M.KUMARASAMY
COLLEGE OF ENGINEERING
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 & ISO 14001:2015 Certified Institution
Thalavapalayam, Karur – 639 113.

A Project Report
on

UNIVERSITY ADMIT ELIGIBLITY PREDICTOR USING MACHINE LEARNING

Submitted in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

Under the Guidance of
Dr. S. SUJANTHI M.E., PH.D.,
Assistant Professor/CSE

Submitted by

TEAM ID: PNT2022TMID15562

927619BCS4108 - SANTHOSH KUMAR P S
927619BCS4106 - SANJAY KIRAN K P
927619BCS4122 - VIGNESHWARAN B
927619BCS4123 - VIGNESHWARAN K

NAALAIYA THIRAN – EXPERIENTIAL PROJECT BASED LEARNING INITIATIVE

18CSE040L - PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY AND ENTREPRENEURSHIP

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING, KARUR

(Autonomous)

Karur - 639 113

November, 2022

ABSTRACT

Student admission problem is very important in educational institutions. This paper addresses machine learning models to predict the chance of a student to be admitted to a master's program. This will assist students to know in advance if they have a chance to get accepted. The machine learning models are multiple linear regression, k-nearest neighbor, random forest, and Multilayer Perceptron. Experiments show that the Multilayer Perceptron model surpasses other models.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 PROJECT OVERVIEW

The global business sector is expanding and is continuously looking for knowledge and experiences that are generally advantageous to people. Young specialists who need to keep their existing jobs are constantly looking for advanced degrees to supplement their knowledge and skills. As a result, during the past ten years, more of her sophomores have applied to take the graduation exams. Getting into her dream university is one of her biggest worries. Clearly college freshmen choose to attend prestigious institutions for their education. Additionally, the majority of the world's international graduates follow a pattern that is centred on the United States. The top institutions provide a variety of courses that can be taken in any order, highly regarded teaching and educational programmes, internationally accepted second There are research degree scholarships available.

In her 4,200+ private and public schools and universities, more than 4,444 of her 10 million international sophomores are enrolled, according to Gauges. In general, Asian nations like India, Pakistan, Sri Lanka, Japan, and China account for a large portion of the undergraduate population in the United States. In addition to the United States, choose the United Kingdom, Germany, Italy, Australia, and Canada. The number of people looking for more in-depth research is rapidly rising in these nations. The main reason sophomores enrol in master's programmes at foreign graduate schools is that there are few openings in these positions and a large number of persons holding them in each nation. Many professional undergraduates have pursued postgraduate courses as a result of this. You can see that US colleges offer a sizable number of computer science bachelor's and master's degrees. These undergraduate degrees are relevant to the study's main subject. Many US universities adhere to comparable standards for undergraduate accreditation. Placement in fitness evaluations and school achievement ratings are just two of the many factors that schools take into account. Rankings for English are based on experience taking English proficiency exams like the TOEFL and IELTS.

Based on the general profile of the applicant's application, the University's Admissions Advisory Board determines whether to accept or reject certain young researchers. This company's records are highlighted with informative areas. A 400-row data collection called Acknowledgment contains seven separate autonomic components. ie

- The result of the Graduate Record Examination (GRE). There are 340 foci in the score.
- The TOEFL exam result for English as a Foreign Language. It has 120 areas of priority.
- Uni .Rating. demonstrates where universities that grant bachelor's degrees stand among other universities. There will be a scale of 1 to 5.
- A Statement of Purpose (SOP), a document created to describe the life, reasons for seeking a certain degree or college, and sources of inspiration. There are five focus points in the score.
- The effectiveness of a letter of recommendation (LOR) confirms the applicant's work history, falsifies veracity, upholds assurance, and attests to your ability. There are five focus points in the score.
- A CGPA (undergraduate) of at least ten.
- Research experience (0 or 1) that could support the application, such as presenting research papers at conferences or completing a right-hand exam for university faculty. One ward variable, the likelihood of affirmation, which ranges from 0 to 1 depending on the input, can be anticipated.

1.2 PURPOSE

This is a Requirements Specification Document for a new web-based University Admissions Predictor – is an AI 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 document describes the scope, objectives and goals of the system. Besides 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.

1.1.1. USERS

Students- The people who will benefit the most from using this system are Indian students. Especially students looking to pursue their higher education from foreign universities, particularly in the United States.

Administrators- The administrator shall be able to access all the data stored in the application.

1.1.2. LOCATION

The system will be available to all users from any location as long as they have an Internet connection. The administrator can also access the website from any location as long as he has the correct login credentials and access to the Internet.

1.2.3. RESPONSIBILITIES

1. The primary responsibilities of the system are:

    - Provide customers access to the prediction model
    - Provide answers to most common FAQs regarding PG Admissions abroad
    - Provide administrator access to all records
    - Provide analysis of how the various academic factors affect university admission

2. Other desired features of the system:

    - Maintaining a profile for each user
    - Password protection for each account

1.2.4. NEEDS

This system is needed so as to answer the queries of students in a compete and concise manner as well as to provide them an as accurate as possible analysis of their chances of admissions to their dream universities.

CHAPTER 2

LITREATURE SURVEY

2.1 EXISTING PROBLEM

Student admission problem is very important in educational institutions. This paper addresses machine learning models to predict the chance of a student to be admitted to a master's program. This document describes the scope, objectives and goals of the system. Besides 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.

Vandit Manish Jain, Rihaan Satia [1]. Every year millions of students apply to universities to begin their educational life. Most of them don't have proper resources, prior knowledge and are not cautious, which in turn creates a lot of problems as applying to the wrong university/college, which further wastes their time, money and energy. With the help of this project, it helps such students who are finding difficulty in finding the right university for them. It is very important that a candidate should apply to colleges that he/she has a good chance of getting into, instead of applying to colleges that they may never get into. This will help in reduction of cost as students will be applying to only those universities that they are highly likely to get into. In this project the algorithms used are Linear Regression, Random Forest, Decision tree, Artificial Neural Network (ANN). The prepared models work to a satisfactory level of accuracy and may be of great assistance to such people. This is a project with good future scope, especially for students of our age group who want to pursue their higher education in their dream college.

Problem Identified - 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.

Sara Aljasmi, Ali Bou Nassif, Ismail Shahin, Ashraf Elnagar [2]. This paper addresses machine learning models to predict the chance of a student to be admitted to a master's program. This will assist students to know in advance if they have a chance to get accepted. The machine learning models are multiple linear regression, k-nearest neighbor, random forest, and Multilayer Perceptron. Experiments show that the Multilayer Perceptron model surpasses other models. In this project the algorithms used are Multiple Linear Regression, K-Nearest Neighbor (KNN), Random Forest. And the accuracy of each algorithm is 0.0343 for Multi Linear Regression, 0.0363 for Random Forest, 0.0544 for K-Nearest Neighbor. And this model can be improved by more models can be conducted on more datasets to learn the model that gives the best performance.

Problems identified – This problem can lead to unstable regression model. In other words, any slight change in the data will lead to a huge change in the coefficients of the multiple linear regression model.

Acharyaet al [3]. proposed a comparative approach by developing four machine learning regression models: linear regression, support vector machine, decision tree and random forest for predictive analytics of graduate admission chances. Then compute error functions for the developed models and compare their performances to select the best performing model out of these developed models the linear regression is the best performing model with R2 score of 0.72.

Problems identified - linear regression to the shot at conceding graduate understudies in expert's projects as a rate. Be that as it may, no more models were performed.

Janani Pet al [4]. proposed a developed project uses machine learning technique specifically a decision tree algorithm based on the test attributes like GRE, TOEFL, CGPA, research papers etc. According to their scores the possibilities of chance of admit is calculated. The developed model has 73% accuracy.
NavoneelChakrabartyet al [5]. In this paper a comparison of different regression models. The developed models are gradient boosting regress or and linear regression model. Gradient boosting regress or have to score of 0.84. That surpassing the performance of linear regression model. They computed different other performance error metrics like mean absolute error, mean square error, and root mean square error. ChithraApoorva et al. [4] proposed different machine learning algorithms for predicting the chances of admission. The models are K- Nearest Neighbor and Linear Regression, Ridge Regression, Random Forest.
Problems Identified - These are trained by features have a high impact on the probability of admission. Out of the generated models the linear regression model has 79% accuracy.

2.2 REFERENCES

1. Acharya MS, Armaan A, Antony AS (2019) a comparison of regression models for prediction of graduate admissions. In: 2019 IEEE International conference on computational intelligence in data science (ICCIDS). IEEE

2. Janani P, HemaPriya V, MonishaPriya S, Prediction of MS Graduate Admissions using Decision Tree Algorithm, International Journal of Science and Research (IJSR) ISSN: 2319-7064 ResearchGate Impact Factor (2018): 0.28 | SJIF (2018): 7.426.

3. Gupta N, Sawhney A, Roth D (2016) Will I get in? Modeling the graduate admission process for American universities. In: 2016 IEEE 16th international conference on data mining workshops (ICDMW). IEEE.

4. NavoneelChakrabarty, Siddhartha Chowdhury, Srinibas Rana ons, A Statistical Approach to Graduate Admissions' Chance Prediction, in book: Innovations in Computer Science and Engineering (pp.333-340), march 2020.

