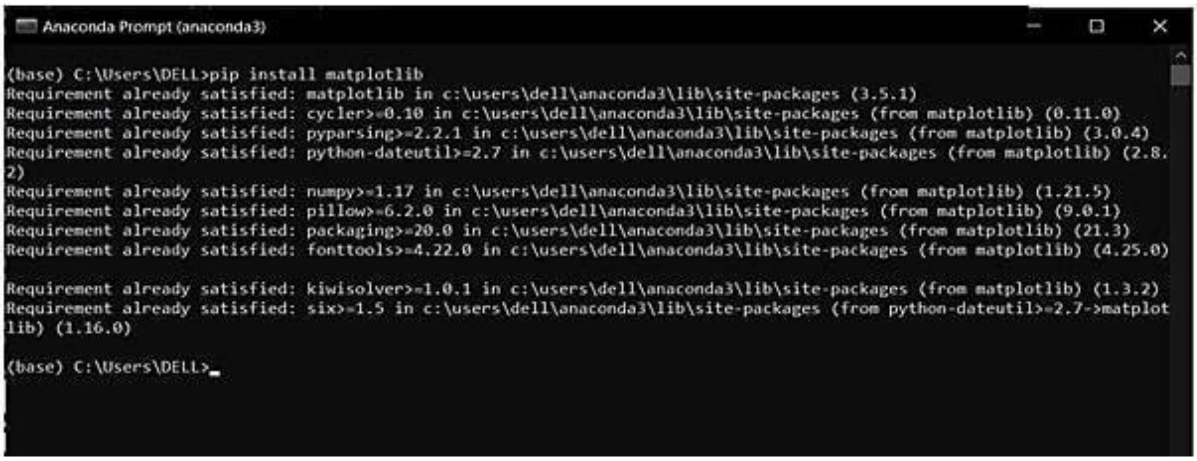
```
(base) C:\Users\DELL>pip install pandas
Requirement already satisfied: pandas in c:\users\dell\anaconda3\lib\site-packages (1.4.2)
Requirement already satisfied: python-dateutil>=2.8.1 in c:\users\dell\anaconda3\lib\site-packages (from pandas) (2.8.2)

Requirement already satisfied: numpy>=1.18.5 in c:\users\dell\anaconda3\lib\site-packages (from pandas) (1.21.5)
Requirement already satisfied: pytz>=2020.1 in c:\users\dell\anaconda3\lib\site-packages (from pandas) (2021.3)
Requirement already satisfied: six>=1.5 in c:\users\dell\anaconda3\lib\site-packages (from python-dateutil>=2.8.1->panda
s) (1.16.0)
```

**Step 5:** Install the Matplotlib package.

To enter the Matplotlib package enter the command in the CMD.exe Command: **Pip install Matplotlib**

**Matplotlib:**

Matplotlib is a comprehensive library for creating static, animated and interactive visualizations in Python. This package comes pre-installed with Anaconda. Matplotlib is a nice visualization library in Python for 2D plotting of arrays. Matplotlib is a cross-platform data visualization library based on NumPy arrays and designed to work with the wider SciPy stack. Introduced by John Hunter in 2002.



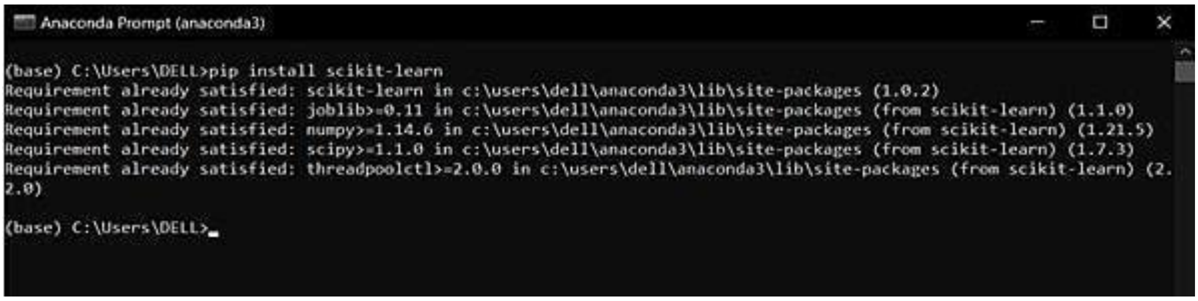
```
(base) C:\Users\DELL>pip install matplotlib
Requirement already satisfied: matplotlib in c:\users\dell\anaconda3\lib\site-packages (3.5.1)
Requirement already satisfied: cycler>=0.10 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (3.0.4)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (2.8.
2)
Requirement already satisfied: numpy>=1.17 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (1.21.5)
Requirement already satisfied: pillow>=6.2.0 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (9.0.1)
Requirement already satisfied: packaging>=20.0 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (21.3)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (4.25.0)

Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\dell\anaconda3\lib\site-packages (from matplotlib) (1.3.2)
Requirement already satisfied: six>=1.5 in c:\users\dell\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplot
lib) (1.16.0)

(base) C:\Users\DELL>_
```

**Step 6:** Install the Scikit-learn package.

To enter the Scikit- learn package enter the command in the CMD.exe Command: **Pip install Scikit-learn**

**Scikit-learn:**

This is a machine learning library for the Python programming language. This package comes pre-installed with Anaconda. Scikit Learn in Python is primarily used to focus on modeling in Python. It was only focused on modeling, not loading data.



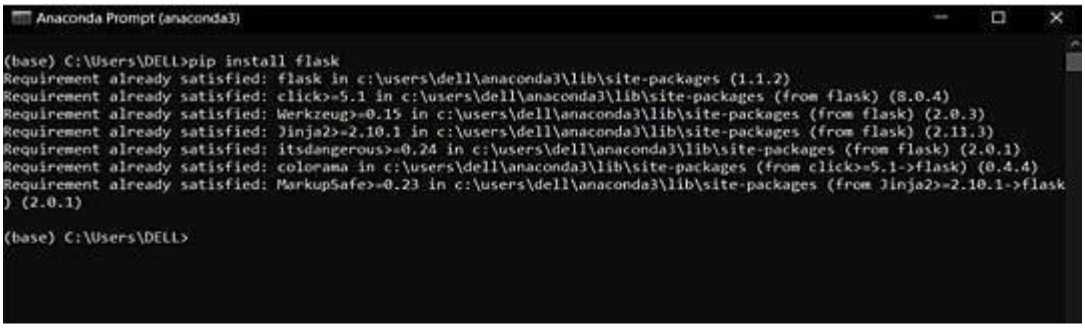
```
(base) C:\Users\DELL>pip install scikit-learn
Requirement already satisfied: scikit-learn in c:\users\dell\anaconda3\lib\site-packages (1.0.2)
Requirement already satisfied: joblib>=0.11 in c:\users\dell\anaconda3\lib\site-packages (from scikit-learn) (1.1.0)
Requirement already satisfied: numpy>=1.14.6 in c:\users\dell\anaconda3\lib\site-packages (from scikit-learn) (1.21.5)
Requirement already satisfied: scipy>=1.1.0 in c:\users\dell\anaconda3\lib\site-packages (from scikit-learn) (1.7.3)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\dell\anaconda3\lib\site-packages (from scikit-learn) (2.
2.0)

(base) C:\Users\DELL>
```

**Step 7:** Install the Flask package.

To enter the Flask package enter the command in the CMD.exe Command: **Pip install Flask**

**Flask:**

Flask is a lightweight WSGI web application framework Flask is a web application framework written in Python. It is developed by Armin Ronacher, who leads an international group of Python enthusiasts called Pocco. Flask is based on the WSGI toolkit tools and the Jinja2 template engine. Both are Pocco projects.



```
(base) C:\Users\DELL>pip install flask
Requirement already satisfied: flask in c:\users\dell\anaconda3\lib\site-packages (1.1.2)
Requirement already satisfied: click>=5.1 in c:\users\dell\anaconda3\lib\site-packages (from flask) (8.0.4)
Requirement already satisfied: Werkzeug>=0.15 in c:\users\dell\anaconda3\lib\site-packages (from flask) (2.0.3)
Requirement already satisfied: Jinja2>=2.10.1 in c:\users\dell\anaconda3\lib\site-packages (from flask) (2.11.3)
Requirement already satisfied: itsdangerous>=0.24 in c:\users\dell\anaconda3\lib\site-packages (from flask) (2.0.1)
Requirement already satisfied: colorama in c:\users\dell\anaconda3\lib\site-packages (from click>=5.1->flask) (0.4.4)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\dell\anaconda3\lib\site-packages (from Jinja2>=2.10.1->flask
) (2.0.1)

(base) C:\Users\DELL>
```

**PROJECT FLOW**

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

● User interacts with the UI (User Interface) to enter Data

● The entered data is analyzed by the model which is integrated

● 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

● Data Collection. ○ Collect the dataset or Create the dataset

● Data Preprocessing. ○ Import the Libraries.

○ Importing the dataset.

○ Checking for Null Values.

○ Data Visualization.

○ Taking care of Missing Data.

○ Label encoding.

○ One Hot Encoding.

○ Feature Scaling.

○ Splitting Data into Train and Test.

● Model Building ○ Training and testing the model

○ Evaluation of Model

● Application Building

○ Create an HTML file

○ Build a Python Code

**PROJECT OBJECTIVES**

● To understand regression and classification problems.

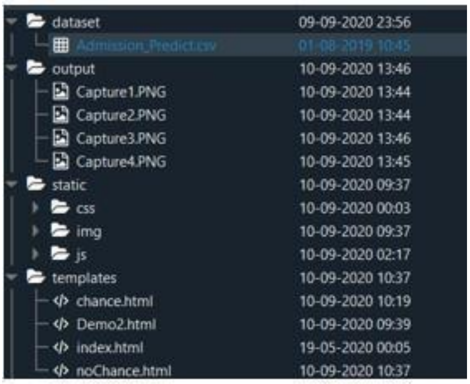
● To grab insights from data through visualization.

● Applying different Machine Learning algorithms to determine the probability of acceptance in a particular university.

● Evaluation metrics build a web application using the Flask framework

**PROJECT STRUCTURE**

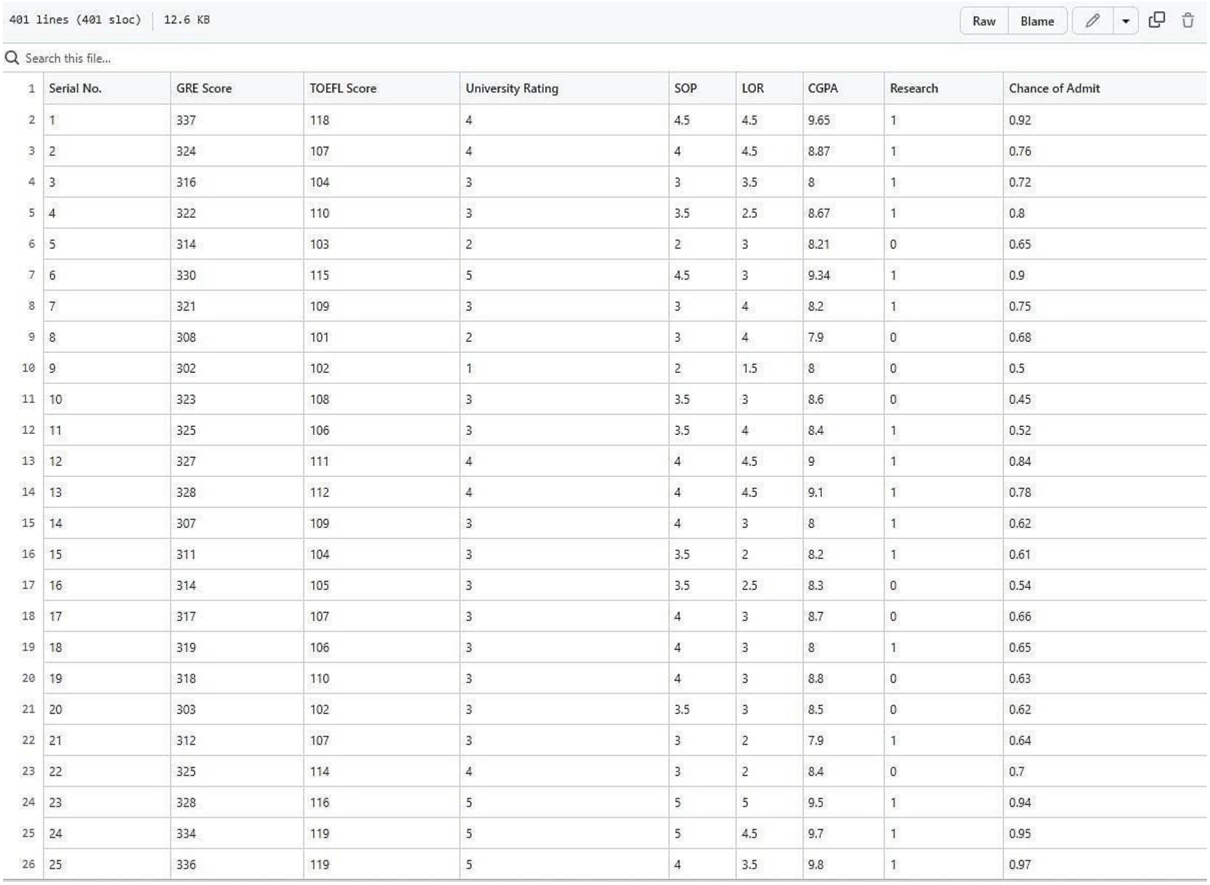
Create a Project folder that contains files as shown below



```
dataset                  09-09-2020 23:56
└ Admission_Predict.csv  01-08-2019 10:45
output                   10-09-2020 13:46
├ Capture1.PNG           10-09-2020 13:44
├ Capture2.PNG           10-09-2020 13:44
├ Capture3.PNG           10-09-2020 13:46
└ Capture4.PNG           10-09-2020 13:45
static                   10-09-2020 09:37
  css                    10-09-2020 00:03
  img                    10-09-2020 09:37
  js                     10-09-2020 02:17
templates                10-09-2020 10:37
├ chance.html            10-09-2020 10:19
├ Demo2.html             10-09-2020 09:39
├ index.html             19-05-2020 00:05
└ noChance.html          10-09-2020 10:37
```

# DATA COLLECTION

The path to common information varies by project type. ML projects use real-time information. Information indexes can be collected from a variety of sources such as documents, data sets, sensors, and other sources, using free information collection from the Internet. Kaggle and the UCI Machine learning Repository are the most commonly used repositories for sorting information for machine learning models. Kaggle is probably the most visited website used for information gathering. Collect the dataset or Create the dataset.



| 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 |

# DATA PRE-PROCESSING

**Importing the Libraries:**

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

● 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.

● 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.

● Seaborn-

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

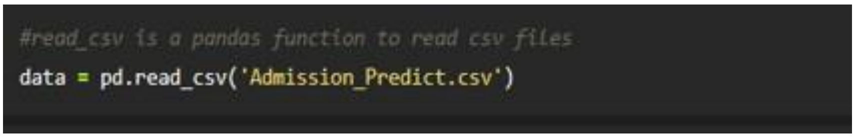
● Matplotlib-

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 command Data= pd.read_csv(r”File_location/filename.csv”)

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

Our Dataset Admission_Predict contains following Columns

1.Serial No.

2. GRE Score

3.TOEFL Score

4.University Rating

5.SOP

6. LOR

7. CGPA

8. Chance of Admit

● Imputing data using the Imputation method in sklearn.

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

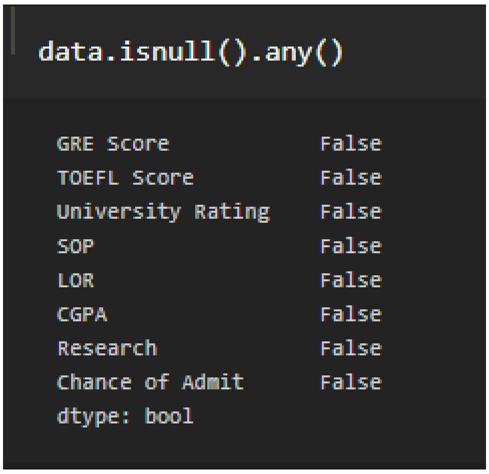
**Handling Missing Values:**

After loading it is important to check the complete information of data as it can indicate many of the 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

● Imputing data using the Imputation method in sklearn.

