

User Manual

Student Management System

Version: 1.0

Date: December 2025

Table of Contents

- 1. [Introduction](#)
 - 2. [System Requirements](#)
 - 3. [Installation and Setup](#)
 - 4. [Getting Started](#)
 - 5. [Features and Operations](#)
 - 6. [Troubleshooting](#)
 - 7. [FAQ](#)
-

1. Introduction

1.1 About the System

The Student Management System is a console-based application designed to help administrators manage student records, course enrollments, and academic performance. Built using Java and Object-Oriented Programming principles, it provides a simple yet powerful interface for educational institutions.

1.2 Key Features

- ✓ Secure admin login system
- ✓ Complete student record management (Create, Read, Update, Delete)
- ✓ Automatic unique ID generation for students
- ✓ Course assignment and management
- ✓ Automatic GPA calculation (4.0 scale)
- ✓ Advanced search capabilities
- ✓ Statistical reports and analytics
- ✓ Comprehensive input validation
- ✓ User-friendly menu-driven interface

1.3 Who Should Use This Manual

- System Administrators
 - School Administrative Staff
 - IT Support Personnel
 - Anyone managing student records
-

2. System Requirements

2.1 Hardware Requirements

- **Processor:** Any modern CPU (1 GHz or higher)
- **RAM:** Minimum 2 GB
- **Storage:** 50 MB free disk space
- **Display:** Console/Terminal support

2.2 Software Requirements

- **Java:** JDK 8 or higher
- **Operating System:** Windows 7+, Linux, or macOS 10.10+
- **Terminal:** Command Prompt, PowerShell, or Unix Terminal

2.3 Checking Java Installation

Open terminal/command prompt and type:

```
java -version
```

You should see Java version information. If not, download and install Java from [Oracle's website](#) or use OpenJDK.

3. Installation and Setup

3.1 Downloading the System

1. Download the Student Management System source code
2. Extract to a folder of your choice (e.g., `C:\StudentManagement`)

3.2 Compilation

Navigate to the source directory and compile:

Windows:

```
cd C:\StudentManagement\src\main\java
javac com\studentmanagement\*.java com\studentmanagement\model\*.java
com\studentmanagement\service\*.java com\studentmanagement\util\*.java
```

Linux/Mac:

```
cd /path/to/StudentManagement/src/main/java
javac com/studentmanagement/*.java com/studentmanagement/model/*.java
com/studentmanagement/service/*.java com/studentmanagement/util/*.java
```

3.3 Running the Application

After compilation, run:

Windows:

```
java com.studentmanagement.StudentManagementSystemApp
```

Linux/Mac:

```
java com.studentmanagement.StudentManagementSystemApp
```

4. Getting Started

4.1 First Launch

When you first launch the application, you'll see a welcome banner:

```
STUDENT MANAGEMENT SYSTEM - OOP PROJECT
Console-Based Application
```

4.2 Logging In

Default Credentials:

- **Username:** admin
- **Password:** admin123

Important Notes:

- You have 3 attempts to log in correctly
- After 3 failed attempts, the system will exit
- Login is case-sensitive

Example Login:

```
--- ADMIN LOGIN ---
Username: admin
Password: admin123
✓ Login successful! Welcome to Student Management System.
```

4.3 Main Menu

After successful login, you'll see the main menu:

MAIN MENU

1. Create New Student

2. View All Students

3. View Student Details

4. Update Student Information

5. Delete Student

6. Search Students

7. Assign Course to Student

8. Remove Course from Student

9. View Statistics

0. Exit

5. Features and Operations

5.1 Create New Student

Menu Option: 1

Purpose: Add a new student to the system with automatic ID generation.

Step-by-Step Guide:

1. Select option 1 from main menu

2. Enter student information:

◦ **First Name:** Letters and spaces only (e.g., "John")

◦ **Last Name:** Letters and spaces only (e.g., "Doe")

◦ **Email:** Valid email format (e.g., "john.doe@example.com")

◦ **Age:** Number between 1 and 149

Example:

--- CREATE NEW STUDENT ---
First Name: John
Last Name: Doe
Email: john.doe@example.com
Age: 20

✓ Student created successfully!
Student ID: STU1001

Validation Rules:

- Names cannot contain numbers or special characters
- Email must have @ and domain (e.g., .com, .edu)
- Age must be a positive number less than 150

Tips:

- The system generates a unique ID automatically (STU1001, STU1002, etc.)
- All fields are required
- If you enter invalid data, you'll be prompted to re-enter

5.2 View All Students

Menu Option: 2

Purpose: Display a list of all students in the system.