5. Vandit Manish Jain, Rihaan Satia DOI: 10.46300/91013.2020.14.13

2.3 PROBLEM STATEMENT DEFINITION

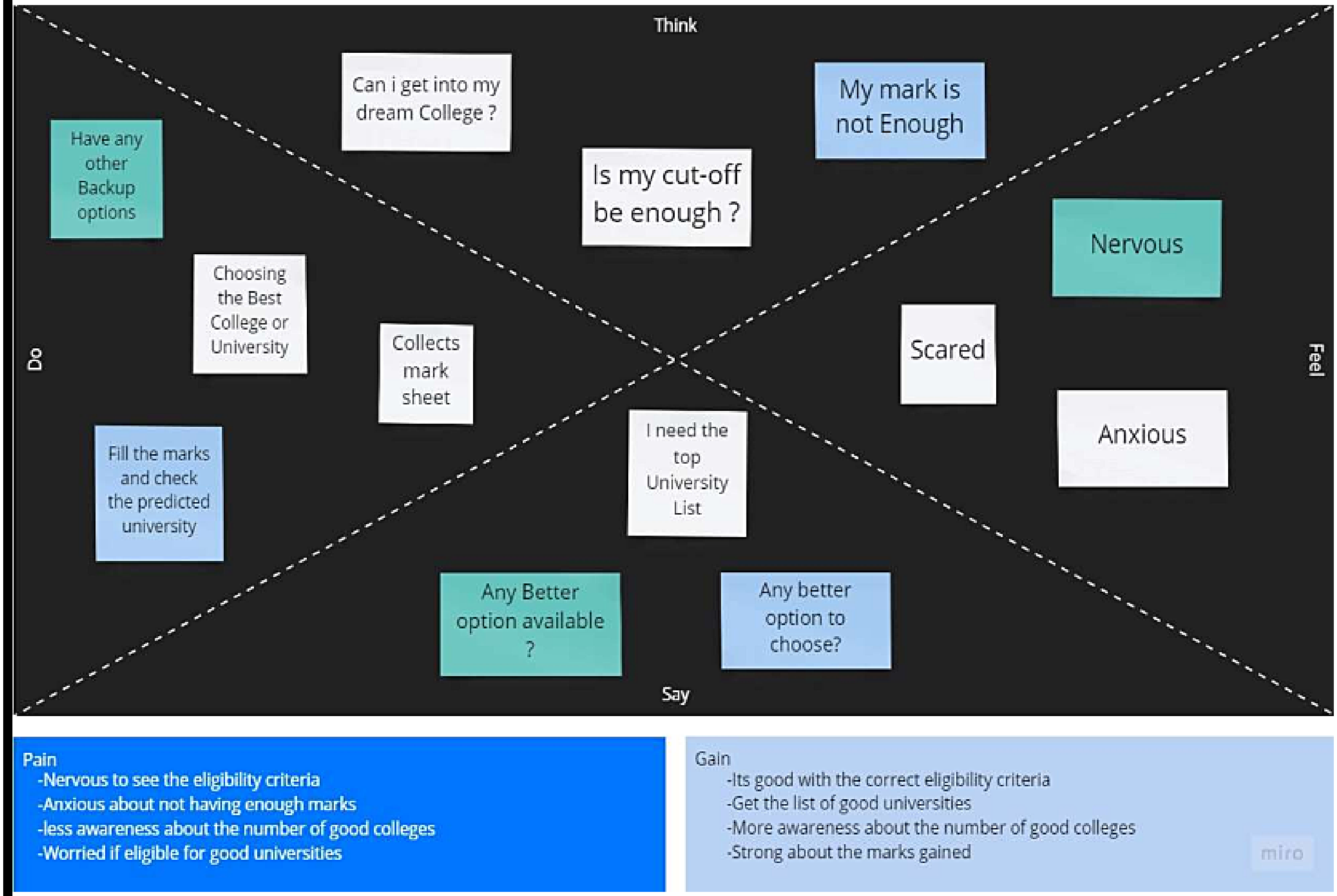
Every year thousands of college graduates apply for the master and PhD programs in US universities from all around the world. Applying to US universities is not an easy task, it involves many steps and procedures to follow. Choosing the right universities or colleges is definitely an another hurdle students have to face. Many students apply for the universities in which they have little chance of acceptance. This leads students of poor economic backgrounds to frustration and anxiety as they only lose surplus amount of money just for applying to those universities. This is because overall university application cost is not affordable for students with low economic backgrounds. US universities application cost for top level universities range from $70 to $90. In the same way total cost to send GRE scores to any individual University is $27 and cost of sending TOEFL Score to any individual university is $19. These stats show students have to throw away lots of hard works and hard-earned money for nothing it they got rejected in universities they have applied for.What if there is a system that could guide students and recommend best universities list and predict their admission chance in those universities according to their profile and scores.

CHAPTER 3

IDEATION & PROPOSED SOLUTION

Ideation is the process where you generate ideas and solutions through sessions such as Sketching, Prototyping, Brainstorming, Brainwriting, Worst Possible Idea, and a wealth of other ideation techniques. Ideation is also the third stage in the Design Thinking process. In this project the ideation phase consist of Empathy map canvas and Ideation and Brainstorming.

3.1 EMPATHY MAP CANVAS


Think
Can i get into my dream College ?
My mark is not Enough
Is my cut-off be enough ?
Have any other Backup options
Nervous
Choosing the Best College or University
Collects mark sheet
Do
Scared
Feel
Anxious
Fill the marks and check the predicted university
I need the top University List
Any Better option available ?
Any better option to choose?
Say
miro

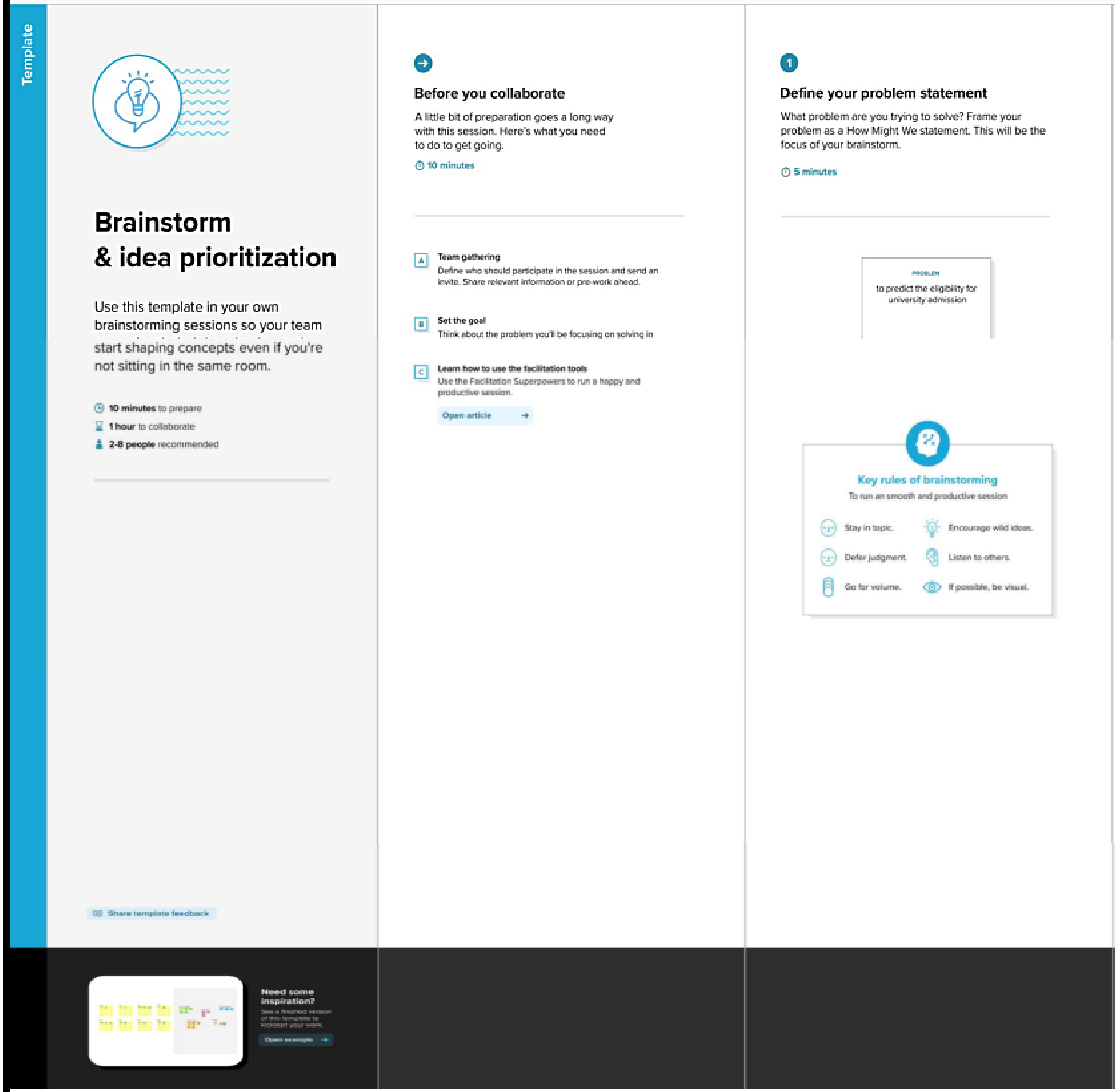
Pain
-Nervous to see the eligibility criteria
-Anxious about not having enough marks
-less awareness about the number of good colleges
-Worried if eligible for good universities

Gain
-Its good with the correct eligibility criteria
-Get the list of good universities
-More awareness about the number of good colleges
-Strong about the marks gained

Fig 3.1-Empathy Map Canvas

3.2 IDEATION & BRAINSTORMING

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



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 2-8 people recommended

Need some inspiration?

