Campus Placement Assistant

A PROJECT REPORT

submitted by

Adarsh S	**DMC18CS002**
Anal P R	**DMC18CS004**
Arjun R	**LDMC18CS030**
Muhammed Shaheen	**DMC18CS019**
Muhammed Yaseen	**DMC18CS020**

To

The APJ Abdul Kalam Technological University
In partial fulfillment of the requirements for the award of the Degree

Of

Bachelors of Technology
In
Computer Science and Engineering

M Dasan Institute of Technology (MDIT)
Ulliyeri P.O, Kozhikode-673620

Department of Computer Science and Engineering

DECEMBER 2021

DECLARATION

We undersigned declare that the project report "Campus Placement Assistant", submitted for the partial fulfillment of the requirements for the award of the degree of the Bachelors of Technology under the university of the APJ Abdul Kalam Technological University, Kerala is a bonafide work, done by us under the supervision of the Ms.Nithya V P (Asst Prof). This submission represents our ideas in our own words and where ideas or words of others have been included; We have adequately and accurately cited and referenced the original sources. We also declare that we have adhered to ethics of academic honesty and integrity and not misrepresented or fabricated any data or idea or fact or source in our submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and the university and can also be evoke penal action from the sorce which has thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other university.

Place :

Date :

ADARSH S   ARJUN R   MUHAMMED SHAHEEN   MUHAMMED YASEEN

M.DASAN INSTITUTE OF TECHNOLOGY

ULLIYERI

Approved by AICTE, New Delhi & Affiliated to APJ Abdul Kalam
Technological University

(KTU)

CERTIFICATE

This is to certify that the report entitled "CAMPUS PLACEMENT ASSISTANT"
by "ADARSH S" (DMC18CS002), "ARJUN R" (LDMC18CS030), "ANAL P R "
(DMC18CS004), "MUHAMMED SHAHEEN" (DMC18CS019) and "MUHAMMED
YASEEN" (DMC18CS020) to the APJ Abdul Kalam Technological University in
partial fulfillment of the requirements for the award of the Degree of Bachelors
of Technology in Computer Science Engineering is a bonafide of seminar carried
by his under our guidance and supervision. This report in any form not been
submitted to any other University or Institute of any purpose.

Project Guide	Project Coordinator	Head of The Department
Ms. NITHYA V P	Ms. SWATHI CHANDRA	Ms. NITHYA V P
Asstistant Professor	Asstistant Professor	Asstistant Professor
Dept. of CSE	Dept. of CSE	Dept. of CSE

ACKNOWLEDGEMENT

ABSTRACT

Each student or job aspirant, dreams to have best career. The campus placement assistant lets students to get a sense of where they are standing and what to do to ensure a decent selection . The placement predictor system helps in predicting whether a student will get placed or not . This system will also be helpful for identifying the areas where the students need to work on for placement. To achieve this, the predictor uses students details like academic marks ,coding skill etc.

Contents

List of Figures

CHAPTER 1

1   INTRODUCTION

According to statistics 1.5 million engineers graduate in India every year. The demand for skilled and qualified students is rising day by day. Thus the company use a good amount of capital in recruiting students from in-campus and off-campus because number of skilled and qualified students are very low. Colleges and Institution needs to focus on practical knowledge of real world rather than completing their syllabus. Placements are the biggest opportunities in the life of a student and they need to be fully prepared while attempting it. Placement Predictor system helps in predicting whether a student will get placement or not. This system can also be helpful for identifying the areas where student needs to work on for placement. This system uses student's details like academic marks, coding skills, etc. This system uses previous year placement statistics and student dataset for the placement prediction so the placement cell of the organization could set up a placement anticipated rundown for the present students. Along these lines it is important to direct an investigation on different placement prediction frameworks. This report studies on various placement prediction framework models.

CHAPTER 2

2 LITERATURE REVIEW

2.1 A Survey on Placement prediction system using Machine learning

2.1.1 Introduction

The purpose of placement management system is to modify the present manual system by the assistance of computerized software system fulfilling their needs, so their valuable data/information is stored for a longer time with simple accessing and manipulation of data. Student's academic achievements and their placement in campus selection is a difficult issue in current manual system. Monitoring the student's progress for his or her campus placement helps in monitoring the student's progression within the academic surroundings. the aim of organizations is to supply superior opportunities to their students.

This proposed student prediction system is most important approach which can be used to differentiate the student data/information on the basis of the student performance. Managing placement and coaching records in any larger organization is quite tough because of the large number of students. This system can classify the student knowledge with ease and can be useful to several educational organizations. There are several classification algorithms and mathematics-based techniques which can be taken nearly as good assets for classifying the students' information set in the education field.

In Our system, Naïve Bayes, SVM, KNN algorithm is applied to predict student performance which can facilitate to identify performance of students and also provides suggestion to improve performance for students such as we are going to classify the student's knowledge set for placement and non-placement classes based on that result, education organizations can give superior training to their students. Based on data received by system, student's performance is analysed in numerous views to check the achievements of the students through their activities and suggests improvement for better placement.

2.1.2   Working

A system is deployed in which the students will register/Login into the system and enter their biodata and skillsets, according to students' academic details the system will identify whether the student is eligible for the placement and recommend the courses to the students. Admin creates the courses and registers students to the respective courses. Admin can view the courses and students along with their attributes. Admin predicts placement status of current students. If the student is eligible for placement mail will be sent to the student from admin and students names will be displayed on the dashboard in their colleges.
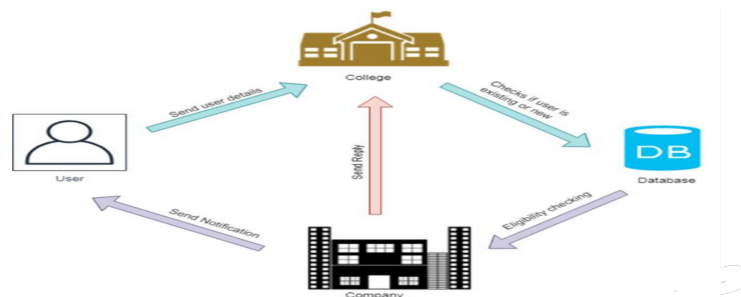
Figure 1: System Architecture

Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model. To better understand this definition let's take a step back into goal of machine learning and model building.
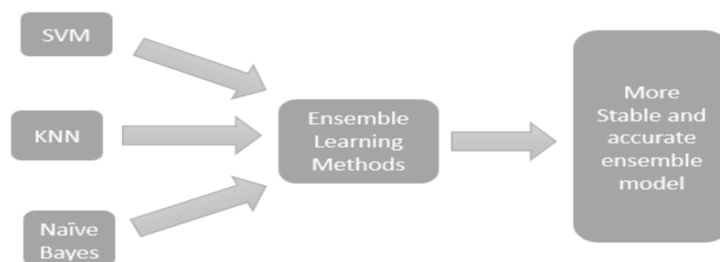
Figure 2: Ensemble Learning

The goal of any machine learning problem is to find a single model that will best predict our wanted outcome. Rather than making one model and hoping this model

is the best/most accurate predictor we can make, ensemble methods take a myriad of models into account, and average those models to produce one final model.

