IBM - PROJECT - 5340 - 1658759200

UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

TEAM ID: PNT2022TMID35642

ABSTRACT

Several factors/scores are taken into account while deciding whether to admit a student to the Master's programme before the student is actually admitted. The selection committee must go through a lengthy, thought-out procedure and conduct extensive study before selecting the best candidates for the Master's programme. The goal of this research is to highlight the top scores that contributed to the student's acceptance to the master's programme. The analysis can appear simple, but care must be taken to take into account factors like the GRE, TOEFL, university rating, SOP, LOR, and CGPA. Any outliers should not have an impact on the decision-making process.

Chapter - 1

INTRODUCTION

1.1 Project Overview

The main goal is to talk about how admittance to universities can be predicted using a variety of variables using logistic regression. For master's programmes, many potential students submit applications. The admittance decision is based on requirements set forth by the specific college or degree programme. In order to forecast graduate school acceptance, the independent factors in this study will be statistically measured. If successful, exploration and data analysis would enable prediction models to improve applicant screening for master's degree programmes, which in turn would provide the admittance to qualified applicants.

1.2 Purpose

This document outlines the specifications for Unipredict, a novel web-based university admissions predictor. An AI-based tool called Unipredict allows users to enter data from their academic transcripts in order to estimate their chances of admission into the university tier they chose. Additionally, it offers a data analysis and demonstrates how different factors can affect admittance chances. The system's scope, aims, and goals are all described in this paper. This document depicts the functional needs with use cases, interaction diagrams, and class models in addition to documenting the nonfunctional requirements

Chapter - 2

LITERATURE SURVEY

A user can use the Education Based Prediction System to help them choose which universities to apply to based on their test results. The following parameters are included in the dataset that is processed: Name of the university, GRE Quantitative and Verbal Scores, TOEFL, and AWA Scores. Many colleges and graduate programmes use the GRE Test (Graduate Record Examinations) as a standard assessment tool for graduate admissions throughout the world.While submitting an application to a college, other variables are also taken into account, such as letters of recommendation (a formal document that attests to a person's work, abilities, or academic performance), test scores, and other relevant information. Co-curricular activities, research papers, and co-curricular activities are all important components of a graduate school application that tell admissions committees who you are, what your academic and professional interests are, and how you'll contribute to the graduate programme you're applying to. Research papers from lesser-known journals or those with a high percentage of plagiarism are not taken into consideration in this case.When a person has finished their undergraduate degree and wants to pursue a postgraduate degree in an area of their choice, it is typically quite difficult for them to determine which universities they should apply to given their GRE and TOEFL scores combined with their GPA at the time of graduation. A lot of time is wasted when candidates apply to institutions that do not accept their score requirements. Applying to numerous institutions with good scores also raises the price. An education predictor system has been created because there aren't many effective solutions accessible to aid with this problem.

2.1 Existing problem

Student information is currently manually entered in colleges. Separate records for each pupil are a laborious undertaking. Updating is required with regard to all of these records. More manual mistakes could occur.

- ★ The student's first year of college.
- ★ He or she first picks up an admissions form at the front desk.
- ★ Fills it out and delivers it to the office.
- ★ If there are any errors in the form, they are repaired once it has been reviewed with papers such as the university's merit list and details.
- ★ The candidate deposits the fees at the time of application.
- ★ The institute assigns the candidate an admission number at the time of application submission.
- ★ The candidate receives a receipt for the deposit of fees.

2.2 References

1. International Journal of computer communications
2. Journal of Network Communications and Emerging Technologies
3. European Journal of Molecular & Clinical Medicine
4. Advanced Computing and communication technologies for high performance applications

2.3 Problem Statement Definition

The Following System enables users to forecast their likelihood of being admitted to a university. The user might not be aware of the complete admissions process and might not know where to turn for information. Using the current year's entrance requirements, the model assists the user in identifying the best university and courses in this situation. Additionally, the model provides FAQ responses.

