**Game Design Document.**

**Colony Sim/Management Game.**

By Ryan Hood.

**Contents**

**Introduction:** Page 2.

**Story:** Page 2.

**Characters:** Page 3 - 4.

**Level and Environment Design:** Page 4.

**Gameplay:** Page 4 - 8.

**Art:** Page 8.

**Sound and Music:** Page 8.

**User Interface and Game Controls:** Page 8 - 10.

**Accessibility:** Page 10.

**Conclusion:** Page 10 - 11.

**Appendices:** Page 11 - 12.

**Introduction**

This document will provide a detailed explanation of what will come together to create this game. Using a mixture of diagrams and textual explanation I will discuss the game concept and how it will be implemented.

**The base game idea is a colony sim**, or in broader terms a management sim; wherein the player will **start on a randomly generated map** (with **randomly placed resources**) and they must build various items and buildings within the map **to help their colonists survive**. Their colonists will have a **variety of wants and needs** which must be fulfilled to keep the game moving.

The aim for this game project will be to **tackle random generation of the maps**, with the added difficulty of **having multiple different levels** (layers) in the maps; it will be important to keep the maps consistent, for example not having randomly floating rocks in the upper layers with no supports below it. The game will also have the objective of experimenting with **pathfinding algorithms** determining which will be the best for this game, investigating both the **shortest path and shortest time algorithms**.

For the implementation of the game I will be **using C++ within visual studio**, but more importantly, **I will be using SFML**: a library which allows for **easy image processing** for simple graphical elements, **networking and sound** (among other things). This allows for both **ease of use as well as flexibility**, which is not always available when using other engines (Unreal Engine for example) and will help to **improve the speed of game development**.

**Story**

**This game will not have a story** per se instead it will have **a random event system**. This means that while the player is doing their own thing **at random intervals an event might affect the overall gameplay** for a short while, for example; the player is creating an expansion for one of their buildings when a raid event triggers, which places randomly generated hostile entities at one of the map edges and they will move towards the player’s colony with the goals of killing and injuring the player’s colonists. The raid event will be considered one of the bad events that could happen to the player, but **there will be a mixture of both good and bad events** that could be **randomly selected**.

The aim for this system will be to **allow for the player to create their own stories** for this game, this will not only **increase the replayability** of the game but also give the player a certain level of connection to the game.

**Characters**

The characters for this game will **vary between each playthrough**. The only characters which will be within the game are **the colonists which will be randomly generated** upon the games beginning. The colonists are one of if not the main concepts to the game, **the player must keep them alive and happy** in order for their colony to thrive.

When the game begins and the colonists are generated **they will start off with a few different systems**, to start with they will be given **a random name** (forename and surname) this will help with the personifying of the colonists. Each of these characters will also be **created with a few stats** (characteristics which determine how effective they work at certain tasks) consisting of:

* **Strength** - The character’s raw physical power.
* **Perception** - The character’s awareness of their surroundings.
* **Endurance** - The character’s general resistance.
* **Charisma** - The character’s social awareness.
* **Dexterity** - The character’s ability to manipulate their surroundings.
* **Intelligence** - The character’s intellect the ability to use and create complex systems.

The aim for the characters is to give the player something to connect to, and because the colonists are the main part of the gameplay, it is important that they are complex and interesting enough to keep the player attention. Therefore the colonists will have a certain **level of editing available** allowing for the player before the game begins to make any changes they want for the colonists, like **changing their name**, or **adjusting their stats**.

Each character will also have a few **status conditions** that **the player must manage**. The values are (But not limited to):

* **Health** - The life of the colonist, if it reaches zero the colonist will die.
* **Thirst** - The need for the character to drink.
* **Hunger** - The need for the character to eat.
* **Sleep** - The general tiredness of the character.
* **Stress** - The mental pressure the colonist is under.

The player must **build objects and buildings** within the game world to help them manage all of these values e.g. the player might be able to build a water pump to provide the colony with water, ext. This component to the characters will be what **pushes all of the other gameplay elements**; mainly the **base building** and the **random events**.

**Level and Environment Design**

At the beginning of the game a **map will be randomly constructed upon a simple grid** (probably around 200 x 200 to begin with), where each **cell on the grid will act as a tile in the map**. Each tile can spawn as one of a small selection of types ranging from stone to dirt (these tiles will determine what the cell will look like).

The map will have **a couple different levels** (or layers) which will **go both above and below the ground level**. This will allow for the player to **have a little more control** over the colony, these extra layers can also be built upon.

