Introduction

One of the primary objectives of this course is to teach you some best practices for building large software systems. In order to do this you must work as a team to complete a project in one semester. This semester you will be building a Game Suite that allows the user to select from at least three (3) games of different types (logic, board, card, etc). You must utilize an Object-Oriented language (i.e Java, C#, C++, etc) to implement your application and you may integrate with an approved framework; the list of approved frameworks is here: <insert link to page>

Requirements

1. You must have a graphical user interface for the user to select a game to play.
   (As a user I want to select a game to play using a graphical user interface so that I can improve my game playing skills.)
2. You must implement three games, including game logic, from at least two of the following categories:
   a. Card Games (solitaire or multiplayer)
   b. Board Game (checkers, battleship, etc)
   c. Logic Games (crossword, wordle, brain teasers)
3. You must provide the user with a way to end the game and return to the game selection screen.
4. You must provide the user with a way to exit the Game Suite application from within the GUI of the application.
5. You must include help text for each game including instructions/rules for game play.
6. You must apply good object-oriented design principles. This includes, at a minimum, encapsulating functionality and leveraging hierarchies to reuse common functionality instead of copy and pasting multiple places.
7. You **do not** have to implement networking for multiplayer games.
8. You must utilize iterative development to analyze, design, and implement your application.
9. You may use a game **logic** engine for one of your games. This is for the logic and rules of game play not the game GUI itself.

Sprint Deliverables

Sprint .5

- Team formation and organization
- Version control setup (i.e. Git)

Sprint 1

- Requirements analysis
- Use Case Diagram
- GUI main screen

Sprint 2

- Complete implementation of Game 1
- Software Architecture overview
- Your pick of one of the other object-oriented analysis diagrams

Sprint 3

- Complete implementation of Game 2
- Your pick of one of the other object-oriented design diagrams

Sprint 4

- Complete implementation of Game 3
- Rationale for (good) design decisions

Sprint 5

- Final Demo Video submission

Extra Credit

TBD