● 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
```

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

- You can also check the number of null values present in the columns by the using isnull().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.

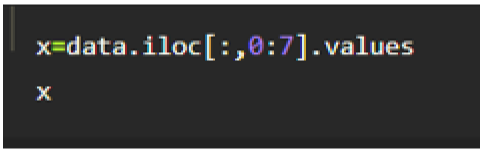
To visualize the dataset we need libraries called Matplotlib and Seaborn. The Matplotlib library is a Python 2D plotting library that allows you to generate plots, scatter plots, histograms, bar charts etc.

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

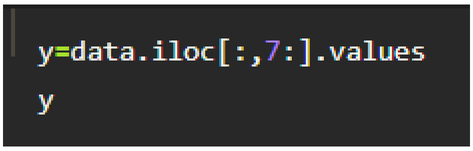
**Splitting Dependent And Independent Columns:**

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.

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



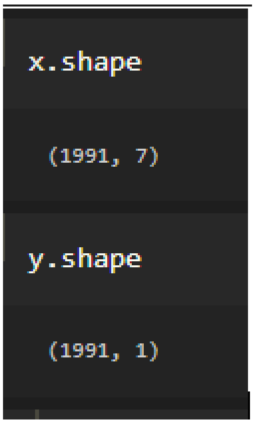
```
x=data.iloc[:,0:7].values
x
```



```
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)
```

● You can see in x we have 1991 rows with 7 columns and y has 1 column with the same number 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.

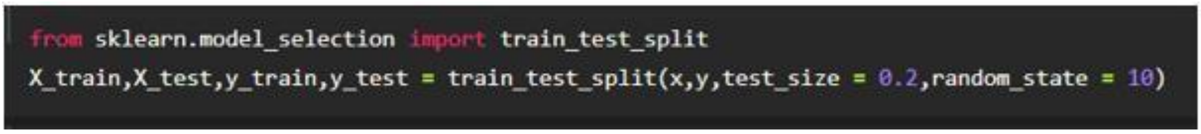
**Test set:** A set of unseen data used solely to evaluate the performance of thefully displayed classifier.

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 dataset into 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.

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



