Project 7

School Grading System
Pierce Angeloff & Rahul Shenoy

[Repository]

Final State of System

We implemented every feature in our Project 5 UML, except the Grade Analysis, and Database upload and download feature. Grade analysis was a tool that in itself could have been our whole project, so we decided to drop the feature, as it was costing us a lot of time and was only a part of the project. DB upload and download was something Rahul spent days on and kept getting errors with the default database continuing to be open, so we were forced to drop the feature due to time constraints. Assignment was made into a class and multiple functions were added to the class (shown in UML).

Final UML Diagram & Comparison

Link (Google Drive)

Overall the finalized UML is a lot different than our original plan and contains so much more. This certainly taught us the value of planning ahead to a greater degree than before as we realized over time our project needed a lot more thorough planning to be successful. A lot of this added complexity was also a result of the Qt framework and how it organizes classes and objects. Our original plan was still the core of our project, built around the Teacher, Student, Course, and Assignment classes. The use of Qt to provide our interface would then effectively add a new class for every window to be displayed, and even classes specifically used for buttons on the screen. Our final project also used a DataTest class that was responsible for test cases and data storage, and while we planned after Project 6 to include a DataManager class for persistent data, this was scrapped due to limited knowledge and struggles with using SQL-lite and csv files.

Third-Party code vs. Original code Statement

Using QT, we heavily consulted QT API and Documentation from https://doc.qt.io/

Outside of this, we consulted these sources on Stack Overflow for a few features such as the Table view for student grades when viewed by a Teacher:
https://stackoverflow.com/questions/37431729/create-a-table-in-qt-and-fill-it-by-the-user

https://stackoverflow.com/questions/12841269/set-qlabel-width
https://stackoverflow.com/questions/57460143/qframe-border-is-not-displayed

Statement on the OOAD process for your overall Semester Project

When we created a factory pattern for the assignments, we found it to have a positive effect on the analysis and design of the project

Coupling was a hard concept to get working, but once we did, we found it to have a positive effect on the analysis and design of the project.

Encapsulation was a hard concept to implement, because the Qt framework and its windows and widgets like to belong to its own classes, and since we had a number of different windows responsible for different things, our code became cluttered. We found encapsulation as a principle to have a positive effect on the project. But the way we implemented it had a negative effect (cluttering).