UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

PROJECT REPORT

Domain : Applied Data Science

Team ID : PNT2022TMID35628

College Name : College of Engineering Guindy

Surya R (2019115112)
Department of Information Science and Technology

Rengaraj Praveen Kumar (2019115078)
Department of Information Science and Technology

Bhairav AJ (2019115026)
Department of Information Science and Technology

Ch Chandrateja Reddy (2019115125)
Department of Information Science and Technology

CONTENTS

1. INTRODUCTION

Project Overview

College admission predictor is a boon to many students. This helps the student not only to help in filling out the application forms but also give the students an idea about their future college by calculating their cut off. When students come from rural places, they find it hard to go along with the formal procedures. So, this application helps them a lot and eases out their fear. Whatever may be their scores, this application helps to find the best colleges. Hence, our proposed computer aided system will help the students to get the list of all colleges in which they could get the admission at the click of a button. The students only have to enter their marks of XII, AIEEE etc. With this application, 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. To grab insights from data through visualization.

- Applying different ML algorithms to determine the probability of acceptance in a particular university.
- Evaluation metrics
- Build a web application using the Flask framework.

Purpose

- 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 to a particular university.
- This analysis should also help students who are currently preparing or will be preparing to get a better idea.

2. LITERATURE SURVEY

Existing problem

In today's world many students are often worried about their chances of admission to university. The main moto of the project is help students to short listing universities with their marks. 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.

Given certain metrics of a student, our task is to predict the probability of the student getting accepted into graduate programs. Statistically, we have seen many students pursue their education away from their native countries. Generally, as the students don't have much of an idea about the procedures, requirements, and details of the universities, they seek help from education consultancy firms to help them successfully secure admission to the universities which are best suitable for their profiles. For this, they have to invest huge amounts of money in consultancy fees. The aim of this research is to develop a system using Applied Data Science.

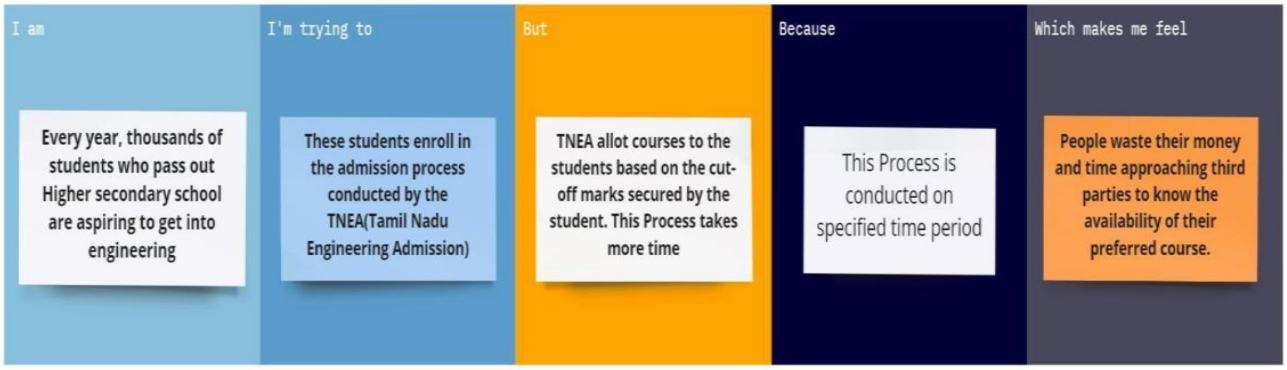
University prediction would be the easiest mode to predict the university/college person is applicable for as well as it would be unbiased and totally transparent. Individually would no more need

to depend upon the consultancies who may be slightly deviated from the list of colleges/universities that may be having contracts with them. Moreover, applying to only colleges/universities where the student has a genuine chance would reduce the application process. Additionally, living expense of the area where colleges/university is located would also be provided on the website.

References

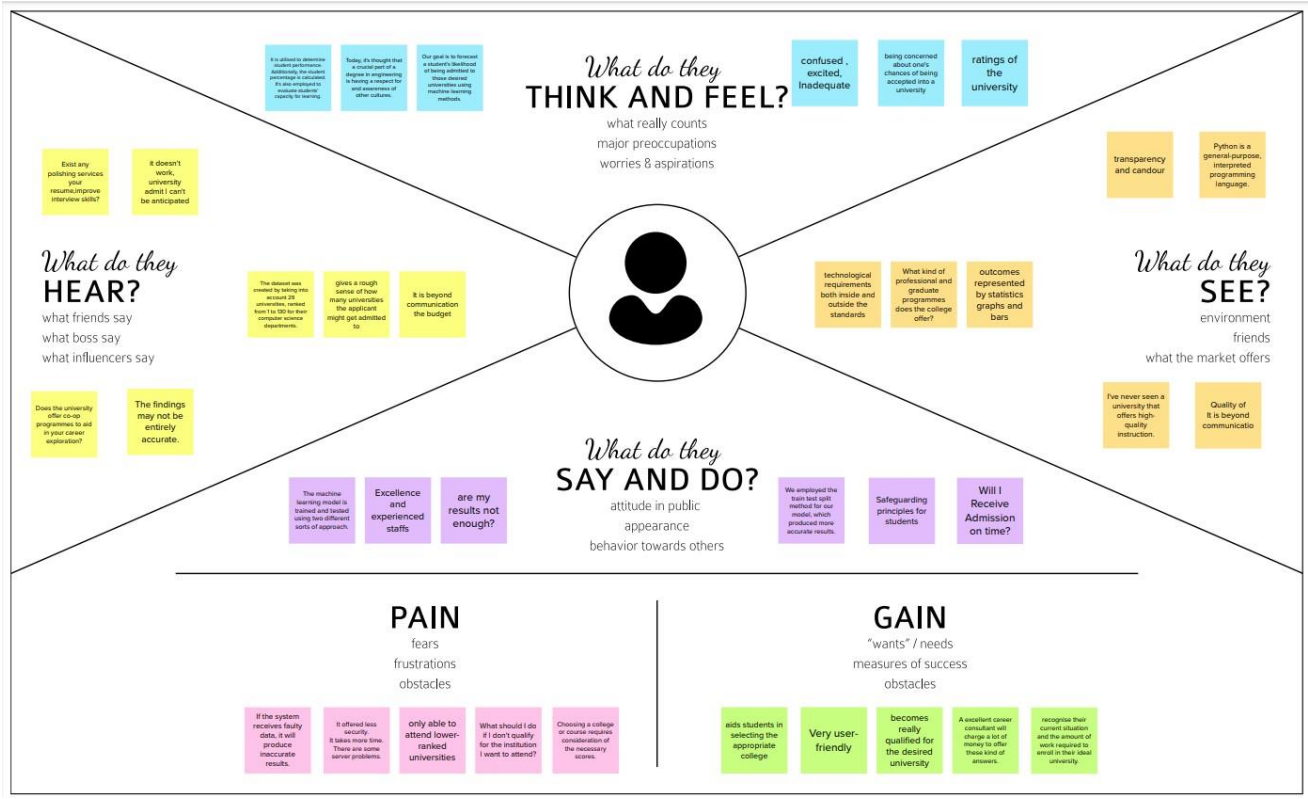
- Sivasangari, A., et al. "Prediction Probability of Getting an Admission into a University using Machine Learning." 2021 5th International Conference on Computing Methodologies and Communicatio(ICCMC). IEEE, 2021.
- Sridhar, Sashank, Siddartha Mootha, and Santosh Kolagati. "A University Admission Prediction System using Stacked Ensemble Learning." 2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA). IEEE, 2020.
- Fathiya, Haseeba, and Lipsa Sadath. "University Admissions Predictor Using Logistic Regression." 2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE). IEEE, 2021.
- Goni, Md Omaer Faruq, et al. "Graduate admission chance prediction using deep neural network." 2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering(WIECON-ECE). IEEE, 2020.

