GSN Challenge

Final Project Report

Preliminary Requirements, Design, Implementation & Testing

Clariant

CSC 492 Team 38

Sam Adkins

Madeleine Ewart

Dev Patel

Tomas Robinson

Kennedy Tate

North Carolina State University

Department of Computer Science

April 23, 2024

# Executive Summary

*Author(s): Dev Patel*

*Reviewer(s)/Editor(s): Kennedy Tate, Sam Adkins*

Our sponsoring company is Clariant, which is a specialty chemical company that produces chemicals for a variety of industries such as agriculture and construction. Additionally, Clariant is known as a chemical industry leader towards achieving sustainability, and the company places pride in their sustainability efforts.

The problem presented to the team by Clariant is that the company wants the team to create a fun and engaging game for Clariant employees and visitors to play that allows them to learn about environmental sustainability along with Clariant’s goals and actions toward achieving sustainability. In addition, the company wants the game to allow players to input new sustainability actions that they can think of as a way of helping the company potentially introduce new solutions for achieving sustainability.

To address this problem, our team has decided to create a game in Unity which will play as a city simulator-like game, where the player plays as Clariant and has to make decisions and manage various resources throughout the game in order to try to get the best score possible by the end of the game, and this game will ultimately help increase awareness on environmental sustainability along with contributing to introducing new potential solutions that could be used to achieve environmental sustainability.

During iteration 0 of the project, the team focused primarily on obtaining info on the project from the sponsor and creating requirements for the game based on the project info given from the sponsor. With iteration 1, the team revised the project requirements and started creating the startup page and game map in Unity along with starting to implement the turn system. In iteration 2, the team has worked on creating art assets for the game and updating the game map with the new assets along with designing the main tile menu that appears when the player clicks a tile. During iteration 3, the team worked on implementing the main tile and sustainable actions menus and creating/integrating the configuration file into the game along with starting to implement the scoring system with the implemented actions. Finally, in iteration 4, the team updated the game map and tile menu/sustainable actions menu UI along with finishing the implementation of the scoring system.

The main requirements for the project revolved around gameplay, mainly detailing the tile functionality in the game along with describing how various gameplay systems like the turn and choice menu systems will work in the game. The functional requirements for the project are focused on the login and user creation functionality that serve as stretch goals for the project. For the design of the project, the team has created some wireframes that illustrate what the player will be able to see upon clicking on various buildings in the game along with a wireframe representing the planned game HUD UI for the game. Additionally, the team has included component diagrams detailing how the frontend and backend systems will work together along with how the game objects will be organized in the code. Finally, the team included some screenshots showing features implemented by the team in Unity such as the startup page and game map along with showing the art assets created by the team. For testing the project, the team has written 10 system tests focused on the gameplay requirements, with 5 of them currently passing while the other 5 will likely be finished in the future by other teams.

# Project Description

*Author(s): Sam Adkins*

*Reviewer(s)/Editor(s): Dev Patel, Kennedy Tate, Madeleine Ewart*

## **Sponsor Background**

Clariant is a multinational specialty chemical company based in Muttenz, Switzerland. They produce chemicals for many industries including adsorbents, adhesives and sealants, agriculture and animal feed, automotive and transportation, aviation, building and construction, catalysts, chemical intermediates, coatings, paints and inks, electronics, and many more. They also pride themselves in their sustainability, and are seen as a leader in the transition of the chemical industry towards sustainability by taking many steps towards a more sustainable future such as converting their power plants to solar and their gas-powered equipment to electric. They have been included in the Dow Jones Sustainability Index since 2013, and are a part of several notable initiatives that work towards sustainability, such as the Alliance to End Plastic Waste, Together for Sustainability, and the Renewable Carbon Initiative.

### Sponsor Information

Amol Kirtikar, Innovation & Sustainability Manager at Clariant

Email: [amol@kirtikar.com](mailto:amol@kirtikar.com)

Mailing Address: 500 E Morehead St #400, Charlotte, NC 28202

## **Problem Description**

People may have a general idea on what environmental sustainability is, but the majority of people aren’t fully aware of the massive impact actions have on environmental sustainability. For example, a pizza company making pizza in a wood-fired oven may not look so bad as the environmental impact with such an action only really seems to be mainly the emissions that come from using the oven. However, there are other environmental impacts that would need to be considered such as the trees that had to be cut to make land for the pizza company building or the emissions that can come when transporting the pizza to various stores through delivery trucks or similar services. Overall, most people don’t really think about the environmental impact daily actions such as these can have, and Clariant therefore wants to get people to try to start thinking more about the impact of their actions on environmental sustainability. Clariant’s dedication to furthering sustainability has led them to want a new way to engage their audiences in the conversation. They want to gamify the learning process, and provide a way for their employees and visitors to learn about Clariant’s goals and future plans in an engaging way that doesn’t feel like the game is trying to blame or preach to the player.

