Brenden Judson

Chips Challenge Description

10/17/18

My design starts with a ChipController class that extends Java’s Application class. This class is composed of the other components of this project and acts as the main class or controller of this project. It has an instances of a Level, Map and a Chip class. The chip class contains information on the character the user controls. It has functions that allow you gather information on it and control its attributes like moving around its location. This class is an extension of the Observable class so that other classes (bugs) can track its location. It also has a refence to the Map class because it needs information on the map like where the walls are. The map class contains the current state of the board. The current state of the board (and also map) is dependent on the current level. Each level needs to implement its own layout for the map, therefore the levels are Implemented with the strategy design pattern. Finally, there is a bug class. This class is the observer of the Chip class. It follows his movements at a slower pace and therefore needs to be notified as chip’s location changes.

If we had further time one thing I would like to fix is that I think there is a good amount of redundancy between the map and level classes. Going forward, I would like to simplify these two classes and the information they contain. For example, both contain grids of the terrain. It could be simplified only have one instance of the grid.

The biggest difference between the final UML diagram and the original one is that I did not have the full scope of the project outlined at the first stage. Specifically, the level strategy was not included in the first stage. I think the final UML diagram does a much better job of outlining the project.