A Project Report for the Bachelor’s in Computer Application

**Student Record Management System**



|  |  |  |
| --- | --- | --- |
| **Name** | **Registration No.** | **Symbol No.** |
| **Babita Khatri** | **2023-1-53-0207** | **23530108** |
| **Laxman Malla** | **2023-1-53-0209** | **23530110** |
| **Niraj Adhikari** | **2023-1-53-0210** | **23530111** |
| **Srishti Sharma** | **2023-1-53-0223** | **23530124** |

**School of Environmental Science and Management**

**Faculty of Science and Technology**

**Pokhara University, Nepal**

**July, 2024**

A Project Report for the Bachelor’s in Computer Application

**Student Record Management System**

**Supervised by**

**Sumant Yadav**

A project submitted in partial fulfilment of the requirements for the degree of project for the Bachelor in Computer Application.

|  |  |  |
| --- | --- | --- |
| **Name** | **Registration No.** | **Symbol No.** |
| **Babita Khatri** | **2023-1-53-0207** | **23530108** |
| **Laxman Malla** | **2023-1-53-0209** | **23530110** |
| **Niraj Adhikari** | **2023-1-53-0210** | **23530111** |
| **Srishti Sharma** | **2023-1-53-0223** | **23530124** |

**School of Environmental Science and Management**

**Faculty of Science and Technology**

**Pokhara University, Nepal**

**July, 2024**

# DECLARATION

We hereby declare that this study entitled **Student Record Management System** is based on our original research work. Related works on the topic by other researchers have been duly acknowledged. The project documentation in this report was mobilized by four below-mentioned second semester undergraduate students of Bachelor of Computer Application (BCA) for the partial fulfillment of the requirement for the project of the second semester under the supervision of **Sumant Yadav.**

Any kind of reproduction of the project and its part for a commercial purpose is strictly prohibited without prior written permission of the author and university. However, this report is open for any kind of academic purpose.

………………………….

Babita Khatri

PU Regd. No. : 2023-1-53-0207

…………………………..

Laxman Malla

PU Regd. No. : 2023-1-53-0209

…………………..……….

Niraj Adhikari

PU Regd. No. : 2023-1-53-0210

…………………………...

Srishti Sharma

PU Regd. No. : 2023-1-53-0223

Date: ………………………

# CERTIFICATE

# ACKNOWLEDGEMENT

All the members of the group have equally contributed in the completion of this project entitled **Student Record Management System.** Foremost, we would like to thank **School of Environmental Science and Management** for assigning this type of project which would help in our future development. We are also thankful to **Sumant Yadav**, the project supervisor for his guidance and support throughout the project.

We would also like to express our sincere thanks to **Kripesh Lamichhane,** subject teacher for his help and interest in our project and also for providing us with his valuable suggestions for the improvement of this project. Additionally, we would like to express our gratitude to all our teachers, seniors, and friends for their support and guidance.

……………………

Babita Khatri

PU Regd. No. : 2023-1-53-0207

……………………

Laxman Malla

PU Regd. No. : 2023-1-53-0209

…………………….

Niraj Adhikari

PU Regd. No. : 2023-1-53-0210

……………………..

Srishti Sharma

PU Regd. No. : 2023-1-53-0223

Date: ………………………

# ABSTRACT

The **STUDENT RECORD MANAGEMENT SYSTEM** is a comprehensive software solution developed to streamline the process of recording and managing students' personal data in educational institutions. This project aims to digitalize the traditional paper-based student record keeping which improves efficiency, accuracy, and accessibility of information. The system allows administrators to maintain student profiles and other relevant details in a centralized database. Features include character line interface for data entry, search, and retrieval, ensuring ease of use for administrators and educators. Through the development of this system, our objective is to enhance administrative effectiveness, reduce paperwork, and provide a reliable platform for managing student data efficiently.