See a finished version of this template to kickstart your work.

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A. Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B. Set the goal
Think about the problem you'll be focusing on solving in

C. Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

1 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM
to predict the eligibility for university admission

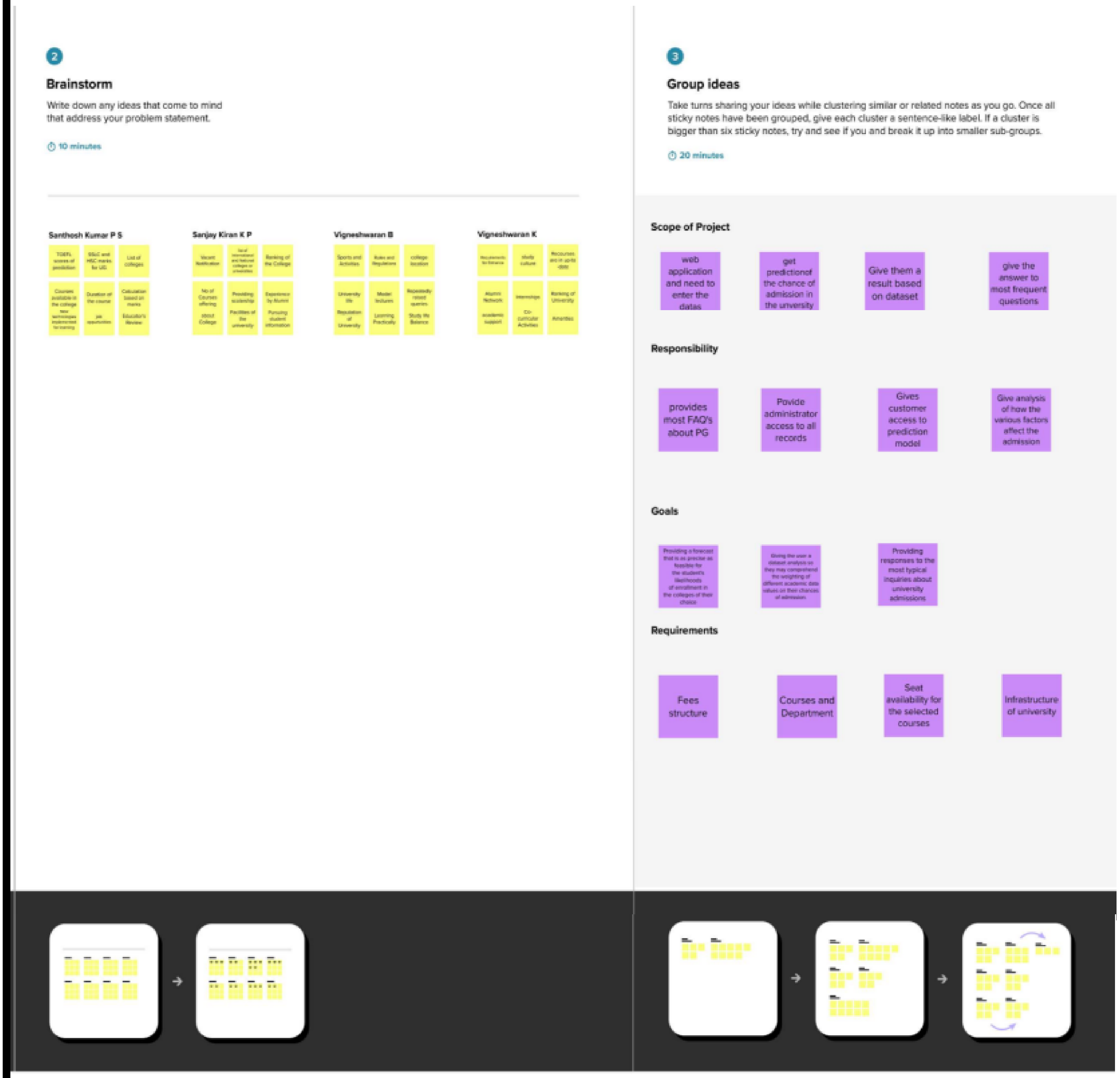
Key rules of brainstorming

To run an smooth and productive session

Stay in topic.	Encourage wild ideas.
Defer judgment.	Listen to others.
Go for volume.	If possible, be visual.

Fig 3.2-Brainstorm Techniques

Step-2: Brainstorm, Idea Listing and Grouping



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

Santhosh Kumar P S

TOEFL scores of prediction

SSLC and HSC marks for UG

List of colleges

Courses available in the college

New technologies implemented for training

Duration of the course

Job opportunities

Calculation based on marks

Educator's Review

Sanjay Kiran K P

Vacant Notification

list of international and National colleges or universities

Ranking of the College

No of Courses offering

about College

Providing scolership

Facilities of the university

Experience by Alumni

Pursuing student information

Vigneshwaran B

Sports and Activities

Rules and Regulations

college location

University life

Reputation of University

Model lectures

Learning Practically

Repeatedly raised queries

Study life Balance

Vigneshwaran K

Requirements for Entrance

Study culture

Resources are in up-to date

Alumni Network

academic support

Internships

Co-curricular Activities

Ranking of University

Amenities

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Scope of Project

web application and need to enter the datas

get predictionof the chance of admission in the unversity

Give them a result based on dataset

give the answer to most frequent questions

Responsibility

provides most FAQ's about PG

Povide administrator access to all records

Gives customer access to prediction model

Give analysis of how the various factors affect the admission

Goals

Providing a forecast that is as precise as feasible for the student's likelihoods of enrollment in the colleges of their choice

Giving the user a dataset analysis so they may comprehend the weighting of different academic data values on their chances of admission

Providing responses to the most typical inquiries about university admissions

Requirements

Fees structure

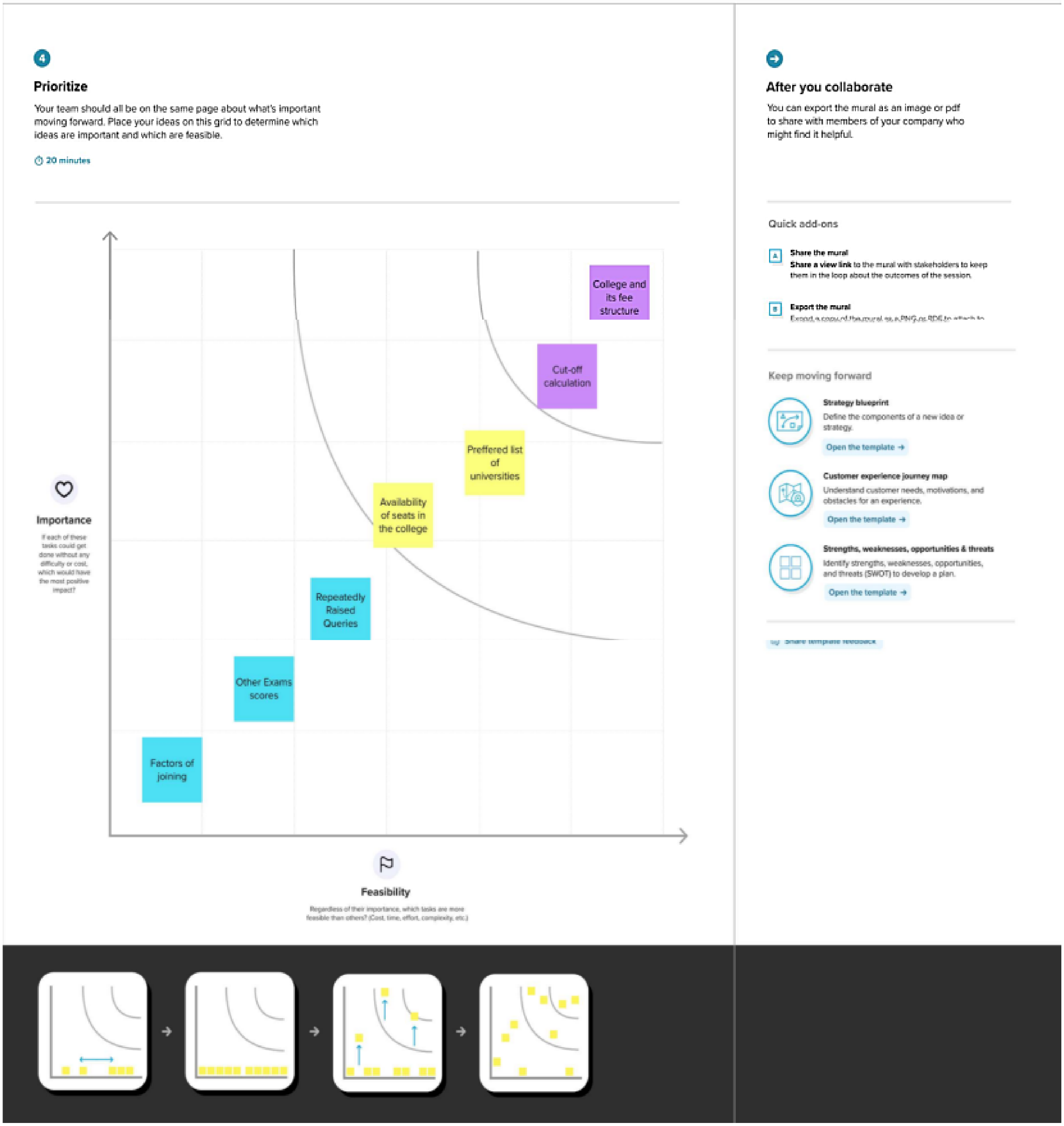
Courses and Department

Seat availability for the selected courses

Infrastructure of university

Fig 3.3-Brainstorm, Idea Listing and Grouping

Step-3: Idea Prioritization



4

Prioritize

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

20 minutes

College and its fee structure
Cut-off calculation
Preffered list of universities
Availability of seats in the college
Repeatedly Raised Queries
Other Exams scores
Factors of joining

