

**PROJECT BASED EXPERIENTIAL LEARNING  
PROGRAM (NALAIYA THIRAN)**

**University Admit Eligibility Predictor**

**A PROJECT REPORT**

**Submitted By**

**Priyadarshan S (312419104099)**

**Lenus M Leonard (312419104074)**

**Kishore D (312419104071)**

**Mohamed Sheik Abdul Kader M (312419104083)**

**Team Id : PNT2022TMID28129**

*in partial fulfilment for the requirement of award of the degree of*

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**



**St. JOSEPH'S INSTITUTE OF TECHNOLOGY**

**CHENNAI - 119**

**ANNA UNIVERSITY : CHENNAI 600 025**

**NOVEMBER-2022**

## INDEX

S.No.	Title	Page No.
1	Introduction	3
2	Literature Survey	4
3	Ideation & Proposed Solution	6
4	Requirement Analysis	9
5	Project Design	12
6	Project Planning & Scheduling	14
7	Coding & Solutioning	16
8	Testing	18
9	Results	20
10	Advantages & Disadvantages	23
11	Conclusion	24
12	Future Scope	25
13	Appendix	25

# 1. INTRODUCTION

## 1.1. Project Overview:

This project University Admission Predictor System is a web based application in which students can register with their personal as well as marks details for prediction the admission in colleges and the administrator can allot the seats for the students. Administrator can add the college details and he batch details. Using this software, the entrance seat allotment became easier and can be implemented using system. The main advantage of the project is the computerization of the entrance seat allotment process. Administrator has the power for the allotment. He can add the allotted seats into a file and the details are saved into the system. The total time for the entrance allotment became lesser and the allotment process became faster.

## 1.2. Purpose:

It helps students to get the list of colleges to which they can apply as the system shortlists the colleges by comparing the student's marks and college's cut off.

**Admin** manages all the system related task such as adding college details, specifying cut-off details, managing college details, viewing student's details and viewing feedback from registered students.

**Student** can register themselves using web portal by entering their personal as well as marks details. Once registration is successful, they can login and view colleges allotted based on their marks.

## **2. LITERATURE SURVEY**

### **2.1. Existing Problem:**

Helps the student by giving them an idea about their future college by calculating their cut off. The students can very easily obtain the list of colleges even branch wise and course wise. This will not only make the admission process easy but also minimizes stress for students. The main objective of our system is to make the right choice of colleges. It helps student for making decision for choosing a right college. Here the chance of occurrence of error is less when compared with the existing system. It is fast, efficient and reliable.

### **2.2. References:**

#### **1. Predicting Academic outcomes.**

-Sarah Alturki, Ioana Hulpus

#### **2. Prediction for University Admission using Machine Learning**

-Chithra Apoorva D A, Malepati ChanduNath, Peta Rohith, Bindu Shree.S, Swaroop.S

#### **3. A comparison of Regression Models for Prediction of Graduate Admissions.**

-S.Mohan Acharya.

