

# **Project report**

On

## **Student Management System**

Created By

**Uttam R. Gondaliya**

**BSC-IT-5**

Under the Guidance of

**PROF. KISHORSINGH VALA**

Submitted To

**GEETANAJALI COLLEGE OF COMPUTER SCIENCE &  
COMMERCE**



Affiliated To

**SAURASTRA UNIVERSITY**

**For Academic Year**

**2025-26**

## Acknowledgement

The project on **STUDENT MANAGEMENT SYSTEM** is developed in **PHP MYSQL** Language at GEETANJALI COLLEGE OF COMPUTER SCIENCE & COMMERCE.

I would like to acknowledge that my project has been completed and I am ensuring that, in this accomplishment, I would like to express my special gratitude to all my teachers and specially to **PROF. KISHORSINH VALA** without their guidance and feedback it is not possible to complete this assignment.

# INDEX

PARTICULARS	PAGE NO
▶ Introduction	
• Abstract	1
• Purpose	1
▶ Project Profile	2
▶ SDLC Process Model	3
▶ Software Resources	5
▶ List of Modules	6
▶ Data Dictionary	7
▶ DFD	
• Context Diagram	15
• Level 1 Diagram	16
▶ E-R Diagram	17
▶ Use Case Diagram	18
▶ Testing – Test Cases	19
▶ Snapshots	26
▶ Limitations	44
▶ Future Enhancement	44
▶ Conclusion	45

---

# INTRODUCTION

## Abstract:

The **Student Management System** is a web-based application developed using **PHP and MySQL** that helps efficiently manage student data. It allows administrators to add, update, delete, and view student details easily. The system streamlines tasks like attendance, grades, and course information management. It improves data accuracy and saves time by automating manual record-keeping.

Admin can add students , parents and teacher data and make a User Role .

## Purpose:

The main purpose of the Student Management System is to simplify and automate the management of student information in educational institutions. It helps administrators and teachers efficiently handle student records, attendance, and performance data. The system ensures accuracy, reduces paperwork, and makes information easily accessible whenever needed.

## Project Profile

**Project Title** : **Student Management System**

**Development Software** : **Visual Studio**

**Front End Tool** : **PHP , BOOTSTRAP**

**Back End Tool** : **MY SQL Server**

**Academic Year** : **2025–26**

**Developed By** : **Created by Uttam R. Gondaliya**

**Submitted To** : **GEETANJALI COLLEGE**

## Process Model

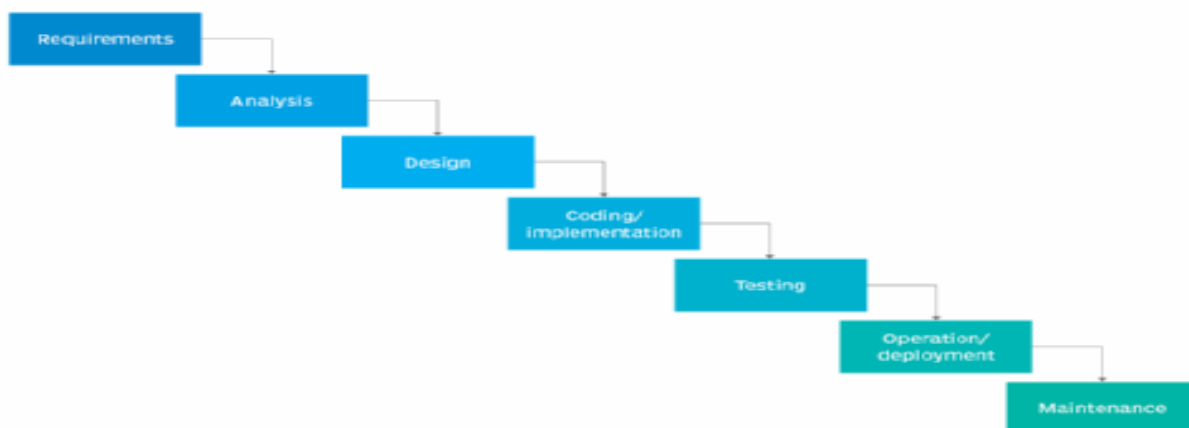
### ➤ The Waterfall Model

The Process Model used in our projects "**Student Management System**" is waterfall model.

The Waterfall Model is one of the oldest and most widely used software development life cycle (SDLC) models. It follows a **linear and sequential flow**, where each phase of the process must be completed before moving on to the next one. The main stages include **Requirement Analysis, System Design, Implementation, Testing, Deployment, and Maintenance**. This model is easy to understand and manage because of its step-by-step structure.

Although the Waterfall Model is simple and well-structured, it has some limitations. Once a phase is finished, it is difficult to go back and make changes. This makes it less flexible for projects where requirements might change during development. Therefore, it is most suitable for **small projects with clear and fixed requirements**. Despite its drawbacks, the Waterfall Model provides a **disciplined and organized approach** to software development and is still used in many industries today.

### Waterfall model



➤ **Reason the waterfall model in the software development cycle**

- **Simple and Easy to Use:**  
The Waterfall Model is straightforward and easy to understand. Each phase has clear objectives and deliverables, making project management simple and structured.
- **Well-Defined Stages:**  
Every stage of development is clearly defined with specific milestones, which helps in better planning, documentation, and tracking of progress.

➤ **Advantages of the waterfall model:**

- This model is simple and easy to understand and use. In, this model phases are processed and completed one at a time and phases do not overlap.
- → Waterfall model works well for smaller projects where sequence is very well understood.

## Software Resources

- Frontend Design Tool : Visual Studio Code
- Backend Design Tool: Xampp SQL Server
- Code-Behind Language :
- Frontend Language : Bootstrap , css , Js
- Backend Language : PHP

### ➤ **Functionality**

- Login

1. Student Login

- Student can see his attendance
- See Exam Marks
- His own Profile
- Can See Notice

2. Admin Login

- Add Teacher details
- Add Student details
- Add Parent details
- Create User Role (Students , Teachers , Parents )

3. Teacher Login

- Take Student Attendance
- Set Exams
- Set Timetable
- Upload Student Marks
- Add Subjects And Classes

4. Parents Login

- See their Student Attendance
- See Exam marks
- See notice



## List of Module

- Login module
- Student , teacher , admin , Parent Login
- Attendance Module
- Exam / Grade Module
- Search Update Student Parent Module

## Data Dictionary

### 1. Table: admin

Column Name	Data Type	Constraint
aid	int(11)	PRIMARY KEY, NOT NULL
fname	varchar(50)	NOT NULL
lname	varchar(50)	NOT NULL
email	varchar(100)	UNIQUE, NOT NULL
phone	varchar(20)	NULL
created_at	timestamp	NOT NULL, DEFAULT Current_Timestamp()

### 2. Table: attendance1

Column Name	Data Type	Constraint
id	int(11)	PRIMARY KEY, NOT NULL
student_id	varchar(50)	UNIQUE (with date), NOT NULL
class_id	varchar(50)	NOT NULL
date	date	NOT NULL
status	enum('Present','Absent')	NOT NULL
recorded_by	varchar(50)	NOT NULL
timestamp	timestamp	NOT NULL, DEFAULT current_timestamp()

### 3. Table: classroom

Column Name	Data Type	Constraint
hno	varchar(50)	PRIMARY KEY, NOT NULL
title	varchar(50)	NOT NULL
location	varchar(50)	NOT NULL
capacity	int(3)	NOT NULL

### 4. Table: exam

Column Name	Data Type	Constraint
id	int(11)	PRIMARY KEY, NOT NULL
subject	varchar(50)	NOT NULL
teacher	varchar(50)	NOT NULL
classroom	varchar(50)	NOT NULL
date	date	NOT NULL
stime	time	NOT NULL

### 5. Table: examresult

Column Name	Data Type	Constraint
exam	int(11)	PRIMARY KEY (part 1), NOT NULL
student	varchar(50)	PRIMARY KEY (part 2), NOT NULL
marks	int(10)	NOT NULL
grade	varchar(10)	NOT NULL

### 6. Table: notice

Column Name	Data Type	Constraint
id	int(11)	PRIMARY KEY, NOT NULL
notice	varchar(1500)	NOT NULL
odience	varchar(100)	NOT NULL
date	datetime	NOT NULL

## 7. Table: parent

Column Name	Data Type	Constraint
pid	int(11)	PRIMARY KEY, NOT NULL
fname	varchar(50)	NOT NULL
lname	varchar(50)	NOT NULL
contact	varchar(20)	NOT NULL
job	varchar(50)	NOT NULL
address	varchar(250)	NOT NULL
gender	varchar(25)	NOT NULL
nic	int(50)	NOT NULL
email	varchar(50)	UNIQUE, NOT NULL

## 8. Table: student

Column Name	Data Type	Constraint
sid	varchar(25)	PRIMARY KEY, NOT NULL
fname	varchar(50)	NOT NULL
lname	varchar(50)	NOT NULL
bday	date	NOT NULL
address	varchar(250)	NOT NULL
parent	int(10)	NOT NULL
gender	varchar(10)	NOT NULL
classroom	varchar(25)	NOT NULL
email	varchar(50)	UNIQUE, NOT NULL

## 9. Table: subject

Column Name	Data Type	Constraint
sid	varchar(50)	PRIMARY KEY, NOT NULL
title	varchar(200)	NOT NULL
description	varchar(500)	NOT NULL

## 10. Table: user

Column Name	Data Type	Constraint	Column Name
role	varchar(50)	NOT NULL	role
email	varchar(50)	PRIMARY KEY, NOT NULL	email
password	varchar(50)	NOT NULL	password

## 11. Table: teacher

Column Name	Data Type	Constraint
tid	varchar(50)	PRIMARY KEY, NOT NULL
fname	varchar(50)	NOT NULL
lname	varchar(50)	NOT NULL
address	varchar(50)	NOT NULL
contact	varchar(50)	NOT NULL
bday	date	NOT NULL
skill	varchar(500)	NOT NULL
gender	varchar(25)	NOT NULL
email	varchar(50)	UNIQUE, NOT NULL

