SOFTWARE ENGINEERING AND PROJECT MANAGEMENT

Submitted by: -

Thulluri Amrutha - RA2011003020017
Nuthakki Lakshmi Priya - RA2011003020001

EX NO: 1 **DATE:**	***Identify the software project, create business case, arrive at a problem statement***

Introduction: -

This document lays out a project plan for the development of "Examination and Result Management System".

The goal of this project is to provide a mobile application for all the students and teachers and Institute's administrators, to build an online system to manage all the records related to examination and result of students and teachers to ease the service of examination and result management.

1.1. Problem Statement: -

In our current examination conduction and result processing system, there is too much manual and paperwork used for conduction of exam and computation of result.

It becomes very difficult to manage record of all students manually which results in heap of files. So, there is always a risk of loss of information due to various reasons.

Also, there is imbalance between the manpower availability and magnitude of the examination work which results in inaccuracy and delay in declaration of result.

Objective:

Our software is designed to efficiently manage all the records of the students, automatically calculate the percentage and grade point of the students.

With the use of our software, the span required for declaration of result and use of man power will be reduced to a great extent.

Also, the student gets a message of the result to their official email or registered mobile number to overcome server issues

Scope of the product:

The "Examination and Result Management System" is a mobile application, which helps students and teachers to find all the up-to-date information related to examination and result.

The Examination and Result Management System is composed of two main components: a client-side application which will run on Android handsets, and a server-side application which will support and interact with client-side queries. The server-side verifies logins as Student, Teacher or Administrator and manages user information. The data will be held in an Access database on the server. The administrator logins to upload information about creating/updating a record of a student or a teacher, or handle the complaints and queries put forward by students/teachers.

1.2. ONE PAGE BUSINESS CASE TEMPLATE:

The Project: -

- The main objective of the project on Examination and Result Management System is to manage the details of student result.
- It manages all the information about student subject results.
- The purpose of the project is to build an application program to reduce the manual work for managing the student result.
- It tracks all the details about the student results.

The History: -

The old manual system was suffering from a series of drawbacks. since whole of the system Was to be maintained with hands the process of keeping, maintaining, and retrieving the information was very tedious and length in the records were never used to be in a systematic order there used to be lots of difficulties in associating any particular transaction with a particular context Li any information was to be found it was required to go through the internet registers, documents there would never exist anything like report generation. where would always be unnecessary consumption of time While Entering records and retrieving records. One more problem was that it Was very difficult to find errors While entering the records. Once the records were entered it was very update these records

Constraints: -

The Software Requirements Specification is produced at the culmination of the Analysis task. The function and performance allocated to software as part of system engineering are refined by establishing a complete information description, a detailed functional and behavioural description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

Approach: -

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made.

The development of this project will require front end and back-end programming knowledge. Thus, it is important to equip with the same.

EX NO: 2 **DATE:**	***Stakeholder and user description, identify the appropriate process models, comparative study with agile model***

2.1. Identifying Stakeholders: -

1. USER:

The persons who will use the project's service are known as users.

In this project, users will be

a) Students
b) Teachers
c) Administrator

Each of these types of users has different the system so each of them has their own requirements.
The students can only use the application to view their profile, result, datasheet, syllabus, and Admit Card. This means that the user is expected to be Internet literate. The user interface will be as intuitive as possible. Thus, technical expertise and Android experience should not be an issue.

2. SPONSOR:

Sponsor is the person or the organization that provides financial support for the project.
As of now, this project is self-sponsored.
Once we upgrade, we may require financial interference of a management.

3. PROGRAM MANAGER:

A Program Manager articulates a program's strategy and objectives and assesses how it will impact.

4. PROJECT MANAGEMENT OFFICE (PMO):

A project management office (PMO) is a team or department that sets and maintains standards for project management throughout an organization.

5. PROJECT MANAGER:

Project Managers (PMs) are responsible for planning, organizing, and directing the completion of specific projects for an organization while ensuring these projects are on time, on budget, and within scope.

6. PROJECT TEAM:

A project team is comprised of the project manager, project management team and the other members who carry out work

Benefits: -

A. Economic Feasibility

This is a very important aspect to be considered while developing a project. We decided the technology based on minimum possible cost factor.
All hardware and software cost has to be borne by the organization.
Overall, we have estimated that the benefits the organization is going to receive from the proposed system will surely overcome the initial costs and the later on running cost for system.

B. Technical Feasibility

This included the study of function, performance and constraints that may affect the ability to achieve an acceptable system. For this feasibility study, we studied complete functionality to be provided in the system, as described in the System Requirement Specification (SRS) and checked if everything was possible using different type of frontend and backend platforms.

C. Operational Feasibility

No doubt the proposed system is fully GUI based that is very user friendly and all inputs to be taken all self-explanatory even to a layman. Besides, a proper training has been conducted to let know the essence of the system to the users so that they feel comfortable with new system. As far our study is concerned the clients are comfortable and happy as the system has cut down their loads and doing.

2.2. IDENTIFY THE APPROPRIATE PROCESS MODELS: -

Process Model

In our software, we are following prototyping Model (Evolutionary) as initially the requirements are not clear.

The requirements are added in the prototype as and when they are understood.

Software reviews are applied at various points during software engineering and serve to uncover the errors and defects that can be removed.

2.3. COMPARISION BETWEEN WATERFALL AND AGILE MODEL: -

WHY AGILE MODEL IS BETTER THAN WATERFALL MODEL?

- The Agile Model is based on iterative development and hence it divides the entire project into smaller parts which reduces the risk factor which is not the case in waterfall model.

- The Waterfall model cannot accept the changes in requirements but in agile model it is easy to change the system requirements.

- In agile model, the entire project is divided into smaller parts which helps to minimize the project risk and to reduce the overall project delivery time requirements.

- In waterfall model since risk factor is high, it is not suitable for complex projects.

- In waterfall model the testing is done in later stage it does not allow identifying the challenges and risks in the earlier phase, so the risk reduction strategy is difficult to prepare, which is not the case in agile model.

- In waterfall model, it follows a sequential approach whereas in agile model it explains the process in order of incremental approach.

- In agile it performs the testing concurrently with software development whereas in waterfall model the testing comes after the build phase only.

- In agile model the distance between the customer and developer is in short whereas in waterfall model it is long.

- In agile there can be done any change in the project but in waterfall model there is no changes throughout the project work.

EX NO: 3 DATE:	***Identify the Requirements, System Requirements, Functional Requirements, Non-functional Requirements***

3.1. REQUIREMENTS:

Requirements are defined during the early stages of the system development as a specification of what should be implemented. A collection of requirements is a requirements document. They may be user level facility description, detailed specification of system behaviour, general system property, a specific constraint on the system or information on how to carry on computation. The three types of requirements are explained below.

3.1.1. SYSTEM REQUIREMENTS:

System requirements are the configuration that a system must have in order for a hardware or software application to run smoothly and efficiently. Failure to meet these requirements can result in installation problems or performance problems. The former may prevent a device or application from getting installed, whereas the latter may cause a product to malfunction or perform below expectation or even to hang or crash.

1. ***Hardware requirements: -***
   - Processor: Intel Pentium 4.0
   - Ram: 2GB
   - Hard disk: 500GB
2. ***Software requirements: -***
   - Operating system: Windows XP.
   - Front-End: HTML, PHP
   - Database: MYSQL
   - Model Design: Rational Rose

3.1.2. FUNCTIONAL REQUIREMENTS: -

1) Subject Information Management

The system will maintain information about various subjects being offered during different semesters of the course. The following information would be maintained for each subject. Subject code, Subject Name, Subject Type (Core / Elective /Lab1 / Lab2 / Mini Project) Semester, Credits.

2) Student Information Management

System will maintain information about various students enrolled in the different courses in different years. The following information would be maintained for each student: Student enrolment No, Student Name, Year of enrolment. The system will allow creation/modification/deletion of new/ existing students and also have the ability to list all the students enrolled in a particular year.

3) Student's Subject Information Management

The system will maintain information about different Elective subjects of different enrolment years in different semesters of students. The following information would be maintained: Student enrolment no, Semester, Student's choices for a particular semester.

4) Marks Information Management

The system will maintain information about marks obtained by various students of different enrolment years in different semesters. The following information would be maintained: Student enrolment no, semester, subject code, internal marks, external marks, total marks and credits.

5) Mark Sheet Generation

The system will generate mark-sheet for every student in different semesters.

6) Report Generation

- Student List Reports
- For each year a report will be generated containing the list of students enrolled in that batch year.
- Student Subject List Report
- For each batch year a report will be generated containing the list of students
- and Elective subject in the semester.
- Semester-wise mark lists
- Rank-wise List Report

7) User Account Management

The system will maintain information about various users who will be able to access the system. The following information would be maintained. Username, User ID, Password and Role.

3.1.4 NON-FUNCTIONAL REQUIREMENTS:

- Readability
- Availability
- Maintainability
- Security
- User Friendly
- Performance
- Efficiency
- Privacy

EX NO: 4 **DATE:**	***Prepare project plan based on scope, find job roles and responsibilities, calculate project effort based on resources***

4.1. PROJECT PLAN: -

- ***PROJECT NAME:***

  "Examination Result Management System"

- ***PROJECT MEMBERS:***

  Our project consists of two members:

  1) N. Lakshmi Priya – RA2011003020001
  2) T. Amrutha – RA2011003020017

- ***MODULES:***

  Student: Student Result Management System Project in PHP
  Admin: Student Result Management System Project in PHP

  Admin Features

  - Admin Dashboard
  - Admin can add/update/ Class
  - Admin can add/update/ Subjects
  - Admin can add/update/ Active/Inactive Subject combination with class
  - Admin can register new student and also edit info of the student
  - Admin can declare/ edit result of a student.
  - Admin can change own password

  Students Features:

  - Student can search their result using valid rollid.
  - Student can download the result in the PDF format.

Here, Student can check their results by entering Roll id. Admin can create & manage Classes, subjects. Add & Manage students and Declare Results. This project is done in PHP. It's easy to operate and understand by users. The design is pretty simple, and user won't find it difficult to understand, use and navigate.

- ***SCHEDULING:***

Task	***Start date***	***End date***
Business case development	27th December 2019	30th December 2019
Identify Stakeholders, Process Models and Required Modules	30th December 2019	6th January 2020
Identify requirements	31st December 2019	6th January 2020
Setting cost estimates and budget	6th January 2020	11th January 2020
Coding	9th January 2020	18th February 2020
UML Diagrams	19th February 2020	10th March 2020
Final Revisions	10th March 2020	14th March 2020

4.2. JOB ROLES AND RESPONSIBILITIES: -

MEMBERS	ROLE AND RESPONSIBILITIES
N. Lakshmi Priya **RA2011003020001**	• **Team leader**: Has the responsibility of coordinating the team, checking in for updates & guiding the team. • **Web Developer**: Has the responsibility of designing the website and interfacing with the server. • **Tester:** Has the responsibility to check if the actual result is matching with expected result.
T. Amrutha **RA20003020017**	• **Team Member**: Has the responsibility of contributing to the documentation. • **Developer**: Has the responsibility of coding modules Login and Update. • **Technical Lead**: is responsible for overall planning, execution, and success of overall complex software solutions to meet customer's needs.

4.3. PROJECT EFFORT BASED ON RESOURCES: -

COCOMO2 (Constructive Cost Model 2) is an algorithmic cost estimation technique proposed by Boehm, which works in a bottom-up manner.

• It is designed to provide some mathematical equations to estimate software projects.

• These mathematical equations are based on historical data and use project size in the form of KLOC.

The COCOMO model uses a multivariable size estimation model for effort estimation.

- **OBJECT POINT = ∑^3 i=1 ∑^3 i=j cij*wij**

	SIMPLE	**MEDIUM**	**COMPLEX**
SCREENS	1	3	2
REPORT	1	3	1
3GL	0	0	1

(1*1+3*2+2*3) +(1*2+3*5+1*8) +(1*10) = 13+25+10
= 48

- **NOP** = **Object Point*(1-% reuse/100)**
  = 48* (1-0)
  = 48

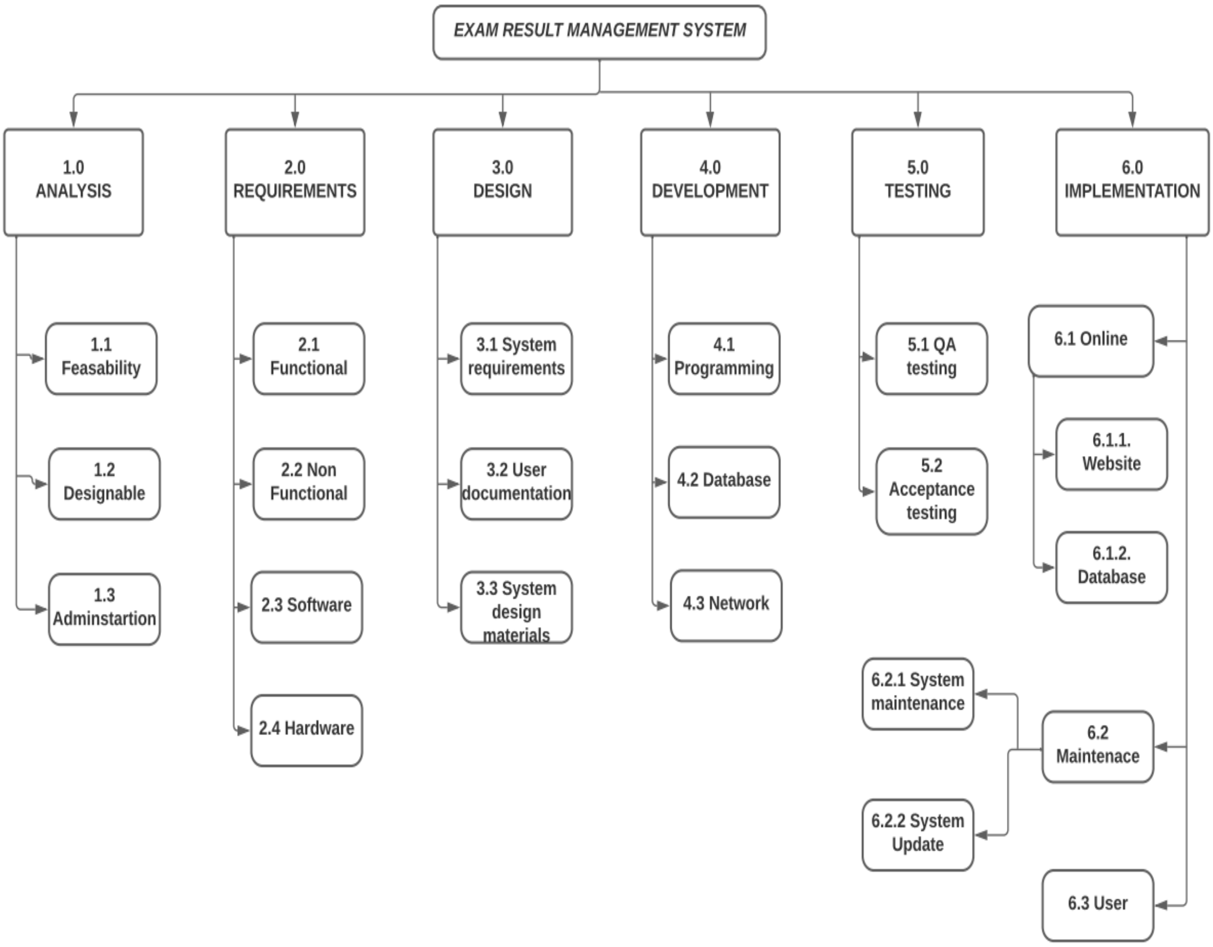
- **EFFORTS = NOP/ PROD**
  = 48/13
  = 3.7

NOP = New Object Point
PROD = Productivity
We have assumed nominal developer experience

EX NO: 5 **DATE:**	***Prepare the work breakdown structure based on timelines, Risk identification plan***

5.1. WORK BREAKDOWN STRUCTURE: -

A Work Breakdown structure is a deliverable-oriented hierarchical decomposition of the work to be executed by the project team to accomplish the project objectives and create the required deliverables. A WBS is the cornerstone of effective project planning, execution, controlling, monitoring, and reporting. All the work contained within WBS is to be identified, estimated, scheduled, and budgeted.



