

UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

A PROJECT REPORT

Submitted by

NARESHKUMAR V 732119205030

ANANTHAN A 732119205004

MYTHESHWARAN G 732119205027

SAKTHIVEL MURUGAN S 732119205043

TEAM ID : PNT2022TMID32265

NANDHA COLLEGE OF TECHNOLOGY, ERODE -52

Vaikalmedu, Perundhurai - 638 052

TABLE OF CONTENTS:

| S. NO | TITLE |
|-------|---|
| 1 | INTRODUCTION |
| 1.1 | Project Overview |
| 1.2 | Purpose |
| 2 | LITERATURE SURVEY |
| 2.1 | Existing problem |
| 2.2 | Problem Statement Definition |
| 3 | IDEATION & PROPOSED SOLUTION |
| 3.1 | Empathy Map Canvas |
| 3.2 | Ideation & Brainstorming |
| 3.3 | Proposed Solution |
| 3.4 | Problem Solution Fit |
| 4 | REQUIREMENT ANALYSIS |
| 4.1 | Functional requirements |
| 4.2 | Non-Functional requirements |
| 5 | PROJECT DESIGN |
| 5.1 | Data Flow Diagrams |
| 5.2 | Solution & Technical Architecture |
| 5.3 | User Stories |

| | |
|-----------|--|
| 6 | PROJECT PLANNING & SCHEDULING |
| 6.1 | Sprint Planning & Estimation |
| 6.2 | Sprint Delivery Schedule |
| 6.3 | Reports from JIRA |
| 7 | CODING & SOLUTIONING |
| 7.1 | Feature 1 |
| 7.2 | Feature 2 |
| 7.3 | Database Schema |
| 8 | TESTING |
| 8.1 | Test Cases |
| 8.2 | User Acceptance Testing |
| 9 | RESULTS |
| 9.1 | Performance Metrics |
| 10 | ADVANTAGES & DISADVANTAGES |
| 11 | CONCLUSION |
| 12 | FUTURE SCOPE |
| 13 | APPENDIX Source Code GitHub and Project Demo Link |

1 . INTRODUCTION

1.1 : Project Overview

- ✓ A Web page is designed for the students where they can check their Admission Eligibility for the respected University.
- ✓ This Web page can get some of the coloums for the students marks that will calculate the students eligibility for that university.

1.2 : Purpose

The Purpose of this project is

- ✓ To reduce the work load of the user and also the use of paper.
- ✓ To enable the online Eligibility Checking for the Universities.
- ✓ To reduce the work load of the Students.
- ✓ It will Automatically calculate the chance of the students.

2 . LITERATURE SURVEY

2.1 : Existing Problem

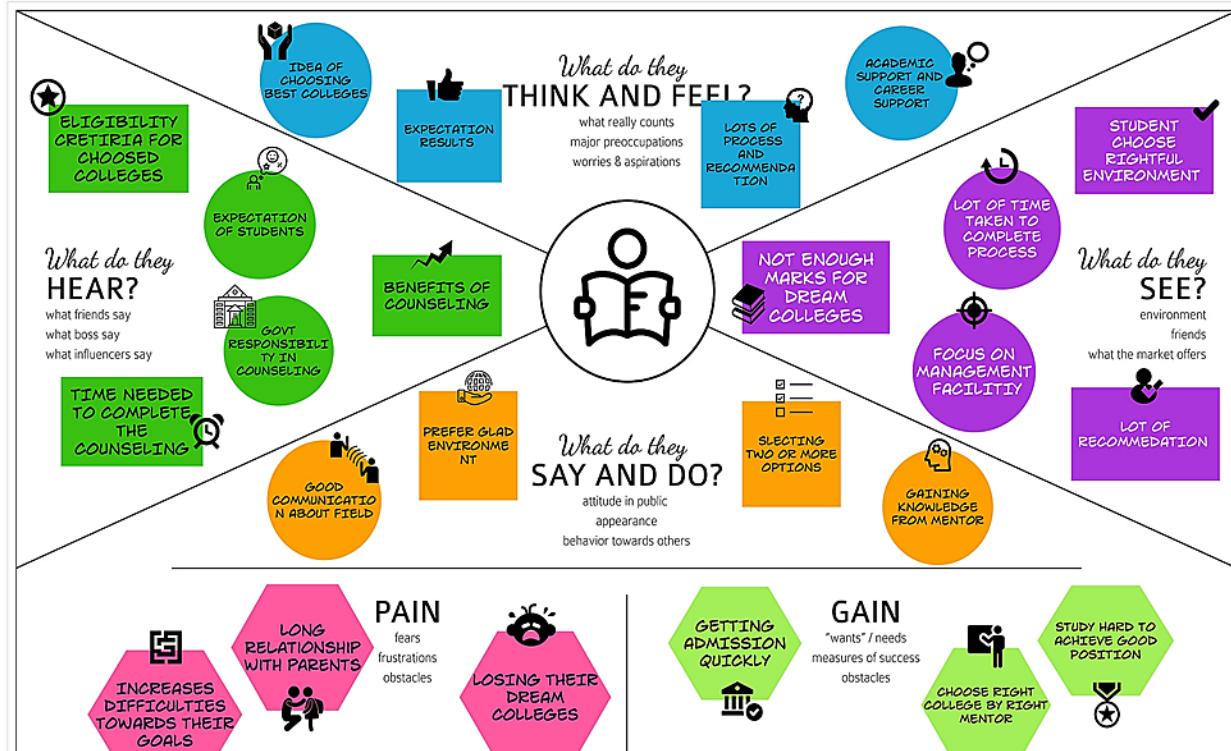
- ✓ Students who need to check their chances of getting admission in the Universities.
- ✓ Students can visit the web site and check their chances.
- ✓ And the Existing Problem can provide the high probability chances for the students who wish to get Admission in the University.
- ✓ This Existing Problem is well user friendly for the students who can visit this site.

2.2 : Problem Statement

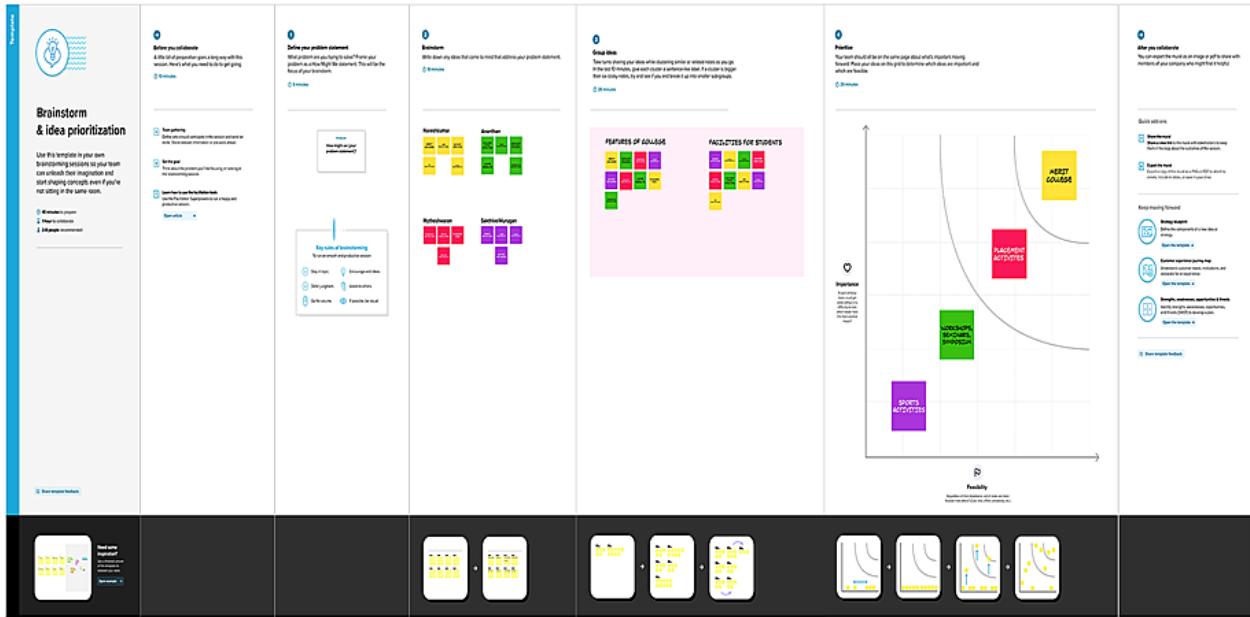
- ✓ Students need to check their Admission Eligibilities in the respective Universities.
- ✓ Students can put their mark details that will calculate and provide the probable chances.
- ✓ Students need this platform to get the idea about the Admissions for the Universities.

