Game Programming Laboratory

Curse of Blobfish: A Submariner's Tale

(Provisional title)

"Think fast or drown !"

TEASER

@Note: Each member should be assigned one or several roles according to the preferences and skills. Respect these two rules: (1) Everyone is a Game Designer; (2) There is one Producer.

Yan Walesch - Producer ,Programmer, Game Designer, QA Engineer
Henry Trinh - Programmer, Game Designer, Visual Artist
Ellen Arlt - Programmer, Game Designer, Visual Artist
Marino Müller - Game Designer, Programmer
Stefan Jokić - Game Designer, Programmer, Visual Artist

Contents

1

Formal Project Proposal

(Max 10 pages)

@Note: Use this chapter for Rough Draft Proposal/Final Proposal

@Note: A formal game proposal makes up the first chapter of your project notebook. The game proposal describes your game idea, provides a detailed development schedule, and gives a qualitative assessment of your project. The proposal should be professionally prepared, expressive, grammatically sound, illustrative of your efforts and process, and easy to understand. A good design effort can easily be hampered by a poor communication of what was done

1.1 Game Description

1.1.1 Overview

Our game will be a 2D retro style (pixel art) level based Platformer. The main idea is to solve puzzle in each level, collect collectibles and fight an end-boss at the end of each level. The game will be played as an two-player co-op (horizontal split-screen)Puzzle solving, Minigames Two different environments (Typically, one player in the submarine and one under water) Main objectives:

- – Submarine: solve minigames to aquire oxygen to keep player underwater alive, help with fights and certain puzzles.
  – Underwater: Solve puzzles and collect as many fisheggs as possible
- Limited time (each crewmate has a limited oxygen supply underwater)
- different characters to choose from with different unique abilities (characters can be

changed at certain points)

- Simple combat (+ end game boss) (e.g. jumping on small enemies, simple shooting attacks of certain characters, picking up and throwing starfish, . . .)

1.1.2 Background Story

In an exciting expedition to the bottom of the sea a crew of four members has been entrusted with the task to locate and secure a vast collection of rare fish species the likes of which humanity has never seen before and might very well be the last of their kind and transport them to a specially constructed safe habitat to save these species in the face of global warming and water pollution. Having caught various species of intelligent and excitingly strange looking fish and, what the four believe to be the most prized of their collection, a tremendous amount of rare fish eggs found in the deepest parts of the sea, the crew is on their way back to the surface.

Then it happens! Someone spills coffee on the controls of the submarine, causing the submarine to malfunction and lose course for just a long enough amount of time for the submarine to scrape a large rock leaving a gaping hole in the submarine through which all the painstakingly collected fish flee and what's more, all the rare fish eggs are dispersed into the sea! Having no capacities to store the fish anymore since all the tanks are now broken, the crew wants to recover at least the fish eggs to ensure their mission wasn't a complete failure. But now the fish are warned and hostile towards the crew and even managed to bring the local fish to their side. Will the crew be able to conquer the dangers of the ocean floor in order to retrieve the eggs and bring them to the promised location?

To secure all eggs the crew will have to face a small number of dangerous fish bosses the final one of which turns out to be mother of the eggs!

1.1.3 Design Decisions

Our game is a two player cooperative adventure 2D platform game with horizontal split screen. The core mechanic consists of solving puzzles in a cooperative fashion in two different environments to progress the story. In addition the game also provides a simple combat system, end bosses at each level and multiple characters with special abilities. To enhance the game experience for the players, we give a time control feature to each player in form of oxygen tanks. During playing one player is responsible for the survival of the person underwater by refilling the oxygen tank. The players succeed when they manage to solve the puzzle or boss fight before the oxygen runs out.

Level Design

As mentioned briefly before, through the entire game we play in two different environments, namely the submarine and the underwater world. The submarine consists of following rooms:

- Oxygen station: Solving mini puzzles here provides underwater person oxygen.

- Battle station: Room where the person assists in boss fight (by e.g. controlling a cannon)
- Light station: Room where the person controls a light beam so that the person under water can see in dark abyssal zones.
- Potentially other rooms with easter eggs (e.g. storage room, control room, etc.)

