

Sprint 7

10 day duration (Ends 23rd of June, 2017) ~ 116 pomodoros of work

Product Backlog Tasks

1. As a player, I have access to a variety of maps that offer me different challenges and opportunities.
2. As a player, I have the ability to save and load my game whenever I want.
3. As a designer, I have a collection of intuitive and robust tools that allow me to design maps efficiently without concerning myself with the technical details of the game's implementation.
4. The map that players play on looks like a cohesive object.
5. The hierarchy of menus and screens works properly and in a way that allows designers to gracefully bypass it.
6. As a player, I am motivated to expand my transportation network and develop my region.

Basic Design

Story

Players are some sort of governing figure attempting to develop a largely empty region by establishing an efficient transportation network that allows societies to grow and complexify.

Mechanics

The current game is an implementation of prototype 3.2. Scoring is a simple measure of the numbers and complexities of various societies, with victory being achieved when a certain point total has been reached.

Aesthetics

The aesthetic style of the game remains simple geometric shapes and flat shades, as going beyond this is beyond the scope of my abilities. I do plan to have a cohesive map structure by the end of development.

Technology

Unity deployed to PCs, and possibly to browsers, as well, depending on performance issues.

Risk List

1. My design isn't based on rigorous and repetitive playtesting, and it seems unlikely I'll be able to playtest it properly.
 - a) I need to continue placing this risk down even though I know I'm not going to address it. I can't get complacent that this decision not to playtest, whether for legitimate or illegitimate reasons, is an acceptable result. I can't let unplaytested games normalize in my mind, given how critical of an aspect of game design it is.
2. The map might not look like a cohesive object, which might severely affect both the game's aesthetics and the differentiation of various neighborhoods.
 - a) The algorithm that I began working on during the end of Sprint 6 should suffice to divide the map into cohesive pieces with few modifications. That will require trying to understand and implement the seemingly complex problem of dividing an undirected graph into its component polygons, but since I've found an algorithm of sorts that claims to perform that task I have a clear way forward for pursuing that task.
3. It might take too long to figure out how to partition the map into polygonal regions of terrain.
 - a) I should make the map-dividing task a non-critical one, instead addressing components of the game that absolutely must exist for it to function properly. Likely it will be of Important priority. That way, even if I can't develop my algorithm in the time remaining, the project will still be in a reasonable state.
4. The game might not have proper victory or defeat conditions.
 - a) I need to establish some activity that occurs when victory is achieved, and figure out what victory looks like on each map. Likely each map will need attached to it some notion of the amount of points required for victory, and I will at the minimum need to create a splash screen that pops up when victory is achieved. I may or may not attempt to record best times on a per-player basis. I can leave that as a desirable task, perhaps.
5. Unexpected problems might arise when I attempt to build the project to a deployable format.
 - a) I will need to start building and testing the program as its deployable very soon, probably no later than the 14th, so that I can resolve any problems that arise in the standalone or the web application if I decide to deploy it there. This will also allow me to check performance in a more realistic context, which may be different inside the editor than outside. Checking the deployable must be a critical task.
6. Leaving ResourceBlobs unsaved might cause frustration and unnecessary difficulties in gameplay. The fact that stockpiles will disappear means that certain configurations at the edge of stability might collapse on a session reload, which is not desirable.
 - a) I had originally considered the saving of ResourceBlobs to be more complex than it was worth, but I can always engage in a half-measure if I find it necessary. It wouldn't be too difficult to save blobs that are in BlobSites by attaching more information to MapNode, no more than a Dictionary<ResourceType, int>. That way I can reconstruct stockpiles and prevent possible destabilizations of delicately balanced configurations.
7. I don't have a collection of maps that demonstrate the engaging complexity of the mechanical system I've built.
 - a) I'll need to get on that then, won't I? Before I do that, I'll need to decide what sorts of challenges might be interesting given the current mechanical structure of the game. What are interesting difficulties I can throw at players? What are problems they'll need to grapple with? What topologies will demonstrate the decisionmaking complexity of the game? These aren't questions I have immediate answers for, nor will I likely be able to generate them over the next two weeks, but I can at least start that process.
8. I don't have a clear way of teaching new players to play the game, or to limit the amount of new information I'm presenting them so that they can learn for themselves more easily.
 - a) I'll need to break the current structure of the game into a series of facts or techniques that must be learned, then design maps that are capable of teaching them. I might build a

system that controls the order by which players can approach maps, so that they start with the maps that teach basic fundamentals before moving on to more complex configurations.