## **Proposed Solution & Project Goals/Benefits**

To achieve such a game, our team has decided to use Unity to build a city-simulator-esque game. The player is in control of Clariant, and is tasked with learning about Clariant’s forays into making their company model more sustainable. In addition, the player should be able to expand their understanding of how a company can be made more sustainable, and be encouraged to think outside the box in innovating their own solutions to this fundamental issue.

The major goals of the project are creating an educational game on environmental sustainability along with allowing players to learn about Clariant’s efforts towards achieving environmental sustainability. By creating an educational game on environmental sustainability, it’ll make learning about environmental sustainability insightful and engaging for people playing the game along with providing the opportunity for players to introduce new potential solutions for achieving environmental sustainability.

# Resources Needed

*Author(s): Tomas Robinson*

*Reviewer(s)/Editor(s): Sam Adkins, Kennedy Tate*

The main resources that the team will be using for the project are shown in Table 1:

| Resource Name | Purpose of Resource | Status | Version | Licensing Information |
| --- | --- | --- | --- | --- |
| Unity | To generate the gameplay framework | Access Obtained | 2023.2.7f1 | Following Unity licenses |
| Art Assets | To make the game more interactive for players | Access Obtained | v1.0 | Created by team / Creative Commons |
| Clariant support resources | To make the game more factual and realistic | Access Obtained | N/A | Clariant disclosure policies and TOS |

Table 1: Project Resources

Unity will be the game engine we use for producing the game, and Unity has some free art assets that can be used for designing the objects and game map. Furthermore, the team is planning on looking for free art assets from other free game art assets along with making some custom art assets for the game. Resources that will be provided by Clariant include info regarding Clariant’s various actions towards sustainability and their impact on factors such as the three types of emissions along with some artwork to use in the game.

# Risks & Risk Mitigation

*Author(s): Tomas Robinson*

*Reviewer(s)/Editor(s): Dev Patel, Kennedy Tate*

A few risks for the project include:

* Legal issues regarding art assets that are found online which would lead to copyright infringements
* Scope creep and changing game requirements
* To mitigate the first risk, the team will primarily look for art assets from free game art websites and/or create their own art assets to avoid any legal issues regarding the development of the game or mechanism for approving the artwork for use in the game
* To mitigate the second risk, the team will maintain constant communication with Clariant to get feedback on the planned gameplay and ensure the requests from Clariant are kept tracked of and applied to the game correctly

# Development Methodology

*Author(s): Madeleine Ewart*

*Reviewer(s)/Editor(s): Tomas Robinson, Kennedy Tate*

Our design methodology is based on iterations. Most iterations last about 2-3 weeks, but a few are longer due to the complexity of the tasks they contain. The first iteration consists mostly of project setup tasks while the last iteration consists mostly of tasks focused on finalizing/polishing the project, with the middle iterations primarily focused on the development and testing of specific features in the game.

Sprints are essentially shorter iterations that focus more on implementing smaller but more in-depth tasks. Because of this, the team chose to have iterations in the form of sprints for this project because the requirements consist of behaviors that build well on each other along with having a relatively consistent length for each iteration. With the iteration development design, core gameplay features such as the game map and tile system were implemented within iteration 0 and 1, while features that build on them like the solar/coal plants and sustainable actions menu system were implemented in later iterations like iteration 2 or 3. The team also wants to create prototypes that are as playable as possible by the end of each iteration to ensure that features are complete before moving on to other features that may or may not build on earlier ones. For this purpose, an appendix was included with screenshots showing the initial planning of the project along with the initial designs and prototypes of certain functionalities in the game such as the startup page and game map.

Iterations will also reduce the risks of the project by dividing the implementation of the core gameplay mechanics and making sure they get the proper focus and time needed to get them working properly. Additionally, planning out the sequence of features to implement will help make the team more aware of the project timeline and deadlines. Furthermore, this helps make the team less likely to leave features unfinished before moving onto other features, making the final product more well-implemented and polished.

During each iteration, the team will test the features implemented in each iteration. Since game development is different from traditional software development, the team will focus more on system testing and usability testing rather than unit testing. With this procedure, each feature will be fully tested by the end of each iteration to ensure that it works correctly and that other features can be effectively built on top of it.