Problem Statement Definition

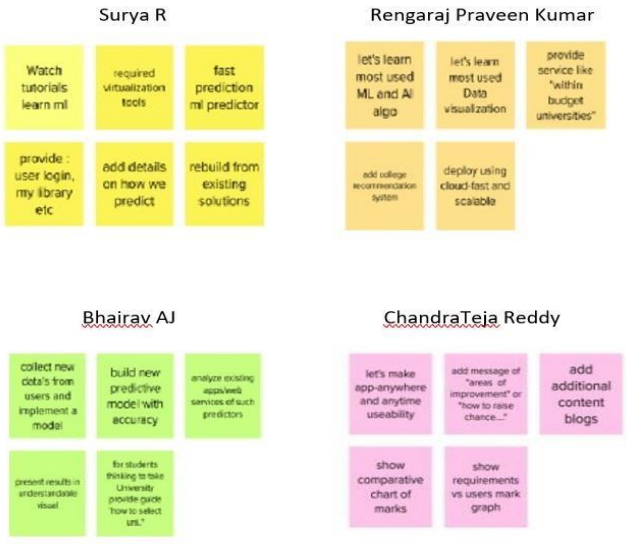


3. IDEATION & PROPOSED SOLUTION

Empathy Map Canvas



Ideation & Brainstorming



3

Group ideas

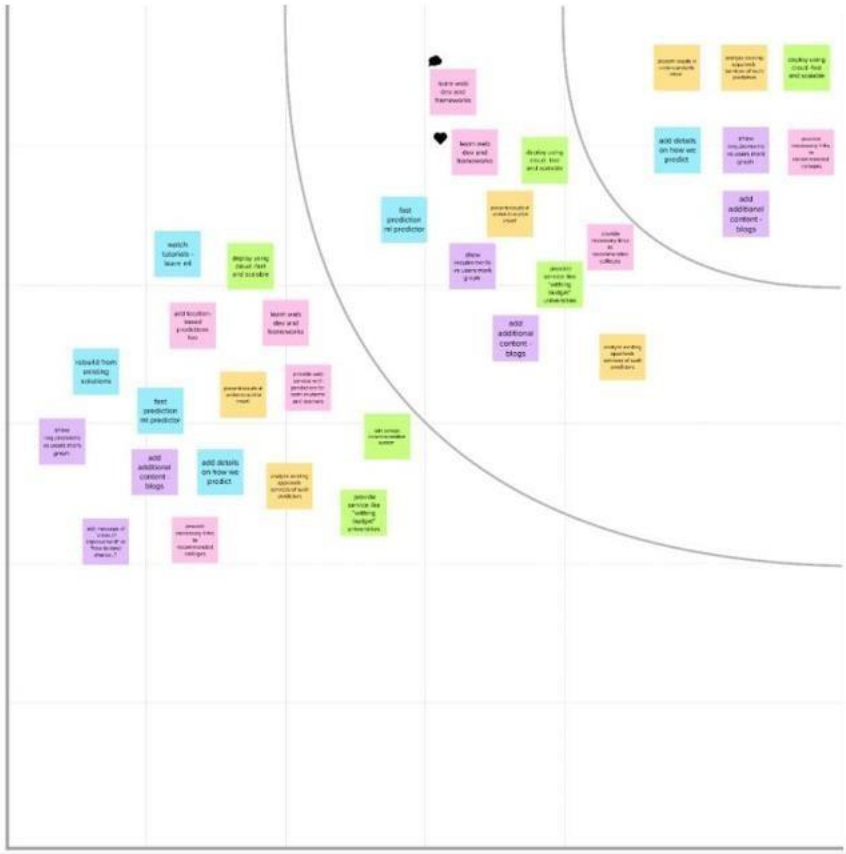
Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

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.



Importance

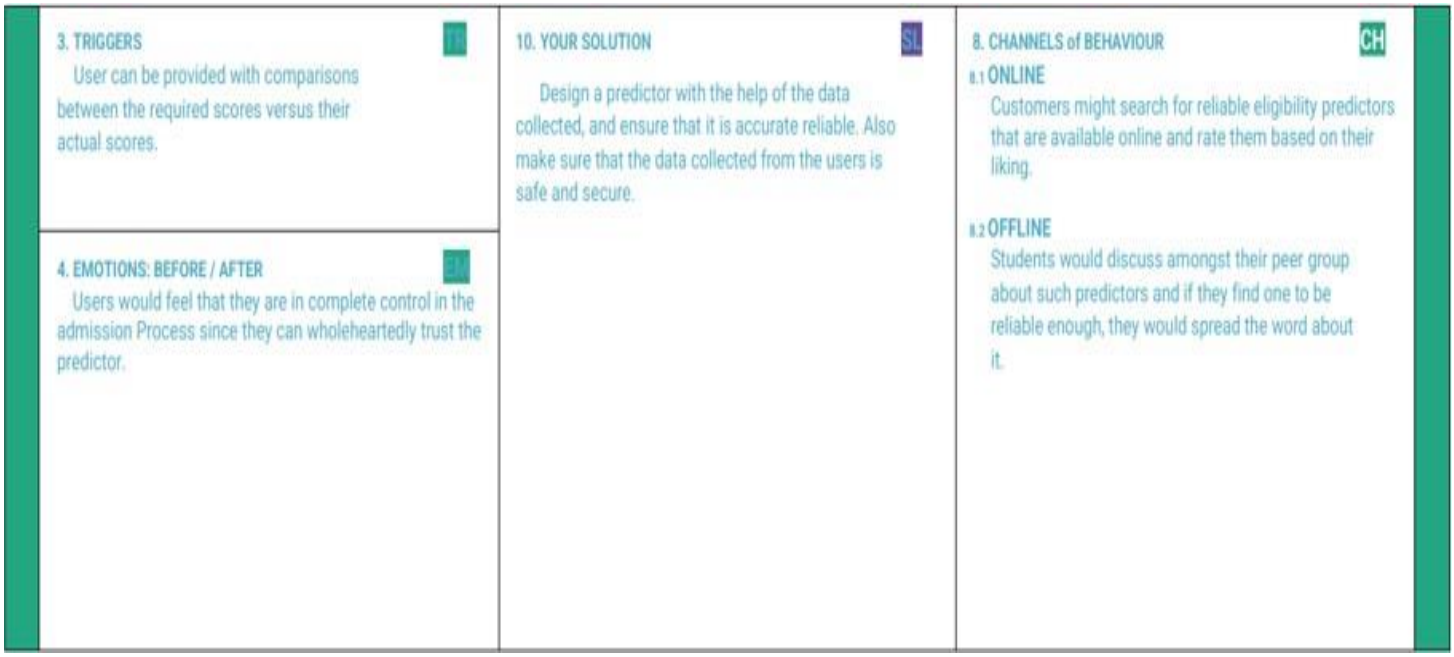
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	We have seen a lot of students pursue their education away from their native countries. Generally as the students don't have much of an idea about the procedures, requirements and details of the universities, they seek help from the education consultancy firms to help them successfully secure the admission in the universities which are best suitable for their profiles. For this, they have to invest huge amount of money as consultancy fees. The aim of this research is to develop a system using machine learning algorithm.
2.	Idea / Solution description	Provide the availability of the preferred course when the student enters the necessary details that the website asks for. The prediction of the course is done based on TNEA rank. The key attributes that will be considered for making the decisions are: • TNEA Rank • Cut Off • Community Rank • Community • Gender For determining the course of the college, we will be using various ML classification models. which model gives the highest accuracy with the help of performance metrics like accuracy-score, precision and recall.
3.	Novelty / Uniqueness	Through this software Students can know what course they might get in Anna University just by entering their Ranks, cut-off marks, community to which they belong and community marks. Students need not face the difficulty of consulting any third party.