3. IDEATION & PROPOSED SOLUTION

3.1 : Empathy Map Canvas



3.2 : Ideation and Brainstorming

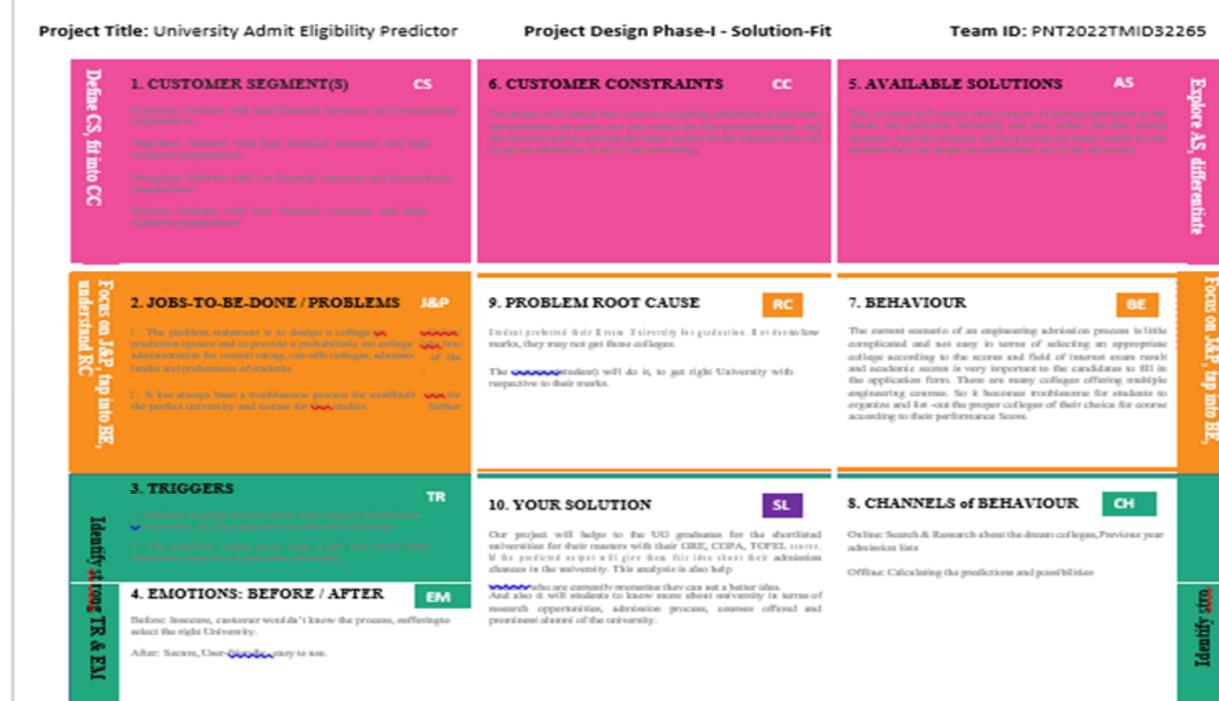


3.3 : Proposed Solution

| S. No. | Parameter | Description |
|--------|--|--|
| 1. | Problem Statement (Problem to be solved) | Students are often worried about their chances of admission to University. The aim of this project is to help students in shortlisting universities with their profiles. The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea to get admitted in a master's program in a university. |
| 2. | Idea / Solution description | Our project will help to the UG graduates for the shortlisted universities for their masters with their GRE, CGPA, TOFEL scores. If the predicted output will give them fair idea about their admission chances in the university. This analysis is also help students who are currently preparing they can get a better idea. And also it will help students to know more about university in terms of research opportunities, admission process, courses offered and prominent alumni of the university. |

| | | |
|----|-------------------------------------|--|
| 3. | Novelty / Uniqueness | The project website can lists various facilities present at the universities and guides to travel to the university where it is located. And also get scholarship opportunities and financial assistance. |
| 4. | Social Impact/Customer Satisfaction | This solution will reduce their concern of getting admission in the dream and particular university and also reduce the fear among students. And this solution will be provide the better results for the Students they can get an admission or not in the university. |
| 5. | Business Model(Revenue Model) | In addition, revenue can be generated by advertising the GRE/TOEFL coaching centers. And the University shall fund the website in Order to maintain and progress it. |
| 6. | Scalability of the Solution | A future update shall have chat space consist of candidate, faculty, current students and alumni. It can scale for universities all around the world. |

3.4 : Problem Solution Fit



4.REQUIREMENT ANALYSIS

4.1 : Functional requirements

Functional Requirements:

Following are the functional requirements of the proposed solution.

| FR No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|--------|-------------------------------|---|
| FR-1 | User Registration | Registration through Form Registration through Gmail Registration through <u>LinkedIN</u> |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | User Login | Login through username and password Login through Gmail Login through <u>LinkedIN</u> |
| FR-4 | Admission Details | Check seat availability Check college infrastructure Check fees details |
| FR-5 | Administration work | Check qualified candidate detail Make allotment |
| FR-6 | Local counsellor | Issue the final allotment order |

4.2: Non-Functional requirements

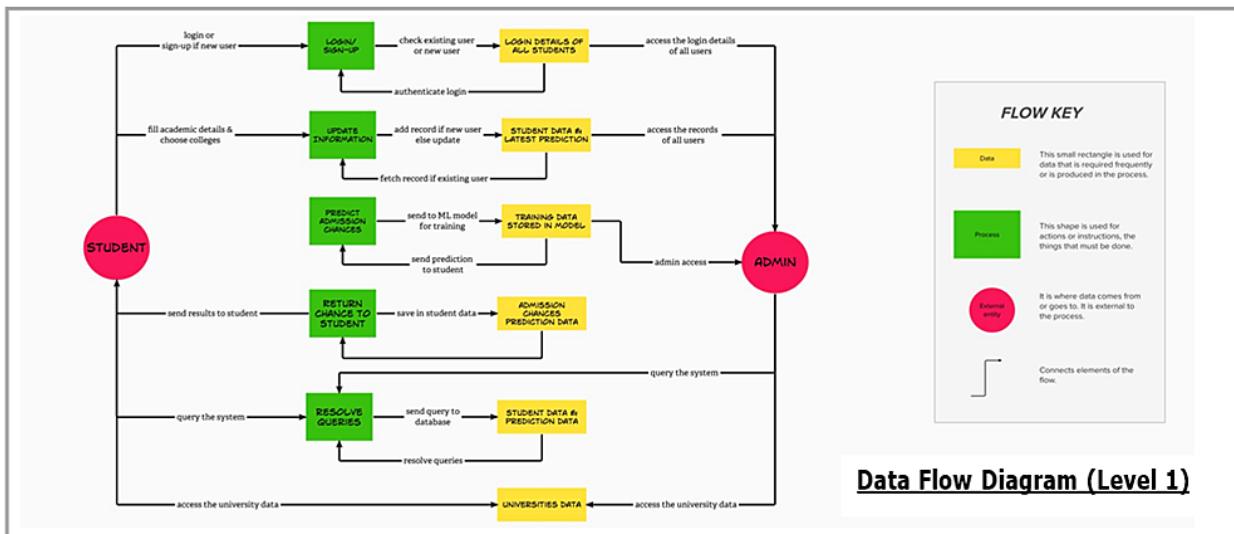
Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

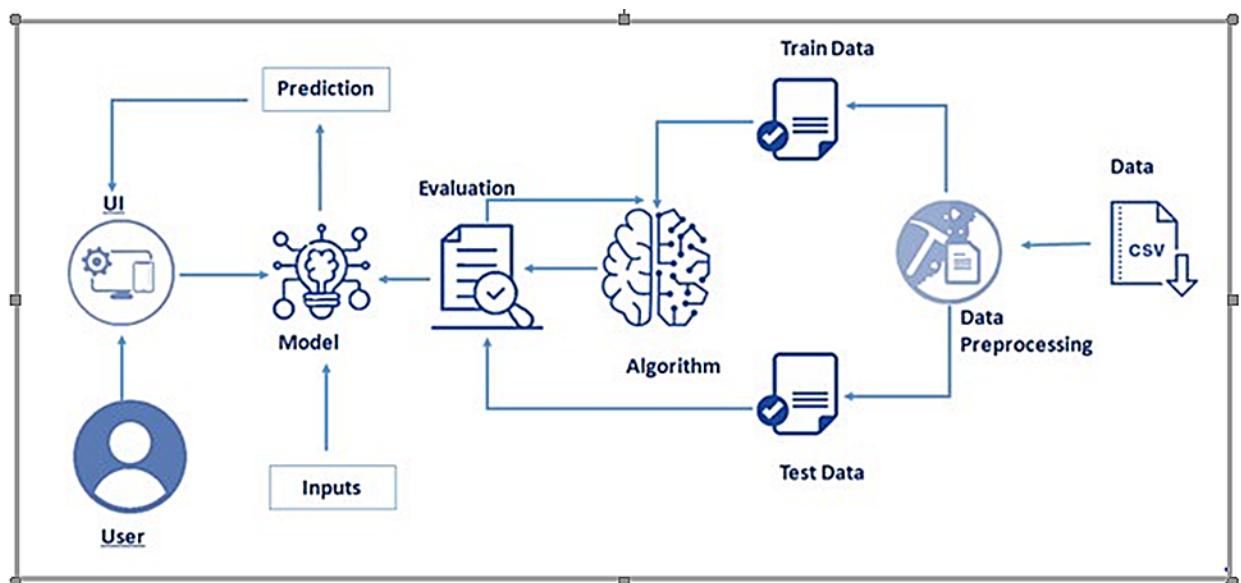
| NFR No. | Non-Functional Requirement | Description |
|---------|----------------------------|--|
| NFR-1 | Usability | i. A logical interface is essential to make easy use of system, speeding up common tasks. ii. The product could be used by two categories of people mainly administrator category and other users. |
| NFR-2 | Security | Some of the factors that are identified to protect the software from accidental or malicious access, use, modification, destruction, or disclosure are described below: i. Keep specific log or history data sets. ii. Utilize certain cryptographic techniques. |

5 . PROJECT DESIGN

5.1 : Data Flow Diagrams



5.2 : Solution & Technical Architecture



5.3 : User Stories

User Stories:

Use the below template to list all the user stories for the product.

