**ABSTRACT**

This project University Admit Eligibility Predictor is created with the idea of making the complicated and tiring process of college admissions easier for students looking to apply for universities by checking what are the possibilities of them getting into their dream universities based on various factors. The students are required to fill in their personal details such as university marks, CGPA, and also additional exam scores such as GRE, IELTS, TOEFL, etc. The administrator of universities can view these details of the students enrolled. Using this software, the entrance seat allotment becomes a lot easier and can be implemented way faster than the normal procedure.

Based on the obtained data set, various models are trained and universities carrying similar properties are suggested so that it increases the chances of students getting into the universities they applied for. Classification algorithms also predict the rate of acceptance for any student to a university. This project also makes use of Machine-Learning model which predicts the eligibility of students for admission in the desired Universities.

The main advantage of this project is the computerization of the seat allotment process. The administrator has the power of allotment. They can add the allotted seats into a file and the details are saved into the system. The total time consumption for the allotment process is vastly reduced which saves time for all.

**CHAPTER-1**

**INTRODUCTION**

* 1. **Purpose of Document**

This is a Requirements Specification Document for our web-based project “UNIVERSITY ADMIT ELIGIBILITY PREDICTOR”. It is an Technology 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. In addition to 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. **Project Summary**

Project Name: UNIVERSITY ADMIT ELIGIBILITY PREDICTOR

Project type: Web Application

Developers: Nashiah Justin A

Ruhi Misbaah R

Anushiya J

Joshina I

Languages used: Python, HTML, Javascript, CSS

Domain: Technology

Development Platform: Anaconda

Data Set Used: Admissions predictor dataset

* 1. **Motivation**

In the current world scenario, it is not enough for a student to just have an Under Graduate 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. A lot of students from India prefer to continue their higher education with foreign universities, especially in the United States, Canada, Germany.

In order 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 or not their applications will be accepted or not as no concrete documentation is available which lists the requirements.

**1.4 Significance**

This project was developed to provide a solution to that problem. 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.

* 1. **Project Scope**

The scope of this project is a web application that allows users to enter their academic data and get predictions of their chances of admissions 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.

**1.6 System Purpose**

* + 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.

**1.6.2 Locations**

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.

**1.6.3 Responsibilities**

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

Other desired features of the system:

* Maintaining a profile for each user
* Password protection for each account
  1. **Limitations and Future Work**
* Requires active internet connection.
* System will provide inaccurate result if data entered incorrectly.
* Future work in the project could include weighing in the features that have been ignored as of yet like percentage seats for Foreign Students. Other criterions like Co-circular achievements, Leadership positions held, job experience etc can also be included as metrics for the model.
  1. **Beneficiaries**
* It helps student for making decision for choosing a right college.
* Here the chance of occurrence of error is less when compared with the existing system.
* It is fast efficient and reliable.
* Avoids data redundancy and inconsistency.
* Very user-friendly.
* Easy accessibility of data.
  1. **Overview of Document**

**Section 1: Functional Objectives**

Each objective gives a desired behaviour for the system and a measure to determine if the final system has successfully met the objective. These objectives are organized by priority. In order for the new system to be considered successful, all high priority objectives must be met

**Section 2: Non-Functional Objectives**

This section is organized by category. Each objective specifies a technical requirement or constraint on the overall characteristics of the system. Each objective is measurable.

**Section 3: Use Case Model**

The specific behavioural requirements of the system are detailed in a series of use cases. Each use case accomplishes a business task and shows the interaction between the system and some outside actor. The system use case diagram depicts the interaction between all uses cases and system actors. A sequence diagram is also provided to show the flow of the application.

**Section 4: Software and Hardware used**

This section details the Hardware, Software and Online tools used for developing the UNIVERSITY ADMIT ELIGIBILITY PREDICTOR project.

**Section 5: Code Snippets**

This section provides some of the major code snippets used to develop the project.

**Section 6: Screenshots**

This section provides Screenshots of the UNIVRSITY ADMIT ELIGIBILITY PREDICTOR Application.

**Section 7: Testing Report**

This section details different test cases needed for the project and how they were implemented.

**Section 8: Validation**

This Section contains a final validation of the system where the system is checked to see if all the promised functional and non-functional requirements as mentioned in the SRS have been fulfilled or not.

**CHAPTER-II**

**FUNCTIONAL OBJECTIVES**

* 1. **High Priority**
* The system shall provide the user access to the AI predictor, wherein the user will be able to fill in a form with their academic transcripts data (GRE Score, TOFL Score, CGPA, SOP Score, LOR Score, Research experience), choose the tier of university they wish to apply to (1-5(top level)) and then get a prediction of their chances of admissions to that level university based on the mapping between their requirements and the student’s results.
* The system shall provide the administrator access to all the records in the database on a “read-only” basis
  1. **Medium Priority**
* The system shall provide all the users with answers to the most common FAQ”s like- “Distribution of University Tiers”, “University Admissions Criteria”
* The system shall allow the user’s details to be stored for the next time they return to the website. If the user chooses to take a new evaluation, the most recent inputs as well as prediction shall replace any previous data

.

* 1. **Low Priority**
* The system shall provide users an analysis of how the various factors mentioned in the form affect their chances of admissions as well as what is the general trend of applications to the various tiers of universities.