4.	Social Impact / Customer Satisfaction	• Students often feel difficult in shortlisting the universities to apply which they tend to wonder if their profile matches the requirement of a certain university. • People do not waste their money and time approaching third parties to know the availability of their preferred course. • Through this software Students can know what course they might get in Anna University • This system reduces dependence on educational consultancies, who charge loads of money to analyse a candidate's Ranks, cutoff marks, community to which they belong and community marks
5.	Scalability of the Solution	• Right now we have used the data set of the last five years. The scalability of this software increases when datasets of previous years are added. • A future update could have chat space where candidates, faculties, current students of the university and alumni can interact and candidates can get their doubts resolved instantly.

Problem Solution fit



4. REQUIREMENT ANALYSIS

Functional requirement

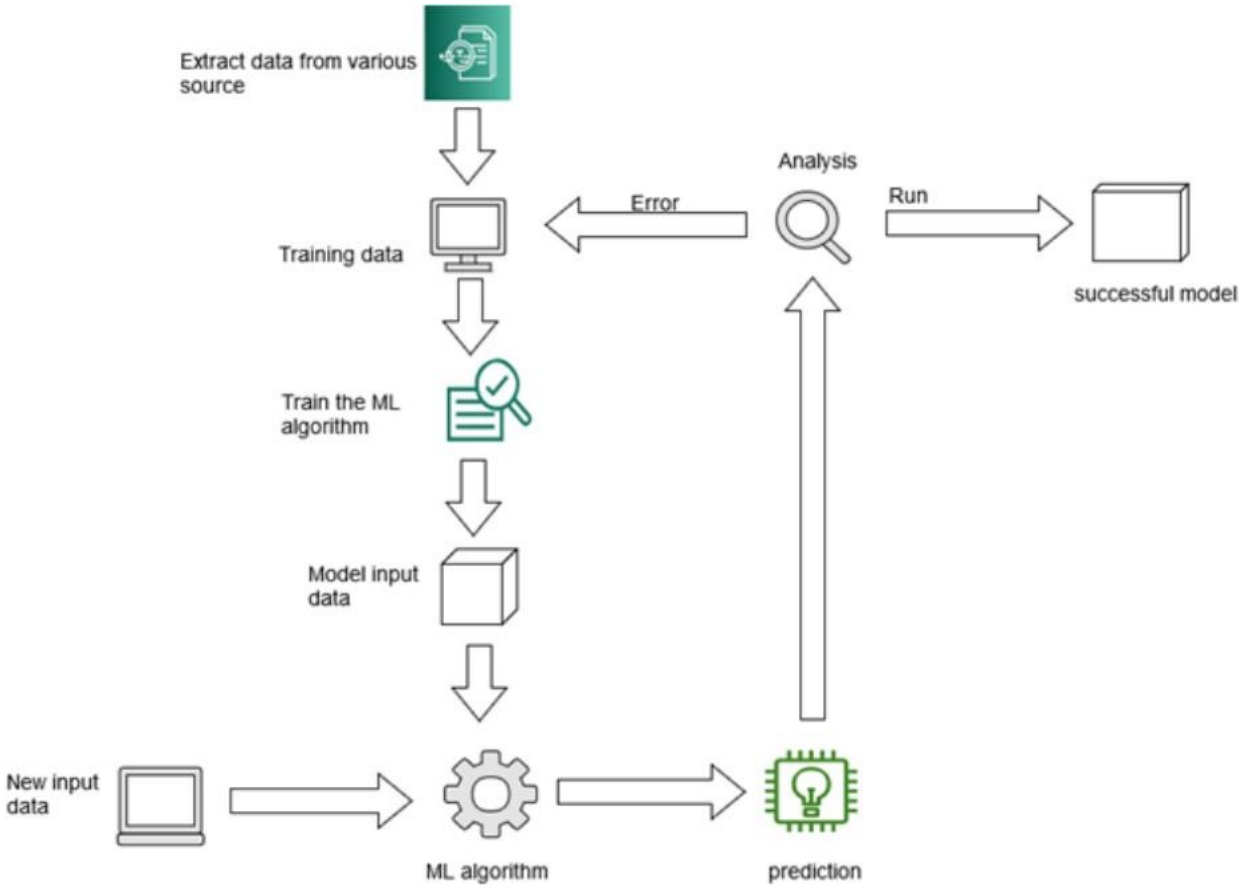
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Data Collection**	The following details of Students' Score are collected: HSC, Community, Rank, Community Rank.
FR-2	**Evaluation**	Using ML algorithms to analyse the data entered by the students and testing the developed ML model with the supplied data.
FR-3	**Prediction**	Prediction is done based on the result of evaluation, the List of Universities for which the students are eligible to apply will be displayed.
FR-4	**Output**	Based on their eligibility, students move forward with the admissions procedure to the predicted university and course.

Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Interactive and Effective UI Visualization of Progress Customer Satisfaction Ease of Learning
NFR-2	**Reliability**	The predictor system will be consistent in order for the system to produce trustworthy and accurate outcomes.
NFR-4	**Performance**	As Decision tree is applied to develop, performance will be more effective.
NFR-5	**Availability**	Users will be able to access the system predictor at any time, anyplace, as needed.
NFR-6	**Scalability**	It can handle any amount of data and perform many computations in a cost-effective and time-saving way.