| User Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|-----------------------------|-------------------------------|-------------------|---|---|----------|----------|
| Student (Mobile / 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 |
| | Login | USN-2 | As a user, I can log into the application by entering email & password | I can choose colleges | High | Sprint-1 |
| | Dashboard | USN-3 | As a user, I can view the details such as my profile and academic details | I can update my profile | Medium | Sprint-1 |
| | Landing page | USN-4 | As a user, I can view the details and latest news about the university | I can access the university landing page | High | Sprint-2 |
| | Queries | USN-5 | As a user, I can ask queries to the system regarding the help/support or technical issues | I can fill and submit the contact form | Medium | Sprint-2 |
| | Admissions | USN-6 | As a user, I can see the previous year cut-off marks | I can check the records of the university | High | Sprint-2 |
| | | USN-7 | As a user, I can predict my eligibility for admission at the university | I can compare the result as either eligible/ not eligible | High | Sprint-3 |
| | Courses Offered | USN-8 | As a user, I can see the courses offered by the university for PG students | I can access the course details | Medium | Sprint-3 |
| Administrator | Authentication | USN-9 | As a admin, I can authenticate the login credentials of user | I can access all the user details | Medium | Sprint-3 |
| | Dashboard | USN-10 | As a admin, I can verify the details of the user | I can confirm the user updating details | High | Sprint-4 |
| | Chances | USN-11 | As a admin, I can send the prediction details to the users | I can provide chances | Medium | Sprint-4 |
| | Solution | USN-12 | As a admin, I can solve the queries of users | I can solve the queries | Low | Sprint-4 |
| | Updating | USN-13 | As a admin, I can update the university database depends on the user confirmation | I can update database | High | Sprint-4 |

6 . PROJECT PLANNING & SCHEDULING

6.1 : Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation(4 Marks):

| 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. | 5 | High | Ananthan.A Nareshkumar.V |
| Sprint-1 | | USN-2 | As a user, I will receive confirmation email once I have registered for the application | 5 | High | Mytheshwaran.G Sakthivel Murugan.S |
| Sprint-2 | | USN-3 | As a user, I can register for the application through Gmail | 6 | Medium | Sakthivel Murugan.S Nareshkumar.V |
| Sprint-1 | Login | USN-4 | As a user, I can log into the application by entering email & password | 5 | High | Nareshkumar.V Mytheshwaran.G |

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story points | Priority | Team Members |
|----------|-------------------------------|-------------------|---|--------------|----------|--|
| Sprint-3 | Selection | USN-5 | As a user, I can confirm the available college or re-apply to other college | 10 | Medium | Mytheshwaran.G Ananthan.A |
| Sprint-4 | Queries | USN-6 | As a user, I can ask queries to the system regarding the help / support or technical issues | 10 | High | Ananthan.A Nareshkumar.V |
| Sprint-1 | Authentication | USN-7 | As a admin, I can authenticate the login credentials of user | 5 | High | Sakthivel Murugan.S Ananthan.A |
| Sprint-2 | Dashboard | USN-8 | As a admin , I can verify the details of the user | 7 | High | Nareshkumar.V Sakthivel Murugan.S |
| Sprint-2 | Prediction | USN-9 | As a admin , I can train the user details with ML algorithm | 7 | High | Ananthan.A Mytheshwaran.G |
| Sprint-3 | Chances | USN-10 | As a admin ,I can solve the queries of users | 10 | High | Nareshkumar.V Ananthan.A Sakthivel Murugan.S Mytheshwaran.G |
| Sprint-4 | Solution | USN-11 | As a admin, I can update the university database depends on the user confirmation | 10 | High | Sakthivel Murugan.S Ananthan.A Nareshkumar.V Mytheshwaran.G |

6.2 : Sprint Delivery Schedule

Project Tracker, Velocity & Burn down Chart: (4 Marks)

| Sprint | Total Story Points | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (as on Planned End Date) | Sprint Release Date (Actual) |
|----------|--------------------|----------|-------------------|---------------------------|---|------------------------------|
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 20 | 29 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 20 | 07 Oct 2022 |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 20 | 10 Oct 2022 |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 20 | 15 Oct 2022 |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

6.3 : Reports From JIRA

The screenshot shows the Jira Software interface for the 'University Admit Eligibility Predictor' project. The left sidebar is collapsed, showing the project name and navigation options like Backlog, Board, and Project pages. The main area displays the 'Backlog' section with four sprint items:

- UAEP Sprint 1** (24 Oct – 29 Oct) - 4 issues: Status 0/20, Buttons: Complete sprint, Start sprint, More.
- UAEP Sprint 2** (31 Oct – 5 Nov) - 3 issues: Status 24/0, Buttons: Start sprint, More.
- UAEP Sprint 3** (7 Nov – 12 Nov) - 2 issues: Status 16/0, Buttons: Start sprint, More.
- UAEP Sprint 4** (14 Nov – 19 Nov) - 2 issues: Status 20/0, Buttons: Start sprint, More.

Below the sprints, there is a section for the **Backlog** (0 issues) with a note: "Your backlog is empty." A button "+ Create issue" is available at the bottom of this section.

Projects / University Admit Eligibility Predictor

UAEP Sprint 1

TO DO IN PROGRESS 4 ISSUES IN REVIEW DONE +

As a user, I can register for the application by entering my email, password, and confirming my password.

