# UNIVERSITY INSTITUTE OF COMPUTING

**PROJECT REPORT ON**

## Student Management System

Program Name: BCA

Subject Name/Code: JAVA PROGRAMMING LAB (22CAP-352)

Submitted by: Submitted to:

**Name: SIDHANT VERMA Name: SUMAN ACHARYA**

**UID:22BCA10290 Designation: Asst.prof Section:22BCA-8-A**

# STUDENT MANAGEMENT SYSTEM

# Sure! Here's a Student Management System Report for a project built using Java programming. This is a structured report that you can use for documentation, submission, or presentation. It includes the necessary sections like Title, Objective, Modules, Technologies Used, Screenshots (optional), and Conclusion.

# 🎯 Objective:

# The aim of the Student Management System is to automate the management of student data, including student registration, course enrollment, marks entry, and generating academic records. The system aims to simplify the process of handling student information in schools or colleges.

# 🛠️ Technologies Used:

# . Programming Language: Java

# . IDE Used: IntelliJ IDEA / Eclipse / NetBeans (choose your IDE)

# . Database: MySQL (or Java File Handling if no DB used)

# .UI Library (if any): Swing / JavaFX

# 📁 Modules of the Project:

# 1. Student Module

# . Add new students

# . Edit student information

# . View student details

# . Delete student records

# 2. Course Module

# . Add new courses

# . Assign courses to students

# . View course details

# 3. Marks Module

# . Enter marks for students

# . Calculate total and average

# . Generate grade report

# 4. Admin Module

# . Login system (Username and Password)

# . Manage Students, Courses, and Marks

# . Generate Reports

# 📷 Screenshots (Optional)

# *(Insert screenshots of GUI or command-line interface)*

# 🧮 Database Structure (if applicable):

# Students Table: | ID | Name | Age | Email | Course\_ID |

# Courses Table: | Course\_ID | Course\_Name | Duration |

# Marks Table: | Student\_ID | Subject | Marks |

# 🔐 Features:

# . Secure admin login

# . Search student by ID or Name

# . View all registered students

# . marksheet / grade reports

# . Error handling and data validation

# . Save and retrieve data from the database or file

# ✅ Output Example (CLI/GUI):

# 1. Add Student

# 2. View Students

# 3. Enter Marks

# 4. Generate Report

# 5. Exit

# Enter your choice:

# 📚 Sample Java Code Snippet:

# public class Student {

# private int id;

# private String name;

# private int age;

# private String department;

# // Constructor

# public Student(int id, String name, int age, String department) {

# this.id = id;

# this.name = name;

# this.age = age;

# this.department = department;

# }

# // Getter and Setter methods

# public int getId() {

# return id;

# }

# public void setId(int id) {

# this.id = id;

# }

# public String getName() {

# return name;

# }

# public void setName(String name) {

# this.name = name;

# }

# public int getAge() {

# return age;

# }

# public void setAge(int age) {

# this.age = age;

# }

# public String getDepartment() {

# return department;

# }

# public void setDepartment(String department) {

# this.department = department;

# }

# // Display Student details

# public void displayStudent() {

# System.out.println("ID: " + id);

# System.out.println("Name: " + name);

# System.out.println("Age: " + age);

# System.out.println("Department: " + department);

# }

# }

# 📌 Conclusion:

# This Student Management System streamlines educational institution operations by efficiently handling student data. It offers easy record keeping, fast data retrieval, and better accuracy. The system is expandable and can be integrated with online platforms or mobile apps in the future.

# 