STUDENT PLACEMENT PREDICTION USING MACHINE LEARNING

BATCH-C10:

Supriya Manda (16241A05F2)

Ayesha Shaik (16241A05C8)

Shivani Silvery (17245A0524)

Santhoshi Rodda(17245A0535)

PROJECT GUIDE:

V.Anusha(Assistant Professor)

ABSTRACT

All students dream to obtain a job offer in their hands before they leave their college. A placement chance predictor helps students to have an idea about where they stand and what to be done to obtain a good placement. A placement predictor is a system that could predict the possibility or the type of company a pre-final year student has chances to be placed. Thus a prediction system could help in the academic planning of an institution for future years. This paper presents a recommendation system that predicts whether the current student will be placed or not, if the student is placed the company is also predicted based on the data of previously placed students.Here we use three different machine learning classification algorithms, namely logistic regression,decision tree and Random forest. These algorithms independently predict the results and we then compare the efficiency of the algorithms, which is based on the dataset. This model helps the placement cell at intervals a corporation to spot the potential students and improve their technical and social skills.

INTRODUCTION

Placement of scholars is one of the vital activities in academic establishments. Admission and name of establishments primarily depends on placements. Hence all institutions strive to Strengthen placement department. Here, the objective is to analyze previous year's student's data and use it to predict the placement chance of the current students. This model is proposed with an algorithm to predict the same.

PROJECT PROTOTYPE

The main Objective of this project is to analyze previous year’s student’s historical data and predict placement possibilities of current students which in turn aids to increase the placement percentage of the institutions.

PROBLEM DEFINITION

The problem is to collect previous students data and analyze those data for new students in institute for prediction purpose.

LITERATURE SURVEY

Title : Prediction of Campus Placement Using Data Mining Algorithm-K nearest neighbor

Author : Mangasuli Sheetal B1 , Prof. Savita Bakare2

Abstract :

Data Mining is “the process of extracting useful information from a large scale data set”. It is a powerful tool to be considered best in the field of education. Educational data mining involves the new methods and its approaches for discovering the knowledge by analysing the database sets to support the decision making process in educational institution. It interprets an effective method for mining the student’s performance based on the database sets to predict and analyse whether a student (he/she) will be recruited or not in the campus placement. The placement of a student not only depends on his academic capabilities but also involves the attributes such as co-curricular activities, communication skills etc. Using these datasets and attributes, predictions are made using the Data Mining Algorithm “K nearest neighbor (KNN)”. The results obtained from each approaches are then compared with respect to their “performance” and “accuracy” levels by graphical analysis and thus the decisions are made towards the best prediction in the campus placement.

EXISTING SYSTEM

They used **WEKA** as the data mining tool to build the model using random tree algorithm. They also used ID3, Bayes Net, RBF network, J48, algorithms on the student data set.The accuracy using ID3 and J48 is 71%,Bayes Net is 70% accurate.It has been observed that most of the approaches are rooted on the decision tree algorithm. However, they emphasis on the algorithms used in order to increase the accuracy of K nearest neighbour.

PROPOSED SYSTEM

This project is proposed with an algorithm to predict the same. Data pertaining to the study were collected from an institution for which the placement prediction is done, and also suitable data pre-processing methods were applied. This proposed model is also compared with other traditional classification algorithms with respect to accuracy, precision and a recall. They resolved that the Random Tree algorithm is more accurate with 73% for the classification/prediction of the model.

ADVANTAGE

The proposal aims to analyze student’s demographic data, study related details and psychological characteristics in terms of final state to figure whether the student will placed or not .

The proposed algorithm performance significantly we compared and state better one .

MACHINE LEARNING MODULES

- Data collection
- Data pre-processing
- Feature Extraction
- Evaluation Model

DATA COLLECTION

Data used in this project is a set student information from college records. This step is concerned with selecting the subset of all available data that you will be working with. ML problems start with data preferably, lots of data (examples or observations) for which you already know the target answer. Data for which you already know the target answer is called *labelled data*.

DATA PRE-PROCESSING

- Organize your selected data by formatting, cleaning and sampling from it.
- Three common data pre-processing steps are:
  1. Formatting
  2. Cleaning
  3. Sampling

- **Formatting:** The data you have selected may not be in a format that is suitable for you to work with. The data may be in a relational database and you would like it in a flat file, or the data may be in a proprietary file format and you would like it in a relational database or a text file.

- **Cleaning:** Cleaning data is the removal or fixing of missing data. There may be data instances that are incomplete and do not carry the data you believe you need to address the problem. These instances may need to be removed. Additionally, there may be sensitive information in some of the attributes and these attributes may need to be anonym zed or removed from the data entirely.

- **Sampling:** There may be far more selected data available than you need to work with. More data can result in much longer running times for algorithms and larger computational and memory requirements. You can take a smaller representative sample of the selected data that may be much faster for exploring and prototyping solutions before considering the whole dataset.

FEATURE EXTRACTION

Next thing is to do Feature extraction is an attribute reduction process. Unlike feature selection, which ranks the existing attributes according to their predictive significance, feature extraction actually transforms the attributes.

EVALUATION MODEL

Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can easily generate over optimistic and over fitted models.

