**VISVESVARAYA TECHNOLOGICAL UNIVERSITY “JNANA SANGAMA”, BELAGAVI - 590 018**



A MINI PROJECT REPORT

on

### “MINI PROJECT TITLE”

***submitted by***

**Prateek Deshbhandari 4SF23IS407**

**Prajwal naik 4SF23IS407**

***In partial fulfillment of the requirements for the V semester***

### DBMS Attendance Management System

of

**BACHELOR OF ENGINEERING**

**in**

**INFORMATION SCIENCE & ENGINEERING**

***under the guidance of***

**Mr. Vasudeva Rao P V Assistant Professor Department of ISE**

at



## SAHYADRI

**College of Engineering & Management An Autonomous Institution Adyar, Mangaluru - 575 007**

**AY: 2024 - 25**

## SAHYADRI

### College of Engineering & Management

**An Autonomous Institution Adyar, Mangaluru - 575 007**

**Department of Information Science & Engineering**



### CERTIFICATE

This is to certify that the **Mini Project** entitled **“Mini Project Title”** has been carried out by **Student Name 1 (Prateek Deshbhandari)** and **Student Name 2 (Prajeal Naik)**, the bonafide students of Sahyadri College of Engineering & Management in partial fulfillment of the requirements for the V semester **DBMS Laboratory with Mini Project (CS522L5C)** of **Information Science & Engineering** of Visvesvaraya Technological University, Belagavi during the Academic Year 2024 - 25. It is certified that all corrections/suggestions indicated for Continuous Internal Assessment have been incorporated in the report deposited in the departmental library of Information Science & Engineering. The mini project report has been approved as it satisfies the academic requirements in respect of mini project work.

———————————– ———————————

Mr. Vasudeva Rao P V Dr. Rithesh Pakkala P.

Assistant Professor Assoc.Professor & Head Dept. of ISE, SCEM Dept. of ISE & CSE(DS), SCEM

**External Practical Examination:**

Examiner’s Name Signature with Date

1. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

## SAHYADRI

### College of Engineering & Management

**An Autonomous Institution Adyar, Mangaluru - 575 007**

**Department of Information Science & Engineering**



### DECLARATION

We hereby declare that the entire work embodied in this Mini Project Report titled **“Mini Project Title”** has been carried out by us at Sahyadri College of Engineering & Management, Mangaluru under the supervision of **Mr. Vasudeva Rao P V** as the part of the V semester **DBMS Attendance Management System Mini Project (CS522L5C)** of **Bachelor of Engineering** in **Information Science & Engineering**. This report has not been submitted to this or any other university.

Student Name 1 (4SF23IS407) Student Name 2 (4SF23IS407)

SCEM, Mangaluru

# Abstract

The Attendance Management System is a software application designed to streamline and automate the process of recording, tracking, and managing attendance for students or employees. It provides a user-friendly interface for instructors, managers, or administrators to efficiently monitor attendance records in real time. The system allows users to register attendance data, generate reports, and maintain accurate attendance history. Features such as automatic calculation of attendance percentages, customizable reporting, and integration with other systems (like payroll or grading) enhance its func- tionality. The system is designed to improve efficiency, reduce errors, and save time compared to manual attendance methods. Additionally, it offers enhanced data security and easy access to historical records for analysis and decision-making. By automating routine tasks, the Attendance Management System ensures greater accuracy, transparency, and accountability in atten- dance tracking The system also supports features like alerts for absenteeism, notifying administrators and users of irregular attendance patterns. Atten- dance Management System enhances productivity, ensures accuracy, and sim- plifies attendance tracking for any organization.

# Acknowledgement

It is with great satisfaction and euphoria that we are submitting the Mini Project Re- port on **“Mini Project Title”**. We have completed it as a part of the V semester **DBMS Attendance Management System Project ()** of **Bachelor of Engineer- ing** in **Information Science & Engineering** of Visvesvaraya Technological University, Belagavi.

We are profoundly indebted to our guide, **Mr. Vasudeva Rao P V**, Assistant Professor, Department of Information Science & Engineering (ISE) for innumerable acts of timely advice, encouragement and We sincerely express our gratitude.

We express our sincere gratitude to **Dr. Rithesh Pakkala P.**, Assoc. Professor & Head, Department of ISE & CSE(DS) for his invaluable support and guidance.

We sincerely thank **Dr. Sudheer Shetty**, Professor, ISE & Vice Principal, Sahyadri College of Engineering & Management, who has constantly motivated us throughout the project work.

We sincerely thank **Dr. S S Injaganeri**, Principal, Sahyadri College of Engineering & Management who have always been a great source of inspiration.

We further thank the non-teaching staff members of the Department of ISE, who have provided necessary support during the course of the work.

Finally, yet importantly, We express our heartfelt thanks to our family & friends for their wishes and encouragement throughout the work.

Student Name 1 (4SF23IS407) Student Name 2 (4SF23IS404)

# Table of Contents

|  |  |
| --- | --- |
| **Abstract**  **Acknowledgement Table of Contents List of Figures** | **i**  **ii iv v** |
| **1 Introduction** | **1** |
| 1.1 Database . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 1 |
| 1.2 Database Management System(DBMS) . . . . . . . . . . . . . . . . . . . | 1 |
| 1.3 Characteristics of the Database Approach . . . . . . . . . . . . . . . . . | 2 |
| 1.4 Advantages of using the DBMS Approach . . . . . . . . . . . . . . . . . | 2 |
| 1.5 Schemas . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 2 |
| 1.6 DBMS Languages . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 2 |
| 1.7 Motivational challenges for the project work . . . . . . . . . . . . . . . . | 3 |
| 1.8 Application of the Project in Real time . . . . . . . . . . . . . . . . . . . | 3 |
| **2 Conceptual Data Modeling** | **4** |
| 2.1 Entity Relationship (ER) Model . . . . . . . . . . . . . . . . . . . . . . . | 4 |
| 2.2 Entities . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 4 |
| 2.3 Attributes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 5 |
| 2.4 Relationships . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 5 |
| 2.5 Notation for ER Diagrams . . . . . . . . . . . . . . . . . . . . . . . . . . | 6 |
| **3 Relational Data Model** | **7** |
| 3.1 Relational Model Concepts . . . . . . . . . . . . . . . . . . . . . . . . . . | 7 |
| 3.2 Relational Model Constraints . . . . . . . . . . . . . . . . . . . . . . . . | 7 |
| **4 Structured Query Language(SQL) Programming** | **9** |
| 4.1 SQL Data Definition and Data Types . . . . . . . . . . . . . . . . . . . . | 9 |
| 4.2 Assertions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 10 |
| 4.3 Triggers . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 10 |

* 1. [Database Programming 10](#_TOC_250010)
  2. [Database Connection 10](#_TOC_250009)
  3. [Stored Procedures 11](#_TOC_250008)

