



Permafrost Game Design Document

Design specification Version 2

| | | |
|-----|----------------------------------|---|
| 1 | Table of Contents | |
| 2 | Introduction | 2 |
| 2.1 | Target Audience | 2 |
| 3 | Specification..... | 2 |
| 3.1 | Platform..... | 2 |
| 3.2 | Game Concept | 2 |
| 3.3 | Story..... | 3 |
| 3.4 | Graphics..... | 3 |
| 4 | Game Play | 3 |
| 4.1 | World..... | 3 |
| 4.2 | Interactable World Objects | 3 |
| 4.3 | Colonists..... | 4 |
| 4.4 | Buildings..... | 4 |
| 4.5 | Temperature | 5 |
| 4.6 | Towers..... | 5 |
| 4.7 | Tutorial..... | 5 |
| 4.8 | Balancing..... | 6 |
| 4.9 | Controls | 6 |
| 5 | Development System | 6 |
| 5.1 | Frameworks..... | 6 |
| 6 | Development schedule | 8 |

2 Introduction

This document specifies a design for the gameplay of the game "Permafrost". Which is based on discussions that happened in team meetings. The document will have a focus on gameplay mechanics and details of how we intent to implement the game.

This document is to be used during the design, implementation and testing of Permafrost. This design document allows for design consistency and as a reference as the initial design of the game.

2.1 Target Audience

Due to the game's intended difficulty, the game is aimed at ages 14+. The game will require multiple strategy elements to be managed by the payer. The game will also exhibit non-linear progression, and there are many strategic decisions that the player must make.

Our research shows that more difficulty games are better suited to an older audience, who are better suited to learning the game's mechanics quicker. Permafrost may not provide instant gratification to the player making intelligent and positive decisions. Younger audiences may lack the attention span required to combat the difficulty of this game, and therefore may perceive boredom or uninterest.

We don't believe it is worth sacrificing depth and complexity in our game mechanics in order to attain to younger audiences. We feel an older audience of late teens and young adults would be the most appropriate demographic to target.

3 Specification

3.1 Platform

Permafrost will be produced for x86 based computer systems that support OpenGL graphics. This includes Microsoft Windows 7, 8.1, and 10, Mac OS X, and Linux Operating systems. This offers a large target market that is well suited to the strategy genre.

Our targeted system requirements for the game will be relatively low. The game will be 2D and we are committed to creating a performant program. We intend for this game to be accessible as possible.

3.2 Game Concept

Permafrost is a single player strategy game where a player controls a group of characters with the aim to manage resources to achieve survival.

Our concept is, top down colony management and survival game set in a distant future. The player will be expected to use gameplay mechanics to manage resources within the colony. Progression will be met with increased difficulty with random enemy encounters and changes to weather conditions.

3.3 Story

Permafrost is set in a post-apocalyptic future. In an attempt to save the world from global warming, humanity built a global system of cooling towers. This however, allowed the AI controlling the towers to overthrow humanity by freezing the earth. The game is set after these events in a hostile frozen wasteland, with humankind fighting for survival. From there the emergent story will be the focus as every game decision is up to the player.

The game will start with an introductory cutscene explaining the events leading the present day.

3.4 Graphics

Permafrost will be a 2D top down 3/4 view game. This is sometimes called the "RPG View" and we feel is best suited for this type of strategy game.

The game will feature an overlay UI along the edges of the screen and will support UI elements such as drop-down menus anywhere on the screen. Each tile will be a 64x64 texture with muted colours to fit the game's atmosphere. The game will contain a number of different textures for multiple different terrain types.



Figure 3:1 - Example of art style of tiles

The graphics of the game will be reflective of the game's atmosphere and will give the player the sense of the cold and warm that the colonists encounter in different scenarios. Warmer regions will have warmer colours, Colder areas will have colder and desaturated colours.

Other effects such as vignette, overlaid precipitation particle effects will be used to add to the atmosphere of the game.

3.5 Sound

Our music will have a slow tempo in a low pitch to create a cold and gloomy feel. It will help us create the experience we are aiming to achieve as it will match the setting. During attacks and defence, the music will have a more upbeat battle tempo.

4 Game Play

4.1 World

The world will consist of procedurally generated zones. Each zone is of a fixed size of 50x50 tiles. Each tile will have one of many terrain types. Interactable objects are located in each zone, each with a different probability of appearing in a given zone.

4.2 Interactable World Objects

An Interactable object is an object that exists in the game world and can be interacted with by colonists. Interacting with an object this way will provide a resource such as food.

Some interactable will require a specific item or tool to be interacted with in certain ways.

Examples of some non-moveable interactables are:

- Trees
- Bushes
- Wild grass
- Stashes or storage boxes

Interactables may have multiple options for which actions can be performed. eg. a Tree could be burned, chopped down, tapped, or its fruit harvested.

Interactable objects can also moving. For example:

- Passive animals; rabbits, pigs, sheep, etc.
- Hostile enemies; Polar bears, Robots, Aggressive Humans etc.

Hostile enemies will be encountered frequently in many zones that the player may explorer. Each zone has a chance of containing a number of both passive and hostile creatures.

4.3 Colonists

Colonists interact with interactable objects. Colonists can pathfind and move around the map to reach a goal. They can perform actions and can be queued with multiple and repeating instructions. The player must instruct colonists with instructions. Each instruction will be between a Colonist and an Interactable. Colonists will also have an inventory for storing resources such as food, and equipment such as clothes and tools.

The game will feature several types of colonists, each with their own specialist task.

The three classes of colonist are Hunter, Builder, and Soldier.

