# UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

**Domain:** Applied Data Science

**Team ID:**

PNT2022TMID18312


**Submitted by**

Gaja Lakshmi J(1919102041)

Deepika C(1919102031)

Dharshini C(1919102035)

Bhuvana K(1919102026)

# TABLE OF CONTENTS

# 1.INTRODUCTION:

The university admission tests find the applicant's ability to admit to the desired university. In recent times, there is a huge competition in the university admission tests. The failure in the admission tests makes an examine e depressed. This paper addresses machine learning models to predict the chance of a student to be admitted to a master’s program. It can help students to improve their preparation to get a chance at their desired university. Many factors are responsible for the failure or success in an admission test.

### 1.1. PROJECT OVERVIEW:

In the current world scenario, it is not enough for a student to just have an Undergraduate degree. Most employers now look for higher qualifications in their new recruits. As a result, the demands for a good higher education are at an all-time high. Numerous students from India prefer to continue their higher education with foreign universities, especially in the United States. To get admitted to these foreign universities, a set of academic requirements are needed. However, because of the sheer number of universities of different levels, students are often stuck in a dilemma till the very last minute as to whether their applications will be accepted or not as no concrete documentation is available which lists the requirements. This project is a new machine learning web-based University Admit Eligibility Predictor. This University Admit Eligibility Predictor is an ML based application that asks for the users to input their academic transcripts data and calculates their chances of admission into the University Tier that they selected. It also provides an analysis of the data and shows how chances of admissions can depend on various factors. This survey analysis should also help students who are preparing or will be preparing to get a better idea for getting admission into the eligible university.

### 1.2 PURPOSE

This is a project report for a new web-based University Admissions Eligibility Predictor. University Admission Eligibility Predictor is an ADS based application that asks for the users to input their academic transcripts data and calculates their chances of admission into the University. It also provides an analysis of the data and shows how chances of admissions can depend on various factors. This document describes the scope, objectives and goals of the system. Additionally describing the non-functional requirements, this document models the functional requirements with use cases, interaction diagrams and class models. This document is intended to direct the design and implementation of the target system in an object-oriented language.

# 2.Literature Survey

A great number of researches and studies have been done on graduation admission datasets using different types of machine learning algorithms. One impressive work by Chary et al. [15] has compared between 4 different regression algorithms, which are: Linear Regression, Support Vector Regression, Decision Trees and Random Forest, to predict the chance of admit based on the best model that showed the least MSE which was multi-linear regression.

In addition, Chakrabarty et al. [16] compared between both linear regression and gradient boosting regression in predicting chance of admit; point out that gradient boosting regression showed better results.

Gupta et al. [17] developed a model that studies the graduate admission process in American universities using machine learning techniques. The purpose of this study was to guide students in finding the best educational institution to apply for. Five machine learning models were built in this paper including SVM (Linear Kernel), AdaBoost, and Logistic classifiers.

Waters and Miikkulainen [18] proposed a remarkable article that helps in ranking graduation admission application according to the level of acceptance and enhances the performance of reviewing applications using statistical machine learning.

Sujay [19] applied linear regression to predict the chance of admitting graduate students in master’s programs as a percentage. However, no more models were performed.

## 2.1. EXISTING PROBLEM

Many machine learning solutions have emerged in the recent years to tackle this problem and provide various predictions, estimations and consultancies so that students can easily make their decisions about applying to the universities with higher chances of admission. In this project, we review the machine learning techniques which are prevalent and provide accurate predictions regarding university admissions. We compare different regression models and machine learning methodologies such as, Random Forest, Linear Regression, Stacked Ensemble Learning, Support Vector Regression, Decision Trees, KNN(K-Nearest Neighbor) etc, used by other authors in their works and try to reach on a conclusion as to which technique will provide better accuracy.

## 2.2. REFERENCES

[1] M. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, “Multi-split Optimized Bagging Ensemble Model Selection for Multi-class Educational Data Mining,” Apple. Intel., vol. 50, pp. 4506–4528, 2020.

[2] F. Salo, M. Injadat, A. Moubayed, A. B. Nassif, and A. Essex, “Clustering Enabled Classification using Ensemble Feature Selection for Intrusion Detection,” in 2019 International Conference on Computing, Networking and Communications (ICNC), 2019, pp. 276–281.

[3] M. N. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, “Systematic ensemble model selection approach for educational data mining,” Knowledge-Based Syst., vol. 200, p. 105992, Jul. 2020.

[4] A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya, and A. Shami, “E-Learning: Challenges and Research Opportunities Using Machine Learning Data Analytics,” IEEE Access, 2018.

[5] M. S. Acharya, A. Armaan, and A. S. Antony, “A Comparison of Regression Models for Prediction of Graduate Admissions,” Kaggle, 2018. .

[6] S. S. Shapiro, M. B. Wilk, and B. T. Laboratories, “An analysis of variance test for normality,” 1965.

[7] G. K. Uyanık and N. Güler, “A Study on Multiple Linear Regression Analysis,” Procedia - Soc. Behav. Sci., vol. 106, pp. 234–240, 2013.

[8] C. López-Martín, Y. Villuendas-Rey, M. Azzeh, A. Bou Nassif, and S. Banitaan, “Transformed k-nearest neighborhood output distance minimization for predicting the defect density of software projects,” J. Syst. Softw., vol. 167, p. 110592, Sep. 2020.

[9] A. B. Nassif, O. Mahdi, Q. Nasir, M. A. Talib, and M. Azzeh, “Machine Learning Classifications of Coronary Artery Disease,” in 2018 International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), 2018, pp. 1–6.

[10] N. S. Altman, “An introduction to kernel and nearest-neighbor non-parametric regression,” Am. Stat., vol. 46, no. 3, pp. 175–185, 1992.

[11] A. B. Nassif, M. Azzeh, L. F. Capretz, and D. Ho, “A comparison between decision trees and decision tree forest models for software development effort estimation,” in 2013 3rd International Conference on Communications and Information Technology, ICCIT 2013, 2013, pp. 220–224.

[12] T. K. Ho, Random Decision Forests. USA: IEEE Computer Society, 1995.

[13] A. B. Nassif, “Software Size and Effort Estimation from Use Case Diagrams Using Regression and Soft Computing Models,” University of Western Ontario, 2012.