5.2. RISK MANAGEMENT: -

DESCRIPTION:

In the modern world, risk management refers to the practice of identifying potential risks in advance by analysing them and taking precautionary steps to curb the risk**.**

- Risk management is the identification, evaluation, and prioritization of risks, controlling the probability or impact of unfortunate events.
- When all risks have been identified, they will then be evaluated to determine their probability of occurrence.
- Plans will be made to avoid each risk, to track each risk to determine if it is more or less likely to occur, and to plan for those risks should they occur.
- The quicker the risks can be identified and avoided, the smaller the chances of having to face those particular risks consequence.

RISKS TO BE HANDLED:

- Computer crash
- Late delivery of results
- End users Resist System
- Low website speed.
- Improper internet connection.
- Maintaining Database.

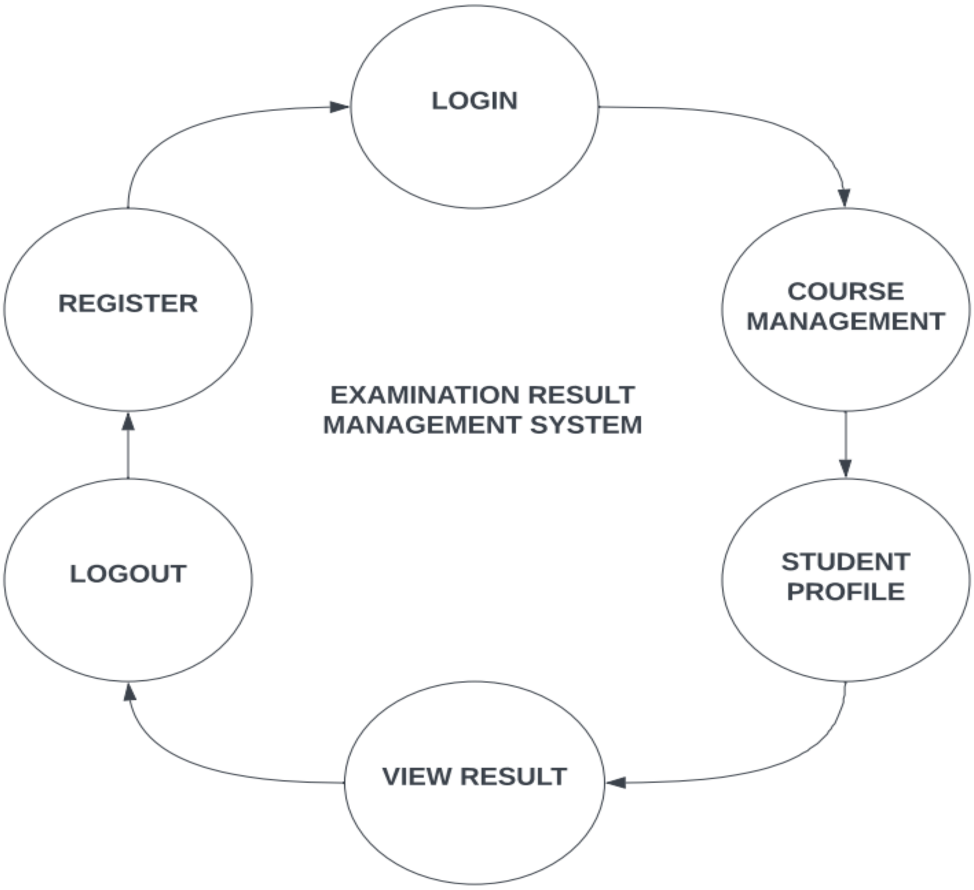
MANAGING RISKS:

- Performing periodic maintenance of the server.
- Using of Captcha and other security protection things to protect from bot attack.
- The bugs must be removed, and the code must pass as many test cases as possible.

EX NO: 6 **DATE:**	***Design a System Architecture, Use Case Diagram, ER Diagram, DFD Diagram, Class Diagram, Collaboration Diagram***

6.1. SYSTEM ARCHITECTURE: -

Here we have used the basic software front end design model in order to represent the system architecture of our software model.



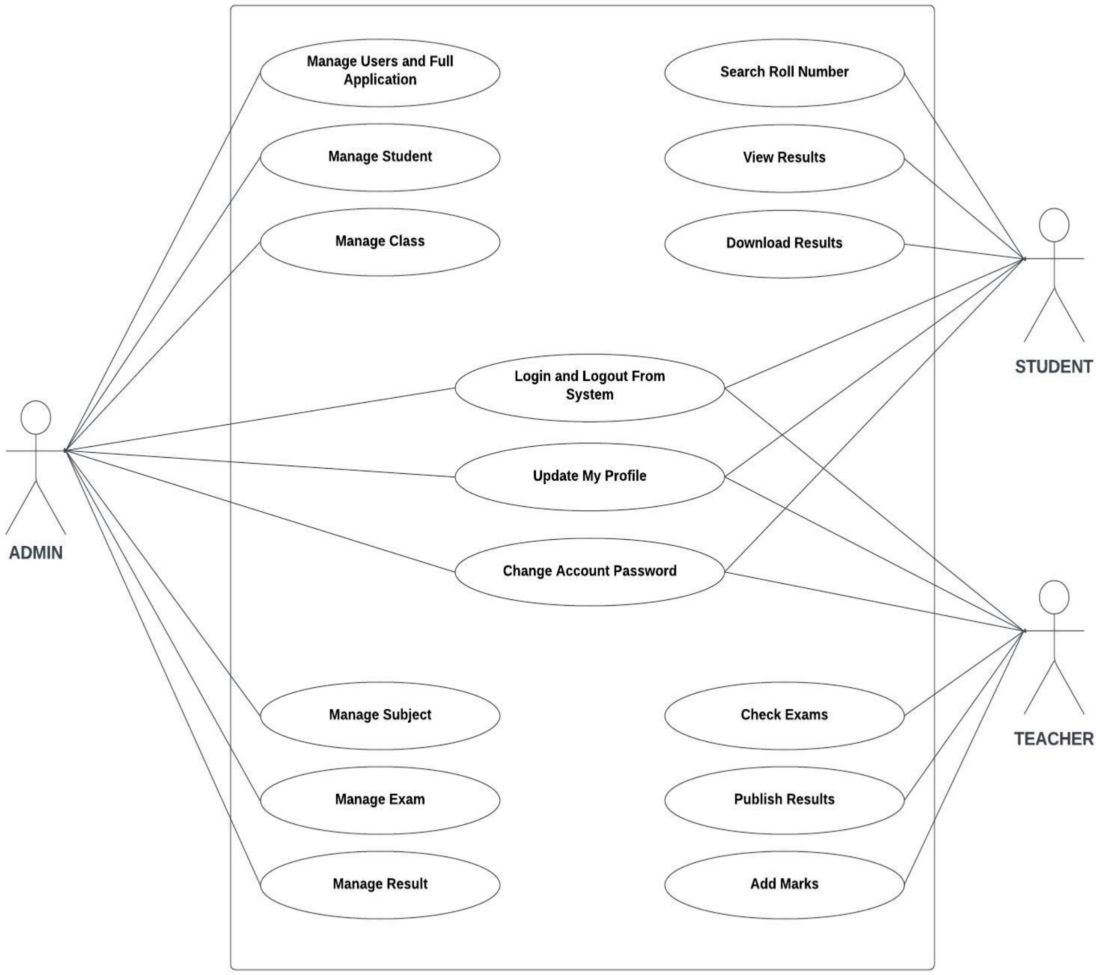
The above is a simple form of system design diagram which uses front end design. This shows a loop of functions that need to be executed when this project is implemented. This is a chain of operations through which this project is implemented.

6.2. MODELING USECASE DIAGRAM AND SCENARIOS: -

6.2.1. USE CASE DIAGRAM DESCRIPTION:

- There are 3 actors:
  - Admin
  - Student
  - Teacher
- The login and logout, update profile, changing password use cases have relations with admin and both teacher and the student.
- The Manage users and fill application, manage student, manage class use cases have relations with Admin.
- The manage subject, manage exam, manage result use cases have relation with the Admin.
- The Search roll number, view results, download results use cases have relation with student

6.2.2. USE CASE DIAGRAM:



6.3. MODELING OF ER DIAGRAM: -

6.3.1. ER DIAGRAM DESCRIPTION:

- An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.
- ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education, and research.

USES OF ER DIAGRAM:

- Database design
- Database troubleshooting
- Business information systems
- Business process re-engineering (BPR)
- Education
- Research

COMPONENTS OF ER DIAGRAM:

ER Diagrams are composed of entities, relationships (Cardinality) and attributes. They also depict cardinality, which defines relationships in terms of numbers.

1] ENTITY: A definable thing—such as a person, object, concept or event—that can have data stored in it.

2] ATTRIBUTES: A property or characteristic of an entity.

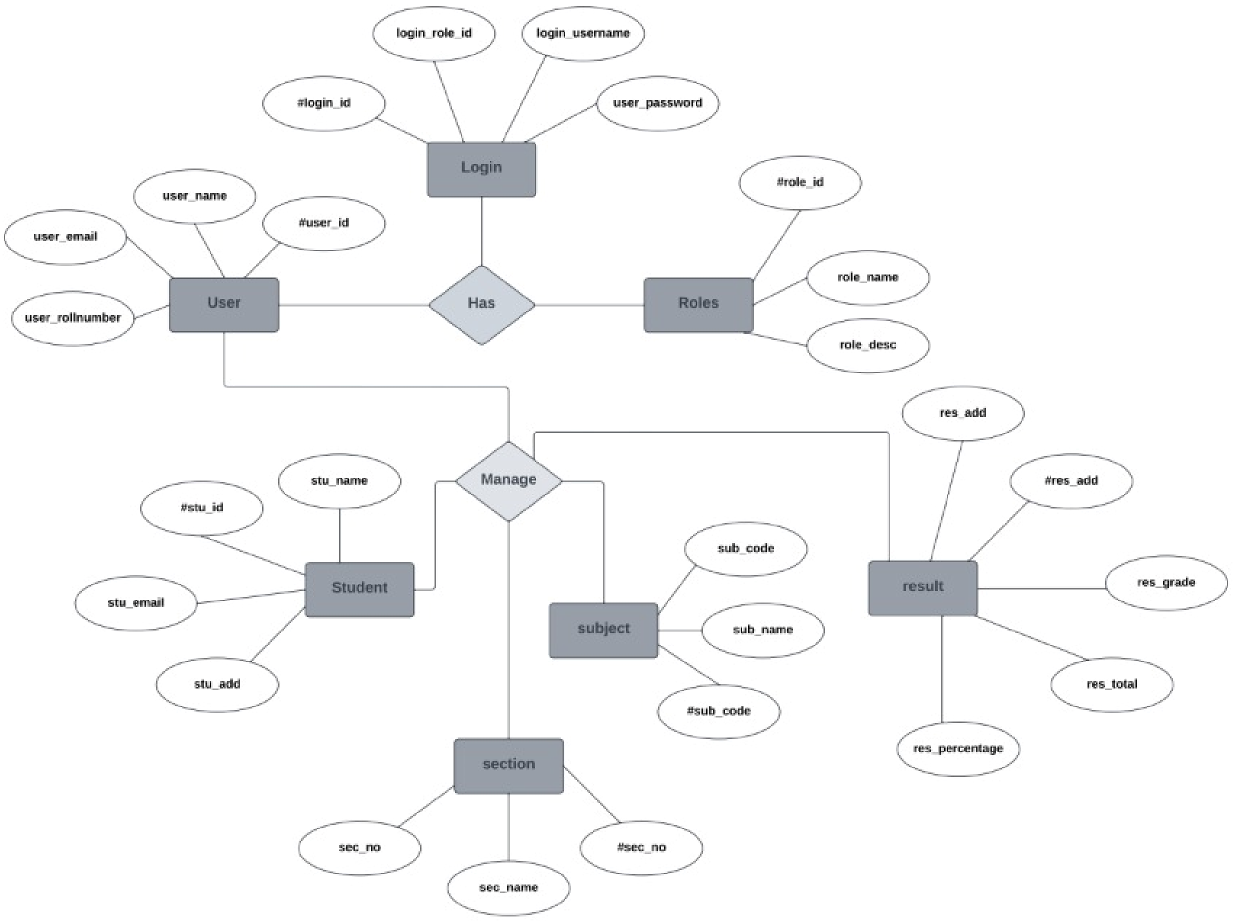
3] KEYS:

PRIMARY KEY(PK): It is unique, cannot be repeated and never null.
FOREIGN KEY(FK): It is not unique and can be repeated.

