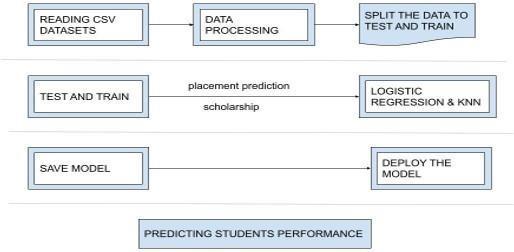
**Chapter 3**

**3.1 Architectural Diagram**

**METHODOLOGY**

**Fig 3.1 Architectural Diagram**

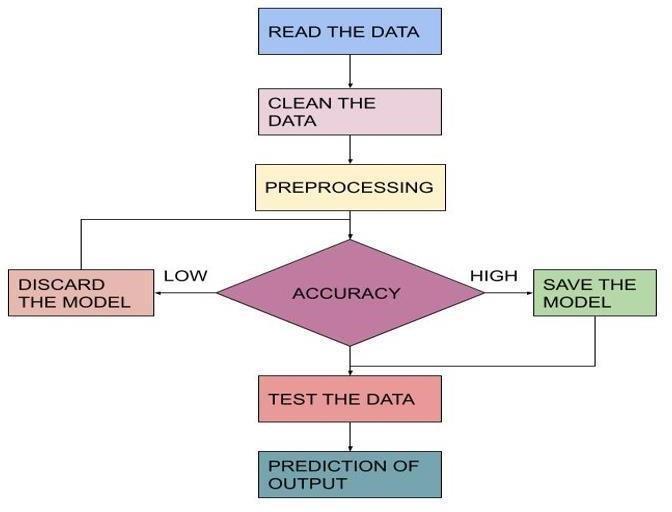
The above diagram shows the Architectural Diagram of the process of prediction. The initial step is extracting the certain features from the CSV file. Since there are lots of datasets extractthe columns they have, segment it and group it together. Once the step is complete data processing is done which cleans the data with null values. Data processing uses the one hot encoding method which converts character value into numeric values. After the completion ofthe process split the data for test and train. Train it using SVM, Neural Network, Naïve Bayes Decision tree, Logistic regression and K Nearest neighbor



**3.2 Detailed Process**

**Fig 3.2 Flow Chart**

The above figure details the process of the project. Firstly read the CSV data collected and sendit to the pre-processing technique where the data is cleaned for null value and then send it to test and train where the data is split for the process. Once the process is complete the tested andtrained data is compressed. Now the accuracy of the algorithms such as logistics regression, decision tree, SVM, Naïve Bayes, and KNN is calculated for the further process. Once the accuracy is calculated save the model of high accuracy and send it for the prediction. The model is sent to a flask where it has a copy of the data and performs the prediction using the given data.



**3.3. Algorithms**

**3.3.1 Logistic Regression**

Logistic regression is another method borrowed from the field of machine learning statistics. which is the way to go for binary classification problems. Logistic regression is named after the logistic function, which is the function used at the core of the method, also called the sigmoid function, which was developed to explain the attributes of population growth in ecology and rapidly. Increases to maximize transport capacity. Environmental. An S-shaped curve that can take the actual value and map it to a value between 0 and 1.

**3.3.2 Naive Bayes**

The naive Bayes classifier is a generative model for classification. Before the advent of deep learning and it's easy-to-use libraries, the Naive Bayes classifier was one of the widely deployed classifiers for machine learning applications. Despite its simplicity, the naive Bayes classifier performs quite well in many applications.