What You'll See:

```
--- ALL STUDENTS ---
Total Students: 3

=====
=====
ID           Name           Email           Age    GPA
=====
=====
STU1001      John Doe      john.doe@example.com    20    3.50
STU1002      Jane Smith   jane.smith@example.com  22    3.75
STU1003      Bob Johnson  bob.j@example.com       19    3.20
=====
=====
```

Information Displayed:

- Student ID
- Full Name (First + Last)
- Email Address
- Age
- Current GPA (0.0 if no courses)

Tips:

- If no students exist, you'll see "No students found in the system"
- Students are displayed in the order they were created

5.3 View Student Details

Menu Option: 3

Purpose: View detailed information about a specific student including all enrolled courses.

Step-by-Step Guide:

- 1. Select option 3
- 2. Enter the Student ID (e.g., STU1001)

Example:

```
--- VIEW STUDENT DETAILS ---
Enter Student ID: STU1001

=====
Student ID: STU1001
Name: John Doe
Email: john.doe@example.com
Age: 20
GPA: 3.50
Enrolled Courses:
  - CS101 - Introduction to Programming (Credits: 3, Grade: 90.0%, Letter:
A)
  - MATH201 - Calculus I (Credits: 4, Grade: 85.0%, Letter: B)
=====
```

Information Displayed:

- All basic student information
- Complete list of enrolled courses with:
 - Course code and name
 - Credit hours
 - Grade percentage
 - Letter grade (A, B, C, D, F)
- Calculated GPA

Tips:

- Student ID search is case-insensitive (stu1001 = STU1001)
- If student doesn't exist, you'll see an error message

5.4 Update Student Information

Menu Option: 4

Purpose: Modify specific fields of a student's record.

Step-by-Step Guide:

- 1. Select option 4
- 2. Enter the Student ID
- 3. Choose which field to update:

- 1: First Name
 - 2: Last Name
 - 3: Email
 - 4: Age
4. Enter the new value

Example:

```
--- UPDATE STUDENT INFORMATION ---
Enter Student ID: STU1001

Current Information:
Student ID: STU1001
Name: John Doe
Email: john.doe@example.com
Age: 20
GPA: 3.50
...

Select field to update:
1. First Name
2. Last Name
3. Email
4. Age
Enter choice: 3
New Email: john.newemail@example.com

✓ Student information updated successfully!
```

Important Notes:

- Student ID cannot be changed (it's permanent)
- Only the selected field is updated
- All validation rules apply to new values
- GPA is not manually updatable (auto-calculated)

Tips:

- You can update one field at a time
- To update multiple fields, repeat the process
- The system shows current info before updating

5.5 Delete Student

Menu Option: 5

Purpose: Remove a student record from the system.

Step-by-Step Guide:

1. Select option 5
2. Enter the Student ID
3. Review the student information displayed
4. Confirm deletion by typing "yes" or "y"

Example:

```
--- DELETE STUDENT ---  
Enter Student ID: STU1001  
  
Student to be deleted:  
Student ID: STU1001  
Name: John Doe  
Email: john.doe@example.com  
...  
  
Are you sure you want to delete this student? (yes/no): yes  
  
✓ Student deleted successfully!
```

Important Warnings:

- ⚠ Deletion is permanent and cannot be undone
- ⚠ All associated courses are also removed
- The system requires explicit confirmation

Tips:

- Type "no" or "n" to cancel deletion
- Review information carefully before confirming
- Deleted student IDs are not reused

5.6 Search Students

Menu Option: 6

Purpose: Find students by ID, name, or email.

Step-by-Step Guide:

1. Select option 6
2. Enter search term (can be partial match)

Search Capabilities:

- Search by Student ID (e.g., "STU1001")
- Search by First Name (e.g., "John")
- Search by Last Name (e.g., "Doe")
- Search by Email (e.g., "example.com")

Example:

```
--- SEARCH STUDENTS ---
Enter search term (ID, name, or email): john

Found 2 student(s):

=====
=====
ID           Name           Email           Age    GPA
=====
=====
STU1001      John Doe       john.doe@example.com    20    3.50
STU1005      Johnny Smith   johnny.s@example.com    21    3.25
=====
=====
```

Tips:

- Search is case-insensitive
- Partial matches work (searching "john" finds "Johnny" too)
- Search across all fields simultaneously
- Empty search term is not allowed

5.7 Assign Course to Student

Menu Option: 7

Purpose: Enroll a student in a course and record their grade. GPA is calculated automatically.