The team is meeting with their sponsor weekly to discuss the team’s progress, ask questions, and get feedback on the progress made on the project. The team discussed their development methodology and iteration timeline with the sponsor early in the semester along with the teaching staff in a task planning meeting, which helped the team solidify their iteration timeline.

The team is using GitHub for version control because even though Unity has its own version control feature built on top of Git, it requires payment for five users, and Git is sufficient for the needs of the project. Additionally, the team is using Fork, which is a Git client that simplifies interacting with Git by allowing users to easily see changes in individual lines and discard changes when necessary. The ability to discard changes is important because Unity sometimes makes small changes to files when the project is opened, so this helps make it easy to switch branches quickly without worrying about committing insignificant changes or discarding them in a more complicated way.

The team is using Discord as the primary team communication channel, where the team discusses the progress on their project planning, implementation, and documentation along with meeting times and anything else relevant to the project. Bugs will be communicated in the Discord, but they will also be added to GitHub issues, which will allow the team to have conversations about the bugs, but also to create more official documentation on all bugs encountered during the development of the project.

# System Requirements

*Author(s): Kennedy Tate, Dev Patel, Madeleine Ewart*

*Reviewer(s)/Editor(s): Tomas Robinson, Sam Adkins*

**1 Requirements**

### 1.1 Overall View

Our project is an isometric, turn-based company environmental simulator game. Because of the nature of the project, the requirements have been split into functional requirements that describe non-gameplay elements of the game, gameplay requirements that describe the gameplay elements of the game, and constraints that describe necessary features that fall outside the other two categories. The functional requirements have four different sections, with the first section focusing on the login page and going over the login process that will be implemented as a stretch goal for the project. The second section is the action log, which is another stretch goal for the project and will be a system that keeps track of each action the player takes and be used to display a summary of their actions at certain points in the game. The third section is the user creation system, which is another stretch goal that will interface with the login system. The final section is the startup page that the user will see upon starting the game and will include various elements such as the game title, login system, and the user creation interface as well as a button that starts the game.

As for the gameplay requirements, that section is split up into seven different sections. The first section is the actors section, which talks about the player actor and the organization actor that, as a stretch goal, will send the player letters that affect their score depending on their performance. The second section is the goals section that goes over the goals of the player during gameplay, which includes the reduction of emissions and costs at Clariant and also details how the player’s score will be tracked. The third section is the rules section, which shows how the player will interact with the game world. The fourth section is the tile section, which details how the tile system works. For instance, the player is able to hover over a tile with their mouse to see information about the tile and they are able to click on that tile to be presented with a menu of choices for that tile. However, the player must have an action point left in their turn to do any of the actions that are presented in the menu for a tile. The fifth section is the turns section where the details of the turn system are discussed. With the turn system, the player is given a certain number of actions each turn that they can use to move or convert tiles as well as other special actions such as buying renewable energy certificates or power purchase agreements. Once the player runs out of action points on their turn the turn advances to the next, their action points are restored and the in-game time is advanced by one month. Once the player reaches a certain time limit, which will be the year 2050, the game is over. The sixth section is the menu of choices which details the menu of choices system. If the player clicks on a tile or the Clariant building they are presented with a menu of choices that show the actions they can perform with that tile, and they are then able to choose one of those actions from the menu as long as they have an action point, but the Clariant building is a special case where the player can see info on the overall score stats across all of the tiles on the game map. The seventh and final gameplay section is the leaderboard and scoring section, which goes over the details of how the player’s scores are calculated and presented at the end of the game along with having an interactive leaderboard which is currently a stretch goal for the project.

The final section in our requirements is the constraints section which shows the features that fall outside of the scopes of functional and gameplay requirements. The first constraint is that we have to make the game accessible without needing to download any game files, which will be done through web hosting, and the final constraint is that the game’s playtime should be between fifteen and thirty minutes.

