

# **COLLEGE MANAGEMENT SYSTEM TO GENERATE STUDENT'S OVERALL PERFORMANCE REPORT USING DECISION TREE ALGORITHM**

**Tribhuvan University  
Institute of Science and Technology**



A Final Year Project Report Submission in  
Partial Fulfillment of the Requirement for the Degree of  
Bachelor of Science in Computer Science and Information Technology

Under the Supervision of

Abhishek Dewan

Vice-Program Coordinator

Submitted by

Faizan Ansari (11092/73)

Nitesh Tuladhar (11097/73)

Ram Sharan Sapkota (11101/73)

Sushek Tamrakar (11119/73)

Submitted to

TRINITY INTERNATIONAL COLLEGE

Department of Computer Science and Information Technology

Dillibazar Height, Kathmandu, Nepal

December 2020

# **COLLEGE MANAGEMENT SYSTEM TO GENERATE STUDENT'S OVERALL PERFORMANCE REPORT USING DECISION TREE ALGORITHM**



A FINAL YEAR PROJECT REPORT SUBMITTED TO  
TRINITY INTERNATIONAL COLLEGE  
DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY  
DILLIBAZAR HEIGHT, KATHMANDU

By

Faizan Ansari (11092/73)

Nitesh Tuladhar (11097/73)

Ram Sharan Sapkota (11101/73)

Sushek Tamrakar (11119/73)

December 2020



## DECLARATION

Project entitled “**College Management System To Generate Student’s Overall Performance Report Using Decision Tree Algorithm**” which is being submitted to the Department of Computer Science and Information Technology, Dillibazar, Kathmandu, Nepal for the fulfillment of the seventh semester under the supervision of **Abhishek Dewan**. This project is original and has not been submitted earlier in part or full in this or any other form to any university or institute, here or elsewhere, for the award of any degree.

## RECOMMENDATION

This is to recommend that **Faizan Ansari, Nitesh Tuladhar, Ram Sharan Sapkota and Sushek Tamrakar** have carried out research entitled “**College Management System To Generate Student’s Overall Performance Report using Decision Tree Algorithm**” for the fulfillment of seventh semester under my/our supervision. To my/our knowledge, this work has not been submitted for any other degree.

They have fulfilled all the requirements laid down by the Trinity International College Department of Computer Science and Information Technology, Dillibazar, Kathmandu, Nepal.

-----  
Mr. Abhishek Dewan

Vice-Program Coordinator

Department of Computer Science and Information Technology,

Trinity International College

Dillibazar, Kathmandu, Nepal

## LETTER OF APPROVAL

[Date: 23/12/2020]

On the recommendation of Mr. Abhishek Dewan, this Project Report submitted by **Faizan Ansari, Nitesh Tuladhar, Ram Sharan Sapkota and Sushek Tamrakar**, entitled “**College Management System To Generate Student’s Overall Performance Report using Decision Tree Algorithm**” in partial fulfillment of the requirement for the award of the bachelor’s degree in Computer Science and Information Technology is a bonafide record of the work carried out under my/our guidance and supervision at Trinity International College, Kathmandu.

## EVALUATION COMMITTEE

.....

.....

External

Mr Satya Bahadur Maharjan

Program Coordinator,

Department of Computer Science and Information Technology

Dillibazar Height, Kathmandu, Nepal

## ACKNOWLEDGEMENTS

We would like to express our deepest appreciation to all those who provided us the possibility to complete this report. A special gratitude we give to our supervisor and Assistant Program Coordinator, **Mr. Abhishek Dewan** for complete support and guidance throughout the project.

Also, we would like to express our gratitude to our Program Coordinator, **Mr. Satya Bahadur Maharjan** , our administrative assistant sir **Mr. Shoyam Bhattarai** our IT assistant sir **Mr. Jitesh Tuladhar** whose encouragement helped us to coordinate our project in a systematic manner.

We would also like to express out special gratitude to all the team members, friends and teachers who helped us reach out the possibilities of our project

Faizan Ansari

Nitesh Tuladhar

Ram Sharan Sapkota

Sushek Tamrakar

December 2020

## **ABSTRACT**

“College Management System To Generate Student’s Overall Performance Report Using Decision Tree Algorithm” is a web application that provides an interface for proper arrangements of student information. The purpose of this application is to maintain the records of students as efficient as possible to generate the overall academic performance report using Decision Tree Algorithm. The creation and management of accurate, up-to-date information regarding a students’ academic career is critically important in the university as well as colleges. College management system handles almost all kind of student information, teacher’s information, academic details and generates a proper report also maintaining college details, course details, curriculum, batch details, placement details and other resource related details. It will also have faculty details, batch execution details, students’ details in all aspects, the various academic notifications to the staff and students updated by the college administration. The web application is based on Python Django Framework which is backend and JavaScript which is front-end.

**Keyword:** *Python, Django, College Management System, Decision Tree Algorithm*



# TABLE OF CONTENTS

DECLARATION .....	i
RECOMMENDATION .....	ii
ACKNOWLEDGEMENTS .....	iv
ABSTRACT.....	v
LIST OF ACRONYMS AND ABBREVIATIONS .....	viii
LIST OF FIGURES .....	ix
LIST OF TABLES.....	x
CHAPTER 1 .....	1
INTRODUCTION .....	1
1.1. Introduction.....	1
1.2. Problem Definition.....	1
1.3. Objectives .....	2
1.4. Scope.....	2
1.5. Limitation.....	3
CHAPTER 2 .....	4
RESEARCH METHODOLOGY.....	4
2.1. Literature Review.....	4
2.2. Framework .....	6
2.3. Algorithm.....	7
2.4. Tools .....	8
CHAPTER 3 .....	9
SYSTEM DEVELOPMENT .....	9
3.1. System requirement .....	9
3.1.1. Functional Requirements .....	9
3.1.2. Non-Functional Requirements .....	10
3.2 Feasibility study .....	10
3.2.1. Economic feasibility .....	10
3.2.2. Technical feasibility .....	10
3.2.4. Legal feasibility .....	10

3.2.5. Operational Feasibility .....	11
3.2.6. Schedule Feasibility.....	11
3.3 System Designs.....	12
3.3.1. Class diagram .....	12
3.3.2. Sequence diagram.....	13
3.3.3. Activity diagram .....	14
3.3.4. Wireframe.....	15
3.3.5. Mind map.....	16
CHAPTER 4 .....	17
IMPLEMENTATION AND TESTING .....	17
4.1 Implementation .....	17
4.1.1 Hardware Implementation .....	17
4.1.2 Software Implementation .....	17
4.2 Testing.....	17
4.2.1. System Testing .....	18
4.2.2. Unit Testing .....	20
4.2.3. Integration Testing.....	20
CHAPTER 5 .....	21
RESULT ANALYSIS.....	21
5.1 Result and discussion .....	21
CHAPTER 6 .....	23
CONCLUSION AND RECOMMENDATIONS .....	23
6.1. Conclusion .....	23
6.2. Recommendation .....	23
REFERENCES .....	24
APPENDIX.....	25
APPENDIX 1 .....	25
APPENDIX 2.....	27

## **LIST OF ACRONYMS AND ABBREVIATIONS**

CSS	Cascading Style Sheet
HTML	Hypertext Markup Language
JS	JavaScript
MIS	Management Information System
PGSQL	PostgreSQL

## LIST OF FIGURES

Figure 1. Framework	6
Figure 2. Use Case Diagram	10
Figure 3. Gantt chart	12
Figure 4. Class diagram	13
Figure 5. Sequence diagram	14
Figure 6. Activity diagram	15
Figure 7: Proposed Home page design of the system	16
Figure 8: Proposed Login page of the system	16
Figure 9: Mind map	17
Figure 10: Decision Tree	22

## **LIST OF TABLES**

Table 1. Login Test	18
Table 2. Student Academic Performance Test Case	19
Table 3. Add New User	20

# **CHAPTER 1**

## **INTRODUCTION**

### **1.1. Introduction**

MIS is a system that helps an organization to manage the overall activities effectively by providing the necessary information. It is a system that studies the individual, groups or organization based on how they evaluate, implement, manage and utilize the systems in order to generate an information to improve efficiency and effectiveness of decision making. Management Information System also helps in analyzing the data from previous data records and make future predictions on both client and employee. This helps to improve the overall organization performance both internally and externally also improvising the sustainability of organization.

The title of the project is College Management System to Generate Student's Overall Performance Report. It is also defined as a web application based on Intranet or Internet that aims to all the levels of management providing information within an academic organization. This system can be used as an information management system as well as to generate a report based on overall academic performance of the student.