Step-by-Step Guide:

1. Select option 7
2. Enter Student ID
3. Enter Course Code (format: 2-4 letters + 3 digits, e.g., CS101)
4. Enter Course Name
5. Enter Credits (1-10)
6. Enter Grade (0-100)

Example:

```
--- ASSIGN COURSE TO STUDENT ---
Enter Student ID: STU1001

Student: John Doe
Course Code (e.g., CS101, MATH201): CS101
Course Name: Introduction to Programming
Credits (1-10): 3
Grade (0-100): 90
```

✓ Course assigned successfully!
Updated GPA: 4.00

Course Code Format Rules:

- Must start with 2-4 uppercase letters
- Must end with exactly 3 digits
- Valid examples: CS101, MATH201, PHYS150
- Invalid examples: C1, CS1, 101CS

Grade Scale:

- A (90-100): 4.0 grade points
- B (80-89): 3.0 grade points
- C (70-79): 2.0 grade points
- D (60-69): 1.0 grade points
- F (0-59): 0.0 grade points

GPA Calculation:
$$\text{GPA} = \text{Sum of (Grade Points} \times \text{Credits)} / \text{Total Credits}$$
Example Calculation:

- Course 1: CS101 (3 credits, 90% = 4.0 points)
- Course 2: MATH201 (4 credits, 85% = 3.0 points)
- $\text{GPA} = (4.0 \times 3 + 3.0 \times 4) / (3 + 4) = 24 / 7 = 3.43$

Tips:

- A student can have multiple courses
- GPA updates automatically after each course assignment
- You can assign the same course multiple times (e.g., if retaken)

5.8 Remove Course from Student

Menu Option: 8

Purpose: Remove a course enrollment from a student's record. GPA is recalculated automatically.

Step-by-Step Guide:

1. Select option 8
2. Enter Student ID
3. Review the list of enrolled courses
4. Enter the Course Code to remove

Example:

```
--- REMOVE COURSE FROM STUDENT ---
Enter Student ID: STU1001

Student: John Doe
Enrolled Courses:
  - CS101: Introduction to Programming
  - MATH201: Calculus I

Enter Course Code to remove: CS101

✓ Course removed successfully!
Updated GPA: 3.00
```

Tips:

- The system shows all enrolled courses before removal
- Course code must match exactly
- GPA is recalculated after removal
- If student has no courses, you'll be notified

5.9 View Statistics

Menu Option: 9

Purpose: View system-wide statistics and analytics.

What You'll See:

```
--- SYSTEM STATISTICS ---

=====
Total Students: 10
Students with Courses: 8
Average GPA: 3.25

Top Students (GPA >= 3.0):
  - Jane Smith (ID: STU1002, GPA: 3.75)
  - John Doe (ID: STU1001, GPA: 3.50)
  - Alice Brown (ID: STU1004, GPA: 3.40)
=====
```

Statistics Provided:

- **Total Students:** Total number of students in system
- **Students with Courses:** Number of students enrolled in at least one course
- **Average GPA:** Mean GPA across all students

- **Top Students:** List of students with GPA 3.0 or higher

Tips:

- Use this to get an overview of system data
 - Identify high-performing students quickly
 - Monitor overall academic performance
-

5.10 Exit Application

Menu Option: 0

Purpose: Safely close the application.

What Happens:

```
Thank you for using Student Management System. Goodbye!  
[Application closes]
```

Important Notes:

- ⚠ All data is stored in memory only
- ⚠ Data is NOT saved between sessions
- ⚠ When you exit, all student records are lost

Tips:

- Make sure to record important information before exiting
 - Future versions may include data persistence
-

6. Troubleshooting

6.1 Common Issues and Solutions

Issue: "java: command not found" or "javac: command not found"

Solution:

1. Verify Java is installed: `java -version`
2. If not installed, download from [Oracle](#)
3. Ensure Java is in your system PATH

Issue: "ClassNotFoundException"

Solution:

1. Ensure you're running from the correct directory
2. Check that all .class files were compiled

3. Use fully qualified class name: `com.studentmanagement.StudentManagementSystemApp`

Issue: "Cannot find or load main class"