This system is helpful for institutions to predict student's campus placement. This system would help reduce tedious job of manual placement system. The placement officer can work on identifying the weaknesses of each students and can suggest improvements so that the students can overcome the weakness and perform to the best of their abilities.

2.1.3   Advantages

- Ensemble methods have higher predictive accuracy, compared to the individual models.

- Ensemble of models is always less noisy and is more stable.

- It reduces the tedious job of manual placement system.

2.1.4   Disadvantages

- The art of ensembling is hard to learn and any wrong selection can lead to lower predictive accuracy than an individual model.

- Ensembling is less interpretable, the output of the ensembled model is hard to predict and explain.

- Ensembling is expensive in terms of both time and space.

- It takes more time to compute.

2.2   Use of ID3 Decision Tree Algorithm for Placement Prediction

2.2.1   Introduction

Every year corporate companies come to colleges in order to recruit students. Recruitment is one of the most essential processes for any organization as they look for skilled and qualified professionals to fill up the positions in their organization. Many companies hire students through campus recruitment process.

Campus recruitment is an efficient way to get the right resources at the right time with minimal cost and within minimum time frame. While the industry hires candidates from different institutes, students too get a chance to start their career with some of the best companies in the corporate sector. The main aim of this paper is to identify relevant attributes based on quantitative and qualitative aspects of a student's profile such as CGPA, academic performance, technical and communication skills and design a model which can predict the placement of a student.

For this purpose ID3 classification technique based on decision tree has been used. The result of this analysis will assist the academic planners to design a strategy to improve the performance of students that will help them in getting placed at the earliest.

2.2.2   Working

Campus placement is a process where companies come to colleges and identify students who are talented and qualified, before they finish their graduation. The proposed system determines the likelihood of placement based on various attributes of a student's profile. Depending on the parameters , manual classification is done whether the student is placed or not placed. A decision tree is then implemented to determine the probable outcome of a student being placed.

	A	B	C	D	E	F	G	H	I	J
1	Dept	CGPA	'Programming Skills'	'Future Studies'	SSC	HSC	Communication skills	Backlog	Internships	Placement
2										
3	CS	8.33	Good	Yes	92	88	Good	0	1	'Not Placed'
4	EXTC	6.78	Good	No	58	56	Average	1	0	'Not Placed'
5	EXTC	6	Good	No	73	64	Good	2	0	'Not Placed'
6	IT	9.3	Poor	Yes	90	86	Average	0	0	'Not Placed'
7	IT	9.67	Good	No	85	82	Good	0	0	Placed
8	CS	6.87	Poor	No	65	67	Average	0	0	'Not Placed'
9	CS	9.22	Poor	Yes	93	86	Average	0	0	'Not Placed'
10	EXTC	7.9	Good	No	82	78	Good	0	1	Placed
11	IT	7.99	Good	No	80	77	Good	0	0	Placed
12	CS	9.9	Average	No	98	92	Good	0	1	Placed
13	CS	8.6	Average	No	93	87	Good	0	1	Placed
14	CS	8.7	Poor	No	76	63	Good	0	0	'Not Placed'
15	CS	6.12	Good	No	89	84	Average	0	0	'Not Placed'
16	IT	7	Good	Yes	94	89	Good	0	1	'Not Placed'

Figure 3: Student Data

The combination of various attributes determines whether the student is placed or not. The quantitative aspects like undergraduate CGPA, Marks obtained in X and XII form the major aspect of a student's academic endeavours. The qualitative aspects like communication and programming skills form a backbone for a student to get placed as each recruiting company desires to hire students that have a sound technical knowledge and ability to communicate effectively. The other factors like internships, backlogs, future studies add value only when the prior requirements are met.

Based on the training set, information gain and entropy is calculated to determine the splitting attribute for constructing the decision tree. The algorithm gives a pruned decision tree with leaves as the decision that is placed or not placed. The primary node consists of programming skills which can accommodate three possible values viz. Good, Average and Poor. If the programming skills are Poor, the student is not placed. Furthermore, if the programming skills are good, the student may be placed based on the academic credential which is CGPA. If the CGPA of the student is above 7, student will be placed otherwise the student will not be placed. Also, if the student has average programming skills, he may still be placed based on other attributes like internships, future studies, communication skills, etc. The final decision tree with four leaf nodes is

CAMPUS PLACEMENT ASSISTANT   Project Report 2022

obtained as shown below. The leaf nodes hold the value whether the student is placed or not placed.
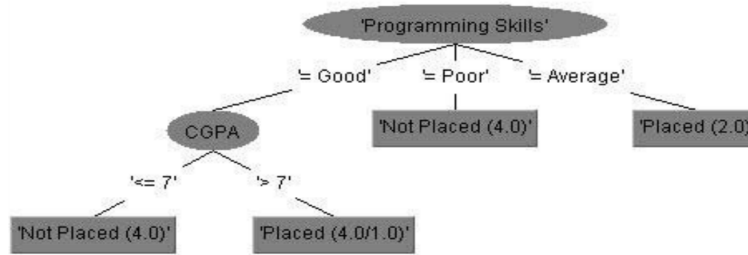

Figure 4: Decision Tree

The root node chosen here is Programming Skills. Further classification is done by calculating information gain and entropy for each attribute. Consider the attribute future studies; it has two possible classes viz. Yes and No. There are four students who wish to pursue future studies and remaining ten out of fourteen who do not have any plans to opt for higher studies. According to the training set, all the students who wish to pursue future studies are not placed. This indicates that all the information is contained in a single class. Hence, entropy becomes zero. Higher value of entropy indicates higher degree of distribution of information among classes. The lowest value of information gain is obtained for programming skills. Thus it is chosen as the root node. Further, the next lowest value (CGPA) is taken as the split node for next level. The subsequent nodes of decision tree at each level are determined by the value obtained in information gain.

