Software Requirements Specification (SRS)
Angle Anglers

Team: 8
Authors: Christopher Coco, Evan Casasanta, Vraj Patel, Sebastian Gyger
Customer: Students in the 5-7th Grade
Instructor: Dr. James Daly

1 Introduction

This document contains the Software Requirement Specification (SRS) for Angle Anglers, an educational fishing video game. This SRS will describe the software's functionality and purpose.

1.1 Purpose

This SRS document's purpose is to describe the system's requirements and constraints. It will also outline the functionality of the software. It will contain a list of requirements for the software, diagrams that outline different components of the game, and describe a prototype of the game. This document will help developers have a solid vision and blueprint for what the project is trying to accomplish. It will also allow clients to learn more about what the project has to offer and the potential benefits of the game.

1.2 Scope

Angle Anglers is an edutainment game designed to improve student engagement and incentivize learning in the classroom by helping students learn about angle concepts in a fun and engaging way. Our intuitive interface and powerful features empower users to clear doubts about angles. Students struggle with mathematics because of a lack of repetition in solving math problems. Our game teaches students about angles through an interactive fishing minigame that improves the students skills through repetition. It will help them recognize patterns in the way to solve the problems which can help them improve their problem solving skills.

1.3 Definitions, acronyms, and abbreviations

Angle Anglers - The title of the game this document is describing.

Player - The user, the person who is playing the game.

Identify Angles - Determining if an angle is acute, obtuse, or right.

Supplementary Angle - Defines the angle that could be added to the current angle to make it equal to 180 degrees.

Rod - A fishing rod that the player can buy.

Hub World - A place for the player to walk around in and interact with various aspects.

Currency - The equivalent of money that the player will earn and spend.

1.4 Organization

1.4.1 Organizational Structure of the SRS:

1. Introduction
    1.1. Purpose
    1.2. Scope
    1.3. Definitions, acronyms, and abbreviations
    1.4. Organization
        1.4.1. Organizational Structure of the SRS
        1.4.2. Rest of the SRS Content
2. Overall Description
    2.1. Product Perspective
    2.2. Product Functions
    2.3. User Characteristics
    2.4. Constraints
    2.5. Assumptions and Dependencies
    2.6. Apportioning of Requirements
3. Specific Requirements
4. Modeling Requirements
    4.1. Use Case Diagram
    4.2. Class Diagram
    4.3. Sequence Diagrams
    4.4. State Diagram
5. Prototype
    5.1. How to Run Prototype
    5.2. Sample Scenarios
6. References
7. Point of Contact

1.4.2 Rest of the SRS Content:

The remaining sections of the SRS would provide detailed specifications for each aspect of the application, including:

- Interfaces
- Design Constraints
- Dependencies
- Requirements
- Diagrams
- Prototype
- Additional Specifications

2 Overall Description

Our game Angle Anglers is designed as an educational game involving different kinds of angles, with the goal of helping students expand their angular knowledge. It is a two dimensional semi open world game that involves fishing.

This section will explain the context of the project and describe its interfaces. It also will explain the product's context and functions. The user characteristics will also be described to know what the ideal users of this product will be. The limitations or restrictions that impact the project are also described. Also the assumptions and dependencies, and the apportioning of requirements are included here.

2.1 Product Perspective

Angle Anglers is an educational fishing game that is a standalone application with the goal of educating middle schoolers about angles and their properties. This game requires no integration with any larger system. However it can be used in a classroom setting where each student is playing on their own personal computer or a school's lab computers. The game does not interact with any external sources, and can be run on most basic computers without an issue. The user interface is very simple, you can move around the map with the arrow keys or wasd, also you can hold down the left shift key in order to sprint. There are many points of interest around the map that you can "walk into" and interact with the interface they provide. The key "F" key zooms your view out allowing you to see the entire map. The space bar allows you to fish, but only if you are standing on the end of a dock. Pressing the "esc" key or using your mouse to click the fishing rod in the top left corner allows you to open up the menu to view information. Minimal memory is needed to run Angle Anglers. When actually fishing, users will be prompted with a meter of sorts where you time your cast to create an angle. From there you will have to answer what kind of angle is shown(acute, obtuse, or right), if wrong you will not be able to catch a fish and will have to try again. If correct you will have to find the supplementary angle using mental math.

Students playing our game learn about angles and their properties, they will learn different kinds, which are, obtuse, acute, and right. They also can increase their quick mental math ability as they have to find supplementary angles of the angle given. The game will help students get prepared for homework assignments and even the test that their teacher will eventually give them on angles.

2.2 User Characteristics

The user is expected to be a middle school student between the years of 5th and 7th grade who is enrolled in a general mathematics class. Their skill level in dealing with angles is expected to be low when first playing the game, but after a while of playing their skill level could be expected to go up exponentially. Users also should have a basic understanding of gaming (knows arrow keys are used to move around) so they can easily figure out the map and what to do when first loaded up.

2.3 Constraints

The math topics explored in this game are of the 7th grade level of the Massachusetts Mathematics Curriculum Framework, limiting the game to basic concepts about angles.

The game should be easy to install and utilize as it should be usable and playable by teachers and students with minimal computer experience. As such there should be clear and concise instructions and an easy to navigate interface.

Since the game is meant to be used for education it is imperative to strike a balance between fun and learning, actual teaching and studying should not be forsaken in order to make the game more fun, while it should be entertaining enough to keep middle-school aged children engaged.

The program must be run on a computer using the Windows operating system, with the Windows version either being 10 or newer in order to guarantee faultless playability. The program relies on input from a keyboard and mouse (or trackpad) and is not tested with any other input devices. Should a user make an attempt to modify the game's code or assets there can be no guarantee of its future functionality.

2.4 Assumptions and Dependencies

Some hardware assumptions of the user include that the user has a functional and modern computer along with a keyboard and mouse(or trackpad) connected to their computer. The only software requirement is that the computer that the user is using has a modern operating system, for example Windows 10 or 11. It is also assumed that users of the game have the ability to read English so they can figure out how to play. Another thing that can be assumed is that students playing our game have basic knowledge on how to play video games, meaning they will not get frustrated not knowing what to do once the game is loaded up. The user is also assumed to have basic knowledge on angles.

2.5 Apportioning of Requirements

There are some means of user customization, you have the ability to purchase new fishing rods and new bait. A few things that have been discussed that are beyond the scope of our current project are online features allowing players to play on the same server. This will allow the game to have a social aspect to it which will increase interaction between students if played in the classroom which has the potential to have them be more engaged in the content. Another feature would be multiple towns and map areas that the player can visit to allow for the game to also have exploration elements to increase engagement. These areas would include different towns and potential traveling to places on boats.

