PROJECT REPORT

Date	17 November 2022
Team ID	PNT2022TMID37486
Team Leader	PRAVEEN KUMAR S (311819205025)
Team Members	MOHAMED IQRAMULLAH S (311819205010) MOHAMED MOOSA M (311819205011) VIGNESH S (311819205031)
Project Name	**UNIVERSITY ADMIT ELIGIBILITY PREDICTOR**

1. INTRODUCTION :

1.1– OVERVIEW :

- The project is implemented using a Machine-Learning model that predicts whether the user is eligible for an admission in the selected rated universities with provided details such as marks and others.

- The algorithm works in such a way that when the user provides the details such as (GRE Score, TOEFL Score, University Rating, SOP, LOR, CGPA, Research) the percentage of chance of admit is displayed.

- The user is provided with a UI (Web based application) in which the user can enter the details mentioned above for prediction.

- The main advantage of this is that the user can avoid long process of having to check the eligibility of a university admission by himself and make use of this application to predict the eligibility / chance of admit.

1.1 – PURPOSE :

- University education is becoming a critical pillar of social and economic life in the twenty-first century. It is crucial not only in the educational process but also in assuring two vital things: a great job and financial stability. Predicting university entrance, on the other hand, might be extremely challenging because pupils are unaware of the admission standards.

- 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 purpose of this project is to make the prediction of eligibility of an admission to a rated university with ease using a UI with the provided user details (GRE Score, TOEFL Score, University Rating, SOP, LOR, CGPA, Research). This also eliminates the possibility of human errors.

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, LOR and under graduate score.

- Bayesian Networks Algorithm have been used to create a decision support network for evaluating the application submitted by foreign students of the university. 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 – PROPOSED SOLUTION :

- These problems can be resolved by using regression algorithms / classification algorithms as they can consider most of the features for prediction. Linear regression / KNN classification / Random forest Regressor can be used as the machine learning model for the model.

- XG boost model can also be used which performs better on small to medium scale datasets but the model giving accurate and desired results only will be selected. The aim of the proposed system is to
address the limitations of the current system. The requirements for the system have been gathered from the defects recorded in the past and also based on the feedback from users of previous metrics tools. Following are the objectives of the proposed system:

- Reach to geographically scattered student.
- Reducing time in activities.
- Paperless admission with reduced man power.
- Operational efficiency.

2.3 - REFERENCES :

1. A University Admission Prediction System using Stacked Ensemble Learning by S. Sridhar, S. Mootha and S. Kolagat in the year 2020.

2. A Comparison of Regression Models for Prediction of Graduate Admissions, IEEE International Conference on Computational Intelligence in Data Science by S. Mohan Acharya in the year 2019.

3. Predicting student university admission using logistic regression by Sharan Kumar Paratala Rajagopan in the year 2020.

4. Engineering & Technology Admission Analysis and Prediction by Bhoite, Sachin & Ajit in the year 2020.

5. Student Admission Predictor by Himanshu Sonawane in the year 2017.

6. A Recommender System for Predicting Students Admission to a Graduate Program using Machine Learning Algorithms by Inssaf El Guabassi, Zakaria Bousalem, Rim Marah and Aimad Qazdar published in the year 2021.

7. Universities admissions predictor by Aanchal Thakur in the year 2020.

8. Multiple Machine Learning Classifiers for students admission into University by Anil B, Akram Pasha and Aditya Kumar Singh in the year 2019.

2.4 – PROBLEM STATEMENT DEFINITION :

1. The problem statement is to design a college prediction/prediction system and to provide a probabilistic insight into college administration for overall rating, cut-offs of the colleges, admission intake and preferences of students.

2. It has always been a troublesome process for students in finding the perfect university and course for their further studies.

3. At times they do know which stream they want to get into, but it is not easy for them to find colleges based on their academic marks and other performances.

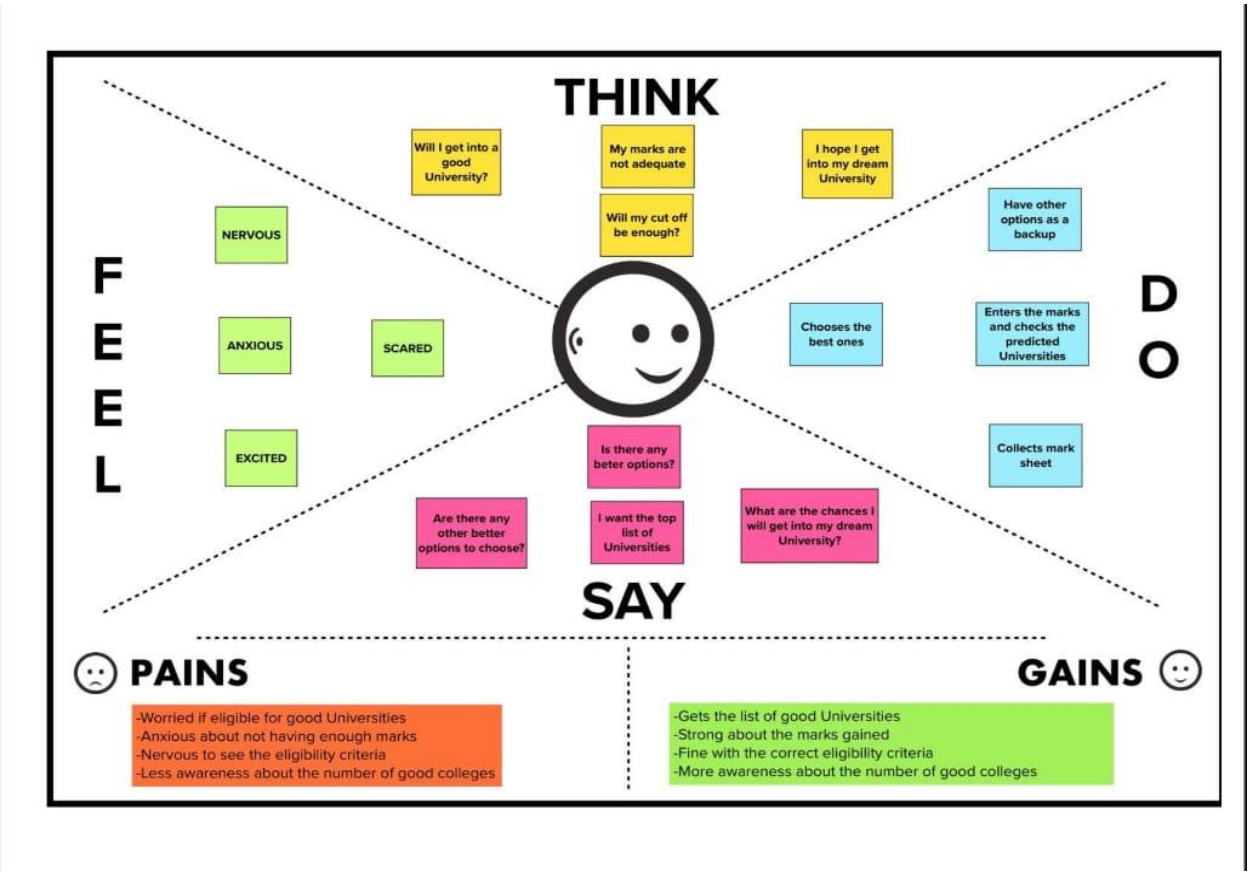
4. We aim to develop and provide a place which would give a probabilistic output of how likely it is to get into a university given their details.

3. IDEATION & PROPOSED SOLUTION :

3.1 – EMPATHY MAP CANVAS :

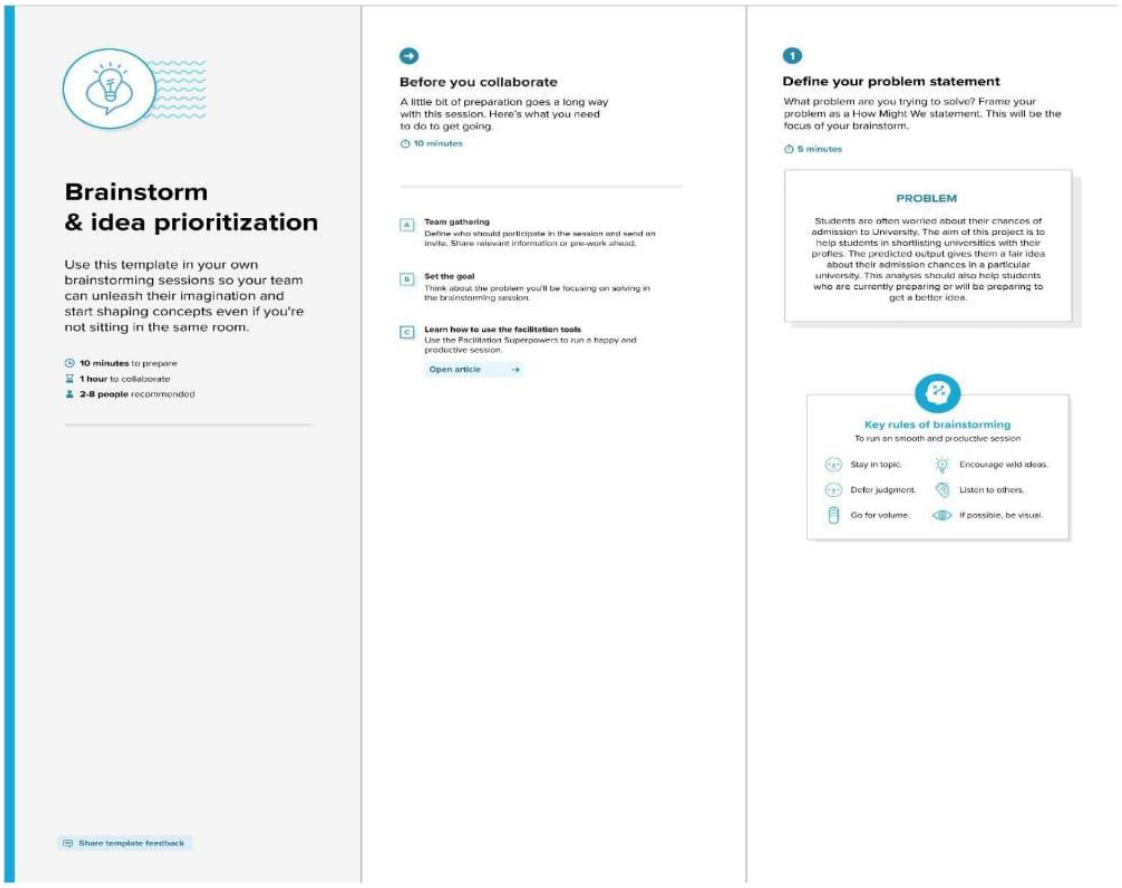
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

3.2 – EMPATHY MAP :

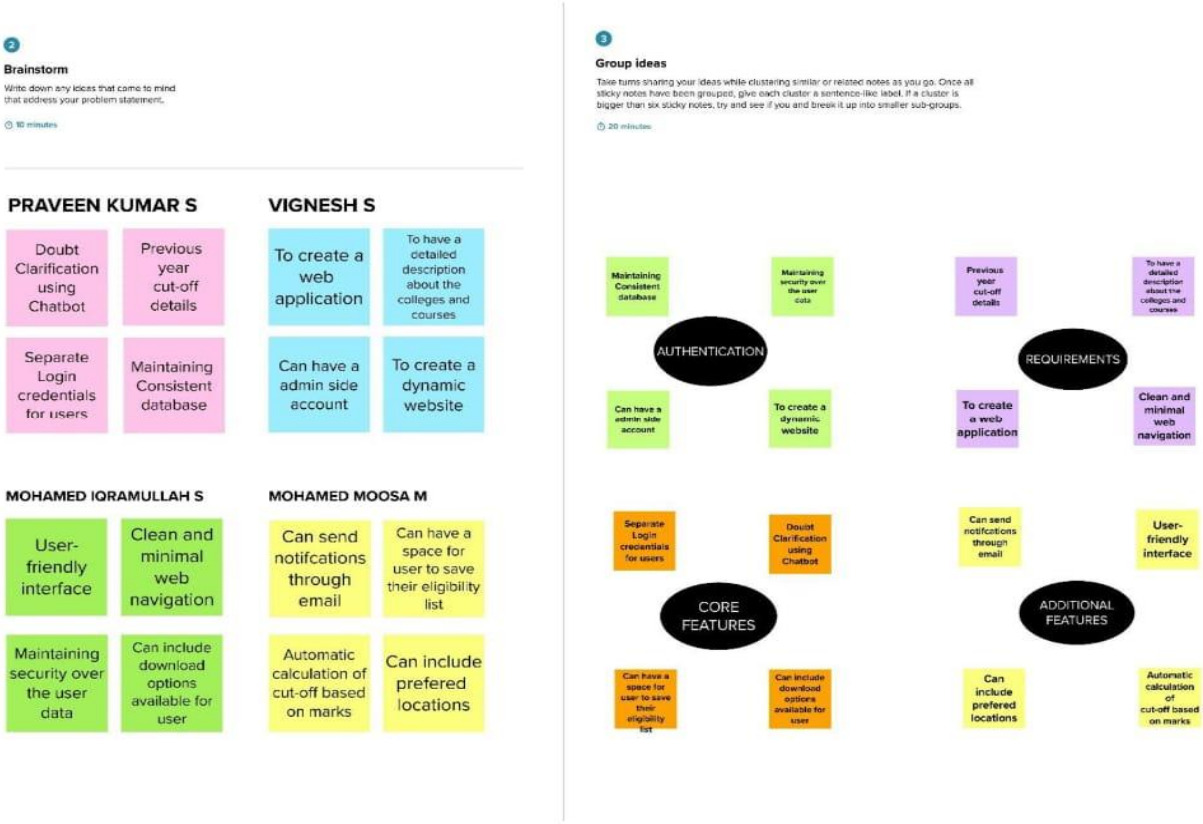


3.3 – IDEATION & BRAINSTORMING :