9. The game's presentation might leave a lot to be desired, impacting player perception and thus player experience.
 - a) There are a variety of things I can try to address this issue. Establishing the map-drawing algorithm stipulated above is one way. I can also look for (or maybe even design myself) a few simple sound effects for certain activities. Maybe when resources are produced they make little popping noises. Maybe UI elements respond with some sort of audio component when buttons are pressed or panels opened. Maybe when societies ascend and descend they make noises. I suppose I could also try to find some music to lay beneath the game, though that seems like a more involved task.
 - b) I could also try to improve the graphical appeal. There may be free assets I could replace my cylinders and boxes with to try and improve appearance. I could even try to build my own simple ones if I felt ambitious. It's harder for me to imagine what that would look like, however. Finding effective models seems harder than finding little noises I can add to certain game tasks.
10. Performance issues may continue to crop up, negatively impacting player experience on well-developed maps and weak hardware.
 - a) I'll have to see what the deployable product looks like, how it performs and if that performance interferes with the core gameplay loop. Performance is, unfortunately, probably not a priority even at this stage of the project, though if I have time I can always profile the game and see what systems need to be made more efficient. If there's any low-hanging fruit I can always try to refactor it.
11. I don't have a way of saving more complex session configurations at design-time, which means that I cannot establish maps for the players to play that contain elements other than nodes and edges.
 - a) I'll need to abandon the notion of the MapAsset and add controls to the editor to allow for the creation of sessions at design time. I'll probably also need to establish a second directory in which sessions as maps are stored, as I doubt the standalone executable wraps up information in the persistent data path. Nor do I want players to accidentally overwrite maps, as they could if I stored them in the same directory.
12. HighwayManagers continue not to update in anything approximating real-time, which could make that display element confusing and user-unfriendly.
 - a) It wouldn't be too hard to set up a coroutine on HighwayManagerDisplay that pings the appropriate Control module, requesting a new display summary based on the ID of the summary it currently has. I could also make the display summary a wrapper for the HighwayManager itself and have HighwayManagerDisplay use coroutines to periodically refresh its display.

Sprint Backlog

To Do

Critical (0 pomodoros of work remaining)

- 1.

Important (9 pomodoros of work remaining)

1. Change the victory system so that it requires a stable or semi-stable configuration in order to reward victory.
 - a) Estimated 1 pomodoro to complete.
2. Implement a second map concept.
 - a) Estimated 8 pomodoros to complete.

Desirable (31 pomodoros of work remaining)

1. Build and test a web application, resolving any issues that might cause it to diverge from the game as viewed in Play Mode in the editor, or determine that it is not worth the effort.
 - a) Estimated 5 pomodoros to complete.
2. Profile the game and identify performance bottlenecks.
 - a) Estimated 5 pomodoros to complete.
3. Change HighwayManagerDisplay so that it periodically updates itself while open.
 - a) Estimated 1 pomodoro to complete.
4. Add sound effects to the creation, destruction, and completion of ConstructionZones.
 - a) Estimated 3 pomodoros to complete.
5. Add sound effects to the complexification and decomplexification of Societies.
 - a) Estimated 3 pomodoros to complete.
6. Modify Society complexities so that they have their own meshes and materials that societies them apply to themselves.
 - a) Estimated 1 pomodoro to complete.
7. Find meshes and materials for the various Society complexities.
 - a) Estimated 8 pomodoros to complete.
8. Add royalty-free music that plays in the background of the game.
 - a) Estimated 3 pomodoros to complete.

In Progress

Critical

- 1.

Important

- 1.

Desirable

1. Add sound effects to the drawing of highways.
 - a) Estimated 2 pomodoros to complete.

Completed

Critical

1. Extend session saving and loading so that it can save map configurations at design time and load them as new games during runtime.
 - a) Estimated 2 pomodoros to complete.
 - b) Took 3 to complete.
2. Change serializable sessions so that they have and display descriptions and the number of points necessary to achieve victory on them.
 - a) Estimated 1 pomodoro to complete.
 - b) Took 2 to complete.
3. Establish a series of lessons that players need to learn in order to play the game properly.
 - a) Estimate 2 pomodoros to complete.
 - b) Took 2 to complete.
4. Implement the creation of highways at design time.
 - a) Estimated 4 pomodoros to complete.
 - b) Took 9 to complete.
5. Consolidate most (or all) of the disparate editor code into a single window, and use the consolidation to improve the experience of designing maps.
 - a) Estimated 5 pomodoros to complete.
 - b) Took 5 to complete.
6. Implement and greenlight all of the unit tests.
 - a) Estimated 3 pomodoros to complete.
 - b) Took 4 to complete.
7. Build a system that makes it easier to set highway permissions at design time.
 - a) Estimated 2 pomodoros to complete.
 - b) Took ~3 to complete.
8. Create a series of maps that teach the necessary lessons.
 - a) Estimated 5 pomodoros to complete.
 - b) Took 7 to complete.
9. Build a system that records the maps a player has beaten and controls player access to maps based on which other maps they've beaten, so that designers can force players to engage in one set of challenges before taking on another.
 - a) Estimated 3 pomodoros to complete.
 - b) Took ~4 to complete.
10. Establish some sort of splash screen that displays when victory has been achieved on a given map.
 - a) Estimated 2 pomodoros to complete.

