# Developing Educational Games: SELFISH

Jason Pang

2034852

Project Dissertation



# Declaration

## Statement 1

This work has not been previously accepted in substance for any degree and is not being con- currently submitted in candidature for any degree.

Signed Jason Pang (2034852)

Date ............................................................ 26/04/2024.

## Statement 2

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by citations giving explicit references. A bibliography is appended.

Signed Jason Pang (2034852)

Date ............................................................ 26/04/2024.

## Statement 3

The University’s ethical procedures have been followed and, where appropriate, ethical approval has been granted.

Signed Jason Pang (2034852)

Date ............................................................ 26/04/2024

## Summary

Selfish is a game designed to study and educate around the self-fertilizing fish Mangrove rivulus(Kryptolebias marmoratus), while keeping the fun of a game for the user.

The game presents a simulation of ecological environment on the mangrove fish, where player will control one of the hermaphrodites mangrove fish in the environment and face on various difficulties including avoid pathogens, finding food and reproduction etc. before dying from old age or infection. Given the variety of pathogens, one must at all costs avoid getting infect if the fish do not have the immunity, occasionally a male mangrove fish will appear, granting the player an option to increase virus immunity by outcrossing with it. With such enriched gameplay, the player will immerse into the experience of being a mangrove and learning their behavior patterns.

An implementation of the game exists as the time of the study start, and it will be the base point of additional requirement for the game, such as including additional factors like temperature, data collection for study purposes etc..

## Acknowledgements

This study starts from the concept created by, and in collaboration with Dr. S. Consuegra, Dept. Bioscience, Swansea University.

The original implementation of the game was written by Matthew Harrison-Jones and developed by Ian Clarke, the implementation was taken as the starting point of study.

## Contents

[Developing Educational Games: SELFISH 1](#_Toc165004297)

[Declaration 1](#_Toc165004298)

[Statement 1 1](#_Toc165004299)

[Statement 2 1](#_Toc165004300)

[Statement 3 1](#_Toc165004301)

[Summary 2](#_Toc165004302)

[Acknowledgements 3](#_Toc165004303)

[Contents 4](#_Toc165004304)

[1 Introduction 5](#_Toc165004305)

[1.1 Motivation 5](#_Toc165004306)

[1.2 Aims and objectives 5](#_Toc165004307)

[1.3 Related Work 5](#_Toc165004308)

[2 Background 6](#_Toc165004309)

[2.1 Mangrove Rivulus 6](#_Toc165004310)

[2.2 Biological Theorem 6](#_Toc165004311)

[3 Specification 7](#_Toc165004312)

[3.1 Requirements 7](#_Toc165004313)

[3.2 Problems to anticipate 7](#_Toc165004314)

[3.3 Tool to use 8](#_Toc165004315)

[4. Implementation 8](#_Toc165004316)

[5. Evaluation 10](#_Toc165004317)

[6. Conclusion 10](#_Toc165004318)

[References 11](#_Toc165004319)

## List of Figures

[1 The act of performing self-fertilizing.](#_bookmark6).................................................................3

## 1 Introduction

### Motivation

Games have existed since the start of humanity, and gone through continuous development ever since its appearance, such length of time resulted in numerous amounts of game categories, but what kind of considerations and principles is needed for a game to be accurately categorized? For educational games, it is normal to consider “having a learning objective” in order to be called educational games, but such way of consideration means that as long as there is a learning objective, it can be called an educational game, so there should be more than that in order to consider a game as educational game.

### Aims and objectives

The aim of this project is to study and understand what principles are needed for developing an educational game. As a starting point, a prototype implementation of the game “selfish” was given and used to satisfy the main aim of this project.

Various objectives are require to transform the prototype implementation into a functional game, this includes: implementing a data collector in form of .csv (comma-separated values) files, the introduction of custom scenarios for the game, the introduction of new factor that affect how the level is played, and bug fixes.