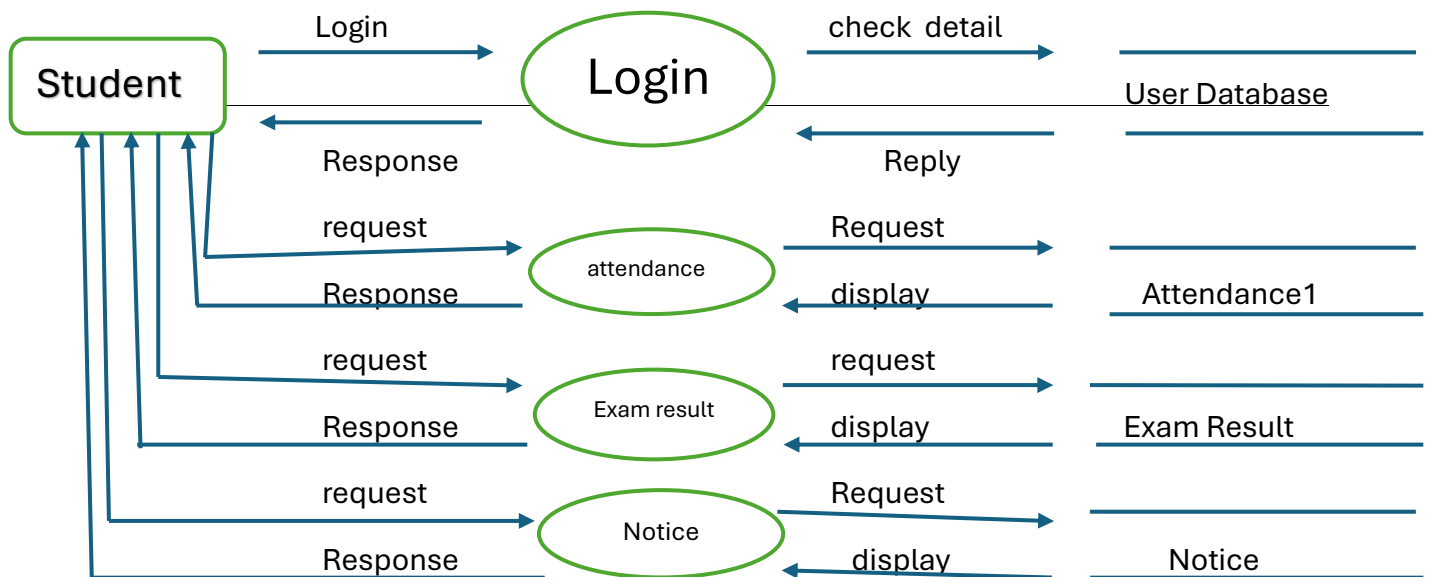
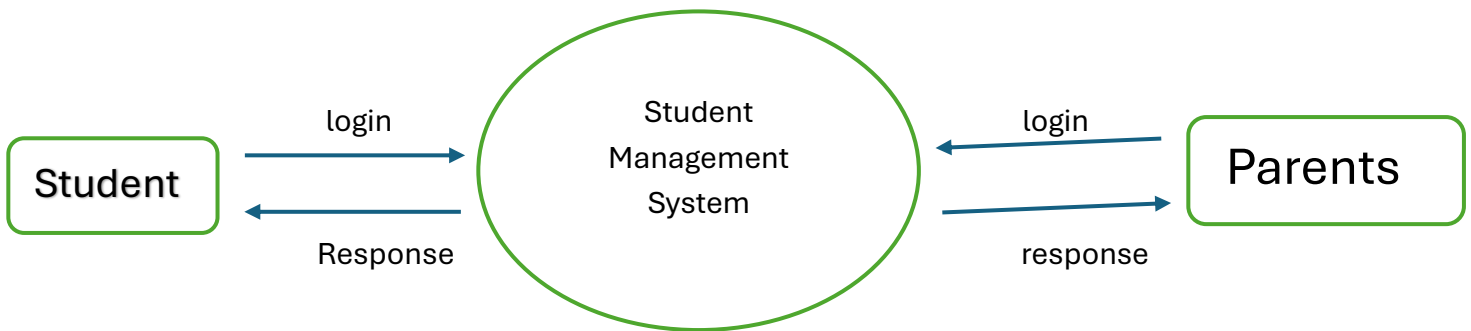


## 12. Table: timetable

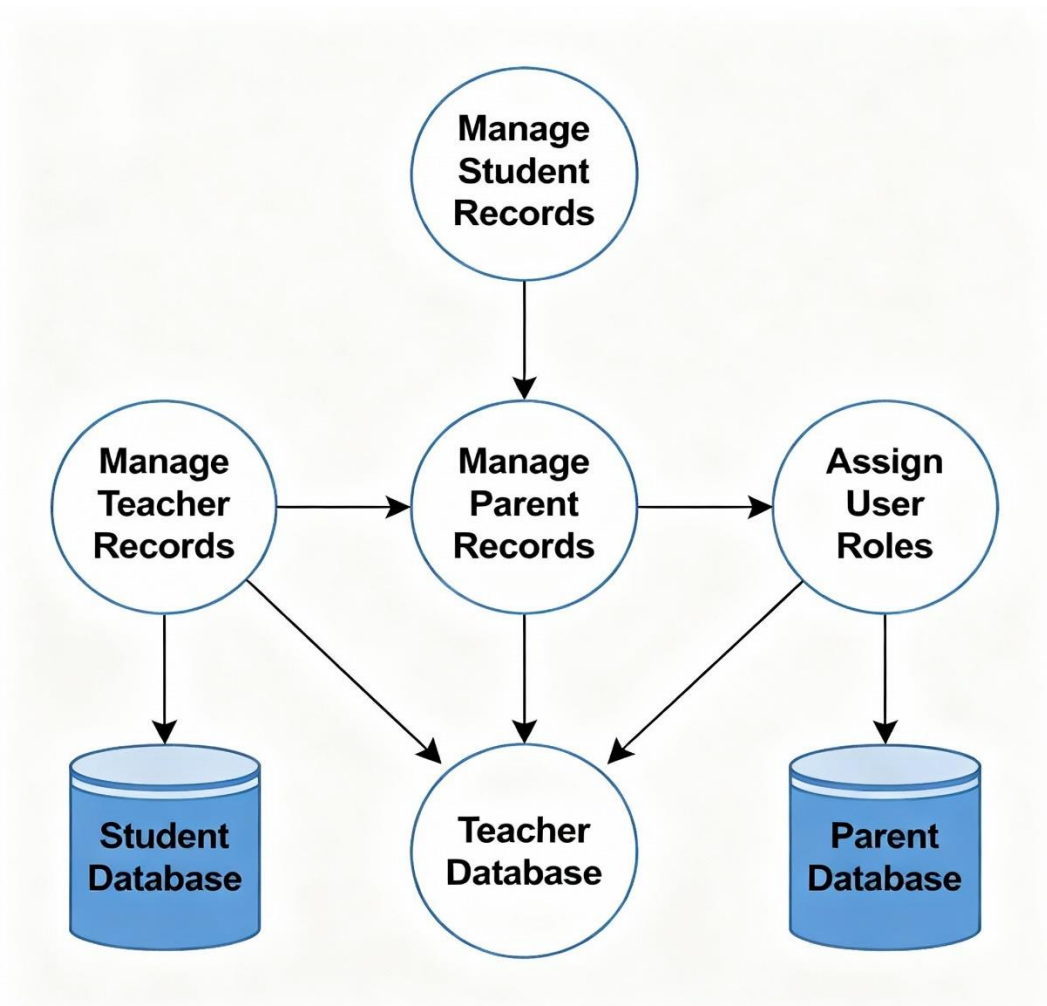
Column Name	Data Type	Constraint
id	int(11)	PRIMARY KEY, NOT NULL
class_id	varchar(50)	NOT NULL
classroom	varchar(50)	FOREIGN KEY → classroom(hno)
subject_id	varchar(50)	FOREIGN KEY → subject(sid)
day	varchar(20)	NOT NULL
start_time	time	NOT NULL
end_time	time	NOT NULL
teacher_id	varchar(50)	NOT NULL
created_at	timestamp	NOT NULL, DEFAULT current_timestamp()

## DFD [Data Flow Diagram]

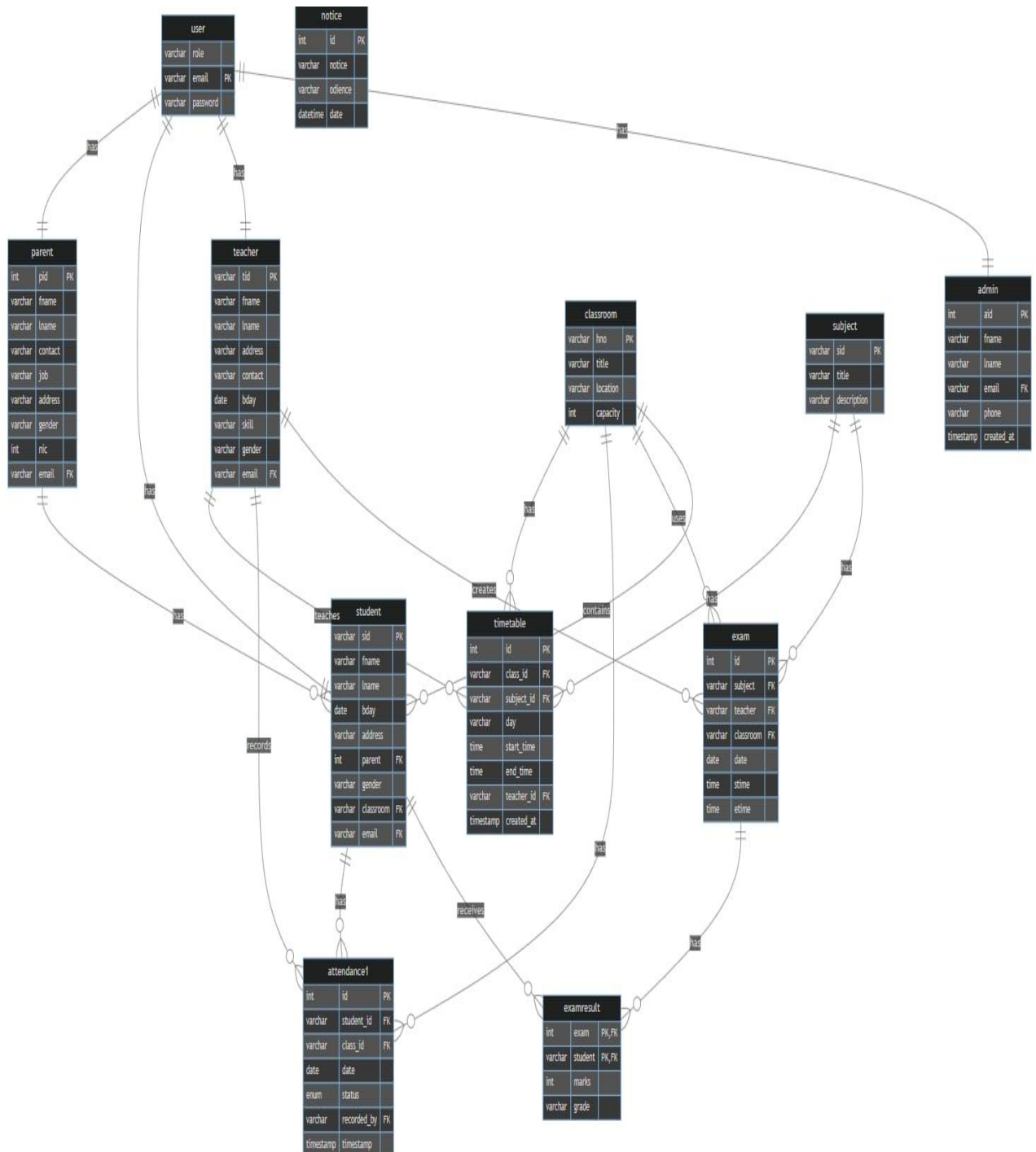
➤ 0 level DFD (context Diagram)



