

**PROJECT REPORT**

**ON**

**STUDENT MANAGEMENT SYSTEM**

**SUBMITTED TO**

Mr. Anuj Kumar

**SUBMITTED BY**

Yashvi Chaudhary(AF04991261)

Charu(AF04991721)

Vanshika(AF04992208)

**Batch code**

ANP- D2406

**Course code**

ITPR

**DR. K.N. MODI INSTITUTE OF ENGINEERING AND  
TECHNOLOGY**

Academic Year:2025 - 26

## **ABSTRACT:-**

The **Student Management System** is a terminal-based software application developed using **Core Java, JDBC, and MySQL**. The purpose of this project is to automate the process of maintaining student records in educational institutions. Manual record keeping is time consuming, error-prone, and inefficient.

This system provides a computerized solution to store, update, delete, and retrieve student-related data efficiently.

## **INTRODUCTION:-**

Educational institutions handle large volumes of student data such as personal details, academic records, course enrollments, and marks. Managing this information manually becomes difficult as the number of students increases.

The **Student Management System** addresses these issues by providing a reliable and structured way to manage data. The system uses **Java** as the core programming language, **JDBC** for database connectivity, and **MySQL** as the backend database. It ensures data accuracy, quick retrieval, and easy management through a terminal-based interface.

## **OBJECTIVES:-**

- To automate student data management
- To reduce paperwork and manual errors
- To maintain a centralized student database
- To provide fast searching and retrieval of data
- To demonstrate CRUD operations using JDBC
- To improve efficiency and accuracy

## **PROJECT CATEGORY:-**

This project belongs to the **Database Management System (DBMS)** category.

It is a terminal-based software application developed using **Core Java**.

**JDBC** is used to connect Java with **MySQL** database.

The project demonstrates **Create, Read, Update, and Delete (CRUD)** operations.

## **SYSTEM ANALYSIS:-**

System analysis involves understanding the requirements and designing a system that fulfills those needs.

The **Student Management System** is designed to manage student records, courses, and marks efficiently.

The system is divided into multiple modules to simplify functionality and improve maintainability.

## **MODULES DESCRIPTION:-**

- 1.Admin Management Admin login and system control.
- 2.Student Management Add, update, view, and delete student records.
- 3.Course Management Add, update, view, and delete courses.
- 4.Marks Management Store and manage student marks.
- 5.Report Management Generate reports and search student data.

## **DATABASE DESIGN:-**

The database is designed using **MySQL**. It consists of tables such as Admin, Student, Course, Enrollment, Marks, and Report\_Log.

Each table is designed with proper primary keys and foreign key relationships to ensure data integrity.

## **ENTITY RELATIONSHIP DIAGRAM:-**

The **ER diagram** represents the relationships between entities such as Student, Course, Admin, and Marks. It shows one-to-many and many-to-many relationships using enrollment tables.

## **DATA FLOW DIAGRAM:-**

The Data Flow Diagram (DFD) illustrates how data flows through the system. It includes:

- Zero Level DFD
- First Level DFD
- Second Level DFD

## **PROCESS LOGIC:-**

**The system follows a simple process logic:**

1. User selects an operation
2. Input is taken through terminal
3. Data is validated
4. Database operation is performed
5. Output is displayed

## **PLATFORM USED:-**

### **Hardware Requirements:**

- Intel i3 or higher
- 4GB RAM
- 500MB storage

### **Software Requirements:**

- JDK 8 or above
- MySQL server
- Eclipse
- JDBC Driver

## **IMPLEMENTATION DETAILS:-**

The project is implemented using layered architecture. **DAO (Data Access Object)** pattern is used to separate logic.

Prepared Statement is used to prevent SQL injection.

Exception handling ensures system stability.

## **TESTING:-**

Testing ensures that the system works as expected. Unit testing is done for each module. Test cases include adding, updating, deleting, and viewing records.

## **RESULTS:-**

The system successfully manages student data. It provides fast data retrieval and secure storage. The terminal interface is user-friendly and efficient.

## **ADVANTAGES:-**

- Easy to use
- Reduces paperwork
- Accurate data management
- Secure and reliable
- Scalable system

## **LIMITATIONS:-**

- Terminal-based interface
- Limited user roles
- Requires basic technical knowledge

## **FUTURE SCOPE:-**

- Web-based version
- Mobile application
- Role-based access
- Cloud integration
- Advanced analytics
- Online fee management

## **CONCLUSION:-**

The **Student Management System** successfully automates student record management. It demonstrates the practical use of **Java**, **JDBC**, and **MySQL**. The project meets all objectives and provides a strong foundation for future upgrades.

**THANK YOU**