In this paper ID3 classification algorithm is used to generate decision rule. The generated decision rule can be used to predict a student's campus placement. The result of this algorithm can be used by the placement-in-charge to identify those set of students that are likely to face problems in campus placement. The classification model can play an important role in increasing the placement statistics. It can be concluded that classification algorithms can be used successfully in order to predict student placement. Further the implementation can be done in development and application of novel computational techniques for the analysis of large datasets.

M-DIT Kozhikode   8   Department of CSE

2.2.3 Advantages

- The best feature of using trees for analytics is that they are easy to interpret and explain. Decision trees are very intuitive and easy to explain.

- Nonlinear relationships between parameters do not affect tree performance.

- Missing values in the data also do NOT affect the process of building a decision tree to any considerable extent.

- Compared to other algorithms decision trees requires less effort for data preparation during pre-processing.

2.2.4 Disadvantages

- A small change in the data can cause a large change in the structure of the decision tree causing instability.

- For a Decision tree sometimes calculation can go far more complex compared to other algorithms.

- Decision tree often involves higher time to train the model.

- Decision tree training is relatively expensive as the complexity and time has taken are more.

2.3 Student Prediction system for Placement training using Fuzzy inference System

2.3.1 Introduction

Proposed student prediction system is most vital approach which may be used to differentiate the student data/information on the basis of the student performance. Managing placement and training records in any larger organization is quite difficult as the student number are high; in such condition differentiation and classification on different categories becomes tedious.

Proposed fuzzy inference system will classify the student data with ease and will be helpful to many educational organizations. There are lots of classification algorithms and statistical base technique which may be taken as good assets for classify the student data set in the education field.

In this paper, Fuzzy Inference system has been applied to predict student performance which will help to identify performance of the students and also provides an opportunity to improve to performance. For instance, here we will classify the student's data set for placement and non-placement classes.

2.3.2 Working

The main objective of this project is to classify large set of student data set using fuzzy logic and predicting student for placement training whether the student is eligible or not for placement training. As we discussed above we are taking two classes for final year student, classes are placement training and non-placement training. For the placement training those students will eligible those will get good marks or good CGPA in exam and remaining student will go for extra classes to improve their performance. From this approach we can predict all students in few times, if we want to classify all students on excel sheet so it will take lot of time, so these approaches are much better compare to other and it will help you.

This proposal may also able to judge the performance of the student continuously. The main advantage of this system is to improve accuracy and speed of the student by conducting an exam and decide some criteria to pass that exam if student fails in exam than some important classes or training will provide to the student it will help to student to improve itself. The advantage of the system is the accuracy of the prediction and speed of the result provided. So the fuzzy system provides the easy way to classify several numbers of student data set. By making a fuzzy inference system and create some rules which will predict the student for pre-defined classes such as placement training and non-placement training.
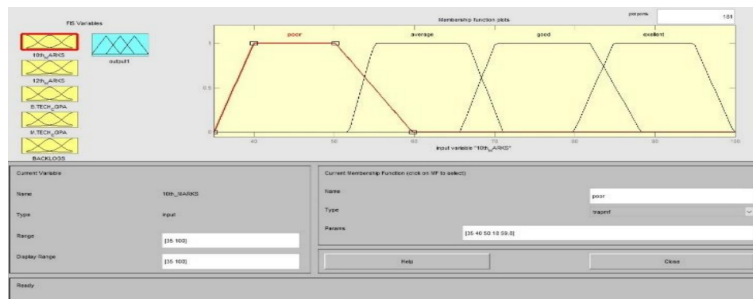
Figure 5: Fuzzy Inference System of each Input

For this system we collected 31 sample data set of M.Tech computer science student from our organization. In this data set put student registration number, name, 10th marks, 12th marks, B.Tech cgpa, M.Tech cgpa and number of backlogs. For classifying data set we skip some part of data set like registration number as well as name also here we take only academic marks and current backlog only and created more than 500

rules for predicting student performance for placement training on the basis of certain parameters.

For example, if (10th marks is poor) and (12th marks is poor) and (B.Tech cgpa is poor) and (M.Tech cgpa is poor) and (backlog is less than 2) then (output is NO). From this fuzzy system can classify thousands number of student's dataset. The main step of fuzzy based student prediction system is creating membership function for each input (10th marks, 12th marks, B.Tech CGPA, M.Tech CGPA, and number of backlogs) and creates output in terms of eligibility corresponding inputs.

Figure 6: Output of Classified Individual student result

This system easily predicts and analyzes lot of student data set for predefined classes by using fuzzy logic which will definitely be a good asset to organization to analyze the information of huge number of student information and data sets. Future studies are required to investigate new hybrid models of fuzzy classification algorithms to improve the performance of prediction system.

2.3.3   Advantages

- It is a robust system where no precise inputs are required.

- The Fuzzy Logic algorithms can be coded using less data, so they do not occupy a huge memory space.

- As it resembles human reasoning, these systems are able to solve complex problems where ambiguous inputs are available and take decisions accordingly.

- These systems are flexible and the rules can be modified.

2.3.4   Disadvantages

- There is no single systematic approach to solve a problem using Fuzzy Logic. As a result, many solutions arise for a particular problem, leading to confusion.

- A major drawback of Fuzzy Logic control systems is that they are completely dependent on human knowledge and expertise.

- You have to regularly update the rules of a Fuzzy Logic control system.

2.4   Student Placement Prediction Using Support Vector machine Algorithm

2.4.1   Introduction

Campus placement plays a vital role in every educational institution in helping students to achieve their goals. All students dream to obtain a job offer in their hands before they leave their college. In this paper, a predictive model is designed which can predict whether a student get placed or not.

The main objective of this project is to analyze the student's academics data, aptitude data and predict the placement possibilities of students to have an idea about where they stand and what to be done to obtain a good placement. Which also aids to increase the placement percentage of the institutions?.

The data has been collected by the institution for which prediction is going to be done and by applying suitable data pre-processing techniques. The model is built by both training and test set which gives accuracy in prediction. Here we use a single supervised machine learning algorithm named support vector machine algorithm. This algorithm independently predicts the results and we then compare the efficiency of the algorithm, which is based on the dataset. This model will help the placement cell to focus on the potential students and help them to improve their technical and other skills.

