

UNIVERSITY ADMIT ELIGIBILITY

PREDICTOR

IBM NALAIYA THIRAN

Project Report Submitted by

JERISH JOHNSON D A(412519106058)

JANARTHANAN S(412519106054)

ARUN KUMAR K(412519106061)

HARISH M U(412519106041)

BACHELOR OF ENGINEERING

In

ELECTRONICS AND COMMUNICATION ENGINEERING

1.INTRODUCTION

1.1 PROJECT OVERVIEW

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.

1.2 PURPOSE

This project Engineering Admission Predictor System is 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

2.Literature Survey

2.1 Existing problem: The problem here is the students can't able to know their chance percentage of getting into the desired college and the difficulty they face here is at what kind of percentage is needed to get into that college.

2.2 References:

- <https://ieeexplore.ieee.org/document/9042216>
- <https://ieeexplore.ieee.org/abstract/document/9397988>
- <https://ieeexplore.ieee.org/document/8862140>
- [https://ieeexplore.ieee.org/abstract/document/9249747\](https://ieeexplore.ieee.org/abstract/document/9249747)

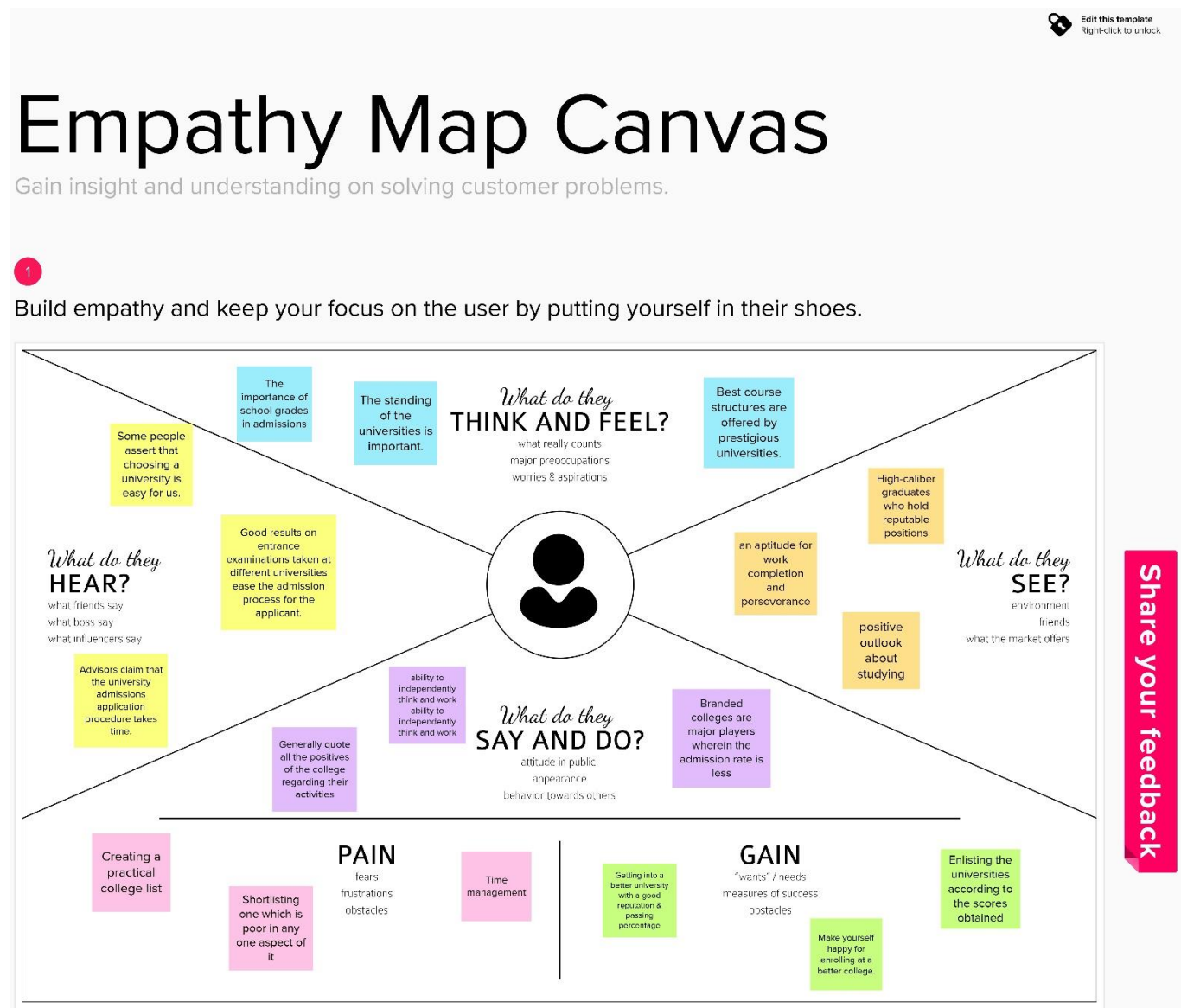
2.3 Problem Statement Definition

Concerns about getting into college are common among students. This project's goal is to assist students in narrowing down institutions based on their profiles. The anticipated results offer them a good indication of their prospects of admission to a particular university. This analysis ought to provide better insight for students who are or will be preparing for exams. The issue here is that the students are unable to determine their likelihood of entering the desired college, and they struggle to understand what kind of percentage is required to enter that institution.

