SKILL-BASED MINI PROJECT

STUDENT DATABASE MANAGEMENT SYSTEM

ABSTRACT

The Student Database Management System (SDMS) is an application designed to manage and streamline the academic records of students across multiple branches of an institution. Developed using C++, the program utilizes essential components of the C++ Standard Library, such as file streams for handling CSV files and vectors for dynamic data storage. The SDMS allows for efficient management of student information, including the addition and removal of students, course registration, and updating of academic marks. This Student Database management System is made based upon the marking schemes of Madhav Institute of Technology and Science, Gwalior. The system has features like updating marks of MidSem-1, MidSem-2, Proficiency, Quiz & Assignment and Endsem. Key functionalities include exporting and importing student data in CSV format, generating detailed reports, and calculating Grade Point Averages (GPA). The system ensures data persistence by saving all changes to a central **database.csv** file, which is read upon program initialization to maintain data integrity. Through a command-line interface, the SDMS provides an accessible and user-friendly platform for academic administrators to handle student records efficiently, ensuring accurate and up-to-date information is readily available for reporting and analysis. This system enhances the administrative capabilities of educational institutions by providing a solution for academic record management.

Chapter 1: INTRODUCTION

The Student Database Management System (SDMS) is a comprehensive solution designed for academic institutions to efficiently store, manage, and access students' academic records across all branches. This system provides a centralized repository for academic data, enabling institutions to maintain accurate records and generate detailed reports in various formats, such as CSV, which can be directly shared with students and other stakeholders.

1.1 About Student Data Management System

The SDMS is developed to facilitate the management of students' academic records, including enrollment details, course registrations, and performance metrics. The system supports various functionalities such as adding and removing students, updating and retrieving marks for different assessments, and calculating grades and CGPA (Cumulative Grade Point Average). It is built to ensure data integrity and provide an easy-to-use interface for administrators.

1.2 Parameters Considered

The system considers several parameters for managing academic records:

- **Student Information:** Name, enrollment number, and list of registered courses.
- **Course Details:** Course name, course code, and individual marks for different assessments (mid-semester exams, end-semester exams, quizzes, assignments, and proficiency tests).
- **Assessment Marks:** Specific marks obtained in various assessments.
- **Grades and CGPA Calculation:** Grades based on marks and overall CGPA calculation for each student.

1.3 Objective

The primary objective of SDMS is to streamline the academic record management process for institutions by:

1. Providing a reliable and user-friendly database system.
2. Ensuring data consistency and accuracy.
3. Enabling easy data import and export in multiple formats.
4. Facilitating quick access to student records and performance reports.
5. Supporting various academic operations such as grade calculation, course management, and student performance analysis.

Chapter 2: TECHNOLOGIES USED

2.1 Tools and Libraries

- **C++ Standard Library:** For data handling and manipulation.
- **File Streams (<fstream>):** For reading from and writing to CSV files.
- **Vector (<vector>):** To store dynamic lists of students and courses.
- **I/O Manipulation (<iomanip>):** For formatting output data.

2.2 Hardware Requirements

- **Processor:** Modern multi-core processor.
- **Memory:** Minimum of 4GB RAM.
- **Storage:** Adequate disk space to store the database files and generated reports.
- **Operating System:** Cross-platform support (Windows, Linux, macOS).

Chapter 3: METHODOLOGY

3.1 Data Structure and Storage

1. **Class Definitions:** The program uses multiple classes to represent various entities in the system.
    - **Course Class:** Stores course information and marks for different assessments.
    - **Student Class:** Stores student information and their associated courses.
    - **Semester Class:** Manages students and courses for a particular semester, including methods for adding students, importing student data from CSV files, and updating marks.
2. **Database Management:**
    - The data is stored in a **database.csv** file, which is read at the start of the program to initialize the in-memory database.
    - Changes made during the program execution are saved back to the **database.csv** file to ensure data persistence.

3.2 Implementation Steps

1. **Reading Database File:**
    - On program startup, the **database.csv** file is read to populate the list of students and their respective courses. This ensures that the latest data is always available for operations.

