**UNIVERSITY ADMIT ELIGIBILITY PREDICTOR**

**NALAIYA THIRAN PROJECT BASED LEARNING**

**PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY, AND ENTREPRENEURSHIP**

**SUBMITTED BY**

                                                     Logesh K(727619BIT015)

                                                     Thinhesh A C (727619BIT029)

                                                     Karthik S (727619BIT039)

                                                     Aribalan M (727619BIT041)

**Bachelor of Technology**

**in**

**Information Technology**

**Dr. Mahalingam College of Engineering and Technology**

**Pollachi - 642003**

**An Autonomous Institution**

**Affiliated to Anna University, Chennai - 600 025**

**November 2022**

**SYNPOSIS**

1. **INTRODUCTION**

1.1 Project Overview

1.2 Purpose

1. **LITERATURE SURVEY**

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

1. **IDEATION & PROPOSED SOLUTION**

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

1. **REQUIREMENT ANALYSIS**

4.1 Functional requirement

4.2 Non-Functional requirements

1. **PROJECT DESIGN**

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

1. **PROJECT PLANNING & SCHEDULING**

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

1. **CODING & SOLUTIONING**

7.1 Feature 1

7.2 Feature 2

1. **TESTING**

8.1 Test Cases

1. **RESULTS**

9.1 Performance Metrics

1. **ADVANTAGES & DISADVANTAGES**
2. **CONCLUSION**
3. **FUTURE SCOPE**
4. **APPENDIX**

Source Code

GitHub & Project Demo Link