By implementing SRMS, educational institutions can significantly reduce administrative workload, improve data accuracy, and enhance communication between students, teachers, and administrative staff. The system ultimately contributes to efficient educational environment. This project is described in the form of a flow chart that explains the phases of student record management system. The flow chart shows the graphical representation of the procedure used for solving the particular problem.

**Keywords:** Student Record Management System, Database, Effectiveness, Efficiency, Accessibility, Effective communication

TABLE OF CONTENT

[DECLARATION ii](#_Toc172214577)

[CERTIFICATE iii](#_Toc172214578)

[ACKNOWLEDGEMENT iv](#_Toc172214579)

[ABSTRACT v](#_Toc172214580)

[LIST OF FIGURES viii](#_Toc172214581)

[LIST OF APPENDICES ix](#_Toc172214582)

[LIST OF ABBREVIATION/ACRONYMS x](#_Toc172214583)

[CHAPTER 1 1](#_Toc172214584)

[INTRODUCTION 1](#_Toc172214585)

[1.1 Background 1](#_Toc172214586)

[1.2 Statement of the problem 2](#_Toc172214587)

[1.3 Project Objectives 2](#_Toc172214588)

[1.4 Scope and Limitation 2](#_Toc172214589)

[CHAPTER 2 3](#_Toc172214590)

[LITERATURE REVIEW 3](#_Toc172214591)

[CHAPTER 3 4](#_Toc172214592)

[METHODOLOGY 4](#_Toc172214593)

[3.1 Planning Phase 4](#_Toc172214594)

[3.2 Requirement Analysis Phase 5](#_Toc172214595)

[3.3 System Design Phase 5](#_Toc172214596)

[3.4 Development Phase 8](#_Toc172214597)

[3.5 Testing Phase 8](#_Toc172214598)

[CHAPTER 4 9](#_Toc172214599)

[RESULTS AND DISCUSSION 9](#_Toc172214600)

[CHAPTER 5 10](#_Toc172214601)

[CONCLUSION AND FUTURE WORKS 10](#_Toc172214602)

[REFERENCES 11](#_Toc172214603)

[APPENDICES 12](#_Toc172214604)

[1 Main Menu 12](#_Toc172214605)

[2 Add Students 12](#_Toc172214606)

[3 Search Students 13](#_Toc172214607)

[4 Modify Students 13](#_Toc172214608)

[5 Delete Students 14](#_Toc172214609)

[6 Exit 14](#_Toc172214610)

[7 txt file: SRMS.txt before modifying 15](#_Toc172214611)

[8 txt file: SRMS.txt after modifying 15](#_Toc172214612)

[9 txt file: SRMS.txt after deleting 15](#_Toc172214613)

# LIST OF FIGURES

|  |  |
| --- | --- |
| **Figures** | **Page No.** |
| Figure 3.1: Methodology Diagram | 4 |
| Figure 3.2: Level 0 DFD of SRMS | 5 |
| Figure 3.3: Level 1 DFD of SRMS | 6 |
| Figure 3.4: Flow Chart of SRMS | 7 |

# LIST OF APPENDICES

|  |  |
| --- | --- |
| **Appendices** | **Page No.** |
| 1. Main Menu | 12 |
| 2. Add Students | 12 |
| 3. Search Students | 13 |
| 4. Modify Students | 13 |
| 5. Delete Students | 14 |
| 6. Exit | 14 |
| 7. text file: SRMS.txt before modifying | 15 |
| 8. text file: SRMS.txt after modifying | 15 |
| 9. text file: SRMS.txt after deleting | 15 |

# LIST OF ABBREVIATION/ACRONYMS

BCA Bachelor’s in Computer Application

CUI Character User Interface

DFD Data Flow Diagram

HOD Head of Department

IDE Integrated Development Environment

PU Pokhara University

Regd. No. Registration Number

SRMS Student Record Management System

# CHAPTER 1

## INTRODUCTION

### 1.1 Background

A school is a centralized place for providing quality education to the students. Many types of activities related to educational institution and students are occurred inside school. Due to this reason, different tasks related to individual student should be recorded systematically by the school management. In order to make this job easier, we are planning to develop a mini–Student Record Management System in C language. This project serves as a practical demonstration of fundamental concepts in C programming. It provides an opportunity for learners to apply their knowledge in a real-world scenario. After the completion of this project, a user can add a new student's record, modify the record of existing students, search if a student exists or not and delete the student's record from the system. Through continuous refinement and enhancement later, this system may evolve to meet the desired needs of educational institutions in managing student records even more effectively (Tang, Y. F., & Zhang, Y. S, 2009, August).