Chapter - 3
IDEATION & PROPOSED SOLUTION

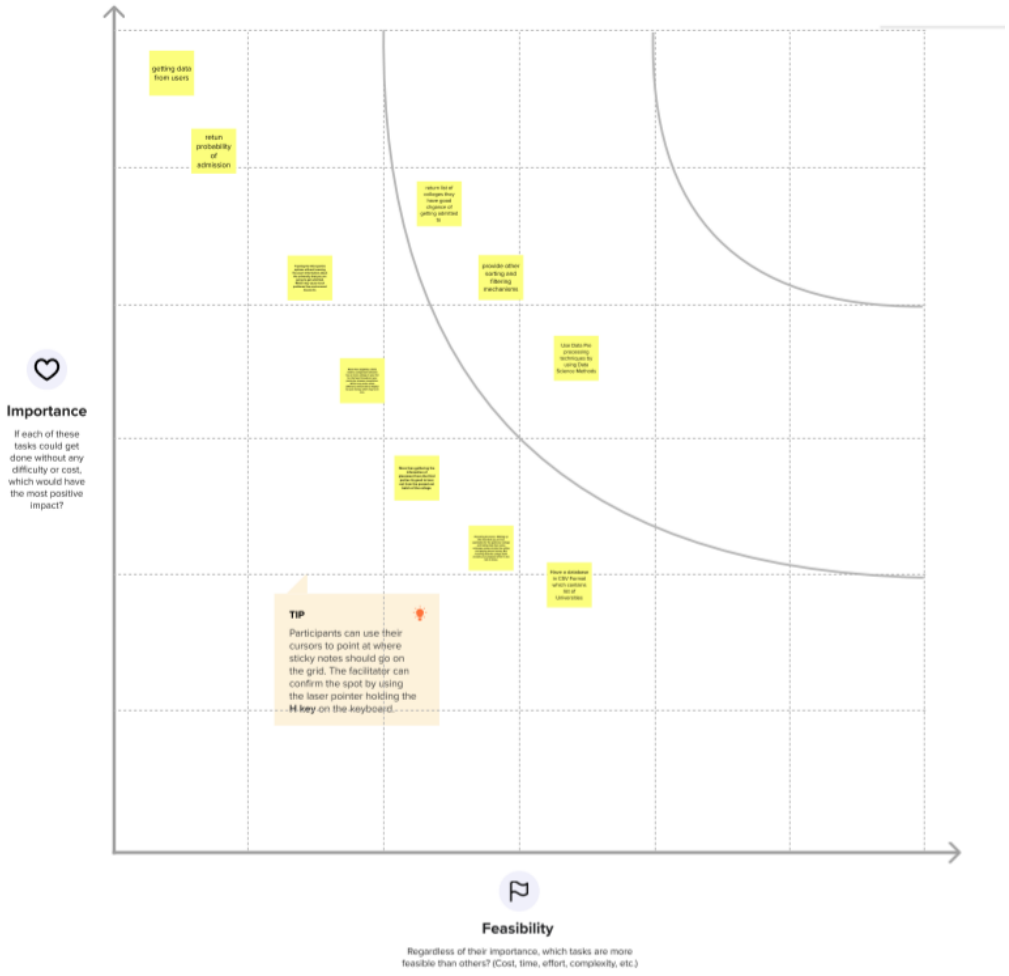
3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes



3.3 Proposed Solution

Use of Machine Learning models to predict current year's admission criteria of different universities and web applications for users to filter based on their preferences. It also provides Them answers to the most common FAQs that arise when thinking of admissions abroad for Postgraduate studies. Custom Filtering based on user specification.The Model can be improved as we gain more data about students.And, Details regarding the applicant's Statement of Purpose essay and Letters of Recommendation (a score can be assigned to rate these) can be used to improve the prediction accuracy. An alternative is to use Natural Language Processing methods to evaluate the essays and letters. In addition to this, the weightage given to the features can be varied according to past trends.

3.4 Problem Solution fit

1. **Customer Segment(s):** Students who have completed 12th grade and have their TOFEL,GRE,LOR,SOP.
2. **Problems:** Data collection is probably the most important step in designing the predictor hence it must be ensured that it is done properly. Customers should be assured of optimum data security in order to sustain their trust in our model.
3. **Triggers:** Users can be provided with comparisons between the required scores versus their actual scores.
4. **Emotions:** Users would feel that they are in complete control in the admission process since they can wholeheartedly trust the predictor.
5. **Available Solutions:** In addition to indicators like grades and GPA, we will also take into account IELTS/TOFEL,GRE, which is important in the admissions process of several colleges, further

increasing the predictor's dependability. Secondly, we will put the model through rigorous tests in order to boost the accuracy of the predictor.

