Last Hope (Open Frameworks Final) Spec

# About this Document

This document describes a proposed deliverable for the final assignment of our Open Frameworks class. This is meant to outline basic functionality of the build as it relates to class. Some details are deliberately left blank to be determined by the speed of the development of the critical elements listed here.

# Gameplay

The gameplay of Last Hope is a variation on a dueling card game mechanic a la Hearthstone, only using real-time agents with autonomous decision-making as units. Each player has a mothership, and the goal of the game is to destroy the other player’s mothership by reducing its health to zero. Players play by deploying units, which are smaller ships, that they appear in the field and then enact behaviors. Deploying these units is the core mechanic of the game.

Players have a pool of points that regenerates over time. Generating a unit costs the player points, and different units have different costs. If the player does not have enough points to generate the unit, the unit cannot be generated. Units are generated by clicking a button named for the unit, which causes the unit to appear right in front of it.

Units have different attributes that determine how they move and how other units respond to them. Unit attributes include:

* Speed
* Target
* Damage Frequency
* Damage Done per Hit
* Health
* Attractiveness to other units

When a unit is deployed, it immediately moves towards its target. When in range of its target, it engages the target, circling it and shooting at it. When a target does damage, it shoots a laser (i.e. draws a line) to its opponent. When a unit is destroyed, it explodes in a pool of particles.

The key role of the lessons of this class comes in the movement of the ships. Ships should not simply sit on screen shooting at each other, so I want to create patterns that emulate ships moving in space. As the game is 2D, I’m going to stick to planar movement for this, but I want to include at least three critical movement effects:

* Ships should never collide with each other, and should design paths that they avoid each other as they move. Each ship’s movement should be a force attracted to the target and repelled by any nearby ships.
* When ships are in combat with each other, they should orbit each other in a loose loop that avoids collision. I’m going to investigate flocking behavior here, as this should sustain a variable number of ships at once.
* When a ship is attacking the mothership, it should do an interesting loop in front of it.