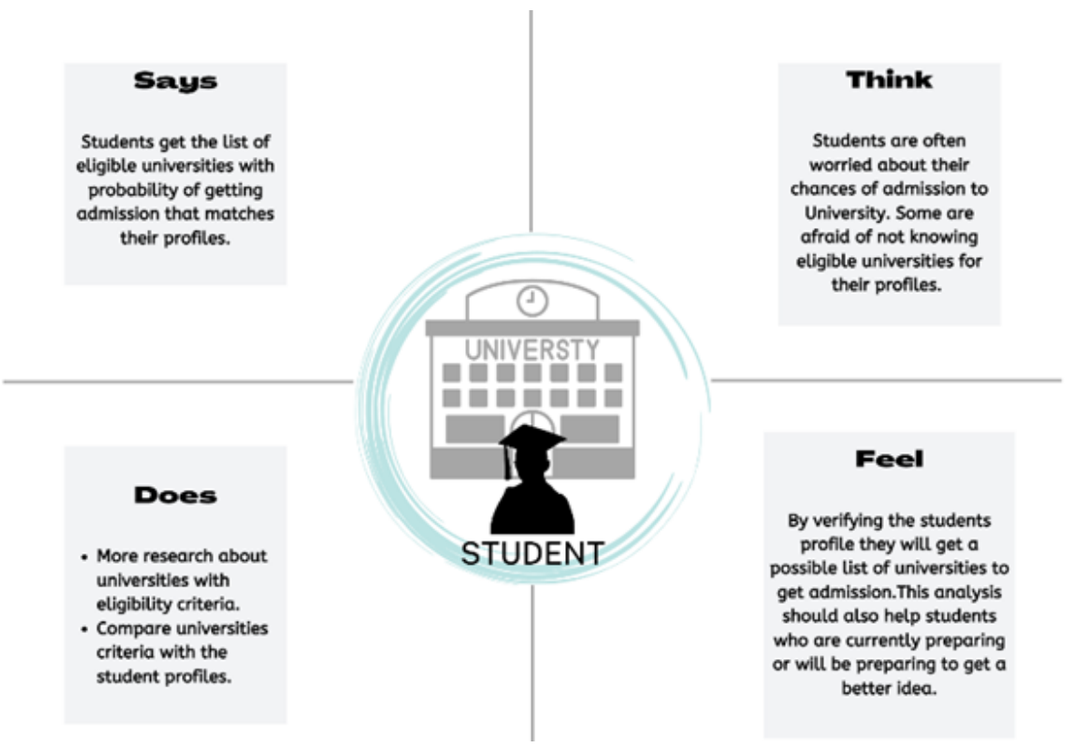
## 2.3. PROBLEM STATEMENT DEFINITON

Educational organizations have always played an important and vital role in society for development and growth of any individual. There are different college prediction apps and websites being maintained but using them is tedious to some extent, due to the lack of articulate information regarding colleges, and the time consumed in searching the best deserving college. The problem statement, hence being tackled, is to design a college prediction/prediction system and to provide a probabilistic insight into college administration for overall rating, cut-

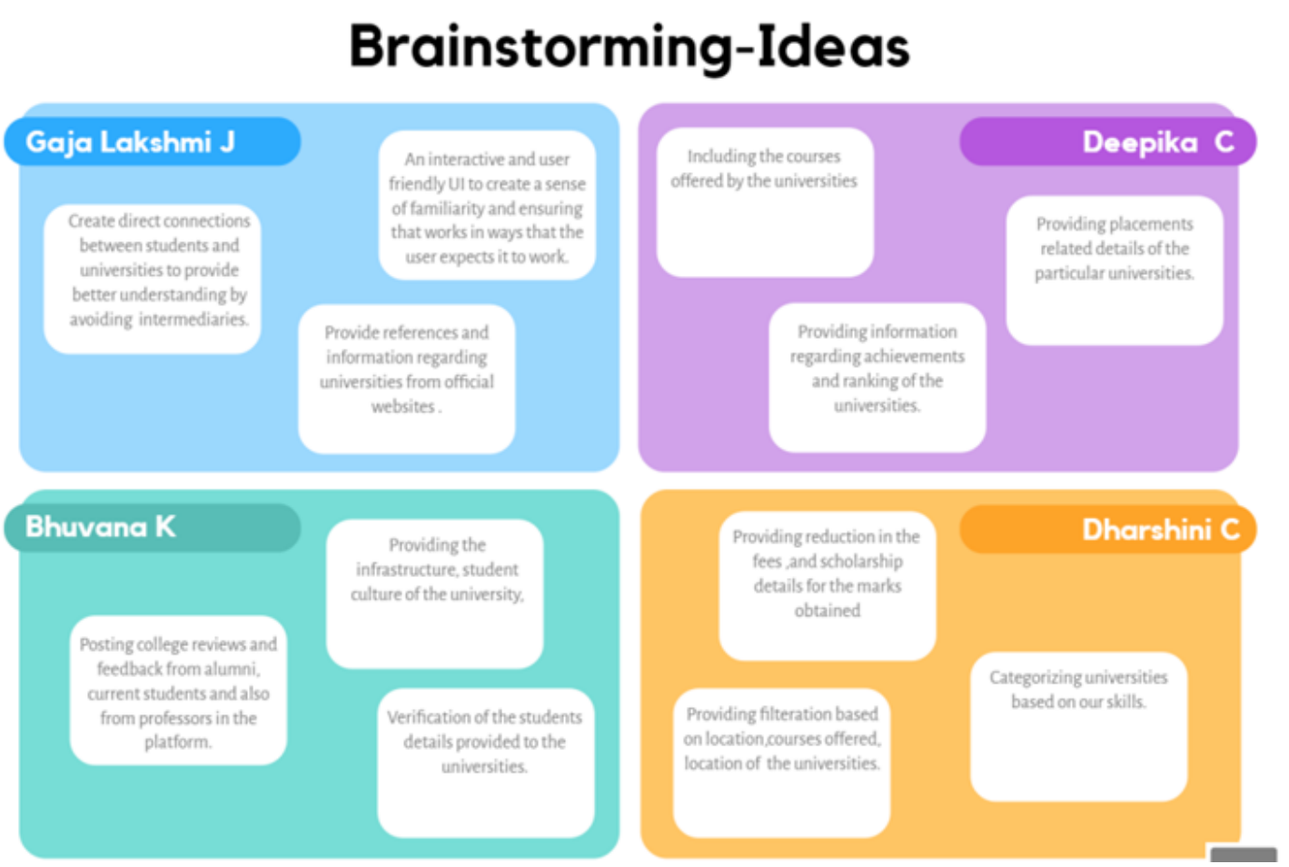
offs of the colleges, admission intake and preferences of students. Also, it helps students avoid spending time and money on counselor and stressful research related to finding a suitable college. It has always been a troublesome process for students in finding the perfect university and course for their further studies. At times they do know which stream they want to get into, but it is not easy for them to find colleges based on their academic marks and other performances. We aim to develop and provide a place which would give a probabilistic output as to how likely it is to get into a university given upon their details.