6. **Customer Constraints:** Spending more money unwanted, Time and Energy in lack of Poor Knowledge.
7. **Behavior:** Customer gathers their data and inputs it to our University Admit Eligibility tracker and checks for chance of admission into universities based on the input got from the user..
8. **Channels of Behaviour:** Customers might search for reliable eligibility predictors that are available online and rate them based on their liking.
9. **Problem Root Cause:** If the obtained data is determined to be erroneous or not enough parameters are taken into account to determine eligibility, the predictor's reliability may be impacted. In addition, clients can stop utilizing our product.
10. **Poor Solution:** Design a predictor with the help of the data collected, and ensure that it is accurate/ reliable. Also make sure that the data collected from the users is safe and secure.

Chapter - 4
REQUIREMENT ANALYSIS

4.1 Functional requirement

4.1.1 User Details

Submit the following documents: GRE Score Marksheet ,TOEFL Score Sheet , Statement of Purpose (SOP),Letter of Recommendation (LOR),CGPA,University rating.

4.1.2 Prediction

Prediction is performed based on the user details using the Logistic Regression Model and the percentage of chance of getting into a university of a particular rating entered by the user is obtained.

4.2 Non-Functional requirements

4.2.1 Usability

- The system doesn't expect any technical prerequisite from the user.
- User friendly.
- The page would not take a lot of time to load the content and display them.

4.2.2 Security

- Data that is fed by users will be highly secure and not accessed by unwanted users.

4.2.3 Reliability

- The system would always strive for maximum reliability due to the importance of data and damages that could be caused by incomplete and incorrect data.

4.2.4 Performance

- Users will be able to handle the process on our website even by having an internet connection with normal speed. There is no need for a high -speed internet connection.
- Traffics can be handled effectively

4.2.5 Availability

- Students will be able to access our application as it will be accessible from any browser and any system capabilities

