**1. General Information**

Fishing Frenzy

Lauren Carrillo, Mason Chen, Anders Choy, Dawson Harris, Will Zhang

*Abstract*

Among the world’s largest problems, biodiversity loss, specifically in the ocean as a result of human activity and misuse, destroys ecosystems all around the world. A way to counteract this damage is moderating fishing and ensuring it is done in a way that sustains the preexisting ecosystem. In Fishing Frenzy, players make the decisions on the types of fishing a fishing company implements and witness both the company’s profit and the ecosystem’s biodiversity fluctuate. This game is meant to teach the harm fishing has on ecosystems, so they can take what they learned and apply it to their everyday lives.

**2. Feedback Synthesis**

*Summarized Reviews:*

The reviews from our stakeholders had been overwhelmingly positive, with suggestions to give more in-depth descriptions of each fishing method [Jessica, Nina, Zihan] and the player’s goal [Snigdha, Alex]. Another review suggested increasing the complexity of the graphics, as one stakeholder mentioned that they were confused when a population fell under 1,000 and the graphic showed no visible fish [Leo]. Similarly, other stakeholders wished that there was more fish imagery [Michael, Jackson].

*Reviewing Criteria from Milestone D:*

1. Clarity of Goal or Objective:

While many stakeholders found the goal clear, many requested that we state it explicitly at the beginning of the game. In this, there was a demand for an explanation of the fishing methods and biodiversity index, instructions on how the game works, background information on the player’s character, how each round works, and how many turns the player has.

1. State Visualization:

The stakeholders found the visualization clear, with the only misconception being that when the fish number drops below 1,000, the bar that shows the species population disappears. However, below this bar, it does state the correct number of fish left of that species.

1. Operators:

The operators were stated to be clear, with the only wanted bonus being for there to be descriptions of each fishing method so that the average person would be able to gain a clearer understanding of exactly what option they are choosing.

1. Credibility of the Model:

The stakeholders found the underlying game model to be credible. They also agreed that the state variables and operators are related to the wicked problem.

1. Time Frame:

The time frame was said to be clear and credible, with the only complaint being that it could be difficult to find for some.

1. Additional Operators?:

Some suggestions:

* The company is able to actively try and conserve biodiversity
* Environmental operators that could increase/decrease pollution and hurricanes
* Players could buy live fish to increase breeding
* Players could establish an estuary/fishery
* Profit could also decrease (not just increase)
* Adding to the “do nothing” operator (fish will reproduce faster)
* Spearfishing
* Apply real-life fishing policies
* Remove some operators
* Adding how much money the act of fishing uses
* Make fish farms

1. Expected Level of Engagement:

Players said that they felt engaged, especially at the beginning. However, as the game went on, they said that it felt repetitive at times, noting that they would appreciate it if the visuals were more complex.

1. Scoring:

Many of the stakeholders stated that they would have liked an explanation as to what the biodiversity score/index means, but the overall scoring indicators were clear and worked for the game.

1. Knowledge Gained:

The stakeholders said that they learned about measuring biodiversity and how different fishing methods affect the biodiversity of a region. They would like to see some real-world applications and examples of these phenomena, given as text in the game.

1. Other Suggestions:

* Progress bar to measure profit
* Make bomb fishing not end the game
* Make operator list shorter
* Add real-life statistics

**3. Specific Responses to the Feedback**

In response to the suggestion of a deeper explanation of each fishing method and goal state, we knew that we need to provide more of a textual explanation to the player on how each fishing method works, and what the player is specifically trying to achieve. In the final version of the game, we included a description of the goal state at the start of the game. Another aspect that we implemented into our game was upgraded visualization, as we implemented the use of the TK3 client, allowing us to add imagery for each of the fish, as well as sound effects, to make the game more engaging. However, although many stakeholders requested there be in-depth descriptions of each fishing method, we were unable to add them into the game. A play style we implemented was “learn by doing.” So, throughout the game, the player learns through trial and error the ways in which different fishing methods will have different effects on the environment, and learn in a similar way to how people in real life learned about the harm different fishing methods could have on an ecosystem. By throwing them into the game with little background knowledge of each fishing method, they have to assess all the variables themselves, and piece together their own explanation as to why certain methods are “bad.”