The Administrator creates the user account, student or staff can access the system to either see the information or to retrieve the data of particular students and overall record of the students on the basis of privileged provided to the particular staff. This project provides an objective system for recording and aggregating information and supports the institution's strategic goals and direction. Since this system supplies the decision-makers with facts, it supports and enhances the overall decision-making process of the academic organization. Here the knowledge of Python with PostgreSQL database is desirable to execute this project.

### **1.2. Problem Definition**

Existing system such as paper-based academic activities that are used in colleges are traditional, outdated and is a complete linear process. Though it is used by most of the organizations, it is still one of the inefficient, time consuming and less effective process. The

problems that an academic organization that uses a traditional paper-based process might face are:

- When the number of records increases it is difficult to maintain the information of each student using the old manual paper-based process.
- Maintaining the records manually leads to error prone and requires more manpower and it consumes more time for processing the records.
- Analyzing the manually written records might lead to less efficient decision making.
- Paper based records are less secure and more prone to physical damage

### **1.3. Objectives**

The main objective of developing the current project entitled College Management System to Generate Student's Overall Performance Report is to build an effective system which is fast, accurate, consistent, reliable, efficient and flexible enough so that it can incorporate any future enhancement. Some of the objectives are as follows:

- To effectively utilize more time and manpower, make online information easily available to the user and at the same time, maintain high level of security.
- To enable every user such as students, staffs and administration to see the information of each student through online from anywhere and anytime.
- To make efficient decisions of student overall academic remark

### **1.4. Scope**

This project deals with the various functioning in College Management System to Generate Student's Overall Performance Report using Decision Tree Algorithm. The main idea is to implement a proper process to system. In our existing system contains many operations such as student search, attendance, exam records, performance of the student, create assignment etc. All these activity takeout manually by administrator or staff as well. So, some of the scope from our project is listed and described.

### **Easier Management of Programs and Course**

Colleges run multiple programs and courses to have a high number of enrollments. To keep manual records of all these courses is a complex task. A college management system can create and manage the different programs that run by your institute.

### **Easier Manage employee and student record**

College management system can manage the complete record of students and other management employees. Dealing with records and give access to students, teachers and other employees to upload information any time they want.

### **Easily Track the Student Performance**

With the help of this project, we can track the student academic overall performance easily whether they are progressing or downgrading.

## **1.5. Limitation**

- This system is built for only one specific educational institute.
- When there is a huge flow in traffic the application is prone to performance issues.
- Only, people who are accustomed to regular use of smartphones or computers can operate this software.



## CHAPTER 2

### RESEARCH METHODOLOGY

#### 2.1. Literature Review

A background study is done to review similar existing systems used to perform student performance analysis. Three existing system are chosen because these systems are similar to the proposed system. Faculty Support System is able to analyze the students' data dynamically as it is able to update of students' data dynamically with the flow of time. Student Performance Analyzer is a powerful analyzer tool used by schools worldwide to perform analysis and displays the analysis data once raw student data is uploaded to the system. The analysis is done by tracking the student or class to get the overall performance of student or class. It helps to identify the students' performance which is below the expected level, at expected level or above the expected level. This would allow the educators or staffs to identify the current students' performance easily. Inminds system enables top and mid-management in UNIMAS to have a clear look on the areas that needed attention by looking at the figures, revenues and risks. The features, ease of use and flexibility provided by the system makes the performance analysis in UNIMAS to be performed in an ideal solution. There are a few features from the existing systems that are employed during the design and implementation phase of the proposed system. These features and functionalities include the user interface, students' performance prediction, illustration displays and report generation. A good user interface provides user-friendly interface as it is easy to be navigate and not complicated.

Meanwhile, the students' performance prediction is included into the proposed system to make sure the objectives are achieved. Furthermore, the generation of reports in Portable Document Format. From these features found in proposed system, all the user requirements would be fulfilled. [1].

College Staff are able to directly access all aspects of a student's academic progress through a secure, online interface embedded in the college's website. The system utilizes user authentication, displaying only information necessary for an individual's duties. Additionally, each sub-system has authentication allowing authorized users to create or update information in that subsystem. All data is thoroughly reviewed and validated on the server before actual record alteration occurs. In addition to a staff user interface, the system plans for student user

interface, allowing users to access information and submit requests online thus reducing processing time. All data is stored securely on servers managed by the college administrator and ensures highest possible level of security. The system features a complex logging system to track all users access and ensure conformity to data access guidelines and is expected to increase the efficiency of the college's record management thereby decreasing the work hours needed to access and deliver student records to users [2].

This is user friendly and easy to use system. All the important data are stored in the database and it avoids any miscalculation. It replaces the old and traditional file (paperwork) storing process; it even provides editing facilities for data which would be tedious through paperwork. It eliminates manual work (Manual system does not mean that we are working with pen and paper, it also includes working on spreadsheets and other simple software's). With this we can retrieve student information instantly. The major benefit of this web portal is to store the student information in one place and can be accessed easily. This is a software application which is responsive in nature [3].

## 2.2. Framework

In this framework model, we describe how the system will operate as:

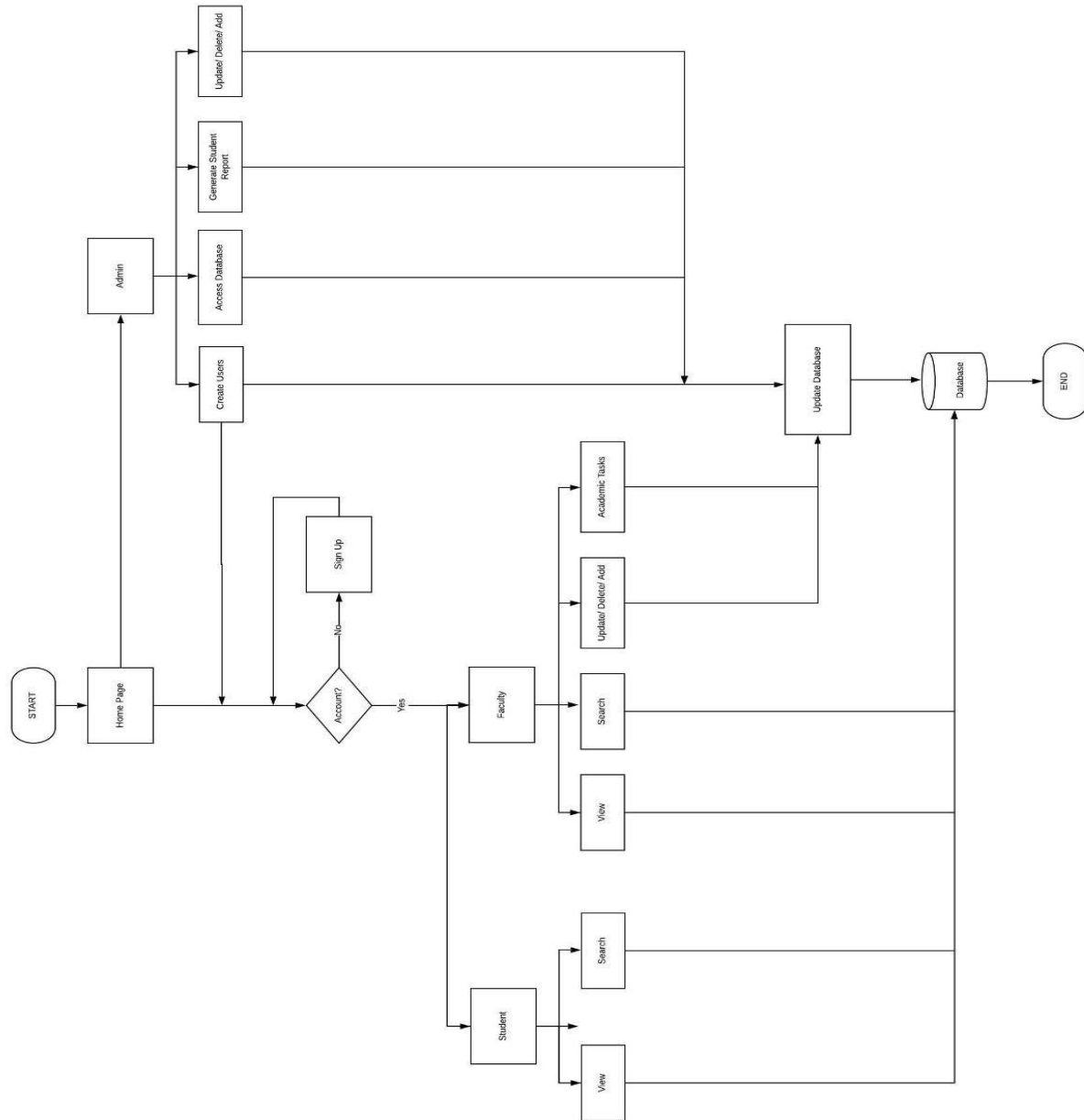


Figure 1. Framework(Flow Chart)

## 2.3. Algorithm

A decision tree is a flow-chart tree structure, where each internal node is denoted by rectangles, and leaf nodes are denoted by ovals. All internal nodes have two or more children node and the internal nodes contain splits, which test the value of an expression of the attributes.

