CS-499-10876-M01

Computer Science Capstone

Milestone 4 Enhancement 3 Databases

2024 C-3 (May-Jun)

Imed Charef

13 Jun 2024

#### **Milestone Four: Enhancement Three: Databases**

#### **Narrative**

### 1. ****Artifact Description:****

**What is the artifact?**

The artifact is a simple calculator application, originally implemented in Java and converted to C++. The primary function of this application is to perform basic arithmetic operations such as addition, subtraction, multiplication, and division.

**When was it created?**

This artifact was initially created as part of my coursework in CS499, where I enhanced and developed various aspects of the application to demonstrate my proficiency in software engineering and integration of database systems.

### 2. ****Justification for Inclusion in the ePortfolio:****

**Why did you select this item?**

I selected this calculator application for my ePortfolio because it showcases my ability to integrate database functionality into a C++ application effectively. The enhancement aligns with the databases category and demonstrates my proficiency in managing data and improving software solutions through the use of databases.

**What specific components of the artifact showcase your skills and abilities in software development?**

* **Database Integration**: I integrated an SQLite database into the application to store and retrieve calculation history. This involved creating a schema, setting up database connections, and implementing CRUD operations within the application.
* **Error Handling and Robustness**: The application now handles invalid inputs and database errors gracefully, ensuring a smooth user experience.
* **UI Interaction**: I enhanced the user interface to allow users to view their calculation history, thus providing an enriched user experience.

**How was the artifact improved?**

The artifact was significantly improved by:

* Adding a database component to store and retrieve calculation history.
* Modifying the application’s architecture to include a DatabaseManager class for handling database operations.
* Updating the user interface to support viewing past calculations, demonstrating the ability to integrate backend database operations with frontend functionality.

### 3. ****Meeting Course Objectives:****

**Did you meet the course objectives you planned to meet with this enhancement in Module One?**

Yes, I believe this enhancement meets the course objectives by demonstrating the ability to design and implement database solutions within a software project. It also showcases proficiency in integrating third-party libraries with a C++ application, which was one of the goals outlined in Module One.

**Do you have any updates to your outcome-coverage plans?**

No significant updates are needed for the outcome-coverage plans. The integration of the database effectively covers the expected outcomes related to database management and software development practices.

### 4. ****Reflection on the Process of Enhancing and Modifying the Artifact:****

**What did you learn as you were creating it and improving it?**

Through the process of enhancing this calculator application, I learned several key aspects of database integration with C++:

* **SQLite Integration**: I gained hands-on experience with setting up and using SQLite within a C++ application, which included compiling the SQLite library, creating tables, and performing SQL operations.
* **Architectural Considerations**: I learned how to design a software architecture that effectively separates concerns, ensuring that the UI, core logic, and database interactions are modular and maintainable.
* **Error Handling**: I deepened my understanding of handling errors and exceptions, particularly in scenarios where database operations could fail or when handling user input.

**What challenges did you face?**

* **Library Integration**: One of the challenges was correctly setting up the SQLite library within the C++ project. This required adjusting the CMake configuration and ensuring that the project could locate and link against the SQLite libraries.
* **Data Consistency**: Ensuring that calculations were correctly stored and retrieved from the database required careful consideration of data types and SQL commands.
* **UI Updates**: Modifying the user interface to incorporate new functionality while maintaining a clean and user-friendly design was challenging but ultimately rewarding.