The levels are taking place in underwater volcanoes, coral reefs, underwater forests, deep seas or contaminated wasteland. Each level is composed of different stages with checkpoints where the players can reset upon failure. In addition the players can also switch during the checkpoints.

Puzzle Integrated Gameplay in Cooperative Fashion

The game has Among Us features where a small window pops up with a puzzle which are mainly located inside the submarine environment. Some of those games can be e.g. riding a king crab while avoiding obstacles, sliding puzzles, memory games, simple mini shooting games etc. Solving them provides the other player with light in dark locations, unlocks closed doors, oxygen refill, giving them more move speed or weakening bosses etc. On the other hand we provide classical obstacle-based puzzles for the player underwater e.g. pushing stones in a way to unlock doors, building platforms out of objects in environment to jump on or killing NPCs to unlock areas. We design the puzzles such that they are solvable by all character types and both player needs to participate to progress the story. Further each characters has special abilities making it easier to solve certain kind of puzzles or have advantages fighting NPCs and bosses.

Characters

Each character has the following basic abilities:

- Jump (underwater: ascend)
- Simple attacks which varies on the type of game character.
- Picking up and throwing items
- Crouching (underwater: descend)

Mad scientist:
His simple attack is a laser gun with a short cooldown. The mad scientist is able to transform himself into half-human/half-fish for a limited amount of time with certain cooldown. During this spanning time the scientist does not require oxygen.

Athlete:
The athlete has increased movement speed, can jump farther and deals more damage. His simple attack allows him to jump on monsters and kill them. He has more oxygen than other characters.

Engineer:
The Engineer has a hacking device which solves small puzzles (in a pop-up window) to disable a trap. The device must be plugged into the respective connector of the trap that he/she wants to

disable. Can also disable multiple traps at once. Stuns enemies for a short duration by zapping them with a taser. Furthermore, he/she can build bridges and other structures to overcome obstacles.

Security Guard:
The Security Guard can use a harpoon to fight (deals most damage out of all). He can grab monsters and throw them. It takes more hits from NPCs to take the security guard down.

NPC and Boss Design

Our game implements different kinds of NPC depended on the environment where the monsters have to be avoided, killed or used to overcome puzzles or obstacles. **Types of boss fights:** The player on the submarine has to distract the boss by solving puzzles. During this time the underwater player has to avoid attacks from the boss. If the boss is successfully distracted the underwater player can do damage to the boss.

1.2 "Big Idea" Bullseye

The most important aspect of this game is the idea of providing a cooperative gameplay while the players operate in different environments most of the time. This concept provides a unique cooperative gaming experience, since the playstyle of each individual player might change the outcome of the game. Furthermore, it encourages communication and cooperation between both players, since tasks in both environments need to be fulfilled to solve the puzzles and ensure the players' survival. Hence, a solid task management and fundamental trust are expected from the players to complete the game, especially due to the fast-paced mini-games and the time pressure provided by the oxygen/health system.

The ability to choose between different characters and change the characters during your playthrough describes a supporting technical component that further encourages the idea of a unique gaming experience. The playstyles of the different players become even more decisive during the game and even provide an intention for the players to replay the story and find different ways to solve the puzzles during the game as most puzzles will be solvable by multiple characters using different character abilities. Changing your team partner could therefore also change your experience.

1.3 Technical Achievement

One core technical element of this game is the inclusion of self generating puzzle/mini-games during the game. As described in previous sections, players need to refill their oxygen tanks during the playthrough by solving time-based mini-games. Using procedural content generation techniques, these mini-games should be newly generated every time a player attempts solving them. This feature should further increase the replayability of the game.

Furthermore, the game should adjust the difficulty of the mini-games to match the players' skill level. The level of difficulty for the corresponding puzzle will be decided using a reinforcement learning algorithm. Hence strong players with lots of experience in puzzle solving should get more difficult mini-games, while beginners or casual players should receive easier puzzles. The level of players might be decided by different features such as time used in previous puzzles, number of steps used during the solving process or the ratio between solved and attempted puzzles. This mechanism ensures that players should neither get bored nor frustrated solving the puzzle.