```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 10)
```

- x_train
- x_test
- y_train

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

- y_test

## 1.2PURPOSE

For anyone pursuing their postgraduate studies, it would be difficult for them to find out what university they may join, based on their GPA, Quants, Verbal, TOEFL and AWA Scores. People may apply to many universities that look for candidates with a higher score set, instead of applying to universities at which they have a chance of getting into. This would be detrimental to their future. It is very important that a candidate should apply to university that he/she has a good chance of getting into, instead of applying to university that they may never get into. There aren’t many efficient ways to find out the university that one can get into, relatively quickly. The Education Based Prediction System helps a person decide what University they can apply to with their scores. The dataset that is used for processing consists of the following parameters: University name, Quants and Verbal Scores (GRE) TOEFL and AWA Scores. The GRE Test (Graduate Record Examinations) is a standardized test used by many universities and graduate schools around the world as part of the graduate admissions process. Other factors are also taken into consideration while applying to University, such as Letter of Recommendation (a formal document that validates someone's work, skills or academic performance), Statement of Purpose (a critical piece of a graduate school application that tells admissions committees who you are, what your academic and professional interests are, and how you'll add value to the graduate program you're applying to), Co-curricular activities and Research papers as well (research papers from journals that are not well known or have a high percentage of plagiarism are not taken into consideration for this case). When a person has completed their undergraduate degree and wants to pursue a Postgraduate degree in a field of their choice, more often than not, it is very confusing for the person to figure out what University they should apply to with the scores that they have obtained in GRE and TOEFL, along with their GPA at the time of their graduation. Many candidates may apply to Universities that do not fall under their score requirements and hence waste a lot of time. Applying to many university with scores also increases the cost. There are not many efficient methods that are available to help address this issue and hence an Education Predictor System has been developed. In the system proposed, a person can enter their scores in the respective fields provided. The system then processes the data entered and produces an output of the list of university that a person could get into, with their scores. This is relatively quick and helps conserve time and money. In order to achieve this we have proposed a novel method utilising Machine Learning algorithms. To maximize the accuracy of our model, we have taken into consideration not one; but several machine learning algorithms. These algorithms include Linear Regression, Decision Tree and Random Forest. More about these algorithms will be covered in the Algorithms section of this paper. These Algorithms are then compared and the algorithm which has the best key performance indicators will be used to develop the Prediction System. We also look forward to incorporate clustering of universities based on a profile and then classifying them as less likely, highly likely acceptance etc.

# CHAPTER 2

# LITERATURE SURVEY

## 2.1 Existing Problem

- The web application assists in prediction of university admission with the help of 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 Multilayer Perceptron. Experiments show that the Multilayer Perceptron model surpasses other models.
- It is very fast at classifying unknown records but It constructs linear boundaries
- Handling multi-dimensional and multi variety data and its requires more time and resources.

## 2.2 References

**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:**

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

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

## 2.3 Problem Statement Definition

Accurate Prediction of University For Eligible and non-Eligible Students. To Filter the University according to the marks (GRE,TOEFL). To understand regression and classification problems. To grab insights from data through visualization. Applying different ML algorithms to determine the probability of acceptance in a particular university. Build a web application using the Flask framework
This paper is divided into four sections

(i) Data Collection

(ii) Comparison of machine learning models on collected data

(iii) Training of system on most promising model

(iv) Testing

# CHAPTER 3

# IDEATION & PROPOSED SOLUTION

## 3. IDEATION AND PROPOSED SOLUTION

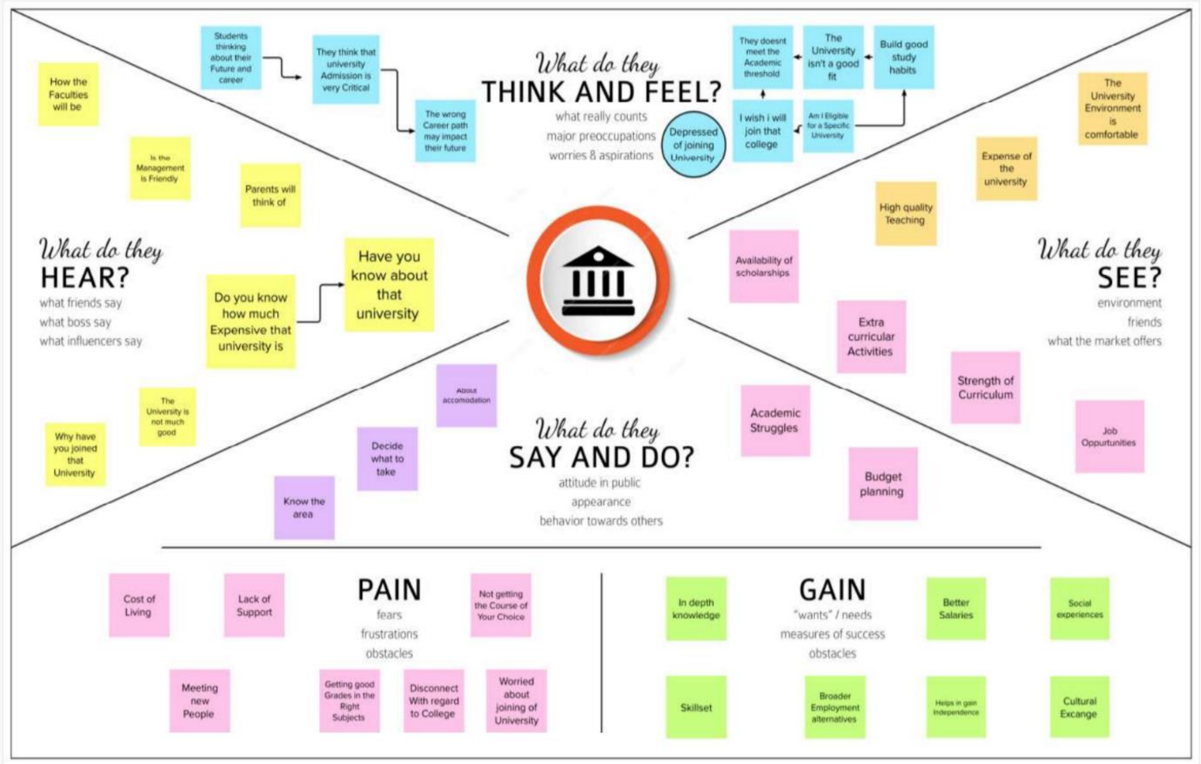
Ideation is the process where you generate ideas and solutions through sessions such as Sketching, Prototyping, Brainstorming, Brainwriting, Worst Possible Idea, and a wealth of other ideation techniques. Ideation is also the third stage in the Design Thinking process. In this project the ideation phase consist of,

- Empathy Map
- Brainstorming
- Proposed Solution
- Problem Solution Fit

### 3.1 Empathy Map Canvas

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviors and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.


What do they
THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Students thinking about their Future and career
They think that university Admission is very Critical
The wrong Career path may impact their future
Depressed of joining University
They doesnt meet the Academic threshold
The University isn't a good fit
Build good study habits
I wish i will join that college
Am I Eligible for a Specific University
What do they
HEAR?
what friends say
what boss say
what influencers say
How the Faculties will be
Is the Management is Friendly
Parents will think of
Do you know how much Expensive that university is
Have you know about that university
The University is not much good
Why have you joined that University
What do they
SEE?
environment
friends
what the market offers
The University Environment is comfortable
Expense of the university
High quality Teaching
Availability of scholarships
Extra curricular Activities
Strength of Curriculum
Job Oppurtunities
What do they
SAY AND DO?
attitude in public
appearance
behavior towards others
About accomodation
Decide what to take
Know the area
Academic Struggles
Budget planning
PAIN
fears
frustrations
obstacles
Cost of Living
Lack of Support
Not getting the Course of Your Choice
Meeting new People
Getting good Grades in the Right Subjects
Disconnect With regard to College
Worried about joining of University
GAIN
"wants" / needs
measures of success
obstacles
In depth knowledge
Better Salaries
Social experiences
Skillset
Broader Employment alternatives
Helps in gain Independence
Cultural Excange

## 3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

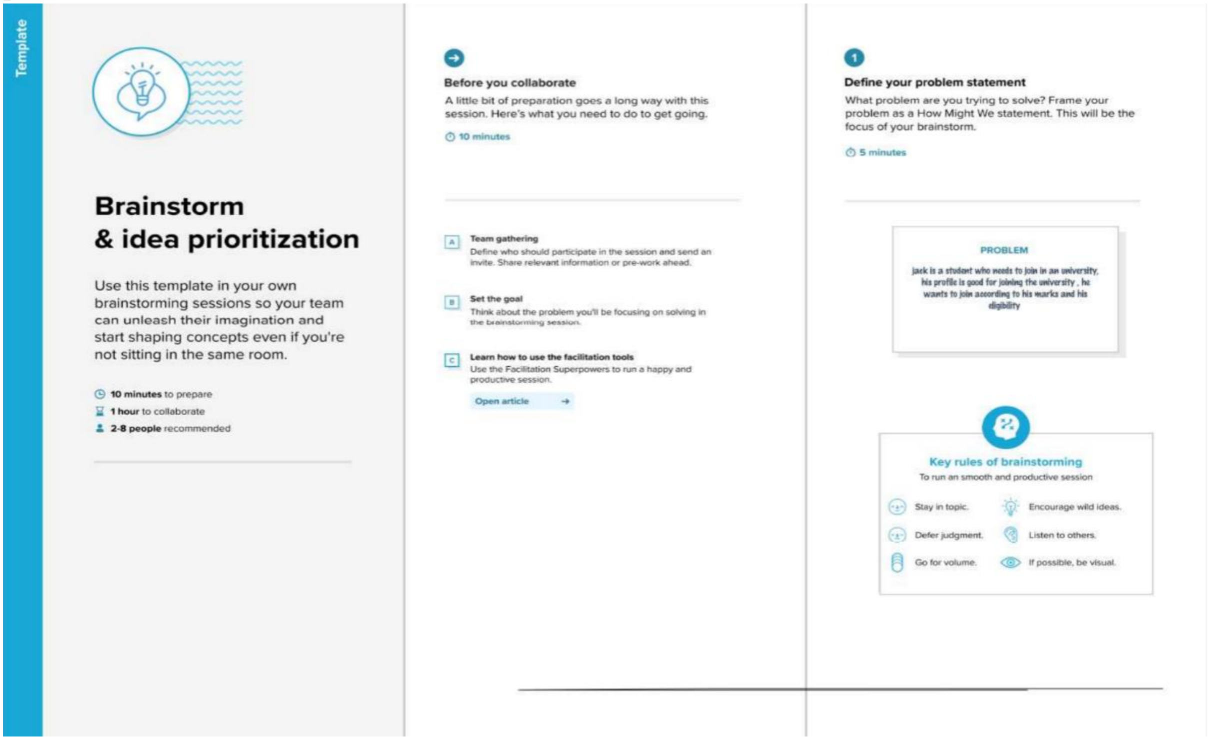
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Step-1: Team Gathering, Collaboration and Select the Problem Statement

In this step team members gather and provide their ideas and collaborate those

ideas and select their problem statement. The ideas should be relevant to their

problem statement



Template

**Brainstorm & idea prioritization**

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

**10 minutes** to prepare
**1 hour** to collaborate
**2-8 people** recommended

**Before you collaborate**

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1 **Define your problem statement**

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM

jack is a student who needs to join in an university, his profile is good for joining the university , he wants to join according to his marks and his eligibility

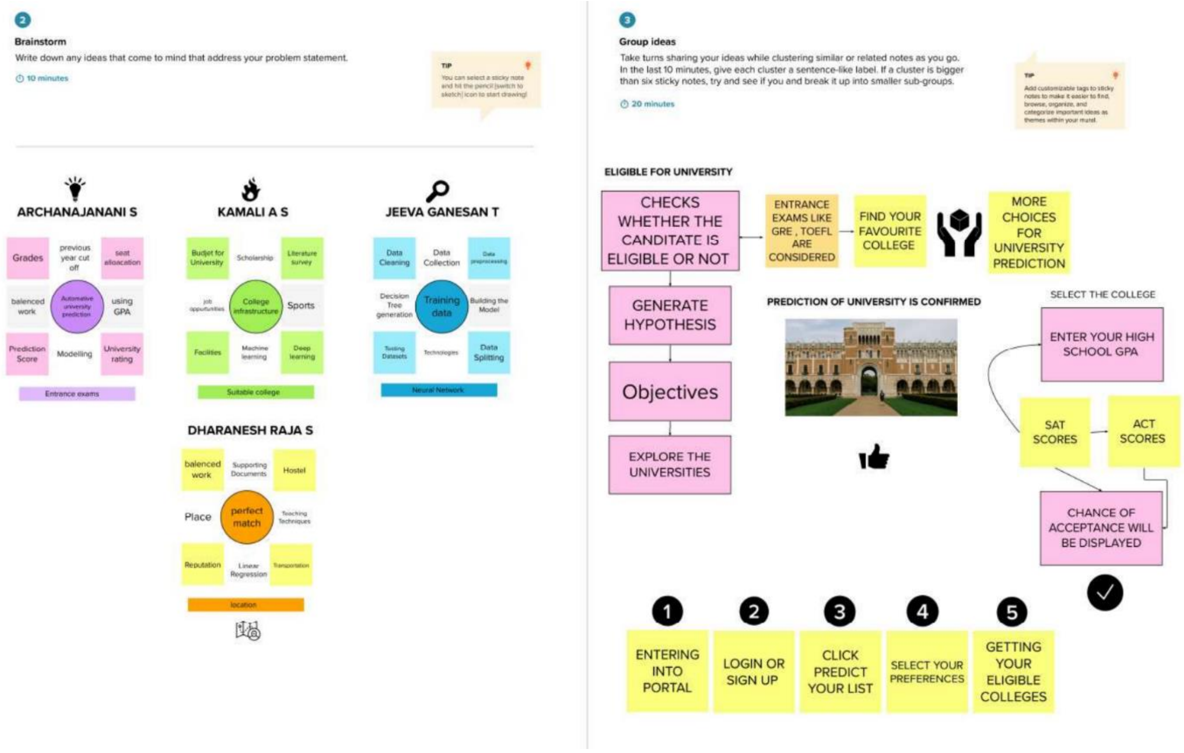
**Key rules of brainstorming**

To run an smooth and productive session

Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping

In this step they put their ideas and views which are prioritized based on their importance and the ideas are grouped. These ideas are categorized according to their relevant classifications.



2

**Brainstorm**

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

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

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.

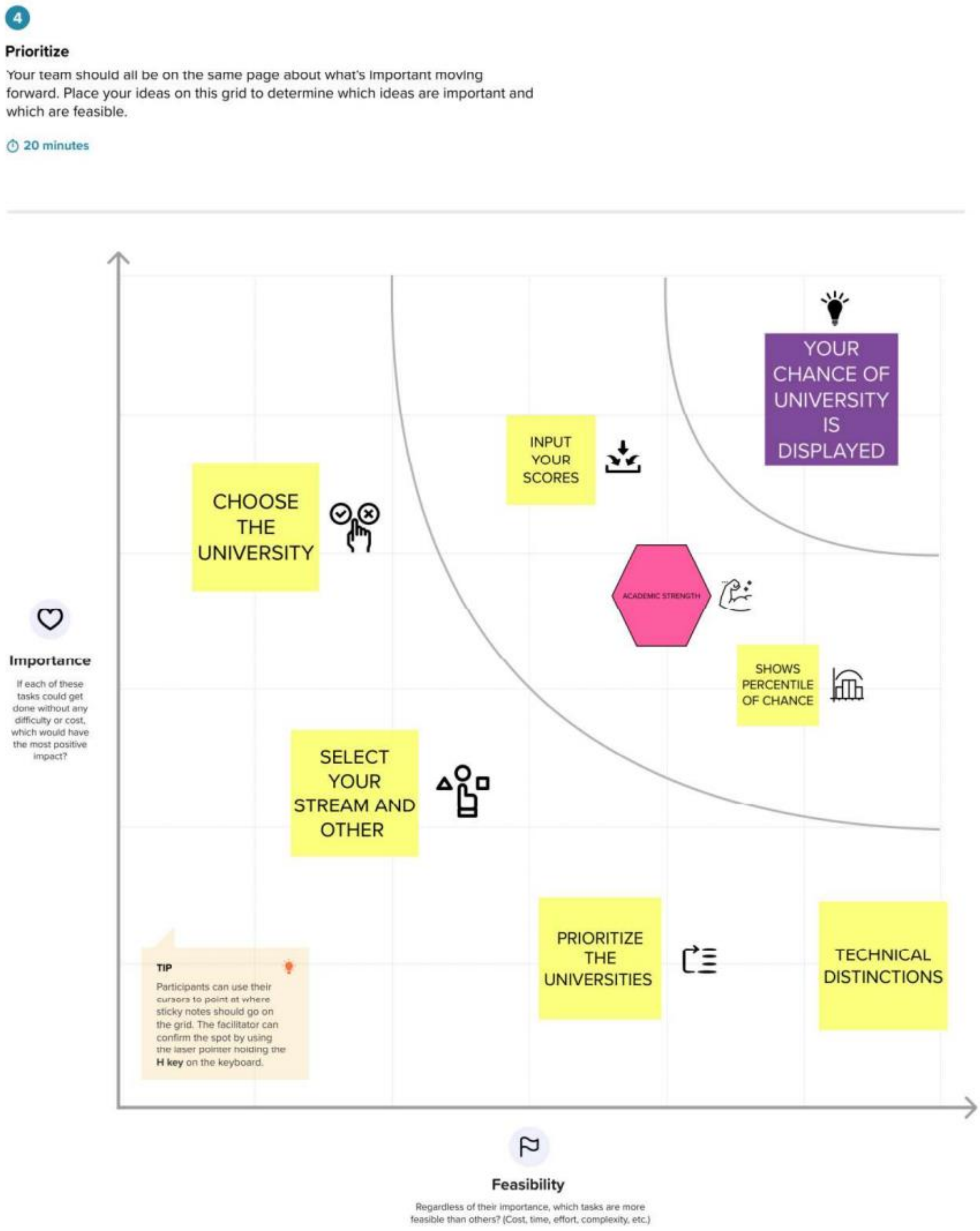
20 minutes

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Step-3: Idea Prioritization

As mentioned, idea prioritization is just a part of the idea management process. Having a structured idea management process and a systematic way of gathering, evaluating and prioritizing new ideas takes time. To make it work, the entire idea management process should be integrated into everyday ways of working.



4

**Prioritize**

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

YOUR CHANCE OF UNIVERSITY IS DISPLAYED
INPUT YOUR SCORES
CHOOSE THE UNIVERSITY
ACADEMIC STRENGTH
SHOWS PERCENTILE OF CHANCE
SELECT YOUR STREAM AND OTHER
PRIORITIZE THE UNIVERSITIES
TECHNICAL DISTINCTIONS

**Importance**

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

**TIP**

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

**Feasibility**

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

## 3.3 Proposed Solution:

Each year in India, lakhs of students are getting graduated and are willing to pursue postgraduation in abroad. A good number of freshly graduated students are not familiar with the requirements and procedures of the postgraduate admission and might spend a considerable amount of money to get advice from consultancy organizations to help them identify their admission chances.This Helps in predicting the eligibility of students to get admission in the best possible university based on their scores and the chance of admission is calculated.Well-designed portal to assist the students in joining better universities according to his/her scores. Helping the students to join a good university is the main aim of this project.

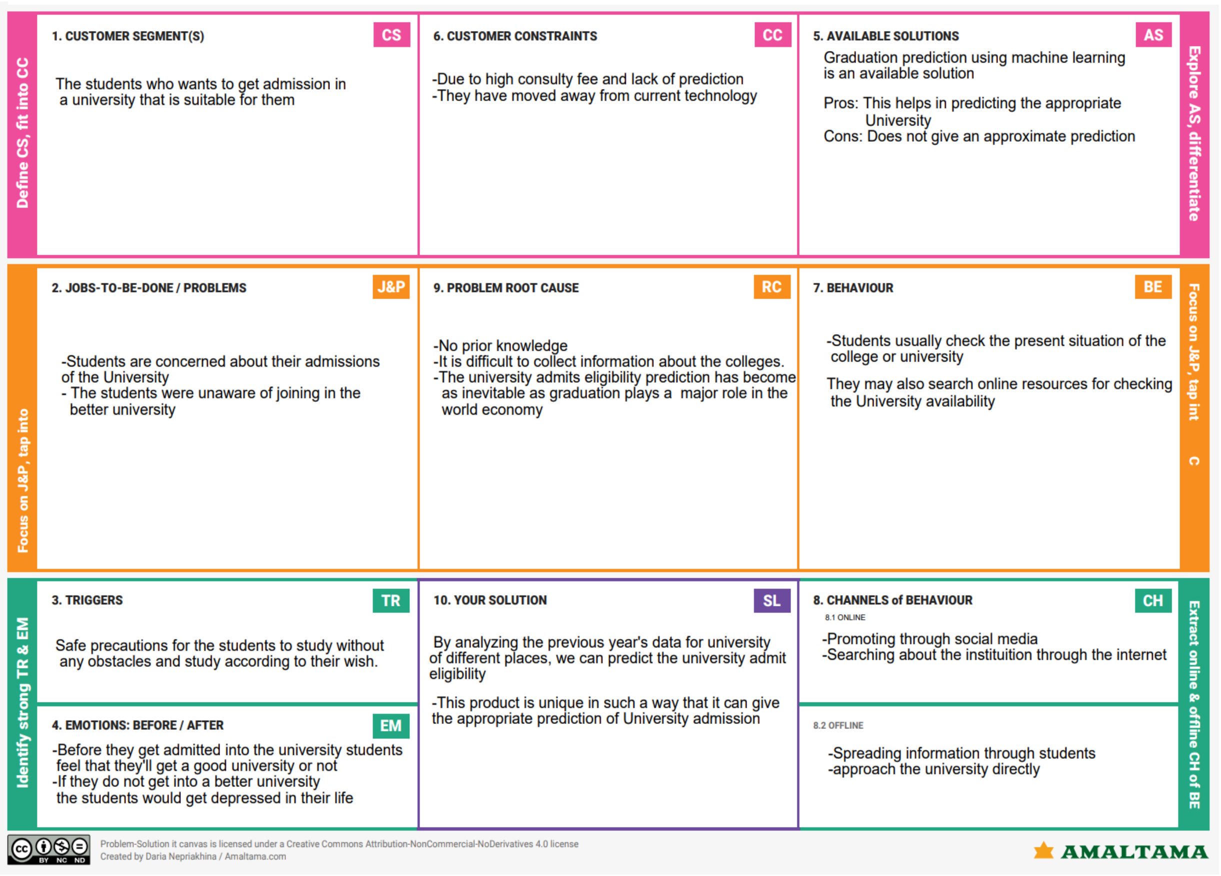
Assisting the students to join a good university reduces their stress and makes them more confident when it comes to admission. Thus opening new opportunities for a person as it contributes to their social and intellectual growth.This project idea can be adapted to a bigger scale than just the local context. Our software system maintains the same high level performance even when the number of users is increased, and the database holds capacity to withstand growing numbers of queries and our operating system performs on different classes of software.

## 3.4 Problem Solution fit:

**Problem – Solution Fit Template:** The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer’s problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why

**Purpose:**

- □ Solve complex problems in a way that fits the state of your customers.