**1.INTRODUCTION**

**1.1 PROJECT OVERVIEW**

Students are often worried about their chances of admission to university. The aim of this project is to help students in shortlisting universities with their profiles. The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea. This project University Admission Predictor System is web based application in which students can register with their personal as well as marks details for prediction the admission in colleges and the administrator can allot the seats for the students. Administrator can add the college details and he batch details. Using this software, the entrance seat allotment became easier and can be implemented using system. 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.

**1.2 PURPOSE**

The primary purpose is to discuss the prediction of student admission to university based on numerous factors and using logistic regression. Many prospective students apply for different courses. The admission decision depends on criteria within the college or degree program. The independent variables in this study will be measured statistically to predict graduate school admission. Exploration and data analysis, if successful, would allow predictive models to allow better prioritization of the applicants screening process to various degree programme which in turn provides the admission to the right candidates.

**2.LITERATURE SURVEY**

**2.1 EXISTING PROBLEM**

Today in colleges student details are entered manually. The student details in separate records are tedious task. Referring to all these records updating is needed. There is a chance for more manual errors.

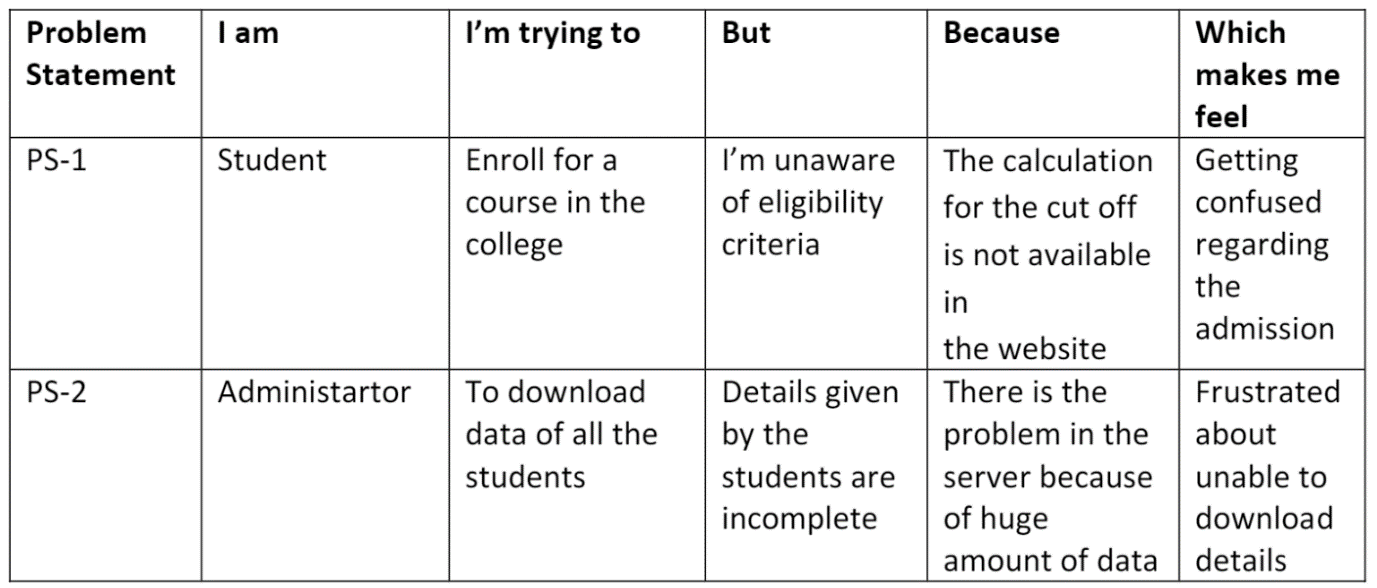
Require much manpower i.e., many efforts, much cost and hard to operate and maintain.