### 1.2 Functional Requirements

**FR1 Login - *Stretch Goal (Not Implemented)***

* **FR1.1:** The player shall enter their username and password in order to play the game, and the username and password must match the username and password registered in the player’s account
* **FR1.2:** Once the player submits their username and password, their previously saved game state shall be available, if they have played before
* **FR1.3:** If the player does not have a registered account, they shall still be able to play the game as a guest
* **FR1.4:** The login system shall authenticate users securely using Clariant’s authentication system

**FR2 Action Log - *Stretch Goal (Partially Implemented)***

* **FR2.1:** User actions shall be saved in a log for every action that is taken
* **FR2.2:** Log shall be seen visible to the user at the end of each month
* **FR2.3:** Log shall be seen visible to the user at the end of the game
* **FR2.4:** Player shall have a notepad that they can write their thoughts or plan of actions if they wish to do so
* **FR2.5:** The log shall be used to calculate statistics about common player actions

**FR3 User Creation - *Stretch Goal (Not Implemented)***

* **FR3.1:** Users shall be able to register a new username in the system
* **FR3.2:** Users shall be able to register a new password in the system
* **FR3.3:** The username shall be unique to the user

**FR4 Startup Page**

* **FR4.1**: Upon loading the game page, the player shall be presented with a startup page
* **FR4.2**: The startup page shall contain the title of the game and an option to start the game
* **FR4.3**: The startup page shall allow the player to create and log into their user account [Stretch Goal, Not Implemented]

### 1.3 Gameplay Requirements

**GR1 Actors**

* **GR1.1:** The player character is an employee who is leading the environmental initiative at Clariant
* **GR1.2:** Organizations like the EPA will send letters to the player character with bonuses/sanctions based on their performance [Stretch Goal, Not Implemented]

**GR2 Goals**

* **GR2.1:** The goal of the game is to implement and update Clariant’s technologies to reduce emissions and costs
* **GR2.2:** The player’s score will be tracked by their actions made in game
* **GR2.3:** The player’s score is based on their emissions, investments, operating costs, product cost, and product carbon footprint (PCF)
* **GR2.4:** The amount of time (in number of turns) it takes the player to reach certain milestones will also be tracked and displayed

**GR3 Rules**

* **GR3.1:** The player will interact with the game world to select options such as converting and moving tiles
* **GR3.2:** Each action will change the player’s emissions, investments, operating costs, product cost, and/or product carbon footprint (PCF)

**GR4 Tiles**

* **GR4.1:** When a tile is clicked, the player is presented with information about the emissions and operational costs from that tile
* **GR4.2**: Clicking on the Clariant Building displays information on the overall investment, operating cost, and different emission values spread across all the tiles

**GR5 Turns**

* **GR5.1:** The game advances to the next turn once the player runs out of action points [Stretch Goal, Not Implemented]
* **GR5.2:** Each turn will advance time by a set amount of time (month)
* **GR5.3:** The player is given a number of actions (action points) each turn that they can use to upgrade or develop tiles [Stretch Goal, Not Implemented]
* **GR5.4:** Once the turn is over, the player’s action points are restored and time advances in the game [Stretch Goal, Not Implemented]

**GR6 Menu Of Choices**

* **GR6.1:** When the player selects the “Sustainable Actions” option in a tile’s main menu, the player will be given a list of options including switching the tile with a different one, replacing the tile with a purchase power agreement, buying a renewable energy certificate, and moving the tile
* **GR6.2:** When the player chooses an action from the menu, information about the investment cost, operating cost, and emission impact of the action shall appear

**GR7 Leaderboard and Scoring**

* **GR7.1:** A player’s scores are calculated based on their emissions, investments, operational costs, product cost, and product carbon footprint (PCF) from the game
* **GR7.2:** Lower scores represent better player performance
* **GR7.3:** Player scores will be compared in an interactive leaderboard [Stretch Goal, Not Implemented]
* **GR7.3:** Player scores will be displayed at the end of gameplay [Stretch Goal, Partially Implemented]

### 1.4 Constraints

* **C1:** Users must be able to play the game without downloading any files
* **C2:** The game must have a playtime of fifteen to thirty minutes for most players

# Design

*Author(s): Madeleine Ewart, Kennedy Tate*

*Reviewer(s)/Editor(s): Tomas Robinson, Dev Patel, Sam Adkins*

## **High-Level Design**



The component diagram (Fig 1) shows the frontend and backend systems of the game. Most of the backend systems have a frontend UI display, represented by the blue arrows. The configuration system is strictly backend because it represents a way of simplifying game updates. The game map and scene change system both deal with game scenes, and are not directly connected with the UI.

**Game Map**

****

The game is built on top of a game map, which contains tiles (Fig 2). The game map is part of the main game display, and the UI will be rendered on top of it.

**Tiles**

****

The following sprites (Fig 3) were created by the team to act as temporary placeholders for the isometric background. They are used to represent the required aspects of the game as described by the original design requirements given by Clariant. However, some of the sprites were changed and/or not used in the game since the sponsor wanted more photorealistic versions of these aspects in the game.