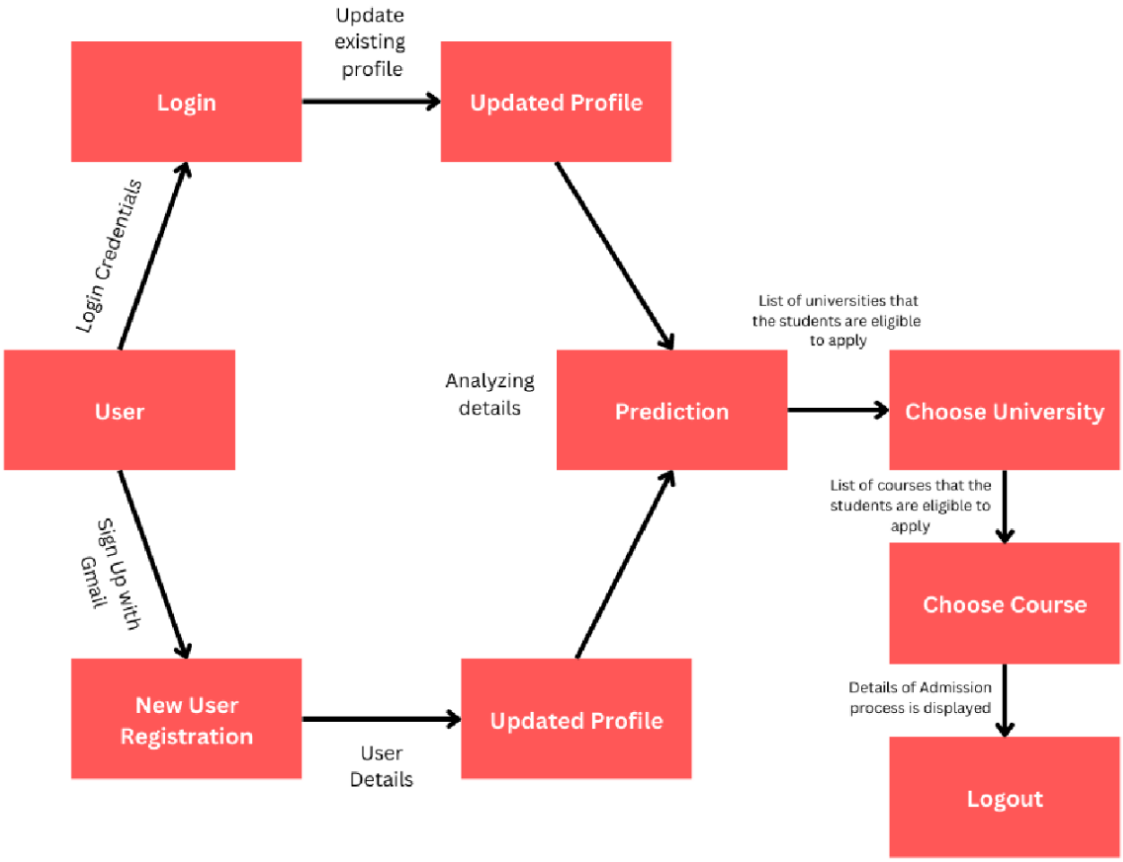
4.2.6 Scalability

- Our website will be easily scalable even when large number of users access our application
- Deployment in IBM cloud platform.