- □ Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behaviour.
- □ Sharpen your communication and marketing strategy with the right triggers and messaging.
- □ Increase touch-points with your company by finding the right problem behaviour fit and building trust by solving frequent annoyances, or urgent or costly problems.
- □ Understand the existing situation in order to improve it for your target group.



Define CS, fit into CC

**1. CUSTOMER SEGMENT(S)** CS

The students who wants to get admission in a university that is suitable for them

**6. CUSTOMER CONSTRAINTS** CC

-Due to high consulty fee and lack of prediction
-They have moved away from current technology

**5. AVAILABLE SOLUTIONS** AS

Graduation prediction using machine learning is an available solution

Pros: This helps in predicting the appropriate University
Cons: Does not give an approximate prediction

Explore AS, differentiate

Focus on J&P, tap into

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

-Students are concerned about their admissions of the University
- The students were unaware of joining in the better university

**9. PROBLEM ROOT CAUSE** RC

-No prior knowledge
-It is difficult to collect information about the colleges.
-The university admits eligibility prediction has become as inevitable as graduation plays a major role in the world economy

**7. BEHAVIOUR** BE

-Students usually check the present situation of the college or university

They may also search online resources for checking the University availability

Focus on J&P, tap int

C

Identify strong TR & EM

**3. TRIGGERS** TR

Safe precautions for the students to study without any obstacles and study according to their wish.

**4. EMOTIONS: BEFORE / AFTER** EM

-Before they get admitted into the university students feel that they'll get a good university or not
-If they do not get into a better university the students would get depressed in their life

**10. YOUR SOLUTION** SL

By analyzing the previous year's data for university of different places, we can predict the university admit eligibility

-This product is unique in such a way that it can give the appropriate prediction of University admission

**8. CHANNELS of BEHAVIOUR** CH

8.1 ONLINE

-Promoting through social media
-Searching about the instituition through the internet

8.2 OFFLINE

-Spreading information through students
-approach the university directly

Extract online & offline CH of BE

AMALTAMA

# CHAPTER 4

# REQUIREMENT ANALYSIS

4.1 Functional requirement

● FR - 1

Functional Requirement (Epic) - User Requirement

- Sub Requirement (Story/Sub-Task) - Registration through Form, Registration through Gmail, Registration through LinkedIn, Registration through Google

● FR-2
Functional Requirement (Epic) - User Confirmation

- Sub Requirement (Story/Sub-Task) - Confirmation via Email, Confirmation via OTP

● FR - 3
Functional Requirement (Epic) – User Details

- Sub Requirement (Story/Sub-Task) - Submit the documents,

GRE or/and TOEFL scoresheet, SAT Scoresheet, Curriculum Vitae (CV) , Statement of Purpose (SOP), Letter of Recommendation

● FR - 4

Functional Requirement (Epic) - User Requirements

- Sub Requirement (Story/Sub-Task) - Upload all the relevant documents in the appropriate location in the website, Based on the uploads, the system will display all the necessary information, The list of all possible university for the candidate will be displayed based on the given information

4.2 Non-Functional requirements

● NFR-1

Non-Functional -Usability

- Requirement Description - User friendly, The page would not take a lot of time to load the content and display them

● NFR-2

Non-Functional -Security

- Requirement Description - Database should be backed up every hour,If any error occur the system should be able to come back to normal operation within an hour

● NFR-3

Non-Functional -Reliability

- Requirement Description - System will run 24/7, System will always work for maximum reliability due to importance of data and damages that may cause by incomplete and incorrect data

● NFR-4

Non-Functional -Performance

- Requirement Description - The Website can efficiently handle traffic by service, viewing this web page using 56- kbps modem connection would not exceed 30 sec

● NFR-5

Non-Functional - Availability

- Requirement Description - Fast and efficient, Avoids data redundancy and inconsistency, The system will work on 7 days a week ,24 hours a day.

● NFR-6

Non-Functional - Scalability

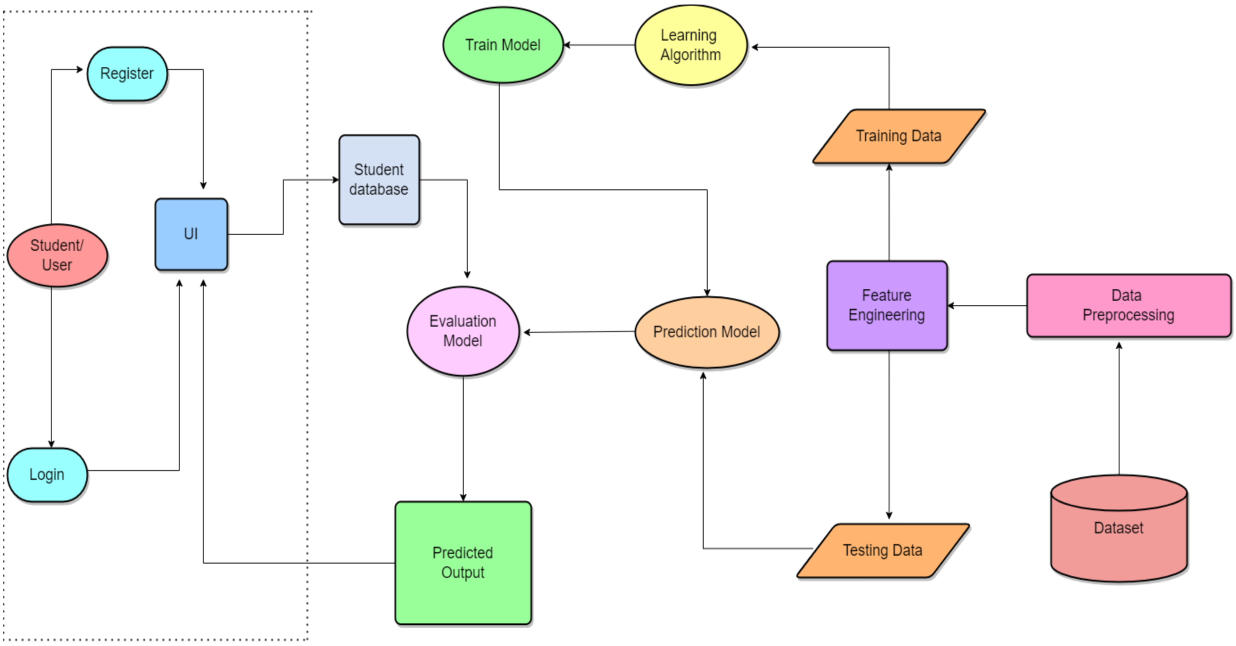
- Requirement Description - It can able to manage large number of users

# CHAPTER 5
# PROJECT DESIGN

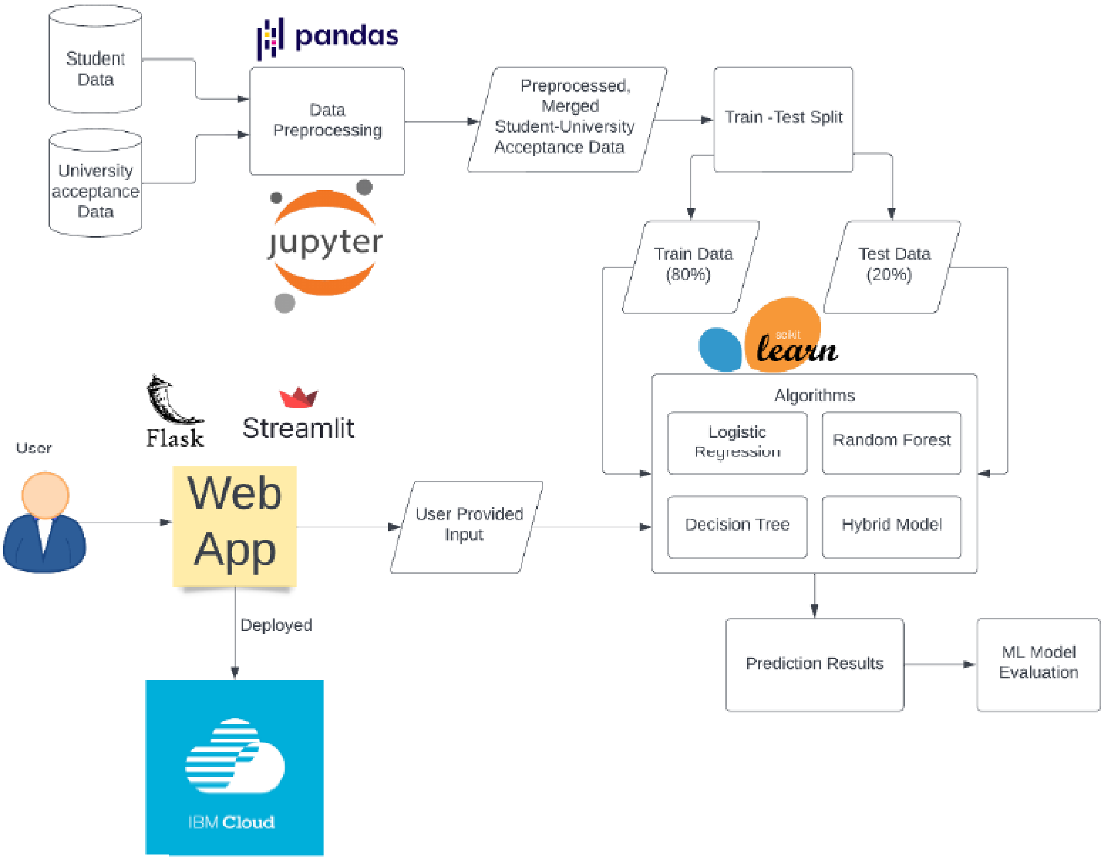
## 5.1 Data Flow Diagrams

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various subprocesses the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

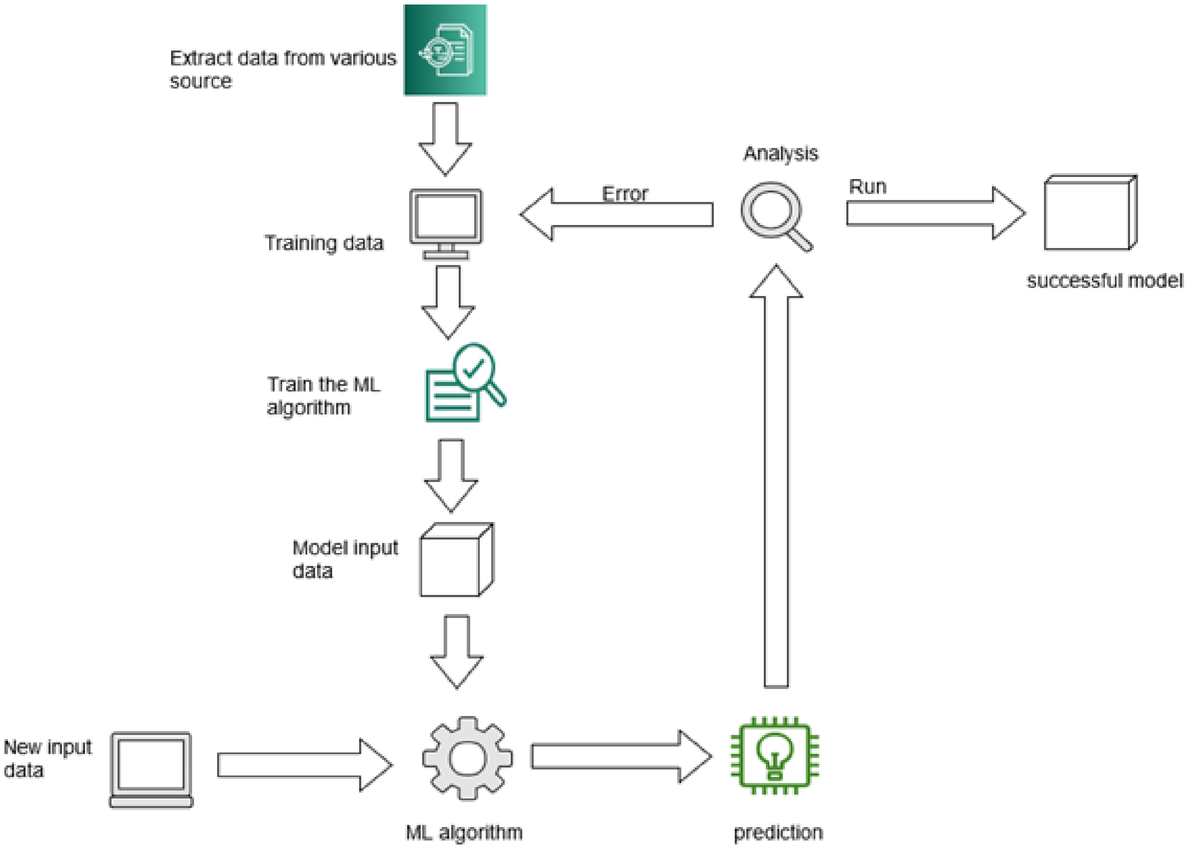
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can graphically depict the right amount of the system requirement. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2 Technical Architecture



## Solution Architecture



## 5.3 User Stories

**Sprint-01:**

Usn-01 - As a user, we can register for the application by entering email, password, and confirming password.

Usn-02 - As a user, we will receive confirmation email once we have registered for the application.

Usn-03 - As a user, we can register for the application through Gmail.

Usn-04 - As a user, we can log into the application by entering email & password.

Usn-05 - Enter all the marks as specified column To check the availability.

**Sprint-02:**

Usn-06 - As a user, we can register for the application through Social medias.

Usn-07 - While entering the home page, I can see profile, customer details and logout.

**Sprint-03:**

Usn-08 - As a customer, we can login to the website by entering email & password.

Usn-09 - As a user, we can log into the application by entering email & password.

**Sprint-04:**

Usn-10 - Based on the accuracy level, the result will be Sorted on ascending order.

Usn-11 - As a user, they can find if they are  eligible for university.

# CHAPTER 6

# PROJECT PLANNING & SCHEDULING

## 6.1 Sprint Planning & Estimation:

**Sprint-01:**

Usn-01 - As a user, I can register for the application by entering my email, password, and confirming my password.

Team members

- JEEVGANESAN T

Usn-02 - As a user, I will receive confirmation email once I have registered for the application

Team members

- DHARANESH RAJA S

Usn-03 As a user, I can register for the application through Gmail

Team members

- ARCHANA JANANI S

Usn-04 As a user, I can log into the application by entering email & password

Team members

- KAMALI A S

**Sprint-02:**

Usn-05 - As a user, I can register for the application through Facebook

Team members

- KAMALI A S

**Sprint-03:**

Usn-06 - User can check the eligibility of various criteria Team members

- ARCHANA JANANI S

**Sprint-04:**

Usn-07 - As a user I can login by username and password for application.

Team members

JEEVA GANESAN T

**6.2 Sprint Delivery Schedule**

**Sprint-1:**

● Total Story Points - 20

● Duration - 6Days

● Sprint Start Date - 24 Oct 2022

● Sprint End Date (Planned) - 29 Oct 2022

● Story Points Completed (as on Planned End Date) - 20

● Sprint Release Date (Actual) – 30 Oct 2022

**Sprint-2:**

● Total Story Points - 20

● Duration - 6 Days

● Sprint Start Date - 07 Nov 2022

● Sprint End Date (Planned) – 12 Nov 2022

● Story Points Completed (as on Planned End Date) - 20

● Sprint Release Date (Actual) – 13 Nov 2022

**Sprint-4:**

● Total Story Points - 20

● Duration - 6 Days

● Sprint Start Date - 14 Nov 2022

● Sprint End Date (Planned) - 19 Nov 2022

● Story Points Completed (as on Planned End Date) - 20

● Sprint Release Date (Actual) - 19 Nov 2022

## 6.2 Sprint Delivery Schedule:

**Project Tracker, Velocity:**

| **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 | 30 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 20 | 06 Nov 2022 |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 20 | 13 Nov 2022 |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 20 | 19 Nov 2022 |

### 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)

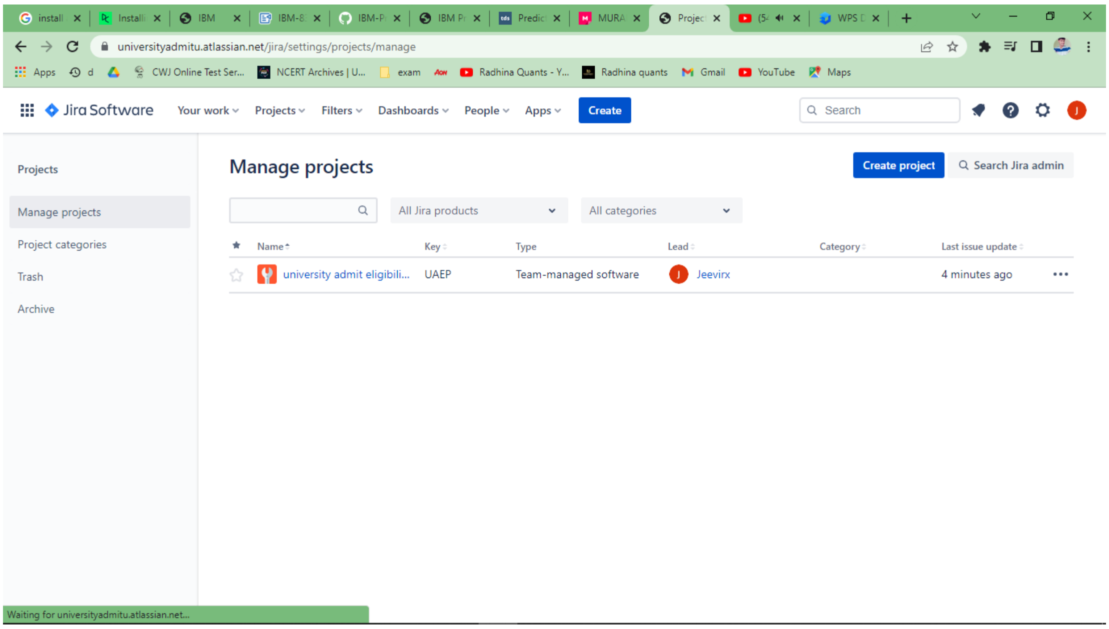
### Burndown Chart:

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 canbe applied to any project containing measurable progress over time.

$$AV = \frac{\textit{sprint duration}}{\textit{velocity}} = \frac{20}{10} = 2$$

## 6.3 Reports from JIRA

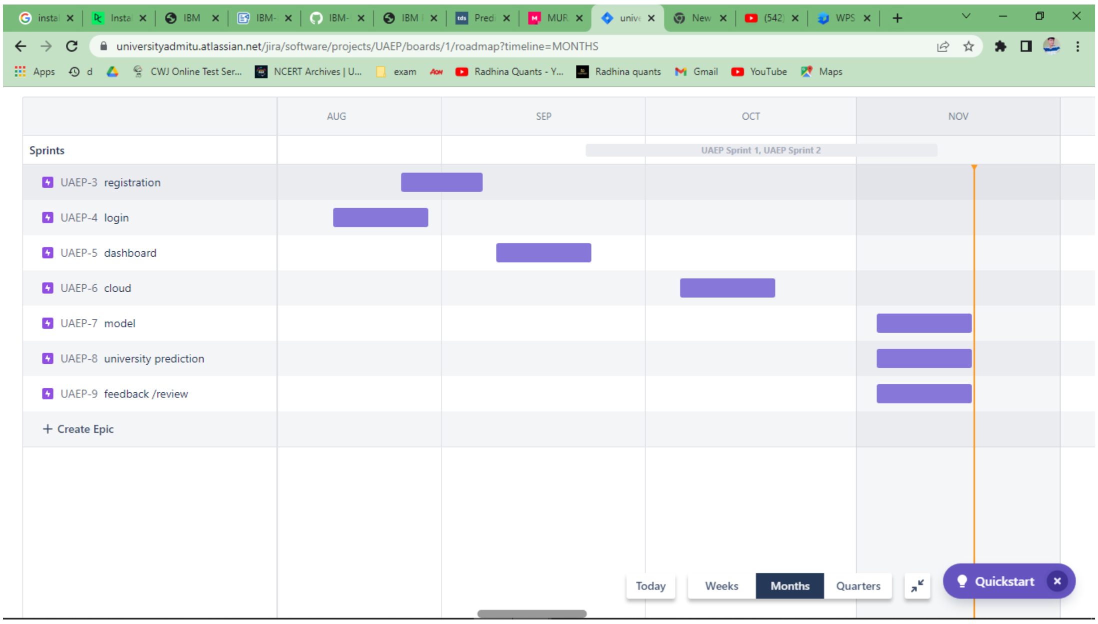
1.Image showing the project created in the JIRA software



Manage projects

| Name | Key | Type | Lead | Category | Last issue update |
|---|---|---|---|---|---|
| university admit eligibili... | UAEP | Team-managed software | Jeevirx | | 4 minutes ago |

2. Image showing the List of Sprints and their time duration in the project



Sprints

UAEP-3 registration

UAEP-4 login

UAEP-5 dashboard

UAEP-6 cloud

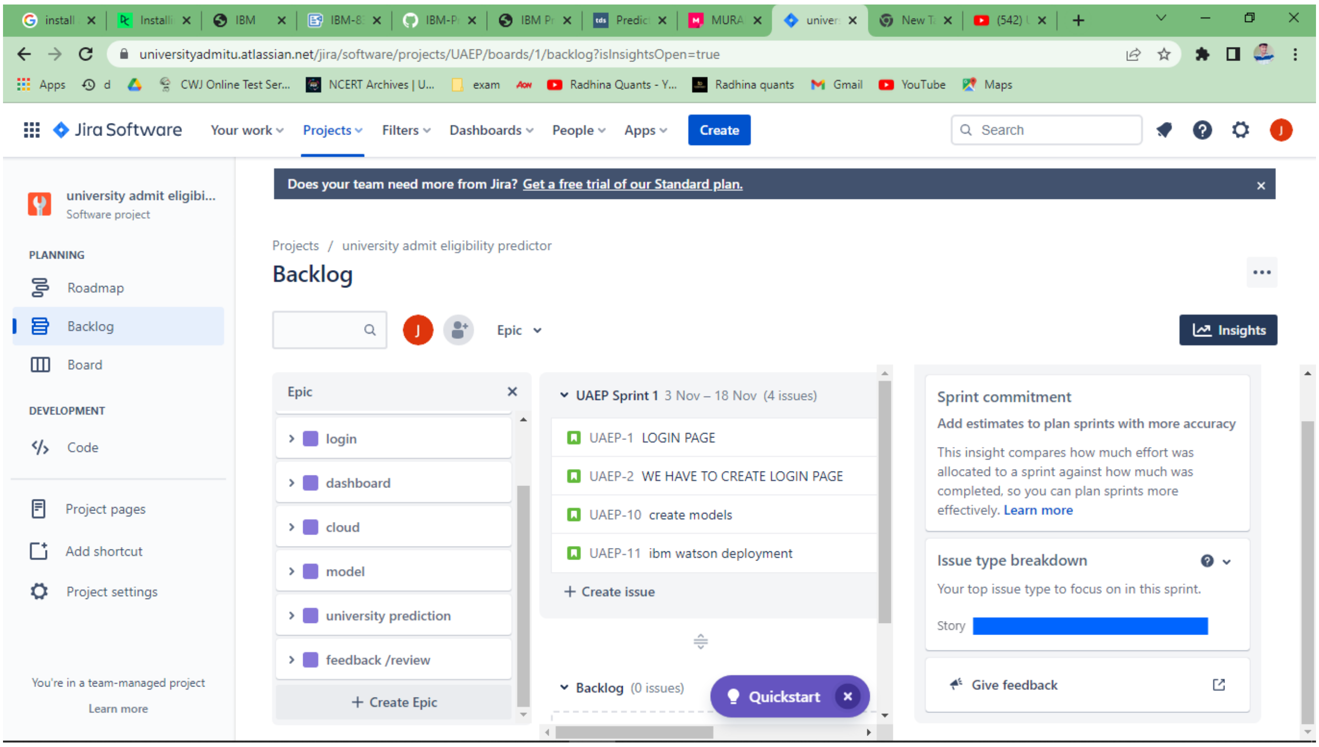
UAEP-7 model

UAEP-8 university prediction

UAEP-9 feedback /review

+ Create Epic

3.Images showing the User Stories involved in each sprints



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / university admit eligibility predictor

Backlog

Epic
- login
- dashboard
- cloud
- model
- university prediction
- feedback /review

UAEP Sprint 1 3 Nov – 18 Nov (4 issues)

- UAEP-1 LOGIN PAGE
- UAEP-2 WE HAVE TO CREATE LOGIN PAGE
- UAEP-10 create models
- UAEP-11 ibm watson deployment

+ Create issue

Backlog (0 issues)

Sprint commitment

Add estimates to plan sprints with more accuracy

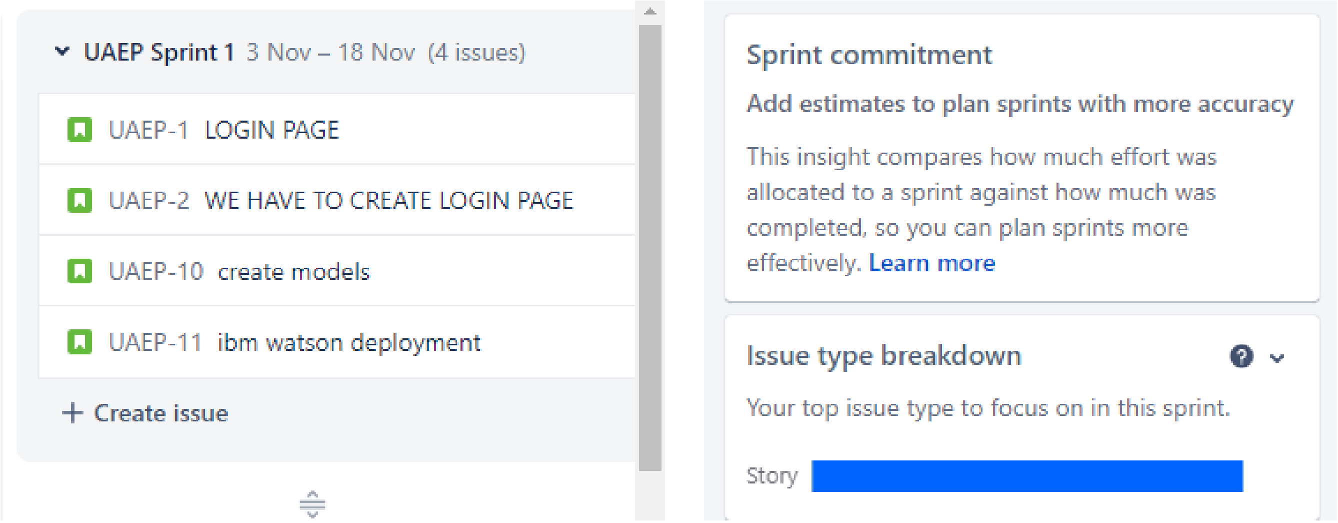
This insight compares how much effort was allocated to a sprint against how much was completed, so you can plan sprints more effectively. Learn more

Issue type breakdown

Your top issue type to focus on in this sprint.

Story

4.Image showing the working progress of the tasks in the selected Spring



UAEP Sprint 1 3 Nov – 18 Nov (4 issues)

- UAEP-1 LOGIN PAGE
- UAEP-2 WE HAVE TO CREATE LOGIN PAGE
- UAEP-10 create models
- UAEP-11 ibm watson deployment

+ Create issue

Sprint commitment

Add estimates to plan sprints with more accuracy

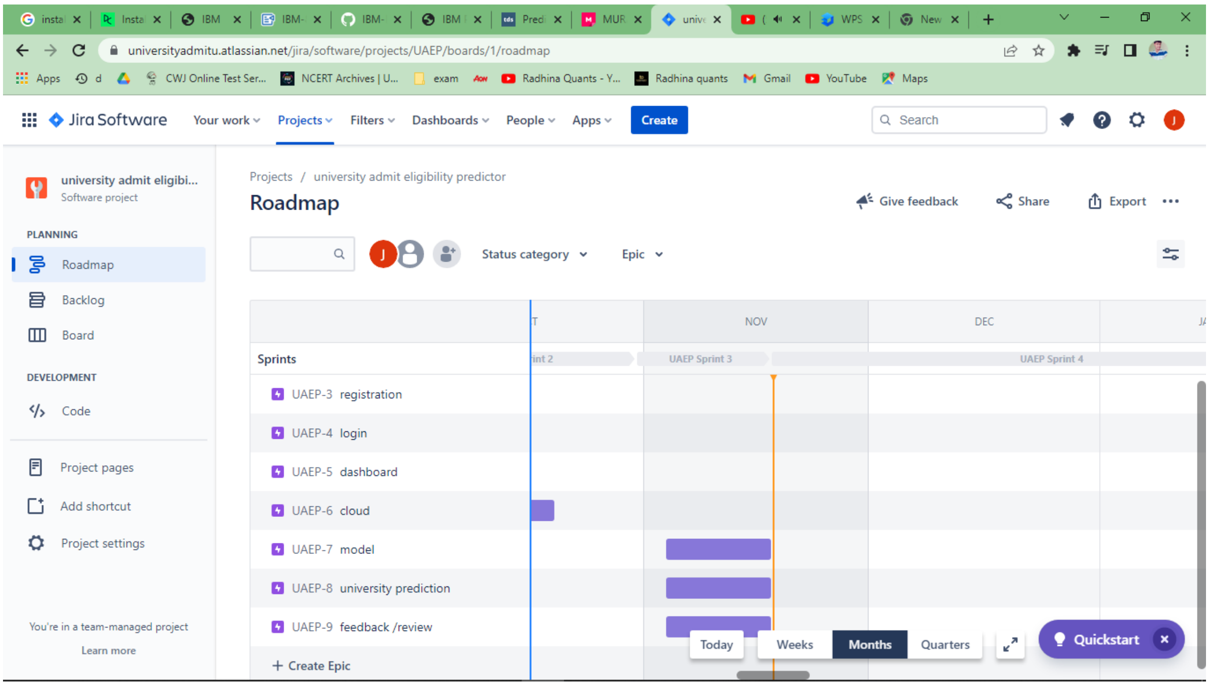
This insight compares how much effort was allocated to a sprint against how much was completed, so you can plan sprints more effectively. Learn more

Issue type breakdown

Your top issue type to focus on in this sprint.

Story

5. Image showing the Roadmap of the Project



Roadmap

Sprints

UAEP-3 registration

UAEP-4 login

UAEP-5 dashboard

UAEP-6 cloud

UAEP-7 model

UAEP-8 university prediction

UAEP-9 feedback /review

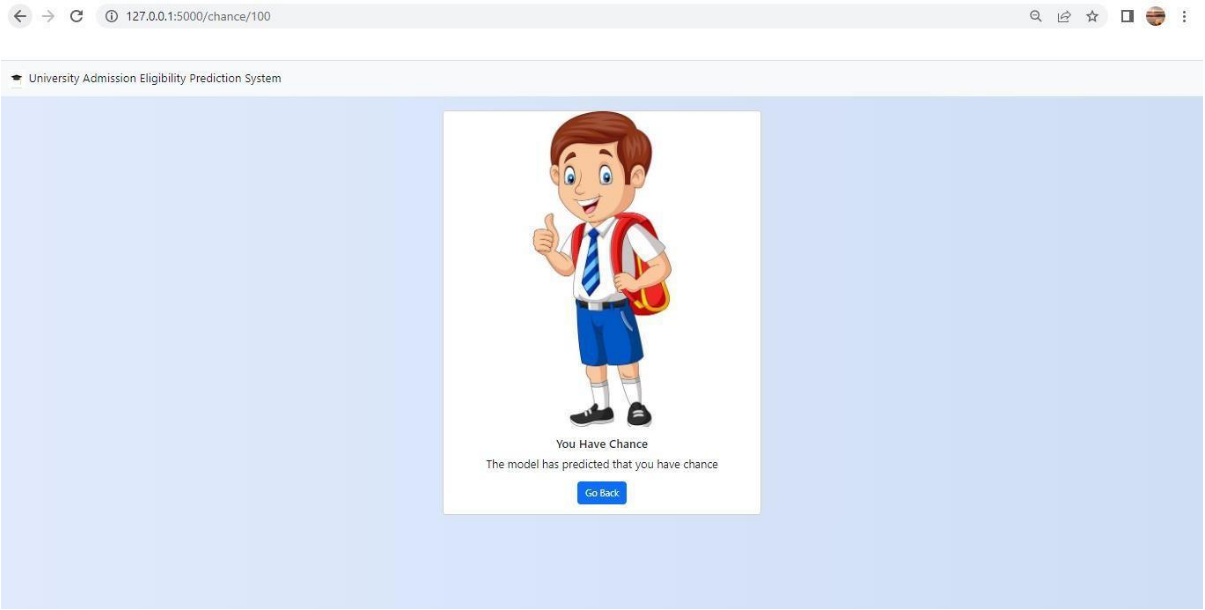
+ Create Epic

# CHAPTER 7

# CODING AND SOLUTIONING

## 7.1 FEATURE 1

The new feature will predict the chances in the admission of the university. The feature was designed in the html code connected with app.py as the backend.


127.0.0.1:5000/chance/100
University Admission Eligibility Prediction System
Go Back

You Have Chance

The model has predicted that you have chance

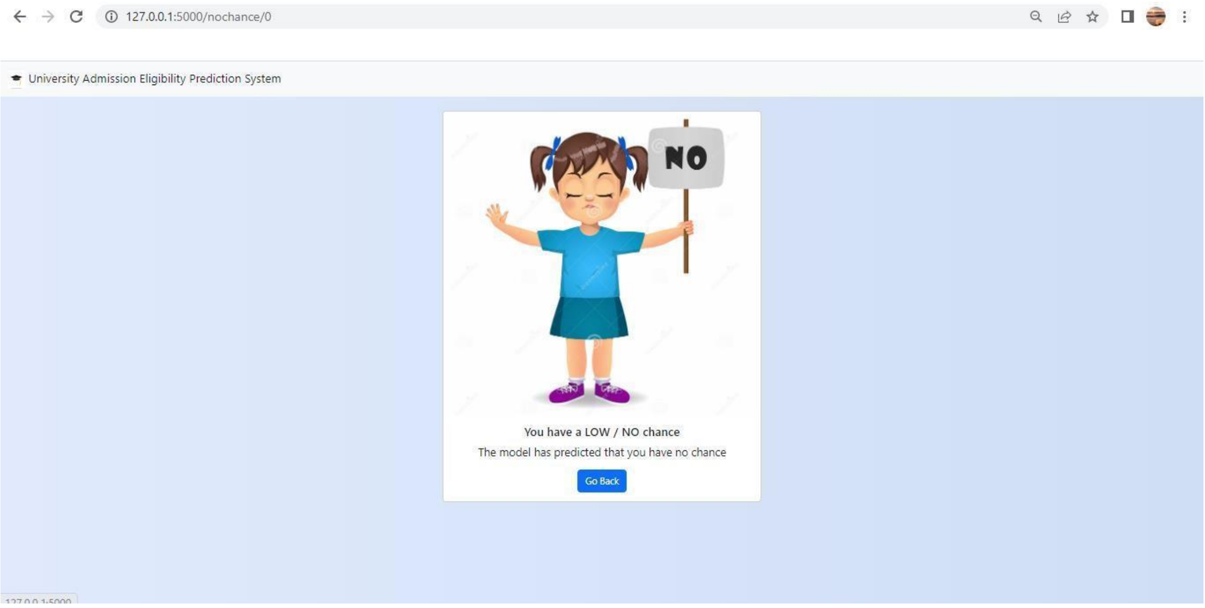
### Source Code