2. **Adding and Managing Students:**
    - **Add Student:** New students can be added by providing their name and enrollment number. The student is then associated with the existing list of courses for the semester.
    - **Import Students:** A CSV file containing student details can be imported to add multiple students at once.
    - **Remove Student:** Students can be removed based on their enrollment number.
3. **Managing Courses:**
    - **Add Course:** New courses can be added by providing the course name and course code. All students are updated with the new course.
    - **Update Marks:** Marks for different assessments (MidSem1, MidSem2, Proficiency, Quiz/Assignment, and EndSem) can be updated by entering the relevant course code and marks for each student.
4. **Calculating Grades and CGPA:**
    - Each course's marks are calculated using predefined weightage and grading criteria.
    - The CGPA is calculated for each student based on the average marks obtained across all registered courses.
5. **Generating Reports:**
    - The system provides functionality to generate and print various reports, including individual assessment marks, course averages, and grades.
    - Reports can be exported in CSV format for easy sharing and record-keeping.
6. **User Interface:**
    - A command-line interface guides the user through various operations such as adding students, updating marks, and generating reports.
    - Prompts ensure that the user provides valid inputs for each operation, maintaining data integrity.

3.3 Program Flow

1. **Initialization:**
    - Read the **database.csv** file to load existing student and course data.
2. **Main Menu:**
    - Display options for managing students, courses, and assessments.
3. **Operations:**

- Based on user selection, perform operations such as adding/removing students, updating marks, and printing reports.

4. **Data Persistence:**
    - After each operation, save the updated data back to the **database.csv** file to ensure changes are not lost.
5. **Exit:**
    - Cleanly exit the program after saving all changes, ensuring the database is up-to-date.

Chapter 4: ALGORITHM

Initialization

1. **Start**
2. **Initialize Variables:**
    - Create an empty list of **students**.
    - Create an empty list of **courses**.
3. **Read database.csv:**
    - Open **database.csv** for reading.
    - For each line in the file:
        - Parse student information and associated courses.
        - Create **Student** and **Course** objects and add them to respective lists.
    - Close **database.csv**.

Main Menu

4. **Display Main Menu:**
    - Options: Add Student, Remove Student, Add Course, Update Marks, Export Marks, Print Marks, Generate Report, Export Report, Display Students, Import Students, Exit.
5. **Get User Selection:**
    - Prompt user for input.

Add Student

6. **If Add Student Selected:**
    - Prompt for student name and enrollment number.

- Create a new **Student** object.
- Add the new student to the **students** list.
- Associate the student with existing courses.
- Save changes to **database.csv**.

Remove Student

7. **If Remove Student Selected:**
    - Prompt for student enrollment number.
    - Search for the student in the **students** list.
    - If found, remove the student from the list.
    - Save changes to **database.csv**.

Add Course

8. **If Add Course Selected:**
    - Prompt for course name and course code.
    - Create a new **Course** object.
    - Add the new course to the **courses** list.
    - Update all students to include the new course.
    - Save changes to **database.csv**.

Update Marks

9. **If Update Marks Selected:**
    - Prompt for course code.
    - Search for the course in the **courses** list.
    - For each student, prompt for marks in MidSem1, MidSem2, Proficiency, Quiz/Assignment, and EndSem.
    - Update the marks in the respective **Course** object for each student.
    - Save changes to **database.csv**.

Export Marks

10. **If Export Marks Selected:**
    - Open a new CSV file for writing.
    - For each student, write their marks to the CSV file.

- Close the CSV file.

Print Marks

11. **If Print Marks Selected:**
    - For each student, print their marks to the console or a connected printer.

Generate Report

12. **If Generate Report Selected:**
    - Prompt for report type: Individual Student Report, Course Report, Overall Performance.
    - Generate the report based on the selected type.
    - Display the report to the user.

Export Report

13. **If Export Report Selected:**
    - Prompt for report type: Individual Student Report, Course Report, Overall Performance.
    - Generate the report based on the selected type.
    - Open a new CSV file for writing.
    - Write the report to the CSV file.
    - Close the CSV file.

Display Students

14. **If Display Students Selected:**
    - For each student in the **students** list, display their information and associated courses.

Import Students

15. **If Import Students Selected:**
    - Prompt for the filename of the CSV file containing student data.
    - Open the CSV file for reading.
    - For each line in the file, parse the student information and add it to the **students** list.
    - Save changes to **database.csv**.

