

UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

A PROJECT REPORT

Domain : Applied Data Science

Team ID : PNT2022TMID47303

College Name : Government College Of Engineering Srirangam

Submitted by

AJITH N

830119104003

BAGHATRAJ N

830119104006

MADHAN PRAKASH T

830119104019

VIGNESH S

830119104052

CHAPTER	TITLE	PAGE NO.
1	INTRODUCTION 1.1 PROJECT OVERVIEW 1.2 PURPOSE	3 3
2	LITERATURE SURVEY 2.1 EXISTING PROBLEM 2.2 REFERENCES 2.3 PROBLEM STATEMENT DEFINITION	4 4 4
3	IDEATION & PROPOSED SOLUTION 3.1 EMPATHY MAP CANVAS 3.2 IDEATION & BRAINSTORMING 3.3 PROPOSED SOLUTION 3.4 PROBLEM SOLUTION FIT	5 6 7 8
4	REQUIREMENT ANALYSIS 4.1 FUNCTIONAL REQUIREMENT 4.2 NON-FUNCTIONAL REQUIREMENTS	9 10
5	PROJECT DESIGN 5.1 DATA FLOW DIAGRAM 5.2 SOLUTION & TECHNICAL ARCHITECTURE 5.3 USER STORIES	11 11 13
6	PROJECT PLANNING & SCHEDULING 6.1 SPRINT PLANNING & ESTIMATION 6.2 SPRINT DELIVERY SCHEDULE 6.3 REPORTS FROM JIRA	14 15 15
7	CODING & SOLUTIONING 7.1 FEATURE 1 7.2 FEATURE 2 7.3 DATABASE SCHEMA	16 19 20
8	TESTING 8.1 TEST CASES 8.2 USER ACCEPTANCE TESTING	21 22
9	RESULTS 9.1 PERFORMANCE METRICS	23
10	ADVANTAGES & DISADVANTAGES	25
11	CONCLUSION	26
12	FUTURE SCOPE	27

13	APPENDIX	
	13.1 SOURCE CODE	28
	13.2 GITHUB & PROJECT DEMO LINK	31

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

Students admission is very important part of their life. It decides how their life is go in the future. This paper addresses machine learning models to predict the chance of a student to be admitted to a university. This project is a web-based University Admission Predictor. It is an AI based application that asks for the users to input their academic transcripts data and calculates their chances of admission into the University Tier that they selected using machine learning models. It also provides an analysis of the data and shows how chances of admissions can depend on various factors.

1.2 PURPOSE

The purpose of creating a web-based University Admission Predictor is to help students to find their chances of getting admitted into their favorite university. This helps in reducing the workload, stress and time consumption. It is fast, reliable and efficient. It avoids data redundancy and inconsistency. It mainly helps the students who are from rural areas and find the students chances of getting admitted into their favourite university.

CHAPTER 2

LITERATURE SURVEY

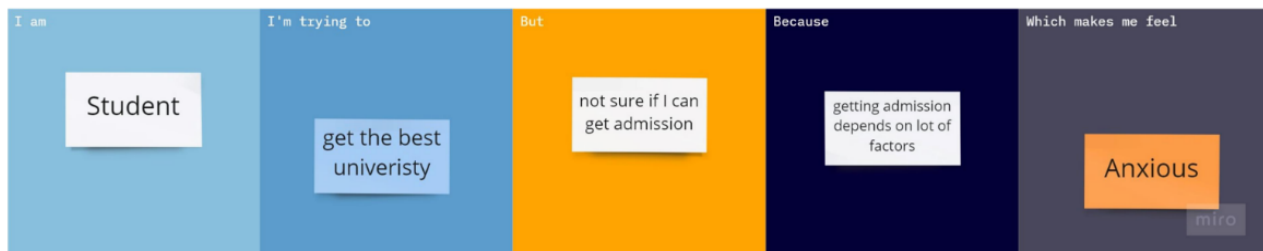
2.1 EXISTING PROBLEM

Previous research done in this area used Naive Bayes algorithm which will evaluate the success probability of student application into a respective university but the main drawback is they didn't consider all the factors which will contribute in the student admission process like TOEFL/IELTS, SOP, etc,. This model was developed to forecast the progress of prospective students by comparing the score of students currently studying at university. The model thus predicted whether the aspiring student should be admitted to university based on various scores of students. Since the comparisons are made only with students who got admission into the universities but not with students who got their admission rejected so this method will not be that much accurate.

2.2 REFERENCES

1. Vandit Manish Jain, Rihaan Satia, "College Admission Prediction using Ensemble Machine Learning Models".
2. Sujay S, "Supervised ML Modelling & Analysis for Graduate Admission Prediction".
3. Mohan S Acharya, Asfia Armaan, Aneeta S Antony, "A Comparison of Regression Models for Prediction of Graduate Admissions".
4. Sashank Sridhar, Siddhartha Mootha, Santosh Kolagat, "A University Admission Prediction System using Stacked EnsembleLearning".
5. A. Sivasangari, V. Shivani, Y. Bindhu, D. Deepa, R. Vignesh, "Prediction Probability of Getting an Admission into a University using ML".