1. Requirements Specification 12
   1. [Hardware Requirements 12](#_TOC_250007)
   2. [Software Requirements 12](#_TOC_250006)
2. Relational Database Design 13
   1. [Entity Relation(ER) Diagram 13](#_TOC_250005)
   2. [Mapping From ER Diagram to Relational Schema Diagram 13](#_TOC_250004)
   3. [Relational Schema Diagram 14](#_TOC_250003)
3. Implementation 16
   1. [Table Structure 16](#_TOC_250002)
   2. [Codes Used For Modules: 17](#_TOC_250001)
4. Results and Discussion 21
5. Conclusion and Future work 24

[References 24](#_TOC_250000)

**List of Figures**

* 1. Schema Diagram of your project work 15
  2. Login Page 21
  3. Adding Customer Details 21
  4. Delete Bill 22
  5. Updating Customer Mobile Number 22
  6. Searching Customer ID 22
  7. Adding Products 23
  8. Trigger for Adding Bill 23
  9. Generated Bill 23

# Chapter 1 Introduction

### Database

he Student Attendance Management System (SAMS) database is a robust and efficient solution designed to streamline the process of managing and recording student attendance. Attendance tracking plays a critical role in educational institutions to monitor student participation and maintain compliance with academic regulations. The SAMS database stores essential information, such as student details, class schedules, attendance records, and reports, essen- tial information, such as student details, class schedules, attendance records, and reports, A SAMS database significantly reduces manual work, allowing institutions to automate attendance calculation, generate reports, and track trends over time. Features like alerts for irregular attendance and real-time dashboards enhance efficiency, transparency, and accountability. Additionally, it ensures secure access through role-based permissions for administrators, , teachers, and students.

### Database Management System(DBMS)

A Database Management System (DBMS) is a software system that facili- tates the creation, organization, storage, and retrieval of data in a structured and efficient manner. It serves as an interface between users and databases, enabling users to define, manipulate, retrieve, and manage.

### Characteristics of the Database Approach

The database approach offers a centralized method of managing data, en- suring consistency and reducing redundancy. It provides data independence by separating data structures from application programs, enabling seamless modifications without affecting functionality. With data abstraction, users can access multiple views of the data, such as physical, logical, and external, which enhances usability and clarity.

### Advantages of using the DBMS Approach

The Database Management System (DBMS) approach provides several key advantages that enhance data management and efficiency. It reduces data redundancy by centralizing storage, ensuring consistency, and saving storage space. Data integrity is maintained through enforced rules and constraints, ensuring data accuracy. DBMS enhances security by controlling access, al- lowing only authorized users to perform specific operations. It ensures data independence, enabling changes to data structures without impacting appli- cations.

### Schemas

A schema is the logical structure that defines the organization, storage, and relationships of data within a database system. It acts as a blueprint, detail- ing the design of the database, including tables, fields, relationships, and con- straints. There are three primary types of schemas. The physical schema spec- ifies how data is physically stored on hardware, including file structures and indexing. The logical schema focuses on the database’s logical design, defin- ing data organization, relationships, and constraints, independent of physical storage. Lastly, the view schema represents user-specific views.

### DBMS Languages

Database Management Systems (DBMS) support various specialized lan- guages to define, manipulate, query, and control data efficiently. The Data

Definition Language (DDL) is used to define the database structure, including creating, altering, and deleting objects like tables and schemas, with com- mands such as CREATE, ALTER, and DROP. The Data Manipulation Lan- guage (DML) allows users to retrieve and modify data, supporting operations like inserting, updating, and deleting records using commands such as IN- SERT, UPDATE, DELETE, and SELECT.

### Motivational challenges for the project work

Lack of Clear Vision: Without a clear understanding of the project goals, it can be difficult to stay motivated. A vague vision can lead to confusion and frustration, decreasing enthusiasm for completing tasks. Procrastination: Delaying tasks or work until the last minute can hinder project progress, leading to increased stress and burnout. This can make it harder to maintain focus and motivation. Burnout: Prolonged work with limited breaks or low social interaction may result in exhaustion. When energy levels dip, motivation significantly decreases, and the quality of work may also suffer. Overwhelming Tasks: Large projects can feel daunting, especially when tasks seem too complex or time-consuming. This might cause demotivation as the magnitude of the project feels unattainable.

### Application of the Project in Real time

This project serves as a powerful tool designed to optimize various aspects of business operations by facilitating efficient data management and decision- making processes. The system enables flexible working practices, as it allows for streamlined handling of tasks that traditionally By centralizing and au- tomating key business functions, such as inventory management, employee in- formation, customer interactions, and sales tracking, the project contributes to significant time savings.

# Chapter 2

**Conceptual Data Modeling**

A Conceptual Data Model (CDM) Student Attendance Management System is a high-level representation of the data structures and relationships within a system, used to define the types of data that will be stored in a database without focusing on how they will be implemented. It provides an abstract view of the data, designed to capture the essential entities, their attributes, and the relationships among them, serving as a blueprint for the database design process.

### Entity Relationship (ER) Model

An Entity-Relationship (ER) Model is a conceptual representation of data that outlines the structure and relationships between entities in a system. For an Attendance Management System, the ER model serves as the foundation for designing the database, helping to identify key entities such as ”Student,” ”Teacher,”session,” and ”Attendance .” It defines attributes like student ID, name, date, and attendance status, and illustrates relationships like students being part of classes and teachers maintaining attendance. This model sim- plifies the system’s data flow, ensuring accurate data storage, retrieval, and management.

### Entities

**”** Student” ”Teacher”Teaches” Admin”Attendance

4

### Attributes

Attributes: SId (Student ID) SName (Student Name) Class Section Password Teacher Attributes: TId (Teacher ID) TName (Teacher Name) Course Password Attendance Attributes: SId (Student ID - Foreign Key) TId (Teacher ID - Foreign Key) ADate

(Attendance Date) Session

Attributes: SId (Student ID - Foreign Key) TId (Teacher ID - Foreign Key) Admin Attributes: AId (Admin ID) AName (Admin Name) Password]Student

Attributes: SId (Student ID) SName (Student Name) Class Section Password Teacher Attributes: TId (Teacher ID) TName (Teacher Name) Course Password Attendance Attributes: SId (Student ID - Foreign Key) TId (Teacher ID - Foreign Key) ADate

(Attendance Date) Session

Attributes: SId (Student ID - Foreign Key) TId (Teacher ID - Foreign Key) Admin Attributes: AId (Admin ID) AName (Admin Name) Password

### Relationships

Entities Involved: Student Teacher Description: Represents the recording of atten- dance for students by teachers on a specific date. Attributes: ADate (Attendance Date) Teaches

Entities Involved: Teacher Session (linked to Student) Description: Represents the assignment of teachers to sessions where they teach specific classes or groups of students. Session

Entities Involved: Student Teacher Description: Represents a session where students and teachers interact, and attendance is managed. Has (Admin Relationship)

