Terrafarm

1. Formal Game Proposal

1.1 Game Description

Terrafarm is a turn-based strategy game set on a previously uninhabited planet, in which the player establishes a foothold to quickly exploit the planet before the collapse of its ecosystem. The player is able to erect several types of buildings to acquire energy and mass, which can be used to expand. While the player's empire grows and might eventually span the entire planet, they also need to deal with climate catastrophes and other hazards that are caused by careless exploitation of the planet's resources.

The player needs to fight against the threats caused by self-induced climate change and acquire as much money as possible before the planet's ecosystem collapses – leading to the inevitable demise of the player's empire. As the world is generated randomly, each game offers a slightly different experience.

1.2 Story

The player is part of an alien species which is rapidly expanding their reach to ultimately colonize the entire galaxy. To reach this goal, they have adapted their biology to quickly transform themselves into industrial buildings. The player settles on an unmarred planet, rich with resources, with the ultimate goal of generating as many resources as possible while interacting with the planets climate and ecosystem.

1.3 World Layout

The shape of the planet is a truncated icosahedron. Each hexagon (or pentagon) serves as a spot where the player can construct a building. The tile itself has a terrain type, which can constrain what type of building can be constructed, or influence its performance in positive or negative ways.

1.4 Resource Types

The game features two basic types of resources: Energy and Mass.

Some buildings generate energy, which is required to sustain the operation of buildings. Other buildings generate mass, which is used to create new buildings. Both can be exchanged for the other at the galactical market (at a reduced rate), and at the end of the game are sold for credits to calculate your score.

1.5 Building Ideas

The player can erect various power plants to generate energy and mass. However, each building has advantages and drawbacks. Some may generate more power in exchange for more pollution, while others may pose different dangers.

- Combustion Power Plant: This power plant burns biomass to generate a fair amount of energy, but also quickly pollutes the environment, possibly reducing the players turn limit due to climate events.

- Fission Reactor: The fission reactor splits atoms to generate vast amounts of energy and pollution. Additionally, it carries the risk of exploding, damaging the player's structures and leaving behind a large financial gap. It is a high-risk, high-reward building!

- Wind Turbines: Wind turbines generate a low amount of pollution, but also a limited amount of energy.

- Mass Extractor: Pulls biomass from the planet's terrain. Uses large amounts of energy.

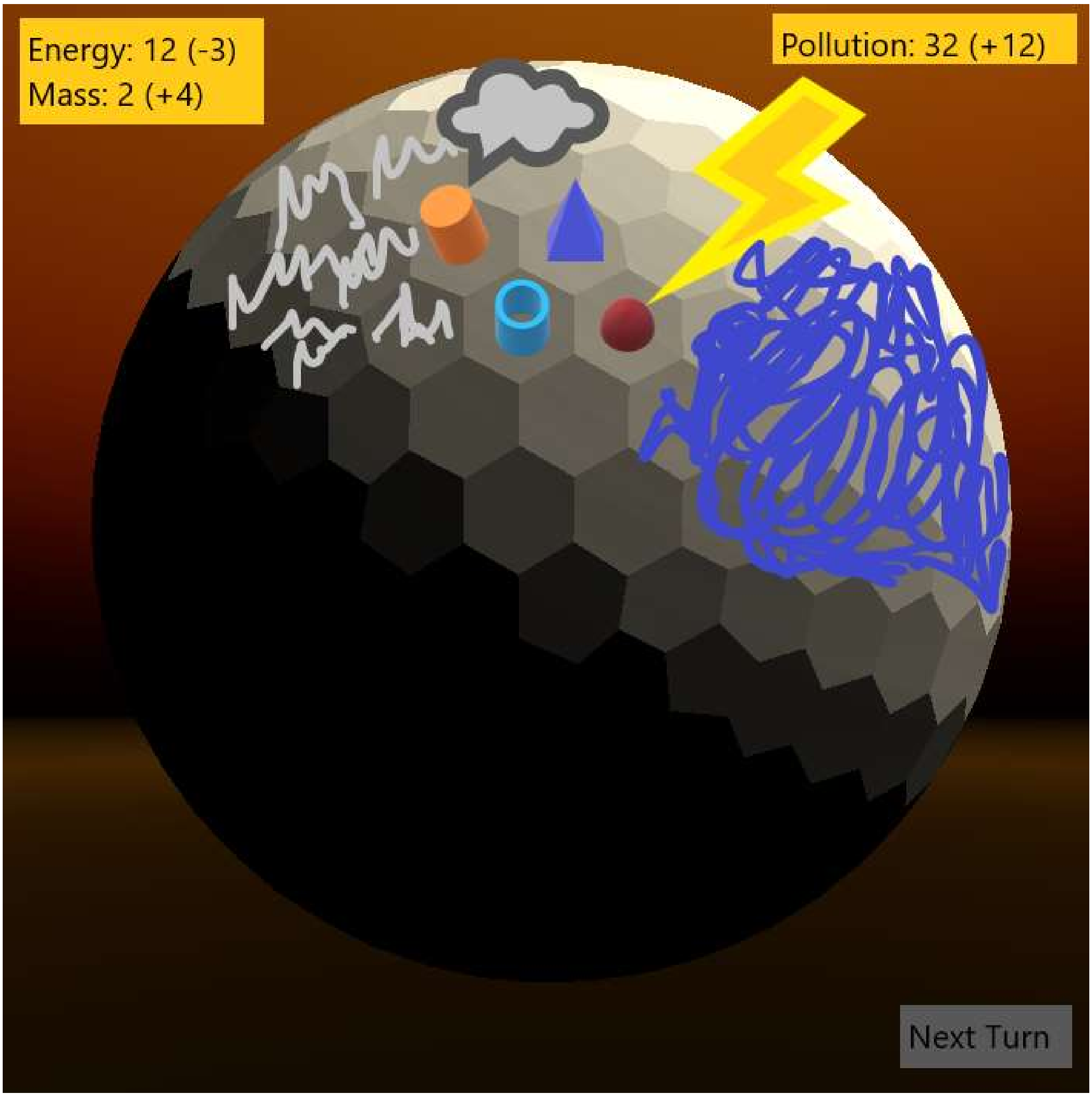
1.6 Climate Effects Ideas

Pollution from buildings will begin to heat up the atmosphere, ultimately making the planet uninhabitable due to high temperatures.