3 Specific Requirements

1. The game will require a computer to be played that meets certain minimum requirements.
    1.1. The computer must be able to receive keyboard input.
    1.2. The computer must be able to receive input from a mouse or trackpad.
    1.3. The computer must have a screen to output images onto.
2. The game shall consist of various elements.
3. The game shall have a hub world that the player can walk around in and explore that will offer various features to help the player in the game.
    3.1. The fishing academy will help the player learn the angle concepts needed to play the game along with explaining how the game works.
    3.2. The shop will allow players to spend their currency on various items.
    3.3. The science lab will show players the encounter tables for each fishing area to help show players where they may be missing fish.
    3.4. There will be areas in the overworld where the player can fish, namely piers at lake and the ocean,
4. The game shall have a minigame that involves fishing.
    4.1. This minigame will be started by fishing in the overworld.
    4.2. The minigame consists of four parts:
        4.2.1. Timing a hit in a 180 degree semicircle.
        4.2.2. Reacting to a fish bite.
        4.2.3. Identifying what type of angle was just formed.
        4.2.4. Calculating the supplementary angle for the line.
    4.3. The fish shall have variable properties.
        4.3.1. The species of fish is determined by an encounter table that is specific to the fishing spot.
        4.3.2. The length will be determined based on the speed with which the minigame was completed.
    4.4. The player will cast a fishing rod to start this minigame.
    4.5. After the fishing rod is cast, the game will generate a fish to bite the bait.
    4.6. When a fish bites, audio and visual cues will prompt the player to react in order to start the minigame.
    4.7. Upon successful completion of the minigame, the fish will be caught.
        4.7.1. Its length will be determined based on the time it took to complete the minigame and the faster the bigger length which will give you more currency.
    4.8. The fish will break off the line if the student does not answer within a given time.

5. Multiple areas to fish in with different fish in each.
    5.1. Different areas that can be found around the map which offer different types of fish in each.
    5.2. The fish in each area are determined by an encounter table with some fish being more rare and others more common.
    5.3. Players will have access to a compendium that will record their progress on collecting all of the fish present in the game.
        5.3.1. It will be divided by area and also record the highest length of that fish and the number caught.
    5.4. Each fish will have a small amount of flavor text that provides the player with some trivia about the species.
    5.5. Each area will have a small amount of flavor text to explain a bit about the type of ecosystem.
6. Currency can be gained and spent by the player.
    6.1. Players will gain currency by catching fish which can be used to purchase different items like bait and new fishing rods.
7. Currency will be able to buy different items.
    7.1. Different finishing rods will affect the difficulty of the game.
        7.1.1. Each better fishing rod will have harder numbers and less time to solve.
        7.1.2. Each better fishing rod will provide more rewards and currency than the others.
    7.2. Bait will be purchasable to increase the odds of different fish.
8. The game must have entertaining music in the background.
    8.1. The game must have an option to turn off the background music.
9. The game must be easy to play and learn from.
    9.1. The game must provide “how-to-play” instructions when the player initially plays the game.
    9.2. The game must provide help to the user when stuck by providing the correct answer to the question.
    9.3. The concepts of identifying angles and calculating supplementary angles must be taught and tested.
    9.4. The game must indicate when it is over.
    9.5. The game must have an option to continue if you fail.
10. The game must have an engaging user interface (UI).
    10.1. The game must have an inviting and bright visual style.
        10.1.1. The game shall be colorful and simple.
        10.1.2. The game shall have large, easy-to-understand prompts and menus.

10.2. The game will be a top-down point of view.

11. The game will contain audio and visual cues to aid the player.

11.1. The visual and audio cues must play in sync to avoid confusion when playing with no sound.

4 Modeling Requirements

The following section contains all of the diagrams for the project. These diagrams include a use case diagram, a class diagram, and two sequence diagrams. Each diagram will have supporting descriptions and content to help better describe it.

4.1 Use Case Diagram

The following diagram contains use cases for Angle Anglers. Every use case has a label in the center of the circle and the lines that connect them represent how each of the use cases interact with each other. The primary functions that the player can perform are Fish at Spot, View Science Lab, Buy Item, and View Academy. These are the core functionality of the game. Fish at Spot will allow the user to fish and play the fishing minigame which is the main gameplay loop of our game. This minigame will teach the concepts of angles along and will reward the player currency that they can use to buy new items. View Science Lab and View Academy both provide the player with information. The academy provides information about how to play the game and the underlying math concepts needed to play the game. The science lab provides information about where you can find each fish. Buy Item lets the player use the currency that they have earned to buy new items like fishing rods which will alter the difficulty of the game.

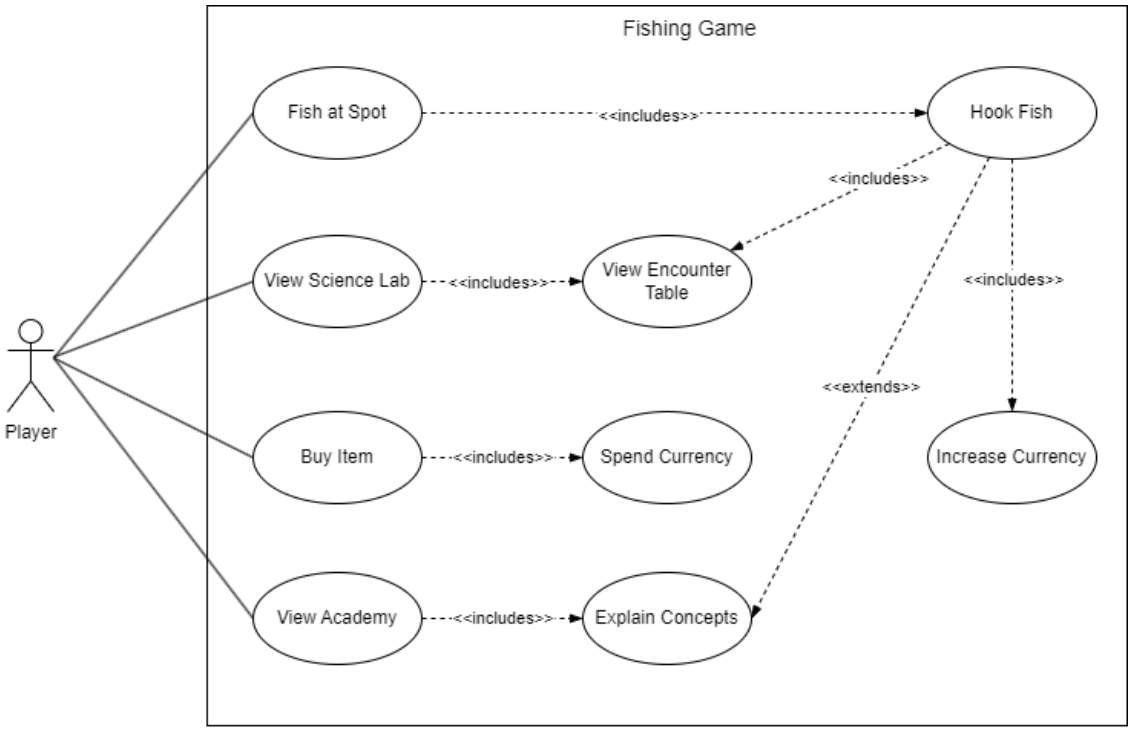

Figure 1 - Use Case Diagram

Use Case Name:	Fish At Spot
Actors:	Player
Description	The player selects one of the fishing spots on the overworld to fish at.
Type:	Primary
Includes:	Hook Fish
Extends:	None
Cross-refs:	Requirement 5
Uses cases:	Hook Fish

