

## **Project Part 3: Refactoring**

### **Project Name**

Santa Horror RPG

### **Team Members**

- Kelsey Dowd
- Bradley Arnot
- Nicholas Pfeufer
- Jeffrey Lipnick

### **Design Changes**

#### **State Design Pattern**

We implemented the State Design Pattern which allows for a better structure when switching between different gameplay states. Originally we thought the Zone class would control this behavior, but after learning the State Design Pattern we saw that it was a better implementation. Zone has been repurposed to function as the gameplay level. When refactoring the class diagram we created a GameState abstract super class. The states, MainMenu, SaveMenu, LoadMenu, and Zone, all inherit from the GameState class and implement the update and render methods. We got rid of the DisplayManager class and divided up the functionality among MainMenu, SaveMenu, and LoadMenu. The GameModel class stores all the game data such as all the GameObj's, which was previously stored in Zone. The current GameState and GameModel are stored in the Game class and the the GameModel is passed into each state through the render and update methods. The State portion of the refactoring has been highlighted in green in the class diagram.

#### **Observer Design Pattern**

We implemented the Observer Design Pattern to handle the input from the player. Originally we had the Zone class listening for and handling key input. After learning the Observer Design Pattern we realized it was a much better way to handle user input. Now we are using a Publish-Subscribe model that more effectively addresses the problem of interfacing with user input. To implement the Observer Design Pattern, we have 2 abstract classes, Subject and Observer. The Keyboard and Mouse event handler classes inherit from Subject and keep track of the Observers they are notifying. The Game class inherits from Observer and implements the update method which is run when either Mouse or Keyboard notify it of user input. The changes in the class diagram are indicated with red.

Diagram can be viewed in more detail on GitHub at “Diagrams/ClassDiagram.[png/svg]”

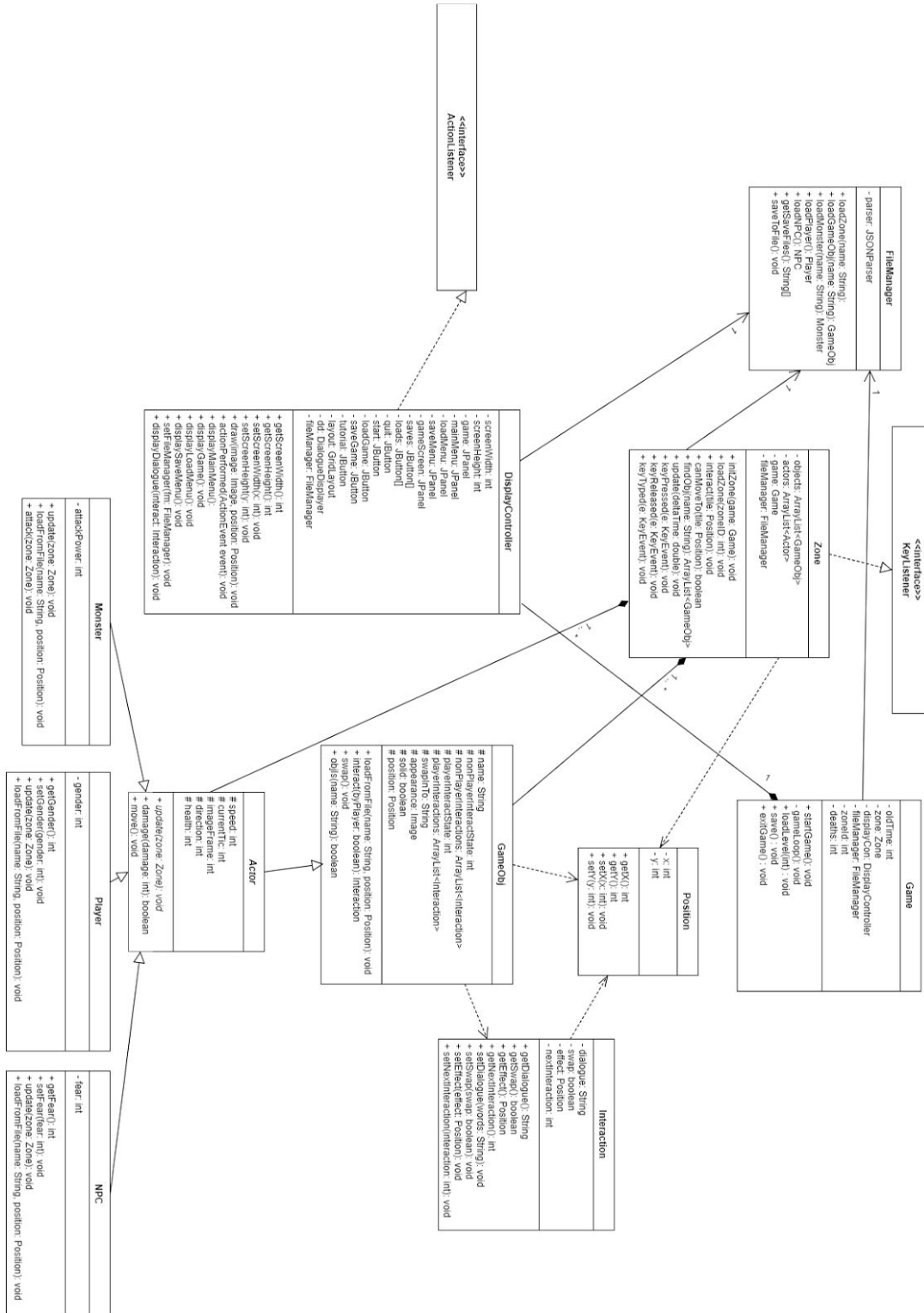


Diagram can be viewed in more detail on GitHub at “[Diagrams/ClassDiagramNew.\[png/svg\]](#)”

