Sprints

Sprint 1
10/20 - 10/23

Alex - Active tower mechanics
Matt - Cleanup the existing code base, wave spawner
Nick - Basics of a tower
David - Work on the enemies: health, speed, spawn rate, positioning.
Thomas - Lighting effects

Sprint Retroactive

Alex - Active tower moves, controls set up to shoot a projectile
Matt - Cleaned up a little bit, UI elements added
Nick - Started the active tower class
David - started projectiles
Thomas - Rotating light camera effect

We reached some of our goals, but we did not reach the overall progress that we had planned. Our game still looks about the same as last week with just a couple of extra things. For the next sprint we will have to improve our overall communication for our project and work on smaller projects if we are dependent on others' work. Progress on towers is currently held up by the lack of a projectile class, but once this is finished we should make some good progress.

Sprint 2
10/24 - 10/28

Alex - work on camera at multiple positions
Matt - enemy class, wavespawnerclass, and getting a basic user interface to work
Nick - adding and getting rid of targets
David - Finish up projectiles
Thomas - Placing towers

Sprint Retroactive

Alex - Set up camera at 4 positions, worked on projectile stuff
Matt - Made a few fixes, worked on AI tower
Nick - Worked on AI tower, including searching for enemies
David - Worked on projectiles
Thomas - Worked on placing towers

Summary - Overall most of our individual goals were met. We added some nice smaller features that will definitely be important in the long-run for our game. One area we missed on was projectiles, which is very important to our game, which we would like to get included in our project 3 submission. We did a better job of meeting our goals this sprint overall, but we could still improve on our group communication. In the last sprint we hope to mainly clean up what we have and take care of housekeeping items.

Sprint 3
10/29-11/1
Alex - cleanup shooting mechanic of player tower work on housekeeping items
Matt - Work on passive towers, tinker with UI
Nick - add different types of towers
David - integrating his projectile class
Thomas - Add support for stat modifiers

Sprint Retroactive

We met most of our goals for this sprint, since we were mainly cleaning things up for presentation, as well as adding in a couple of things behind the scenes that will help us in the future. Overall we worked well as a group here to knock out some of the documentation and diagrams.

Sprint 4
11/2 - 11/16
Matt - Add UI elements to display score, health, etc. Delete grass blocks when tower placed
Alex - Player tower projectile physics, implement rotating cannon for player tower
Thomas - Cleanup wave-spawning mechanics to increase difficulty with progression. Lighting improvements
Nick - Cleanup wave-spawning mechanics to increase difficulty with progression
David - add “playerwallet” to implement money system

Sprint Retroactive
We met almost all of our goals for this sprint. At this stage we have almost all the elements of a “minimum viable product”, which once finished we can expand upon.

Sprint 5
11/17-11/30
Matt - UI improvements, toggling UI for tower purchase, Buff towers, display wave in UI
Alex - Enemies destroyed when their health reaches 0, player tower projectile collision, display money from enemy death
Nick - Laser Towers
Thomas - UI improvements

David - Documentation

Sprint Retroactive

We met most of our goals for this sprint. Overall progress was stalled by our busy schedules and Thanksgiving break, but we made good improvements despite this. We have many elements in place of a full game, but it is not yet working as a basic functioning game.

Sprint 6
11/31 - 12/08
Thomas - add start screen
Alex - Handle purchase of towers
Matt - tower placement bug fixes, UI of prices for towers
Nick - start wave tower highlight, implementation for buff towers.
David - documentation

Sprint Retroactive

We met our goals successfully during this sprint. At this stage we have a fully functioning game. Overall the game needs to be more balanced, which will require user testing (by ourselves) and others. From this point on we are adding to the game, rather than getting it into a functional state