Dungeon Crawler Design Proposal

# Project Description

This game will consist of a player that will be controlled with WASD and points at the mouse at all times and fires its gun when the mouse is clicked. The player will also be able to dodge in the direction that they are moving using shift or if they are not moving it will be toward the mouse. There will be different type of enemies with different attack patterns. Some will have pathing which will track the player, some will aim at the player, some will react to specific player actions, etc. There will be walls which will stop movement and bullets. There will be different items that the player can have: damage reducing, health increasing, damage increasing, different types of guns that shoot differently and have different damage stats. There will be saving.

# Competitive Analysis

This game is similar to the game [Enter the Gungeon](https://store.steampowered.com/app/311690/Enter_the_Gungeon/). It will have similar mechanics: dodging, shooting, getting better weapons, semi-random loot drops, a store. It will also have similar objectives: defeating bosses to get to chests (maybe just one boss for my game). I will also not add features such as limited ammo.

[Hyper Light Drifter](https://store.steampowered.com/app/257850/Hyper_Light_Drifter/) is also similar. While the gameplay is fairly different aside from the use of dashing, I hope to partially emulate the enemy design specifically the bosses.

Unlike both these games I will have a level editor. I will also have a each “room” take up one screen and going out a door will then load another “room”.

# Structural Plan

I have a class folder where I will define classes. I will have a “player.py”. The player class will have health, an angle, a position, equipped gun, equipped items, a constant width and height. It will have a bag which holds obtained items, obtained gun, and money. Player class will also have methods: shoot (calls the gun’s shoot function), move, dash, draw, etc.

I will have “items.py”. I will have a item class which holds what the item modifies, health, damage, etc. I will have a static function which returns a list of all the items.

I will have “guns.py”. It will have a static function which returns a list of all the guns. Each gun will be its own subclass of the gun class excluding the default gun. Each will have a shoot function which does something different for each gun.

I will have “enemy.py” which will define an enemy class which has attributes like health, hitbox (this will be a tuple of coordinates assuming the top left of the image is at 0,0), The basic enemy will move to be a random radius away from the player with nothing obstructing their view. Some enemies may just stand still but have different attack based on where the player is a what they are doing.

# Algorithmic Plan

The most complex algorithms will be path finding and hit detection.

For path finding the first problem will be is there a straight shot to the player. I will probably use code similar to hit detection to see if a line between the enemy and the player is obstructed.

# Timeline Plan

By 8/6: I want to have the character movement fully implement along with the basic enemy type (hopefully with path finding too).

By 8/7: I want to have the first Room completely done

By 8/8: I want to have the first level done will have a couple rooms with chests with random drops. Probably 2 or 3 enemies including the boss will also have random drops.

By 8/9: add saving

By 8/10: Depending on time constraints add money, store, and maybe level editor.

Maybe add usable items?

# Version Control Plan

Github

# Changes

I don’t think I will have time to add items, but beside that not much is different than my plan.

# Changes 2

I will not be adding items or additional guns. You can save and load levels from the level editor. I didn’t go exactly by my timeline and ended up adding the store after the first room was done, but besides that I stuck pretty faithfully to it (also did not add random drops).