4] CARDINALITY: Defines the numerical attributes of the relationship between two entities.

- One to one
- Many to one
- One and only
- Zero to one
- One or many
- Zero or many

6.3.2. ER DIAGRAM:



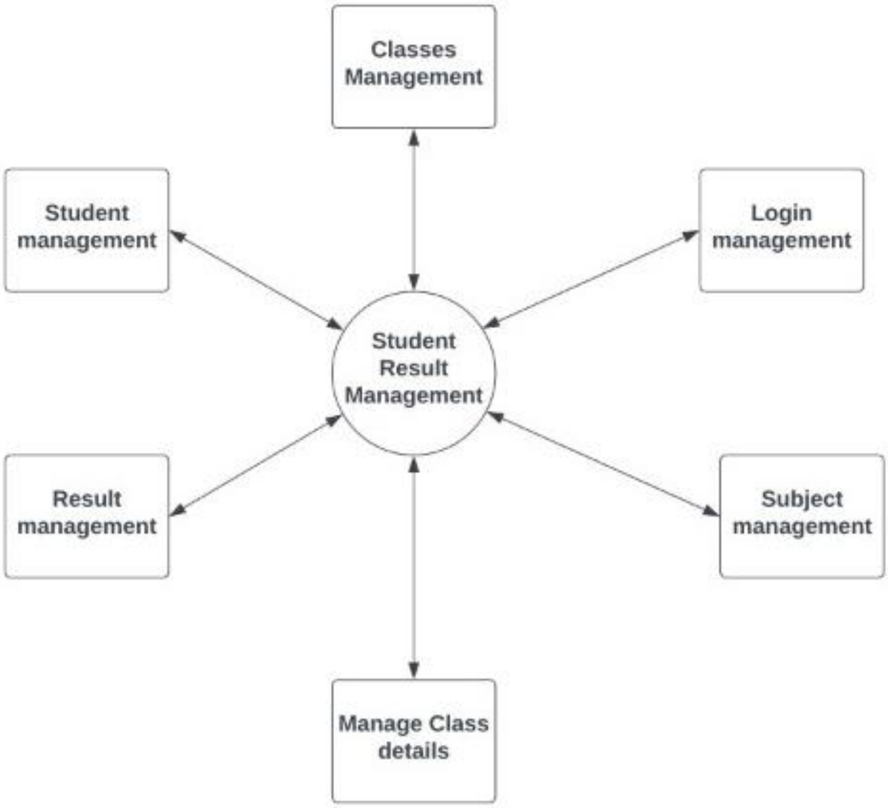
6.4. MODELING OF DATA FLOW DIAGRAM: -

6.4.1. DATA FLOW DIAGRAM DESCRIPTION:

Result Management System Data flow diagram is often used as a preliminary step to create an overview of the Result Management without going into great detail, which can later be elaborated it normally consists of overall application dataflow and processes of the Result Management process. It contains all of the user flow and their entities such all the flow of Student, Exam, Class, Subject, Result, Teacher, Semester. All of the below diagrams has been used for the visualization of data processing and structured design of the Result Management process and working flow

Main entities and output of First Level DFD (1st Level DFD):

- Processing Student records and generate report of all Student
- Processing login records and generate report of login credentials
- Processing Class records and generate report of all Class
- Processing Subject records and generate report of all Subject
- Processing Result records and generate report of all Result
- Processing Teacher records and generate report of all Teacher
- Processing Semester, records and generate report of all Semester



6.5. MODELING OF CLASS DIAGRAM: -

6.5.1. CLASS DIAGRAM DESCRIPTION:

Class diagrams are one of the most useful types of diagrams in UML as they clearly map out the structure of a particular system by modelling its classes, attributes, operations, and relationships b/w objects.

COMPONENTS OF CLASS DIAGRAM:

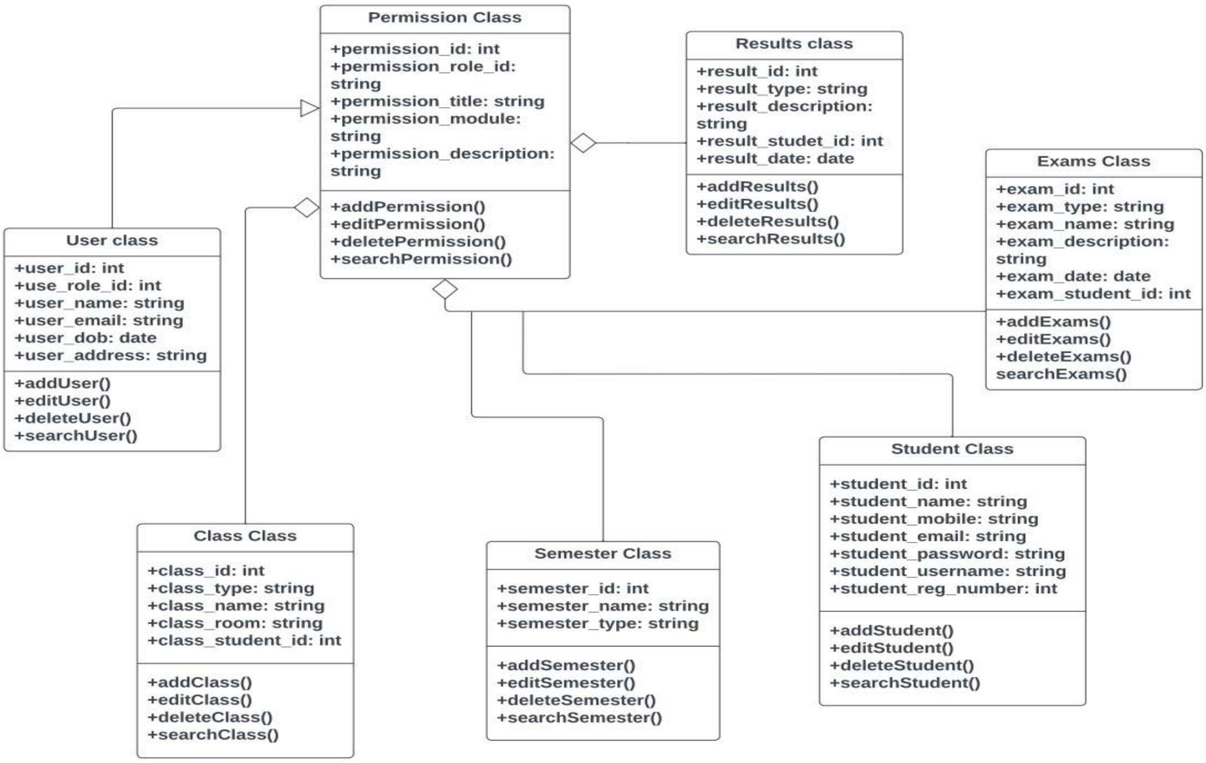
The standard class diagram is composed of three sections:

- UPPER SECTION: Contains the name of the class. This section is always required, whether you are talking about the classifier or an object.

- MIDDLE SECTION: Contains the attributes of the class. Use this section to describe the qualities of the class. This is only required when describing a specific instance of a class.

- BOTTOM SECTION: Includes class operations (methods). Displayed in list format, each operation takes up its own line. The operations describe how a class interacts with data.

6.5.2. CLASS DIAGRAM:



6.6 MODELING OF COLLABORATION DIAGRAM: -

6.6.1. COLLABORATION DIAGRAM DESCRIPTION:

Communication diagrams, formerly known as collaboration diagrams, are almost identical to sequence diagrams in UML, but they focus more on the relationships of objects—how they associate and connect through messages in a sequence rather than interactions.

COMPONENTS OF COMMUNICATION DIAGRAM:

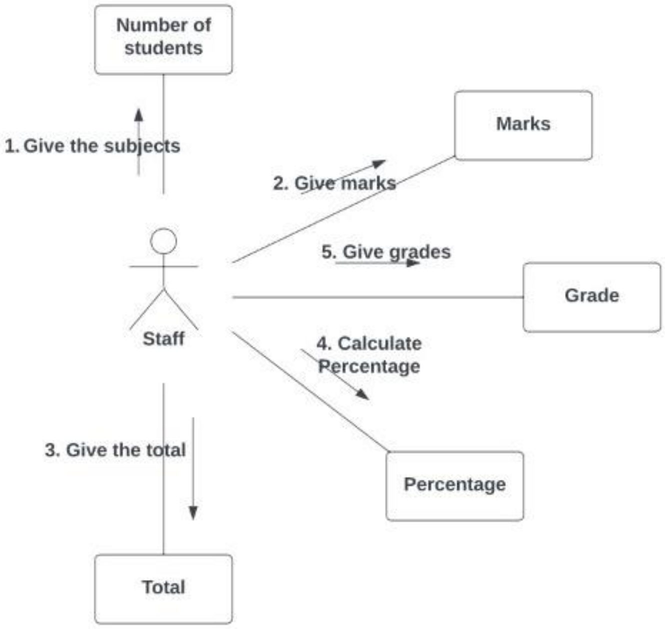
1]OBJECTS:
Objects can be classed as either a supplier or a client. Suppliers call the function that supplies the message. Clients send the message to the supplier, who receives it. It is represented by rounded rectangle.

2]ACTORS:
Stick figure represents the actor. It is the instances that invokes the interaction. Each actor has a specific name and a role.

3]LINKS:
A straight line connecting two objects indicates a relationship between them. The two objects are able to send messages to each other.

4]MESSAGES:
Typically, messages will have a number and description next to them. The number determines the order in which messages should be read.

6.6.2. COLLABORATION DIAGRAM:



EX NO: 7 DATE:	***State and Sequence Diagram, Deployment Diagram, Sample Frontend Design***

7.1. STATECHART DIAGRAM: -

7.1.1. STATECHART DIAGRAM DESCRIPTION:

State diagram describes the behaviour of a single object in response to a series of events in a system. This UML diagram models the dynamic flow of control from state to state of a particular object within a system.

COMPONENTS ARE:

• Initial State:

A filled circle followed by an arrow represents the student's login (object's) initial state.

• States

States in state chart diagram represent situations during the life of an Object. You can easily illustrate a state in Smart Draw by using a rectangle with rounded corners.

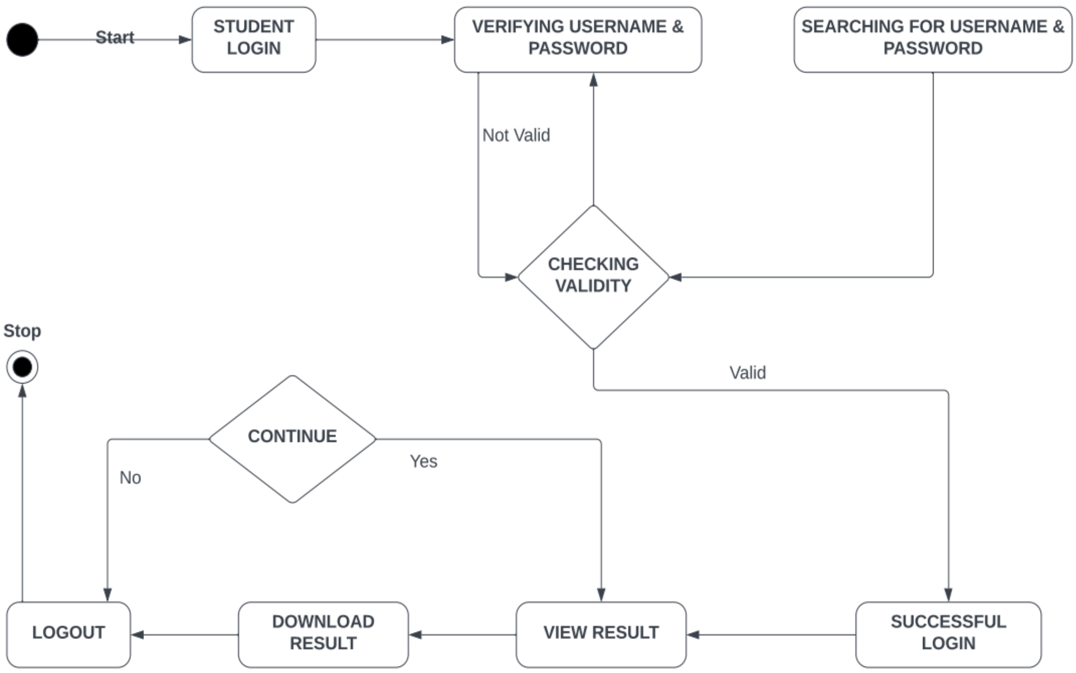
• Transition

A solid arrow represents the path between different states of an object of Exam result management system.

• Final State

An arrow pointing to a filled circle nested inside another circle represents the (object's) result.

7.1.2. STATECHART DIAGRAM:



7.2. SEQUENCE DIAGRAM: -

7.2.1. SEQUENCE DIAGRAM DESCRIPTION:

Sequence diagram are a popular dynamic modelling solution in UML because they specifically focus on lifelines, or the processes and objects that live simultaneously, and the messages exchanged between them to perform a function before the lifeline ends.

COMPONENTS IN SEQUENCE DIAGRAM:

1] ACTOR:

Stick figure represents the actor. Shows entities that interact the external objects of the system.

2] OBJECTS:

Rectangular boxes represent the object, demonstrates how an object will behave in the context of the system.