**P(A | B) = P(B | A) P(A) P(B)**

With Bayes Theorem one can find probability that happens because B has occurred. Here is B-proof and A is the hypothesis. The assumptions made here are predictions / characteristics are independent. It is the presence of a particular feature that does not affect the other side. Therefore, it is called innocent. **3.3.3 Random Forest Classifier**

Random Forest is also called a supervised learning algorithm. Both classification and regression are available. It is also the most flexible and easy to use algorithm. A forest is made up of trees. A random forest builds a decision tree. Here is an example of the logistic regression equation:

**y = e^(b0 + b1\*x) / (1 + e^(b0 + b1\*x))**

on a randomly selected sample of data, makes predictions from each tree, and selects the optimal solution through voting. It technically is a method (based on the divide-and-conquer approach) of decision trees that is generated on a randomly data split dataset. **3.3.4 K-Nearest Neighbour (KNN)**

K-nearest neighbour is the non-parametric lazy algorithm. The nearest neighbours are selected based on Euclidean distance calculated between x and y vectors given in the equation (2). The result of KNN varies for different values of K . A large value of K will cause overlapping in classes, while a smaller value of K increases computations.

**Euclidean Distance = √∑**𝑘 𝑖**=1 (**𝑥𝑖 **−**𝑦𝑖**)2**

**3.3.5 Decision Tree Classifier**

The tree structure is similar to the flowchart in which the internal nodes of the decision tree indicate the function (or attribute), the branch indicates the decision rule, and each leaf node shows the result. The top node of the decision tree is called the root node. Learn how to split based on attribute values. Call recursive partitioning to recursively split the tree. This flowchart-like structure is useful for decision-making. Therefore, decision trees are very easy to understand and also to interpret. The basic idea behind any decision tree algorithm is as follows: a. Select the best decision attribute using ASM to split the record. b. Make the attribute as a decision tree node and also break the dataset into a smaller set of subsets. c. Now start the tree building process by repeating the steps recursively for each child until one of the conditions matches: i.all the tuples are now belonging to the same value of attributes. ii. There seems to be no more remaining attributes. iii. There seems to be no more instances.

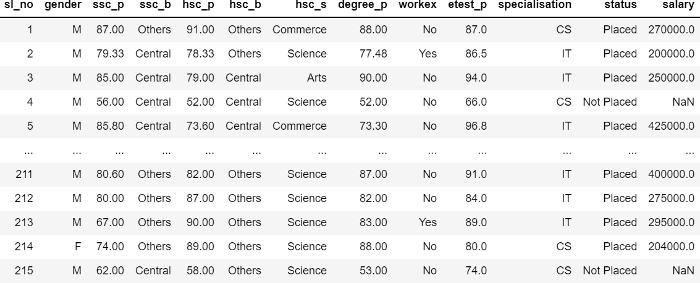
**3.3.6 Support Vector Machine(SVM)** Support vector or SVM machine is one of the most commonly monitored learning algorithms used to classify and regression issues. Mostly, however, it is used to classify problems in learning machines. The aim of the SVM algorithm is to prepare the best line or decision-making limitthat can separate the N-dimensional space in classes, so it can easily place new data points in the future category. This best decision-making limit is called a hyperplane. SVM Select ExtremePoints / Vectors to create Super Flat. These extreme cases are referred to as support vectors, and because the algorithm is referred to as a carrier vector. The SVM classifier offers better accuracy and faster predictions. It also uses less memory because it uses a subset of educational points during the decision-making phase. SVMs work clearly with separation margins and higher dimensional spaces.

**`** **Chapter 4**

**4.1 Datasets**

**IMPLEMENTATION**

Two datasets were collected to determine the performance of students. For the placement analysis, the influencing factors of academic and employability that aids in a student's placement. The dataset was downloaded from Kaggle. The Campus placement information includes secondary and higher secondary school percentages and specialization. It also includes degree specialization, type and Work experience and salary offers to the placed students. The sample data collected consisted of approximately 215 students that included 12 attributes. For Scholarship analysis, we created a dataset that included a total of 5 attributes of 181 students. This includes Students Name, Scholarship Name, Caste, Marks of