Additionally, during gameplay, other random events may happen depending on the player's pollution:

- Lightning Strike: May hit player's building causing it to take large damage
- Earthquake: Deals low damage over a wide area
- Ion Storm: Lowers the production of a building for the next X turns

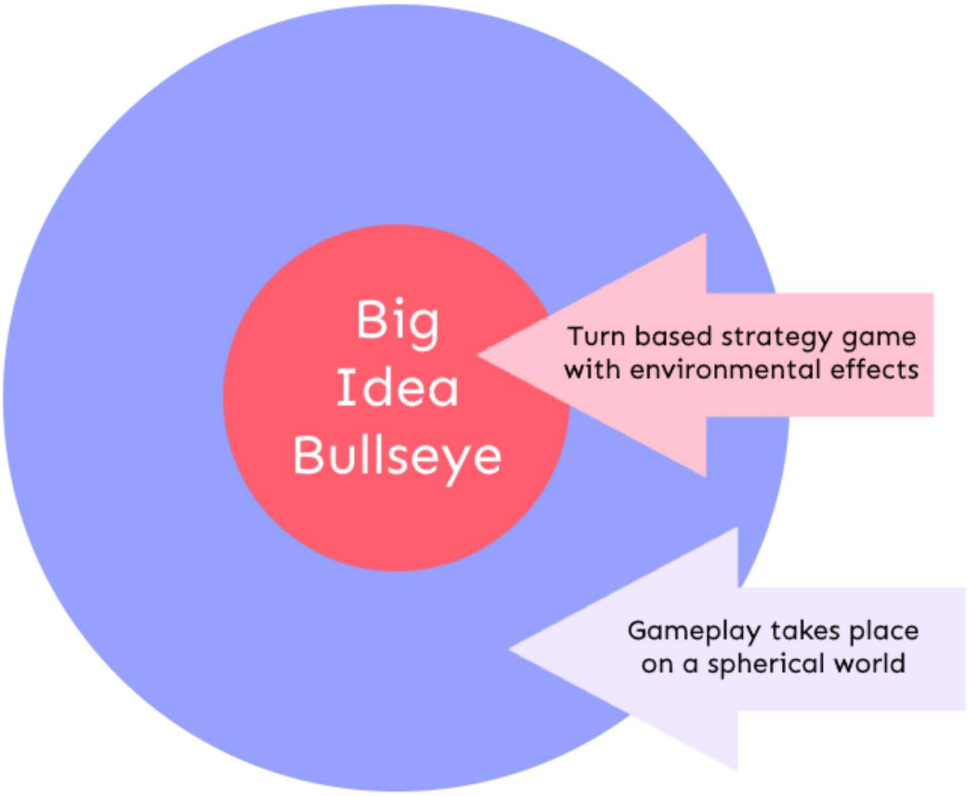
1.7 Concept Art



1.8 Technical Achievement

Our technical achievement will come from the fact that the game takes place on a truncated icosahedron, so extra precautions need to be taken in areas such as graphics, camera control and coordinate systems. Additionally, we would like to incorporate procedural elements into the world layout such as mountains and water to increase replayability.

1.9 Big Idea Bullseye


Big Idea Bullseye
Turn based strategy game with environmental effects
Gameplay takes place on a spherical world

1.10 Development Schedule

1.10.1 Layered Development Schedule

Functional Minimum

- World Generation
- Input System
- Camera Controls
- Selectable Starting Location
- Buildable Buildings
- Collectable Resources

Low Target

- World features (Water, Mountains)
- Climate effect depending on player-generated pollution
- Simple UI

Desirable Target

- Buildings only placeable on certain terrain
- More building types
- More climate effects
- High scores
- Background Music
- Sound effects

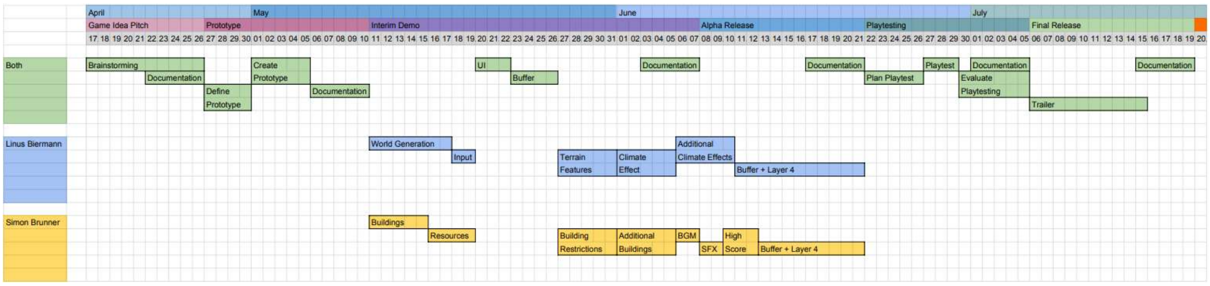
High Target

- Fancier graphics
- Save / Load
- Animated Assets
- More resource types
- Upgradable buildings
- Terrain modification
- More world features
- Implement Story
- World Events