This system will provide the interface for two types of users: one is an admin who creates an account for the student and another is the student who can check his/her records which will be helping students, teachers and management staffs. It’s like a digital file cabinet for the school. Having a system like this is crucial for schools and other educational institutes too. It makes everything easier to manage like adding new student’s details and modifying or deleting the existing details. Furthermore, in contemporary era characterized by technological advancement, employing software solutions for managing educational record is a smart decision. By working on this project, we’re not only learning about programming but we are also tackling the real-world problem in education sector. If the system will be improved more, it will be a valuable tool for the schools everywhere in the future (Student Information Management System Project, 2019).

### 1.2 Statement of the problem

The current process of managing student’s record is often manual, time consuming and prone to errors. Teachers are manually recording information on a paper-based registers, which is difficult to manage and update. It is also difficult to search for student’s record if the records are large in number (Radhika Bhanushali and et. al.).

### 1.3 Project Objectives

A Student Record Management System (SRMS) is a software solution designed:

to simplify administrative processes associated with managing student’s record

to efficiently manage student related data in digital form to make faculty jobs more accessible by giving them an easy place to find and sort information.

**Key functionalities:**

Add Students’ Information (Enrolling)

Search Students’ Information (Viewing)

Modify Students’ Information (Updating)

Delete Students’ Information (Deleting)

### 1.4 Scope and Limitation

The SRMS will streamline the process of managing student records, making it easier for school administrators to add, modify, search, and delete student information. This eliminates the need for manual recordkeeping, which is often time-consuming and prone to errors. By providing a digital solution for managing student records, it allows teachers and management staff to focus more on educational activities rather than administrative tasks.

But our system also has some sorts of limitations. The system can handle a maximum of 100 student records which limits the number of students that can be managed. The system uses a command-line interface, which might not be as user-friendly. There is no authentication mechanism in this system i.e., any user who can run the program can access and modify all student records.

# CHAPTER 2

## LITERATURE REVIEW

The management of student records has evolved significantly over the years, transitioning from time consuming manual paper-based systems to sophisticated digital systems. This evolution reflects broader trends in information technology and the increasing importance of efficient data management in educational institutions. Historically, student records were maintained in physical files, which posed numerous challenges such as data redundancy, inconsistency, and the risk of physical damage or loss. This method of recording data was too time consuming and prone to errors. These systems were labor-intensive and inefficient, making it difficult to manage large volumes of data effectively, provide security and protect from physical loss and damages. Adopting new technologies can be met with resistance from staff accustomed to traditional methods. With the development of computers in the late 20th century, educational institutions began adopting basic computerized systems to manage student records. These systems reduced redundancy and improved data accuracy. Those systems typically couldn’t integrate functionalities such as data bases to keep the records for future use, student enrollment tracking, academic performance monitoring, and automated reporting, contributing to streamlined administrative processes and enhanced decision-making capabilities (Bailey, 1993).

To overcome the problems in the above-mentioned projects, we have built this system for the systematic recording of the student’s personal records. Our project, the 'Student Record Management System' builds upon these principles by incorporating command line interface and database management techniques to provide a comprehensive solution tailored to the specific needs of educational institutions. The dynamic file handling in this system helps to retrieve the information of very large number of students in a very short period of time with a very less efforts. The system helps in easier access of the students’ records in the educational organization.