Since, all the work is done in papers, so it is very hard to locate a particular student record when it is required.

**2.2 REFERENCES**

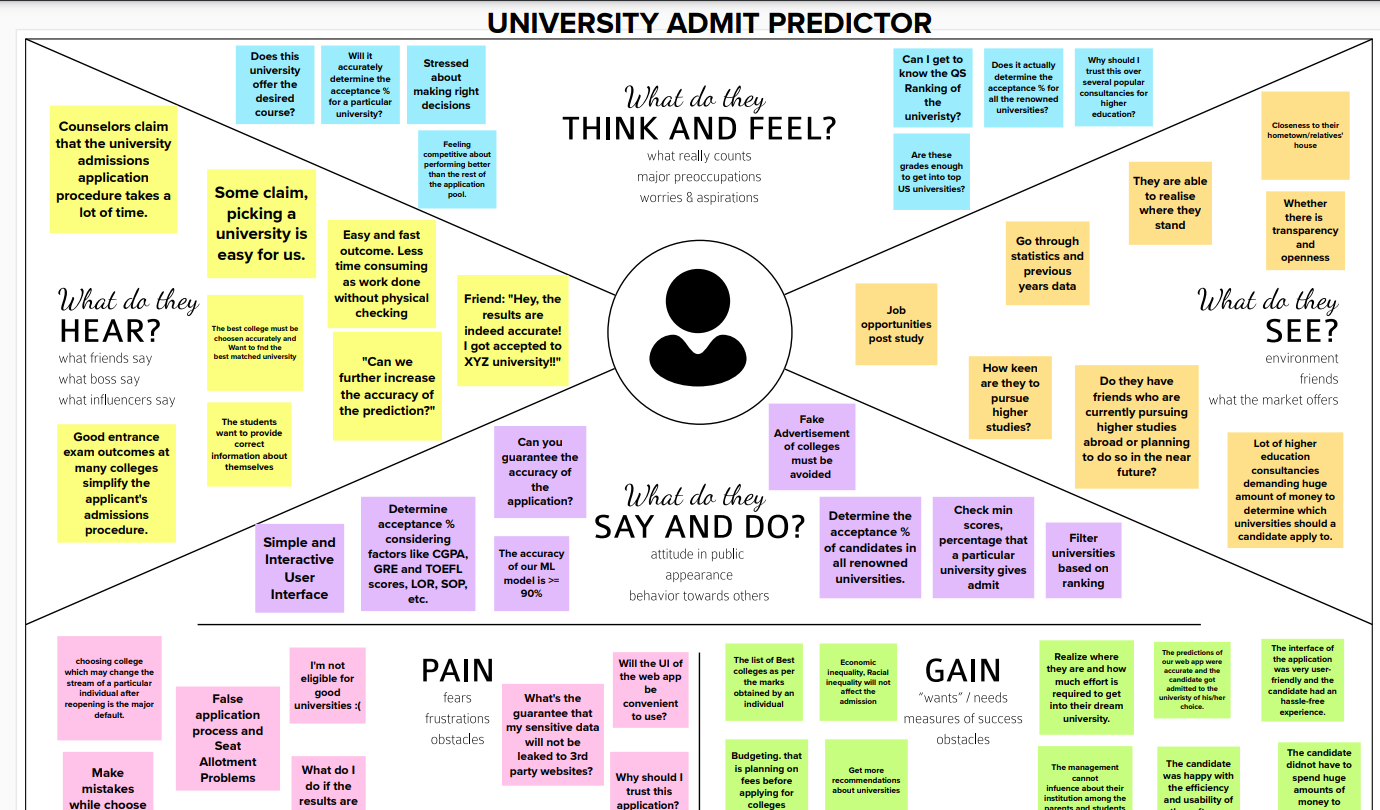
* Abdul Fatah S; M, A. H. (2012). Hybrid Recommender System for Predicting College Admission, pp. 107–113.
* Bibodi, J., Vadodaria, A., Rawat, A. and Patel, J. (n.d.). Admission Prediction System Using Machine Learning.
* Eberle, W., Simpson, E., Talbert, D., Roberts, L. and Pope, A. (n.d.). Using Machine Learning and Predictive Modelling to Assess Admission Policies and Standards.
* Jamison, J. (2017). Applying Machine Learning to Predict Davidson College’ s Admissions Yield, pp. 765–766.
* Mane, R. V. (2016). Predicting Student Admission decisions by Association Rule Mining with Pattern Growth Approach, pp. 202–207.