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



Step-2: Brainstorm, Idea Listing and Grouping



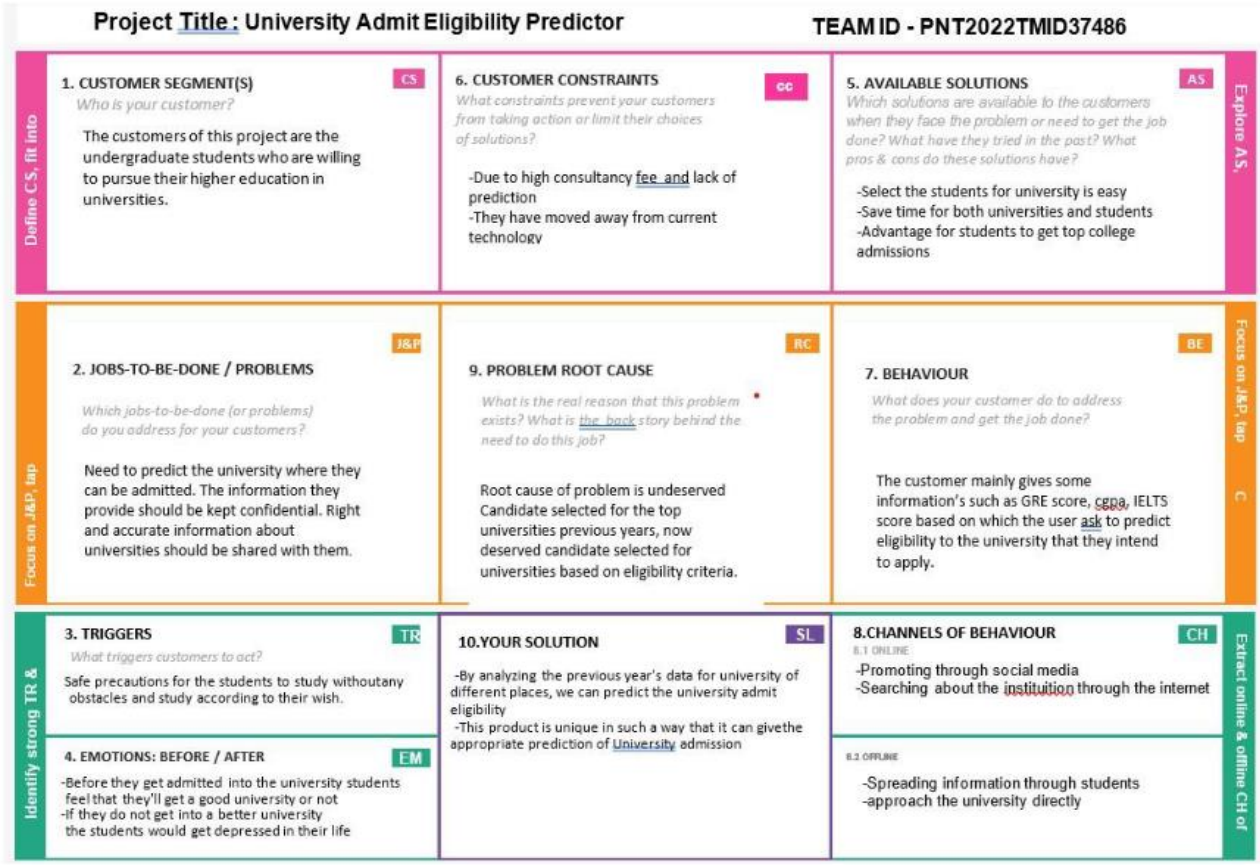
Step-3: Idea Prioritization

3.4 – PROPOSED SOLUTION :

S.NO.	PARAMETER	DESCRIPTION
1.	Problem statement (Problem to be solved)	• Students are often worried about their chancesof admission to university. • The aim of this project is to help students inshortlisting universities with their profiles. • The predicted output gives them a fair idea about their admission chances in a particularuniversity. • This analysis help students who are currentlypreparing or will be preparing to get a better idea.
2.	Idea / Solution description	• The aim of this project is to help students inshortlisting universities with their profiles. • The predicted output gives them a fair idea abouttheir admission chances in a particular university. • This analysis should also help students who are currently preparing or will be preparing to get abetter idea regarding the admission process. • A model is developed which analyses the dataprovided by the user and evaluates it in accordance with the algorithm developed to predict the eligibility of the user for the specified university.

3.	Novelty / Uniqueness	• We aim to design the model in such a way that it takes certain non-academic factors which influencethe admission process into consideration as well. • This further enhances the accuracy of the predictor.This attribute is not considered in most predictors available in the market.
4.	Social Impact / CustomerSatisfaction	• This predictor would provide a clarity to passed outstudents who might be confused regarding their future with respect to university admissions. • The students can apply to universities based ontheir eligibility chances.
5.	Business Model (Revenue Model)	• Such predictors have a huge demand in the market since students who complete their schooling are always in need of tools like this to plan out their university admissions.
6.	Scalability of the solution	• The scope of this predictor is very wide as a large number of universities could be brought within therange of this predictor depending on the requirements of the user. • Hence, this solution is largely scalable in nature.

3.5 – PROBLEM SOLUTION FIT :



4. REQUIREMENT ANALYSIS :

4.1 – FUNCTIONAL REQUIREMENTS :

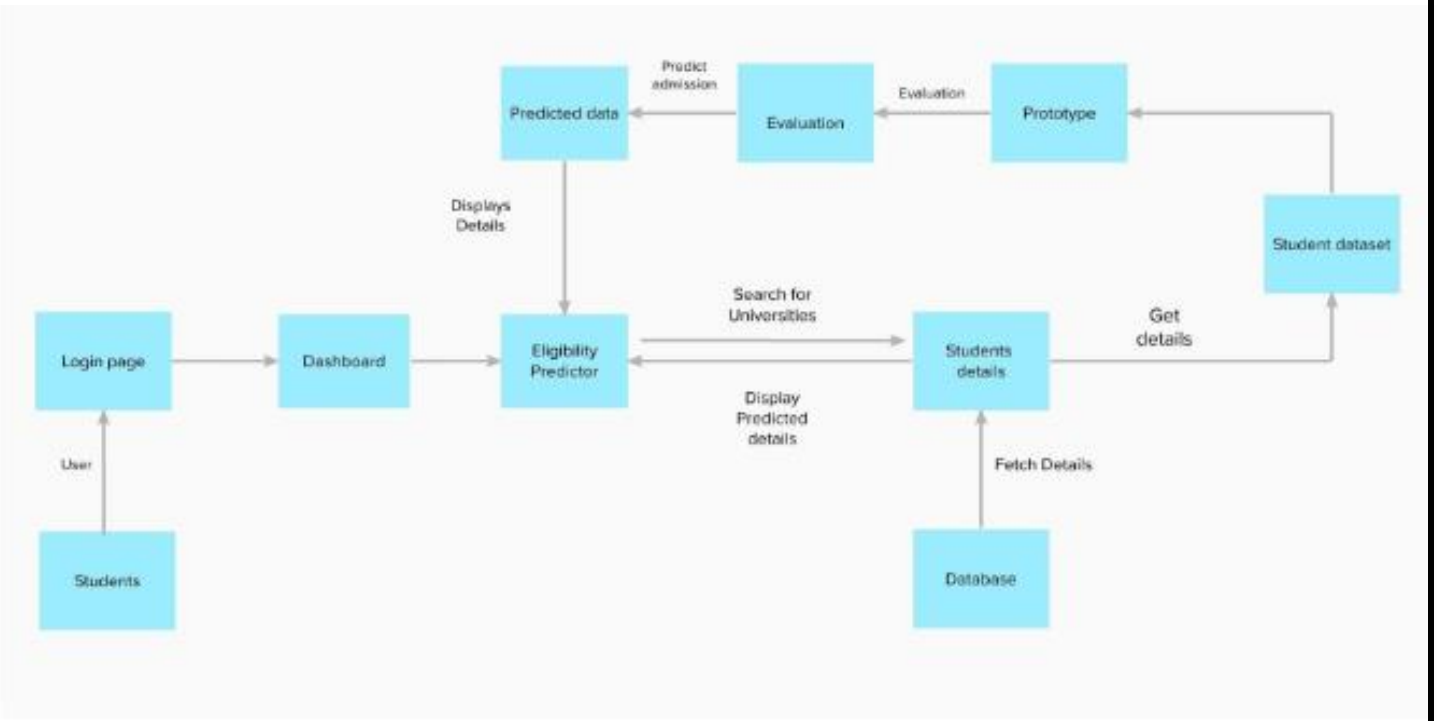
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	**User Registration**	• Registration through Form • Registration through Gmail • Registration through LinkedIN
FR-2	**User Confirmation**	• Confirmation via Email • Confirmation via OTP
FR-3	**User Requirements**	• All the needed files are been asked tofeed in the website. • Based on the uploads, the system would collect all the necessary information. • The information includes the list of allthe possible universities and streams.
FR-4	**User Details**	• Has to feed some documentsScore Sheets • Letter of Recommendation (LOR)Statement of Purpose (SOP) Curriculum Vitae (CV)

4.2 - NON-FUNCTIONAL REQUIREMENTS :

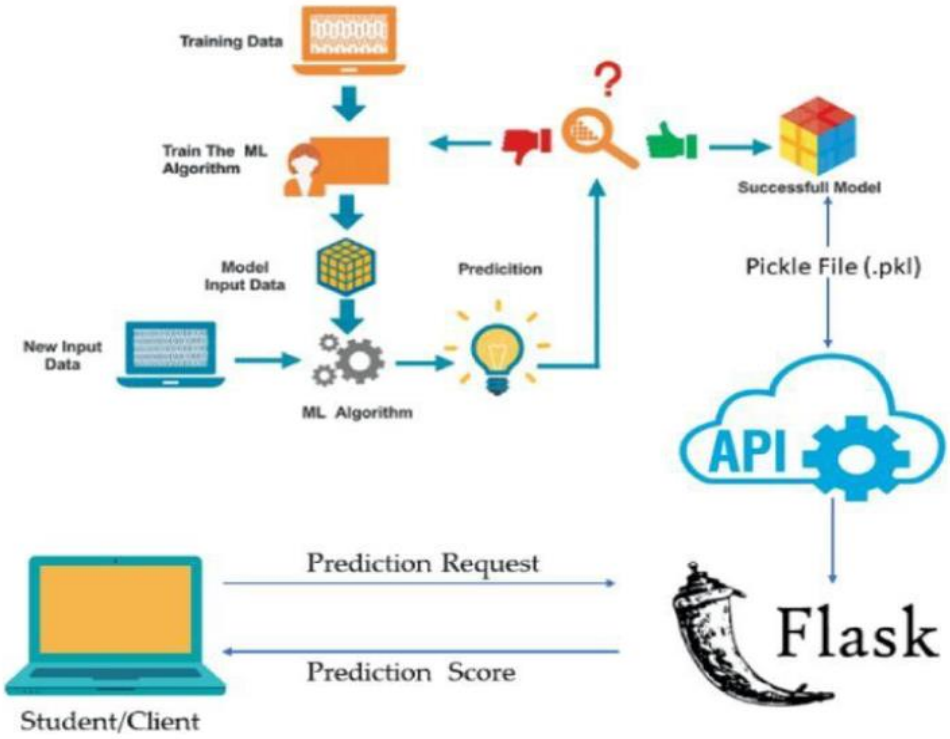
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	• Our website is very user friendly. • There is no need for any technical skill in order to access our website. • The page would not take a lot of time to load the content.
NFR-2	**Security**	• The user who is having the valid credentials can able to access our site. • Under any error, the system should be able to come back to regular operation in under an hour. • Use any cryptographic techniques. • Check data integrity for critical variables.
NFR- 3	**Reliability**	• The system would always strive for maximum reliability due to the importance of data and damages that could be cause by incomplete and incorrect data. • Data corruption is prevented by applying the possible backup procedures and techniques.
NFR- 4	**Performance**	• User can able to access in our website with internet connection. • Traffics can be handled effectively. • The database should be able to accommodate a minimum of 10,000 records of students.
NFR- 5	**Availability**	• Fast and efficient. • Students can access our website from any of the available browser. • Increase of the hardware and database failure a replacement page will be show and for database back should be retrieved from data folder.
NFR- 6	**Scalability**	• A sizable number of users be able to access the system at the same time. • It must therefore be able to manage numerous concurrent users. • The system must be scalable enough to support 10,00,000 visits at the same time while maintaining optimal performance.