# CHAPTER 3

## METHODOLOGY

This Student Record Management System is designed in order to manage the students’ information in a command line interface by using C language. Different steps were followed in order to develop the entire system. Here’s the breakdown of each of the phases.

Figure 3.1: Methodology Diagram

### 3.1 Planning Phase

At the beginning of the development of this project, all the requirements were gathered and if any changes should have to be made, it has the feedback mechanism too (Scribd, 2024).

### 3.2 Requirement Analysis Phase

In this phase, we had conducted a seminar for teachers, students and other school staffs in order to understand their needs towards digital platform. Surveys and interviews were conducted with the students and teachers. Research was conducted to understand the existing student record management systems and analyze the area of improvement in the existing systems. In this phase, we had also studied about the feasibility of the system in the society after the completion of the project on different basis like social feasibility, economic feasibility, technical feasibility, time feasibility, etc. (Eludire, A. A, 2011)

### 3.3 System Design Phase

The overall architecture of the Student Record Management System was designed in this phase which includes designing the user interface to interact with the system. The interface that we have developed is Character User Interface (CUI) in which the user can interact with the system through text-based menus (Bharamagoudar, S. R., Geeta, R. B., & Totad, S. G. , June 2013).

Data Flow Diagram: DFD is a graphical representation of the flow of data within the system how the data enters into and leaves from the system, which process changes the data and where the data is stored. 

Figure 3.2: Level 0 DFD of SRMS

Level 0 DFD shows the proper overview of the system. In this DFD, the flow of data between the admin (user), system and database is shown. When the user requests any process in the system, the system requests to the database and responds back to the user on the basis of respond provided by the database. The user interacts with the system to manage student records. SRMS system handles adding, searching, modifying, and deleting student records. SRMS.txt file stores student information.