2.3 PROBLEM STATEMENT DEFINITION



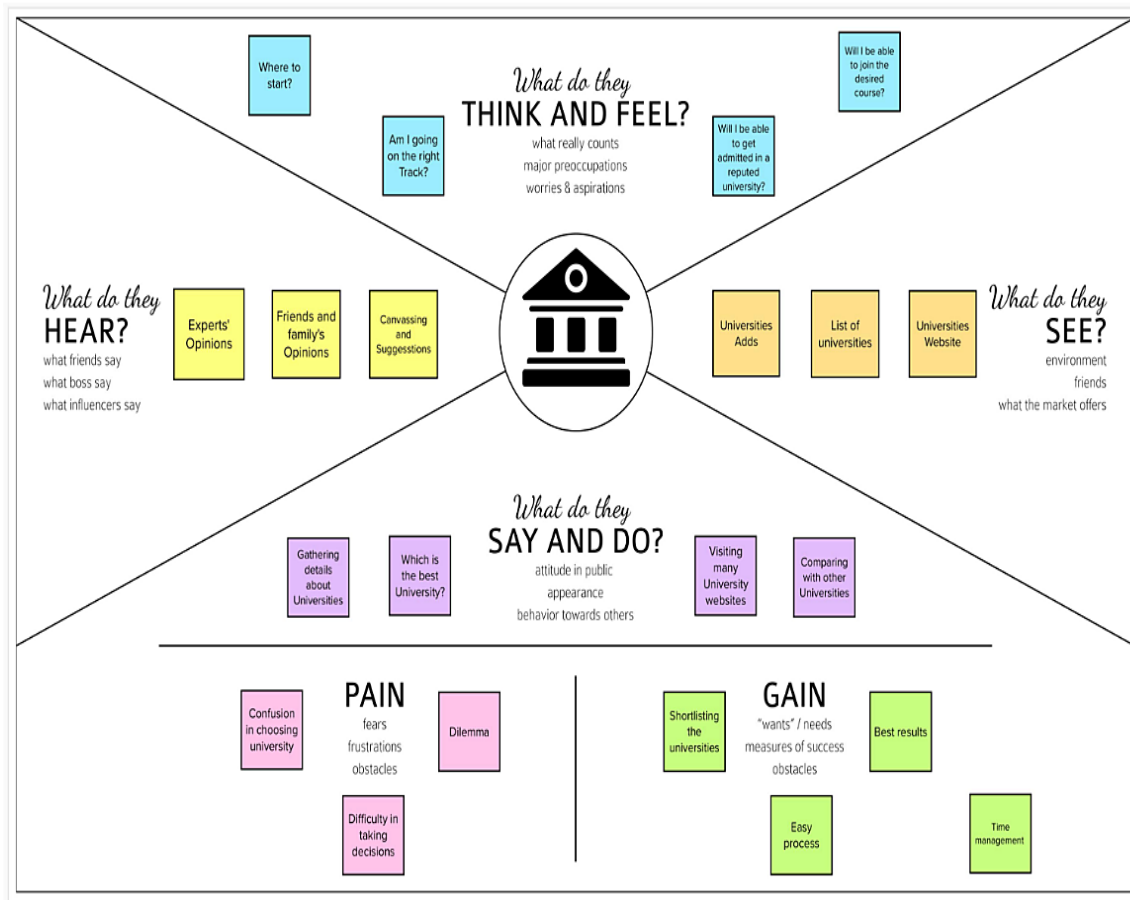
CHAPTER 3

IDEATION & PROPOSED SOLUTION

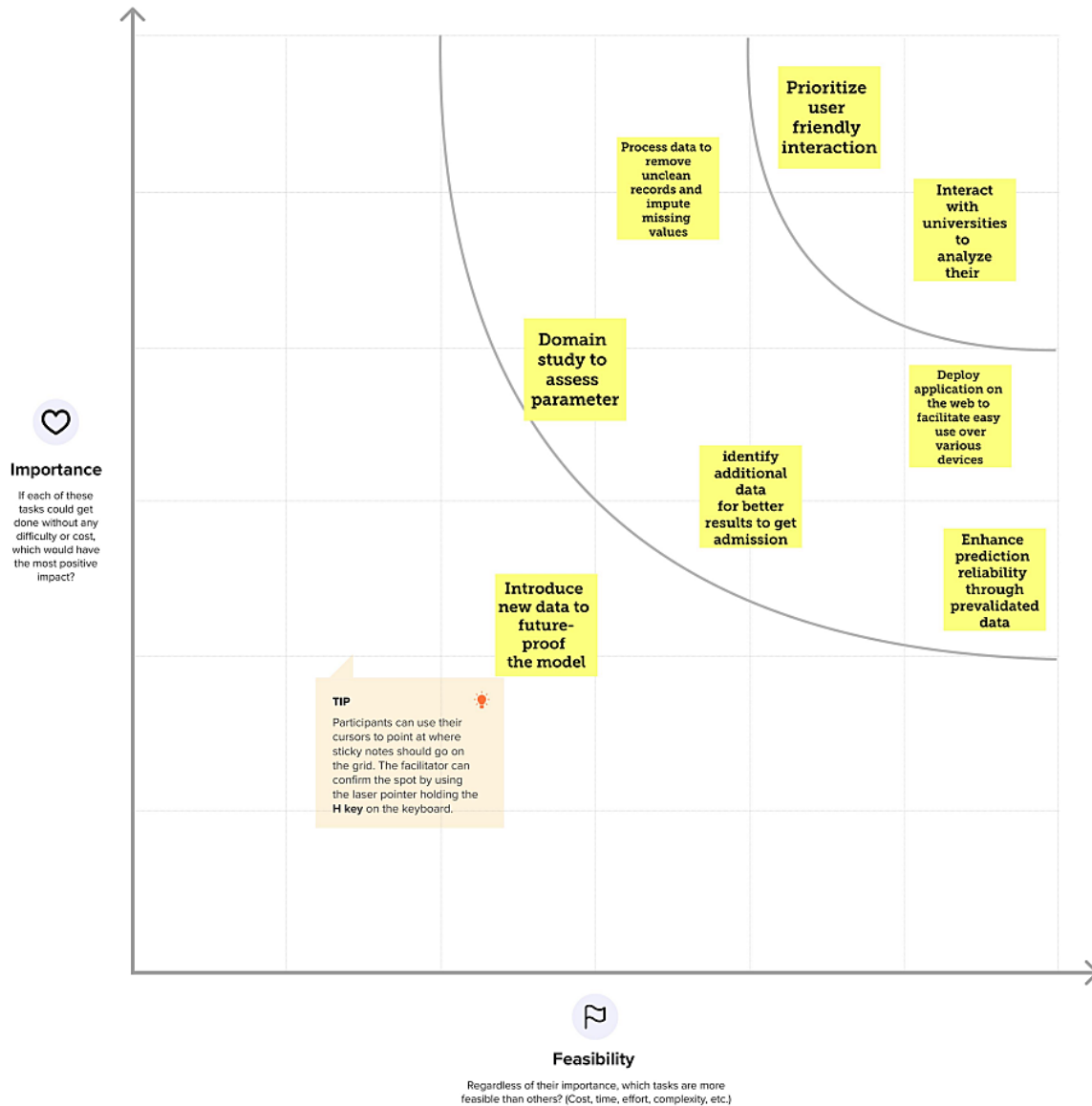
3.1 EMPATHY MAP CANVAS

Empathy Map Canvas:

University Admit Eligibility Predictor



3.2 IDEATION AND BRAINSTORMING



3.3 PROPOSED SOLUTION

S.NO	PARAMETERS	DESCRIPTION
1	Problem Statement	I am a student who is trying to choose a right university based on score but I couldn't because I am not aware of eligibility criteria which makes me feel frustrated.
2	Idea/Solution description	This idea helps students to get the list of university to which they can apply as the system shortlist the colleges by comparing the students marks and cut off to predict the possibility of admission or not.
3	Novelty/Uniqueness	The main advantages of the project are the computerization of the entrance seat allotment process. This makes the process easier and takes very less time when compared to the existing system.
4	Social Impact/Customer Satisfaction	It helps the students for making decision for right college. Here the chance of occurring errors is comparatively less. It is fast and efficient. Avoid the data repetition and inconsistency. It helps you to understand as to how your profile can be further improved to secure an admit in your preferred college. It can guide you whether you need to take GRE or not, in order to improve your chances of admit in your target university.
5	Business Model	Institutions are under increasing pressure to admit more students, retain these students, and do their best to ensure the students success. This project not only relieves students from pressure but also helps these institutions to get relieved from their pressure.
6	Scalability of the Solution	This will also help you to finalize your dream university with a realistic roadmap, with the help of your information coupled with a bit of reality check on your academic scores such as GRE, TOFEL, etc., working experience and your CGPA. On the other hand, we have connoisseurs who shall work with you to amplify your prospects of receiving offers by ensuring that the universities that you apply, do not digress from your profile, and chiefly your ambition. The concern is privacy and ownership for both students and teachers.

