

PROJECT REPORT

PROJECT TITLE: University Admit Eligibility Predictor

TEAM ID: PNT2022TMID14375

TEAM MEMBERS:

NANDHINI R

NIVETHITHA A S

PRAJNA A

PREETHI S

1. INTRODUCTION

1.1 Project Overview

In the present conditions, students regularly have difficulty finding a fitting institution to pursue higher studies based on their profile. There are some advisory administrations and online apps that recommend universities but they ask huge consultancy fees and online apps are not accurate. So, the aim of this research is to develop a model that predict the percentage of chances into the university accurately. This model provides also the analysis of scores versus chance of prediction based on historical data so that students can understand whether their profile is suitable or not. The proposed model uses linear regression and random forest algorithms but cat boost algorithm is giving highest accuracy.

1.2 Purpose

The primary purpose of the University Admit Eligibility Predictor is to help the student to find the chance to get their desired University and the percentage of getting them inside the University with surity. This give 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.

2. LITERATURE SURVEY

2.1 Existing problem

Decision making by applying data mining methods is being used in many service organizations. Educational bodies gradually started to use the business intelligence techniques to identify the current progress in their institutions. Numerous factors which have an impact in academia will be vivid to the educationalists while applying data mining techniques on the academic data. By employing the data mining methodologies, we could identify different patterns which aid institutions to take strategic decisions to improve the students' academic performance. Potential graduate students will have a dilemma on identifying the universities for their post graduate admissions and on the other hand an average graduate student would be uncertain on getting post graduate admission in a reputed university based on their academic scores. In this study, we applied the classification techniques such as Logistic Regression, KNN Classification, Support Vector Classification, Naive Bayes Classification, Decision Tree Classification and Random Forest Classification on the given academic admission dataset.

2.2 References

- [1] Selvaprabu Jeganathan, Saravanan Parthasarathy and P. M. Ashok Kumar, "PREDICTING THE POST GRADUATE ADMISSIONS USING CLASSIFICATION TECHNIQUES"
- [2] Akkem Yaganteeswarudu, "MULTI DISEASE PREDICTION MODEL BY USING MACHINE LEARNING AND FLASK API"
- [3] A. Sivasangari, V. Shivani, Y. Bindhu, D. Deepa, R. Vignesh, ":

PREDICTION PROBABILITY OF GETTING AN ADMISSION INTO A UNIVERSITY USING ML"

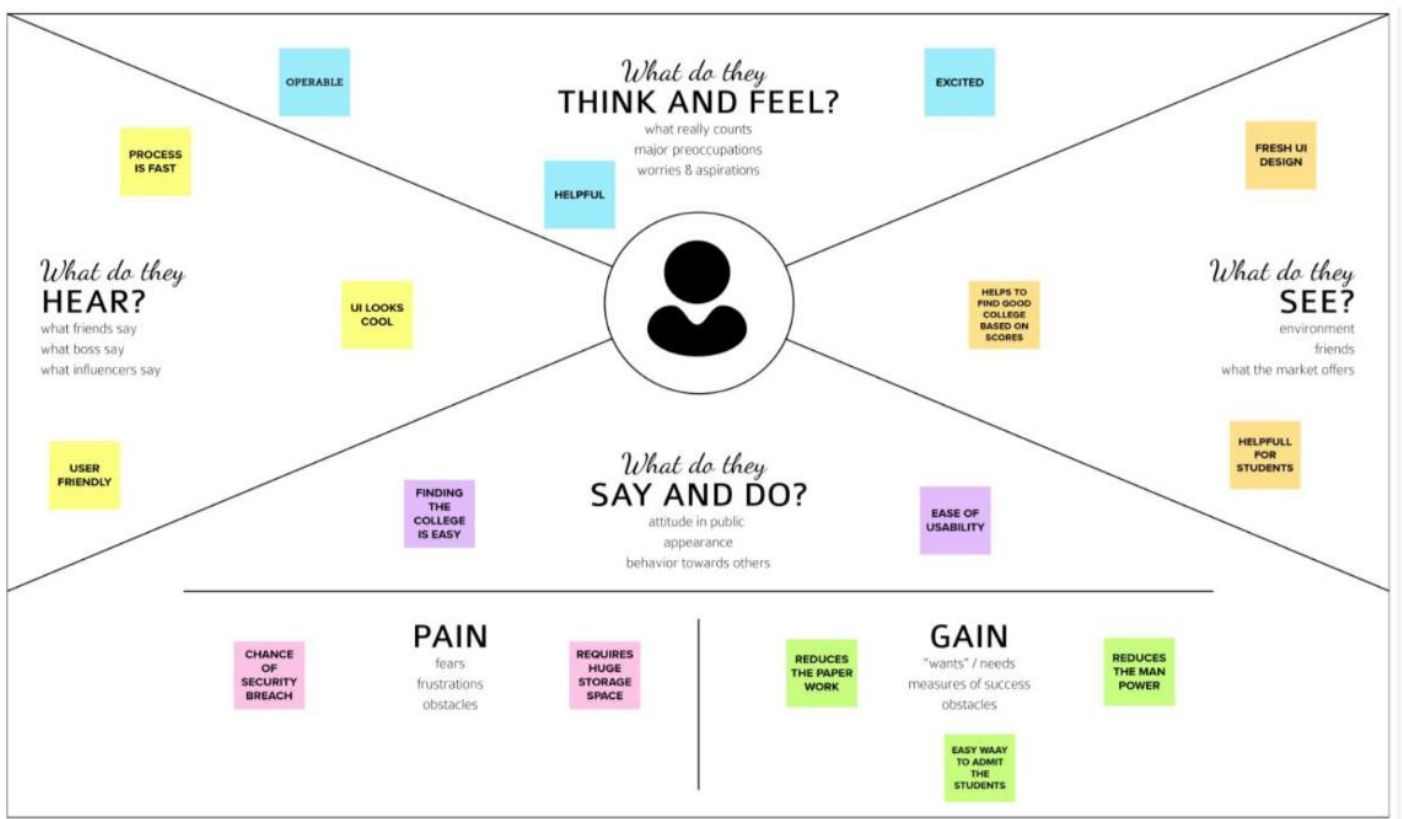
[4] S. Sridhar, S. Mootha and S. Kolagati, "A UNIVERSITY ADMISSION PREDICTION SYSTEM USING STACKED ENSEMBLE LEARNING"

