Student Performance Analyzer System

Final Year Project

By

Muhammad Sabahat Gulbaz

Supervised By

Dr. Ghazanfar Farooq

Department of Computer Science
Quaid-i-Azam University
Islamabad

Session (2014-2018)

In the name of Allah, the Most Merciful, the Most Kind.

ACKNOWLEDGEMENT

In the name of ALLAH, Most Beneficent, Most Merciful.

First, I thank a lot to almighty Allah for his blessings and grace in completing my project. At the start, it was looking very hard for me but with the passage of time things seems to be simple with the help of almighty Allah. I thank a lot to my Parents because without their support it was impossible for me to reach at this stage. I thanks to my all teachers Dr. M. Afzal Bhatti, Dr. Mudassir Azam Sindhu, Dr. Ghazanfar Farooq, Dr. Shuaib Karim, S.M. Naqi, Dr. Mubasher Mushtaq, Dr. Akmal Khan Khattak, Dr. Khalid Saleem, Dr. Onaiza Maqbool, Dr. Rabeeh Ayaz Abbasi, Miss Ifrah Farrukh Khan, Miss Memona Afsheen and for their kind support and cooperation throughout my degree. Special thanks to my project supervisor Dr. Ghazanfar Farooq, for their cooperation during implementation of my project. Continuous suggestions of teachers and supervisor enabled me to implement this project in better way. At the end, I thank to my friends and fellows who encouraged and motivated me throughout my degree.

Contents

Chapter 1

Software Project Management

1.1 Introduction

This chapter briefly describes student performance analyzer system for Analysis of student performance. It then highlights the problem that has been addressed in this work. It also elaborates objectives, project organization and project planning. Finally, this chapter explains report structure.

1.1.1 Project Overview

Our project is a web based student performance analyzer application. User of this application can perform statistical analysis on different data set e.g CGPA/GPA and can identify problem areas.

Opportunity

The Information Age, also called the Computer Age, the Digital Age and the New Media Age, is coupled tightly with the advent of personal computers. These computers are used in every part of life to get information. The academic performance of student is usually stored in various formats like files, documents, records etc. The available data would be analyzed to extract useful information. It becomes difficult to analyze student data by applying statistical techniques or other traditional database management tools. Hence there is a need to develop an automated tool for student performance analysis that would analyze student performance and will guide them by displaying the areas where they need improvement. There is no such system exist in Computer Science Department, Quaid-i-Azam University Islamabad. This system will take Department of Computer Science toward advancement.

Goal

The goal of the project is to design such system which will enable computer science department to analyze student's performance and to identify the factors that affects the students' performance quickly. This is a web based application providing basic analysis facilities on student's result.

1.1.2 Project Deliverables

Project deliverables are the products resulted during a project execution. Deliverables are necessary for completion of project. These deliverables test the milestone achieved during a project. Each deliverable is a step toward finishing of project.

1. **SYSTEM OVERVIEW**

In this deliverable product overview, will be explained.

2. **WHAT IS PROPOSED SYSTEM?**

In this document, the proposed system will be explained. What is proposed system and system details will be explained. Motivation, related work and proposed solution will also be explained.

3. **PROJECT MANAGEMENT PLAN**

In this deliverable, project planning will be discussed. Tasks will be identified and defined.

4. **SOFTWARE REQUIREMENT SPECIFICATIONS**

This deliverable will explain software requirement specifications. In this deliverable purpose of the product, scope of the product, product perspective like software interfaces, hardware interfaces, user interfaces and communication interfaces will be described. User characteristics and main software features will also be described.

5. **FUNCTIONAL REQUIREMENTS**

This deliverable will describe functional requirement of the software. In this deliverable use cases, will be identified and explained in fully dressed format.

6. **DATABASE REQUIREMENTS**

This deliverable will be analysis of database requirements of the system. This will explain what database will be used? Which entities will be stored in database? Which tables will be created in database?

7. **SOFTWARE DESIGN DOCUMENTATION**

This deliverable will be analysis of system design including system architectural design, data flow diagram, sequence diagrams and user interfaces.

8. SOFTWARE TEST DOCUMENTATION

This document deliverable will be the software testing documentation. It will include software testing approaches, test cases and test logs.

9. ADMIN MODULE IMPLEMENTATION

This deliverable will be the implementation of back end module of system.

10. CLIENT/FACULTY MODULE IMPLEMENTATION

This deliverable will be the implementation of client side of system from where users can search notifications.

1.2 Project Organization

Project organization involves several steps like choosing software process model, tools and techniques used for the project development.

1.2.1 Software Process Model

Software process models are used to develop software in a systematic way. There are lot of process models available. Different software models are used for different scenarios. Examples of software process models are Water Fall model, Prototyping, Incremental Development, Spiral Development etc. The approach used for development of this system is Agile Software Development.

What is Agile Software Development?

Agility is the ability to create and respond to change to succeed in an uncertain and turbulent environment. Agile Software Development is an umbrella term for a set of methods and practices based on the values and principles expressed in the Agile Manifesto. Solutions evolve through collaboration between self-organizing, cross-functional teams utilizing the appropriate practices for their context.

Reason to choose Agile Software Development

Agile Software Development has a lot of advantages. It is iterative approach mean features are delivered incrementally and product continuous to be developed. Agile software development is used to develop software quickly. This software development technique involves active user throughout the development. This technique embrace changes leading to development of right product.

Chosen Agile Process Model

There are different agile process models are available.

- Extreme Programming (XP)
- Adaptive Software Development (ASD)
- Dynamic Systems Development Method (DSDM)
- Scrum
- Crystal
- Feature Driven Development (FDD)
- Agile Modeling (AM)

Each of these model has its own advantages based on specific scenarios. Agile process model used for development of this system is Extreme Programming (XP). XP relies on object-oriented approach. Key activities involved are Planning, Design, Coding and Testing.

Roles and Responsibilities

As I am a solo person doing this project. So all the roles and responsibilities are on my end. Gathering information, Analysis, Design, Implementation and Testing fall in this category.

1.2.2. Tools and Techniques

1. **Tools used for development of Document**

   Tool used for the documents production is Microsoft Word 2016.

2. **Tools used for creating diagrams and figures**

   ArgoUML is the leading open source UML modeling tool and includes support for all standard UML 1.4 diagrams. It will be used for creating UML Diagrams is ArgoUML, Other tools recommended tools are Pencil Tool, Microsoft Visio.

3. **Tools used for development of Project Plan**

   Tool that will be used for development of Project Plan is ProjectLibre.

4. Languages used for software development

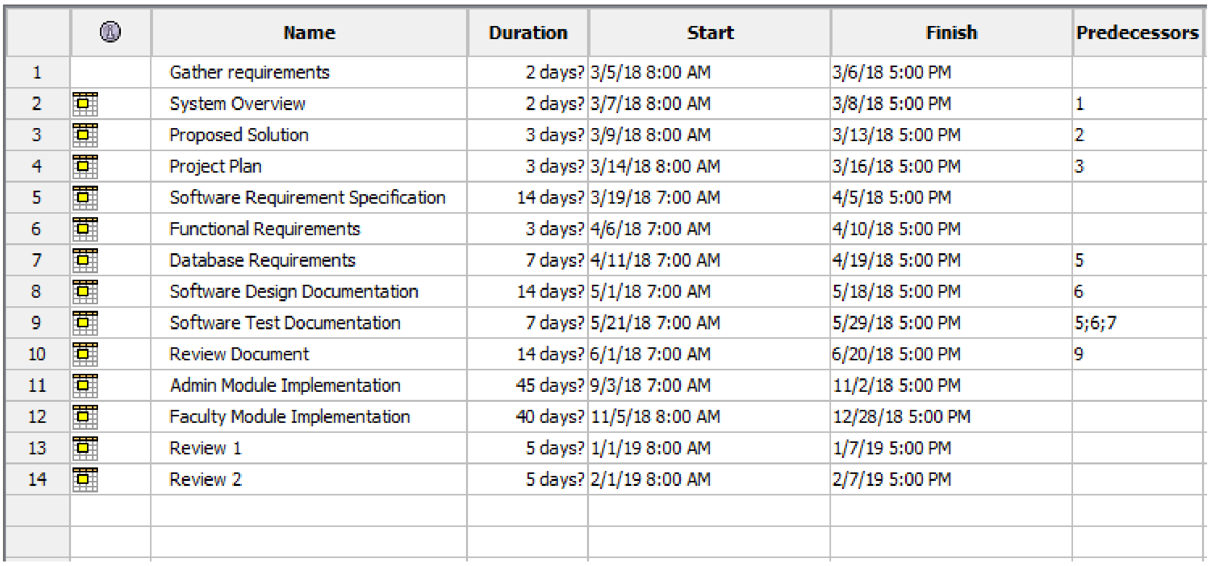
Languages used for the software development will be

- HTML5 **(**Markup Language**)**
- CSS **(**Cascading Style Sheets**)**
- JavaScript and JQuery
- PHP
- MySQL

1.3 Project Management Plan

Software project management is an umbrella activity within software engineering. It begins before any technical activity is initiated and continues throughout the modeling, construction, and deployment of computer software.

Project plan for this project is constructed using project Libre. Below are the tasks are their predecessor and time allocation for each task.



	ⓘ	Name	Duration	Start	Finish	Predecessors
1		Gather requirements	2 days?	3/5/18 8:00 AM	3/6/18 5:00 PM	
2	▦	System Overview	2 days?	3/7/18 8:00 AM	3/8/18 5:00 PM	1
3	▦	Proposed Solution	3 days?	3/9/18 8:00 AM	3/13/18 5:00 PM	2
4	▦	Project Plan	3 days?	3/14/18 8:00 AM	3/16/18 5:00 PM	3
5	▦	Software Requirement Specification	14 days?	3/19/18 7:00 AM	4/5/18 5:00 PM	
6	▦	Functional Requirements	3 days?	4/6/18 7:00 AM	4/10/18 5:00 PM	
7	▦	Database Requirements	7 days?	4/11/18 7:00 AM	4/19/18 5:00 PM	5
8	▦	Software Design Documentation	14 days?	5/1/18 7:00 AM	5/18/18 5:00 PM	6
9	▦	Software Test Documentation	7 days?	5/21/18 7:00 AM	5/29/18 5:00 PM	5;6;7
10	▦	Review Document	14 days?	6/1/18 7:00 AM	6/20/18 5:00 PM	9
11	▦	Admin Module Implementation	45 days?	9/3/18 7:00 AM	11/2/18 5:00 PM	
12	▦	Faculty Module Implementation	40 days?	11/5/18 8:00 AM	12/28/18 5:00 PM	
13	▦	Review 1	5 days?	1/1/19 8:00 AM	1/7/19 5:00 PM	
14	▦	Review 2	5 days?	2/1/19 8:00 AM	2/7/19 5:00 PM	

Figure 1. 1 Time Schedule

1.3.1 Gant Chart

A Gantt chart is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis.

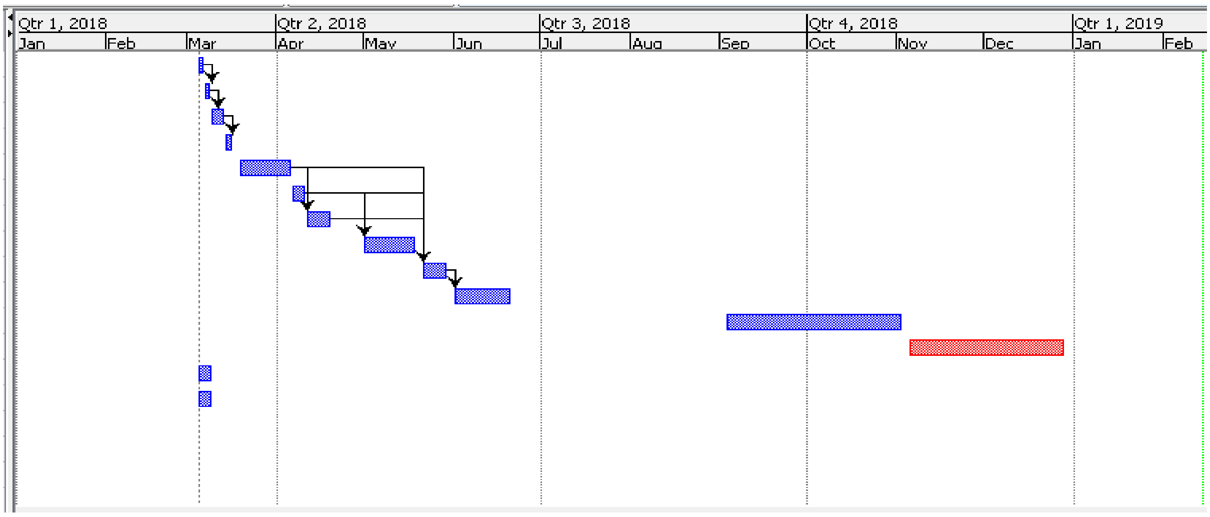


Figure 1. 2 Gantt chart

Chapter 2

Software Requirement Specification

2.1 Introduction

In this chapter, software requirements will be discussed. Project overview, proposed system, motivation behind this system, related work etc. will be explained.

2.1.1 Product Overview

This product is a web based student performance analyzer application. User of this application can perform statistical analysis on different data set e.g CGPA/GPA and can identify problem areas. System will provide a simple user interface. The system is designed for the department of computer science, Quaid-i-Azam University Islamabad. It will facilitate department office and faculty members.

2.1.2 Specific Requirements

This chapter presents a detailed analysis of functional and non-functional requirements of the developed application. The deliverable result, at the end of this phase, is Software Requirement Specification (SRS) document. The detailed requirements analysis phase should be carried out carefully because it decides what to build. It also describes the summary of the functions that the software will perform.

