Atrius – DevLog #1

Introduction/Elevator Pitch

Kairos is an RTS Game set in a corrupted fantasy world in which you must fight to survive and reclaim the land. As the player, you will be able to build your kingdom and train units to help your people survive the corruption. Be careful however, because the corruption is always growing closer. Train up units in holy magic to help push the corruption back. Using many unique tools and strategies, you will be able to cleanse the world of this evil.

In Kairos, every world is unique and randomly generated. Changing the playstyle of every run. Challenge yourself with a deeply corrupted world, or have some fun building and expanding in a world with slow-growing corrupted land. While the idea of corruption has been used by many games in the past, we believe Kairos will bring a new experience to the concept.

Our team is developing using the Unity Engine and visual studio. We manage our project tasks using Jira and our project repository is stored on GitHub. We keep ourselves on track using agile-scrum team management.

Over the past few weeks, we've put a lot of focus into perfecting our map generation and unit behavior. These are the core of any RTS game, and we feel that completing these 2 items has been a major win for our team. As such, we'd like to give you a sample of what they look like.

We use a is voxel-based world and perform random generation using a modified noise algorithm, which gives us very realistic looking terrain. We wanted to make sure that our players get the experience of completely purifying the world, so, for primary gameplay, our world is generated as continent. This makes the corruption spread and purification much easier to gauge and more realistic when it comes to keeping your kingdom alive.

For our unit behavior there 2 primary components. Pathfinding, which allows the units to move around the map, as well as a flocking algorithm, which is what makes the units move in a natural manner and allows them to avoid each other. For Pathfinding we use the A\* algorithm which allows for finding efficient paths on the map. However, if units follow these paths directly, they move way to rigid, and it takes too much time to calculate the paths for a large group of units. To solve this, we also implemented a modified boids algorithm. Using this algorithm, units are able to avoid one another as well as move together as a unified group. This also allows us to run pathfinding a single time for a large selection of units, making unit movement much more efficient.

At the beginning of the quarter, the whole team felt as if we were in the red. Things weren't progressing as fast as we had hoped, and we were getting nervous that we wouldn't be able to finish our product on time. During our meeting with Professor Cameron, he helped us more clearly define what our product should look like. Following his advice, we have decided to focus on creating a bare-bones gameplay demo. This demo will showcase all of the core features required for an RTS Game. Use those as building blocks, we hope to be able to speed up our development and finish the game on time. Our goal is to complete this by Febuary 21st.