```
{% extends 'index.html' %}
{% block body %}
<div class="container text-center p-4">
<div class="d-flex justify-content-center">
<div class="card" style="width: 34rem;">
<img src="..\static\img\chance.png" class="card-img-top" alt="...">
<div class="card-body">
<h5 class="card-title">You Have Chance</h5>
<p class="card-text">The model has predicted that you have chance</p>
<a href="/home" class="btn btn-primary">Go Back</a>
</div>
</div>
</div>
```

```
</div>
{% endblock %}
```

## 7.2 FEATURE 2

The new feature will predict the low chances in the admission of the university. The feature was designed in the html code connected with app.py as the backend.


127.0.0.1:5000/nochance/0
University Admission Eligibility Prediction System
NO
Go Back
127.0.0.1:5000

You have a LOW / NO chance

The model has predicted that you have no chance

**Source Code**

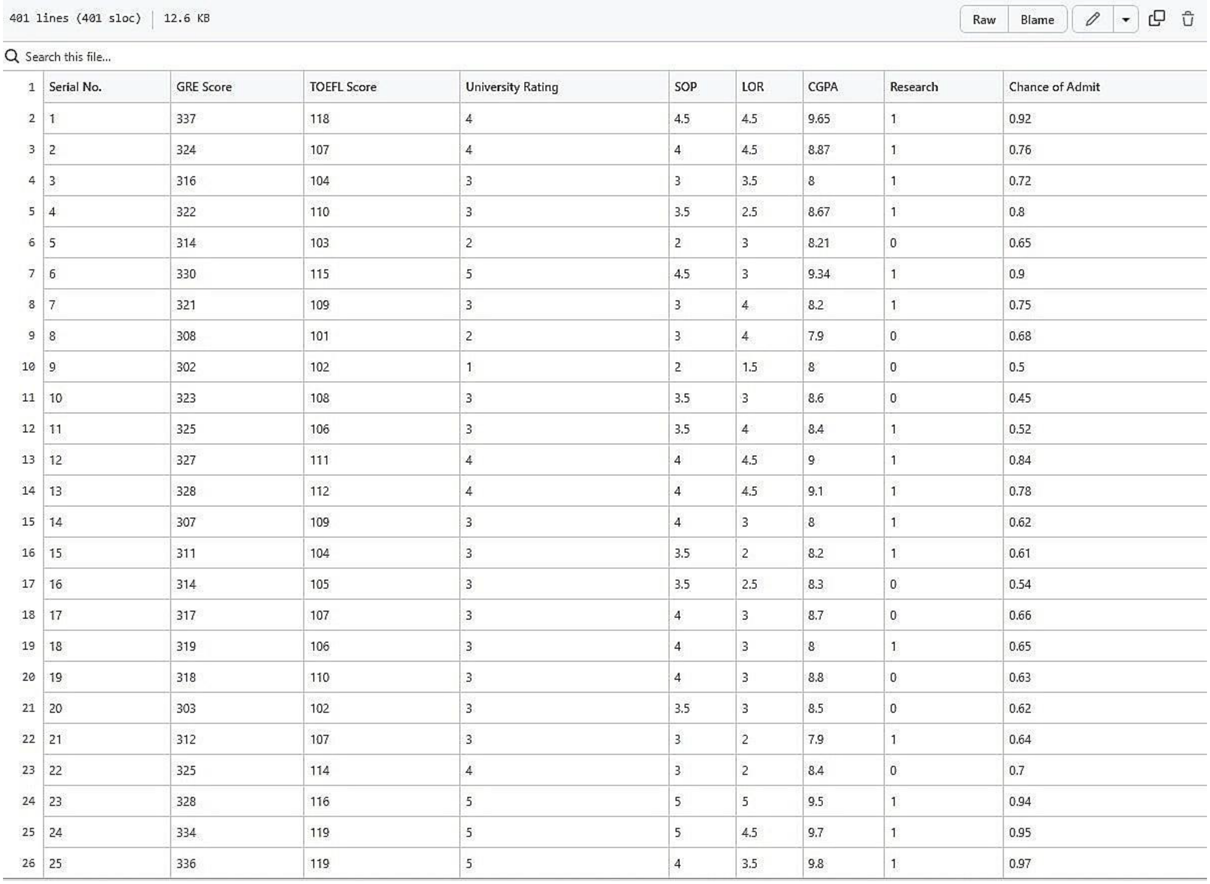
```
{% extends 'index.html' %}
{% block body %}
<div class="container text-center p-4">
<div class="d-flex justify-content-center">
<div class="card" style="width: 34rem;">
<img src="..\static\img\Nochance.jpg" class="card-img-top" alt="...">
<div class="card-body">
<h5 class="card-title">You have a LOW / NO chance</h5>
<p class="card-text">The model has predicted that you have no chance</p>
<a href="/home" class="btn btn-primary">Go Back</a>
</div>
</div>
</div>
```

</div>

{% endblock %}

## 7.3 Database Schema

The database used here in this project was Admission_Predict.csv. The sample screenshot of the database 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 |

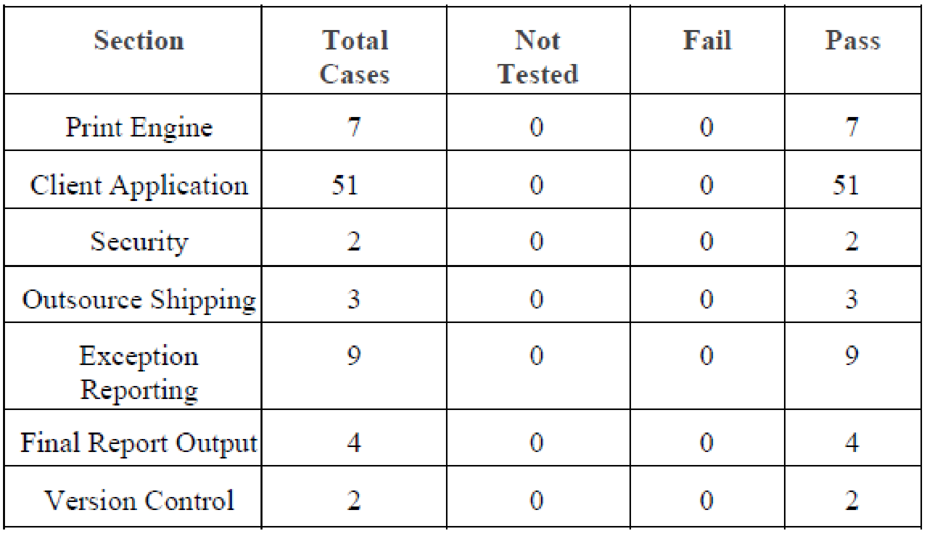
# CHAPTER 8

# 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 |

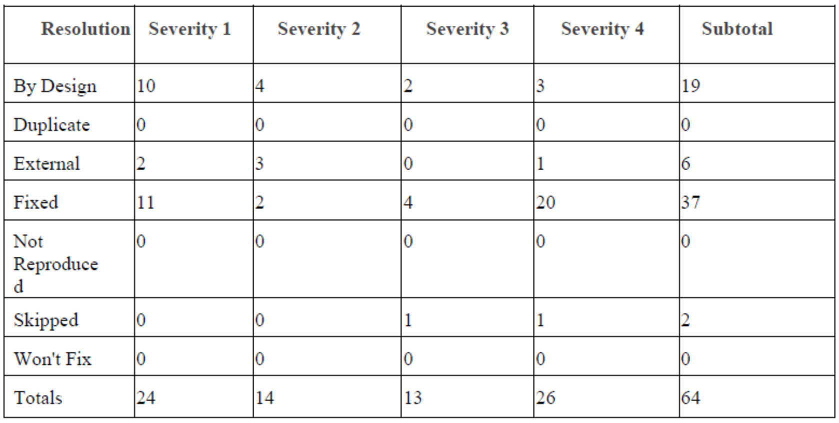
## 8.2 USER ACCEPTANCE TESTING

- **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).

- **Defect Analysis**

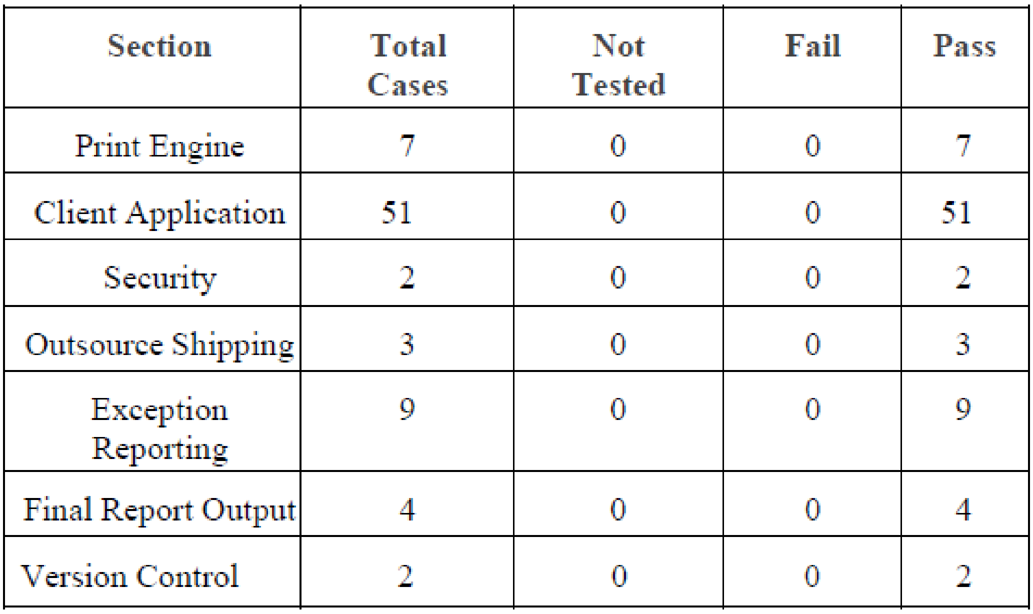
This report shows the number of resolved or closed bugs at each severity level, and how they were resolved



| 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 Reproduced | 0 | 0 | 0 | 0 | 0 |
| 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

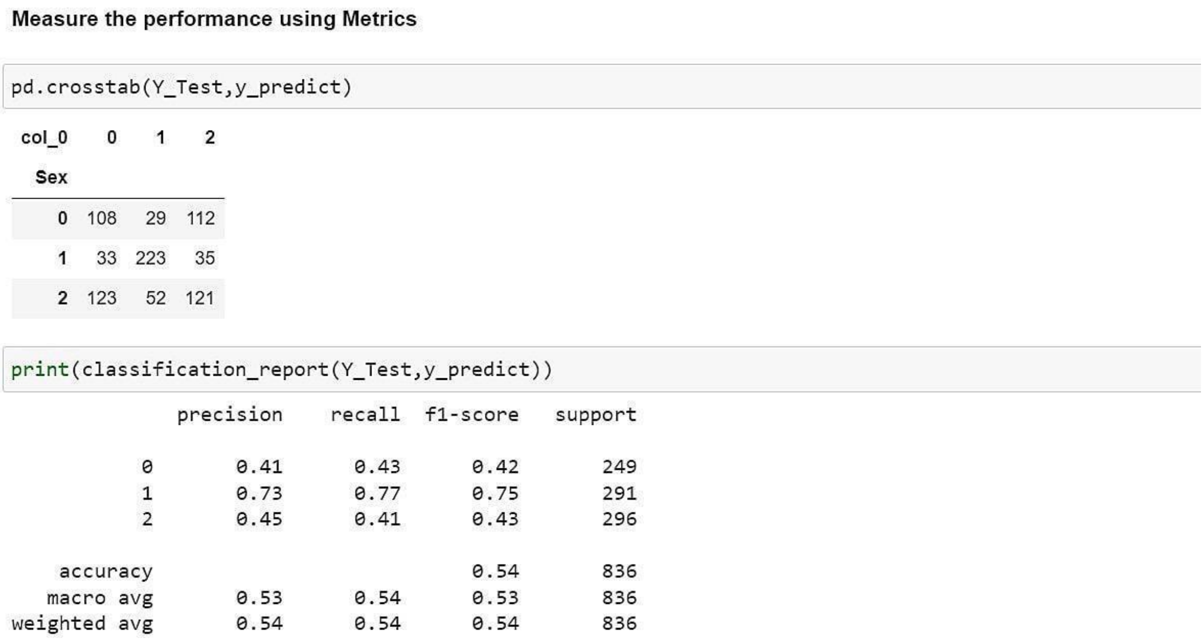


| Section | Total Cases | Not Tested | Fail | Pass |
|---|---|---|---|---|
| Print Engine | 7 | 0 | 0 | 7 |
| Client Application | 51 | 0 | 0 | 51 |
| Security | 2 | 0 | 0 | 2 |
| Outsource Shipping | 3 | 0 | 0 | 3 |
| Exception Reporting | 9 | 0 | 0 | 9 |
| Final Report Output | 4 | 0 | 0 | 4 |
| Version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

# CHAPTER 9

## 9. RESULTS

## 9.1 PERFORMANCE METRICS



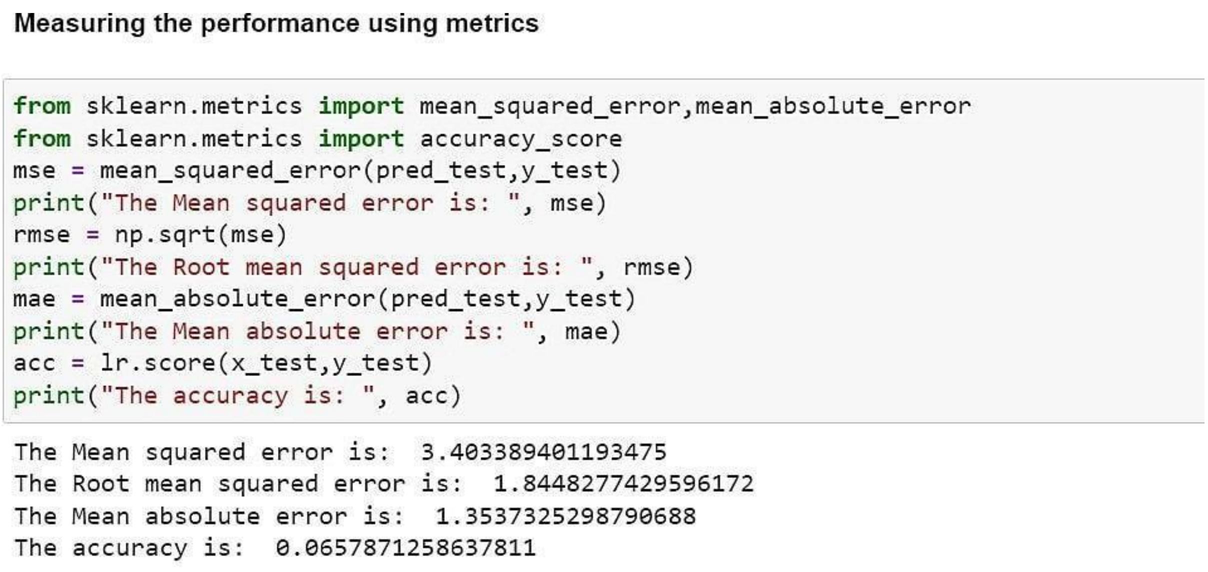
**Measure the performance using Metrics**