5. PROJECT DESIGN :

5.1 – DATA FLOW DIAGRAMS :



5.2 – SOLUTION ARCHITECTURE :



5.3 – TECHNICAL ARCHITECTURE :

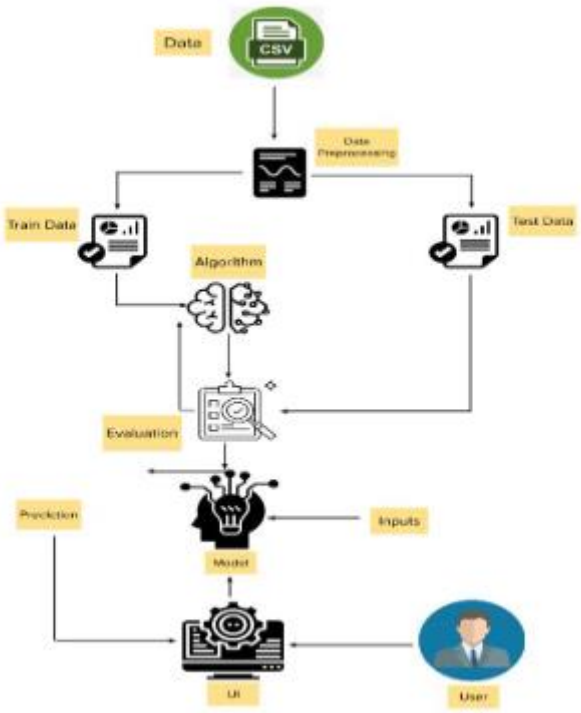


Table-1: COMPONENTS & TECHNOLOGIES:

S.NO	CHARACTERISTICS	DESCRIPTION	TECHNOLOGY
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript etc.
2.	Application Logic-1	Logic for a process in the application	Python (jupyter)
3.	Application Logic-2	Logic for a process in the application	IBM Watson Assistant
4.	Database	Data Type, configurations etc.	CSV
5.	External API	Purpose of External API Used in the application	List of eligible universities
6.	Machine Learning Model	Purpose of Machine Learning Model	KNN, Random Forest, Decision tree etc.
7.	Infrastructure for server or cloud	Application Deployment on Local system / cloud Local Server Configuration: Cloud server Configuration:	Local, cloud etc.

Table-2: APPLICATION CHARACTERISTICS:

S.NO	CHARACTERISTICS	DESCRIPTION	TECHNOLOGY
1.	Open-Source Frameworks	Flask is imported for front end purpose and python is used for backend purpose	Python (flask)
2.	Security Implementations	The user profile will be secured	Encryptions, IAM controls, OWSAP etc
3.	Scalable Architecture	The accurate list of eligible universities and its description will be provided	Random forest ML algorithm
4.	Availability	Anyone can visit our website anytime	IBM load balancer
5.	Performance	By applying through our website, the user can have knowledge according to their eligibility criteria.	Random forest ML algorithm

5.4 – 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 register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
Customer (Web user)	Registration	USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
Customer (Web user)	Registration	USN-3	As a user, I can register to the application through Gmail	I can access my account	Medium	Sprint-1
Customer (Web user)	Login	USN-4	As a user, I can log into the application by entering email & password	I can access my account	High	Sprint-1
Customer (Web user)	Update Profile	USN-5	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
Customer (Web user)	Choose University	USN-6	As a user, I will be able to view the list of Universities that the students are eligible to apply.	I can choose the University from the List of University provided in the drop down menu.	High	Sprint-3
Customer (Web user)	Choose Course	USN-7	As a user, I will be able to view the list of courses that the students are eligible to apply.	I can choose the course from the List of courses provided in the drop down menu.	Medium	Sprint-3
Customer (Web user)	Admission Process	USN-8	As a user,I will be able to view the details of Admission process like date and venue of certification verification.	I can view the details of Admission process being displayed at the end of prediction.	Low	Sprint-4
Administrator	Authentication	USN-9	As a admin , the login credential of the user is authenticated my me.	I can retrieve and make use of all the user details.	High	Sprint-1
Administrator	Update Profile	USN-10	As a admin,I can verify the user entered details.	I can confirm and access the user details.	High	Sprint-2
Administrator	Prediction	USN-11	As a admin,I can test the trained ML model by analysing the user details by ML algorithms like Logistic Regression.	I can test the user data with the trained ML model.	High	Sprint-3
Administrator	Output	USN-12	As a 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 AND SCHEDULING :

6.1 - SPRINT PLANNING & ESTIMATION :

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	PRAVEEN
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	IQRAM
Sprint-1	Login	USN-3	As a user, I can log into the application by entering email & password	2	High	MOOSA
Sprint-2	Dashboard	USN-4	As a user, I will provide my schooling information in the application	3	High	MOOSA VIGNESH
Sprint-2		USN-5	As a user I will view the colleges which are under my criteria.	2	High	PRAVEEN IQRAM
Sprint-3		USN-6	As an Administrator I provide the eligibility criteria of the college for the students.	5	Medium	PRAVEEN VIGNESH
Sprint-4		USN-7	As a user after checking my college details I can view the customer care phone number for addition information.	5	Medium	IQRAM MOOSA

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	23 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	30 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	06 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	13 Nov 2022	19 Nov 2022	20	19 Nov 2022

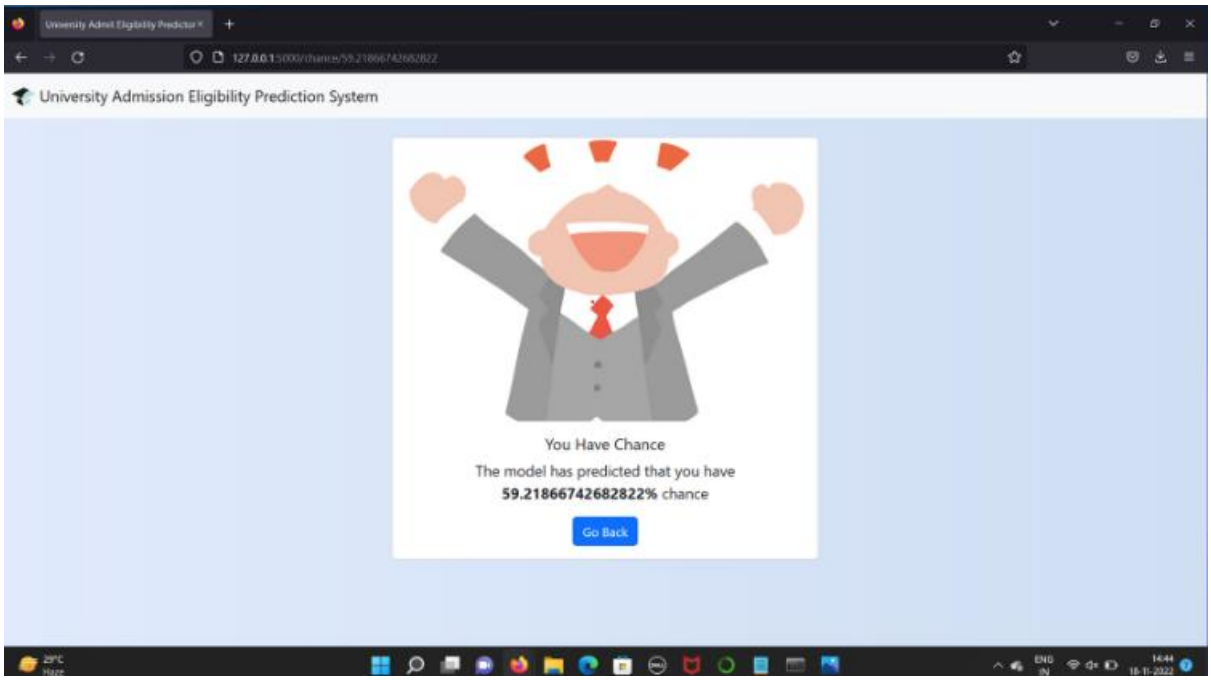
BURNDOWN CHART :

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

7. CODING AND SOLUTIONING :

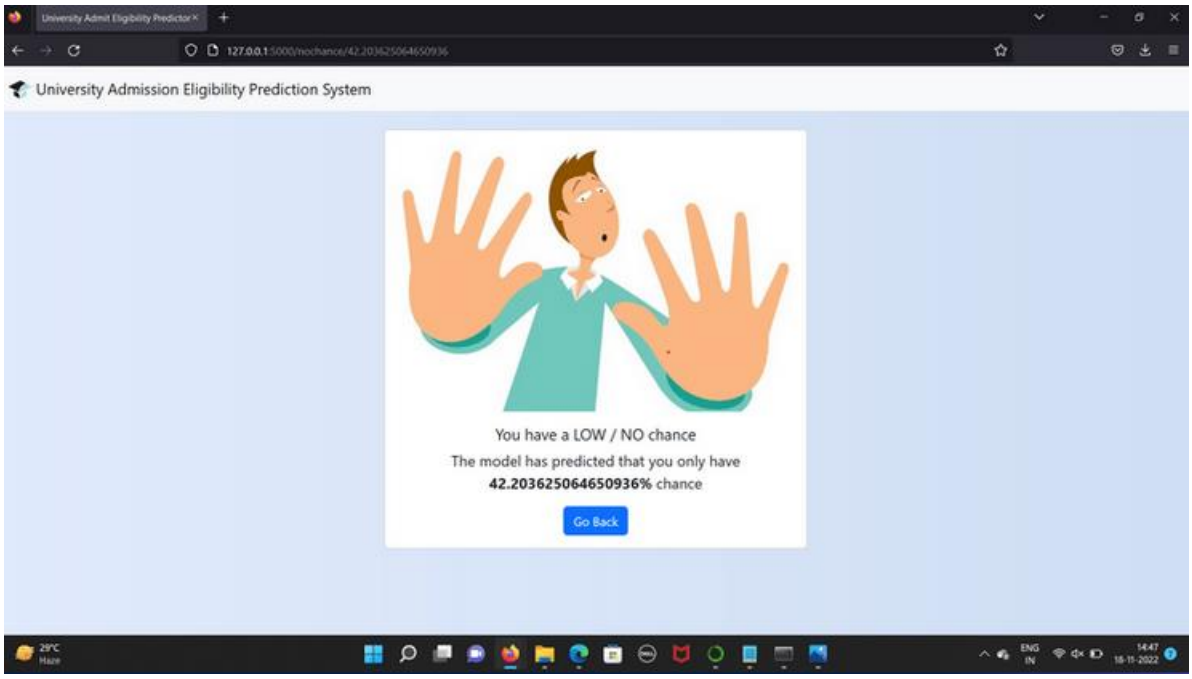
7.1 – FEATURE 1 :

This feature was included to demonstrate that pupils who have achieved success have greater opportunities.