### C4.5

C4.5 known as J48 in WEKA is a successor of ID3 developed by Quinlan (1992). It is also based on Hunt's algorithm. J48 handles both categorical and continuous attributes to build a decision tree.

$$Gain(S, A) = Entropy(S) - \sum_{v \in value(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (1)$$

Where Values (A) is the set of all possible values for attribute A, and  $S_v$  is the subset of S for which attribute A has value v (i.e.,  $S_v = \{s \in S \mid A(s) = v\}$ ). The first term in the equation for Gain is just the entropy of the original collection S and the second term is the expected value of the entropy after S is partitioned using attribute A. The expected entropy described by this second term is simply the sum of the entropies of each subset, weighted by the fraction of examples That belong to Gain (S, A) is therefore the expected reduction in entropy caused by knowing the value of attribute A.

$$Gain\ Ratio(S, A) = \frac{Gain(S, A)}{Split\ Information(S, A)} \quad (2)$$

### Classification and Regression Tree (CART).

CART was introduced by Breiman et al. (1984). It is also based on Hunt's algorithm. CART handles both categorical and continuous attributes to build a decision tree. It handles missing values. CART uses Gini Index as an attribute selection measure to build a decision tree. Unlike

ID3 and J48 (C4.5) algorithms, CART produces binary splits. Hence, it produces binary trees. Gini Index measure does not use probabilistic assumptions like ID3, C4.5. CART uses cost complexity pruning to remove the unreliable branches from the decision tree to improve the accuracy.

$$\text{Gini Index: } 1 - \sum_j p_j^2$$

Gini index of a pure table which consist of single class is zero because the probability is 1 and  $1 - 1^2 = 0$ . Similar to Entropy, Gini index also reaches maximum value when all classes in the table have equal probability [4].

## 2.4. Tools

The Tools used in the project are listed below.

### **FRONT-END:**

1. HTML
2. CSS
3. JavaScript

### **BACK-END:**

1. Python
2. Django
3. PostgreSQL

## CHAPTER 3

### SYSTEM DEVELOPMENT

#### 3.1. System requirement

This requirement specification provides a detailed feature of requirement needed by programmer or developing a project.

##### 3.1.1. Functional Requirements

- System should describe the behavior under specific conditions
- The system should be clear for both development team and stakeholders
- System must implement to enable users to accomplish certain tasks

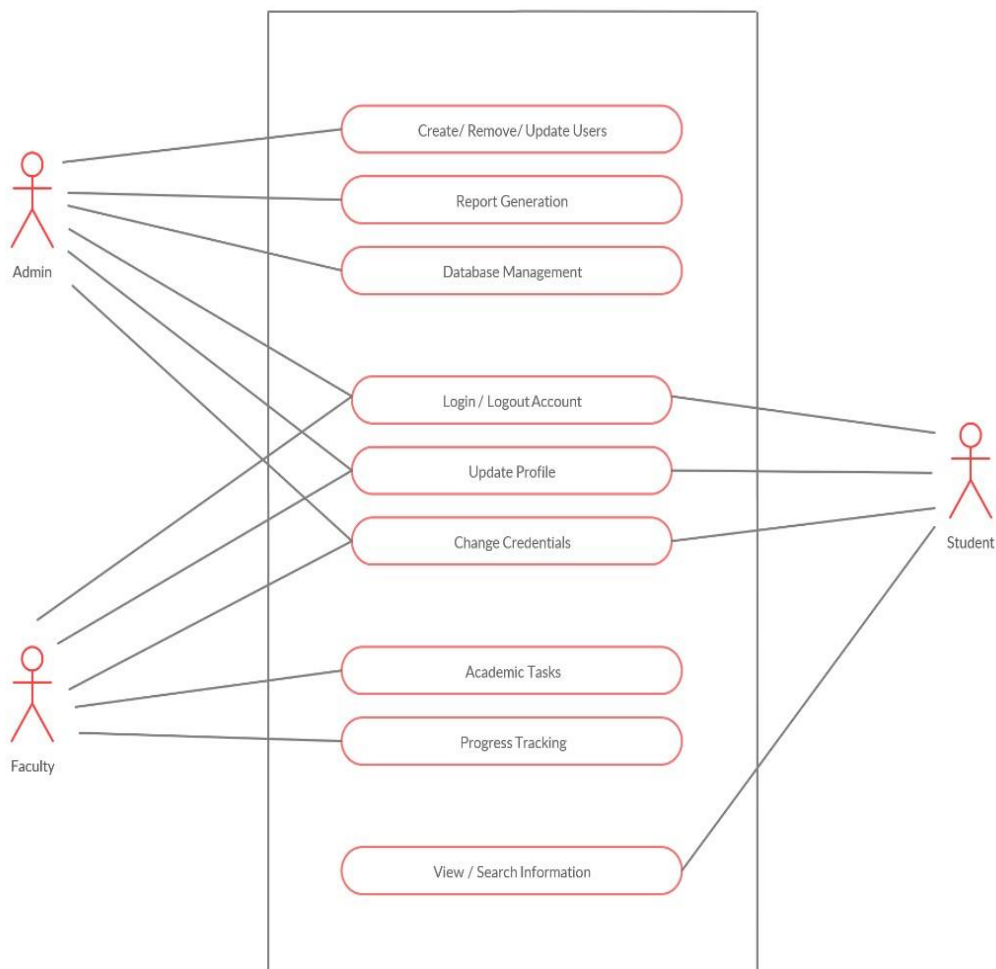


Figure 2. Use Case Diagram

### **3.1.2. Non-Functional Requirements**

- System should functionally perform all the requirements that are specified.
- The database should take the database backup for the safety requirements.
- Authorities must be specified depending on the roles of the user.
- System should be user-friendly and easy to operate.
- The system should be cross browser compatible.

## **3.2 Feasibility study**

It is both necessary and prudent to evaluate the feasibility of a project at earliest possible time. Wastage of manpower and financial resources and untold professional embarrassment can be avoided if an ill-conceived system is recognized early in the development phase. To know whether the project is feasible or not, several checks such as economical, technological, operational, etc. was carried out.

### **3.2.1. Economic feasibility**

- As of our Project, the web application doesn't require any software cost however, the user must have PC/laptop with internet connection
- The cost and benefits associated with the proposed system are economically feasible as all the resources required are easily accessible and development tools are available for free.

Thus, the project was economically feasible.

### **3.2.2. Technical feasibility**

- Every educational institute have technical resources to access the internet by the help of browser.
- Also, every educational institute has technical personnel who can operate the web application and servers.

### **3.2.4. Legal feasibility**

- The implementation of this project doesn't violate any rules or standards define by the Government of Nepal (GON).

- The Concepts and Techniques taken from other sources are properly cited. It also doesn't violate any copyright act.

### 3.2.5. Operational Feasibility

- Any project that can be implemented in real-world is beneficial. The website can be surf using the internet from the PC or smartphone. It only requires a web browser to be installed on user's computer
- As the project can be implemented in real world for beneficial purpose on behalf of the existing system making it totally computer-based system

### 3.2.6. Schedule Feasibility

A Gantt chart represents project schedule with tasks on the vertical axis and time intervals on the horizontal axis. The following chart is created on the basis of obtained information and data using Free Gantt chart excel:

ID	Task Name	Start	Finish	Duration	Mar 2020					Apr 2020				May 2020			
					3/1	3/8	3/15	3/22	3/29	4/5	4/12	4/19	4/26	5/3	5/10	5/17	
1	Planning	03/03/2020	09/03/2020	7d													
2	Designing layout	10/03/2020	16/03/2020	7d													
3	Design	16/03/2020	10/04/2020	26d													
4	Writing the code	10/04/2020	10/05/2020	31d													
5	Debugging	10/05/2020	15/05/2020	6d													
6	Testing and Verification	15/05/2020	21/05/2020	7d													
7	Finishing	21/05/2020	28/05/2020	8d													

