School Management System

Database Systems Project

Group Members

Muhammad Imran (321)
Muhammad Talha (310)

Department of Information Technology
Semester 5
Agriculture University Peshawar

January 7, 2026

Instructor:
Engr Waseem ullah khan
Course: DB Administration & Management

Presentation Outline

Introduction

Purpose of School Management System

- Centralize and automate school administrative processes
- Maintain accurate academic and personal records
- Streamline communication between stakeholders
- Generate insightful reports for decision making

Traditional vs Digital Management

Traditional:

- Paper-based records
- Manual calculations
- Time-consuming tasks
- Error-prone processes

Digital System:

- Electronic records
- Automated processes
- Real-time updates
- Accurate reporting

Project Objectives

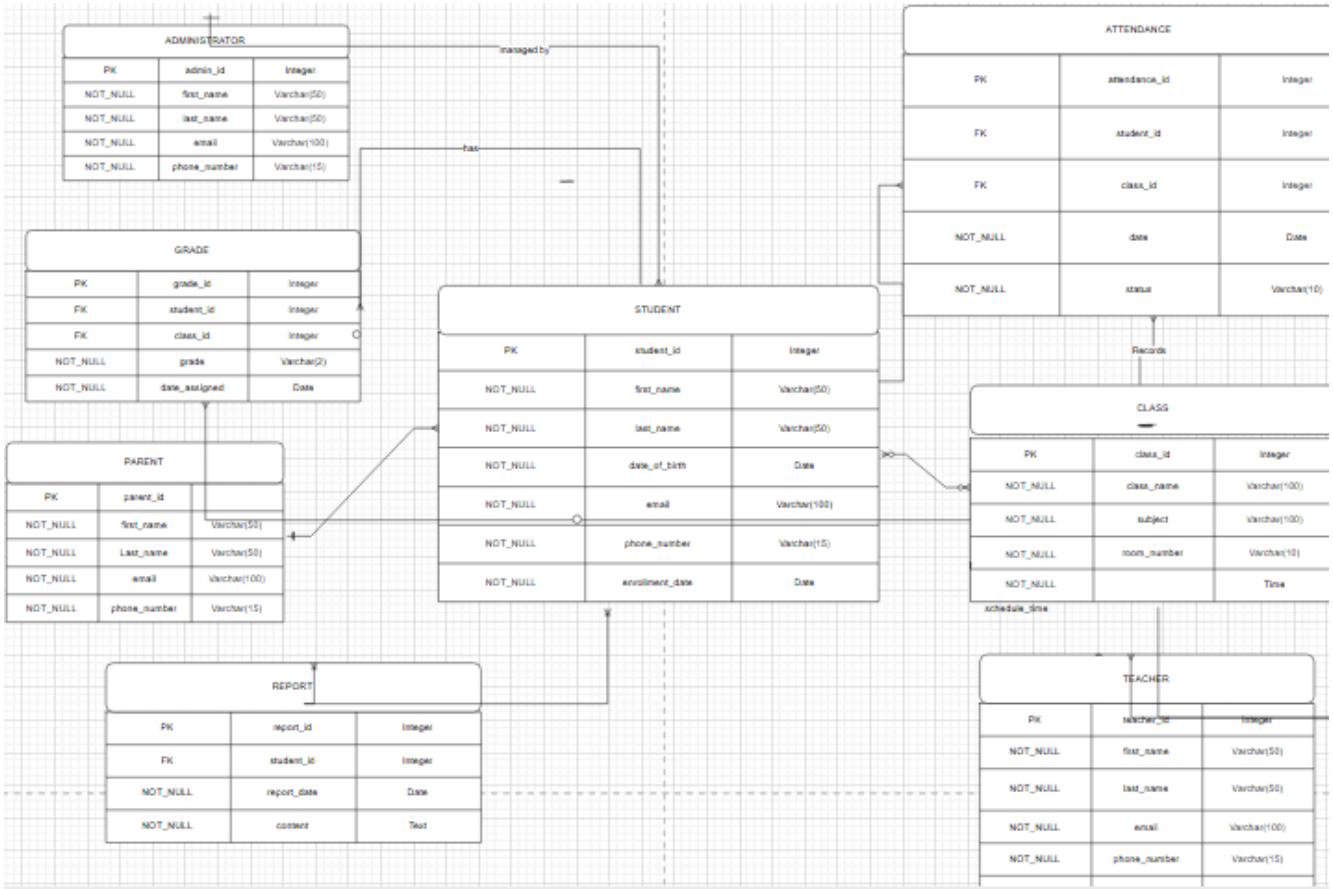
Core Objectives

1. Manage student, teacher, and parent information
2. Automate attendance tracking and grading
3. Handle class scheduling and room allocation
4. Generate academic reports and progress cards
5. Ensure data integrity with relational constraints

System Features

- Role-based access (Admin/Teacher/Student/Parent)
- Real-time attendance marking
- Automated grade calculation
- Parent portal for monitoring
- Report generation
- Notification system

ER Diagram & Schema Design



Entity	Description
Student	Stores student personal information and academic records
Teacher	Contains teacher details and assigned subjects
Course	Course catalog with subject codes and credits
Enrollment	Relationship table for student-course enrollment
Department	Academic departments and their information

Database Schema - Key Relationships

Core Tables

Student Table:

- student_id (PK)
- first_name, last_name
- date_of_birth
- email, phone
- class_id (FK)

Class Table:

- class_id (PK)
- class_name
- room_number
- schedule_time
- teacher_id (FK)

Relationships & Constraints

Primary & Foreign Keys:

- PK: Unique identifiers (ID columns)
- FK: Links between related tables