- b) Took ~1 to complete.
- 11. Build some number of interesting challenges or situations that players might find engaging.
 - a) Estimated 2 pomodoros to complete.
 - b) Took 1 to complete.
- 12. Implement one of the map concepts.
 - a) Estimated 8 pomodoros to complete.
 - b) Took ~5 to complete.
- 13. Build and test a standalone deployable, resolving any issues that might cause it to diverge from the game as viewed in Play Mode in the editor.
 - a) Estimated 5 pomodoros to complete.
 - b) Took 3 to complete.
- 14. Repair all of the maps that were broken by changes to the Sessions module.
 - a) Estimated 8 pomodoros to complete.
 - b) Took 1 to complete.
- 15. Resolve the issues with SerializableSession incompatibility.
 - a) No estimate provided.
 - b) Took 8 pomodoros to complete.
- 16. Rebuild all of the maps that were broken.
 - a) Estimated 5 pomodoros to complete.
 - b) Took 6 to complete.

Important

- 1. Transfer a version of the HexGrid from MerchantRepublics into UnityCustomUtilities.
 - a) Estimated 5 pomodoros to complete.
 - b) Took 4 to complete.
- 2. Create and test a subclass of HexGrid that creates hexes properly.
 - a) Estimated 3 pomodoros to complete.
 - b) Took ~ 3 to complete.
- 3. Build a system that assigns TerrainTypes to hexes in the HexGrid based on the nearest MapNode, within some maximum distance.
 - a) Estimated 2 pomodoros to complete.
 - b) Took 1 to complete.
- 4. Alter the terrain grid so that it associates hexes with their nearest node and outlines terrain regions centered on nodes.
 - a) Estimated 3 pomodoros to complete.
 - b) Took 10 to complete.
- 5. Integrate the HexGrid-driven terrain drawer into SessionManager and SerializableSession.
 - a) Estimated 3 pomodoros to complete.

- b) Took 1 to complete.
- 6. Add new unit tests to account for new behaviour.
 - a) Estimated 5 pomodoros to complete.
 - b) Took 10 to complete, when wrapped up with the below task.
- 7. Greenlight all existing unit tests.
 - a) Estimated 3 pomodoros to complete.
 - b) Took 10 to complete, when wrapped up with the above task.
- 8. Extend session serialization so that it saves stockpiles of resources within MapNodes, and also the current position of the camera.
 - a) Estimated 1 pomodoro to complete.
 - b) Took 1 to complete.
- 9. Add sound effects to the pressing of various buttons and the clicking of various toggles.
 - a) Estimated 2 pomodoros to complete.
 - b) Took 3 to complete.
- 10. Add sound effects to the selection and deselection of MapNodes.
 - a) Estimated 1 pomodoro to complete.
 - b) Took 1 to complete.
- 11. Add sound effects to the opening and closing of various menus.
 - a) Estimated 2 pomodoros to complete.
 - b) Took 3 to complete.

Desirable

- 1.

Abandoned

Critical (hours of work remaining)

1.

Important (hours of work remaining)

1. Find or create unit tests that prove the correctness of the polygon detection algorithm I found, as well as its component algorithms.
 - a) Estimated 8 pomodoros to complete.
 - b) Task and its related components were ultimately considered too complex, too difficult to implement, and not useful for the problem at hand.
2. Greenlight the unit tests for the polygon detection algorithm, and its component algorithms.
 - a) Estimated 8 pomodoros to complete.
 - b) Task and its related components were ultimately considered too complex, too difficult to implement, and not useful for the problem at hand.
3. Build unit tests to prove the correctness of my map partitioning algorithm.
 - a) Estimated 5 pomodoros to complete.
 - b) Task and its related components were ultimately considered too complex, too difficult to implement, and not useful for the problem at hand.
4. Greenlight the map partitioning algorithm unit tests.
 - a) Estimated 5 pomodoros to complete.
 - b) Task and its related components were ultimately considered too complex, too difficult to implement, and not useful for the problem at hand.
5. Integrate the map partitioning algorithm into MapGraph proper.
 - a) Estimated 1 pomodoro to complete.
 - b) Task and its related components were ultimately considered too complex, too difficult to implement, and not useful for the problem at hand.

Desirable (hours of work remaining)

1.

Review

Retrospective