**4. Self-Assessment:**

*Scores and Justifications:*

1. Clarity of Goal or Objective: (5)

The goal of the game is clear – to choose fishing methods that deal the least amount of damage to the ecosystem (measured by the biodiversity score) while also generating the most profit for the fishing company.

1. State Visualization: (5)

Although the design of the visualization is simple, it is clean and straightforward and gives the player a way to physically see the progression of the fish population. The different sections are already labeled with text, to give the player an easy way to decipher between the different columns.

1. Operators: (5)

The list of operators is clear and is sufficient in terms of what our player needs to understand. They allow the player to see every single option that is available to them, and provide the player with a wide range of options that logistically and realistically work. All 18 operators involving the different fishing methods for different fish species are introduced at the initial state and stay available as long as the preconditions are met.

1. The credibility of the Model: (4)

The model of population dynamics and its relation to biodiversity loss had to be simplified for the sake of the game. While the general theme of population fluctuations and growth that eventually levels out is accurate, the game has to disregard several factors, including prey and predator relationships and external factors (human-related activities not including fishing and natural disasters). Additionally, the populations and birth rate across all species of fish have been standardized, which doesn’t follow real-world relationships. Still, this was a choice that was deliberately made, as it doesn’t take away from the potential lessons learned.

1. Time Frame: (5)

The timeframe for our game is several rounds (a total of 20), each of which spans 3 months. This

allows for a more realistic representation of the time needed for a species to recover its population

via reproduction. Additionally, real-world fishing operations can span many months, so our time

per round is realistic.

1. Additional Operators?: (N/A)

An operator that would be interesting to add to the game would be one that would allow the player to buy different nets with their profit money in order to catch less bycatch.

1. Expected Level of Engagement: (4)

This game is plenty engaging, as there are many different operators given to the player, meaning that they can play the game differently every time. There are also “random” events that occur as the game progresses, which makes sure that the player never gets too comfortable with their position in the game. In order to make this game more engaging, the graphics could be more complex and interactive.

1. Scoring: (4)

The primary score mechanism that the game is scored through is the final biodiversity score at the end. If the score is too low, meaning that there were fewer fish in the ocean than when the game started, the player will lose the game. If the player maintains a high biodiversity score, they will win the game. However, the player cannot cheat the game and not fish at all in order to keep the biodiversity score up. They must return back to work with a profit, or else they will be fired and lose the game. To make this more compelling, there could be certain thresholds that the player could meet in order to satisfy different biodiversity score quotas or to satisfy the company’s profit margin.

1. Knowledge Gained: (4)

In this game, the player learns how biodiversity loss occurs in the ocean, specifically through different fishing methods. The player observes how, over time, the consequences of overfishing and bycatch pile up and end up destroying an ecosystem. The player learns about the tradeoffs this company undergoes ‒ giving up potential profit in order to conserve biodiversity ‒ and can apply that thought process to real-world examples. It would be interesting if instead of focusing on just fishing, there were more components that contribute to biodiversity loss, and not just in the ocean. In the future, maybe the game designers would be able to shift the game so that the player would have to manage a few different companies – for example, a farming company on land, and the effects of that company would affect the biodiversity of the land, as well as the runoff affecting the biodiversity of the ocean.

1. Other Suggestions: (N/A)

A suggestion for this game would simply be a more complicated form of visualization. While what the game has right now is sufficient for engagement ‒ it gives the player a visual representation of current fish populations ‒ it is still extremely simple, with little graphics. In another version of the game, it would be interesting to see how it can be visualized in a way that is more complex, like graphics for the various fishing methods and fish species. This would increase both the engagement of the player and also increase their knowledge of the process.