7.2 – FEATURE 2 :

This is the feature to show the results as “No” or “Lower chance of Admission”.



7.3 – TECHNOLOGIES USED :

SOFTWARE :

- Python
- Anaconda Navigator
- Jupyter Notebook/Google collab
- Spyder
- IBM Watson Studio

FRAMEWORKS:

- Python-Flask
- Anaconda Navigator
- IBM Cloud
- Python Web Frameworks

8. TESTING :

8.1 – TEST CASES :

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

8.2 – USER ACCEPTANCE TESTING :

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	19
Duplicate	0	0	0	0	0
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	1	2
Won't Fix	0	0	0	0	0
Totals	24	14	13	26	64

9. RESULTS :

9.1 – PERFORMANCE METRICS :

In [25]:
```
y_predict=rgr.predict(X_test)
```

In [26]:
```
from sklearn.metrics import mean_squared_error, r2_score,mean_absolute_error
import numpy as np
print('Mean Absolute Error:', mean_absolute_error(y_test, y_predict))
print('Mean Squared Error:', mean_squared_error(y_test, y_predict))
print('Root Mean Squared Error:', np.sqrt(mean_squared_error(y_test, y_predict)))
```

```
Mean Absolute Error: 0.043484769325589964
Mean Squared Error: 0.003981572845017372
Root Mean Squared Error: 0.06309970558582165
```

In [28]:
```
from sklearn.linear_model._logistic import LogisticRegression

lore = LogisticRegression(random_state=0, max_iter=1000)

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

In [29]:
```
y_pred = lr.predict(X_test)
```

In [30]:
```
from sklearn.metrics import accuracy_score, recall_score, roc_auc_score, confusion_matrix

print('Accuracy Score:', accuracy_score(y_test, y_pred))
print('Recall Score:', recall_score(y_test, y_pred))
print('ROC AUC Score:', roc_auc_score(y_test, y_pred))
print('Confussion Matrix:\n', confusion_matrix(y_test, y_pred))
```

```
Accuracy Score: 0.9666666666666667
Recall Score: 1.0
ROC AUC Score: 0.75
Confussion Matrix:
 [[ 2  2]
 [ 0 56]]
```

10. ADVANTAGES AND DISADVANTAGES :

10.1– ADVANTAGES :

- Aids pupils in forecasting their admissions chances.
- Answers the question of which university is the best.
- Assists in locating the right university for each student.
- Simplifies the admissions procedure.
- Convenient platform.

10.2 – DISADVANTAGES :

- This digital approach can only be implemented if there are electronic equipment, such as a PC or laptop, with a strong internet connection.
- Results are solely based on the information provided by the students.
- If the students don't enter their marks correctly, inaccurate predictions will result.

11. APPLICATIONS :

• The main advantage of the project is the computerization of the entrance Seat allotment process.
• The total time for the entrance allotment became lesser and the allotment process became faster.
• Admissions are handled more easily and more simply.
• Automated method of estimating admissions probability.
• Eliminates the need to attend career guidance programmes or counselling
sessions to select the best university.
• An interactive and user-friendly tool for determining admission predictions.

12. CONCLUSION :

Millions of students apply to universities each year to start their academic careers. Most of them lack the necessary tools, background information, and caution, which leads to a host of issues including applying to the wrong university or college, which further wastes their time, money, and effort. We have made an effort to assist students who are having trouble selecting the best university for them with the aid of our initiative. Applying to colleges where one has a strong possibility of admission is crucial, as opposed to applying to schools where one may never be accepted.

13. FUTURE SCOPE :

This project's scope is a web application that lets users enter their academic information to acquire estimates of their chances of admission to thetier of universities of their choice. It also offers an analysis based on the data set employed that demonstrates how the various parameters affect the likelihood of admission.

14. APPENDIX :

SOURCE CODE :

Demo2.html

```
{% extends 'index.html' %}
{% block body %}
<div class="p-4">
<div class="row mb-3">
<div class="col-4">
<h2 class="text-responsive-h">
ENTER YOUR DETAILS AND CHECK YOUR CHANCES OF
ADMISSION
</h2>
<p class="text-responsive">

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 predicted 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 who will be preparing for exams.
</p>
<div class="d-flex justify-content-right">
<img src="../static/img/animated.gif" border="0" alt="..." />
</div>
```

```
</div>
<div class="col-8">
<div class="card p-2 ms-2 my-2">
<div class="card-body">
<h5 class="card-title pb-4">
Enter the details
</h5>
<form action="/" method="post" id="theForm">
<div class="row mb-3">
<label for="gre" class="col-lg-2 col-form-label">GRE
Score:</label>
<div class="col-lg-10">
<input type="number" class="form-control" id="gre"
name="gre" min="250" max="340" required>
</div>
</div>
<div class="row mb-3">
<label for="tofel" class="col-lg-2 col-form-label">TOFEL
Score:</label>
<div class="col-lg-10">
<input type="number" class="form-control" id="tofel"
name="tofel" min="50" max="120" required>
</div>
</div>
<div class="row mb-3">
<label for="university_rating" class="col-lg-2 col-form-
label">University Rating:</label>
<div class="col-lg-10">
<input type="number" class="form-control"
id="university_rating" step="0.01" name="university_rating" min="1"
max="5" required>
</div>
</div>
<div class="row mb-3">
<label for="sop" class="col-lg-2 col-form-
label">SOP:</label>
<div class="col-lg-10">
```

```
<input type="number" class="form-control" id="sop"
name="sop" step="0.01" min="1" max="5" required>
</div>
</div>
<div class="row mb-3">
<label for="lor" class="col-lg-2 col-form-
label">LOR:</label>
<div class="col-lg-10">
<input type="number" class="form-control" id="lor"
name="lor" step="0.01" min="1" max="5" required>
</div>
</div>
<div class="row mb-3">
<label for="cgpa" class="col-lg-2 col-form-
label">CGPA:</label>
<div class="col-lg-10">
<input type="number" class="form-control" id="cgpa"
name="cgpa" step="0.01" min="5" max="10" required>
</div>
</div>
<fieldset class="row mb-3">
<legend class="col-form-label col-sm-2 pt-
0">Research:</legend>
<div class="col-sm-10">
<div class="form-check">
<input class="form-check-input" type="radio"
name="yes_no_radio" id="gridRadios1" value="1">
<label class="form-check-label" for="yes_no_radio">
Yes
</label>
</div>
<div class="form-check">
<input class="form-check-input" type="radio"
name="yes_no_radio" id="gridRadios2" value="0" checked>
<label class="form-check-label" for="yes_no_radio">
No
</label>
```

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

Chance.html

```
{% extends 'index.html' %}
{% block body %}
<div class="container text-center p-4">
```

```
<div class="d-flex justify-content-center">
<div class="card" style="width: 34rem;">
<img src="..\static\img\chance.jpeg" class="card-img-top" alt="...">
<div class="card-body">
<h5 class="card-title">You Have Chance</h5>
```

```
<p class="card-text">The model has predicted that you have
<strong>{{content[0]}}%</strong> chance</p>
<a href="/home" class="btn btn-primary">Go Back</a>
</div>
</div>
</div>
</div>
{% endblock %}
```

noChance.html

```
{% extends 'index.html' %}
{% block body %}
<div class="container text-center p-4">
<div class="d-flex justify-content-center">
<div class="card" style="width: 34rem;">
<img src="..\static\img\noChance.jpeg" class="card-img-top" alt="...">
<div class="card-body">
<h5 class="card-title">You have a LOW / NO chance</h5>
<p class="card-text">The model has predicted that you only have
<strong>{{content[0]}}%</strong> chance</p>
<a href="/home" class="btn btn-primary">Go Back</a>
</div>
</div>
</div>
</div>
{% endblock %}
```

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
```

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1, user-scalable=no">
<link rel="stylesheet" type="text/css" rel="noopener" target="_blank"
href="../static/css/styles.css">
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min
.css"
rel="stylesheet" integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9B
v1W
TRi" crossorigin="anonymous">
<script type="text/javascript" src="../static/js/script.js" async></script>
<title>University Admit Eligibility Predictor</title>
</head>
<body>
<nav class="navbar navbar-expand-lg bg-light">
<div class="container-fluid">
<a class="navbar-brand text-responsive-h" href="/">
<img src="..\static\img\logo.png" alt="Logo" width="30" height="24"
class="d-inline-block align-text-top ">
University Admission Eligibility Prediction System
</a>
</div>
</nav>
{% block body %}
<h1> Index Page </h1>
{% endblock %}
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundl
e.min.j
s" integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V
8Qb
sw3" crossorigin="anonymous"></script></body></html>
```

app.py

```
from flask import Flask, render_template, redirect, url_for, request
import requests
app = Flask(__name__)
@app.route("/", methods=['POST', 'GET'])
def index():
if request.method == 'POST':
arr = []
for i in request.form:
val = request.form[i]
if val == '':
return redirect(url_for("demo2"))
arr.append(float(val))
# deepcode ignore HardcodedNonCryptoSecret: <please specify a
reason
of ignoring this>
API_KEY =
"wf8mge_OQdwVO8ao2kmWCtfxOfLWl8442SH44V85v2Ls"
token_response =
requests.post('https://iam.cloud.ibm.com/identity/token',
data={
30
"apikey": API_KEY,
"grant_type": 'urn:ibm:params:oauth:grant-type:apikey'
})
mltoken = token_response.json()["access_token"]
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' +
mltoken}
payload_scoring = {
"input_data": [{"fields": ['GRE Score',
'TOEFL Score',
'University Rating',
'SOP',
'LOR ',
'CGPA',
'Research'],
"values": [arr]
```

```
}]
}
response_scoring = requests.post(
'https://us-south.ml.cloud.ibm.com/ml/v4/deployments/8308fd4c-24a5-46ab-96fa-263657ae4ad0/predictions?version=2022-10-18',
json=payload_scoring,
headers=header
).json()
result = response_scoring['predictions'][0]['values']
if result[0][0] > 0.5:
return redirect(url_for('chance', percent=result[0][0] * 100))
else:
return redirect(url_for('no_chance', percent=result[0][0] * 100))
else:
return redirect(url_for("demo2"))
@app.route("/home")
def demo2():
return render_template("demo2.html")
@app.route("/chance/<percent>")
def chance(percent):
return render_template("chance.html", content=[percent])
@app.route("/nochance/<percent>")
def no_chance(percent):
return render_template("noChance.html", content=[percent])
@app.route('/<path:path>')
def catch_all():
return redirect(url_for("demo2"))
if __name__ == "__main__":
app.run()
```

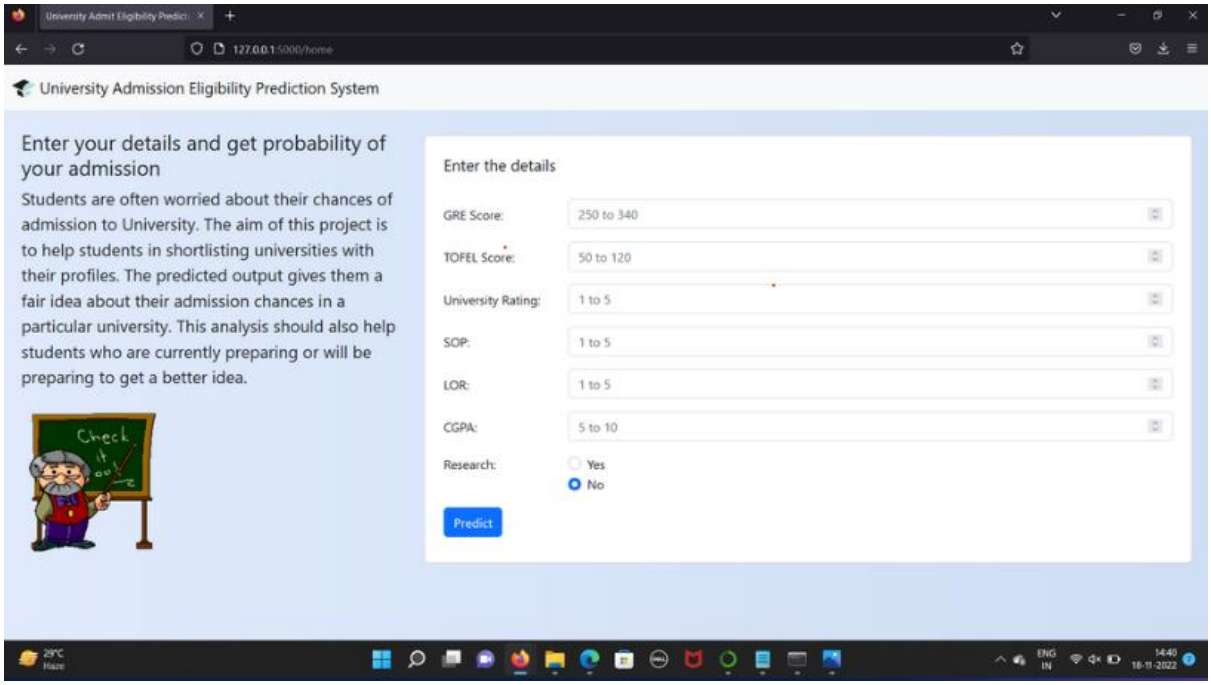
styles.css