2.3 Problem Statement Definition

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.


3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming

Ideation & Brainstorming3.3



Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and 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

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 the brainstorming session.

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

[Open article](#) ➔

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](#)

How might we (your problem statement)?

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.

2

Brainstorm

Write down any ideas that comes to mind that address your problem Statement

🕒 10 minutes

Nandhini R

Match tutorials learn ml	required visualization tools	test prediction ml predictor
provide : user login, my library etc	add details on how we predict	rebuild from existing solutions

Nivethitha

let's learn most used ML and AI algo	let's learn most used Data visualization	provide service fee "within budget university"
add college recommendation system	deploy using cloud-fast and scalable	

Preethi S

collect new data's from users and implement a model	build new predictive model with accuracy	analyse existing available services of such products
present results in understandable visual	for students thinking to take University provide guide how to select uni.	

Prajna A

learn web dev and frameworks	add location based predictions too	provide necessary links to recommended colleges
provide "stop" editor	provide web service with predictor for both students and parents	

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

Requirements



Application Type



Core Features



Additional Features



Extras



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



3.3 Proposed Solution

3.	Novelty / Uniqueness	In addition to the application process itself being a challenging one, students require a great deal of dedication and effort to complete the overall application process. If students were relieved from the step of selecting the best colleges and universities to apply to, it would definitely make their life easier.
4.	Social Impact / Customer Satisfaction	Graduates who might be confused about their future with respect to university admissions can benefit from this predictor. Students can apply to universities based on their chances of eligibility.
5.	Business Model (Revenue Model)	Financially, this project could benefit from the students' admission fees but they may want to select their college in advance. However, that is what this project does for prediction. In this project, this problem has been addressed by modeling a recommender system based on various classification algorithms. The required data was obtained from thegradcafe.com. The data set was used to train various models and one optimal model and some similar property-bearing universities were recommended so that a student's chances of getting into the university were maximized.
6.	Scalability of the Solution	In this project, this problem has been addressed by modelling a recommender system based on various classification algorithms. To predict the best University for a student, his or her GPA, GRE (Verbal and Quant) Score, and TOEFL Score are used as attributes. K nearest neighbour has been applied to predict best University and K means clustering has been used to find more similar universities. Support Vector Machine and Random forest has been used to predict the admission chance of particular student on specific University.