# 3.IDEATION & PROPOSED SOLUTION

## 3.1. Empathy Map Canvas




## 3.2. Ideation & Brainstorming

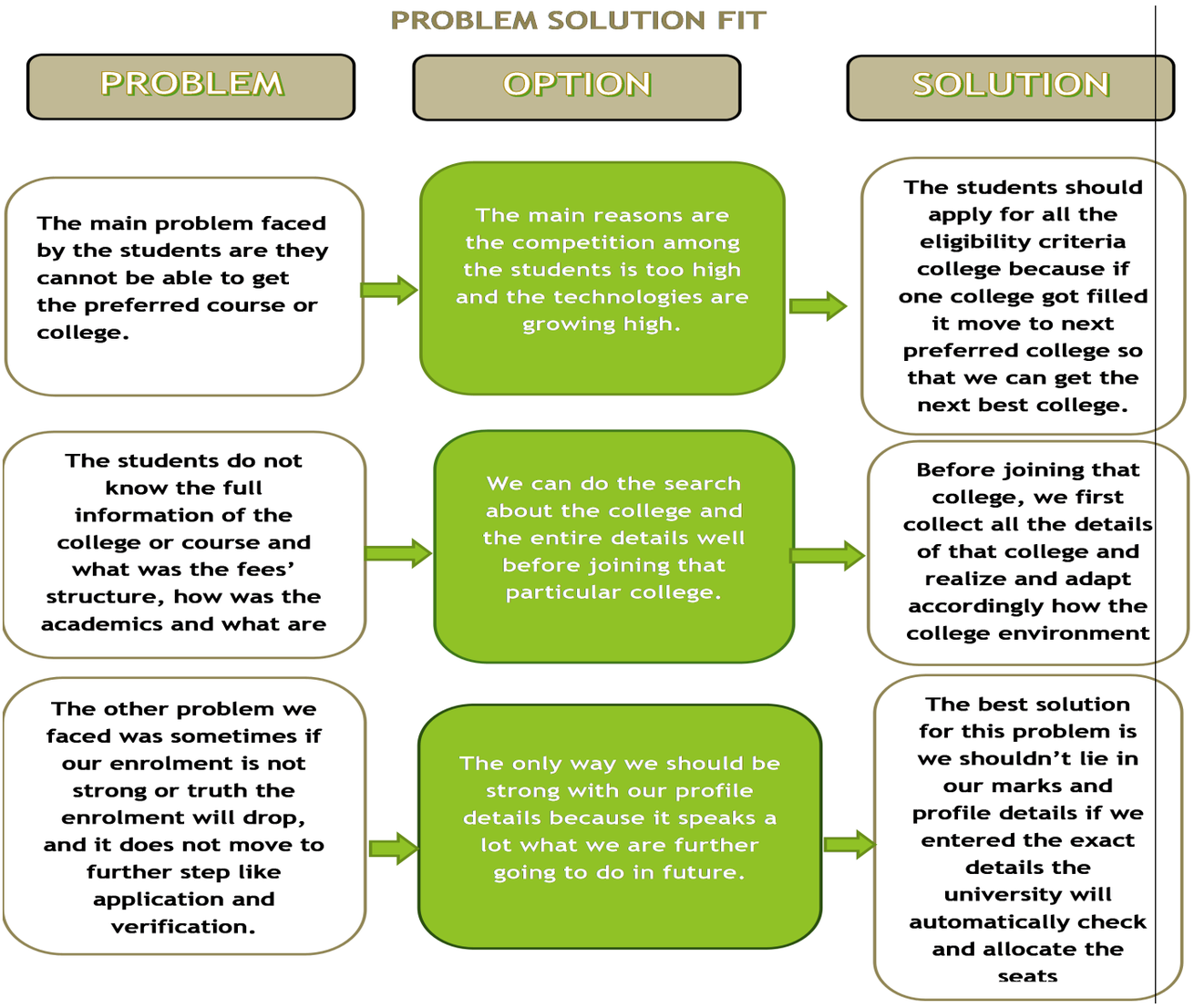


### 3.3. Proposed Solution

| S.No | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | Most of the students who apply and sit for the admission test do not have the guarantee of admission opportunities in the university because of the limited number of seats.<br>Students must overcome the barrier of admission test and qualify in the examination to secure their seats. Such students must go through a long time of mental stress or illness before or after the admission test.<br>But with the aid of modern technologies and strategies e.g. educational data mining, this predictor can reduce the problem and make students aware of it in the admission test.<br>If any student can know the pre-examination and post-examination status of a particular university for undergraduate admission, it will be a great benefit for him/her to take the necessary steps to improve the admission test's performance so that he/she can get a chance at the desired university.<br>Hence, we want to help the students to judge and improve themselves before or after the admission test using this system. |
| 2. | Idea / Solution description | University Admit Eligibility Predictor is an AI-based application that asks the users to input their academic transcripts data and calculates their chances of admission into the University Tier that they selected.<br>It also provides an analysis of the data and shows how chances of admissions can depend on various factors.<br>It can help students to improve their preparation to get a chance at their desired university.<br>Many factors are responsible for the failure or success in an admission test.<br>Educational data mining helps us to analyze and extract information from these factors.<br>Here, we apply several machine learning algorithms on a collected dataset to estimate the probability of getting admission to the university after attending the university tests.<br>We also evaluate and compare the performance levels of those algorithms based on two different evaluation metrics – accuracy and test scores. |
| 3. | Novelty / Uniqueness | We at University Admit Eligibility Predictor are here to provide a solution to a problem that not only do we provide a single platform that documents all the requirements as well as the different tiers of universities, but our website also incorporates an AI Model that was built after considering many leading Machine Learning Algorithms, to provide the most accurate prediction of how much of a chance of admissions does a student's current grades and other academic transcripts allow them in the tier of universities of their choice |
| 4. | Social Impact / Customer Satisfaction | The primary responsibilities of the system are:<br>Provide customers access to the prediction model.<br>Provide answers to the most common FAQs regarding PG Admissions abroad.<br>Provide administrator access to all records.<br>Provide analysis of how the various academic factors affect university admission.<br>Other desired features of the system:<br>Maintaining a profile for each user.<br>Password protection for each account. |

| | | |
|---|---|---|
| 5. | Business Model (Revenue Model) | Initially, a good amount of time was spent on understanding the problem statement by understanding the concerns of students regarding the current application process. The objectives of the research were defined in this process.<br>Maintaining a profile for each user and Password protection for each account.<br>A Database will also be implemented for the system so that students can save their data and review and edit it as they progress with the most recent predictions being saved with their profile.<br>Multiple machine learning models were developed to predict the likelihood of success of the student's application in a particular university. The user interface was developed to allow the users to access these models.<br>Maintaining a profile for each user<br>• Password protection for each account<br>Maintaining a profile for each user<br>• Password protection for each account<br>Maintaining a profile for each user<br>• Password protection for each account |
| 6. | Scalability of the Solution | The scope of this project is a web application that allows users to enter their academic data and get predictions of their chances of admission in the university tier of their choosing. It also provides them answers to the most common FAQ's that arise when thinking of admissions abroad for Post Graduate studies. It also provides an analysis based on the data set used that shows how the different parameters affect chances of admissions. A Database will also be implemented for the system so that students can save their data and review and edit it as they progress with the most recent predictions being saved with their profile. |