SOFTWARE AND HARDWARE REQUIREMENTS

SOFTWARE

python 2.7

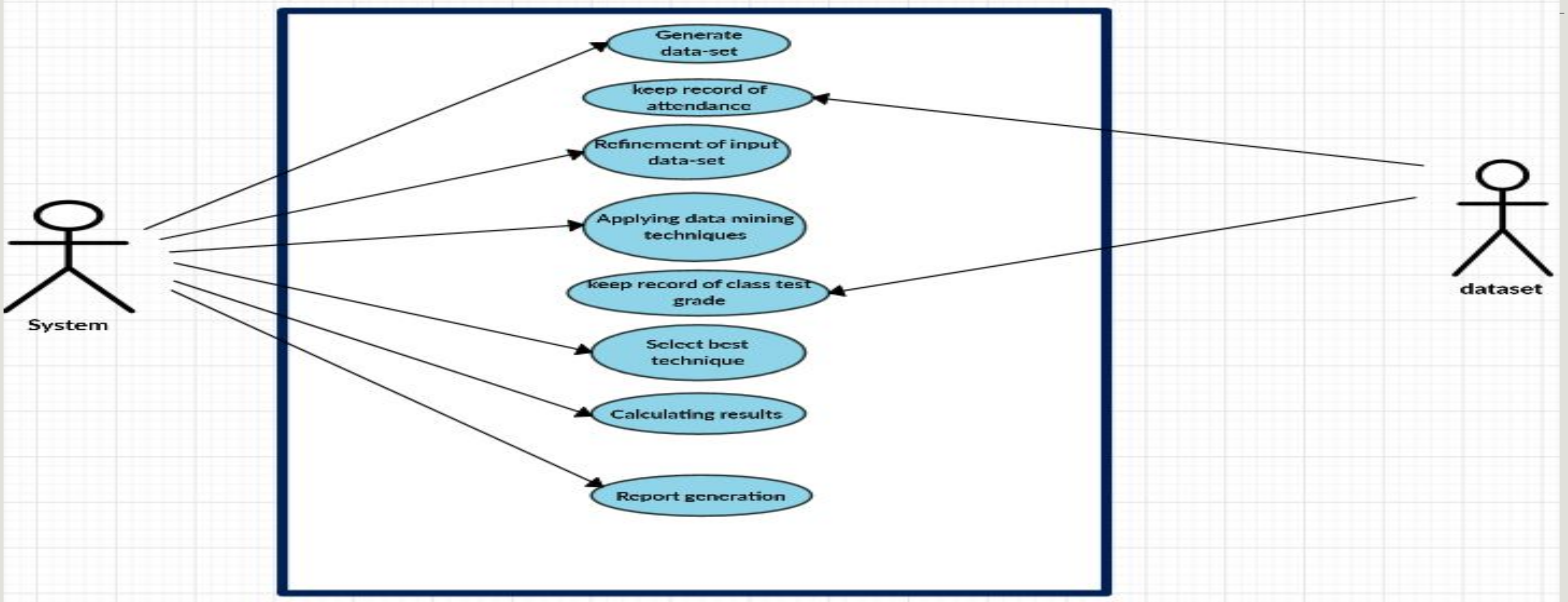
anaconda navigator

HARDWARE

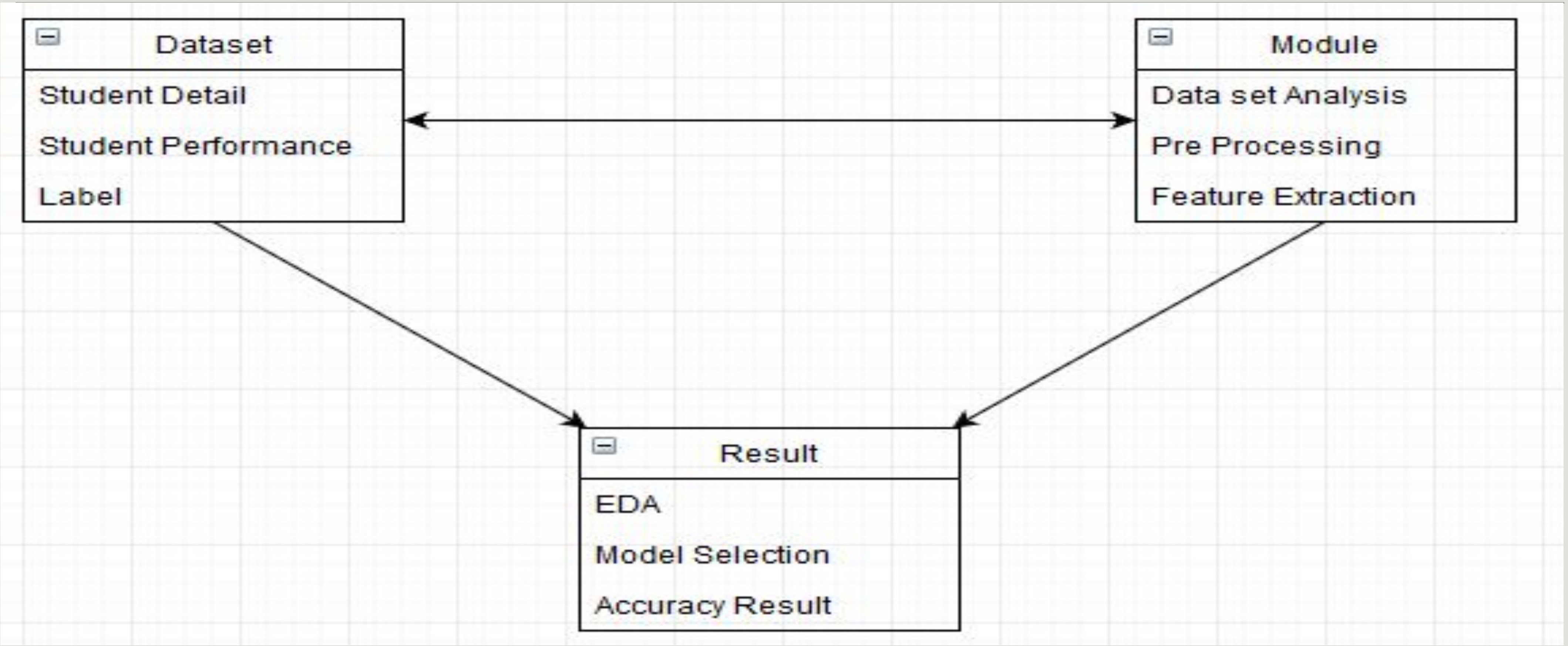
windows 7,8,10 64bit

ram 4gb

UML Diagram – Use Case


Generate data-set
keep record of attendance
Refinement of input data-set
Applying data mining techniques
keep record of class test grade
Select best technique
Calculating results
Report generation
System
dataset