**Turn System**

The current turn system is based upon a system of “action points” in which the player gets to make a certain number of decisions in a fixed amount of time. Upon using the final number of action points, the counter resets and time progresses in the game, giving the player more time in a later year to make changes to the company.

**Letter System**

The letter system will trigger when the turn system reaches a certain time. At that time, a letter or other message will appear on the screen and give the player relevant information. The letter system was a stretch goal and was not implemented.

**Menu System**

The menu system is one that is built into the game tiles. This allows the player to click on specific tiles and make decisions based specifically on the type of tile that is selected. When the player finishes making a decision, the menu disappears out of view for convenience.

**Configuration System**

The configuration system will allow parts of the game to be changed via configuration files, specifically text describing Clariant’s real life environmental contributions, images, and anything else relevant. The purpose of the configuration files is to create a simple way for Clariant to edit certain aspects of the game after the completion of the project, without the need to download and use the Unity editor.

**Logging System**

In order for Clariant to get useful information from the gameplay of employees and visitors, the logging system must be detailed regarding the choices made by players and the actions they took during the game. Therefore, there must be a logging system that is constantly updated during gameplay with the actions performed by players and possibly a way to make suggestions for the guided actions that they can possibly take.

**Scoring System**

At the end of the game the player will be given a score based on the three criteria of scope emissions along with their operational and investment costs. Scope 1 emissions are direct emissions directly related to the player and the company emissions that they allow to occur. Scope 2 emissions are emissions such as electricity that are not directly caused by the player, but are tied to player usage of certain resources. Scope 3 emissions are related to the suppliers that the players use such as with resources that are given to them by other third parties. The player’s score at the end of the game will be represented with multiple scores like scope 1 and 2 being one score, scope 3 will be a separate score, total investments, and operating costs, product costs, product carbon footprint, as well as time as numeric values/scores.

**Scene Change System**

The scene change system is a very simple component which allows for switching between scenes in the game such as the startup page and main game page.

## **Low-Level Design**



The component design diagram (Fig 4) is used to keep track of the individual components of each of the game objects so that the overall planning of adding in components is well structured and there are no unknown components that could cause issues down the line. The Tiles class maintains the various resources/buildings that players can interact with in the game, which will interact with the game map represented with the Game Map class. The Metrics class maintains the various factors in the game such as the operating and investment costs that’ll be used for determining the player’s score throughout the game. The User Interface class represents the various UI elements used in the game like the UI panels for displaying the different metric values and displaying the sustainable actions menu.

**GUI Design**



The game UI (Fig 4) will be rendered on top of the game map, and it will mainly display the various scores that are kept tracked of and impacted by the player’s actions in the game. The Settings icon on the bottom left of the screen allows players access to the “Quit Game” button when clicked on, and clicking that button quits the game.



The following is the startup page (Fig 5) that players will see when starting the game, and the “Play Game” button is the button that lets players start the game when clicked on.





The main tile menu (Fig 6) is the menu that appears when a player clicks on a tile, and it displays stats about the tile along with giving the player an option to move the tile or do a sustainable action with the tile. After clicking the “Sustainable Actions” button, the sustainable actions menu (Fig 7) appears and at first only displays the possible actions a player can do for the tile. After an option is chosen, then the rest of the menu is filled up with text indicating the action chosen and its impact on the players’ various scores before prompting the player to enter a value and either confirm/cancel the action with their respective buttons.

# Implementation

*Author(s): Dev Patel*

*Reviewer(s)/Editor(s): Madeleine Ewart, Kennedy Tate*

## **Iteration Definition & Current Status**

## Iteration 0: Jan 18 - Feb 2 *(Completed)*

This iteration consisted of:

* Planning and setting up the project
* Participated in formal task planning with the teaching staff
* Creating requirements for the project and discussing it with the sponsor and teaching staff
* Setting up initial Unity project and getting familiar with Unity
* Starting the design of the game map in Unity
* Creating initial design wireframes

This iteration was a little longer than the others because initial expectations were being established and requirement elicitation took longer than expected.