### 3.4. Problem Solution Fit




# 4.REQUIREMENT ANALYSIS

### 4.1. 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 | 1. Registration through Form<br>2. Registration through Gmail<br>3. Registration through LinkedIN |
| FR-2 | User Confirmation | 1. Confirmation via Email<br>2. Confirmation via OTP |
| FR-3 | User filling the required details | 1. Enter the marks or percentage obtained in SSLC and HSC.<br>2. Enter the UG percentage or CGPA for PG admission.<br>3. Enter the preferred location. |
| FR-4 | Analyzing | 1. Analysis user credentials and compare with universities criteria. |
| FR-5 | Predicting | 2. Predicting the probability for getting admissions in the universities by analyzing various machine learning algorithms. |

### 4.2. Non-functional Requirements:

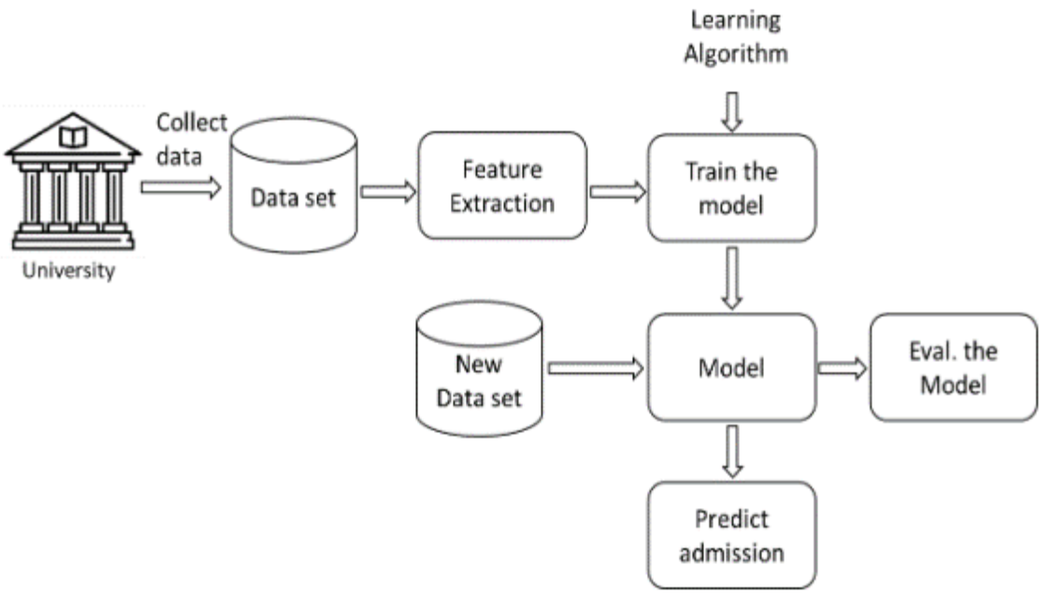
Following are the non-functional requirements of the proposed solution.

| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | Usability | 1. No training is required to use the website.<br>2. The form, home, about, FAQ and analysis pages load up within 10 seconds.<br>3. The results from the predictor should not take more than 30 seconds. |
| NFR-2 | Security | 1. The system shall provide password protected access to the website to all users – students and admins. |
| NFR-3 | Reliability | 2. The system shall be completely operational all hours of the day unless system failure or upgradation work is to be performed<br>3. Down time after a failure shall not exceed 24 hours . |
| NFR-4 | Performance | 1. The system can support any number of users at a time.<br>2. The meantime to view a webpage over a 56Kbps modem connection shall not exceed 5 seconds. |
| NFR-5 | Availability | 1. The system will be able to incorporate more features without major reengineering.<br>2. The system web site shall be viewable from Internet Explorer 4.0 or later, Netscape Navigator/Communicator 3.0 or later and the America Online web browser version 3.0 or later. |
| NFR-6 | Scalability | 1. The scope of this project is a web application that allows users to enter their academic data and get predictions of their chances of admission in the university tier of their choosing.<br>2. It also provides them answers to the most common FAQ’s that arise when thinking of admissions abroad for Post Graduate studies.<br>3. It also provides an analysis based on the data set used that shows how the different parameters affect chances of admissions.<br>4. A Database will also be implemented for the system so that students can save their data and review and edit it as they progress with the most recent predictions being saved with their profile. |