2.4.2   Working

In this paper we use machine learning techniques to predict the placement status of students based on a dataset. The parameters in the dataset which are considered for the prediction are Quantitative scores, Logical Reasoning scores, Verbal scores, Programming scores, CGPA, internal marks, external marks, list of students placed in a company The placement prediction is done by machine learning Algorithm using SVM.
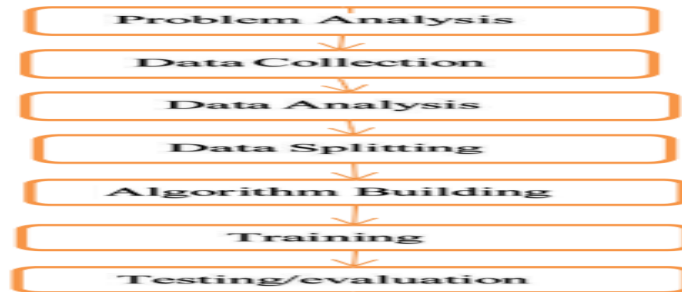

Figure 7: System Architecture

1. **Data Collection** Sample data has been collected from college placement department. As an input for model prediction, which consist of all the required dataset.

2. **Data Preparation & Pre-processing** Data preparation is a step in a data analysis process in which data from one or more sources is cleaned, transformed and enriched to improve the quality of data prior to its use. The collected data were then pre-processed to fill the missing data and made compatible for further processing.

3. **Data Splitting** Splitting the Dataset into Training set and Test Set ,Now the next step is to split our dataset into two. Training set and a Test set. We will train our machine learning models on our training set, i.e our machine learning models will try to understand any correlations in our training set and then we will test the models on our test set to examine how accurately it will predict. A general rule of the thumb is to assign 80to test set.

4. **Algorithm Building** SVM algorithm is appied on the dataset. SVM stands for Support Vector Machine. It is also a supervised machine learning algorithm that

can be used for both classification and regression problems. However, it is mostly used for classification problems. A point in the n-dimensional space is a data item where the value of each feature is the value of a particular coordinate. Here, n is the number of features you have. After plotting the data item, we perform classification by finding the hyper-plane that differentiates the two classes very well. Now the problem lies in finding which hyper-plane to be chosen such that it is the right one. The Support Vector Machine (SVM) helps in identifying the hyperplane for classifying the data samples. In the case of multiple hyperplanes, the one which has maximum distance from the data points was chosen for better classification.

5. **Evaluation and Testing:** The performance measurement of the model was evaluated with the help of various metrics like accuracy, sensitivity, F1-score and precision. The performance visualization of the multi-class classification problem was analyzed using a graphical plot AUC (Area under the Curve) ROC (Receiver Operating Characteristics) curve that reveals the analytical ability of a binary classifier system as its discrimination threshold. The ROC curve is generated by plotting the true positive rate against false-positive rates at various threshold rates. The best algorithm based on the performance parameters was selected to predict the placement category of students. Based on the details provided by the students, the placement category could be predicted and the result would be displayed along with the suggestions for further improvement.

From the study it is clear that the student dataset containing academic and placement details are a potential source for predicting the future placement chances and It is clear that SVM gives an accuracy of 100. This prediction can enlighten students to identify their capabilities and improve accordingly. This system also helps in the academic planning of an institution to prepare proper strategies and improve the placement statistics for the future years.

2.4.3 Advantages

- SVM works relatively well when there is a clear margin of separation between classes.

- SVM is more effective in high dimensional spaces & SVM is relatively memory efficient.

- SVM is effective in cases where the number of dimensions is greater than the number of samples.

2.4.4 Disadvantages

- SVM algorithm is not suitable for large data sets.

- SVM does not perform very well when the data set has more noise i.e. target classes are overlapping.

- In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform.

2.5   Recruitment System with Placement Prediction

2.5.1   Introduction

The availability of information and the facility for the user to take action on the information collected have been revolutionized by the use of the Internet and the World Wide Web. The placement process can be managed using the internet which arises a need to develop a web-based placement management system specifically by the recruiters and the software engineers that can be used as a Recruitment system (Online TnP portal).

This system can be used as an application for both candidates and recruiters. Advanced features for recruiters are available as they can shortlist candidates for further rounds according to their requirements on the basis of the probability obtained. The current recruitment system recruiters do not possess candidate information apart from his/her CV.

This proposed system aims to analyze the candidate performance and recommend candidates fittest for the job using Random Forest Regressor algorithm that will help to maximize the placement probability of candidates easing the recruiter's task. Random Forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. This system will provide ease and efficiency in recruitment process.

2.5.2   Working

The proposed system is a Web application meant to be used for recruitment process. It can be used by organizations as a tool for effective recruitment by analyzing the candidate's fitness for the job. And colleges can use the system to get an idea about the probability of the student to be placed prior to the placement drive with the help of Placement prediction feature. This system recommends the candidates on the basis of their likelihood to get placed. It considers parameters like candidate's SSC marks, HSC marks, CGPA, gender to predict the placement probability. Machine learning technique is used to implement Random Forest Regressor algorithm. The model is first trained on a dataset of any previous placement drives and then used to predict the probability of the candidate to get placed. The system consists of two phases –

1. Real-time placement prediction system (dynamic).

2. Probable candidate list generator(static).

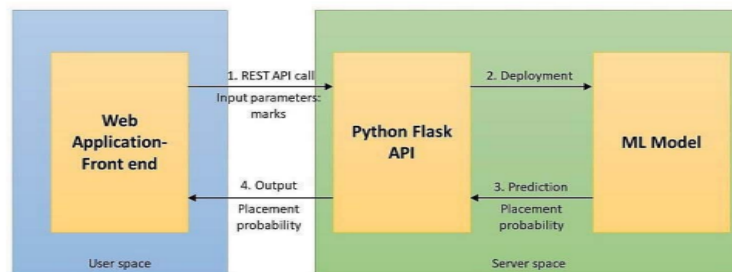**Real-time placement prediction system (dynamic):**

Figure 8: Dynamic predictor

This phase is for walk-in candidates where no previous record of him is available in the system. The following steps are carried out:

Step1: Placement candidate enters required credentials.

Step2: These input parameters will be passed to the Flask API using REST API call.

Step3: The data forwarded by API will be given to trained ML model for calculation of placement probability.