Class Diagram



Dataset

Student Detail

Student Performance

Label

Module

Data set Analysis

Pre Processing

Feature Extraction

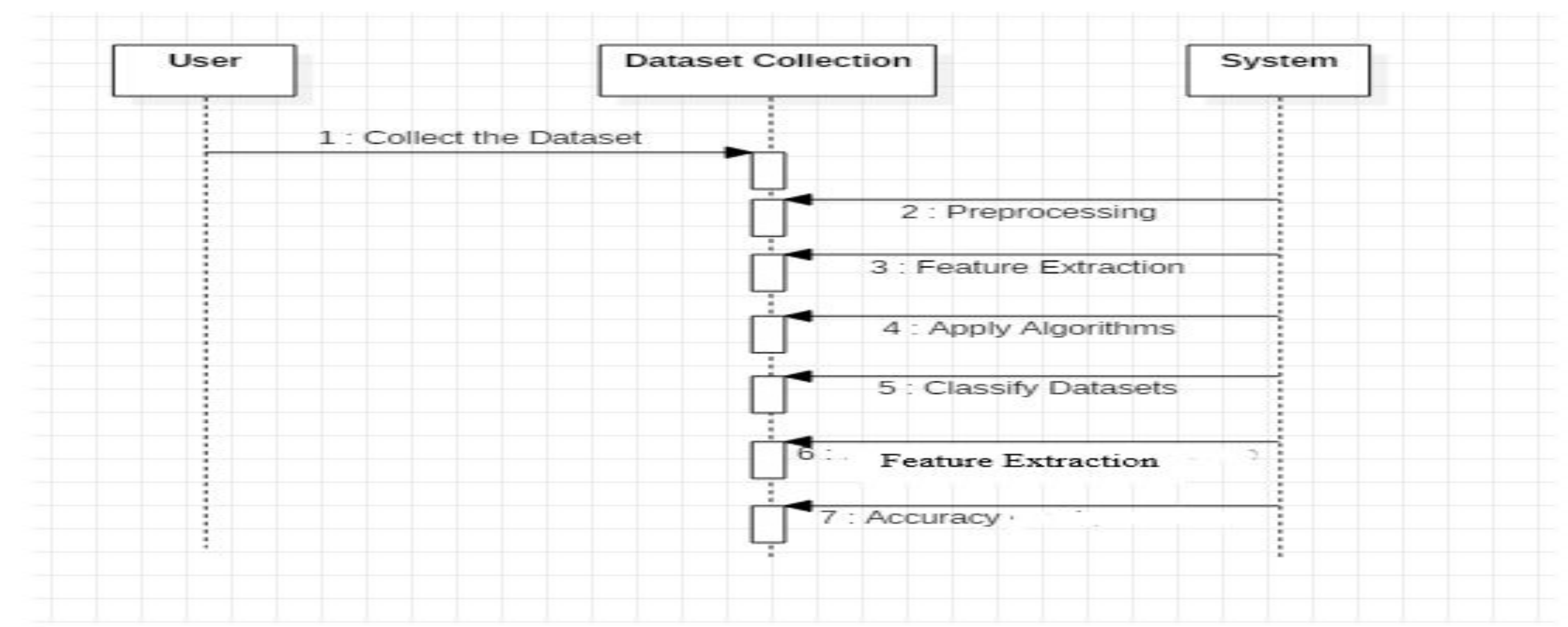
Result

EDA

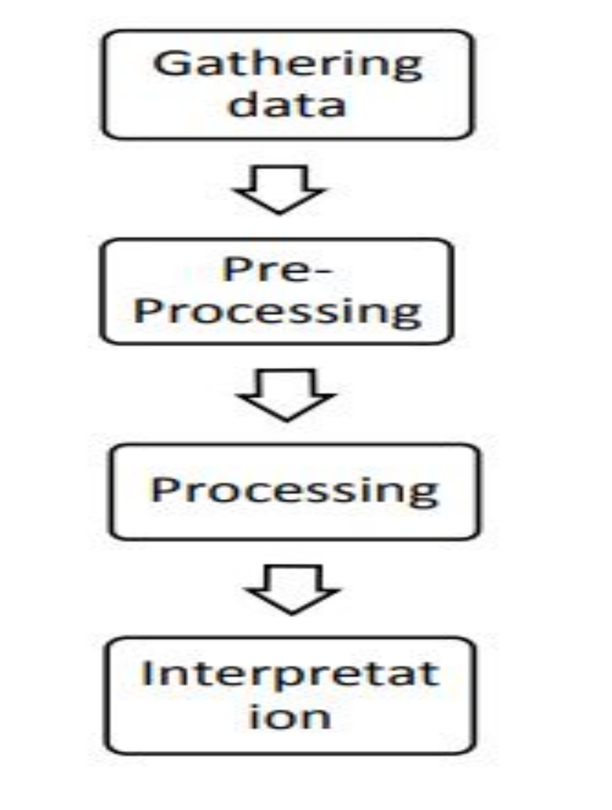
Model Selection

Accuracy Result

Sequence Diagram



FLOWCHART AND SYSTEM ARCHITECTURE

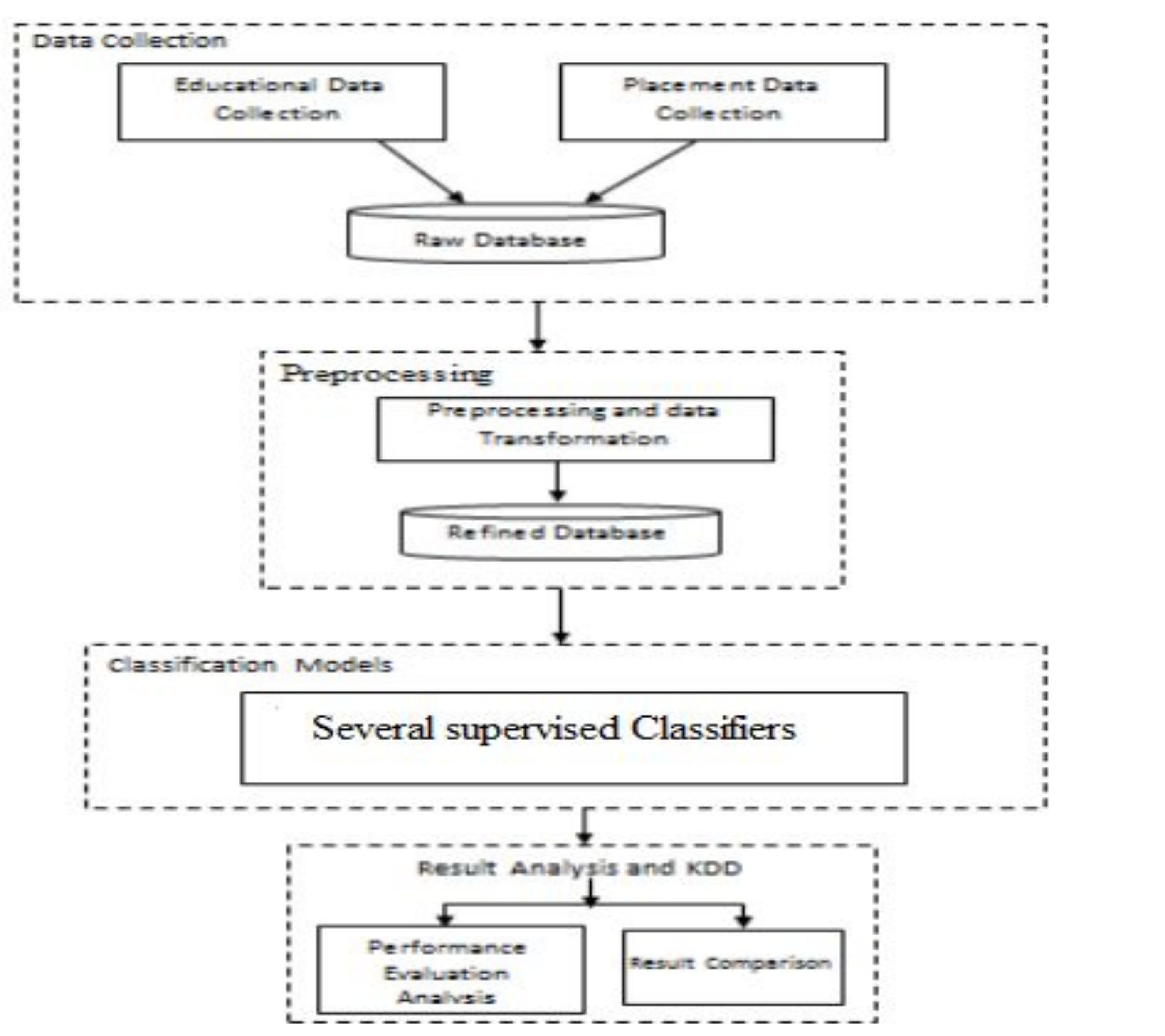


Gathering data

