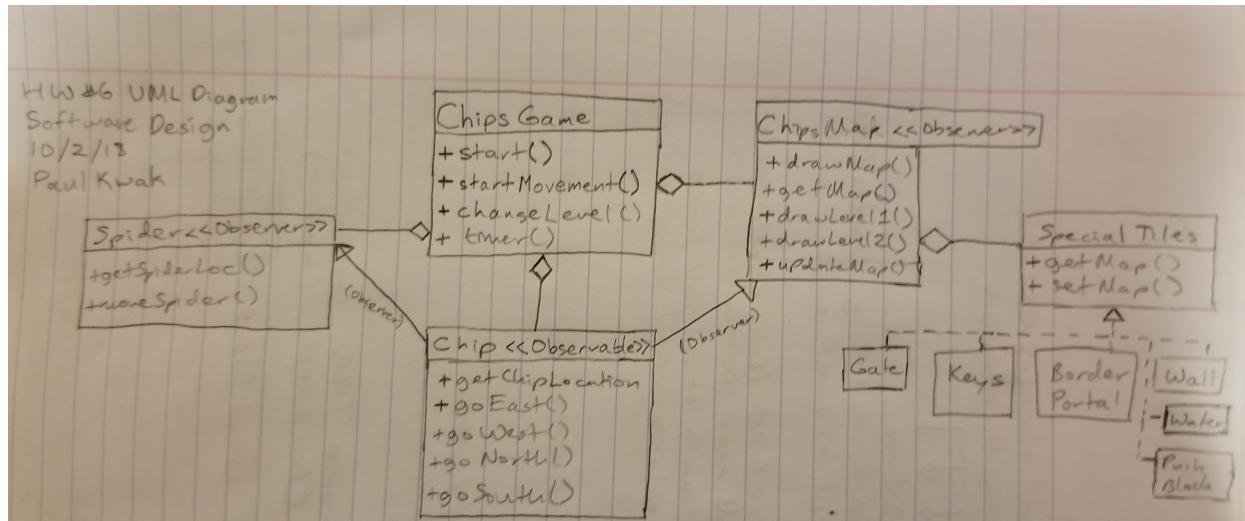
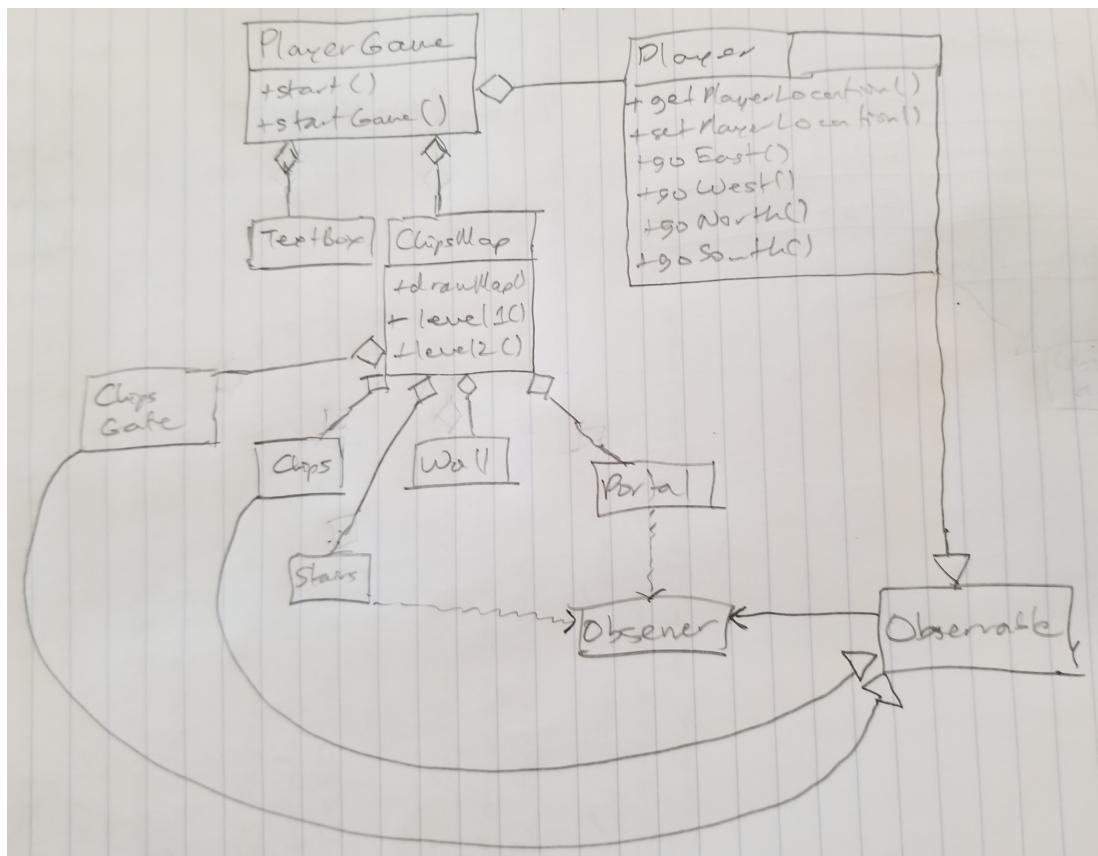


Paul Kwak
 Software Design
 Chips Challenge
 10/18/18

Old UML



New UML



The differences between my old and new uml diagram is that I decided not to go with enemy spiders, keys, but instead went with stairs. I also decided to keep a lot of the strategies I had initially. This leads into the next part of this deliverable:

Patterns Used:

Pattern Name: Singleton

Class Name	Role in Pattern
ChipsMap	Singleton

Purpose: This is to ensure that only one version of the map will be used at all times. Since there will be multiple levels that need to be instantiated, it's important that there's only one map that they're being instantiated upon.

Pattern Name: Observer

Class Name	Role in Pattern
ChipsGate	Observable
Chips	Observable
Portal	Observer
Stairs	Observer
Player	Observable

Purpose: This allows a lot of the functionality in terms of just running the game, as the player's sprite will need to interact with a lot of objects, as well as objects needing to interact with other objects.

Pattern Name: Strategy

Class Name	Role in Pattern
ChipsMap	Parent
ChipsGate	Child
Chips	Child
Portal	Child
Stairs	Child

Purpose: This allows a lot of special tiles to be created from just using the board class as well.

The final design followed a lot of the strategy patterns that I had initially in my intermediate design. However, in terms of implementation, things were tricky to sort out at first, since some special tiles could have changed their status from being immovable to replaceable. Also, I wish I could have implemented the strategy pattern a little differently, since ChipsMap contains a lot of the special tiles which may make it more of a god class. I also would have liked to add more

features and work on creating an actual level. Also, I would work on making a level factory or something of the sort.