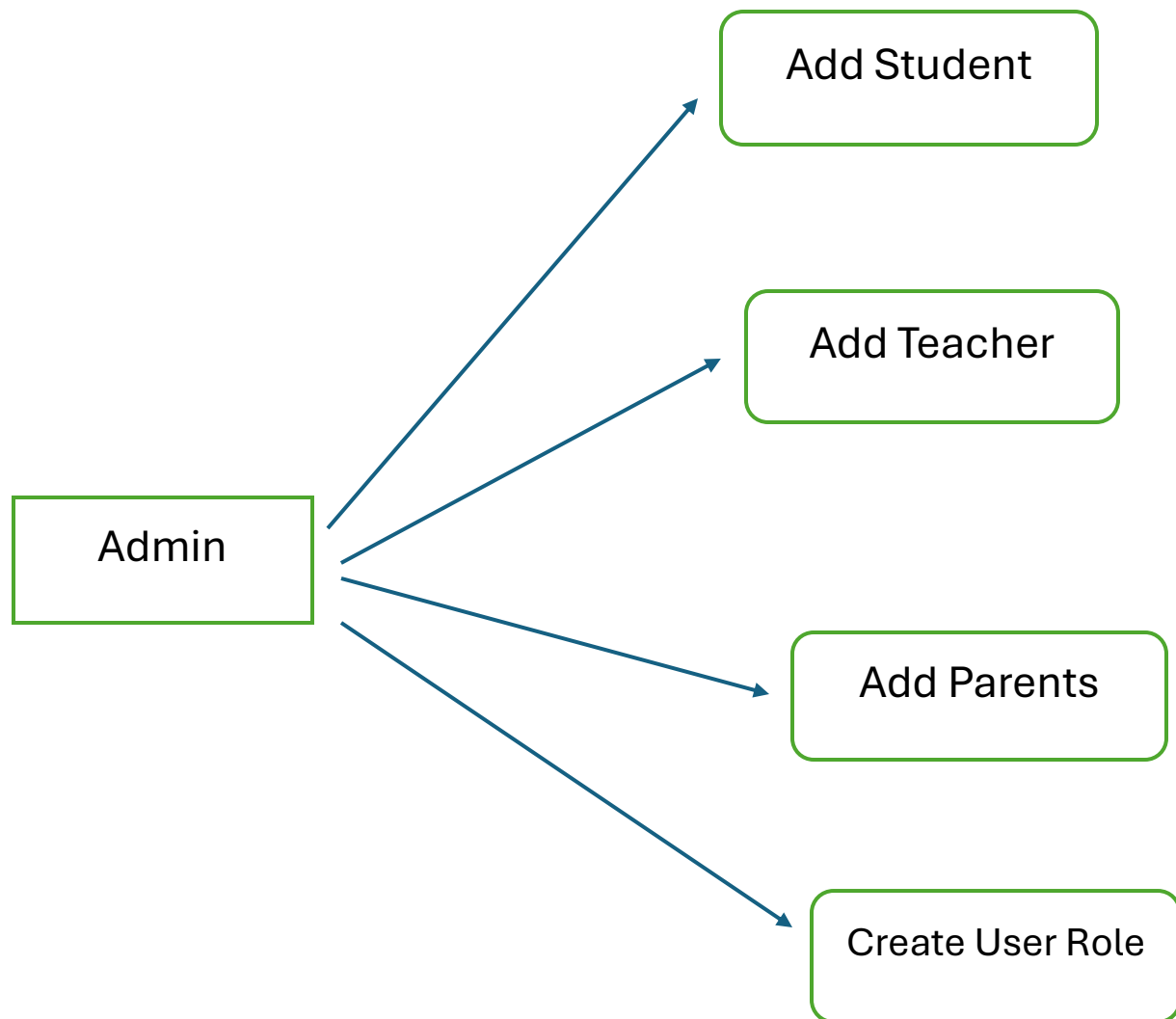
➤ 1 Level DFD



## ERD [Entity Relationship Diagram]



## Use case Diagram



## Test Cases

### Login

Sr. No	Module	Test Scenario	Test Steps	Result	Remarks
1	Login	Check login with valid credentials	1. Open login page 2. Enter valid email & password 3. Click Login	User redirected to correct panel based on role	Pass if redirected correctly
2	Login	Check login with invalid credentials	1. Open login page 2. Enter invalid email or password 3. Click Login	Error message displayed "Invalid credentials"	Pass if error shown
3	Admin Panel	Add a new student	1. Login as admin 2. Go to Add Student 3. Fill details 4. Submit	Student added to database and visible in list	Pass if student saved
4	Student Panel	View exam results	1. Login as student 2. Go to Exam Results section	Display student's exam results	Pass if correct results shown
5	Teacher Panel	Take attendance for a class	1. Login as teacher 2. Go to Attendance 3. Select class 4. Mark present/absent 5. Submit	Attendance recorded in database	Pass if attendance saved
6	Parent Panel	View child attendance	1. Login as parent 2. Go to Attendance section	Display child's attendance	Pass if correct attendance shown

Admin

Sr. No	Module	Test Scenario	Test Steps	Result	Remarks
1	Admin	Add Student Details - Valid	1. Login as admin 2. Go to Add Student 3. Fill all required fields 4. Click Submit	New student added successfully and visible in student list	Pass if student saved correctly
2	Admin	Add Student Details - Invalid	1. Login as admin 2. Go to Add Student 3. Leave required fields empty 4. Click Submit	Error message displayed "Please fill all required fields"	Pass if error shown correctly
3	Admin	Add Teacher Details - Valid	1. Login as admin 2. Go to Add Teacher 3. Fill all required fields 4. Click Submit	New teacher added successfully and visible in teacher list	Pass if teacher saved correctly
4	Admin	Add Teacher Details - Invalid	1. Login as admin 2. Go to Add Teacher 3. Enter invalid email/phone 4. Click Submit	Error message displayed "Invalid input"	Pass if error shown correctly

## PROJECT REPORT

5	Admin	Add Parent Details - Valid	1. Login as admin 2. Go to Add Parent 3. Fill all required fields 4. Click Submit	New parent added successfully and visible in parent list	Pass if parent saved correctly
6	Admin	Add Parent Details - Invalid	1. Login as admin 2. Go to Add Parent 3. Leave required fields empty 4. Click Submit	Error message displayed "Please fill all required fields"	Pass if error shown correctly
7	Admin	Assign User Role - Valid	1. Login as admin 2. Go to User Role assignment 3. Select user and role 4. Click Submit	User role assigned successfully and saved in database	Pass if role assigned correctly
8	Admin	Assign User Role - Invalid	1. Login as admin 2. Go to User Role assignment 3. Do not select a role 4. Click Submit	Error message displayed "Please select a role"	Pass if error shown correctly



## Teacher

Sr. No	Module	Test Scenario	Test Steps	Result	Remarks
1	Teacher	Add Subject - Valid	1. Login as teacher 2. Go to Add Subject 3. Fill subject details 4. Click Submit	Subject added successfully and visible in list	Pass if subject saved correctly
2	Teacher	Add Subject - Invalid	1. Login as teacher 2. Go to Add Subject 3. Leave required fields empty 4. Click Submit	Error message displayed "Please fill all required fields"	Pass if error shown correctly
3	Teacher	Add Class - Valid	1. Login as teacher 2. Go to Add Class 3. Fill class details 4. Click Submit	Class added successfully and visible in list	Pass if class saved correctly
4	Teacher	Add Timetable	1. Login as teacher 2. Go to Add Timetable 3. Fill timetable details 4. Click Submit	Timetable added successfully	Pass if timetable saved correctly
5	Teacher	Take Attendance Class-wise	1. Login as teacher 2. Go to Attendance 3. Select class 4. Mark present/absent 5. Submit	Attendance recorded successfully	Pass if attendance saved

## PROJECT REPORT

6	Teacher	Send Notice	1. Login as teacher 2. Go to Add Notice 3. Enter notice details 4. Click Submit	Notice added and visible to students/parents	Pass if notice sent correctly
7	Teacher	Set Exam	1. Login as teacher 2. Go to Set Exam 3. Fill exam details 4. Click Submit	Exam created successfully	Pass if exam saved correctly
8	Teacher	Set Exam Result	1. Login as teacher 2. Go to Upload Result 3. Select exam & student 4. Enter marks 5. Submit	Result uploaded successfully	Pass if result saved correctly