3.4 Problem Solution fit

Project Title: University Admit Eligibility Predictor

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMD14375

<p>1. CUSTOMER SEGMENT(S) Who is your customer? i.e. working parents of 0-3 y.o. kids</p> <p>College students who recently graduated and wish to be admitted to prestigious universities</p>	<p>6. CUSTOMER CONSTRAINTS What constraints prevent your customers from taking action or being more efficient? i.e. spending power, budget, no bank, network connection, available storage</p> <p>There might be a lack of trust in the predictor's accuracy or reliability, causing customers to refrain from using it. Furthermore, users would need to enter confidential information into the model. The predictor might be avoided by a certain segment of customers due to concerns about data misuse.</p>	<p>5. AVAILABLE SOLUTIONS Which solutions are available to the customers when they face the problem?</p> <p>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 vs. an alternative to digital computing</p> <p>As well as grades and GPA, we'll also take into account certain non-academic factors that may play a role in university admissions, further improving the predictor's accuracy. Secondly, we will put the model through rigorous tests in order to boost the accuracy of the predictor.</p>
<p>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.</p> <p>Designing a predictor requires a lot of data collection, so it is important that it is done right. Customers should be assured of optimum data security in order to have them retain their trust in our predictor.</p>	<p>9. PROBLEM ROOT CAUSE What is the root reason that this problem exists? What is the basic story behind the need to do this job? i.e. customers have to do it because of the change in regulations</p> <p>If inaccurate data is collected or not enough factors are taken into account to judge eligibility, the predictor's reliability may be compromised. The second reason may be that customers may refrain from using our product if they perceive it to be a cyberattack.</p>	<p>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 less</p> <p>Direct: Students will visit all universities they are interested. In order to get admission, she wishes to contact the students studying there a university that is desired. Find out what the requirements are as well as taking the necessary measures to meet admission requirements</p> <p>Indirect: Pay an agency to help students find required criteria in the desired universities and visit only those selective universities and get the job done.</p>
<p>3. TRIGGERS What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution on the news</p> <p>The chances of getting into the universities of your choice can often make students anxious and tense. With less time and effort, and less expense, students peers may have a wide selection of colleges to choose from.</p>	<p>10. YOUR SOLUTION If you are working on an existing business, define your current solution first. If in the startup, and ahead, how much is the reality</p> <p>Design a predictor with the help of the data collected, and ensure that it is accurate / reliable. Also make sure that the data collected from the users is safe and secure.</p>	<p>8. CHANNELS of BEHAVIOUR 8.1 ONLINE What kind of actions do customers take online? i.e. search online channels from IT</p> <p>customers might search for reliable eligibility predictors that are available online and rate them based on their liking.</p> <p>8.2 OFFLINE What kind of actions do customers take offline? i.e. attend a show channels from IT and use them for customer development</p> <p>Students would discuss amongst their peer group about such predictors and if they find one to be reliable enough, they would spread the word about it.</p>
<p>4. EMOTIONS: BEFORE / AFTER</p> <p>Insecure and unaware of the process, suffering to select the most appropriate university. Rapacious agent and missing out on possible universities.</p> <p>Result: Secure, user-friendly, and aware of the process. Costs are reduced, and universities that are feasible are not missed.</p>		

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	31October 2022
Team ID	PNT2022TMID14375
Project Name	Project – University Admit Eligibility Predictor
Maximum Marks	4 Marks

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through Facebook
FR-2	Authentication of user	An OTP is sent to the registered phone number and email to authenticate the user.
FR-3	User Data (input) User confirmation	A confirmation mail/SMS is sent to the user after the successful registration
FR-4	User Data (input)	Details like CGPA, IELTS/TOEFL score, projects done, GRE score are collected from the user
FR-5	Log in/Log out	Users can login using their mail id and password. They can logout as and when required.
FR-6	Editing user profile	The users must have an option to edit their profile even after the initial registration is over.
FR-7	Chat box facility	A chat box to provide the answers to FAQs and resolve any issues in the functioning.
FR-8	Video tutorial	A video tutorial explaining the working of the predictor should be made available for the convenience of the user.
FR-9	Previous admission records	Admission records of the universities in the years before the current academic year, should be made available to the user.

4.2 Non-Functional requirements

Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	The predictor must be easy to use and the UI should be smooth and decluttered.
NFR-2	Security	It should be ensured that necessary security features are in place to safe guard users' data from activities like data theft
NFR-3	Reliability	The reliability of the predictor must be maintained by providing the customer close-to-accurate results every single time.
NFR-4	Performance	The performance of the predictor is entirely dependent on its accuracy and the time taken by it to come up with the results.
NFR-5	Availability	It must be made accessible through any browsers to ensure that it is available to a wide spectrum of users.
NFR-6	Scalability	The predictor must be designed in such a way that its range/scope can easily be increased without any massive changes
NFR-7	Serviceability	Customer service must be provided through chat box/chat bots to resolve any issues that they might face and to resolve their queries.
NFR-8	Manoeuvrability	The platform must be easily manoeuvrable.