3] ACTIVATION BOXES:

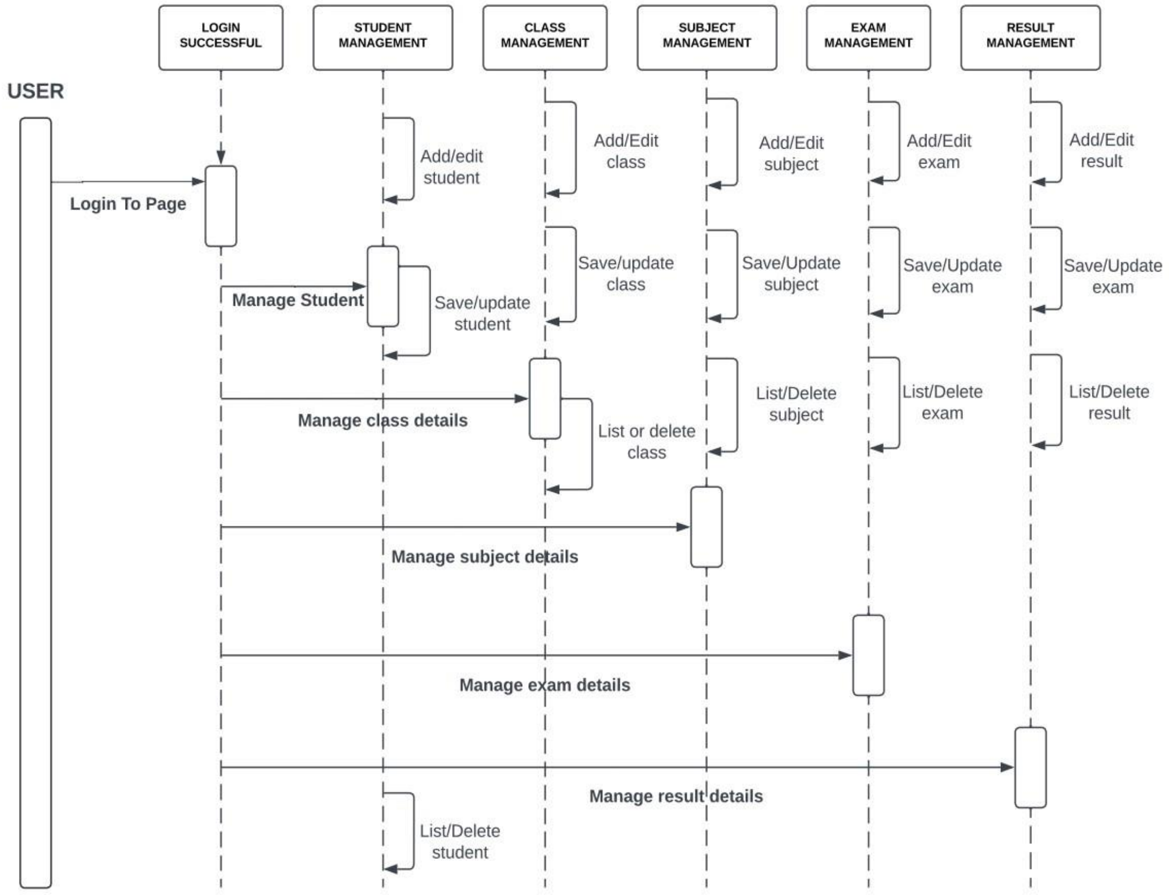
Represents the time needed for an object to complete a task. The longer the task will take, the longer the activation box becomes.

4] MESSAGE SYMBOLS:

We use the following arrows and message symbols to show how information is transmitted between objects. These symbols may reflect the start and execution of an operation or the sending and reception of a signal.

- SYNCHRONOUS MESSAGE: Represented by a solid line with a solid arrowhead. This symbol is used when a sender must wait for a response to a message before it continues. The diagram should show both the call and the reply.
- ASYNCHRONOUS MESSAGE: Represented by a solid line with a lined arrowhead. Asynchronous messages don't require a response before the sender continues. Only the call should be included in the diagram.
- REPLY MESSAGE: Represented by a dashed line with a lined arrowhead, these messages are replies to calls.
- DELETE MESSAGE: Represented by a solid line with a solid arrowhead, followed by an X. This message destroys an object.

7.2.2. SEQUENCE DIAGRAM:



7.3 DEPLOYMENT DIAGRAM: -

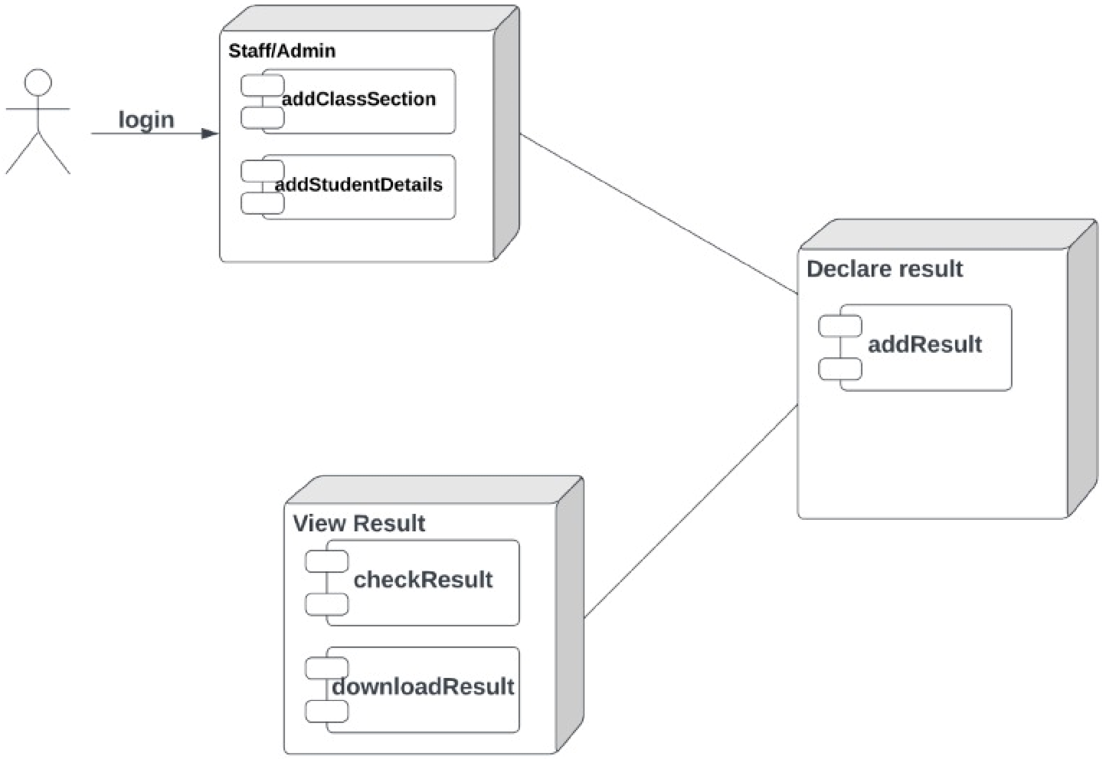
7.3.1. DEPLOYMENT DIAGRAM DESCRIPTION:

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

Deployment diagrams are typically used to visualize the physical hardware and software of a system. Using it you can understand how the system will be physically deployed on the hardware.

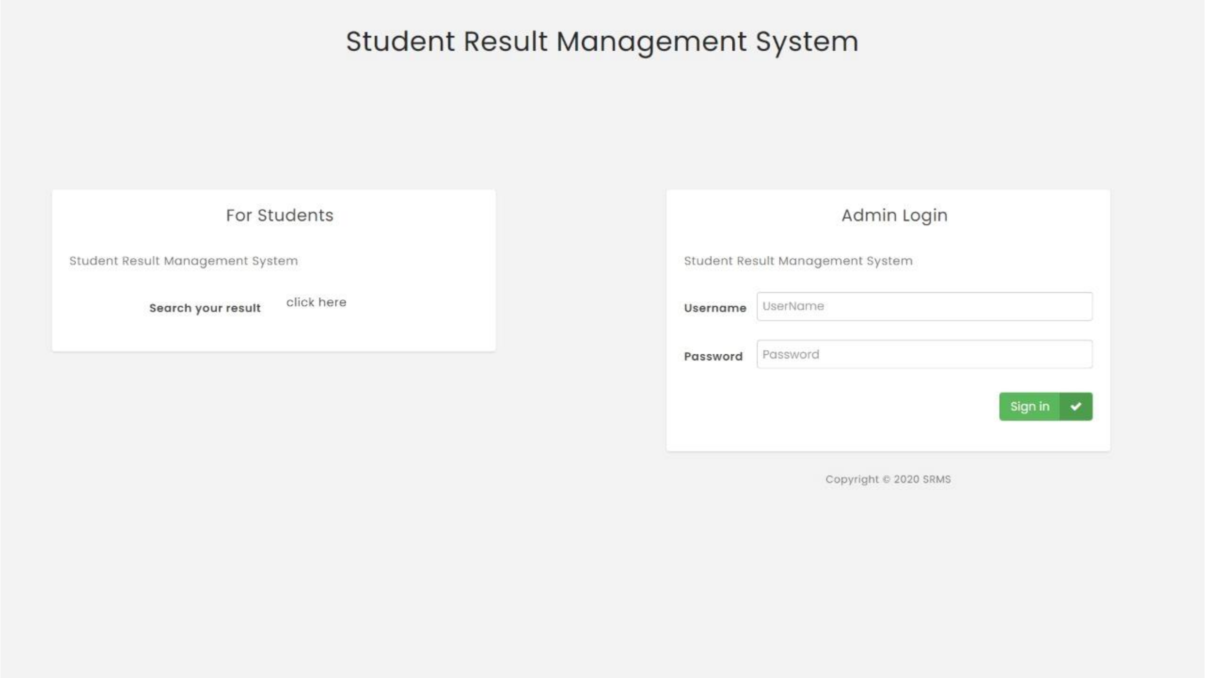
Deployment diagrams help model the hardware topology of a system compared to other UML diagram types which mostly outline the logical components of a system.

7.3.2. DEPLOYMENT DIAGRAM:



7.4. SAMPLE FRONTEND DESIGN: -

The figures 7.4.1shows the login page of both student and admin and 7.4.2 show the login the page for student to view result. And possesses a button that permits the student print the same results.



Student Result Management System

For Students

Student Result Management System

Search your result click here

Admin Login

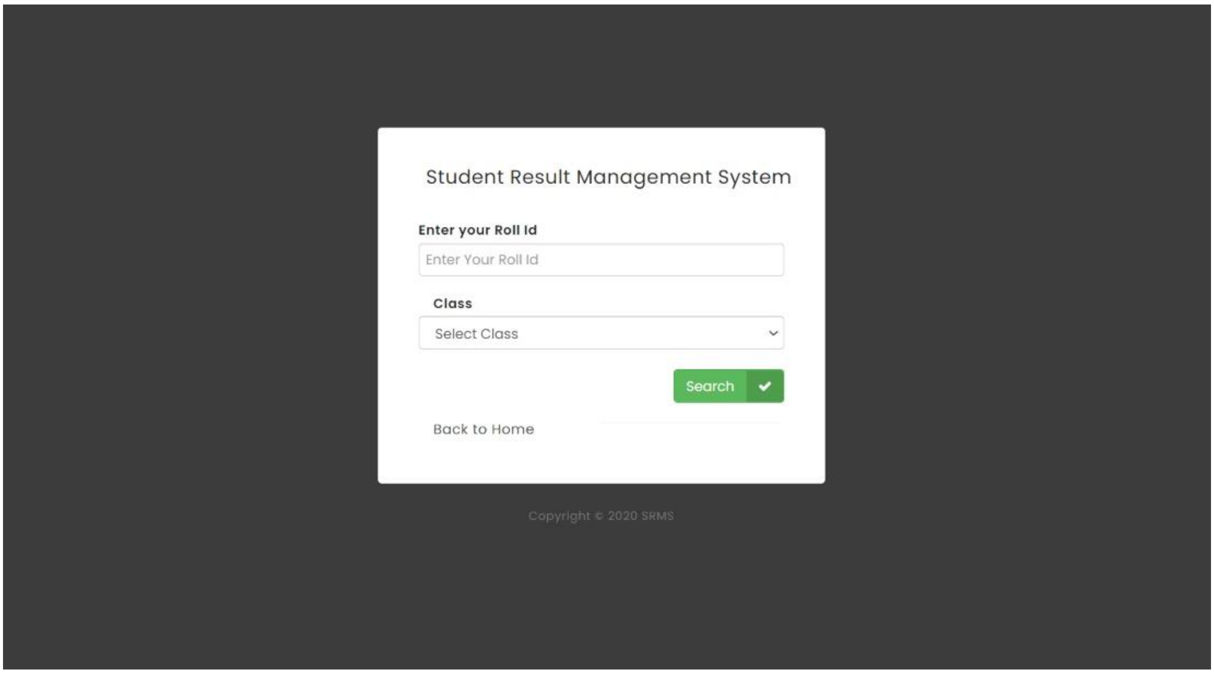
Student Result Management System

Username: UserName

Password: Password

Copyright © 2020 SRMS

Fig.7.4.1 Login page



Student Result Management System

Enter your Roll Id: Enter Your Roll Id

Class: Select Class

Copyright © 2020 SRMS

Fig.7.4.2. Student Login page to view result

EX NO: 8 **DATE:**	***Module Description, Module Implementation (phase 1) Using Agile***

8.1. MODULE DESCRIPTION: -

Following are the main Module of this Online Student Result Management system.

1. Login Module - At Admin Side
2. Classes Management Module - At Admin Side
3. Student Management Module - AT Admin Side
4. Exam Management Module - At Admin Side
5. User Management Module - At Admin Side
6. Profile Management Module - At Admin Side
7. Result Management Module - At Admin & Student Side

Features of Student Result Management System:

- Student can get result by Search Result by entering their roll number.
- Student can download Result in PDF format.
- This is Multiuser System, one user can create number of Sub user.
- Admin can see analytics data of total number of results publish, total exam data, total student data, total subject data and total classes data.
- Admin can Add Edit and Delete Classes data.
- Admin can enable and disable the status of Classes under this System.
- Admin can Add New Subject in particular classes and he or she can also edit or remove Subject data also.
- Admin can Enable and disable status of Subject of particular class.
- Admin can Add, Edit and Delete Student data.
- Admin user can Enable and Disable the status of student.
- Admin user can Add, Edit and Delete Exam data.
- Admin user can Enable and Disable the status of student.
- Admin user can create new user and he or she can also edit user data also.
- Admin can disable the login of sub user and he or she can also enable the login of sub user also.
- Admin and Sub User can edit their profile details.
- Sub user can add edit and delete result data and Admin can see all user result data in single page.
- Sub user and Admin can disable the any student result for publish on internet and they can also enable it also.