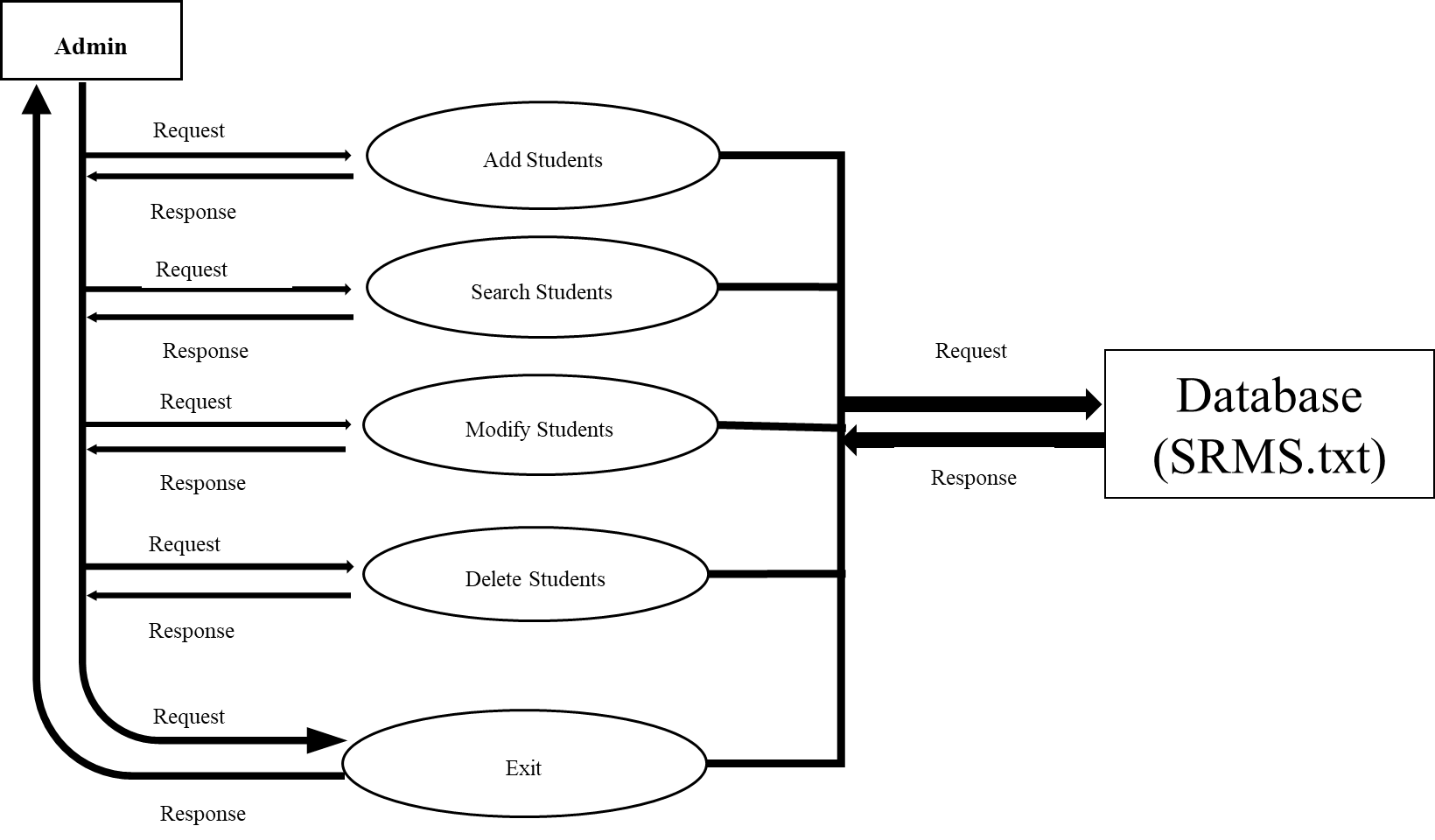


Figure 3.3: Level 1 DFD of SRMS

Level 1 DFD represents the flow of data in between the system and each of the processes in the system and recording them into the database. This DFD captures the flow of data within the Student Record Management System at a high level. The user interacts with the SRMS system to perform operations like adding, searching, modifying, and deleting student records. Then the system interacts with database to store and retrieve student’s data. The processes handle data inputs, modifications, deletions, and save the updated data back to the database (Edsembli, 2024).

Flow Chart: The flow chart below is the graphical representation of SRMS that explains the working method of each of the blocks. When the program is executed, the main menu presents options to the user for different operations where the user is asked to enter the choice. If the user chooses 1, firstly, it checks if the maximum number of students (100) is reached. If yes, it displays the maximum number reached message. If not, prompts user to enter details for a new student and displays a confirmation message after the details are added. If the user chooses 2, prompts user to enter a student ID and searches for the student in the records. If record is found, it displays the student's details and if not, displays an error message. If the user chooses 3, prompts user to enter a student ID and modify the records if the student is found with a confirmation message. If student is not found, displays an error message. If the user chooses 4, prompts user to enter a student ID and delete the records if found with a confirmation message. If student is not found, displays an error message. If the user chooses 5, the system saves all records to a file and exit from the system displaying an exit message. If the user chooses invalid choice, an error message is displayed and the system is returned to the main menu.



Figure 3.4: Flow Chart of SRMS

### 3.4 Development Phase

During the development phase of the Student Record Management System, we started by defining the core functionalities required for managing student records efficiently. Algorithms were developed to perform operations such as adding, updating, and deleting records. We used dev C++ platform with GCC compiler to write our code in Integrated Development Environment (IDE). Additionally, we somehow incorporated error handling mechanisms to manage invalid inputs and ensure the reliability of the system.

### 3.5 Testing Phase

In the testing phase, have checked it carefully to make sure it works well and is dependable. We’ve tested each part on its own, then put them together to see if they work smoothly. We’ve also tried out different situations, like when something unusual happens, to make sure the system can handle it. Plus, we have asked our friends to use the system to try it out and tell us what they think. Furthermore, if we find any problems, we'll fix them, so the system works great for everyone.