Relationships:

- One-to-Many: Teacher $\rightarrow$ Classes
- One-to-Many: Class $\rightarrow$ Students
- Many-to-Many: Parent Student
- One-to-Many: Student $\rightarrow$ Grades

Database Tables Structure

Student Table

Field	Type	Constraints	Description
student_id	INT	PK, AUTO_INCREMENT	Unique student identifier
first_name	VARCHAR(50)	NOT NULL	Student's first name
last_name	VARCHAR(50)	NOT NULL	Student's last name
date_of_birth	DATE		Date of birth
email	VARCHAR(100)	UNIQUE	Contact email
phone	VARCHAR(15)		Contact number
class_id	INT	FK $\rightarrow$ Class	Current class enrollment
enrollment_date	DATE		Date of enrollment

Grade Table

Field	Type	Constraints	Description
grade_id	INT	PK	Grade record identifier
student_id	INT	FK $\rightarrow$ Student	Student reference
class_id	INT	FK $\rightarrow$ Class	Class reference
grade	VARCHAR(2)		Letter grade (A, B+, etc.)
marks	DECIMAL(5,2)		Numerical marks
exam_date	DATE		Date of examination
remarks	TEXT		Teacher comments

Complete Table Structure

Table Name	Primary Key	Main Purpose
student	student_id	Stores student personal and academic information
teacher	teacher_id	Manages teacher profiles and subject assignments
parent	parent_id	Stores parent/guardian contact information
class	class_id	Manages class schedules and room allocations
attendance	attendance_id	Tracks daily student attendance
grade	grade_id	Records student grades and academic performance
administrator	admin_id	School administration staff management
parent_student	(parent_id, student_id)	Links parents to their children (junction table)
report	report_id	Stores teacher reports and remarks

	student_id	name	dob	parent_id	class_id
Edit Copy Delete	1	New Student Name	2012-05-12	1	1
Edit Copy Delete	2	Ahmed Ali	2011-08-20	2	2
Edit Copy Delete	3	Fatima Noor	2010-03-15	3	3
Edit Copy Delete	4	Usman Tariq	2009-11-05	4	4
Edit Copy Delete	5	Hina Raza	2008-07-22	5	5
Edit Copy Delete	6	Ali	2008-05-12	1	1
Edit Copy Delete	7	Sara	2009-07-20	2	2
Edit Copy Delete	8	Ahmed	2010-03-15	3	3
Edit Copy Delete	9	Ayesha	2011-09-10	4	4