5. PROJECT DESIGN

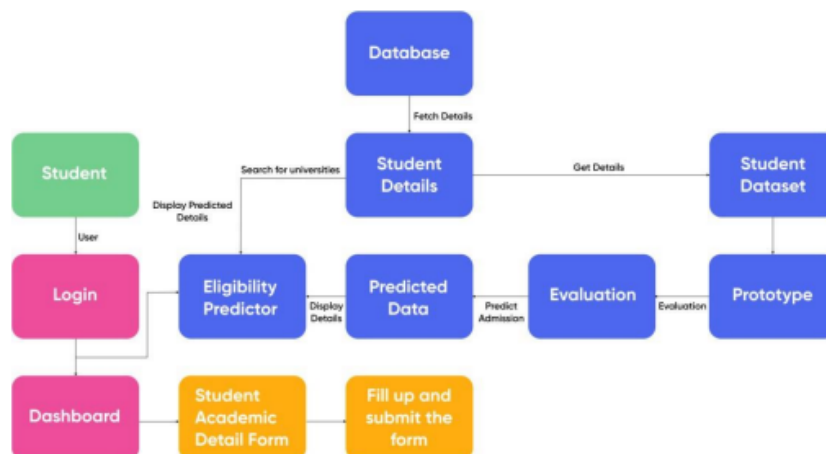
5.1 Data Flow Diagrams

PROJECT DESIGN PHASE-II DATA FLOW DIAGRAM & USER STORIES

Date	31 October 2022
Team ID	PNT2022TMID14375
Project Name	University Admit Eligibility Predictor
Maximum Marks	4 Marks

Data Flow Diagrams:

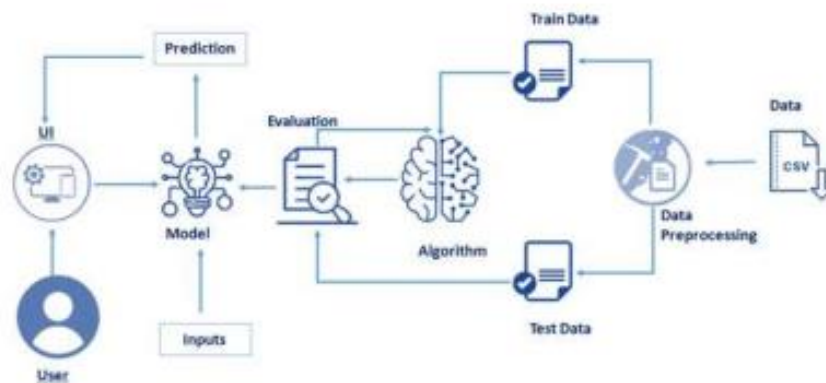
A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture

Solution Architecture

Date	09 October 2022
Team ID	PNT2022TMID14375
Project Name	University Admit Eligibility Predictor
Maximum Marks	4 Marks



Project Design Phase-II Technology Stack (Architecture & Stack)

Date	31 October 2022
Team ID	PNT2022TMID14375
Project Name	Project- University Admit Eligibility Predictor
Maximum Marks	4 Marks