During the generation process, **the ground level will be the first to be generated** allowing for the upper and lower levels to be **generated with the same topography**. Each map is guaranteed to have at least **a source of water** (being either a river or a lake) **some dirt** (the main floor tile) and **some rock** formations (forming the main bulk of the lower levels and potentially mountains).

The main idea of a randomly generated map is to **add variation to the gameplay**, preventing having the same map twice. This will **help the replayability** of the game allowing for the player to **play multiple playthroughs** on many **different maps** with **different layouts** of buildings, mountains, lakes and rivers.

See **Appendix A** for more info on the basic map generation.

On each of the maps, there will also be a scattering of **resources on the maps**, which will be **building materials**, the main resource which will be spawned will be wood… During the map generation, on each dirt tile, there will be a chance that the tile will have a tree on it. (See the Gameplay section to learn more about the resources). Suffice to say during the map generation there will be resources added into the map, ranging from plants (e.g. trees) to resources inside rock (e.g. iron or gold).

**Gameplay**

This section will cover all of the gameplay elements which will create the game in great detail, explaining **how they work** and the ways they may **interact with other systems** within the game. This section will be split into **two important topics**; the **required elements**, the ones which will need to be implemented to form the game, and a **wish list** which will be completed at the end of the deadline permits it.

**Required Gameplay Elements**

**Base building:** this is one of the key gameplay elements for this type of game. This will involve the player being able to create buildings within the game world using resources from said world. Using the tile/grid system the map is generated with also allows for the easy placement of ‘blocks’ in the world; for example, if the player wishes to create a wooden wall in the map they will place the block onto a tile, thus forming a single wooden wall tile. This system will work for all of the building blocks, if the player wishes to build a wooden floor then they place the tile down where they want and the tile becomes that allowing for complex buildings and structures to be created easily and quickly. However, the furniture which can be placed will work slightly differently, it will be overlayed on top of the tile, while the placement will still work in a similar way (tile locked - preventing the player from placing an object outside or in the middle of a tile), the tile will still be shown below the object. E.g. if a chair would be placed on some dirt then the dirt would be shown below the chair, but both of them would still be displayed, additionally, furniture will not be placeable within walls.

**Random events:** The story element for this game will be delivered through a random event system, this will allow for the players to have unique experience between each playthrough, it will also differ between each player. This will allow for the player to have a special connection to this game, giving them their own story to tell; for example (<https://ludeon.com/forums/index.php?topic=41809.0>), this is just one place and there are many more. In terms of implementation I aim to have a list of possible events and at random intervals one of them will be selected, causing either something good for the player or something bad. It’s all down to a random number generator. For more info on the potential events see the Random Events spreadsheet (<https://docs.google.com/spreadsheets/d/1R_aV6u4QltcNEDkX95QkIbhnMbnDyVBAm5HdpnOkSio/edit?usp=sharing>).

**Colonists:** The colonists will have a few different elements associated with them, the first of them being their skills. The skills (as mentioned above) will be the key to how well they will perform certain tasks. The second component to them will be the tasks that they perform;

* Building. After a player has designated what they want to be to be built, if the colony has the required resources then the colonist will bring those resources over to the object and begin to construct it. Each item will take a different amount of time to build added on the character will have a stat modifier that adjusts the building time.
* Medical. After one of the colonists has been injured then it will be up to one of the other colonists to tend to their wounds.
* Transporting. After a resource has been harvested from somewhere on the map, a character tasked with transporting will pick up the item and bring it to a zone marked on the map. Allowing for all of the player's resources to be stored in one place.
* Mining. This will allow for the player to assign a piece of rock somewhere in the world map and the colonist assigned to this task will break down the rock, collecting the resource and freeing up the tile which was rock previously.
* Harvesting. This is the task of collecting a resource from a plant spawned on the map. This plant could be, for example, a tree, and if the player wants some wood they would assign a harvesting order on a tree and the assigned colonist will cut the tree and collect its wood.
* Hunting (Wish List). This task represents the task of finding and killing an animal on the map and collecting the resources from it (meat and hide). The character will use their equipped weapon on the hunt.
* Taming (Wish List). This task will require one of the colonists to attempt to tame an animal using some food from the colony. If the taming attempt is successful then the animal will become a member of the colony (possibly moving and remaining in a designated zone), from where the player can then harvest different resources from different animals, for example, a chicken will lay eggs for the colony, but they can also be killed at any point for the meat and leather.
* Crafting (Wish List). The player will be able to assign a crafting task (possibly at a specific workbench) and the assigned colonist will then take the required resources and construct the item. The item could possibly be a weapon…
* Cooking (Wish List). This will allow for the assigned colonist to process raw food like meat and create a more edible food for the colony.