2.1.3 Proposed System

In the department of computer science, student's result data stored in paper based record files of different categories. Different categories are Student Registration Information, Student semester courses marks and student result information. Analysis of student's performance from these record is quite a difficult task. Performance analysis from the CSV based file is time consuming thing. These operations are purely paper based. The proposed solution for managing student data in an advanced way is to make a web based system. System will provide basic data managing and performance analysis facilities. Analysis will be performed under different techniques. For example Analysis can be perform on the basis of gender, semester base, on the bases of batch, also on the bases of CGPA and GPA. And result will be shown in the form of graphs.

Collection of Data

In department of Computer Science, official student result stored in different places in the form of CSV based file records. To make system of performance analysis, Data will be collected from the office.

2.1.4 Motivation

This is era of technology. Everything is being automated. Traditional paper based file records are being converted to online storage systems. These systems provide a lot of functionality like searching of record. Online systems are a great way to get information when needed in no time. In the Department of Computer Science, Quaid-i-Azam University Islamabad, no such online system exists. The motivation behind this system is to facilitate Department of Computer Science to make analysis of student performance. This system is designed for the Department of Computer Sciences, QAU Islamabad. In traditional paper files, analysis of data is a challenging and time-consuming thing. This system will enable computer science department to make analysis in matter of seconds. This project will be a step toward advancement.

2.1.5 Related Work

A) Study of Existing Systems Similar to proposed System:

Here we have chosen three existing systems as study of these helped us to propose our system.

Student Performance Analysis System (SPAS) [1]

Chew Li Sa et.al have proposed a framework named Student Performance Analysis System (SPAS) which is able to predict the students' performance in course "TMC1013 System Analysis and Design", which in turns assists the lecturers from Information System department to identify students that are predicted to have bad performance in course "TMC1013 System Analysis and Design". The proposed system offers student performance prediction through the rules generated via statistical technique. The statistical technique used in this project classifies the students based on students' grade and gender.

IUS performance analysis on student [2]

IUS predict student performance on the bases of semester work load.

Science direct Review on predicting student performance [3]

A research paper in which predict student performance on the bases of no of semester at university using statistical techniques.

2.1.6 Purpose

Paper based file records are traditional way to store data, documents etc. These file records are of different categories like a file record for student's related documents, file record for faculty related documents, file record for scholarships documents. These file records are maintained to keep a record of the documents. Whenever someone needed the document, they can get the document from the file record. The available data would be analyzed to extract useful information. It becomes difficult to analyze student data by applying old statistical techniques or other traditional database management tools. Hence there is a need to develop an automated tool for student performance analysis that would analyze student performance and will guide them by displaying the areas where they need improvement.

2.1.7 Scope

Student performance Analyzer system is a web based system to make analysis of student performance.

- To develop a system for students' performance analysis.
- To identify the factors that affects the students' performance.
- To develop a student monitoring tool which will counsel students in order to provide them an insight regarding their academics & areas where improvement is required.

2.2 Interfaces

Interfaces for the system includes software interfaces, hardware interfaces, user interfaces and communication interfaces.

1. **Software Interfaces**

Since the application is web based. It can be easily viewed on desktop computers, laptops and smartphones through web browsers. The required software to run the application are:

- Any updated web browser meeting the today's web technology requirements.

2. **Hardware Interfaces**

The hardware interfaces required to run the application are following. The application can be run on any on the following.

- Desktop computers
- Laptops

3. **User Interfaces**

   User can interact with the application through mouse, or if his device has touch functionality then the user can interact using touch gestures.

4. **Communication Interfaces**

   Communication interface for system is web interface.

2.2.1 Software Product Features

Software has two types of users. One is administrator who will add student data to the system etc. and the others are Faculty Members. Faculty members will only perform the analysis and browse the result.

Administrator Functionality

1. Login
2. Create batch
3. Add Student registration CSV file
4. Add Student courses marks CSV file
5. Add Student Result information CSV file

Faculty Member Functionality

1. Sign up
2. Login
3. Search Student's data

4. Perform Analysis on the gender basis

5. Perform analysis on the GPA/CGPA bases

6. Can check failure ratio against a semester or against a course

2.2.2 Use Cases Description

Use cases description is given below.

Use Cases

Below are the identified use cases for the whole system.

1. Sign Up

2. Login

3. Upload student registration information

4. Upload student semester courses marks

5. Upload student status information

6. View student’s performance

2.2.3 Use Cases Diagram

Use-Case diagram help to determine the functionality and features of the software from the user's perspective. Use case diagram of Student performance Analyzer system is given below.

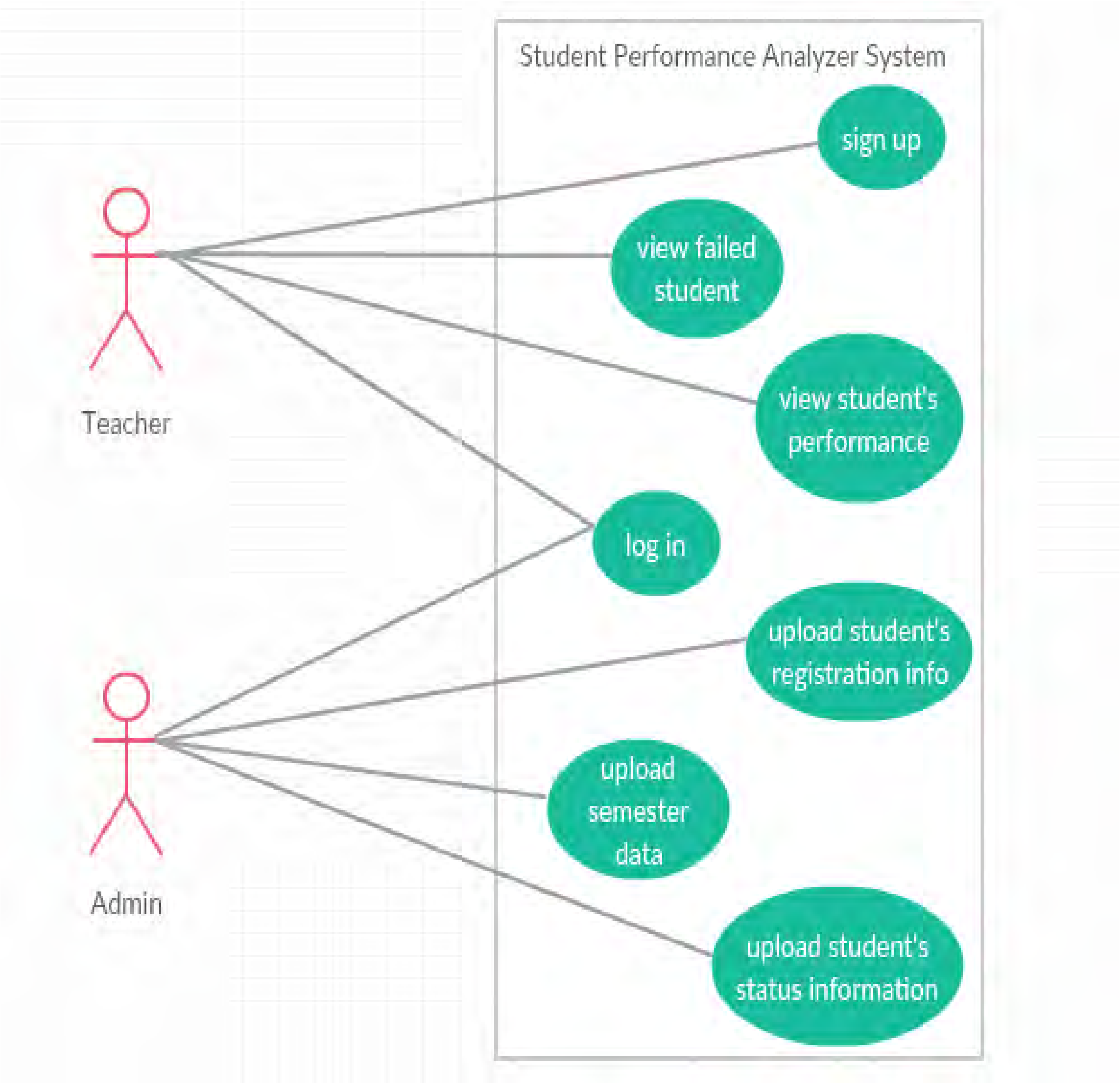

Student Performance Analyzer System
sign up
view failed student
view student's performance
log in
upload student's registration info
upload semester data
upload student's status information
Teacher
Admin

Figure 2. 1 Use case diagram

Use Cases Description

1. Sign Up

The Faculty Member first must sign up to use the system.

Use Case ID	UC-01
Use Case Name	Sign Up
Primary Actor	Faculty Member
Pre-Condition	The user has accessed the website.
Post-Condition	The user has successfully signed up.
Main Success Scenarios	1. The System prompts the user for a username and password or Register new account. 2. The System prompts user for registration information, Username, password, etc. 3. The user enters information. 4. System verifies information and creates account.
Alternative Flows	3a. The page is not loaded successfully. 4a. The user enters incorrect information.
Special Requirements	No special requirements are required other than a web browser And internet.
Frequency of occurrence	Whenever there will be a new user.

Table 2. 1 Signup UC Description

2. Log in

The admin and teacher first must log in to use the system.

Use Case ID	UC-02
Use Case Name	Sign-in
Primary Actor	Admin, Faculty Member
Pre-Condition	The user has accessed the website and has account on the System.
Post-Condition	The user has successfully logged in.
Main Success Scenarios	1. The System prompts the user for a username and password. 2. The user enters username and password. 3. The user clicks log in button. 4. System verifies information. 5. User is logged in.
Alternative Flows	2a. The user enters wrong input. 4a. System does not verify information.
Special Requirements	No special requirements are required other than a web browser and internet.
Frequency of occurrence	Whenever the session expires, user should login.

Table 2. 2 Login UC Description

3. View Student’s Performance

Faculty Member will be logged in and view Student Performance.

Use Case ID	UC-03
Use Case Name	View Students Performance
Primary Actor	Faculty Member
Pre-Condition	The user should be logged in.
Post-Condition	Results with statistical analysis.
Main Success Scenarios	1. Faculty Member perform analysis on student data 2. System Extract Meaning full information. 3. System return the results.
Alternative Flows	2a. The user enters wrong input. 4a. System does not verify information.
Special Requirements	No special requirements are required other than a web browser And internet.
Frequency of occurrence	Whenever the session expires, user should login.

Table 2. 3 View Students Performance Use Case description

4. View failed Students

Faculty Member will be logged in and view failed Student.

Use Case ID	UC-03
Use Case Name	View Failed Student
Primary Actor	Faculty Member
Pre-Condition	The user should be logged in.
Post-Condition	Result with failed students name, registration no and with CGPA.
Main Success Scenarios	1. Faculty Member perform analysis on student data 2. System Extract Meaning full information. 3. System return the results.
Alternative Flows	2a. The user enters wrong input. 4a. System does not verify information.
Special Requirements	No special requirements are required other than a web browser And internet.
Frequency of occurrence	Whenever the session expires, user should login.

Table 2. 4 View Failed Student Use Case Description

5. Upload student registration Info

Admin will be logged in and add student registration file to system.

Use Case ID	UC-04
Use Case Name	Upload Student's Registration Info
Primary Actor	Admin
Pre-Condition	The user should be logged in.
Post-Condition	File is uploaded successfully.
Main Success Scenarios	1. Admin will create a batch 2. Admin will add student registration file. 3. System read file and store in database. 4. System return success if record successfully uploaded otherwise return fail.
Alternative Flows	2a. The user enters wrong input file. 4a. System does not verify information.
Special Requirements	No special requirements are required other than a web browser and internet.
Frequency of occurrence	Whenever Admin create batch, admin will upload registration file.

Table 2. 5 Student registration file upload Use Case description

6. Upload semester data

Admin will be logged in and add student's semester courses marks file to system.

Use Case ID	UC-05
Use Case Name	Upload semester data
Primary Actor	Admin
Pre-Condition	The user should be logged in.
Post-Condition	File is uploaded successfully.
Main Success Scenarios	1. Admin will create a batch 2. Admin will add student semester courses marks file. 3. System read file and store in database. 4. System return success if record successfully uploaded otherwise return fail.
Alternative Flows	2a. The user enters wrong input file. 4a. System does not verify information.
Special Requirements	No special requirements are required other than a web browser and internet.
Frequency of occurrence	Whenever Admin create batch, admin will upload courses marks file against each semester

Table 2. 6 upload student semester courses marks Use Case description

7. Upload student status information

Admin will be logged in and add student status information file to system.

Use Case ID	UC-06
Use Case Name	Upload student status information
Primary Actor	Admin
Pre-Condition	The user should be logged in.
Post-Condition	File is uploaded successfully.
Main Success Scenarios	1. Admin will create a batch 2. Admin will add student status information file. 3. System read file and store in database. 4. System return success if record successfully uploaded otherwise return fail.
Alternative Flows	2a. The user enters wrong input file. 4a. System does not verify information.
Special Requirements	No special requirements are required other than a web browser and internet.
Frequency of occurrence	Whenever Admin create batch, admin will add status information file against each semester

Table 2. 7 Upload Student Status Information Use Case Description

2.2.4 Software System Attributes

Reliability

Reliability of system is the ability of a computer program to perform its intended functions and operations in a system's environment, without experiencing failure (system crash). This system should perform all operations without failure. It should give proper analysis results with precision more than 90 %.

Availability

The system must be available for use when it is needed. The web site should have 99.9 % uptime.

Security

- System should be secure. The system should not be used by any unauthorized user.
- System should not allow any fake sign up, login
- Secure database
- No bogus entries to database
- No unauthorized changes in data

Maintainability

The applications will be developed in such a way that it will facilitate maintainability.

Portability

As the system is a web based application. It can be used on smartphone and laptops as well as on desktop computers. The system will provide responsive interface.