5. PROJECT DESIGN

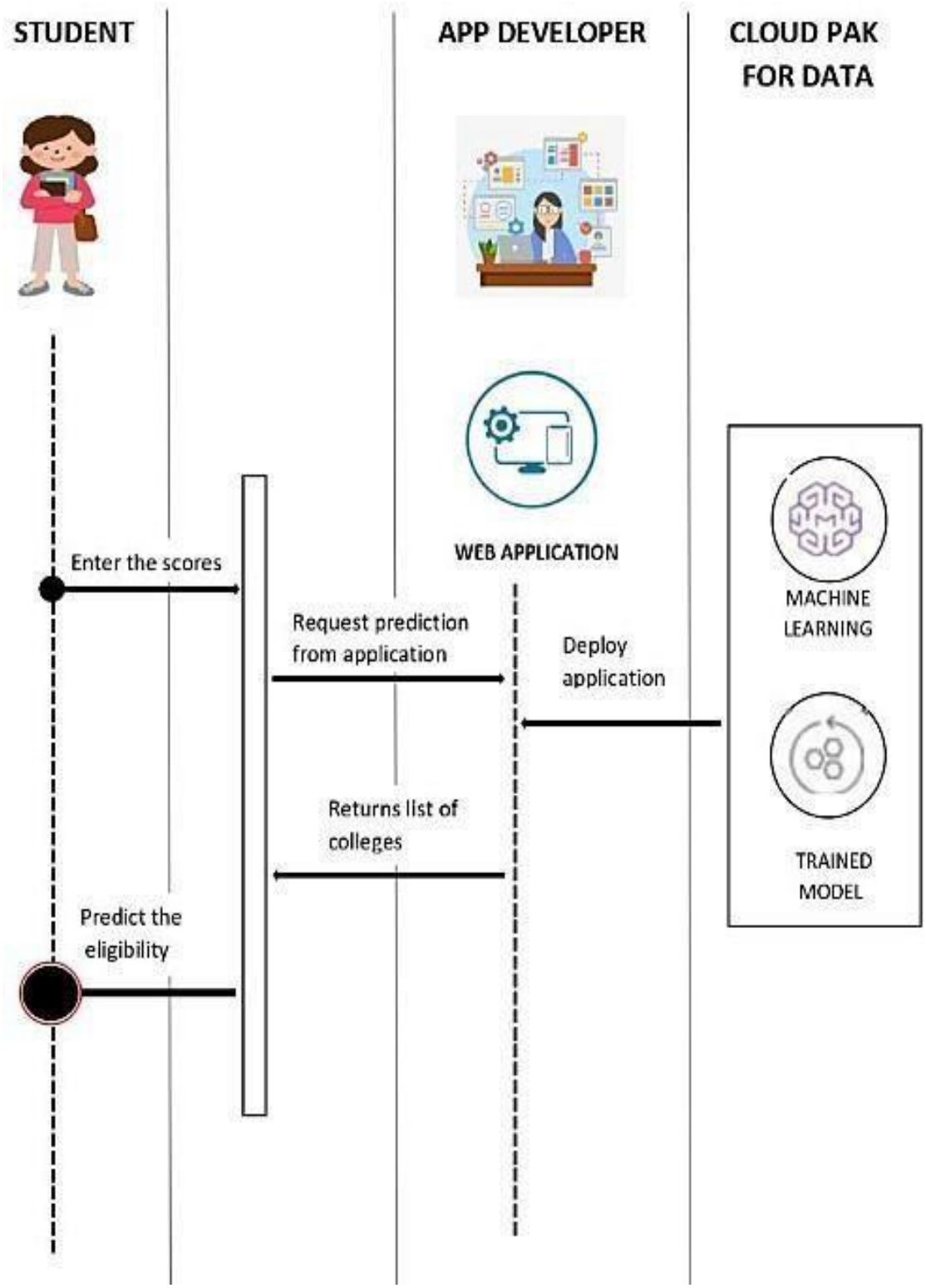
Data Flow Diagram



Solution & Technical Architecture

Solution Architecture

1. This solution 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.
2. The chance of occurrence of error is less when compared with the existing system. 3.First, Enter the scores in the fields.
4. Next, request the prediction from web application.
5. Returns the list of colleges based on scores.
6. Predicts the eligibility score.
7. Recommending best suitable universities to students based on their cut off marks, community rank, rank and also predicting admission probability.


STUDENT
APP DEVELOPER
CLOUD PAK FOR DATA
WEB APPLICATION
MACHINE LEARNING
TRAINED MODEL
Enter the scores
Request prediction from application
Deploy application
Returns list of colleges
Predict the eligibility

Technical Architecture

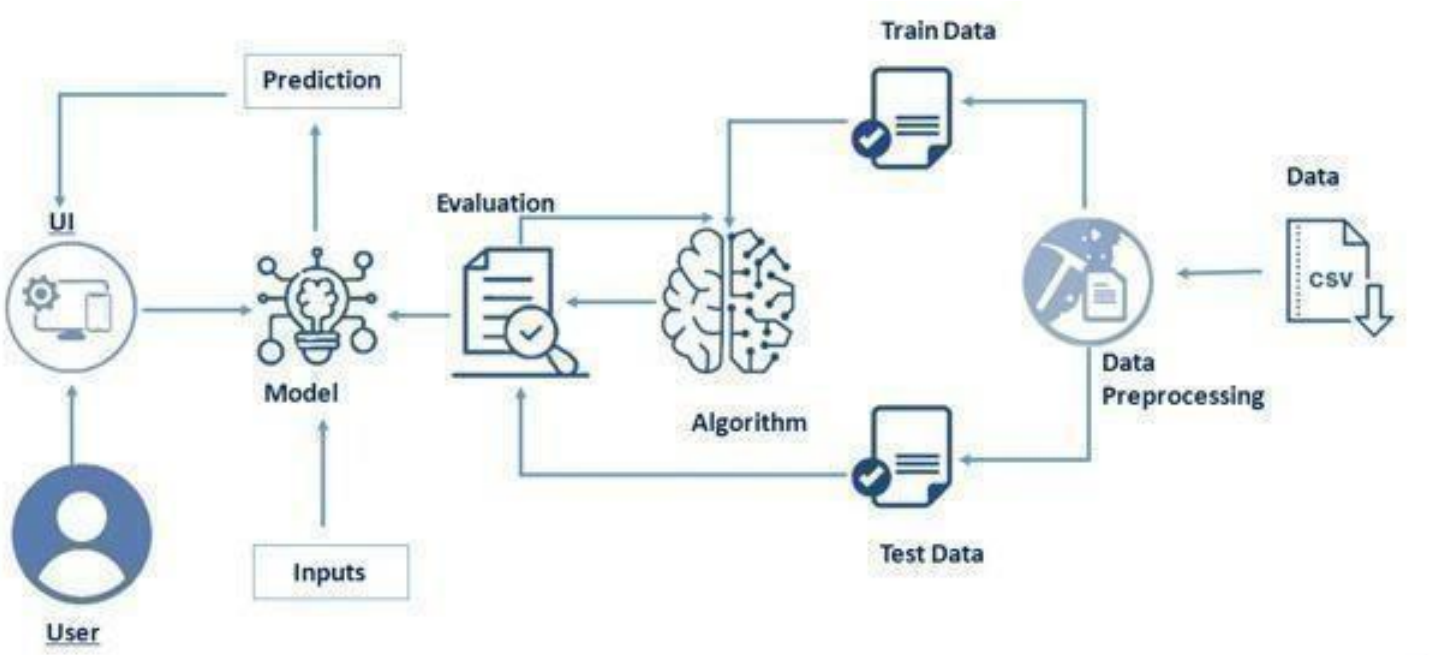


Table-1: Components &Technologies:

S.No	Component	Description	Technology
1	User Interface	The Front-end part of the application	HTML, CSS
2	Libraries	Import Libraries into data	Numpy, Pandas, Seaborn, Matplotlib
3	File Storage	File storage requirements	Local File System
4	Machine Learning Model	Purpose of Machine Learning Model	Admission Prediction Model
5	Training and testing data	Purpose of training and testing data	Decision Tree algorithm
6	Accuracy	Accuracy of the tested and trained data	Score

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technologies Used
1	Open-Source Frameworks	List the open-source frameworks used	Flask Framework
2	Scalable Architecture	Many computations can be done in a time saving and effective way	Decision Tree
3	Availability	Our web application is available at any time and at any place	IBM Load Balancer
4	Performance	As Decision Tree is applied to develop the performance will be more effective	Decision Tree

User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)		USN-1	As a user, I can receive university details and their ranking.	I can only view(read-only)	Medium	Sprint-1
		USN-2	As a user, I can review the experience of the students in the university	I can access the review sections	Medium	Sprint-2
		USN-3	As a user, I can fill out the general and educational details in the form provided	I have read and write access to the forms filled.	High	Sprint-1
	Predictor	USN-4	I can view the list of universities in which I am eligible to get an admission	I can receive the final result as whether eligible or not	High	Sprint-2

6. PROJECT PLANNING & SCHEDULING

Sprint Planning & Estimation

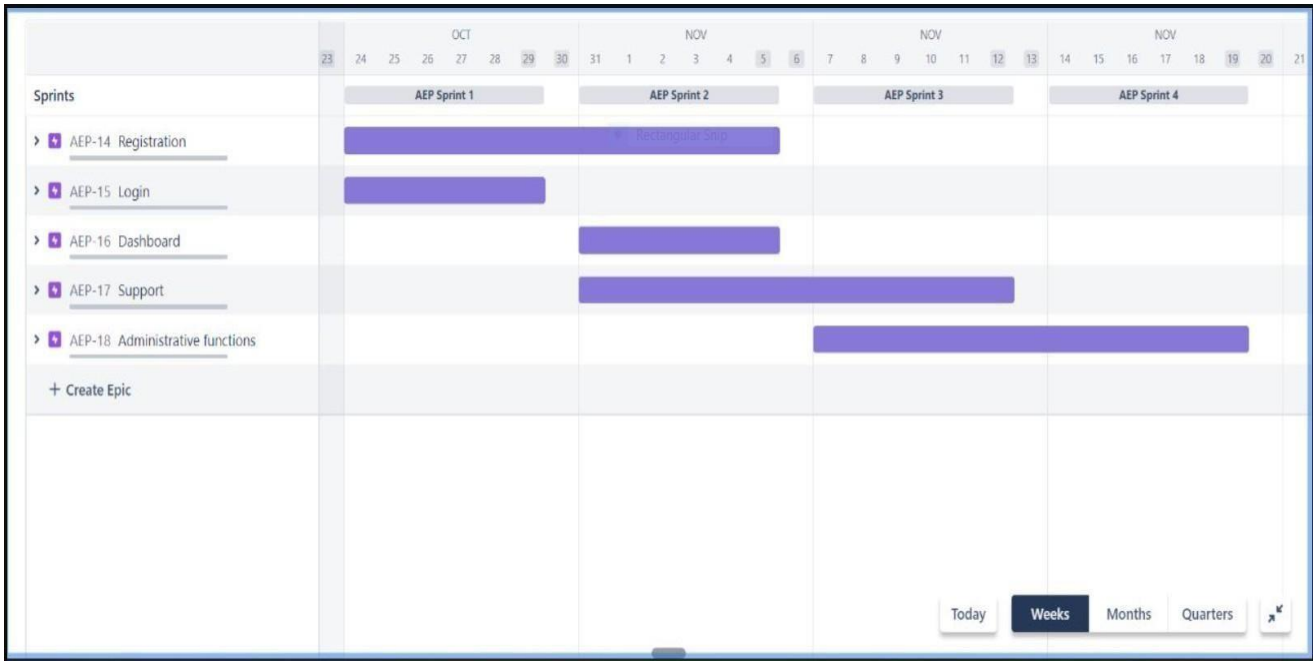
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team member
Sprint-1	Exploratory Data Analysis	USN-1	Perform data cleaning if required and perform univariate, bivariate and multivariate analysis.	1		1,3
Sprint-2	User Data	USN-2	User enter the data that are consist of 4 parameters such as Rank, Community, Community Rank, Cut Off Mark.	1	Medium	2,3
Sprint-3	Prediction	USN-3	Taking a given parameter to predict the label in separate page.	1	High	1,4
Sprint-4	Integrate the web app with the deployed model.	USN-4	Use the deployed model in IBM Watson through the scoring endpoint by making an	3	Medium	2,4

			API call with the IBM cloud API key.			

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	19	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Reports from JIRA



7. CODING & SOLUTIONING

Feature 1

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

Feature 1

Index.html

```
<!DOCTYPE html>
<html>
<head>
        <title>University Admit Eligibility Predictor</title>
</head>
<link rel="preconnect" href="https://fonts.gstatic.com">
<link
href="https://fonts.googleapis.com/css2?family=Raleway:wght@100&display=swap"
rel="stylesheet">
<link
href="https://fonts.googleapis.com/css2?family=Noto+Sans+HK:wght@500&display=sw
ap" rel="stylesheet">
<style type="text/css">

        h1,h2{
                font-family: 'Times New Roman', serif;
                color: black;
        }
        h2,h1,form,p,b{
                text-align: left;
                color: black;
        }
        label,p,b{
                font-family: 'Arial', sans-serif;
                color: black;
```

```
        }
        .elements{
                padding-top: 1px;
        }

</style>
<body>

        <img src="https://www.linkpicture.com/q/Screenshot-2022-11-10-185544_1.png"
style= "position: -webkit-sticky; position: sticky; top: 0;" align="right">
        <h1 style="font-size: 3rem; text-decoration-line: underline; text-decoration-
thickness: auto;">University Admit Eligibility Predictor</h1>

        <p style="font-size: 2rem; font-family: 'Arial', sans-serif;"> <strong> ABOUT
</strong></p>
        <p style="font-size: 1.5rem; font-family: 'Helvetica', sans-serif;">Enter your details
to predict whether you'll get an admission or not .</p>

        <form action="/predict" method="post" class="elements" style="font-size: 1rem;">
                <p style="font-size: 2rem; font-family: 'Arial', sans-serif;"> <strong>
DETAILS </strong></p>
                <p class="elements"><label>Community</label> </p>
                <select name="com" style=" border-radius: 8px;">
                        <option value="1">BC</option>
                        <option value="2">SC</option>
                        <option value="3">OC</option>
                        <option value="4">BCM</option>
                        <option value="5">MBC</option>
            <option value="6">SCA</option>
            <option value="7">ST</option>
         </select>
                <br>
                <p class="elements">Rank</p>
                <p><input type="text" name="rank" value="Score range 0-50000" style="
border-radius: 8px;"></p>
                <p class="elements">Com.Rank</p>
                <p><input type="text" name="comr" value="Score range 0-170000" style="
border-radius: 8px;"></p>
                <p class="elements">CutOff</p>
                <p><input type="text" name="co" value="Score range 150-201" style="
border-radius: 8px;"></p>
                <p class="elements"> <input type = "Submit" value = "Submit" style="
border-radius: 8px;"/> </p>
        </form>

</body>
```

```
</html>
```