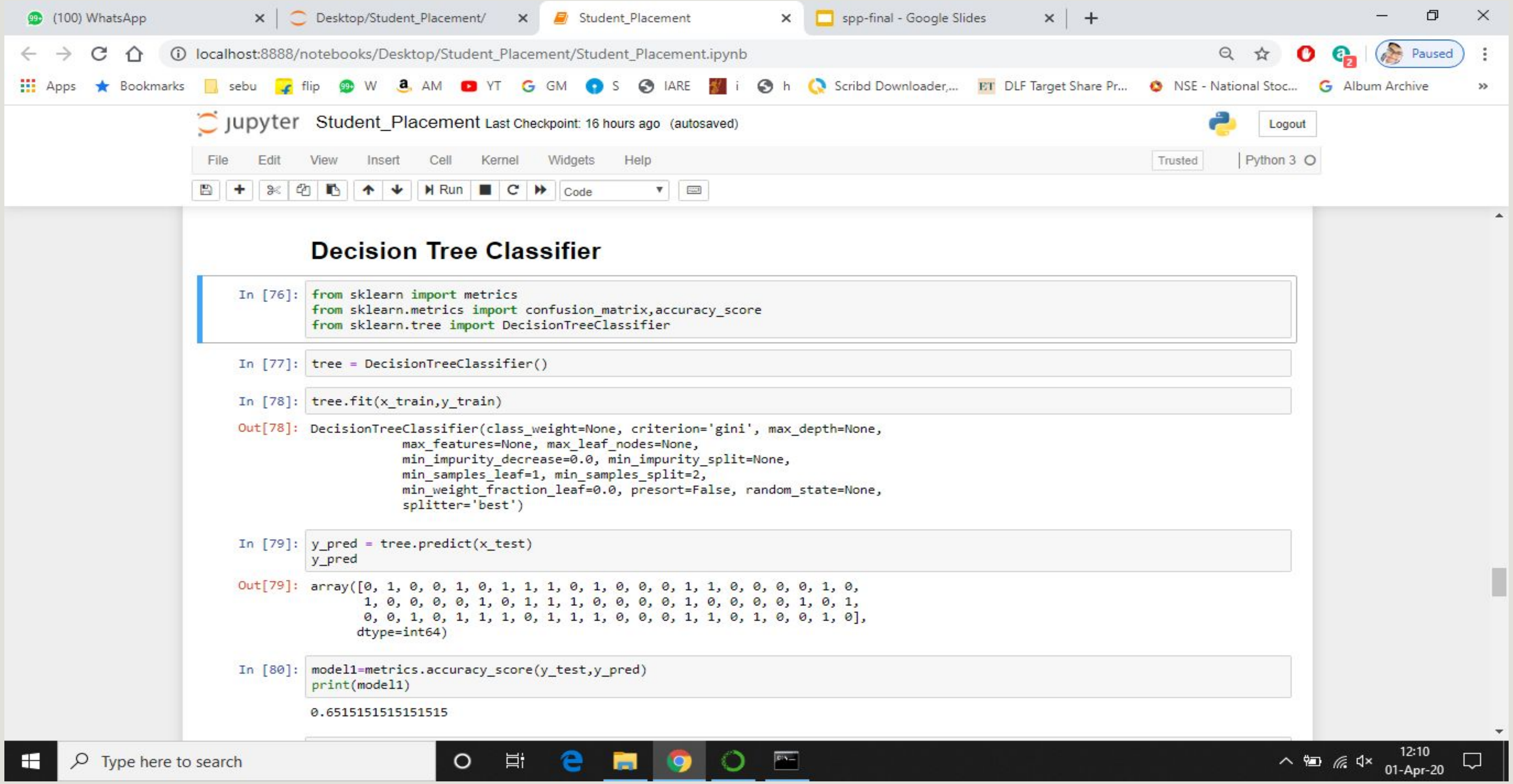
Pre-Processing

Processing

Interpretat ion


Data Collection
Educational Data Collection
Placement Data Collection
Raw Database
Preprocessing
Preprocessing and data Transformation
Refined Database
Classification Models
Several supervised Classifiers
Result Analysis and KDD
Performance Evaluation Analysis
Result Comperison