Use Case Name:	Hook Fish
Actors:	Player
Description	After selecting a spot to fish in, the user will wait for a fish to bite and once one does they will attempt to catch it. If they are successful they will receive rewards and if they fail their answers will be corrected.
Type:	Primary
Includes:	Increase Currency, View Encounter Table
Extends:	Explain Concepts
Cross-refs:	Requirement 4
Uses cases:	Increase Currency, View Encounter Table, Explain Concepts

Use Case Name:	View Encounter Table
Actors:	System
Description	Retrieves and displays information about the fish appearance rates for the area.
Type:	Secondary
Includes:	None
Extends:	None
Cross-refs:	Requirement 5.1
Uses cases:	None

Use Case Name:	Increase Currency
Actors:	System
Description	Increases the internal value that represents the player's current amount of currency.
Type:	Secondary
Includes:	None
Extends:	None
Cross-refs:	Requirement 6
Uses cases:	None

Use Case Name:	Explain Concepts
Actors:	System
Description	Explain how to play the game, its features, and the underlying math concepts to the game.
Type:	Secondary
Includes:	None
Extends:	None
Cross-refs:	Requirement 9
Uses cases:	None

Use Case Name:	View Science Lab
Actors:	Player
Description	The player enters the science lab which provides information about the fish appearance rates in each area.
Type:	Primary
Includes:	View Encounter Table
Extends:	None
Cross-refs:	Requirement 3.3
Uses cases:	View Encounter Table

Use Case Name:	Buy Item
Actors:	Player
Description	The player purchases different items from the shop which will aid the player and give them unique effects or alter the difficulty.
Type:	Primary
Includes:	Spend Currency
Extends:	None
Cross-refs:	Requirement 7
Uses cases:	Spend Currency

Use Case Name:	Spend Currency
Actors:	System
Description	Decreases the internal value that represents the player's current amount of currency.
Type:	Secondary
Includes:	None
Extends:	None
Cross-refs:	Requirement 7
Uses cases:	None

Use Case Name:	View Academy
Actors:	Player
Description	The player enters the academy which serves the purpose of holding information about the game.
Type:	Primary
Includes:	Explain Concepts
Extends:	None
Cross-refs:	Requirement 3.1
Uses cases:	Explain Concepts

4.2 Class Diagram

The following diagram contains information about the classes that make up Angle Anglers. Their name, fields, and methods are all outlined. All of the different classes that make up the game revolve around the player and their interactions with them. The player itself holds information about the player like profile name and currency and also has the actions of casting their fishing rod and reeling in a fish. One of the things that the player can interact with is fishing spots. These spots are where fish can be caught. Fishing spots all have an associated encounter table which holds the information about the probabilities of finding each fish in that area. Encounter tables are affected by a difficulty which is determined by the player's fishing rod. The difficulty will affect the numbers generated for a problem along with increasing probabilities when the difficulty is higher.. The next thing that the player can interact with is the Store. The Store handles all purchases that the player may make and holds items that the player can buy. These items are from a common interface which holds basic information about that item type. The two item types that inherit from this interface are the Rod and Bait classes. The Rod class contains an effect which is a mapping to a difficulty. The bait class contains the type which is the rarity that it will effect and a reroll amount that it offers. The final thing that the player can interact with is the Compendium. This is used to keep track of game progress that the player makes like statistics of their gameplay and the amount of fish that they have caught. The Compendium holds information about a Fish Species which is a class that represents each kind of fish and holds various information about them.

4.2.1 Class Diagram

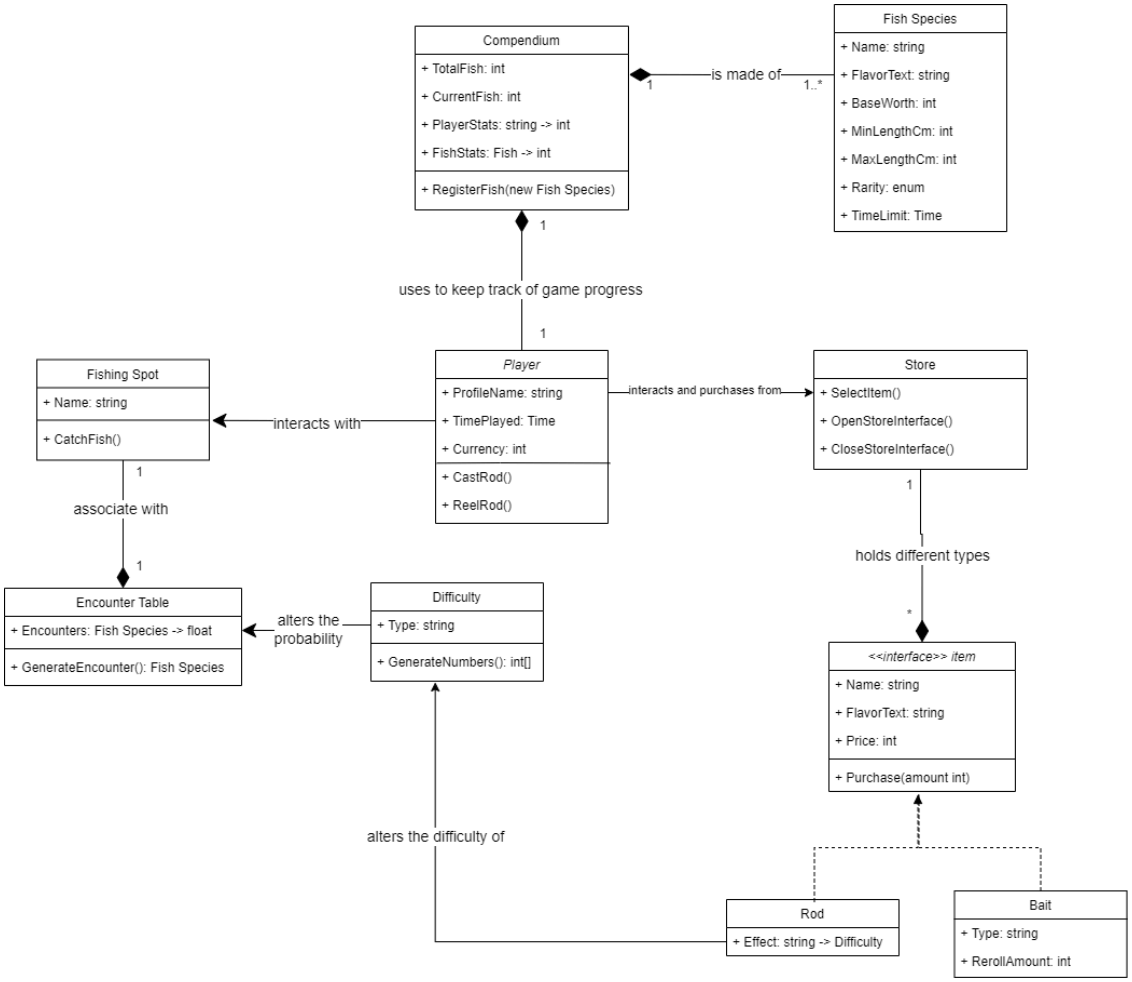

Figure 2 - Class Diagram

4.2.2 Class Descriptions:

Class Name	Player		
Description	Represents the player and their actions.		
Extends	None		
Attributes	ProfileName	string	The player's profile name that they assign to themselves.
	TimePlayed	Time	The amount of time that the player has played the game for.
	Currency	int	The amount of currency that the player has.
Operations	CastRod()		Casts the players fishing rod in the area that they are standing.
	ReelRod()		Reels in the players fishing rod.