```
* {
margin: 0;
padding: 0;
border: 0;
}
```

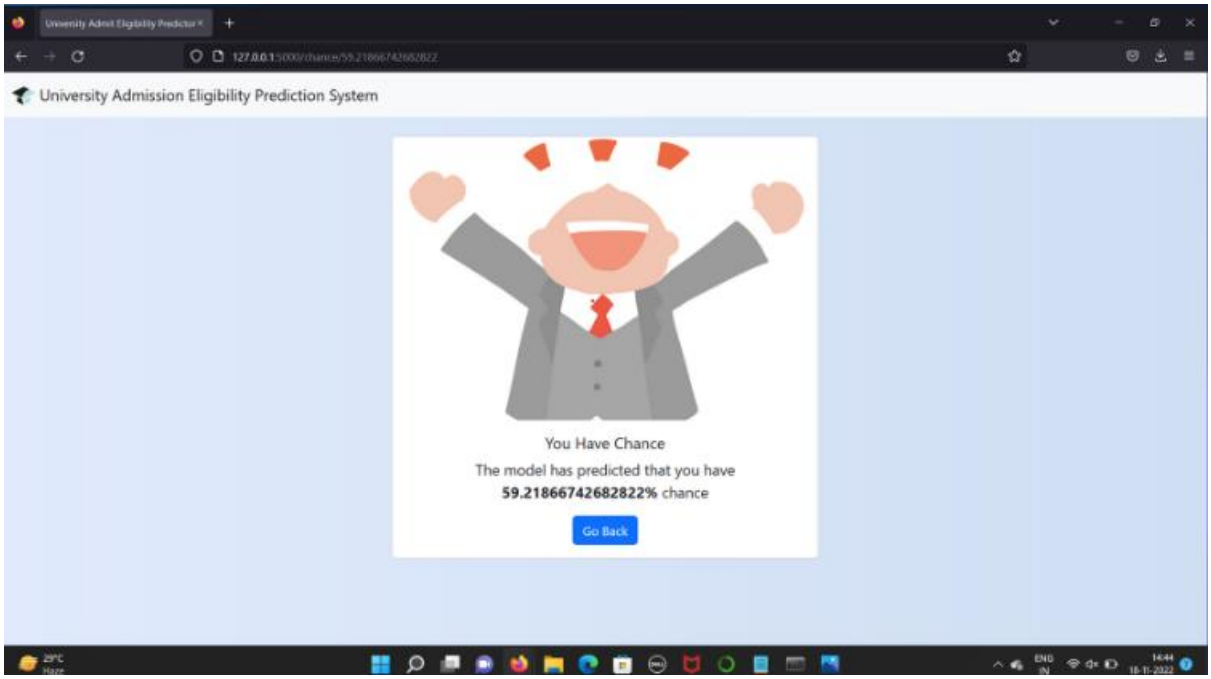
```
body {
font: 62.5%/1.5 "Lucida Grande", "Lucida Sans", Tahoma, Verdana,
sans-
serif;
background: #d2b48c;
background: -webkit-linear-gradient(to right, #d2b48c, #cfdef3);
background: linear-gradient(to right, #e0eafc, #d2b48c);
color: #000000;
text-align:center;
}
h1 {
font-size: 2.2em;
}
h2 {
font-size: 2.0em;
}
h4{