REGISTRATION

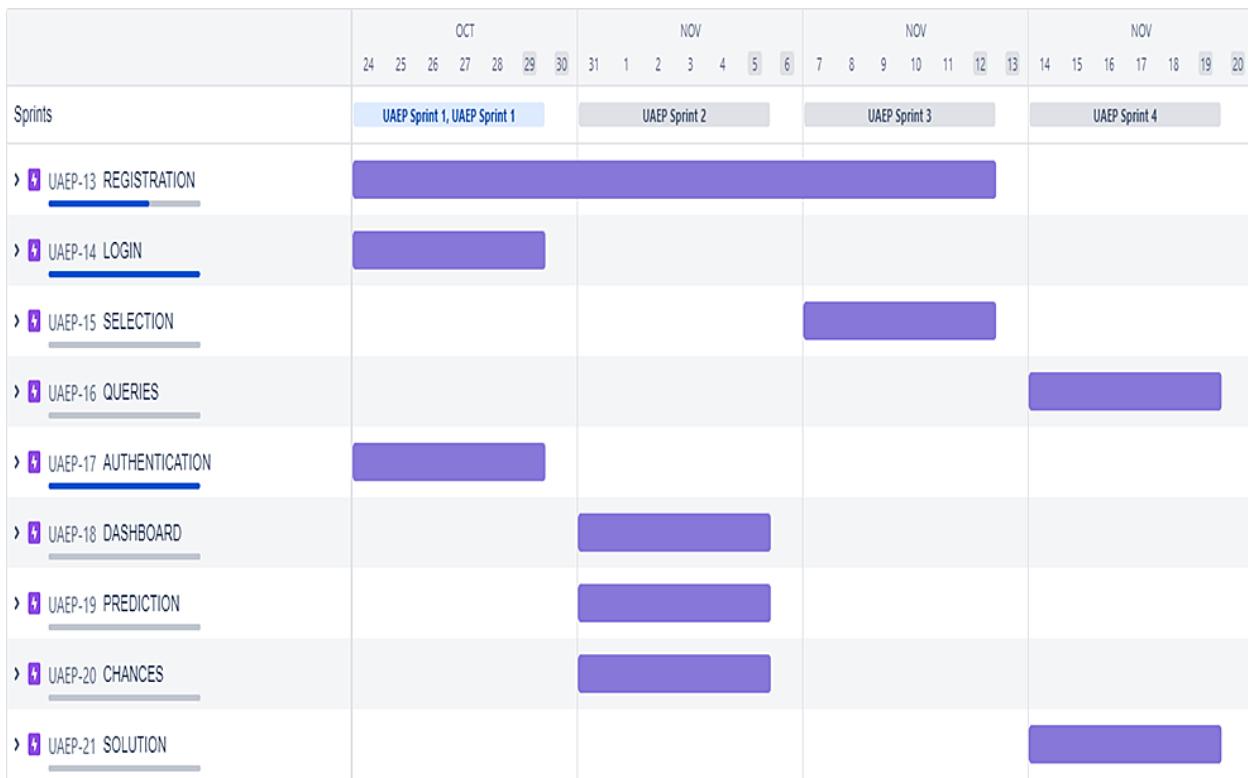
UAEPI-1

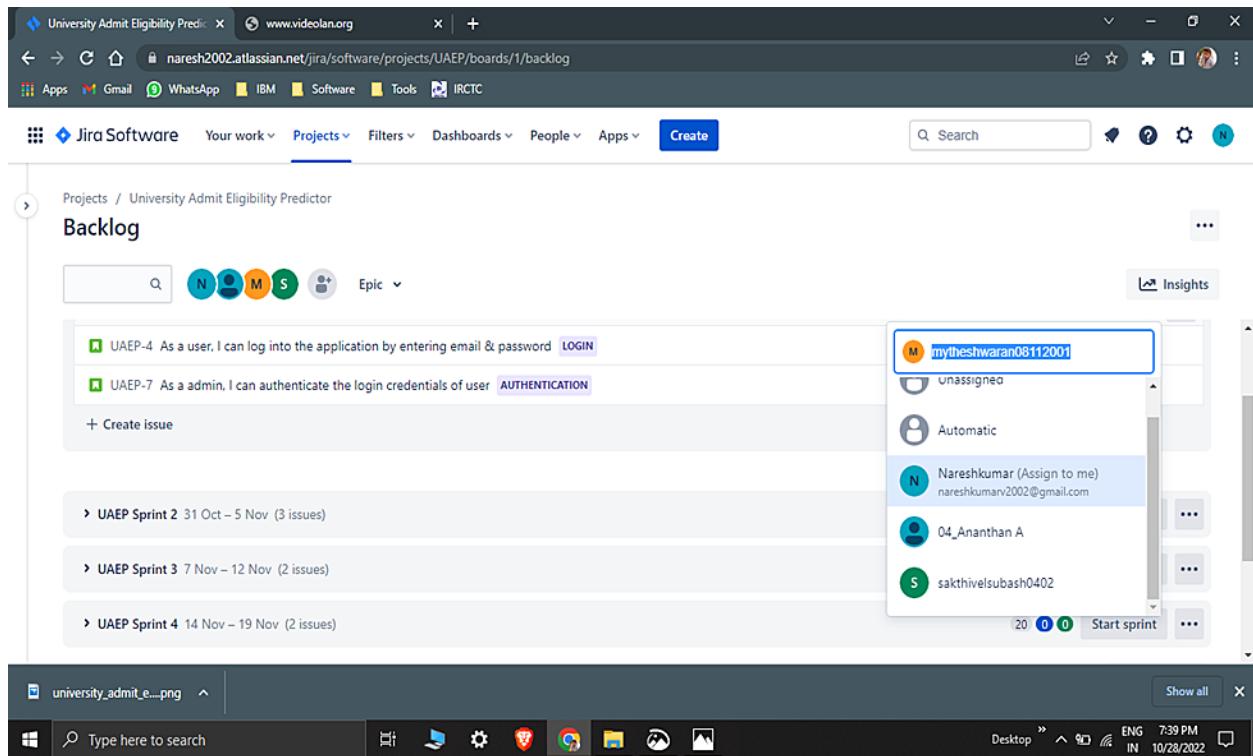
As a user, I will receive confirmation email once I have registered for the application.

REGISTRATION

UAEPI-2

GROUP BY None Insights





7 . CODING & SOLUTIONING

7.1 : Feature 1

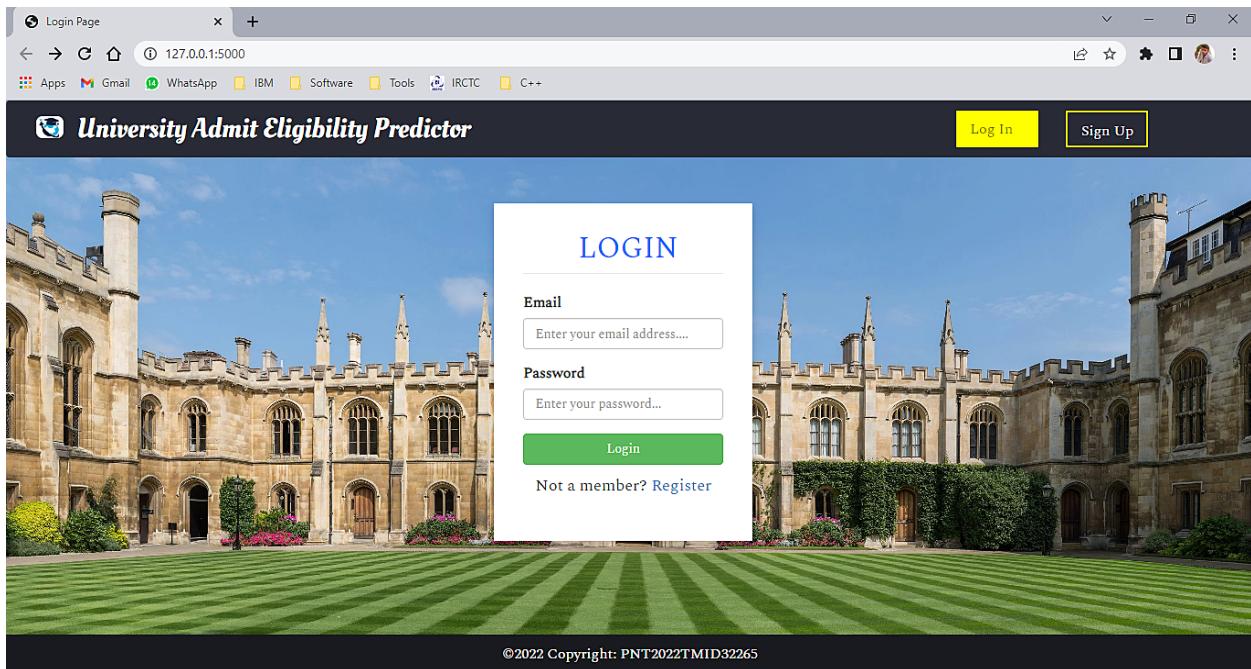
UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

Description :

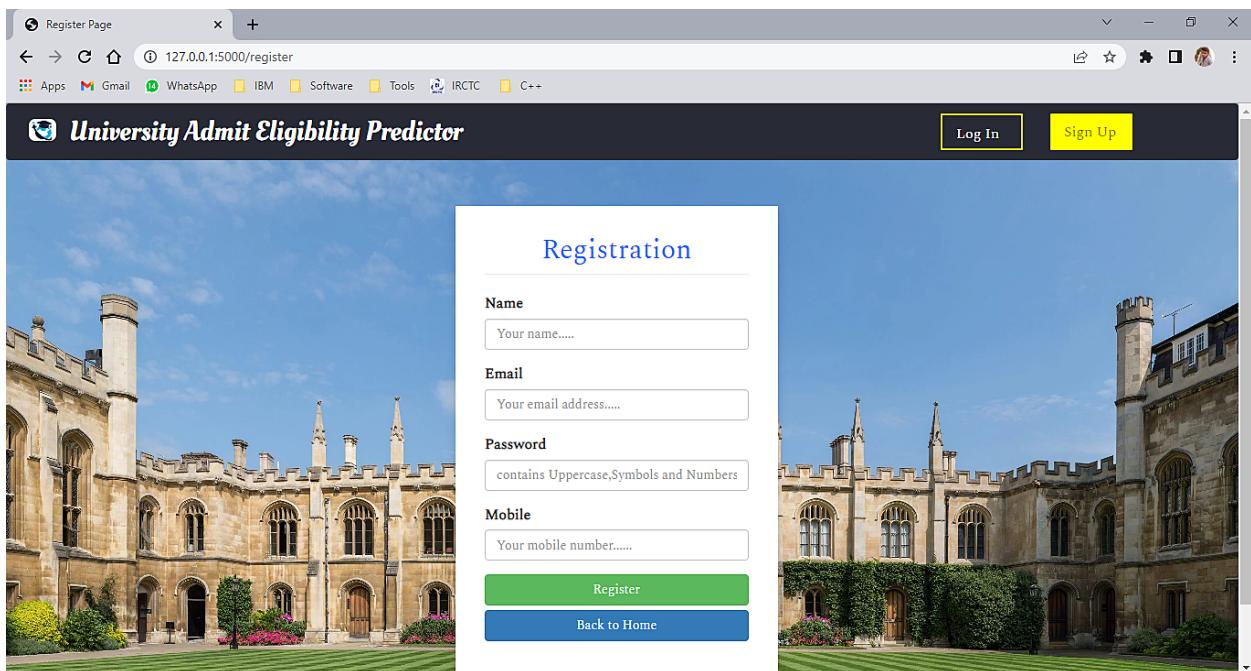
In feature 1 we have designed a webpage using flask and vscode to check the admission prediction .The user can login in into the webpage using username and password . After successful login, the user wil be redirected to the check eligibility form. In this form ,users are asked to check their admission prediction

HOME PAGE:

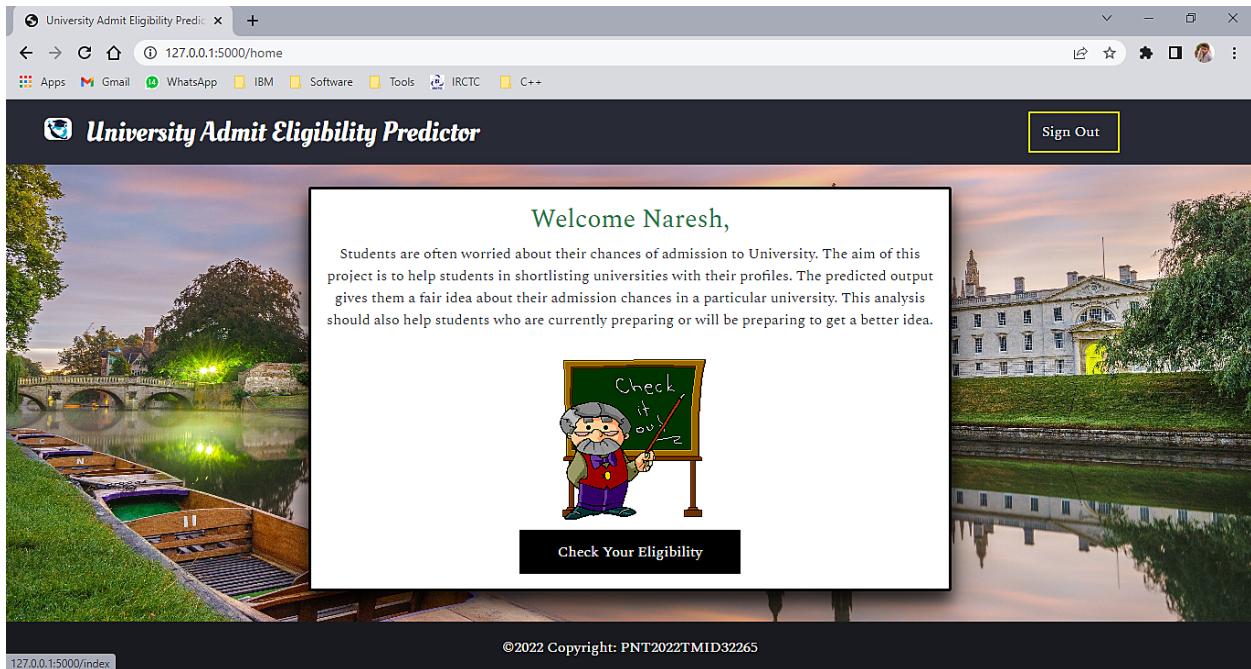
login page :



signup page:



check_eligibility_page:



7.2 : Feature 2

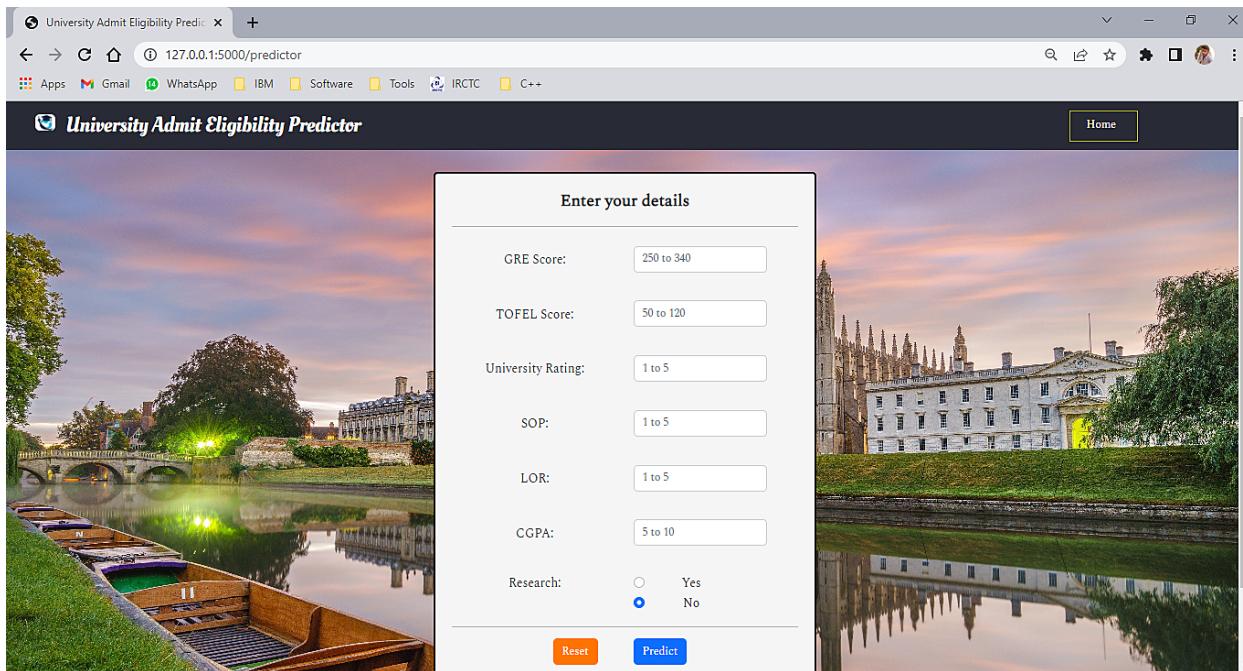
CHECK THE ELIGIBILITY PREDICTION

Description :

In this we have include the predictor page that colud be asked the users to fill their marks in the respected tables and calculate their prediction for the Universities.

If the user have a chance they can be redirected to the respective pages are the chance and nochance pages in this web pages.

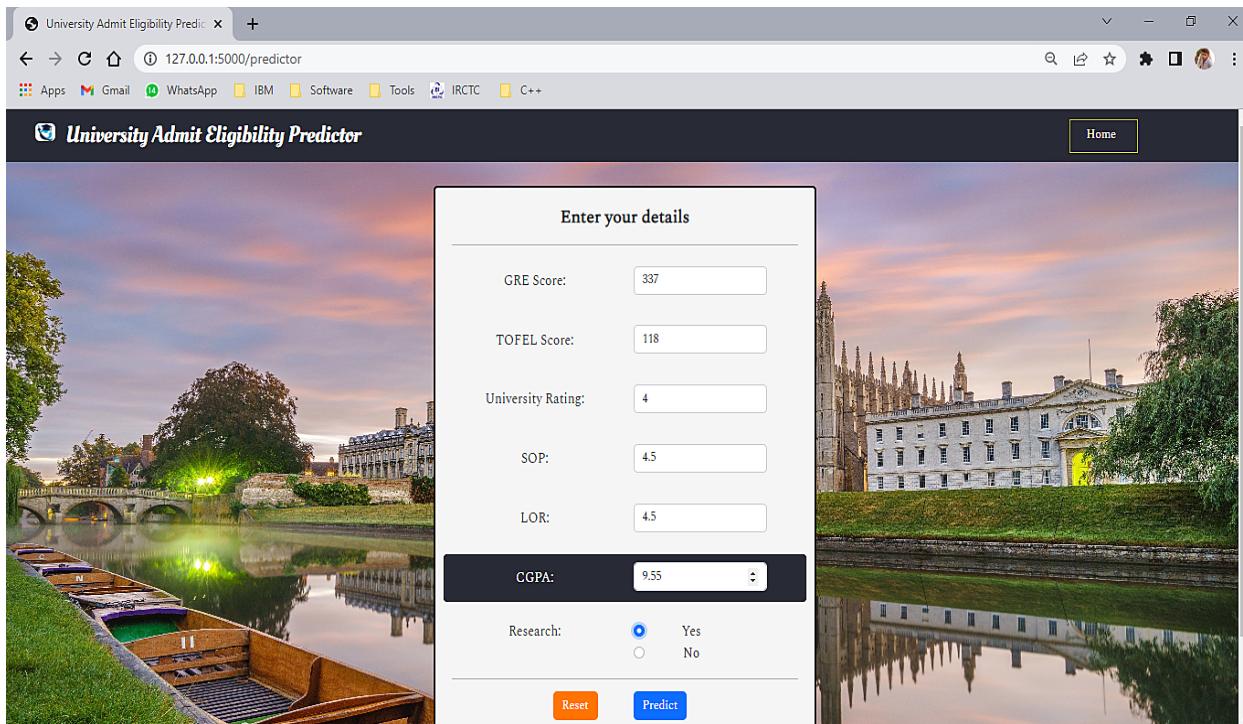
Prediction page:



The screenshot shows the prediction page of the "University Admit Eligibility Predictor". The page has a dark header with the title "University Admit Eligibility Predictor" and a "Home" button. The main content area contains a form titled "Enter your details" with the following fields:

| | |
|--------------------|--|
| GRE Score: | 250 to 340 |
| TOFEL Score: | 50 to 120 |
| University Rating: | 1 to 5 |
| SOP: | 1 to 5 |
| LOR: | 1 to 5 |
| CGPA: | 5 to 10 |
| Research: | <input type="radio"/> Yes <input checked="" type="radio"/> No |

At the bottom of the form are two buttons: "Reset" (orange) and "Predict" (blue).

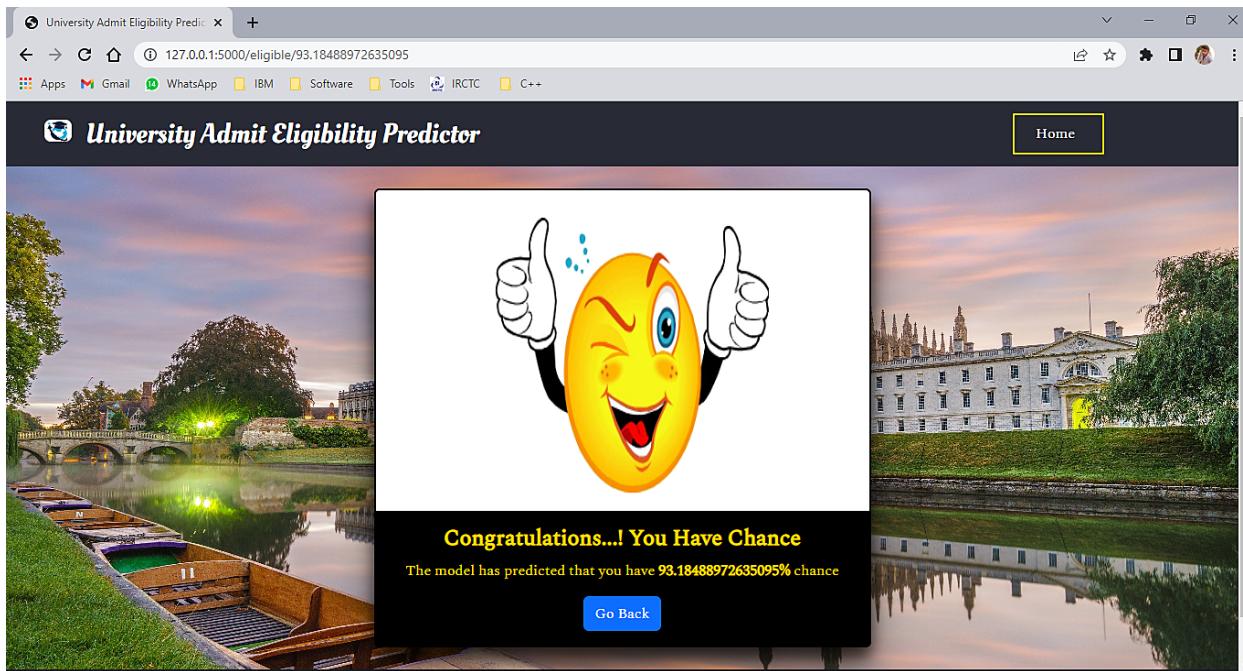


This screenshot shows the prediction page after inputting new values. The form fields now contain:

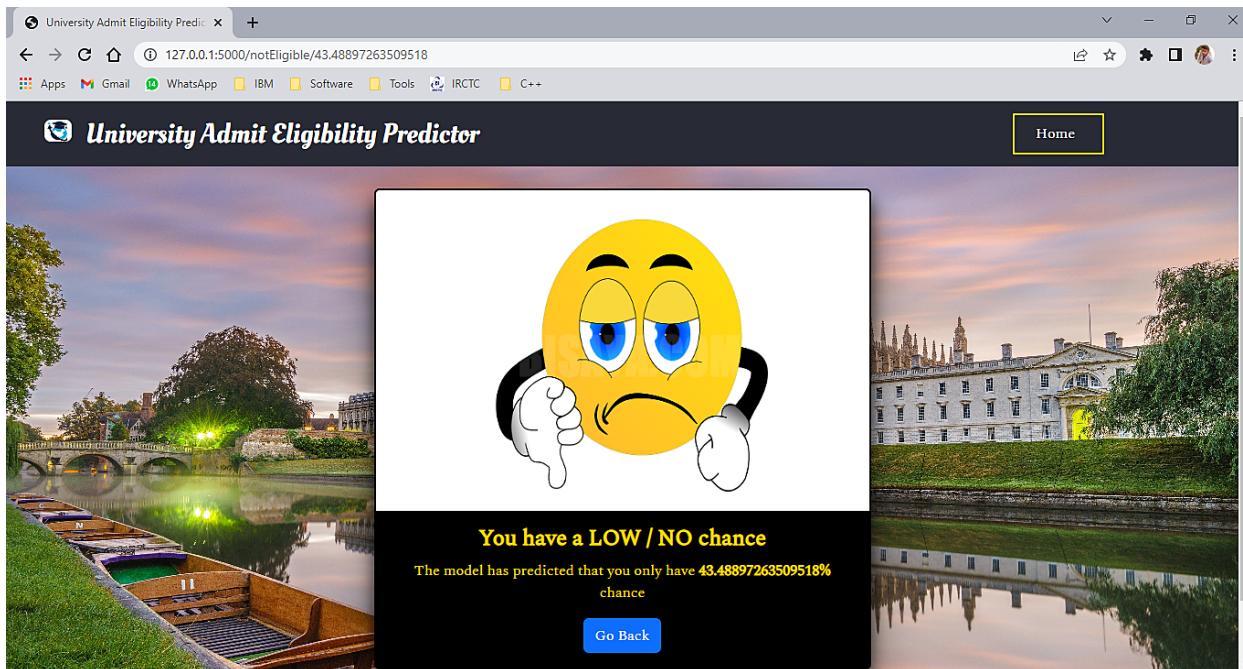
| | |
|--------------------|--|
| GRE Score: | 337 |
| TOFEL Score: | 118 |
| University Rating: | 4 |
| SOP: | 4.5 |
| LOR: | 4.5 |
| CGPA: | 9.55 |
| Research: | <input checked="" type="radio"/> Yes <input type="radio"/> No |

The "Predict" button is visible at the bottom of the form.

Chance page:



NoChance page:



APP.PY

```
from flask import Flask, render_template,flash,redirect,url_for,session,request
import sqlite3
import requests

app = Flask(__name__)
app.secret_key="123"

con=sqlite3.connect("database.db")
con.execute("create table if not exists students(pid integer primary key,name text,email text,mobile integer,password text)")
con.close()

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

@app.route('/login',methods=["GET","POST"])
def index1():
    if request.method=='POST':
        email=request.form['email']
        password=request.form['password']
        con=sqlite3.connect("database.db")
        con.row_factory=sqlite3.Row
        cur=con.cursor()
        cur.execute("select * from students where email=? and password=?",(email,password))
        data=cur.fetchone()
        if data:
            session["name"]=data["name"]
```

```

session["email"]=data["email"]
session["password"]=data["password"]
return redirect(url_for("home"))

else:
    flash("Username and Password Mismatch","danger")
return redirect(url_for("login"))

@app.route('/home',methods=["GET","POST"])
def home():
    return render_template("home.html")

@app.route('/register',methods=['GET','POST'])
def register():
    if request.method=='POST':
        try:
            name=request.form['name']
            email=request.form['email']
            mobile=request.form['mobile']
            password=request.form['password']
            con=sqlite3.connect("database.db")
            cur=con.cursor()
            cur.execute("insert      into
students(name,email,mobile,password)values(?, ?, ?, ?)",(name,email,mobile,password))
            con.commit()
            flash("Record Added Successfully","success")
        except:
            flash("Error in Insert Operation","danger")
    finally:
        return redirect(url_for("login"))
        con.close()

```

```

return render_template('register.html')

@app.route('/logout')
def logout():
    session.clear()
    return redirect(url_for("login"))

@app.route("/index", 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("predictor"))
            arr.append(float(val))

    # deepcode ignore HardcodedNonCryptoSecret: <please specify a reason of ignoring
    this>

API_KEY = "wf8mge_OQdwVO8ao2kmWCtxOfLWl8442SH44V85v2Ls"
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}
payload_scoring = {
    "input_data": [{"fields": [ 'GRE Score',
        'TOEFL Score',
        'University Rating',
        'SOP',
        'LOR ',
        'CGPA',
        'Research']}]}

app.route('/predict', methods = ['POST'])
def predict():
    json_input = request.json
    payload_scoring["input_data"] = json_input["input_data"]
    response = requests.post('https://us-central.ml.cloud.ibm.com/ml/v1/predictions', headers=header, json=payload_scoring)
    return response.json()

```

```

        '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('eligible', percent=result[0][0]*100))
else:
    return redirect(url_for('notEligible', percent=result[0][0]*100))
else:
    return redirect(url_for("predictor"))

@app.route("/predictor")
def predictor():
    return render_template("predictor.html")

@app.route("/eligible/<percent>")
def eligible(percent):
    return render_template("eligible.html", content=[percent])

@app.route("/notEligible/<percent>")
def notEligible(percent):

```

```

return render_template("notEligible.html", content=[percent])

@app.route('/<path:path>')

def catch_all():
    return redirect(url_for("predictor"))

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

```

7.3 : Database Schema

DB Browser for SQLite - C:\Users\Nareshkumar V\Documents\IBM\Docs\Final Deliverable\database.db

The screenshot shows the DB Browser for SQLite interface with the following details:

- Toolbar:** File, Edit, View, Tools, Help.
- Menu Bar:** New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project, Attach Database, Close Database.
- Table Selection:** Database Structure, Browse Data, Edit Pragmas, Execute SQL.
- Table Name:** students
- Table Headers:** pid, name, email, mobile, password.
- Data Rows:**

| pid | name | email | mobile | password |
|-----|--------------|--------------------------------|------------|--------------|
| 1 | Mytheshwaran | mytheshwaran08112001@gmail.com | 9876543210 | admin123@ |
| 2 | Admin | admin@mail.com | 1234567890 | admin123@ |
| 3 | Naresh | naresh@gmail.com | 9876543210 | Naresh123!@# |
- Pagination:** 1 - 3 of 3.
- Search:** Go to: 1.

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 | TC for Automation(Y/N) | Bug ID | Executed By |
|--------------|--------------|---------------|---|----------------|---|-------------------------------|-------------------------------|---------------------|--------|---------------------------|------------------------|--------|--------------|
| UAEP_TC_001 | Functional | Login Page | user can put the correct login id and password | Flask , vscode | 1.Click on the name and password box 2.Fill the box with the required data 3.Click the login button | Login.html | login page get display | Working as expected | Pass | Steps are clear to follow | y | null | Nareshkumar |
| UAEP_TC_002 | Functional | Sign Up Page | new user can sign up with their details | Flask , vscode | 1.Click on the user details and fill 2.Fill the box with the required data 3.Click the sign up button | SignUp.html | Sign up page display | Working as expected | Pass | Steps are clear to follow | y | null | Mytheshwaran |
| UAEP_TC_003 | Functional | Home Page | Verify user is able to see the submitted data when user click on predict button | Flask , vscode | 1.Click on the input text box 2.Fill the box with the required data 3.Click the predict button | home.html | home page get display | Working as expected | Pass | Steps are clear to follow | y | null | Nareshkumar |
| UAEP_TC_004 | Functional | Home Page | Verify the UI elements in home page | vscode | 1.Click on the input text box 2.Fill the box with the required data 3.Click the predict button | home.html | retrieve to prediction result | Working as expected | Pass | Steps are clear to follow | y | null | Mytheshwaran |
| UAEP_TC_005 | Functional | chance Page | Verify the UI elements in chance page | Flask , vscode | 1.See your UAEP prediction result 2.Also you can see the back to home link 3.Click the link to back to home | .chance.html | predict the chances | Working as expected | Pass | Steps are clear to follow | y | null | Ananthan |
| UAEP_TC_006 | Functional | chance Page | Verify user is able to go back to home | Flask , vscode | 1.See your UAEP prediction result 2.Also you can see the back to home link 3.Click the link to back to home | .chance.html | retrieve to home page | Working as expected | Pass | Steps are clear to follow | y | null | Ananthan |
| UAEP_TC_007 | Functional | noChance Page | Verify the UI elements in noChance page | Flask , vscode | 1.See your UAEP prediction result 2.Also you can see the back to home link 3.Click the link to back to home | noChance.html | predict the chances | Working as expected | Pass | Steps are clear to follow | y | null | Nareshkumar |
| UAEP_TC_008 | Functional | noChance Page | Verify user is able to go back to home | Flask , vscode | 1.See your UAEP prediction result 2.Also you can see the back to home link 3.Click the link to back to home | noChance.html | retrieve to home page | Working as expected | Pass | Steps are clear to follow | y | null | Mytheshwaran |