**CHAPTER-III**

**NON-FUNCTIONAL REQUIREMENTS**

**3.1 Reliability**

* The system shall be completely operational all hours of the day unless system failure or up gradation work is to be performed.
* Down time after a failure shall not exceed 24 hours.

**3.2 Usability**

* No training is required to use the website.
* The form home, about, FAQ and analysis pages load up within 10 seconds.
* The results from the predictor should not take more than 30 seconds.

**3.3 Performance**

* The system can support any number of users at a time.
* The mean time to view a web page over a 56Kbps modern connection shall not exceed 5 seconds

.

**3.4 Security**

* The system shall provide password protected access to the website to all users –students and administrations both.

**3.5 Online user documentation and help**

* The system shall provide a web page that explains how to navigate the site. This page should be customized based on what pages that user is allowed to access.
* This help page should be accessible from all other pages.
  1. **Supportability**
* The system will be able to incorporate more features without major reengineering. The system web site shall be viewable from Internet Explorer 4.0 or later, Netscape Navigator/Communicator 3.0 later and the America Online web browser version 3.0 or later.

**CHAPTER-IV**

**USE CASE MODEL**

**4.1 Use Case Diagram**

**4.2 Use Case Descriptions**

* **Login User**

|  |  |
| --- | --- |
| Use Case Name: | Login User |
| Summary | In order to access the system and the prediction model, a user must sign in. It also determines whether a user is an Administrator, in which case, it offers different functionalities. |
| Basic Flow: | 1. The use case starts when user indicate that he wants to login. 2. The system requests the username and password. 3. The user enters his username and password. 4. The system verifies the username and password against all registered users. 5. The system starts a login session and displays the home page. |

* **View Home Page**

|  |  |
| --- | --- |
| Use Case Name: | View Home Page |
| Summary | The user can now view the home page of the app and get access to the other tabs. |
| Basic Flow: | 1. The user case starts when the user gets logged in to the system. 2. The Home page is now visible to the user which holds all the information about how to work with the different elements of the application. |

* **View About Page**

|  |  |
| --- | --- |
| Use Case Name: | View About Page |
| Summary | The user can now view the about page of the app and get access to the other tabs. |
| Basic Flow: | 1. The use case starts when the user clicks on the about tab on top of the Home page. 2. The About page is now visible to the user which holds all the information about how, why and by whom the application was developed. |

* **View Queries**

|  |  |
| --- | --- |
| Use Case Name: | View Queries |
| Summary | The user can now view the queries the app provides answers to and get access to the other tabs. |
| Basic Flow: | 1. The use case starts when the user clicks on the “Queries” tab. 2. The drop-down menu appears for the user to choose which query he/she wants the answer to. |

* **View Analysis**

|  |  |
| --- | --- |
| Use Case Name: | View Analysis |
| Summary | The user can now view the analysis the app provides of how the various academic scores affect their chances of admissions. |
| Basic Flow: | 1. The use case starts when the user clucks on the “Analysis” tab. 2. The drop-down menu appears for the user to choose which analysis he/she wants to view. |
| Extensions Points: | Extension point are:   1. GRE Score Analysis. 2. TOEFL Score Analysis. 3. CGPA Analysis. 4. Research Experience Analysis. |

* **Full Predictor Form**

|  |  |
| --- | --- |
| Use Case Name: | Full Predictor Form |
| Summary | The Students can now access the admissions by filling up the form and getting their predicted chances of admissions. |
| Basic Flow: | 1. The use case starts when the user clicks on the “predictor” tab 2. The user can access the form and fill up all the data required by the system. All data fields must be filled. 3. On clicking the submit button, the data is sent to the predictor and the results are generated. 4. The resultant prediction of chances of admissions are returned to the user. |

* **View All Records**

|  |  |
| --- | --- |
| Use Case Name: | View All Users |
| Summary | The administrator can access all the records for all users that are registered in the system. |
| Basic Flows: | 1. The use case starts when the users is verified to be an administrator. 2. Once the admin clicks on the view all tab, all the records currently stored in the system become available to him/her. |

* **View User Profile**

|  |  |
| --- | --- |
| Use Case Name | View User Profile |
| Summary | The user can view his/her own profile. Students can also see the most recent values of their scores and prediction as stored in the system. |
| Basic Flow: | 1. The use case starts when the user clicks on the “Queries” tab. 2. The drop-down menu appears for the user to choose which query he/she wants the answer to. |

* **Sign Out**

|  |  |
| --- | --- |
| Use Case Name: | Sign Out |
| Summary | The user can sign out of the system and be redirected to the login age. |
| Basic Flow: | 1. The use case starts when the user clicks on the “Logout” tab. 2. The user session is ended, and he/she gets redirected to the login page. |

**CHAPTER-V**

**SOFTWARE AND HARDWAR PLATFORMS**

**5.1 Hardware**

A home PC -capable of handling light ML processing.

Device Specifications:

1. I5 10th Gen processor
2. 8 GB RAM
3. 64 bit Operation System
   1. **Software**

* **Anaconda (Jupiter Notebook)**

Project Jupiter is a non-profit organization created to “develop open-source software, open-standards, and services for interactive computing across dozens of programming languages”. [2] Spun off from I Python in 2014 by Fernando Perez, Project Jupiter supports execution environments in several dozen languages. Project Jupiter’s name is a reference to the three core programming languages supported by Jupiter, which are Julia, Python and R, and also a homage to Galileo’s notebooks recording the discovery of the moons of Jupiter.