Chapter - 5

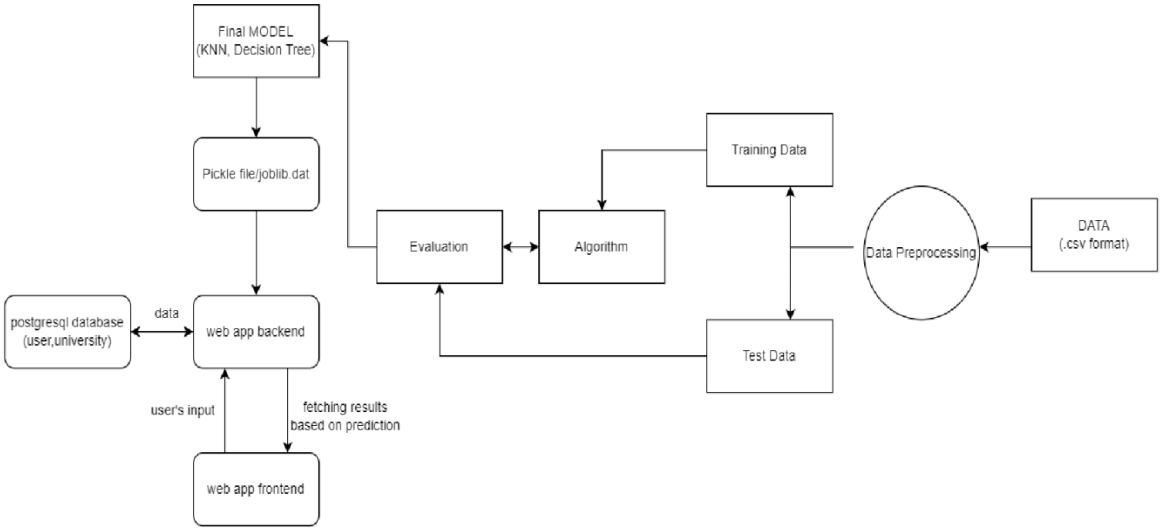
PROJECT DESIGN

5.1 Data Flow Diagram

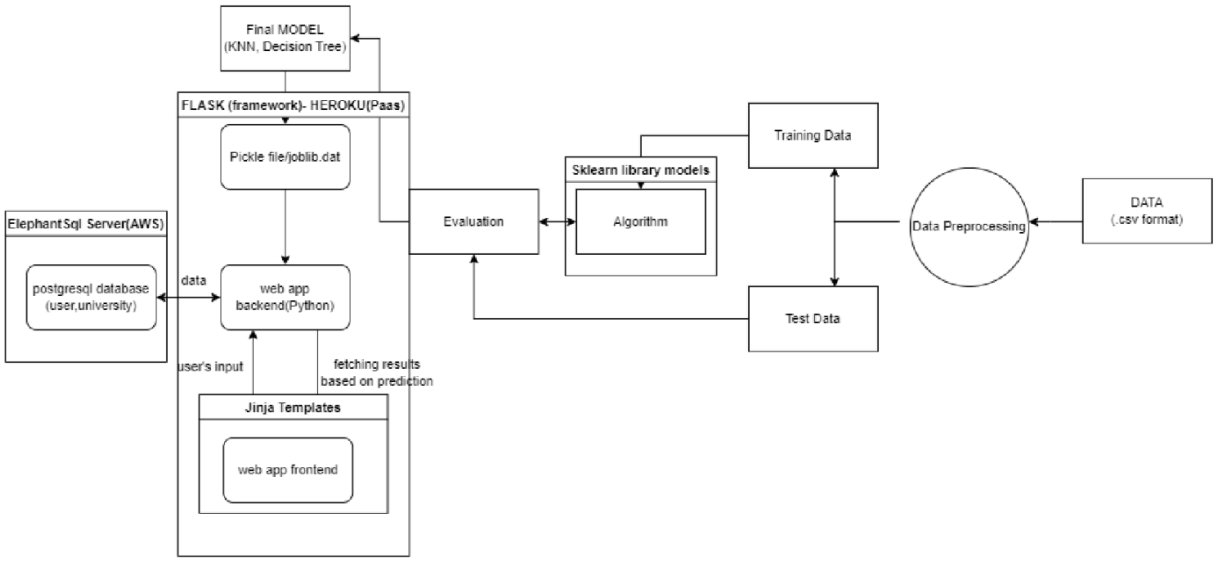


5.2 Solution & Technical Architecture

Solution Architecture:



Technical Architecture:



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile 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
		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
		USN-3	As a user, I can register for the application through Gmail	I can access my account	Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password	I can access my account	High	Sprint-1
	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
	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 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 provided in the drop-down menu	Medium	Sprint 3
Customer Care Executive	Admission Process	USN-8	As a user, I will be able to view the details of the admission process like date, venue of certification verifications.	I can view the details of admission process being displayed at the end of the prediction.	Low	Sprint 4
Administrator	Authentication	USN-9	As a admin, the login credentials of the user is authenticated by me.	I can retrieve and make use of all the user details.	High	Sprint 1
	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
	Prediction	USN-11	As a Admin, I can test the trained ML model by analysing the user details by ML Algos	I can test the user data with trained ML Model	High	Sprint 3
	Output	USN-12	As a admin, I can upload the confirmation of the user for prediction into Database.	I can keep track of user data & activities by storing database.	High	Sprint 4

Chapter - 6

PROJECT PLANNING & SCHEDULING

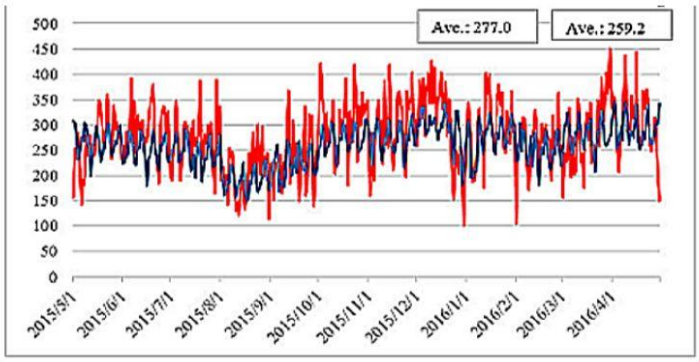
6.1 Sprint Planning & Estimation

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	10	High	All
Sprint-1	Gmail notification	USN-2	As a user, I will receive confirmation email once I have registered for the application	5	Low	All
Sprint-2	Login	USN-3	As a user, I can login with my credential and get a personalized dashboard	5	Low	All
Sprint-2	View institutes	USN-4	As a user, I can view list of institutes	10	Low	All
Sprint-3	prediction	USN-5	As a user, I can provide my scores for prediction	10	High	All
Sprint-4	preferences	USN-6	As a user I can select and filter results based on my preferences	20	High	All