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

Share the mural
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

Export the mural
Export a copy of the mural as a PNG or PDF to attach to

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

Share template feedback

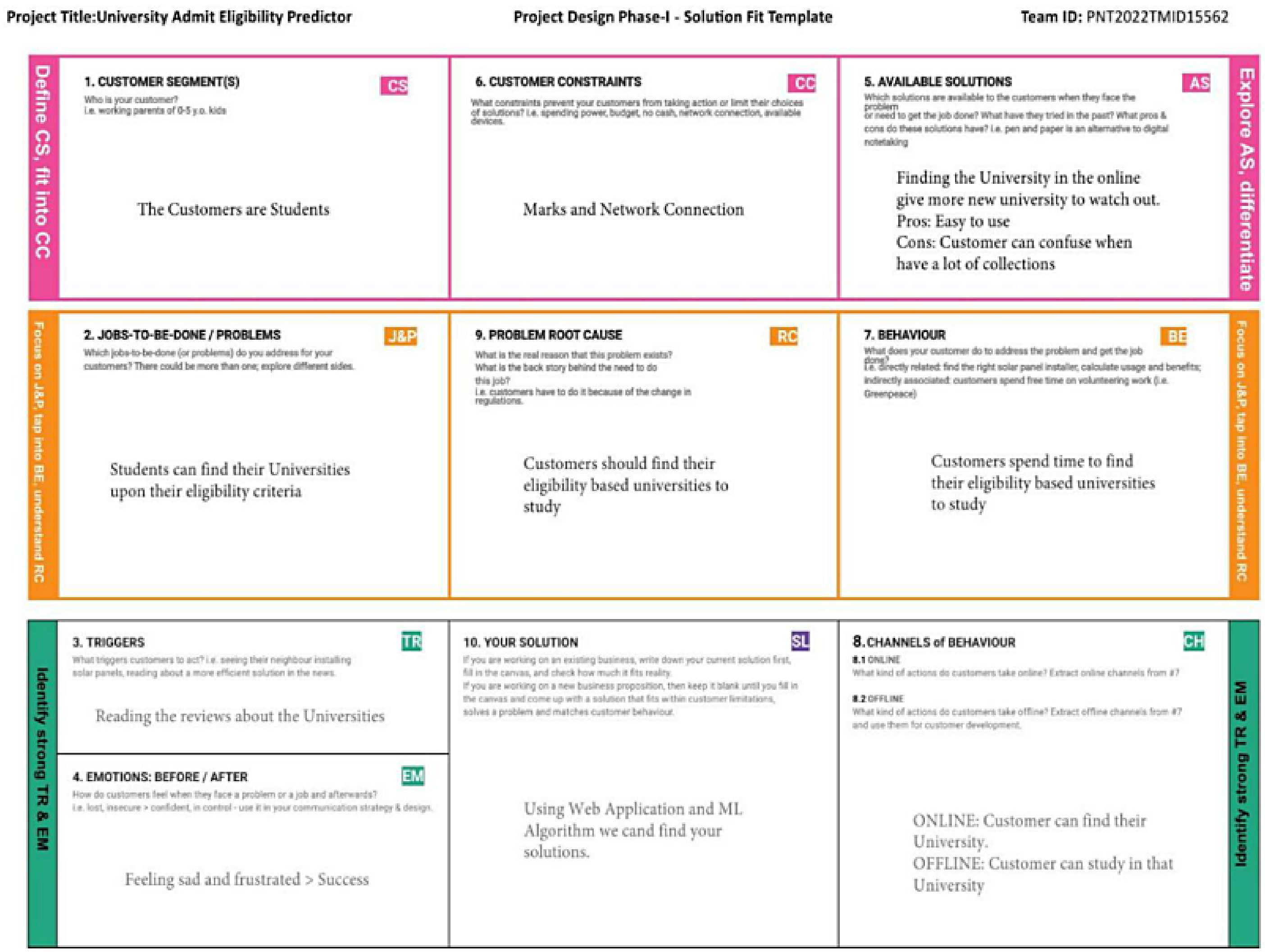
Fig 3.4-Idea Prioritization

3.3 PROPOSED SYSTEM

S.NO	PARAMETER	DESCRIPTION
1.	Problem Statement (Problem to be solved)	Students can enter their grades and personal information in the University Admission Predictor System, a web-based application system. This aids in predicting their college admissions. Administrators can add information about the college and the batch specifics. Utilizing this application makes entry seat allocation simpler and efficient. The entrance's computerization is the project's principal benefit.
2.	Idea / Solution description	Students can register their grades and personal information on the web-based application system called University Admission Predictor System. This aids in forecasting their college enrolment. A manager may add the information about the college and the cohort. Using this Application, the allocation of the front seat becomes more effective and simpler.
3.	Novelty / Uniqueness	The project website can include a summary of the many amenities offered by the institutions as well as directions to get there. additional get options for financial aid and scholarships assistance.
4.	Social Impact / Customer Satisfaction	This method will lessen student’s anxiety as well as their worry about being admitted to the university of their dreams. And this method will yield better outcomes for the students, determining whether they will be admitted to the institution or not.
5.	Business Model (Revenue Model)	In addition, revenue can be generated by advertising the Entrance exam coaching Centre and the University shall fund the website in order to maintain and progress it.

6.	Scalability of the Solution	A conversation room with candidates, instructors, current students, and alumni will be available in a future update. It is scalable for colleges everywhere in the world.

3.4 PROBLEM SOLUTION FIT



Project Title:University Admit Eligibility Predictor

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID15562

1. CUSTOMER SEGMENT(S)

Who is your customer?
i.e. working parents of 0-5 y.o. kids

The Customers are Students

6. CUSTOMER CONSTRAINTS

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Marks and Network Connection

5. AVAILABLE SOLUTIONS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Finding the University in the online give more new university to watch out.
Pros: Easy to use
Cons: Customer can confuse when have a lot of collections

2. JOBS-TO-BE-DONE / PROBLEMS

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

Students can find their Universities upon their eligibility criteria

9. PROBLEM ROOT CAUSE

What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

Customers should find their eligibility based universities to study

7. BEHAVIOUR

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Customers spend time to find their eligibility based universities to study

3. TRIGGERS

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

Reading the reviews about the Universities

4. EMOTIONS: BEFORE / AFTER

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Feeling sad and frustrated > Success

10. YOUR SOLUTION

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

Using Web Application and ML Algorithm we cand find your solutions.

8. CHANNELS of BEHAVIOUR

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

ONLINE: Customer can find their University.
OFFLINE: Customer can study in that University

Define CS, fit into CC
CS
CC
AS
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
J&P
RC
BE
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
TR
EM
SL
CH
Identify strong TR & EM

Fig 3.5-Solution Fit Model

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

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 2. Registration through Gmail
FR-2	User Confirmation	3. Confirmation via Email 4. Confirmation via OTP
FR-3	User Details	Submit the documents 5. GRE or/and TOEFL scoresheet 6. Curriculum Vitae (CV) 7. Statement of Purpose (SoP) 8. Letter of Recommendation
FR-4	User Requirements	9. Upload all the relevant documents in the appropriate location in the website 10. Based on the uploads, the system would scrape all the necessary information 11. The list of all possible university for the candidate would be displayed based on the scraped information

4.2 NON-FUNCTIONAL REQUIREMENT

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

NFR No.	NON-FUNCTIONAL REQUIREMENT	DESCRIPTION
NFR-1	Usability	1. The system doesn't expect any technical pre-requisite from the user i.e.; even the naïve user can access it 2. The UI would focus on recognize over recall 3. User friendly 4. Reduced focus on Short Term memory load Focus on Internal Locus of Control 5. The page would not take a lot of time to load the content and display them (< 30 seconds) 6. The fields in the site would be self-explanatory
NFR-2	Security	1. Only the authenticated user could utilize the services of the site. 2. Database should be backed up every hour 3. Under any error, the system should be able to come back to normal operation in under an hour.
NFR-3	Reliability	1. The system would always strive for maximum reliability due to the importance of data and damages thar could be cause by incomplete and incorrect data 2. The system will run 7 days a week, 24 hours a day
NFR-4	Performance	1. The website can efficiently handle the traffic by service the request as soon as possible 2. Viewing this webpage using a 56-kbps modem connection would not exceed 30 seconds (quantitatively, the mean time)
NFR-5	Availability	1. Minimal data redundancy 2. Less prone to errors 3. Fast and efficient 4. The system will run 7 days a week, 24 hours a day
NFR-6	Scalability	1. Since an academic portal is crucial to the courses that use it, it is crucial that a sizable number of users be able to access the system at the same time. 2. The admission season is probably when the system will be under the most strain. 3. It must therefore be able to manage numerous concurrent users.

