1. Niko Bazos, Team 3 Musketeers
2. My roles for this assignment was Project Lead/Game View.
3. I worked on the Game1 and Player classes.
   1. In the Game1 class I worked on:
      1. expanding on every game state to give it a stylized look and use.
      2. implementing a custom font and making it work in-game.
      3. player direction so now if the player presses the left or right arrow key the character faces that direction.
      4. enemy direction based on where the player is on screen so that now the enemies will always face you.
      5. implementing in-game UI such as health, ammo, and wave number.
      6. implementing animation and I made a sprite sheet, however, I could not get it to work as intended.
      7. enemy movement so that half of the enemies move in a different direction from the others.
      8. Dez and I fixed the bullets so that they fly in the direction the character is facing.
   2. In the Player class I worked on:
      1. adding a property for the amount of ammo
      2. Adding a UseBullet() method that decreases the amount of ammo left
      3. Adding a GainBullet() method that increases the amount of ammo based on an enemy death
4. Current major bugs/issues that need to be fixed are the amount of health lost upon collision with enemy objects, the drawing of the enemy bullets, and the animation of the character assets. Some bugs that were introduced when I integrated my personal code with the overall project was the game crashing due to the True Type Font I wanted to use not being converted in the proper format (now fixed), as well as when I attempted to implement my spritesheet I was unable to effectively cut up and play each frame. My plan for tracking down and fixing these bugs is to first comment out certain areas of code and observe its effect on the game. After analyzing these effects I will search for alternative algorithms or alter the current one with trial and error in order to correct the issue. I will utilize the debugger, however, for our particular issues I think the trial and error with commenting will prove more effective in understanding the issues at hand.
5. The design, architecture, look, or timeline has not changed since Milestone 2.

Name: Alex Pierce

Team Name: Three Musketeers

Role: Architect

This milestone I finished work on the map tool. I connected the map tool and the main game so that when you created a map with the map tool the file is immediately ready to run in the main game. Originally I tried to have the file serialize the scenery method but when I realized what serializing actually does I decided to have information save to a binary file and have the main game decode the information that it is sent.

The only bug as far as the map tool is concerned is the lack of a background texture. We can manually set this in the next milestone.

I messed around a lot with the scenery and scenery converter classes so that they would transfer the relevant information between the map tool and the main game.

Alex Martinelli

Musketeers

Role - Miscellaneous additions and fixes

This milestone I worked on enemy shooting, as well as finishing the map tool. I fixed the X and Y position picker for the map tool to allow for the user to use the mouse. A strange problem arose when doing this. The form being used also got the X and Y when clicked, but the map tool had no valid X and Y positions. This caused the map tool to crash. To fix this, I wrote an if statement restricting the click event to be used only when the mouse’s X and Y position were between 0 and the viewport’s width and height. I also helped Alex P. in fixing the save function of the map tool. Finally, I worked on the enemy shooting. I created a method for the enemy class that handled the shooting and was called in the update method. While I can confirm through testing that the shoot method works, I’m having problems with actually drawing the bullets. Either the game crashes whenever I try to draw the bullets in, or the bullets don’t draw at all. I’m still working on this, and will finish it for milestone 4.

Dezmon Gilbert – Team 3 Musketeers

Game Development and Algorithmic Problem Solving II

Professor Kevin Bierre

April 24, 2016

For this milestone my role was primarily **Gameplay.** In the Game1 class, I created the following methods: StartGame, NextWave, and Death. The StartGame method resets the score and wave count to zero and calls the NextWave method. The NextWave method increments the wave count, clears the list of enemies, and creates enemies with random locations to be added to the list using a for loop. The amount of enemies to make is decided by an algorithm that is 2 times the wave count plus 3. The Death method goes through the list of enemies and determines which enemies have died and removes them from the list. In the Enemy class I created a method, EnemyDeath, which will set enemy IsActive property to false when the enemy health is 0. It also returns a bool to which is to help with if statements in Game1. I also made a new Draw method in Enemy that only draws the enemy while the IsActive property is true.

I had a lot of troubles working with the Update method. Mainly my issue was the enemy dying too quickly, the score incrementing forever, and figuring out the best way to call the NextWave method(i.e. using a variable to keep track of amount killed enemies and when that is equal to the list Count property then call the NextWave method. The major issue was that I tried to do everything in the Update method of Game1. I fixed these issues by making methods outside of the Update method that would handle this for me. This was especially helpful in getting to the next wave since it was difficult to remove an enemy from the list without the game crashing in the Update method.

There are currently no bugs on the parts that I worked on. The only issues I want to work on are putting more enemy types in and working on the “wants”.

Not much has changed for my part from milestone 2 since we didn't change how the input or game states are implemented.