**2.3 PROBLEM STATEMENT DEFINITION**

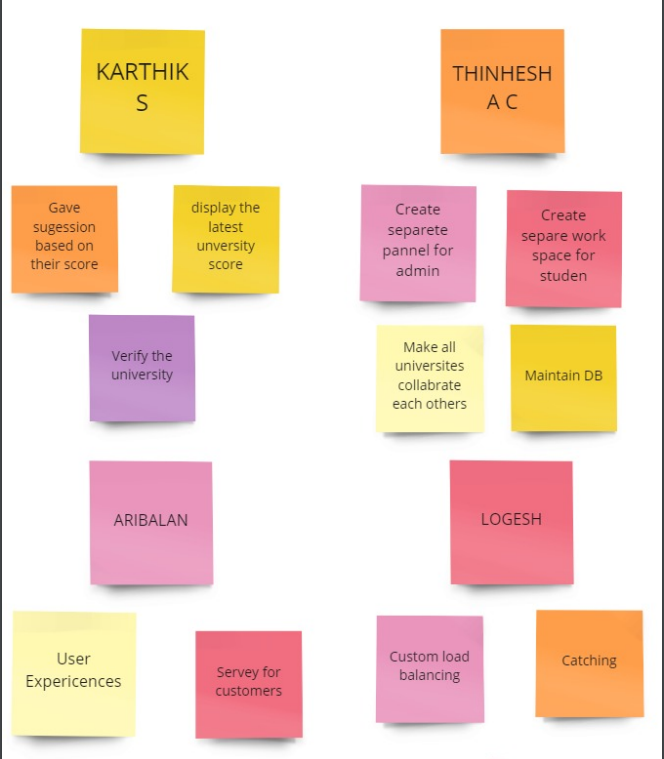
****

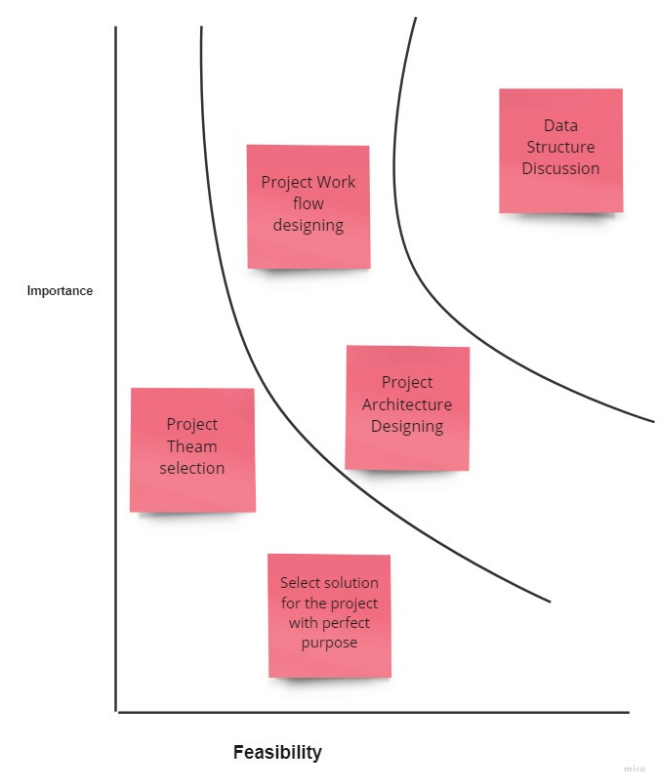
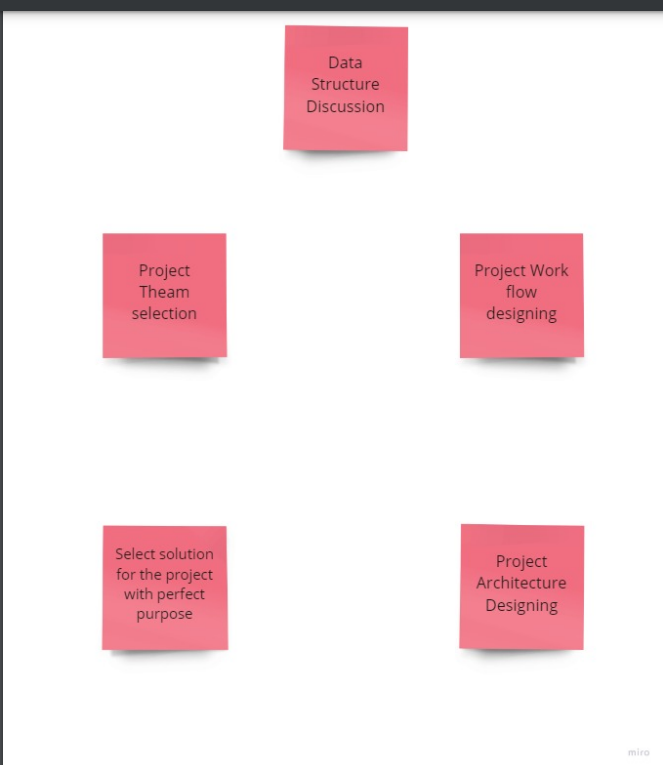
**3.IDEATION AND PROPOSED SOLUTION**