```
pd.crosstab(Y_Test,y_predict)
```

| col_0 | 0 | 1 | 2 |
|---|---|---|---|
| **Sex** | | | |
| **0** | 108 | 29 | 112 |
| **1** | 33 | 223 | 35 |
| **2** | 123 | 52 | 121 |

```
print(classification_report(Y_Test,y_predict))
```

```
              precision    recall  f1-score   support

           0       0.41      0.43      0.42       249
           1       0.73      0.77      0.75       291
           2       0.45      0.41      0.43       296

    accuracy                           0.54       836
   macro avg       0.53      0.54      0.53       836
weighted avg       0.54      0.54      0.54       836
```



**Measuring the performance using metrics**

```
from sklearn.metrics import mean_squared_error,mean_absolute_error
from sklearn.metrics import accuracy_score
mse = mean_squared_error(pred_test,y_test)
print("The Mean squared error is: ", mse)
rmse = np.sqrt(mse)
print("The Root mean squared error is: ", rmse)
mae = mean_absolute_error(pred_test,y_test)
print("The Mean absolute error is: ", mae)
acc = lr.score(x_test,y_test)
print("The accuracy is: ", acc)
```

```
The Mean squared error is:  3.403389401193475
The Root mean squared error is:  1.8448277429596172
The Mean absolute error is:  1.3537325298790688
The accuracy is:  0.0657871258637811
```

# CHAPTER 10

# ADVANTAGES AND DISADVANTAGES

## ADVANTAGES

- Time efficiency
- It helps student for making decision for choosing a right university.
- Here the chance of occurrence of error is less when compared with the existing system.
- It is fast, efficient and reliable.
- Avoids data redundancy and inconsistency.
- user-friendly.
- Easy accessibility of data.

## DISADVANTAGES

- Sometimes it may produce inappropriate results because of data.
- System will provide inaccurate results if data entered incorrectly.
- Requires internet

## 11 CONCLUSION

Every year millions of students apply to universities to begin their educational life. Most of them don’t have proper resources, prior knowledge and are not cautious, which in turn creates a lot of problems as applying to the wrong university, which further wastes their time, money and energy. With the help of our project, we have tried to help out such students who are finding difficulty in finding the right university for them. It is very important that a candidate should apply to university that he/she has a good chance of getting into, instead of applying to university that they may never get into. This will help in reduction of cost as students will be applying to only those universities that they are highly likely to get into. Our prepared models work to a satisfactory level of accuracy and may be of great assistance to such people. Student admission problem is very important in educational institutions. In this project addresses machine learning models to predict the chance of a student to beadmitted. This will assist students to know in advance if they have a chance to getaccepted. 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 and random forest. Experiments show that the Multiple linear 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.

## 12.FUTURE SCOPE

This is a project with good future scope, especially for students of our age group who want to pursue their higher education in their dream university. And advertisement of different Universities could be placed in a web-app to generate revenue through apps. From the proposed work we are able to identify which University we are obtaining. So, in future we can develop a representation, which gives us a list of Universities in which we can obtain admission.

## 13.1 Source Code

## 13. APPENDIX

## PYTHON CODE

Uploading the python code,

### IMPORT STATEMENTS

**In [1]:**

```
import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

%matplotlib inline
```

### LOAD THE DATASET

**In [2]:**

```
import os, types

import pandas as pd

from botocore.client import Config

import ibm_boto3

def iter (self): return 0 # @hidden_cell

# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.

# You might want to remove those credentials before you share the notebook. cos_client = ibm_boto3.client(service_name='s3',

ibm_api_key_id='T6FhPnWEPrnR91XKAfpiopbqTZ8j-gbLtjakMGexd6v0', ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token", config=Config(signature_version='oauth'), endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'university-donotdelete-pr-1ijujvyruwxy5c' object_key = 'Admission_Predict.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']

# add missing iter method, so pandas accepts body as file-like object

if not hasattr(body, " iter "): body. iter = types.MethodType( iter , body )

data = pd.read_csv(body)
```

data.head()

**Out[2]:**

| 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 |

**In [3]:**

data.drop(["Serial No."], axis=1, inplace=True)

**In [4]:**

data.describe()

**Out[4]:**

| GRE Score | TOEFL Score | University Rating | SOP | LOR | CGPA | Research | Chance of Admit | |
|---|---|---|---|---|---|---|---|---|
| **count** | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.000000 | 400.000000 |
| **mean** | 316.807500 | 107.410000 | 3.087500 | 3.400000 | 3.452500 | 8.598925 | 0.547500 | 0.724350 |
| **std** | 11.473646 | 6.069514 | 1.143728 | 1.006869 | 0.898478 | 0.596317 | 0.498362 | 0.142609 |
| **min** | 290.000000 | 92.000000 | 1.000000 | 1.000000 | 1.000000 | 6.800000 | 0.000000 | 0.340000 |
| **25%** | 308.000000 | 103.000000 | 2.000000 | 2.500000 | 3.000000 | 8.170000 | 0.000000 | 0.640000 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **50%** | 317.000000 | 107.000000 | 3.000000 | 3.500000 | 3.500000 | 8.610000 | 1.000000 | 0.730000 |
| **75%** | 325.000000 | 112.000000 | 4.000000 | 4.000000 | 4.000000 | 9.062500 | 1.000000 | 0.830000 |
| **max** | 340.000000 | 120.000000 | 5.000000 | 5.000000 | 5.000000 | 9.920000 | 1.000000 | 0.970000 |

**In [5]:**

```
data.info()
```

**Out[5]:**

```
<class 'pandas.core.frame.DataFrame'> RangeIndex: 400 entries, 0 to 399
Data columns (total 8 columns):
# Column Non-Null Count Dtype
0 GRE Score 400 non-null int64
1 TOEFL Score 400 non-null int64
2 University Rating 400 non-null int64
3 SOP 400 non-null float64
4 LOR 400 non-null float64
5 CGPA 400 non-null float64
6 Research 400 non-null int64
7 Chance of Admit 400 non-null float64 dtypes: float64(4), int64(4)

memory usage: 25.1 KB
```

**In [6]:**

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

**Out[6]:**

```
GRE Score 0
TOEFL Score 0
University Rating 0
```

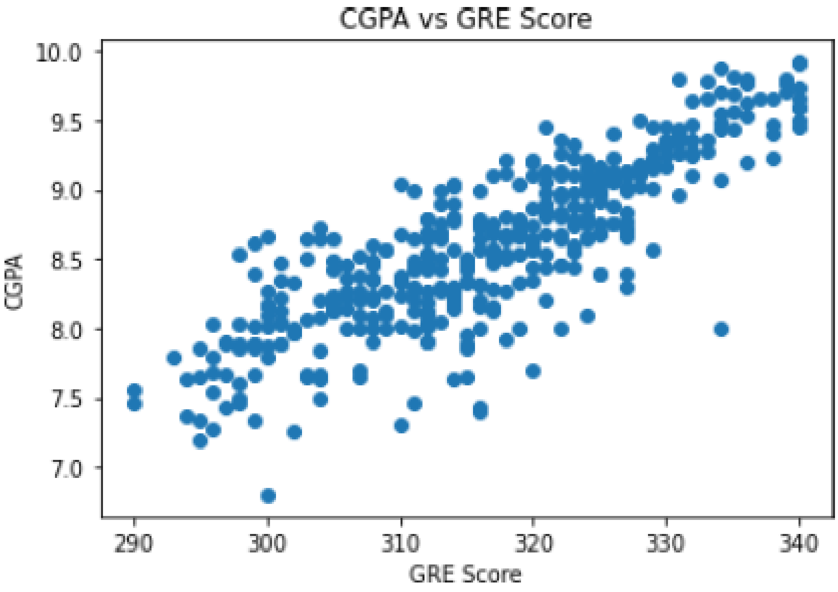
```
SOP 0

LOR 0

CGPA 0

Research 0

Chance of Admit 0

dtype: int64
```

**VISUALIZATION**

**In [7]:**

```
plt.scatter(data['GRE Score'],data['CGPA'])

plt.title('CGPA vs GRE Score')

plt.xlabel('GRE Score')

plt.ylabel('CGPA')

plt.show()
```

**Out [7]:**



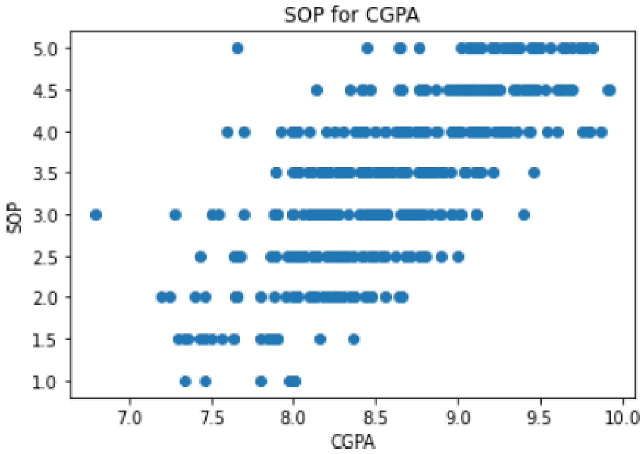
**In [8]:**