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 [ProductName] 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 | 9 | 3 | 1 | 2 | 15 |
| Duplicate | 1 | 0 | 3 | 0 | 4 |
| External | 2 | 3 | 0 | 1 | 6 |
| Fixed | 9 | 2 | 5 | 19 | 35 |
| Not Reproduced | 0 | 0 | 1 | 0 | 1 |
| Skipped | 0 | 1 | 1 | 1 | 3 |
| Won't Fix | 0 | 4 | 2 | 0 | 6 |
| Totals | 21 | 13 | 13 | 23 | 69 |

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 |
|---------------------|-------------|------------|------|------|
| Print Engine | 6 | 0 | 0 | 6 |
| Client Application | 49 | 0 | 0 | 49 |
| Security | 3 | 0 | 0 | 3 |
| Outsource Shipping | 4 | 0 | 0 | 4 |
| Exception Reporting | 8 | 0 | 0 | 8 |
| Final Report Output | 3 | 0 | 0 | 3 |
| Version Control | 2 | 0 | 0 | 2 |

9 . RESULTS

9.1 : Performance Metrics

Software quality is a measurement of something intangible, “how good” a software product really is. Some of the aspects of software quality taken are

- i. Scalability
- ii. Speed
- iii. Stability
- iv. Reliability
- v. Security
- vi. Maintainability and code quality

LOAD TEST

| | |
|---------------------------|---|
| Scenario Name | Load Test – University Admit Eligibility Predictor |
| Scenario Type | Load Test – Duration 1 hour |
| Scenario Objective | To Simulate the peak load and to monitor the performance of the Website |
| Steps | The online load will be maintained at steady state |
| Entry Criteria | All the monitors are in ready state |
| Exit Criteria | Response met the criteria and test completion report is agreed |

STRESS TEST

| | |
|---------------------------|--|
| Scenario Name | Stress Test - University Admit Eligibility Predictor |
| Scenario Type | Stress Test |
| Scenario Objective | Objective is to verify that the application can handle the projected growth and to discover the breaking point |
| Steps | Ramp up to 150% of peak volume and continuously increase load until breaking point |
| Entry Criteria | All the monitors are in place Test Data is set up Peak load test completed successfully |
| Exit Criteria | Test completion report is agreed upon as per expectation |

ENDURANCE / SOAK TEST

| | |
|---------------------------|---|
| Scenario Name | Soak Test – University Admit Eligibility Predictor |
| Scenario Type | Endurance – Duration 8 hours |
| Scenario Objective | To discover memory issues and bottlenecks that might occur under daily usage of the application |
| Steps | Steady state is maintained for 8 hours with half of the peak load |
| Entry Criteria | All the monitors are in place Test Data is set up Peak load test completed successfully |
| Exit Criteria | Test completion report is agreed upon as per expectation |

10 . ADVANTAGES

- ✓ It helps student for making decision for choosing a right college.
- ✓ Here the chance of occurrence of error is less when compared with the existing system.
- ✓ It is fast, efficient and reliable.
- ✓ Very user-friendly and Easy accessibility of data.

DISADVANTAGES

- ✓ Academic Risks.
- ✓ Compliance Risks.
- ✓ Financial and Operational Risks.
- ✓ Reputational and Strategic Risks.

11 . CONCLUSION

- ✓ University Admit Eligibility Predictor is designed to reduce the work load of the user and also the use of paper.
- ✓ A Web page is designed for the users where they can check their admission prediction by seeing the available admission on the universities.
- ✓ After putting the marks in the prediction page users can get the idea about the admissions for the universities.
- ✓ All the details of the users will be stored in the database with a unique ID and they can be retrieved back when the users can login again.

12 . FUTURE SCOPE

- ✓ Future indeed holds for big data science as there is a huge amount of data in the age of the internet, waiting to be utilized to make predictions , decisions and inventions.
- ✓ One of the most significant benefit of the University Admit Eligibility Predictor is that they are always open for business and can accept the predictions.
- ✓ Universities are used this projects to get their work easily and simpler manner to provide the Admissions to the students.

13 . APPANDIX

Source Code :

login.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">

    <title>Login Page</title>

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi"
crossorigin="anonymous">
<link rel="stylesheet" href="{{url_for('static',filename='css/bootstrap.min.css')}}">
```

```
<link rel="stylesheet" href="{{url_for('static',filename='myfont/css/all.min.css')}}">
<script src="{{url_for('static',filename='js/jquery.min.js')}}"></script>
<style>
    @import url('https://fonts.googleapis.com/css2?family=Oleo+Script&family=Spectral&display=swap');

body{
    background-image : url('{{url_for('static',filename='img/home_front.jpg')}}");
    /* background-color: lightgray; */
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    position: relative;
    font-family: 'Spectral', serif;
}
.container-fluid{
    margin-left: -400px;
}
.navbar{
    background-color: #282a35;
}
.navbar-brand span{
    font-family: 'Oleo Script', cursive;
    font-size: 30px;
    color: white;
    margin-left: 10px;
}
.navbar-brand img{
    border-radius: 7px;
    margin-top: -4px;
}
```

```
.navbar-nav{  
    margin-right: 100px;  
}  
.nav-item{  
    margin-left: 15px;  
    margin-right: 15px;  
    font-size: 16px;  
    width: 90px;  
    height: 40px;  
}  
.nav-item a{  
    text-decoration: none;  
    margin-left: 0px;  
    margin-top: -5px;  
    margin-left: 10px;  
}  
.nva{  
    background-color: yellow;  
}  
.nvia{  
    border: yellow solid 2px;  
    color: white;  
}  
.nvia:hover{  
    background-color: yellow;  
    color: black;  
}  
.nl-inactive{  
    color: white;  
}  
.shadow{  
    box-shadow:3px 3px 10px black;
```

```
padding:30px;
background-color: white;
margin-top: -50px;
margin-left: 60px;
font-size: 16px;
border: white solid 2px;
}

.shadow:hover{
    border: black solid 2px;
    box-shadow: 0 12px 16px 0 rgb(255, 255, 255),0 17px 50px 0 rgb(190, 183, 183);
}

.page-header{
    margin-top: 0;
}

.page-header span{
    margin-left: 60px;
    font-size : 32px;
}

.fg-p-a{
    margin-top: 10px;
    font-size: 18px;
}

.foot-span{
    color:white;
    font-size: 16px;
}

</style>

</head>

<body>

<nav class="navbar navbar-expand-lg">
    <div class="container-fluid">
```

```

<a class="navbar-brand text-responsive-h" href="/">
    
        <span>University Admit Eligibility Predictor</span>
    </a>
</div>

<ul class="navbar-nav">
    <li class="nav-item nva">
        <a class="nav-link nl-active" href="/">Log In</a>
    </li>
    <li class="nav-item nvia">
        <a class="nav-link nl-inactive" href="/register">Sign Up</a>
    </li>
</ul>

</nav>

<div class="container" style="margin-top:100px">
    <div class="col-md-offset-1 col-md-10" >
        <div class="col-md-offset-3 col-md-5">
            <div class="shadow">
                <form action="/login" method="POST" autocomplete="off">
                    <h3 class="page-header text-primary"><span>LOGIN</span></h3>

                    {% with messages = get_flashed_messages(with_categories=true) %}

                    {% if messages %}
                        {% for category,message in messages %}
                            <div class="alert alert-{{category}}">{{message}}</div>
                        {% endfor %}
                    {% endif %}
                    {% endwith %}

                    <div class="form-group">

```

```
<label>Email</label>
    <input type="text" name="email" class="form-control"
placeholder="Enter your email address...." required>
</div>
<div class="form-group">
    <label>Password</label>
    <input type="password" name="password" class="form-control"
placeholder="Enter your password..." required>
</div>
<div class="form-group">
    <input type="submit" value="Login" class="btn btn-success btn-block">
    <div class="fg-p-a">
        <center>
            <p>Not a member? <a href="/register">Register</a></p>
        </center>
    </div>
</div>
</form>
</div>
</div>
</div>
<footer class="text-center text-lg-start fixed-bottom" style="margin-bottom: 0px;">
    <div class="text-center p-3" style="background-color: #191a1f">
        <center>
            <span class="foot-span">©2022 Copyright: PNT2022TMID32265</span>
        </center>
    </div>
</footer>
</body>
</html>
<script>
```