# 5.PROJECT DESIGN

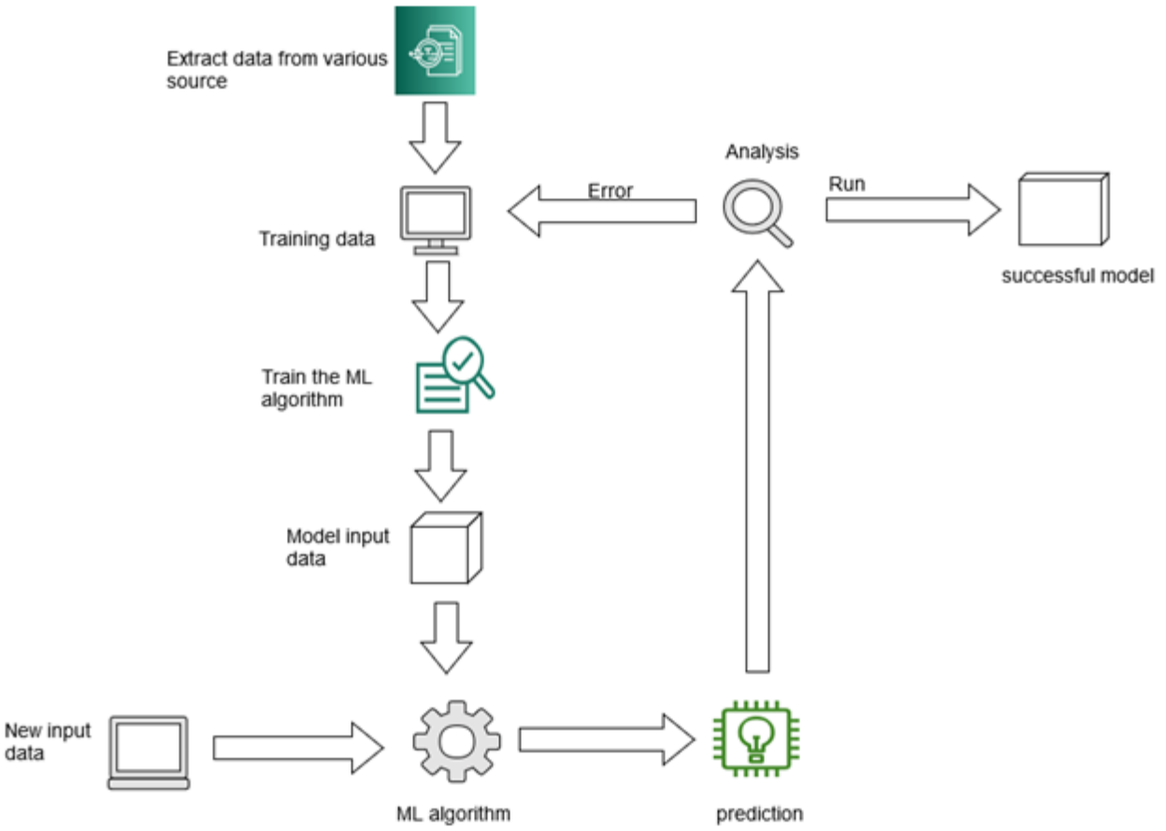
## 5.1. DATA FLOW DIAGRAMS

**System Overview**




**Data Flow Diagram**

## 5.2. SOLUTION & TECHNICAL ARCHITECTURE




## 5.3. USER STORIES

| Phases | Phase 1 *Motivation* | Phase 2 *Search and apply for university* | Phase 3 Getting rejection for admission | Phase 4 *Providing details in UAEP* | Phase 5 Predicting Eligible universities | Phase 6 *Apply for the favorite university* |
|---|---|---|---|---|---|---|
| Activities Performed | Students want to apply for the universities for higher studies. | Searches for admission in the universities on various websites and apply for it. | Post search, student applies for the university to which they are interested but got rejected since criteria not met with university. | Students provide the scores, area of interest, preferred location and etc. in the UAEP predicting website. | Predict the list of eligible universities with probability for admission to which they can apply to it. | Apply to the top eligible university and application for admission got selected. |
| Emotions | Happy and Excited | Happy as the student finds numerous universities. | Disappointed as the first admission application got rejected. | Very happy as he uses UAEP for prediction by providing the students interested fields, scores etc. | Happy! Apply to the eligible university and find the favorite university on the predicted results. | Predicts the eligible university using UAEP. Apply to the favorite university listed after prediction and gets admission. |
| Overall Experiences | Good | Good | Bad | Good | Good | Good |
| Customer Expectations | Easy availability of the websites, universities, and admission options. | Student thinks it as an easy-to-use search engine that can make the university search process easy. | Student expects to get admission in first try. | Easy and convenient for students to know the eligible universities. | By seeing the list of eligible universities student get excited. | getting admission in first try after predicting makes them happier. |

# 6. PROJECT PLANNING & SCHEDULING

## 6.1. SPRINT PLANNING & ESTIMATION

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | 2 | High | Gaja Lakshmi J & Deepika C |
| Sprint-1 | | USN-2 | As a user, I will receive confirmation email onceI have registered for the application. | 1 | High | Gaja Lakshmi J &Deepika C |
| Sprint-1 | Login | USN-3 | As a user, I can log into the application by entering email and password. | 3 | Medium | Dharshini C & Bhuvana K |
| Sprint-2 | Update Profile | USN-4 | As a user, after logging in i will have to update-my profile by providing all the required details. | 2 | Medium | Dharshini C & Bhuvana K |
| Sprint-3 | Analysis | USN-5 | As a user I have to analyses the possible universities which is matches with my profile. | 3 | High | Gaja Lakshmi J &Deepika |
| Sprint-1 | Authentication Process | USN-6 | As admin, the login credential of the user is authenticated by me. | 3 | medium | Dharshini C & Bhuvana K |
| Sprint-3 | prediction | USN-7 | As admin I can test the trained ML model by analyzing the user details by ML algorithms like logistic Regression. | 5 | High | Gaja Lakshmi J & Deepika C |
| Sprint-1 | Output | USN-8 | As adminI can upload the confirmation of userfor the prediction into the Database. | 5 | High | Dharshini C & Bhuvana K |
| Sprint-4 | Logout | USN-9 | As a user, after all the process is over then Ican log out of the application. | 2 | Low | Gaja Lakshmi J &Bhuvana K |

### 6.2. SPRINT DELIVERY SCHEDULE

| Sprint | Total StoryPoints | Duration | Sprint Start Date | Sprint End Date (Planned) | Story Points Completed (as on Planned End Date) | Sprint ReleaseDate(Actual) |
|---|---|---|---|---|---|---|
| Sprint-1 | 8 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 8 | 29 Oct 2022 |
| Sprint-2 | 5 | 6 Days | 31 Oct 2022 | 05 Nov2022 | 5 | 05 Nov2022 |
| Sprint-3 | 15 | 6 Days | 07 Nov2022 | 12 Nov2022 | 15 | 12 Nov2022 |
| Sprint-4 | 3 | 6 Days | 14 Nov2022 | 19 Nov2022 | 3 | 14 Nov2022 |

# 7.CODING & SOLUTIONING

## app.py:

```
import pandas as pd
from flask import Flask, request, jsonify, render_template
import pickle

import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "your api key"
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}

app = Flask(__name__)
model = pickle.load(open('linear_regression_model_sc.pkl', 'rb'))


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

@app.route('/predict', methods=['GET','post'])
def predict():

	GRE_Score = int(request.form['GRE Score'])
	TOEFL_Score = int(request.form['TOEFL Score'])
	University_Rating = int(request.form['University Rating'])
	SOP = float(request.form['SOP'])
	LOR = float(request.form['LOR'])
	CGPA = float(request.form['CGPA'])
	Research = int(request.form['Research'])

	# 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":
[GRE_Score,TOEFL_Score,University_Rating,SOP,LOR,CGPA,Research]}]}
	response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/80d03055-cefc-404a-8502-
e1be5e99c2ae/predictions?version=2022-11-13', json=payload_scoring,
   headers={'Authorization': 'Bearer ' + mltoken})
	print("Scoring response")
	print(response_scoring.json())
	final_features = pd.DataFrame([[GRE_Score, TOEFL_Score, University_Rating, SOP, LOR, CGPA, Research]])

	predict = model.predict(final_features)

	output = predict[0]

	return render_template('predict.html', Admission_Prediction='Admission chances are {}'.format(output))

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

# index.html:

```
<!DOCTYPE html>
<html>
<head>
 <SCRIPT language=Javascript>
   <!--
   function check(e, value) {
     //Check Charater
     var unicode = e.charCode ? e.charCode : e.keyCode;
     if (value.indexOf(".") != -1)
       if (unicode == 46) return false;
     if (unicode != 8)
       if ((unicode < 48 || unicode > 57) && unicode != 46) return false;
   }
   //-->
 </SCRIPT>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<style>
   body {
       font-family: Arial, Helvetica, sans-serif;

   }

* {
  box-sizing: border-box;
}

/* style the container */
    .container {
        position: relative;
        border-radius: 5px;
        background-image: linear-gradient(180deg, #F0F97E, #00E0D8);
        padding: 40px;
    }

/* style inputs and link buttons */
input,
.btn {
  width: 50%;
  padding: 12px;
  border: none;
  border-radius: 6px;
  margin: 5px 0;
  opacity: 0.75;
  display: inline-block;
  font-size: 17px;
  line-height: 20px;
  text-decoration: none; /* remove underline from anchors */
  margin-left:130px;
}


input:hover,
.btn:hover {
  opacity: 1;
}


/* style the submit button */
input[type=submit] {
  background-color: #4CAF50;
  color: white;
  cursor: pointer;
  width: 25%;
  margin-left: 200px;
}

input[type=submit]:hover {
  background-color: #45a049;
}

/* Two-column layout */
.col {
  float: left;
  width: 50%;
  margin: auto;
```

```
  padding: 0 60px;
  margin-top: 6px;
}

 .row:before {
  content: "";
  display: table;
  clear: both;
    }
 /* Clear floats after the columns */
  .row:after {
  content: "";
  display: table;
  clear: both;
}

/* vertical line */
.vl {
  position: absolute;
  left: 50%;
  transform: translate(-50%);
  border: 2px solid #004138;
  height: 480px;
}

/* text inside the vertical line */
.vl-innertext {
  position: absolute;
  top: 50%;
  transform: translate(-50%, -50%);
  background-color: #00A89A;
  border: 1px solid #40E0D0;
  border-radius: 50%;
  padding: 10px 10px;
}

/* hide some text on medium and large screens */
.hide-md-lg {
  display: none;
}

/* bottom container */
    .bottom-container {
        text-align: center;
        background-color: #00E0D8;
        opacity:0.85;
        border-radius: 0px 0px 4px 4px;
    }

/* Responsive layout - when the screen is less than 650px wide, make the two columns stack on top of each other instead of next to each other */
@media screen and (max-width: 650px) {
    .col {
        width: 100%;
        margin-top: 0;
        margin-left:40px;
    }
  /* hide the vertical line */
  .vl {
    display: none;
  }
  /* show the hidden text on small screens */
  .hide-md-lg {
    display: block;
    text-align: center;
  }
}
</style>
</head>
<body>

<div class="container">
  <form action="/predict" method="POST" autocomplete="off">
      <div class="row">
          <div class="bottom-container">
              <h1 style="color:#00090E">University Admit Eligibility Predictor</h1>
              </div>
              <div class="vl">
                  <span class="vl-innertext"></span>
              </div>
```

```
            <div class="col">
               <h3 style="color:#025043">About</h3>
                    <p>The aim of this university admit eligibility predictor is to predict the chances of admission to an university based on student's profile.</p>
               <h3 style="color:#025043">Instructions for Input</h3>
               <ul>
                  <li>GRE Score (out of 340)</li>
                  <li>TOEFL Score (out of 120)</li>
                  <li>University Rating (out of 5)</li>
                  <li>Statement of Purpose {SOP} (out of 5)</li>
                  <li>Letter of Recommendation {LOP} Strength (out of 5)</li>
                  <li>Undergraduate CGPA (out of 10)</li>
                  <li>Research Experience (Either 0 or 1)</li>

               </ul>
            </div>

            <div class="col">
               <div class="hide-md-lg">
               </div>

               <input type="number" name="GRE Score" placeholder="GRE Score" required="required" min="0" max="340" />
               <input type="number" name="TOEFL Score" placeholder="TOEFL Score" required="required" min="0" max="120" />
               <input type="number" name="University Rating" placeholder="University Rating" required="required" min="1" max="5" />
               <input type="number" name="SOP" placeholder="SOP" required="required" onkeypress="return check(event,value)" step="0.1" min="1" max="5" />
               <input type="number" name="LOR" placeholder="LOR" required="required" onkeypress="return check(event,value)" step="0.1" min="1" max="5" />
               <input type="number" name="CGPA" placeholder="CGPA" required="required" onkeypress="return check(event,value)" step="0.01" min="1"
max="10" />
               <input type="number" name="Research" placeholder="Research" required="required" min="0" max="1" />

               <input type="submit" value="Predict"/>
            </div>

         </div>
</form>

</div>

<div class="bottom-container">
  <div class="row">
    <h3 style="color:#00090E">Be bold, Be courageous, Be your best.</h3>
  </div>
</div>

</body>
</html>
```

# predict.html:

```
<!DOCTYPE html>
<html lang="en-US">
<head>
   <meta charset="utf-8" />
   <meta name="viewport" content="width=device-width" />
   <title>Predicted Result</title>
   <style>
      body {
         font-family: Arial, Helvetica, sans-serif;
         margin: 0;
      }

      html {
         box-sizing: border-box;
      }

      *, *:before, *:after {
         box-sizing: inherit;
      }

      .column {
         float: left;
         width: 33.3%;
         margin-bottom: 16px;
         padding: 0 8px;
      }

      .card {
         box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
         margin: 8px;
```

```
        }

        .prediction {
            padding: 220px;
            text-align: center;
            color: white;
            background-image: linear-gradient(180deg, #F0F97E, #00E0D8);
        }

        .title {
            color: grey;
        }

        .button-1 {
            background-color: #45a049;
            border-radius: 8px;
            border-style: none;
            box-sizing: border-box;
            color: #FFFFFF;
            cursor: pointer;
            display: inline-block;
            font-family: "Haas Grot Text R Web", "Helvetica Neue", Helvetica, Arial, sans-serif;
            font-size: 14px;
            font-weight: 500;
            height: 40px;
            line-height: 20px;
            list-style: none;
            margin: 0;
            outline: none;
            padding: 10px 16px;
            position: relative;
            text-align: center;
            text-decoration: none;
            transition: color 100ms;
            vertical-align: baseline;
            user-select: none;
            -webkit-user-select: none;
            touch-action: manipulation;
        }