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	15	6 Days	24 Oct 2022	29 Oct 2022	15	29 Oct 2022
Sprint-2	15	6 Days	31 Oct 2022	05 Nov 2022	15	05 Nov 2022
Sprint-3	10	6 Days	07 Nov 2022	12 Nov 2022	10	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 : Flask App

```python
from flask import Flask, render_template, redirect, url_for, request
import requests, joblib

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("index"))
            arr.append(float(val))
        Serial_No =1

        gre = float(request.form['gre'])
        tofel = float(request.form['tofel'])
        university_rating = float(request.form['university_rating'])
        sop = float(request.form['sop'])
        lor = float(request.form['lor'])
        cgpa = float(request.form['cgpa'])
        yes_no_radio = float(request.form['yes_no_radio'])

        X =[[gre,tofel,university_rating,sop,lor,cgpa,yes_no_radio]]


        # NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
        API_KEY = "6S922LYgmx66Kf3lp7WQqCjc9jyywHJwB9IXjJKdGdW6"
        token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"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}

        # NOTE: manually define and pass the array(s) of values to be scored in the next line
        payload_scoring = {"input_data": [{"field": [["GRE Score","TOEFL Score","University Rating","SOP","LOR ","CGPA", "Research"]], "values": X}]}

        response_scoring =
        requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/4d27e05d-bdd3-4437-aa02-e0989a1ed38c/predictions?version=2022-11-18', json=payload_scoring,
        headers={'Authorization': 'Bearer ' + mltoken})
        response_scoring = response_scoring.json()
```

```python
            #model = joblib.load('model.pkl')
            #result = model.predict(X)[0]
            result = response_scoring['predictions'][0]['values'][0][0][0]
            if result> 0.5:
                return redirect((f'/chance/{round(result*100)}'))
            else:
                return redirect((f'/no_chance/{round(result*100)}'))
        else:
            return redirect(url_for("demo"))


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

@app.route("/chance/<percent>")
def chance(percent):
    return render_template("chance.html", percent=percent)

@app.route("/no_chance/<percent>")
def no_chance(percent):
    return render_template("noChance.html", percent=percent)

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

7.2 Feature - 2 : UI

Index.html

```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">
    <link rel="stylesheet" href="../static/style.css">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">

    <title>University Eligibility Predictor</title>
</head>
<body>
    <h1 class="text-center mt-3">University Eligibility Predictor</h1>


    <div class="heading">
        <div class="ml-1 p-4" >
            <!-- <img src="/static/trend.png" alt="like">-->

        </div>
        <div class="ml-1  p-4">

        </div>
        <div class="ml-2 p-4">

        </div>
    </div>

    <div class="col-6 " id="main" >
        <div class="card p-2 ms-2 my-2" style="background-color: rgb(212, 205, 205);">
            <div class="card-body" >
                <h5 class="card-title pb-4 text-center">
                    Enter the details
                </h5>
                <form action="/" method="post"  style="background-color:rgb(212, 205, 205) ;" 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" placeholder="250 to 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" placeholder="50 to 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" placeholder="1 to 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" placeholder="1 to 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" placeholder="1 to 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" placeholder="5 to 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 justify-content-center">
                        <div class="col-lg-2 mb-2 me-3">
                            <button type="submit" class="btn btn-primary  m-auto" style="padding:8px 25px; "  id="button">Predict</button>
                        </div>

                </form>
                </div>
            </div>
        </div>

</body>
</html>
```

Chance.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">
    <link rel="stylesheet" href="../static/style.css">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">

    <title>Document</title>
</head>
<body>

    <h1 class="text-center" id="title" ><strong>Congratulations you are eligible</strong></h1>
    <div class="m-5 d-flex flex-row ">
        <div class="col-sm-8">
            <div class="alert alert-success" style="height:70px;">
                <strong><h3>There is high chance about {{ percent }} percentage chance for you in this rating university.</h3></strong>
             </div>

        </div>
       <div class="m-5" id="mark1"><span><img id="resultImage" src="../static/green thumb.jpg"></span></div>
       </div>
       <div class="m-5 d-flex flex-row ">

        <div class="">
            <!--<img id="image"src="/static/success1.gif" alt="success">-->
          </div>
       </div>

       <div class="d-flex justify-content-center">
        <button class="btn btn-primary " style="margin-top: 0;"><a href="/" style="color:white; text-decoration : none" >Go Back</a></button>
       </div>
</body>
</html>
```