Extras

- Asymmetric Multiplayer
- Online leaderboard
- Mobile platform

1.10.2 Timeline



1.10.3 Task Outline

Task	Description	Member	Planned	Actual
World Generation	- generate truncated icosahedron with variable size - layout data to easily access buildings, tiles, etc - foundation for world features - selectable starting location - flat shading	Linus	15	
Input System	- selectable tiles - camera controls	Linus	4	
Buildings	- integrate 4+ buildings - resource generation - pollution / effect on the environment	Simon	15	
Resources	- energy - mass - credits	Simon	5	
UI	- user interface for buildings, construction, resources and various other information panes	Both	10	
Core	- integrate all systems to create a playable prototype	Both	10	
Terrain Features	- mountains, water, ... - make data easily accessible for building perks / detriments / restrictions	Linus	5	
Building	-make buildings placeable on certain	Simon	10	

Restrictions	tiles -impact of the environment (mountains, water) on resource generation			
Climate Effect	- layout foundation different types of effects with variable sizes, effects, … - create visuals for effects	Linus	10	
Additional Content	- more buildings - more climate effects - extra resource types? - upgradable buildings - world events - terrain modification - add deeper story integration - asymmetric multiplayer - online leaderboards - mobile platform?	Both	∞	

1.11 Assessment

We hope to appeal to a wide audience through the combination of simple graphics, strategic elements and slower gameplay. Players will have time to think about where they want to place buildings, and how those buildings will interact with the resources they have, the terrain around them and their remaining time. However, since we have only half the members at a team of two, we must be cautious with our planning and make sure the scope of the game does not get too large.

Our plan is as follows: Work on the core features (buildings, resources, world generation, climate effects) and finish these as quickly as possible to establish a good baseline, then work on additional content to hopefully create an engaging experience.

2. Prototype

A turn-based strategy game like this one is perfectly suited for a physical paper prototype, as there are no reaction-based action elements involved. In theory, the entire game could be played on paper, without the need for a computer.

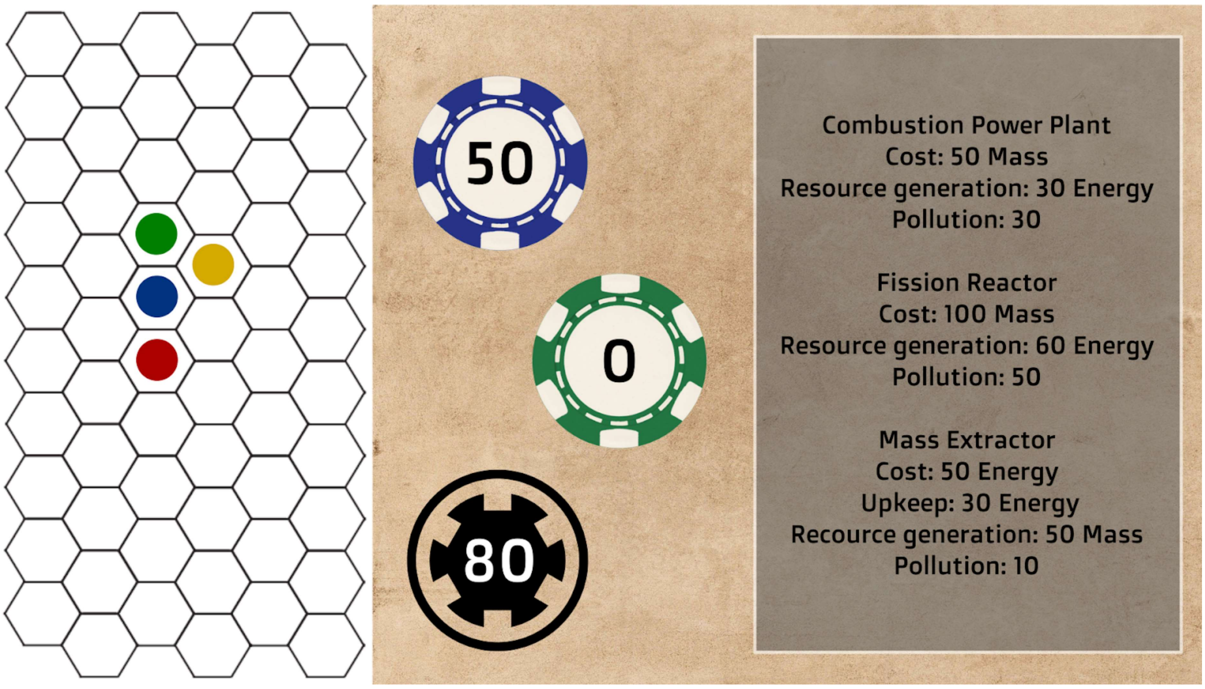
2.1 Prototype Description

First, we printed out the hexagonal grid from our game. We used a flat piece of paper instead of a spherical object, as the underlying shape of the grid structure has only very little impact on the gameplay. For our buildings, we used pieces from other board games such as small figurines and dice. To keep track of resources, money, and pollution we used poker chips.

Then we set up some basic rules that the player needs to follow in order to play the game. Generally the game runs in two phases. The first phase resembles the gameplay of the actual video game: The player uses his resources to construct new buildings or trade for money.

In the second phase, the player needs to calculate by hand what will be calculated automatically in the finished game:

- Pay the energy upkeep for each building
- All buildings generate resources and pollution
- The Player rolls dice to see if any random events happen
- If the pollution exceeds the pollution limit, end the game. Otherwise, return to phase 1



Combustion Power Plant
Cost: 50 Mass
Resource generation: 30 Energy
Pollution: 30

Fission Reactor
Cost: 100 Mass
Resource generation: 60 Energy
Pollution: 50

Mass Extractor
Cost: 50 Energy
Upkeep: 30 Energy
Recource generation: 50 Mass
Pollution: 10

2.2 Prototype Insight

The goal of the prototype is to answer a few core questions and hopefully discover early flaws in the core game design to save time later on.

Is the game fun? Playing the game turn for turn and watching your empire and resources grow is already a rewarding experience, and we think with a few tweaks it can definitely grow into something that is engaging and fun.