Step4: This placement probability will be displayed on the web page back through Flask API and the candidate record will be saved in a csv file.
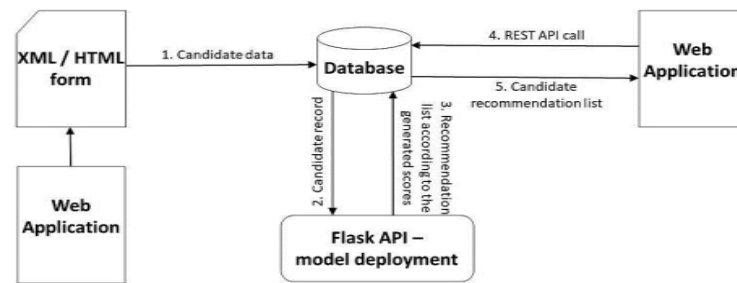
Probable candidate list generator(static):

Figure 9: Static predictor

This phase is for candidates who enter their data through a form or whose records are already present in the database of the system.

Step1: HTML/XML form is filled by candidate to store data in database using browser.

Step2: Flask API is used to implement ML algorithms on database data.

Step3: A trained ML model is implemented on the candidate records and scores are generated.

Step4: A candidate recommendation list is generated based on the scores.

Step5: Whenever a placement client requests the list, it will be retrieved from database and displayed on the web page.

Prediction Process in both phases: A pickle file is created, in which scikit-learn is imported which is a Machine Learning module. Using scikit-learn, Random Forest Regressor algorithm is trained on a dataset which contains historic data of placement drive. Now this pickle file is read by the Flask API file and the trained model is fitted on the current candidate records.

The proposed system consisting of dynamic prediction uses Machine learning to predict the placement probability of candidates dynamically using the parameters such as CGPA, HSC marks, SSC marks. It overcomes the limitations of current recruitment system which displays discrete values and gives an idea about placement to the candidates. The Scikit learn module provides us with a Random Forest Regressor algorithm which helps in generating probabilities with accuracy for large datasets and hence is comfortably suited for this purpose.

2.5.3   Advantages

- It reduces overfitting in decision trees and helps to improve the accuracy.

- It is flexible to both classification and regression problems.

- Normalising of data is not required as it uses a rule-based approach.

- It works well with both categorical and continuous values & it automates missing values present in the data.

2.5.4   Disadvantages

- It requires much computational power as well as resources as it builds numerous trees to combine their outputs.

- It also requires much time for training as it combines a lot of decision trees to determine the class.

- Due to the ensemble of decision trees, it also suffers interpretability and fails to determine the significance of each variable.

CHAPTER 3

3   PROBLEM STATEMENT

Previous Literatures Survey's have used machine learning algorithms like :

1. Ensemble learning (with SVM, KNN and Naive Bayes) for recruiters and colleges to predict the placement.

2. ID3 algorithm for academic planner to predict the placement.

3. Fuzzy logic for analyzing student perfomance to predict the placement.

4. SVM for placement cell to predict the placement.

5. Random forest regressor for recruiters to predict the placement.

Eventhough the algorithms has classified the placement probablity as can be Placed or not, the expected accuracy was not attained with any of the above mentioned algorithm as the used data set was small, algorithm was complex, algorithm was not efficient etc... . So in this paper we have decided to use huge dataset and select the best algorithms out of the following:

1. Logistic Regression

2. KNN Algorithm

3. SVM Algorithm

4. XGBoost Classifier

5. Decision Tree Algorithm

CHAPTER 4

4   PROPOSED SYSTEM

4.1   Introduction

The importance of campus placement assistant system is to modify the present manual system where there the data are unorganized manner. Due to unorganized data the stakeholders namely Placement Officers and Recruiters face difficulty in selecting the best student from the campus. The students should also need to have a clear picture of the placements going in the campus, so an effective computerized system will help the stakeholders to select the best student and the student to understand their level that is whether they can be placed or not.

The

4.2   Anvil

Anvil is a new way to build web apps, with nothing but Python. This documentation will tell you all about how it works. You do not need to know anything about HTML, Javascript or website development to use Anvil. All you need is some familiarity with the Python programming language.Anvil fills in these gaps by allowing you to build a full-stack web app using only Python. You can build a user interface with a simple drag and drop UI (or build it with code if you insist), plot with your favorite Python plotting library (Plotly, Matplotlib, etc.), and then deploy to the web in one click. No servers or containers to deal with.

Structure of an Anvil app

An Anvil app is made up of:

1. A User Interface, which you design with a drag-and-drop designer.

2. Client-side Python code, which runs in the web browser.

3. Server-side Python code, which runs on Anvil's servers.

4. A built-in database (Data Tables), which stores your data.

5. Some Python code running on your computer, which can also interact with your app.

What else you can do. Anvil also has built-in support for:

1. One-click hosting and deployment

2. Versioning your app with Git version control

3. Sending and receiving emails

4. Exposing and using HTTP APIs

5. Integrating with services from: Google, Microsoft, Facebook and anything else with a Python SDK!

Deep Notebook Deepnote is a seriously slick Python notebook, hosted in the cloud, with incredible real-time collaboration. It's great for working with other data scientists. But what happens when you want to share your project with non-programmers? They need an easy-to-use interface – so you need to deploy what you've built. Deepnote has been used to integrate the machine learning code to Anvil.

4.3   Python Packages

NumPy NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

At the core of the NumPy package, is the ndarray object. This encapsulates n-dimensional arrays of homogeneous data types, with many operations being performed in compiled code for performance. There are several important differences between NumPy arrays and the standard Python sequences:

- NumPy arrays have a fixed size at creation, unlike Python lists (which can grow dynamically). Changing the size of an ndarray will create a new array and delete the original.

- The elements in a NumPy array are all required to be of the same data type, and thus will be the same size in memory. The exception: one can have arrays of (Python, including NumPy) objects, thereby allowing for arrays of different sized elements.

- NumPy arrays facilitate advanced mathematical and other types of operations on large numbers of data. Typically, such operations are executed more efficiently and with less code than is possible using Python's built-in sequences.

pandas pandas is a Python package that provides fast, flexible, and expressive data structures designed to make working with "relational" or "labeled" data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python. Additionally, it has the broader goal of becoming the most powerful and flexible open source data analysis / manipulation tool available in any language. It is already well on its way towards this goal. Pandas makes it simple to do many of the time consuming, repetitive tasks associated with working with data, including:

- Data cleansing

- Data fill

- Data normalization

- Merges and joins

- Data visualization

- Statistical analysis

- Data inspection

- Loading and saving data etc..