Exit

16. **If Exit Selected:**
    - Save all changes to **database.csv**.
    - End program.

Chapter 5: RESULTS AND CONCLUSIONS

5.1 Output

```
PS A:\AKSHARA RATHORE (08) & VAIBHAV SHARMA(69)\MINI> .\studentDatabase
                WELCOME TO STUDENT DATABASE MANAGEMENT SYSTEM
                          SKILL BASED MINI PROJECT
                                  Made by
                        AKSHARA RATHORE (0901AD231008)
                        VAIBHAV SHARMA (0901AD231069)

1. ADD BRANCH
2. ADD COURSE
3. ADD STUDENTS
4. IMPORT STUDENT LIST
5. UPDATE MARKS
6. REMOVE STUDENT
7. DISPLAY STUDENTS
8. DISPLAY MARKS
9. GENERATE REPORT
10. EXPORT MARKS
11. EXPORT MARKS FOR PARTICULAR EXAM
12. EXPORT/SAVE DATABASE
13. CLEAR SCREEN
0. EXIT

Enter Function no.: 1
Enter Branch Name: AIADS
BRANCH ADDED SUCCESSFULLY

Enter Function no.: 2
Enter Branch: AIADS
Enter semester: 2
Enter Course name: OOPs
Enter Course Code: 3270223
COURSE ADDED SUCCESSFULLY

Enter Function no.: 3
Enter Branch: AIADS
Enter Semester: 2
Enter exit to stop
Enter name: AKSHARA RATHORE
Enter enrol: 0901AD231008
Enter name: VAIBHAV SHARMA
Enter enrol: 0901AD231069
Enter name: exit
STUDENTS ADDED SUCCESSFULLY

Enter Function no.: 5
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. All Student
2. Particular Student
0. Return
Enter choice: 1

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 1
Enter marks
AKSHARA RATHORE : 20
VAIBHAV SHARMA : 19

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 2
```

```
Enter marks
AKSHARA RATHORE : 19
VAIBHAV SHARMA : 18

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 3
Enter marks
AKSHARA RATHORE : 9
VAIBHAV SHARMA : 10

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 4
Enter marks
AKSHARA RATHORE : 19
VAIBHAV SHARMA : 19

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 5
Enter marks
AKSHARA RATHORE : 50
VAIBHAV SHARMA : 49

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 0
MARKS UPDATED SUCCESSFULLY

Enter Function no.: 7
Enter Branch: AIADS
Enter Semester: 2
             STUDENT LIST
S.no.     Enrollment no.      Name
1         0901AD231008        AKSHARA RATHORE
2         0901AD231069        VAIBHAV SHARMA
```

```
Enter Function no.: 8
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
5. EndSem Marks
6. Course Average
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
Enter Choice: 9
COURSE:                                                    3270223
S.no.     Enrollment no.      Name                         MidSem 1       MidSem 2       Average Marks
1         0901AD231008        AKSHARA RATHORE              20             19             19
2         0901AD231069        VAIBHAV SHARMA               19             18             18

Enter Function no.: 8
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
5. EndSem Marks
6. Course Average
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
Enter Choice: 8
COURSE:                                                                  3270223
S.no. Enrollment no.  Name                         Mid1  Mid2  Prof. Q/A   EndSem  Avg Marks  Grade
1     0901AD231008    AKSHARA RATHORE              20    19    9     19    50      97         A+
2     0901AD231069    VAIBHAV SHARMA               19    18    10    19    49      96         A+

Enter Function no.: 5
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. All Student
2. Particular Student
0. Return
Enter choice: 2

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 2
```

```
Enter exit to stop
Enrollment no.: 0901AD231069
Enter New Marks: 20
Enrollment no.: exit

Enter 1 to update MidSem1 Marks
Enter 2 to update MidSem2 Marks
Enter 3 to update Proficiency Marks
Enter 4 to update Quiz & Assignment Marks
Enter 5 to update EndSem Marks
Enter 0 to return
Enter Choice: 0
MARKS UPDATED SUCCESSFULLY

Enter Function no.: 8
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
5. EndSem Marks
6. Course Average
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
```

```
Enter Choice: 8
COURSE:                                                    3270223
