



Student Record Management System

Final Project Presentation
5/12/2025

Submitted by:
Nistha Aryal
Mohammad Nemer

Submitted to:
Oke Onwuka

Introduction



For our final project in COMPSI-2 with Professor Oke, we created a Student Record Management System using C++. This project showcases important C++ concepts like file handling, pointers, sorting, modular design, and now even a Graphical User Interface (GUI) using Windows Forms. The addition of the GUI makes it easier for users to interact with the system through buttons, text boxes, and forms, enhancing its user-friendliness and aligning it with real-world application development.

We picked this project to apply classroom ideas in real life. By creating a Student Record Management System in C++, we honed skills like linked lists, file I/O, data management, and GUI development with Windows Forms.

Rather than just a coding task, we built something useful and easy to use. Adding a GUI made our system interactive and more realistic. This project enhanced our backend skills and introduced us to front-end design, making it both practical and thrilling.



Project Implementation

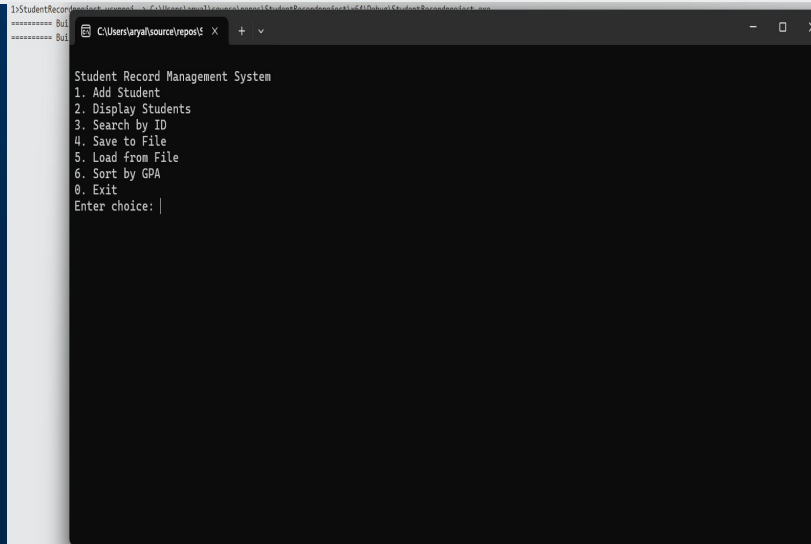
- Console and GUI interface for managing student records
- Add, display, search, and sort student information
- File I/O using text files to save and load records
- Linked list implemented using pointers and structures
- Sorting by GPA using bubble sort (descending order)
- Code modularized across multiple .cpp and .h files
- GUI developed using Windows Forms in Visual Studio

Demo

The demo showcases both console and GUI functionalities:

- Adding student records via form inputs (GUI and console)
- Displaying records and sorting them by GPA
- Searching for students by ID
- Saving and loading data from students.txt
- Interactive Windows Forms GUI with input fields and buttons
- Console menu still supports full functionality

Screenshots and live demo will show both console and GUI views of the system.

A screenshot of a Windows console application window titled "Student Record Management System". The window has a dark background and a light-colored title bar. The menu is displayed in a light color against the dark background. The menu options are: 1. Add Student, 2. Display Students, 3. Search by ID, 4. Save to File, 5. Load from File, 6. Sort by GPA, 0. Exit. Below the menu is a prompt "Enter choice: |". The window's address bar shows the file path "C:\Users\arya\source\repos\StudentRecordManagementSystem\StudentRecordManagementSystem.exe".

```
Student Record Management System
1. Add Student
2. Display Students
3. Search by ID
4. Save to File
5. Load from File
6. Sort by GPA
0. Exit
Enter choice: |
```

Console menu at startup lets users add, display, sort, search, and manage student records.

Future Work and Conclusion

- Enhance the current GUI with more advanced features (like validation or error prompts)
- Add update/delete functionality to student records
- Refactor code to use classes instead of structs for better OOP design
- Explore GUI alternatives like Qt or cross-platform libraries
-

Conclusion:

This project was an incredible chance to get hands-on with real-world programming using C++. By working on both console and GUI versions, we really got to grasp file handling, modular design, and user interaction. It not only honed our technical abilities but also provided us with a strong project to proudly display in our portfolios.