Matplotlib Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc. **Seaborn** Seaborn is a data visualization library built on top of matplotlib and closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data. One has to be familiar with Numpy and Matplotlib and Pandas to learn about Seaborn.

Seaborn offers the following functionalities:

- Dataset oriented API to determine the relationship between variables.

- Automatic estimation and plotting of linear regression plots.

- It supports high-level abstractions for multi-plot grids.

- Visualizing univariate and bivariate distribution.

4.4   Git Hub

GitHub, Inc. is a provider of Internet hosting for software development and version control using Git. It offers the distributed version control and source code management functionality of Git, plus its own features. GitHub is an online software development platform used for storing, tracking, and collaborating on software projects. It enables developers to upload their own code files and to collaborate with fellow developers on open-source projects.

A GitHub repository can be used to store a development project. It can contain folders and any type of files (HTML, CSS, JavaScript, Documents, Data, Images). A GitHub repository should also include a licence file and a README file about the project. A GitHub repository can also be used to store ideas, or any resources that you want to share.

A GitHub branch is used to work with different versions of a repository at the same time. By default a repository has a master branch (a production branch). Any other branch is a copy of the master branch (as it was at a point in time). New Branches are for bug fixes and feature work separate from the master branch. When changes are ready, they can be merged into the master branch. If you make changes to the master branch while working on a new branch, these updates can be pulled in.

Pull Requests are the heart of GitHub collaboration. With a pull request you are proposing that your changes should be merged (pulled in) with the master. Pull requests show content differences, changes, additions, and subtractions in colors (green and red). As soon as you have a commit, you can open a pull request and start a discussion, even before the code is finished.

CHAPTER 4

5   SOFTWARE REQUIREMENTS& HARDWARE REQUIREMENTS

5.1   SOFTWARE REQUIREMENTS

- WEB APP - Any web Browser of the last OS

- ANDROID DEVICE - Android 4.1 (API level 16) or higher.

- IOS DEVICE - IOS 9.0 Later

5.2   HARDWARE REQUIREMENTS

- WEB APP - Pentium 4 processor or higher, and the computer must have approximately 100MB of free hard drive space and 128MB of RAM.

- ANDROID DEVICE : Intel Atom Processor Z2520 1.2 GHz or faster, the the device must have minimum 850 MB of storage and 512 MB of RAM.

- iOS DEVICE : Any iphone.

CHAPTER 5

6 SYSTEM DESIGN

6.1 Data Flow Diagram

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself.

There are several notations for displaying data-flow diagrams. The notation presented above was described in 1979 by Tom DeMarco as part of structured analysis.

For each data flow, at least one of the endpoints (source and / or destination) must exist in a process. The refined representation of a process can be done in another data-flow diagram, which subdivides this process into sub-processes.

The data-flow diagram is a tool that is part of structured analysis and data modeling. When using UML, the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site-oriented data-flow plan.

Data-flow diagrams can be regarded as inverted Petri nets, because places in such networks correspond to the semantics of data memories. Analogously, the semantics of transitions from Petri nets and data flows and functions from data-flow diagrams should be considered equivalent.
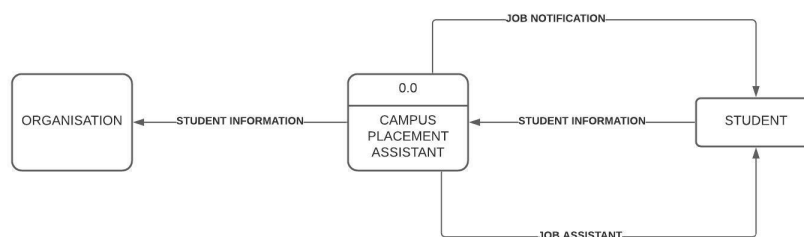
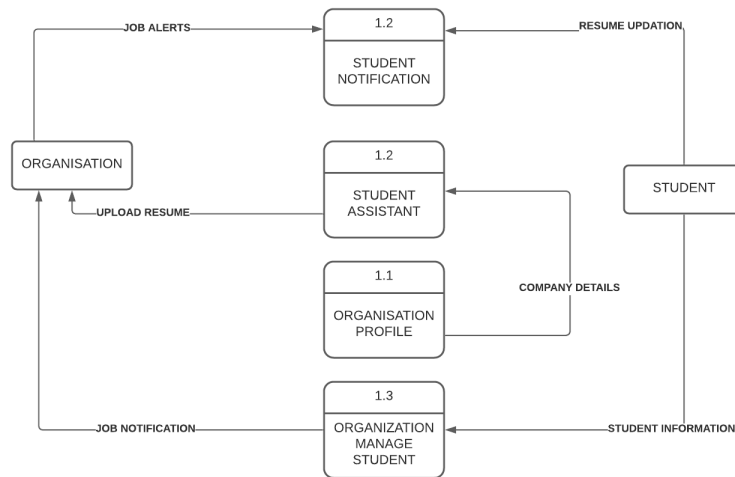**Level 1.1**