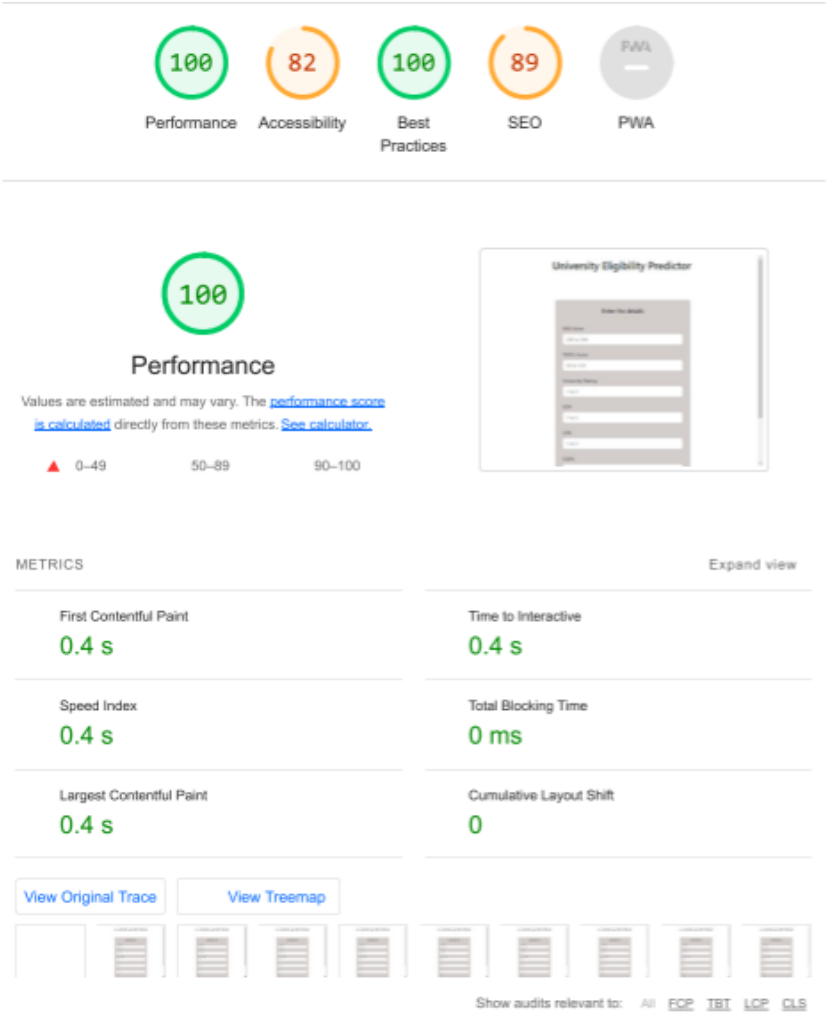
Nochance.html

```
<!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">
    <link rel="stylesheet" href="../static/style.css">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">

    <title>Document</title>
</head>
<body>

    <h1 class="text-center" style="margin-top:25px; font-style: italic; font-weight:400; font-size: 50px;"><strong>Sorry you are not Eligible</strong></h1>
    <div class="m-5 d-flex flex-row ">
        <div class="col-sm-8">
            <div class="alert alert-warning" style="height:70px;">
                <strong><h3>There is low chance about {{ percent }} percent chance for you in this rating university.</h3></strong>
              </div><br>

        </div>
       <div class="m-5" id="mark2"><span><img id="resultImage" src="../static//failure.jpg"></span></div>
       </div>
       <div class="m-5 d-flex flex-row ">

        <div class="">

          </div>
       </div>

       <div class="d-flex justify-content-center">
        <button class="btn btn-primary " style="margin-top: 0;"><a href="/" style="color:white; text-decoration : none" >Go Back</a></button>
       </div>
</body>
</html>
```

Chapter - 8

TESTING

8.1 Test Cases



8.2 User Acceptance Testing

A crucial step in the development process is user acceptance testing (UAT). It not only alerts the development team to aspects that don't yet satisfy the needs of the users, but it also offers governance a better notion of progress if done as early and frequently as possible. Defects become costly and challenging to rectify if UAT is postponed.