```
plt.scatter(data['CGPA'],data['SOP'])

plt.title('SOP for CGPA')

plt.xlabel('CGPA')
```

plt.ylabel('SOP')

plt.show()

**Out [9]:**



**In [9]:**

data[data.CGPA >= 8.5].plot(kind='scatter', x='GRE Score', y='TOEFL Score',color="BLUE")

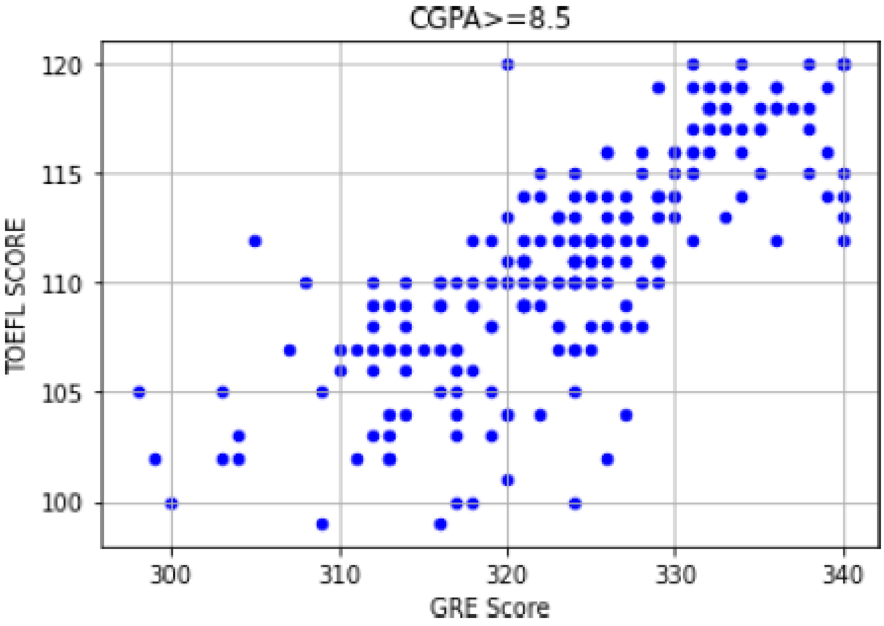
plt.xlabel("GRE Score")

plt.ylabel("TOEFL SCORE")

plt.title("CGPA>=8.5") plt.grid(True)

plt.show()

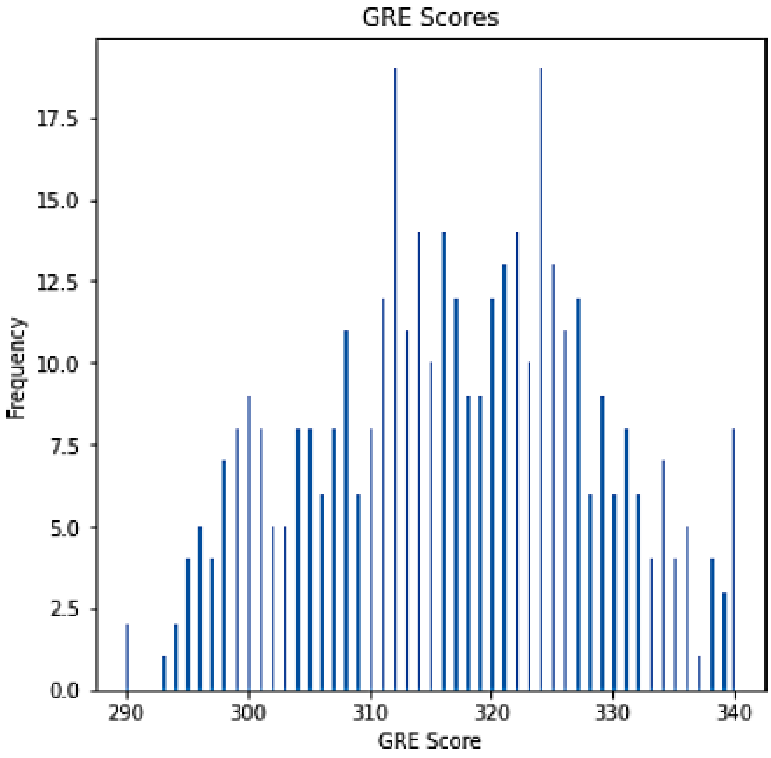
**Out [9]:**



**In [10]:**

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

plt.title("GRE Scores")

plt.xlabel("GRE Score")

plt.ylabel("Frequency") plt.show()
```

**In[11]:**



**In[11]:**

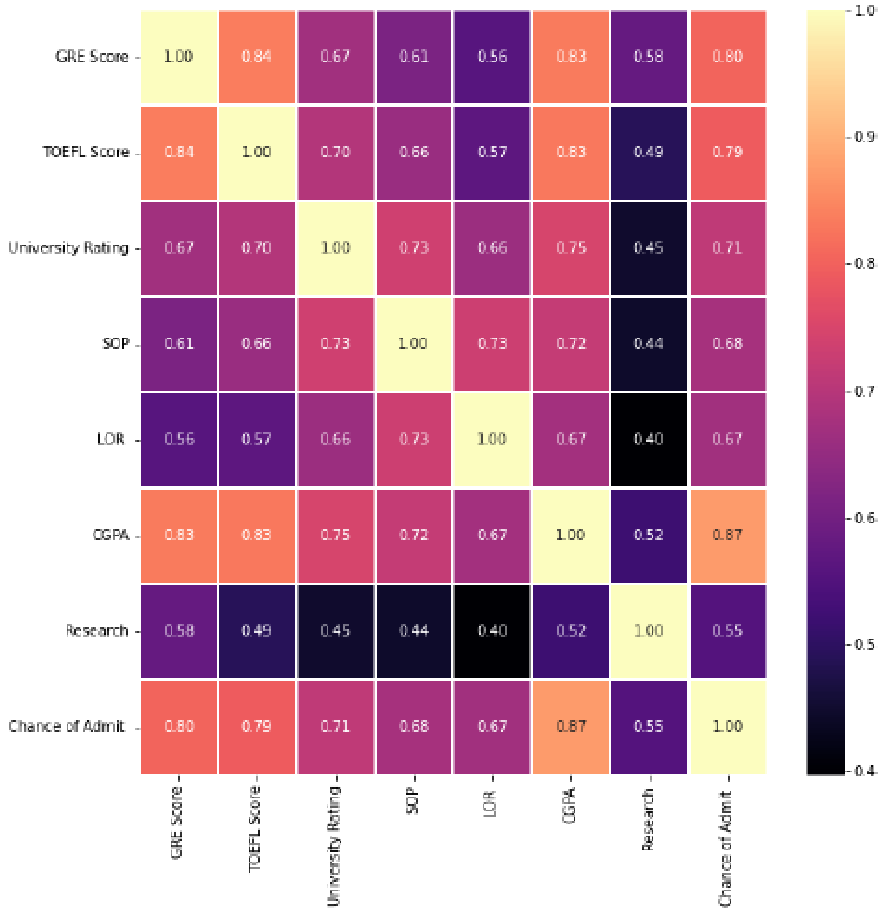
```
p = np.array([data["TOEFL Score"].min(),data["TOEFL Score"].mean(),data["TOEFL
Score"].max()])

r = ["Worst","Average","Best"]

plt.bar(p,r)

plt.title("TOEFL Scores")
plt.xlabel("Level")
plt.ylabel("TOEFL Score")

plt.show()
```

**In[12]:**

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

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

plt.show()
```

**Out[13]:**

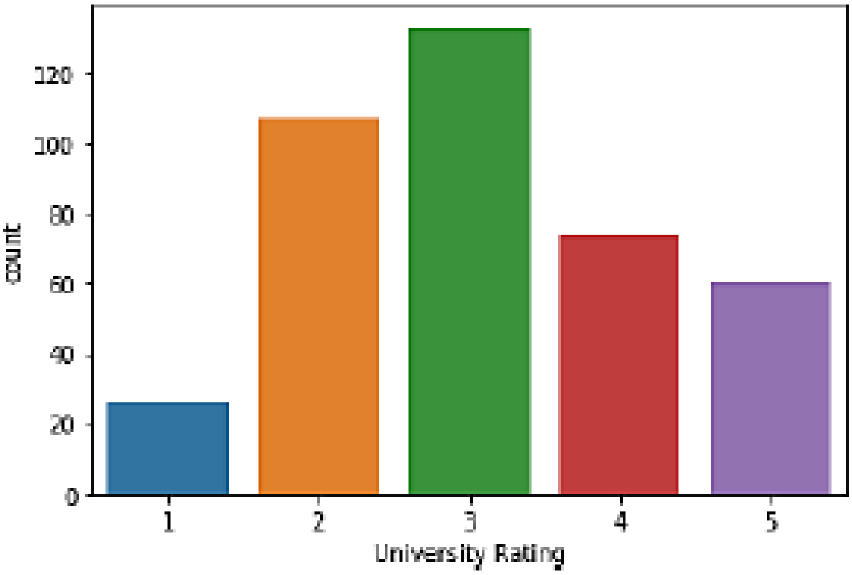


**In[14]:**

```
data.Research.value_counts()
sns.countplot(x="University Rating",data=data)
```

**Out[14]:**

```
<AxesSubplot:xlabel='University Rating', ylabel='count'>
```

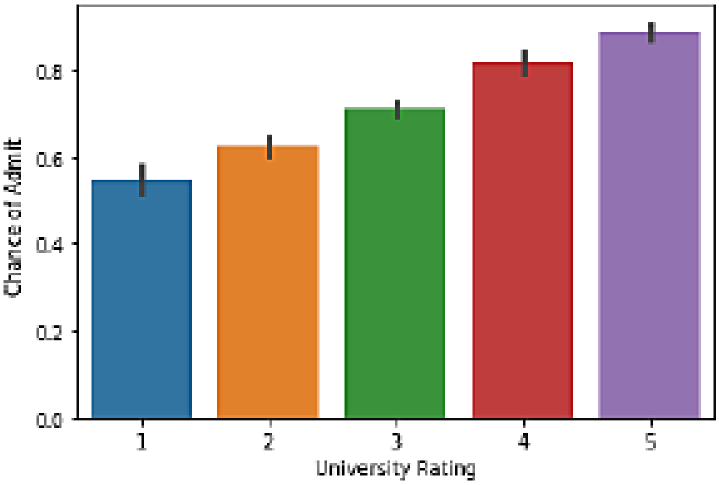


**In[15]:**

```
sns.barplot(x="University Rating", y="Chance of Admit ", data=data)
```

**Out[15]:**

```
<AxesSubplot:xlabel='University Rating', ylabel='Chance of Admit '>
```



**TRAINING AND TESTING SPLIT**

**In [16]:**

```
X=data.drop(['Chance of Admit '],axis=1) #input data_set y=data['Chance of Admit '] #output
labels
```

**In [17]:**

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15)
```

**MODELING AND TRAINING**

**In [18]:**

```
from sklearn.ensemble import GradientBoostingRegressor rgr =
GradientBoostingRegressor()

rgr.fit(X_train,y_train)
```

**Out[18]:**

```
GradientBoostingRegressor()
```

**In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook. On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.**

**In [19]:**

```
rgr.score(X_test,y_test)
```

**Out[19]:**

```
0.7214021715194154
```

**In [20]:**

```
y_predict=rgr.predict(X_test)
```

**In [21]:**

```
        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: 0.061115035673946834

Mean Squared Error: 0.007194293635482686 Root Mean Squared Error:
0.08481918200196631
```

**In [22]:**

```
y_train = (y_train>0.5) y_test = (y_test>0.5)
```

**In [23]:**

```
from sklearn.linear_model._logistic import LogisticRegression lore = LogisticRegression(random_state=0, max_iter=1000)

lr = lore.fit(X_train, y_train)
```

**In [24]:**

```
y_pred = lr.predict(X_test)
```

**In [25]:**

```
from sklearn.metrics import accuracy_score, recall_score, roc_auc_score, confusion_matrix 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))
```

**Out [25]:**

```
Accuracy Score: 0.9166666666666666

Recall Score: 1.0

ROC AUC Score: 0.7222222222222222

Confussion Matrix: [[ 4 5]

[ 0 51]]
```

**SAVING THE MODEL**

**In [26]:**

```
import pickle
```

**In [27]:**

```
pickle.dump(lr, open("university.pkl", "wb")) #logistic regression model
```

**HOSTING THE MODEL**

**In [28]:**

```
import pickle
```

**In [29]:**

```
lr = pickle.load(open("university.pkl", "rb")) #logistic regression model
```

**In [30]:**

```
pip install -U ibm-watson-machine-learning
```

**Out [30]:**

```
Requirement already satisfied: ibm-watson-machine-learning in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (1.0.255)

Collecting ibm-watson-machine-learning

Downloading ibm_watson_machine_learning-1.0.256-py3-none-any.whl (1.8 MB)

|████████████████████████████████| 1.8 MB 22.1 MB/s eta 0:00:01
Requirement already satisfied: packaging in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from ibm-watson-machine-learning) (21.3)

Requirement already satisfied: importlib-metadata in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (4.8.2)

Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from ibm-watson-machine-learning) (1.26.7)

Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from ibm-watson-machine-learning) (0.3.3)

Requirement already satisfied: pandas<1.5.0,>=0.24.2 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (1.3.4)

Requirement already satisfied: ibm-cos-sdk==2.11.* in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-watson-machine-learning) (2.11.0)

Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from ibm-watson-machine-learning) (2.26.0)

Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from ibm-watson-machine-learning) (0.8.9)

Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from ibm-watson-machine-learning) (2022.9.24)

Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning)
(2.11.0)

Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning)
(0.10.0)

Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm-watson-machine-learning)
(2.11.0)

Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*-
>ibm- watson-machine-learning) (2.8.2)
```

```
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-
3.9/lib/python3.9/site- packages (from pandas<1.5.0,>=0.24.2->ibm-watson-machine-
learning) (2021.3)

Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-
3.9/lib/python3.9/site- packages (from pandas<1.5.0,>=0.24.2->ibm-watson-machine-
learning) (1.20.3)

Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from python-dateutil<3.0.0,>=2.1->ibm-cos-sdk-core==2.11.0->ibm-cos-
sdk==2.11.*->ibm-watson-machine-learning) (1.15.0)

Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from requests->ibm-watson-machine-learning) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-
3.9/lib/python3.9/site- packages (from requests->ibm-watson-machine-learning) (3.3)

Requirement already satisfied: zipp>=0.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-
packages (from importlib-metadata->ibm-watson-machine-learning) (3.6.0)

Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/envs/Python-
3.9/lib/python3.9/site-packages (from packaging->ibm-watson-machine-learning) (3.0.4)
Installing collected packages: ibm-watson-machine-learning

Attempting uninstall: ibm-watson-machine-learning

Found existing installation: ibm-watson-machine-learning 1.0.255 Uninstalling ibm-watson-
machine-learning-1.0.255:

Successfully uninstalled ibm-watson-machine-learning-1.0.255 Successfully installed ibm-
watson-machine-learning-1.0.256

Note: you may need to restart the kernel to use updated packages.
```

**In [31]:**

```
from ibm_watson_machine_learning import APIClient import json
```

**In [32]:**

```
uml_credentials = {

"url": "https://us-south.ml.cloud.ibm.com",

"apikey": "poJ22ua6BCG9qY33B8fkgnz1bnP1f9DZqUlF9NkBM1bZ"

}

client = APIClient(uml_credentials)
```

**In [33]:**

```
def guid_from_space_name(client, space_name): space = client.spaces.get_details()

idr = []

for i in space['resources']: idr.append(i['metadata']['id'])
```

```
return idr
```

**In [34]:**

```
space_uid = guid_from_space_name(client, "university") print(space_uid[0])
```

```
4f0253e5-f162-4eec-84ba-72e01fb69ab9
```