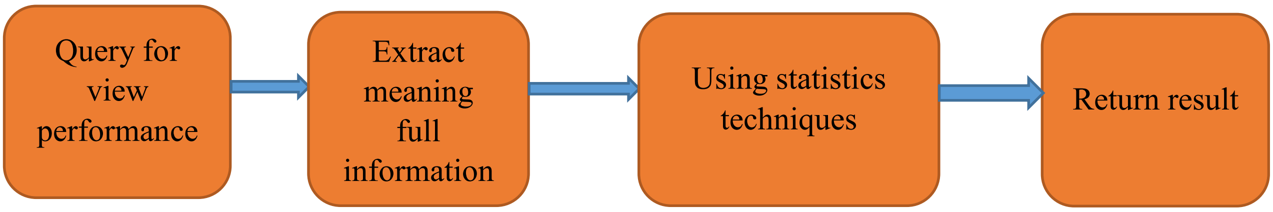
Performance

- Application should not crash on any user action
-
- Application should provide responsive interfaces on all devices
- Applications should have 80 % speed.
- Page load time should be less than or equal to 2 seconds.

2.2.5 System Functionality

The system functionality specifies those function that a system must perform.

System will extract meaning full information from large data using statistical techniques.


Query for view performance
Extract meaning full information
Using statistics techniques
Return result

2.2.6 Database Requirements

This application will have database. We will use the XAMPP server to store the data.

Because in web based system it is easy to get data from XAMPP server.

-

Entity Relationship Diagram

Entity relationship diagram for system is show below

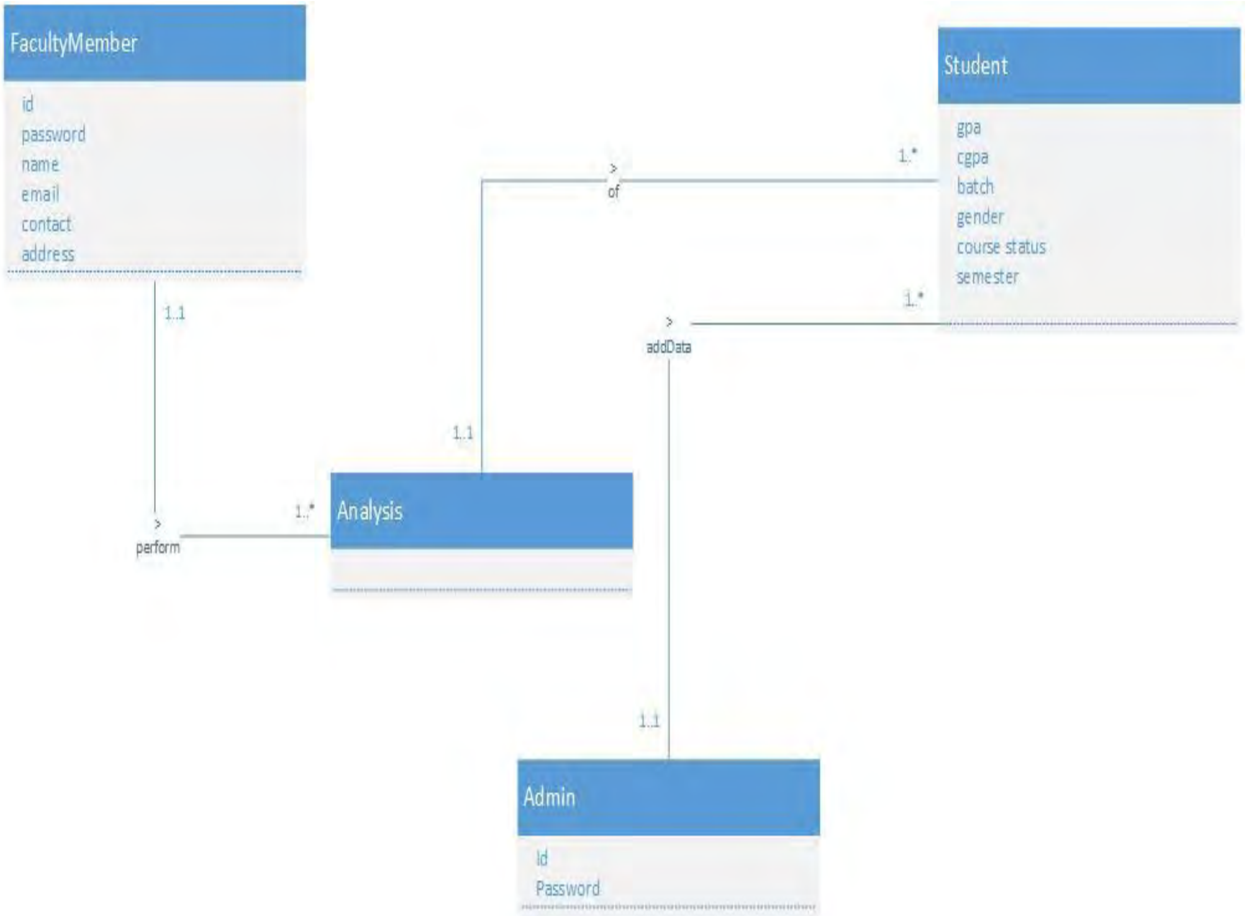


Figure 2. 2 Entity Relationship Diagram

Chapter 3

Software Design Description

3.1 Introduction

This chapter introduces Software Design in more details. In this chapter we will discuss sequence diagrams, data flow diagram, database design, software architecture, detailed description of user interfaces. A software design description (SDD) is a written description of a software product, that a software designer writes in order to give a software development team overall guidance to the architecture of the software project. An SDD usually accompanies an architecture diagram with pointers to detailed feature specifications of smaller pieces of the design.

3.1.1 Design Overview

Design of system explains overall design of system. It conveys the purpose of different components and modules of system. Design of system is built in a step by step procedure. It involves software architecture, domain model, data flow diagram, database design and user interface design.

3.1.2 Requirements Traceability Matrix

Software requirements should always be traceable. Through this matrix each requirement can be traced throughout the development life cycle.

Below table shows traceability matrix.

Req. Id	**Functional Requirement**	**Architectural/Design document**	**Test Case**	**language**	**status**
UC 1	Sign up	Software requirement specification	TC-01	PHP	implemented
UC 2	login	Software requirement specification	TC-02	PHP	implemented
UC 3	View students' performance	Software requirement specification	TC-03	PHP	implemented
UC 4	View failed students	Software requirement specification	TC-04	PHP	implemented
UC 5	Upload student registration file	Software requirement specification	TC-05	PHP	implemented
UC 6	Upload semester data	Software requirement specification	TC-06	PHP	implemented
UC 7	Upload student status information	Software requirement specification	TC-07	PHP	implemented

Table 3. 1 Traceability Matrix

3.2 Software Architectural Design

An architecture is the set of significant decisions about the organization of a software system, the selection of the structural elements and their interfaces by which the system is composed, together with their behavior as specified in the collaborations among those elements, the composition of these structural and behavioral elements into progressively larger subsystems, and the architectural style. Architectural design is the resolution of the requirements in the design of the software, the hardware and networking, operations, policies, and so forth.

3.2.1 Chosen System Architecture

3-Tier architecture is used to make the Performance Analyzer for CS Department (PAFCD). Basically, Three-tier (layer) is a client-server architecture in which the user interface, business process (business rules) and data storage and data access are developed and maintained as independent modules or most often on separate platforms. Basically, there are 3 layers, tier 1

(presentation tier, GUI tier), tier 2 (business objects, business logic tier) and tier 3 (data access tier). These tiers can be developed and tested separately. The Three-tier Architecture contains:

Presentation tier

This is the topmost level of the application. The presentation tier displays information related to such services as browsing merchandise, purchasing and shopping cart contents. It communicates with other tiers by which it puts out the results to the browser/client tier and all other tiers in the network. In simple terms, it is a layer which users can access directly (such as a web page, or an operating system's GUI).

Application tier (business logic, logic tier, or middle tier)

The logical tier is pulled out from the presentation tier and, as its own layer, it controls an application's functionality by performing detailed processing.

Data tier

The data tier includes the data persistence mechanisms (database servers, file shares, etc.) and the data access layer that encapsulates the persistence mechanisms and exposes the data. The data access layer should provide an API to the application tier that exposes methods of managing the stored data without exposing or creating dependencies on the data storage mechanisms. As with the separation of any tier, there are costs for implementation and often costs to performance in exchange for improved scalability and maintainability.

Architecture Diagram

Three-tier architecture of this system has interface tier, business logic and database. Interface tier is from where user interacts with application. Business logic is where the whole application logic works. Database tier is responsible for storing and maintaining data. System architecture diagram for this system is given below

Architecture diagram for Performance Analyzer for CS Department in fig 3.1

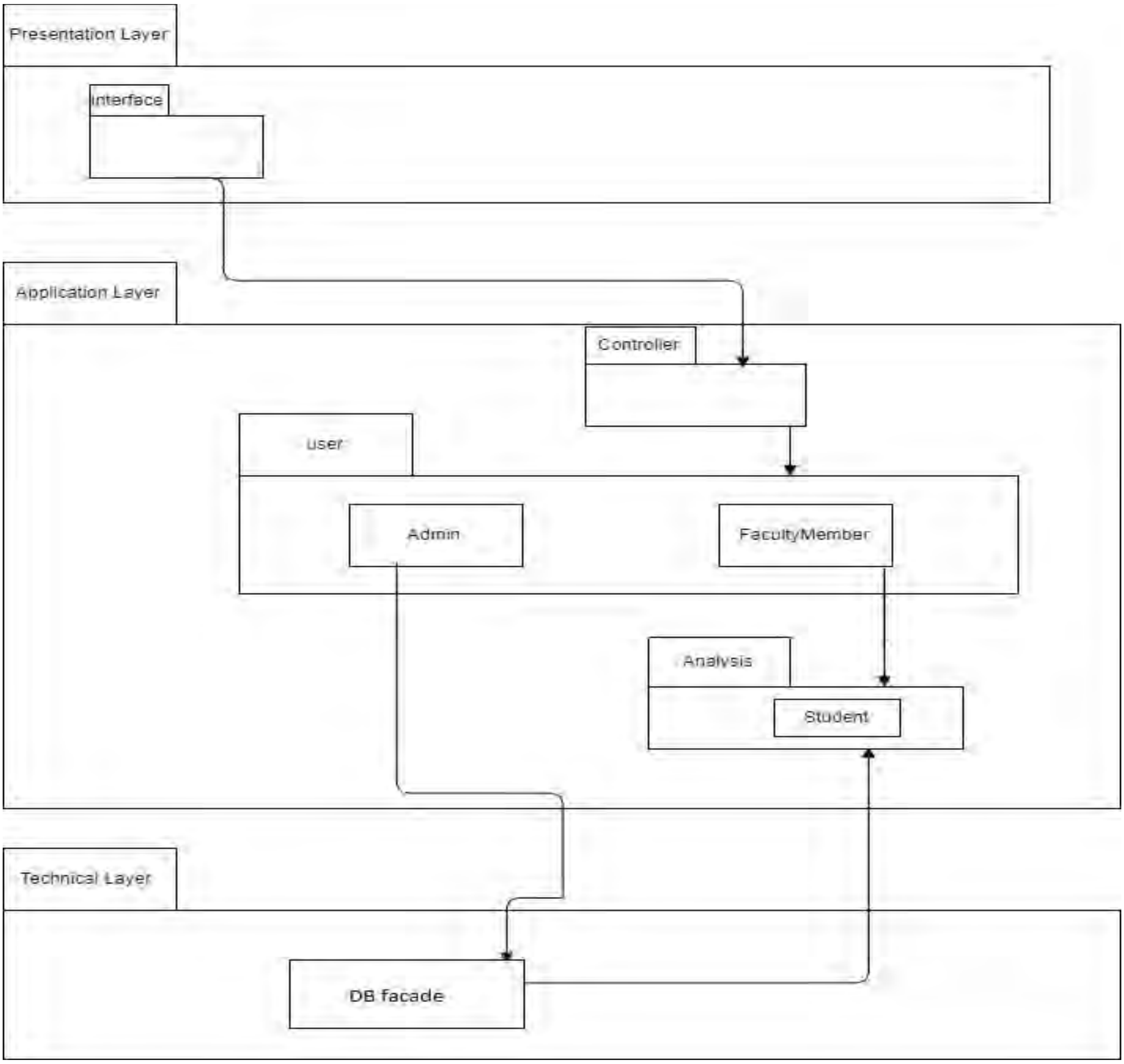


Figure 3. 1 Logical View of layers in PAFCD

3.3 Detailed Design

Detailed designed for the system includes relational database design, system sequence diagrams, data flow diagrams and user interface design.

3.3.1 System Sequence Diagrams (SSD)

A SSD is a sequence diagram that shows, for a particular scenario of a use case, the events that external actors generate, their order, and possible inter-system events. SSD is visual summaries of the individual use cases.