## Student

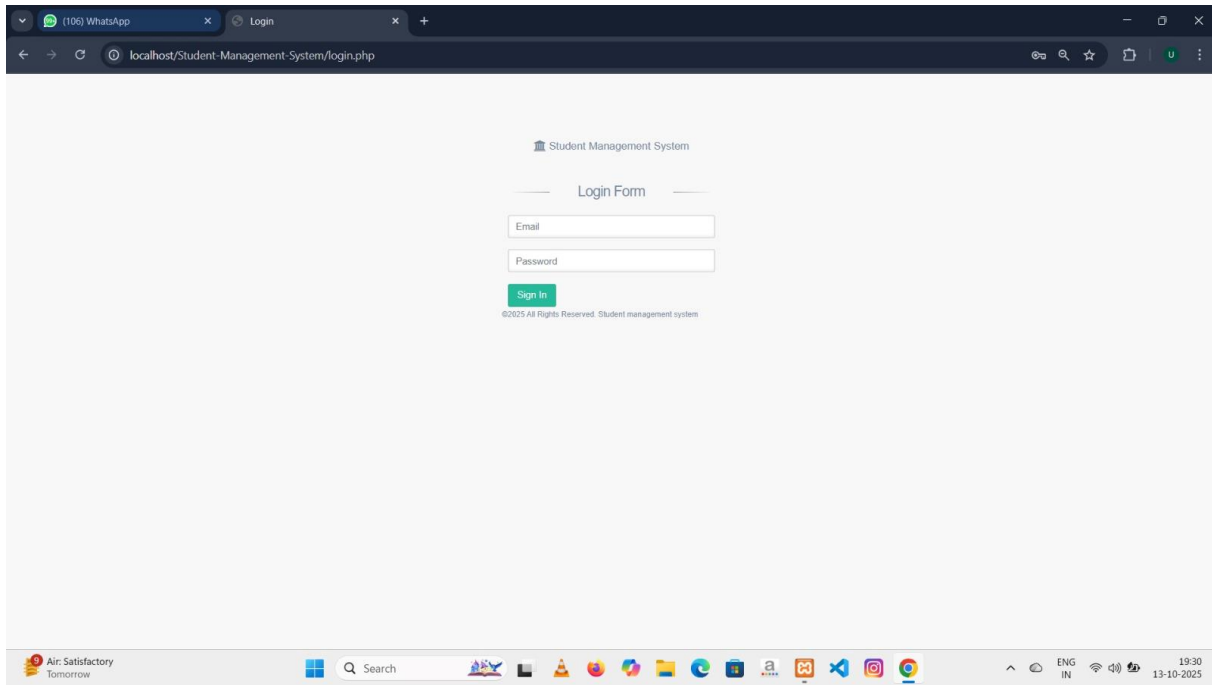
Sr. No	Module	Test Scenario	Test Steps	Result	Remarks
1	Student	View Profile	1. Login as student 2. Go to Profile section	Student's personal details displayed	Pass if correct details shown
2	Student	View Attendance	1. Login as student 2. Go to Attendance section	Attendance record of the student displayed	Pass if correct attendance shown
3	Student	View Exam Marks	1. Login as student 2. Go to Exam Results section	Exam marks uploaded by teacher displayed	Pass if correct results shown
4	Student	View Notices	1. Login as student 2. Go to Notices section	Notices uploaded by teacher/admin displayed	Pass if notices visible
5	Student	View Timetable/Schedule	1. Login as student 2. Go to Timetable section	Weekly timetable displayed	Pass if timetable correct
6	Student	Profile Update - Invalid	1. Login as student 2. Go to Profile 3. Enter invalid data 4. Submit	Error message displayed "Invalid input"	Pass if error shown correctly
7	Student	Logout	1. Click logout button	Redirected to login page	Pass if logout works

Parent

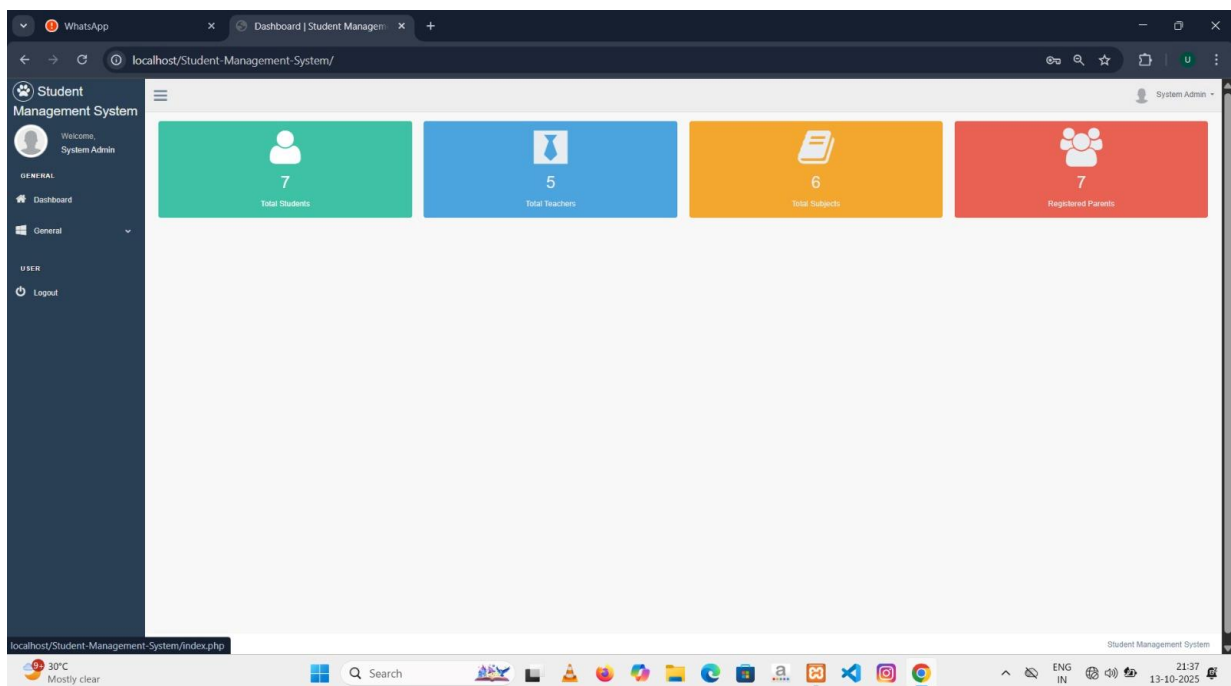
Sr. No	Module	Test Scenario	Test Steps	Result	Remarks
1	Parent	View Child Attendance	1. Login as parent 2. Go to Attendance section	Attendance record of the child displayed	Pass if correct attendance shown
2	Parent	View Child Exam Result	1. Login as parent 2. Go to Exam Results section	Exam marks of the child displayed	Pass if correct results shown
3	Parent	View Notices	1. Login as parent 2. Go to Notices section	Notices uploaded by teacher/admin displayed	Pass if notices visible
4	Parent	View Child Name	1. Login as parent 2. Go to Profile/Child Info section	Child's name and details displayed	Pass if correct child name shown
5	Parent	Profile Update - Invalid	1. Login as parent 2. Go to Profile 3. Enter invalid data 4. Submit	Error message displayed "Invalid input"	Pass if error shown correctly
6	Parent	Logout	1. Click logout button	Redirected to login page	Pass if logout works

## Snapshots

### Login page



- This is login Page



## Add teacher

The screenshot displays the 'Add teacher' page in the Student Management System admin panel. The page is divided into two main sections: a form for adding a new teacher on the left and a table of existing teachers on the right.

**Add teacher form fields:**

- Teacher ID
- First Name
- Last Name
- Date of Birth (dd-mm-yyyy)
- Gender (Male/Female)
- Email
- Address
- Contact
- Skills
- Update Teacher button

**All Teachers Table:**

TID	Name	DOB	Gender	Address	Contact	Skills	Actions
deep	Deep Hariyani	2000-10-16	Male	malyasan	98565481696	maths teacher	<a href="#">Update</a>
ENG-2	ntin trivedi	1995-02-26	Male	dhasa , botad , gujarat	9725689816	english	<a href="#">Update</a>
Hin-3	rajni maheta	1993-06-07	Male	lathi , sahanad socirty	989354589	hindi	<a href="#">Update</a>
phy-1	karadam sathvare	1982-06-28	Male	ahemdabad	97925089660	physics	<a href="#">Update</a>
ss-2	tushar sondagar	2025-09-25	Male	talaja borla	9725689816	social science	<a href="#">Update</a>

Showing 1 to 5 of 5 entries

- This is the page for add teacher in admin panel , where admin can add teacher .

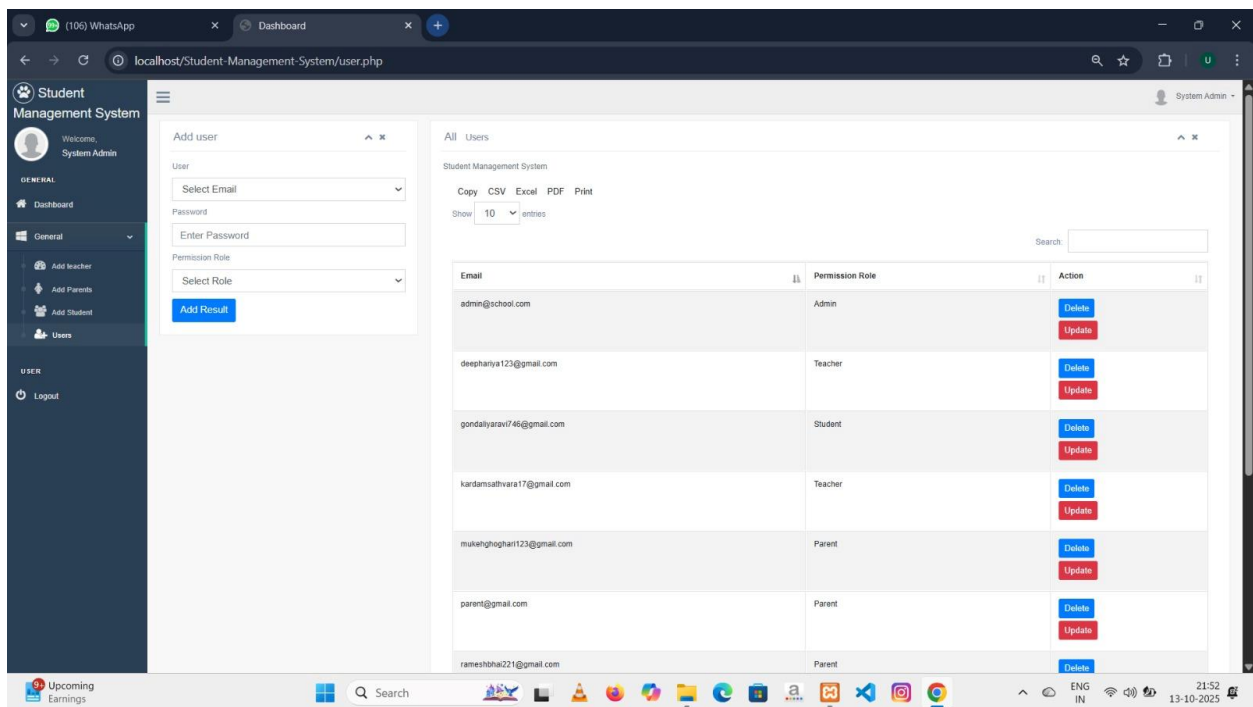
## Add Parent

The screenshot displays the 'Add Parent' interface of a Student Management System. On the left, there is a sidebar with navigation options: Dashboard, General (Add Teacher, Add Parents, Add Student, Users), and User (Logout). The main content area is divided into two sections. The left section is the 'Add parent' form, which includes input fields for First Name, Last Name, Adhar card, Gender (radio buttons for Male and Female), Email, Address, Contact, and Occupation, followed by an 'Add Parent' button. The right section, titled 'All parents', shows a table of existing parents with columns for Parent ID, Full Name, Adhar card, Gender, Address, Contact, Occupation, Students, and Actions (Update/Delete). The table contains 11 entries. Below the table, it indicates 'Showing 1 to 7 of 7 entries' and provides 'Previous' and 'Next' navigation links.