There were however a few minor issues we ran into when prototyping our early concept.

One observation regards the exchange of energy for money. In the game, the player can trade energy for money – and gaining as much money as possible is the ultimate goal of the game, as it is based on a highscore system. While playing the prototype we realized that there is no real reason to make this exchange earlier than the last round of the game. This leads to a 1:1 relation between energy and money. To address this issue, we plan to add interest to the players current credits on every round. This incentivises the player to exchange a part of his energy for credits early for a larger return on investment as the rounds go by, introducing an additional strategic element.

Another important observation that we made regards pollution and the pollution limit. In our prototype, every building generates pollution. As soon as the pollution exceeds the pollution limit, the game ends. The method to decide how many turns this takes is a very important part of the design, as it will decide in part how long a gameplay session is. By introducing interest on credits, we create the possibility of the ideal strategy being a minimal base while selling energy each turn to maximize profit via interest. We want to incentivise the player to construct more buildings, instead of maximizing the profit by playing as many turns as possible while earning money purely by exploiting the interest mechanic and doing nothing each turn. We plan to address this by looking into other pollution to turn ratios, iterating on building values, as well as playing around with additional potential "win" condition that give the player a credit bonus. For now it is something to keep in mind while iterating.

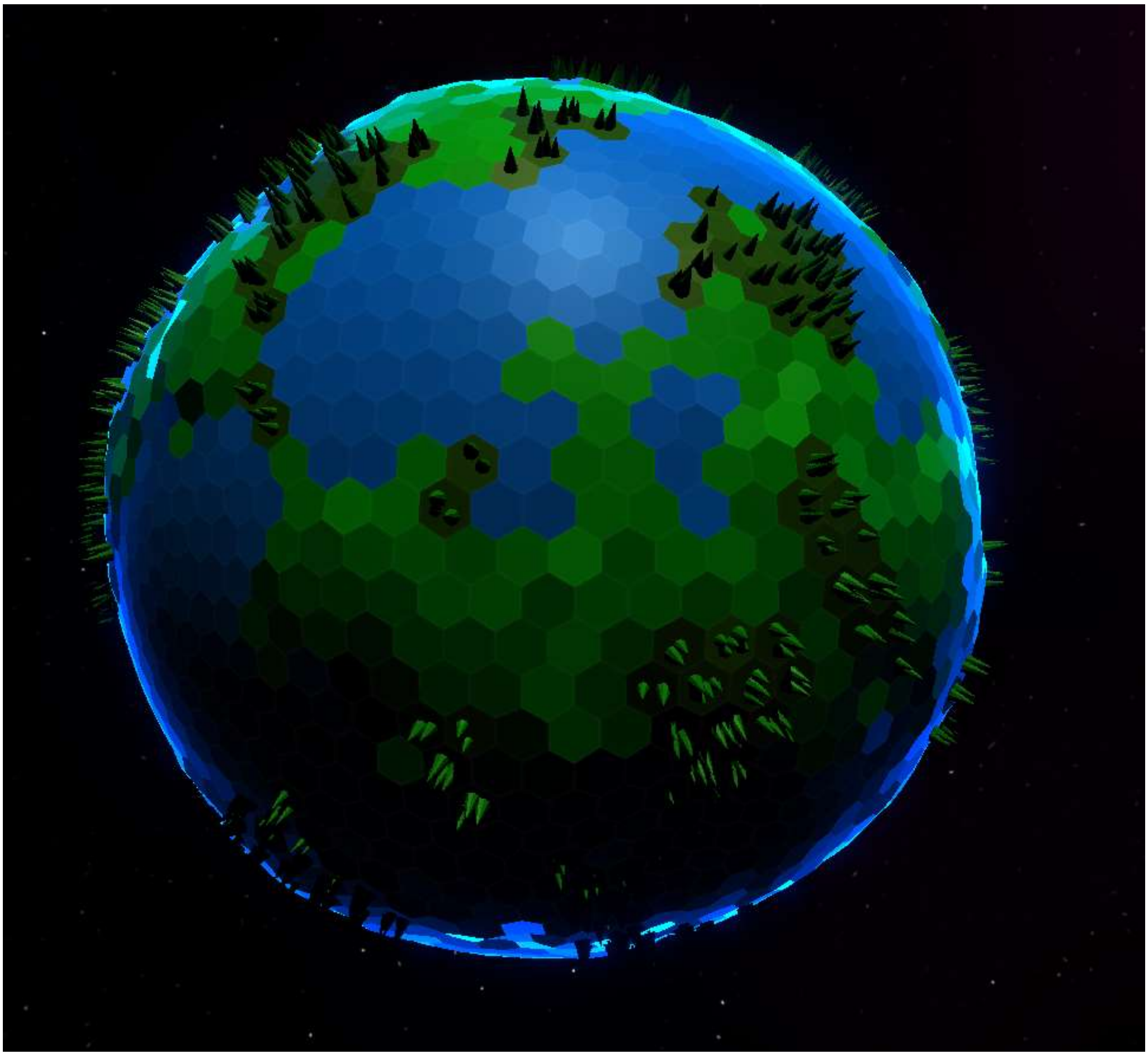
One general observation that also was mentioned in the reviews on our formal game proposal is that we should focus more on strategy and less on RNG-heavy elements. To achieve this, we plan to tune climate events to be more predictable, such as rising water levels or spreading fires. This should make them feel less unforgiving and also add more dynamic gameplay as players can react to the events and plan accordingly. Additionally, we plan to expand on the world generation aspects to create more strategic opportunities. The player must more carefully consider his starting location, how much room he has to build, as well as which tiles are nearby. We also plan to add more building-building interaction, such as buildings that modify the surrounding buildings in some way such as increasing resource generation or reducing pollution production. This way, the player will more carefully have to plan the layout of his buildings, rather than just focusing on the raw statistics provided by each building. This will hopefully lead to more interesting and strategic decision making during gameplay.