A big component for the colonist will be the way that they move around the colony, therefore a certain level of consideration for their movement must be taken. Therefore during implementation, I will be testing and comparing the way in which a path finding algorithm finds the path. After some research into pathfinding algorithms (<https://en.wikipedia.org/wiki/Pathfinding>) I have decided that A\* will most likely be the best for this type of game, with an added consideration for the heuristics that will be used testing both the shortest path (the smallest number of notes from the start to the end) and shortest time (the shortest estimated time from start to end).

**Additional Gameplay Elements (Wish List)**

**Ranged Combat:** This will be the initial form of combat within the game, where two or more colonists will fire their weapons towards each other with the intent to injure. Each weapon will have three main stats as well as a classification. The first being its power, the damage to the colonist if hit by the projectile. The second The weapon’s aiming time/fire rate, in other words how long it will take to fire the next projectile. The third, the weapon’s weight, so if the weapon is heavy it will reduce the colonist's movement speed so the player may have to balance a weapons damage over its weight. And finally the classification, there will be three separate classifications;

* Close Range - Covers weapons which require the character to get far closer to the opponent to be able to deal damage.
* Long Range - Covers weapons which are able to fire from long distances allowing for reach.
* Automatic - Covers weapons of both long and short range with the added bonus of firing a volley of around three projectiles each trigger.

**Melee Combat:** This will work in a similar way to ranged weapons, they will have the same stats, but have different classifications. The main difference between ranged and melee is that for this form of combat, the colonist must be in an adjacent tile to their enemy. The different classifications will affect the type of damage they do, they are;

* Slashing - A blade that will cut the opponent.
* Bashing - A blunt object that will bludgeon the opponent.
* Stabbing - A sharp object that will pierce the opponent.

**Animals:** An additional feature that I would like to include is animals, initially starting off with small animals, such as birds but later to include larger animals. Animals like all of the other resources will spawn randomly on the map and will have two main collectable resources; their meat and hide. These will be used to provide an easy source of food for the colony, but they will be tamable to allow for a slow income of additional resources.

**Zoning:** This will allow for the player to choose a section of tiles in the grid and assign what type of resources will be stored there. This will allow for the player to easily keep track of their resources. I aim to add additional zones into the game, allowing for better managing of colonists and colony resources. For example, an animal zone which will prevent the animals assigned to it to leave the zone, and possibly another zone which limits colonists from leaving a certain zone.

**Crafting Items:** I also aim to add in a basic crafting system into the game wherein the player will create an object in the map and it will give them access to recipes to create, this could be a mixture of food, weapons and various other things. The crafting will include the assignment of the item, the colonist collecting the resources required to build that item and the colonist spending time constructing the item.

**Art**

For this game, I aim to use a rather **simple art style**. I will use **basic blocky objects** for the furniture; using room plans for inspiration. These room plans have made **use of blocks and circles to effectively convey the information** on the page. For example, a bed is a rectangle with a rounded rectangle towards the head of the bed.

I aim to implement tiles for each of the grid cells, this would also mean that I require some **textured art for each of the tiles**, at the moment I aim to implement **three simple tiles** into the game: **water**, **dirt** and **rock**. I plan to create three similar tiles for each of the sections **allowing for slight variation**.

For the colonists I plan to **start off with white boxes** for **testing purposes**, but if time permits I would like to implement **more detailed pixel art** using **games like prison architect and rimworld** as a baseline for the game.

**Sound and Music**

The music in games is **extremely important**, it is used to **convey a certain feeling** while in the game. The main feeling that I want to convey whilst the player is in the game, is **a sense of calm**. The reasoning for this is that I aim for this game to be **a relaxing experience**, yes there may be some situations which might become a little stressful but this **game should ultimately be calm**.

**User Interface and Game Controls**

When it comes to UI it is important to make it both **easy to understand** and use. Therefore I plan to use **a rather simple overlay** for my UI. It will contain **large text** as well as **an image** to convey the action the button will be used for; for example, a button to assign a colonist to cut a tree will have an image of an axe on it to signify the act of cutting a tree.

I aim to include **three different sizes for the UI**, a **small**, **medium** and **large**; this will allow for the user to **adjust the user interface** to better fit with both their needs and their setup. This will require a lot of configuration to ensure all of the data is accurately displayed.