3.IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

A concise, easy-to-understand image called an empathy map condenses information about a user's activities and view points. It is a useful tool that enables teams to better understand their users.



3.2 Ideation & Brainstorming

Everyone in a team is urged to participate in the process of original thought that yields problem solutions during a brainstorming session. In order to generate a wealth of unique solutions, volume over quality is prioritised, unexpected ideas are embraced and expanded upon, and everyone is urged to participate.

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

To determine a student's eligibility for admission to a university, develop a thorough prediction system that considers a range of elements, including the CGPA, Technical Skills, English proficiency test, and statement of purpose.

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 icon to edit it (even to start drawing!)

Jerish Johnson D A

Harish M U

Arun Kumar k

Janarthanan S

Give precise information about the listed university.

Details provided by peers and specialists regarding university life

Obtain score inputs

Documents and References

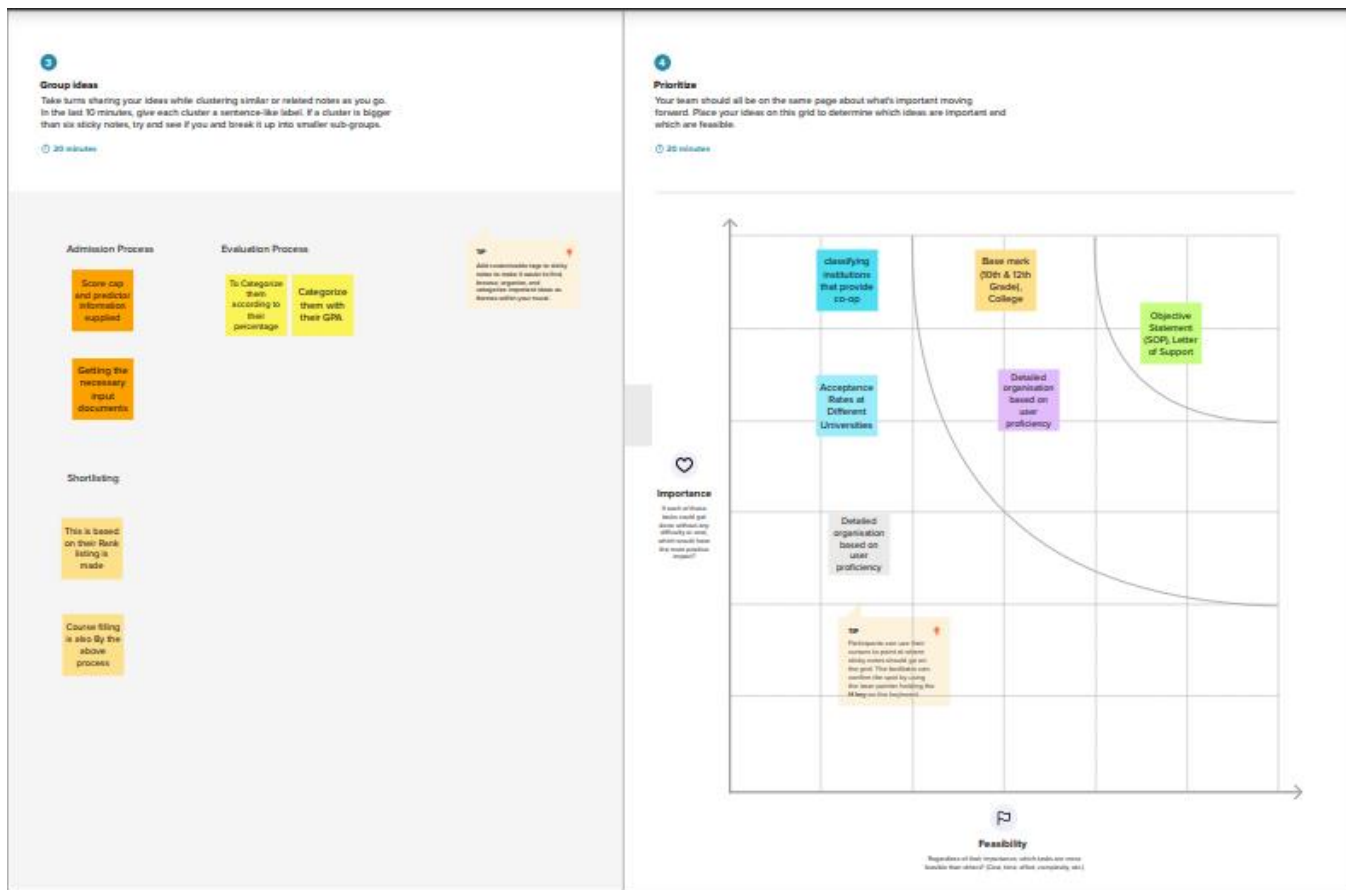
a list of potential courses that the user is qualified for