Class Name	Fishing Spot		
Description	Represents a spot where the player can fish.		
Extends	None		
Attributes	Name	string	The name of the fishing spot.
Operations	CatchFish()		Generates a fish for the player to catch.

Class Name	Encounter Table		
Description	Represents the probabilities of the fish that can be caught in the fishing spot.		
Extends	None		
Attributes	Encounters	Map<Fish Species, Float>	Maps a Fish Species to a probability.
Operations	GenerateEncounter(): Fish Species	Generates a fish species based on the table.	

Class Name	Difficulty		
Description	Represents the difficulty of the game.		
Extends	None		
Attributes	Type	string	The type of difficulty.
Operations	GenerateNumbers()	Generates numbers for the questions based on the type of the difficulty.	

Class Name	Store	
Description	Represents the store where the player can buy different items.	
Extends	None	
Operations	SelectItem()	Select an item to purchase from the store.
	OpenStoreInterface()	Open the store GUI that displays the information and items available at the store.
	CloseStoreInterface()	Close the store GUI.

Class Name	Item (interface)		
Description	An interface that acts as a basis for different item types.		
Extends	None		
Attributes	Name	string	The name of the item.
	FlavorText	string	A short description of the item that is displayed to the player.
	Price	int	The amount of currency that the item costs.
Operations	Purchase(amount int)		Purchase an X amount of that item.

Class Name	Rod		
Description	Represents a fishing rod that the player can use. Different fishing rods have different effects on the difficulty.		
Extends	Item		
Attributes	Effect	Map<string, Difficulty>	A mapping of the rod type to a difficulty.

Class Name	Bait		
Description	Represents bait that the player can use and buy. Bait can be used to increase the rerolls on finding a certain rarity of fish.		
Extends	Item		
Attributes	Type	string	The rarity that the bait increases a chance for.
	RerollAmount	int	The amount of rerolls the bait provides.

Class Name	Compendium		
Description	Represents a collection of data about the player's gameplay.		
Extends	None		
Attributes	TotalFish	int	Represents the total amount of fish in the game.
	CurrentFish	int	The current amount of those total fish that the player has caught.
	PlayerStats	Map<string, int>	A table of different statistics about the player. For example, total fish caught, total currency earned, etc.
	FishStats	Map<Fish Species, int>	A table of different statistics about each species of fish. For example, the number of this type caught.
Operations	RegisterFish(new Fish)	Registers a new fish to the compendium.	

Class Name	Fish Species		
Description	Represents a species of fish.		
Extends	None		
Attributes	Name	string	The name of the species.
	FlavorText	string	A short little description of the fish.
	BaseWorth	int	The base amount of current that the fish is worth.
	MinLengthCm	int	The minimum length in centimeters that the fish can be.
	MaxLengthCm	int	The maximum length in centimeters that the fish can be.
	Rarity	enum	The rarity of the fish that can be from Common, Rare, or Legendary
	TimeLimit	Time	The amount of time before the fish gets away while attempting to catch it.

4.2.3 Relationship Descriptions:

Player purchases and items from the store - The player uses their currency to buy and purchase different things that inherit from the item interface from the store.

The store holds different types of Items - The store holds different types of items that it has available for purchase.

Compendium is made of Fish - The compendium holds information about the fish that the player has caught like whether they have found it or the total amount caught.

Player uses Compendium to keep track of game progress - The compendium holds a lot of information about fish and player statistics which shows the player their progress through the game.

Player interacts with Fishing Spot - The player interacts with a fishing spot to play the fishing minigame which is the main gameplay of the game.
Fishing Spot associates with an Encounter Table - Each fishing spot will have an encounter table mapped to it to make them each different. Certain fish may be available in one but not another.

Difficulty alters the probability of Encounter Tables - The difficulty will make the game hard but also make more rare Fish in the encounter table have a higher chance of appearing for a risk reward aspect.

Rods alter the difficulty - The current rod that the player has equipped will alter the difficulty.

4.3 Sequence Diagrams

The following sequence diagrams represent two different processes that are a part of Angle Anglers. The first is the process for catching fish and the other is the process for catching an item. Both subsections will contain the sequence diagram itself and then a description describing the process.

4.3.1 Catching a Fish

This sequence diagram shows the process of the player catching a fish. The player begins by casting their rod which they then wait for a fish to bite. After one has bit, they need to reel in the fish. If they successfully react in time, the system will start the minigame which requires generating a problem that is based on the difficulty of the game. Then, the player is given the problem which they are then required to solve. If they solve it successfully a fish is then generated and returned to the player to be used to update their stats, collection, and increase their currency.