CHAPTER 5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

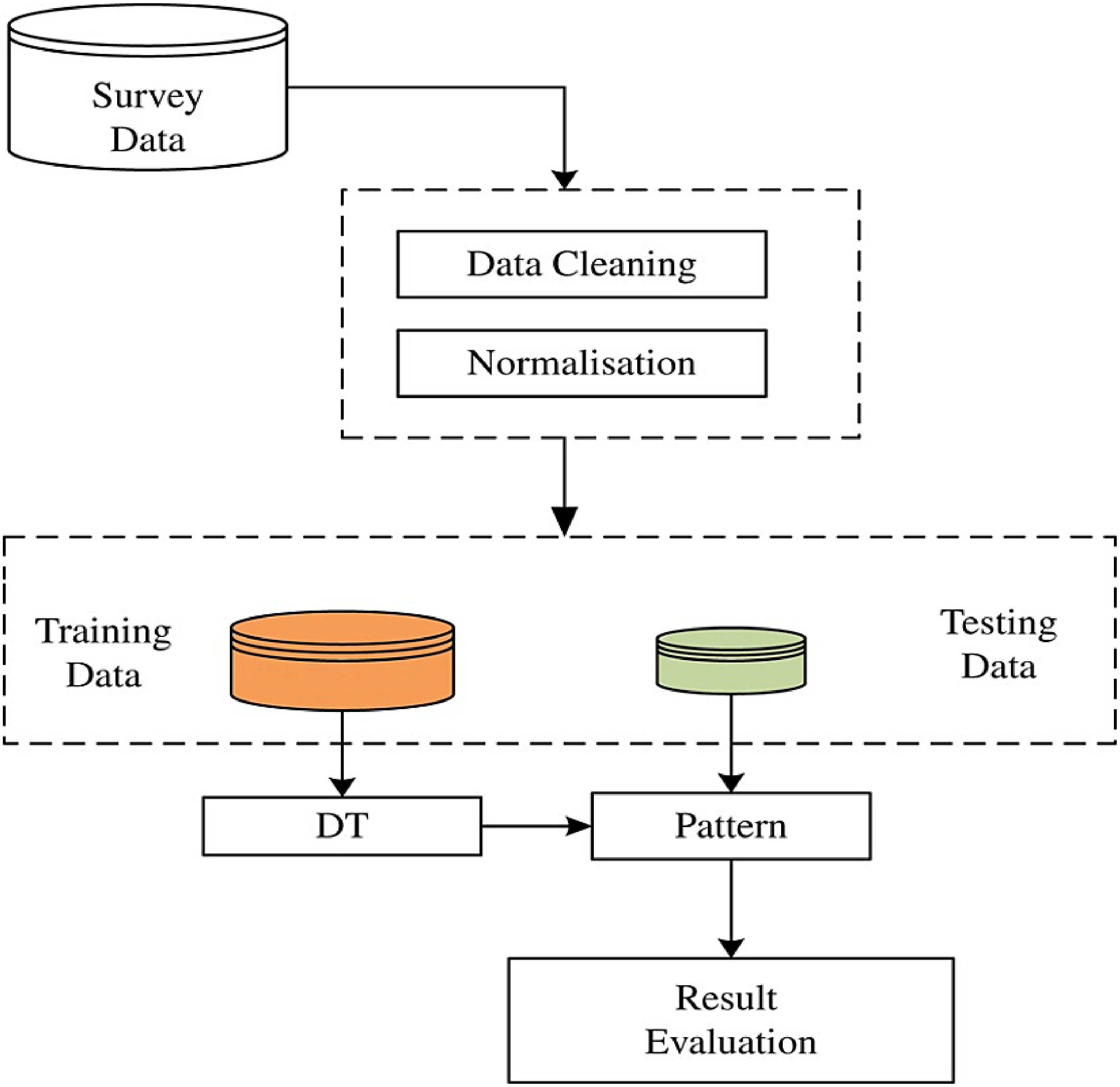


Fig 5.1-Data Flow Diagram

5.2 SOLUTIONAL & TECHNICAL ARCHITECTURE

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2.

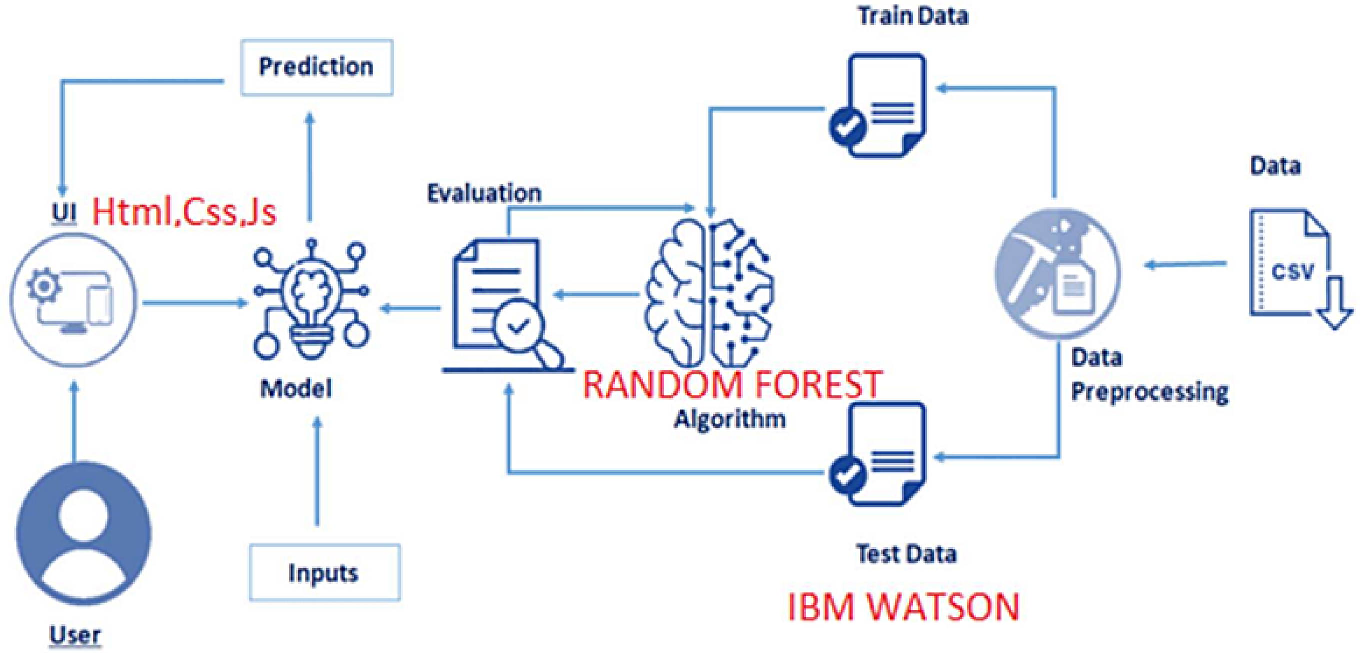


Fig 5.2-Technical Architecture

Table-1: Components & Technologies:

S.NO.	COMPONENT	TECHNOLOGY
1.	User Interface	Html, CSS, is
2.	Application Logic-1	Python
3.	Application Logic-2	IBM Watson
4.	Machine Learning Model	Random Forest

Table-2: Application Characteristics:

S.NO.	CHARACTERISITICS	TECHNOLOGY
1.	Open-Source Frameworks	Flask
2.	Performance	It can handle about 100 requests per second

5.3 USER STORIES:

USER TYPE	FUNCTIONAL REUIREMENT	USER STORY NO.	USER STORY/ TASK	ACCEPTANCE CRITERIA	PRIORITY	RELEASE
Student (Mobile user)	Registration	USN-1	As a student, I can register for the application by entering my email, password, and confirming my password.	I can access my account/ dashboard	High	Sprint -1
		USN -2	As a student, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-2
		USN -3	As a student, I can register for the application through Gmail.		Medium	Sprint -1

	Login	USN -4	As a student, I can log into the application by entering email & password.		High	Sprint -2
	Dashboard	USN -5	While entering the home page, I can see profile, student details and logout.		High	Sprint-3
Student (Web user)	Registration	USN -6	As a student, I can register via website using Email and password	I can receive confirmation Email & click confirm	High	Sprint -1
Student (Web user)	Login	USN -7	As a student, I can login to the website by entering email & password		High	Sprint-2
Administrator	Home Page	USN-8	Enter all the marks as specified column to check	I can able to see list of available universities	High	Sprint-3

			availability			
	Evaluation	USN -9	Based on the accuracy level, the result will be Sorted on ascending order		Medium	Sprint-4
	Results	USN-10	As a student, I can choose eligible of my university		High	Sprint-4

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Functional Requirement	User Story Number	User Story/ Task	Story point	Priority	Team Members
Sprint -1		USN – 1	As a user, I can register for the application by entering my email, password, and confirming my password.	User Registration	High	2
Sprint -1		USN – 2	As a user, I will receive confirmation email once I have registered for the application		High	1
Sprint - 2		USN – 3	As a user, I can check the eligibility criteria for various universities by uploading the necessary	2	Low	2

			documents			
Sprint - 3		USN – 4	As a user, I can register for the desired university through Gmail and can also upload further course completion documents if necessary	2	Medium	2
Sprint - 4	User login	USN – 5	As a user, I can log into the application by entering email & password	1	High	2
	Dashboard		As a user, I can log into the application by entering email & password			4