building a user-friendly website

College Facilities

Cost Estimator

Several peer testimonials and a blogspot creation should be linked.

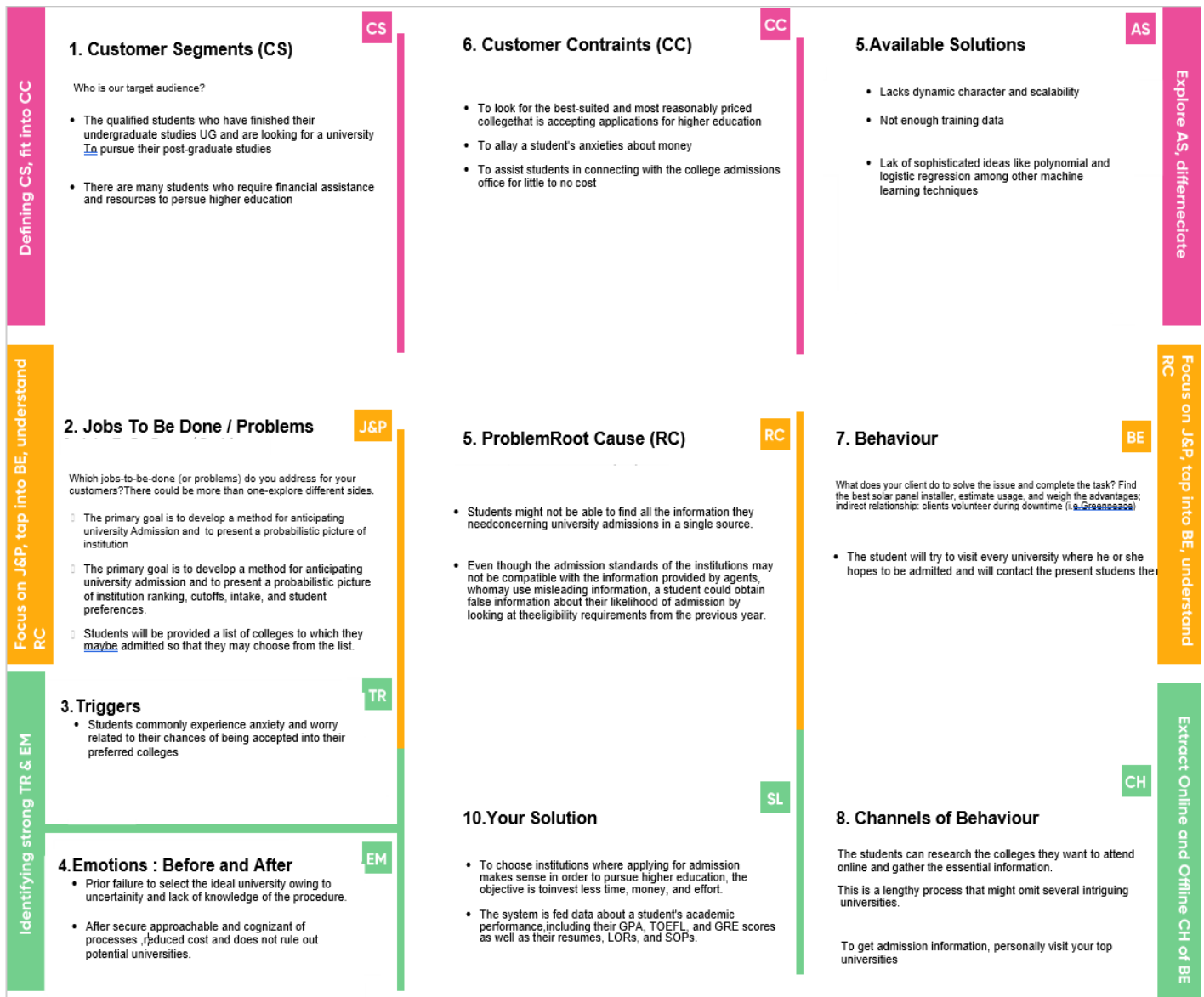


3.3 Proposed Solution

S. meter	ription
em Statement (Problem to be solved)	heck the University eligibility criteria of a student by their cut-off mark.
/ Solution description	der to immediately benefit from preparing selves in accordance with the university's rements, students and parents need a mechanism se and list the universities available for their cut-o .
lty / Uniqueness	hine language is used in order to analyse the fastest o know whether the eligibility ia is met for the particular college
l Impact / Customer Satisfaction	nterface is much easier to use so that anyone make f it easier way to know their eligibility criteria

	less Model (Revenue Model)	can be deployed so that based on this usability ges get admission and the institutions and colleges e charged for this t added in the list.