1.4 Development Schedule

1.4.1 Layered Task Breakdown

@Note: You can't accurately anticipate how long each step in your project is going to take. Consequently, you need to make a detailed development schedule that is layered.

Functional Minimum

- Create two different environments (submarine and underwater world)
- Submarine: popup puzzle games (mini-game) for oxygen station (using predefined games)
- Underwater: Build 1 level with simple NPC enemies and puzzles
- Underwater: Add collectable eggs to the levels and associated counter
- Develop 1 playable character with basic abilities
- Create an oxygen/health system, which lowers the oxygen level over time, if the player is not inside of the submarine.

Low Target

- Submarine: add multiple other Stations and their respective functionalities (i.e. battle station allowing players to drop bombs/use a machine gun, control station allowing to control an outside spotlight, switch station allowing a player to leave the submarine, ...)
- Underwater: create darker sections allowing players to make use of the newly added spotlight functionality
- Add a bossfight at the end of the first level
- Add a second playable character and develop different basic abilities for each of the characters.

Desired Target

- Make use of procedural content generation techniques to generate the mini-games for the submarine oxygen tank station.
- Develop multiple(2-3) levels with different enemies and puzzles, including a bossfight at the end of each level.
- Create an additional 2 playable Characters and add more complex/ special abilities for each character.
- Add simple sound effects such as a background theme
- Add a starting menu, as well as a pause/resume and a save/load function.
- Give the players the ability to switch the used character.

High Target

- Add more sound effects, such as battle theme and attack or dying sounds
- Make use of reinforcement learning to adapt the difficulty of the mini-games at the oxygen tank station to the players skill level
- Develop special abilities for some NPC (i.e. sonar for dolphins to confuse players for short amount of time, blasting ink for octopuses decreasing a player's field of view if hit, etc...)
- Add small movements to the background
- Add grabbable and consumable items, i.e. throw starfish as shuriken, health positions, etc...
- Underwater: implementing a reward/achievement function regarding the number of collected eggs (i.e. certain amounts of eggs unlock an item/character/cutscene, etc.)

Extras

- Add more levels & characters
- Use advanced AI for NPC and bossfights
- Compose own music and sound effects :D
- Include more complex puzzles e.g. integrating camera control or blurring mechanisms

1.4.2 Task List

WIP

Task	Description	Who	Hrs	Actual
1	Character Design	Henry		
2	Environment Design (underwater)			
3	Environment Design (submarine)			
4	Boss Design			
5	NPC Design			
6	Implementation of submarine stations			
7	Physics			
8	Puzzle Design			
9	Puzzle Implementation			
10	Self generating mini-games			
11	Battle mechanic implementation			
12	Character Implementation			
13	Sound			
14	Trailer			
15	Poster Design			

@ Note: Provide a table showing who is responsible for each task, how many hours will each task require, etc.

1.4.3 Timeline

@ Note: Provide a Gantt chart when each task will be started and finished, etc.

TODO

1.5 Assessment

- The focus of the game is fast-paced puzzle solving and cooperative interaction between the two players in different environments. The strength of the game is that it adapts to each player's preferred playstyle (are they mechanically skilled? or are they better at solving puzzles?) by means of different characters with unique abilities that allow to progress through the levels in different ways (e.g. jump over a trap or solve a puzzle to disable it) but also by means of the two different environments (solving mini-puzzles in the submarine vs solving puzzles that may involve movement and fighting NPCs).

- The game is targeted towards players who enjoy solving puzzles which should ideally not be too complicated, but still require some sort of ingenuity. It is also targeted towards players that enjoy playing in a coordinated manner with another player.
- The players should feel time pressured and try to work together to progress. The failure of one player will cause the other player to fail as well. If the player in the submarine doesn't supply the player under water with enough oxygen, he/she will suffocate and the level must be restarted. This can happen if any of the two, or both players, is not fast enough in solving the puzzles. We hope that this aspect gets the players more immersed in the game and motivates the players to communicate and work well with each other. As a consequence, however, players may blame each other when they fail and they are unsure who is at fault. Such discussions between the players can be fun, but we certainly hope that it doesn't break any friendships ;) The puzzles should also be fun to solve, have sufficient diversity between them and an adequate difficulty (perhaps dynamically adapting to the player's skill, at least with regard to the mini-puzzles in the submarine). We also want to encourage cooperative interaction between the players even further by implementing tasks in different levels that, for instance, involve having the player in the submarine control a light beam so that the player in certain dark parts under water can progress through the level, or having the player in the submarine control a cannon that can fight off NPCs under water.