6.2 SPRINT DELIVERY SCHEDULE

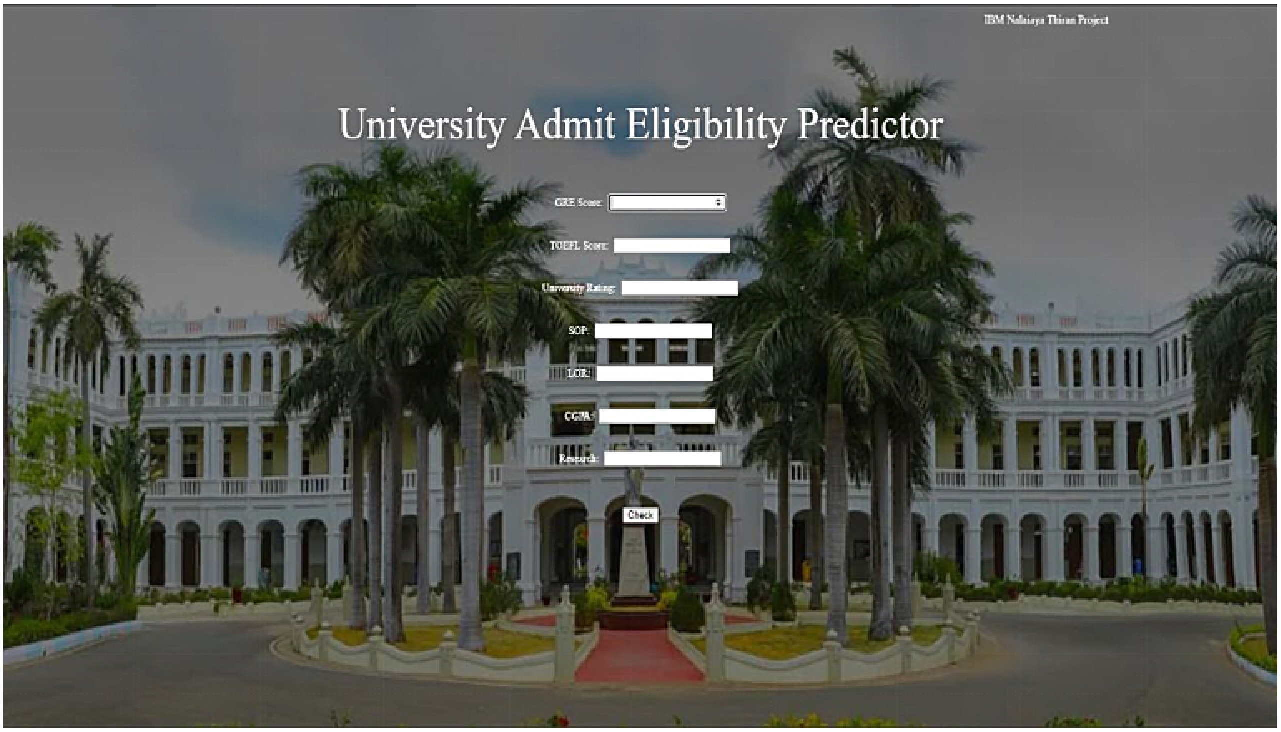
Sprint	Total Story points	Duration	Sprint Start date	Sprint End date	Story points Completed(as on planned End date)	Sprint Release date
Sprint – 1	20	6 days	24 Oct 2022	29 Oct 2022	20	30 Oct 2022
Sprint – 2	20	6 days	31 Oct 2022	05 Nov 2022	20	06 Nov 2022
Sprint – 3	20	6 days	07 Nov 2022	12 Nov 2022	15	13 Nov 2022
Sprint - 4	20	6 days	14 Nov 2022	19 Nov 2022	25	20 Nov 2022

CHAPTER 7

CODING & SOLUTION

7.1 FEATURE 1 (RANDOM FOREST)

The new function will forecast the likelihood of receiving university admission. The functionality was created using html code, with app.py serving as the backend.



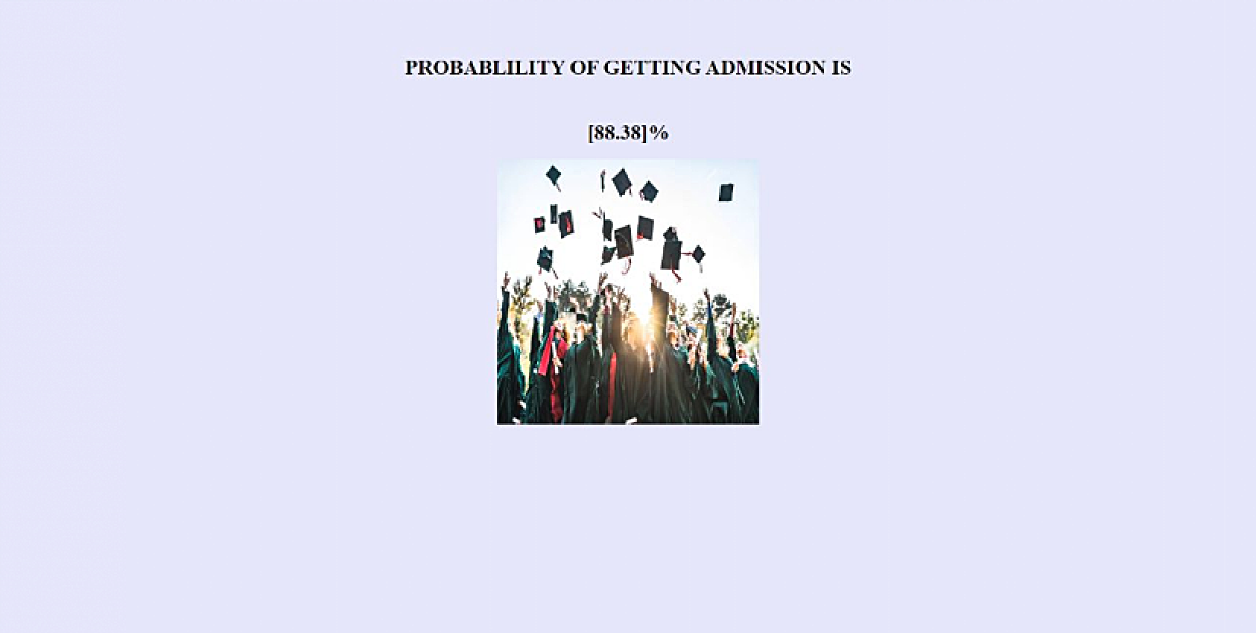
Source Code:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Flaskimio</title>
<style>
body {
background: linear-gradient(
                rgba(10,10,10, .50),
                rgba(10,10,10, .50)),
                url(
'https://images.collegedunia.com/public/image/1a09101fad610b0e76351db713cf8a4b.png?tr=w-800,h-533,c-
force');
                background-repeat: no-repeat;
                background-size: cover;
}
.container {
  border: 2px solid #ccc;
  padding: 10px;
  width: 20em;
height:21em;
background-color:white;
}


.hello{
opacity: 0.5;
}
</style>
</head>

<body>
  <marquee style="color: white;">IBM Nalaiaya Thiran Project</marquee>
  <br>
  <br>
  <br>
<center><p style="font-size:60px;color:white;">University Admit Eligibility Predictor</p></center>
<form action="/val" method="post"><center>
  <label for="GRE Score" style="font-size:15px;color:white;" >GRE Score:</label>
  <input type="number" id="GRE Score" name="GRE Score"><br><br><br>


  <label for="TOEFL Score" style="font-size:15px;color:white;">TOEFL Score:</label>
  <input type="number" id="TOEFL Score" name="TOEFL Score">
<br>
<br>
<br>
  <label for="University Rating" style="font-size:15px;color:white;">University Rating:</label>
  <input type="number" id="University Rating" name="University Rating">
<br>
<br>
<br>
  <label for="SOP" style="font-size:15px;color:white;">SOP:</label>
  <input type="number" id="SOP" name="SOP">
```

```
<br>
<br>
<br>
  <label for="LOR" style="font-size:15px;color:white;">LOR:</label>
  <input type="number" id="LOR" name="LOR">
<br>
<br>
<br>
  <label for="CGPA" style="font-size:15px;color:white;">CGPA:</label>
  <input type="number" id="CGPA" name="CGPA">
<br>
<br>
<br>
  <label for="Research" style="font-size:15px;color:white;">Research:</label>
  <input type="number" id="Research" name="Research">
<br>
<br>
<br>