font-size: 1.6em;
}
p {
font-size: 1.2em;
}
input.text
{
padding: 3px;
border: 1px solid #999999;
}
img {
max-width: auto;
height: auto;
}
.text-responsive {
font-size: calc(50% + 0.6vw + 0.6vh);
}
.text-responsive-h {
font-size: calc(% + 0.6vw + 0.6vh);
}
```

OUTPUT SCREENSHOTS

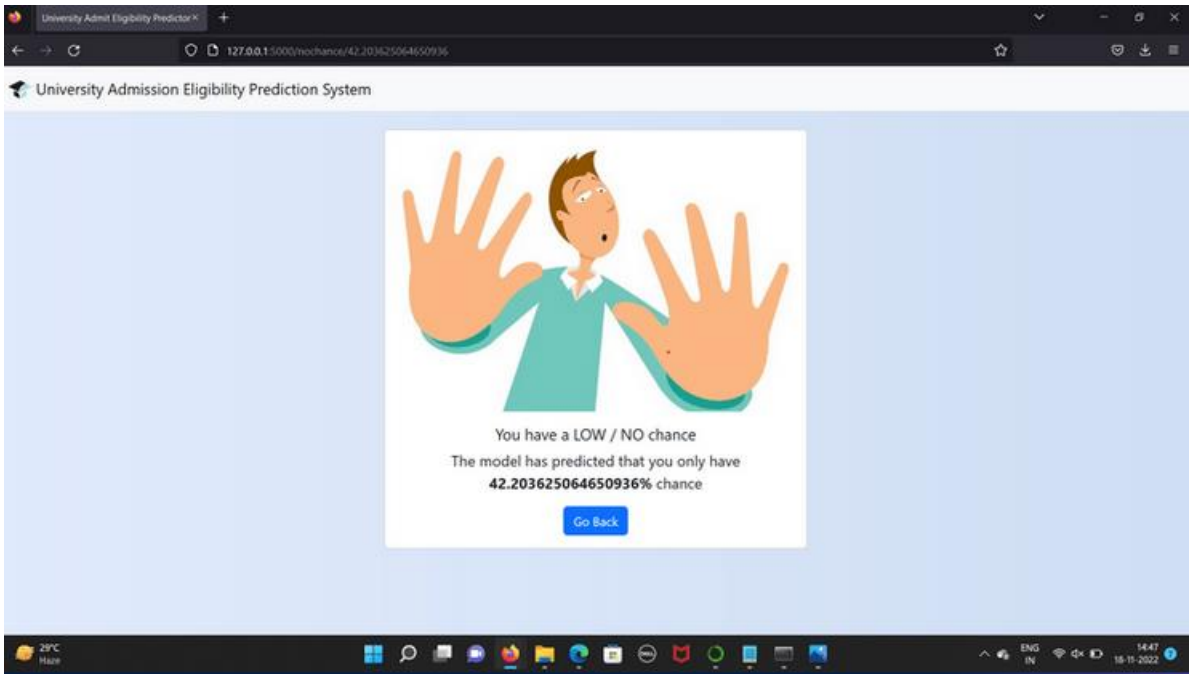
HOMEPAGE



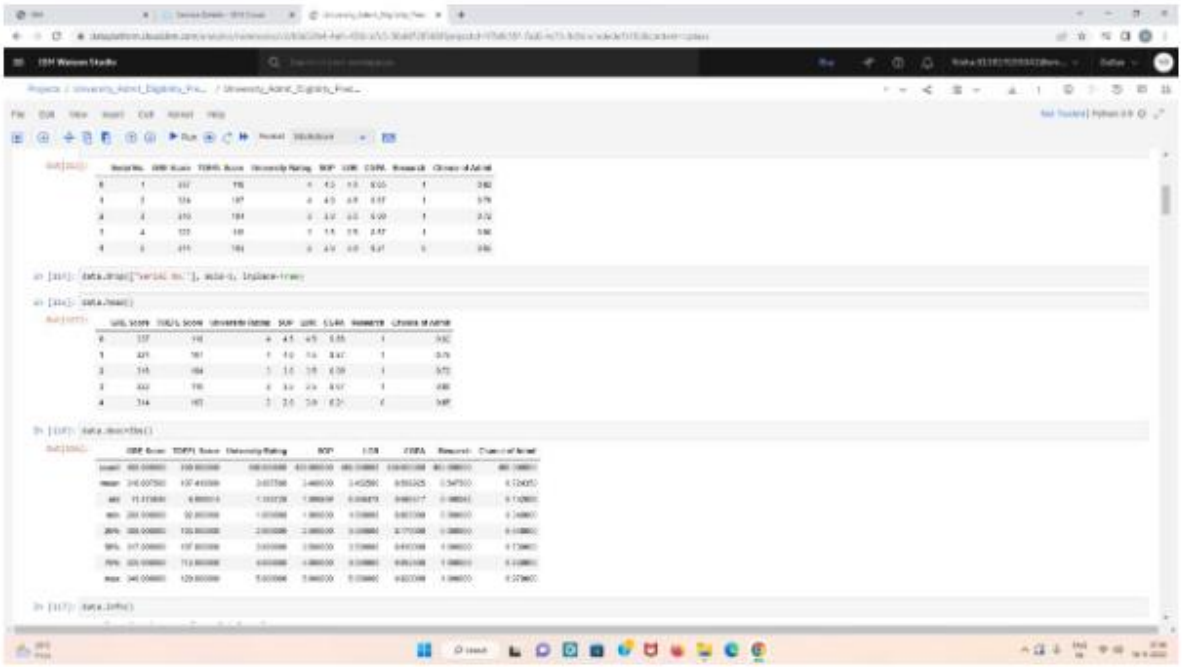
CHANCE

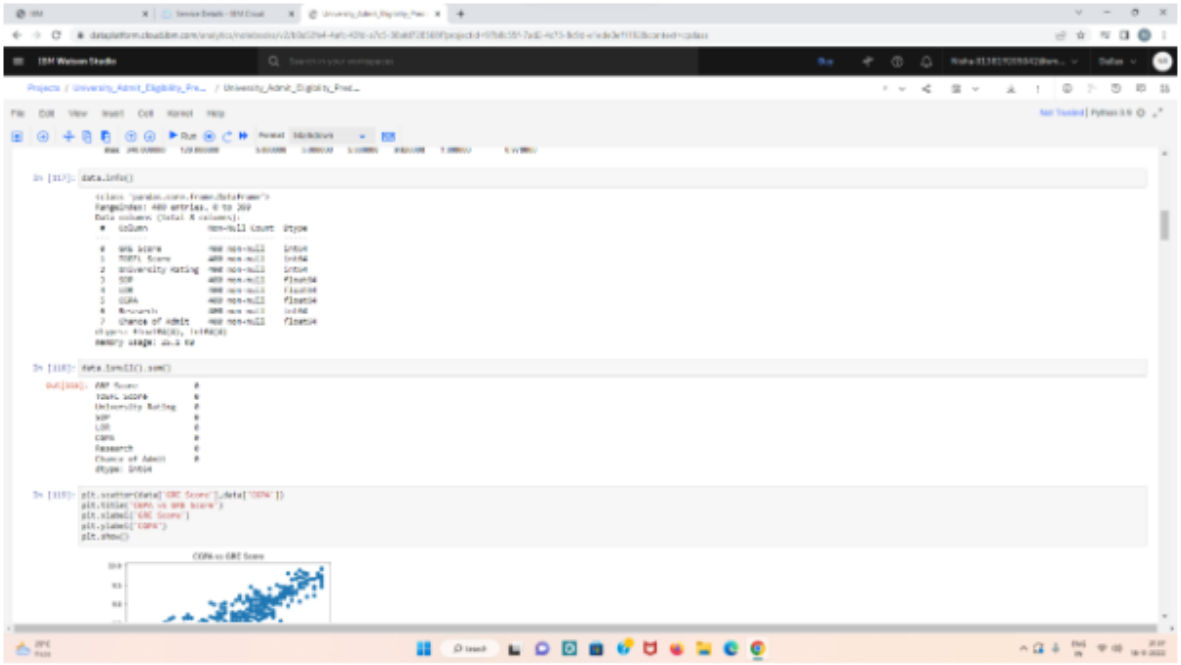