Parent ID	Full Name	Adhar card	Gender	Address	Contact	Occupation	Students	Actions
1	rameshbhai sondagar	2140063647	Male	borla talaja bhavnagar	9792566880	brick work	Students: kushar sondagar	<a href="#">Update</a> <a href="#">Delete</a>
2	rameshbhai gondaliya	2147483647	Male	Laxminagar Society Dhamel Road	9725688454	bricks work	Students: Uttam gondaliya	<a href="#">Update</a> <a href="#">Delete</a>
7	mukeshbhai ghoghari	2147483647	Male	amrell pratapara	9394689513	bricks work	Students: prince ghoghari	<a href="#">Update</a> <a href="#">Delete</a>
8	Elipinbhai Sarvaiya	2147483647	Male	kharavedha , jamnagar	9654782135	iron work	Students: Harshil Sarvaiya	<a href="#">Update</a> <a href="#">Delete</a>
9	Raghadass Hariyani	2147483647	Male	maliyasan , rajkot	9532568914	diamond worker	Students: Deep Hariyani	<a href="#">Update</a> <a href="#">Delete</a>
10	maheesh narola	66595656	Male	somnath	95889646	nil	Students: Prince narola	<a href="#">Update</a> <a href="#">Delete</a>
11	vikrambhai vadukar	56981478	Male	veraval	985623147	business owner	Students: Yug Vadukar	<a href="#">Update</a> <a href="#">Delete</a>

- This is a parent page for admin, where parent can add student data and it automatically gives name of child which is connected with child .

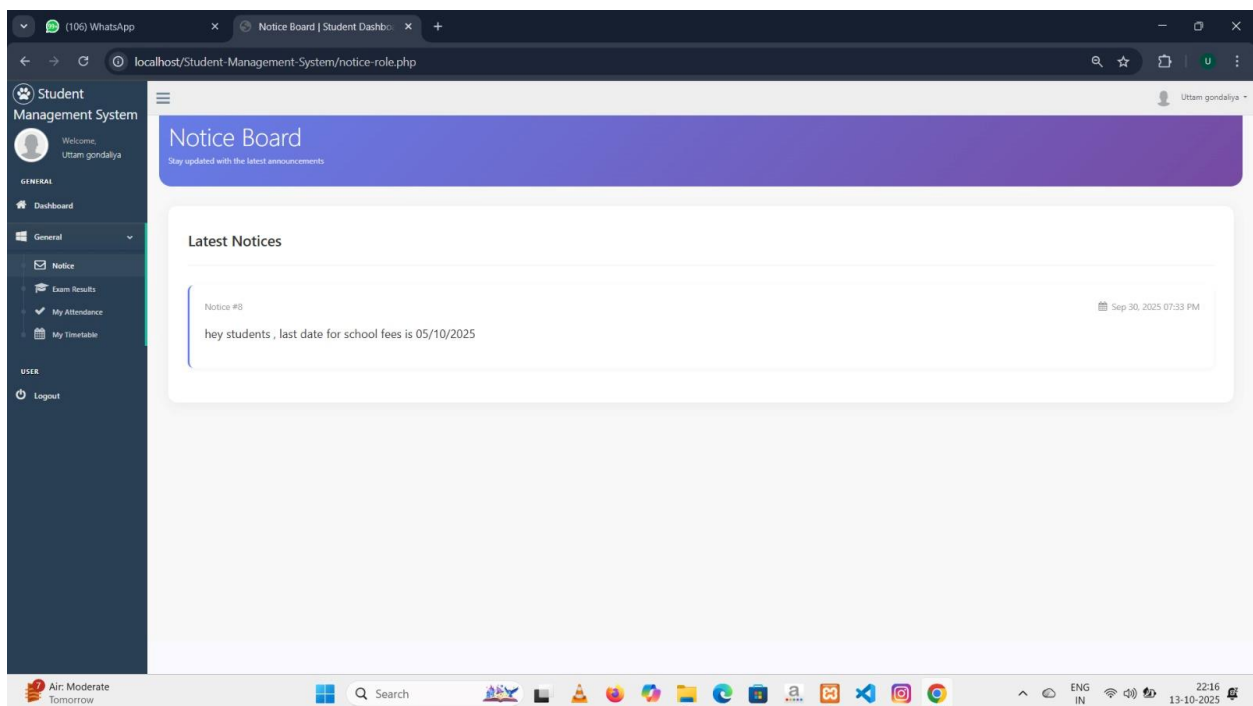
## User Role



- This is a Page of User Role where admin can set a User role for Student , Teacher and Parent so the can login into System .

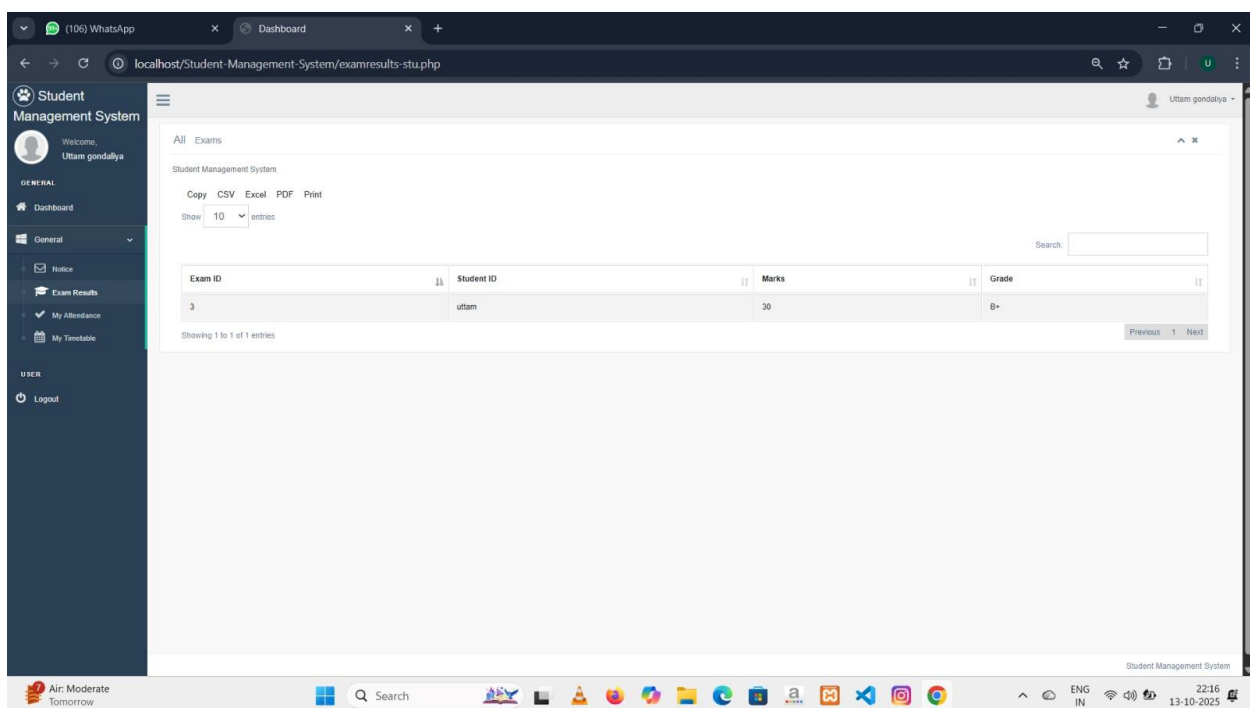


## Notice Student



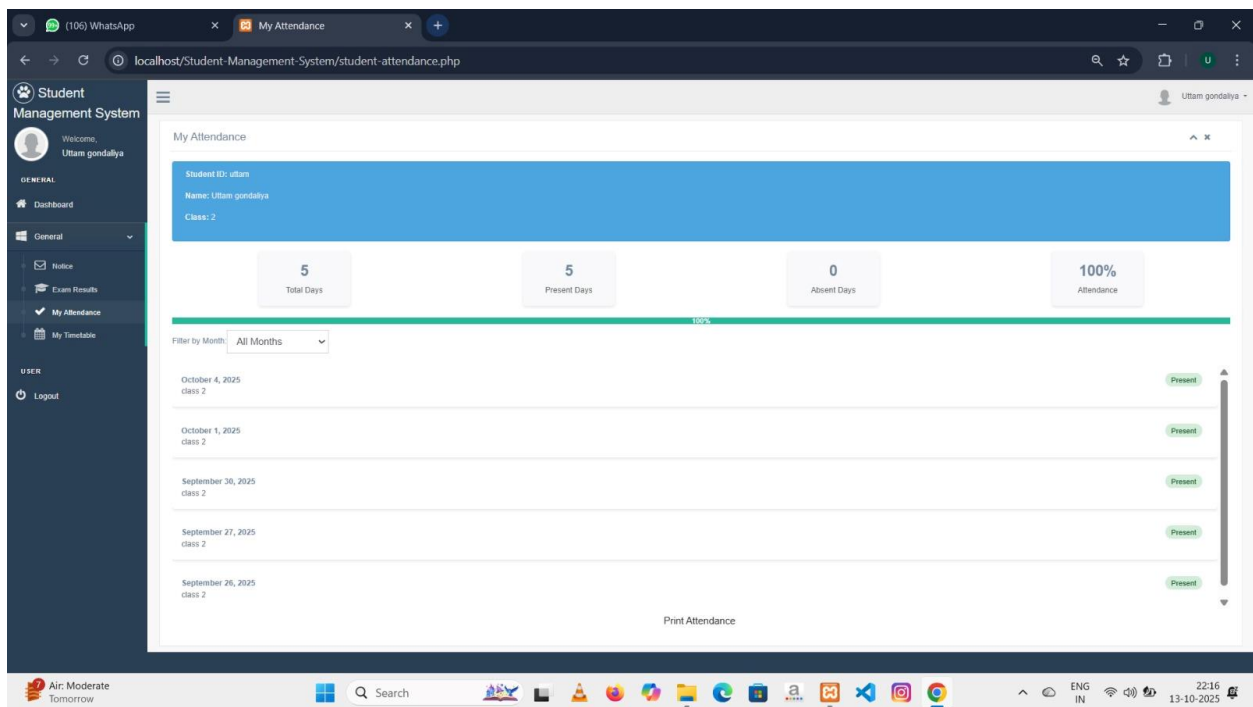
- This is Notice Section page in student panel , where student can see the important notices , which is send by teachers .