**3.1 EMPATHY MAP CANVAS**

****

**3.2 IDEATION & BRAINSTORMING**

****

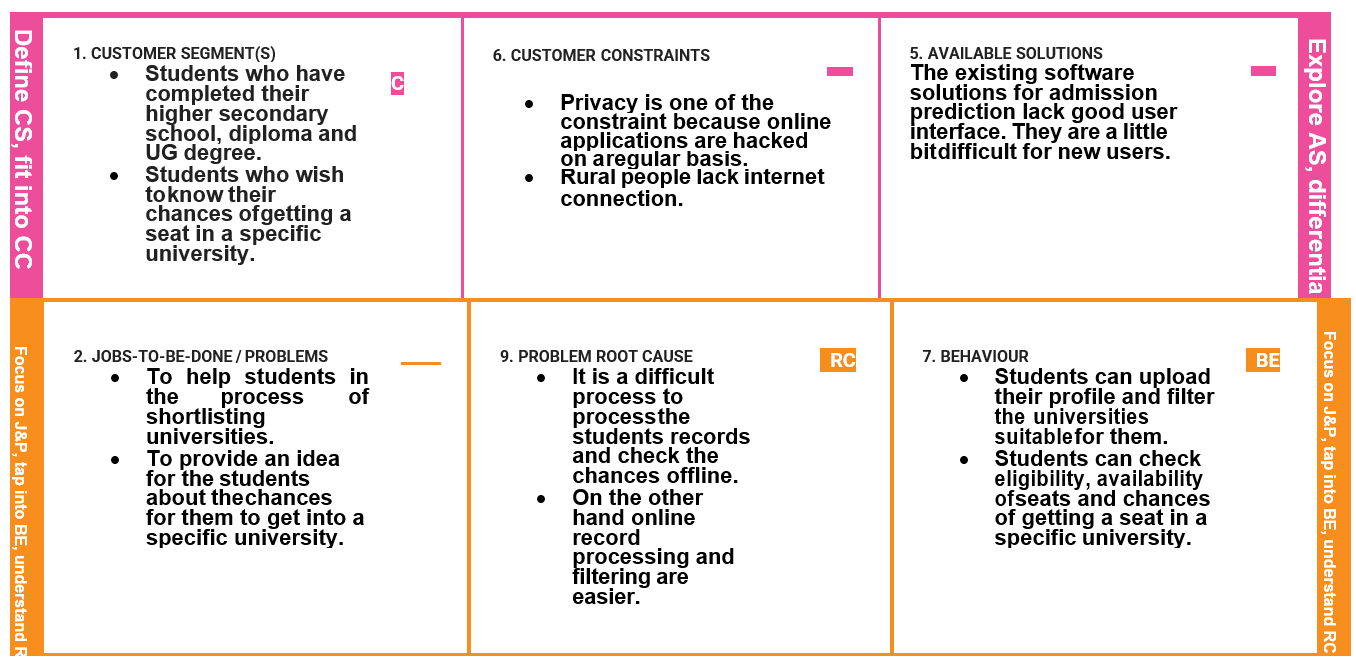
****

**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. It also aims to make a direct connection between the students and the universities and avoid any intermediaries. |
| 2. | Idea / Solution description | This project intends to calculate the probability of acceptance in a particular grad-school after assessing the candidate’s profile.  The key attributes that will be considered for making the decisions are:   1. GRE & TOEFL Scores 2. Undergrad CGPA 3. SOP & LOR 4. Corporate Work Experience / Research Experience 5. Extracurriculars   For determining the % of acceptance, we will be using various ML models such as Logistic Regression, Multiple Linear Regression, Decision Tree & Random Forest and assess which model gives the highest accuracy with the help of performance metrics like accuracy- score, precision and recall. |
| 3. | Novelty / Uniqueness | * We intend to develop a novel deep learning- based hybrid model that has a better accuracy than the existing traditional ML models. |

|  |  |  |
| --- | --- | --- |
|  |  | * The web-app will also provide feedback on the parameters where the candidate is lacking so that he can improve on those areas. |
| 4. | Social Impact / Customer Satisfaction | * Students often feel difficult in shortlisting the universities to apply which they tend to wonder if their profile matches the requirement of a certain university. * Moreover, the cost of applying to a university is extremely high making it critical that students shortlist universities based on their profile. * A university admission prediction system is quite useful for students to determine their chances of acceptance to a specific university. * This system reduces dependence on educational consultancies, who charge loads of money to analyse a candidate’s profile and determine the universities he/she should apply to. |
| 5. | Business Model (Revenue Model) | * Advertisements of different universities could be placed in the web-app to generate revenue through ads. * In the future, a separate premium plan could be created where the students can directly interact with the professors and alumni of the   university through video calls. |
| 6. | Scalability of the Solution | * A future update could have chat space where candidates, faculties, current students of the university and alumni can interact and candidates can get their doubts resolved instantly. * To deal with huge volumes of data in the future (Both - applicants and university details), cloud-based storages (IBM cloud, AWS, GCP, AZURE) and NoSQL databases (MongoDB, Redis, etc.) could be used instead of the traditional RDBMS storage. * Alternatively, distributed big-data processing techniques could be explored if the no. of users using the website increase exponentially during the course of time. |