During the prototype we did not overly concern ourselves with balancing as we think this is generally easier to do in an alpha development stage anyways. There will be numerous features, like the terrain as well as additional buildings that have a large impact on the balance and can't easily be simulated as a prototype. We plan to adjust the separate values for buildings and terrain iteratively once we are in a position to test the game, as this is rather difficult and tedious with a pure paper prototype.

3. Interim Demo

Good progress has been made, and we are currently roughly halfway through our layer 3 features.

3.1 World Generation

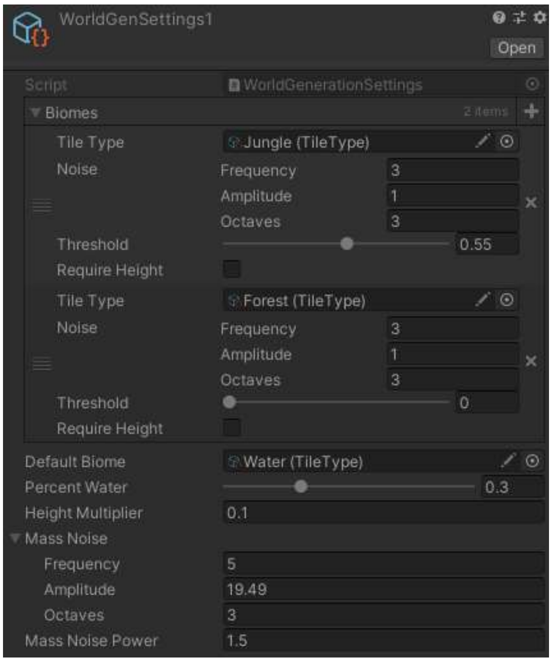


The world's shape is generated by iteratively truncating an icosahedron. Due to the algorithm, we are somewhat restricted in choosing our world size and have settled on 2432 hexagons as a fitting size.

Depending on the seed of the world, a template of world layouts is chosen. These define the biome generation parameters and thereby make it possible to have varied worlds by creating many of these layouts.

Once a layout is chosen, biomes are created using simplex noise and a few simple rules. Each tile type is defined as a scriptable object making it easy to create additional tiles. These scriptable objects also carry additional information, such as which object to spawn on it to make it look nicer. Since we have many tiles and potentially even more objects, the tile objects are handled using Unity ECS.

Additionally, each tile carries a "mass value", which gives a bonus to mass extractors. This intends to make base planning more interesting, and that climate events can change the amount of mass a tile carries. To make it easier to plan their base, the player can visualize these mass values at any time.



WorldGenSettings1

Open

Script WorldGenerationSettings

Biomes 2 items

Tile Type Jungle (TileType)

Noise Frequency 3

Amplitude 1

Octaves 3

Threshold 0.55

Require Height □

Tile Type Forest (TileType)

Noise Frequency 3

Amplitude 1

Octaves 3

Threshold 0

Require Height □

Default Biome Water (TileType)