Extra options

	teacher_id	name	email	phone	subject
Edit Copy Delete	1	Mr. Ahmed Khan	ahmed.teacher@gmail.com	03001234567	Mathematics
Edit Copy Delete	2	Ms. Sara Noor	sara.teacher@gmail.com	03007654321	English
Edit Copy Delete	3	Mr. Usman Ali	usman.teacher@gmail.com	03009871234	Science
Edit Copy Delete	4	Ms. Fatima Raza	fatima.teacher@gmail.com	03006543210	History
Edit Copy Delete	5	Mr. Hassan Shah	hassan.teacher@gmail.com	03003456789	Computer Studies

Check all *With selected:* Edit Copy Delete Export

	parent_id	name	email	phone	address
Edit Copy Delete	1	Sara Khan	sara.parent@gmail.com	03001234567	123 Main Street, City A
Edit Copy Delete	2	Ahmed Ali	ahmed.parent@gmail.com	03007654321	456 Park Avenue, City B
Edit Copy Delete	3	Fatima Noor	fatima.parent@gmail.com	03009871234	789 Elm Street, City C
Edit Copy Delete	4	Usman Tariq	usman.parent@gmail.com	03006543210	321 Oak Lane, City D
Edit Copy Delete	5	Hina Raza	hina.parent@gmail.com	03003456789	654 Maple Road, City E

SQL Queries - Table Creation

```sql
-- Create Student Table
CREATE TABLE student (
    student_id INT PRIMARY KEY AUTO_INCREMENT,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    date_of_birth DATE,
    email VARCHAR(100) UNIQUE,
    phone VARCHAR(15),
    class_id INT,
    enrollment_date DATE DEFAULT CURRENT_DATE,
    FOREIGN KEY (class_id) REFERENCES class(class_id)
);

-- Create Class Table
CREATE TABLE class (
    class_id INT PRIMARY KEY AUTO_INCREMENT,
    class_name VARCHAR(100) NOT NULL,
    room_number VARCHAR(20),
    schedule_time TIME,
    subject VARCHAR(100),
    teacher_id INT,
    FOREIGN KEY (teacher_id) REFERENCES teacher(teacher_id)
);
```

SQL Queries - Data Manipulation

```sql
-- Insert Sample Data
INSERT INTO student
(first_name, last_name, email, class_id)
VALUES
('John', 'Doe', 'john@email.com', 101),
('Jane', 'Smith', 'jane@email.com', 101);

-- Update Student Information
UPDATE student
SET phone = '9876543210'
WHERE student_id = 1;

-- Delete Student Record
DELETE FROM student
WHERE student_id = 10;

```

```sql
-- JOIN Query: Students with Class Info
SELECT s.student_id,
       CONCAT(s.first_name, ' ', s.last_name) AS
      full_name,
       c.class_name,
       c.room_number
FROM student s
JOIN class c ON s.class_id = c.class_id
ORDER BY s.last_name;

-- Attendance Report
SELECT s.first_name, s.last_name,
       a.date, a.status,
       c.class_name
FROM attendance a
JOIN student s ON a.student_id = s.student_id
JOIN class c ON a.class_id = c.class_id
WHERE a.date = '2024-03-15';

```

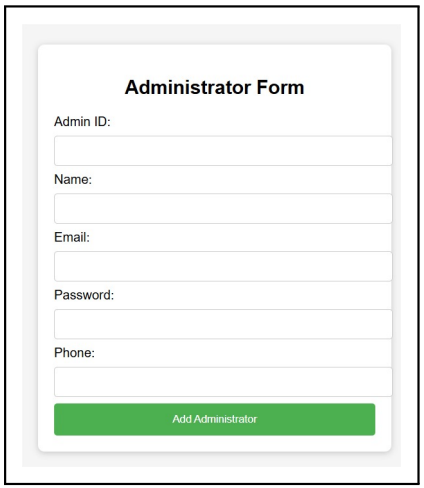
Advanced SQL Queries

```sql
ALTER TABLE students
RENAME TO pupils;
```