## Iteration 1: Feb 2 - Feb 16 *(Completed)*

This iteration consisted of:

* Starting the design of the game startup page
* Updating and finalizing the low-level and high-level design
* Creating art assets for the game
* Updating the game map with the new team assets created

## Iteration 2: Feb 16 - Mar 8 *(Completed)*

This iteration consisted of:

* Updating the design of the game startup page
* Starting the implementation for the moving tile functionality
* Starting the design of the main tile menu
* Starting documentation on the project
* Planning/Writing the system tests for the game

## Iteration 3: Mar 8 - Mar 29 *(Completed)*

This iteration consisted of:

* Starting the implementation of the main tile menu
* Creating and integrating a configuration file into the game
* Starting the implementation of the scoring system
* Updating the UI of the game map and main tile menu
* Having the sponsor playtest the game
* Running system tests on the game
* Updating project documentation

## Iteration 4: Mar 29 - April 23 *(Completed)*

This iteration consisted of:

* Updating the game map with new art assets
* Finalizing the UI of the game map and main tile menu
* Designing and implementing the sustainable actions menu
* Finishing project documentation (Consisted of the deployment/user/development guides and the FPR)
* Participating in the final handoff meeting and the Posters & Pies event

This iteration took the longest because final expectations/requirements were being established and polishing/finalizing the systems implemented in the game took a little longer than expected.

## **Security Considerations**

Since the project contains the login and user creation features as stretch goals for the game, security concerns revolving around protecting user info and authenticating valid users will likely have to be addressed. Because of this, the login system will need some kind of encryption/decryption system for validating registered users along with ensuring the integrity of registered users by making sure hackers aren’t able to access users’ info or login as an impersonated valid user.

## **Project Folder Structure**

The project is structured with the following nested folder structure:  
  
Unity File Structure

* Clariant
  + Assets
    - Actions
    - Configuration
      * Actions.csv
      * Config.xlsx
      * DefaultGameSettings.cs
      * Startingpoint.csv
    - Images and Sprites
      * Backgrounds
      * Clariant Logos
      * Game Map
      * Team Art
      * UI
    - Prefabs
    - Scenes
      * Unused Scenes
        + Choice Menu.unity
        + Main UI.unity
      * Final Score.unity
      * Game Map.unity
      * Startup Page.unity
    - Scripts
      * ButtonScripts
        + Building.cs
        + CoalPlant.cs
        + PPA.cs
        + REC.cs
        + SolarPlant.cs
        + Swap.cs
        + Tile.cs
      * Input
        + CustomCursor.cs
        + MouseInput.cs
        + TileClick.cs
      * Resource Scripts
        + Emmision1Count.cs
        + Emmision2Count.cs
        + Emmision3Count.cs
        + InvestmentCount.cs
        + ProductCarbonFootprintCount.cs
        + UnitProductCostCount.cs
      * Scene Displays
        + SceneLoader.cs
        + SwitchPanel.cs
        + TogglePanelVisibility.cs
      * Time
        + TimeCount.cs
        + TimeManager.cs
        + TimePointSystem.cs
      * Tooltips
        + Tooltip.cs
        + TooltipSystem.cs
        + TooltipTrigger.cs
      * CameraSystem.cs
      * CSVReader.cs
      * GameManager.cs
      * QuitGame.cs
    - TextMeshPro
    - Tile Palette
      * Ground.prefab
  + Library
  + Logs
  + Packages
  + ProjectSettings
  + Temp
  + UserSettings

The project is organized using the default Unity folder structure, where most of the code for the game is placed within the Assets folder. Inside the Assets folder are additional folders such as the Configuration, Scenes, Scripts, Prefabs and Images and Sprites folders that contain assets and various code files used for creating the game in Unity. For example, the Images and Sprites folder contains art assets used in Unity, with the Game Map folder containing Unity art assets used for creating the game map and the Team Art folder containing the art assets created by the team. The Scripts folder contains code for the various scripts used for running the primary gameplay systems featured in the game, like the time and move tile systems. The Scenes folder contains the game pages in Unity that’ll be used in the game such as the startup page and game map page, with the Prefabs folder containing the buttons and text panels that will be featured in the game like the Login button and game map UI text panels.

## **Project Configuration/Settings**

If a developer wanted to install and work on the game, they would need to install Unity, in particular the version 2023.2.7f1 along with Visual Studio Code 2019 in order to work with the game within Unity. The configuration file used for the game is the DefaultGameSettings.cs file contained within the Scripts folder, which defines the main variables that’ll be used in the backend for representing various parts of resources players will interact with in the game and how the player will be scored, like the investment and cost amounts for resources such as forests and solar plants. Furthermore, the configuration serves as a way of defining the initial amounts of the variables declared in the file to define the player’s initial resource amounts and score when first starting the game.