Entities Involved: Admin]Attendance

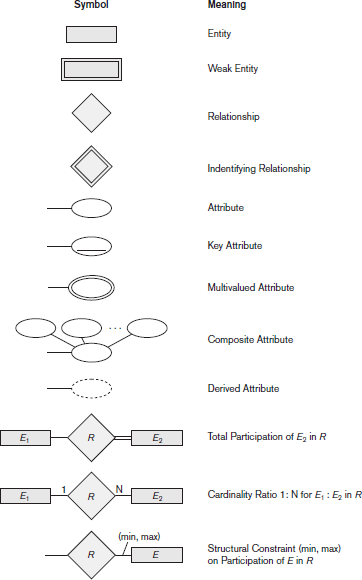
Entities Involved: Student Teacher Description: Represents the recording of atten- dance for students by teachers on a specific date. Attributes: ADate (Attendance Date) Teaches

Entities Involved: Teacher Session (linked to Student) Description: Represents the assignment of teachers to sessions where they teach specific classes or groups of students. Session

Entities Involved: Student Teacher Description: Represents a session where students and teachers interact, and attendance is managed. Has (Admin Relationship)

Entities Involved: Admin

### Notation for ER Diagrams



Summary of the Nota-

tion for ER Diagrams

# Chapter 3

**Relational Data Model**

### Relational Model Concepts

Relational Model Concepts for Student Attendance Management System: Relations (Tables): A relation in the relational model refers to a table that organizes data into rows and columns. For the Student Attendance Manage- ment System, several tables can be designed, such as: Students: Contains information about each student, such as StudentID, Name, Class, Contact Details. Courses: Stores course information, such as CourseID, CourseName, and InstructorID. Instructors: Stores information about instructors, such as InstructorID, Name, and Department. Attendance: Records student atten- dance for each class session, storing attributes like AttendanceID, StudentID, CourseID, Date, and Status (Present/Absent). Classes: Contains details of class sessions, including ClassID, Date, CourseID, and InstructorID. At- tributes (Columns): Attributes represent the characteristics or properties of the entities in the system. In the Students table, for instance, attributes may include StudentID, Name, DateOfBirth, Email, and Address. Each column contains data of a specific type, like integers, strings, or dates.

### Relational Model Constraints

Foreign Key Constraint: Maintains referential integrity between related entities. For example, SId in the Attendance table must correspond to a valid SId in the Student table, and TId must correspond to a valid TId in the Teacher table.

7

Mini Project Title Chapter 3

Not Null Constraint: Ensures that critical attributes, such as SName (Student Name) or ADate (Attendance Date), cannot have null values, preserving the completeness of records.

Unique Constraint: Guarantees that certain fields, like SId, TId, or AId, are unique within their tables, preventing duplicate records.

Domain Constraint: Restricts the values for specific attributes. For instance, Atten- dance Status might be limited to values like ”Present” or ”Absent.”

Check Constraint: Enforces conditions for specific attributes. For example, Password fields might require a minimum length for security.]Primary Key Constraint: Ensures that each entity has a unique identifier, such as SId for students, TId for teachers, and AId for admins. These attributes uniquely identify records in their respective tables.

Foreign Key Constraint: Maintains referential integrity between related entities. For example, SId in the Attendance table must correspond to a valid SId in the Student table, and TId must correspond to a valid TId in the Teacher table.

Not Null Constraint: Ensures that critical attributes, such as SName (Student Name) or ADate (Attendance Date), cannot have null values, preserving the completeness of records.

Unique Constraint: Guarantees that certain fields, like SId, TId, or AId, are unique within their tables, preventing duplicate records.

Domain Constraint: Restricts the values for specific attributes. For instance, Atten- dance Status might be limited to values like ”Present” or ”Absent.”

Check Constraint: Enforces conditions for specific attributes. For example, Password fields might require a minimum length for security.

Department of Information Science & Engineering, SCEM, Mangaluru Page 8

# Chapter 4

**Structured Query Language(SQL) Programming**

### SQL Data Definition and Data Types

In SQL, Data Definition Language (DDL) is used to define the structure of a database, including the creation of tables, attributes, and relationships. Below is a breakdown of SQL DDL and common data types for an Attendance Management System

INT or BIGINT: For IDs such as SId, TId, AId. DECIMAL or FLOAT: For numerical data requiring precision (e.g., percentages). String Data Types:

VARCHAR(n): For attributes like names, course names, and passwords where variable- length text is needed. CHAR(n): For fixed-length strings like Section. Date and Time Data Types:

DATE: For attributes like ADate to store attendance dates. DATETIME or TIMES- TAMP: If time tracking is required along with the date. Boolean Data Types (in some databases):

BOOLEAN: To store true/false values, such as attendance status (can also use TINYINT(1) in MySQL).]Numeric Data Types:

INT or BIGINT: For IDs such as SId, TId, AId. DECIMAL or FLOAT: For numerical data requiring precision (e.g., percentages). String Data Types:

VARCHAR(n): For attributes like names, course names, and passwords where variable- length text is needed. CHAR(n): For fixed-length strings like Section. Date and Time Data Types:

9

DATE: For attributes like ADate to store attendance dates. DATETIME or TIMES- TAMP: If time tracking is required along with the date. Boolean Data Types (in some databases):

BOOLEAN: To store true/false values, such as attendance status (can also use TINYINT(1) in MySQL).

### Assertions

For the Student Attendance Management System (SAMS) in a DBMS project, assertions can be used to ensure data integrity and enforce certain conditions that should always hold true. In this context, assertions can be used to vali- date conditions related to attendance, enrollment, and relationships between students, session, and teachers,attributes.

### Triggers

Here are a few examples of how triggers can be used in your SAMS project:]In a Student Attendance Management System (SAMS) in a DBMS project, triggers can be used to automatically perform certain actions when specific changes occur in the database. Triggers can be helpful for enforcing business rules, auditing data, and maintaining data integrity.

Here are a few examples of how triggers can be used in your SAMS project:

### Database Programming

Database Programming involves writing code to interact with databases, typ- ically for tasks such as querying, inserting, updating, and deleting data. It often uses SQL (Structured Query Language) along with a programming lan- guage (such as Python, Java, PHP, or C) to perform operations on a database.

### Database Connection

XAMPP Official Website

Install MySQL and a Web Server like Apache (or use tools like XAMPP, WAMP,

or MAMP). Install MySQL client libraries (often bundled with PHP). Use MySQLi or PDO (PHP Data Objects) to connect to the database. Step-by-Step:]PHP is often used to interact with a database (such as MySQL) in web development. To use MySQL with PHP, you need to ensure the following:

Install MySQL and a Web Server like Apache (or use tools like XAMPP, WAMP, or MAMP). Install MySQL client libraries (often bundled with PHP). Use MySQLi or PDO (PHP Data Objects) to connect to the database. Step-by-Step:

Host: The server where the database is hosted (e.g., localhost or an IP address). Username: The username to access the database (e.g., root). Password: The pass- word associated with the username. Database Name: The specific database you want to connect to.]Provide Connection Parameters: To connect to the database, you need parameters such as:

Host: The server where the database is hosted (e.g., localhost or an IP address). Username: The username to access the database (e.g., root). Password: The password associated with the username. Database Name: The specific database you want to con- nect to.