**Fig 4.1 Insights of Dataset**

**TABLE 1. STUDENT PLACEMENT DATASET ATTRIBUTES**

**Attribute name Datatype Category**

Sl No Numerical Range 0-215

Gender Nominal M, F

ssc\_p Numerical Secondary Education

percentage- 10th Grade

ssc\_b Nominal Board of Education-

Central/ Others

hsc\_p Numerical Higher Secondary Education

percentage- 12th Grade

hsc\_b Nominal Board of education-

Central/Others

hsc\_s Nominal Specialization in Higher

Secondary Education-

Science, Commerce, Arts

degree\_p Numerical Degree Percentage

workex Nominal Work Experience- Yes/No

etest\_p Numerical Employability test

percentage (conducted by

the college)

specialization Nominal Specialization -CS/ IT

status Nominal Status- Placed/ Not Placed

salary Numerical Salary after placed

**TABLE 2. STUDENT SCHOLARSHIP DATASET ATTRIBUTES**

**Attribute name Datatype Category**

Sl no Numerical Range from 1-181

Students Name Nominal Name of Students

Scholarship Name Nominal Name of scholarship

Caste Nominal Catholic, Muslim, Others

Marks of Previous Year Numerical Marks of Previous Year

Gender Nominal M, F

Annual Income Numerical Income less than 2,00,000

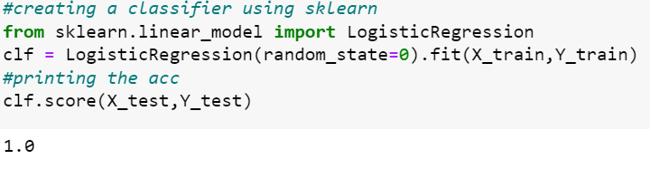
**4.2 Experimental Setup**

Our experiment begins by collecting the dataset. Once the collection of the dataset is completed, the preprocessing of that experimental data begins. The pre-processing includes the cleaning of the dataset, analysis of the dataset, and training of the model with the dataset.

**4.2.1 Import all the Required Libraries** For the language we used, if it need to identify any function we worked, we need to import all the required libraries into it. All the necessary libraries need to be imported into the working environment like Pandas, NumPy, Matplotlib, Pickle, and Scikit-learn.

**4.2.2 Data set Importing**

The collected dataset which is in CSV format, needs to be imported into the workspace. As the columns of sl. No and salary are not required to predict student placement status, those two columns are dropped. Similarly, for Scholarship, sl. No and Student Name is dropped. **4.2.3 Dealing with the Missing Values** Missing values are ones whose data does not appear in the relevant column. A dataset with a higher



**4.2.4 Label Encoding the Data** Label encoding is mainly done to make sure that all the data is in numerical format. The categories in the ‘hsc\_s’ and ‘degree\_t’ are split into individual columns on applying the dummy encoding. It uses ‘1’indicating ‘YES’ and ‘0’ indicating ‘NO’. Here, the number of newly created columns equals to the number of categories. The following columns ['gender','ssc\_b','hsc\_b' ’hsc\_s’,'workex','specialisation','status'] are also converted to ‘0’ and ‘1’s. Similarly, for the Scholarship Analysis, caste is encoded to Catholic-1, Muslim-2, Others-3.

**4.2.5 Dataset Splitting** Data set need to be divided into two sections, Training and Testing data. Training data is the major part of the dataset which is used to feed the Machine learning model to recognise the patterns. Testing data is data which is used to compute the accurate result of the model. Data can be splitted in the ratio of 70:30, 60:40, 80:20. In our Project, the dataset is divided into 80:20 ratio of Training and Testing respectively. For Placement Analysis, the Training data considered was Gender, ssc\_p, ssc\_b, hsc\_p, hsc\_b, hsc\_s, degree\_p, workex, etest\_p, specialization and for testing data we used status. For Scholarship Analysis, the Training data considered were caste, marks of previous year, gender, annual income and Testing they used scholarship Name.

