

DHAKA INTERNATIONAL UNIVERSITY



A Mini Project Report On

“Student Management System “

Submitted in partial fulfillment of the requirement for the ‘Object Oriented Programming Course Project’ of 6th semester Bachelor of Engineering in Computer Science and Engineering.

Submitted by

Md Saihan Alam (12)

Under the guidance of

Raiyan Ul Islam

Lecturer, Dept of CSE (DIU)

Description :

The Student Management System is a Java-based application designed to streamline the administrative tasks of managing student information within an educational institution. It provides functionalities for adding new students, searching for students by their ID, and generating result cards displaying enrolled courses and grades.

Key Feature :

- **Add Student :** Enables administrators to input a new student's name and unique ID, creating a new student record in the system.
- **Search Student :** Allows users to find a specific student by entering their ID, providing immediate feedback on the student's existence in the database.
- **Print Result Card :** Generates a detailed result card for a student upon entering their ID, showing their name, ID, list of enrolled courses, and respective grades.
- **User Interface :** The application features a simple command-line interface (CLI) for user interaction, offering intuitive menu options to perform various operations seamlessly.

Classes and Structure :

- **Person Class :** Represents a basic entity with attributes name and id, serving as a superclass for Student.
- **Student Class :** Extends Person and includes additional attributes courses (list of courses enrolled) and grades (list of corresponding grades).
- **Student Management System Class:** Manages a collection of Student objects, providing methods to add students, search for students by ID, and print their result cards.
- **ManagementSystem :** It is a interface ensures that any class implementing it will provide the necessary functionality to manage students.

Source Code :

Github : <https://github.com/Saihan-DIU/Student-Management-System/blob/main/StudentManagementSystem12/src/main/java/com/saint/studentmanagementsystem12/StudentManagementSystem12.java>

Data Flow Diagram :

The data flow diagram for the Student Management System is as follows:

1. External Entities :

- **User:** Interacts with the system to perform operations such as adding students, searching for students, and printing result cards.

2. Processes :

- **Add Student:** Allows users to add new students, including their courses and grades.
- **Search Student:** Allows users to search for students by their ID.
- **Print Result Card:** Allows users to print a student's result card.

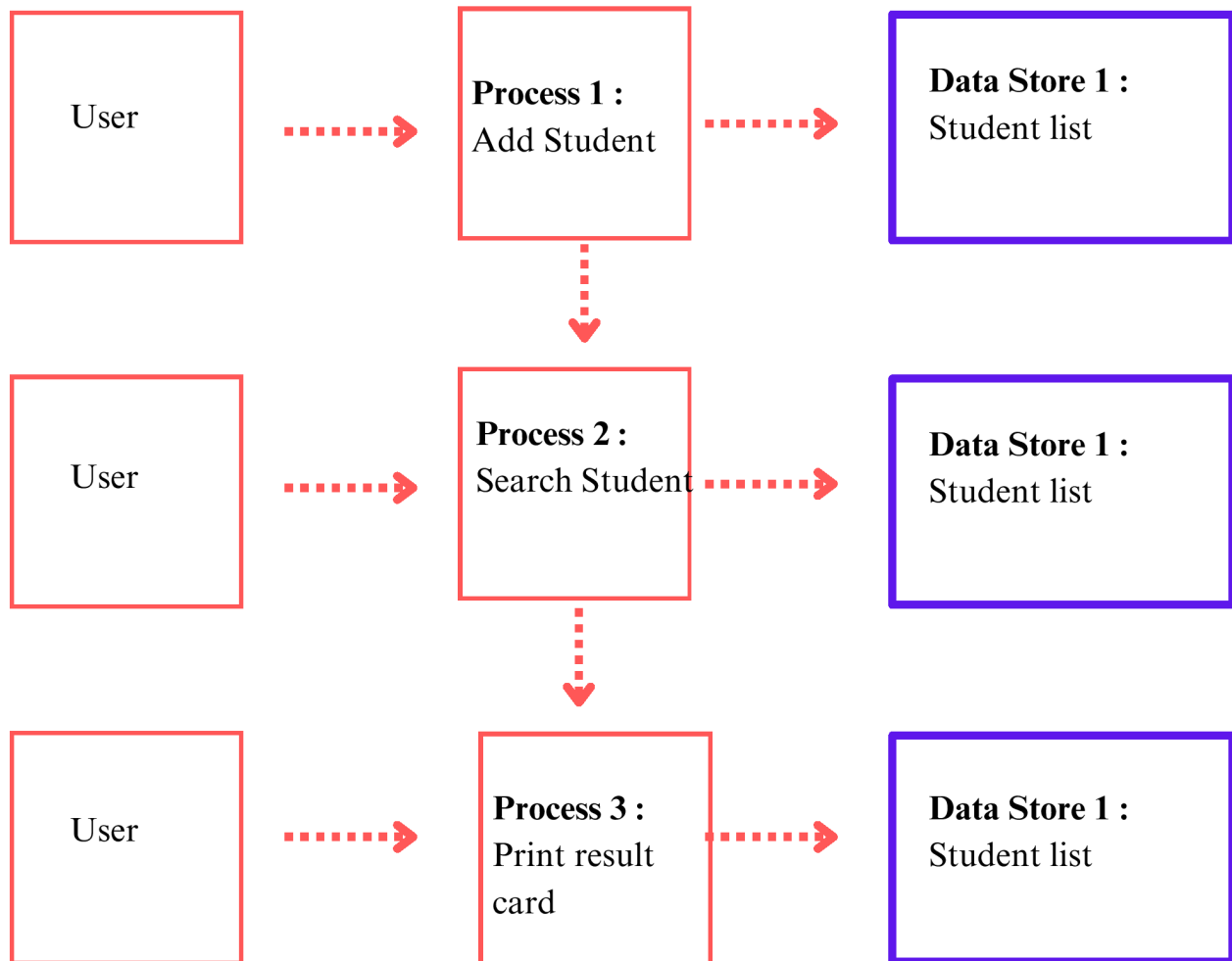
3. Data Stores :