<br></center>
  <center><button type="submit">Check</button></center>
</form>
</body>
</html>
```

7.2 FEATURE 2 (FLASK CONNECTION)

The new feature will forecast the low likelihood of university acceptance. The function was created using HTML code and used app.py as the backend.



PROBABLILITY OF GETTING ADMISSION IS

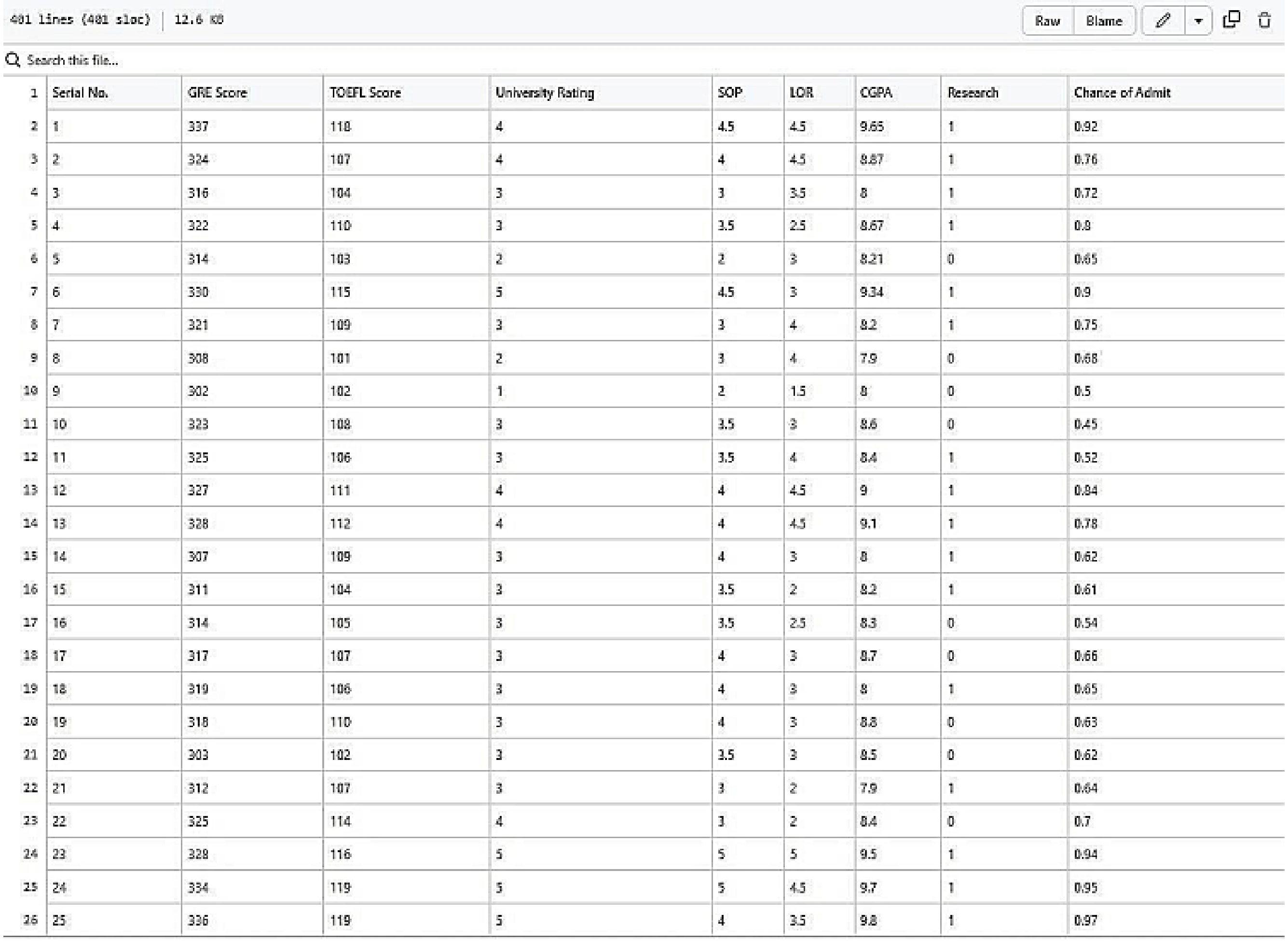
[88.38]%

Source Code:

```
<html>
<head>
<style>
body {
  background-color: #E6E6FA;}
</style>
</head>
<body >
<br>
<br>
<br>
<center><h1>PROBABLILITY OF GETTING ADMISSION IS </h1></center>
<center><h1>{{answer2}}%</h1></center>
<center><img alt="Qries" src="https://media.istockphoto.com/id/1066324992/photo/graduation-day.jpg?s=612x612&w=0&k=20&c=cleRpjTZbo430AbH-luZFYMMNqPwhwyTnFgWMbi_AiI=" width="400" height="400"></center>
</body>
</html>
```

7.3 DATABASE SCHEMA

This study made use of the database Admission Predict.csv. The database's sample screenshots include:



Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
1	337	118	4	4.5	4.5	9.65	1	0.92
2	324	107	4	4	4.5	8.87	1	0.76
3	316	104	3	3	3.5	8	1	0.72
4	322	110	3	3.5	2.5	8.67	1	0.8
5	314	103	2	2	3	8.21	0	0.65
6	330	115	5	4.5	3	9.34	1	0.9
7	321	109	3	3	4	8.2	1	0.75
8	308	101	2	3	4	7.9	0	0.68
9	302	102	1	2	1.5	8	0	0.5
10	323	108	3	3.5	3	8.6	0	0.45
11	325	106	3	3.5	4	8.4	1	0.52
12	327	111	4	4	4.5	9	1	0.84
13	328	112	4	4	4.5	9.1	1	0.78
14	307	109	3	4	3	8	1	0.62
15	311	104	3	3.5	2	8.2	1	0.61
16	314	105	3	3.5	2.5	8.3	0	0.54
17	317	107	3	4	3	8.7	0	0.66
18	319	106	3	4	3	8	1	0.65
19	318	110	3	4	3	8.8	0	0.63
20	303	102	3	3.5	3	8.5	0	0.62
21	312	107	3	3	2	7.9	1	0.64
22	325	114	4	3	2	8.4	0	0.7
23	328	116	5	5	5	9.5	1	0.94
24	334	119	5	5	4.5	9.7	1	0.95
25	336	119	5	4	3.5	9.8	1	0.97

Fig 7.3.1 Database schema

CHAPTER 8

TESTING

8.1 TEST CASES

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		7
Customer Application	51	0	0	51
Security	2		0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.2 USER ACCEPTANCE TESTING

PURPOSE OF DOCUMENT:

This document's goal is to provide a brief explanation of the University Admit Eligibility Predictor project's test coverage and open issues as of the project's release for user acceptance testing (UAT).

DEFECT ANALYSIS:

This report lists the number of bugs that have been fixed or closed at each severity level, along with how they were fixed.

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

TEST CASE ANALYSIS

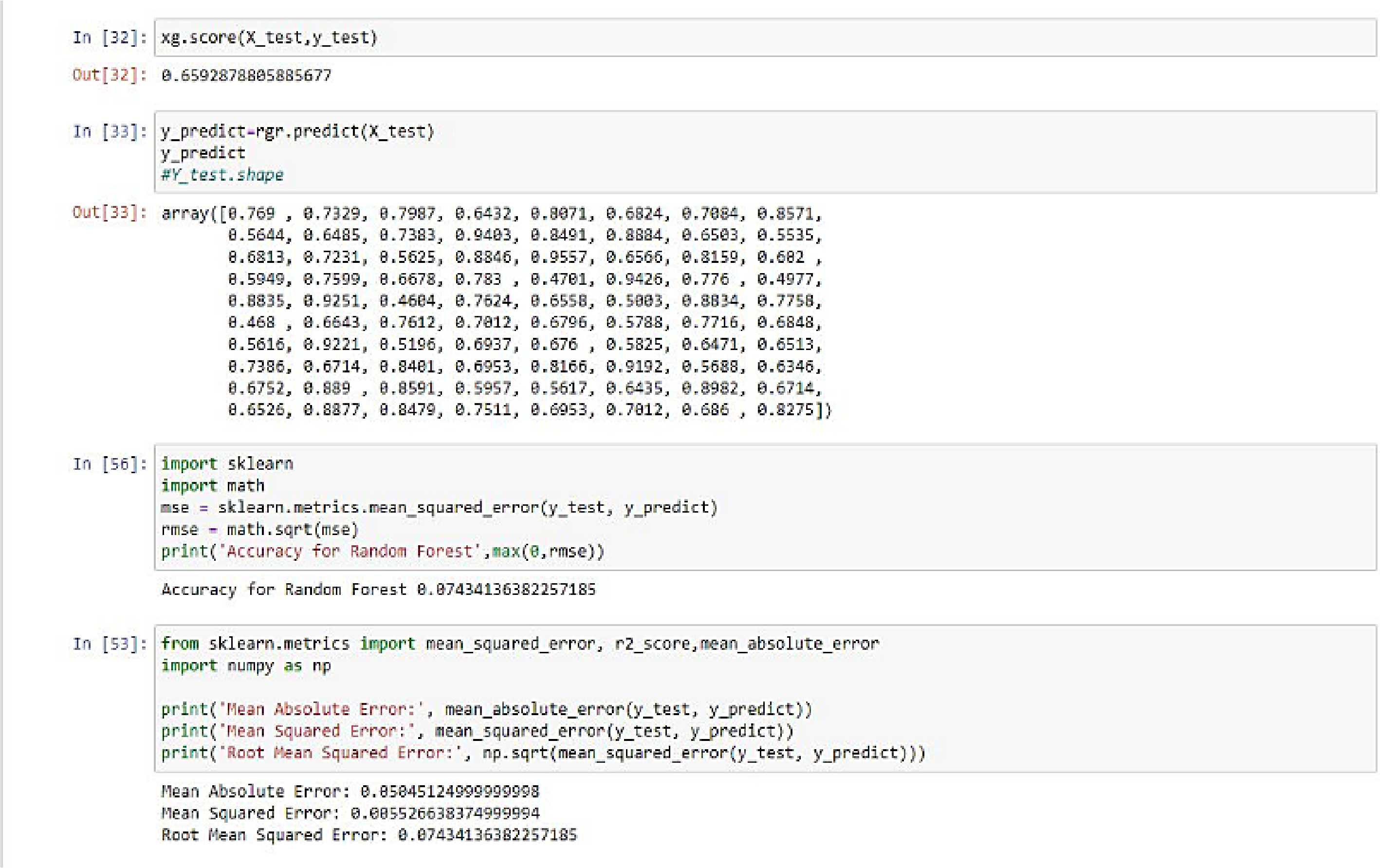
The number of test cases that have succeeded, failed, and not been tested is displayed in this report.

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

CHAPTER 9

RESULTS

9.1 PERFORMANCE METRICS



```
In [32]: xg.score(X_test,y_test)