S.no. Enrollment no.  Name                    Mid1  Mid2  Prof. Q/A   EndSem  Avg Marks  Grade
1     0901AD231008    AKSHARA RATHORE         20    19    9     19    50      97         A+
2     0901AD231069    VAIBHAV SHARMA          19    20    10    19    49      97         A+

Enter Function no.: 9
Enter Enrollment no.: 0901AD231008
1. Display
2. Export as TXT
0. return
```

```
Enter Choice: 1
 ------------------------------------------------------------
          Madhav Institute of Technology & Science
    Race Course Road, Gole ka Mandir, Gwalior,M.P.-474005
 ------------------------------------------------------------
                       REPORT CARD
 ------------------------------------------------------------
Name:          AKSHARA RATHORE     Roll no.:     0901AD231008
Course:        B.tech              Branch:       AIADS
Semester:      2                   Status:       Regular
 ------------------------------------------------------------
Course Code    Course               Score     Grade
 ------------------------------------------------------------
3270223         OOPs                97        A+
 ------------------------------------------------------------
Result Des.:        PASS      CGPA:               10
 ------------------------------------------------------------

Enter Function no.: 12
DATABASE EXPORTED/SAVED SUCCESSFULLY

Enter Function no.: 10
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. TXT
2. CSV
```

```
3. EXIT
Enter Choice: 2
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
5. EndSem Marks
6. Course Average
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
Enter Choice: 1
Enter file name (without extension) : AIADS sem 2 midsem 1
MARKS EXPORTED SUCCESSFULLY

Enter Function no.: 10
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. TXT
2. CSV
3. EXIT
Enter Choice: 2
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
```

```
5. EndSem Marks
6. Course Average
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
Enter Choice: 8
Enter file name (without extension) : AIADS SEM 2 ALL MARKS
MARKS EXPORTED SUCCESSFULLY

Enter Function no.: 4
Enter Branch: AIADS
Enter Semester: 3
Enter filename (wihtout .csv): student
STUDENTS IMPORTED SUCCESSFULLY
```

```
Enter Function no.: 7
Enter Branch: AIADS
Enter Semester: 3
STUDENT LIST
S.no.     Enrollment no.      Name
1         0901AD231001        ADITI SHARMA
2         0901AD231002        GAURI SHARMA
3         0901AD231008        AKSHARA RATHORE
4         0901AD231069        VAIBHAV SHARMA
5         0901AD231070        VANSH RATHORE
6         0901AD231071        VEDIKA SHARMA

Enter Function no.: 10
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. TXT
2. CSV
3. EXIT
Enter Choice: 1
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
5. EndSem Marks
6. Course Average
```

```
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
Enter Choice: 8
Enter file name (without extension) : MARKSLISTAIDS
MARKS EXPORTED SUCCESSFULLY

Enter Function no.: 10
Enter Branch: AIADS
Enter Semester: 2
Enter Course Code: 3270223
1. TXT
2. CSV
3. EXIT
```

```
Enter Choice: 1
1. MidSem 1 Marks
2. MidSem 2 Marks
3. Proficiency Marks
4. Quiz/Assignment Marks
5. EndSem Marks
6. Course Average
7. Course Grade
8. ALL MARKS
9. ALL MIDSEM MARKS
Enter Choice: 2
Enter file name (without extension) : MIDSEM2
MARKS EXPORTED SUCCESSFULLY

Enter Function no.: 6
Enter Branch: AIADS
Enter Semester: 3
Enter enrollment no. to remove student: 0901AD231070
STUDENT REMOVED SUCCESSFULLY
```

```
Enter Function no.: 7
Enter Branch: AIADS
Enter Semester: 3
STUDENT LIST
S.no.     Enrollment no.      Name
1         0901AD231001        ADITI SHARMA
2         0901AD231002        GAURI SHARMA
3         0901AD231008        AKSHARA RATHORE
4         0901AD231069        VAIBHAV SHARMA
5         0901AD231071        VEDIKA SHARMA

Enter Function no.: 0
THANKYOU FOR USING STUDENT DATABASE MANAGEMENT SYSTEM
SKILL BASED MINI PROJECT
Made by
AKSHARA RATHORE (0901AD231008)
VAIBHAV SHARMA (0901AD231069)
PS A:\AKSHARA RATHORE (08) & VAIBHAV SHARMA(69)\MINI>
```