3.4 PROBLEM SOLUTION FIT

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS Who is your customer? I.e. working parents of 0-5 y.o. kids <div>Students who have completed HSC.</div>	6. CUSTOMER CONSTRAINTS CC What constraints prevent your customers from taking action or limit their choices of solutions? I.e. spending power, budget, no cash, network connection, available devices. <div>Waste of Money, Time and Energy due to Poor Knowledge.</div>	5. AVAILABLE SOLUTIONS AS Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? <div>Seat allotment, Eligibility Criteria due to Entrance Exam like NEET, JEE, etc.,</div>	Explore AS, differentiate
	2. JOBS-TO-BE-DONE / PROBLEMS J&P Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides. <div>Want to get admission in preferred University.</div>	9. PROBLEM ROOT CAUSE RC What is the real reason that this problem exists? What is the back story behind the need to do this job? I.e. customers have to do it because of the change in regulations. <div>Due to high Competition and not meeting the required criteria.</div>	7. BEHAVIOUR BE What does your customer do to address the problem and get the job done? I.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (I.e. Greenpeace) <div>At the Last Moment Spend more Money and Time to get into the preferred University.</div>	
Identify strong TR & EM	3. TRIGGERS TR What triggers customers to act? I.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. <div>Watching Other Students trying into getting into one's Favourite University.</div>	10. YOUR SOLUTION SL If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. <div>We would create an application that helps students to get the list of colleges by comparing the student's marks and college's cut off and predicting admission probability. It is fast, efficient and reliable. It helps you to understand as to how your profile can be further improved to secure an admit in your preferred university.</div>	8. CHANNELS of BEHAVIOUR CH 8.1 ONLINE What kind of actions do customers take online? Extract online channels from #7 <div>They will search online about the preferred university and the criteria to join the University</div>	Extra Online & Offline of CH and BE
	4. EMOTIONS: BEFORE / AFTER EM How do customers feel when they face a problem or a job and afterwards? I.e. lost, insecure > confident, in control - use it in your communication strategy & design. <div>Confused, Anxious over whether one getting Admission in the University.</div>	8.2 OFFLINE What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. <div>Visiting College campus, Enquire students, Academic representatives and nearby people about the University.</div>		

CHAPTER 4

REQUIREMENTS ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

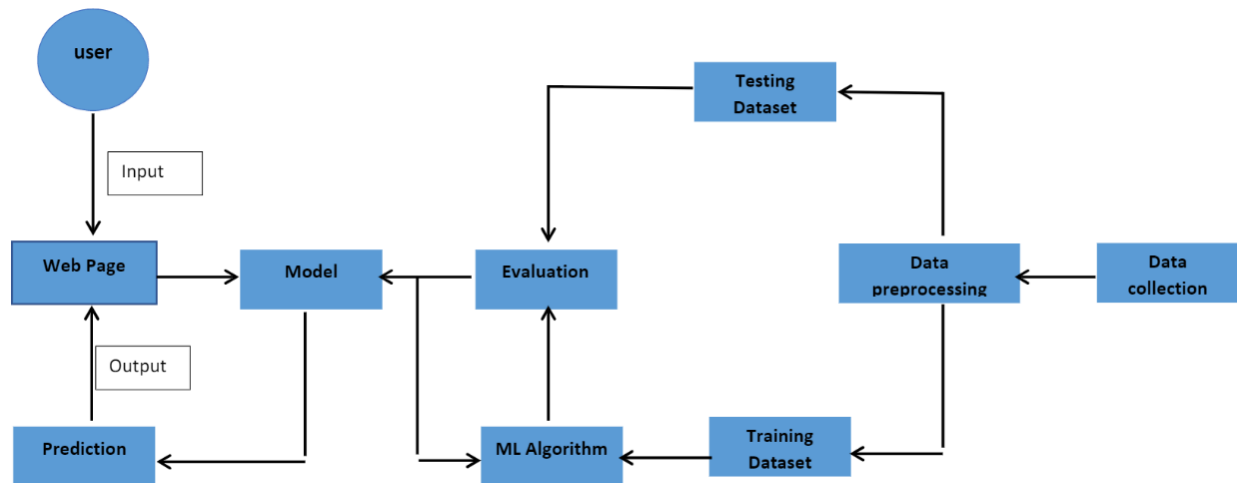
FR NO.	FUNCTIONAL REQUIREMENT (EPIC)	SUB REQUIREMENT (STORY/ SUB-TASK)
FR-1	User Registration	Registration through Form
FR-2	User Information	<p>All the grades and scores necessary for the user's admission will need to be provided.</p> <p>These include,</p> <ul style="list-style-type: none"> a. English Proficiency Test score - TOEFL score out of 120 marks b. Knowledge Evaluation Test score - GRE score out of 340 marks c. High School / Undergraduate CGPA out of 10 point d. Collect other info about University Rating, SOP, LOR and Research data from the users.
FR-3	Result Display	<p>The user should complete the following tasks to get their admission prediction:</p> <ul style="list-style-type: none"> a. enter the test scores required for admission prediction. b. The user's chances of acceptance will be provided and sent through mail or SMS.
FR-4	User Confirmation	<p>Confirmation via Email</p> <p>Confirmation via OTP</p>

4.2 NON - FUNCTIONAL REQUIREMENTS

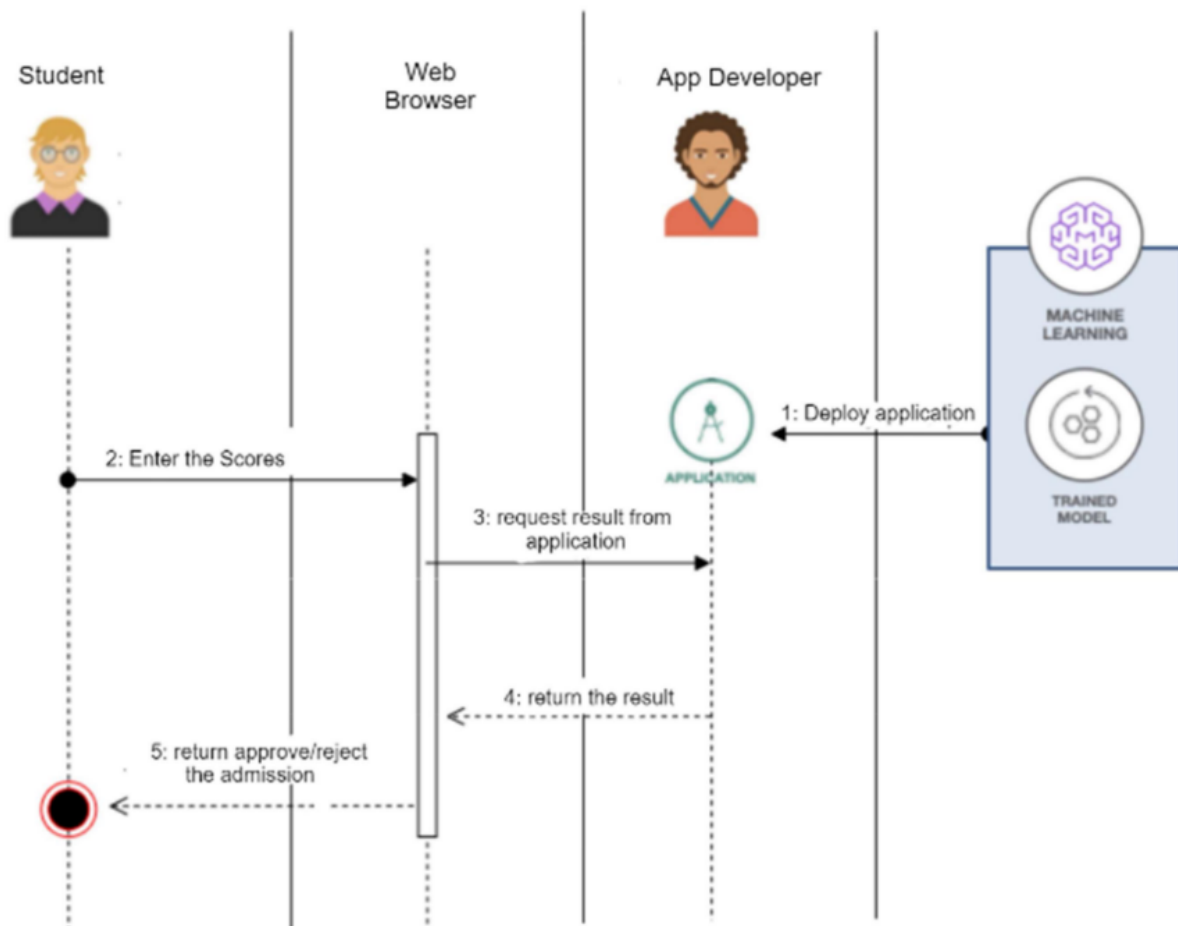
FR NO.	NON-FUNCTIONAL REQUIREMENT	DESCRIPTION
NFR-1	Usability	<ul style="list-style-type: none">a. User-Friendly.b. No technical Experience is required to use the website.c. It takes less time to show the output.
NFR-2	Security	<ul style="list-style-type: none">a. Standard authentication protocols will be implemented.b. Data is secure.
NFR-3	Reliability	<ul style="list-style-type: none">a. High accuracy so it can be reliable for the users to make decisions.b. Easy-to-use interface, thus the user can share or recommend the solution to friends.
NFR-4	Performance	<ul style="list-style-type: none">a. This system can support any number of users at a time.b. Efficiently optimized to provide results as soon as possible given the speed of the user's internet connection.
NFR-5	Availability	<ul style="list-style-type: none">a. The solution will be available 24/7.b. Avoids data redundancy and inconsistency.c. It is fast, efficient and reliable.d. A chance of occurrence of error is less when compared to existing system.
NFR-6	Scalability	<ul style="list-style-type: none">a. The accuracy of the results can also be improved by integrating another ML approach if it is found to be more effective.b. The system can be improved to handle more concurrent users if available capacity occurs.