## Exam Result Student



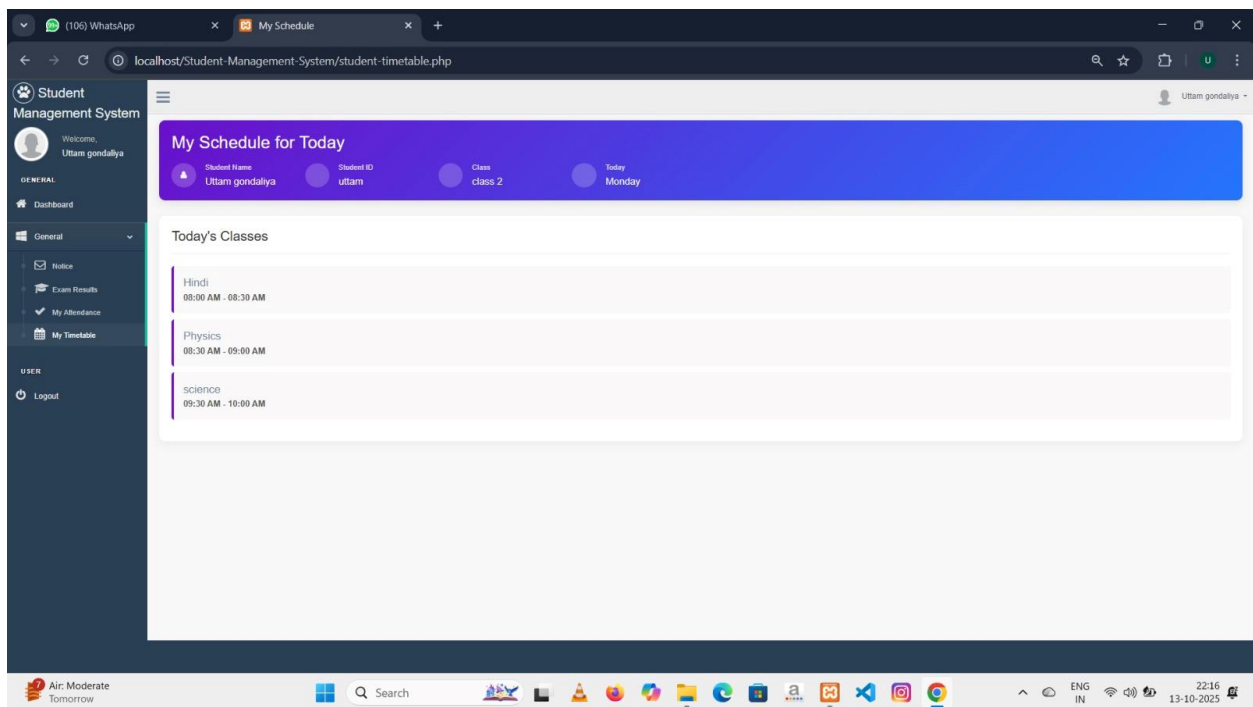
- This is a page for a Exam result show in Student p, which is Uploaded by teachers.

## Student Attendance



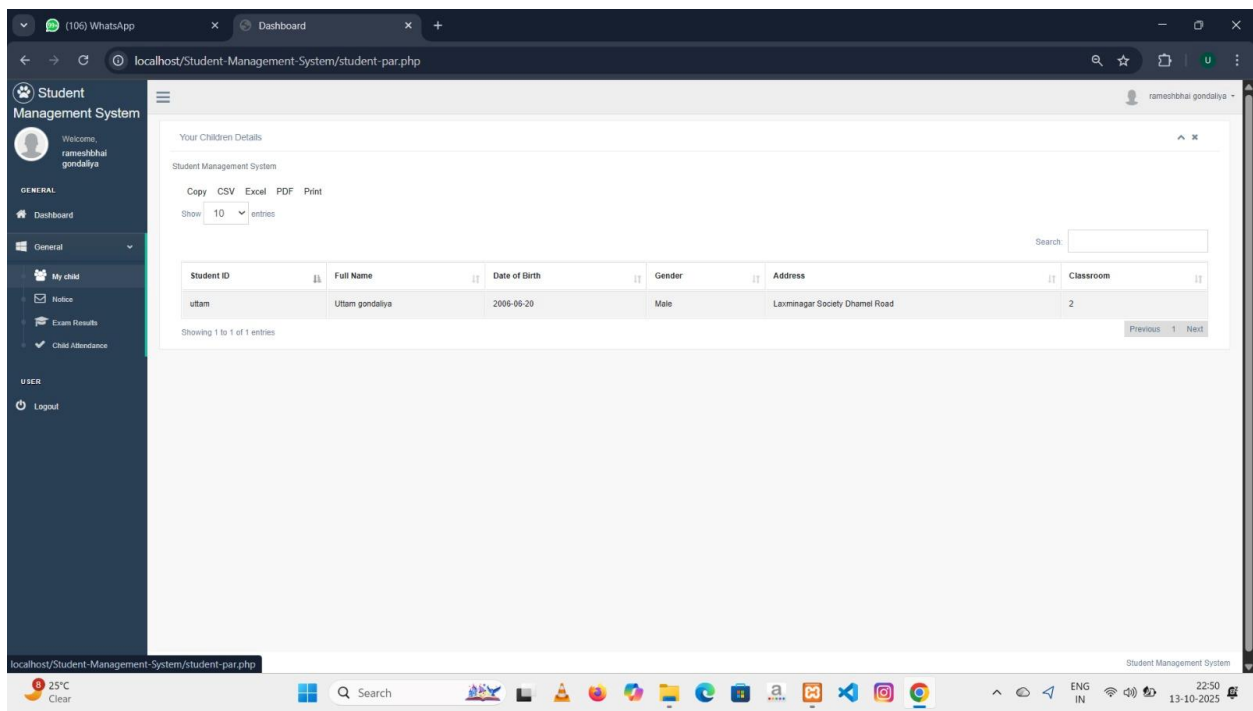
- This is a page for a student attendance, where it shows to student their attendance daily and also a monthly attendance report.
- How many days student present and how many days student absent.

## Student timetable



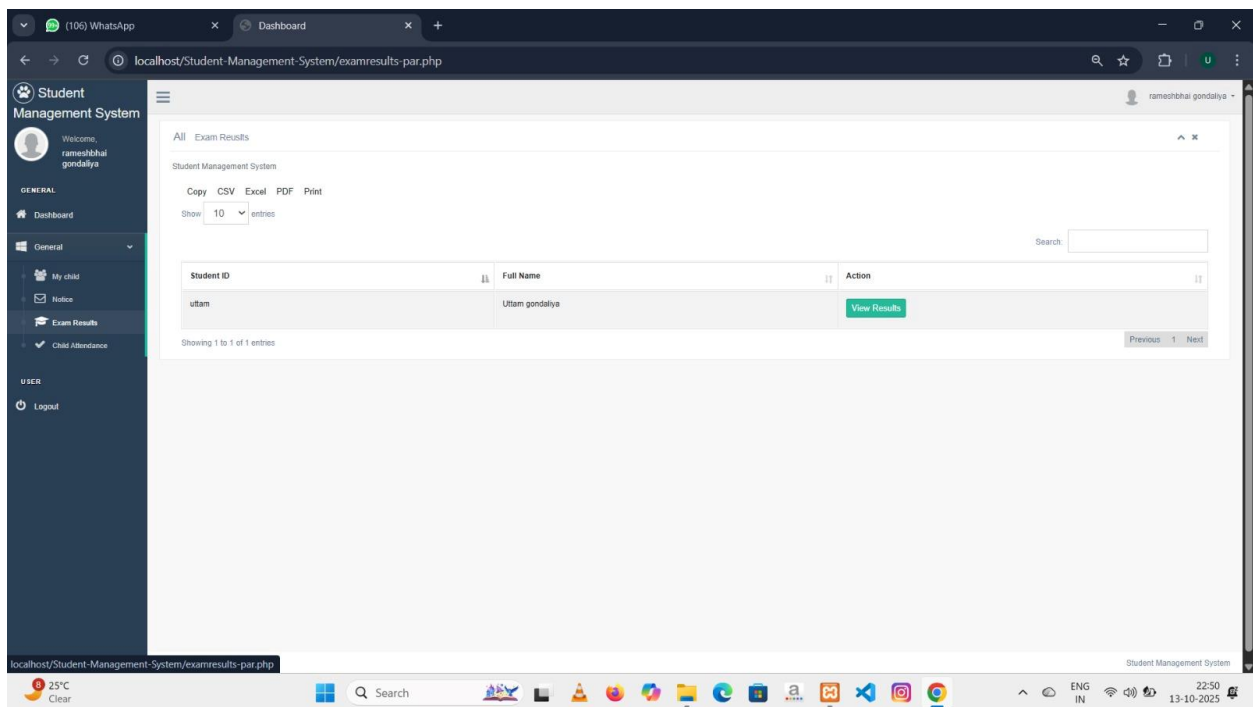
- This is a page for a student time table Where it shows regular schedule .

## Parent Child detail



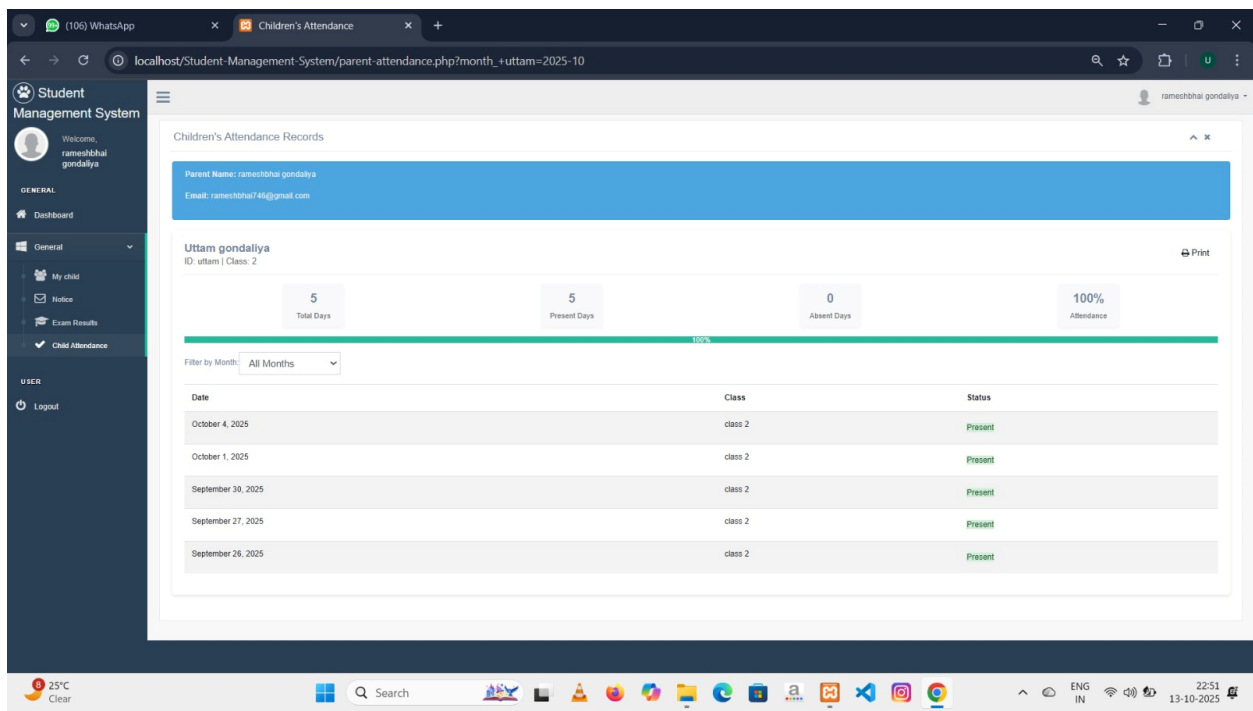
- This is a page where parents can see his child detail.