### Related Work

The prototype implementation of the game “selfish” is provided, originated from Matthew Harrison-Jones, from the concept by Dr S. Consuegra and developed by Ian Clarke, is a game about life as a specific type of fish called “Mangrove Rivulus”, where out of 10 levels player is given the option to be gradually introduced to the simulated environment through first 5 levels or the introduction levels of the game, learning about how to eat, survive, avoid predator and reproduce offspring or battle against all odds with what you learn through the last 5 levels or the advanced levels.

Upon inspecting the code, a prototype of making a custom level is also provided, but was disable for unknown reason, this gave a good starting point of making custom scenarios for the game.

## Background

### Mangrove Rivulus

Mangrove rivulus or Kryptolebias marmoratus, is self-fertilizing fish that consists mostly of hermaphrodites (means that it has both male and female reproductive organs), and some males, the amount of males is depends on the requirement for genetic diversity (for example, an increase in local parasite population will leads to an increase in male density), with an average of 3%~8% and up to 25% recorded in harsh environment[1], otherwise group of mangrove will mostly be the same as a result of long-term self-fertilize as it is more beneficial than performing outcross.

### 2.2 Biological Theorem

As the player mature and have enough food, they are given the option of self-fertilizing or outcross with a male, it is the player’s strategy to consider which way do they want to go for, by choosing self-fertilizing, it is likely to implied that there’s is no immediate threat and the offsprings will survive just as fine as you or the chances of finding male are rare to the point that self-fertilizing is your only option, in the other hand, performing outcross implies threats are noticeable and more generic diversity is needed for the species to survive and extend its bloodline, creating more generic variation that leads to a more resistance offspring to adapt the harsh environment.