8.2. MODULE IMPLEMENTATION USING AGILE: -

The fig.8.2.1 shows the dashboard to create and manage the student classes, subjects, student, and result by the admin (staff).

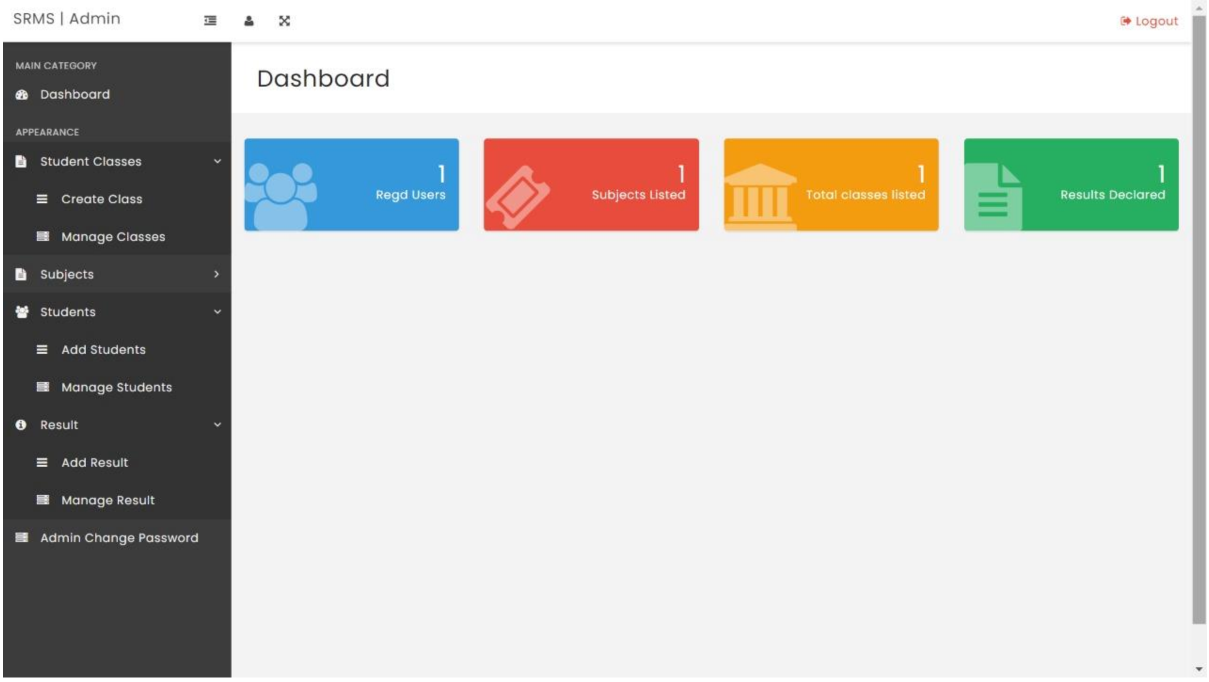
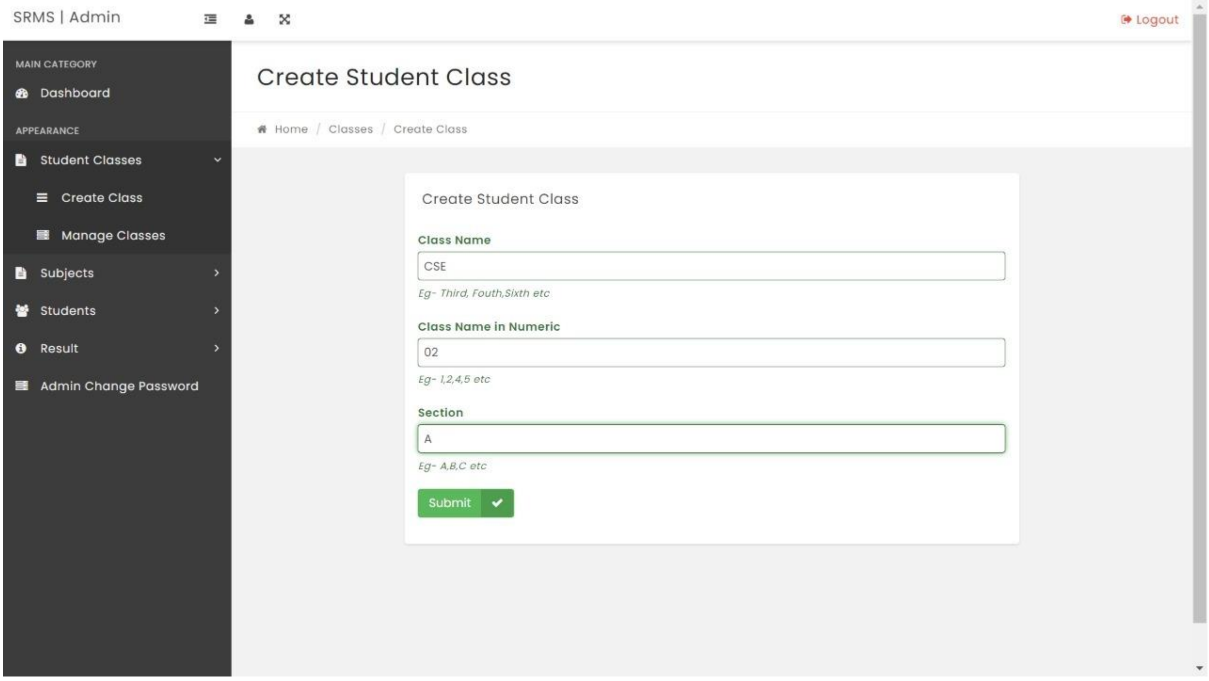


Fig.8.2.1. Dashboard

EX NO: 9 **DATE:**	***Module Implementation, Scrum Master to Induce New requirements in Agile Development***

The fig.9.1 show the page to create student class, which is done by the admin(staff), where the staff should add the class name, class number and section. Then click on the submit button to add the class. Fig.9.2. shows the page where the admin can manage student class if any changes occur.



Create Student Class

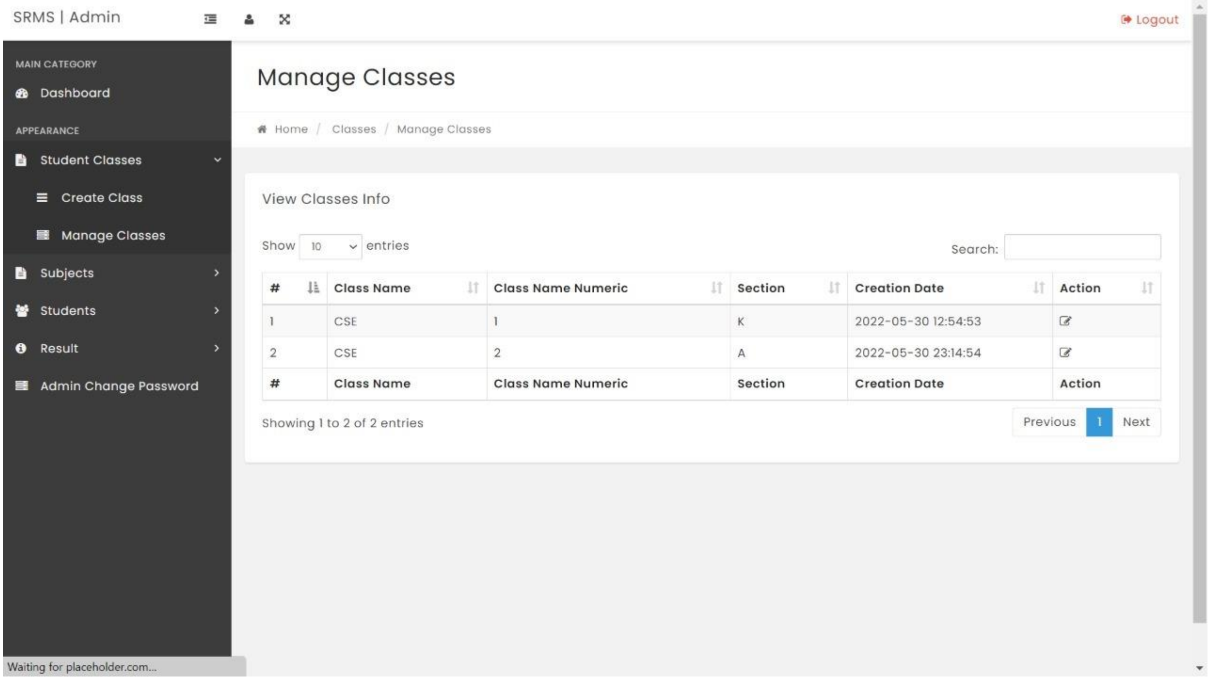
Create Student Class

Class Name: CSE
Eg- Third, Fouth,Sixth etc

Class Name in Numeric: 02
Eg- 1,2,4,5 etc

Section: A
Eg- A,B,C etc

Fig.9.1. Student class creation page



Manage Classes

View Classes Info

Show 10 entries

Search:

#	Class Name	Class Name Numeric	Section	Creation Date	Action
1	CSE	1	K	2022-05-30 12:54:53	
2	CSE	2	A	2022-05-30 23:14:54	
#	Class Name	Class Name Numeric	Section	Creation Date	Action

Showing 1 to 2 of 2 entries

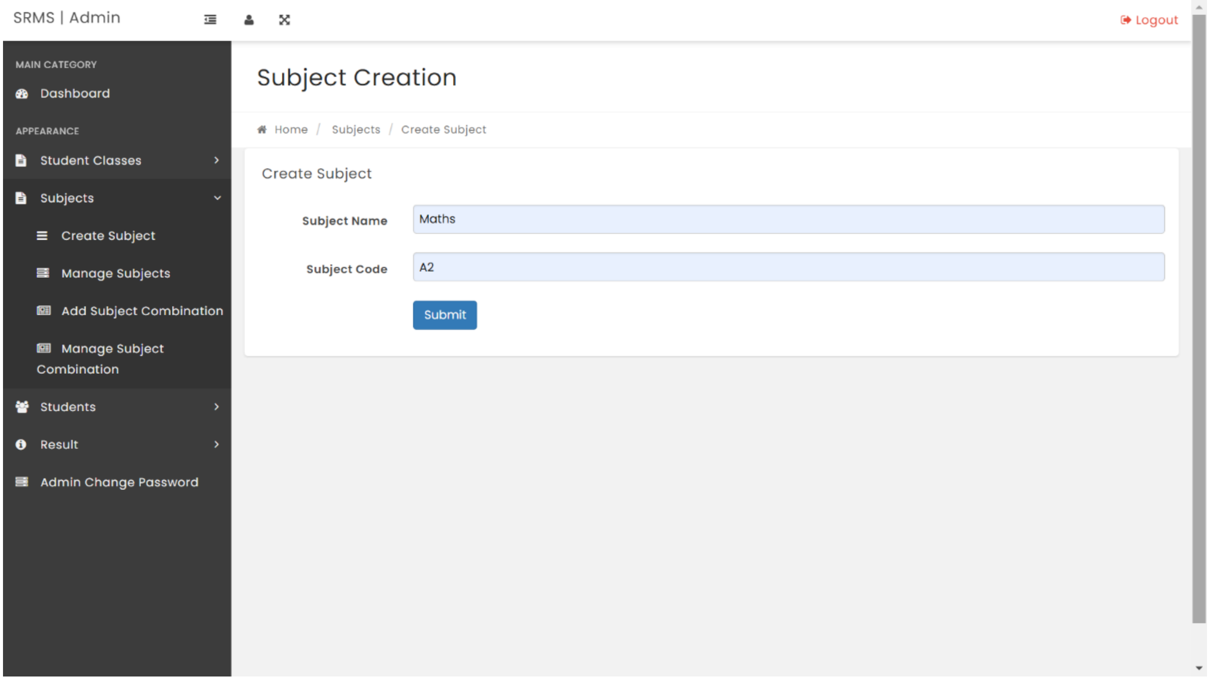
Fig.9.2. Class Management page

EX NO: 10

DATE:

Module Implementation (Phase 2), Scrum Master to Induce New Issues in Agile Development

In Subjects, there are 4 different components to add and manage different subjects to the different classes. As shown in fig.10.1, Subject name and subject code is generated by admin(staff).



Subject Creation

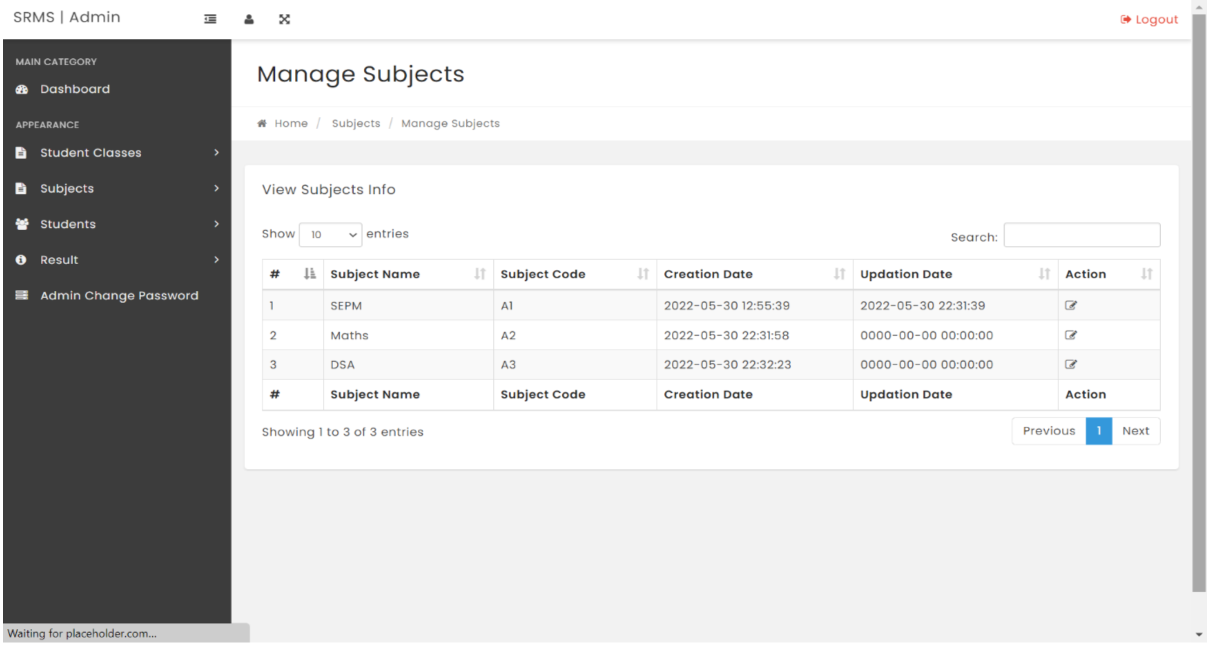
Create Subject

Subject Name: Maths

Subject Code: A2

Fig.10.1.Subject creation page

Second component is to manage subjects as shown in fig.10.2 you can view all the subject's information and can update if any changes required.