### Stored Procedures

A Stored Procedure is a set of SQL queries that are stored and executed in the database. It is typically used to perform a specific task or operation like inserting, updating, deleting, or querying data. Stored procedures allow you to encapsulate logic, improving code reusability, security, and performance.

# Requirements Specification

### Hardware Requirements

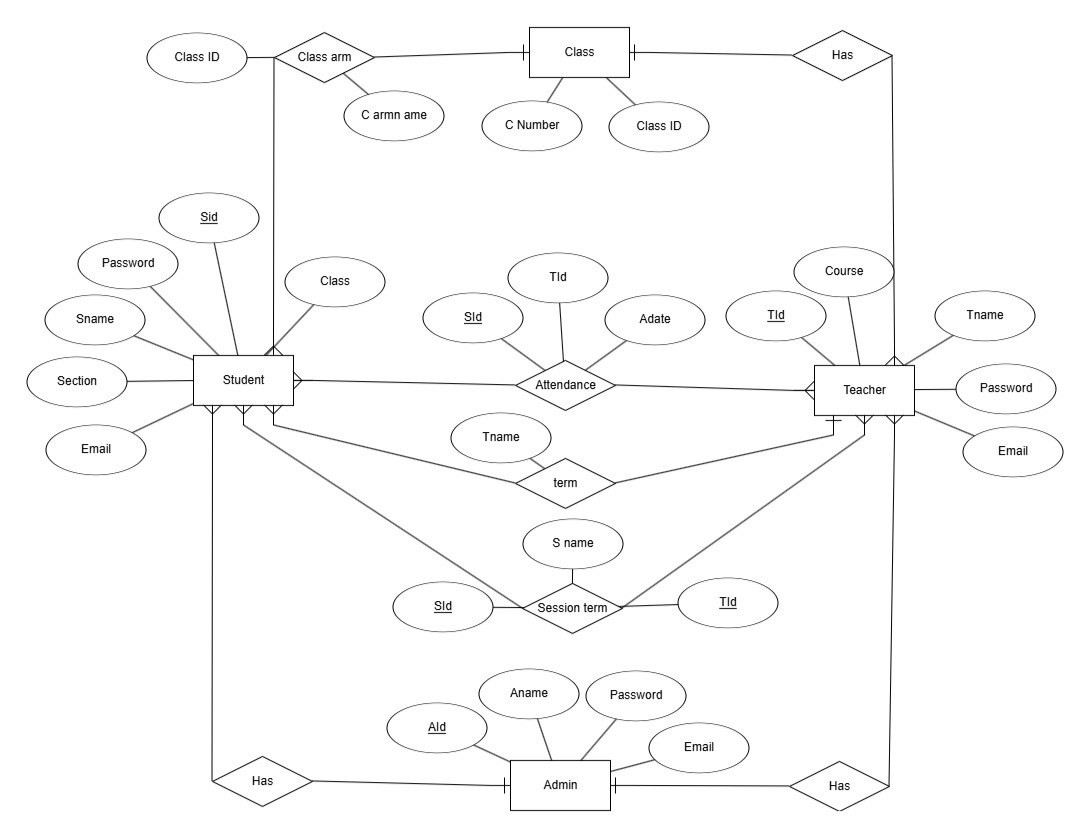
* + - Processor : Any Processor more than 500 MHz
    - RAM : 4GB
    - Hard Disk : 512GB
    - Input Device : Standard Keyboard and Mouse
    - Output Device : Monitor

### Software Requirements

* + - Database : •XAMPP
    - Programming Language :•CSS,Java script,PHP,SQL
    - IDE :AMPPA 8.2.4
    - Operating System : Windows 10

# Relational Database Design

### Entity Relation(ER) Diagram



### Mapping From ER Diagram to Relational Schema Diagram

Include a table highlighting all the Seven steps involved in the conversion of ER Model into the Relational Model

In our database we have the following mappings:

Step 1 : Mapping of Regular Entities.

Explain the Regular entity types in your ER model. Resolve them and draw the Rela- tional Model after the mapping of regular entities

Step 2: Mapping of Weak Entity Types.

Explain the Weak entity types in your ER model. Resolve them and redraw the Rela- tional Model after the mapping of Weak entities

Step 3: Mapping of binary 1:1 Relation Types.

Explain the 1:1 relations in your ER model. Resolve them and redraw the Relational Model after the mapping of 1:1 relations

Step 4: Mapping of binary 1:N Relation Types.

Explain the 1:N relations in your ER model. Resolve them and redraw the Relational Model after the mapping of 1:N relations

Step 5: Mapping of binary M:N Relation Types.

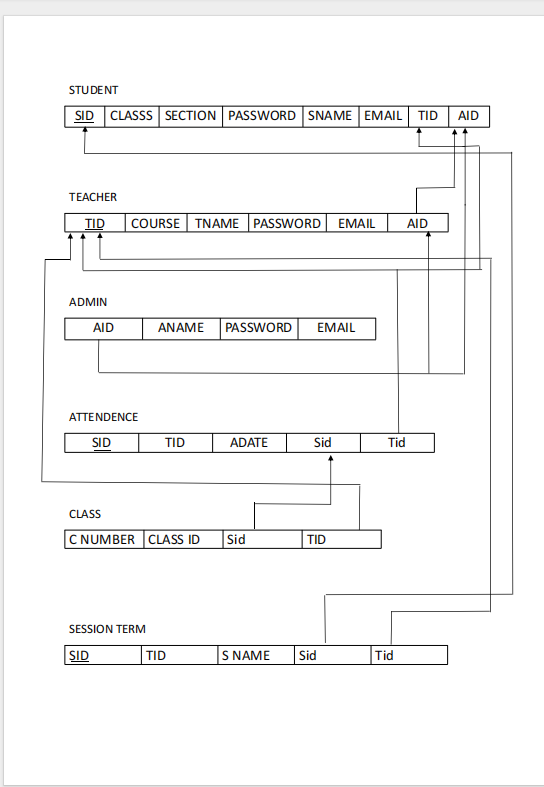
Explain the M:N relations in your ER model. Resolve them and redraw the Relational Model after the mapping of M:N relations

Step 6: Mapping of multivalued attributes.

Explain the multivalued attributes in your ER model. Resolve them and redraw the Relational Model after the mapping of multivalued attributes

### Relational Schema Diagram

A schema is a pictorial representation of the relationship between the database tables in the database that is created. The database schema of a database system is its structure described in a formal language supported by the database management system (DBMS). The term ”schema” refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language. The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modelled in the database.

