Final Report: Student Record System

Team Members:

Nistha Aryal

Mohammad Nemer

Course: COMPSCI-2

Instructor: Oke Onwuka

Date: 5/12/2025

Project Background and Inspiration

As Computer Science students, we were eager to embark on a project that would not just showcase our understanding of fundamental C++ concepts but would, in addition, prove to be beneficial in real life. We decided to work on a Student Record System, which is a common and handy tool in schools, and which would assist us in handling organized data effectively. This project enabled us to simulate a mini-database wherein we can add student records, arrange them, and retrieve them all inside a C++ console application.

Design and Implementation

We designed our program with several source files to offer modularity as well as readability.

• main.cpp – Manages the main menu, user input, and function calls.

•student.h – Declares the Student structure and function prototypes.

• student.cpp – Contains function definitions like adding, searching, sorting, and printing student records.

The major features that we implemented are:

• Adding new student records (ID, name, GPA)

• Showing all student records

Searching for a student by ID.

• Sorting student records by GPA using Bubble Sort

 • Saving and loading records using File I/O

All of the student data is held dynamically in vectors, which are read from or written to a .txt file to allow persistence.

Applied C++ Concepts

This project enabled us to research and implement the following C++ concepts:

- Structures and Pointers – For storing student data and passing by reference

- File Handling (Streams) – For reading and writing data to external files

- The sorting method employed in sorting the records by GPA was Bubble Sort.

- Vectors and Functions – To store data and maintain neat, modular code

Problems Encountered

One of the main problems for us was getting the sorting feature to function and implementing file read/write properly. At first, sorting did not function as anticipated because GPA was a float type, which means that proper comparison was needed. File handling was also problematic, such as reading from the wrong format or losing data. We conquered these problems through testing and correcting the mistakes several times.

Insights Gained

This assignment gave us hands-on experience with using in-class concepts in a real-world-style program. We gained familiarity with working with structures, file streams, and sorting algorithms. We also learned how to collaborate effectively, split tasks, and debug as a team.

Future Developments

- Among the improvements we plan to make in the future are:

- Including functionality to delete and update records

- Enhancing input validation to facilitate user error handling.

- Creating a graphical interface (GUI) to enhance its usability.

- Perhaps using linked lists or higher-level data processing structures

Conclusion:

Ultimately, this project provided us with an amazing experience of honing our C++ programming skills and our teamwork skills. We understood how to translate software development concepts into the real world, and we are proud both of the final product and of the learning process involved.