**5. Game Structures and Mechanics:**

The structure of Fishing Frenzy is a turn-based game involving the decisions of WARMD company’s fishing method manager. The game involves feedback cycles of different rounds (spanning 3 months each), each of which includes an update of the game state, meaning the profit earned by the company, biodiversity score, biodiversity index, and populations of each fish species change. There are no stages in the game where new or different sets of operators are introduced - every available operator is revealed to the player at the initial state. As long as the precondition is met, any of these operators will be selectable by the player. There are no “special tricks” to the scoring mechanism. The player will have to experiment with each fishing method to find a balance between profit and maintaining biodiversity.

**6. Explanation of the Model:**

Fishing Frenzy models a singular phenomenon, albeit a simplified version for gamifying purposes: population dynamics. According to [ScienceDirect](https://www.sciencedirect.com/topics/earth-and-planetary-sciences/population-dynamics), population dynamics is defined as “the analysis of the factors that affect the increase, stability, and decrease of populations over time.” To model this in regards to fishing, the variables of population number and reproduction rates were introduced, along with a population “cap” at 10,000 individual fish per species. Populations fluctuate depending on the external factors (in this case human activity via fishing and occasionally events such as pollution) and reproduction rates, while the population cap represents the leveling-out or stabilization found in s-shaped population curves ‒ known as carrying capacity (occurs when a population of fish has been left alone). To simplify population dynamics, the factors of natural prey, predators, and mutual competition with other individuals were removed. In the creation of the variables, the team initially sought to incorporate accurate ratios of fish populations and reproductive rates. However, virtually no studies with usable values could be found on most of the species. Additionally, it was decided that standardizing every population (6,000) and reproductive rates (0.5) would make the game easier to understand for the player. Two small details included in the game were the unique prices of each species of fish, which were researched to simulate the real-world [market](https://www.statista.com/statistics/512015/retail-price-per-kilogram-leading-seafood-united-kingdom-uk/) for salmon, cod, and tuna, as well as [halibut](https://www.statista.com/statistics/196489/average-annual-price-of-pacific-halibut-in-the-us-since-2000/), [striped bass](https://fishmasters.com/how-much-does-striped-bass-cost-per-pound/), and [pompano](https://wildseafoodmarket.com/product/pompano/). The specific fishing methods that were restricted to only certain species were also researched so both these details added some complexity to the player’s perspective. For three more variables, potential choices include a level for marine plant life, prey populations (algae, shrimp, smaller fish), and predator populations (larger fish, sharks, seals, dolphins).

**7. Learning Retrospective**

*Lessons Learned in the Creation of this Game:*

1. Technical Coding

Ren:

When it came to visualization, something that we learned was that sometimes, it is necessary to take the route that is tedious and time-consuming in order to achieve what we wanted to achieve. Although there may be a faster way to do it, this way was easier for us to understand and execute, even if it took a little more time and effort to achieve.

Mason:

When we wanted to represent bycatch in each fishing method, just don’t forget to add bycatch as a variable. By analyzing fishing methods, such as long lines fishing, gillnet, and trawling, we could know which will bring the smallest impact and which will bring the biggest impact.

Anders:

From setting up state variables and attributes of various fish species to writing functions and algorithms for different fishing methods, I have learned how to more efficiently set up working systems and troubleshoot bugs or errors that occur here and there. It was all about trial and error and experience. Without much familiarity with classes and objects in Python before this course, this project allowed me to explore my creativity even with the limited structure of the SOLUZION clients. I believe it is very important to be able to adapt to different environments, hardware, and clients when it comes to coding, and I have been taught more about this with our game.

Dawson:

When creating this game, I learned a lot of things. This was my first time ever creating a graphical interface with Python, so using Tkinter was a whole new process for me to learn. Using Tkinter definitely had me frustrated at times because of how exhausting it was, but I actually am very proud of how we incorporated it into our games. Although our UI is simple, I learned a lot about the Tkinter library and how it functions. Also, when we were creating our game, I learned a lot more about state functions and using classes in Python. I am used to classes and namespaces since I mainly program in C#, but it was really cool to learn how to do stuff like that using a language as simple as Python.