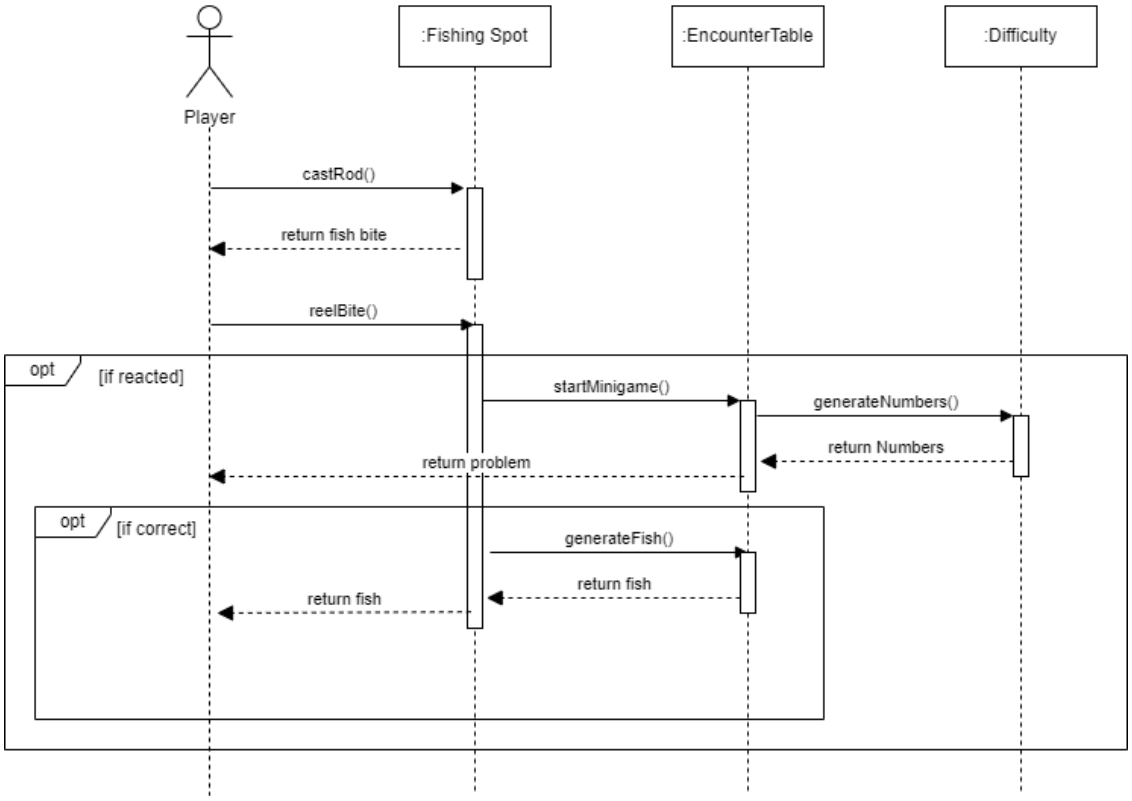

Figure 3 - Catching a Fish Sequence Diagram

4.3.2 Buying an Item

This sequence diagram shows the process of a player buying an item from a shop. The first thing that happens is the player triggers opening the store interfacing. This could be done by interacting with something or entering the shop building. Once the interface is open the player will select an item that is available in the store. The selected item will be purchased if the player has enough currency and once the selected item is purchased it is then returned to the player to check if they have it. After that the player can choose to close the store interface or buy another item.

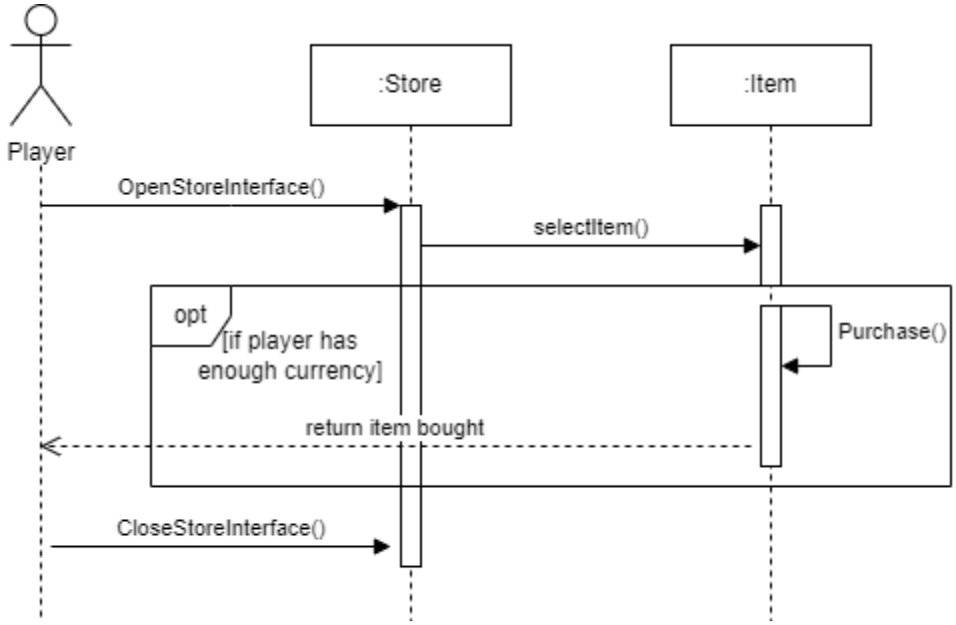

Figure 4 - Buying an Item Sequence Diagram

4.4 State Diagram

Below is a state diagram that describes the flow of the states that the player will be in while playing the game. The player starts on the main menu and then can either choose to quit the game or start it. The player has some options while walking around the world. The first is the main gameplay loop. fishing. The player starts by preparing to cast a rod and if they do that successfully the next states revolve around the questions that are given. If correct they move on and if they are incorrect then the correct answer will be displayed. If a fish is successfully caught it will be displayed and then the player is returned to either casting a rod or just walking away. The other major function is entering a building which will display an interface for the contents of that building. The academy and science lab both have information displayed and then the player can exit, however the store allows the player to buy items as well. The other thing that can be done is pausing the game. When you pause a pause menu will be displayed which allows for going back to the main menu or just exiting the game.