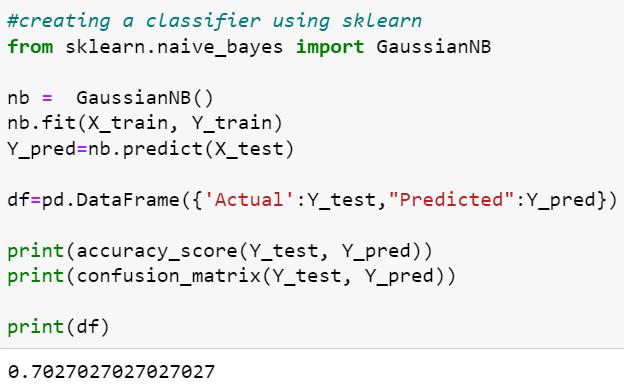
**4.2.6 Algorithms Used**

For Placement Analysis, we used Logistic Regression Algorithm

**Fig 4.2.2 Pseudocode of Logistic Regression**

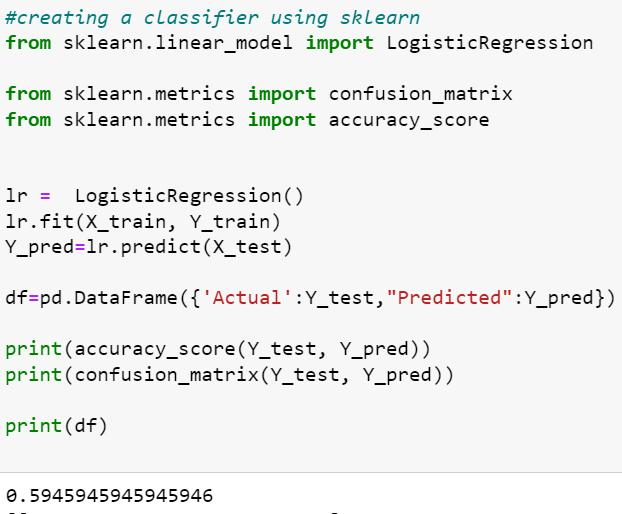
**Fig 4.2.3 Model Save of Logistic Regression**

For Scholarship Analysis, we used KNN, Logistic Regression, Naïve Bayes, Decision Tree, and SVC. • Logistic Regression