# CHAPTER 4

## RESULTS AND DISCUSSION

Student Record Management System (SRMS) results in an organized management of students’ record including their name, parents name, address, id and contact digitally. This system provides an effective platform to store students’ information. Unlike the traditional method, we don't have to search the student record manually. We can directly find the stored students’ information just by entering the students ID. This helps in time management and reduces workload of the staffs. The records can even be modified or updated which productively aids in reducing errors. Similarly, in case of no need of the students’ details in case of disenrollment of any of the students, the records can be deleted from the file. This system helps to keep effective and efficient track of students’ information in a well manner.

We have got the following outcomes from this system which surely provides the aids to the educational institutions especially for those institutions having very large no. of students:

1. Adding Students’ Details
2. Searching Students’ Details by Student ID
3. Modifying Student’s Details if Found
4. Deleting the existing Records of the students
5. Saving the data into the file
6. Exiting from the program

We evaluated its functionality and performance based on predefined criteria. This system successfully demonstrated capabilities in managing student records through the features like adding, searching, modifying, and deleting students’ data. We analyzed its efficiency on handling various operations.

Overall, this project met the expectations by delivering a reliable and practical solution for managing student records using C programming language. In the future, it is expected that this system will be developed even to record the academic information and progress of the students such as academic reports, attendance, results etc.

# CHAPTER 5

## CONCLUSION AND FUTURE WORKS

Conclusion:

In conclusion, the development of Student Record Management System (SRMS) in C language effectively demonstrates fundamental principles of data handling and user interaction within command-based applications. This project was designed with five key functionalities: adding records, searching for records, modifying records, deleting records and exiting from the system.

Overall, this project not only reinforces core programming concepts in C, such as file handling and data structures, but also emphasizes the importance of creating user-centric applications. Through repeated testing and improvements, each feature was adjusted to make managing student records easily and user-friendly. This mini project serves as a solid foundation for more complex systems and highlights the versatility of the C language in developing functional and efficient software solutions.

Future Works:

The system is developed with so much accuracy but this system is just a practical application of C programming. So, a lot of improvements can be done in the future to make this system more advanced. Firstly, we will develop this system into a user-friendly Graphical User Interface (GUI) so that it will be easier for the user to use this system. Next, we will integrate this system to record the academic information and progress of the students such as academic reports, attendance, results, etc. Continuous improvement and updates will also be done in this system. The system will also meet up with the emerging technologies in the recent future.

# REFERENCES

Acharya K. (2019). Student Information Management System Project. *SSRN 4765869*.

Bailey. (1993). Computerized Student Information Systems. *Journal of Educational Administration*.

Bharamagoudar, S. R., Geeta, R. B., & Totad, S. G. . (June 2013). Web Based Student Information Management System. *International Journal of Advanced Research in Computer and Communication Engineering, 2*(6), 2342-2348.

*Edsembli*. (2024, April 26). Retrieved from https://www.edsembli.com

Eludire, A. A. (2011). The design and implementation of student academic record management system. *Research Journal of Applied Sciences, Engineering and Technology, 3*(8), 707-712.

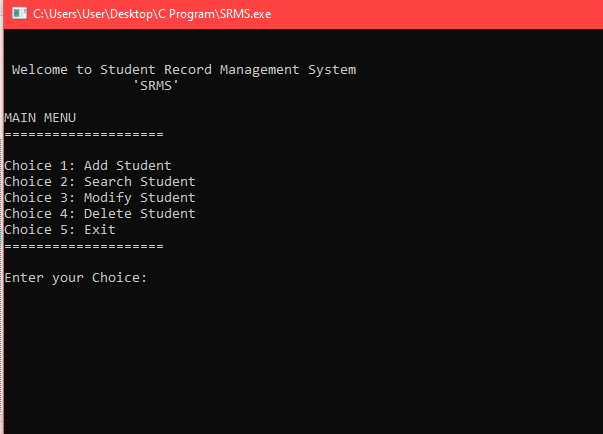
Radhika Bhanushali and et. al. (2022 July, April 22). Student Management System. *Ijraset Journal For Research in Applied Science and Engineering Technology*. Retrieved from https://www.ijraset.com