SSD-1 (Register Faculty Member)

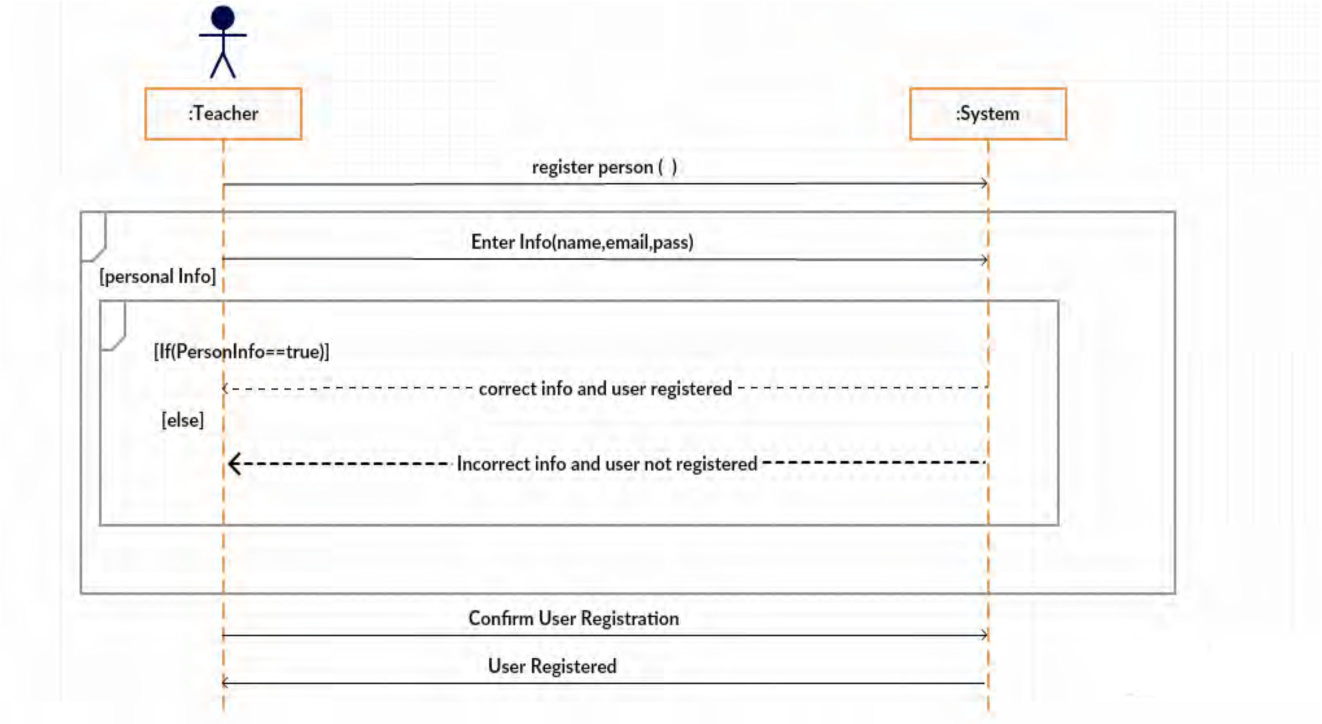

:Teacher
:System
register person ( )
Enter Info(name,email,pass)
[personal Info]
[If(PersonInfo==true)]
correct info and user registered
[else]
Incorrect info and user not registered
Confirm User Registration
User Registered

Figure 3. 2 Register Faculty Member

SSD-2 (View Student Performance)

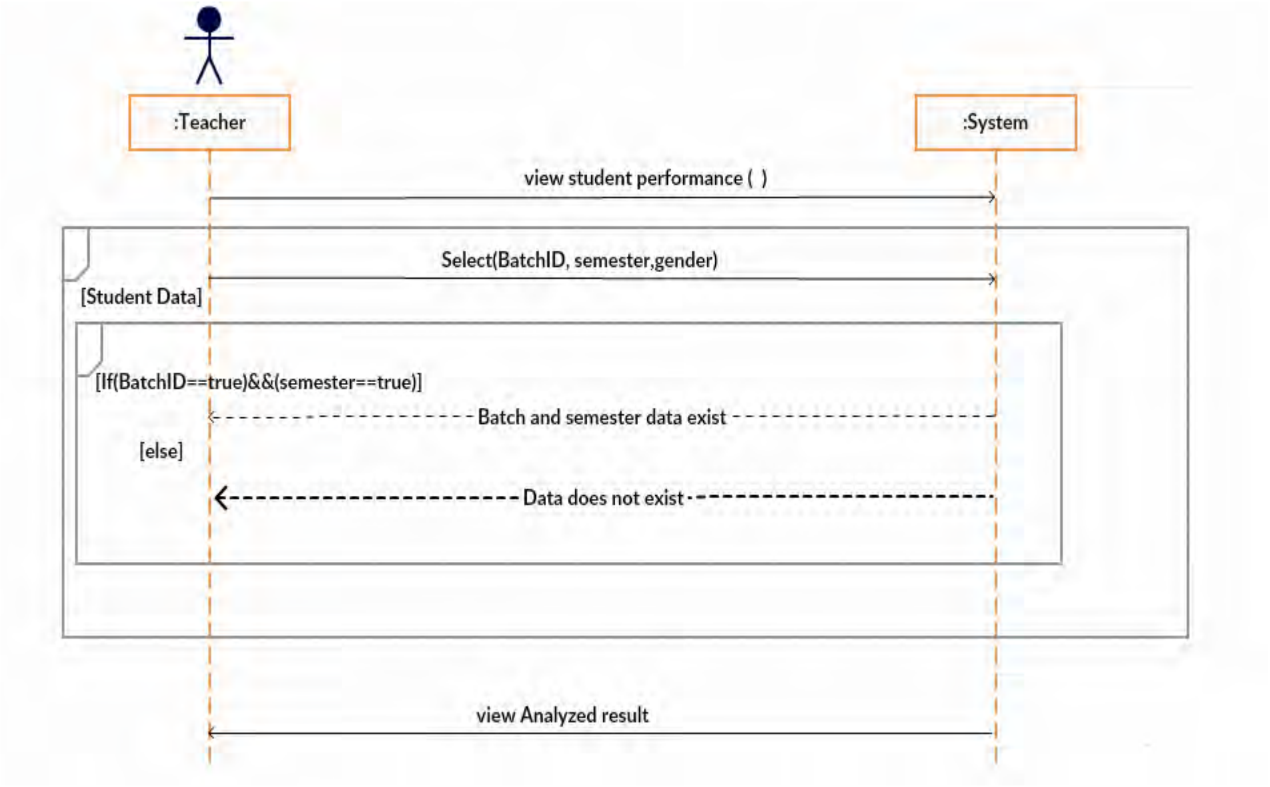

:Teacher
:System
view student performance ( )
Select(BatchID, semester,gender)
[Student Data]
[If(BatchID==true)&&(semester==true)]
Batch and semester data exist
[else]
Data does not exist
view Analyzed result

Figure 3. 3 View Student Performance

SSD-3 (Upload Student Registration Info)

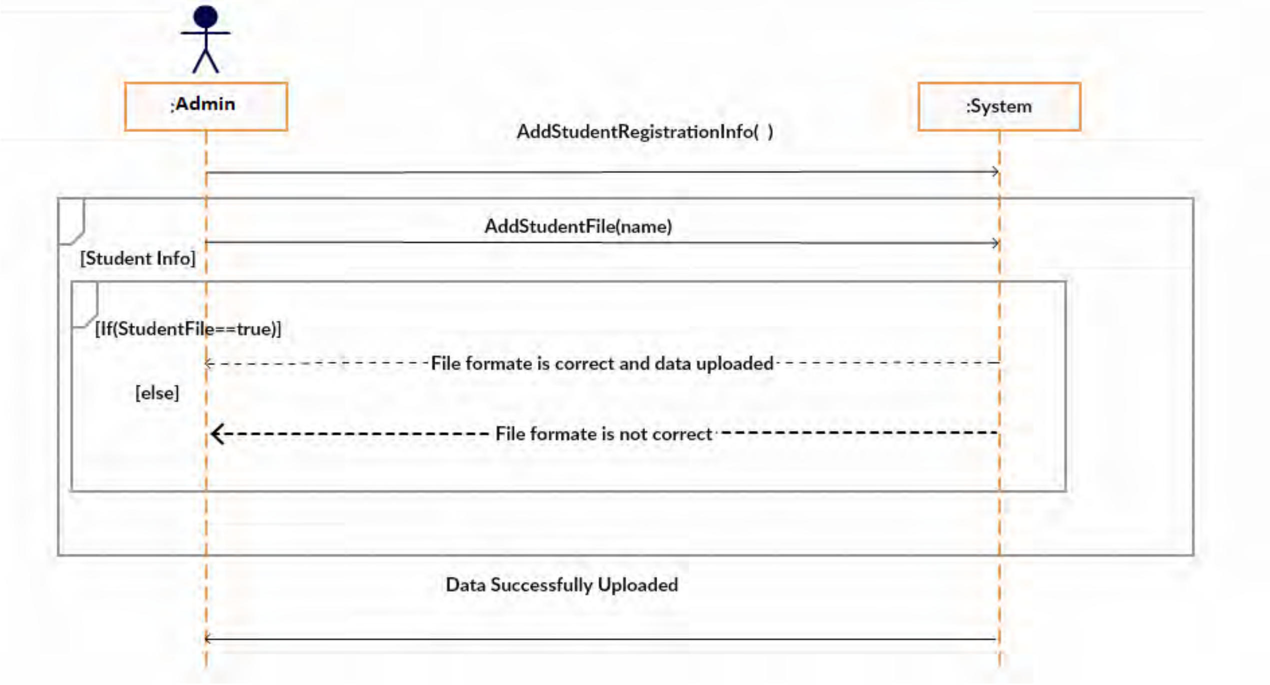
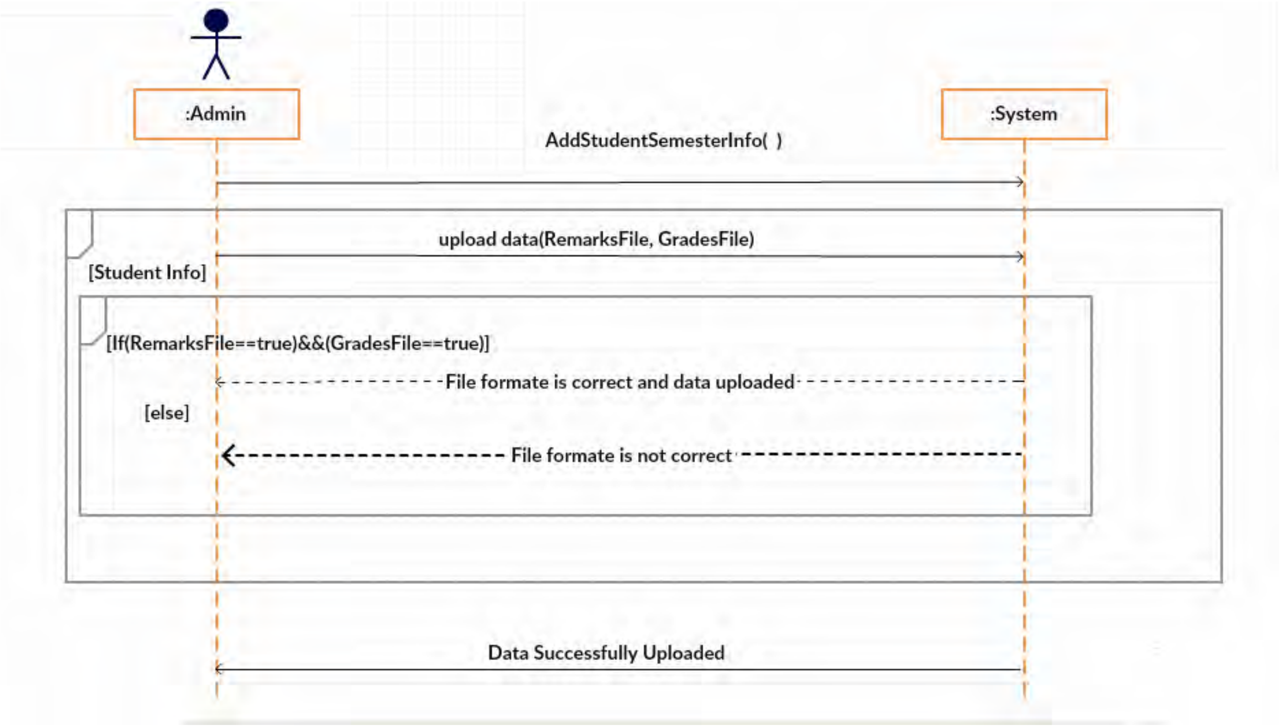

:Admin
:System
AddStudentRegistrationInfo( )
AddStudentFile(name)
[Student Info]
[If(StudentFile==true)]
File formate is correct and data uploaded
[else]
File formate is not correct
Data Successfully Uploaded

Figure 3. 4 Upload Student Registration Info

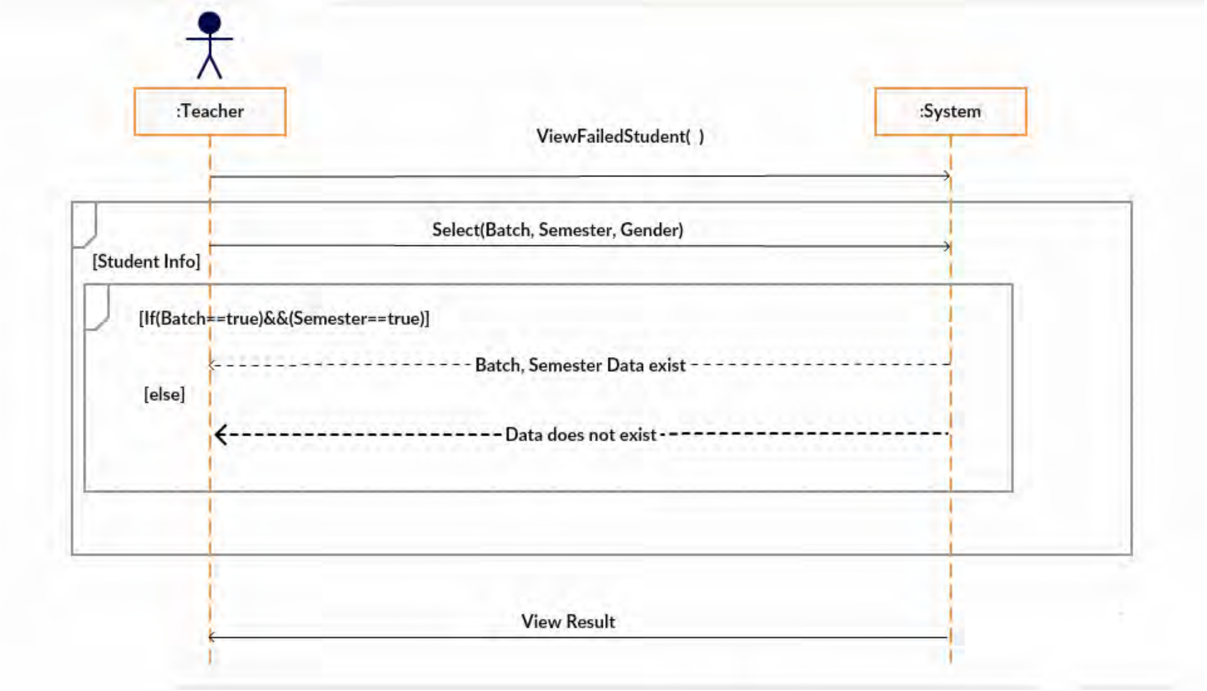
SSD-4 (Upload Semester Data)



:Admin
:System
AddStudentSemesterInfo( )
upload data(RemarksFile, GradesFile)
[Student Info]
[If(RemarksFile==true)&&(GradesFile==true)]
File formate is correct and data uploaded
[else]
File formate is not correct
Data Successfully Uploaded

Figure 3. 5 Upload Semester Data

SSD-5 (View Failed Student)



:Teacher
:System
ViewFailedStudent( )
Select(Batch, Semester, Gender)
[Student Info]
[If(Batch==true)&&(Semester==true)]
Batch, Semester Data exist
[else]
Data does not exist
View Result

Figure 3. 6 View Failed Student

3.3.2 Data Flow Diagrams

A data flow diagram (DFD) is a way of representing a flow of a data of 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. A data flow diagram has no control flow, there are no decision rules and no loops.