**3.4 PROBLEM SOLUTION FIT**

****

**Table

Description automatically generated**

**4.REQUIREMENT ANALYSIS**

**4.1 FUNCTIONAL REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **FR NO** | **Functional Requirement** | **Sub- Requirement(Story/Sub-Task)** |
| FR-1 | User Registration | Registration through forms by  providing correct details. |
| FR-2 | User Login | Login by providing username and  Password |
| FR-3 | User Profile | Complete user profile by providing  the Student Academic details. |
| FR-4 | User Data Collection | The following information about students' scores is gathered:  If they are PG applicants, their ,  HSC SSLC  CGPA. |
| FR-5 | Evaluation | Analysing the data entered by the pupils using ML algorithms and putting the ML model that has been produced to the test using the  supplied data. |
| FR-6 | Prediction | The list of universities to which the students are qualified to apply will be shown after the prediction is made based on the findings of the  evaluation. |
| FR-7 | Output | The list of universities to which the students are qualified to apply will be shown after the prediction is made based on the findings of the  evaluation. |

**4.2 NON-FUNCTIONAL REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **NFR No** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | 1. Interactive and powerful progress visualisation 2. Customer Satisfaction 3. Easy to Learn |
| NFR-2 | **Security** | 1. User details are secured from unauthorized parties. 2. When the programme isn't being used, it automatically logs out to prevent unauthorised users from   accessing the user's account. |
| NFR-3 | **Reliability** | The users can find universities based  on their preferred location and results. |
| NFR-4 | **Performance** | The website will provide the list of  universities within 30 seconds. |
| NFR-5 | **Availability** | The system predictor will be accessible to users wherever they are and  whenever they need it. |
| NFR-6 | **Scalability** | It can handle any volume of data and carry out several computations  efficiently and quickly. |

**5. PROJECT DESIGN**

**5.1 DATA FLOW DIAGRAM**

**Diagram

Description automatically generated**

**5.2 SOLUTION AND TECHNICAL ARCHITECTURE**

**Diagram

Description automatically generated**

**5.3 USER STORIES**

**Table

Description automatically generated**

**6. PROJECT PLANNING AND SCHEDULING**

**6.1 SPRINT PLANNING AND ESTIMATION**

**Text

Description automatically generated with medium confidence**

**6.2 SPRINT DELIVERY SCHEDULE**

**Table

Description automatically generated**

**6.3 REPORTS FROM JIRA**

**Chart

Description automatically generated with medium confidence**

**Graphical user interface, text, application

Description automatically generated**

**7. CODING AND SOLUTIONING**

**7.1 FEATURE 1**

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("home"))

arr.append(float(val))

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

API\_KEY = "\_y5HXZoBlP-pbdVhiMEyIryHoMWR2xyJcXeiSGPJlRuz"

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'],

"values": [arr]

}]

}

response\_scoring = requests.post(

'https://jp-tok.ml.cloud.ibm.com/ml/v4/deployments/62772227-bda0-483c-9213-98ad2f100980/predictions?version=2022-11-17',

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("home"))

@app.route("/home")

def demo2():

return render\_template("home.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("home"))

if \_\_name\_\_ == "\_\_main\_\_":

app.run()

**7.2 FEATURE 2**

{% extends 'index.html' %}

{% block body %}

<div class="p-4">

<div class="row mb-3">

<div class="col-4">

<h2 class="text-responsive-h">

Enter you Scoring to view prediction

</h2>

<p class="text-responsive">

Students can Use their marks for prediction the admission in colleges and the administrator can allot the seats for the students. It will help student to make a better choice.

</p>

<div class="d-flex justify-content-right">

<img src="../static/Image/YES OR NO.jpg" class="card-img-top" 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 Score

</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-success" id="button">Let's try</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>