Figure 6.1: Schema Diagram of your project word

# Chapter 7 Implementation

### Table Structure

1. CREATE TABLE ‘tbladmin‘ ( ‘Id‘ int(10) NOT NULL, ‘firstName‘ var- char(50) NOT NULL, ‘lastName‘ varchar(50) NOT NULL, ‘emailAddress‘ varchar(50) NOT NULL, ‘password‘ varchar(50) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;
2. CREATE TABLE ‘tblattendance‘ ( ‘Id‘ int(10) NOT NULL, ‘admissionNo‘ varchar(255) NOT NULL, ‘classId‘ varchar(10) NOT NULL, ‘classArmId‘ varchar(10) NOT NULL, ‘sessionTermId‘ varchar(10) NOT NULL, ‘status‘ varchar(10) NOT NULL, ‘dateTimeTaken‘ varchar(20) NOT NULL ) EN- GINE=MyISAM DEFAULT CHARSET=latin1;
3. CREATE TABLE ‘tblclass‘ ( ‘Id‘ int(10) NOT NULL, ‘className‘ var- char(255) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;
4. CREATE TABLE ‘tblclassarms‘ ( ‘Id‘ int(10) NOT NULL, ‘classId‘ var- char(10) NOT NULL, ‘classArmName‘ varchar(255) NOT NULL, ‘isAssigned‘ varchar(10) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE ‘tblclassteacher‘ ( ‘Id‘ int(10) NOT NULL, ‘firstName‘ varchar(255) NOT NULL, ‘lastName‘ varchar(255) NOT NULL, ‘emailAd- dress‘ varchar(255) NOT NULL, ‘password‘ varchar(255) NOT NULL, ‘pho- neNo‘ varchar(50) NOT NULL, ‘classId‘ varchar(10) NOT NULL,

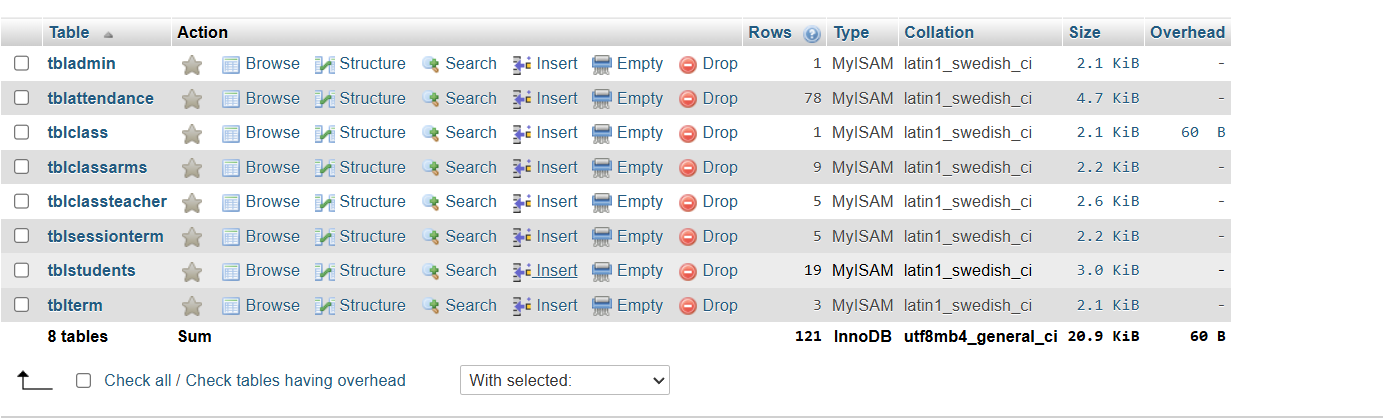
6.varchar(10) NOT NULL, ‘dateCreated‘ varchar(50) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

1. CREATE TABLE ‘tblsessionterm‘ ( ‘Id‘ int(10) NOT NULL, ‘sessionName‘ varchar(50) NOT NULL, ‘termId‘ varchar(50) NOT NULL, ‘isActive‘ var- char(10) NOT NULL, ‘dateCreated‘ varchar(50) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE ‘tblstudents‘ ( ‘Id‘ int(10) NOT NULL, ‘firstName‘ var- char(255) NOT NULL, ‘lastName‘ varchar(255) NOT NULL, ‘otherName‘ varchar(255) NOT NULL, ‘admissionNumber‘ varchar(255) NOT NULL, ‘pass- word‘ varchar(50) NOT NULL, ‘classId‘ varchar(10) NOT NULL, ‘classArmId‘

varchar(10) NOT NULL, ‘dateCreated‘ varchar(50) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE ‘tblterm‘ ( ‘Id‘ int(10) NOT NULL, ‘termName‘ varchar(20) NOT NULL ) ENGINE=MyISAM DEFAULT CHARSET=latin1;



### Codes Used For Modules:

Insert

1.INSERT INTO ‘tbladmin‘ (‘Id‘, ‘firstName‘, ‘lastName‘, ‘emailAddress‘, ‘password‘) VALUES (1, ’Admin’, ”, [’admin@mail.com’,](mailto:admin@mail.com) ’D00F5D5217896FB7FD601412CB890830’);

INSERT INTO ‘tblattendance‘ (‘Id‘, ‘admissionNo‘, ‘classId‘, ‘classArmId‘, ‘sessionTermId‘, ‘status‘, ‘dateTimeTaken‘) VALUES (162, ’ASDFLKJ’, ’1’, ’2’, ’1’, ’1’, ’2020-11-01’), (163, ’HSKSDD’, ’1’, ’2’, ’1’, ’1’, ’2020-11-01’),

(164, ’JSLDKJ’, ’1’, ’2’, ’1’, ’1’, ’2020-11-01’), (172, ’HSKDS9EE’, ’1’, ’4’,

’1’, ’1’, ’2020-11-01’), (171, ’JKADA’, ’1’, ’4’, ’1’, ’0’, ’2020-11-01’),

2.CREATE TABLE ‘tblclass‘ ( ‘Id‘ int(10) NOT NULL, ‘className‘ varchar(255) NOT NULL ) ENGINE=MyISAM DEFAULT

CHARSET=latin1;

INSERT INTO ‘tblclass‘ (‘Id‘, ‘className‘) VALUES (1, ’Seven’), (3, ’Eight’), (4, ’Nine’);

4.INSERT INTO ‘tblclassarms‘ (‘Id‘, ‘classId‘, ‘classArmName‘, ‘isAssigned‘) VALUES (2, ’1’, ’S1’, ’1’), (4, ’1’, ’S2’, ’1’), (5, ’3’, ’E1’, ’1’), (6,

’4’, ’N1’, ’1’);

5.INSERT INTO ‘tblclassteacher‘ (‘Id‘, ‘firstName‘, ‘lastName‘, ‘emailAddress‘, ‘password‘, ‘phoneNo‘, ‘classId‘, ‘classArmId‘, ‘dateCreated‘) VALUES (1, ’Will’, ’Kibagendi’, [’teacher2@mail.com’,](mailto:teacher2@mail.com) ’32250170a0dca92d53ec9624f336ca24’, ’09089898999’, ’1’, ’2’, ’2022-10-31’),

(4, ’Demola’, ’Ade’, [’teacher3@gmail.com’,](mailto:teacher3@gmail.com) ’32250170a0dca92d53ec9624f336ca24’, ’096720 02882’, ’1’, ’4’, ’2022-11-01’),