Technical Architecture

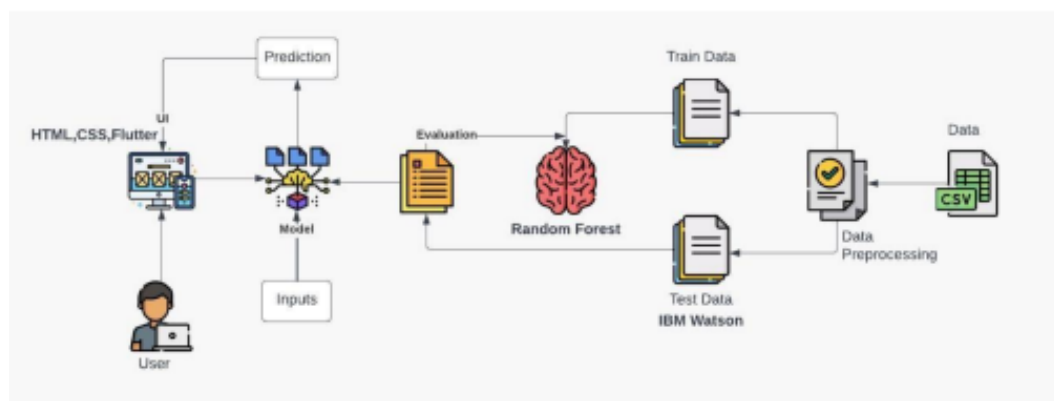


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1	User Interface	How user interacts with application and its features.	HTML, CSS, JavaScript etc.
2	Application Logic-1	The user fills the data into his profile which is then fed into the model to calculate the chances	Python [Jupyter]
3	Application Logic-2	The model predicts the eligibility chances of the user for different universities based on the input data	IBM Watson STT, Python
4	Database	Data of the names of the universities and their corresponding cut-offs and exam scores for admission	Imported through pandas in a csv format
5	Machine Learning Model	Predicts the output using the ML algorithm	KNN, Decision tree, Random Forest, etc.
6	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud	IBM cloud, local cloud

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Python for backend and Flask for front end	Python, Flask
2.	Security Implementations	To ensure the security of the data provided by the user	Encryption, OWASP
3.	Scalable Architecture	The model is scalable in nature because its scope can be increased easily.	Random forest ML algorithm, Logistic regression
4.	Availability	The model is available to anyone, anywhere, anytime	IBM load balancer
5.	Performance	The chances are predicted with a greater accuracy	Random forest ML algorithm

5.3 User Stories

User Stories

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Landing page	USN-1	As a user, I am able to view information and details about the university	I am able to access the university landing page	Medium	Sprint-1
		USN-2	As a user, I am able to view the current news about the university	I am able to access the latest news	Medium	Sprint-1
		USN-3	As a user, I am able to fill a form to contact the university with queries	I am able to fill and submit the contact form	Low	Sprint-2
		USN-4	As a user, I am able to go through the social media pages of the university	I am able to reach out to them via social media	Medium	Sprint-1
		USN-5	As a user, I am able to see testimonials of students who have passed out from that university	I am able to access the testimonials	Medium	Sprint-1
	Admissions	USN-6	As a user, I am able to see the cut off marks of past years.	I am able to download the previous year cut-off details	High	Sprint-2
		USN-7	As a user, I am able to access details of previous alumni.	I am able to access the details of alumni of the university	Medium	Sprint-2
		USN-8	As a user, I am able to predict my admission eligibility for the university	I am able to get result as either eligible/not eligible	High	Sprint-2

	Courses offered	USN-9	As a user, I am able to view the courses offered by the university for PG students	I am able to access the course details	Medium	Sprint-3
	Events	USN-10	As a user, I am able to check the various technical events about to happen in the university	I can register for the events	Low	Sprint-3
	E-books	USN-11	As a user, I am able to download and read e-books relating to visa formalities	I can download the e-books	High	Sprint-3

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Project Planning Phase

Project Planning (Product Backlog, Sprint Planning, Stories, Story points)

Date	22 October 2022
Team ID	PNT2022TMID14375
Project Name	Project - University Admit Eligibility Predictor
Maximum Marks	8 Marks

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

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Set	USN-1	Performing Data Analysis , Choosing perfect model(ML),Checking Error Matrix.	5	High	Nivethitha AS
Sprint-2	Designing Data Entry page	USN-2	As a user,I can enter the details of marks ,to predict universities\Designing User Interface Page.	5	High	Nandhini R
Sprint-3	Result Page	USN-3	As a user, I can Predict the Chances of Universities.	5	Medium	Prajna A
Sprint-4	Python And Flask	USN-4	Integrating Backend and Frontend with using Flask.	5	High	Preethi S

6.3 Sprint Delivery Schedule

Project Tracker, Velocity & Burndown Chart: (4 Marks)

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

Velocity:

Imagine we have a 6-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \text{SPRINT DURATION} / \text{VELOCITY} = 20/6 = 3.33$$