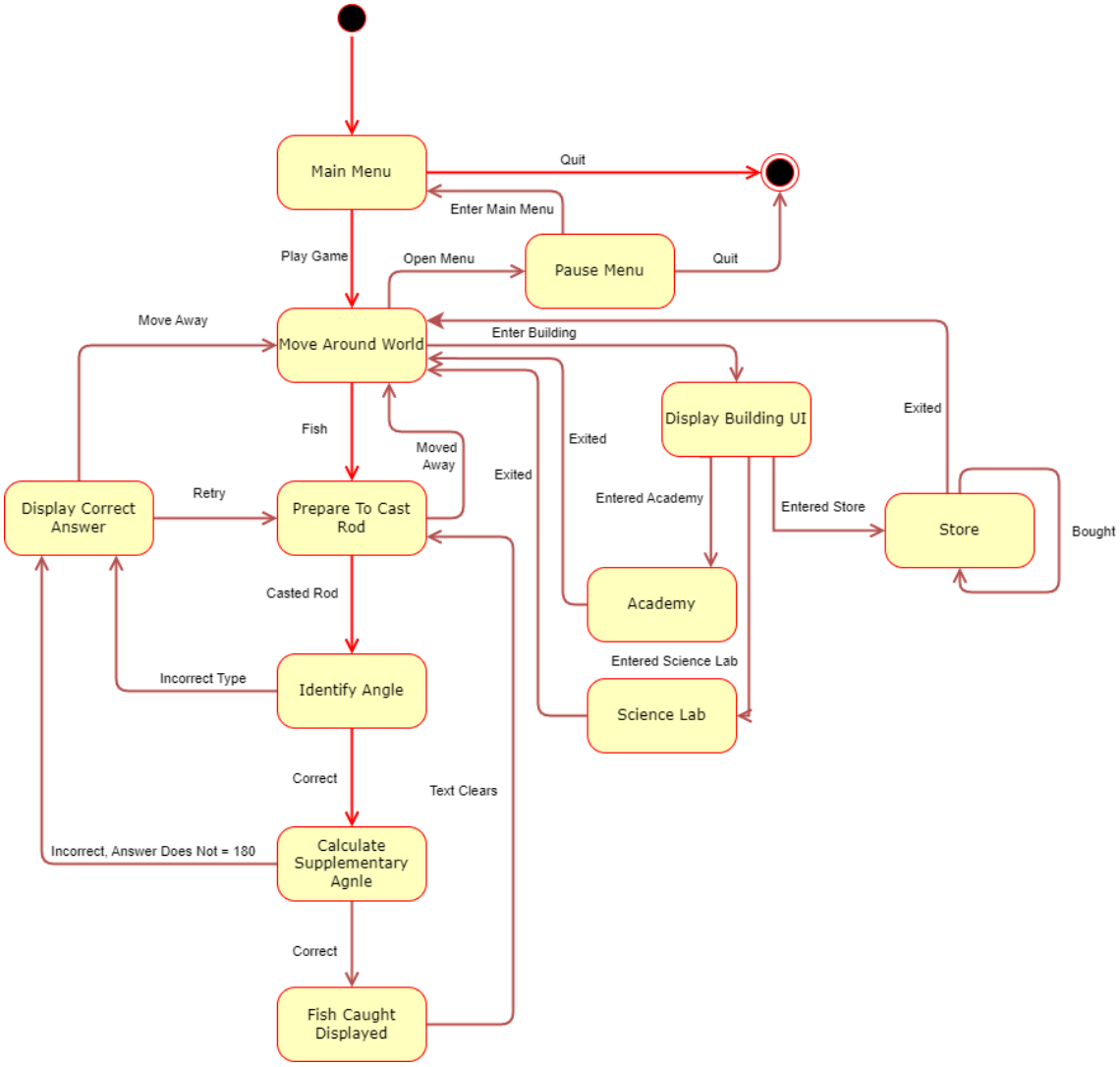

Figure 5 - State Diagram

5 Prototype

The prototype of Angle Anglers will include a main menu and a small portion of the game. The player will be able to walk around a small hub world with a few features implemented. The player will have the ability to also run to move at a faster pace and zoom out the view of the whole map. The features that the player can interact with is fishing which provides the basic fishing minigame and multiple buildings the player can enter. The fishing minigame will have the player cast their line and attempt to line up a line with a certain spot on a semicircle. After this they will wait and react to a fish biting their line. After that they will be asked to identify the angle and find its supplementary angle to be successful at catching the fish. Any wrong answer will result in the fish getting away. There will be two areas for the player to fish in a pond and a beach dock which will offer six different fish per area. There will also be three buildings for the player to visit. The academy which will offer how to play the game screen. The science lab which will include information about the encounter rates for fish. And finally, the store which will allow the player to buy new fishing rods with currency that they earn from fishing. These fishing rods will also increase the rewards you can gain but make the numbers harder.

5.1 How to Run Prototype

The game can be downloaded from our website (https://angle-anglers.github.io/download.html) under the downloads table. Once there click on the downloads button and it will take you to a Google Drive link with the game zipped. Unzip the game and do not remove any of the folders or files; it may break something. Once unzipped open the AngleAnglers.exe file and enjoy the game. At the current moment the game is only compiled for the Windows operating system.

5.2 Sample Scenarios

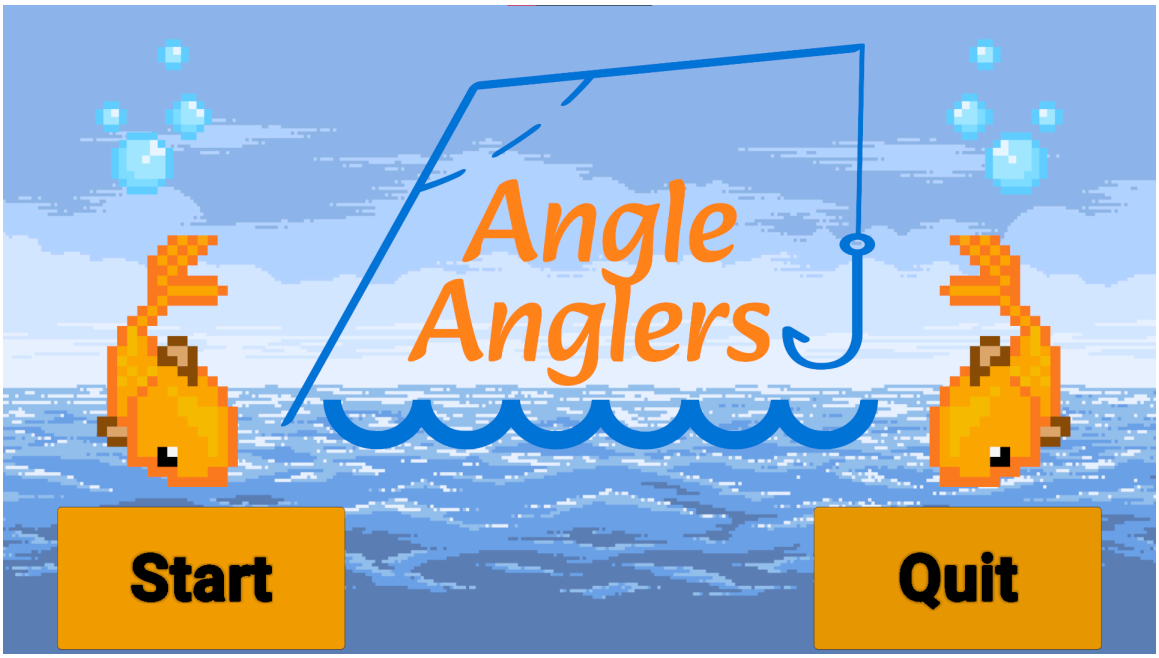

Figure 6 - The main menu of the game

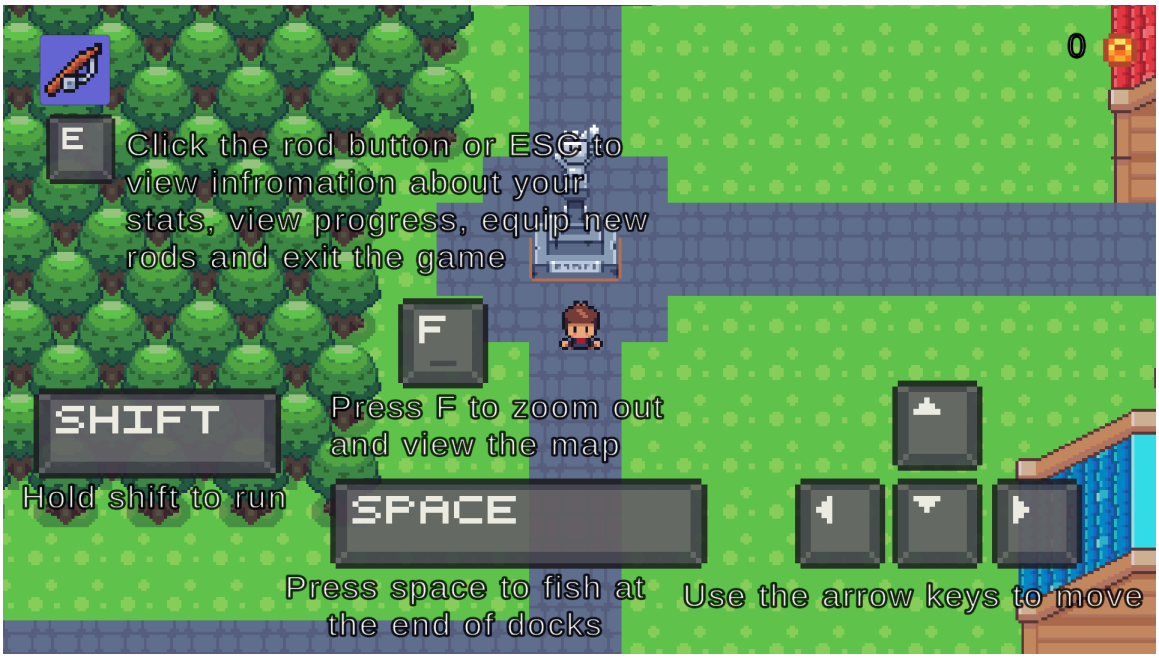

Figure 7 - An overlay that displays the controls when you first load into the game