Will:

This course was my first official learning experience with Python. Throughout the projects covered in this class, I learned the fundamentals and beyond. Specifically in this project, I gained more Python knowledge on classes, conditionals, and general programming structure. I learned how to apply my newly gained knowledge on wicked problems and the classical theory of problem-solving to the formulation of our game with Python. My teammate Anders was a crucial part of this learning experience; he helped me understand why the code worked and often helped me debug my issues. I also gained experience working with the SOLUZION clients, Tkinter, and sharing our code on GitHub and VS Code liveshare.

1. Wicked Problem and its Formulation

Ren:

For me, I think the thing that was one of the hardest parts of learning about the wicked problem was having to simplify it so much in order to fit the parameters of our game. Because of this, we had to learn to choose the parts of biodiversity loss that we *really* wanted players to learn through our game. In terms of wicked problems, this aspect of our learning process highlighted the true vastness and complexity of a wicked problem, and how it must be broken down in order to get a deep understanding of it.

Mason:

At the beginning of Project C, I thought Biodiversity Loss was kind of a huge topic to be covered. When preparing for the first presentation, I found articles and learned about the cause and effects of biodiversity loss, some solutions, and how to prevent this wicked problem. When our game was decided to formulate ocean biodiversity loss, I also found some additional resources like blast fishing which was banned in 1997. We need to be aware that ocean biodiversity loss has also been caused by water pollution and natural disasters, which means this is related to another wicked problem.

Anders:

Learning and understanding how our wicked problem Biodiversity Loss is very complex and consists of several characteristics including being a symptom of other problems, it was difficult to figure out what to focus on with our game. When the decision was being made, however, I thought that overfishing and loss of ocean biodiversity is a great direction since this issue is often overlooked and is well fitted as a game concept. During the production of our game, I have also learned that it is extremely challenging to make it close to real life because existing data is not complete and there are so many aspects to the wicked problem. These experiences and our research allowed me to realize how “wicked” and impactful these real-world problems are.

Dawson:

When I saw that our group was assigned to create a game on Biodiversity Loss, I was actually a bit nervous about creating a game about this wicked problem. This is because it was the one prompt that I did not have an idea for a game already thought of in my head. I thought that coming up with a game for this wicked problem might be a bit difficult because I knew that the contents of Biodiversity are broad and can expand from plants to animals, etc, and I wasn’t exactly sure what we would choose in designing our game. Luckily, when we chose a fishing game, we were able to find lots of resources about fishing companies and how they can harm ecosystems. With the resources all of us found, we tried our best to implement some of the facts we learned into our game. A great example of this is the various fishing methods we included in our operators. Because these methods of fishing still cause damage to ecosystems, it was important for us to try to make our game educational about the damages caused by our assigned wicked problem.

Will:

It was interesting to learn about wicked problems in general (their characteristics), but more so to dive into a specific one and attempt to formulate it as a game. Our team picked biodiversity loss, a complex problem with ties to many other environmental problems (climate change, human-caused pollution, resource overuse, habitat destruction). We researched how biodiversity loss impacts marine ecosystems in the ocean and aimed to simulate that in our game. For our formulation, we had to narrow down the scope of the phenomenon, and therefore pick and choose which aspects to keep while still making it realistic enough to be educational for the player.

1. Group Coordination

Ren:

Since this was a team project, we needed to figure out what time worked with everyone’s schedules, and this ended up with a lot of late nights working on this project. Time management was a huge part of what allowed us to finish this project on time, and something that was important to learn when it comes to a multi-person multi-part project.

Mason:

For the presentation and slides in Project C, we had several rounds of discussions and meetings. I researched several resources and picked up essential information from the articles that are useful for our game design. I learned that it is important to respond to your teammates if they point out a mistake on your part or they feel confused. You need to be responsible for your part so that you won’t hold back your team’s process. Since this is my first programming course, I realize there are a few things I can help with code writing, but I still tried to write a few codes in the precondition of not affecting the main project.

