Colin Young
Instructor: Ed Keenan
Course: GAM 386

Game Design Spec

Title: Turret Commander

Game Summary

Turret Commander is a touch-based action game. It's very simple, and has only one level. It incorporates elements of Asteroids and Tower Defense in a fun, naturally touch-based package.

Basic Gameplay: The player controls a Turret on the stage, which is contiguous (wraps around) like in Asteroids. Figure 2 contains a screen shot of the turret (with no enemies).

The game takes place on a 3x3 grid of iPhone-screen sized screens. There are enemies (essentially asteroids) that move towards bases in the center of each grid square. Each one has a damage meter, and if they all drop down (or if three in a row are out) you lose. Your ship is invincible and hitting enemies causes them to explode. You move by shaking, dragging, and attack by touching and rotating (see the controls section for more).

Reference Material

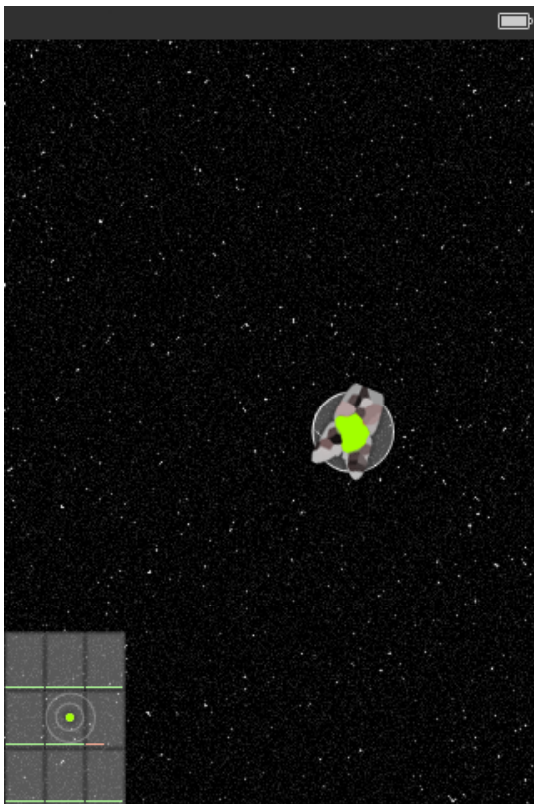None other than Asteroids and the art section.

Game detail

In a typical level, the game starts with a 3x3 grid with a few enemies. As the round goes on, more and more enemies come into the different screens. They only come on from the sides, which your turret cannot move outside of (there will be a little 'laser fence' or something to block it).

The player will score based on how many waves they last for. Waves will either be a certain amount of time long, or will end when all enemies are dead. If time allows, I'll implement a points counter for killing enemies or pickups that regenerate/resurrect your bases.
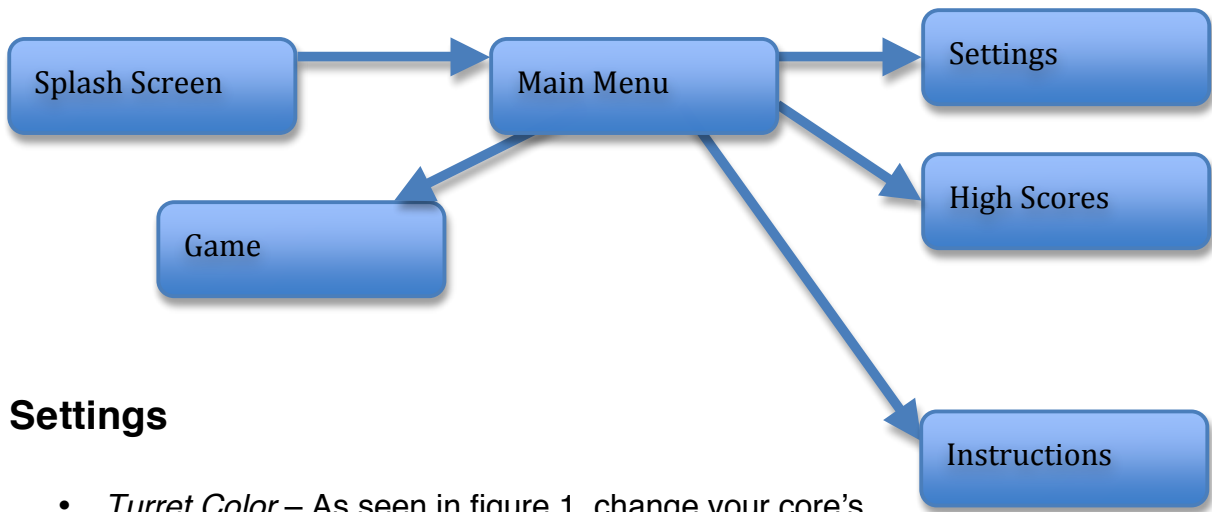
Sound

There will be background music (a friend has an electronic music project) and explosions. Warning sirens will sound when the user has only a few zones left or there is about to be 3 in a row knocked out. Lastly, there will be shooting sounds for the turret bullets.

Art

User Interface

```
Splash Screen  →  Main Menu  →  Settings

Main Menu  →  Game

Main Menu  →  High Scores

Main Menu  →  Instructions
```

Settings

- *Turret Color* – As seen in figure 1, change your core's energy color
- *Difficulty* – controls the number of enemies and their speed
- *Tap-and-drag* to toss your turret

Controls

- *Touch to shoot* – bullets simply go where enemies are (no collisions – it's just if you clicked on an enemy or not)
- *Tap-and-drag* to toss your turret
- *Shake gesture* to jump to hotspot.
- *Rotate Left-right when powered up* to kill all enemies on screen, but switch orientation, requiring an adjustment and making the game more difficult.

The twist is that the game offers a MiniMap (It's a 3x3 grid) a la Starcraft. It's not a square, however; it's slightly elongated (made up of iPhone-screen-shaped zones). If a large amount of enemies are in a zone, you can move to it by shaking your phone in a simple shake gesture. If you 'toss'-drag (pick the turret up and throw it) in a direction, you can also move around that way (this allows you more control than the jump, which simply moves you to a hotspot. You must protect the zones, or they'll die.

Risks/Hazards

The game sounds complex, and this worries me, although I have taken care to ensure that the changes will require me to create robust, reusable code. For example, the rotate gesture (probably the hardest part) will require me to code the grid and the interface consistently and adaptively. Coding it in that fashion

would help me immeasurably if, for example, I ever wanted to adapt it to an iPad version. I'm actually hoping to sell the game, so I want to challenge myself upfront. Collisions confused me, so I wanted to make them as simply as possible – the only collisions I'll need to handle are if an enemy touches your turret or your bases, it explodes, which should be a relatively simple collision, since they cannot bounce and will only move on a single vector. There is no AI on the enemies, just static, randomized, single-direction motion, and user-controlled motion is either drag-based or a single vector.