#### **4. A Data Mining Application In A Student Database**

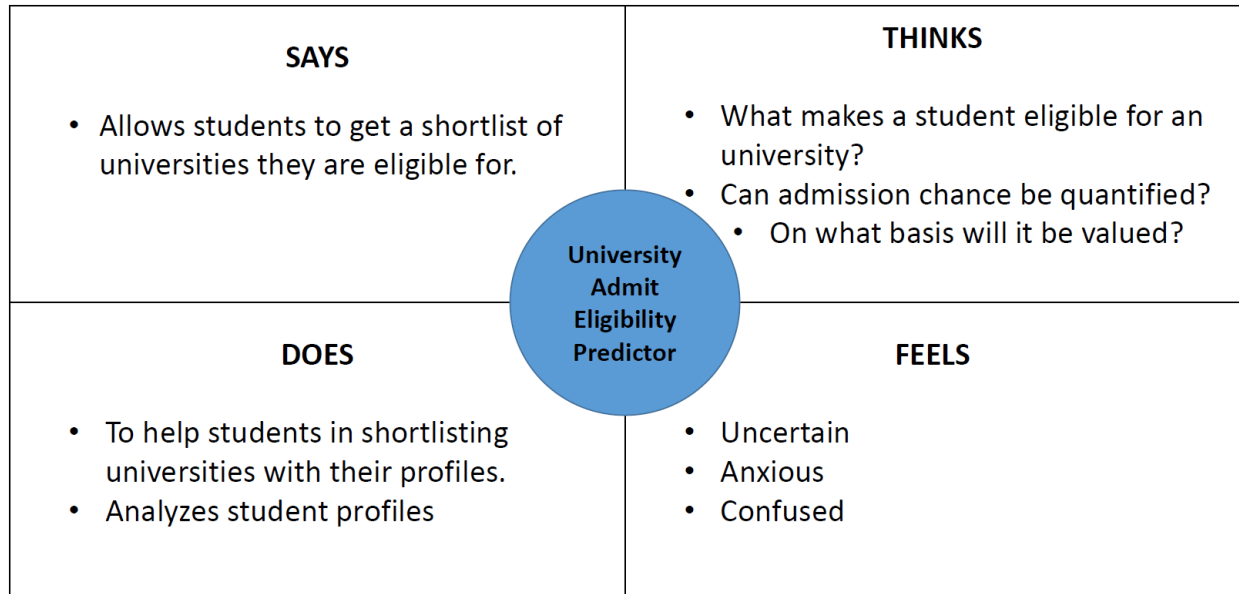
-Senol Zafer Erdogan, Mehpare Timor

### **2.3. Problem Statement Definition:**

Choosing the right universities or colleges is a problem students definitely have to face. Many students apply for the universities in which they have little chance of acceptance. This leads students of poor economic backgrounds to frustration and anxiety as they only lose surplus amount of money just for applying to those universities. The drawback of the currently available resources is that they are very limited and also they are not truly dependable taking into consideration of their accuracy and reliability. The aim of this project is to develop a system for Student Admission Predictor. Which will help the students to identify the chances of their application to a university being accepted.

### 3. IDEATION & PROPOSED SOLUTION

#### 3.1 Empathy Map Canvas:



#### 3.2 Ideation & Brainstorming:

The three best ideas, based on feasibility and importance:

- Using K-Nearest Neighbours Algorithm: It is an algorithm which is used widely for classification and regression problems. Due to its simplicity and effectiveness, it is easy to implement and understand. Distance is calculated between the unseen data sample and the all other data samples already present in the data-set. Depending on the value of K, that many nearest neighbours are selected and their class is identified. Multiple values of K should be tried and tested, and the value of K at which best performance is observed must be selected for the model. This will help find the best university option.

- Using Logistic Regression: Logistic regression algorithm is used to identify the probability of occurrence of an event based on single predictor variable.

Multivariate Logistic regression can be used to determine the probability of the occurrence of an event based on multiple predictor variables. Logistic Regression is also a supervised machine learning algorithm which used data with predetermined classes to create a model and perform predictive analysis on unseen data.

- Using Decision Tree: It is a supervised machine learning algorithm. Due to its simple logic, effectiveness and interoperability it is the most widely used classification algorithm. The model works by creating a tree-like structure by dividing the data-set into several smaller subsets based on different conditional logic. The main components of the decision tree are the decision nodes, leaf nodes and the branches. Different university choices can be different nodes and one can be picked.

### **3.3 Proposed Solution:**

University and College research being one part of the university application process is itself an arduous and lengthy task. This issue being a big problem for students has not been solved till now. There are recognized sites which filter the best universities and colleges based on the location, tuition fees, major and degree but none of them have used machine learning algorithms to solve the issue. Hence, we have done this research project to solve that issue to some extent with the use of data mining techniques.