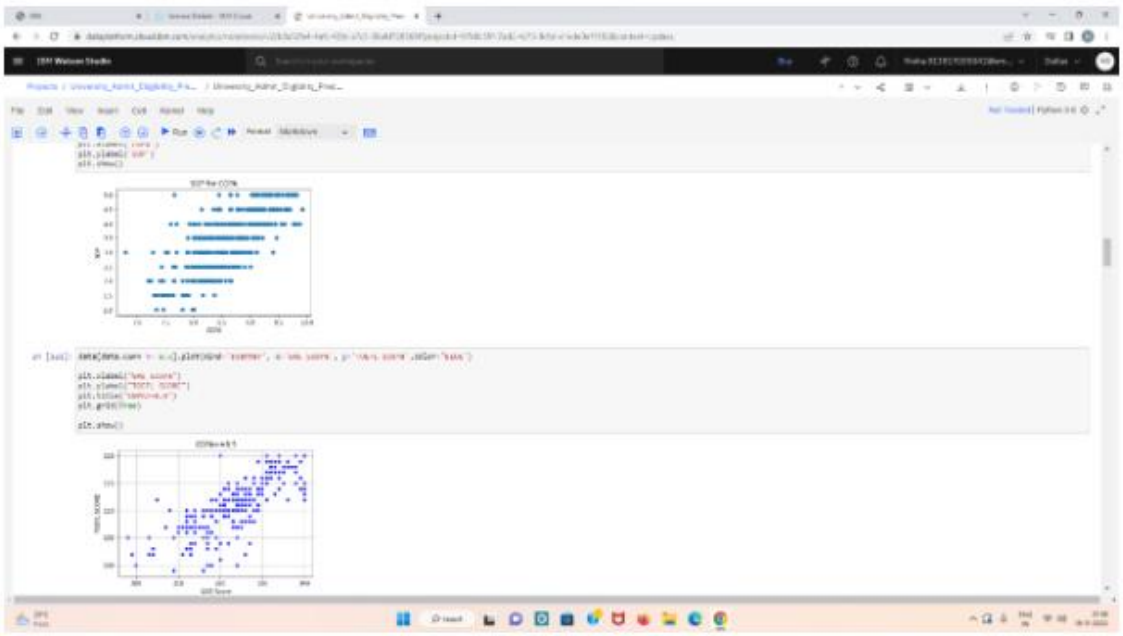
NOCHANCE

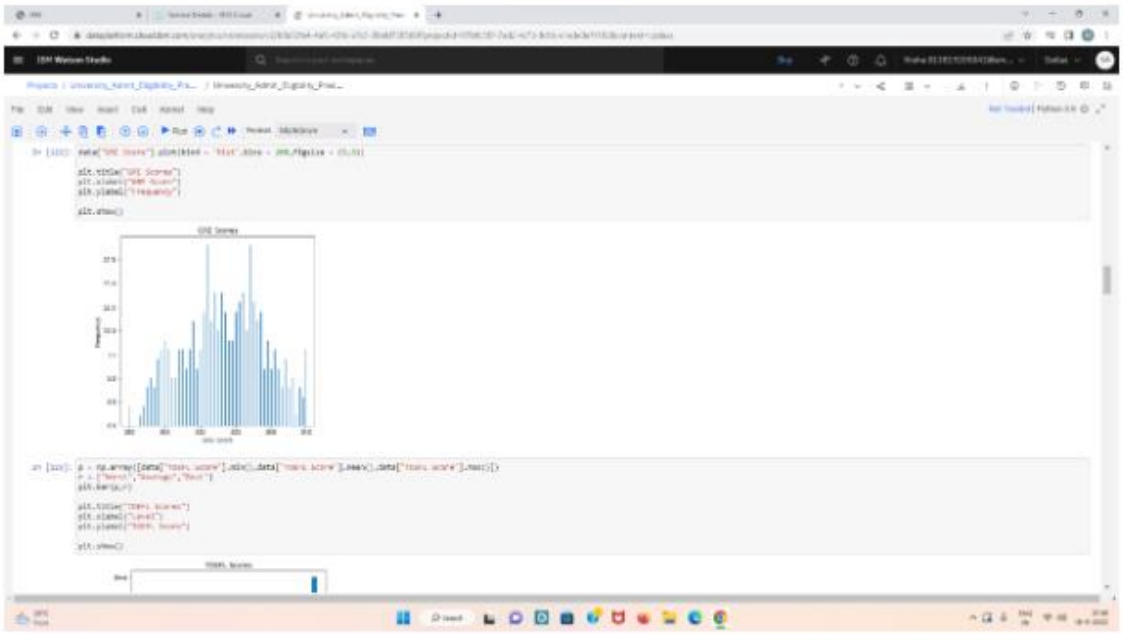


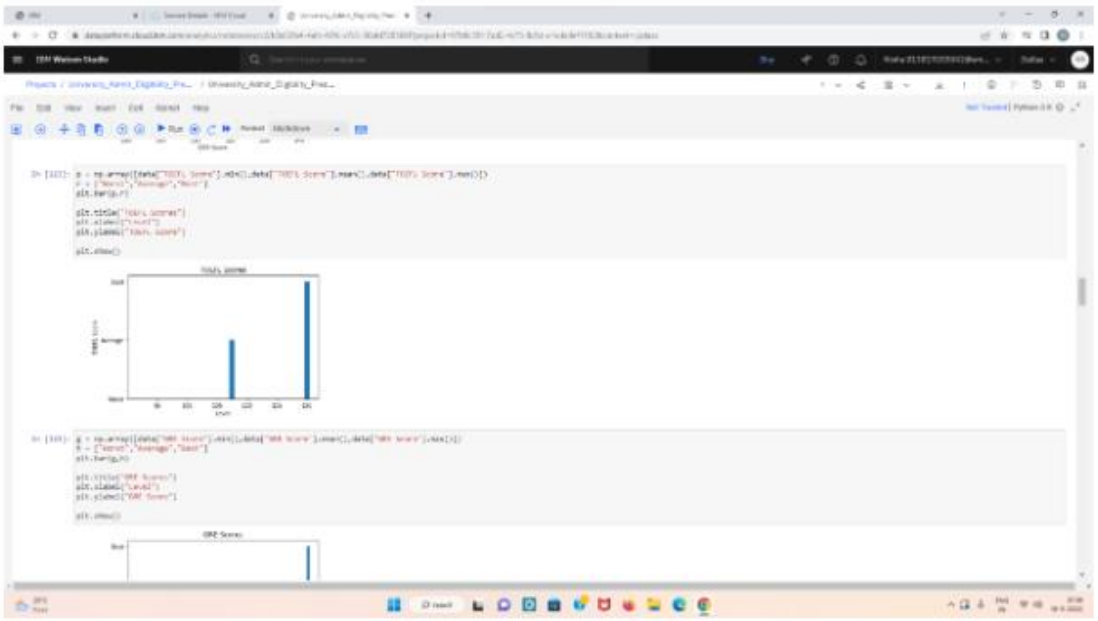
UNIVERSITY_ADMIT_ELIGIBILITY_PREDICTOR.ipynb

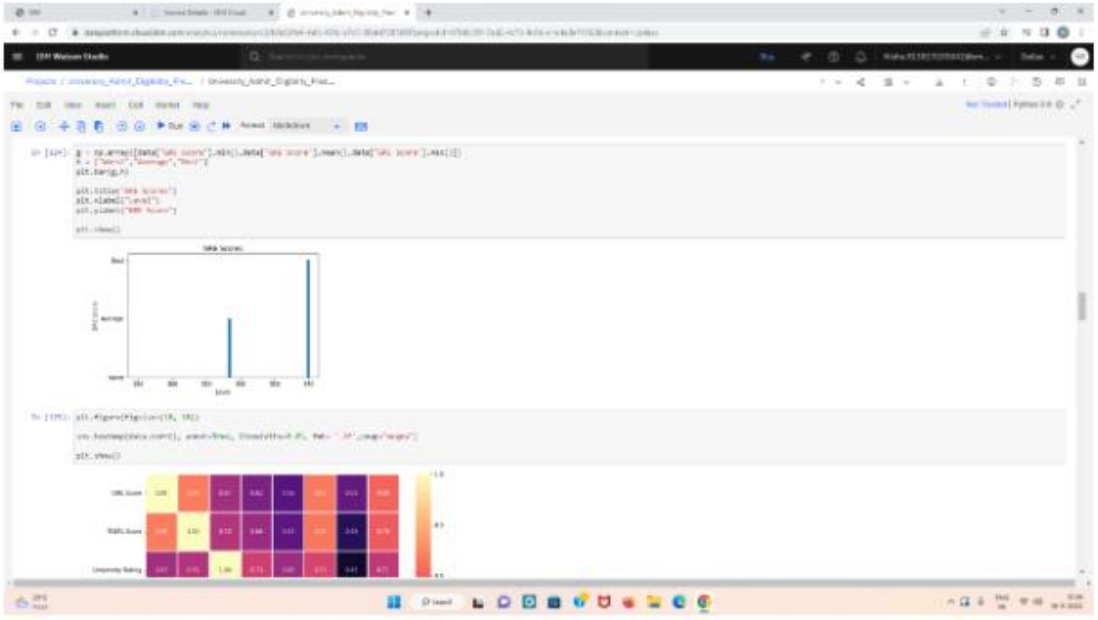


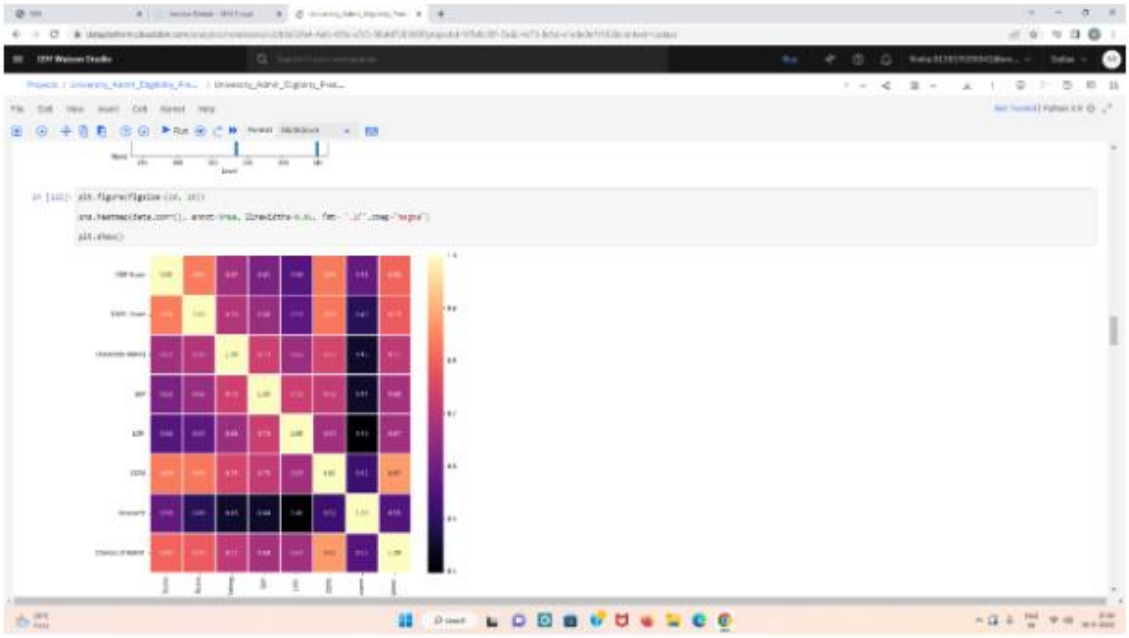


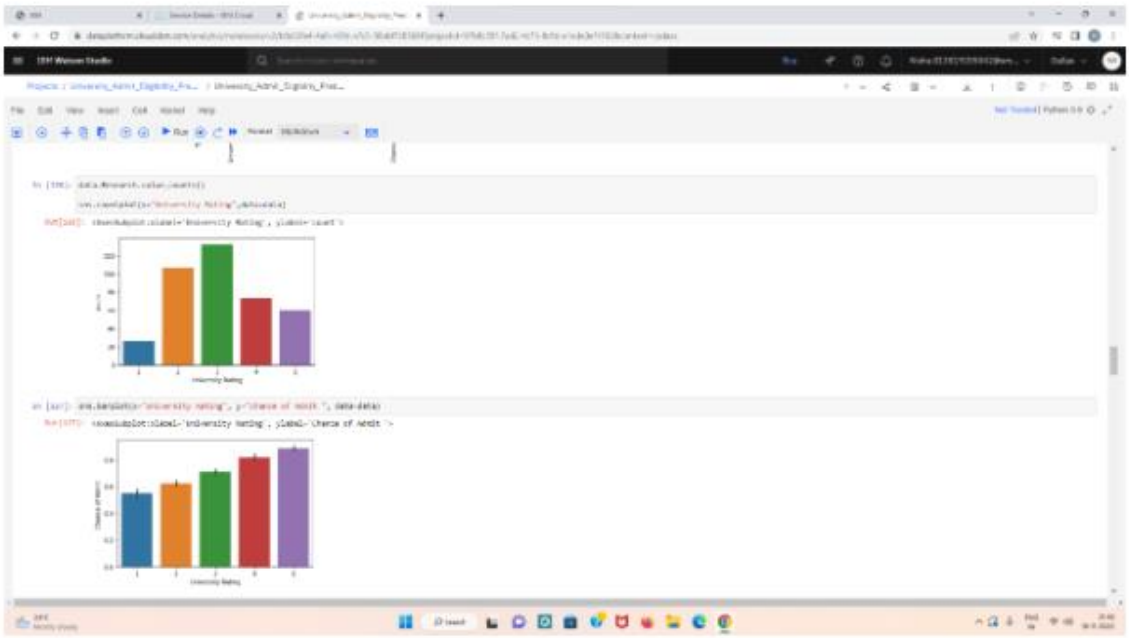


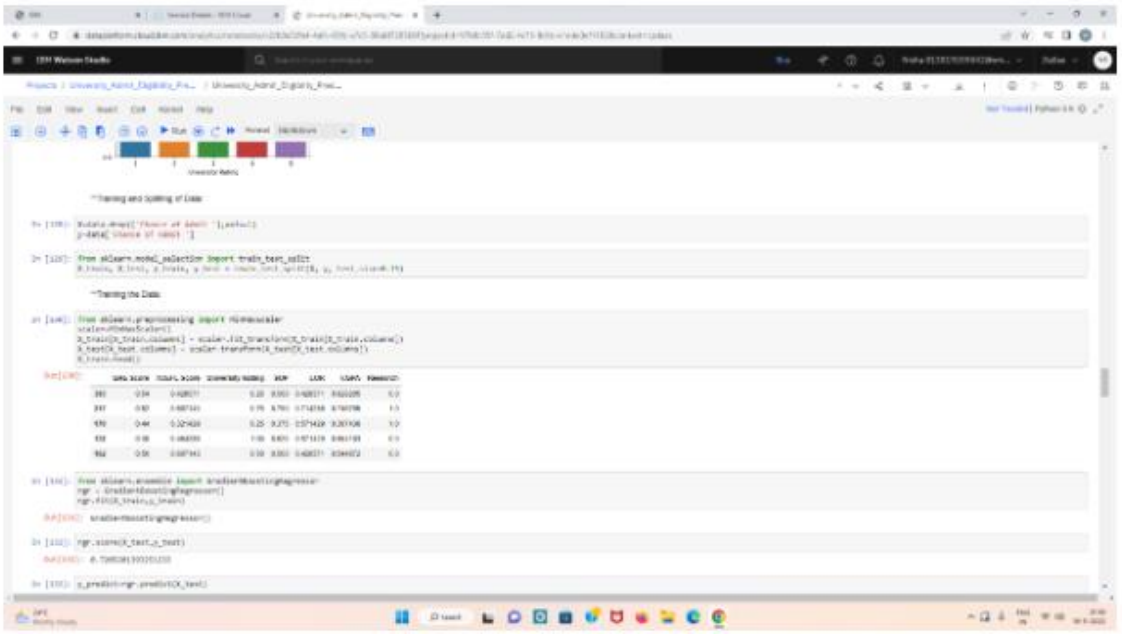


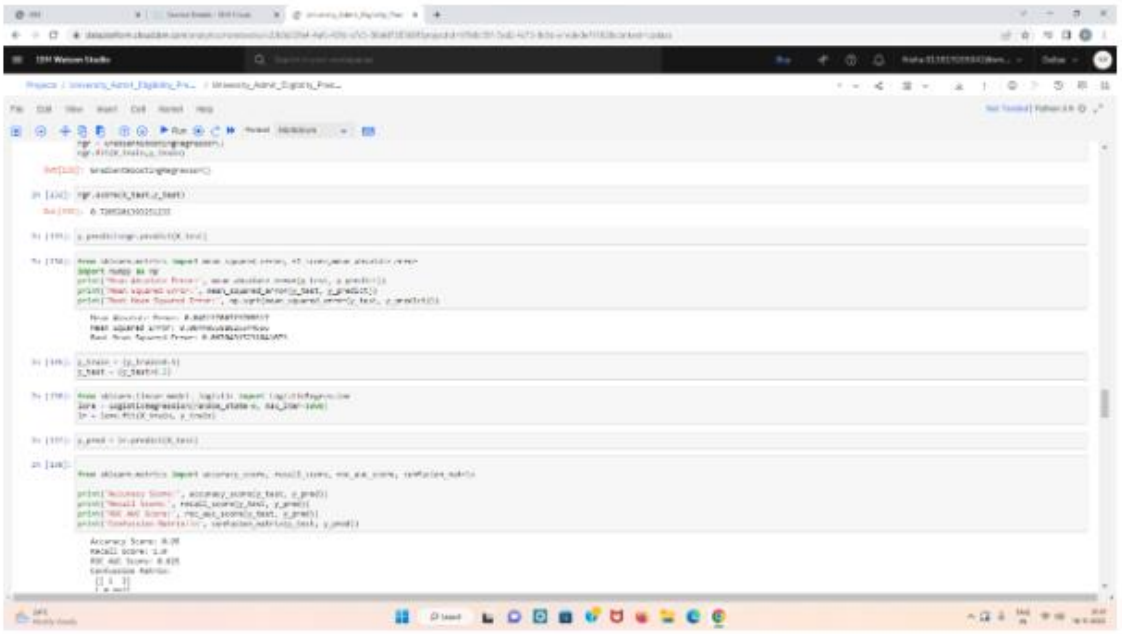