DFD-1 (Context level Data Flow Diagram)

The Context Diagram shows the system under consideration as a single high-level process and then shows the relationship that the system has with other external entities (systems, organizational groups, external data stores, etc.).

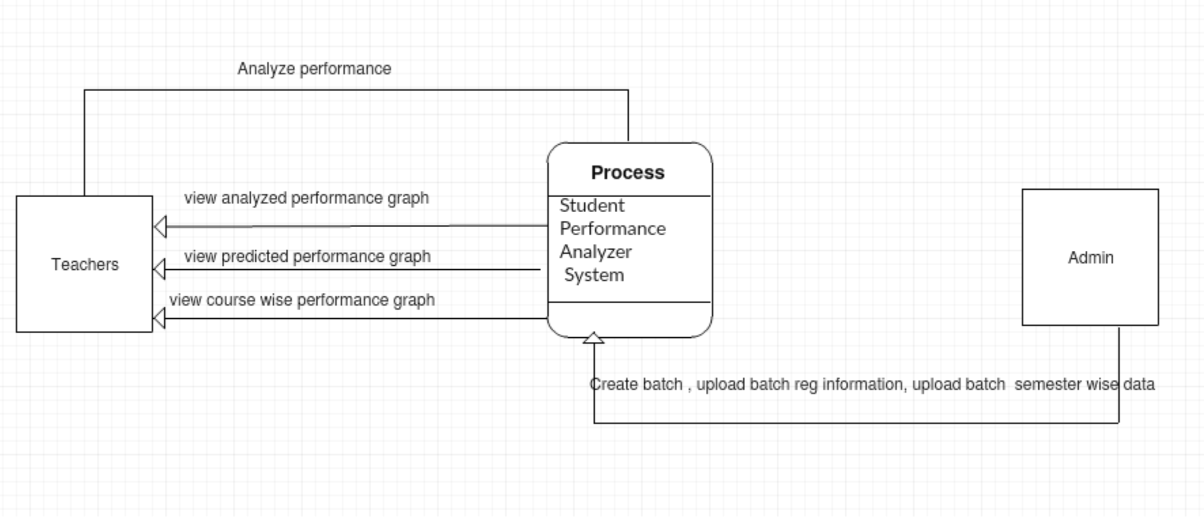

Analyze performance
Process
Student Performance Analyzer System
view analyzed performance graph
Teachers
view predicted performance graph
view course wise performance graph
Admin
Create batch , upload batch reg information, upload batch semester wise data

Figure 3. 7 context level dataflow diagram of system

DFD-2 (2nd level admin side DF diagram)

This diagram show the complete flow how the admin will interact with the system

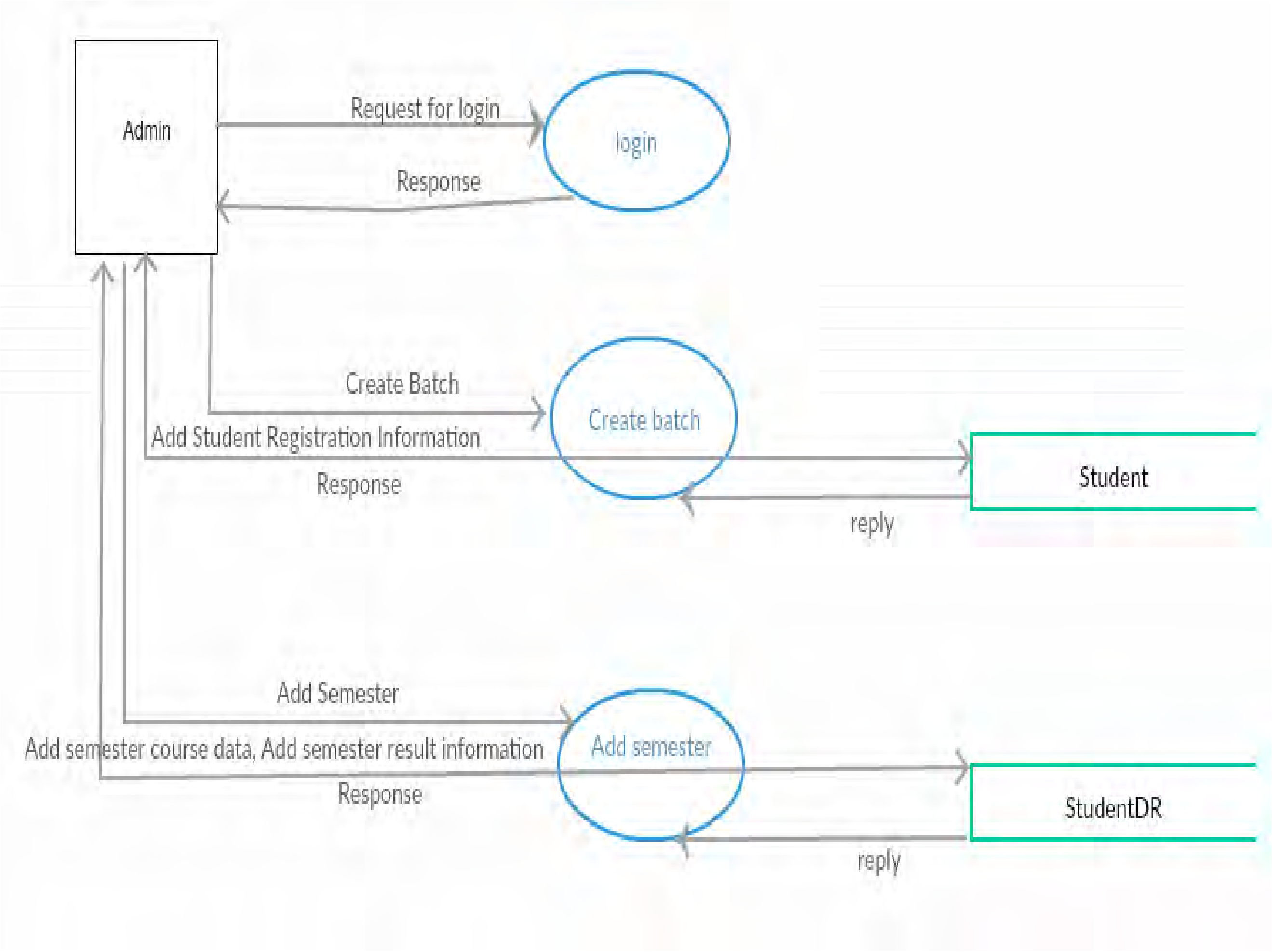


Figure 3. 8 2nd level admin side DF diagram

DFD-3 (2nd level Client/Faculty side DF diagram)

This diagram show the complete flow how the Teacher will interact with the system.

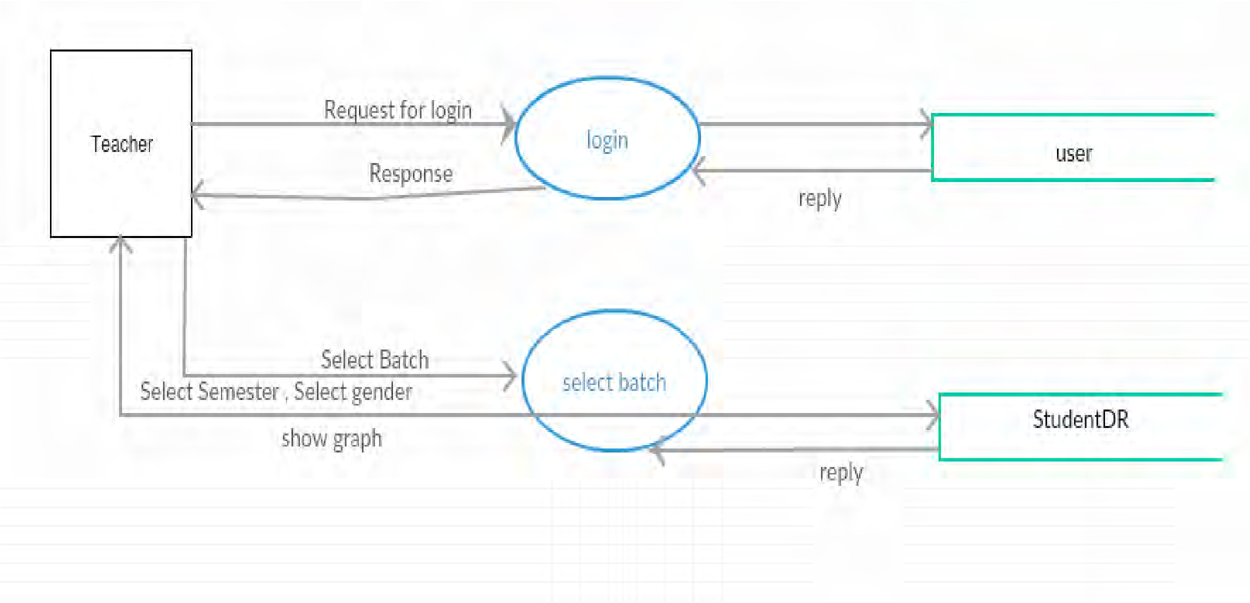


Figure 3. 9 2nd level teacher side DF diagram

3.3.3 Activity Diagram

An activity is a state of doing something. The activity diagram describes the sequencing of activities. Activity diagram depicts the dynamic behavior of a system or part of a system through the flow of control between actions that the system performs. It is similar to a flowchart except that an activity diagram can show concurrent flows. Activity diagrams for this system is given in following:

AD-1 Add Student Data (Activity diagram)

Create batch, Upload Student Registration File, Upload Student Remarks file and Upload Student Grades File by administrator is under the umbrella of manage product.

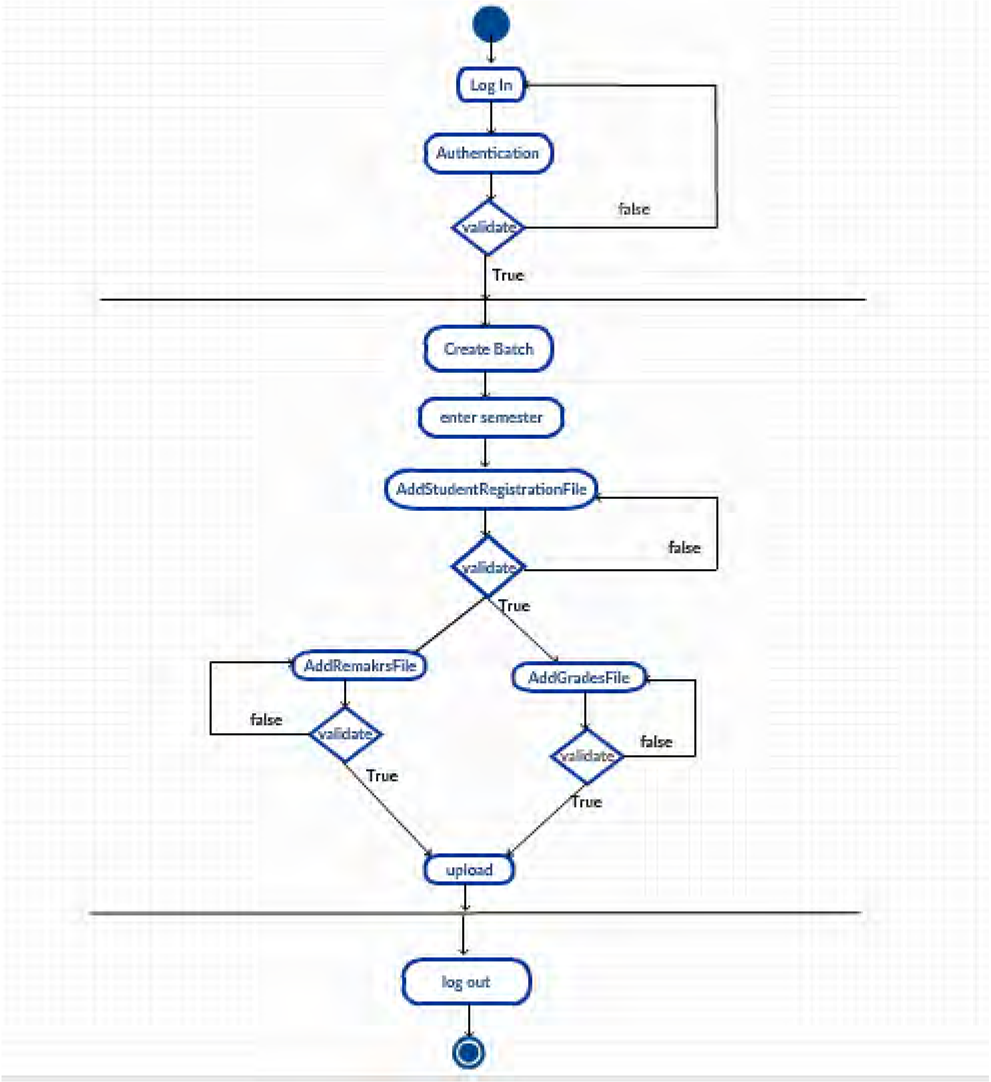


Figure 3. 10 Add Student Data Activity Diagram

AD-2 View Students Performance Result (Activity diagram)

