Introduction:  
Norathian Arcadia is a captivating 2D RPG game crafted in Python, offering players the exciting opportunity to embark on a quest to explore numerous dungeons in search of rare items and valuable loot. Developed using the Pygame library, the game promises a rich experience of adventure, combat, and character growth. Throughout this essay, I will discuss the achievements made during development, the envisioned goals that could not be accomplished by the deadline, and the valuable insights gained from the game development process.

Concept:

The game was envisioned to offer players the exciting choice of becoming a Warrior, Mage, or Rogue, akin to the iconic character selection in Diablo 1. Although animations were meticulously programmed for each class, only the combat mechanics for the Warrior were fully developed within the given timeframe. The grand plan was to create eight captivating 2D dungeons, each housing various valuable items that, unbeknownst to the player, would prove essential to conquer another dungeon of their choosing. Throughout the journey, players would venture through these dungeons, amassing an arsenal of gear, some of which were incredibly rare, necessitating multiple clears for acquisition. The ultimate challenge was to connect the dots, allowing players to deduce the importance of specific items for their quest. Regrettably, the game's current state only allows players to begin in the default dungeon, engage in combat with a few adversaries, acquire loot, and equip items. Despite not achieving the full vision, the development process provided valuable insights and paved the way for potential future expansions and refinements.  
Here is an over view of the class behaviors with a class diagram  
A screenshot of a computer

Description automatically generated

Player Character: As mentioned above the character would have had three options but we just had one.   
I created the character spritesheets, and enemy spritesheet from <https://sanderfrenken.github.io/Universal-LPC-Spritesheet-Character-Generator/#?body=Body_color_light&head=Human_male_light>. The spritesheets contain walking, attacking , casting, and shooting from various directions. The player may press q for melee attack, e for stab, r for shoot and f for cast. Q however, is the only attack that interacts with anything but the others will trigger the animations.   
An example of the default character before any armor is equipped:  
A cartoon of a person holding a sword

Description automatically generated  
The character after full armor.   
A pixelated video game character

Description automatically generated  
Note any combination of those items requires its’ own spritesheet. For example, the top spritesheet is ‘defaultChar.png’. The bottom one is ‘HCALS.png’. This stands for head, chest, arms, legs, longsword. Every combination of loot had to be added which is 5c1 + 5c2+ 5c3+ 5c4+ 5c5 combos. This took a while but I am proud I figured it out. The character can kill an enemy, walk over the item, once it disappears that means you picked it up. You then can press I to open the inventory and click on the item. That will equip it. Note sometimes the alignment is a bit off and you must click around slightly.

Exploration and Movement:   
Player navigates using WASD to move around. The 2d dungeons are just PNG images where I used pygame.rect on the locations I wanted walls to build the collision to make the image appear to be interactive. For instance I had my character print out location statements according to where he was on the grid. I then wrote those downs and went in and built the walls based on those areas. They could be toggled on and off to trouble shoot and test them. The enemies also had to be programmed to collide with the environment.

Combat System:   
This involved the player triggering the agro radius of enemies either in a patrol state or not patrol state. The player can enter in and out of it dropping agro. While the player is in the agro radius the enemy will continue to attack the player with an attack delay of 2 seconds. The player was intended to not kill the enemies in one hit but I never got to working the defense and hit points for the enemies so the character “one-shot” enemies. The player however has 100Hps and each attack from an enemy takes 10 hit points from the player. The enemy and player but have hit boxes that cannot collide with each other however, sometimes they do get sticky on each other and I have to figure that out still. Once the player dies he drops loot. Again I did not intend on characters dropping loot every single time. It is still set to one random piece of loot between sword, chest, arms, legs , helmet and every single kill drops one piece. This would have been changed eventually.

Character Development:  
Currently the state of the game is the character just changes his image and there is not much more development than that. The armor of course should have added defense, maybe health, and the weapon should have increased attack range and hit box range. Of course since the enemies are already being one shot that would need to be changed for any weapons to even feel any impact. Bigger the weapons hit box the longer I was going to allow the attack delay.  
  
Below is a Player State Diagram, Enemy State Diagram, and Game State Diagram.  
A screenshot of a computer

Description automatically generated  
A screenshot of a computer

Description automatically generated A screenshot of a diagram

Description automatically generated