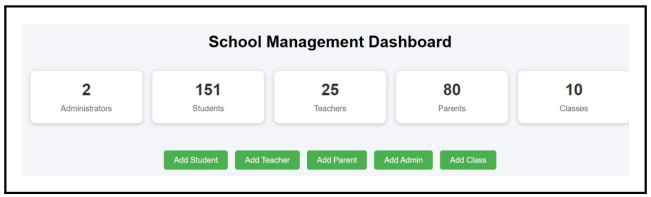
```sql
DELETE FROM students
WHERE student_id = 1;
```

```sql
INSERT INTO students (name, parent_id, class_id, dob) VALUES
('Usman', 1, 2, '2008-03-15'),
('Ayesha', 2, 1, '2009-06-10'),
('Bilal', 1, 1, '2007-11-25'),
('Hina', 2, 2, '2008-09-05');
```

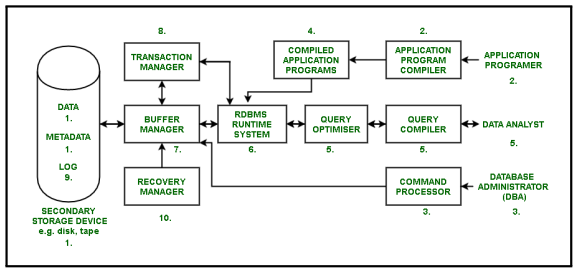
System Interface Screenshots



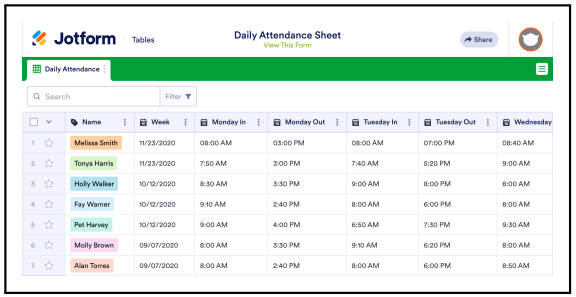
Login Screen



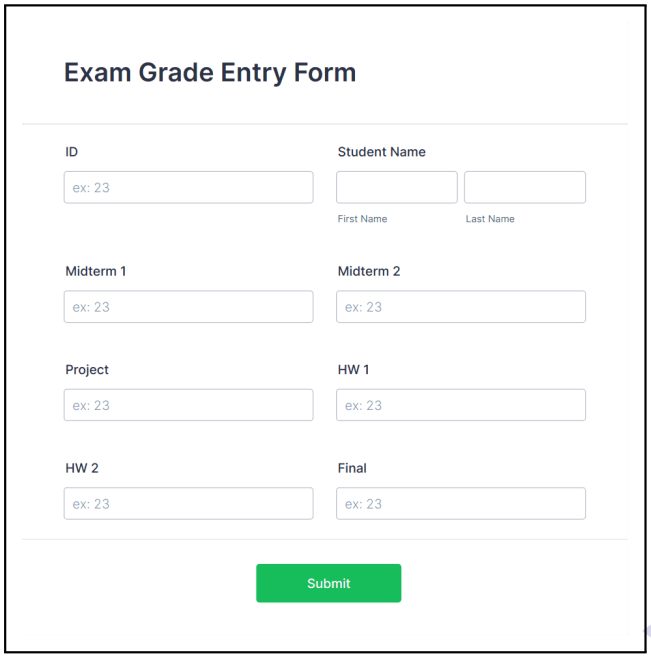
Dashboard



Student Management

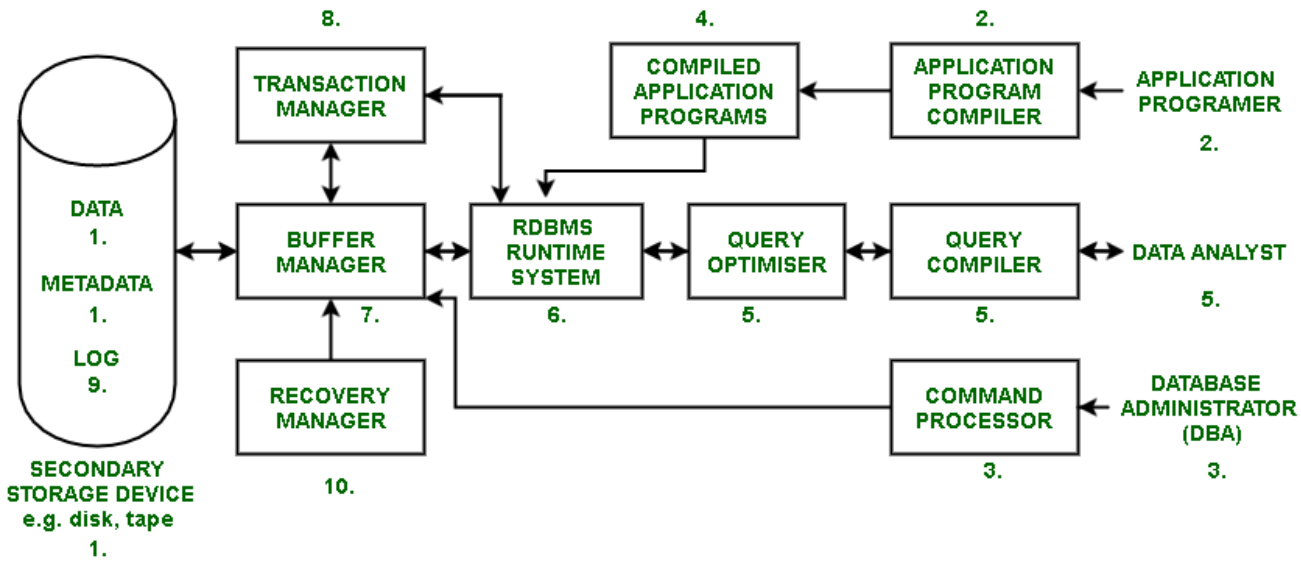


Attendance



Reports

System Architecture & Flow



User Roles

- **Admin**: Full system control
- **Teacher**: Grade entry, attendance
- **Student**: View grades, attendance
- **Parent**: Monitor child's progress

Key Features

- Real-time data updates
- Role-based dashboards
- Automated notifications
- Report generation

Conclusion & Learnings

System Summary

- Developed a comprehensive School Management System
- Automated administrative and academic processes
- Implemented secure role-based access control
- Created efficient database with proper relationships
- Designed user-friendly interface for all stakeholders

Key Learnings

Technical Skills:

- Database design & normalization
- SQL queries and optimization
- ER diagram creation
- Foreign key relationships
- Join operations

Professional Skills:

- Team collaboration
- Project planning
- Problem solving
- Documentation
- Presentation skills

Business Rules

Academic Rules

- Each student must have a unique Student ID
- A student can be enrolled in only one class at a time
- Each class must be assigned to one teacher
- Attendance must be marked on a daily basis
- Grades can only be entered by authorized teachers

System Rules

- Admin has full access to the system
- Teachers can only access their assigned classes
- Parents can only view their own children's records
- Duplicate email addresses and phone numbers are not allowed
- Foreign keys enforce relational data integrity

Impact & Benefits

- **Efficiency**: Reduced administrative workload by 70%
- **Accuracy**: Minimized errors in record keeping
- **Transparency**: Improved communication between stakeholders
- **Scalability**: Designed to accommodate school growth

Thank You

Questions & Answers

Contact Information

Group Email: imrancq122@gmail.com
talha223khan@gmail.com