CHAPTER 5 PROJECT DESIGN

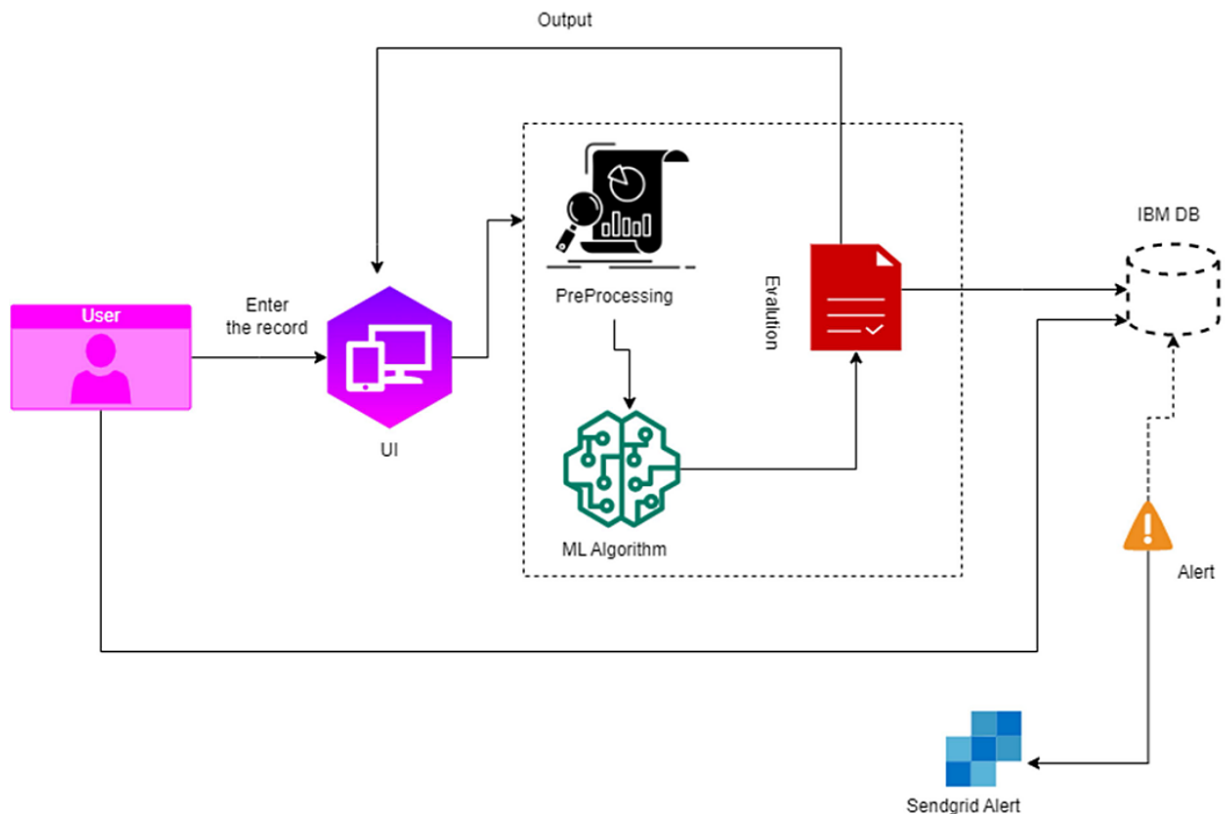
5.1 DATA FLOW DIAGRAMS



5.2 SOLUTION ARCHITECTURE



TECHNICAL ARCHITECTURE



5.3 USER STORIES

USER TYPE	FUNCTIONAL REQUIREMENT	USER STORY NUMBER	USER STORY / TASK	ACCEPTANCE CRITERIA	PRIORITY	RELEASE
Customer	Registration	USN -1	As a user, I can register for the website by entering my email, password and confirming my password	I can access my account and dashboard	High	Sprint - 1
		USN - 2	As a user I will receive confirmation email once I have registered for the website	I can receive confirmation email & click confirm	High	Sprint - 1
		USN -3	As a user I can also enter the marks without registering.	I can enter scores	Medium	Sprint - 3
	Login	USN - 4	As a user, I can log into the website by entering email & password	I can access the Dashboard	High	Sprint - 2
	Dashboard	USN - 5	As a user, I can enter the scores	I can find eligibility	High	Sprint - 3
		USN - 6	As a user I can view the prediction	I can See the results	High	Sprint - 3
		USN - 7	As a viewer I can view my results in the email.	I get results in my email	Medium	Sprint - 4

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

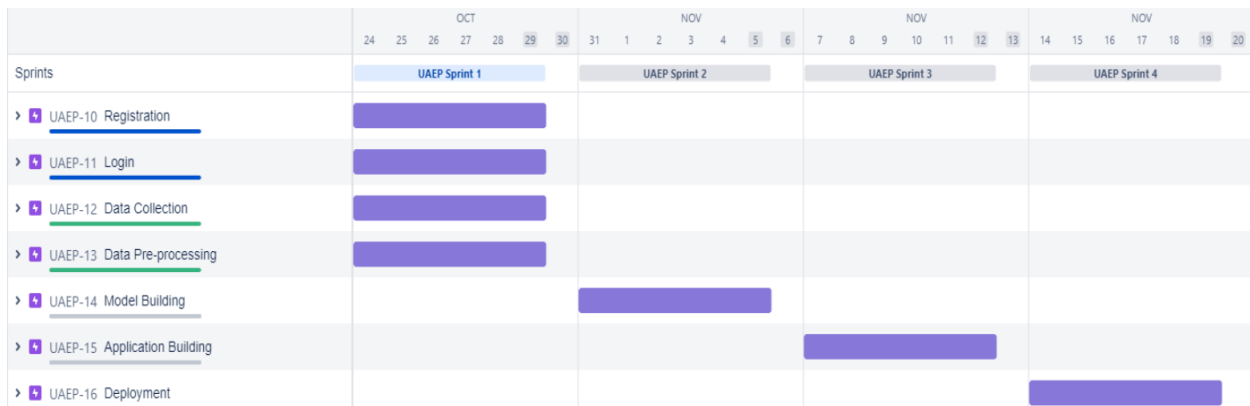
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	Medium	Madhan Prakash, Vignesh
Sprint-1	Login	USN-2	As a user, I can log into the application by entering email & password	1	High	Madhan Prakash, Vignesh
Sprint-1	Data Collection	USN-3	Gathering the information from various resources	1	Medium	Ajith, Baghatraj, Madhan Prakash, Vignesh
Sprint-1	Data Preprocessing	USN-4	To Convert and clean the raw data	2	High	Ajith, Baghatraj
Sprint-2	Model Building	USN-5	Using cleaned dataset, Model can be buildusing ML Algorithm	2	High	Ajith, Baghatraj
Sprint-2		USN-6	Training the classification model	1	High	Ajith, Baghatraj
Sprint-3	Application Building	USN-7	Building Python code and run the application	1	Medium	Madhan Prakash, Vignesh