This activity will be performed by the Teacher

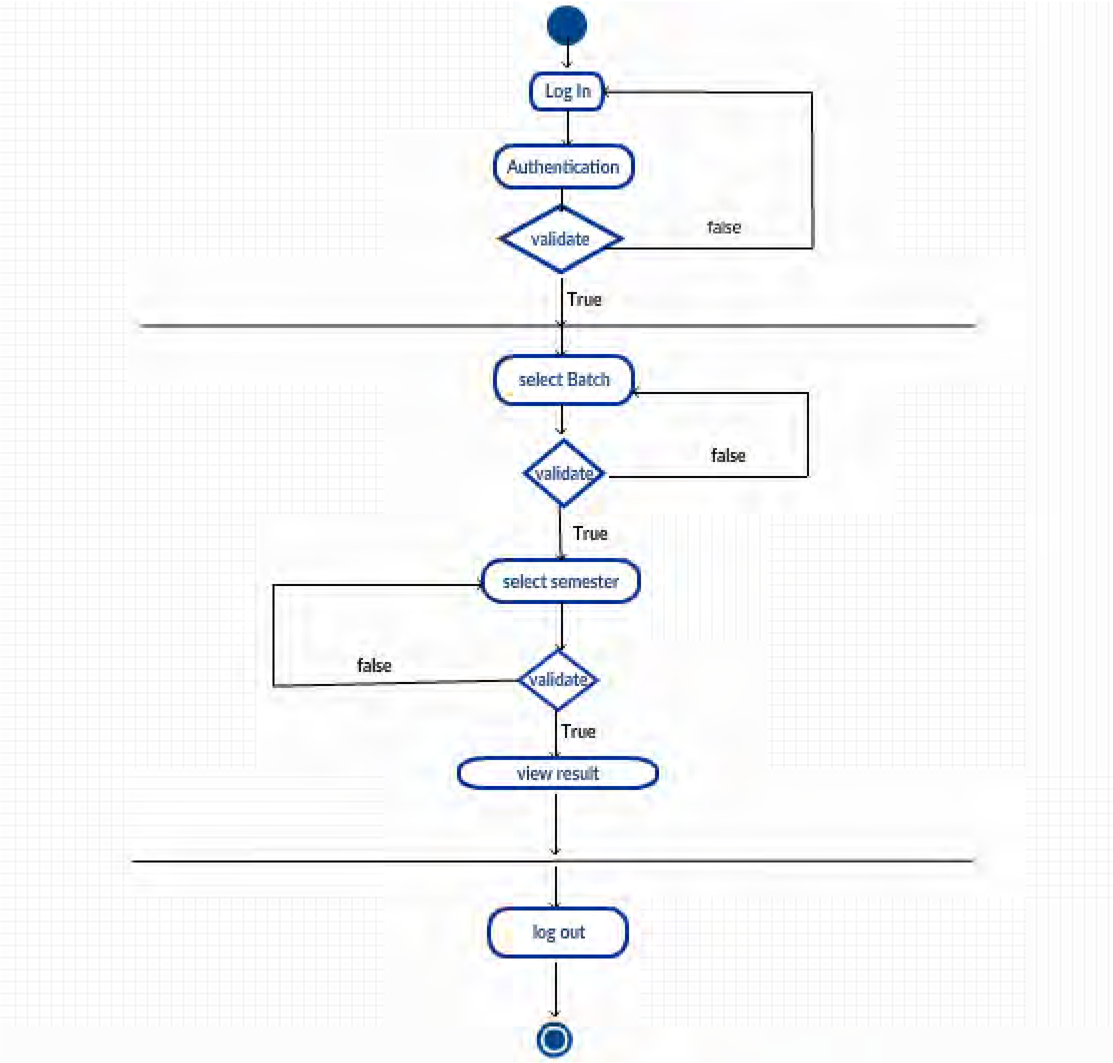


Figure 3. 11 View Student Performance Result

Chapter 4

SYSTEM IMPLEMENTATIONS

4.1 Introduction

This chapter is related to system implementation. The chapter mentions the tools, API's used to develop 'Performance Analyzer for CS Department'. Language selection is about the programming language and database selection is about the database that is used for storing and retrieving data. After the design phase, the implementation phase comes. In this phase we decide how to implement our design and which techniques to use.

4.2 Language Selection

PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages. PHP is a widely-used, free, and efficient alternative to competitors such as Microsoft's ASP.

4.3 Database Selection

XAMPP server is selected for data storage. XAMPP Server have a relational database management system. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet).

4.4 Software Used

1. Following software is used in developing student performance analyser Web-based Application:
2. Creating MySQL Database using XAMPP
3. Notepad++ for writing php, html, css code
4. Microsoft Visio for use case diagram, domain model, ERD diagram.
5. Project libre for software project management plan.
6. Adobe Photoshop for image manipulations (animations and provide special effect to images).

4.5 User Interfaces Design

The user interface (UI) is the part of the software most visible to the user and is one of the most important to get right. It includes prototypes of screen images of the system. There are two approaches for prototyping system design.

Low Fidelity

In low fidelity prototyping, screens images are drawn by hand.

High Fidelity

In high fidelity prototyping, screen images a designed with variety of screen design software like visual basic, pencil tool etc.

4.5.1 User Interface Screens

UI -1 Log in

User should login to use the system. He will enter email and password and click login.

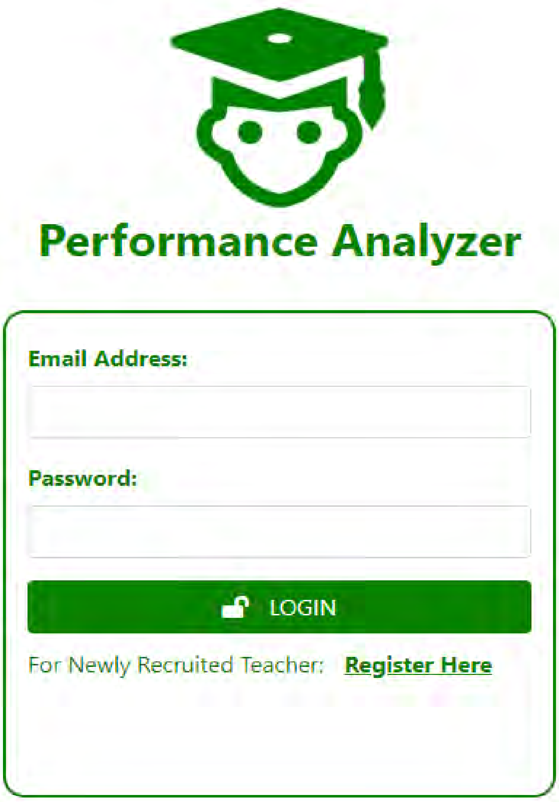
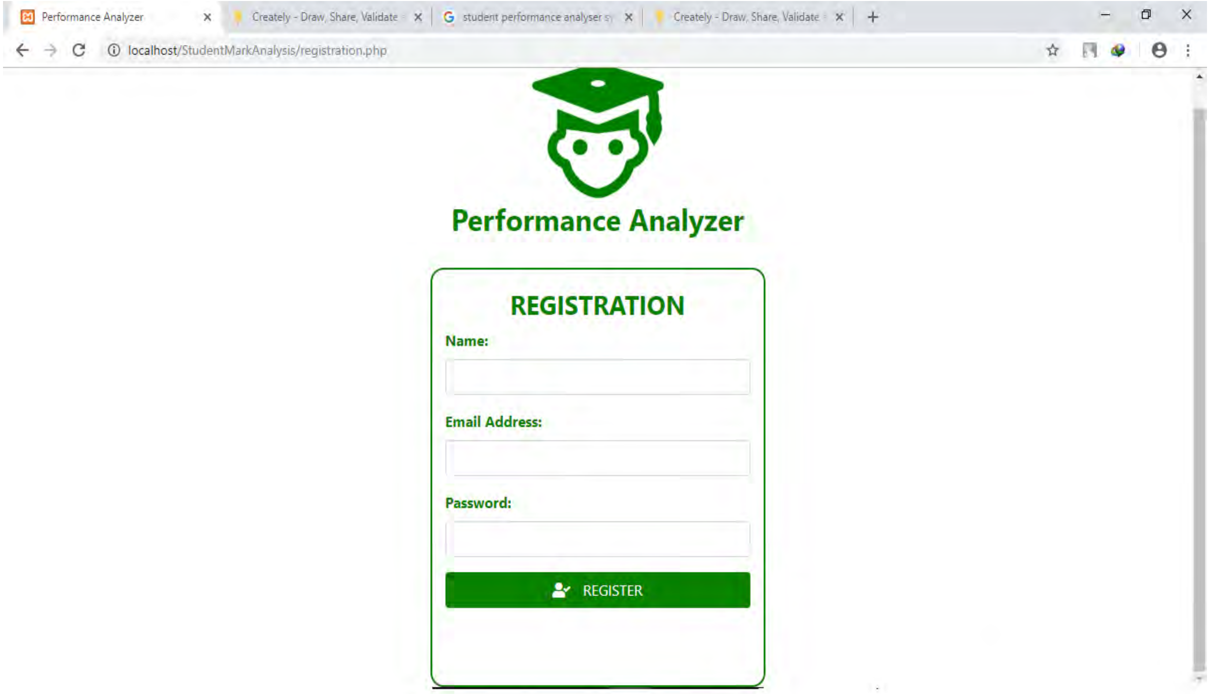


Figure 4. 1 Login interface

UI -2 Sign up

Faculty Member will sign up to use the system. He will enter email and password, name and click sign up.

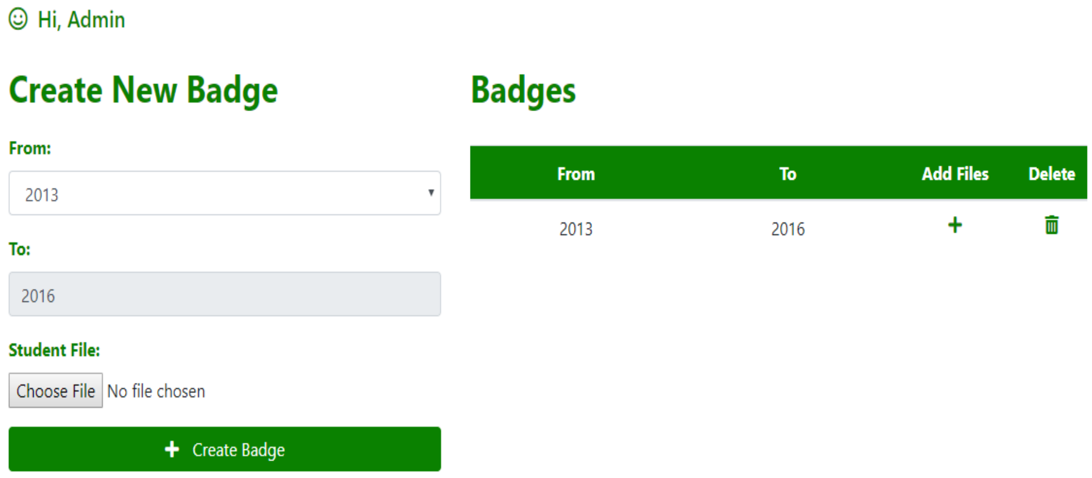


Performance Analyzer
Creately - Draw, Share, Validate
student performance analyser sy
Creately - Draw, Share, Validate
localhost/StudentMarkAnalysis/registration.php
Performance Analyzer
REGISTRATION
Name:
Email Address:
Password:
REGISTER

Figure 4. 2 Sign up screen

UI-3 Upload Student Registration file

Admin will log in to use the system. He will create batch and upload student registration information file.