Out[32]: 0.6592878805885677

In [33]: y_predict=rgr.predict(X_test)
         y_predict
         #Y_test.shape

Out[33]: array([0.769 , 0.7329, 0.7987, 0.6432, 0.8071, 0.6824, 0.7084, 0.8571,
                0.5644, 0.6485, 0.7383, 0.9403, 0.8491, 0.8884, 0.6503, 0.5535,
                0.6813, 0.7231, 0.5625, 0.8846, 0.9557, 0.6566, 0.8159, 0.682 ,
                0.5949, 0.7599, 0.6678, 0.783 , 0.4701, 0.9426, 0.776 , 0.4977,
                0.8835, 0.9251, 0.4604, 0.7624, 0.6558, 0.5003, 0.8834, 0.7758,
                0.468 , 0.6643, 0.7612, 0.7012, 0.6796, 0.5788, 0.7716, 0.6848,
                0.5616, 0.9221, 0.5196, 0.6937, 0.676 , 0.5825, 0.6471, 0.6513,
                0.7386, 0.6714, 0.8401, 0.6953, 0.8166, 0.9192, 0.5688, 0.6346,
                0.6752, 0.889 , 0.8591, 0.5957, 0.5617, 0.6435, 0.8982, 0.6714,
                0.6526, 0.8877, 0.8479, 0.7511, 0.6953, 0.7012, 0.686 , 0.8275])

In [56]: import sklearn
         import math
         mse = sklearn.metrics.mean_squared_error(y_test, y_predict)
         rmse = math.sqrt(mse)
         print('Accuracy for Random Forest',max(0,rmse))

         Accuracy for Random Forest 0.07434136382257185

In [53]: from sklearn.metrics import mean_squared_error, r2_score,mean_absolute_error
         import numpy as np

         print('Mean Absolute Error:', mean_absolute_error(y_test, y_predict))
         print('Mean Squared Error:', mean_squared_error(y_test, y_predict))
         print('Root Mean Squared Error:', np.sqrt(mean_squared_error(y_test, y_predict)))

         Mean Absolute Error: 0.05045124999999998
         Mean Squared Error: 0.005526638374999994
         Root Mean Squared Error: 0.07434136382257185
```

CHAPTER 10

ADVANTAGES & DISADVANTAGES

10.1 ADVANTAGES

- Easy to predict the eligible University
- Identify to geographically scattered student
- Reducing time in activities
- Centralized data handling
- Paperless admission with reduced man power
- Operational efficiency

10.2 DISADVANTAGES

- Low computer literacy and Internet access
- Online applications make it easier for fraudsters to manipulate the application process.
- A task that requires financial and infrastructural resources
- Unintentional system failures or server crashes may disrupt the entire online admission process.

CHAPTER 11

CONCLUSION

This system, which we constructed for the first time in Python employing ML algorithms and other front end languages like html, css, and java script, has proven to be more challenging than anticipated. While it can seem straightforward to complete a few forms and process the data, there is much more that goes into choosing applications than this. Every time improvements and new features were implemented, ideas for more features or ways to make the system easier to use became obvious. A project in and of itself, balancing the completion of these required features with the suggestions for improvement as well as remembering everything that had to be done, was made possible by the fact that adding one feature made it possible to add another necessary feature. . Finding out what needs to be debugged can often be a very simple task compared to debugging itself. Since so many components of the admissions system are interconnected, if an error is found on one page, it may be a display error, a problem with how the information is read from the database, or even a problem with how the information was initially stored in the database. Each time, all three must be verified. When the apparent reason of an issue is not immediately clear, this slows down the process and can be frustrating. The language used must be straightforward and simple to comprehend, and compatibility is crucial. . The mobility of this system would not have been able to duplicate if it had not been created as a fully web-based application.

CHAPTER 12

FUTURE SCOPE

The future scope of this project is very broad.

Few of them are:

- 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.

CHAPTER 13

APPENDIX

Source Code:

Connectivity.py:

```
import pickle
loaded_reg = pickle. load(open('randomreg_chronic', 'rb'))
loaded_scaler = pickle. load(open('scaler_admin', 'rb'))

import numpy as np
import pandas as pd

from flask import Flask, request, redirect, render_template
app = Flask(__name__)
@app.route("/",methods=['GET', 'POST'])
def index():
    return render_template('index.html')
@app.route("/val",methods=['POST'])

def val():
    test=[]
    if request.method == 'POST':
        test.append(int(request.form.get("GRE Score")))
        test.append(int(request.form.get("TOEFL Score")))
        test.append(int(request.form.get("University Rating")))
        test.append(int(request.form.get("SOP")))
        test.append(int(request.form.get("LOR")))
        test.append(int(request.form.get("CGPA")))
        test.append(int(request.form.get("Research")))
    print(test)
    test_df=pd.DataFrame(test)
    test_df=np.array(test_df).reshape(1, -1)

    ans1=loaded_scaler.transform(test_df)
    print(ans1)
    ans2=loaded_reg.predict(ans1)
    print(ans2)

    return render_template('rename.html',answer2=ans2)

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

Kernal.py:

```
# This Python 3 environment comes with many helpful analytics libraries installed
# It is defined by the kaggle/python docker image: https://github.com/kaggle/docker-python
# For example, here's several helpful packages to load in

import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

# Input data files are available in the "../input/" directory.
# For example, running this (by clicking run or pressing Shift+Enter) will list all files under the input directory

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# Any results you write to the current directory are saved as output.

import pandas as pd
import matplotlib.pyplot as plt
admission = pd.read_csv('C:/Users/Sinegalatha/Downloads/sanjaykiren/archive/Admission_Predict.csv')
admission.head() # to see the top five records of the data sets
admission.shape # to  see what is the shape of data_set our data set has 400 records and 9 fields
admission.columns #to see the name of the fields
admission.describe() # to see the mathematical values of the data sets i.e mean, standar_deviation,
minimum_value,maximum_value, counts etc.
admission.info() #to see the type of values in every fields i.e int ,float etc
admission.isnull().sum() # to see that if dataset has any null_values or not
# the dataset has no null_values so there is no need to fill null_values
# now preparing the input data_set and out_labels
X=admission.drop(['Serial No.','Chance of Admit '],axis=1) #input data_set
X.shape
y=admission['Chance of Admit '] #output labels
y.shape
admission.sample(5)
plt.scatter(admission['GRE Score'],admission['CGPA'])
plt.title('CGPA vs GRE Score')
plt.xlabel('GRE Score')
plt.ylabel('CGPA')
plt.show()
plt.scatter(admission['CGPA'],admission['SOP'])
plt.title('SOP for CGPA')
plt.xlabel('CGPA')
plt.ylabel('SOP')
plt.show()
```

```
admission[admission.CGPA >= 8.5].plot(kind='scatter', x='GRE Score', y='TOEFL Score',color="BLUE")

plt.xlabel("GRE Score")
plt.ylabel("TOEFL SCORE")
plt.title("CGPA>=8.5")
plt.grid(True)

plt.show()
admission["GRE Score"].plot(kind = 'hist',bins = 200,figsize = (6,6))

plt.title("GRE Scores")
plt.xlabel("GRE Score")
plt.ylabel("Frequency")

plt.show()

p = np.array([admission["TOEFL Score"].min(),admission["TOEFL Score"].mean(),admission["TOEFL
Score"].max()])
r = ["Worst","Average","Best"]
plt.bar(p,r)

plt.title("TOEFL Scores")
plt.xlabel("Level")
plt.ylabel("TOEFL Score")

plt.show()

g = np.array([admission["GRE Score"].min(),admission["GRE Score"].mean(),admission["GRE
Score"].max()])
h = ["Worst","Average","Best"]
plt.bar(g,h)
plt.title("GRE Scores")
plt.xlabel("Level")
plt.ylabel("GRE Score")

plt.show()
import seaborn as sns

plt.figure(figsize=(10, 10))

sns.heatmap(admission.corr(), annot=True, linewidths=0.05, fmt= '.2f',cmap="magma")

plt.show()
admission.Research.value_counts()

sns.countplot(x="University Rating",data=admission)
admission.Research.value_counts()

sns.countplot(x="University Rating",data=admission)
sns.barplot(x="University Rating", y="Chance of Admit ", data=admission)
#splittin the input data(x) and output labels(y) into train data and test data
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.20) # test_size defins the volume of train
data and test data here 0.2 means 20% of the data belongs to the test data
X_train.shape
X_test.shape
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
X_train[X_train.columns] = scaler.fit_transform(X_train[X_train.columns])
X_test[X_test.columns] = scaler.transform(X_test[X_test.columns])
X_train.head()
```

```
import pickle
pickle. dump(scaler, open('scaler_admin', 'wb'))
from sklearn.ensemble import RandomForestRegressor
rgr=RandomForestRegressor()
rgr.fit(X_train,y_train)
import pickle
pickle. dump(rgr, open('randomreg_chronic', 'wb'))
rgr.score(X_test,y_test)
pip install xgboost
import xgboost as xgb
xg = xgb.XGBRegressor()
xg.fit(X_train,y_train)
xg.score(X_test,y_test)
y_predict=rgr.predict(X_test)
y_predict
#Y_test.shape
from sklearn.metrics import mean_squared_error, r2_score,mean_absolute_error
import numpy as np

print('Mean Absolute Error:', mean_absolute_error(y_test, y_predict))
print('Mean Squared Error:', mean_squared_error(y_test, y_predict))
print('Root Mean Squared Error:', np.sqrt(mean_squared_error(y_test, y_predict)))
```

GITHUB AND PROJECT DEMO LINK:

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-21365-1659778700

PROJECT DEMO LINK:

https://drive.google.com/file/d/1U1kaQNQUzWrJ4zfk3z-ZVHBDY69-VU9x/view?usp=share_link