	bility of the Solution	most scalable that the seats available in thecollege based on the students ranking based on their cut-o s the seats are allotted based on their rankings

3.4 Problem Solution Fit



4. REQUIREMENT ANALYSIS

4.1 Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Data Collection	The data such as GRE score,cut-off marks are collected.
FR-4	Evaluation	The evaluation is done using the students cutoff marks,GRE score.
FR-5	Prediction	Prediction is done based on the users given data by sing using the machine learning algorithm.
FR-6	Output	Based on their cut-off marks the eligibility percentageof the selected college is shown.

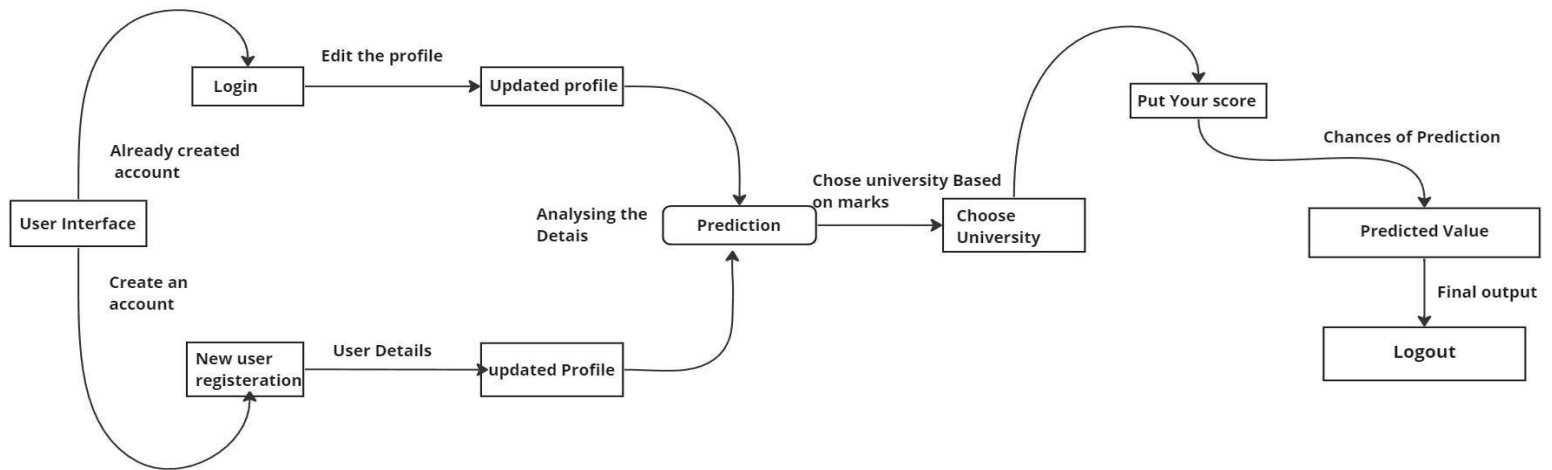
4.2 Non Funtional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Ease of Access UI Satisfaction
NFR-2	Security	Encryption is used Login used ID and password is required.
NFR-3	Reliability	The predictor system will be consistent in order forthe system to produce trustworthy and accurate outcomes.
NFR-4	Performance	A logistic regression is used