            .button-1:hover,
            .button-1:focus {
                background-color: #F082AC;
            }
    </style>
</head>
<body>
    <div class="prediction">
        <h1 style="color: #00A89A">Predicted Result</h1>
        <br />
        <h2 style="text-align: center; color: #025043">{{Admission_Prediction}}</h2>
        <br />
        <a href="/" class="button-1" role="button">Go Back</a>
    </div>
</body>
</html>
```

# 8.TESTING

## 8.1. Test case:

The following section contains the report of the testing phase of the software.

| | |
|---|---|
| Project Name | University Admit Eligibility Predictor |
| Project Type | Applied Data Science Based Web Application |
| Developer | Gaja Lakshmi J, Deepika C, Dharshini C, Bhuvana K |
| Language | Python, html, css |
| Total Number of Test cases | 45 |
| Number of Test cases executed | 45 |
| Total Number of Test cases passed | 44 |
| Total Number of Test cases failed | 1-System not useful for Blind People |
| Positive Test cases | 33 |
| Negative Test cases | 12 |

## 8.2. User Acceptance Testing

### 1. Defect Analysis

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

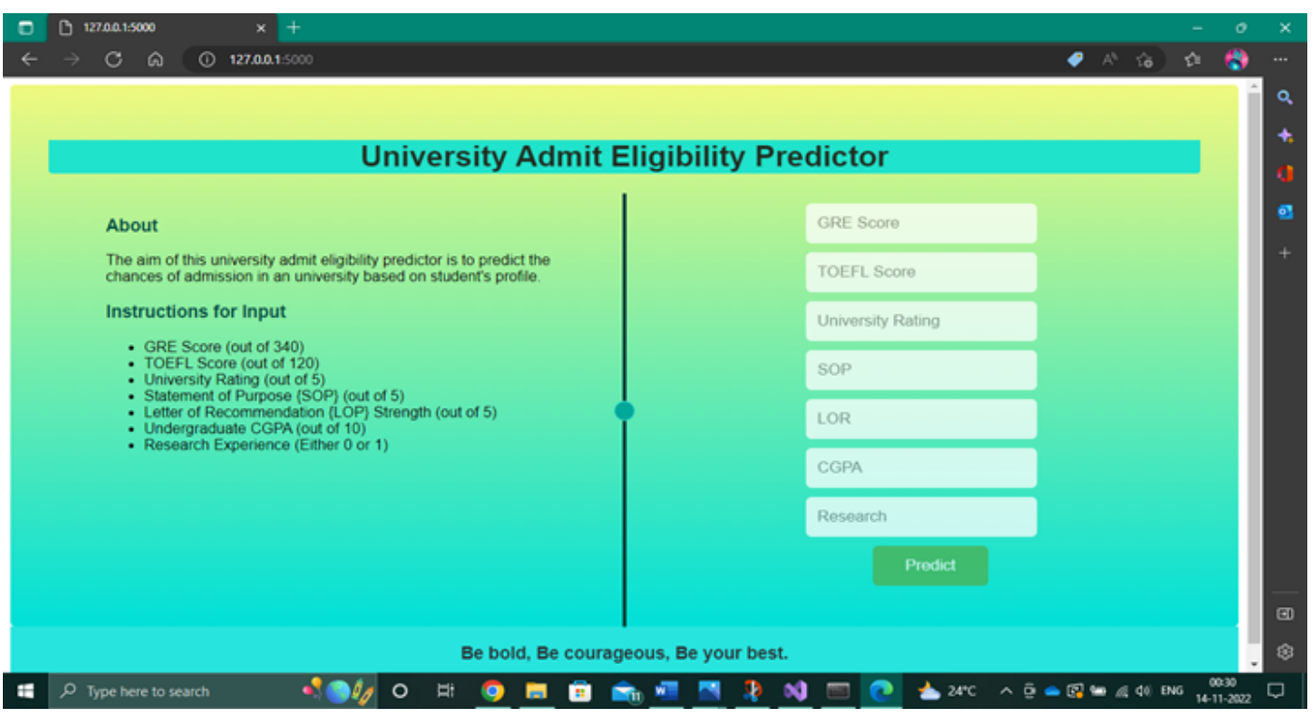
| Resolution | Severity 1 | Severity 2 | Severity 3 | Severity 4 | Subtotal |
|---|---|---|---|---|---|
| By Design | 10 | 4 | 2 | 3 | 19 |
| Duplicate | 1 | 0 | 3 | 0 | 4 |
| External | 2 | 3 | 0 | 1 | 6 |
| Fixed | 11 | 2 | 4 | 20 | 38 |
| Not Reproduced | 0 | 0 | 1 | 0 | 1 |
| Skipped | 0 | 0 | 1 | 1 | 2 |
| Won't Fix | 0 | 5 | 2 | 1 | 8 |
| Totals | 23 | 14 | 13 | 26 | 76 |

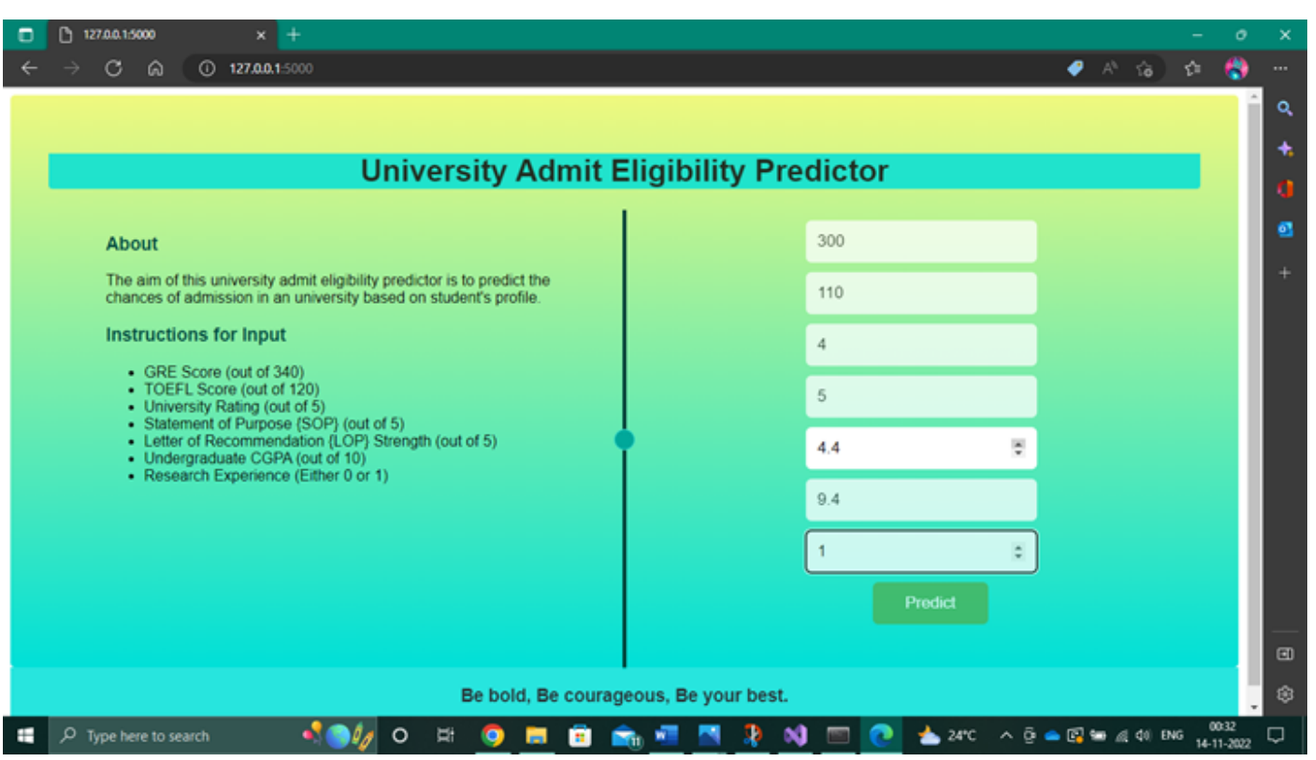
## 1. Test Case Analysis

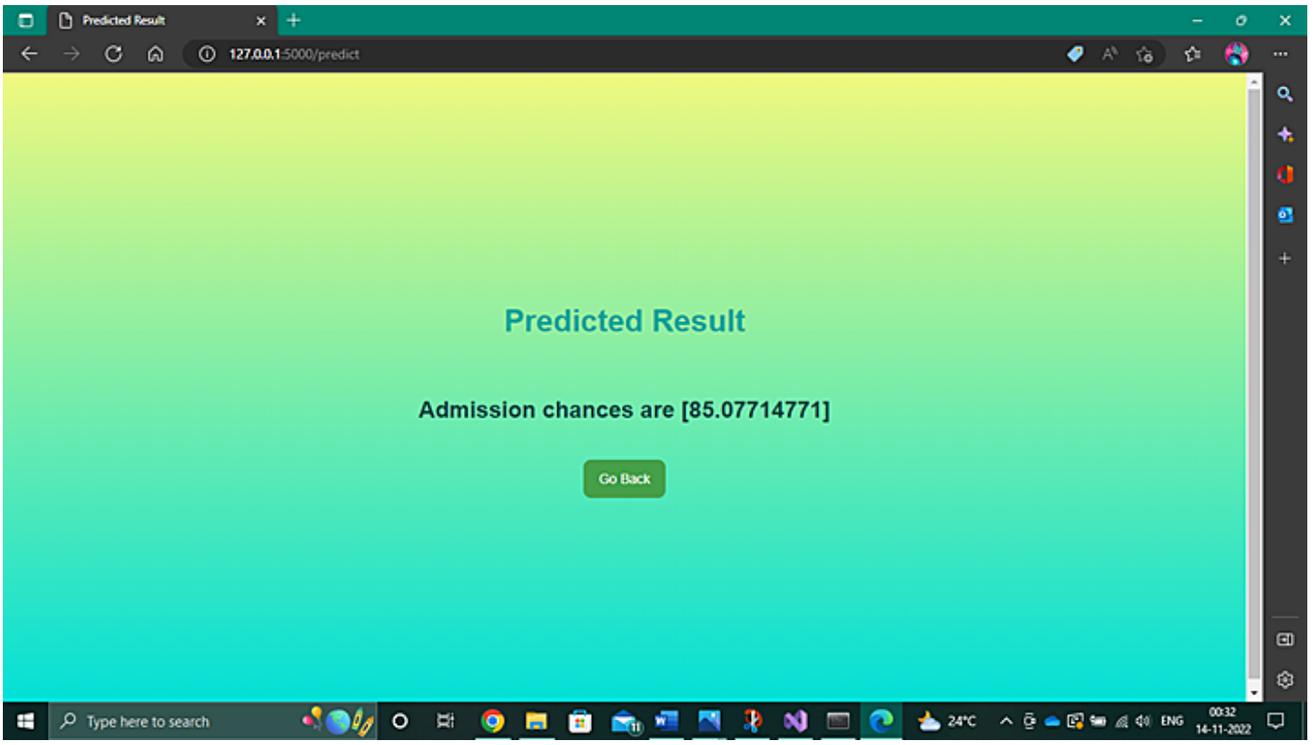
This report shows the number of test cases that have passed, failed,and untested

| Section | Total Cases | Not Tested | Fail | Pass |
|---|---|---|---|---|
| Print Engine | 7 | 0 | 0 | 7 |
| Client Application | 23 | 0 | 0 | 23 |
| Security | 2 | 0 | 0 | 2 |
| Outsource Shipping | 3 | 0 | 0 | 3 |
| Exception Reporting | 9 | 0 | 0 | 9 |
| Final Report Output | 4 | 0 | 0 | 4 |
| Version Control | 2 | 0 | 0 | 2 |