Figure 3: Gantt chart



## 3.3 System Designs

### 3.3.1. Class diagram

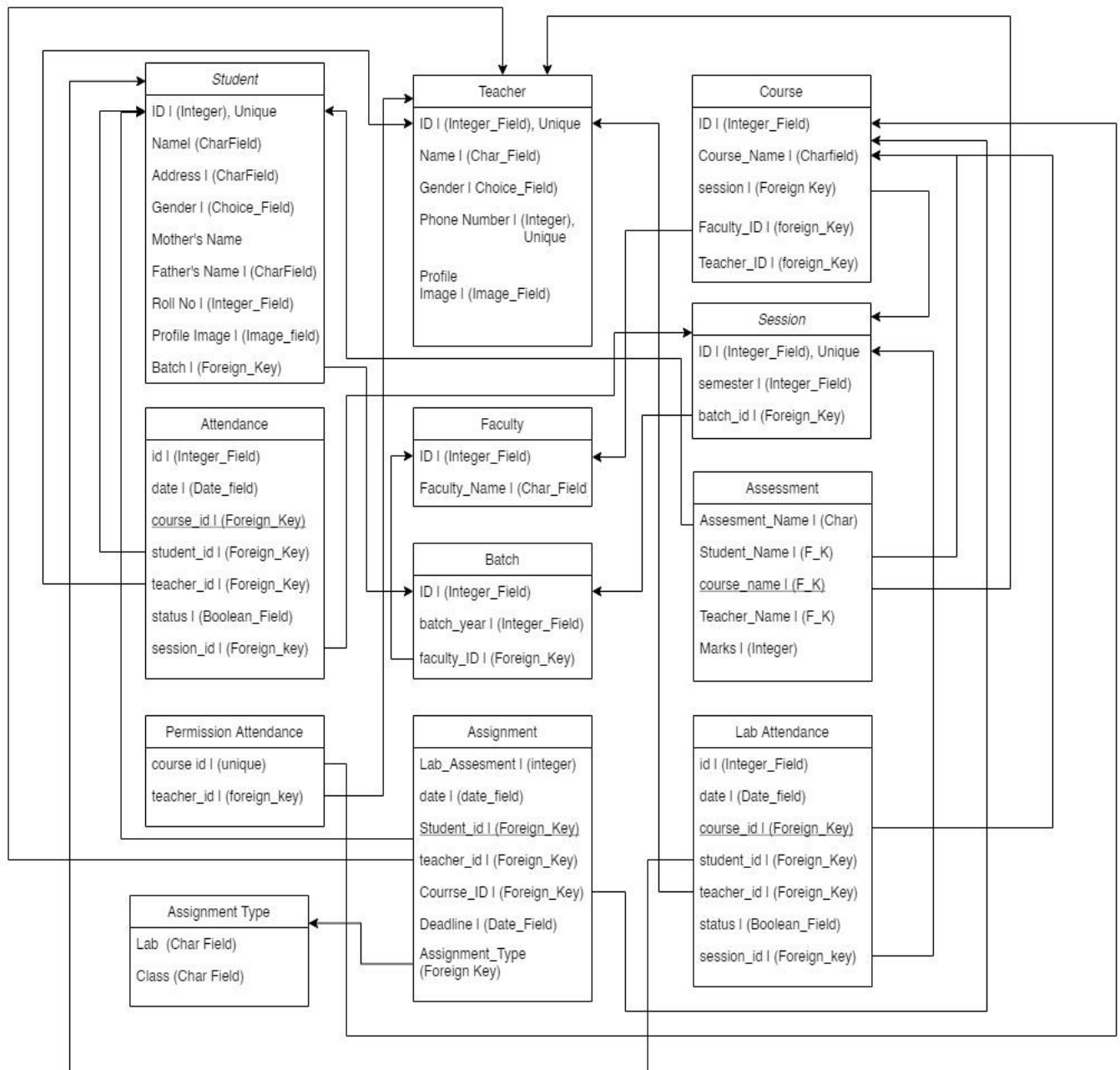


Figure 4: Class Diagram

### 3.3.2. Sequence diagram

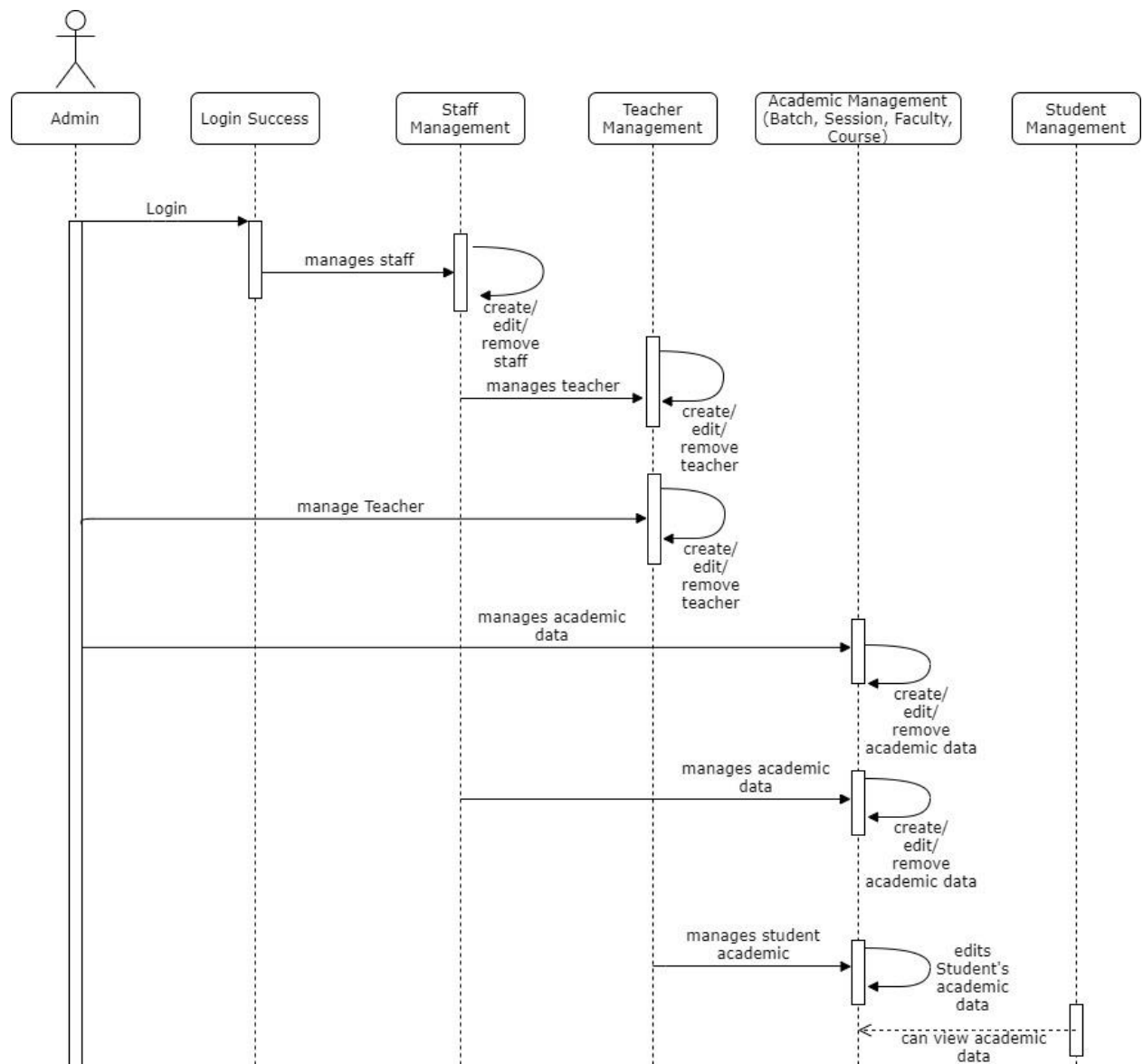


Figure 5: Sequence Diagram

### 3.3.3. Activity diagram

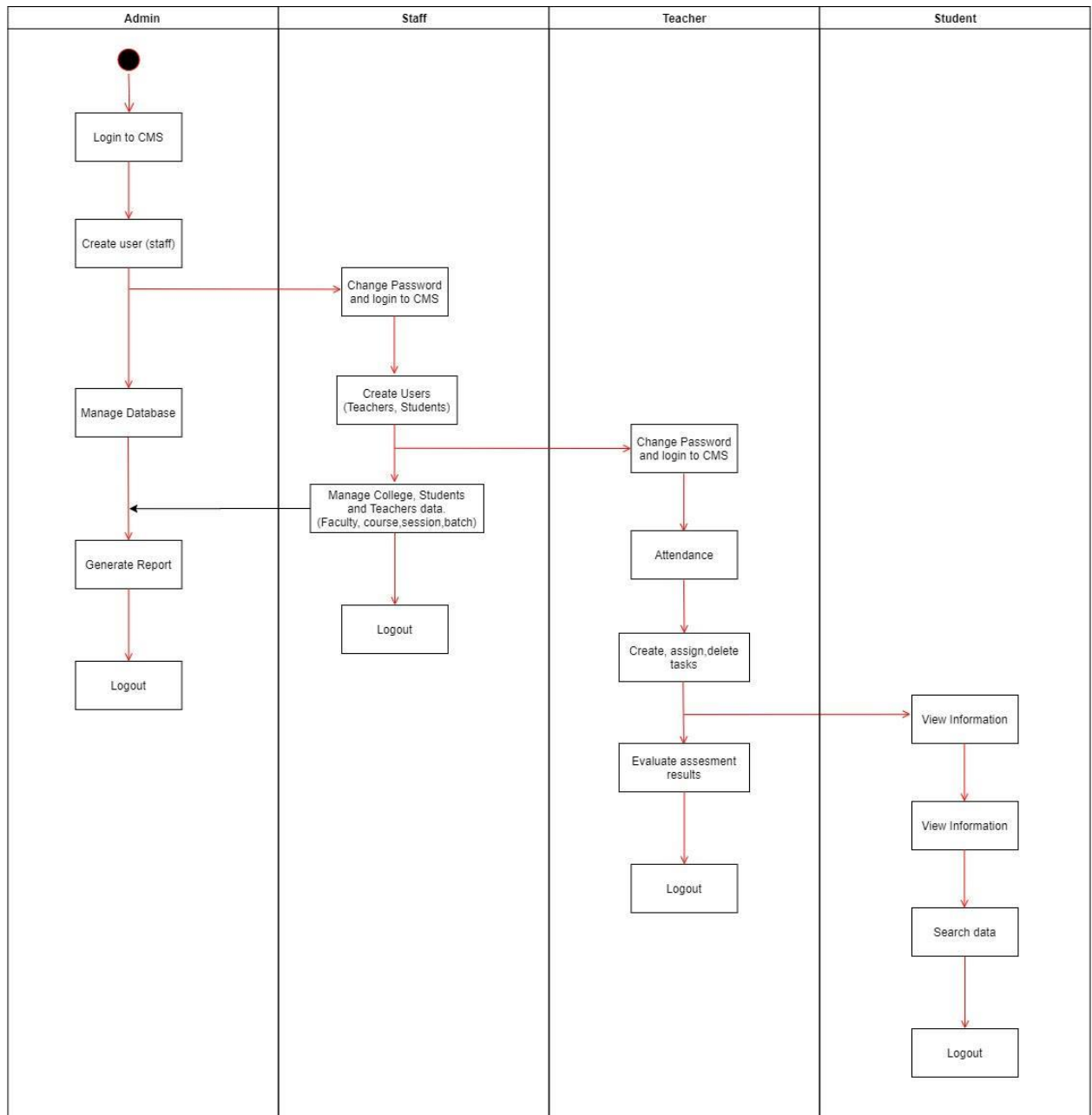
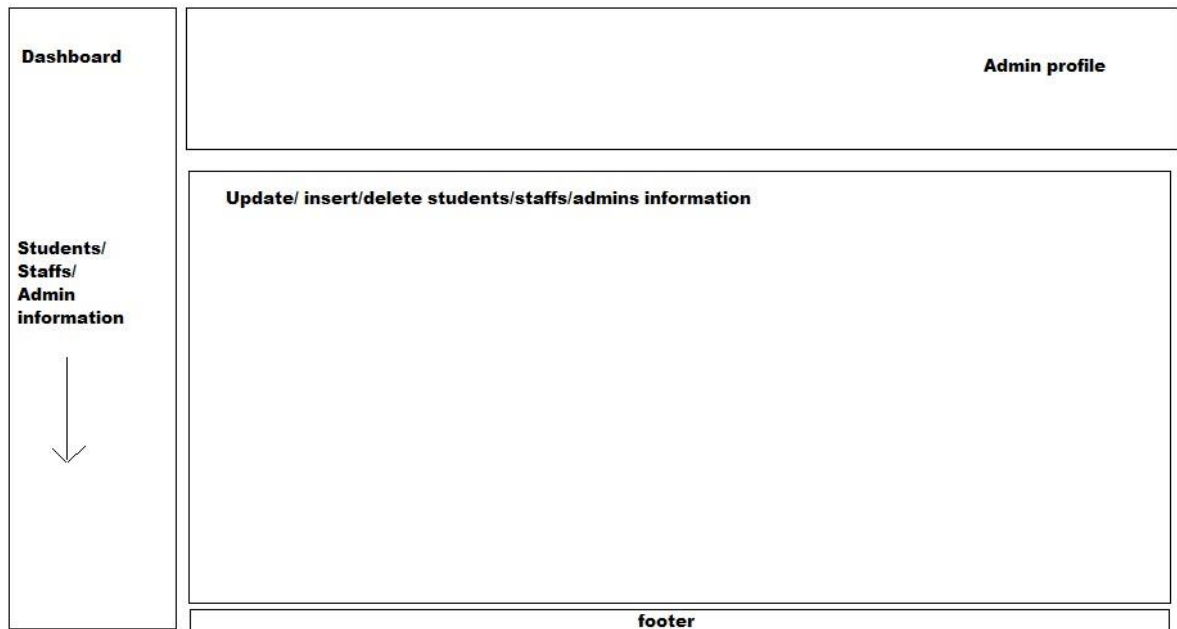


Figure 6: Activity Diagram

### 3.3.4. Wireframe



The wireframe shows a login form with the following components: a label **Username** above a text input field; a label **Password** above another text input field; a link *forgot password?