From: 2013

To: 2016

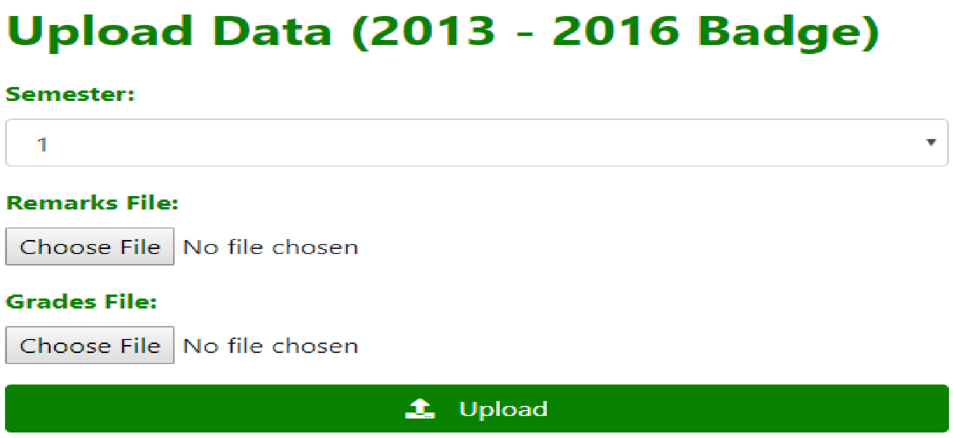
Student File: No file chosen

From	To	Add Files	Delete
2013	2016	+	🗑

Figure 4. 3 Upload student registration information file screen

UI-4 Upload Student courses marks and status screen

Admin will log in to use the system. He will create batch and upload student courses marks and their status information file



Semester: 1

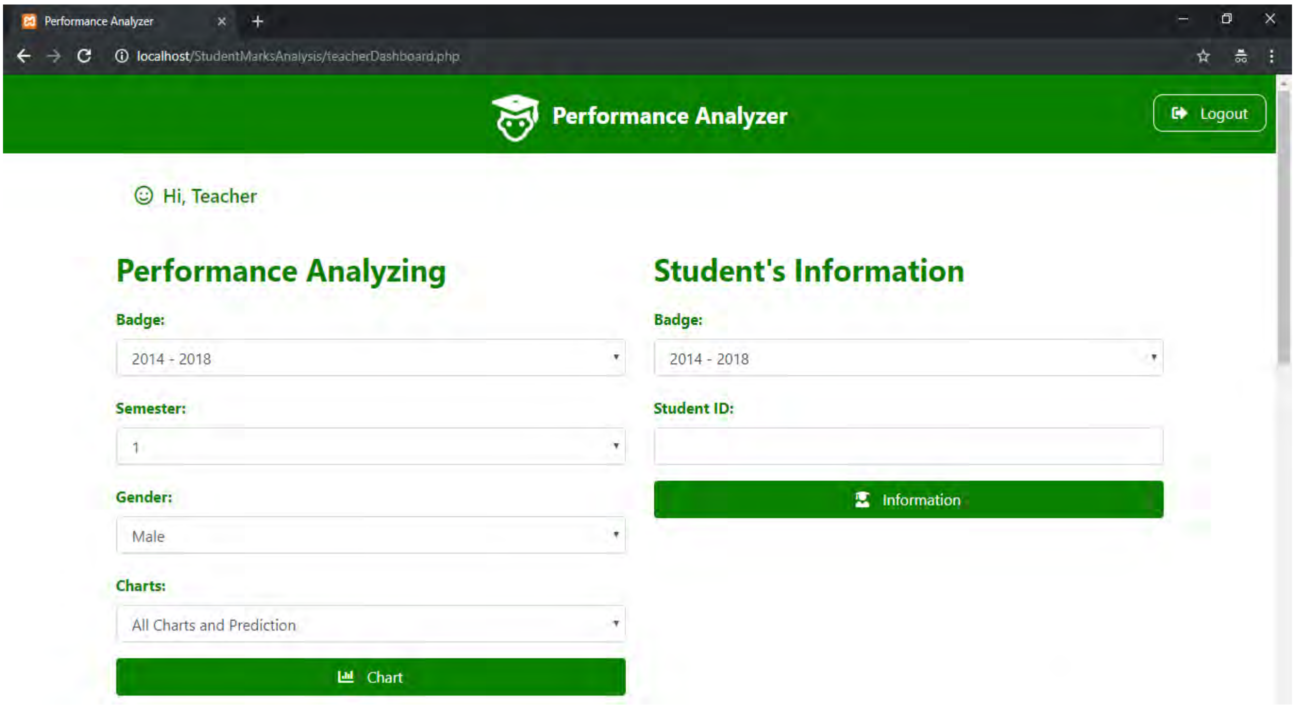
Remarks File: No file chosen

Grades File: No file chosen

Figure 4. 4 Student courses marks and status information uploading screen

UI-5 view performance result

Teacher will login firstly and then he will select batch, semester and gender, select all chart and prediction or course wise prediction and then click on chart. And he can also view the individual student result information by entering the registration no of student.



Hi, Teacher

Performance Analyzing

Badge: 2014 - 2018

Semester: 1

Gender: Male

Charts: All Charts and Prediction

Student's Information

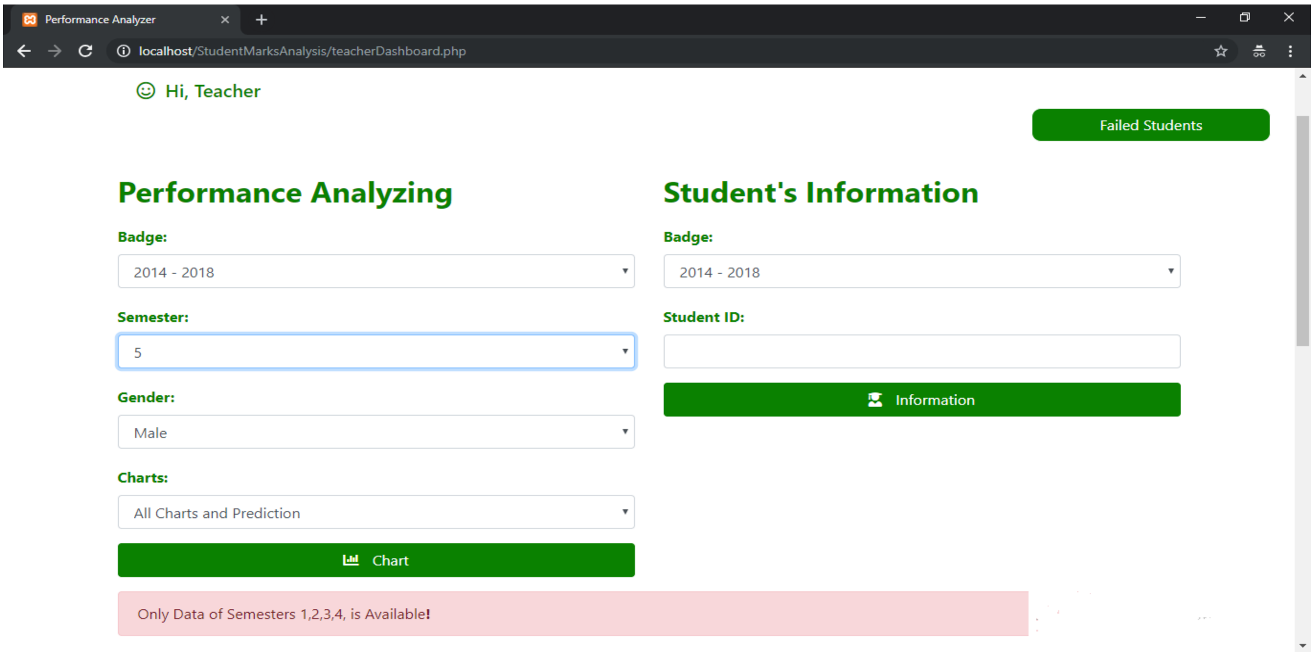
Badge: 2014 - 2018

Student ID:

Figure 4. 5 view performance result

UI-6 Search semester data

Faculty Member will login and check which batches data is available and he will also check semester's data against each batch.



Hi, Teacher

Performance Analyzing

Badge: 2014 - 2018

Semester: 5

Gender: Male

Charts: All Charts and Prediction

Student's Information

Badge: 2014 - 2018

Student ID:

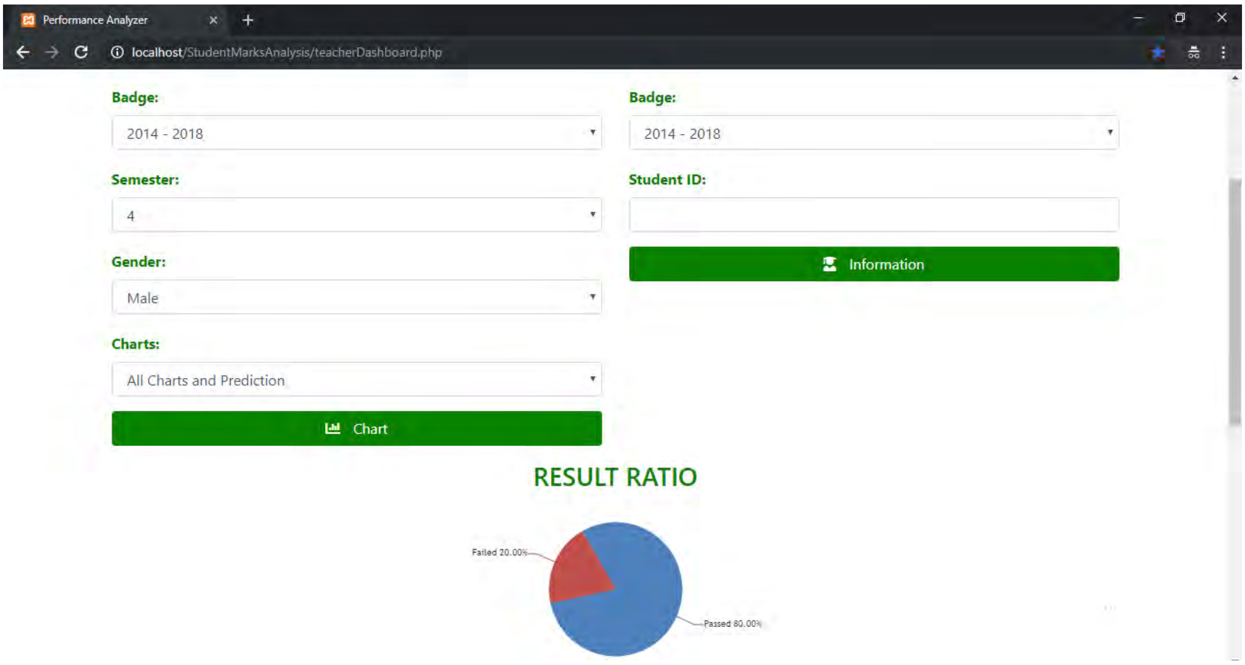
Only Data of Semesters 1,2,3,4, is Available!

Figure 4. 6 Available Batches and Semesters Screen

4.5.2 System Screens

SC-1 Performance result

Teacher will select a batch for which he want to make a performance analysis. And he will also select semester and gender of that batch.



Badge: 2014 - 2018

Semester: 4

Gender: Male

Charts: All Charts and Prediction

Badge: 2014 - 2018

Student ID:

Figure 4. 7 performance result screen

SC-2 Predicted result

When Teacher will perform analysis on the bases of the previous semester result then system will also predict next semester result with failure ratio.