```
$(document).ready(function(){
    $(".alert").hide(3000)
});
</script>
```

Signup.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">

    <title>Register Page</title>

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1WTRi" crossorigin="anonymous">
    <link rel="stylesheet" href="{{url_for('static',filename='css/bootstrap.min.css')}}">
    <link rel="stylesheet" href="{{url_for('static',filename='myfont/css/all.min.css')}}">
    <script src="{{url_for('static',filename='js/jquery.min.js')}}"></script>

    <style>
        @import
url('https://fonts.googleapis.com/css2?family=Oleo+Script&family=Spectral&display=swa
p');
```

body{

```
background-image : url("{{url_for('static',filename='img/home_front.jpg')}}");
background-position: center;
background-repeat: no-repeat;
background-size: cover;
position: relative;
font-family: 'Spectral', serif;
}

.container-fluid{
    margin-left: -400px;
}

.navbar{
    background-color: #282a35;
}

.navbar-brand span{
    font-family: 'Oleo Script', cursive;
    font-size: 30px;
    color: white;
    margin-left: 10px;
}

.navbar-brand img{
    border-radius: 7px;
    margin-top: -4px;
}

.navbar-nav{
    margin-right: 100px;
}

.nav-item{
    margin-left: 15px;
    margin-right: 15px;
    font-size: 16px;
    width: 90px;
    height: 40px;
}
```

```
}

.nav-item a{
    text-decoration: none;
    margin-left: 0px;
    margin-top: -5px;
    margin-left: 10px;
}

.nva{
    background-color: yellow;
}

.nvia{
    border: yellow solid 2px;
    color: white;
}

.nvia:hover{
    background-color: yellow;
    color: black;
}

.nl-inactive{
    color: white;
}

.shadow{
    box-shadow:3px 3px 10px black;
    padding:30px;
    background-color: white;
    margin-top: -50px;
    margin-left: 80px;
    font-size: 16px;
    border: white solid 2px;
    margin-bottom: 20px;
}

.shadow:hover{
```

```

    border: black solid 2px;
    box-shadow: 0 12px 16px 0 rgba(0, 0, 0, 0.685),0 17px 50px 0 rgba(0, 0, 0, 0.486);
}

.page-header{
    margin-top: 0;
}

.page-header span{
    margin-left: 10px;
    font-size : 32px;
}

.foot-span{
    color:white;
    font-size: 16px;
}


```

</style>

</head>

<body>

```

<nav class="navbar navbar-expand-lg">
    <div class="container-fluid">
        <a class="navbar-brand text-responsive-h" href="/">
            
            <span>University Admit Eligibility Predictor</span>
        </a>
    </div>
    <ul class="navbar-nav">
        <li class="nav-item nvia">
            <a class="nav-link nl-inactive" href="/">Log In</a>
        </li>
        <li class="nav-item nva">
            <a class="nav-link nl-active" href="/register">Sign Up</a>
        
```

```

</li>
</ul>
</nav>

<div class="container" style="margin-top:100px">
<div class="col-md-offset-3 col-md-5">
<div class="shadow">
<form method="POST" action="/register">
<h2 class="page-header text-primary text-center">Registration</h2>
<div class="form-group">
<label>Name</label>
<input type="text" class="form-control" name="name" placeholder="Your
name....." required>
</div>
<div class="form-group">
<label>Email</label>
<input type="text" class="form-control" name="email" placeholder="Your
email address....." required>
</div>
<div class="form-group">
<label>Password</label>
<input type="text" class="form-control" name="password"
placeholder="contains Uppercase,Symbols and Numbers....." required>
</div>
<div class="form-group">
<label>Mobile</label>
<input type="text" class="form-control" name="mobile"
placeholder="Your mobile number....." required>
</div>
<div class="form-group">
<input type="submit" class="btn btn-success btn-block" value="Register">
<a href="{{url_for('login')}}" class="btn btn-primary btn-block">Back to

```

```

Home</a>
    </div>
    </form>
    </div>
    </div>
</div>

<footer class="text-center text-lg-start" style="margin-bottom: 0px;">
    <div class="text-center p-3" style="background-color: #191a1f">
        <center>
            <span class="foot-span">©2022 Copyright: PNT2022TMID32265</span>
        </center>
    </div>
</footer>
</body>
</html>
<script>
    $(document).ready(function(){
        $("form").attr("autocomplete","off");
    });
</script>

```

Home.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 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 Admit Eligibility Predictor</title>
```

```
<style>
    @import
url('https://fonts.googleapis.com/css2?family=Oleo+Script&family=Spectral&display=swa
p');
```

```
body{
    background-image : url("{{url_for('static',filename='img/check_front.jpg')}}");
    background-position: center;
    background-repeat: no-repeat;
    background-size: cover;
    position: relative;
    font-family: 'Spectral', serif;
}
.navbar{
    background-color: #282a35;
}
.navbar-brand{
    margin-left: 30px;
}
.navbar-brand span{
    font-family: 'Oleo Script', cursive;
    font-size: 30px;
    color: white;
    margin-left: 10px;
}
.navbar-brand img{
```

```
border-radius: 7px;  
margin-top: -4px;  
}  
  
.navbar-nav{  
margin-right: 100px;  
width: 150px;  
height: 45px;  
}  
  
.nav-item{  
margin-left: 15px;  
margin-right: 15px;  
font-size: 16px;  
width: 100px;  
}  
  
.nav-item a{  
text-decoration: none;  
color: white;  
margin-left: 5px;  
}  
  
.nva{  
border: yellow solid 2px;  
color: white;  
}  
  
.nva:hover{  
border: yellow solid 2px;  
background-color: yellow;  
color: black;  
}  
  
.card-home{  
border: white solid 3px;  
border-radius: 3px;  
}
```

```

.card-home:hover{
    border: black solid 3px;
    box-shadow: 0 12px 16px 0 rgba(0, 0, 0, 0.685),0 17px 50px 0 rgba(0, 0, 0, 0.486);
}

.btn-home{
    background-color: white;
    padding: 10px 40px;
    border: black solid 2px;
    border-radius: 0;
    margin-top: 10px;
    color: black;
}

.btn-home:hover{
    background-color: black;
    border: black solid 2px;
    color: white;
}

.foot-span{
    color:white;
    font-size: 16px;
}


```

</style>

</head>

<body>

```

<nav class="navbar navbar-expand-lg">
    <div class="container-fluid">
        <a class="navbar-brand text-responsive-h" href="/">
            
            <span>University Admit Eligibility Predictor</span>
        </a>
    
```

```

<ul class="navbar-nav">
  <li class="nav-item nva">
    <a class="nav-link nl-active" href="/">Sign Out</a>
  </li>
</ul>
</div>
</div>
</nav>

<div class="container text-center p-4">
  <div class="d-flex justify-content-center">
    <div class="card card-home" style="width: 44rem;">
      <div class="card-body">
        <h3 class="text-success">Welcome {{session["name"]}}</h3>
        <p class="card-text">Students are often worried about their chances of admission to University. The aim of this project is to help students in shortlisting universities with their profiles. The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea.</p>
        <center>
          
        </center>
        <a href="/index" class="btn btn-primary btn-home" >Check Your Eligibility</a>
      </div>
    </div>
  </div>
</div>

</body>
<footer class="text-center text-lg-start fixed-bottom" style="margin-bottom: 0px;">
  <div class="text-center p-3" style="background-color: #191a1f">
    <center>

```

```

<span class="foot-span">©2022 Copyright: PNT2022TMID32265</span>
</center>
</div>
</footer>
</html>

```

Predictor.html :

```

{% extends 'index.html' %}

{% block body %}

<div class="p-4">
<div class="row row-sec mb-3">
<div class="col-4">
<div class="card card-sec p-2 ms-2 my-2">
<div class="card-body card-pred"><center>
<h5 class="card-title pb-4">
    Enter your details
</h5>
<hr class="hor-line">
<form action="/index" method="post" id="theForm">
<div class="row mb-3">
    <label for="gre" class="col-lg-6 col-form-label">GRE Score:</label>
    <div class="col-lg-5">
        <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-6 col-form-label">TOFEL Score:</label>
    <div class="col-lg-5">
        <input type="number" class="form-control" id="tofel" name="tofel"
min="50" max="120" placeholder="50 to 120" required>
    </div>
</div>

```

```
</div>
</div>
<div class="row mb-3">
    <label for="university_rating" class="col-lg-6 col-form-label">University
Rating:</label>
    <div class="col-lg-5">
        <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-6 col-form-label">SOP:</label>
    <div class="col-lg-5">
        <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-6 col-form-label">LOR:</label>
    <div class="col-lg-5">
        <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-6 col-form-label">CGPA:</label>
    <div class="col-lg-5">
        <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">
```

```
<label class="col-form-label col-sm-6 pt-0">Research:</label>
<div class="col-sm-5">
    <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>

<hr class="hor-line">
<div class="row row-btn lg-3">
    <div class="col-lg-2 mb-2 me-3">
        <a href="/predictor"> <button type="button" class="btn-res btn btn-primary" >Reset</button></a>
    </div>
    <div class="col-lg-2 mb-2 me-3">
        <button type="submit" class="btn-sub btn btn-primary" id="button">Predict</button>
    </div>
    </div>
    <div class="row row-btn lg-3"><center>
        <div class="col-lg-2 col-load" id="spinner">
```

Chance.html :

```
{% extends 'index.html' %}
```

{% block body %}

![...](../static/img/chance1.png)

Congratulations...! You Have Chance

The model has predicted that you have {{content[0]}}% chance

[Go Back](/home)

```
</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 card-eligible" style="width: 34rem;">
            
            <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 %}
```

GITHUB LINK :

<https://github.com/IBM-EPBL/IBM-Project-13808-1664003647>

PROJECT DEMO LINK :

https://drive.google.com/file/d/1O4jgcJOqf662kxlVhDKAKR37Ke01Yrca/view?usp=share_link