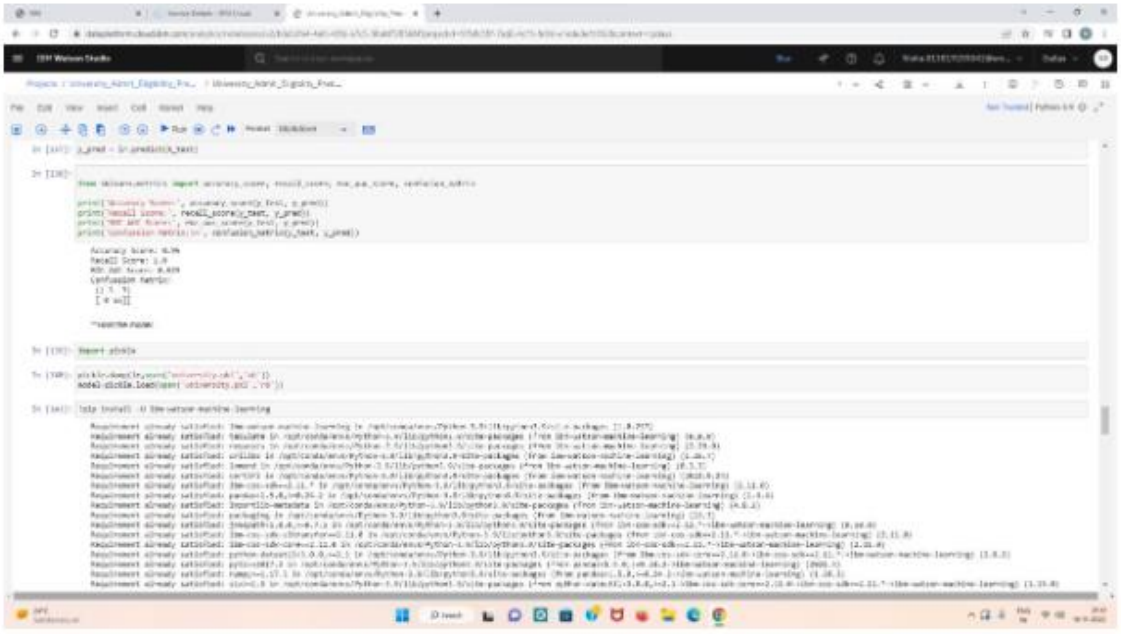


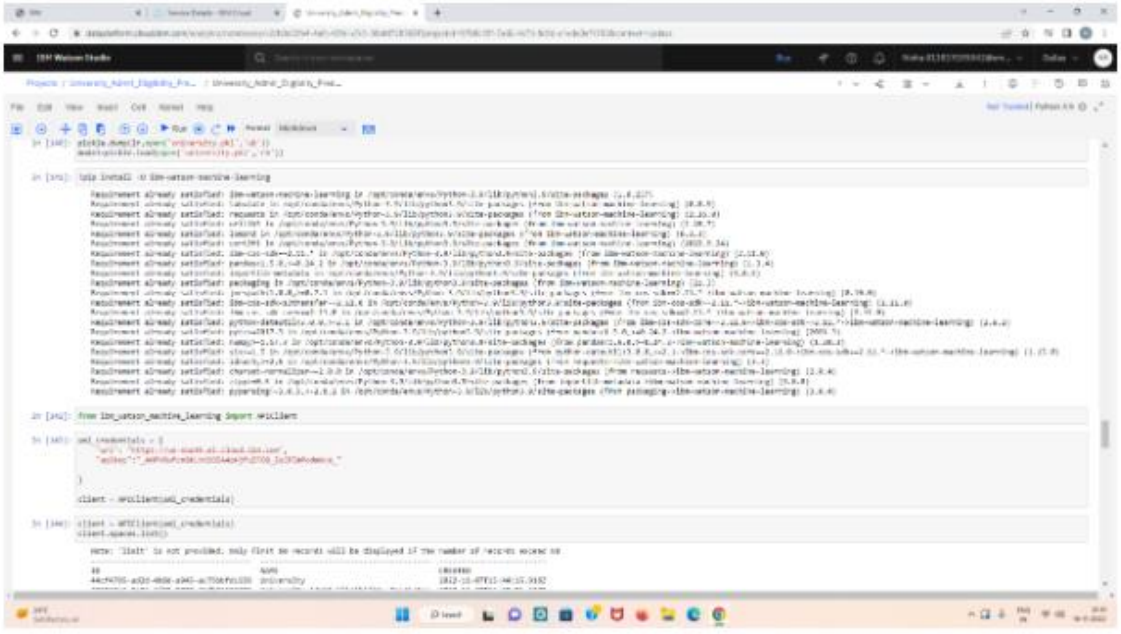


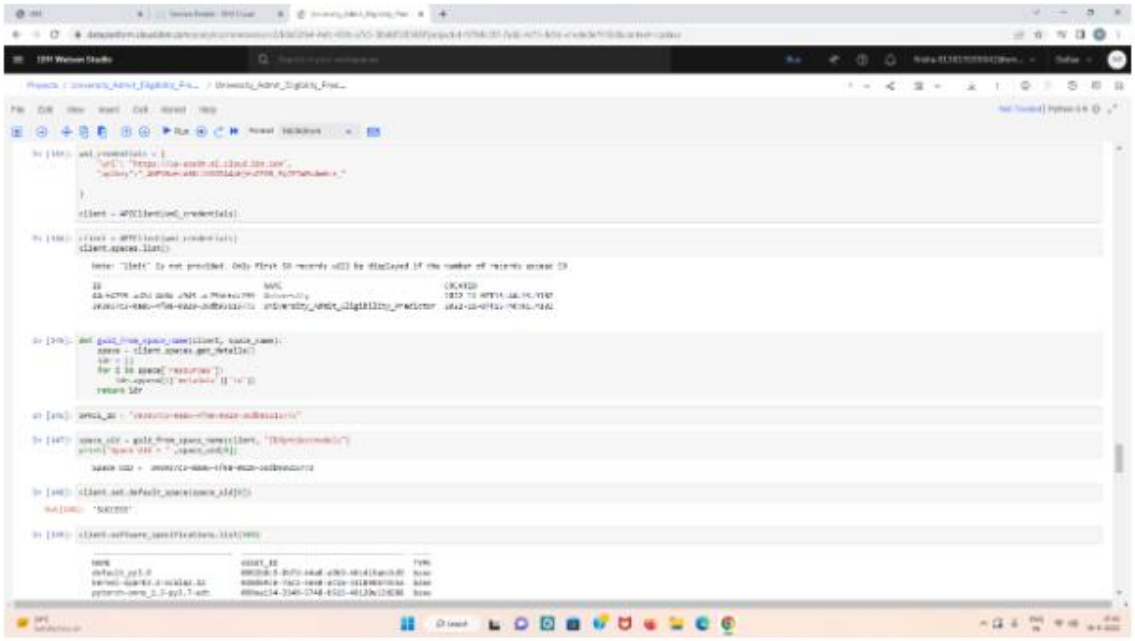


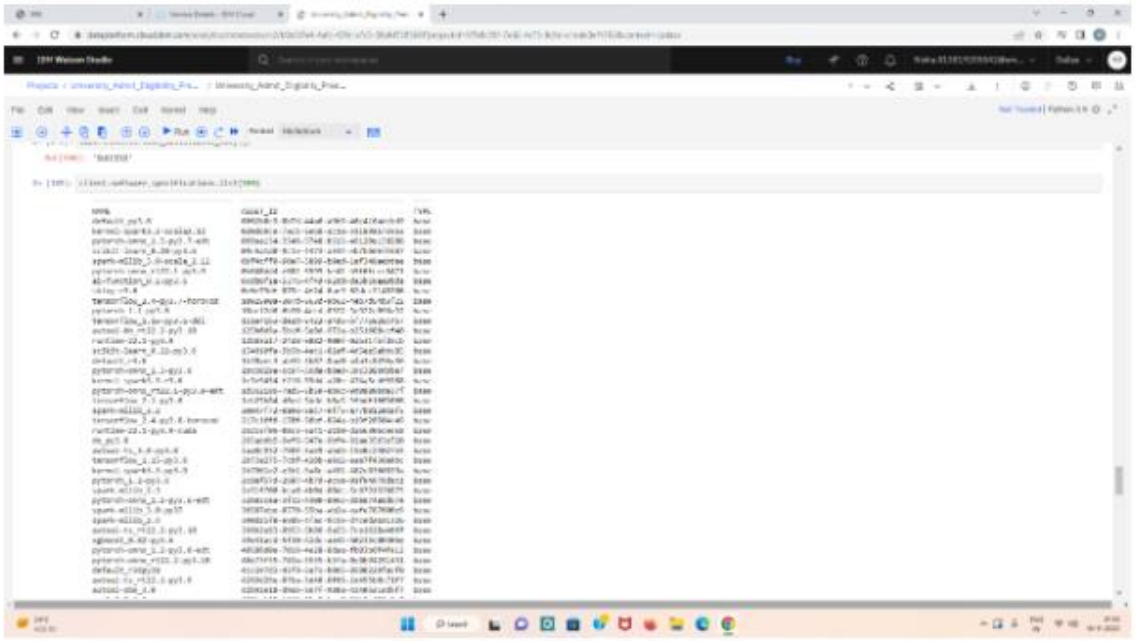


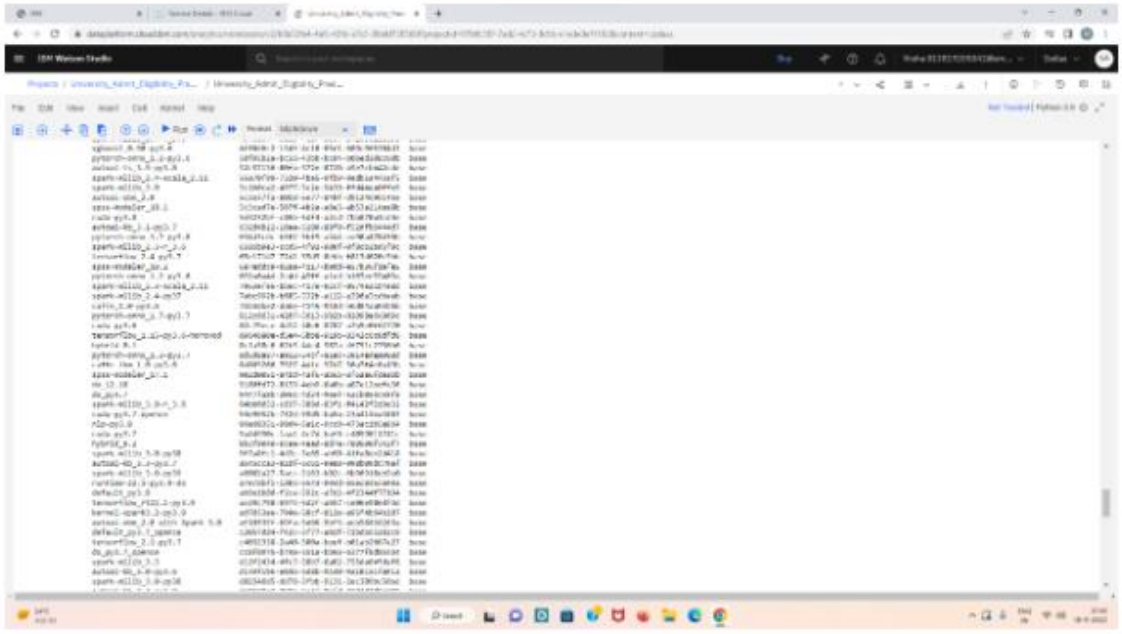


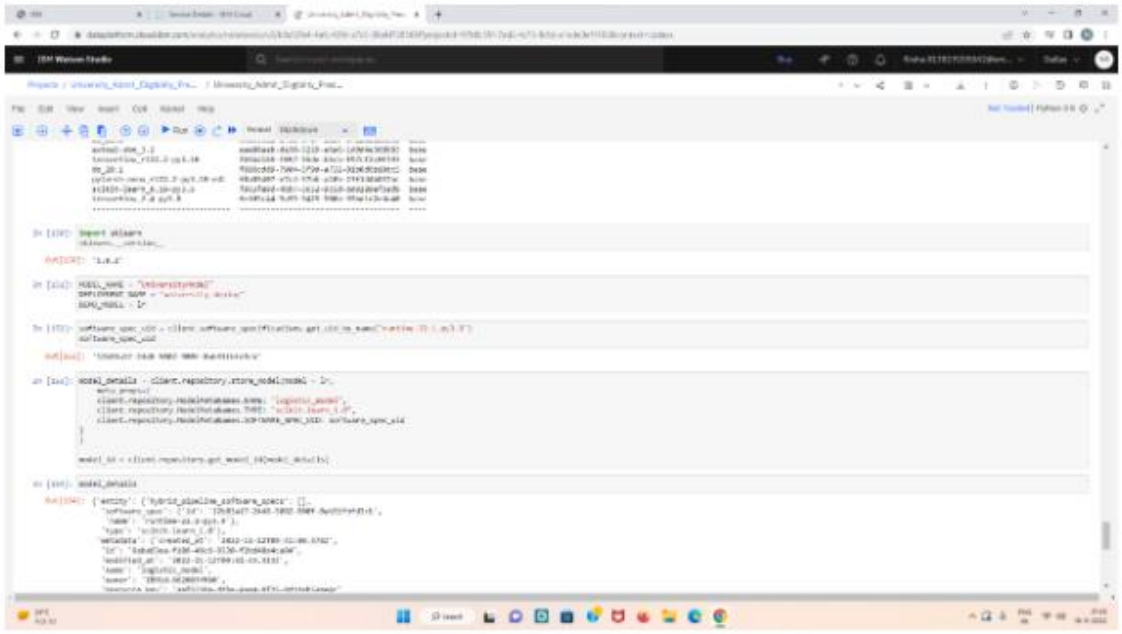


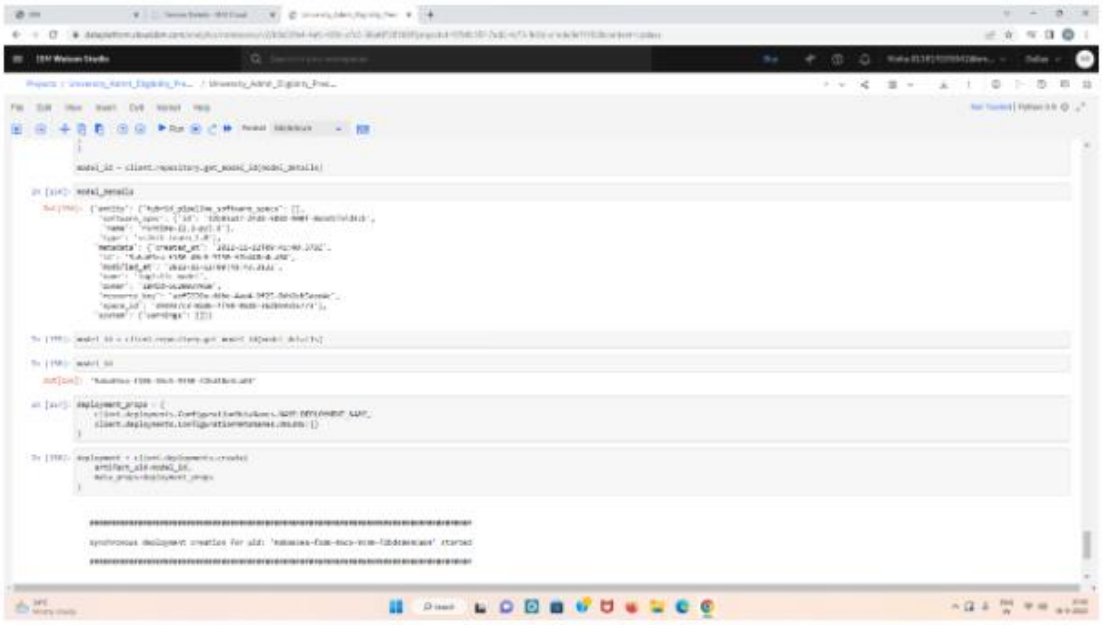


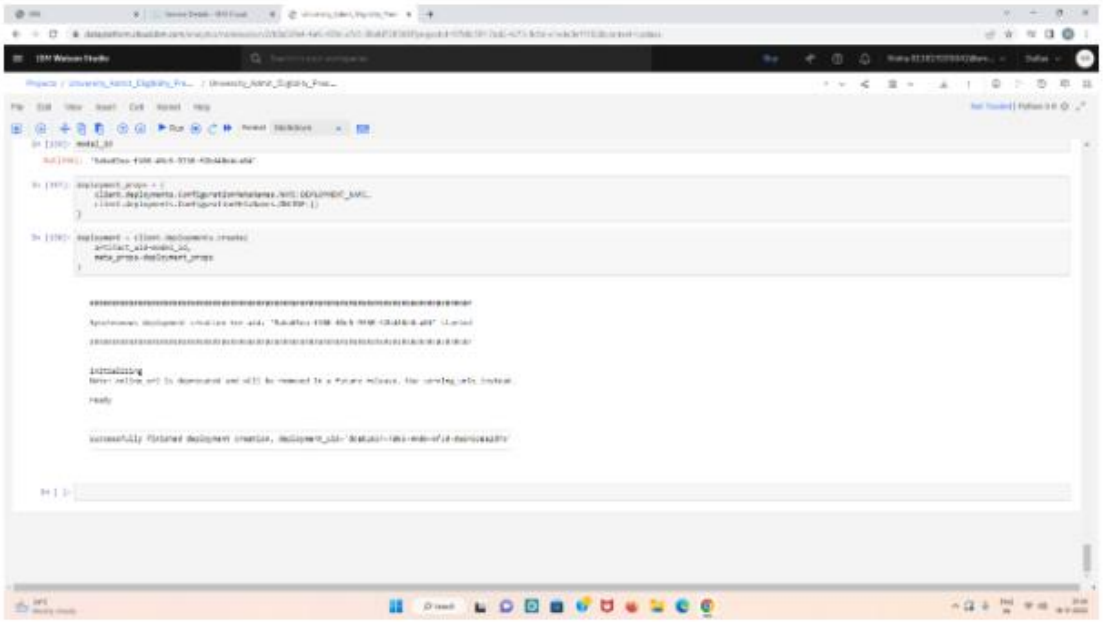












GITHUB LINK

GitHub : https://github.com/IBM-EPBL/IBM-Project-43293-1660715280