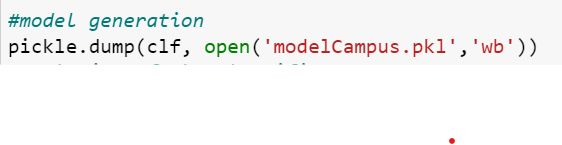


**Fig 4.2.4 Pseudocode of Logistic Regression**

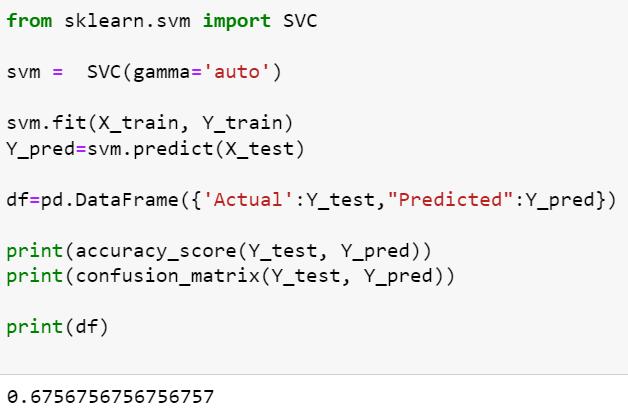
• Naïve Bayes



**Fig 4.2.5 Pseudocode of Naïve Bayes**



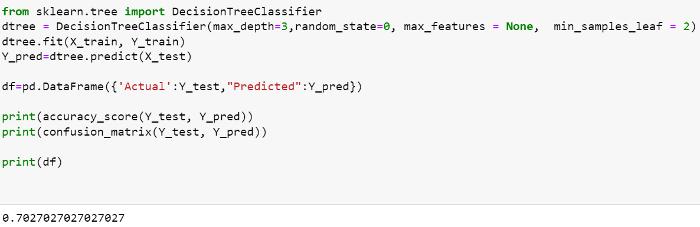
• KNN



**4.2.6 Pseudocode of KNN**

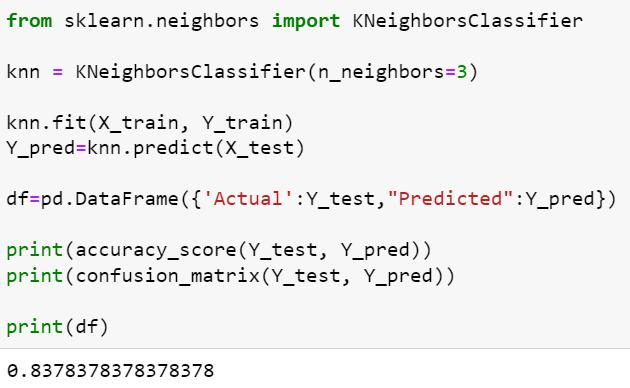
**4.2.7 Model save of KNN**

• Decision Tree



**Fig 4.2.8 Pseudocode of Decision Tree**

• SVM





**Fig 4.2.9 Pseudocode of SVM** **4.3 Software Used**

**4.3.1 Flask**

Flask is a micro web framework written in Python. Flask was created by Armin Ronacher of Pocoo, an international group of Python enthusiasts formed in 2004. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object- relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Flask is based on the Werkzeg WSGI toolkit; a toolkit that implements requests, response objects, and utility functions. This enables a web frame to be built on it. and the Jinja2 template engine, a web template system that combines atemplate with a specific data source to render a dynamic web page.

**4.3.2 Visual Studio**

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code. Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a code profiler, designer for building GUI applications, web designer,class designer, and database schema designer. It accepts plug-ins that expand the functionalityat almost every level including adding support for source control systems (like Subversion and Git) and adding new toolsets like editors and visual designers for domain- specific languages or toolsets for other aspects of the software development lifecycle (like the Azure DevOps client: Team Explorer). **4.3.3 Jupyter Notebook**

The Jupyter Notebook App is a server-client application that allows editing and runningnotebook documents via a web browser. The Jupyter Notebook App can be executed on a localdesktop requiring no internet access or can be installed on a remote server and accessed through the internet. In addition to displaying/editing/running notebook documents, the Jupyter Notebook App has a “Dashboard” a “control panel” showing local files and allowing to open notebook documents or shutting down their kernels.

**4.4 Hardware used with Justification**

**•**Processor: Intel(R) core (TM) i5

• RAM: 8.00 GB

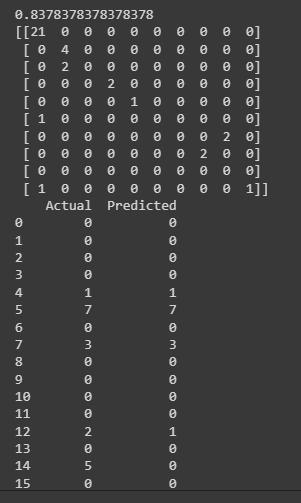
• Speed: 1.80GHz

• Hard Disk: 1TB

• System Type: 32/64-bit Operating System **Chapter 5**

**RESULTS AND DISCUSSION**

Our solution enables instructors to know student levels in advance and identify the students who need extra help by efficiently and accurately predicting student performance using machine learning techniques.



• Logistic Regression provides the maximum accuracy, which is 100%, in placement analysis.

**Fig 5.1.Accuracy of Logistic Regression**

• KNN provides a highest accuracy of 83 % in scholarship analysis.



