**P5 Tower Defense Game**

**6/03: DAY 1: Implementing the Tower System 0.0-0.2**

* Designed tower class displays and a temporary green background.
* Planned the enemy track by sketching it on paper for reference.
* Developed the tower placement system after researching and learning the required logic.
* Each tower can be placed on designated properties, which then become unavailable.
* Future plan to implement tower selling and replacement mechanics.

On the first day of the Capstone Project, I started by drafting all the blueprints visually and creating the tower class display with a simple green background. I carefully planned the enemy track by drawing it on paper to guide development and allow for easy adjustments. I then focused on implementing the tower placement system, which proved more challenging than expected and required online research. Players can place towers on predefined properties, which get replaced by the tower and become unusable afterward. I intend to add functionality for selling and swapping towers in future sessions.

**6/04: DAY 2: Implementing the Gold System 0.3**

* Set initial gold amount to 350 to balance early waves.
* Introduced variable tower costs depending on type (e.g., archers cheaper than bomb shooters).
* Added a build menu appearing upon clicking tower properties, showing tower options with costs.
* Implemented color-coded affordability indicators for tower purchase options.
* Included hotkeys beside each tower in the build menu for quick selection.

During this session, I worked on the gold system that manages resources for tower placement and upgrades. I started with a balance in mind by giving the player 300 gold, enough for early wave defense without overwhelming firepower. Each tower has a cost varying by type, which led to adding a build menu accessible by clicking tower properties. The menu displays three tower types with their costs and hotkeys. To improve usability, I coded color cues: towers that the player can afford show their cost in gold, while those out of budget appear in red.

**6/05-6: DAY 3: The Track and Tower Targeting 0.7**

* Created a simple linear track running right to left with bends.
* Developed enemy pathing using an array of coordinates for sequential movement.
* Added health bars above enemies to visually indicate damage taken.
* Designed towers with unique ranges, damage, and attack rates for strategic variety.
* Simplified targeting by having towers attack the enemy with the lowest x-coordinate within range.

This day focused on constructing the enemy path and refining tower targeting mechanics. I designed a linear track starting from the right side of the screen to the left with a few bends. Enemy movement is controlled by passing an array of target points to each enemy instance, which they follow sequentially. Each enemy shows a health bar for player feedback. Towers were given distinct stats (range, damage, and fire rate) to encourage strategic play. The linear track enabled a simple targeting system where towers detect enemies within range using distance calculations and attack the enemy closest to the exit (lowest x-value).

**6/09-10: DAY 4: Range Visualization and Enemy Variety 0.8**

* Added a low-opacity green circle around selected towers to visualize attack range.
* Implemented a parent enemy class to manage shared properties for all enemy types.
* Created diverse enemy subclasses with different speeds, health, and rewards.
* Planned an armor system to reduce damage from physical attacks and promote tower variety.
* Aimed to encourage strategic balance by punishing overuse of specific tower types.

While testing and balancing the game, I realized it was difficult to judge tower ranges without a visual. To address this, I added a translucent green circle around selected towers showing their effective attack radius. I also set up a parent enemy class encapsulating shared attributes like health, speed, and reward, from which different enemy types inherit. This allowed me to create diverse enemies (some fast and weak, others slow and tough) to challenge players in varied ways. I implemented an armor mechanic that reduces damage from physical attacks (like arrows and bombs), encouraging players to use the wizard tower strategically. This system will prevent players from spamming only one or two tower types without consequences. I plan on later making tips pop up throughout the game. For example, to explain how the armor works and which enemies have armor. Additionally, a visual on the enemies that have armor that way the player does not have to memorize which ones do and which ones don’t as that would be annoying. Currently, there are 6 kinds of enemies in the game. On the final wave I intend on adding a boss kind of enemy that will have a special power. Whether that is summoning enemies, or teleporting around the track, something to make him unique and memorable.

**6/11: DAY 5: Lives & upgrading system 0.9**

* Implemented a win/loss condition
* Towers can now be upgraded increasing their range and damage. Max upgrade cap: level 3
* Towers can now be sold for a fraction of their original cost. 70%
* Upgraded towers receive different visuals to indicate they’re upgraded.
* Display stats of selected tower on bottom of the screen.

The winning condition of the game is to survive all of the waves of enemies. The level starts with 20 lives and if enemies get past, you lose lives. The amount of lives lost depends on the enemy. Stronger enemies cost more lives than weaker enemies. The final boss costs all 20 lives meaning you have to beat him in order to win the game. If the number of lives drops to 0, the game is lost and you must start over from wave 1. Losing the game may be demoralizing however, I see it as a way of obtaining information about future rounds. Now that you’ve lost, you can be better prepared for what you lost against and improve your strategy. Adding tower upgrades felt necessary as the game would be too boring and final once all plots of land have a tower on them. I made it so that upgrading towers can sometimes be better than having more towers (quality > quantity). This encourages the player to consider the next waves of enemies and decide whether a higher fire rate is better, or having more damage. Furthermore, I added the selling mechanic. To sell a tower, you have to select it and hit s. This removes the tower from the plot and returns 70% of the money invested into it. Eg. if you sell a level 2 archer tower, you would get 70% of its total cost. “Returned money = (level 1 cost + level 2 cost) \* 0.7”. I added this so if the player feels a tower isn’t contributing enough, they can replace it with a better fitting tower. This further promotes player engagement, rather than just placing the towers. Lastly, I added the stats of each tower on the bottom of the screen when they’re selected. The stats are the damage, range, and fire rate of the tower. This helps the player know how much damage a tower is actively doing and how often. With this information, the player can judge when and where a tower would be best fit.

**6/12: DAY 6: Wave composition & refinements 1.0**

* Added a wave system. Waves of enemies can be called by pressing the spacebar.
* Added enemies to waves in an ascending manner. (early waves are easier, later waves are harder.)
* Added a highlight to the tower plots. Makes it easier to tell which one is selected.
* Added SFX to the towers attacks, enemy deaths, and BG music.

The waves of enemies are pre-defined in a 2d array. The first parameter being the wave number. The second parameter being the enemies that appear on said wave, and their order. To further vary the waves, certain waves have a shorter spawn delay, meaning the enemies will spawn more grouped together.

**6/14: DAY 7: Balance changes & QOL changes 1.1**

* Nerfed/Buffed certain enemies to better balance the game.
* Buffed the artillery tower as there was no point in building it before.
* Added a blue flag near the enemy exit and a red flag near the enemy entrance
* Added SFX to tower building, tower selling, tower upgrades, wave calling, and pre battle music/build music.
* Improved enemy sprites.
* Added a win and loss screen.

At this point of the project, the framework is finished and all thats left is to polish it up. I adjusted some enemies and towers. I noticed that at the start of the game, the player has no way to tell which way the enemies are coming from. To fix this, I added a red flag for the enemies and a blue flag for the player. Also, this entire time the enemies have just been circles of different sizes. So, I went ahead and remodeled a bunch of enemies to give them weapons or armor so they feel more complete. Finally, more sound effects were added to fill the emptiness of sound. Then the win and lose screen.