# Testing

*Author(s): Tomas Robinson*

*Reviewer(s)/Editor(s): Kennedy Tate*

## **Overall View**

Given the gamified nature of the project, the most appropriate form of testing was black box and acceptance testing given how difficult unit testing can be in a game setting. Despite that, the team tried to do some unit testing by creating some unit tests in Unity using Unity’s built-in unit testing, but wasn’t able to get it working properly. Testing was done towards the end of development in the form of gameplay testing to make sure that the functionality and requirements are met in accordance with what was requested from the project sponsor. In order to ensure that the tests are consistent and repeatable, there will be a very clear and concise checklist given to testers to use throughout the testing process to ensure there are no issues regarding clarity of instructions for playing the game.

## **Unit Testing**

Although testing of this type may be limited and not as thorough as for other systems it is still possible to perform some white box testing by using some tools provided by the Unity development environment. This may allow for some issues to be caught that a human tester may not catch that some tools may be able to thereby making the system components of the game higher quality. In addition, it may also be possible to use static analysis tools for the individual C# scripts used in the game to ensure that there are no bugs or vulnerabilities included in the game code that could pose an issue before testing begins. The beginnings of a unit testing segment has been introduced to the project under a file named GameStart so that it can be tested more in depth for future developers who attempt to add more features to the game in the future. The Unity Testing Framework was what is currently being used to help write the unit test methods and currently only focuses on the game when it first starts up to ensure that all the values are properly displayed.

## **Acceptance Testing**

Testing for acceptance will be very straightforward and simple provided that the requirements and rubrics are followed meticulously, and this was done by the sponsor. For the testing process, the sponsor was given a small playable beta so that he could test the various gameplay systems and make sure that all desired functionality was working properly.

| **Test ID** | **Description** | **Testing Steps** | **Expected Result** | **Actual Result** |
| --- | --- | --- | --- | --- |
| GR4.1: ToolTipHUD | Makes sure that the tool tip functions properly when the cursor hovers certain labels/buttons | Prerequisite(s):  The game is running  1. Move mouse over a UI-labeled box or menu button | A textbox message should appear near the UI-labeled box or menu button the mouse is hovering over | Partially implemented (Only certain UI-labeled boxes and menu buttons have tooltips implemented) |
| GR3.2: Scope1HUD | Ensures that the scope 1 emission value tied to the Clariant building is displayed | Prerequisite(s):  The game is running  1. Click the Clariant building and observe the menu stats | The scope 1 amount should be displayed in the menu that appears | The scope 1 emissions value is displayed as 1 of the possible stats shown for the Clariant building |
| GR3.2:  Scope2HUD | Ensures that the scope 2 emission value tied to the Clariant building is displayed | Prerequisite(s):  The game is running  1. Click the Clariant building and observe the menu stats | The scope 2 amount should be displayed in the menu that appears | The scope 2 emissions value is displayed as 1 of the possible stats shown for the Clariant building |
| **Test ID** | **Description** | **Testing Steps** | **Expected Result** | **Actual Result** |
| GR3.2: Scope3HUD | Ensures that the scope 3 emission value tied to the Clariant building is displayed | Prerequisite(s):  The game is running  1. Click the Clariant building and observe the menu stats | The scope 3 amount should be displayed in the menu that appears | The scope 3 emissions value is displayed as 1 of the possible stats shown for the Clariant building |
| GR3.2:  PowerHUD | Ensures that the power usage HUD tied to the Clariant building is displayed | Prerequisite(s):  The game is running  1. Click the Clariant building and observe the menu stats | The power output should be displayed in the menu that appears | The power output value is displayed as 1 of the possible stats shown for the Clariant building |
| GR4.2:  MoveTile | Makes sure that the tile selected by the cursor can be placed on any arbitrary free tile | Prerequisite(s):  The game is running and it is the beginning of the game  1. Select a tile on the game map  2. Click the “Move Asset” button in the tile menu that appears  3. Click the mouse on any free space on the map | The tile should be visibly placed at the position where the mouse was clicked at | The chosen tile can be seen at the location where the player clicked on the mouse to place the tile |
| **Test ID** | **Description** | **Testing Steps** | **Expected Result** | **Actual Result** |
| GR4.2:  ConvertTile | Makes sure that the chosen tile can be partially converted to a different tile | Prerequisite(s):  The game is running and the power plant is on the map  1. Select the power plant and click the Sustainable Actions button  2. Select the option to convert to a solar plant and press Confirm | A solar plant appears for the player and can be placed on any free space on the map | A solar plant appears for the player and is able to be placed any available space on the map |
| GR5.4:  TimeStep | Upon running out of action points in a given time ensure that the timeline advances and resets the action points | Prerequisite(s):  The game is running and there’s 1 action point left  1. Observe the time  2. Select a tile and select a sustainable action for the tile  3. Observe the time again | The action points should be reset to the default as well as the time being advanced by a month | Not implemented |
| GR7.1:  GameEnd | Make sure that the game ends when the timeline reaches the given end of the chosen limit | Prerequisite(s):  The game is 1 month away from the end time  1. Select a tile and do any sustainable action on the tile  2. Observe the end message | There should be a message that pops up stating that the user has run out of time which has been set | Not implemented |
| **Test ID** | **Description** | **Testing Steps** | **Expected Result** | **Actual Result** |
| GR7.1:  ScoreShown | After the game successfully ends, display the scores of the player | Prerequisite:  The game has ended after the user has run out of time  1. The player clicks on the show score button | The scores of the player are displayed as well as being recorded for observation in logging | Not implemented |