Percent Water 0.3

Height Multiplier 0.1

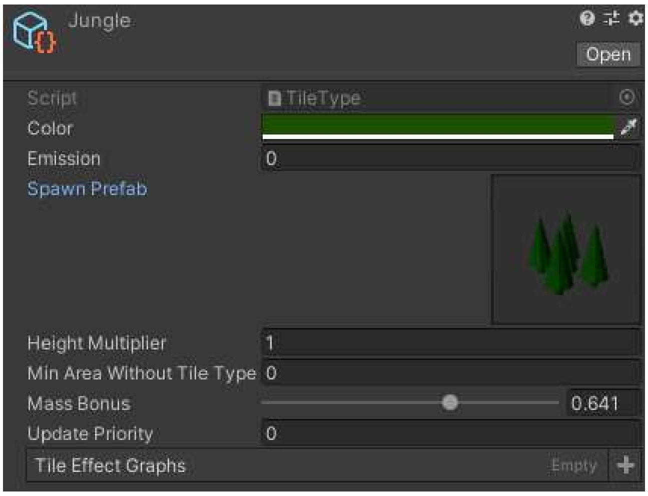
Mass Noise

Frequency 5

Amplitude 19.49

Octaves 3

Mass Noise Power 1.5



Jungle

Open

Script TileType

Color 🟩

Emission 0

Spawn Prefab

Height Multiplier 1

Min Area Without Tile Type 0

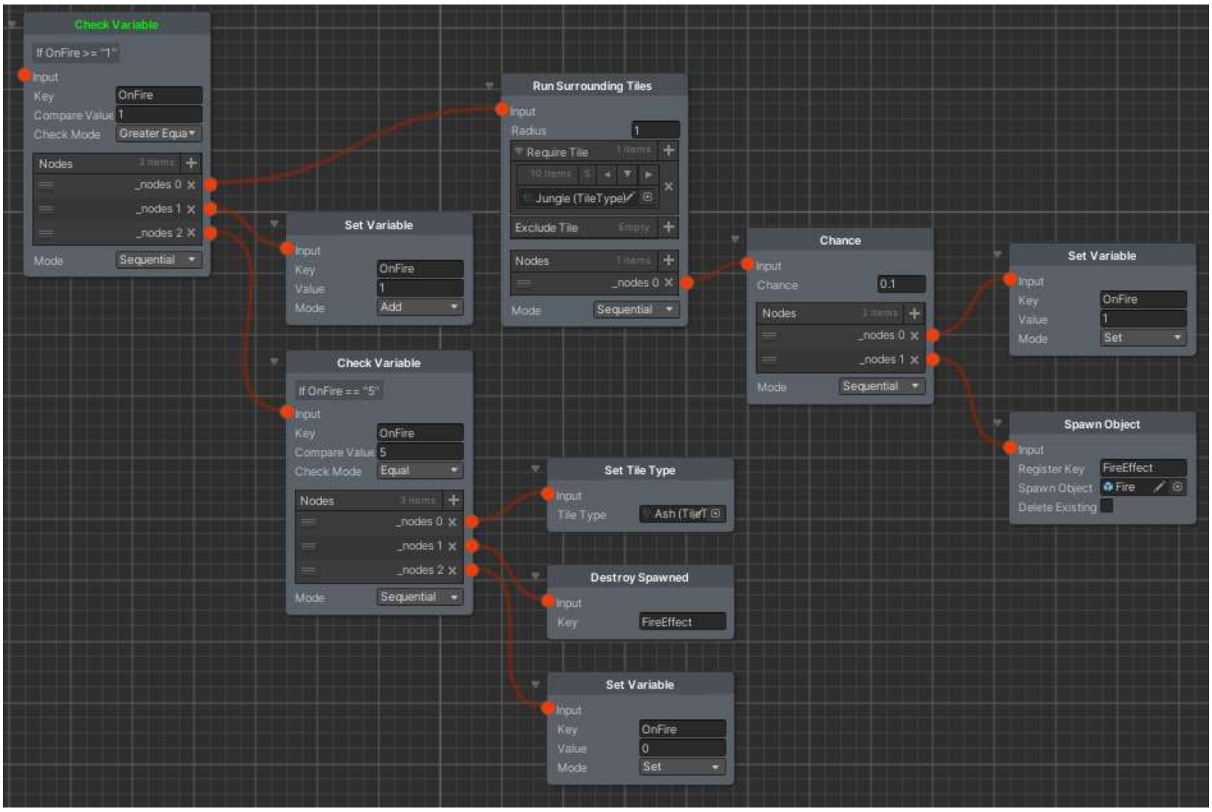
Mass Bonus 0.641

Update Priority 0

Tile Effect Graphs Empty

3.2 Climate Effects

Since climate effects can vary greatly, they are implemented using a graph system. Each graph node is executed sequentially (although concurrent execution is also possible), giving time to pan the camera or show effects. Once a graph for a tile has executed, the next tile will execute its graphs. This gives flexibility and a high-level overview to designing new climate effects. New nodes can quickly be added to enable even more complex effects.



Check Variable
If OnFire >= "1"
Input
Key: OnFire
Compare Value: 1
Check Mode: Greater Equa
Nodes: 3 items
_nodes 0
_nodes 1
_nodes 2
Mode: Sequential

Set Variable
Input
Key: OnFire
Value: 1
Mode: Add

Run Surrounding Tiles
Input
Radius: 1
Require Tile: 1 items
10 items
Jungle (TileType)
Exclude Tile: Empty
Nodes: 1 items
_nodes 0
Mode: Sequential

Check Variable
If OnFire == "5"
Input
Key: OnFire
Compare Value: 5
Check Mode: Equal
Nodes: 3 items
_nodes 0
_nodes 1
_nodes 2
Mode: Sequential

Chance
Input
Chance: 0.1
Nodes: 2 items
_nodes 0
_nodes 1
Mode: Sequential

Set Variable
Input
Key: OnFire
Value: 1
Mode: Set

Spawn Object
Input
Register Key: FireEffect
Spawn Object: Fire
Delete Existing

Set Tile Type
Input
Tile Type: Ash (TileT

Destroy Spawned
Input
Key: FireEffect

Set Variable
Input
Key: OnFire
Value: 0
Mode: Set

Example graph for the spreading fire climate effect

3.3 Resources

Energy, Mass and Pollution are internally all treated as resources (even though pollution probably isn't a resource the player wants to obtain), and it is very easy to add more types of resources to the game. For example, money will likely be added as resource type at a later stage of development, once the implementation of the high score feature takes off.

3.4 Buildings

The buildings are processed one by one after each turn. An update priority ensures that certain buildings are processed first. For example, mass extractors are processed rather early, as the player needs to pay upkeep for them. If the player can't afford the upkeep, the mass extractors do not generate mass in this turn, even though the player's other power plants might generate additional energy afterwards.

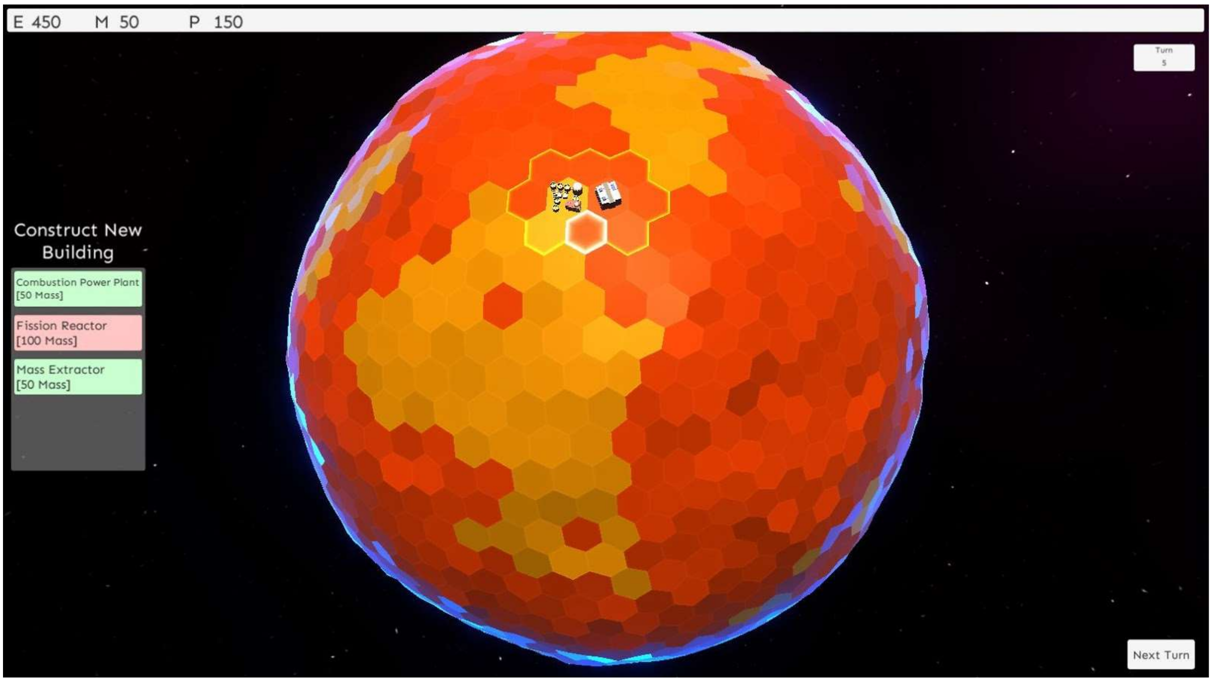
Each building can contain one or several building processors, that generate resources. For example, the combustion power plant has two building processors that generate a flat amount of resources each turn, one that generates energy and one that generates pollution. The mass extractor, on the other hand, uses a slightly more complex resource generator that takes the mass value of the tile the mass extractor resides on into account.