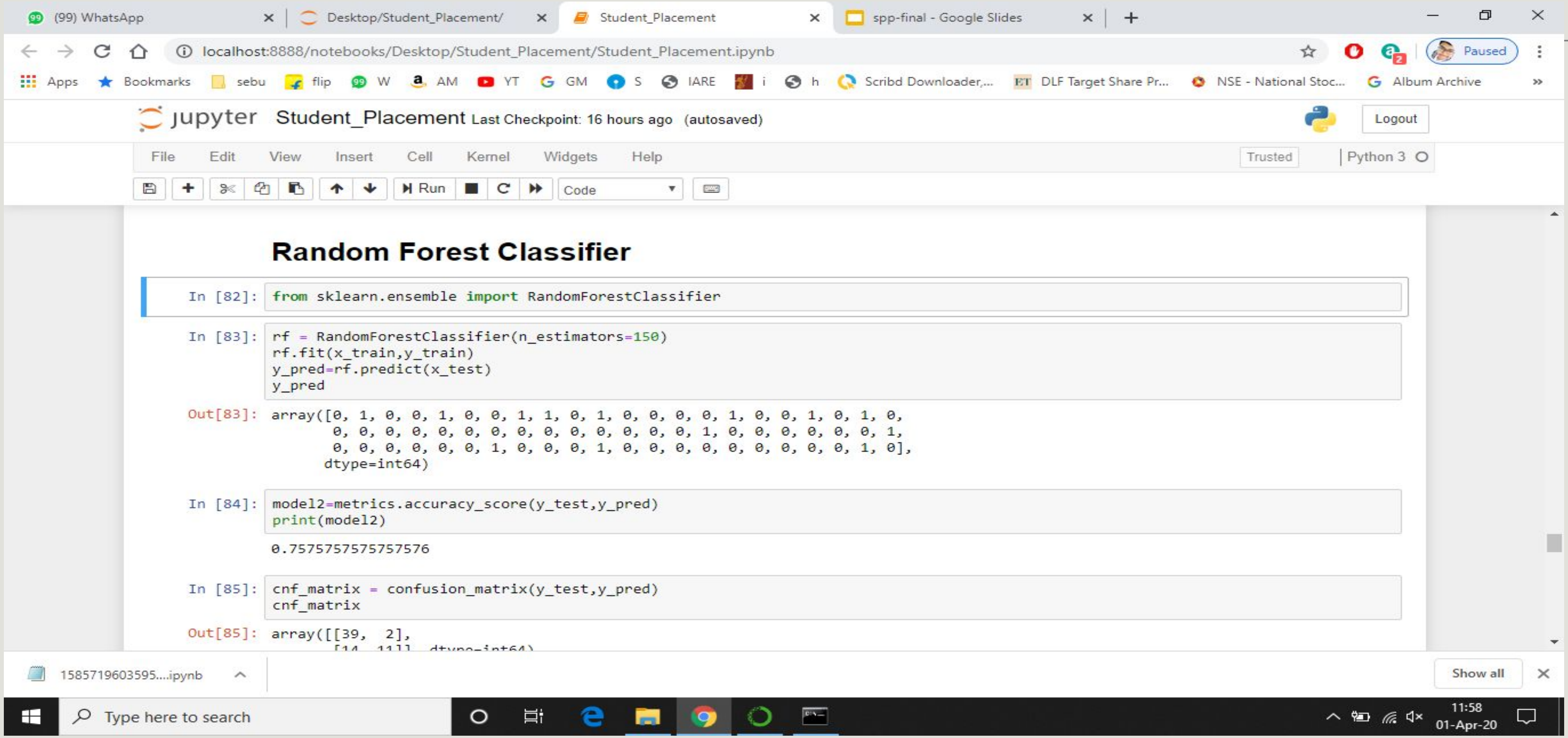
CODE



Decision Tree Classifier

```
In [76]: from sklearn import metrics
         from sklearn.metrics import confusion_matrix,accuracy_score
         from sklearn.tree import DecisionTreeClassifier

In [77]: tree = DecisionTreeClassifier()

In [78]: tree.fit(x_train,y_train)

Out[78]: DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
                     max_features=None, max_leaf_nodes=None,
                     min_impurity_decrease=0.0, min_impurity_split=None,
                     min_samples_leaf=1, min_samples_split=2,
                     min_weight_fraction_leaf=0.0, presort=False, random_state=None,
                     splitter='best')

In [79]: y_pred = tree.predict(x_test)
         y_pred

Out[79]: array([0, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 0,
                1, 0, 0, 0, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1,
                0, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0, 0, 1, 0],
               dtype=int64)

In [80]: model1=metrics.accuracy_score(y_test,y_pred)
         print(model1)

         0.6515151515151515
```