NFR-5	Availability	Since it runs on the cloud it can be accessed anywhere,anytime,anyplace.
NFR-6	Scalability	It has the ability to handle any amount of data that is given by the user.

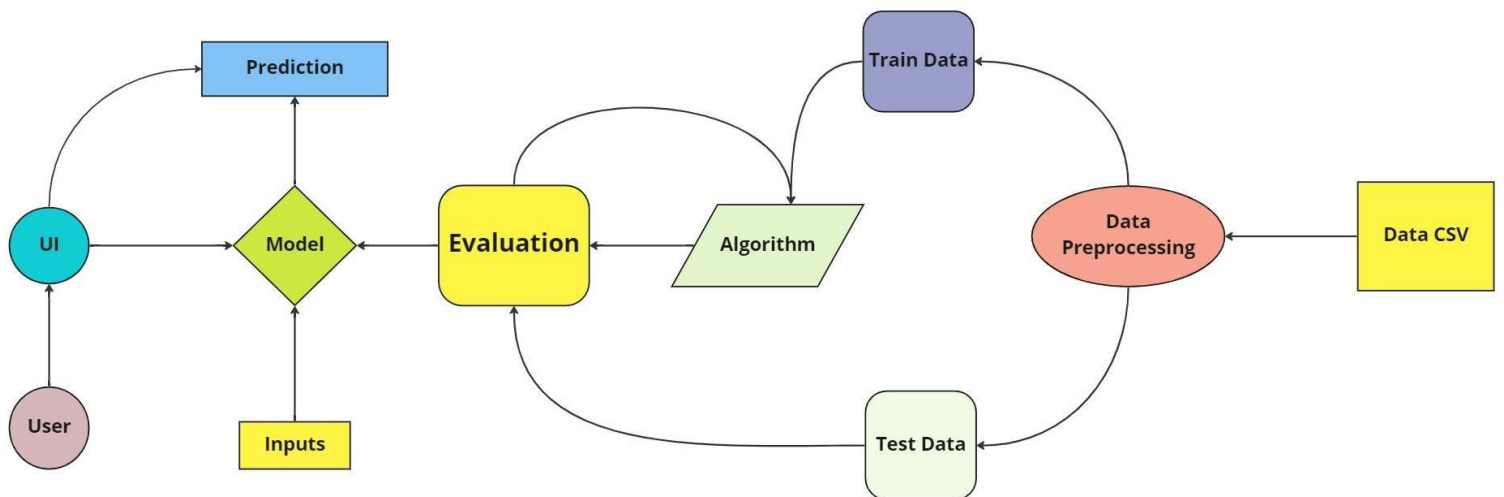
5.PROJECT DESIGN

5.1 Data Flow Diagram



miro

5.2 Solution and Technical Architecture



miro

Table:Components and Technologies

S.No	Component	Description	Technology
1	User Interface	The Front-end part of the application	HTML,CSS
2	Application Logic-1	Logic for a process in the application	Python
3	Application Logic-2	Logic for a process in the application	IBM Watson
4	Application Logic-3	Logic for a process in the application	IBM Watson
5	Database	Data type ,Configuration.	IBM cloud
6	Cloud Database	Database services on cloud	IBM DB2,IBM Cloudbant,etc.
7	Libraries	Import Libraries into data	Numpy,Pandas,Seaborn,Matplotlib
8	File Storage	File storage requirements	Local File System
9	Machine Learning Model	Purpose of Machine Learning Model	Admission Prediction Model
10	Training and testing data	Purpose of training and testing data	Logistic Regression algorithm
11	Accuracy	Accuracy of the tested and trained data	Root Mean Squared Logarithmic Error(RMSLE),Mean Squared Error(MSE),Statistics.
12	Infrastructure	Cloud Local Server Configuration	Local

Table : Application Characteristics

S.No	Characteristics	Description	Technologies Used
1	Open-Source Frameworks	The opensource Framework used here is python.	Flask Framework
2	Security Implementations	The user profile has been stored in a secured way in the cloud.	Encryptions
3	Scalable Architecture	Many computations can be done in a time saving and effective way using ML.	Logistic Regression
4	Availability	Our web application is available at anytime and at any place	IBM Balancer
5	Performance	As logistic regression is applied to develop the performance will be more effective	Logistic Regression

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	Registration	USN-1	As a user, I can create an account so that I can access it.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I can register to the application through Gmail	I can access my account	Medium	Sprint-1
	Login	USN-3	As a user, I can log into the application by entering the login credentials..	I can access my account	High	Sprint-1
	Edit Profile	USN-4	As a user, after logging in, I will have to update my profile by providing all the required details.	I can complete the profile to proceed with the prediction process.	High	Sprint-2