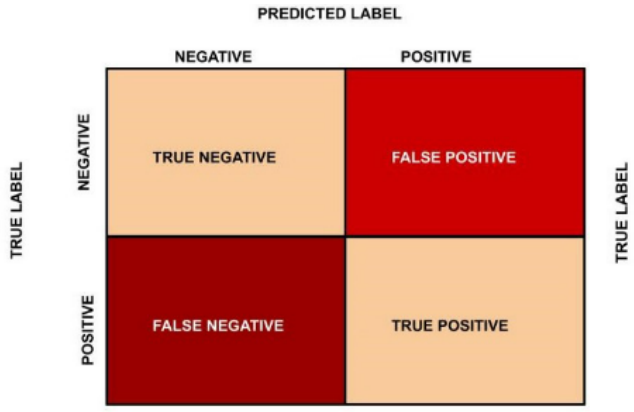
Chapter - 9

RESULTS

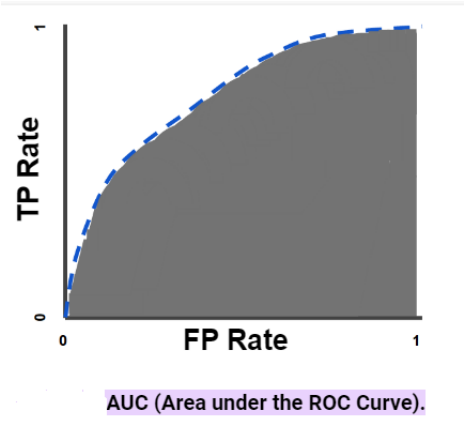
9.1 Performance Metrics

```
Accuracy Score: 0.9333333333333333
Recall Score: 1.0
ROC AUC Score: 0.6
Confussion Matrix:
 [[ 1  4]
 [ 0 55]]
```

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN} \quad Recall = \frac{TP}{TP + FN}$$



Confusion Matrix



AUC (Area under the ROC Curve).

ADVANTAGES & DISADVANTAGES

Advantages

- ★ Know the percentage - not just a yes or no but the percentage of success
- ★ Lower investigation and Time consumption- as ML does the prediction
- ★ Provide Relevant Material - does not ask for unnecessary user data
- ★ Good user experience-as UI/UX is simple and easy to use.

Disadvantages

- ★ Significant investments required - for development of enterprise level applications.
- ★ Unable to capture changes - as data changes we need to ingest newer data(such as release of final admission criteria by some university) and dynamically retrain and deploy ML models using messaging systems like Apache kafka and RabbitMQ services .
- ★ Privacy concerns - more security and privacy at enterprise level could be added.

Chapter - 11

CONCLUSION

The findings of this investigation seem to suggest that it makes a significant contribution to the response variable "Chance of Admit." The likelihood of admission increases with greater GRE and TOEFL scores. Based on the fore mentioned parameters, the model may be used to predict the likelihood of admission with accuracy .This approach will assist institutions forecast admission decisions and streamline their selection and timetable processes. The model supported the premise that GRE, TOEFL, and other test scores are required for admission to Master's degree programmes. The collection of more data from students at various universities with comparable selection criteria to determine applicants for master's programmes would probably considerably improve this model. Hence, We have achieved building a Admission Eligibility Predictor ML model and built a website so that it's easy for users to avail the model’s prediction capability through a simplistic Web UI interface .

Chapter - 12

FUTURE SCOPE

The project can be extended to support a real time system that retrains ML model everytime a university publishes its final admission criteria and deploys the new model immediately with zero downtime between switching the active ML models**.**

Chapter - 13

APPENDIX Source code GitHub & Project Demo Link

Github Link:
https://github.com/IBM-EPBL/IBM-Project-5340-1658759200

Demo Link:
https://github.com/IBM-EPBL/IBM-Project-5340-1658759200/tree/main/Final%20Deliverables/Demo_Video