Figure 10: level 1.1 DFD

Level 1.2

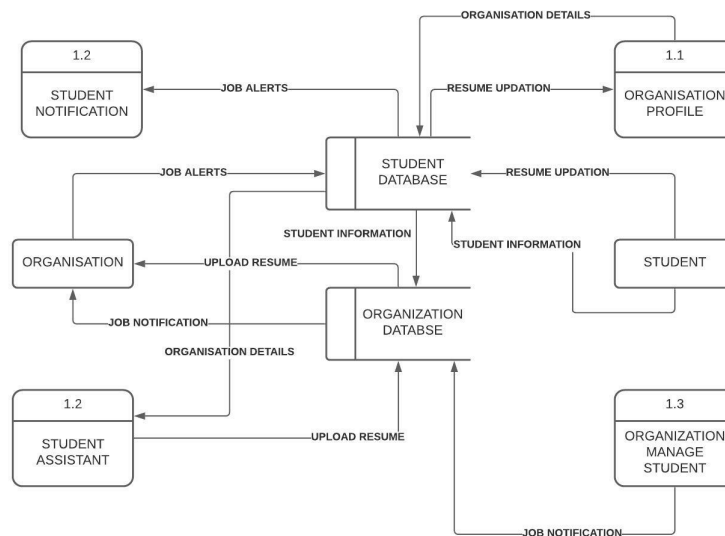Figure 11: level 1.2 DFD

Level 1.3

Figure 12:   level 1.3 DFD

6.2   UML DIAGRAM

6.2.1   Sequence Diagram

A sequence diagram is an interaction diagram that emphasis the time ordering of the messages; a collaboration diagram is an interaction diagram that emphasizes the structural organization of the objects that send and receive messages. Sequence diagrams and collaboration diagrams are isomorphic, meaning that you can take one and transform it into the other. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lines, different processes or objects that lives simultaneously, and as horizontal arrow, the messages exchanged between them, in the order in which they occur. This allows the Specification of Simple's Runtime Scenarios In A Graphical Manner.
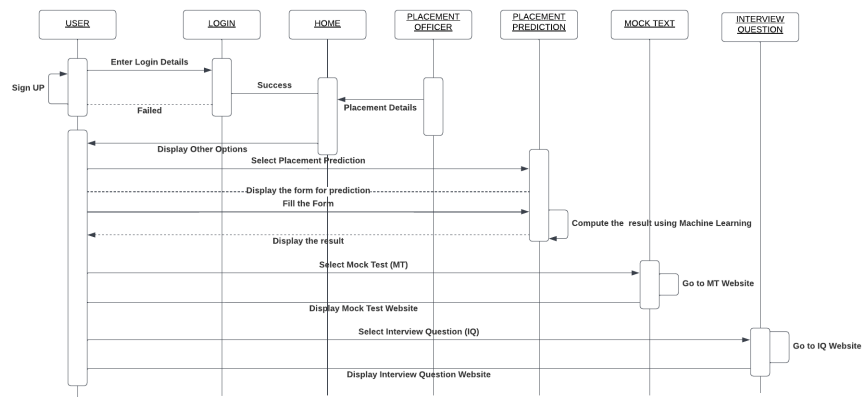

Figure 13: Sequence Diagram

6.2.2   Use Case Diagram

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify and organize system requirements' use case diagram at its simplest is representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

The purpose of use case diagram is to capture the dynamic aspect of a system. Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analyzed to gather its functionalities use case are prepared and actors are identified.
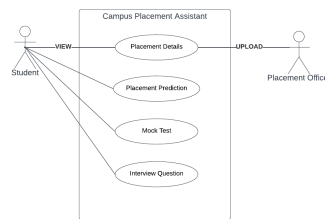

Figure 14: Use Case Diagram

6.2.3   Activiy Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows). Activity diagrams show the overall flow of control. Activity diagrams are constructed from a limited number of shapes, connected with arrows. The most important shape types:

- Rounded rectangles represent actions;

- Diamonds represent decisions;

- Bars represent the start (split) or end (join) of concurrent activities;

- A black circle represents the start (initial node) of the workflow; An encircled black circle represents the end (final node).

Arrows run from the start towards the end and represent the order in which activities happen. Activity diagrams may be regarded as a form of flowchart. Typical flowchart techniques lack constructs for expressing concurrency. However, the join and split symbols in activity diagrams only resolve this for simple cases; the meaning of the model is not clear when they are arbitrarily combined with decisions or loops.
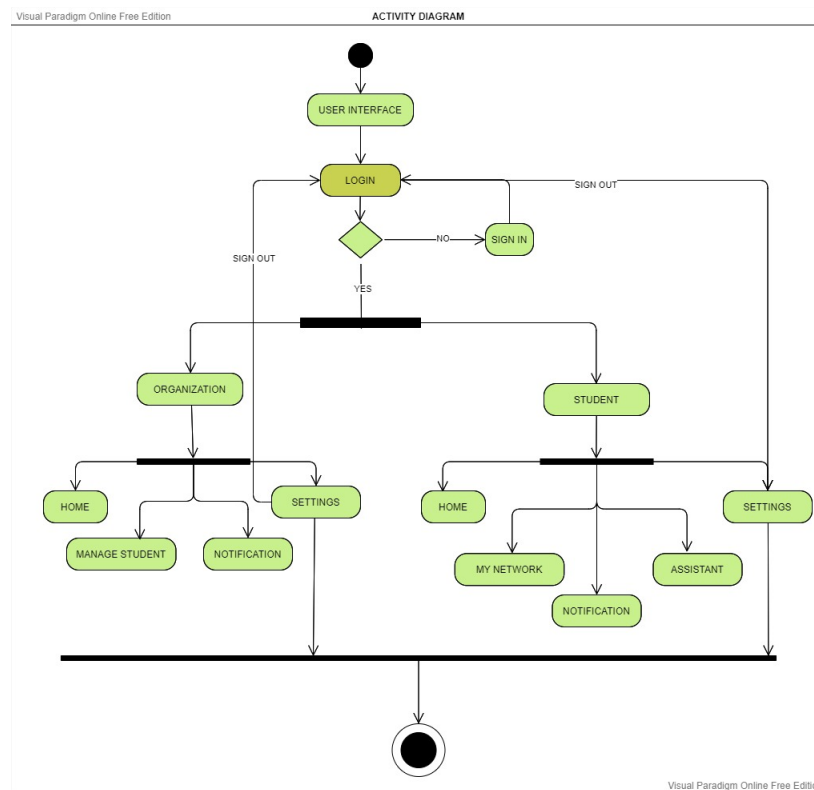
Figure 15: Activity diagram of Student

6.2.4   Class Diagram

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling, translating the models into programming code. Class diagrams can also be used for data modelling.