	Choose University	USN-5	As a user, I will be able to view the list of Universities that the students are eligible to apply based on the cutoff given.	I can put the cut off mark and the GRE score so that I can see the universities and chances of getting it	High	Sprint-3
	Choose Course	USN-6	As a user, I will be able to view the list of courses that the students are eligible to apply.	I can be able to see the list of colleges available to me.	Medium	Sprint-3
	Admission Process	USN-7	As a user, I will be able to view the chances of getting that college based on my cutoff given.	I can view the details of Admission process being displayed at the end of prediction.	High	Sprint-4
Administrator	Authentication	USN-8	As an admin, the login credential of the user is authenticated by me.	I can retrieve and make use of all the user details.	High	Sprint-1
	Update Profile	USN-9	As an admin, I can verify the user entered details.	I can confirm and access the user details.	High	Sprint-2

	Prediction	USN-10	As an admin, I can test the trained ML	I can test the user data	High	Sprint-3
			model by analysing the user details by ML algorithms like Logistic Regression.	with the trained ML model.		
	Output	USN-12	As an admin, I can upload the confirmation of user for the prediction into the Database.	I can keep track of user data and activities by storing in database.	High	Sprint-4

6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning and 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	low	Harish,Arun kumar
Sprint-1	Entering cutoff marks	USN-2	As a user I can enter my cut off scores	3	High	Harish , Arun kumar
Sprint-1	UI	USN-3	The user interface is created	4	high	Jerish Johnson,Janarthanan
Sprint-2	Model Building	USN-4	Using the dataset the model can be build using ML algorithm	6	high	Arunkumar,Janarthanan
Sprint-2		USN-5	Training the classification model	5	High	Janarthanan harish
Sprint-3	Application buiding	USN-6	Building the python code to run the application	5	high	Jerish,harish
Sprint-3	Testing	USN-7	Testing the ML model	3	low	Harish,Arun kumar
Sprint-3		USN-8	Predicted result is shown on the website	4	Medium	Jerish,harish
Sprint-4	Link the model	USN-9	Linking the HTML and the python code	4	hign	Jerish,janarthanan
Sprint-4	Deployment of the project	USN-10	Deployment in the IBM cloud	6	high	Jerish Johnson,Janarthanan

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	3	29 October
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	4	5 November
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	4	10 November
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	5	15 November

6.3 Reports From jira



7.CODING AND SOLUTIONING

7.1 Feature 1

Predict Your Result

GRE (Out of 340)