### 3.4 Problem Solution fit:

#### PROBLEM – SOLUTION FIT

Define CS, fit into	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> <ul style="list-style-type: none"><li>• Students who have completed their under graduation.</li><li>• Students who have completed their high school.</li><li>• The students who have undertaken exams like GRE and TOEFL exams.</li></ul>	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span> <ul style="list-style-type: none"><li>• Searching the right and best-suitable college from the wide range of options of colleges that are available for admissions.</li><li>• Reduce cost incurred to travel or communicate with respective universities</li><li>• Getting admission at the preferred location.</li></ul>	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> <ul style="list-style-type: none"><li>• They lack essential criteria that needs to be considered while predicting the feasibility of getting admission in the desired university.</li><li>• Lacks dynamic nature and scalability.</li><li>• Incomplete information.</li><li>• Absence of powerful ideas like polynomial and logistic regression and other algorithms.</li></ul>	Explore AS.
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> <ul style="list-style-type: none"><li>• The major task is to design a university admission prediction system and to provide a probabilistic insight into the university rating, cutoffs, intake count and the students' university preferences.</li><li>• The students are to be provided with a list of universities which match their eligibility.</li><li>• The system must do the aforementioned tasks effectively as well as efficiently.</li></ul>	<b>9. PROBLEM ROOT CAUSE</b> <span>RC</span> <ul style="list-style-type: none"><li>• There may not be a place where the students can find all the admission related information of the universities.</li><li>• The students may not be aware of the eligibility criteria of various universities in and around the world.</li><li>• The agents may use untrustworthy information.</li></ul>	<b>7. BEHAVIOUR</b> <span>BE</span> <ul style="list-style-type: none"><li>• Direct: The students will try to visit all the universities that he/ she wishes to get admission. Get notified about the criteria to get admission and also take necessary measures to meet the criteria.</li><li>• Indirect: Pay for an agency that helps the students to find the required criteria in the desired universities and visit only those selective universities and get the job done.</li></ul>	
Focus on J&P, tap into BE, understand	<b>3. TRIGGERS</b> <span>TR</span> <ul style="list-style-type: none"><li>• Students often get tensed and anxious about their admission chances of their desired universities.</li><li>• The students' peers may get lot of colleges to choose from, with lesser time and effort and lesser expenses.</li></ul>	<b>10. YOUR SOLUTION</b> <span>SL</span> <ul style="list-style-type: none"><li>• The focus is to reduce the time, effort and money spent on finding the universities where admission is feasible for pursuing higher education.</li><li>• The system uses a pre-trained machine model (ML, IBM Cloud and Watson Studio) to predict the feasibility of admission in desired university based on the provided student data.</li><li>• The output of the system is the list of possible universities for the student to apply for admission</li></ul>	<b>8. CHANNELS OF BEHAVIOUR</b> <span>CH</span> <p><b>ONLINE</b></p> <ul style="list-style-type: none"><li>• The students may browse the Internet to research about their desired universities and get to know required information.</li><li>• This is a time-consuming task and may miss out some <u>universities of interest</u></li></ul> <p><b>OFFLINE</b></p> <ul style="list-style-type: none"><li>• Visit the desired universities in person and gather admission details.</li></ul>	Focus on J&P, tap into BE, understand
	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> <ul style="list-style-type: none"><li>• Before: Unaware of the process, suffering to select the best-suited university.</li><li>• After: Secure, user-friendly and aware of process.</li></ul> Reduced cost and does not miss out feasible universities			
Identify strong TR & EM	Extract online & offline CH of BE			



## 4. REQUIREMENT ANALYSIS

### 4.1 Functional requirement:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Profile	User should fill in their following details: Name High school and UG Marks Specialization GRE or/and TOEFL Score Sheet Languages Known Nationality Contact Details
FR-4	User Requirements	Upload all the relevant documents. Based on the uploads, the system would scrape all the necessary information.
FR-5	User Preference	Select the Preferred Countries
FR-6	Display List of Colleges	According to the user profile

		and uploaded documents filter out the colleges according to the user preference Display the list of colleges the student is eligible for.
--	--	--

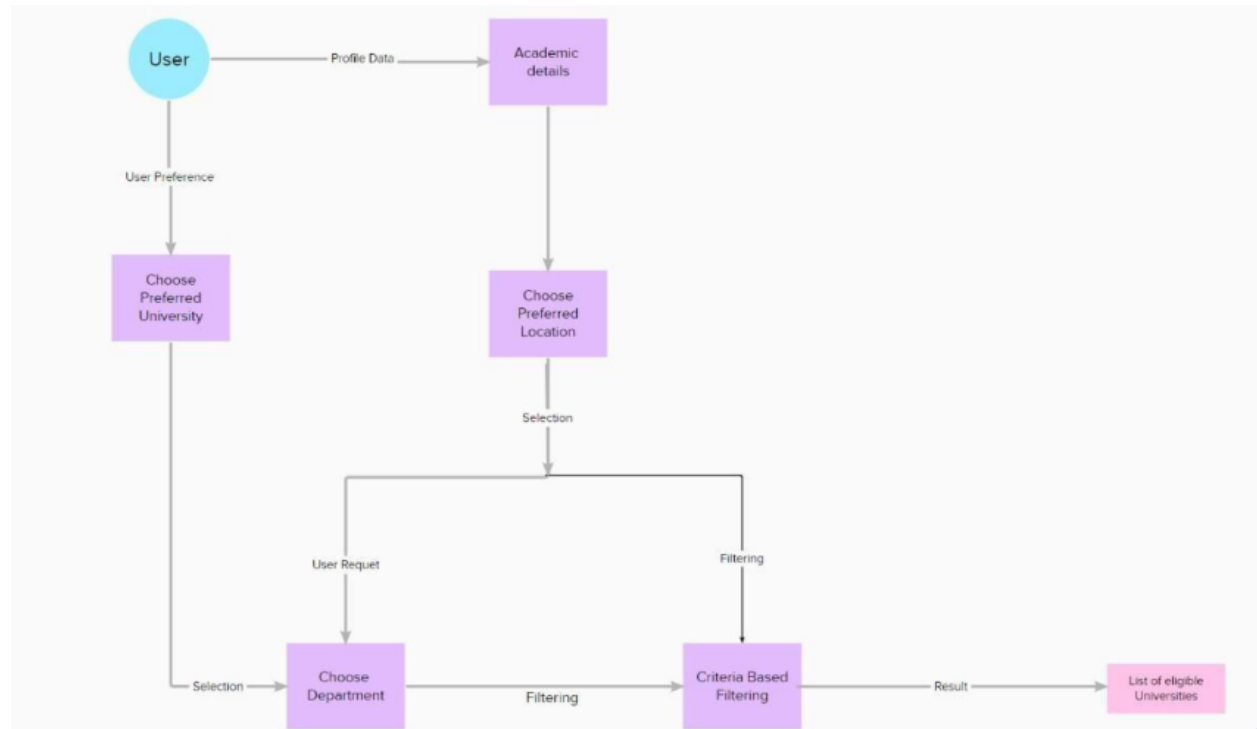
#### 4.2 Non-Functional requirements:

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The system doesn't require technical skills. User friendly. Easily Understandable. Faster results (doesn't take a lot of time for loading)
NFR-2	Security	Only the authenticated user could utilize the services of the site. Database should be backed up every hour.
NFR-3	Reliability	The system should run under stated conditions without any failure  The system would always strive for maximum reliability due to the importance of

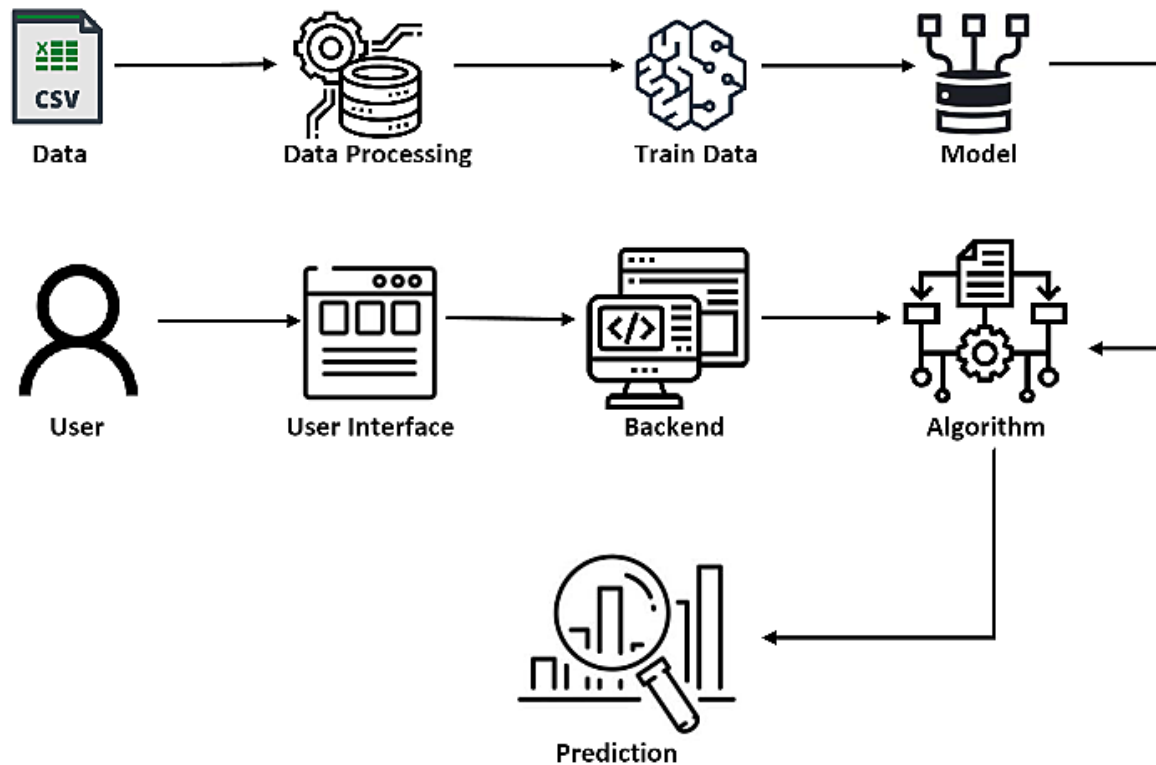
		data and damages that could be cause by incomplete and incorrect data.
NFR-4	Performance	<p>The website can efficiently handle the traffic.</p> <p>Viewing this webpage using a 56 -kbps modem connection would not exceed 30 seconds (quantitatively, the meantime).</p>
NFR-5	Availability	<p>Easy access.</p> <p>Minimal data redundancy.</p> <p>Less prone to errors.</p>
NFR-6	Scalability	<p>Provides concurrent access.</p> <p>The admission season is probably when the system will be under the most strain.</p> <p>Manages a huge amount of users.</p>