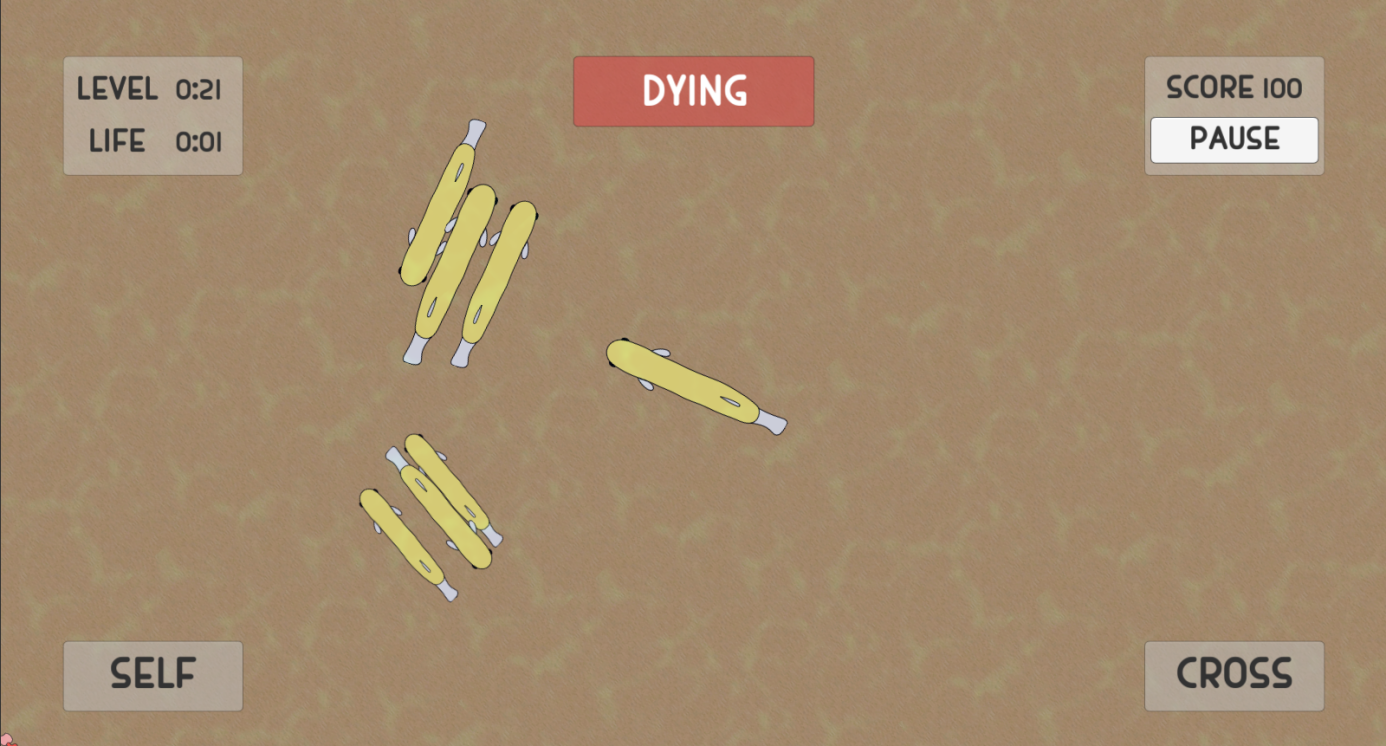


Fig1. The act of performing self-fertilizing

## Specification

### Requirements

For the game to work as planned, there are features that is required to be implemented, these will be divided into two categories:

* + 1. Functional Requirements
    2. Non-Functional Requirements

By fulfilling the requirements stated, this will enrich and complete the prototype implementation and get a better scope on understanding the principles of developing educational games.

#### 3.1.1 Functional Requirements

This includes:

1. A folder for storing collected data
   1. A Class for handling data collection and writing

2. A new factor to be introduced (possibly temperature)

3. A custom scene to simulate different scenarios outside of provided levels

4. Bug Fixes

#### 3.1.2 Non-Functional Requirements

Features including:

1. Maintain stable performance throughout the whole game.
2. Accurate simulation that is close to the real world.

### Problems to anticipate

#### 3.2.1 Data Collection

There are a few problems to anticipate during development, such as limiting data collection to write to one time per second, ensure file is saved to the correct level when implementing a manager to collect data from game so there is no thousands of lines or thousands of files when collecting and write data for just 1 level.

#### 3.2.2 Introducing New Factors

Introducing new factors into the game is also a huge problem, even with careful design one cannot fully predict what the outcome might be as some build off existing functions, thus might break the game with significant or hidden errors and requires more time to do debug and testing.

#### 3.2.3 Custom Scenario

The disabled custom scenario might be a indicate of broken functionality and require extra care or even scrap and rebuild the code for worst case scenario, when the bug fix for custom scenario is done, introduced factor needs to be implemented to assure the accuracy of creating a custom scenario.

#### 3.2.4 Bugs

The biggest problem to anticipate is bug fixes, with all the above mentioned can cause a bug, and the bugs the prototype implementation brought with, it will be the most time-consuming part of the development for the game to work.

### Tool to use

#### 3.3.1 Unity

For this project Unity is favored to develop the implementation as it is one of the most popular game engines, as a renown game developing engine that has been used worldwide for years, it has one of the largest user base and forum for developers to exchange and understand various questions as well as a refined documentation.

Another reason for the use of unity is the environment of implementation run on, upon received the given implementation and inspect the file inside, there are files that indicate the prototype implementation is worked on a unity editor, with the use of unity it is easier to start working on the implementation as it requires little to no conversion to make the code running.

## Implementation

With some work, I have successfully implemented the requirements mentioned above, such as having data collection for the game, a working custom scenario for the game including the implementation of temperature factor and various bug fixes.