Table 2: Project System Test Plan/STP

The acceptance testing was also done in the form of playtesting by the sponsor to ensure that the requirements were being met according to what was wanted and along with this was the fact that the supervisor of the sponsor was also giving feedback which helped to refine the final version by increasing the efficiency of finding issues and helping to fix the iterations at the end of the project through clear insights. However, this was also the main reason why only 60% of the system tests are passing since the requirements of the project kept changing throughout the semester with the sponsor, which slowed our progress with implementing the requirements for the project compared to what the team initially expected to have completed.







# Task Plan

*Author(s): Sam Adkins*

*Reviewer(s)/Editor(s): Madeleine Ewart*

| Team 38 Task Plan | | | |
| --- | --- | --- | --- |
| Item | Owner(s) | Due Date | Status |
| First Sponsor Meeting | All | 1/18/2024 | Complete |
| Iteration 0 -   1. Project Requirements 2. Startup Page Design 3. Game Map Design | 1. All | 1/18/2024  -  2/2/2024 | Complete |
| 2. Madeleine Ewart |
| 3. Sam Adkins |
| Oral Progress Report 1 Presentation | Tomas Robinson | 2/8/2024 | Complete |
| Iteration 1 -   1. Create Art Assets 2. Update Game Map 3. Create LL/HL Diagrams | 1. Tomas Robinson, Kennedy Tate | 2/2/2024  -  2/16/2024 | Complete |
| 2. Dev Patel, Sam Adkins | Complete |
| 3. Madeleine Ewart | Complete |
| Iteration 2 -   1. Update Startup Page 2. Main Tile Menu Design 3. Move Tile Functionality | 1. Madeleine Ewart | 2/16/2024  -  3/8/2024 | Complete |
| 2. All |
| 3. Sam Adkins |
| Oral Progress Report 2 Presentation | Sam Adkins, Madeleine Ewart | 3/7/2024 | Complete |
| Interim Project Report | All | 3/8/2024 | Complete |
| Iteration 3 -   1. Implement Tile Menu 2. Implement Config File 3. Score System Functionality 4. Update Game Map/Tile Menu UI | 1. Sam Adkins, Madeleine Ewart | 3/8/2024  -  3/29/2024 | Complete |
| 2. Tomas Robinson | Complete |
| 3. Kennedy Tate, Sam Adkins | Complete |
| 4. Dev Patel | Complete |
| Posters and Pies Rough Draft | All | 3/24/2024 | Complete |
| Oral Progress Report 3 Presentation | Dev Patel, Kennedy Tate | 4/2/2024 | Complete |
| Iteration 4 -   1. Finalize Game Map/Tile Menu UI 2. Implement SA Menu 3. Update Score System Functionality | 1. Dev Patel | 3/29/2024  -  4/12/2024 | Complete |
| 2. Sam Adkins |
| 3. All |
| Finalize Posters and Pies Posters | All | 4/16/2024 | Complete |
| Final Presentation | All | 4/18/2024 | Complete |
| Final Written Report and Guides Due | All | 4/23/2024 | Complete |

Table 3: Project Task Plan

**Team Contact Information**

Sam Adkins, Sponsor Contactor, sameadkins@gmail.com, (919) 888-2833

Madeleine Ewart, Development Manager, madelafin@gmail.com, (910) 988-2405

Dev Patel, Proofreader/Submitter, dpamaze01@gmail.com , (980) 422-5588

Tomas Robinson, Team Leader, tomasrobinson2002@gmail.com, (980) 395-5862

Kennedy Tate, Stenographer, ktat320@gmail.com, (252) 341-2015

# 

# Appendix





















****