**Fig 5.2. Accuracy of KNN**

**Chapter 6**

**CONCLUSION AND FUTURE SCOPE**

Predicting student’ performance is mostly useful to help the educators and learners improving their learning and teaching process. Several tools have been designed till today for betterment and evaluation of student’s performance. The result produced by these tools can help in decision making that improves student’s performance. This project assesses the usefulness of machine learning algorithms to predict the student’s with inadequate result. The prime aim of this project is to identify the students who are eligible for placements at an early stage, so they will have enough time to rework and attain a satisfactory result which gives eligibility to placements. Secondly, the project designed to predict the student’s eligible for scholarship which helps the student’s in empowering academic and career goals by removing the financial barriers. We pre-processed the two different datasets for placement and scholarship having 252 and 181 instances respectively and applied a set of machine learning algorithms, logistic regression, k-NN, decision tree, naïve bayes, SVM, random forest classifier, to come up with most appropriate prediction model. Logistic regression prevails by achieving the accuracy of 100% and k-NN of 83%. The chosen model of high accuracy is then deployed to flask where prediction process takes place. The interpretation reveals CGPA/percentage of the academics, grades in pre-requisite courses, caste and annual income as features which quantifies student’s eligible for placement and scholarship. The instructor must present additional care to the student’s who are not eligible for placements and provide knowledge to improve with academics and placement processes, so that at the end of academics they are eligible to place in a reputed company. In future, we would like to extend the model to provide with feedback direct to the student’s as well as the instructor so that both instructor and student’s can improvise the academics with their own manner. This will increase the efficiency of learning and progress with both placements and academics. We also intend to extend the prediction of multiple scholarships for a student and the prediction of suitable company a student can hold with his academic and co-curricular activities. We aim to apply model in other courses and extend the notion to entire set of courses in the institution.

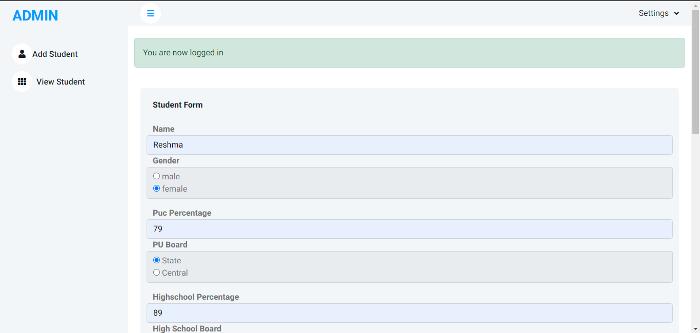
**7. REFERENCES**

[1] Dindar Mikaeel Ahmed, Adnan Mohsin Abdulazeez, Diyar Qader Zeebaree. 2021- July. Predicting University’s Students Performance based on Machine Learning. 2021 IEEE International Conference on Automatic Control and Intelligent systems (I2CACIS 2021) [2] V. Uday Kumar, Azmira Krishna, P. Neelakanteshwara, Zeelan Basha. 2020- September. Advanced prediction of performance of a student in an university using Machine learning technique. ICESE 2020. IEEE Xplore pn: CFP20V66-ART [3] M. Irfan Uddin. 2021-August. Performance Prediction for Higher Education Student’s using Deep learning. Hindawi Complexity. Volume 2021, Article Id 9958203. [4] Ijaz Khan, Abdul Ahmed, Nafaa jabeur, Mohammad Mahadi. 2021-September.An AI approach to monitor students performance and devise preventive measures. Khan et al.Smart Learn.Environ. [5] Amirah Shahiri, Wahidah Hussain, Nur’Ani Abdul Rashid, 2020. A Review on Predicting Student’s Performance using Data Mining Techniques .Procedia computer science 72. [6] Atul Prakash Prajapati, Sanjeev Sharma, Manish Sharma. 2019-October. Student’s performance analysis using Machine learning Tools. International journal of scientific and engineering research. Vol 8. ISSN 2229-5518 [7] Ahajjam Tarik, Haidar Aissa, Farhaoui Yousef. 2021-March. Artificial Intelligence and Machine Learning to predict student’s performance during Covid-19. The 3rd International workshop on Big Data and Business Intelligence (BDBI 2021). [8] Juan L Rastrollo, Juan A. Gomez-Pulido, Arturo Duran. 2020-Febraury. Analyzing and predicting student’s performance by means of Machine Learning. [9] F.Okubo, A shimanda, T Yamashita, H Ogata. 2019-March. A neural Network Approach for student’s performance prediction. [10] Byung- Hak, Ethan Vizitei, Varun Ganapati. 2019. Student performance prediction with Deep Learning. CA 9404