## 5. PROJECT DESIGN

### 5.1 Data Flow Diagrams:



## 5.2 Solution & Technical Architecture:



## 5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Profile	USN-1	As a user, I can Give my academic information in the profile section	I can access my dashboard	High	Sprint-1
		USN-2	As a user, I will be able to select a location that I prefer	I can receive the list of location in the dropdown to select	High	Sprint-1
	Search	USN-3	As a user I can search for my preferred university	I can use the search bar	Medium	Sprint-2
	User Preference	USN-4	As a user, I can select my preferred university from the list to check my eligibility for the particular university	I can use the dropdown list provided to select the university	Medium	Sprint-2
		USN-5	As a user, I can select my preferred location	I can select my preferred location	High	Sprint-1
		USN-6	As a user, I will be able to select my preferred department	I can select a department from the dropdown list	Medium	Sprint-1
	Result	USN-7	As a user, I can view the list of universities that I am eligible in accordance to my preferred location	I can view the list of universities filtered by the model	High	Sprint-3
		USN-8	As a user, I can access the link to the university that I am eligible from the list	I can access the university link	Medium	Sprint-3
		USN-9	As a user, I can access the location link of the university that I am eligible from the list	I can access the university location link	Low	Sprint-3
		USN-10	From the list of universities, I can select and view the eligibility for the particular university	I can view the eligible university	Medium	Sprint-3

## 6. PROJECT PLANNING & SCHEDULING

### 6.1 Sprint Planning & Estimation:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Mohamed Sheik Abdul kader, Kishore
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Lenus M Leonard
Sprint-2		USN-3	As a user, I can register for the application through Facebook	2	Low	Mohamed Sheik Abdul Kader
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	Priyadarshan
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Priyadarshan, Lenus M Leonard
Sprint - 2	Dashboard , Update Profile	USN -6	As a user, after login, I will update my profile by providing all the required details	4	High	Priyadarshan , Lenus M Leonard, Kishore
Sprint- 3	Choose University	USN -7	As a user, I will be able to view the list of Universities that the students are eligible to apply	4	Medium	Lenus M Leonard, Priyadarshan , Kishore , Mohamed Sheik Abdul Kader
Sprint -3	Choose Course	USN -8	As a user, I will be able to view the details of Admission process like date and venue of certification verification	2	Low	Kishore , Lenus M Leonard
Sprint - 4	Admission Process	USN -9	As a user, I will be able to view the list of courses that the students are eligible to apply	3	High	Priyadarshan , Lenus M Leonard , Mohamed Sheik Abdul Kader
Sprint - 1	Authentication	USN-10	As a admin, the login credential of the user is authenticated my me	2	High	Priyadarshan , Mohamed Sheik Abdul Kader
Sprint - 2	Update Profile	USN -11	As a admin, I can verify the user Details which are entered by the user	4	High	Priyadarshan, Lenus M Leonard, Mohamed Sheik Abdul Kader, Kishore
Sprint - 3	Prediction	USN -12	As a admin, I can test the trained machine learning model by analyzing the user details by machine learning Algorithms.	3	High	Lenus M Leonard, Kishore , Mohamed Sheik Abdul Kader
Sprint -4	Output	USN - 13	As a admin, I can upload the confirmation of user for the prediction into the database.	3	High	Priyadarshan ,Lenus, Sheik ,Kishore

## 6.2 Sprint Delivery Schedule:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

## 6.3 Reports from JIRA:

University Admit Eligib...  
Software project

PLANNING

Roadmap

Board

DEVELOPMENT

Code

Project pages

Add shortcut

Project settings

Projects / University Admit Eligibility Predictor

Roadmap

Q

L

Status category ▾

	SEP	OCT
<div>UAEP-1 Registration</div>		
<div>UAEP-2 Login</div>		
<div>UAEP-3 Authentication</div>		
<div>UAEP-4 Prediction</div>		
<div>+ Create Epic</div>		