TOEFL (Out of 120)

CGPA (Out of 10)

Predict

F:\> Project Fianl jerish johnson > Final Deliverables > templates > index.html > ...

```
1
2 <!DOCTYPE html>
3 <html>
4 <!--From https://codepen.io/frytyler/pen/EGdtg-->
5 <head>
6   <meta charset="UTF-8">
7   <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
8   <title>Graduate Admission</title>
9   <link href="https://fonts.googleapis.com/css?family=IBM+Plex+Sans&display=swap" rel="stylesheet">
10  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
11  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" integrity="sha384-Gn5384xqQ1aolwXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6J"
12
13
14 </head>
15
16 <body style="background-color: #181818">
17   <div class="card" style="width: 28rem;margin: 0 auto;float: none;margin-bottom: 10px;margin-top: 30px;font-family: 'IBM Plex Sans', sans-serif;background-color: #282828">
18   <article class="card-body">
19     <h4 class="card-title text-center mb-4 mt-1" style="color: white">Predict Your Result</h4>
20     <hr>
21     <form action="{{ url_for('predict')}}" method="post">
22     <div class="form-group">
23     <div class="input-group">
24       <div class="input-group-prepend">
25         <span class="input-group-text"> <i class="fa fa-book"></i> </span>
26       </div>
27       <input name="GRE" class="form-control" placeholder="GRE (Out of 340)" type="text" required/>
28     </div> <!-- input-group.// -->
29   </div> <!-- form-group// -->
30   <div class="form-group">
31   <div class="input-group">
32     <div class="input-group-prepend">
33       <span class="input-group-text"> <i class="fa fa-book"></i> </span>
34     </div>
35     <input name="TOEFL" class="form-control" placeholder="TOEFL (Out of 120)" type="text" required/>
36   </div> <!-- input-group.// -->
37 </div> <!-- form-group// -->
38 <div class="form-group">
39 <div class="input-group">
40   <div class="input-group-prepend">
```

```
41     <span class="input-group-text"> <i class="fa fa-book"></i> </span>
42   </div>
43   <input name="CGPA" class="form-control" placeholder="CGPA (Out of 10)" type="text" required/>
44 </div> <!-- input-group.// -->
45 </div> <!-- form-group// -->
46
47 <div class="form-group">
48 <button type="submit" class="btn btn-primary btn-block" style="background-color: #484848;border-color: #484848"> Predict </button>
49 </div> <!-- form-group// -->
50 <p class="text-center" style="color: white">{{ prediction_text }}</p>
51 </form>
52 </article>
53 </div> <!-- card.// -->
54 <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCKR/rE9/QpG6aZGJwFDMVNA/GpGFF93hXpG5KkKl" crossorigin="anonymous"></script>
55 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="
56 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PVCmY1" crossorigin="anc
57 </body>
58 </html>
```

Flask File launch website into IBM cloud

```
1  import numpy as np
2  from flask import Flask, request, jsonify, render_template
3  import pickle
4  from gevent.pywsgi import WSGIServer
5  import os
6
7  app = Flask(__name__)
8  model = pickle.load(open('model.pkl', 'rb'))
9
10 @app.route('/')
11 def home():
12     return render_template('index.html')
13
14 @app.route('/predict', methods=['POST'])
15 def predict():
16     """
17     For rendering results on HTML GUI
18     """
19     int_features = [float(x) for x in request.form.values()]
20     final_features = [np.array(int_features)]
21     prediction = model.predict(final_features)
22
23     output = round(prediction[0], 2)*100
24
25     if(output>100):
26         return render_template('index.html', prediction_text='Chance of Admission: 100%')
27     if(output<0):
28         return render_template('index.html', prediction_text='Chance of Admission: 0%')
29
30     return render_template('index.html', prediction_text='Chance of Admission: {}'.format(output))
31 port = os.getenv('VCAP_APP_PORT', '8080')
32
33 if __name__ == "__main__":
34     app.secret_key=os.urandom(12)
35     app.run(debug=True,host='0.0.0.0',port=port)
36
```