Feature 2

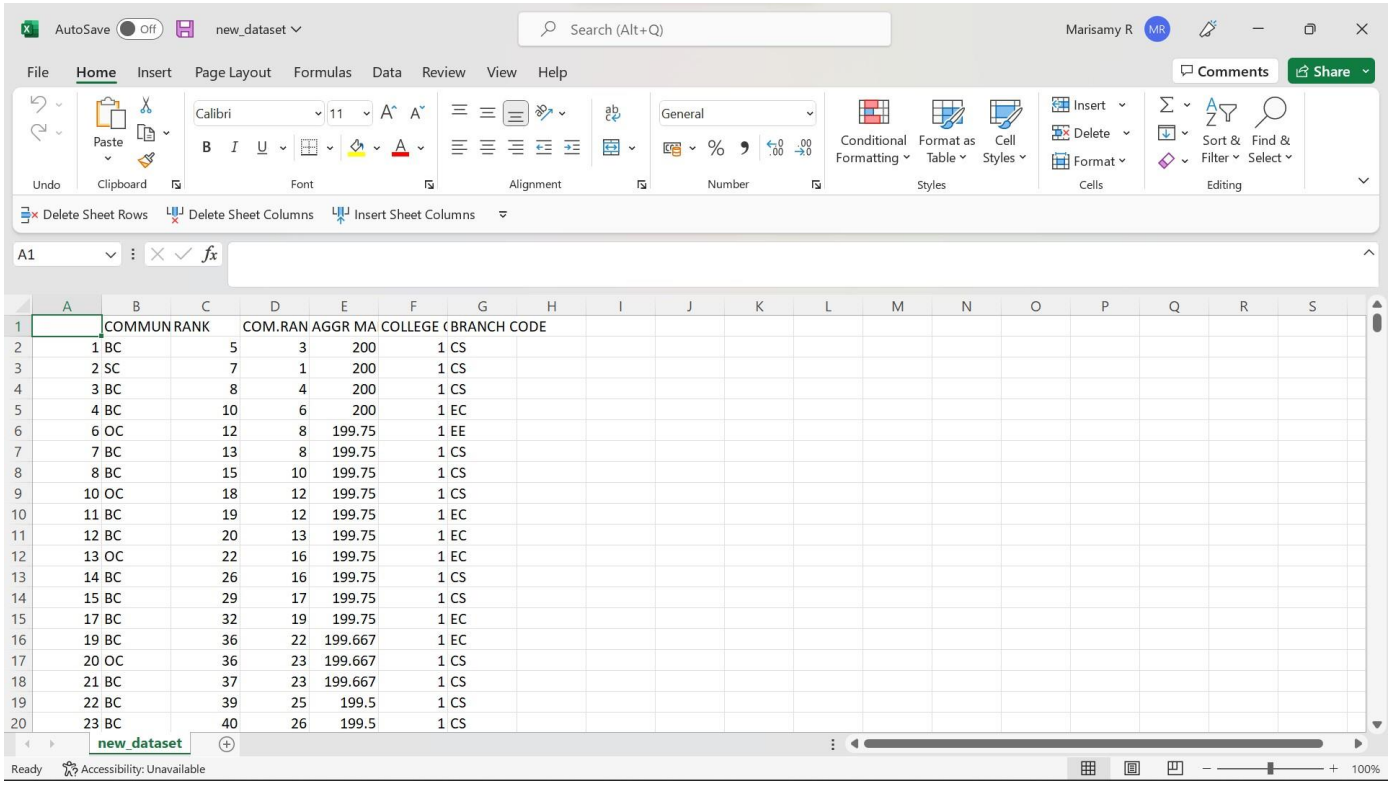
Chance.html

```
<!DOCTYPE html>
<html>
<head>
	<title>eligibility</title>
</head>
<body>
	<img
src="https://i.pinimg.com/736x/d7/6f/89/d76f89a73987e8b831253bb9a4b8cfa5.jpg"
style="float: right;">
	<div style="padding-top: 15%">
	<h2>Predicting chance of admission</h2>
	<h3>A Machine Learning Web App Using Flask</h3>
	<p>Prediction : <b>You've a chance to get the admission !</b></p>
   </div>
</body>
</html>
```

No Chance.html

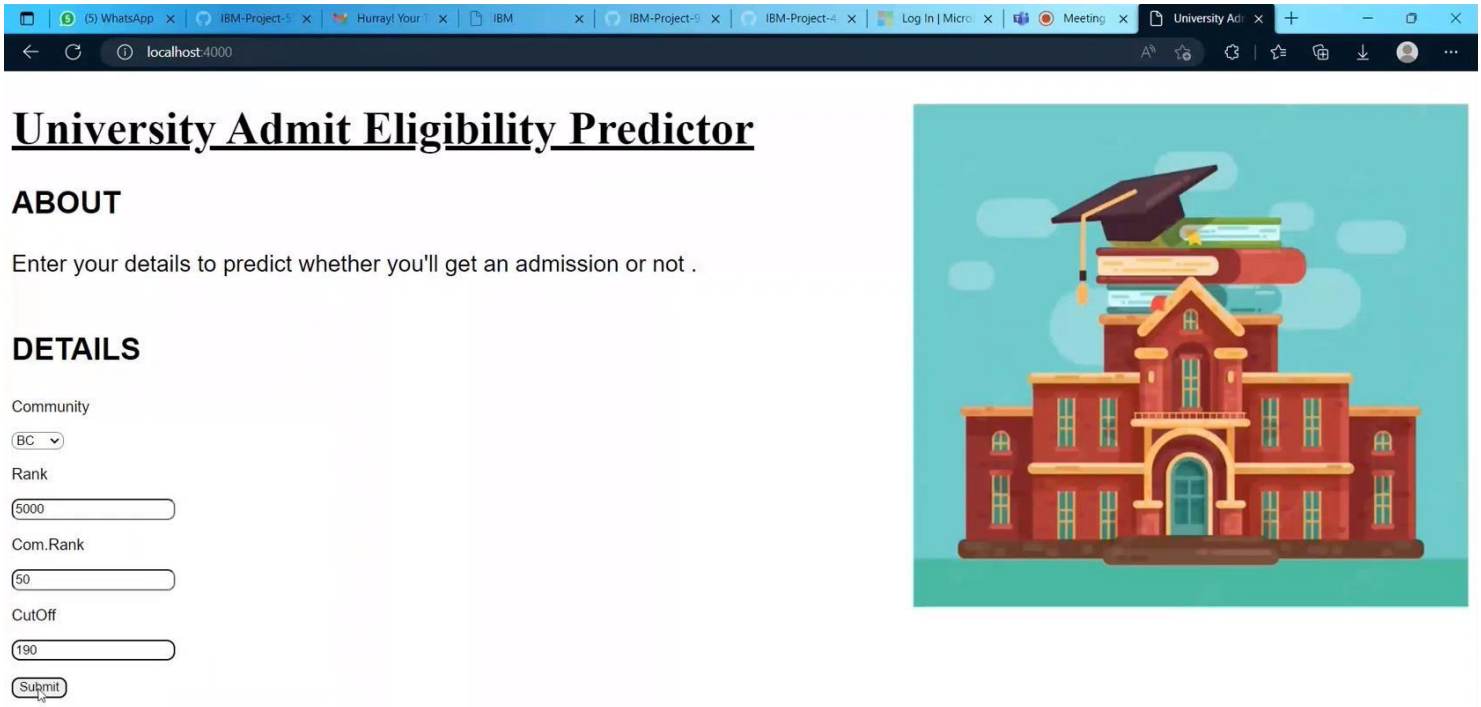
```
<!DOCTYPE html>
<html>
<head>
	<title>eligibility</title>
</head>
<body>
	<img src="https://ih1.redbubble.net/image.436264386.5214/st,small,507x507-
pad,600x600,f8f8f8.u5.jpg" style="float: right; " height="50%" width="50%">
	<div style="padding-top: 15%">
	<h2>Predicting chance of admission</h2>
	<h3>A Machine Learning Web App Using Flask</h3>
	<p>Prediction : <b>You don't have a chance!</b></p>
	</div>
</body>
</html>
```