Manage Subjects

View Subjects Info

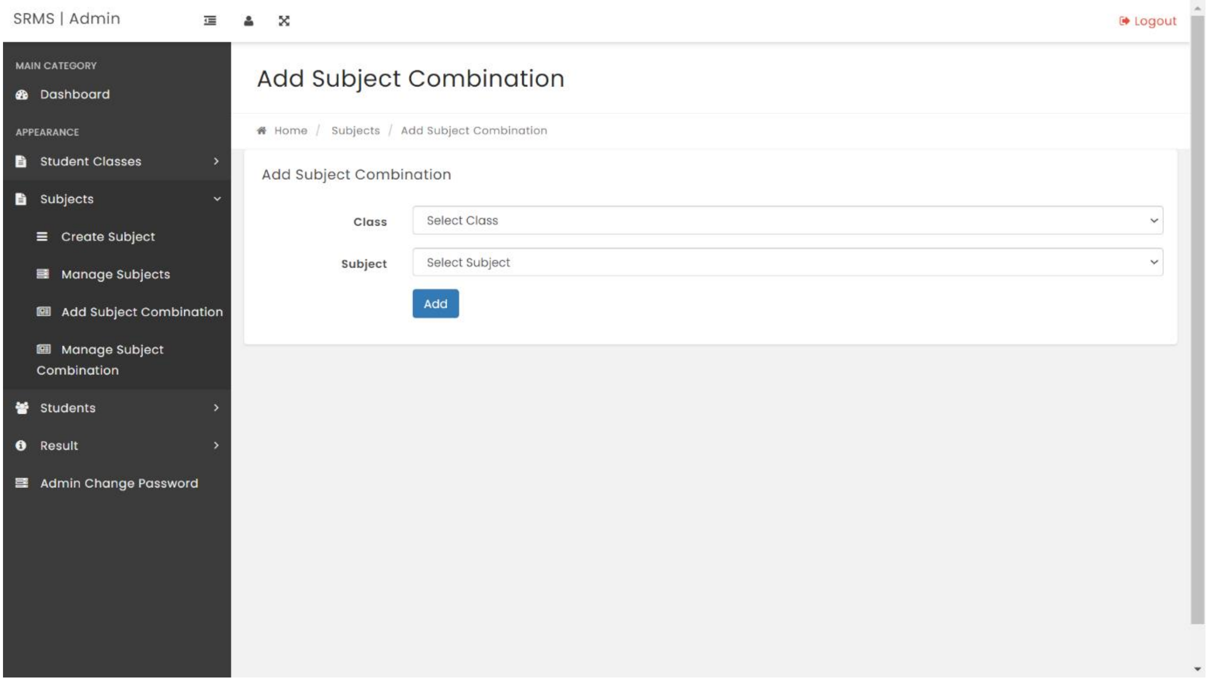
Show 10 entries

#	Subject Name	Subject Code	Creation Date	Updation Date	Action
1	SEPM	A1	2022-05-30 12:55:39	2022-05-30 22:31:39	✎
2	Maths	A2	2022-05-30 22:31:58	0000-00-00 00:00:00	✎
3	DSA	A3	2022-05-30 22:32:23	0000-00-00 00:00:00	✎
#	Subject Name	Subject Code	Creation Date	Updation Date	Action

Showing 1 to 3 of 3 entries

Fig.10.2. Subject Management page

The fig.10.3 show the page to add subject combination. Subject combination is to allocate a subject(s) to a particular section. After giving the subject and class click on add. So that the subject combination is done with the particular section or class.



Add Subject Combination

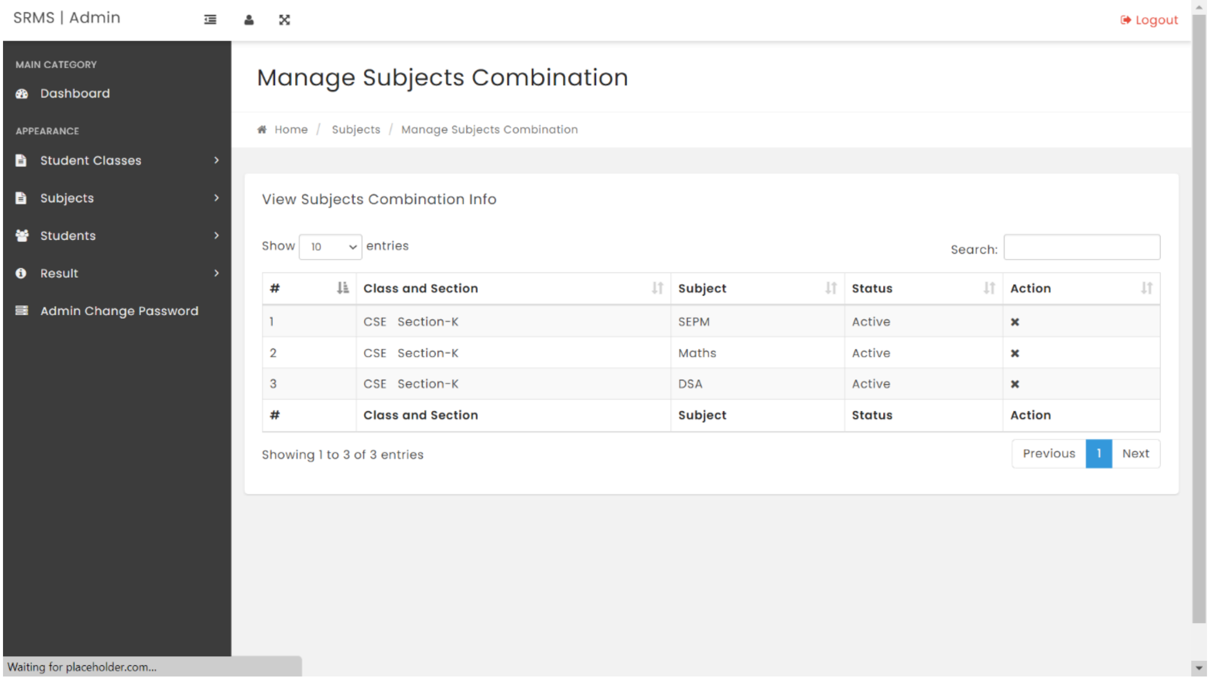
Add Subject Combination

Class: Select Class

Subject: Select Subject

Fig.10.3. Page to Add Subject Combination

The fig.10.4 shows the page to manage subject combination. In fourth component you can check and review all the subjects allocated to classes. Changes can be done here. You can delete any subject allocated to a class if not required.



Manage Subjects Combination

View Subjects Combination Info

Show 10 entries

Search:

#	Class and Section	Subject	Status	Action
1	CSE Section-K	SEPM	Active	✖
2	CSE Section-K	Maths	Active	✖
3	CSE Section-K	DSA	Active	✖
#	Class and Section	Subject	Status	Action

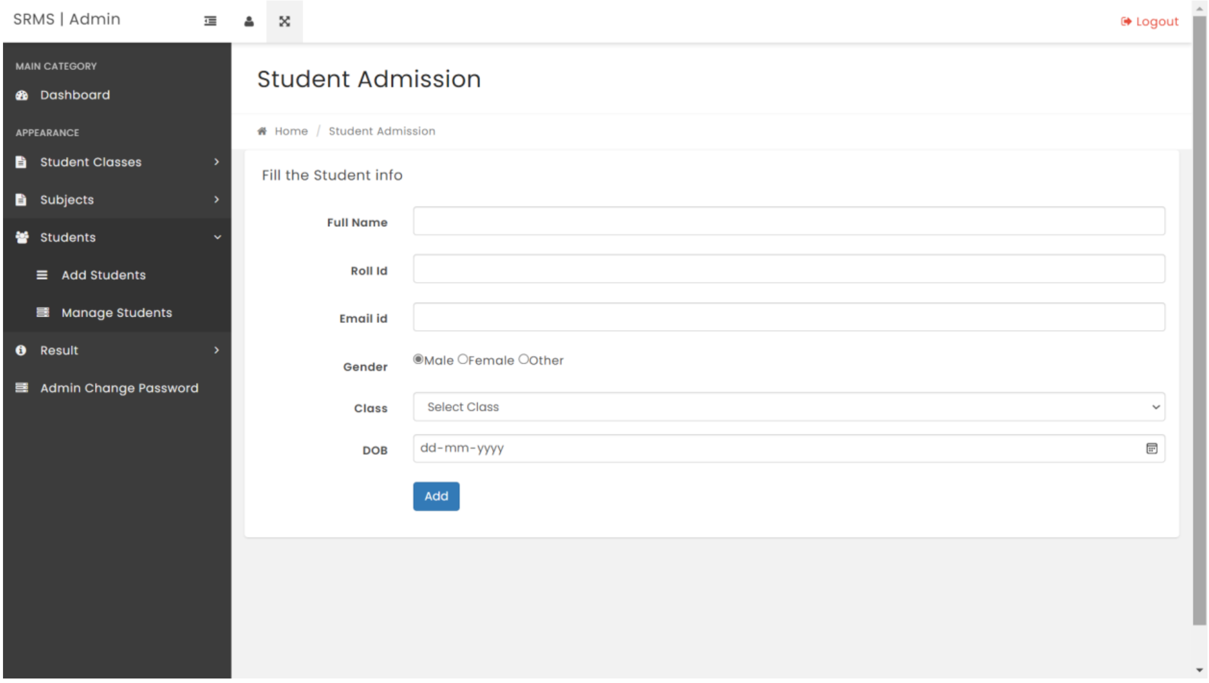
Showing 1 to 3 of 3 entries

Fig.10.4. Page to Manage Subject Combination

Here in students we have two different components

- Add students
- Manage students

In fig 10.5. student admission is shown where the admin have to fill the details of student i.e., name,roll number,email id, gender, class, and date of birth. By clicking on add button student is added to respective class.



Student Admission

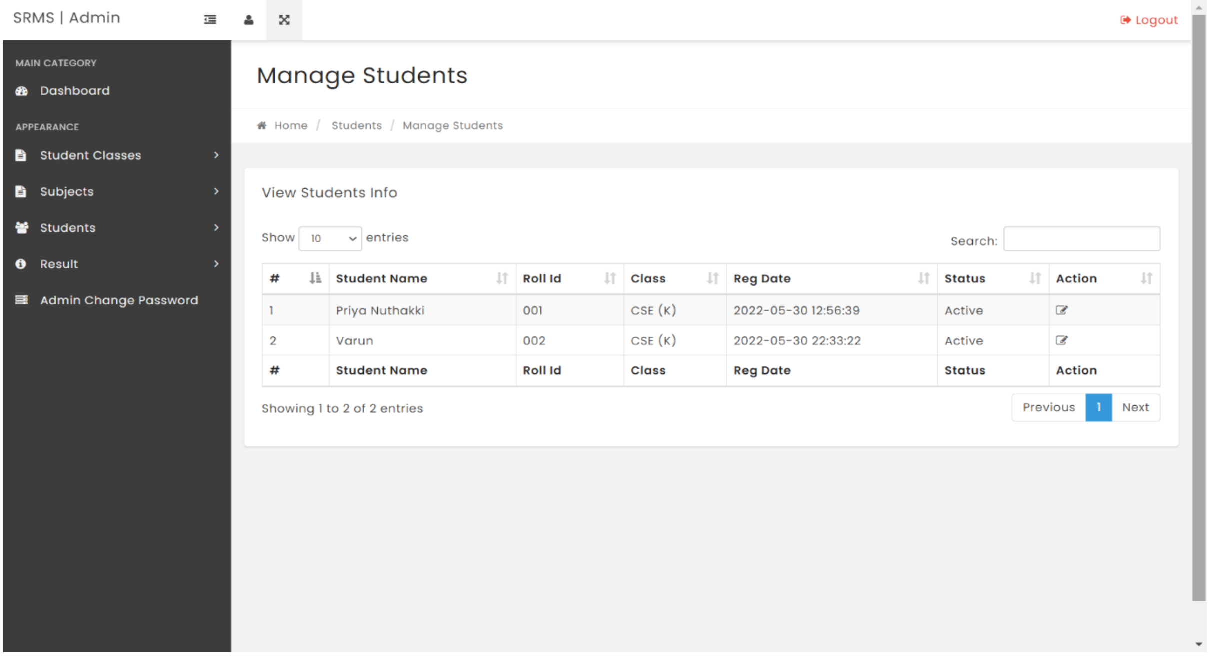
Fill the Student info

Full Name	
Roll Id	
Email id	
Gender	● Male ○ Female ○ Other
Class	Select Class
DOB	dd-mm-yyyy

Add

Fig.10.5. Student admission page

In manage Students page, the admin can view all the student details such as section, registration date. Also, about their active status. As shown in fig.10.6.