## Parent child result



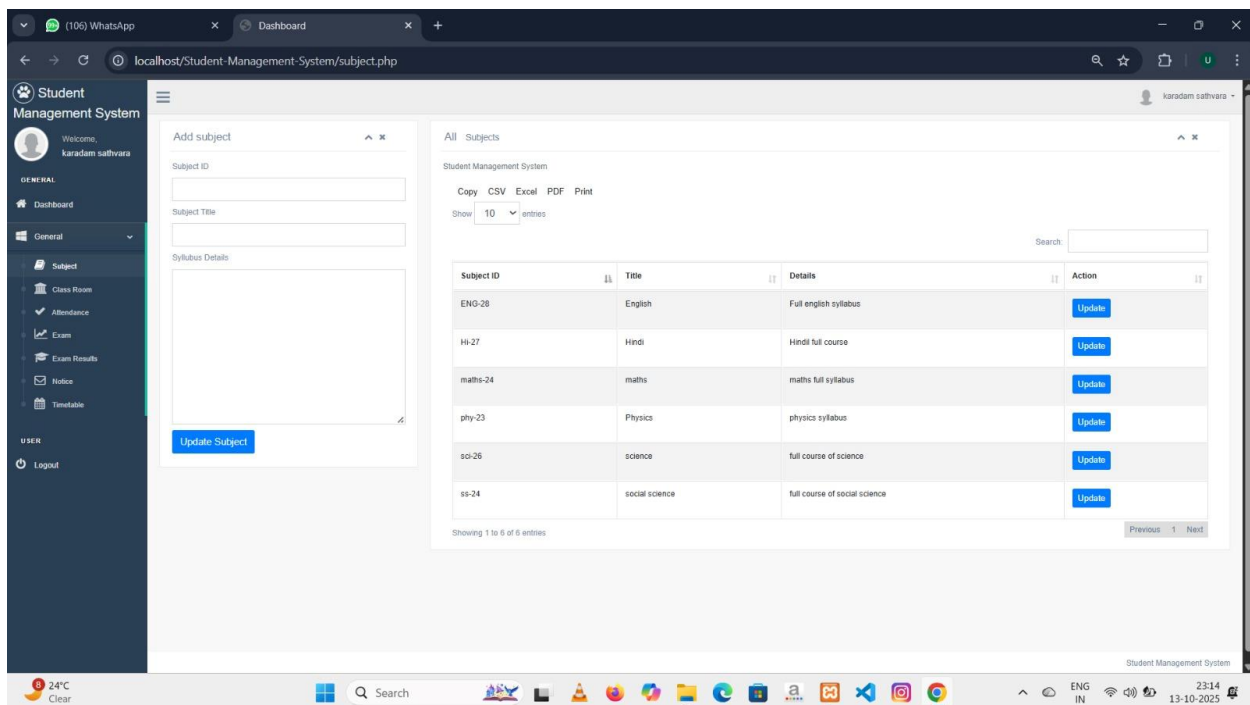
- This is the page where parents can see his child result.

## Child attendance



- This is a page where parents can see his child attendance and also can print it .

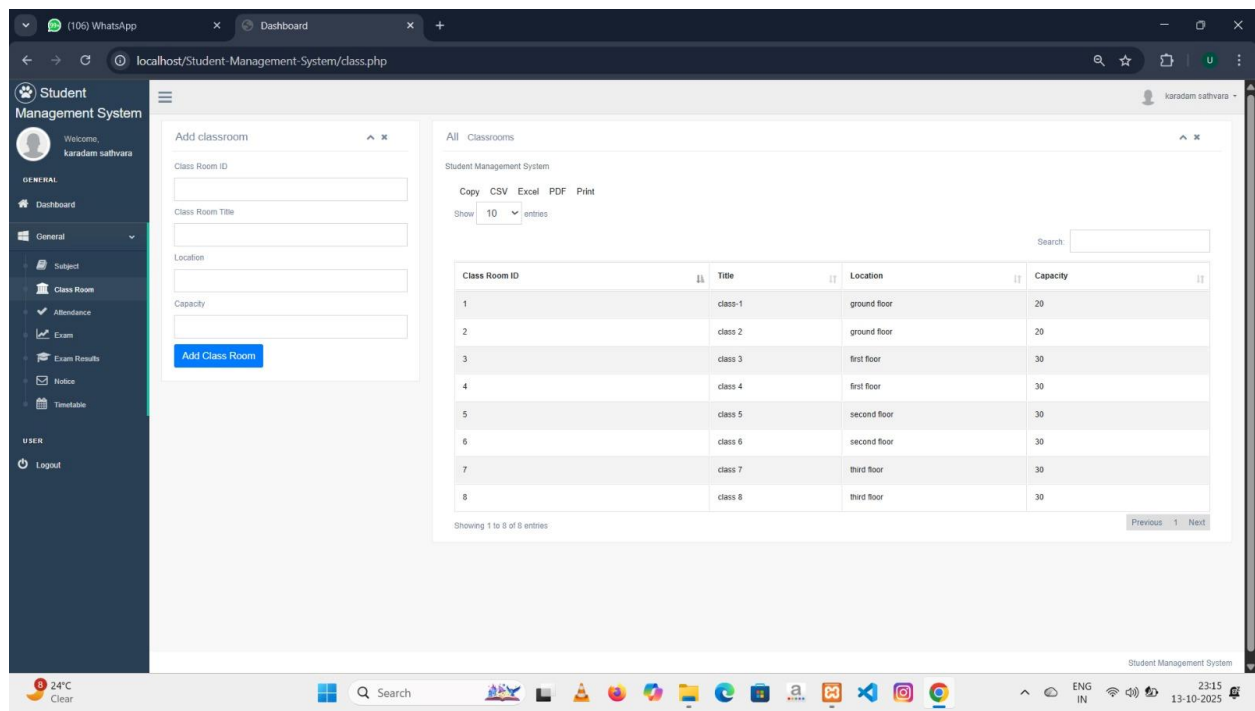
## Teacher Subject



- Teacher Can add subject from this page.

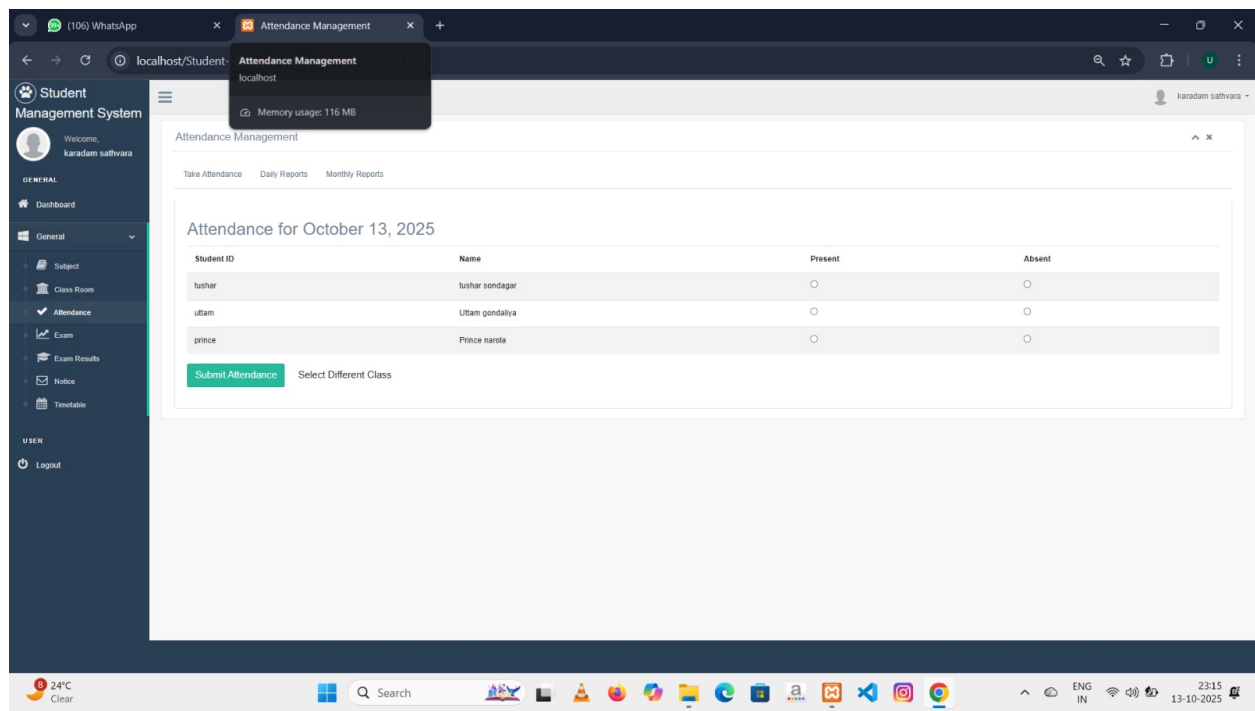


## Teacher Class



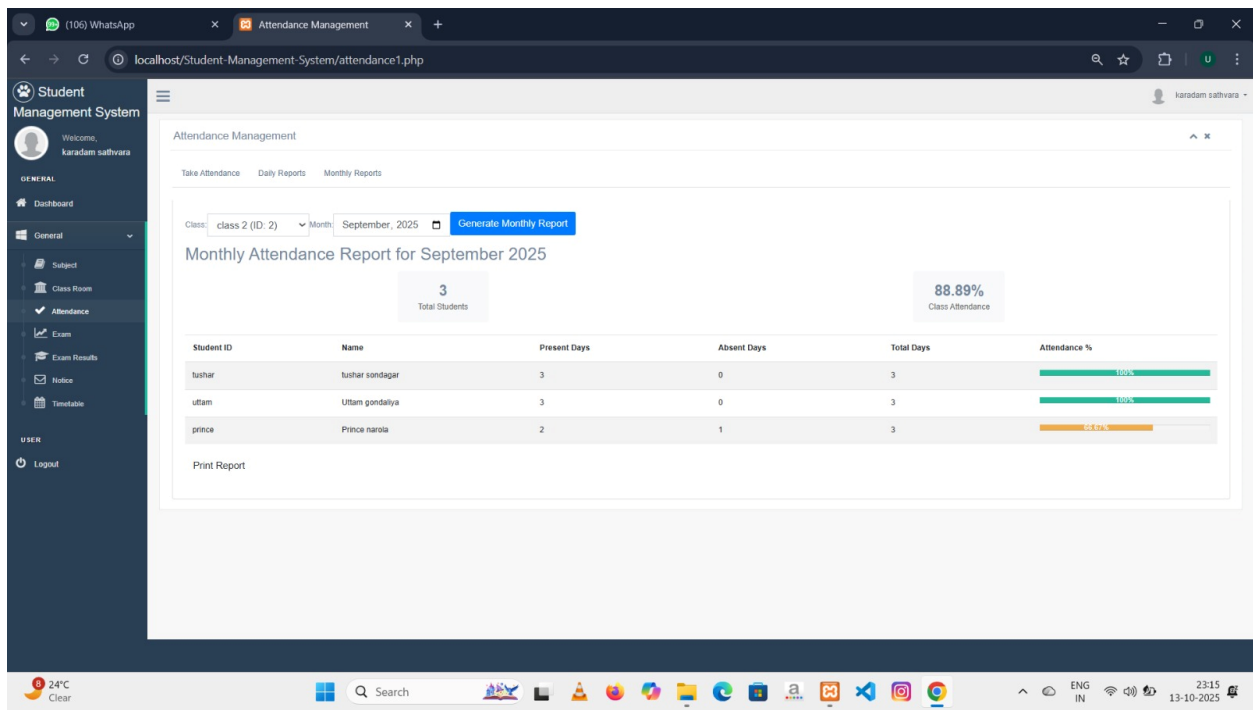
- Teacher can add Classes from this page.