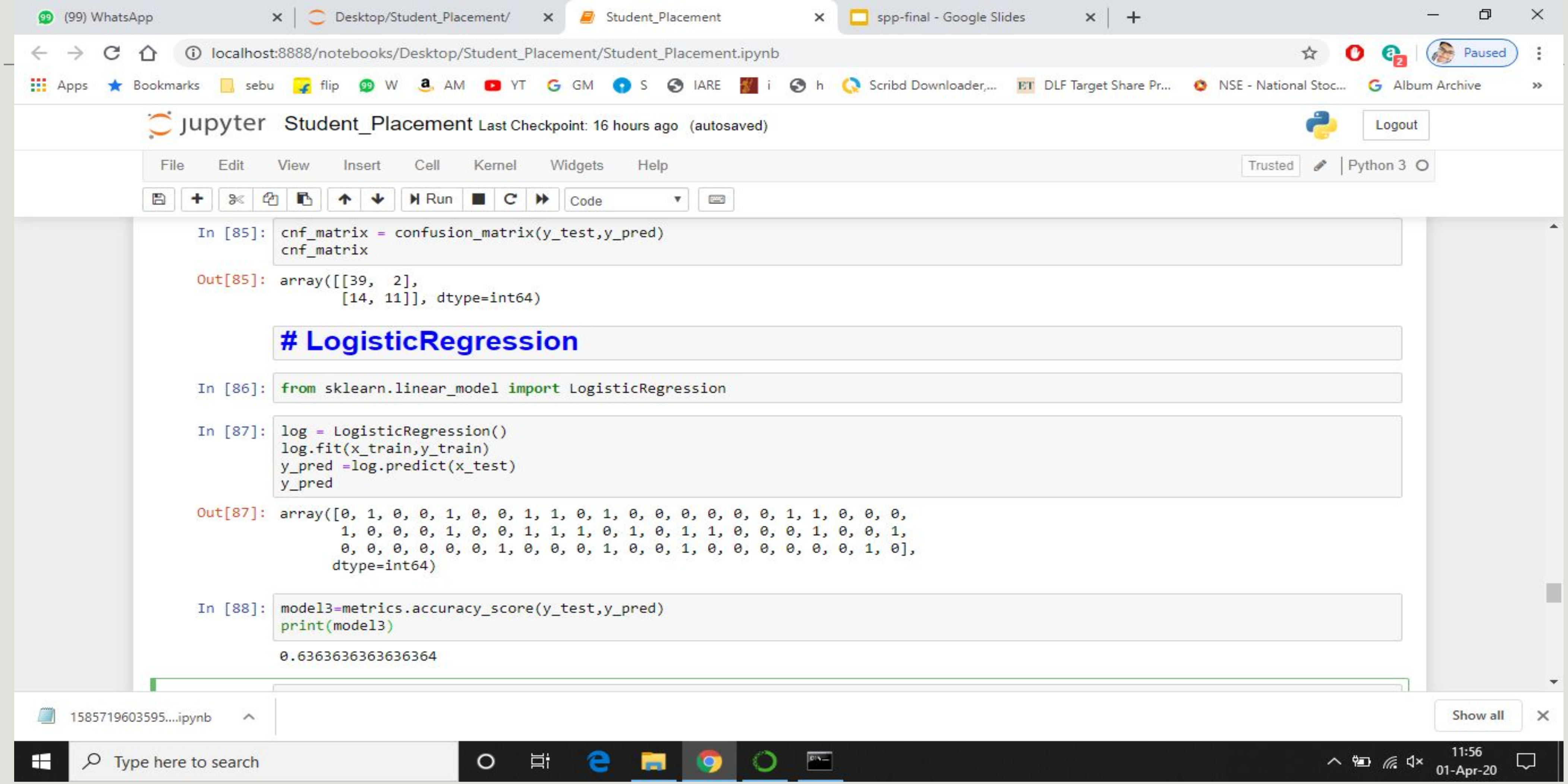
CODE



Random Forest Classifier

```
In [82]: from sklearn.ensemble import RandomForestClassifier

In [83]: rf = RandomForestClassifier(n_estimators=150)
         rf.fit(x_train,y_train)
         y_pred=rf.predict(x_test)
         y_pred

Out[83]: array([0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0,
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1,
                0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0],
               dtype=int64)

In [84]: model2=metrics.accuracy_score(y_test,y_pred)
         print(model2)

         0.7575757575757576

In [85]: cnf_matrix = confusion_matrix(y_test,y_pred)
         cnf_matrix

Out[85]: array([[39,  2],
                [14, 11]], dtype=int64)
```

CODE

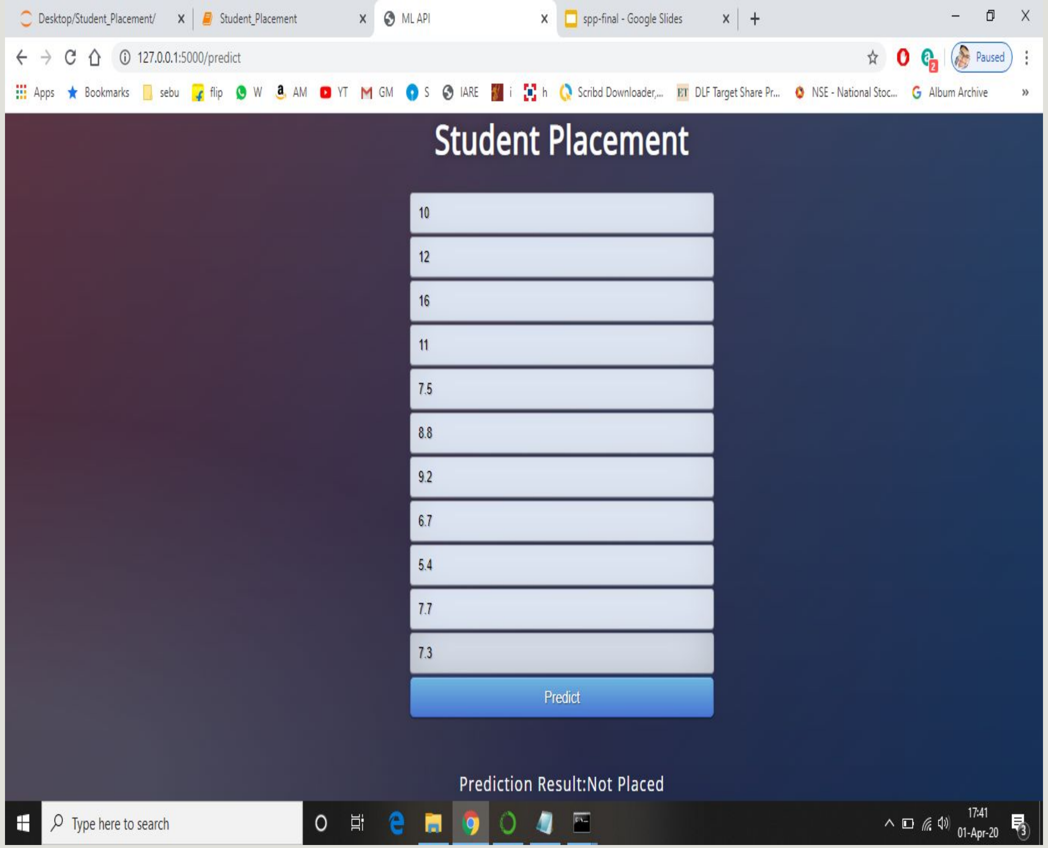


```
In [85]: cnf_matrix = confusion_matrix(y_test,y_pred)
         cnf_matrix

Out[85]: array([[39,  2],
                [14, 11]], dtype=int64)
```