(5, ’Ryan’, ’Mbeche’, [’teacher4@mail.com’,](mailto:teacher4@mail.com) ’32250170a0dca92d53ec9624f336ca24’, ’7014560000’, ’3’, ’5’, ’2022-10-07’),

(6, ’John’, ’Keroche’, [’teacher@mail.com’,](mailto:teacher@mail.com) ’32250170a0dca92d53ec9624f336ca24’, ’0100000030’, ’4’, ’6’, ’2022-10-07’);

6.INSERT INTO ‘tblsessionterm‘ (‘Id‘, ‘sessionName‘, ‘termId‘, ‘isActive‘, ‘dateCreated‘) VALUES (1, ’2021/2022’, ’1’, ’1’, ’2022-10-31’), (3,

’2021/2022’, ’2’, ’0’, ’2022-10-31’);

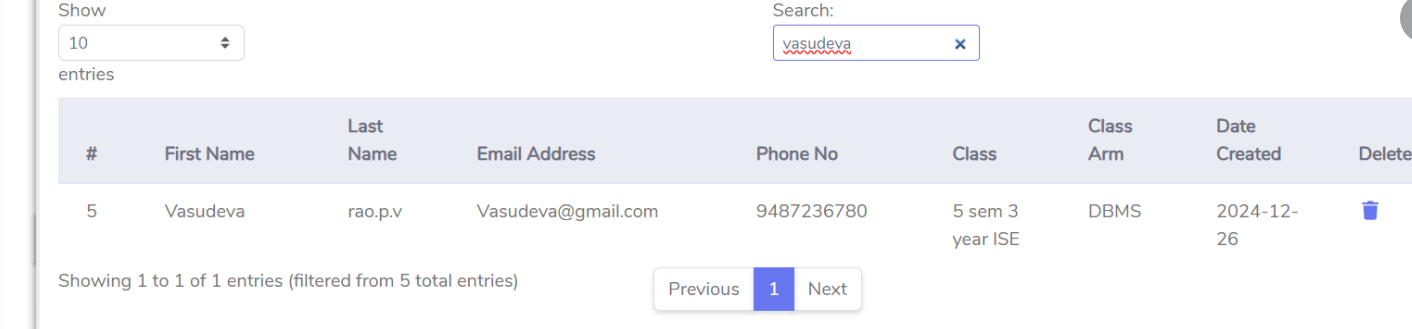
Delete:

DELETE FROM ‘tblattendance‘ WHERE ‘Id‘ = 162;

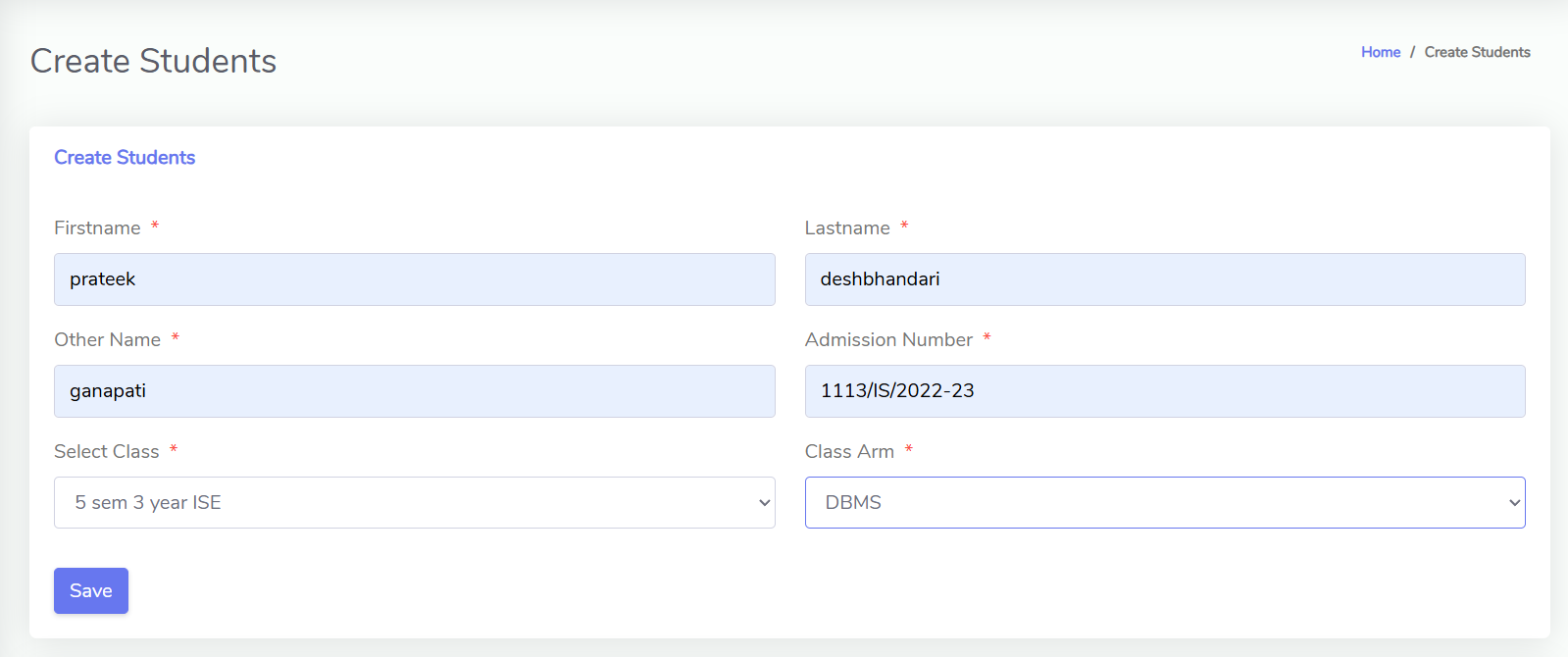
Update:

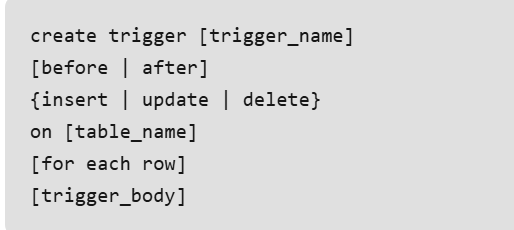
UPDATE ‘tblattendance‘ SET ‘status‘ = ’Absent’ WHERE ‘Id‘ = 170;

Search Teacher:

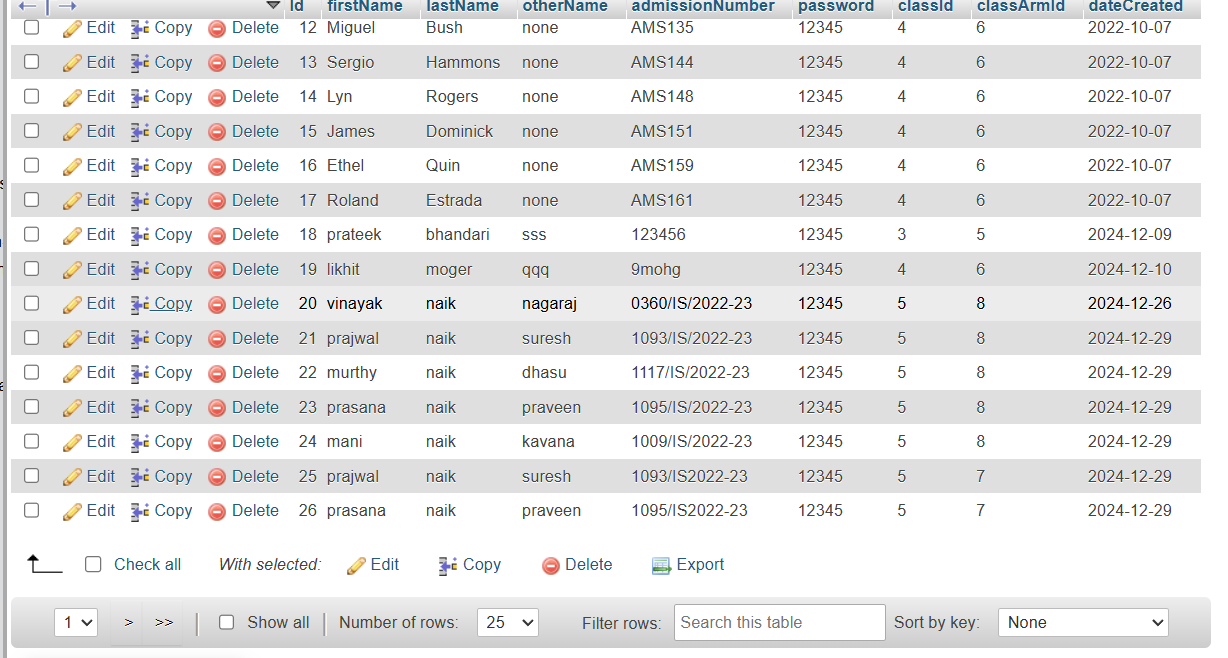


Display To Create Studennt:

Trigger:



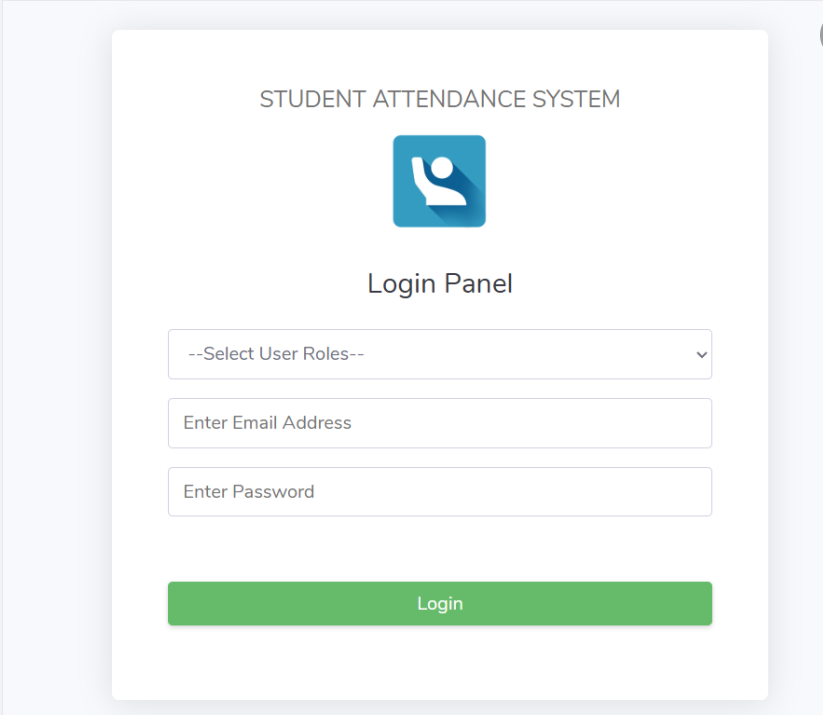
Stored Procedure Student:



Chapter 8

**Results and Discussion**

Login Page:

 Figure 8.1: Login Page

This is our login page.Here the cashier can select his Id from a dropdown list and can login.

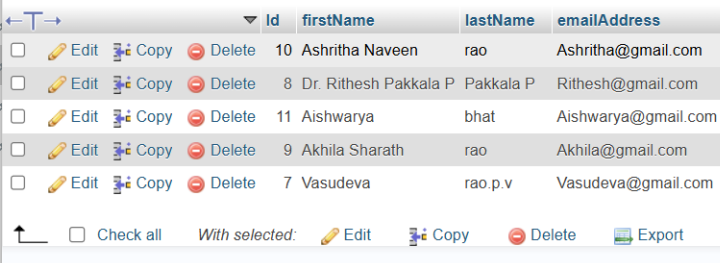
student Details:



Figure 8.2: Adding Student Details

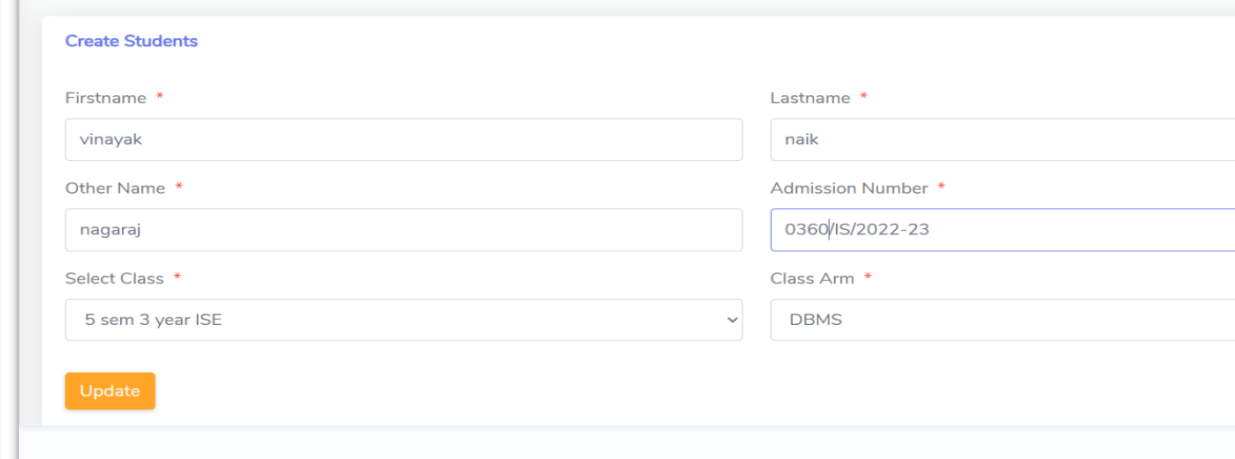
21

Deleting Teacher:

 Figure 8.3: DeleteTeacher

Here cashier can delete teacher of the selecting that particular row.