* positioned below the password field; and a **Login** button located at the bottom of the form.

Figure 7-8: WireFrame

### 3.3.5. Mind map

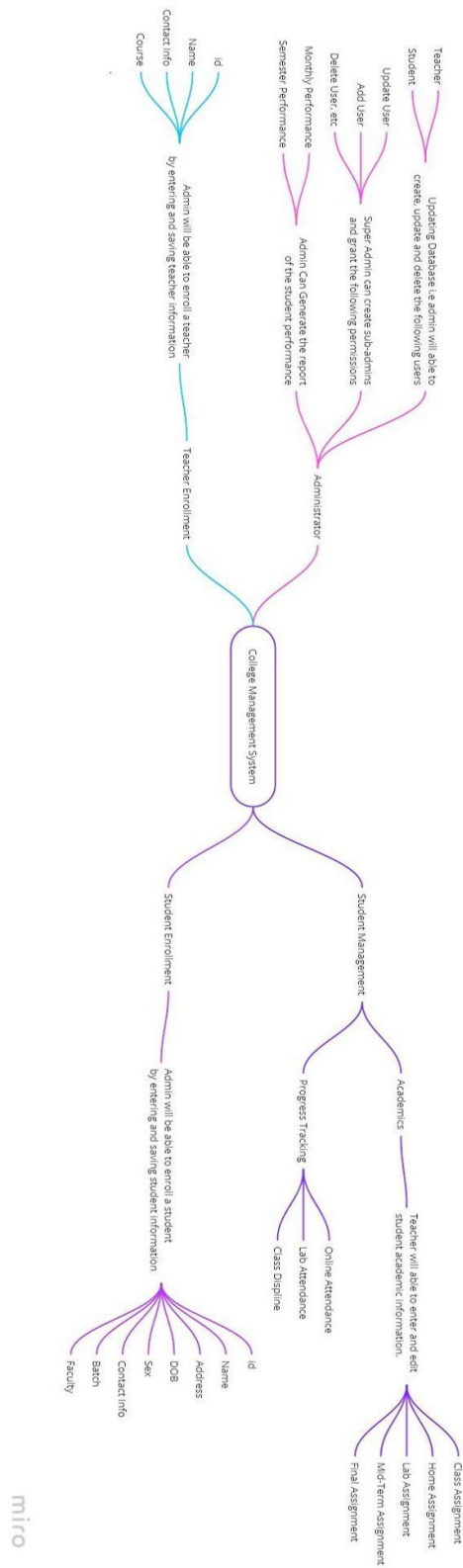


Figure 9: Mind Map

## **CHAPTER 4**

### **IMPLEMENTATION AND TESTING**

#### **4.1 Implementation**

##### **4.1.1 Hardware Implementation**

For hardware implementation, developers need a proper machine to code their website. The machine should be able to provide everything a developer needs to create quality products. They should be up to date and easy to use.