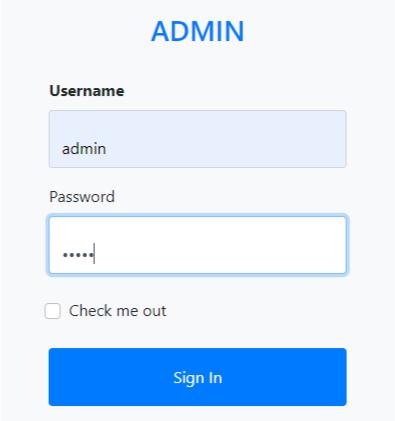
**8. APPENDIX**



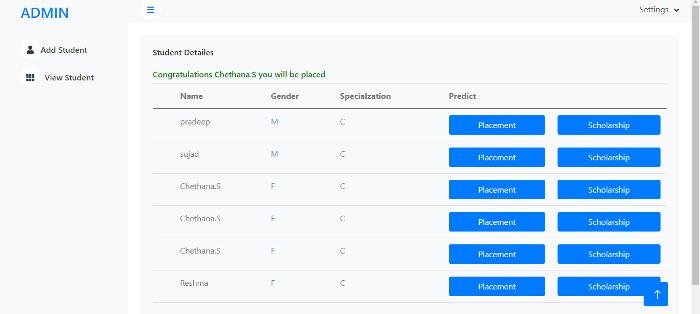
**Fig 8.1: Home page**



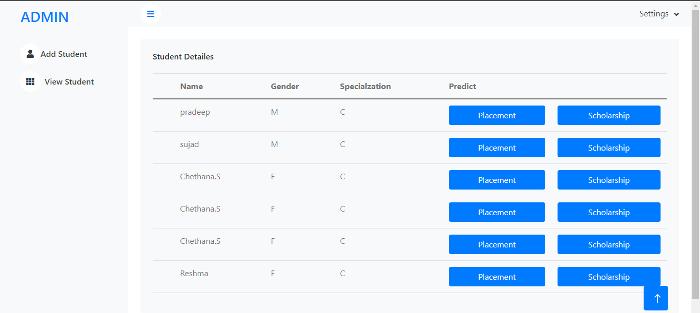
**Fig 8.2: Login Page**



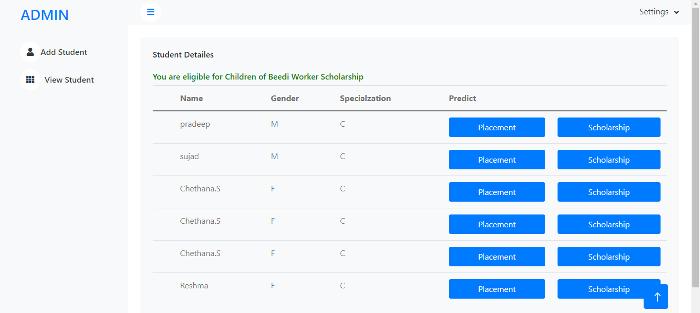
**Fig 8.3: Inserting details of Students**



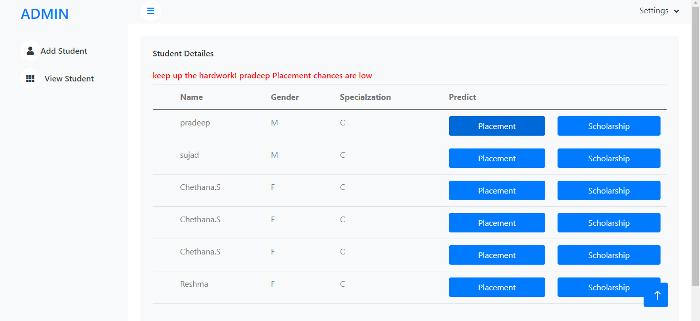
**Fig 8.4: Viewing the Student Details**



**Fig 8.5: Prediction a particular student is placed**



**Fig 8.6: Prediction of Particular student is not placed.**



**Fig 8.7: Predicting which scholarship a student will be eligible for.**

Improved Students Performance Prediction Using Machine Learning Techniques

**Fig 8.8: Predicting which scholarship a student will be eligible for.**

