**Student Management System**

**Summary**

The Student Management System is a web-based application developed to streamline administrative tasks related to managing student information. The system provides an intuitive admin panel for adding new students and fetching student records based on various criteria. This report outlines the key features, technologies used, implementation details, and future considerations for the project.

**Objectives**

The primary objectives of the Student Management System include:

1. Efficiently manage student information.

2. Allow administrators to add new students with relevant details.

3. Facilitate the retrieval of student records based on specified criteria.

**Technologies Used**

* Java Servlets
* Apache Tomcat
* NetBeans IDE
* MySQL Database

**System Architecture**

The Student Management System follows a three-tier architecture:

1. Presentation Tier :

Servlets for handling user requests and managing the flow of data.

2. Application Tier:

* Java business logic for processing data.
* JDBC for database connectivity.
* Apache Tomcat as the servlet container.
* MySQL database for storing student information.

**Features**

A. Adding Students

* Admin panel with a form for adding new students.
* Servlet to handle form submissions and interact with the database.

B. Fetching Student Records

* Admin panel functionality for retrieving student records.
* Servlet to process user queries and fetch data from the database.

**Implementation Details**

A. Database Setup

* MySQL database schema design.
* JDBC connection for database interactions.

B. Servlets

* Separate servlets for adding and fetching students.
* Proper request handling using `doGet` and `doPost` methods.

**User Interface**

* + Forms for data input and result display for fetched records.

**Future Considerations**

* + Implementing additional features like editing and deleting student records.
  + Enhancing user authentication and authorization mechanisms.
  + Improving the user interface for a more user-friendly experience.

**Conclusion**

The Student Management System project successfully achieves its objectives of providing an efficient solution for managing student information. The use of Java Servlets, Apache Tomcat, and MySQL database ensures a robust and scalable system. Future enhancements can further improve the system's functionality and user experience.