## 7. CODING & SOLUTIONING

### 7.1: Code:

```
import pandas
as pd

from flask import Flask, request, jsonify, render_template
import pickle

app = Flask(__name__)

model = pickle.load(open('regressor.pkl', 'rb'))

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

@app.route('/predict', methods=['GET', 'post'])
def predict():

    GRE_Score = int(request.form['GRE Score'])
    TOEFL_Score = int(request.form['TOEFL Score'])
    University_Rating = int(request.form['University Rating'])
    SOP = float(request.form['SOP'])
    LOR = float(request.form['LOR'])
    CGPA = float(request.form['CGPA'])
    Research = int(request.form['Research'])
```



```

final_features = pd.DataFrame([[GRE_Score, TOEFL_Score, University_Rating, SOP,
LOR, CGPA, Research]])

predict = model.predict(final_features)

output = predict[0]

return render_template('index.html', prediction_text='Admission chances are
{}'.format(output))

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

```

## 7.2. Features:

### 1. Enter Details:

The student can enter their details, which include:

- GRE Score
- TOEFL Score
- University Rating
- Statement Of Purpose Strength
- Letter Of Recommendation Strength
- Undergraduate CGPA
- Research Experience

### 2. Result:

The student will receive a percentage result which gives their chance of admission.

## 8. Testing

### 8.1. Test Case:

Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
Functional	Home page	Verify if user is able to enter the GRE score	GRE marks	1.Enter the GRE score by clicking in the GRE Score field	GRE score	User should be able to enter the GRE score	Working as expected	Pass	Normal test case	Y	BUG ID-1	Priyadarshan S
UI	Home page	Verify if user enters above the limit	nil	1.Verify user enters the score below 340	GRE score	If the user enters above the limit,the application should display "Value must be below 340"	Working as expected	Pass	Normal test case	Y	BUG ID-2	Lenus M leonard
Functional	Home page	Verify if user is able to enter the TOFEL score	TOFEL marks	1.Enter the TOFEL score by clicking in the TOFEL Score field	TOFEL score	User should be able to enter the TOFEL score	Working as expected	Pass	Normal test case	Y	BUG ID-3	Mohamed sheik abdul kader
UI	Home page	Verify if user enters above the limit	nil	1.Verify user enters the score below 120	TOFEL score	If the user enters above the limit,the application should display "Value must be below 150"	Working as expected	Pass	Normal test case	Y	BUG ID-4	Kishore D
Functional	Home page	Verify if user is able to enter the university rating	nil	1.Enter the University Rating by clicking in the University Rating field	Rating	User should be able to enter the university rating	Working as expected	Pass	Normal test case	Y	BUG ID-5	Mohamed sheik abdul kader
UI	Home page	Verify if user enters above the limit	nil	1.Verify user enters the score below 5	Rating	If the user enters above the limit,the application should display "Value must be below 5"	Working as expected	Pass	Normal test case	Y	BUG ID-6	Priyadarshan S
Functional	Home page	Verify if user is able to enter SOP	nil	1.Enter the SOP by clicking in the SOP field	SOP	User should be able to enter the SOP	Working as expected	Pass	Normal test case	Y	BUG ID-7	Lenus M leonard
UI	Home page	Verify if user enters above the limit	nil	1.Verify user enters the score below 5	SOP	If the user enters above the limit,the application should display "Value must be below 5"	Working as expected	Pass	Normal test case	Y	BUG ID-8	Kishore D
Functional	Home page	Verify if user is able to enter LOR	nil	1.Enter the LOR by clicking in the LOR field	LOR	User should be able to enter the LOR	Working as expected	Pass	Normal test case	Y	BUG ID-9	Lenus M leonard
UI	Home page	Verify if user enters above the limit	nil	1.Verify user enters the score below 5	LOR	If the user enters above the limit,the application should display "Value must be below 5"	Working as expected	Pass	Normal test case	Y	BUG ID-10	Kishore D
Functional	Home page	Verify if user is able to enter CGPA	CGPA	1.Enter the CGPA by clicking in the CGPA field	CGPA	User should be able to enter CGPA	Working as expected	Pass	Normal test case	Y	BUG ID-11	Priyadarshan S
UI	Home page	Verify if user enters above the limit	CGPA	1.Verify user enters the CGPA less than 10	CGPA	If the user enters above the limit,the application should display "Value must be less than 10"	Working as expected	Pass	Normal test case	Y	BUG ID-12	Mohamed sheik abdul kader
Functional	Home page	Verify if user is able to enter research	nil	1.Enter the Research experience by clicking in the research experience field and enter 0 for no experience and 1 for experience	nil	User should be able to enter research experience data	Working as expected	Pass	Normal test case	Y	BUG ID-13	Kishore D
UI	Result page	Verify if user is able to see the result of prediction	nil	1.Click on the predict test box	nil	User should be able to see the result of prediction	Working as expected	pass	Normal test case	Y	BUG ID-14	Mohamed sheik abdul kader
UI	Result page	Verify user is able to have a chance	nil	1.Click on the predict test box	nil	Application should display "Admission chances are"with percentage	Working as expected	Pass	Normal test case	Y	BUG ID-15	Priyadarshan S