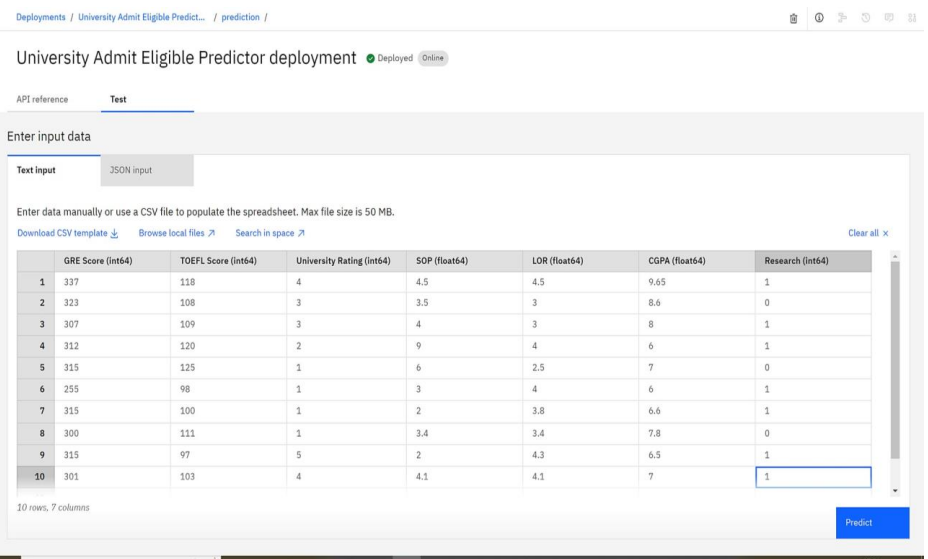
</div>

{% endblock %}

**8.TESTING**

**8.1 TEST CASES**

If the student is eligible for the university, it will give output as True. Otherwise, the output will be False.

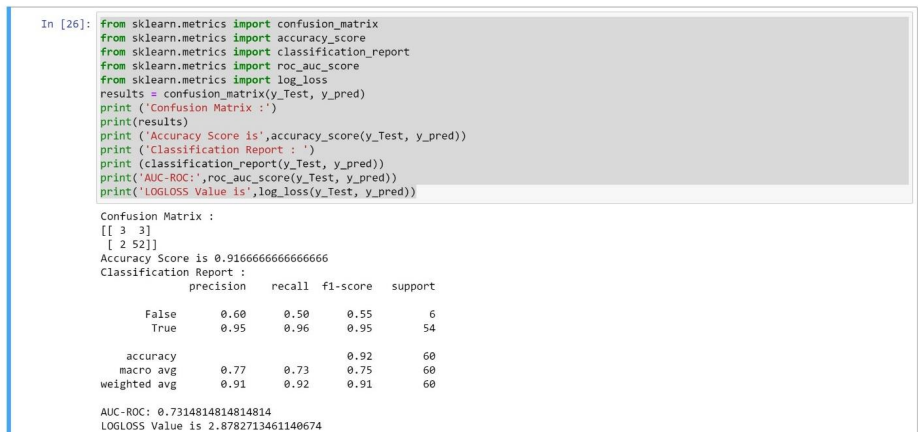


Table

Description automatically generated

**9. RESULTS**

**9.1 PERFORMANCE METRICS**

****

**10. ADVANTAGES AND DISADVANTAGES**

**ADVANTAGES:**

* It helps students 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.
* Avoids data redundancy and inconsistency.
* Very user-friendly.
* Easy accessibility of data.

**DISADVATAGES:**

* Required active internet connection.
* System will provide inaccurate results if data entered incorrectly.

**11. CONCLUSION**

The results of this examination appear to indicate that it greatly contributes to the response variable ‘Chance of Admit’. Higher the GRE, TOEFL score then higher the admit chances. The model predicts 91.5% accuracy and can be used for predicting the admit chances based on the above factors. This model will be helpful for the universities to predict the admission and ease their process of selection and timelines. As part of the hypothesis, the model proved that admission to Master’s degree program is dependent on GRE, TOEFL and other scores. This model would likely be greatly improved by the gathering of additional data of students from different universities which has similar selection criteria to choose the candidates for Master’s program.

**12. FUTURE SCOPE**

* This can be implemented in less time for proper admission process.
* This can be accessed anytime anywhere, since it is a web application provided only an internet connection.
* The user had not need to travel a long distance for the admission and his/her time is also saved as a result of this automated system.

**13. APPENDIX**

**SOUCE CODE LINK**

https://drive.google.com/drive/folders/1uvuErcuLCxM8NlmmSZgzuN3FNBmCyAq7?usp=sharing

**GITHUB LINK**

https://github.com/IBM-EPBL/IBM-Project-2182-1658465378.git

**DEMO LINK**

https://drive.google.com/file/d/15hMwhOuawjN0ZPRy-AZaMon0B0jsomCb/view?usp=share\_link