*Scribd*. (2024, April 24). Retrieved from https://www.scribd.com

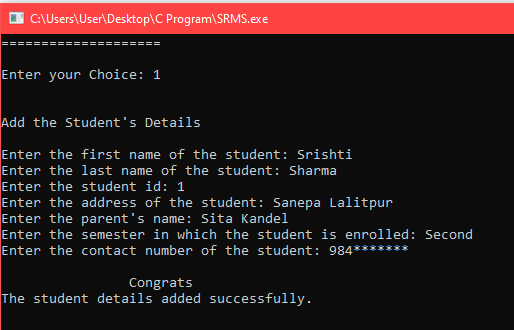
Tang, Y. F., & Zhang, Y. S. (2009, August). Design and implementation of college student information management system. *IEEE International Symposium on IT in Medicine & Education*, 1044-1048.

# APPENDICES

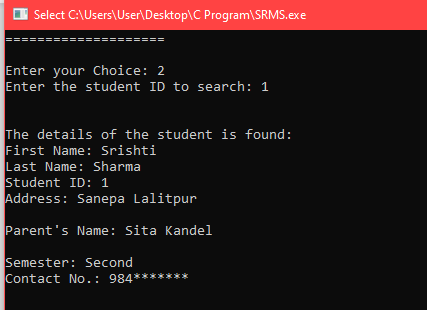
## 1 Main Menu



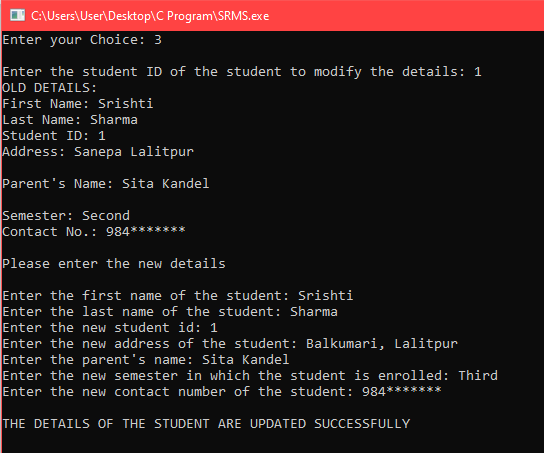
## 2 Add Students



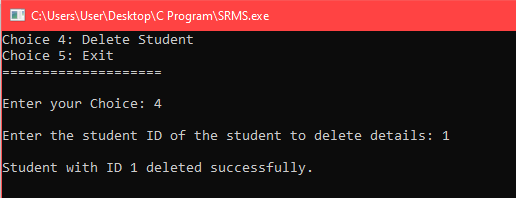
## 3 Search Students



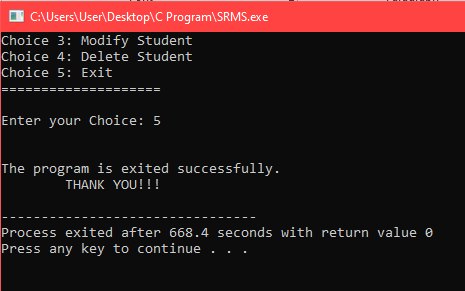
## 4 Modify Students



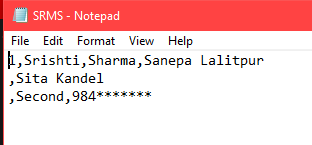
## 5 Delete Students



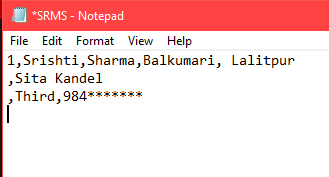
## 6 Exit



## 7 txt file: SRMS.txt before modifying



## 8 txt file: SRMS.txt after modifying



## 9 txt file: SRMS.txt after deleting