Anders:

From this game design project, I have learned a lot about collaborating with a team. Although we have different opinions and ways of working at times, good communication and adaptability are key. We always discussed issues, decisions, and reflections, which lead to a positive environment and outcome. I have also learned that splitting up the work, doing parts that we are more familiar with, and helping out teammates when needed is a very good method of doing this project. In a short span of time, we have had struggles and obstacles, but the breakthroughs and successes, to me, are the most fulfilling. With little teamwork experience, I had a fun experience working with my team.

Dawson:

In all of the other computer science courses I took before coming to the University of Washington, all of my CS assignments were solo projects. Because of this, I learned how to collaborate with a group while programming. While working on this project, I learned many things about working in a group. I learned how the Scrum process worked, and I even learned how to use tools like GitHub. I think the biggest challenge about working on a group project is coordinating a time when we are all free to meet up and work on the game. This was new to me since I’m used to just being able to work on projects at my own pace at any hour of the day.

Will:

Before this class, I had some prior knowledge of the SCRUM system (it was taught to me in my senior year JAVA class). However, through the lectures about the process, I learned how to better implement it to divide the workload, keep each other on the track, and consistently make progress. As the only one in my team that lives off campus, I had to communicate with my group about which days to stay on campus and work in person. When that wasn’t necessary, I joined voice or video calls on our team’s Discord server to work on the project remotely. This constant communication via SCRUM meetings in class/in person and talking online allowed for good coordination of our progress.

**8. Code Explanation**

*State Representation*

The string representation of game state is in the order of profit, biodiversity index, biodiversity score, salmon population, tuna population, cod population, pompano population, striped bass population, halibut population, bycatch, round(year and quarter), event.

An example would be: (Gross Profit: 25590k, Biodiversity Index: 0.831, Biodiversity Score: 97.6, Salmon left: 6500, Tuna left: 8000, Cod left: 2000, Pompano left: 5000, Striped Bass left: 6500, Halibut left: 2000, Bycatch: 2500)

Year 2026 Quarter 1

A factory had released tons of pollution into the ocean and fish populations are decreased by 1000.

The 3 key state variables would be fishList, biodiversityScore, and event. “fishList” includes all the fish species in our game as objects with their attributes. “biodiversityScore” determines the biodiversity of our game’s ecosystem, which is calculated using Simpson’s diversity index in a percentage format. Even though the variable “event” is only an integer, it represents the events which happen in real life and decrease fish populations/biodiversity.

*Sample Operator and Code Excerpt*

Operator 1: Use longlines to fish Salmon

This operator decreases the salmon population by 2000, increases bycatch by 200, and increases profit by the number of salmon caught multiplied by its price. I implemented this with a general move() state transfer function that manages all the events and fish reproduction while including this fishing\_method() function below, which selects the fish species and amount being caught and returns the profit. I also programmed a Fish() class in order to store the details, including name, population, price, reproduction rate, of our different fish species.

def fishing\_method(self, method, species):

if method == 6: #bomb fishing

psum = 0

for f in self.fishList:

psum += f.number \* f.price

f.number -= f.number

return psum

elif method == 1 and species != 3 and species != 5: #longlines targets specific species except for halibut and pompano

self.fishList[species].number -= 2000

self.bycatch += 200

return 2000 \* self.fishList[species].price

*Special Packages or Modules Needed*

The libraries required to run our game include Tkinter and PIL. Pillow (PIL) must be installed in order to run the Tk3 version of Fishing Frenzy. The version of Pillow we are using is 9.2.0. Installing Pillow requires Pip. Pip v22.2.2 can be installed by downloading and running [this python file](https://bootstrap.pypa.io/get-pip.py). After installing Pip, you can install Pillow by typing “py -m pip install --upgrade Pillow” in Windows’ command prompt.