Figure 8 - The games pause menu (Will be updated for V2)

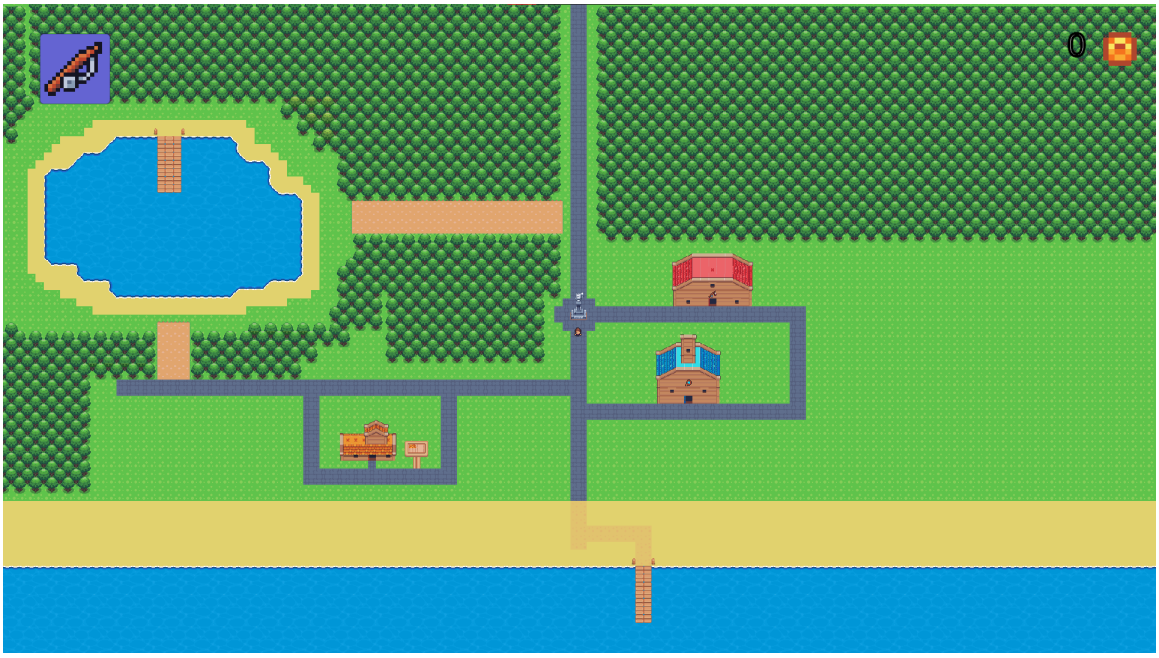

Figure 9 - A zoom out of the map

Figure 10 - The store where the play can buy new rods (Will be updated for V2)