Database Schema

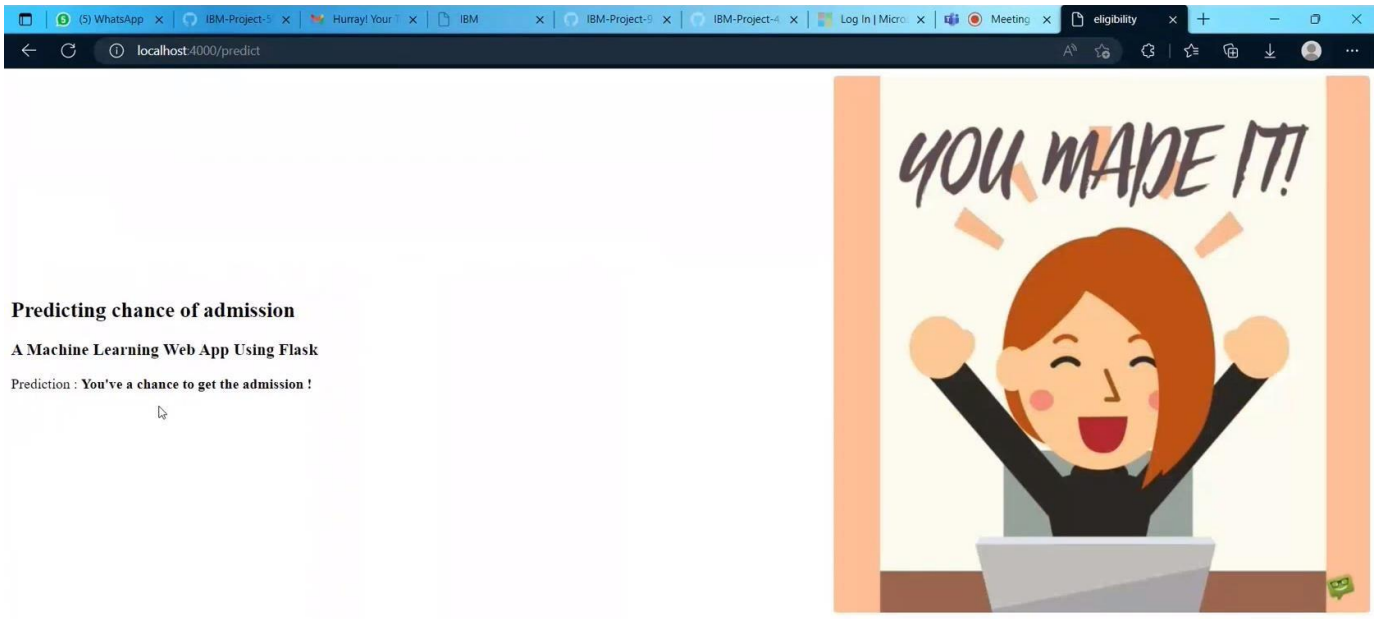


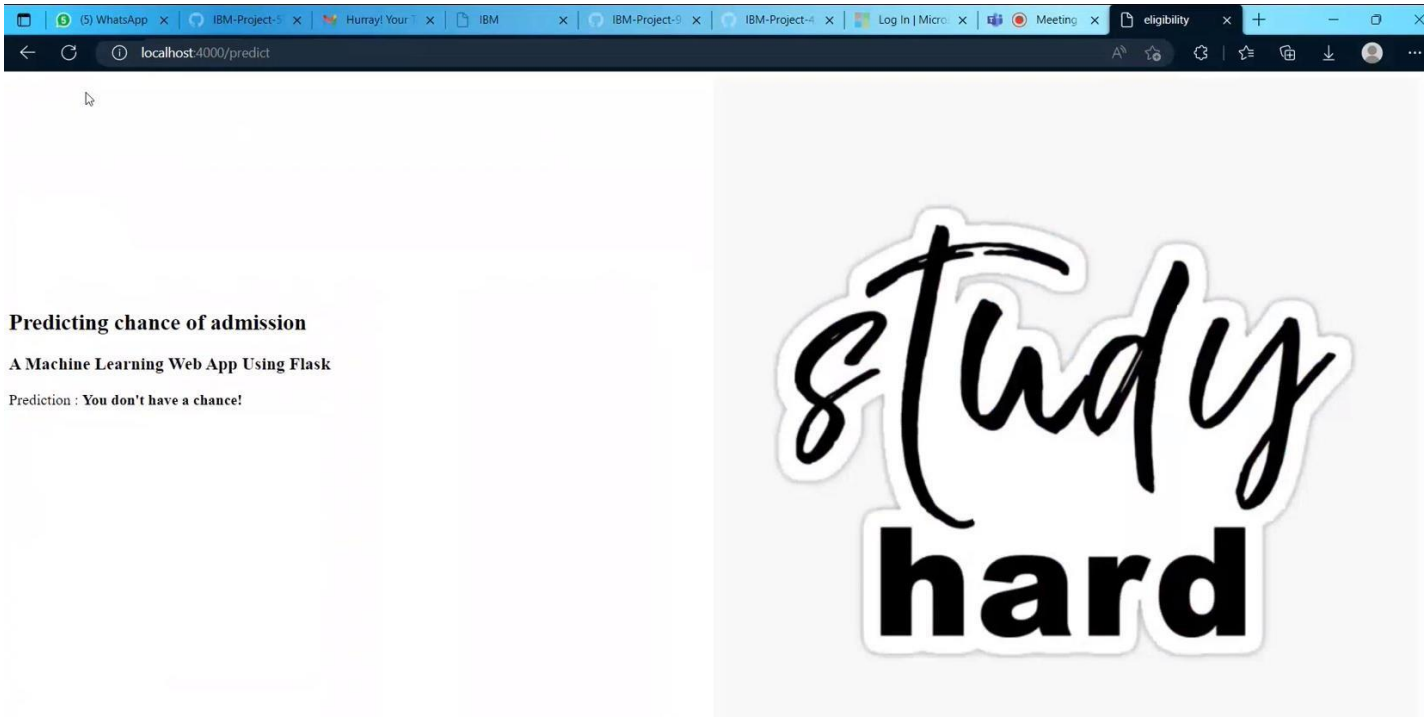
	A	B	C	D	E	F	G
1		COMMUN	RANK	COM.RAN	AGGR MA	COLLEGE (	BRANCH CODE
2	1	BC	5	3	200	1	CS
3	2	SC	7	1	200	1	CS
4	3	BC	8	4	200	1	CS
5	4	BC	10	6	200	1	EC
6	6	OC	12	8	199.75	1	EE
7	7	BC	13	8	199.75	1	CS
8	8	BC	15	10	199.75	1	CS
9	10	OC	18	12	199.75	1	CS
10	11	BC	19	12	199.75	1	EC
11	12	BC	20	13	199.75	1	EC
12	13	OC	22	16	199.75	1	EC
13	14	BC	26	16	199.75	1	CS
14	15	BC	29	17	199.75	1	CS
15	17	BC	32	19	199.75	1	EC
16	19	BC	36	22	199.667	1	EC
17	20	OC	36	23	199.667	1	CS
18	21	BC	37	23	199.667	1	CS
19	22	BC	39	25	199.5	1	CS
20	23	BC	40	26	199.5	1	CS