Manage Students

View Students Info

Show 10 entries

Search:

#	Student Name	Roll Id	Class	Reg Date	Status	Action
1	Priya Nuthakki	001	CSE (K)	2022-05-30 12:56:39	Active	✎
2	Varun	002	CSE (K)	2022-05-30 22:33:22	Active	✎
#	Student Name	Roll Id	Class	Reg Date	Status	Action

Showing 1 to 2 of 2 entries

Previous 1 Next

Fig.10.6. Page to Manage Student

EX NO: 11 **DATE:**	***Module Implementation (Phase 3) Scrum Master to Induce New requirements in Agile Development, Scrum Master to Induce New Issues in Agile Development, Code development***

11.1. MODULE IMPLEMENTATION (PHASE 3): -

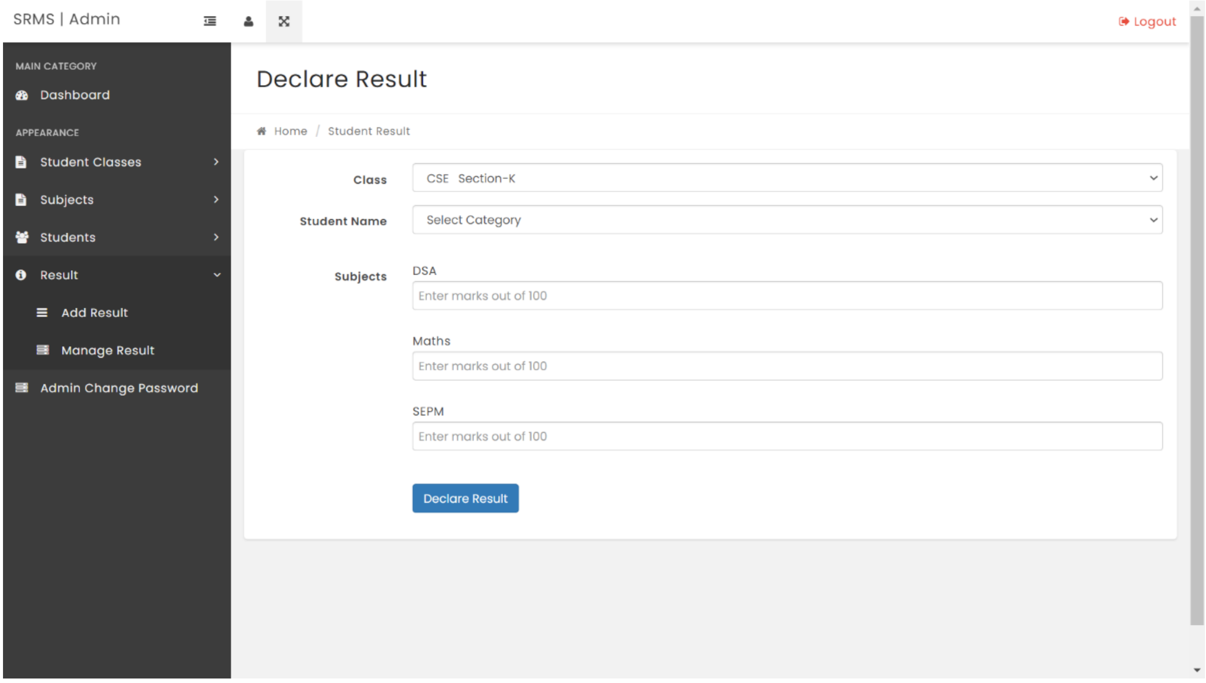
In result component, the admin can add and manage result.

In Declare result page, first the admin has to choose the class and then the student name from the list.

In subjects the admin can add the results to the subjects respectively.

After clicking on declare, the result is displayed in the list of manage results where the admin can check and update the result if required.

After declaring result the student can check their result from the website by giving their roll id and section.



Class: CSE Section-K

Student Name: Select Category

Subjects

DSA: Enter marks out of 100

Maths: Enter marks out of 100

SEPM: Enter marks out of 100

Fig.11.1. Result Declaration page

In the last component of dashboard, Admin can change their password.
First current password should be given, then new password and retype the new password in confirm password column and enter change.
The password will be changed, and page will log out after changing password.

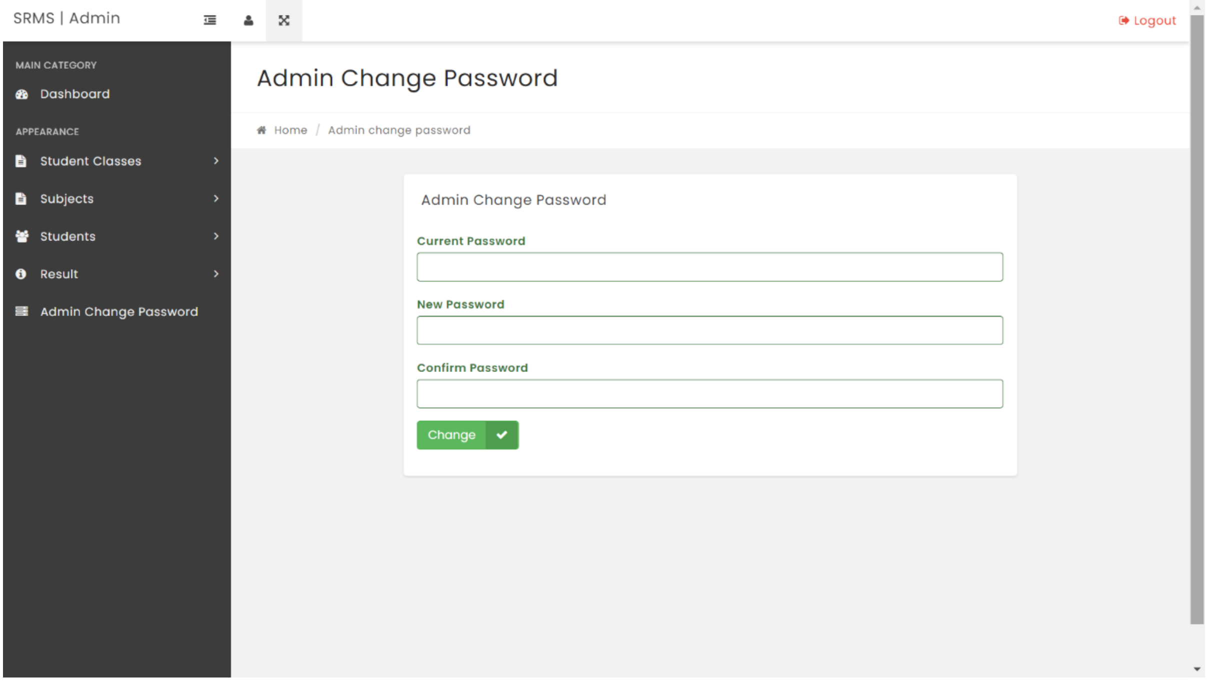


Fig.11.2

The final part of the project is to view the result by the student.
Student should enter their roll id and class to view the result.

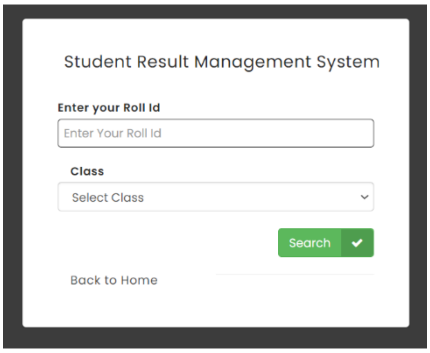
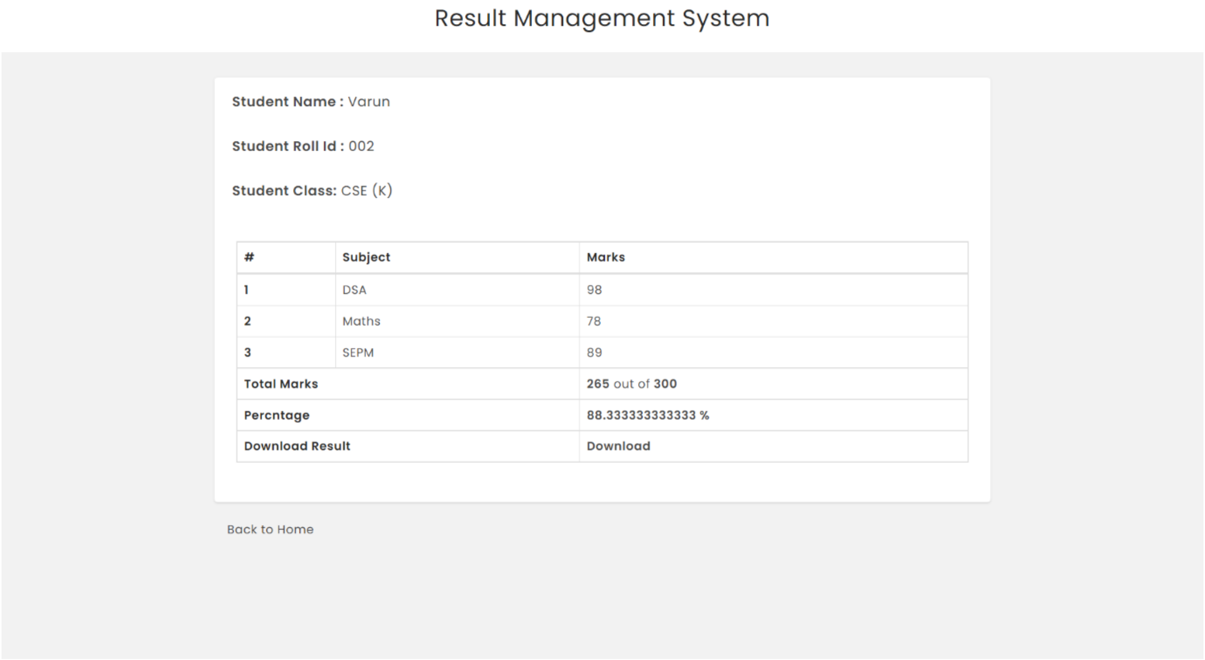


Fig.11.3

After clicking on Search in Fig.11.3. the result sheet is displayed as shown in Fig.11.4. The student sees their total marks and percentage. Also, student can do wnload the result in the form of pdf.



Result Management System

Student Name : Varun

Student Roll Id : 002

Student Class: CSE (K)

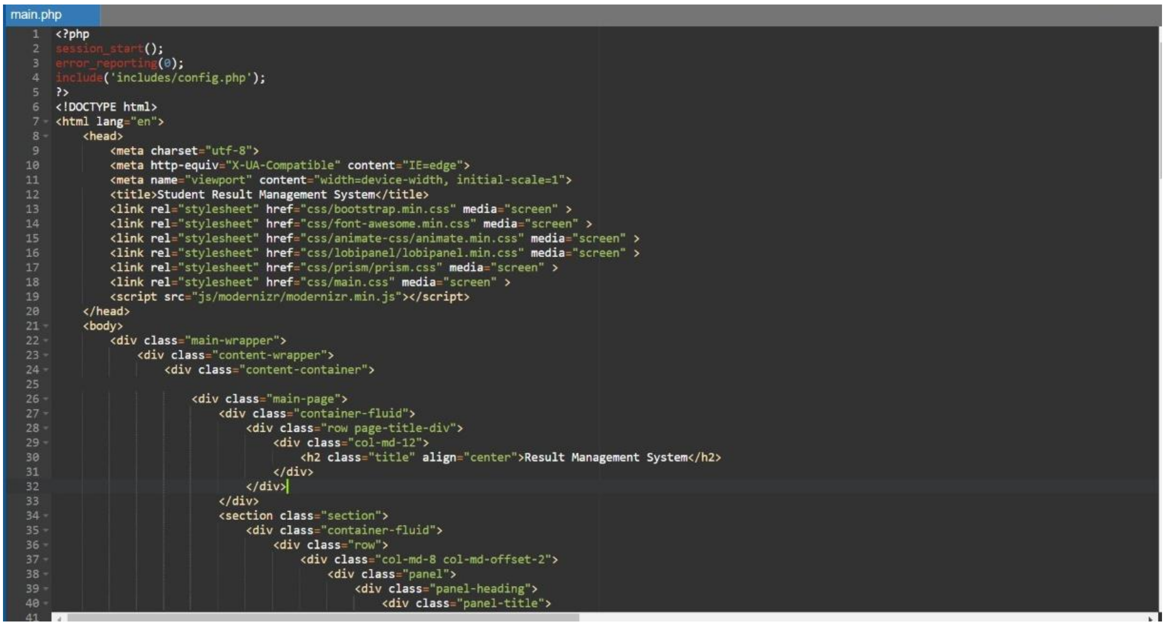
#	Subject	Marks
1	DSA	98
2	Maths	78
3	SEPM	89
Total Marks		265 out of 300
Percntage		88.333333333333 %
Download Result		Download

Back to Home

Fig. 11.4

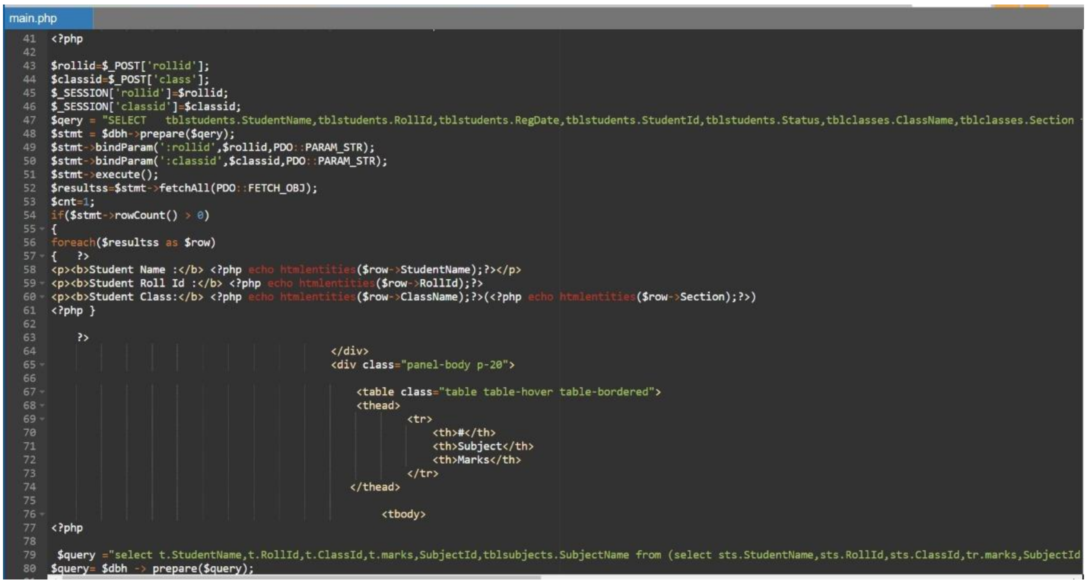
11.2. CODE DEVELOPMENT: -

The Exam Result Management System is developed using PHP, MYSQL, HTML, CSS, JAVASCRIPT. Functional decomposition of the system and its key modules are provided to explain the major functionalities proffered by the system. Also, use case diagram is presented to show the different categories of the system users and the various functionalities associated the different system user



