Game name: Dragon and Dungeons

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Description: Turn-based role playing game. Your goal is to survive, defeat the dragon and enter it’s treasure room.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

User Level Description: Player is able to equip/unequip armor and weapons, sell/buy items from shop, pickup/drop items, fight enemies, use potions,go around rooms, unlock doors and chest and levelup.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Implementation Description: We set up a google doc with general goals in mind and we slowly checked off what we did. Discussed it between each other when we get the chance.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Special Features: Turn-based combat with movement priority.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Patterns:

Singleton: Created an instance of GameWorld in GameWorld.cs

Command: Command.cs takes the name and second and use that to parse and see if a command is available. The commands we made usually have the name of the command followed by Command.cs. Example is GoCommand.cs or BackCommand.cs.

Template Method: We created interfaces in GameInterffaces.cs to create doors, delegate rooms, chest, items and key items.

Adapter: We created an entity interface in GameInterfaces.cs and use that as the base of our items and key items interfaces.

Notification: GameWorld.cs and Room.cs. Checks if you enter a certain room or say a certain word.

Delegates: Room.cs. Check if you enter a room to have the shop get more items. Check if you say the correct word in a trap room. If it is wrong, you take damage.

Facade: In GameInterfaces.cs we used void Open() and void Close().

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Bugs: TrapRooms after saying the correct word will damage you if you say the wrong word after you are allowed to leave. When you enter a certain room, the shop restocks, however if you enter it repeatedly, it just adds more items instead of deleting the room after you activate it once. RestartCommand.cs does not work quite as well. Restart just remove items permanently in the session and enemies that have been defeated do not come back.