We created the buildings using low poly models from the asset store, as neither of us is proficient in modelling.

3.5 UI

Our UI currently comprises several panels that show information about the currently selected tile and the player's resources. There are buttons to select a starting position and to advance one turn. A sidebar allows the player to construct new buildings or modify existing ones. Starting a new game is currently done using a debug UI, so we need to add some sort of main menu later on.

The art style of the UI is rather simple, as we used Unity's default sprites for buttons and panels.



3.6 Audio

We implemented a basic Audio Manager that allows game objects to play any sound whenever it is necessary. However, we didn't yet decide, which sound effects and background music we want to use.

3.7 Issues

3.7.1 Models

The use of models from the asset store is up to change, as this approach involves several concessions. First of all, it is hard to maintain a coherent art style, as not all types of building are covered within one asset pack, and different asset packs can vary a lot in style.

Additionally, as most buildings are viewed from the top and take up only very little space on the screen, it is rather hard to spot lots of the models' details. Furthermore, the most important aspect of the buildings' visual design is for the player to be able to identify each building within a split second, which becomes harder for more detailed models. Therefore, it is very likely that we will switch back to models that consist of only a few primitives, but that are coloured very diverse.

3.7.2 Turn Processing Order

We have a lot of things on each turn that need to happen sequentially, and in the correct order. Every turn update, buildings need to be processed, which can include tile and resource modifications. Additionally, tiles need to be updated, meaning their climate effects need to be updated. These, in turn, have different effects that need to be played, and may or may not modify underlying tiles or even destroy buildings. Depending on the order these things are executed in, the results may vary. Whether or not this is a problem, we will see in the actual playtesting of a game. If it is, we have systems in place (like the priority ordering of buildings and tiles) to combat this to some extent.

4. Alpha Release

We made a lot of progress, finished layer 3 of our layered development schedule and implemented most of our high target features.

4.1 World Generation

The hexagons of the world mesh are now connected via quads, to properly enable elevated hexagons. Additionally, the water is now animated to subtly move, making it look less static. Each tile now also has a wind direction value that is used in various parts of gameplay.

4.1 High Scores

We added Credits as the fourth resource to our three existing resources Energy, Mass and Pollution. These can be gained by exchanging energy for credits at a 10:1 ratio (for now). Additionally, the player gets 5% interest each turn. This opens up more strategies, such as buying credits early on to profit off of the interest in later turns. A shortage of energy can, however, cause the player to lack the upkeep required for buildings.

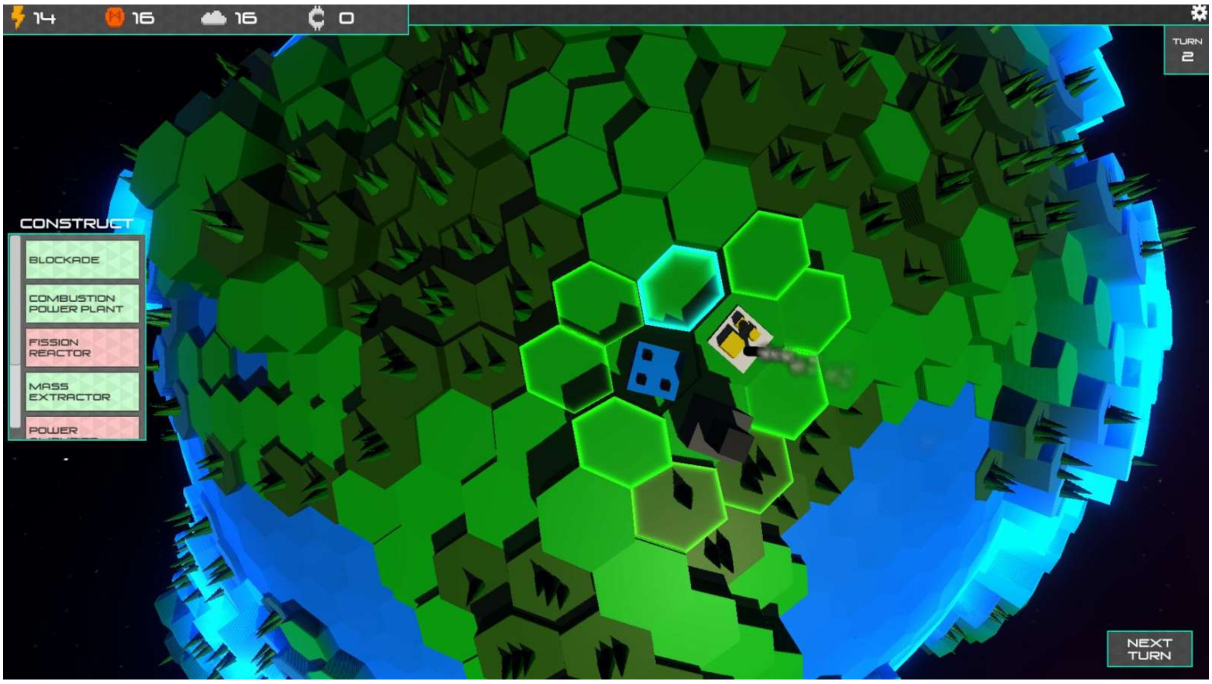
Credits will be the ultimate resource that will determine the player's score. No matter how many buildings were placed or how many turns the player was able to sustain the planet, all that matters is the amount of credits earned, which determines their highscore.