The hardware devices used for the project are:

- Processor
- Hard Disk
- Ram Memory

##### **4.1.2 Software Implementation**

For software implementation, developers need a fast and efficient platform that they can use to code their website.

The software used to code the website are:

- Sublime Text 3
- Virtual Studio Code

For the hardware implementation, developers need a proper machine to code their webs

#### **4.2 Testing**

The testing process involves taking a sample data of the students from a particular class in a college. The sampled data is placed in the database and is processed accordingly using specific tools, resources and personnel, and within the specified cost and scheduled constraint to generate the student analysis report.

#### 4.2.1. System Testing

System testing is a level of the software testing where, completed and integrated software is tested. The purpose of this test is to evaluate the systems compliance with the specified requirements. In other word, the process of testing an integrated system to verify that it meets specified. After combining the module system testing has been done to test whether the system has met the expectation, if expectation is far away, rework has been done to meet the expectation.

#### Login Test

S. No	Test Cases	Expected Result	Actual Result
1.	Login with empty/invalid username and password	Display Error in Login	As Expected
2.	If Login User is Student	Redirect to the Student Dashboard	As Expected
3.	If Login User is Teacher	Redirect to the Teacher Dashboard	As Expected
4	If Login User is Admin	Redirect to the Admin Dashboard	As Expected

Table 1: Login Test

### Student Academic Performance Test Case

S. No	Test Cases	Expected Result	Actual Result
1.	Sending empty data in the algorithm parameters.	Display N/A	As Expected
2.	If Student Overall Academic Performance in Assignment, Lab-Assignment, Attendance, Lab-Attendance, Class Test, Mid-Assessment and Final Assessment is good	Good	As Expected
3.	If Student Overall Academic Performance in Assignment, Lab-Assignment, Attendance, Lab-Attendance, Class Test, Mid-Assessment and Final Assessment is Average	Average	As Expected
4	If Student Overall Academic Performance in Assignment, Lab-Assignment, Attendance, Lab-Attendance, Class Test, Mid-Assessment and Final Assessment is Bad	Bad	As Expected

Table 2: Student Academic Performance Test Case



#### 4.2.2. Unit Testing

Unit testing is a best practice that ensures your code is working how it was intended. It is a critical habit that should be developed early on in your coding adventures. Unit testing is done to check whether the particular module work as expectation or not, rework has been done for the module that doesn't meet the expectation.

#### Add New User

Objective Add New User	
Input	<ul style="list-style-type: none"><li>○ Email</li><li>○ Date Of Birth</li><li>○ Full Name</li><li>○ Address</li><li>○ Contact Number</li><li>○ Gender</li><li>○ Is the User Faculty Member?</li><li>○ Is the User Student?</li><li>○ Is the User Admin</li></ul>
Expected Output	Password is Auto Generated and sent it to the respective user's mail.
Error Info	<ul style="list-style-type: none"><li>○ Fill the fields</li><li>○ This username already exists, try different</li><li>○ Invalid email or password</li></ul>

Table 3: Add New User

#### 4.2.3. Integration Testing

For the Integration Testing, the testing is done based on whether the user is admin, faculty or as student. If the user is admin, he/she is able to perform admin authorized activities such as creating/updating/deleting user, adding events, payments, notice, etc. Likewise if the user is Teacher, he/she is able to perform staff activities such as taking attendance, creating assignment and give marks in assessment. Lastly, if the user is Student, they are able to view all their data including their performance.

## **CHAPTER 5**

### **RESULT ANALYSIS**

#### **5.1 Result and discussion**

As a result of this project, we can replace all the academic tasks that are based on paper and to generate the overall performance analysis report of the student that can be monitored and controlled remotely. Also, it allows students and teachers to view and access the data respectively from anywhere.

Here is the Decision Tree for the student performance analysis. The figure below represents the decision flow of the algorithm that determines the overall performance of the student for a particular semester. Based on seven parameters, the algorithm determines whether the student's performance is good, average or bad.

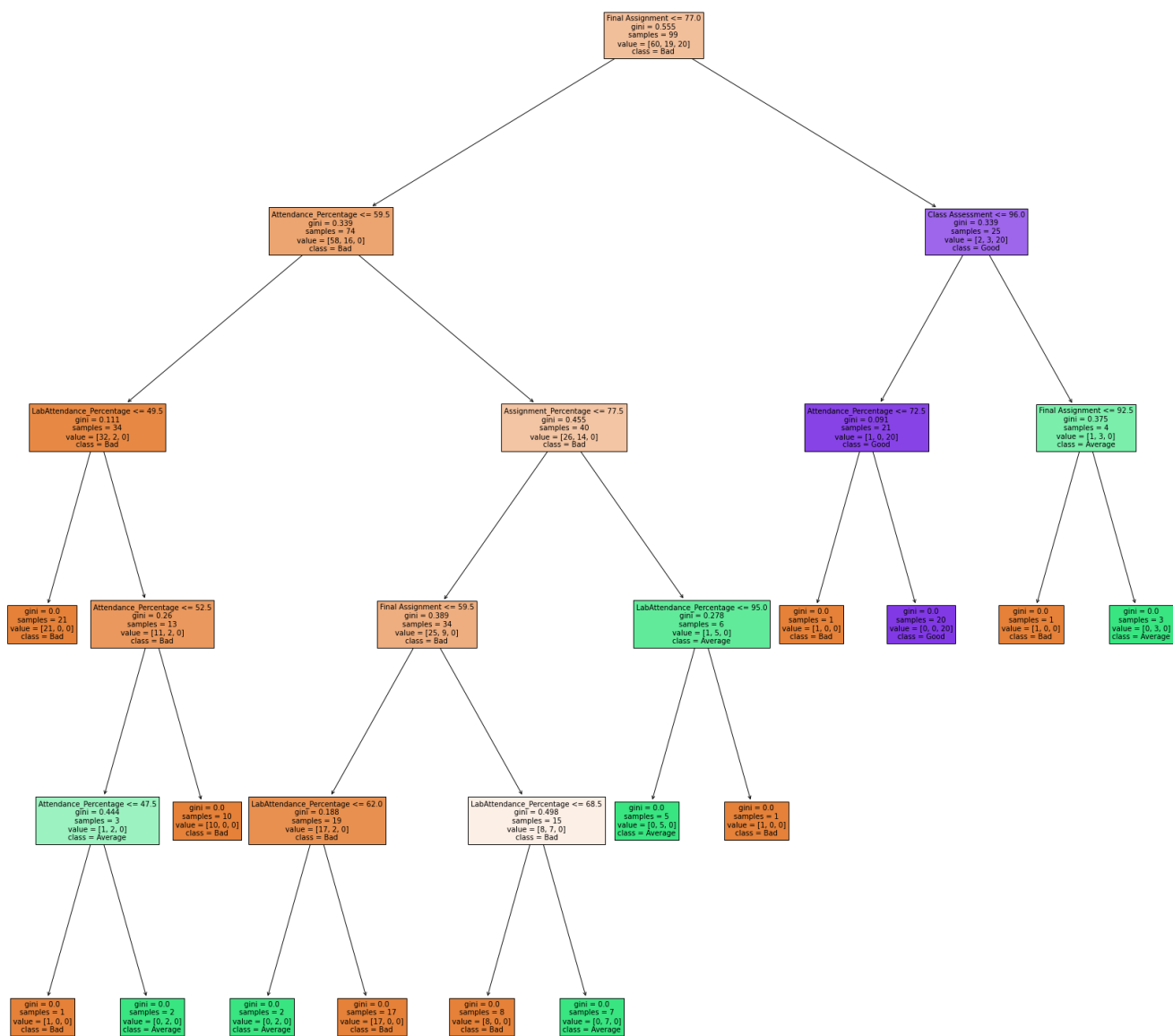


Figure 10: Decision Tree

## **CHAPTER 6**

### **CONCLUSION AND RECOMMENDATIONS**

#### **6.1. Conclusion**

This Project Titled "COLLEGE MANAGEMENT SYSTEM TO GENERATE STUDENT'S OVERALL PERFORMANCE REPORT USING DECISION TREE ALGORITHM" is designed mainly for the purpose of making student's academic performance report and also managing academic data of the college. This system will help us in managing the activities like attendance, class assessment, generating performance report, etc. Using this system, we can also view or update data and information about students and staff easily. We can also manage all the information of all the aspects of a college, its student, faculties, Department, marks and other extracurricular activities. Using this System, Teachers also can check student attendance anytime. It also helps teacher to announce the result. College administration can also manage college work easily. This system helps us to manage all the information of all the aspects of a college, its student faculties, marks, etc. College Management system provides the easiest way to manage all the functionalities of a college. By the system you can register new student and their course details. We can also check our result online by logging to the system. And more interestingly new students in the system can be added and details of the students can be checked.

