CS179N Final Writeup

gcort002, jbeto001, wli025, mcao004, lle018

June 2018

1 Assets and Resources

All assets in Nothing Is Sacred, including character sprites, backgrounds, user interface elements, icons, full-screen artworks, and music, were created by team members and are original to us, with the following exception:

"After The Fall" title score provided by Purple Planet Music, with attribution per their **usage permissions page**. A link to the song's download source can be found by **clicking here**.

2 Achievements and Omissions

Using our original project proposal as a checklist, we were successful in implementing the following features in our game:

- Fully-illustrated map with panning and zooming.

- Original art assets for player and enemy units visible on map.

- Menu icons for all tradeable goods in game.

- Original title screen artwork.

- Real-time clock with display and "fast forward" button to speed up passage of game time.

- Character movement using spline drawing.

- All major classes related to game logic.

- Basic economic system which accomplishes the following:

    · Produces and consumes goods (on a per city basis) resulting in an active, live economy.

    · Adjusts prices based on local supply of a given good in a given city.

- Government units with patrol, chase, and apprehend states, and a visible field of view.

- Global government A.I. which monitors cities and responds to changes in their states.

- In-game musical score.

We were unsuccessful in implementing the following features originally stated in our project proposal:

* Fully-connected and implemented Government A.I. strategy including storehouse raids.

* Sound effects for menu interactions, trade transactions, Unit movements, and so forth.

* Endgame conditions based on the values of Tyranny and Complicity remain inert and unimplemented.

* The following functions related to Heat remain unimplemented:

  · Buying or selling "hot" items in a city results in an increase of Heat in that city.
  · The Heat value of a city should decay over time as long as the player does not engage in activity there.
  · "Hot" items should command a greater price on the market. As it stands, there is no price modifier to take into account the illegality of a particular good.

The main reason we missed the above-objectives were related to problems implementing other features: namely the inventory.

The inventory is the primary means that the player has for buying, selling, and transferring goods. The main problem we had with the inventory is that it produced "two-way street" logical errors.

For example, whenever a city produces or consumes goods, the types of good-icons which appear in its storehouse change. When a city produces goods, more icons should appear. When a city consumes goods, fewer icons should appear. In this scenario, logical value changes produce UI changes.

As another example, if a player sells food to a city, the quantity of food in its goods-hashmap should be updated to reflect the purchase. In this scenario, a UI change produces logical changes.

We spent a great deal of time trying to get the logical mutator functions and inventory display functions modular enough so that information could flow in

both directions. We encountered numerous errors (null reference exceptions to goods that appeared in the visual menu but "didn't exist" in the structure tracking their positions, cities starving in spite of that fact that food was being sold to them, etc), and that bug fixing took up time that we had intended to devote to implementing our missed features.

The other main issue was that many of the economic system and Government A.I. functions could not be implemented until the large pieces of the game were built, connected, and operating relatively bug-free. You can't run a raid on a storehouse and clear it out until you have a city, and a storehouse, and things inside of it to be taken. Since implementing and fixing the inventory took so long, many of these features were deferred to the final week and only received a few hours of work.

In the end, the final project ended up being very close to what we originally aimed for.