CSCI 2300, Spring 2019

Team Project

Core Logic Implementation with Unit Tests [100 points]

Overview

For this part of the team project you need to implement and test the core logic of your application. The core logic is contained in the classes that do not have any Java Swing API components. In other words, these are the ‘model’ classes that represent and change the state of the application. If you look at your sequence diagrams, these will be the classes on the right side.

Details

Using your class diagrams and sequence diagrams, determine which classes contain the core logic. If you are unsure, please reach out to me and I’ll help you with this step.

Implement the functions of your core classes that are called by other classes. Note, that to do this, you may need to implement some private functions or additional classes that are used by the core classes. Provide unit tests for your code: each method of each class must have at least one unit test (feel free to add more than one unit test per function).

Submit

Submit your code for the core classes and unit tests to your project team git repo. Note, all code submitted to the master branch must compile. I will compile it with: javac -cp $CLASSPATH *.java

Grading

Your grade will be based on the following criteria:

[50 points] Core logic implementation:

- Is all the necessary code present and working correctly (30 points)?
- Does your solution avoid code duplication (5 points)?
- Does your code follow Single Responsibility Principle, Open Closed Principle, and Liskov Substitution Principle (10 points)?
- Does the code follow CSCI 2300 coding standards (5 points)?

[30 points] Unit tests:

- Did you provide at least one unit test per function?

[20 points] Peer evaluation results: I will distribute a peer-evaluation form when this phase of the project is due. A portion of your grade will be determined by how well you worked with the rest of the team and your contribution to the project (as evaluated by your teammates).