Sprint-3		USN-8	Predicted Result has been sent to user registered mail	1	Medium	Ajith, Baghatraj, Madhan Prakash, Vignesh
Sprint-4	Implementation of the application and deployment on cloud	USN-9	Deployed on IBM Cloud	2	High	Ajith, Baghatraj, Madhan Prakash, Vignesh

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 ReleaseDate (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



CHAPTER 7

CODING & SOLUTIONING

7.1 FEATURE 1

- Analysed university admission statistics.
- Languages : Python
- IDE : Anaconda
- Libraries : Recommended

app.py

```
1  from flask import render_template, Flask, request, redirect, url_for, g, flash, session
2  from flask_mysql import MySQL
3  import MySQLdb.cursors
4  import re
5  import numpy as np
6  import pandas as pd
7  import joblib
8
9  from send_mail import send_email, fail_mail, linear_mail
10
11  app = Flask(__name__)
12  app.secret_key = '12345'
13  app.config['MYSQL_HOST'] = 'localhost'
14  app.config['MYSQL_USER'] = 'root'
15  app.config['MYSQL_PASSWORD'] = ''
16  app.config['MYSQL_DB'] = 'uaeap'
17
18  mysql = MySQL(app)
19
20
21  """Prediction code"""
22  scaler = joblib.load(open("./models/scaler.pkl", "rb"))
23  print('Scaler Model Loaded')
24  university = joblib.load(open("./models/university.pkl", "rb"))
25  print('Naive Model Loaded')
26  linear_model = joblib.load(open("./models/linear_model.pkl", "rb"))
27  print('Linear Model loaded')
28
29  # MAIN PAGE
30  @app.route('/', methods=['GET', 'POST'])
31  def main():
32      return render_template('index.html')
33
34
35
36  @app.before_request
37  def load_user():
38
39
40
41
42
43
44
45  #LOGIN page
46  @app.route('/loginpage', methods = ['GET', 'POST'])
47  def loginpage():
48      if request.method == 'POST':
49          username = request.form.get('username')
50          password = request.form.get('password')
51          cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
52          cursor.execute('select * from details where username = %s AND password = %s', (username, password))
53          account = cursor.fetchone()
54
55          if account:
56              session['loggedin'] = True
57              session['username'] = account['username']
58              session['email'] = account['email']
59              return render_template('prediction.html', username=session['username'], logout='logout')
60          else:
61              return render_template('sign-in.html', msg = 'username and password not found')
62      return render_template('sign-in.html')
63
64
65  #logout
66  @app.route('/logout')
67  def logout():
68      session.pop('username', None)
69      return render_template('index.html')
70
71  #predict page
72  @app.route('/predict', methods=['GET', 'POST'])
```

```

74     if request.method == 'POST':
75         toefl = request.form.get('toefl')
76         sop = request.form.get('sop')
77         lor = request.form.get('lor')
78         cgpa = request.form.get('cgpa')
79         gre = request.form.get('gre')
80         rating = request.form.get('rating')
81         researchs = request.form.get('researchs')
82         model = request.form.get('models')
83         if toefl == '':
84             msg = 'enter the TOEFL marks'
85             return render_template('prediction.html',msg = msg)
86         elif sop == '':
87             msg = 'enter the SOP marks'
88             return render_template('prediction.html',msg = msg)
89         elif lor == '':
90             msg = 'enter the LOR marks'
91             return render_template('prediction.html',msg = msg)
92         elif gre == '':
93             msg = 'enter the GRE marks'
94             return render_template('prediction.html',msg = msg)
95         elif cgpa == '':
96             msg = 'enter the CGPA marks'
97             return render_template('prediction.html',msg = msg)
98         elif researchs == 'Select any one':
99             msg = 'Please select whether you researched about your admission'
100             return render_template('prediction.html',msg = msg)
101         elif model == 'Select any one':
102             msg = 'Please select whether you Naive Bayes Algorithm or Linear Regression Algorithm about your admission'
103             return render_template('prediction.html',msg = msg)
104         elif researchs == 'Research':
105             researchs = 1
106         elif researchs == 'No Research':
107             researchs = 0
108
109
110
111     if g.record == -1:
112
113         # input_lst = [gre,toefl,rating,sop,lor,cgpa,researchs]
114         new_input = {
115             'GRE Score':gre,
116             'TOEFL Score':toefl,
117             'University Rating':rating,
118             'SOP':sop,
119             'LOR': lor,
120             'CGPA':cgpa,
121             'Research':researchs
122         }
123         print(new_input)
124         def predict_input(input):
125             input_df = pd.DataFrame([input])
126             print(input_df)
127             input_df[input_df.columns] = scaler.transform(input_df[input_df.columns])
128             print(input_df)
129             if model == 'linear':
130                 pred = linear_model.predict(input_df)
131             else:
132                 pred = university.predict(input_df)
133             return pred
134
135         prediction = predict_input(new_input)
136         if model == 'naivebayes':
137             if prediction == 1:
138                 send_email(g,email)
139                 return render_template('success.html')
140             elif prediction == 0:
141                 fail_mail(g,email)
142                 return render_template('fail.html')
143
144         elif model == 'linear':
145             predict = prediction * 100
146             linear_mail(g,email,predict)
147
148         return render_template('lineat_output.html',prediction= f'({predict})%')
149
150     elif g.record == 0:
151         # input_lst = [gre,toefl,rating,sop,lor,cgpa,researchs]
152         new_input = {
153             'GRE Score':gre,
154             'TOEFL Score':toefl,
155             'University Rating':rating,
156             'SOP':sop,
157             'LOR': lor,
158             'CGPA':cgpa,
159             'Research':researchs
160         }
161         print(new_input)
162         def predict_input(input):
163             input_df = pd.DataFrame([input])
164             print(input_df)
165             input_df[input_df.columns] = scaler.transform(input_df[input_df.columns])
166             print(input_df)
167             if model == 'linear':
168                 pred = linear_model.predict(input_df)
169             else:
170                 pred = university.predict(input_df)
171             return pred
172
173         prediction = predict_input(new_input)
174         if model == 'naivebayes':
175             if prediction == 1:
176                 return render_template('success.html')
177             elif prediction == 0:
178                 return render_template('fail.html')
179
180         elif model == 'linear':
181             predict = prediction * 100
182             return render_template('lineat_output.html',prediction= f'({predict})%')
183
184     if g.record == 1:

```

```

181         predict = prediction *100
182         return render_template('lineat_output.html',prediction= f'{predict}%')
183
184     if g.record == 1:
185         return render_template('prediction.html',username=session['username'],logout = 'logout')
186     elif g.record ==0:
187         return render_template('prediction.html')
188 #register page
189 @app.route('/register',methods = ['GET','POST'])
190 def register():
191     if request.method == 'POST':
192         username = request.form['username']
193         email = request.form['mail']
194         password = request.form['password']
195         confirm_password = request.form['confirm-password']
196         cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
197         cursor.execute('SELECT * FROM details WHERE username= %s',(username,))
198         account = cursor.fetchone()
199
200         if account:
201             msg = f'{username} already exist please enter an another username'
202             return render_template('sign-up.html',msg = msg)
203         elif not re.match(r'[^@]+\@[^@]+\.[^@]+', email):
204             msg = 'Enter the valid email id'
205             return render_template('sign-up.html',msg = msg)
206         elif not re.match( "^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*#?&])[A-Za-z\d@$!%*#?&]{6,20}$", password):
207             msg = 'password must be at least 8 character and on special character and one capital letter'
208             return render_template('sign-up.html',msg = msg)
209         elif password !=confirm_password:
210             msg = 'Password and confirm_password must be equal'
211             return render_template('sign-up.html',msg = msg)
212         elif not username or not password or not email:
213             msg = 'Please fill out the form!'
214             return render_template('sign-up.html',msg = msg)
215         else:
216             cursor.execute('Create table if not exists details(username varchar(150),email varchar(150),password varchar(150))')
217             cursor.execute('insert into details value(%s,%s,%s)',(username,email,password))
218
219     username = request.form['username']
220     email = request.form['mail']
221     password = request.form['password']
222     confirm_password = request.form['confirm-password']
223     cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
224     cursor.execute('SELECT * FROM details WHERE username= %s',(username,))
225     account = cursor.fetchone()
226
227     if account:
228         msg = f'{username} already exist please enter an another username'
229         return render_template('sign-up.html',msg = msg)
230     elif not re.match(r'[^@]+\@[^@]+\.[^@]+', email):
231         msg = 'Enter the valid email id'
232         return render_template('sign-up.html',msg = msg)
233     elif not re.match( "^(?=.*[a-z])(?=.*[A-Z])(?=.*\d)(?=.*[@$!%*#?&])[A-Za-z\d@$!%*#?&]{6,20}$", password):
234         msg = 'password must be at least 8 character and on special character and one capital letter'
235         return render_template('sign-up.html',msg = msg)
236     elif password !=confirm_password:
237         msg = 'Password and confirm_password must be equal'
238         return render_template('sign-up.html',msg = msg)
239     elif not username or not password or not email:
240         msg = 'Please fill out the form!'
241         return render_template('sign-up.html',msg = msg)
242     else:
243         cursor.execute('Create table if not exists details(username varchar(150),email varchar(150),password varchar(150))')
244         cursor.execute('insert into details value(%s,%s,%s)',(username,email,password))
245         mysql.connection.commit()
246         return render_template('sign-in.html')
247     return render_template('sign-up.html')
248
249 if __name__ == '__main__':
250     app.run(debug=True)

```

7.2 FEATURE 2

send_mail.py

```

1  |
2  | import smtplib, ssl
3  | ## email.mime subclasses
4  | from email.mime.multipart import MIMEMultipart
5  | from email.mime.text import MIMEText
6  | # The pandas library is only for generating the current date, which is not necessary for sending emails
7  | import pandas as pd
8  |
9  | # Define the HTML document
10 | def send_email(email):
11 |     html = '''
12 |         <html style='color:white; text-align:center;background-color:green;padding:8px 18px;border-radius:3px'>You are Eligible to Apply </html>
13 |         <img src='https://www.pngmart.com/files/1/Blushing-Emoji-PNG-File.png' style='margin-left:20%' width='150px' height = '150px'>
14 |         <p style='text-align:center;'>Thank you for visiting our website. Hope you had a great experience</p>
15 |         ...
16 |
17 |     # Set up the email addresses and password. Please replace below with your email address and password
18 |     email_from = 'gcesuap@gmail.com'
19 |     password = 'terc awcj webf raiv'
20 |     email_to = email
21 |
22 |     # Generate today's date to be included in the email Subject
23 |     date_str = pd.Timestamp.today().strftime('%Y-%m-%d')
24 |
25 |     # Create a MIMEMultipart class, and set up the from, to, Subject fields
26 |     email_message = MIMEMultipart()
27 |     email_message['from'] = email_from
28 |     email_message['to'] = email_to
29 |     email_message['Subject'] = f'Prediction Report Email- {date_str}'
30 |
31 |     # Attach the html doc defined earlier, as a MIMEText html content type to the MIME message
32 |     email_message.attach(MIMEText(html, "html"))
33 |     # Convert it as a string
34 |     email_string = email_message.as_string()
35 |
36 |     # Connect to the Gmail SMTP server and Send Email
37 |     context = ssl.create_default_context()
38 |     with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server:
39 |         server.login(email_from, password)
40 |         server.sendmail(email_from, email_to, email_string)
41 |
42 | def fail_email(email):
43 |     html = '''
44 |         <html style='color:white; text-align:center;background-color:green;padding:8px 18px;border-radius:3px'>You are not Eligible to Apply </html>
45 |         <img src='https://i.pinimg.com/564x/6b/91/e5/6b91e5e2f77ca9133f7a25fba2d5b23.jpg' style='margin-left:20%' width='150px' height = '150px'>
46 |         <p style='text-align:center;'>Thank you for visiting our website. Hope you had a great experience</p>
47 |         ...
48 |
49 |     # Set up the email addresses and password. Please replace below with your email address and password
50 |     email_from = 'gcesuap@gmail.com'
51 |     password = 'terc awcj webf raiv'
52 |     email_to = email
53 |
54 |     # Generate today's date to be included in the email Subject
55 |     date_str = pd.Timestamp.today().strftime('%Y-%m-%d')
56 |
57 |     # Create a MIMEMultipart class, and set up the from, to, Subject fields
58 |     email_message = MIMEMultipart()
59 |     email_message['from'] = email_from
60 |     email_message['to'] = email_to
61 |     email_message['Subject'] = f'Prediction Report Email- {date_str}'
62 |
63 |     # Attach the html doc defined earlier, as a MIMEText html content type to the MIME message
64 |     email_message.attach(MIMEText(html, "html"))
65 |     # Convert it as a string
66 |     email_string = email_message.as_string()
67 |
68 |     # Connect to the Gmail SMTP server and Send Email
69 |     context = ssl.create_default_context()
70 |     with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server:
71 |         server.login(email_from, password)
72 |         server.sendmail(email_from, email_to, email_string)
73 |
74 | def linear_email(email, predict):
75 |     html = f'''
76 |
77 |         <img src='https://www.tenforce.com/wp-content/uploads/2020/08/undraw_approve_gwp7-el603987686875.png' style='margin-left:20%' width='150px' height = '150px'>
78 |         <html style='color:white; text-align:center;background-color:green;padding:8px 18px;border-radius:3px'>Your chance of eligibility is : </html>
79 |         <h3 style='text-align:center;font-size:24px;color:green;padding:10px'>{predict}</h3>
80 |         <p style='text-align:center;'>Thank you for visiting our website. Hope you had a great experience</p>
81 |         ...
82 |
83 |     # Set up the email addresses and password. Please replace below with your email address and password
84 |     email_from = 'gcesuap@gmail.com'
85 |     password = 'terc awcj webf raiv'
86 |     email_to = email
87 |
88 |     # Generate today's date to be included in the email Subject
89 |     date_str = pd.Timestamp.today().strftime('%Y-%m-%d')
90 |
91 |     # Create a MIMEMultipart class, and set up the from, to, Subject fields
92 |     email_message = MIMEMultipart()
93 |     email_message['from'] = email_from
94 |     email_message['to'] = email_to
95 |     email_message['Subject'] = f'Prediction Report Email- {date_str}'
96 |     # Attach the html doc defined earlier, as a MIMEText html content
97 |     # type to the MIME message
98 |     email_message.attach(MIMEText(html, "html"))
99 |     # Convert it as a string
100 |    email_string = email_message.as_string()
101 |
102 |    # Connect to the Gmail SMTP server and Send Email
103 |    context = ssl.create_default_context()
104 |    with smtplib.SMTP_SSL("smtp.gmail.com", 465, context=context) as server:
105 |        server.login(email_from, password)
106 |        server.sendmail(email_from, email_to, email_string)
107 |
108 |

```

7.3 DATABASE SCHEMA

Server: 127.0.0.1 » Database: uaep » Table: details

Table structure | Relation view

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 username	varchar(150)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	2 email	varchar(150)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/>	3 password	varchar(150)	utf8mb4_general_ci		No	None			Change Drop More

☐ Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext

Print Propose table structure Move columns Normalize

Add 1 column(s) after password Go

Indexes

No index defined!