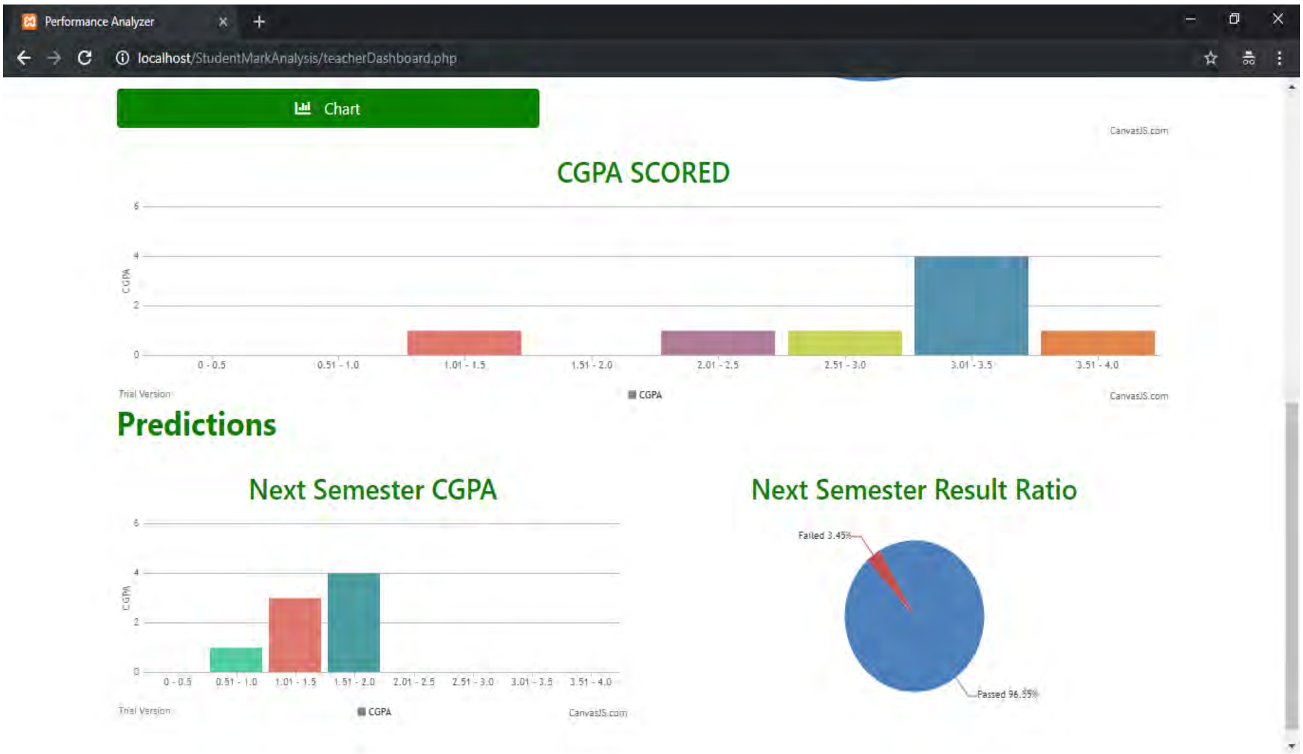


Figure 4. 8 predicted result

SC-3 Course wise Analyzed result

Course wise student passed and failure ratio screen is

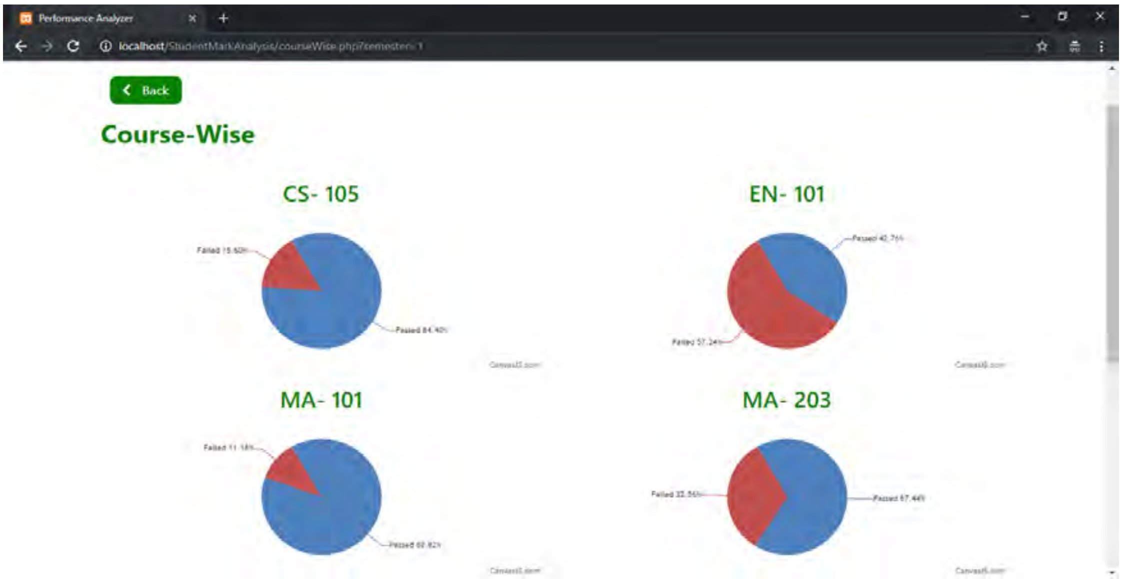
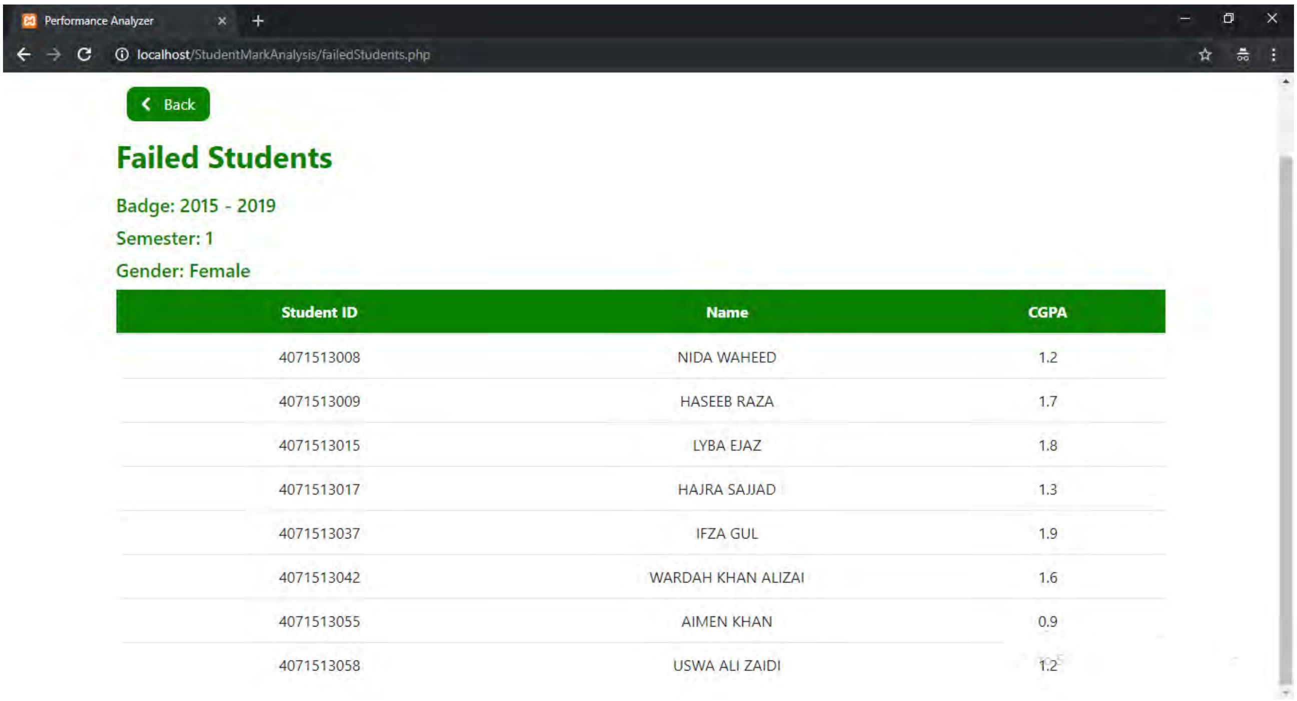


Figure 4. 9 Course wise pass fail ratio screen

SC-5 Failed Students Result Screen

When Teacher will perform analysis on the bases of the previous semester result then system will also show failed students name with registration no and CGPA. All those students will show to teacher whose CGPA is less than 2.0



Failed Students

Badge: 2015 - 2019

Semester: 1

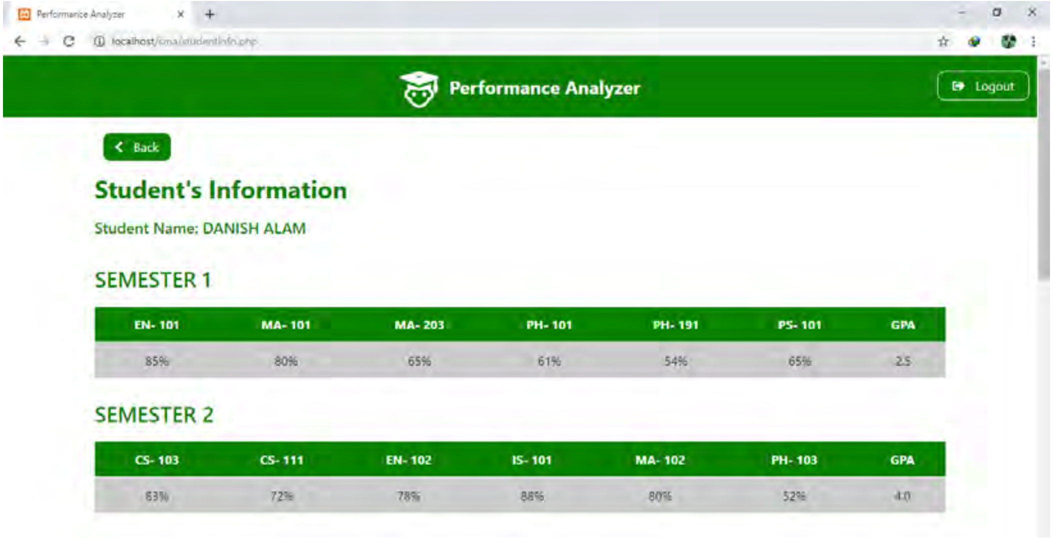
Gender: Female

Student ID	Name	CGPA
4071513008	NIDA WAHEED	1.2
4071513009	HASEEB RAZA	1.7
4071513015	LYBA EJAZ	1.8
4071513017	HAJRA SAJJAD	1.3
4071513037	IFZA GUL	1.9
4071513042	WARDAH KHAN ALIZAI	1.6
4071513055	AIMEN KHAN	0.9
4071513058	USWA ALI ZAIDI	1.2

Figure 4. 10 Failed Student Result Screen

SC-6 Students Result Screen

Teacher will login and he will enter student name and can view students semesters result



Student's Information

Student Name: DANISH ALAM

SEMESTER 1

EN- 101	MA- 101	MA- 203	PH- 101	PH- 191	PS- 101	GPA
85%	80%	65%	61%	54%	65%	2.5

SEMESTER 2

CS- 103	CS- 111	EN- 102	IS- 101	MA- 102	PH- 103	GPA
63%	72%	78%	88%	80%	52%	4.0

Figure 4. 11 Student Result Screen

Chapter 5

Software Test Documentation

5.1 Introduction

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. It is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements. Testing documentation involves the documentation of artifacts that should be developed before or during the testing of Software. This chapter specifying the test approach, test plan which include features to be tested and features not to be tested, testing tools and techniques and test cases that involve a set of steps that can be used while performing testing tasks.

5.1.1 Test Approach

Test approach is implementation of the test strategy for a specific project. It describe how the product risks of the stakeholders are mitigated at the test-level, which types of testing are to be performed, and which entry and exit criteria apply. The Test Approach describe the sources (project documentation) for requirements that will be used to drive the test design. This includes the modeling outputs (e.g., outputs from the process modeling activities such as use cases and task scenarios) from which test cases are developed. The choice of test approaches or test strategy is one of the most powerful factor in the success of the test effort and the accuracy of the test plans and estimates. The testing approach for (Performance Analyzer for CS Department) is Unit Testing. Unit test is the smallest testable part of an application like functions, classes, procedures, interfaces. Unit testing is a method by which individual units of source code are tested to determine if they are fit for use. So, we construct Performance Analyzer for CS Department in step by step manner, that’s why unit testing is suitable to test because the main focus of unit testing to segregate each part of the program and test that the individual parts are working correctly. This means that for any function or procedure when a set of inputs are given then it should return the proper values. It should handle the failures gracefully during the course of execution when any invalid input is given. White Box Testing method is used for executing the unit test.

5.2 Test Plan

A test plan describes how testing will be accomplished on a software product, together with the resources and schedule needed. It is the most important activity to ensure that there is initially a list of tasks and milestones in a baseline plan to track the progress of the project. It also defines the size of the test effort.

5.2.1 Feature to be Test

All the major functions (functional requirements) of (Performance Analyzer for CS Department) are to be tested. Following are the list of features to be tested:

- Login
- Signup
- Analyze Performance
- Upload Result

5.2.2 Features not to be Test

All the features are tested to increase reliability of Performance Analyzer for CS Department .So all features are to be tested.

5.2.3 Testing Tools and Environment

Test Environment consists of elements that support test execution with software, hardware and network configured. Manual test environment is used in Performance Analyzer for CS Department. Manual testing is crucial for testing software applications more thoroughly. ApTest Manager is a web-based tool for managing manual testing. ApTest Manager solves challenges by recording and tracking the tests used in your software and hardware testing projects and the results of running them. Testing information becomes instantly available to everyone - always accurate, always current. It also allows you complete access to monitor the progress of projects during every phase from start to finish.

5.3 Test Cases

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application. It is good practice to make your test cases atomic.

5.3.1 Login Test Case

Test Case ID	TC-01
Actors	Faculty Member/Admin
Purpose	Login to account

Input	**Frequency**	**Expected Output**	**Observed Output**	**Verdict**
User enters his correct email and password.	2	User successfully login.	User successfully login.	Pass
User enters his incorrect Id and password.	1	User does not exist.	User does not exist.	Pass
User submits login form with empty Id and empty password.	4	Empty fields are not allowed.	Empty fields are not allowed.	Pass

Table 5. 1 Login Test Case

5.3.2 Sign up Test Case

Test Case ID	TC-02
Actors	Non-Registered Faculty Member
Purpose	Register Faculty Member

Input	**Frequency**	**Expected Output**	**Observed Output**	**Verdict**
Faculty Member enters data in all the required fields with correct information.	2	Faculty Member successfully registered in system.	Faculty Member successfully registered in system.	Pass
Faculty Member enters incorrect information like wrong mail syntax (sabahat@gmail).	2	Enter valid email for proceed.	Enter valid email for proceed.	Pass
Faculty Member submits correct information but user of that name already exist.	2	Faculty Member name already exist in our system.	Faculty Member name already exist in our system.	Pass

Table 5. 2 Sign Up Test Case

5.3.3 Upload Result Test Case

Test Case ID	TC-003
Actors	Admin
Purpose	Upload student information.

Input	**Frequency**	**Expected Output**	**Observed Output**	**Verdict**
Admin select badge and select Registration information csv file	10	Data successfully uploaded	Data successfully uploaded	pass
Admin enters invalid csv file	2	Data will not upload	Data will not upload	pass
Admin create badge without registration information CSV fie	1	Empty input field	Empty input field	pass
Admin select semester, courses data and result information	5	Data uploaded successfully	Data uploaded successfully	Pass
Admin select semester and add wrong csv files	2	Syntax of file does not match	Syntax of file does not match	pass

Table 5. 3 Upload Result Test Case

5.3.4 View Performance Result Test Case

Test Case ID	TC-004
Actors	Faculty Member
Purpose	Analyse student performance.

Input	**Frequency**	**Expected Output**	**Observed Output**	**Verdict**
Faculty Member selects Batch, semester, gender and prediction criteria	5	Successfully analysed student performance	Successfully analysed student performance	pass
Faculty Member check performance without selecting batch and semester	2	Student performance graph will not show.	Student performance graph will not show	pass

Table 5. 4 View Performance Result

Chapter 6

Conclusions and Future Enhancements

This document describes the project conclusions and future enhancements i.e. what type of new features can be added with time.

6.1 Conclusion

The purpose behind this project is to develop a web base application for CS faculty of QAU. The proposed system will display results of student performance on a single click action by the user, thus inducing automation and reducing efforts of staff in analyzing student performance manually. The System is completely developed and testing is performed from different aspects to develop confidence on system.

6.2 Future Enhancements

It is possible to enhance the application in future. Here are some features that can be added to application:

- Individual Student performance can be predict on the basis of previous result.