### 8.2 User Acceptance Testing:

#### i) 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).

#### ii) 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	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

### iii) Test Case Analysis:

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

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

## 9. Results

### 9.1. Performance Metrics:

```
In [47]: mae = metrics.mean_absolute_error(y_test, predlinear)
mse = metrics.mean_squared_error(y_test, predlinear)
rmse = np.sqrt(mse) # or mse**(0.5)
r2 = metrics.r2_score(y_test, predlinear)
```

```
In [48]: chart = {
    'Metric': ["MAE", "MSE", "RMSE", "R2-SCORE"],
    'LINEAR_REGRESSION': [mae, mse, rmse, r2],
}
chart = pd.DataFrame(chart)
```

```
In [49]: display(chart)
```

	Metric	LINEAR_REGRESSION
0	MAE	0.043051
1	MSE	0.003313
2	RMSE	0.057560
3	R2-SCORE	0.807216

```
In [43]: model = LinearRegression(normalize=True)
model.fit(X_test, y_test)
# model.score(X_test, y_test)
predlinear = model.predict(X_test)
print ("Accuracy : ", model.score(X_test, y_test)*100)
methodDict = {}
methodDict['Linear Regression'] = model.score(X_test, y_test)*100

Accuracy : 80.7216438856893
```

### 9.2. Tune The Model:

#### Hyperparamter Tuning:

- The number of features is important and should be tuned in random forest classification.
- Initially all parameters in the dataset are taken as independent values to arrive at the dependent decision of Chronic Kidney Disease or No Chronic Kidney Disease.
- But the result was not accurate so used only 8 more correlated values as independent values to arrive at the dependent decision of Chronic Kidney Disease or not.

### 9.3. Validation Model:

It involves partitioning the training data set into subsets, where one subset is held out to test the performance of the model. This data set is called the validation data set.

Cross validation is to use different models and identify the best:

```
In [64]: scores = cross_val_score(model, X_train, y_train, scoring='r2', cv=5)
scores |

Out[64]: array([0.81813967, 0.77169539, 0.83989563, 0.74719974, 0.78589678])

In [65]: avg_score=scores.mean()

In [67]: print ("Cross Validation Scores : ",scores)
print ("Average CV Score : ",avg_score)
print ("Number of CV Scores used in Average : ",len(scores))

Cross Validation Scores : [0.81813967 0.77169539 0.83989563 0.74719974 0.78589678]
Average CV Score : 0.7925654408790849
Number of CV Scores used in Average : 5
```

### 9.4. Screenshots:

**University Admit Eligibility Predictor!**

A Simple Web App to Predict The Chances of Getting Admission in Universities Based on Student's Profile

**Input Guide**

- GRE Score (out of 340)
- TOEFL Score (out of 120)
- University Rating (out of 5) - the category of the target university
- Statement of Purpose (SOP) Strength (out of 5)
- Letter of Recommendation (LOP) Strength (out of 5)
- Undergraduate CGPA (out of 10)
- Research Experience (0 for NONE and 1 for YES)