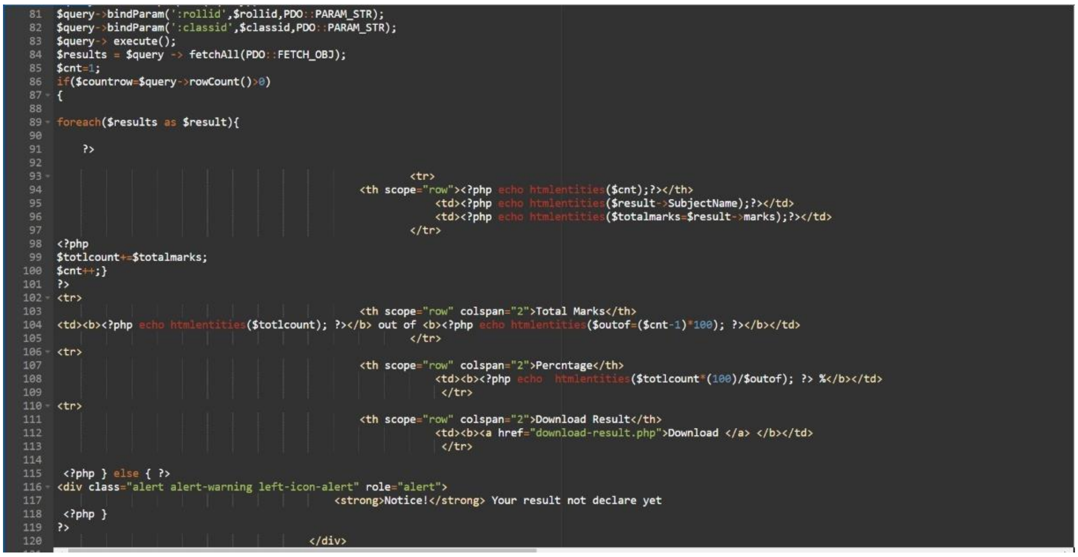
main.php

```
<?php
session_start();
error_reporting(0);
include('includes/config.php');
?>
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <title>Student Result Management System</title>
        <link rel="stylesheet" href="css/bootstrap.min.css" media="screen" >
        <link rel="stylesheet" href="css/font-awesome.min.css" media="screen" >
        <link rel="stylesheet" href="css/animate-css/animate.min.css" media="screen" >
        <link rel="stylesheet" href="css/lobipanel/lobipanel.min.css" media="screen" >
        <link rel="stylesheet" href="css/prism/prism.css" media="screen" >
        <link rel="stylesheet" href="css/main.css" media="screen" >
        <script src="js/modernizr/modernizr.min.js"></script>
    </head>
    <body>
        <div class="main-wrapper">
            <div class="content-wrapper">
                <div class="content-container">

                    <div class="main-page">
                        <div class="container-fluid">
                            <div class="row page-title-div">
                                <div class="col-md-12">
                                    <h2 class="title" align="center">Result Management System</h2>
                                </div>
                            </div>
                        </div>
                        <section class="section">
                            <div class="container-fluid">
                                <div class="row">
                                    <div class="col-md-8 col-md-offset-2">
                                        <div class="panel">
                                            <div class="panel-heading">
                                                <div class="panel-title">
```

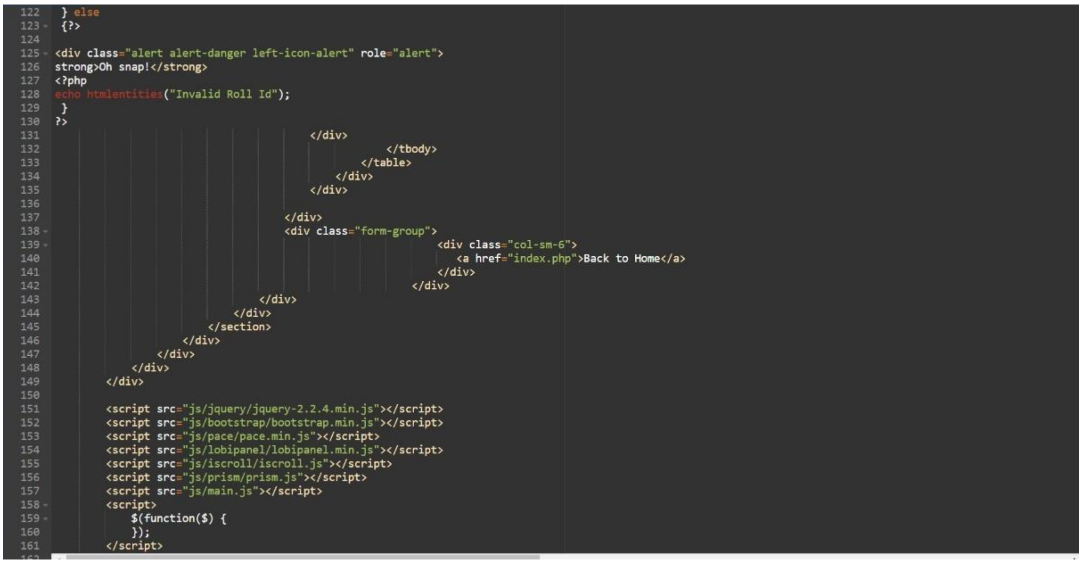


main.php

```
<?php

$rollid=$_POST['rollid'];
$classid=$_POST['class'];
$_SESSION['rollid']=$rollid;
$_SESSION['classid']=$classid;
$qery = "SELECT   tblstudents.StudentName,tblstudents.RollId,tblstudents.RegDate,tblstudents.StudentId,tblstudents.Status,tblclasses.ClassName,tblclasses.Section
$stmt = $dbh->prepare($qery);
$stmt->bindParam(':rollid',$rollid,PDO::PARAM_STR);
$stmt->bindParam(':classid',$classid,PDO::PARAM_STR);
$stmt->execute();
$resultss=$stmt->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($stmt->rowCount() > 0)
{
foreach($resultss as $row)
{   ?>
<p><b>Student Name :</b> <?php echo htmlentities($row->StudentName);?></p>
<p><b>Student Roll Id :</b> <?php echo htmlentities($row->RollId);?>
<p><b>Student Class:</b> <?php echo htmlentities($row->ClassName);?>(<?php echo htmlentities($row->Section);?>)
<?php }

    ?>
                                            </div>
                                            <div class="panel-body p-20">

                                                <table class="table table-hover table-bordered">
                                                <thead>
                                                        <tr>
                                                            <th>#</th>
                                                            <th>Subject</th>
                                                            <th>Marks</th>
                                                        </tr>
                                               </thead>

                                                    <tbody>
<?php

 $query ="select t.StudentName,t.RollId,t.ClassId,t.marks,SubjectId,tblsubjects.SubjectName from (select sts.StudentName,sts.RollId,sts.ClassId,tr.marks,SubjectId
$query= $dbh -> prepare($query);
```



```
$query->bindParam(':rollid',$rollid,PDO::PARAM_STR);
$query->bindParam(':classid',$classid,PDO::PARAM_STR);
$query-> execute();
$results = $query -> fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($countrow=$query->rowCount()>0)
{

foreach($results as $result){

    ?>

                                                        <tr>
                                                <th scope="row"><?php echo htmlentities($cnt);?></th>
                                                            <td><?php echo htmlentities($result->SubjectName);?></td>
                                                            <td><?php echo htmlentities($totalmarks=$result->marks);?></td>
                                                        </tr>
<?php
$totlcount+=$totalmarks;
$cnt++;}
?>
<tr>
                                                <th scope="row" colspan="2">Total Marks</th>
<td><b><?php echo htmlentities($totlcount); ?></b> out of <b><?php echo htmlentities($outof=($cnt-1)*100); ?></b></td>
                                                        </tr>
<tr>
                                                <th scope="row" colspan="2">Percntage</th>
                                                            <td><b><?php echo  htmlentities($totlcount*(100)/$outof); ?> %</b></td>
                                                             </tr>
<tr>
                                                <th scope="row" colspan="2">Download Result</th>
                                                            <td><b><a href="download-result.php">Download </a> </b></td>
                                                             </tr>

 <?php } else { ?>
<div class="alert alert-warning left-icon-alert" role="alert">
                                            <strong>Notice!</strong> Your result not declare yet
 <?php }
?>
                                        </div>
```



```
 } else
 {?>

<div class="alert alert-danger left-icon-alert" role="alert">
strong>Oh snap!</strong>
<?php
echo htmlentities("Invalid Roll Id");
 }
?>
                                        </div>
                                                    </tbody>
                                                </table>
                                            </div>
                                        </div>

                                    </div>
                                    <div class="form-group">
                                                            <div class="col-sm-6">
                                                               <a href="index.php">Back to Home</a>
                                                            </div>
                                                        </div>
                                </div>
                            </div>
                        </section>
                    </div>
                </div>
            </div>
        </div>

        <script src="js/jquery/jquery-2.2.4.min.js"></script>
        <script src="js/bootstrap/bootstrap.min.js"></script>
        <script src="js/pace/pace.min.js"></script>
        <script src="js/lobipanel/lobipanel.min.js"></script>
        <script src="js/iscroll/iscroll.js"></script>
        <script src="js/prism/prism.js"></script>
        <script src="js/main.js"></script>
        <script>
            $(function($) {
            });
        </script>
```

EX NO: 12 DATE:	***Master Test Plan, Test Case Design (Phase 1)***

12.1. MASTER TEST PLAN: -

TESTING OBJECTIVE	**FOCUSING ON PERFORMANCE ISSUE**
Test Items	Login system, Registration system, Uploading documents, Payment System
Features to be tested	Login verification, Registration feature, Uploading documents feature, Payment feature
Features not to be tested	Database Connectivity, Payment verifier, Two way pinging tool
Approach	Method – Manual Testing
Required Hardware/Software	A PC with 8 GB RAM, Internet Connectivity
Risks	Instability of the product
Testers & Schedule	Tester: SELIN RIONA V Scheduling Information: 25th April 2021, 3:00 PM
Estimate	Rs500/- (Excluding Tax and other charges)

12.2. TEST CASE DESIGN: -

➢ ***Testing:***

- The process of executing a system with the intent of finding an error.
- Testing is defined as the process in which defects are identified, isolated, subjected for rectification and ensured that product is defect free in order to produce the quality product and hence customer satisfaction.
- Quality is defined as justification of the requirements

- Defect is nothing but deviation from the requirements.
- Testing --- The presence of bugs
- Testing can demonstrate the presence of bugs, but not their absence
- Debugging and Testing are not the same thing!
- Testing is a systematic attempt to break a program or the AUT
- Debugging is the art or method of uncovering why the script /program did not execute properly.

- **Testing Methodologies:**
  - **Black box Testing**: is the testing process in which tester can perform testing on an application without having any internal structural knowledge of application. Usually Test Engineers are involved in the black box testing.
  - **White box Testing:** is the testing process in which tester can perform testing on an application with having internal structural knowledge. Usually, The Developers are involved in white box testing.
  - **Gray Box Testing:** is the process in which the combination of black box and white box techniques are used.

- **Positive Test Case:**
  - The positive flow of the functionality must be considered
  - Valid inputs must be used for testing.
  - Must have the positive perception to verify whether the requirements are justified.

- **Negative Test Case: -**
  - Must have negative perception.
  - Invalid inputs must be used for test.

EX NO: 13 **DATE:**	***Manual Testing***

13.1. MANULE TESTING: -

TEST AREA	INPUT	TEST DESCRIPTION	OUTPUT/RESULT
Login Module	Username and Password	Permits the user to enter into the application	Tested
Application Module	Fill the Application form	Allows the user to apply Passport	Tested
Upload Documents Module	Upload Documents required	Allows the user to upload documents	Tested
Payment Module	Click validate and pay option	Checks whether the payment feature is functioning and secure	Tested

EX NO: 14 **DATE:**	***User Manual, Analysis of Costing, Effort and Resources***

14.1. USER MANUAL: -

14.1.1 Introduction:

The "Examination and Result Management System" application allows students a simple interface to access their account from a mobile device to view their Result, Syllabus, Date sheet etc. This document will provide instructions for using the application.

14.1.2. Getting Started:

Download and install the "Examination and Result Management System" application available on play store or AppleStore. The application is compatible with Android versions 5.0 and above.

14.1.2a. Quick Start: Students:

Step 1: Tap the "Examination and Result Management System" icon in your device menu. The Login screen will be brought up.

Step 2: Students need to Log In to use the application by providing their username and password.

Step 3: Once you are logged in, your profile will be opened.

Step 4: Now, a student can view their Result, Date sheet, Syllabus and Admit Card.

Step 5: To view his Result Date sheet/Syllabus/Admit Card, he simply needs to click on the respective button displayed on his profile.

14.1.2b. System Requirements:

- Smartphone with Android versions 5.0 and above.
- Internet connection for Application to function.

14.1.3. Troubleshooting:

Missing or Incorrect Password or E-Mail. A message will be displayed in the event Try again with proper credentials to access

14.2. ANALYSIS OF COSTING, EFFORT AND RESOURCES: -

- ***DEVELOPMENT OF PROJECT:***

RESOURCE REQUIREMENT	COST
Computer with core i7 8th gen processor, at least 8GB of RAM, running on windows 10.	Rs. 65000/-
Code	Open Source
Printing	Rs. 500/-

- ***SERVER-END:***

RESOURCE REQUIREMENT	COST
My SQL	Enterprise Edition Rs. 10000/-
http web services	Std edition Rs. 5000/-
UPS	Rs. 2500/-

- ***OTHER COSTS:***

Employee salary	-
Maintenance cost	Rs. 1000/- per month