7. CODING & SOLUTIONING

7.1 Feature 1

- IBM Watson Platform
- Web UI
- Python Code
- HTML
- CSS
- JS

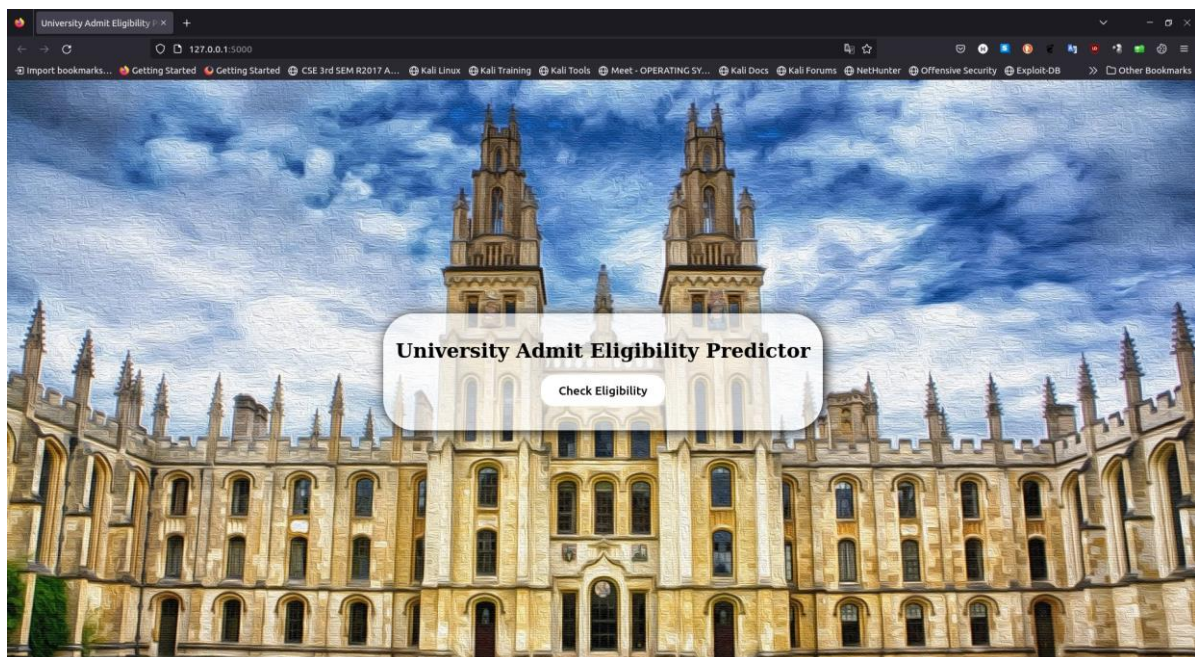
7.2 Feature 2

- Index
- Chance
- Nochance
- Demo2

8. TESTING AND RESULTS

8.1 Test Cases

TEST CASE 1



TEST CASE 2

The screenshot shows a web browser window with the title "University Admit Eligibility Predictor". The background image is a collage of a city skyline and a historic building. The form is a semi-transparent overlay with the following fields:

- GRE Score:
- TOEFL Score:
- University Rank:
- SOP:
- LOR:
- CGPA:
- Research:

At the bottom of the form are two buttons: "Submit" and "Back".