**In [35]:**

```
client.set.default_space(space_uid[0])
```

**Out[35]:**

```
'SUCCESS'
```

**In [36]:**

```
client.software_specifications.list()
```

```
NAME ASSET_ID TYPE

default_py3.6 0062b8c9-8b7d-44a0-a9b9-46c416adcbd9 base kernel-spark3.2-scala2.12
020d69ce-7ac1-5e68-ac1a-31189867356a base pytorch-onnx_1.3-py3.7-edt 069ea134-3346-
5748-b513-49120e15d288 base scikit-learn_0.20-py3.6 09c5a1d0-9c1e-4473-a344-
eb7b665ff687 base spark-mllib_3.0-scala_2.12 09f4cff0-90a7-5899-b9ed-1ef348aebdee base
pytorch-onnx_rt22.1-py3.9 0b848dd4-e681-5599-be41-b5f6fccc6471 base ai-function_0.1-
py3.6 0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda base shiny-r3.6 0e6e79df-875e-4f24-8ae9-
62dcc2148306 base

tensorflow_2.4-py3.7-horovod 1092590a-307d-563d-9b62-4eb7d64b3f22 base pytorch_1.1-
py3.6 10ac12d6-6b30-4ccd-8392-3e922c096a92 base tensorflow_1.15-py3.6-ddl 111e41b3-
de2d-5422-a4d6-bf776828c4b7 base runtime-22.1-py3.9 12b83a17-24d8-5082-900f-
0ab31fbfd3cb base scikit-learn_0.22-py3.6 154010fa-5b3b-4ac1-82af-4d5ee5abbc85 base
default_r3.6 1b70aec3-ab34-4b87-8aa0-a4a3c8296a36 base

pytorch-onnx_1.3-py3.6 1bc6029a-cc97-56da-b8e0-39c3880dbbe7 base kernel-spark3.3-r3.6
1c9e5454-f216-59dd-a20e-474a5cdf5988 base pytorch-onnx_rt22.1-py3.9-edt 1d362186-
7ad5-5b59-8b6c-9d0880bde37f base tensorflow_2.1-py3.6 1eb25b84-d6ed-5dde-b6a5-
3fbdf1665666 base spark-mllib_3.2 20047f72-0a98-58c7-9ff5-a77b012eb8f5 base
tensorflow_2.4-py3.8-horovod 217c16f6-178f-56bf-824a-b19f20564c49 base runtime-22.1-
py3.9-cuda 26215f05-08c3-5a41-a1b0-da66306ce658 base do_py3.8 295addb5-9ef9-547e-
9bf4-92ae3563e720 base

autoai-ts_3.8-py3.8 2aa0c932-798f-5ae9-abd6-15e0c2402fb5 base tensorflow_1.15-py3.6
2b73a275-7cbf-420b-a912-eae7f436e0bc base

kernel-spark3.3-py3.9 2b7961e2-e3b1-5a8c-a491-482c8368839a base pytorch_1.2-py3.6
2c8ef57d-2687-4b7d-acce-01f94976dac1 base spark-mllib_2.3 2e51f700-bca0-4b0d-88dc-
5c6791338875 base pytorch-onnx_1.1-py3.6-edt 32983cea-3f32-4400-8965-dde874a8d67e
base spark-mllib_3.0-py37 36507ebe-8770-55ba-ab2a-eafe787600e9 base spark-mllib_2.4
390d21f8-e58b-4fac-9c55-d7ceda621326 base xgboost_0.82-py3.6 39e31acd-5f30-41dc-
ae44-60233c80306e base pytorch-onnx_1.2-py3.6-edt 40589d0e-7019-4e28-8daa-
```

```
fb03b6f4fe12 base default_r36py38 41c247d3-45f8-5a71-b065-8580229facf0 base autoai-
ts_rt22.1-py3.9 4269d26e-07ba-5d40-8f66-2d495b0c71f7 base autoai-obm_3.0 42b92e18-
d9ab-567f-988a-4240ba1ed5f7 base pmml-3.0_4.3 493bcb95-16f1-5bc5-bee8-81b8af80e9c7
base spark-mllib_2.4-r_3.6 49403dff-92e9-4c87-a3d7-a42d0021c095 base xgboost_0.90-
py3.6 4ff8d6c2-1343-4c18-85e1-689c965304d3 base pytorch-onnx_1.1-py3.6 50f95b2a-
bc16-43bb-bc94-b0bed208c60b base autoai-ts_3.9-py3.8 52c57136-80fa-572e-8728-
a5e7cbb42cde base spark-mllib_2.4-scala_2.11 55a70f99-7320-4be5-9fb9-9edb5a443af5
base spark-mllib_3.0 5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9 base

autoai-obm_2.0 5c2e37fa-80b8-5e77-840f-d912469614ee base spss-modeler_18.1 5c3cad7e-
507f-4b2a-a9a3-ab53a21dee8b base cuda-py3.8 5d3232bf-c86b-5df4-a2cd-7bb870a1cd4e
base autoai-kb_3.1-py3.7 632d4b22-10aa-5180-88f0-f52dfb6444d7 base pytorch-onnx_1.7-
py3.8 634d3cdc-b562-5bf9-a2d4-ea90a478456b base spark-mllib_2.3-r_3.6 6586b9e3-ccd6-
4f92-900f-0f8cb2bd6f0c base tensorflow_2.4-py3.7 65e171d7-72d1-55d9-8ebb-
f813d620c9bb base spss-modeler_18.2 687eddc9-028a-4117-b9dd-e57b36f1efa5 base

Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

**In [37]:**

```
import sklearn sklearn. version
```

**Out[37]:**

```
'1.0.2'
```

**In [38]:**

```
MODEL_NAME = 'university' DEPLOYMENT_NAME = 'uni' DEMO_MODEL = lr
```

**In [39]:**

```
software_spec_uid = client.software_specifications.get_id_by_name('runtime-22.1-py3.9')
```

**In [40]:**

```
model_props = {

client.repository.ModelMetaNames.NAME: MODEL_NAME,
client.repository.ModelMetaNames.TYPE: 'scikit-learn_1.0 ',
client.repository.ModelMetaNames.SOFTWARE_SPEC_UID: software_spec_uid

}
```

**In [41]:**

```
model_details = client.repository.store_model( model = DEMO_MODEL,

meta_props = model_props, training_data = X_train, training_target = y_train

)

model_details
```

**Out[41]:**

```
{'entity': {'hybrid_pipeline_software_specs': [], 'label_column': 'Chance of Admit ',

'schemas': {'input': [{'fields': [{'name': 'GRE Score', 'type': 'int64'},

{'name': 'TOEFL Score', 'type': 'int64'},

{'name': 'University Rating', 'type': 'int64'},

{'name': 'SOP', 'type': 'float64'},

{'name': 'LOR ', 'type': 'float64'},

{'name': 'CGPA', 'type': 'float64'},

{'name': 'Research', 'type': 'int64'}], 'id': '1',

'type': 'struct'}],

'output': []},

'software_spec': {'id': '12b83a17-24d8-5082-900f-0ab31fbfd3cb', 'name': 'runtime-22.1-py3.9'},

'type': 'scikit-learn_1.0'},

'metadata': {'created_at': '2022-11-03T05:20:49.371Z', 'id': '566cfcae-49ab-4bd3-b5df-
abc981fa27b9', 'modified_at': '2022-11-03T05:20:51.730Z',

'name': 'university',

'owner': 'IBMid-6630041JHH',

'resource_key': 'a61934d2-41d0-413d-9f54-49589e7c7741', 'space_id': '4f0253e5-f162-4eec-
84ba-72e01fb69ab9'}, 'system': {'warnings': []}}
```

**In [42]:**

```
model_id = client.repository.get_model_id(model_details) model_id
```

**Out[42]:**

```
'566cfcae-49ab-4bd3-b5df-abc981fa27b9'
```

**In [43]:**

```
deployment_props = {
client.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_NAME,
client.deployments.ConfigurationMetaNames.ONLINE: {}

}

deployment = client.deployments.create( artifact_uid = model_id,

meta_props = deployment_props

)
```

**Out [43]:**

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

```
Synchronous deployment creation for uid: '566cfcae-49ab-4bd3-b5df-abc981fa27b9' 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='28aea4f7-0bec-4310-82bf-06e502d2cd4d'
```

## HTML CODES

Uploading Html codes

## Chance.html

```
{% extends 'index.html' %}

{% block body %}

<div class="container text-center p-4">

<div class="d-flex justify-content-center">

<div class="card" style="width: 34rem;">

<img src="..\static\img\chance.png" class="card-img-top" alt="...">

<div class="card-body">

<h5 class="card-title">You Have Chance</h5>

<p class="card-text">The model has predicted that you have chance</p>

<a href="/home" class="btn btn-primary">Go Back</a>

</div>

</div>

</div>

</div>

{% endblock %}
```

## Demo2.html

```
{% extends 'index.html' %}

{% block body %}

<div class="p-4">

<div class="row mb-3">

<div class="col-4">
```

```html
<h2 class="text-responsive-h">
Enter your details and get probability of your admission
</h2>
<p class="text-responsive">
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.
</p>
<div class="d-flex justify-content-right">
<img src="../static/img/anime.png" class="card-img-top" alt="..." />
</div>
</div>
<div class="col-8">
<div class="card p-2 ms-2 my-2">
<div class="card-body">
<h5 class="card-title pb-4"> Enter the details
</h5>
<form action="/" method="post" 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" 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" required>
</div>
</div>
```

```
<div class="row mb-3">

Rating:</label>

<label for="university_rating" class="col-lg-2 col-form-label">University

<div class="col-lg-10">

<input type="number" class="form-control" id="university_rating"

step="0.01" name="university_rating" min="1" max="5" required>

</div>

</div>

<div class="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"
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"
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" 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">
<div class="col-lg-2 mb-2 me-3">
<button type="submit" class="btn btn-primary" id="button">Predict</button>
</div>
<div class="col-lg-2" id="spinner">
<div class="spinner-border text-primary m-1" role="status">
<span class="visually-hidden">Loading...</span>
</div>
<div class="spinner-grow text-primary m-1" role="status">
<span class="visually-hidden">Loading...</span>
</div>
</div>
</form>
</div>
</div>
</div>
</div>
</div>
```

```
{% endblock %}
```

## 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">
<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">
<link rel="stylesheet" type="text/css" rel="noopener" target="_blank" href="../static/css/styles.css">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">
<script type="text/javascript" src="../static/js/script.js" async></script>
<title>University Admit Eligibility Predictor</title>
</head >
<body>
<nav class="navbar navbar-expand-lg bg-light">
<div class="container-fluid">
<a class="navbar-brand text-responsive-h" href="/">
<img src="..\static\img\hat1.png" alt="Logo" width="30" height="24" class="d-inline- block align-text-top ">
University Admission Eligibility Prediction System
</a>
</div>
</nav>
{% block body %}
<h1> Index Page </h1>
{% endblock %}
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Qbsw3"
crossorigin="anonymous"></script>

</body>

</html>
```

## Nochance.html

```
{% extends 'index.html' %}

{% block body %}

<div class="container text-center p-4">

<div class="d-flex justify-content-center">

<div class="card" style="width: 34rem;">

<img src="..\static\img\Nochance.jpg" class="card-img-top" alt="...">

<div class="card-body">

<h5 class="card-title">You have a LOW / NO chance</h5>

<p class="card-text">The model has predicted that you have no chance</p>

<a href="/home" class="btn btn-primary">Go Back</a>

</div>

</div>

</div>

</div>

{% endblock %}
```

## Script.js

```
const button = document.getElementById('button');

const theForm = document.getElementById('theForm');

 const loading = document.getElementById('spinner');

const disableButton = () => {

console.log('Submitting form...');

button.disabled = true;

button.className = "btn btn-outline-primary";

button.innerHTML = "Predicting..." loading.style.display = "block"

};
```

```
const enableButton = () => {
 console.log('Loading window...');
 button.disabled = false;

button.className = "btn btn-primary" button.innerHTML = "Predict" loading.style.display = "none"

}

theForm.onsubmit = disableButton; window.onload = enableButton;
```

## Styles.css

```
* {

margin: 0;

padding: 0;

border: 0;

}

body {

font: 62.5%/1.5 "Lucida Grande", "Lucida Sans", Tahoma, Verdana, sans-serif; background: #e0eafc;

background: -webkit-linear-gradient(to right, #e0eafc, #cfdef3); background: linear-gradient(to right, #e0eafc, #cfdef3);

color: #000000; text-align:center;

}

h1 {

font-size: 2.2em;

}

h2 {

font-size: 2.0em;

}

h4 {

font-size: 1.6em;

}

p {

font-size: 1.2em;

}

input.text
```

```
{
padding: 3px;
border: 1px solid #999999;
}
img {
max-width: auto; height: auto;
}
.text-responsive {
font-size: calc(50% + 0.6vw + 0.6vh);
}


.text-responsive-h {
font-size: calc(80% + 0.6vw + 0.6vh);
}
```

Footer

# REQUIRED IMAGES IN THE HTML CODES

## SOURCE CODE

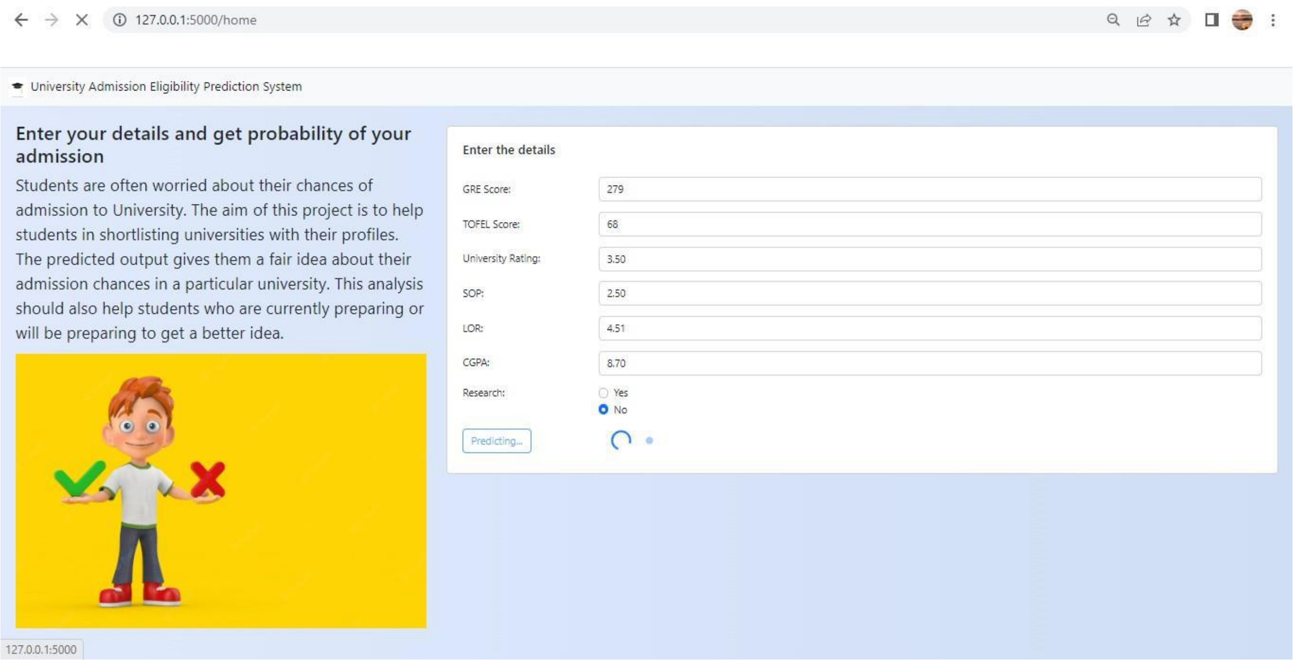
## PYTHON CODE

### App.py

```
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("demo2")) arr.append(float(val))

# deepcode ignore HardcodedNonCryptoSecret: <please specify a reason of ignoring this>
API_KEY = "poJ22ua6BCG9qY33B8fkgnz1bnP1f9DZqUlF9NkBM1bZ" 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/28aea4f7-0bec-4310-82bf-06e502d2cd4d/predictions?version=2022-11-03',
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("demo2"))
@app.route("/home") def demo2():
return render_template("demo2.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("demo2"))

if name == " main ": app.run()
```

# OUTPUT :



127.0.0.1:5000/home

University Admission Eligibility Prediction System

**Enter your details and get probability of your admission**

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.

**Enter the details**

GRE Score: 279

TOFEL Score: 68

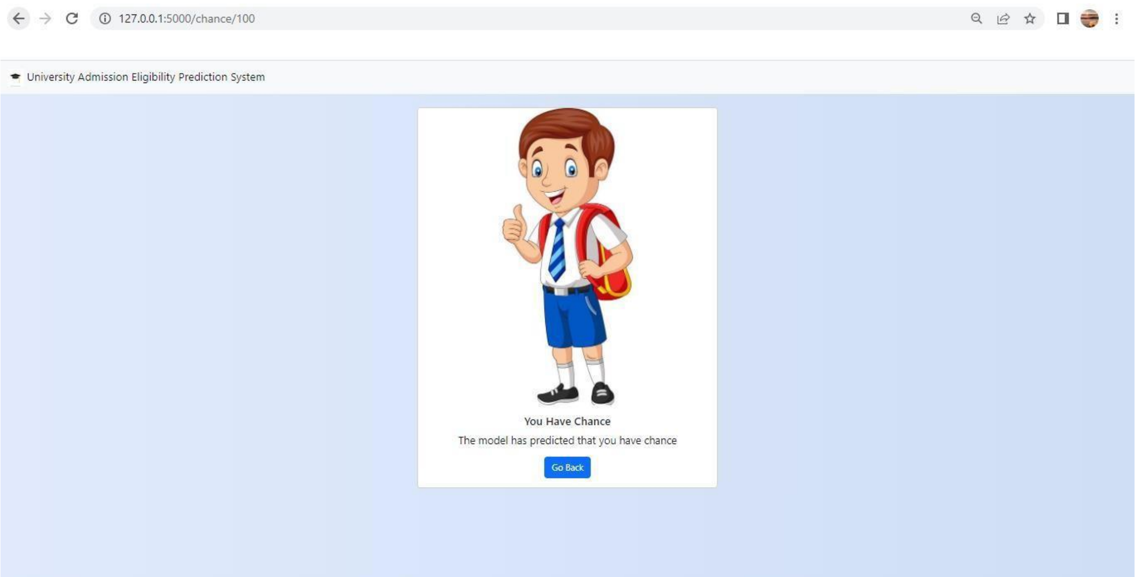
University Rating: 3.50

SOP: 2.50

LOR: 4.51

CGPA: 8.70

Research: ○ Yes ● No



127.0.0.1:5000/chance/100

University Admission Eligibility Prediction System

You Have Chance

The model has predicted that you have chance

### 13.2 GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-8373-1658916650

### 13.3 PROJECT DEMO LINK:

**https://www.youtube.com/watch?v=4T4lSDYnOD0&feature=youtu.be**