

Tower Defence Game

Aidan Saull

January 1, 2020

Basics

- Players play as government contractors tasked with saving the Earth
- Same starting contract amount
- After the first wave, the defense budget is split according to exactly how much work each company did
- In subsequent rounds, the defense budget is spent relative to how much each contractor contributed relative to their budget. (if all contractors do poorly relative to their contracts, they can all secure more money later)
- The player who had the worst performance will lose money due to damages if any enemies get through to the Earth.
- Game ends once the Earth is destroyed or all waves are cleared (maybe, waves may be infinite)
- Winner is the player with the biggest overall profit.

Gameplay

- Enemies have health and shields, some weapons work better on health and some on shields
- We do the math on how much it “costs” to destroy a point of shield vs a point of health and adjust performance scores accordingly

- First round enemies have only health and no shield
- Later enemies will tend to continue on a health and shield percent trend, however sometimes a round will counter the trend.
- Turrets are placed in any circular orbit. Orbits below 400km will decay after three rounds no matter what, all other orbits will never decay.
- Cost to put a turret into orbit increase as the orbit height increases. This is rewarded due to the fact that late game turrets have large ranges much greater than say 400km. Putting a 2000km range turret 600km above the earth is inefficient as range is wasted. Additionally, some of the best high range turrets cannot shoot through the Earth, and they will therefore see more if they are higher. Thirdly, high orbit turrets can achieve higher performance (idk what I meant by this). Fourthly, some enemies will move slowly at a high orbit, but very quickly through lower orbits.
- Turrets orbit at appropriate speeds. This means that turrets may, through bad luck have a very poor performance in any given round.

Economy - based on obsevation

- *At no point in the development of this game will I ever attempt to allow spending no money to be a legitimate strategy.*
- This section will be filled as observations about the economy round to round are made.

Phases

The game will be divided into two main phases, Purchasing and Defending.

Purchasing

During this phase the game will be paused, and all satellites will be at their “default” positions. The player will be able to:

- Purchase satellites

- Repair satellites
- Adjust the orbit of satellites that are capable of orbit adjustment

Defending

During this phase, time will become unpaused and the satellites will begin their orbits around the planet. The player will not be able to do anything except for watch.

Classes

GameController

Private Attributes

- `player_list` - list of Corporation
- `base_health`
- `round_number`
- `enemy_shield_percent` - apprx. percent of enemies health that is shield.
- `enemy_shield_pierce_percent` - apprx. percent of enemies that fire shield piercing weapons
- `enemy_types` - list of Enemy
- `game_state` - what stage the game is on. Build, Combat, or Recap
- `defence_budget`

Public Attributes

-

Methods/Functions

- `calculate_budgets` - returns the a list of budgets for each of the corporations
- `start_game`
- `start_round` - begins spawning of enemies and activates defences
- `end_round` - sums up the round, awards contracts and adjusts global budget

Corporation

Private Attributes

- `budget`
- `funds`
- `performance` - Score of enemies destroyed

Public Attributes

-

Methods/Functions

- `get_budget` - returns last round's budget
- `set_budget(budget)` - sets the budget for next round
- `get_performance(performance)` - returns last round's performance
- `update` - will monitor for player input when placing defences
- `place_defence(defence_type, position)`

Defence

A generic defense. Only children of this class will ever be instantiated.

Private Attributes

- health
- shield
- altitude
- current_target - an `Enemy` that this turret is currently targeting
- upgrades - a dictionary with keys as upgrade types, and values as upgrade descriptions
- current_upgrades - a dictionary with keys as upgrade types, and values as the current level of that upgrade.

Public Attributes

-

Methods/Functions

- update - set enemy to target, call attack, call move - *Not Implemented*
- attack(enemy) - attack designated enemy
- upgrade(upgrade_type) - upgrade this defence by the specified type once, returns True if completed, or False if not possible.
- move - simply orbit the Earth.

Enemy

A generic enemy. Only children of this class will ever be instantiated.

Private Attributes

- health
- shield
- current_target - a defence that this turret is currently targeting

Public Attributes

-

Methods/Functions

- `update` - set defence to target, call attack, call move - *Not Implemented*
- `attack(enemy)` - attack designated defence
- `move` - choose a path to follow down to the surface