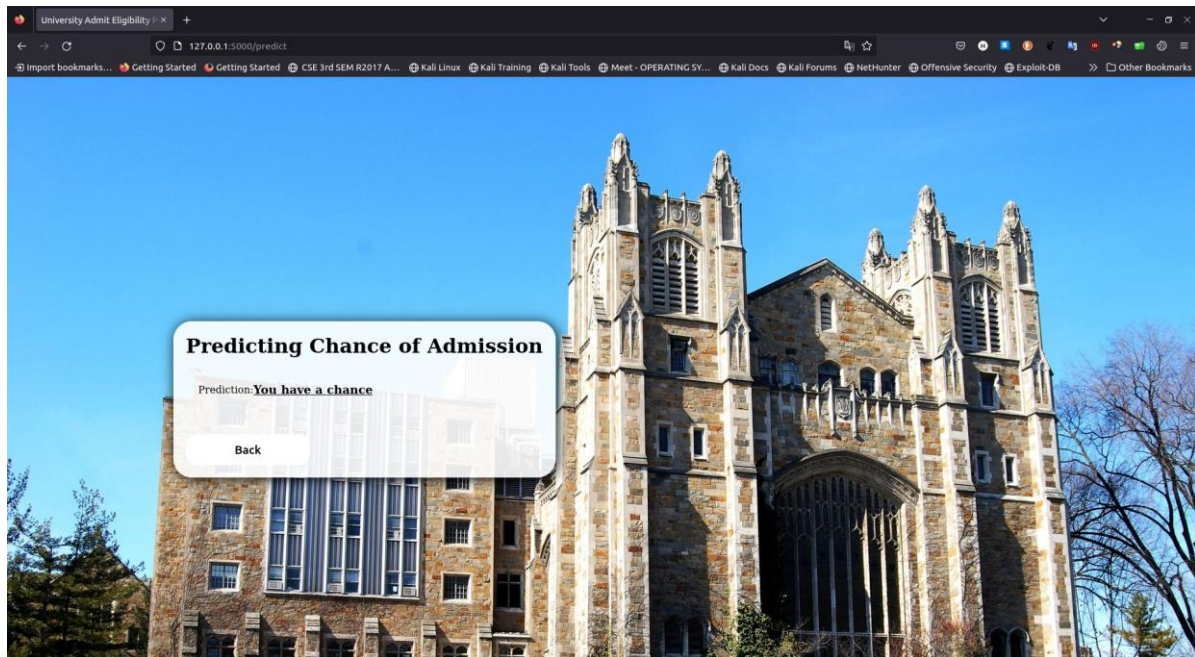
TEST CASE 3

The screenshot shows the same web application as in Test Case 2, but with numerical values entered into the form fields:

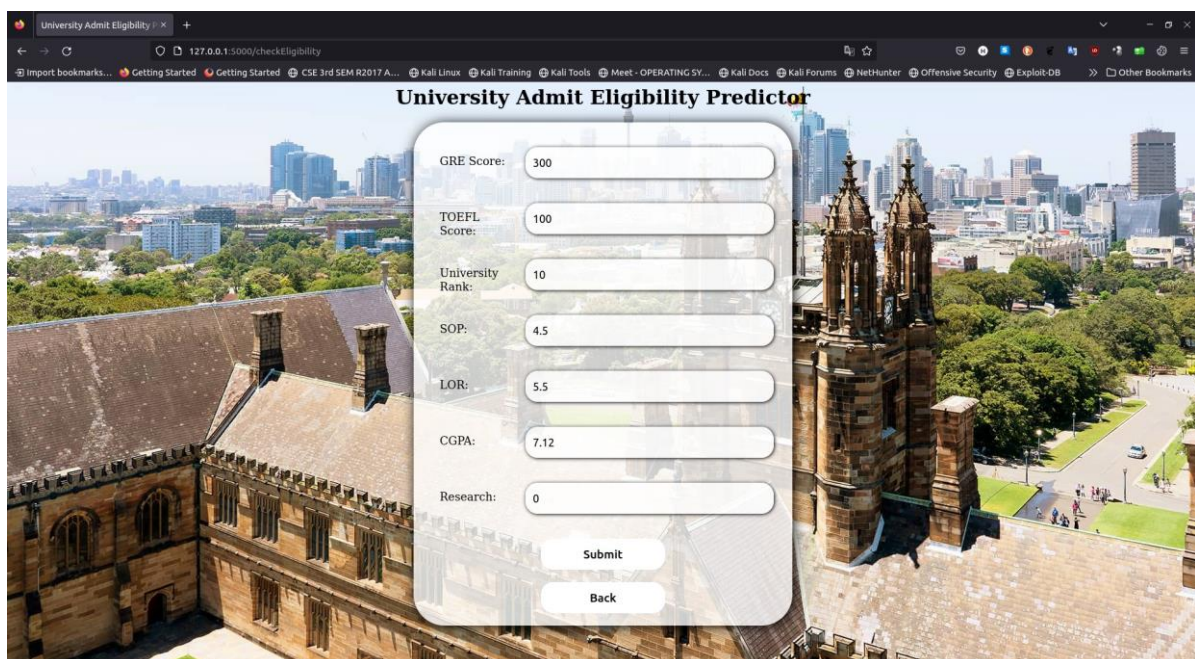
- GRE Score:
- TOEFL Score:
- University Rank:
- SOP:
- LOR:
- CGPA:
- Research:

The "Submit" and "Back" buttons remain at the bottom of the form.