# 9.RESULTS






## 10.ADVANTAGES & DISADVANTAGES:

**Advantages:** We at University Admit Eligibility Predictor are here to provide solutions to major problems. Not only do we provide a single platform that documents all the requirements as well as the different tiers of universities, but our website also incorporates an AI Model that was built after considering many leading Machine Learning Algorithms, to provide the most accurate prediction of how much of a chance of admissions does a student's current grades and other academic transcripts allow them in the tier of universities of their choice.

**Disadvantages:** Issues of web security other than credentials protection within the website are not part of this project.

## 11.CONCLUSION

The main objective of this research was to develop a prototype of the system that can be used by the students aspiring to pursue their education in the USA. Multiple machine learning algorithms were developed and used for this research. Linear Regression proved to be best-fit for development of the system when compared with the Logistic regression model. The model can be used by the students for evaluating their chances of getting shortlisted in a particular university with an average accuracy of 75%. The linear Regression algorithm was used to predict the universities which were best suitable for a student based on their profile. The Linear Regression algorithm proved to be 80% accurate. A simple user interface was developed to make the application interactive and easy to use for the users from a non-technical background. Python, HTML and CSS were used to create the user interface. The overall objective of the research was achieved successfully as the system allows the students to save the extra amount of time and money that they would spend on education consultants and application fees for the universities where they have fewer chances of securing admission. Also, it will help the students to make better and faster decisions regarding application to the universities.

## 12.FUTURE SCOPE

In future, more data related to additional universities and courses can be added to the system. Also, the system can be enhanced to a web-based application by making changes to the flask code. Other classification algorithms can be evaluated to resolve the problem. if they perform better than the current algorithm, the system can be easily updated to support the new algorithm by changing the server code in the flask app.

## 13.APPENDIX

GITHUB LINK: https://github.com/IBM-EPBL/IBM-Project-26388-1660025833

Demo_Link:https://drive.google.com/file/d/1X7PIndXb1Hpn0nk5B3INi54BFO4qH9SS/view?usp=share_link