Highscores can be viewed in the menu, where a player can also change their highscore name if they want to.

4.2 Fancier Graphics / Animated Assets

We switched the models of our buildings back to simple primitives. This makes them more easily distinguishable and provides a cleaner look in general. To make the game look less static, we added tweening and animations to building construction. Additionally, some buildings have moving parts, such as the mass extractor, wind turbines or power amplifier.

Our UI now also looks a lot more visually appealing, as we added custom graphics to all UI elements. We implemented a title menu, where the player can immediately start a new game, or decide to set a custom seed first, and a small settings menu, where the player can set the volume of music and sound effects, as well as forfeit early. Many of the UI elements are tweened to make transitions look better.



Gameplay scene showing new UI and some example buildings

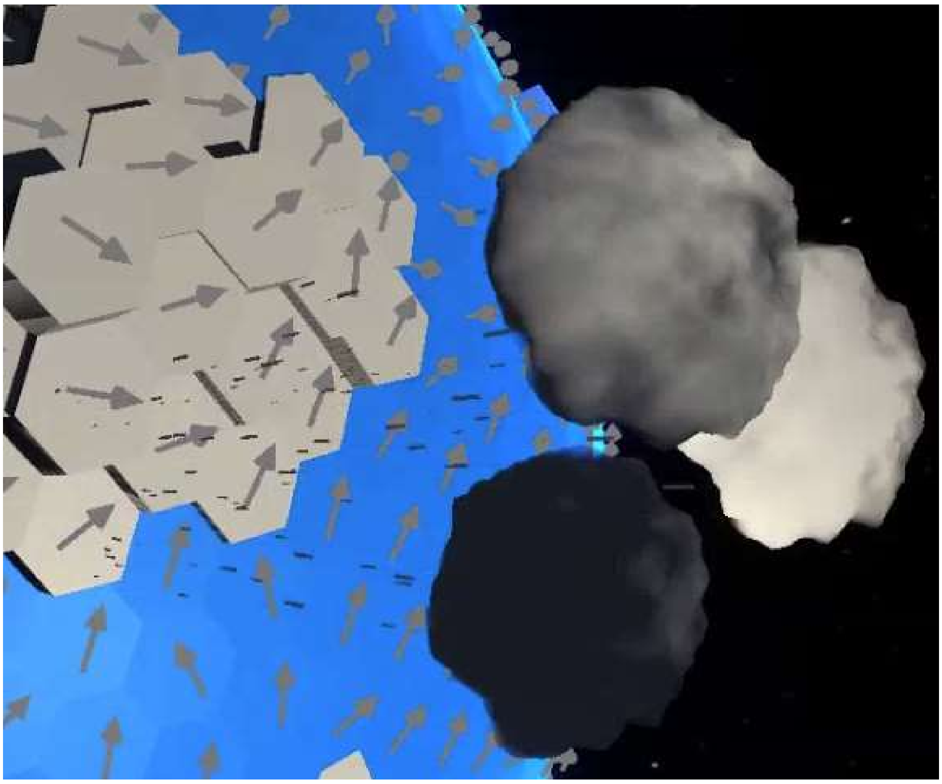
4.3 Quality of Life Improvements

Many quality of life improvements have been added.

- Right-clicking a building in the construction menu gives additional information
- Hovering over a building in the construction menu shows how much mass it costs
- Pressing F3 toggles per-building output UI elements to have a better overview of what is being produced where
- A help menu has been added to the main menu which shows a few tips to the player as well as how some mechanics work

4.4 More Climate Effects

As announced previously, we implemented more climate effects, namely lightning strikes and rain. Rain extinguishes fire and sometimes causes lightning to strike the tile underneath, damaging the building and potentially destroying it. Rain clouds always spread downwind, which can be visualized with F2. Additionally, fire also now only spreads downwind, giving the player a better way to plan ahead. Meteor impacts are now rarer, but deal impact damage and spread more fire. The flood effect was deprecated in favour of raising the water level over time. Tiles that fall below the waterline are converted to water and destroy potential buildings on them if they cannot reside in water.



Rain effect with wind direction visualization enabled

4.5 More Building Types

We implemented the feedback that we got during the presentation of the interim demo, and added streets and bridges that can be used to expand the player's territory. Streets can be placed on most tiles, with liquids being the only exceptions, while bridges can only be placed on water tiles. Furthermore, streets and bridges need to be built on tiles that are adjacent to other streets/bridges, or the player's base. These can be used to quickly expand outward from the initial location to reach another part of the world.

We also added:

- The wind turbine, an energy-generating building with a great energy to pollution ratio, but expensive in mass when built
- The blockade, which blocks the advance of fire and is very durable
- The power amplifier, which boosts the energy generation of the surrounding buildings

4.6 Background Music / Sound Effects

We started the development of a basic audio manager during the interim demo phase. During the last two weeks, we further fleshed out our audio manager and started to actually implement some sound effects and background music. The game starts by playing calm menu music and then smoothly transitions to a randomly selected track once the game has been started.

4.6 Issues

There still are some minor bugs in the game that need to be fixed. The next big step is playtesting the game and balancing all the various factors.