8.TESTING

8.1 Test Cases

				Date	16-Nov-22								
				Team ID	PNT2022TMD04094								
				Project Name	University Admit Eligibility Predictor								
				Maximum Marks	4 marks								
Test case ID	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
LoginPage_TC_002	UI	Index	Verify the UI elements in home page		1.Enter URL and click go 2.Enter the Scores 3.Click the Submit button	http://127.0.0.1:5000/home	Working as expected	Working as expected	Pass				
LoginPage_TC_003	Functional	Chance	Verifying whether the student is eligible for admission		1.As per the Entered Model Value. 2.Getting above 50%. 3.You have a Chance will get displayed	http://127.0.0.1:5000/chance/90.174255759468	Working as expected	Working as expected	Pass				
LoginPage_TC_004	Functional	NoChance	Verifying whether the student does not have a chance of admission		1.As per the Entered Model Value. 2.Getting below 50%. 3.You have a Low/Nochance will get displayed	http://127.0.0.1:5000/nochance/41.52682121752442	Working as expected	Working as expected	Pass				

8.2 User Acceptance testing

8.2.1 Defect Analysis

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

8.2.2 Test case analysis

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 Performane Metrics

S.No.	Parameter	Values	Screenshot
1.	Metrics	<p>Regression Model:</p> <p>MAE - 0.04439100712864568 ,</p> <p>MSE -0.003969262204179716</p> <p>RMSE - ,0.063</p> <p>R2 score -0.8279343840569254</p> <p>Classification Model:</p> <p>Confusion Matrix - , Accuray</p> <p>Score- & Classification Report -</p>	<pre> from sklearn.metrics import r2_score, mean_squared_error, mean_absolute_error from math import sqrt RMSE = float(format(np.sqrt(mean_squared_error(y_test_orig, y_predict_orig)), '.3f')) MSE = mean_squared_error(y_test_orig, y_predict_orig) MAE = mean_absolute_error(y_test_orig, y_predict_orig) r2 = r2_score(y_test_orig, y_predict_orig) adj_r2 = 1-(1-r2)*(n-1)/(n-k-1) print('RMSE =', RMSE, '\nMSE =', MSE, '\nMAE =', MAE, '\nR2 =', r2, '\nAdjusted R2 =', adj_r2) </pre> <p> RMSE = 0.063 MSE = 0.003969262204179716 MAE = 0.04439100712864568 R2 = 0.8279343840569254 Adjusted R2 = 0.8047717049876654 </p>

2.	Tune the Model	Hyperparameter Tuning - Validation Method -	<pre> from sklearn import datasets from sklearn.tree import DecisionTreeClassifier from sklearn.model_selection import StratifiedKFold, cross_val_score X, y = datasets.load_iris(return_X_y=True) clf = DecisionTreeClassifier(random_state=42) sk_folds = StratifiedKFold(n_splits = 5) scores = cross_val_score(clf, X, y, cv = sk_folds) print("Cross Validation Scores: ", scores) print("Average CV Score: ", scores.mean()) print("Number of CV Scores used in Average: ", len(scores)) </pre> <p> Cross Validation Scores: [0.96666667 0.96666667 0.9 0.93333333 1. Average CV Score: 0.9533333333333334 Number of CV Scores used in Average: 5 </p>
----	-------------------	--	--

10. ADVANTAGES AND DISADVANTAGES

Advantages:

- It aids students in picking the appropriate college.
- When compared to the current system, this one has a lower mistake probability.
- It is quick, effective, and dependable.
- Prevents inconsistent and redundant data.
- Very approachable.
- Data are readily available

Disadvantages:

- A live internet connection is necessary.

- If data is entered incorrectly, the system will produce inaccurate results.

11. CONCLUSION

Thus, it can be said that our model, which uses the Multiple Linear Regression Algorithm, offers a reliable prediction score and that users can guess the scores accurately. The website's user interface is straightforward and makes it easy for users to use the feature.

12. FUTURE SCOPE

By including information about each university, we hope to improve the user experience by allowing users to learn more about the local culture, alumni testimonials, university rankings, and other factors.

13. APPENDIX

Source Code : <https://github.com/IBM-EPBL/IBM-Project-22193-1659807644/tree/main/Final%20Deliverables>

Project Demo: <https://github.com/IBM-EPBL/IBM-Project-22193-1659807644/blob/main/Final%20Deliverables/Project-Demonstration-Video.mp4>