Solution:

- 1. Run from the `src/main/java` directory
- 2. Use correct package path
- 3. Verify compilation was successful (check for `.class` files)

Issue: Login fails repeatedly

Solution:

- Double-check credentials:
 - Username: `admin` (lowercase)
 - Password: `admin123`
- Login is case-sensitive
- After 3 failures, restart the application

Issue: Invalid input errors

Solution:

- Check format requirements for each field
- Names: letters and spaces only
- Email: must include `@` and domain
- Age: 1-149
- Course codes: 2-4 letters + 3 digits

Issue: Student not found

Solution:

- Verify the Student ID is correct
- IDs are case-insensitive but must match
- Use option 2 to see all student IDs
- Student may have been deleted

6.2 Input Validation Messages

Message	Meaning	Solution
"Invalid name"	Name contains numbers/symbols	Use only letters and spaces
"Invalid email format"	Email doesn't follow format	Use format: name@domain.com
"Invalid age"	Age out of range	Enter age between 1-149
"Invalid course code"	Wrong format	Use format: CS101, MATH201
"Invalid grade"	Grade out of range	Enter grade between 0-100

Message	Meaning	Solution
"Invalid credits"	Credits out of range	Enter credits between 1-10

7. FAQ

Q1: Can multiple users log in simultaneously?

A: No, this is a single-user system designed for one administrator.

Q2: Is the data saved when I exit?

A: No, all data is stored in memory and lost when the application closes. This is intentional for the current version.

Q3: Can I change the admin password?

A: The current version has hardcoded credentials. You would need to modify the source code to change them.

Q4: What happens if I enter the same student twice?

A: Each student gets a unique ID, so you can create multiple records with similar information. They are treated as separate students.

Q5: Can a student be enrolled in the same course twice?

A: Yes, this allows for scenarios where students retake courses.

Q6: How is GPA calculated?

A: GPA is calculated as a weighted average: $\text{Sum of (Grade Points} \times \text{Credits)} / \text{Total Credits}$, on a 4.0 scale.

Q7: What is the maximum number of students?

A: The system can handle thousands of students, limited only by available memory.

Q8: Can I export student data?

A: The current version does not support data export. This feature may be added in future versions.

Q9: What if I accidentally delete a student?

A: Deletion is permanent. There is no undo function. Always confirm carefully before deleting.

Q10: Can I assign courses without grades?

A: No, a grade (0-100) is required when assigning a course. You can enter 0 as a placeholder if needed.

Q11: Does the system validate course codes against a database?

A: No, course codes only need to follow the format (letters + numbers). Any valid format is accepted.

Q12: Can I update a course grade after assignment?

A: The current version requires removing and re-adding the course with the new grade.

8. Best Practices

8.1 Data Entry

- Always double-check student information before saving
- Use consistent email domain for your institution
- Use standard course code formats for your school

8.2 Regular Operations

- Review student list regularly (Option 2)
- Use search function to quickly find students (Option 6)
- Check statistics to monitor overall performance (Option 9)

8.3 Data Management

- Keep a backup list of student IDs
 - Document any important student information externally
 - Confirm deletions carefully
-

9. Getting Help

9.1 Support Resources

- **User Manual:** This document
- **System Requirements:** Section 2
- **Troubleshooting:** Section 6
- **FAQ:** Section 7

9.2 Additional Information

For questions or issues not covered in this manual, please contact your system administrator or IT support team.

10. Version History

Version	Date	Changes
1.0	December 2025	Initial release

Document Prepared By: Documentation Team

Last Updated: December 2025

For: Student Management System v1.0

Quick Reference Card

Main Menu Options:

1 - Create Student	6 - Search Students
2 - View All Students	7 - Assign Course
3 - View Student Details	8 - Remove Course
4 - Update Student	9 - View Statistics
5 - Delete Student	0 - Exit

Default Login:

Username: admin

Password: admin123

Valid Input Formats:

Names: Letters and spaces only

Email: name@domain.com

Age: 1-149

Course Code: CS101, MATH201 (2-4 letters + 3 digits)

Grade: 0-100

Credits: 1-10

Grade Scale (4.0 GPA):

A: 90-100 (4.0)

B: 80-89 (3.0)

C: 70-79 (2.0)

D: 60-69 (1.0)

F: 0-59 (0.0)

End of User Manual