#### **6.2. Recommendation**

Our project is depended on number of users on our database where we have used SQLite which is default database of the Python Framework Django. Also, teacher cannot be assigned to the multiple courses in same batch for now. These issues can be overcome and the system will be further improved in the coming days.

- PostgreSQL is the recommended database in the python Django framework and will be replaced once the project is deployed and fully functional.
- The Database Architecture can be changed to overcome the matter of resolving one to many fields in database of teacher.

## REFERENCES

- [1] B. R. S, G. R. B and T. G. S, *Web Based Student Information Management*, vol. II, no. 6, pp. 2042-2348, 2013.
- [2] C. B. S, *An Innovative Information System for College Management*, vol. III, no. 6, pp. 140145, 2018.
- [3] C. L. Sa, D. H. b. A. Ibrahim, E. D. Hossain and M. b. Hossin, "Proposed System," *Student Performance Analysis System (SPAS)*.
- [4] S. Patnaik, K. k. Singh, . R. Ranjan and N. Kumari, "JavaScript," *College Mangement System*, vol. III, no. 5, pp. 659-661, 2016.
- [5] S. Y. Kayode, H. M. Abiola, S.-I. T. Tosin and A. N. Babatunde, "Decision Tree Algorithm," *Student's Performance Analysis Using Decision Tree Algorithms*, vol. XV, pp. 55-62, 2017.

# APPENDIX

## APPENDIX 1

### Student Performance Analysis

```
import pandas as pd
import numpy as np
from sklearn import tree
from InternalMarksPrediction.models import TotalMarks
from student.models import Student

def algorithm(request):

    df = pd.read_csv('dataset/student_performance_report.csv')
    head = df.head()
    independent = df.drop('Performance', axis='columns')
    target = df['Performance']

    model = tree.DecisionTreeClassifier(max_leaf_nodes=10,random_state=10,
criterion='gini')

    model.fit(independent,target)

    score= model.score(independent,target)

#DATABASE QUERY
account = Student.objects.get(user_id=request.user.id)
total_marks = TotalMarks.objects.get(user_id = account.id)
attendance = total_marks.total_attendance_marks
```

```

lab_attendance = total_marks.total_lab_attendance_marks
assignment = total_marks.total_assignment_marks
lab_assignment = total_marks.total_lab_assignment_marks
class_assesment = total_marks.total_class_assesment_marks
mid_assessment = total_marks.total_mid_assessment_marks
final_assessment = total_marks.total_final_assessment_marks

predict = model.predict([[attendance,lab_attendance, assignment, lab_assignment,
class_assesment, mid_assessment,final_assessment]])

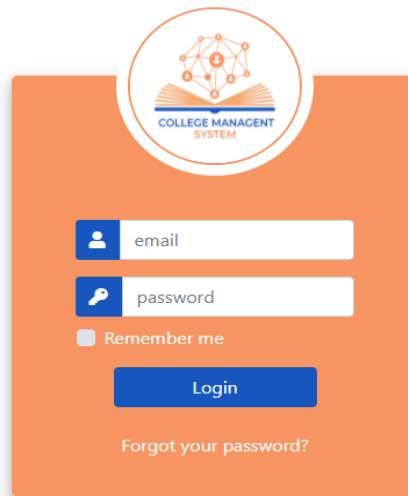
predicts = str(predict)
print(type(predicts))

if predict == 0:
    predict = "Bad"
elif predict == 1:
    predict = 'Average'
elif predict == 2:
    predict = "Good"

return predict

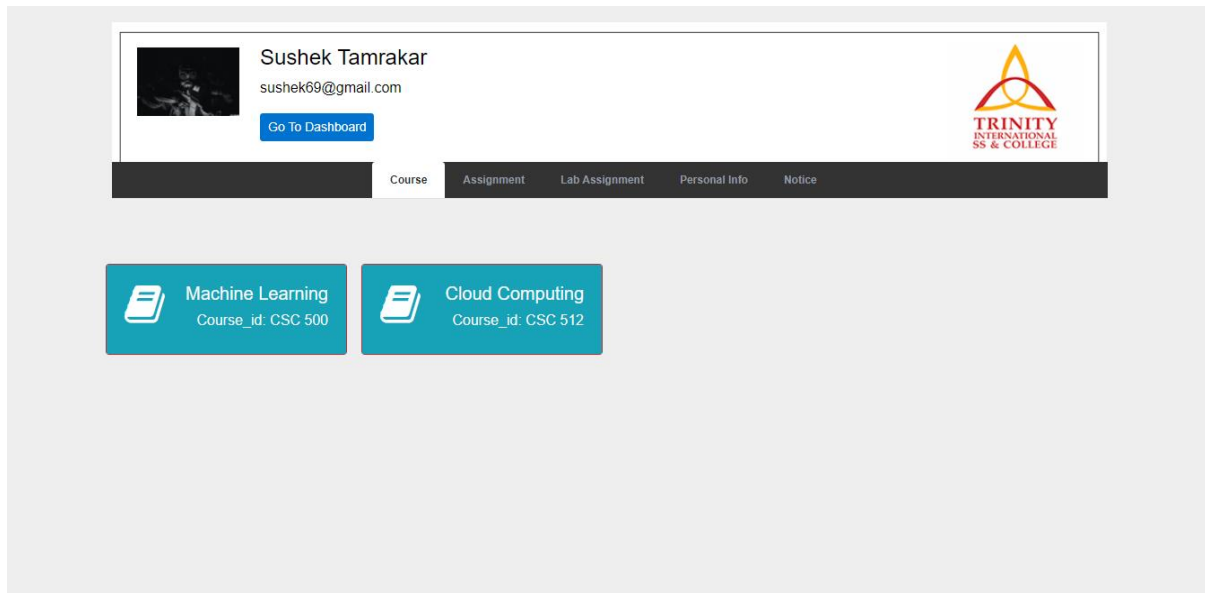
```

## APPENDIX 2



The login page features a central orange card with a white circular logo at the top. The logo contains a network diagram and the text 'COLLEGE MANAGEMENT SYSTEM'. Below the logo, there are two input fields: 'email' with a person icon and 'password' with a key icon. A 'Remember me' checkbox is located below the password field. A blue 'Login' button is positioned below the checkbox. At the bottom of the card, there is a link that says 'Forgot your password?'.

Figure 10: Login Page



The student's homepage displays a user profile at the top left with a profile picture, the name 'Sushek Tamrakar', and the email 'sushek69@gmail.com'. A 'Go To Dashboard' button is located below the profile information. To the right of the profile is the 'TRINITY INTERNATIONAL SS & COLLEGE' logo. Below the profile and logo is a navigation bar with the following links: 'Course', 'Assignment', 'Lab Assignment', 'Personal Info', and 'Notice'. The 'Course' link is currently selected. Below the navigation bar, there are two course cards. The first card is for 'Machine Learning' with 'Course\_id: CSC 500'. The second card is for 'Cloud Computing' with 'Course\_id: CSC 512'.