LogisticRegression

```
In [86]: from sklearn.linear_model import LogisticRegression

In [87]: log = LogisticRegression()
         log.fit(x_train,y_train)
         y_pred =log.predict(x_test)
         y_pred

Out[87]: array([0, 1, 0, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0,
                1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1,
                0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0],
               dtype=int64)

In [88]: model3=metrics.accuracy_score(y_test,y_pred)
         print(model3)

         0.6363636363636364
```

OUTPUT



Student Placement

10

12

16

11

7.5

8.8

9.2

6.7

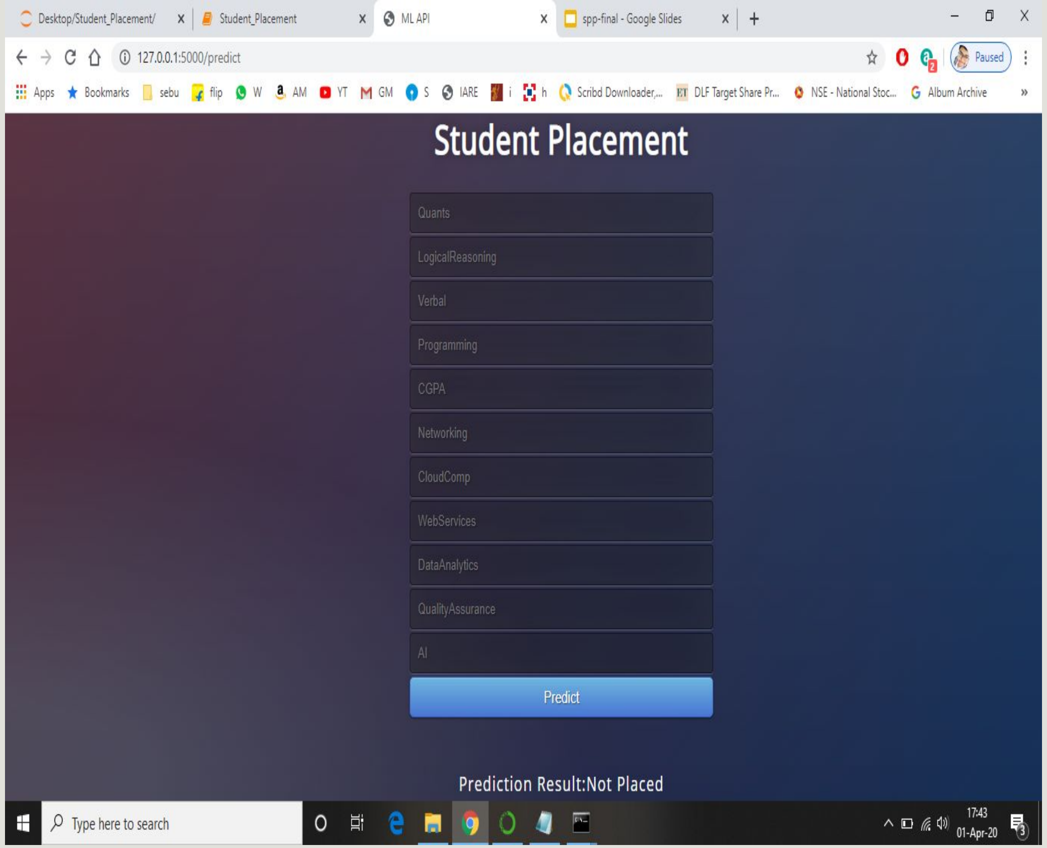
5.4

7.7

7.3

Predict

Prediction Result:Not Placed



Student Placement

Quants

LogicalReasoning

Verbal

Programming

CGPA

Networking

CloudComp

WebServices

DataAnalytics

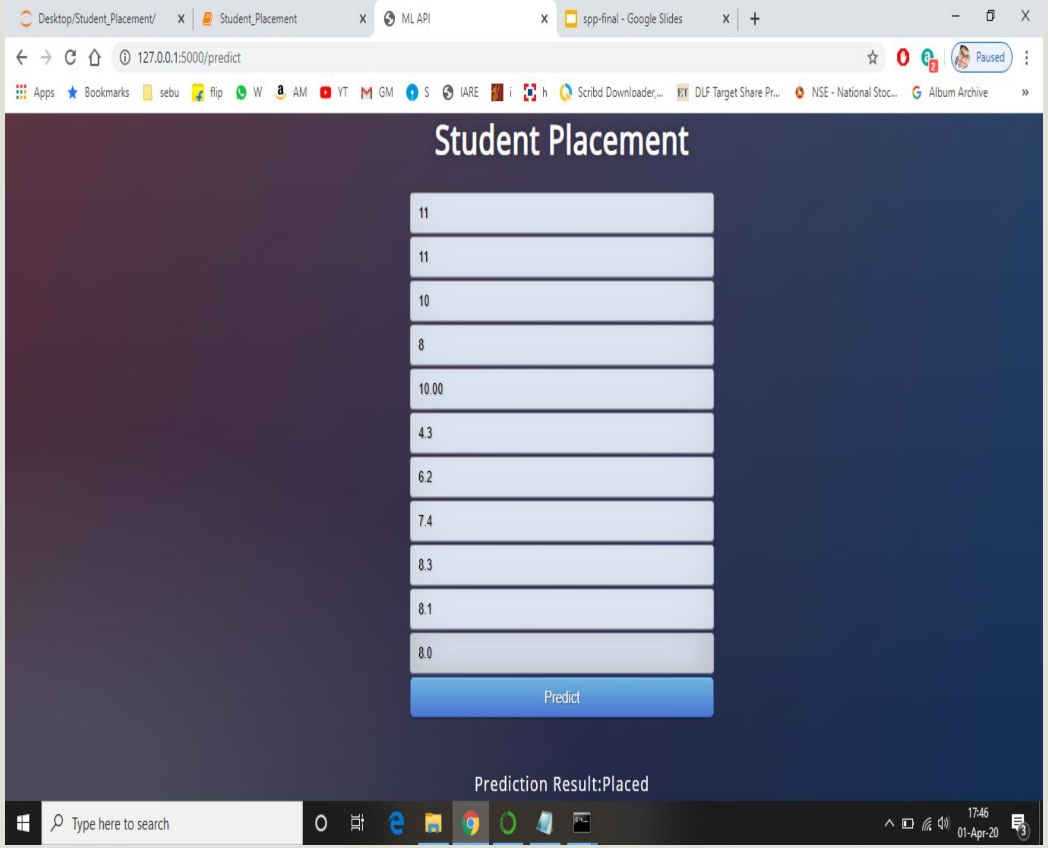
QualityAssurance

AI

Predict

Prediction Result:Not Placed

OUTPUT



Student Placement

11

11

10

8

10.00

4.3

6.2

7.4

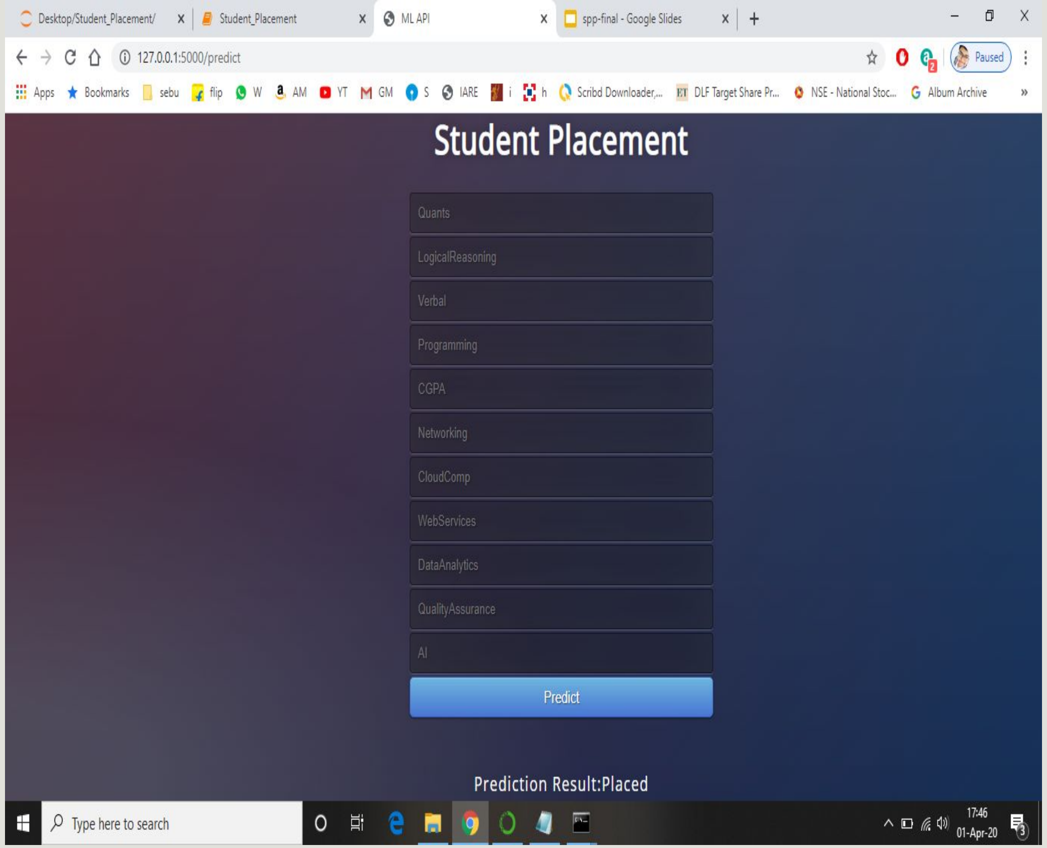
8.3

8.1

8.0

Predict

Prediction Result:Placed



Student Placement

Quants

LogicalReasoning

Verbal

Programming

CGPA

Networking

CloudComp

WebServices

DataAnalytics

QualityAssurance

AI

Predict

Prediction Result:Placed

Conclusion

Finally, placement analysis for students which has been a major problem is being countered.The work reported in this project indicates the importance of machine learning techniques with supervised learning algorithms intended to understand the performance of algorithm with respect to student records,where we analyse the performance of student and categorize it into two classes as placed or not placed with a good accuracy .

REFERENCES

- Mangasuli Sheetal B, Prof. Savita Bakare “Prediction of Campus Placement Using Data Mining Algorithm Fuzzy logic and K nearest neighbour” International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 6, June 2016 .
- Ravi Tiwari and Awadhesh Kumar Sharma, “A Data Mining Model to Improve Placement”, International Journal of Computer Applications (0975 – 8887) Volume 120 – No.12, June 2015
- Ms.sonal patil, Mr.Mayur Agrawal, Ms.Vijaya R. Baviskar “Efficient Processing of Decision Tree using ID3 & improved C4.5 Algorithm”, International Journal of Computer Science and Information Technologies, Vol. 6 (2) , 2015, 1956-1961