#### 4.1 CSV Manager

The CSV Manager is a class whose sole purpose is to collect various data from the game and write the collected data into a .csv format file, The manager will first validate to see if there is a folder to write data in, when you enter a level, the manager will collect what level you are in, try create a file in a specific folder with the format of “Level-(current level)-(local time)”, and append the file header into the file, the headers includes:

* + 1. the current time passed on the level
    2. the player’s current total fish
    3. the current total fish on the level
    4. the amount of self-fertilizing done
    5. the amount of outcross done
    6. the current score
    7. the temperature if you are playing on a custom level

More data will be collected during the playthrough of a level, and a series of checks will be performed to determine if the manager should append to the file or create a new file. In the CSV Manager class, the function CheckIfTimeIsRight will check if the current time + 1 is the same as the last recorded time, which serves as a file separator so only the necessary data is appended to file but not all data to a single file. To avoid thousands of lines into a file in just 60 seconds of gameplay, statistic collect has been limited to collect once every second until the level ends.

#### 4.2 Custom Scenario

When re-enabling the disabled custom scenario, no major bugs have been found and only new functions are added, such as function for calling CSV Manager and temperature change.

#### 4.3 Temperature

The temperature factor is added to custom scenario only as a factor to control the percentage of male spawn on self-fertilize and outcross. In an attempt to provide a more accurate simulate environment, the lower the temperature is, the more male will be produce on self-fertilize and outcross, and the higher the temperature, the more herms will spawn[2], this is done by setting a curve on self-fertilizing and outcross male ratio, and there is a chance on the temperature being randomly changed by +1/-1 each second.

#### 4.4 Bug fixes

Upon receiving the implementation, there is a few major bugs on the code that can drastically change the game, such as the camera will stay on the same spot until the level end when the player dies and have a offspring, this is fixed by removing a line of unnecessary code so the camera can get an instance of your offspring instead of dead because one field is left empty. Another major bug is that the AI will not move, no matter if it’s the fish, fish food or the viruses, this is fixed by using the AddComponent function instead of GetComponent function and the fish world is much more lively after the fix.

## Evaluation

#### 5.1 Principles of Developing Educational Games

During development, I have come to a realization that educational games are not just about letting kids learn through simple games, it is much more than that. By utilizing simulation player have a higher immersive experience to the game by portraying themselves as a living fish and make decisions based on how healthy it is to the player and it’s offspring, avoiding predators and survive, such creation of engagement leads to higher focus and increase in learning efficiency.

Since it is a game for investigating selection hypotheses, with enough data collection, it can help researchers to perform analysis on why such selection is made, educational games are not just about educating on one specific group of people but as much variant as one hope for.

With the gradually increase in difficulty, it is beneficial and allow a longer and higher engagement time with the game as the game showing challenge to the player’s skill level, giving players a sense of danger and a will for improvement in order to out skill the challenge provide by the stage.

#### 5.2 Expanding the project to different directions

To investigate on the selection hypothesis, a manager class is created to collect multiple categories of data while the game is running, such as the number of fish you own, the amount of self-fertilize and outcross perform, written file is categorized through different levels. The data can then be used to perform analysis on why such selection exists.

As a simulation game, the custom scenario has been restored to great working conditions so player can have some fun on how the want the game to go, such as making a easy simulation by limiting the amount of pathogens and predators exists and giving yourself more immunity, or give yourselves a really hard challenge like reducing food spawns, male spawns or even herm spawns. Data collection is applied on the on the custom levels as well. More factors are introduced in the custom levels to introduce a more chaotic playground for players.

## Conclusion

In conclusion, developing educational games is more than just getting some code in and make a game out, a lot of consideration and principles is taken into account for a educational game, no goals and it is just a regular game, no user engagement and it will just be a boring moving picture that does not grab the player’s attention, and some game can be used for research purpose such as collecting data in a simulation game to study any reason behind a decision or more, by taking some of these principles and consideration while developing, it is easy to smash education and entertainment together and make improving fun.

## References

1. Wikipedia page of Mangrove rivulus, URL: <https://en.wikipedia.org/wiki/Mangrove_rivulus>
2. Biology of Mangrove Rivulus, URL:

https://fishbase.mnhn.fr/summary/3213