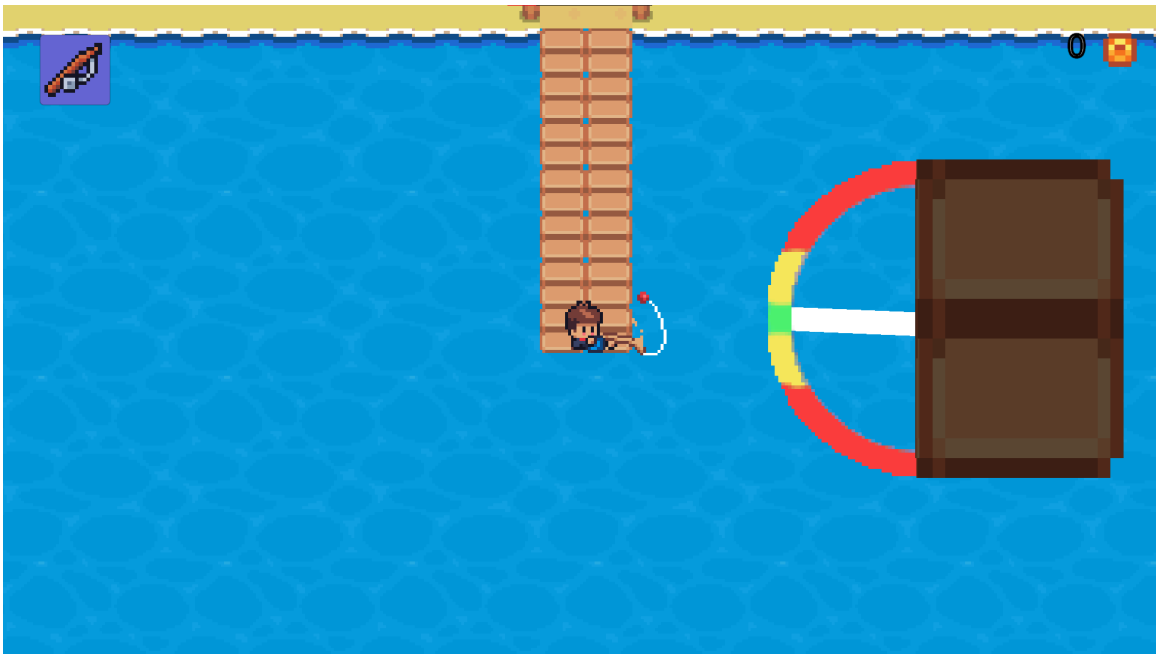

Figure 11 - The player casts their rod on the beach

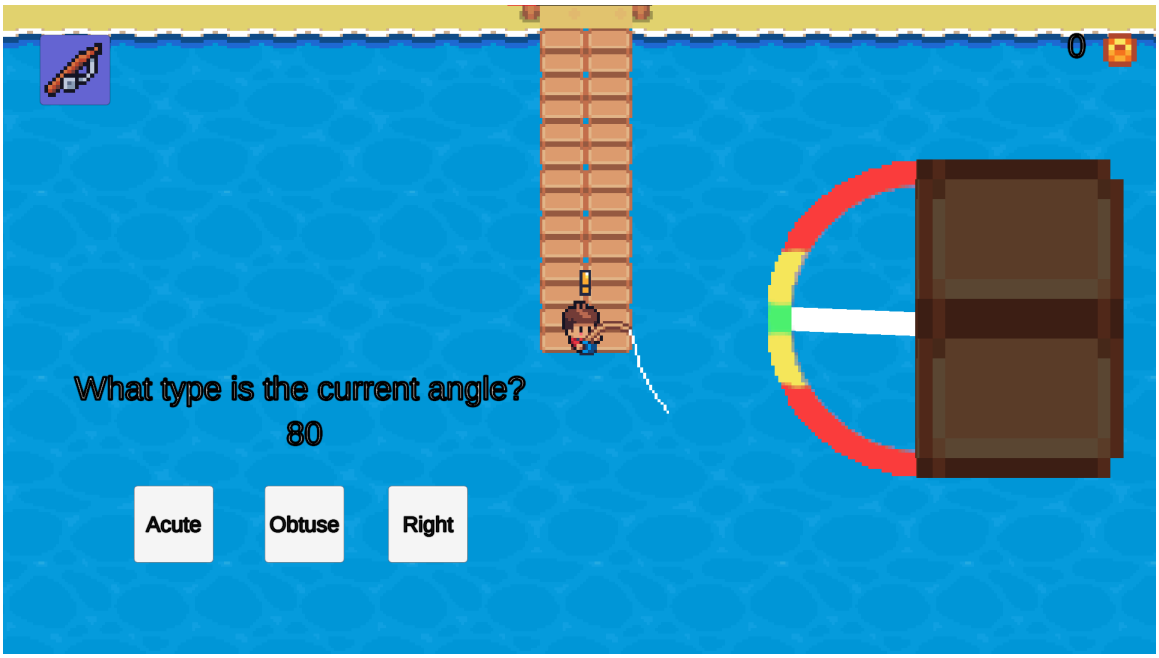

Figure 12 - The player identifies an angle while catching a fish

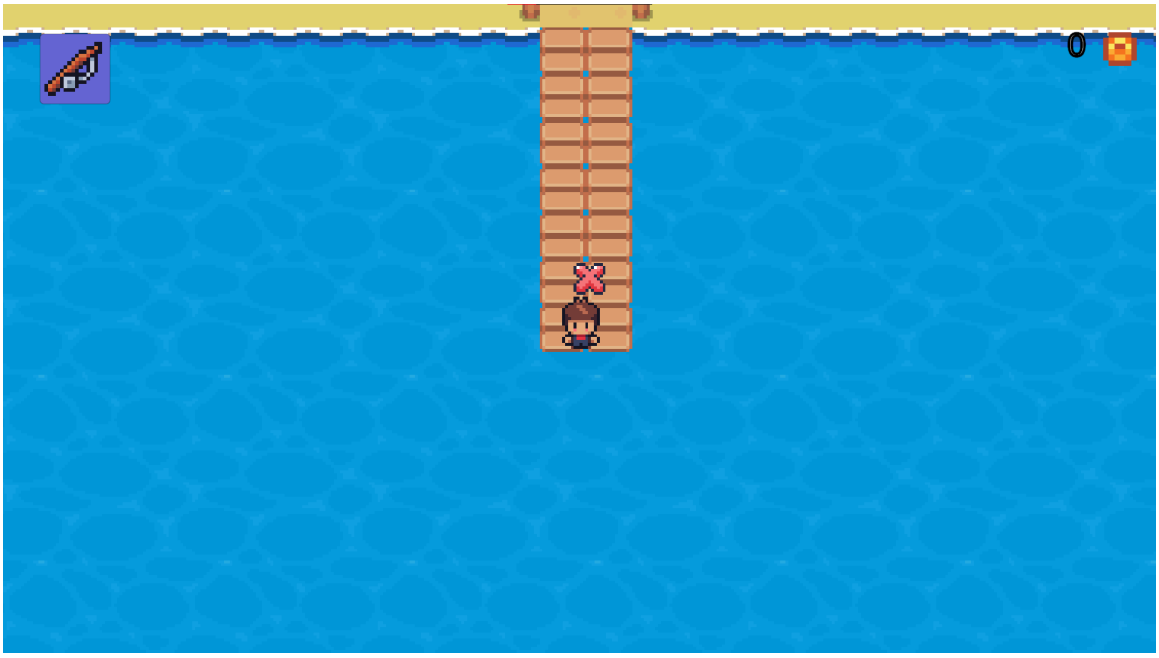

Figure 13 - The player got the wrong answer

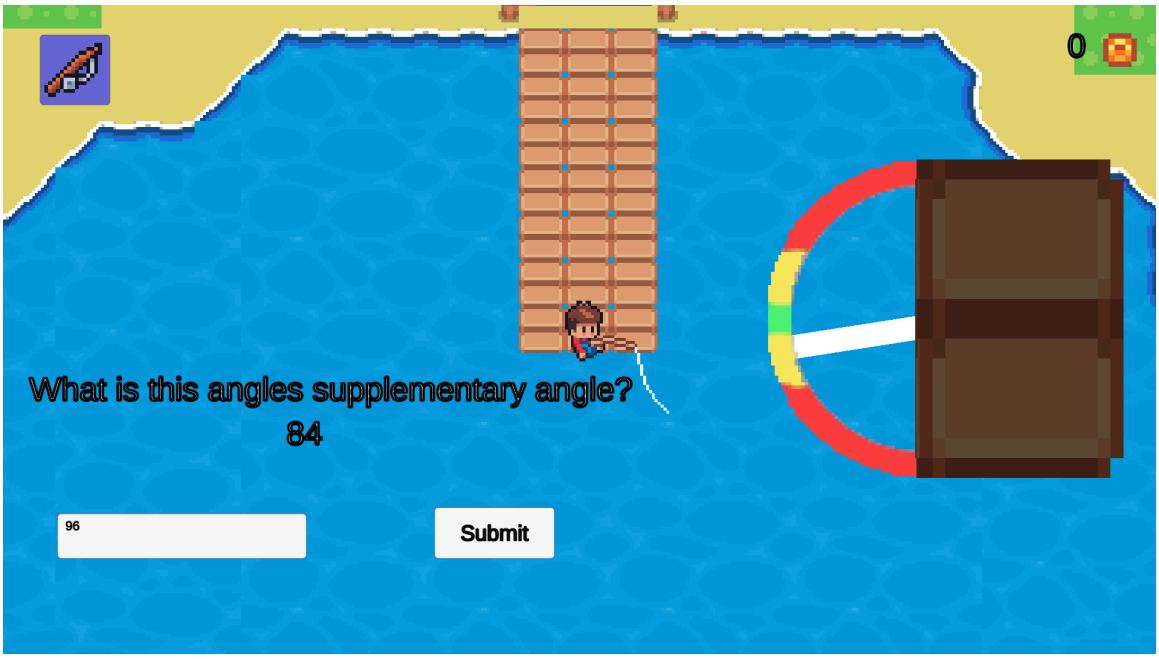

Figure 14 - The player finding the supplementary angle

Figure 15 - The player successfully catches a fish

6 References

[1] *Massachusetts Curriculum Framework for Mathematics,* Massachusetts Department of Elementary and Secondary Education, 2017, https://www.doe.mass.edu/frameworks/math/2017-06.pdf

[2] E. Casasanta, C. Coco, S. Gyger, V. Patel, *Angle Anglers*, Software Engineering Group 8, 2024, https://angle-anglers.github.io/index.html

[3] *Unity Documentation,* Unity Technologies, 2005-2024, https://docs.unity.com

[4] *Ocean Background,* KnoblePersona, 2016, https://opengameart.org/content/ocean-background

[5] *Simple Keys*, beamedeighth (Itch.io), 2022, https://beamedeighth.itch.io/simplekeys-animated-pixel-keyboard-keys

[6] *Sunnyside World*, danieldiggle (Itch.io), 2024, https://danieldiggle.itch.io/sunnyside

7 Point of Contact

For further information regarding this document and project, please contact **Prof. Daly** at University of Massachusetts Lowell (james_daly at uml.edu). All materials in this document have been sanitized for proprietary data. The students and the instructor gratefully acknowledge the participation of our industrial collaborators.