Create an index on 1 columns Go

Partitions

CHAPTER 8

TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments
LoginPage_TC_OO1	UI	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button		1.Enter URL and click go 2.Verify login/Signup popup displayed or not 3.If you want view your score 4.Click to see Eligible button	http://127.0.0.1:5000/	Application should show below UI elements: a.navbar which contain sign-in button and about link button b.Click to You are Eligible Button c.About Our project d.Our team member details e.Footer	Working as expected	Pass	
LoginPage_TC_OO2	UI	Home Page	Click the button to view user eligible percentage		1.Enter URL and click go 2.Click on 'Click to find you are Eligible' 3.If User click it will return to predict page: 4.Without Login	http://127.0.0.1:5000/predict	Application should show below UI elements: a.GRE test box b.Toefl test box c.university rating text box d.Sop text box e.Lor text Box d.Cgpa Text box d.Research Drop down box	Working as expected	Pass	
LoginPage_TC_OO3	Functional	Login page	Verify user is able to log into application with Valid credentials		1.Enter URL(https://shopnizer.com/) and click go 2.Click on Sign-in option 3.Enter Valid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: vgnesh157@gmail.com password: Vikki@12345	User should navigate to user account Predict page	Working as expected	Pass	
LoginPage_TC_OO4	Functional	Register page	Enter a username ,email, password,confirm password		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.I you are not the user click the Sign up button below 5.Regisiter Page will return 6.Enter all the credential And Click register		Application should return Register page a. Username Text Box b. email Text Box c. Password text box d. Confirm Password Text Box	Working as expected	Pass	
LoginPage_TC_OO4	Functional	Register page	Verify the email format is correct		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.I you are not the user click the Sign up button below 5.Regisiter Page will return 6.Enter all the credential And Click register	Email-vgnesh157@gmail.com	Application should show 'Enter a valid Email Address'	Working as expected	Pass	
LoginPage_TC_OO5	Functional	Register page	verify the password and the confirm password are match		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.I you are not the user click the Sign up button below 5.Regisiter Page will return 6.Enter all the credential And Click register	password:Vikki@12345 confirm password:vikki@12345	Application should show 'password and confirm password is not match'	Working as expected	Pass	
	Functional	sign in page	Verify username and password are present in the database		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.Enter username and password	username:Vignesh confirm_password:vikki@12346	Application should show 'Username and password is mis match'	Working as expected	Pass	
	Functional	Predict page	Verify username and password are present in the database		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.Enter username and password 5.click the sign_in button 6.Return Predict Page	username:Vignesh confirm_password:Vikki@12345	Application should show below UI elements: a.GRE test box b.Toefl test box c.university rating text box d.Sop text box e.Lor text Box d.Cgpa Text box d.Research Drop down box	Working as expected	Pass	
	UI & Functional	Success Page And send success email	Check whether the result is success		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.Enter username and password 5.click the sign_in button 6.Click the predict button Predict Page	GRE marks,toefl marks, university rating, sop mark ,lor marks,cgpa,Research	Application should return the success page,send succes email to respected email the enter by the user	Working as expected	Pass	
	UI & Functional	Fail Page And send success email	Check whether the result is Fail		1.Enter URL and click go 2.Click on sign in option 3.It will go to the login page 4.Enter username and password 5.click the sign_in button 6.Click the predict button Predict Page	GRE marks,toefl marks, university rating, sop mark ,lor marks,cgpa,Research	Application should return the Fail page,send Fail email to respected email the enter by the user	Working as expected	Pass	

8.2 USER ACCEPTANCE TESTING

1. Purpose of Document

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

2. Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	5	3	2	20
Duplicate	2	0	1	0	3
External	3	3	0	0	6
Fixed	12	6	3	2	23
Not Reproduced	3	2	0	0	5
Skipped	0	0	1	0	1
Won't Fix	0	0	0	0	0
Totals	30	16	8	4	58

3. 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	5	0	0	5
View Prediction Page	5	0	0	5
Enter the Scores	15	0	0	15
View Success Results	8	0	0	8
View Failure Results	7	0	0	7
Version Control	3	0	0	3

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS

UAEP

About Sign-In

University Admit Eligibility Predictor

We help you to find that you are eligible for higher education

Click to find you are Eligible

U A E P

Home

Enter your GRE(Out of 340)

Enter your TOEFL(Out of 120)

Enter your university rating

Enter your SOP(1-5)

Enter your LOR(1-5)

Enter your CGPA(Out of 10)

Select any one

Select any one

Predict



Predicting the chance of admission:




Congratulations. You have a chance.

 Copyright © All right reserved



Predicting the chance of admission:

Sorry. You don't have a chance.

   Copyright © All right reserved

CHAPTER 10

ADVANTAGES & 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 existing system.
- It is fast, efficient and reliable.
- Avoids data redundancy and inconsistency.
- 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.

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.

CHAPTER 11

CONCLUSION

Student admission problem is essential 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 their favourite universities.

CHAPTER 12

FUTURE SCOPE

The future scope of this project is enormous. 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 as a 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 scores 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.

CHAPTER 13

APPENDIX

13.1 SOURCE CODE

Index

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
8   <meta http-equiv="X-UA-Compatible" content="IE=edge">
9   <meta name="viewport" content="width=device-width, initial-scale=1.0">
10  <title>UAEP</title>
11  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.carousel.min.css">
12  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.theme.min.css">
13  <script type="text/javascript" src="https://code.jquery.com/jquery-1.12.0.min.js"></script>
14  <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.carousel.min.js"></script>
15  <link rel="stylesheet" href="/static/css/home-style.css">
16  <script src="/static/js/owl-carousel.js"></script>
17  <script src="/static/js/script.js"></script>
18 </head>
19 <body>
20   <div class="navbar">
21     <div class="nav-container">
22       <h3 class="nav-icon">UAEP</h3>
23       <div class="nav-content">
24         <ul>
25           <li><a href="#about-us">About</a></li>
26           <li><a href="/loginpage">Sign-In</a></li>
27         </ul>
28       </div>
29     </div>
30   </div>
31   <div class="home">
32     <div class="home-container">
33       
34       <h3 class="title-home">
35         University Admit Eligibility Predictor
36       </h3>
37       <p class="home-content">
38         We help you to find that you are eligible for higher education
39       </p>
40       <a href="/predict">
41         Click to find you are Eligible
42       </a>
43     </div>
44   </div>
45   <div class="about-us" id = 'about-us'>
46     <div class="container">
47       <h3>
48         What we Do
49       </h3>
50       <p class="about-us-content">
51         UAEP is a new web-based University Admissions Predictor.It is an AI based application that asks for the users to input their academic transcripts data and
52       </p>
53     </div>
54   </div>
55   <div class="about" id="about">
56     <div class="container">
57       <div class="row">
58         <div class="col-md-12">
59           <div id="testimonial-slider" class="owl-carousel">
60             <div class="testimonial">
61               <div class="pic">
62                 
63               </div>
64               <h3 class="title">Team Leader</h3>
65               <p class="description">Aspiring data analyst with a strong math and statistics background; proficient in documenting, interpreting, and analyzing
66               </p>
67               <div class="testimonial-content">
68                 <div class="testimonial-profile">
69                   <h3 class="name">Ajith N</h3>
70                   <span class="post">ML Engineer</span>
71                 </div>
72               </div>
73             </div>
74           </div>
75           <div class="testimonial">
76             <div class="pic">
77               
78             </div>
79             <h3 class="title">Team Members-1</h3>
80             <p class="description">Positive driven and confident web developer with an apt for learning new technology.Exceptionally creative and talented
81             </p>
82             <div class="testimonial-content">
```