- **Student List:** A list that stores Student objects, maintaining all student records.

4. Data Flow :

- **User Input:** Data entered by the user (student details, course names, grades).
- **Student Data:** Information related to students, including their names, IDs, courses, and grades.

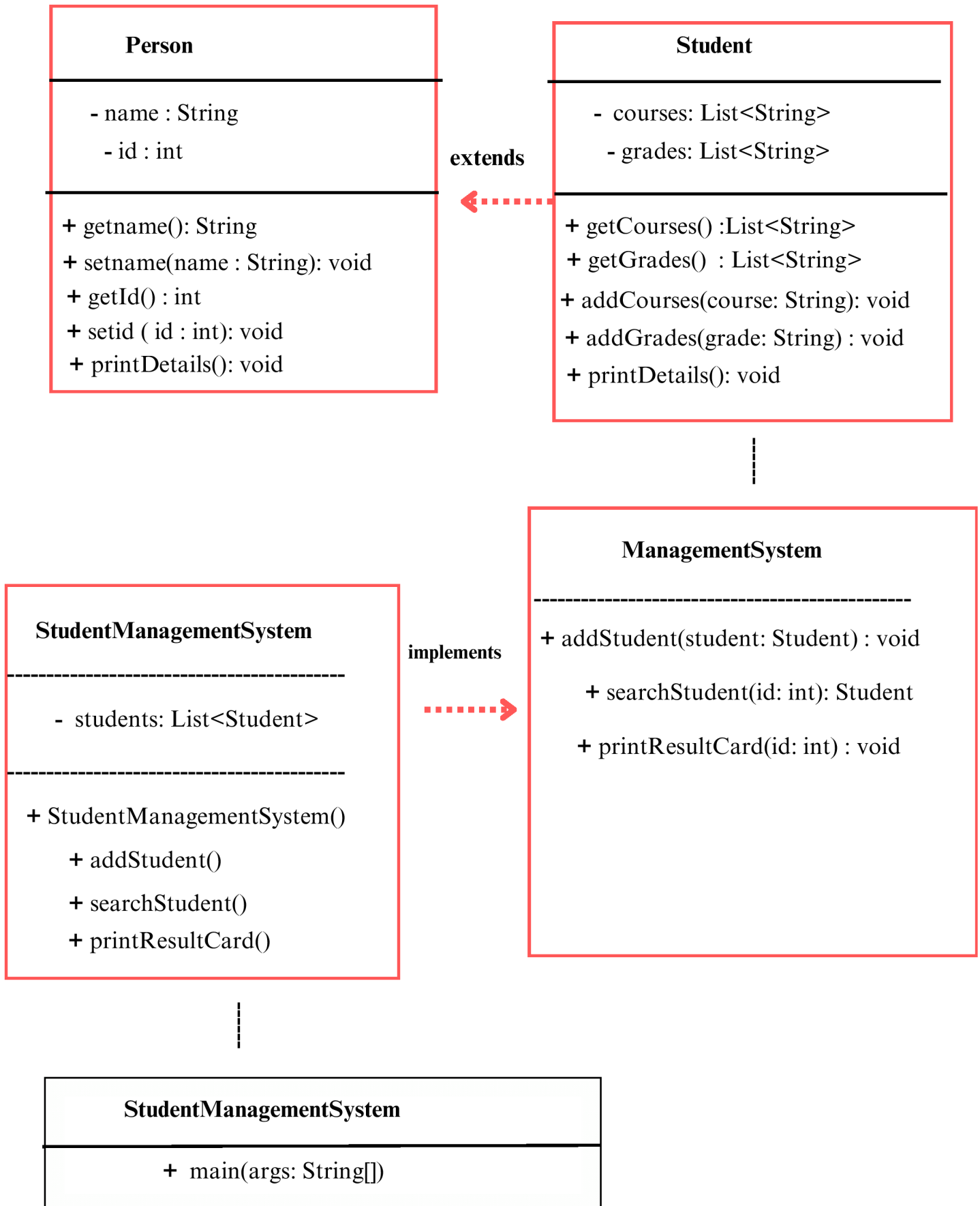
Data Flow Diagram :



UML Diagram Notes :

- **Inheritance** is represented by a solid line with a hollow arrow pointing to the superclass (**Person** from **Student**).
- **Association** is represented by a line connecting **StudentManagementSystem** to **Student**.
- **Attributes** are shown with a minus sign (**-**) for private and **Methods** are shown with a plus sign (**+**) for public visibility.

UML Diagram :



Console Interface :

```
1. Add student
2. Search student
3. Print result card
4. Exit
- Enter your choice: |
```

Console Input : Adding a student in the system ,

```
1. Add student
2. Search student
3. Print result card
4. Exit
Enter your choice: 1
Enter name: Saihan
Enter id: 12
Enter course name (or 'done' to finish): CSE
Enter grade for CSE: 4.00
Enter course name (or 'done' to finish): done
Student added successfully.
```

Console Input : Searching a student in the system ,

```
1. Add student
2. Search student
3. Print result card
4. Exit
Enter your choice: 2
Enter id: 12
Student found: Saihan
```

Console Input : Printing the result of a student in the system ,

```
1. Add student
2. Search student
3. Print result card
4. Exit
Enter your choice: 3
Enter id: 33
Name: Md Saihan Alam
ID: 33
Courses: [CSE, GED - 201]
Grades: [4.00, 3.85]
```

Future Improvement :

In the future to enhance the Student Management System, several features can be added:

1. **Showing Attendance :** It will show a student attendance .
2. **Top Student :** It will show the merit list of the student in the semester.
3. **Student Profiles:** Enhance the system to include additional student details, such as contact information and enrollment status.
4. **Grade Calculation:** Add features to compute and display average grades, and perhaps track student performance over time.
5. **User Interface Enhancements:** Develop a graphical user interface (GUI) for a more intuitive and interactive experience, moving beyond the command-line interface.

Project Conclusion :

This project successfully implements a basic Student Management System using Java. The system allows for the addition of students, searching by ID, and printing of result cards.

The design effectively supports the core functionalities needed for managing student data, and adheres to object-oriented principles. Future improvements could focus on enhancing error handling, adding data persistence, and developing a more user-friendly interface. Overall, this system provides a solid foundation for further development and refinement.