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of objectoriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.
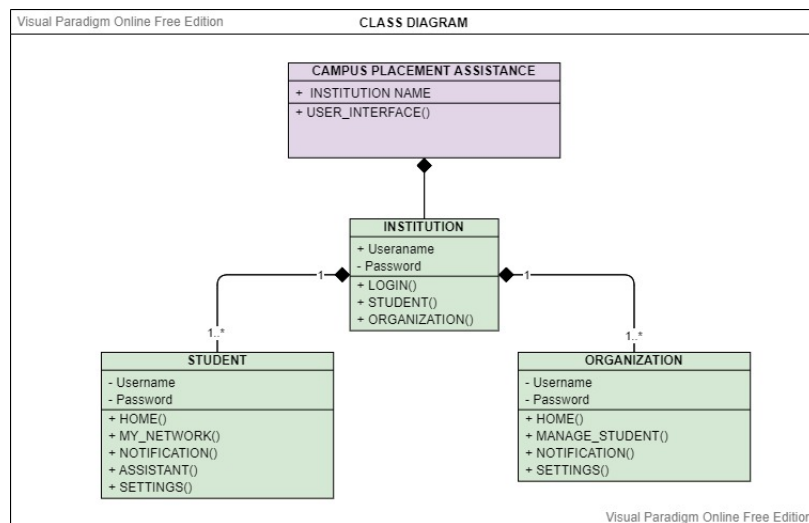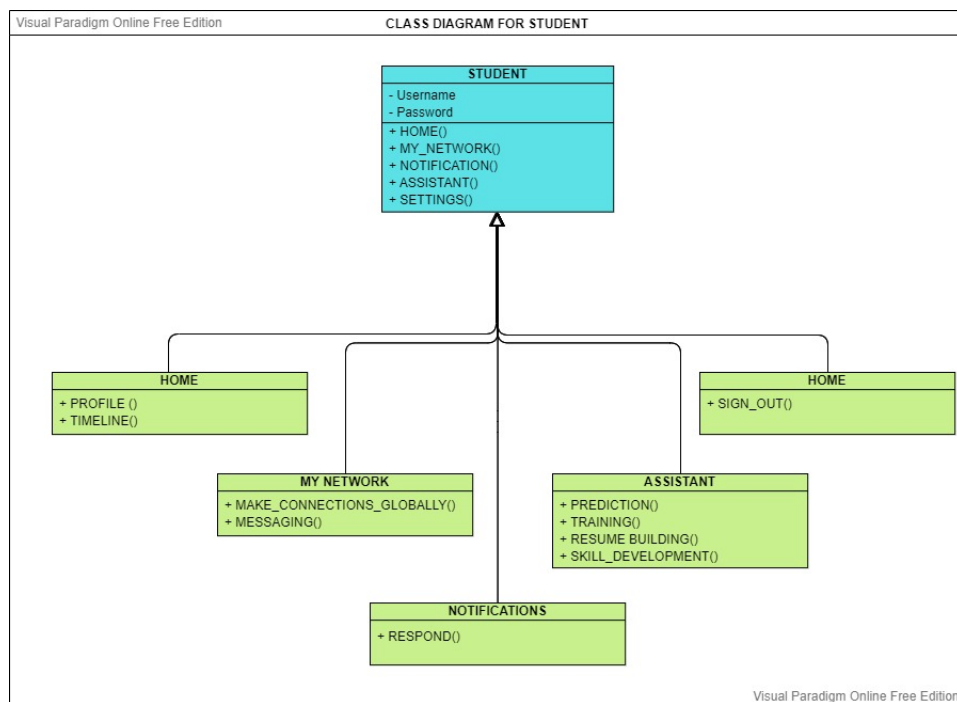
Figure 16: Class Diagram
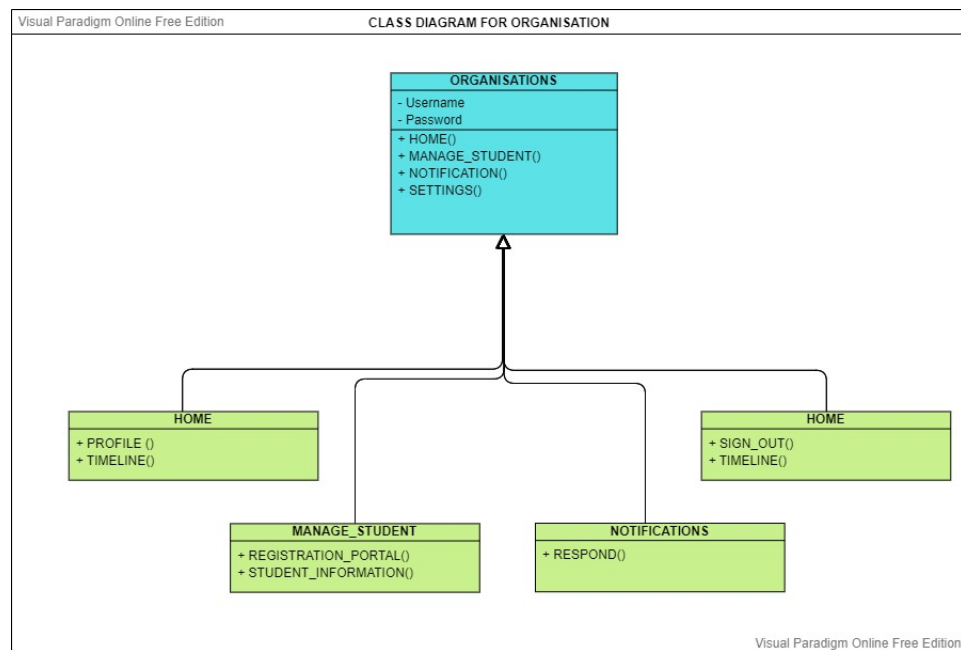

Figure 17: Class Diagram

Figure 18: Class Diagram

CHAPTER 6

7   CONCLUSION

The various Machine Learning algorithms are discussed in the reference of Student survey. The collected dataset is preprocessed enough to utilize effectively. The classification of structure dataset is executed in Weka with all possible nodes and attributes. The dynamic analyze is carried out on structural dataset with different parameters and attributes. The applied ML algorithms is helping to generate high recommendations for prediction of student engagement in various social media platform as LinkedIn, Facebook, Instagram, Snapchat, Telegram, Twitter, Whatsapp, Etc. The different algorithm has derived its best possible outcomes in association of student social engagement. It concludes that random forest can be most desirable option for the same study. Here from the facial expressions determining the behaviors of individual students and analysing their depression.

References

[1] Performance Comparison of Machine Learning Algorithms for Prediction ofg Students' Social Engagement https://doi.org/10.1109/ICCMC51019.2021.9418260

[2] An EDM Approach to the Analysis of Students' Engagement in Online Courses from Constructs of the Transactional Distance https://doi.org/10.1109/ICALT.2016.17

[3] Using Ranking and Multiple Linear Regression to Explore the Impact of Social Media Engagement on Student Performance https://doi.org/10.1109/ICALT.2016.140

[4] Student Academic Performance and Social Behavior Predictor using Data Mining Techniques https://doi.org/10.1109/CCAA.2017.8229794

[5] Student Engagement Predictions in an e-Learning System and Their Impact on Student Course Assessment Scores https://doi.org/10.1155/2018/6347186

[6] Cluster and Logistic Regression Distribution of Students' Performance by Classification