## Teacher Attendance



- This is a page where teacher first select the class then it shows the student of that class with present absent option then teacher select in option then submit .

## Attendance Report



- This is a page where it shows monthly attendance report .

## Exam Set

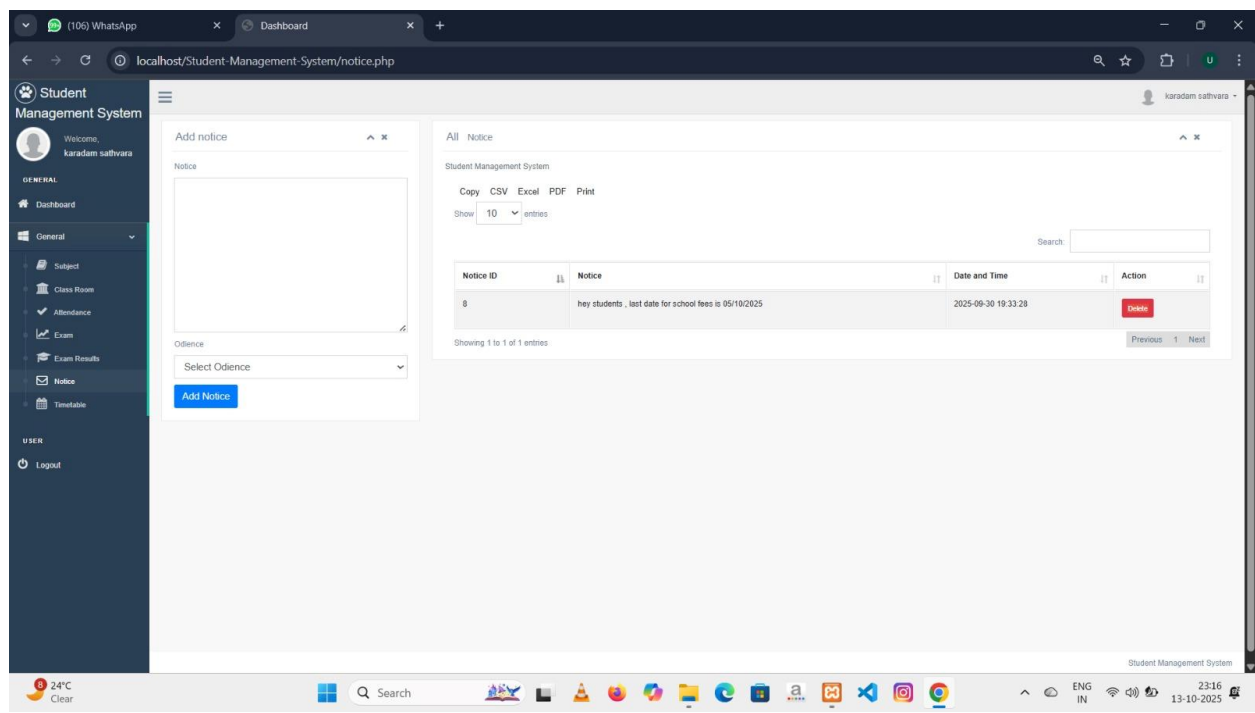
The screenshot displays the 'Exam Set' interface of a Student Management System. On the left is a dark sidebar with a 'Student Management System' header and a user profile for 'karadam sahvira'. Below this is a 'GENERAL' section with a 'Dashboard' link and a 'General' dropdown menu containing 'Subject', 'Class Room', 'Attendance', 'Exam', 'Exam Results', 'Notice', and 'Timetable'. The 'Exam' option is currently selected. The main content area is titled 'Add exam' and contains several form fields: 'Subject' (a dropdown menu), 'Exam Hall (Class Room)' (a dropdown menu), 'Teacher in Charge' (a dropdown menu), 'Date' (a date picker), 'Start Time' (a time picker), and 'End Time' (a time picker). An 'Add Exam' button is located at the bottom of this form. To the right of the form is a section titled 'All Exams' which includes a 'Student Management System' header, a 'Copy CSV Excel PDF Print' menu, a 'Show 10 entries' dropdown, and a search bar. Below these is a table with the following data:

Exam ID	Subject	Teacher in Charge	Location (Classroom)	Date	Start Time	End Time
3	phy-23	phy-1	2-B	2006-06-20	10:00:00	11:00:00

Below the table, it says 'Showing 1 to 1 of 1 entries' and there are 'Previous' and 'Next' navigation buttons. The bottom of the page shows a Windows taskbar with a search bar, various application icons, and a system tray displaying '24°C Clear', 'ENG IN', and the date '13-10-2025'.

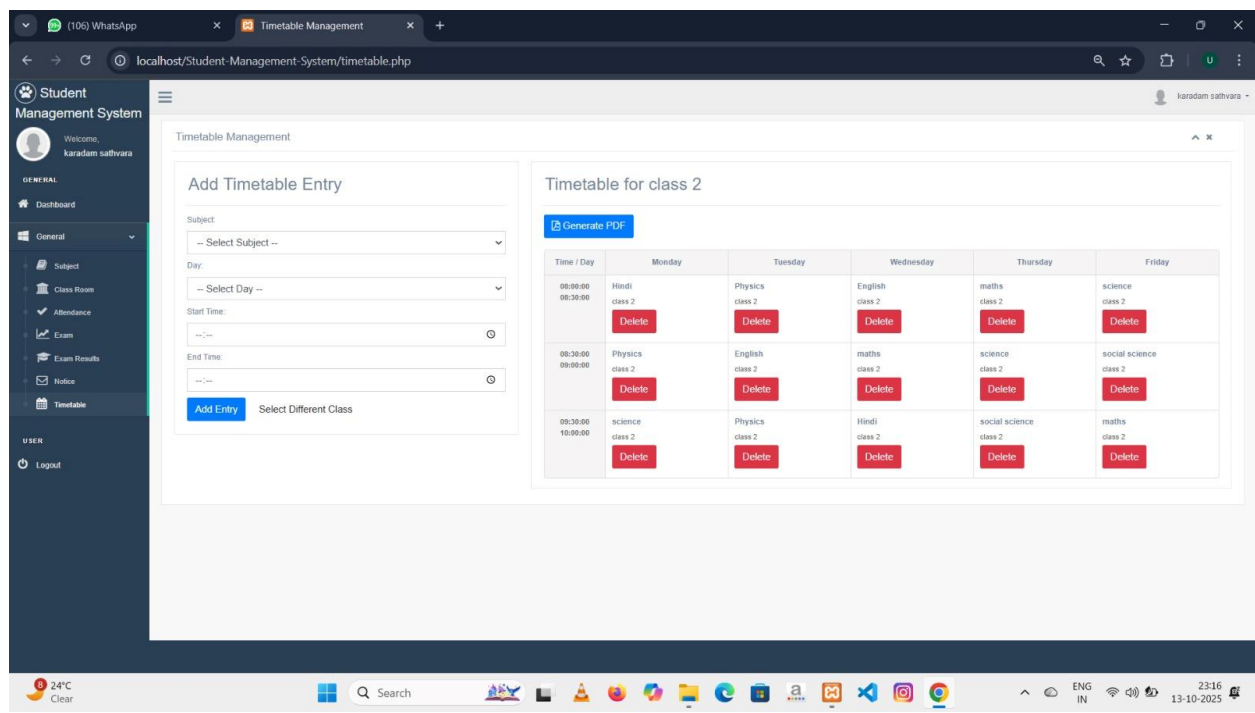
- Teacher can set Exam (Test) from this page .

## Exam Result



- This is a page where teacher can sent a notice to Student and Parents .

## Timetable



- This is the page where teacher can set a timetable for different class.
- First of all dropdown menu of class come then select class and then click submit button then create a time table.
- You can also generate timetable print.

## Limitations

Though we tried our best in developing this software but as limitation are major part of any software so are of our software. Some limitation of innovator to investors portal are as follows:

- Limited functionality is available.
- If database gets big amount of data, it gets much time to load.
- Specific user roles are not provided in this software. That's why this can only be used by admin.

## Future Enhancement

In the future, the **Student Management System** can be enhanced by integrating advanced technologies such as Artificial Intelligence (AI) and Data Analytics to automate attendance tracking, performance evaluation, and personalized learning recommendations. Cloud integration can be implemented to provide real-time access to data for students, teachers, and parents from anywhere. Additionally, features like biometric authentication, chatbot support for queries, and mobile application compatibility can improve user experience and system efficiency. These enhancements will make the system more intelligent, secure, and user-friendly for all stakeholders.

## Conclusion

### Conclusion:

The **Student Management System** project successfully provides a digital solution for managing student-related information efficiently within an educational institution. It helps administrators easily handle student, teacher, and parent records through a centralized platform. The system automates key processes such as registration, attendance, examination management, and communication, thereby reducing manual work and improving accuracy. It ensures that data is well-organized and accessible, making the overall management process more efficient and transparent.

Overall, this project demonstrates how technology can simplify educational administration. By using PHP and MySQL, the system offers reliability, flexibility, and scalability for future improvements. It not only benefits the administration but also creates better communication between teachers, students, and parents. With future enhancements like mobile app support, AI-based analytics, and cloud integration, this project can evolve into a complete digital ecosystem for modern educational management.