Neon Commander

Product Scope Documentation

Dancing Emu Studios (Heath, Jarrod and Richard)

Advanced Diploma in Professional Game Development

Assessment 4 - Cross Platform Development

Table of Contents

Justification

Our group is required to create a polished game for multiple platforms us to pass the course. This will also give us experience at making a game from start to finish and working in a group. So to solve this issue, our group (Heath, Jarrod, Richard) is going to make a game for the PlayStation Vita and the Windows operating system.

Scope Statement

This project will consist of making a potentially publishable game for both the Windows operating system and the PlayStation Vita. The game will be in a two dimensional top down vector shooter, inspired by retro games such as Asteroids and Gravitar as well as more recent games like Geometry Wars. The game will feature: At least three types of enemies each with different behaviours, at least five different power ups for the player to collect and use against the enemies and possibly more than one game mode. The game will have an art style heavily featuring outlines and neon effects. Some of the milestones of this project include: Having the required documentation to actually start creating the game completed before the 3rd of September, having all of the art completed by the 30th of September and having all the documentation and game completed before October 16th. The deadline for the entire project is on October 17th.

Deliverables

Week 1 *(Wed 3rd - Tues 9th)*
- All game states basically implemented on PC and Vita.
- Camera implemented and working as intended.
- Player moving around correctly.

Week 2 *(Wed 10th - Tues 16th)*
- Player fully implemented.
- Seeker enemy fully implemented.
- Missile enemy fully implemented.

Week 3 *(Wed 17th - Tues 23rd)*
- Multi-direction shooter enemy fully implemented.
- *TBD Enemy* fully implemented.
- Power ups started.
- Improved art.

Week 4 *(Wed 24th - Tues 30th)*
- Power ups completed.
- All enemies fully implemented.
- All art has been improved.

Week 5 *(Wed 1st - Tues 7th)*
- Implementing endless mode.
- Implementing score attack mode.
- Tweaking game balance.

Week 6 *(Wed 8th - Tues 14th)*
- Polishing GUI features. (Improving art)
- Finishing documentation.
- Tweaking game balance.

Project Exclusions

This project will not produce an engine or a framework, as we are using MonoGame and the XNA Content Pipeline. It will not feature multiplayer on any platform, online or co-op.

Constraints

Time
We have roughly seven weeks to complete the project, with 20 hours a week spent at university. With three people in the group, that's a minimum of 60 work hours a week which should be more than enough to complete the project.

Money and Resources
All of us in the group have extremely low amounts of money to spend on resources. This means that all of the resources we gather must be free or be made ourselves. This is a potentially large hindrance as art can be time consuming to make and is easier to outsource. If the required resources cannot be gotten then our only option is to reduce the project scope.

Acceptance Criteria

Fully Implemented Seeker Enemy
- Enemy follows player using forces. Starts following the player as soon as it is spawned.
- Must not be doing a sine or cosine calculation every frame.
- When killed it must explode with particle effects called by itself.

Fully Implemented Missile Enemy
- Enemy launches towards the player using forces. Only launches when the player is in range and in line of sight.
- If the enemy collides with any other enemy it must explode, killing everything within the explosion radius.
- When killed it must explode with particle effects called by itself.

Fully Implemented Multi-Directional Shooter Enemy
- Enemy must shoot bullets in multiple directions that don't collide with enemies, only the player.
- When killed it must explode with particle effects called by itself.

Fully Implemented Power ups
- Powerup must behave as intended. *(As outlined in the design document)*

Fully Implemented Player
- Player must be able to move and shoot, freely and intuitively, without bugs. *(Through testing)*
- Must have a method to alter the players powerup status. *(Shielded, multi fire, etc…)*