Figure 11: Student's Homepage



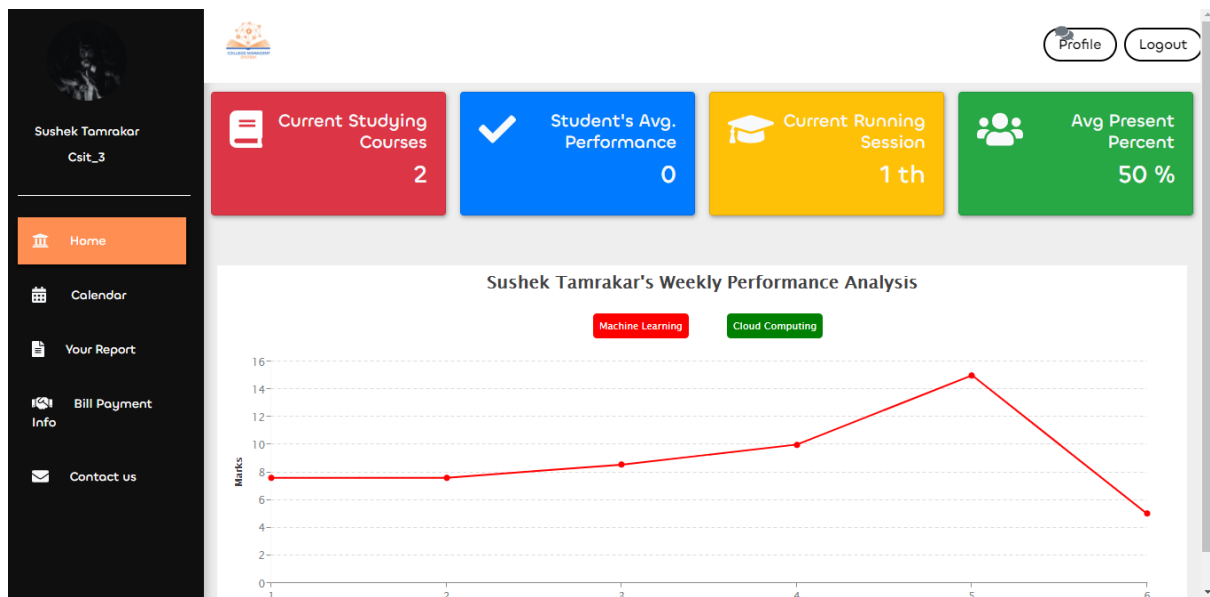


Figure 12: Student's Dashboard

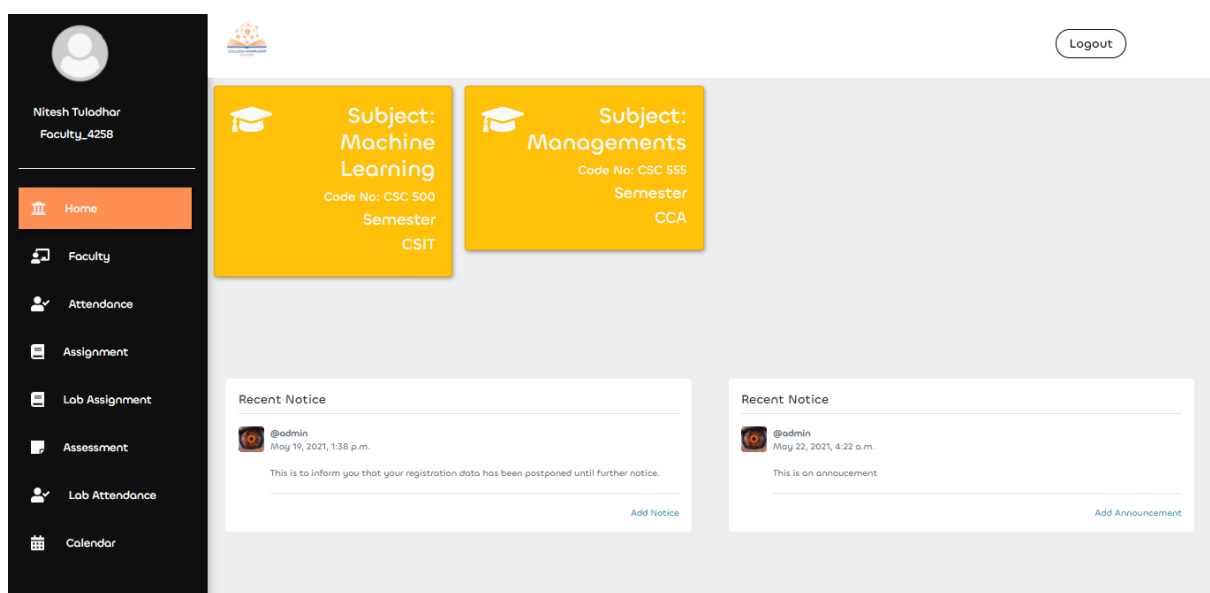


Figure 13: Teacher's Dashboard

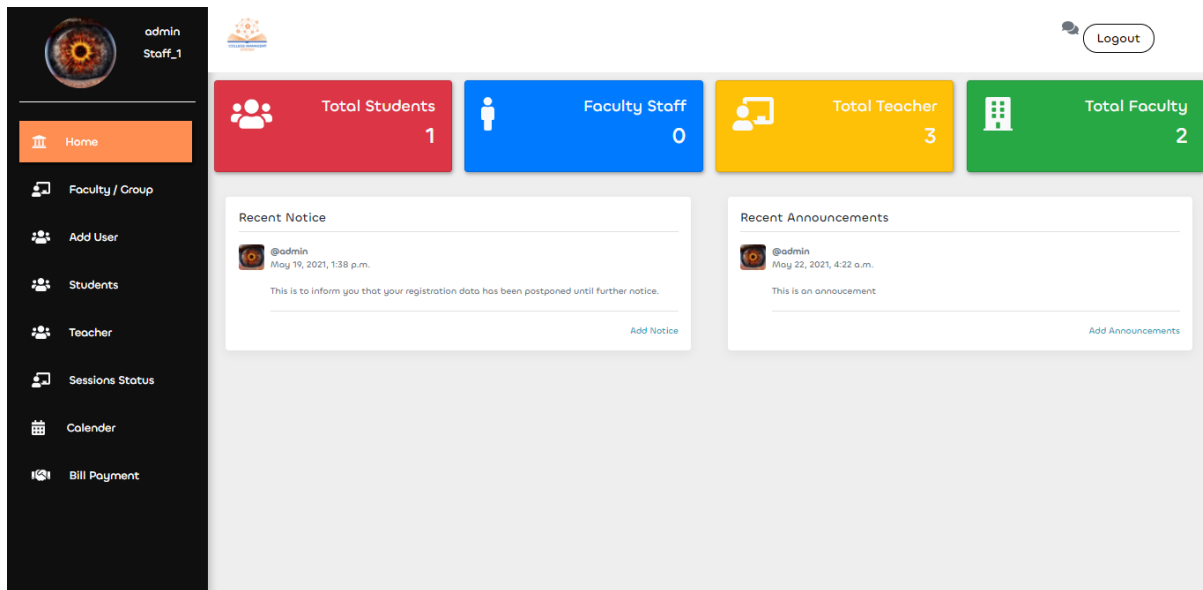


Figure 14: Admin's Dashboard

The screenshot shows the 'Add New Users' page. The sidebar is identical to Figure 14, with 'Add User' selected. The main content area has a white background with the title 'Add New Users'. The form contains the following fields:
 

- Email:
- Date Of Birth:  with a calendar icon
- Full Name:
- Address:
- Gender:  with a dropdown arrow
- Contact Number:
- Is active:  with a dropdown arrow
- Is admin:  with a dropdown arrow

Figure 15: Add New Users Page

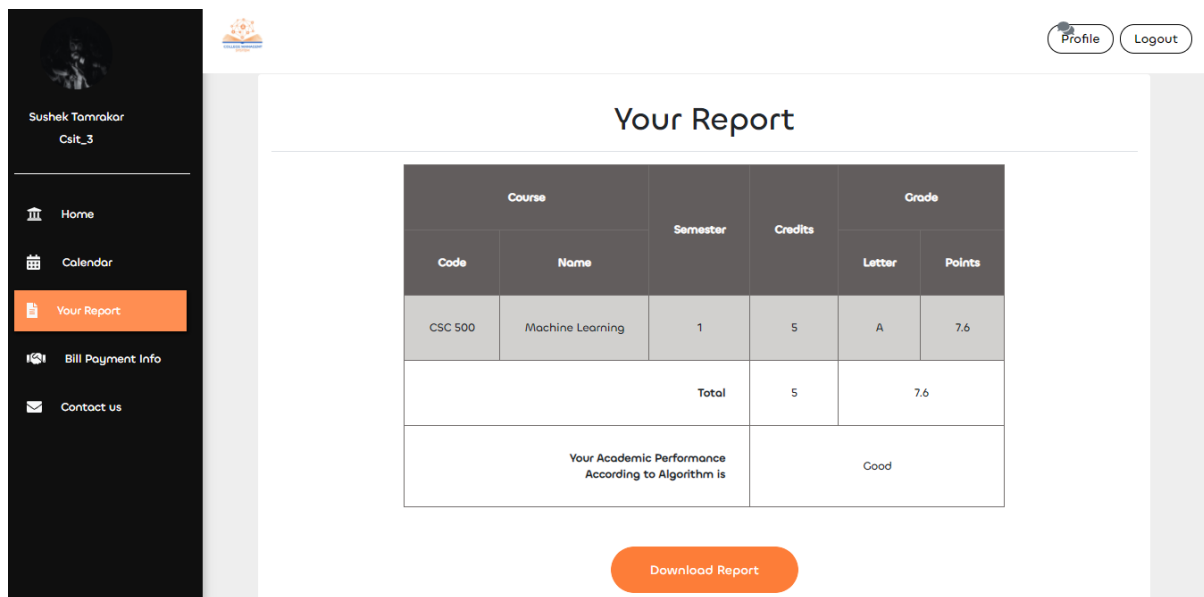


Figure 16: Student Performance Report