Input



Output


localhost:4000/predict
eligibility
Predicting chance of admission
A Machine Learning Web App Using Flask
Prediction : You've a chance to get the admission !
YOU MADE IT!


localhost:4000/predict
eligibility
Predicting chance of admission
A Machine Learning Web App Using Flask
Prediction : You don't have a chance!
study
hard

8. TESTING

Test Cases

Steps to Execute:

Verify Home Page with below UI elements:
1.Community, 2.Rank, 3.Community Rank, 4.CutOff Mark.
Application should show above UI elements. Click the text boxes and enter the inputs.
Then Click Submit Button to know your Classification.

S.No	Test Data	Expected Result	Status
1.	Community – MBC Rank – 2700 Community Rank – 700 CutOff Mark – 189.5	Application should show "You have a chance to get the admittion"	Pass
2.	Community –OC Rank – 27000 Community Rank – 1700 CutOff Mark – 190	Application should show "You don't have a chance"	Fail
3.	Community – MBC Rank – 27000 Community Rank – 1700 CutOff Mark – 190	Application should show "You have a chance to get the admittion"	Pass
4.	Community – SC Rank – 1740 Community Rank – 10 CutOff Mark – 185	Application should show "You have a chance to get the admittion"	Pass

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	0	0	2	1	3
Duplicate	1	0	0	0	1
External	0	0	1	0	1
Fixed	0	1	1	0	2
Not Reproduced	0	1	0	0	1
Skipped	0	0	0	0	0
Won't Fix	0	1	0	0	1
Totals	1	3	4	1	9

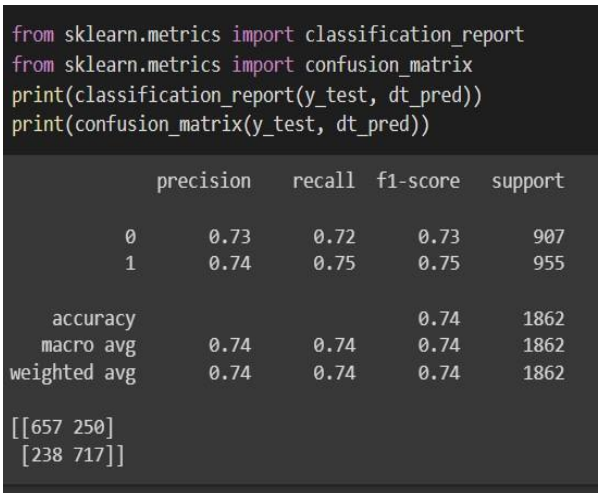
Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
View Home Page	10	0	3	7
Enter the scores	20	0	3	17
Click Submit button	2	0	0	2
Image displayed	10	0	2	8
Selecting from Drop down	7	0	0	7
Final Report Output	30	0	10	20
Version Control	5	0	2	3

9. RESULTS

Performance Metrics

S. No	Parameter	Values	Screenshot
1.	Metrics	**Decision Tree Classifier Model:** Accuracy Score	
2.	Performance Testing	**Measure the performance of the decision tree model**: Precision, Recall and F1-Score those are used for testing the performance of the model.	

10. ADVANTAGES

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

DISADVANTAGES

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

11. CONCLUSION

Student admission problem is very important in educational institutions. In this project addresses machine learning models to predict the chance of a student to be admitted. This will assist students to know in advance if they have a chance to get accepted. In this paper, machine learning models were performed to predict the opportunity of a student to get admitted to a master's program. The machine learning model included is Decision Tree Classifier model. 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

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

- This can be implemented in less time for proper admission process.
- This can be accessed anytime anywhere, since it is a web application provided only an internet connection.
- The user had not need to travel a long distance for the admission and his/her time is also saved asa result of this automated system.
- The scope of this project is a web application that allows users to enter their academic data and get predictions of their chances of admissions in the university tier of their choosing.
- It also provides an analysis based on the data set used that shows how the different affect chances of admissions.
- A Database will also be implemented for the system so that students can save their data and review and edit it as they progress with the most recent predictions being saved with their profile.
- Future work in the project could include weighing in the features that have been ignored as of yet like percentage seats for Foreign Students.
- Other criterion's like Co-curricular achievements, Leadership positions held, job experience etc can also be included as metrics for the model.

13. APPENDIX

Source Code

```python
import pickle
from flask import Flask , request, render_template
from math import ceil
app = Flask(__name__)
model = pickle.load(open("IBMDT.pkl","rb"))

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

@app.route('/predict',methods = ['GET','POST'])
def admin():
    Com = request.form["com"]
    Rank = request.form["rank"]
    ComRank = request.form["comr"]
    CutOff = request.form["co"]

    preds=[[Com,Rank,ComRank,CutOff]]
    xx=model.predict(preds)
    if (xx == 1):
        return render_template("chance.html")
    return render_template("nochance.html")
if __name__ == '__main__':
    app.run(debug = False, port=4000)
```

Github & Project Demo Link

Github Link: https://github.com/IBM-EPBL/IBM-Project-9377-1658998418
Project Demo Link: https://github.com/IBM-EPBL/IBM-Project-9377-1658998418/blob/main/Final%20Deliverables/IBM_DEMO.mp4