**Update STUDENT Details**:

 Figure 8.4: Updating Customer Mobile Number

To implement functionality for updating a customer's mobile number and proceeding with billing, you can break it down into two major operations:

**Teacher Details** :

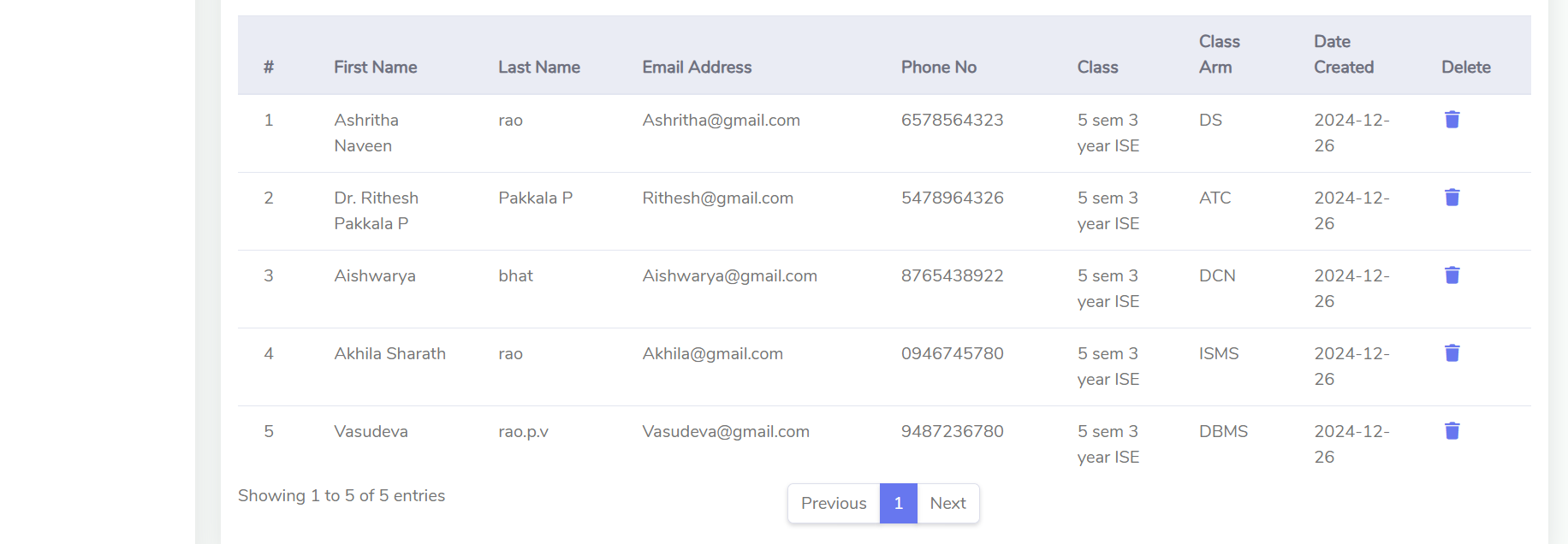


Figure 8.6:Teacher Details

It looks like you are asking for a system to manage teacher details and student attendance.

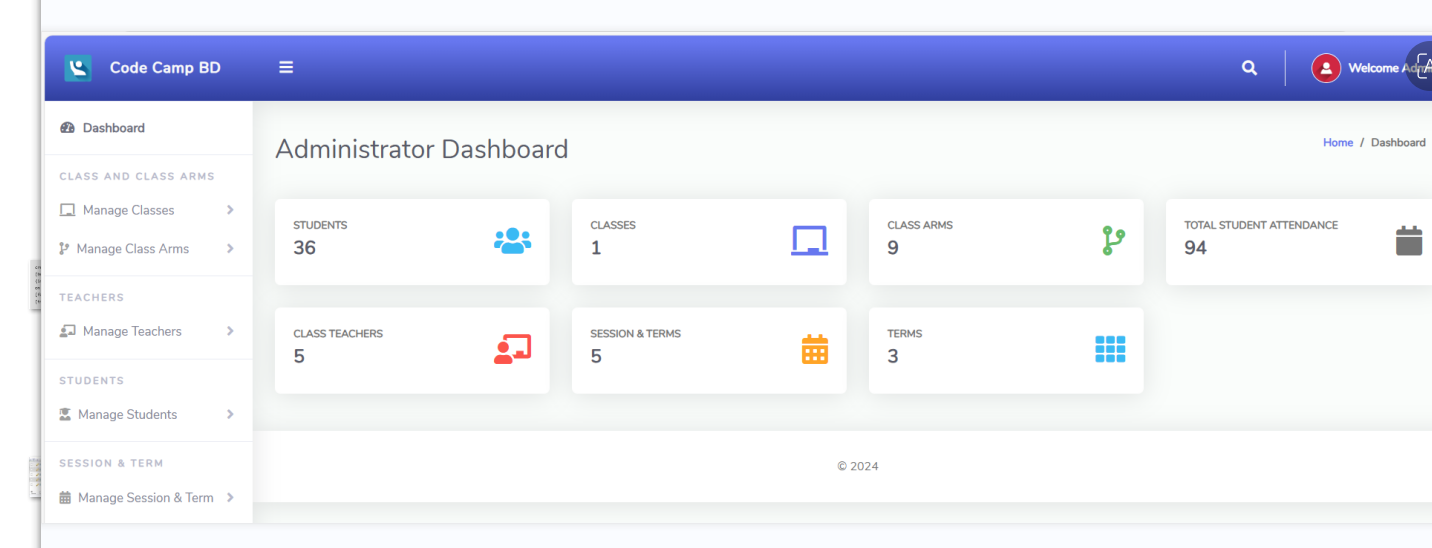
Trigger:



Figure 8.7: Trigger for Adding Bill

Trigger will be violated if will try to generate bill between 8am to 8pm.

**Administrator Display:**

Figure 8.8:Administrator Display

# Chapter 9

**Conclusion and Future work**

In conclusion, this project not only addressed the challenges of manual attendance management but also demonstrated the capabilities of relational databases in managing structured data. It can be used in schools, colleges, and universities to minimize errors and ensure data integrity in attendance records.]The Attendance Management System project aimed at developing a database management system (DBMS) solution for track- ing and managing attendance efficiently. The system was designed to automate the process of marking and recording attendance, allowing both administrators and students to access real-time attendance data. Through the use of SQL queries, the system ensures accurate and timely recording of attendance data, improving the workflow for educational institutions. The system is built to provide ease of use, security, and scalability for future enhancements.

In conclusion, this project not only addressed the challenges of manual attendance management but also demonstrated the capabilities of relational databases in managing structured data. It can be used in schools, colleges, and universities to minimize errors and ensure data integrity in attendance records.

# References

1. Database systems Models, Languages, Design and Application Programming, Prateek and prajwal, 6th Edition, Pearson.
2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, Mc- Graw Hill.
3. Silberschatz Korth and Sudharshan: Database System Concepts, 6th Edition, Mc- Graw Hill, 2013.
4. Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implemen- tation and Management, Cengage Learning 2012.