```

82         <div class="testimonial-profile">
83             <h3 class="name">Madhan Prakash T</h3>
84             <span class="post">Web Designer</span>
85         </div>
86     </div>
87 </div>
88 <div class="testimonial">
89     <div class="pic">
90         
91     </div>
92     <h3 class="title">Team Member-2</h3>
93     <p class="description">Computer Engineer with solid coding background and exceptional ability to apply cutting edge UI frameworks (Angular, Re
94     <div class="testimonial-content">
95         <div class="testimonial-profile">
96             <h3 class="name">Vignesh S</h3>
97             <span class="post">Web Developer</span>
98         </div>
99     </div>
100 </div>
101 <div class="testimonial">
102     <div class="pic">
103         
104     </div>
105     <h3 class="title">Team Member-3</h3>
106     <p class="description">Creative data analyst with exceptional information gathering skills looking for a challenging opportunity to incorporat
107     <div class="testimonial-content">
108         <div class="testimonial-profile">
109             <h3 class="name">Baghatraj N</h3>
110             <span class="post">Data Analyst</span>
111         </div>
112     </div>
113 </div>
114 </div>
115 </div>
116 </div>
117 </div>
118 </div>
119 <section class="footer">
120     <p class="footer-content">
121         <a href="https://github.com/IBM-EPBL/IBM-Project-37700-1660318565"><i class="fa fa-github"></i></a>
122         <p>Copyright <i class="fa fa-copyright"></i> All right reserved</p>
123     </p>
124 </section>
125 </body>
126 </html>

```

```

88     <div class="testimonial">
89         <div class="pic">
90             
91         </div>
92         <h3 class="title">Team Member-2</h3>
93         <p class="description">Computer Engineer with solid coding background and exceptional ability to apply cutting edge UI frameworks (Angular, Re
94         <div class="testimonial-content">
95             <div class="testimonial-profile">
96                 <h3 class="name">Vignesh S</h3>
97                 <span class="post">Web Developer</span>
98             </div>
99         </div>
100     </div>
101     <div class="testimonial">
102         <div class="pic">
103             
104         </div>
105         <h3 class="title">Team Member-3</h3>
106         <p class="description">Creative data analyst with exceptional information gathering skills looking for a challenging opportunity to incorporat
107         <div class="testimonial-content">
108             <div class="testimonial-profile">
109                 <h3 class="name">Baghatraj N</h3>
110                 <span class="post">Data Analyst</span>
111             </div>
112         </div>
113     </div>
114 </div>
115 </div>
116 </div>
117 </div>
118 </div>
119 <section class="footer">
120     <p class="footer-content">
121         <a href="https://github.com/IBM-EPBL/IBM-Project-37700-1660318565"><i class="fa fa-github"></i></a>
122         <p>Copyright <i class="fa fa-copyright"></i> All right reserved</p>
123     </p>
124 </section>
125 </body>
126 </html>

```

Chance

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
8   <meta http-equiv="X-UA-Compatible" content="IE=edge">
9   <meta name="viewport" content="width=device-width, initial-scale=1.0">
10  <title>UAE</title>
11  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.carousel.min.css">
12  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.theme.min.css">
13  <script type="text/javascript" src="https://code.jquery.com/jquery-1.12.0.min.js"></script>
14  <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.carousel.min.js"></script>
15  <link rel="stylesheet" href="/static/css/home-style.css">
16  <script src="/static/js/owl-carousel.js"></script>
17  <script src="/static/js/script.js"></script>
18 </head>
19 <body>
20   <div class="navbar" style="background-color: black;>
21     <div class="nav-container">
22       <h3 class="nav-icon">UAE</h3>
23       <div class="nav-content">
24         <ul>
25           <li><a href="/">Home</a></li>
26           <li><a href="/logout">logout</a></li>
27         </ul>
28       </div>
29     </div>
30   </div>
31   <div class="success">
32     <div class="success-content">
33       
34     </div>
35     <div class="success-content">
36       <div class="msg">
37         <h3>Predicting the chance of admission:</h3>
38         <br>
39         <br>
40         <span style="font-weight: 600;">Congratulations, You have a chance.</span>
41       </div>
42     </div>
43   </div>
44 </div>
45 <div>
46   <section class="footer">
47     <p class="footer-content">
48       <a href="https://github.com/IBM-EPBL/IBM-Project-37700-166018555"><i class="fa fa-github"></i></a>
49       <p>Copyright <i class="fa fa-copyright"></i> All right reserved</p>
50     </p>
51   </section>
52 </div>
53 </body>
54 </html>
```

No Chance

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5
6   <meta name="viewport" content="width=device-width, initial-scale=1">
7   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
8   <meta http-equiv="X-UA-Compatible" content="IE=edge">
9   <meta name="viewport" content="width=device-width, initial-scale=1.0">
10  <title>UAE</title>
11  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.carousel.min.css">
12  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.theme.min.css">
13  <script type="text/javascript" src="https://code.jquery.com/jquery-1.12.0.min.js"></script>
14  <script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/owl-carousel/1.3.3/owl.carousel.min.js"></script>
15  <link rel="stylesheet" href="/static/css/home-style.css">
16  <script src="/static/js/owl-carousel.js"></script>
17  <script src="/static/js/script.js"></script>
18 </head>
19 <body>
20   <div class="navbar" style="background-color: black;>
21     <div class="nav-container">
22       <h3 class="nav-icon">UAE</h3>
23       <div class="nav-content">
24         <ul>
25           <li><a href="/">Home</a></li>
26           <li><a href="/logout">logout</a></li>
27         </ul>
28       </div>
29     </div>
30   </div>
31   <div class="success">
32     <div class="success-content">
33       
34     </div>
35     <div class="success-content">
36       <div class="msg">
37         <h3>Predicting the chance of admission:</h3>
38         <br>
39         <p>{{predict_value}}</p>
40         <br>
41         <span style="font-weight: 600;">Sorry, You don't have a chance.</span> </p>
42       </div>
43     </div>
44   </div>
45 <div>
46   <section class="footer">
47     <p class="footer-content">
48       <a href="http://"><i class="fa fa-github"></i></a>
49       <a href="http://"><i class="fa fa-twitter"></i></a>
50       <a href="http://"><i class="fa fa-facebook"></i></a>
51       <p>Copyright <i class="fa fa-copyright"></i> All right reserved</p>
52     </p>
53   </section>
54 </div>
55 </body>
56 </html>
```

13.2 GITHUB & PROJECT DEMO LINK

Github Link : <https://github.com/IBM-EPBL/IBM-Project-37700-1660318565>

Demo Link : https://drive.google.com/file/d/17l8JsX2zKgmRhXAEK8wmeQ3pquiPgCo-/view?usp=share_link