MINI > AIADS SEM 2 ALL MARKS.csv

```
1  COURSE: 3270223
2  S.no.,Enrollment no.,Name,MidSem 1,MidSem 2,Proficiency,Quiz/Assignment,EndSem,Course Marks,Course Grade
3  1,0901AD231008,AKSHARA RATHORE,20,19,9,19,50,97,A+
4  2,0901AD231069,VAIBHAV SHARMA,19,20,10,19,49,97,A+
5
```

MINI > database.csv

```
Branch,AIADS
Sem,1,
Sem,2,OOPs,3270223,
AKSHARA RATHORE,0901AD231008,20,19,9,19,50,
VAIBHAV SHARMA,0901AD231069,19,20,10,19,49,
Sem,3,
ADITI SHARMA,0901AD231001,
GAURI SHARMA,0901AD231002,
AKSHARA RATHORE,0901AD231008,
VAIBHAV SHARMA,0901AD231069,
VANSH RATHORE,0901AD231070,
VEDIKA SHARMA,0901AD231071,
Sem,4,
Sem,5,
Sem,6,
Sem,7,
Sem,8,
```

	A	B	C	D	E	F	G	H	I	J	K
1	COURSE: 3270223										
2	S.no.	Enrollment no.	Name	MidSem 1	MidSem 2	Proficiency	Quiz/Assig	EndSem	Course Ma	Course Grade	
3	1	0901AD231008	AKSHARA RATHORE	20	19	9	19	50	97	A+	
4	2	0901AD231069	VAIBHAV SHARMA	19	20	10	19	49	97	A+	
5											
6											

MINI > AIADS sem 2 midsem 1.csv

```
MIDSEM 1 MARKS: 3270223
S.no.,Enrollment no.,Name,Marks
1,0901AD231008,AKSHARA RATHORE,20
2,0901AD231069,VAIBHAV SHARMA,19

```

MINI > MIDSEM2.txt

```
                MIDSEM 2 MARKS: 3270223
S.no.       Enrollment no.      Name                    Marks
1           0901AD231008        AKSHARA RATHORE         19
2           0901AD231069        VAIBHAV SHARMA          20

```

5.2 Results and Discussions

The Student Database Management System (SDMS) generates a variety of outputs designed to facilitate the management and analysis of students' academic records. The primary outputs include detailed reports and data exports in multiple formats, ensuring easy access and distribution of information. When the user selects the **Export Marks** option, the system creates a CSV file containing the marks of all students across different assessments, allowing for straightforward data sharing and record-keeping. The **Print Marks** feature enables printing students' marks directly to the console or a connected printer, providing a quick overview of academic performance. The **Generate Report** functionality produces comprehensive reports based on the selected type (e.g., individual student report, course report, or overall performance), displaying the information on the screen for immediate review. These reports can also be exported to CSV files through the **Export Report** option, ensuring that detailed analyses can be shared or stored as needed. The **Display Students** option provides a list of all students along with their enrolled courses and associated marks, offering a clear and organized view of the current academic database. These outputs collectively ensure that the SDMS effectively supports academic administration by providing accurate, accessible, and versatile data management solutions.

5.3 Future Scopes

The Student Database Management System has significant potential for future development and enhancement. Some areas for future work include:

- **Integration with Other Systems:** Enhancing the system to integrate with other institutional software, such as Learning Management Systems (LMS) and Human Resource Management Systems (HRMS), to create a more cohesive IT infrastructure.
- **Mobile Application Development:** Creating a mobile version of the SDMS to provide on-the-go access for administrators, teachers, and students, increasing the system's accessibility and convenience.
- **Advanced Analytics and Reporting:** Incorporating advanced data analytics tools to provide deeper insights into student performance trends and predictions, helping educators tailor their teaching strategies.
- **Cloud-Based Solutions:** Migrating the system to a cloud-based platform to improve scalability, accessibility, and data security, allowing institutions to manage their data more effectively.
- **Graphical User Interface (GUI):** Developing a user-friendly interface to replace the console-based interaction, making it accessible to non-technical users.
- **Database Integration:** Transitioning from file-based storage to a database management system (DBMS) like MySQL or SQLite for improved data handling and scalability.