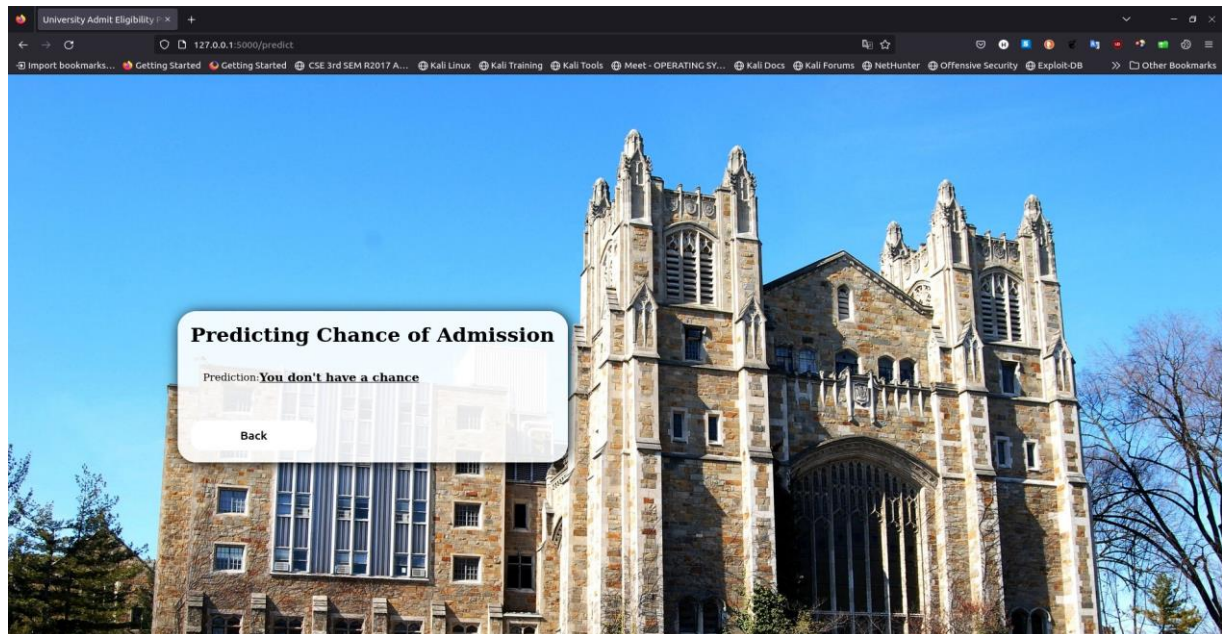
TEST CASE 4



TEST CASE 5



TEST CASE 6



9. ADVANTAGES

- Easy prediction of University based on the scores secured.
- It helps student for making decision for choosing the right college.
- It avoids data redundancy and inconsistency.

10. DISADVANTAGES

- Only few selected university are available for the prediction.
- A system will provide inaccurate result if data entered incorrectly.

11. CONCLUSION

In University Admit Eligibility Predictor students can register with their personal as well as marks details for predicting the admission in the colleges and administrator can allot the seats for the students.

12. FUTURE SCOPE

In the updated version of this software, it will contain features that we can select more number of universities for prediction and the system will provide correct results even if the data has been entered wrong.

13. APPENDIX

13.1 Source Code

```
from flask import Flask, render_template, request
import requests
```

```
API_KEY = "aq6ah02xEItNc_Ceas6Xq6GdfyCs1tRv6SUS0gzVbr3i"
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__, static_url_path='')
@app.route('/')
def index():
    return render_template('index.html')

@app.route('/checkEligibility')
def checkEligibility():
    return render_template('Demo2.html')

@app.route('/predict', methods=['POST'])
def predict():
    greScore = int(request.form['greScore'])
    toeflScore = int(request.form['toeflScore'])
    univRank = int(request.form['univRank'])
    sop = float(request.form['sop'])
    lor = float(request.form['lor'])
```

```

cgpa = float(request.form['cgpa']) research =
int(request.form['research'])
    array_of_input_fields = ['greScore', 'toeflScore', 'univRank', 'sop', 'lor',
'cgpa', 'research']
    array_of_values_to_be_scored = [greScore, toeflScore, univRank, sop,lor,
cgpa, research]
    payload_scoring = {"input_data": [{"fields": [array_of_input_fields],
"values": [array_of_values_to_be_scored]}]}
        response_scoring      =      requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/1c59d2d9-3eb5-49fd-858a-
80ea675b9c1e/predictions?version=2022-11-18', json=payload_scoring,
headers={'Authorization': 'Bearer ' + mltoken}) predictions =
response_scoring.json()
prediction = predictions['predictions'][0]['values'][0][0]

if prediction:
    return render_template('chance.html') else:
    return render_template('noChance.html')

```

```

if __name__ == "__main__":
    app.run()

```

13.2 GitHub

<https://github.com/IBM-EPBL/IBM-Project-42443-1660662766>