2

Prototype

(Min 3, Max 5 pages)

@Note: The key goal of this part of the project is to develop a prototype of your game that distills out the core game play. The prototype should incorporate the game mechanics while providing only a crude approximation of other features like artwork.

2.1 Prototype Setup

@Note: Include sketches and photos of your prototype in such a way that you can demonstrate how the prototype works and how the gameplay is modeled. How did you model environment, characters, and other features of the game?

2.2 Playing Experience

@Note: Your experience playing the game. Was it fun?

2.3 Findings and Conclusion

@Note: Explain what you have learned from creating the prototype. What has proved to be harder (or easier) than expected? What design revisions have you made to your game based on your experience creating the prototype?

3

Interim Report

(Max 5 pages)

3.1 Progress

@Note: Describe how many layers you have finished. You can include screen shots to help explain your game so far, and text to describe how a user would interact with it. Our hope is that you have completely finished layer 2 and are well into layer 3.

3.2 Challenges

@Note: Explain what has proved to be harder (or easier) than expected. What design revisions have you made to your game as a result of what you've learned with the implementation? Discuss the implementation challenges you faced. Were there aspects that you wanted to build but were unable to do so?

3.3 Future Work

@Note: What are the planned tasks that will be implemented next? Shortly explain.

4

Alpha Release

(Max 5 pages)

@Note: Follows the same guidelines as the interim report chapter

4.1 Progress

@Note: Comment on how far you have progressed and show us what is exciting about your game. Ideally, you will have met the goals outlined in layer 3 (your desired target) and possibly part or all of layer 4 (your high target). You can include screenshots.

4.2 Challenges

@Note: Explain what has proved to be harder (or easier) than expected. What design revisions have you made to your game as a result of what you've learned with the implementation? Discuss the implementation challenges you faced. Were there aspects that you wanted to build but were unable to do so?

4.3 Future Work

@Note: What are the planned tasks that will be implemented next? Shortly explain.

5

Playtest

(Max 5 pages)

5.1 Playtesting Session

@Note: Describe who you recruited for playtesting and how you organized the playtesting sessions. If possible, include some photos.

5.2 Questions and Comments

@Note: List the questions you chose to ask the testers. Summarize their answers. Comment on overall trends you learned from the exercise, as well as any specific suggestions that were particularly useful.

5.3 Design Revisions

@Note: Finally, describe any changes you made to your game based on the playtesting.

6

Conclusion

(Max 5 pages)

6.1 Final Results

@Note: In this chapter, first provide a summary of your final results including screenshots from your final game. Comment on any significant changes from the alpha release.

6.2 Experience

@Note: Here you should provide commentary about your experience during the class. How well did your initial design ideas materialize into the final game. Were you able to follow your development schedule, or did you deviate significantly from it? How did the different elements of the project structure (development schedule, prototype, playtesting, etc.) contribute to or hinder your progress?

6.3 Personal Impressions

@Note: Did it meet your expectations? Are you happy and proud of your game? Do you feel there wasn't enough time or that the schedule was too compressed?

@Note: You might also consider these questions:

- What was the biggest technical difficulty during the project?
- What was your impression of working with the theme?
- Do you think the theme enhanced your game, or would you have been happier with total freedom?
- What would you do differently in your next game project?
- What was your greatest success during the project?
- Are you happy with the final result of your project?
- Do you consider the project a success?
- To what extend did you meet your project plan and milestones (not at all, partly, mostly, or always)?
- What improvements would you suggest for the course organization? (Perhaps in D1 evaluation)?
- Did you like using MonoGame?