For the implementation of the UI, I aim to create **four main buttons** which will hang at **the bottom of the screen**; these buttons will have **the function of opening a list** of more buttons with a function associated with the pressed button. For example, **the Actions button will have a list of buttons for the different actions** available to the player.

The button categories will be:

* **Actions** - A list of actions the player can assign like chop wood, mine rock, ext.
* **Construction** - A list of buildable objects for the player to place on the map.
* **Colonists** - An ordered list of all of the current colonists with data relating to them, like their current task and their stats.
* **Zones** - Giving the player the ability to create, edit and delete zones within the map at will.

To find more information see **Appendix D**.

The controls for the game will be fairly simple.

To move around the map will be (WASD) **W to move up, A to move left, S to move down and D to move right**. An **alternate set** of these will be **the arrow keys**. Towards the bottom right of most keyboards. I shall also allow for the mouse to move around the map as well, so **if the mouse touches one of the edges of the window, the view will move in that direction**.

There will be **a zoom feature** in the game **allowing for the maps focus to be adjusted** during gameplay, this will be **controlled using the mouse wheel**, rolling it forward zooms in focusing on a smaller section of the map, but making the items in frame bigger, while rolling it back does the opposite.

Due to the map being a multi-level map, this will require **a way to traverse up and down the layers**, as well as having **a button in the UI** the **Page Up and Page down buttons will also work to change the current layer** on the map.

Most actions within the game will be controlled with **the LMB (left mouse button) it will be used to activate buttons select items and place objects**. The LBM will be the main control item for the game.

**Accessibility**

I aim to make the game **as accessible as possible**, this will involve me including **a number of features** into the game to achieve that. While browsing methods of increasing accessibility I found (<http://gameaccessibilityguidelines.com/>), this provided details **to improve accessibility**, here are a few options that can easily be developed into the game;

* To begin with, I will have **an options menu** which will remember the settings the player have selected upon startup.
* Next, I will include **an option to rebind the keys** to anything on the keyboard, as well as having **a number of keyboard shortcuts**.
* I will make the game **playable in both fullscreen and windowed**, allowing for the use of an onscreen keyboard.
* I will also **allow for the UI (user interface) to be resized**, primarily having three size options small, medium and large each with increasing font size allowing for people to easily see the UI.
* I will follow **a colour scheme which is not too garish**, ensuring similar colour are not overlayed on top of each other. Following **if I have additional time I will create a color blind mode**.
* I will **not use sound or coloured text to convey important information** within the game.
* I will have a selection of **multiple different difficulties** for this game ranging from easy to hard.
* I aim to add in **a system to both speed up and slow down the gameplay** allowing for people to take their time whilst playing.
* I will have **a controls menu within the options screen**, allowing for people to easily refresh on the controls.

**Conclusion**

And this concludes the game design document. This document covers the entire project, spanning the main gameplay items including a simple wish list for items I ideally want to add. I have also covered some additional considerations for this project including User Interface designs and accessibility items I aim to include for the game.

Following this document will be some appendices they will expand upon some items touched upon within this document and allow more information on items if required.

**Appendices**

Appendix A - Diagram showing the map and how the multiple levels will look for the player. It is a simple drawing consisting of straight lines and basic connotations on each square of the grid separating the different components of the map. The map shows only the minimalist detail showing the general idea of the generation without the inclusion of resources or player made buildings in the world.

Appendix B - Simple diagram of a potential building which will be constructible within the game. It shows a building with multiple walls, doors a couple windows, and some stairs. One of the main concepts for this diagram is to also propose how the multiple layers of the map should affect the options for building, as shown by the diagram there are three layers, both continuing the building from the ground level.

Appendix C - Is a collection of various mood boards which will serve to act as inspiration for the art style for the game. The final art style for the game is currently not decided and during the implementation process, the games art will be created as it is needed. There are three boards; furniture, character and texture.

Appendix D - An interface design for the game. It has multiple levels showing the hierarchy of menus within the game. This will allow for the game UI to be easily implemented. This is an initial draft and is subject to change upon a moments notice.

Appendix E - the project plan for the project, this also includes the Gantt chart linked within the document. It covers the main milestones and the implementation strategy for the project.

Appendix F - Bibliography for the project, It contains a small list of most if not all of the resources which I have used to create this document and any additional documents created for this project.

Appendix G - Project contract, a confirmation of the deliverables and objectives for this project, signed by both me (Ryan Hood) and my supervisor (Shengxiang Yang).

Appendix H - Ethical review form, an evaluation of my project to ensure it is an ethical project.

Appendix I - A global checklist, this will ensure my project follows De Montfort University’s global guidelines.