4.4 Buildings

Buildings are a type of interactable object, they will provide unique actions such as crafting, heat production, and resource storage. Buildings will be crafted by colonists and require a certain amount of resources to build.

Examples of buildings and their instructions are:

| Building Type | Instructions |
|----------------------|----------------------------------|
| Farm | Sow, Harvest |
| Kitchen / Oven | Cook meal, Butcher Animal |
| Camp fire | Cook meal, Fuel |
| Storage room | Store Resource, Collect Resource |

Excluded from this table are the instructions for Build and Demolish which all buildings will share.

Some buildings may have the same function as others, but perform tasks faster, or require different / less resources. This allows for upgraded building types and colonist progression. Eg Campfire → Fireplace → Oven.

4.5 Temperature

Each Tile in the game world will have a Temperature this temperature will be somewhat random but influenced by terrain and surrounding tiles and zones. Colonists will have a habitable temperature range which humans can survive in. This can be influenced by equipment such as clothes. The food intake of a Colonist will be dependent on their temperature. Colder colonists will require more food. This will be the main limiting mechanic that would prevent endless exploration and requires strategy and planning on the players part to allow for exploration and survival of their colonists.

The player can use buildings to increase the temperature of nearby tiles. Campfires will be one of the first buildings the player can create. Out of all the buildings, Towers will have the most heating abilities, Towers will show a noticeable heating effect in neighbouring zones.

4.6 Towers

Towers play a large role in the overarching goal for the player. Around the game world there will be many zones that contain a cooling tower. This tower will be guarded by AI Robots. The tower can be taken over and claimed by your colonists, converting the tower into a heating tower. It will require significant progression and resources for this to happen and is a high-risk endeavour. Once the tower is taken over by the colonist, the temperature of nearby tiles and neighbouring zones will be affected. Allowing the player to colonise other zones and explore the world further.

We intend to design the game to be played in a number of ways. A player might want to setup a base around a single tower, and then perform short explorations into other zones to gather resources. Or the player could have multiple small bases in different zone to allow for exploration over a further distance. A more nomadic method may be favourable however not necessary. The game will be balanced in a way to give the player options for how they want to play the game.

4.7 Tutorial

The game will contain an objective system which will guide the player and introduce core mechanics and rules of the game. The tutorial is designed to present the core mechanics of

- Controlling colonists and Issuing instructions.
- Resource and colonists need.
- Temperature system.
- Building system.
- Enemy and combat system.
- Zone system and towers.

This objective system will guide the player up into the point of capturing their first tower and building a simple and sustainable base. From then on emergent story will occur where the player is in total control over the direction their colony takes.

Towers do, however, require a constant fuel supply to heat, and the amount of heating will be dependent on the amount of burnable fuel or electricity that can be provided.

4.8 Balancing

The game will start of quite easy and manageable, with the difficulty increasing overtime.

The difficulty of the game will be balanced with a metric of how much progression the player has made in the game. The better the player's colony is performing, the more difficult. To increase the difficulty of the game, the frequency of enemy encounters will be increased. the temperature may fall slightly. This progression metric will contain several factors including the level of equipment colonists have. The amount or value of resources.

This dynamic balancing ensures that the player can recover from periods of decline. However, it is important that fixed forms of balancing occur to ensure that the perceived difficulty increase is not too rapid, and that ultimate goals can be reached whilst still providing a challenge. We also want the methods of progression to be intuitive and after the tutorial, for the player to understand the way that they wish to play the game. Good balancing is required to allow for this.

4.9 Controls

The game will have three main control schemes:

1. Mouse click and drag controls for panning the camera. UI elements for cycling between colonists and issuing instructions.
2. WASD for panning camera, Hotkeys for cycling through colonists. Hotkeys for issuing instructions and to cycle between buildings.

All keyboard keys and mouse behaviour will be customisable through the options menu.

Instructions will be given to colonists through the control schemes mentioned above. UI elements will be used to allow for selection of different actions, colonists, and interactable (buildings, animals, trees etc.)

The intended control experience sees the player utilizing primarily the mouse input options but providing keyboard shortcuts and options for more experienced players who want a quicker method for interacting with the game.

We are not prioritising controller support since we don't believe that it is a good match for gameplay. However it may be a feature that we rereview later on in development.

5 Development System

5.1 Frameworks

The game will be created using MonoGame 3.7.1 Cross Platform GL Framework. This allows the project to be built for any OpenGL supporting operating systems. This strikes a good balance between flexibility of adding our own features and having absolute control over the game's core structure, and the speed of production. Using MonoGame will allow us to produce a high-quality game with novel game mechanics, cross platform access, and a relatively large gameplay scope.

We will be using Tiled TMX format for loading tile sets.

For development process we will initially use the Open Source Myra UI framework. We intend for this to be replaced with our own solution either in the MVP or the Final product.

6 Development schedule

The table below shows the forecast versions of the game and the features we intend for them to include.

| | Version | Key features |
|----------|---------|--|
| The Hack | 0.1 | Tile Map Loading from File |
| | 0.2 | Game object manager |
| | 0.3 | Colonist pathfinding |
| | 0.4 | Colonists Interacting with interactable |
| | 0.5 | UI system |
| | 0.5 | Pathfinding interactives |
| MVP | 1.X | Zone system and map serialisation Procedural world generation |
| | | Finalizing buildings Resource system |
| | | Hostile Creatures and AI Towers Temperature System |
| | | Menu and Save system New UI system |
| | | |
| Final | 2.X | Equipment System |
| | | Zone temperature balancing Combat balancing Resource Balancing |
| | | Frame rate optimisation |
| | | Tutorial Intro cutscene |
| | | |