300

110

4

4

5

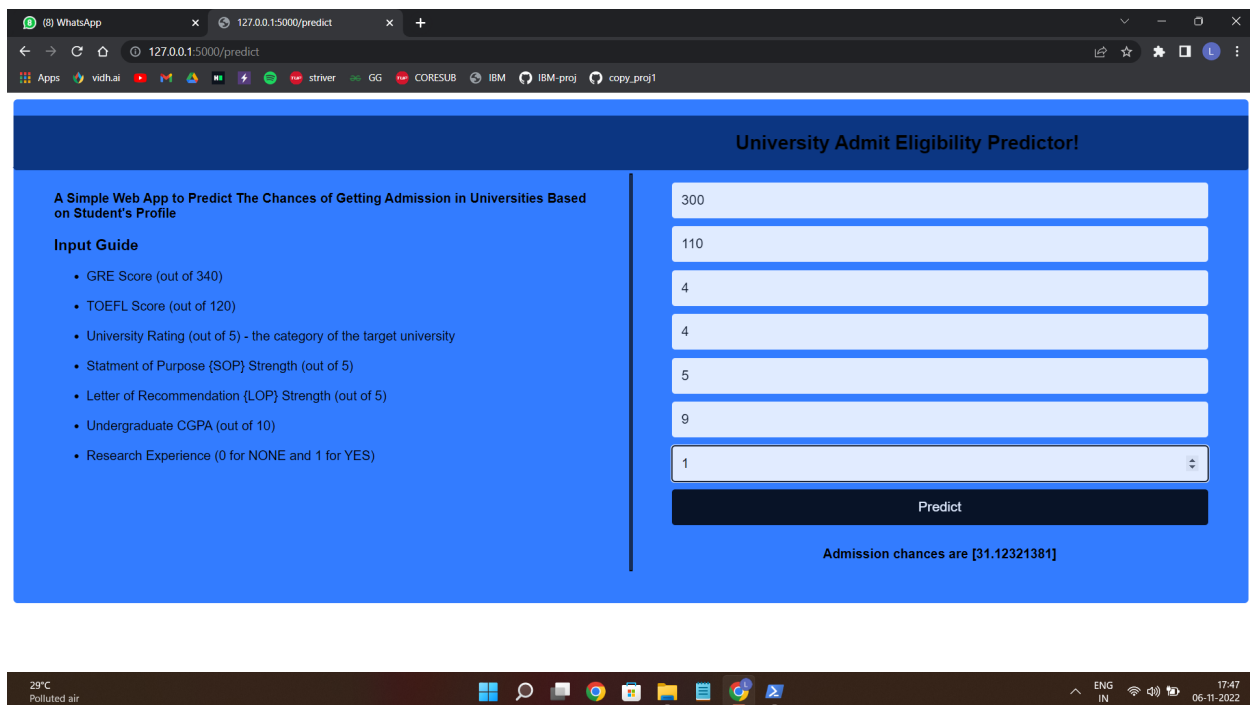
9

1

Predict

Admission chances are [31.12321381]

ENTER DETAILS



**RESULT**

## **10. Advantages and Disadvantages**

### **Advantages:**

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

### **Disadvantages:**

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

## 11. Conclusion

The numbers of the students seeking further higher education abroad keeps increasing every year. And this number won't go down in upcoming years either. In our technological and competitive world, the students constantly feel the need to upskill themselves and fare better than their peers and competitors every day and thus after their under graduation, a vast number of students tend to leave for higher education abroad. To facilitate an easy migration, and to provide them with a tool which can help them shortlist their potential college/university based on various parameters will not just empower them in finding their preferred institutions but also ensure that it saves their precious time as well as their hard-earned money at the same time. Many researchers have tried to come up with ideas using machine learning, data mining, and generic algorithm from time to time, each with their own benefits and drawbacks. If we go through the research done till date, the success rate of hybrid systems with the implementations of more than one algorithm concurrently is higher with low error rates, as the work done by each algorithm in any hybrid system relies on each of them only for the specific task and hence later collaboration ensures higher accuracy. In this regard, the stacked ensemble approach, which itself is based on an ensemble approach which tries to bring different algorithms together to gain better accuracy, fares better than all the other approaches discussed and reviewed in this project.



## 12. FUTURE SCOPE

- We can begin with building the system for colleges under Anna University alone.
- From there, the system can be expanded to include universities around India and then all over the world.
- In this way, the system is scalable and can be developed further.

## 13. APPENDIX

**13.1. Source Code Link :** <https://drive.google.com/file/d/1eShTrzigXjemtWu-8WdGf1rKYmVRiOIY/view?usp=sharing>

**3.2. GitHub & Project Demo Link:**

GitHub link: <https://github.com/IBM-EPBL/IBM-Project-28322-1660110687>

Project Demo Link:

The screenshot displays a web application titled "University Admission Enquiry Predictor". Below the title, a subtitle reads: "A Simple Web-App to Predict The Chances of Getting Admission at Universities Based on Student's Profile".

**Input Guide:**

- GRE Score (out of 340)
- TOEFL Score (out of 120)
- University Rating (out of 5) - (in category of the target university)
- Statement of Purpose (SOP) Strength (out of 5)
- Letter of Recommendation (LOR) Strength (out of 5)
- Undergraduate GPA (out of 4.0)
- Research Experience (YES/NO and YES/YES)

**Form Fields:**

- Score: 200
- TOEFL Score: 8
- University Rating: [dropdown menu]
- SOP: [dropdown menu]
- LOR: [dropdown menu]
- GPA: [dropdown menu]
- Research: [dropdown menu]

**Buttons:** A "Predict" button is located at the bottom right of the form area.