Software Engineering Project Report

Depths of Fear

University of Illinois Chicago

October 2024

Whitney Le
Stephen Harayo
Tuan Dinh
Walter Flores

Table of Contents

List of Figures

List of Tables

I Project Description

Depths of Fear Is a horror survival game set in the mysterious depths of the ocean. Players take on the role of an ocean researcher exploring both underwater and on-land environments, uncovering the dark secrets of an abandoned marine research project that has led to a genetic mutation of sea creatures. The player must manage resources like oxygen, evade predators, and solve puzzles to survive.

The game combines exploration, survival, and horror while raising awareness about real-world ocean conservation issues such as pollution, overfishing, and climate change. Through immersive gameplay, great storytelling, and great visuals of the deep sea, the game offers both a horror and educational experience as players duel the danger of lurking in the abyss and the consequences of human pollution.

1 Project Overview

This project is being developed to show the potential of ocean-based horror games while bringing awareness of the critical state of our oceans. The deep sea is a perfect setting for a horror game that can bring fear, isolation, and suspense, making it a unique experience compared to traditional horror games. By combining real-world environmental themes, the game aims to bring an entertaining experience while educating players about the fragility of marine ecosystems and the impact of human activities, such as pollution and overfishing, on ocean life.

2 The Purpose of the Project

This project is being developed to show the potential of ocean-based horror games while bringing awareness of the critical state of our oceans. The deep sea is a perfect setting for a horror game that can bring fear, isolation, and suspense, making it a unique experience compared to traditional horror games. By combining real-world environmental themes, the game aims to bring an entertaining experience while educating players about the fragility of marine ecosystems and the impact of human activities, such as pollution and overfishing, on ocean life.

2a The User Business or Background of the Project Effort

This project is being developed within the game development industry, specifically focusing on horror-survival games. The game development business is centered on creating immersive and interactive experiences for players to engage emotionally and intellectually. In our case, the project seeks to create an experience combining horror elements with environmental storytelling. The game addresses a growing audience interested in environmental conservation and awareness.

The critical part of this project is the increasing concern over ocean conservation, including the threats posed by climate change, pollution, and the destruction of marine ecosystems. With many real-world initiatives that are focused on ocean preservation, this game brings up the excellent storytelling power that video games

provide to bring awareness to these issues. At the same time, it provides gamers with a thrilling experience that horror and survival genres offer.

This project is designed to engage players through gameplay and education. Players will navigate the deep underwater world with tremendous environmental interaction. Through this exploration, we aim to teach players about ocean conservation, gathering information that connects to real-world ecological themes. The ultimate goal is to bring awareness and teach users about the ocean through gameplay.

Motivation:

This project is a great opportunity to blend entertainment with education, reaching an audience of gamers who are interested in environmental issues while giving a thrilling and immersive experience. Video games provide the advantage of delivering interactive narratives that influence the player's experience. By putting ocean conservation within a horror setting, the game aims to bring a deeper understanding of environmental issues, motivating players to learn and take action in their daily lives.

Considerations:

Current environmental campaigns struggle to capture the attention of younger audiences, especially in a meaningful and engaging way. This game seeks to address this by making it a more interactive and immersive experience to bring about these issues. The game's success relies on its interactive gameplay and educational components. This ensures a more lasting impression regarding the importance of ocean protection. Ocean pollution is a severe and occurring problem, which continues to grow nowadays.

2b Goals of the Project

This project is being carried out to raise awareness about ocean conservation while providing a different path to raise awareness of it with gaming. From the user's perspective, the primary goal is to use the power of interactive entertainment to create a greater understanding and emotional connection to environmental issues that affect the ocean, particularly to the gaming audience, which tackles a wide range of people of different ages.

This project is not just a game but a tool that educates players about the impact of human actions in marine environments. We hope a sense of responsibility and action towards ocean conservation efforts inspires the client. The game achieves a broader goal of blending entertainment with environmental advocacy.

2c Measurement

To know when we have met our goals through this project, measurable results should focus on educational and entertainment success. The key measurable goals are:

1. **Engagement with ocean conservation themes:** Players should engage with in-game content related to ocean conservation, for example, reading journals, completing side quests focused on conservation, or saving marine wildlife.
2. **Positive player feedback:** The game should receive a certain level of player satisfaction, particularly in reviews due to its balance of entertainment and environmental messaging. Survey questioning users if their perspectives have changed before and after playing the game
3. **Impact on ocean conservation awareness:** The game should create an impact in terms of partnership with conservation organizations or campaigns.
4. **Success in reach and sales:** The game should achieve a set number of sales indicating a reach in the audience.

3 The Scope of the Work

The work of this project brings awareness and teaches players about ocean conservation through a horror survival game. This game will be used to entertain and inform players about the state of the ocean, human impact on the marine ecosystem, and the need for conservation.

3a The Current Situation

Currently, the client, an organization focused on game development with environmental awareness, is using traditional campaigns and media to promote ocean conservation. These efforts often involve websites, documentaries, social media campaigns, or environmental organizations. While these approaches provide valuable information, they lack the interactive engagement that a video game can offer, which appeals to younger audiences and gamers who haven't been informed.

The client's existing process relies on passive forms of communication, like documentaries, articles, or social media discussions. These methods do not deeply immerse individuals in the experience of why it is so important to save our ocean and the consequences of environmental degradation in an interactive way like a video game can.

There are not that many games that fully integrate entertainment with ocean conservation education in an engaging way. The current approaches are limited by the lack of immersive engagement like video games can and reach a broader audience. This misses the opportunity to create a lasting impact through interactive learning.

3b The Context of the Work

The work in this project involves the creation of an interactive horror- survival game that integrates ocean conservation education. The primary focus is on immersing players in an engaging experience that teaches marine ecosystems, human impacts on the ocean, and the current situation that we are facing in a thrilling gaming experience.

Defining the boundaries for the study of the work and requirements is essential to ensure that the product fits seamlessly into its intended environment. Without a clear

understanding of the work, and how it interacts with external systems, there can be a risk that the product becomes misaligned with the goals or needs of the stakeholders. By doing so, we ensure the team remains on track to deliver a product that meets its educational and entertainment goals while taking only the necessary elements of these external components.

Game Publishing Platforms: Distribution of the game to users, handling of updates, DLC, and patches.

Ocean Conservation Organizations: COllaborating on in-game educational content, awareness campaigns, and potential fundraising opportunities.

Player Communities: Players discuss their experiences, share information, and engage in conversations about the game's themes and mechanics.

Game Development tools: The development tools are critical to the creation of the game, especially for realistic environments and the interactive experience.

3b The Context of the Work

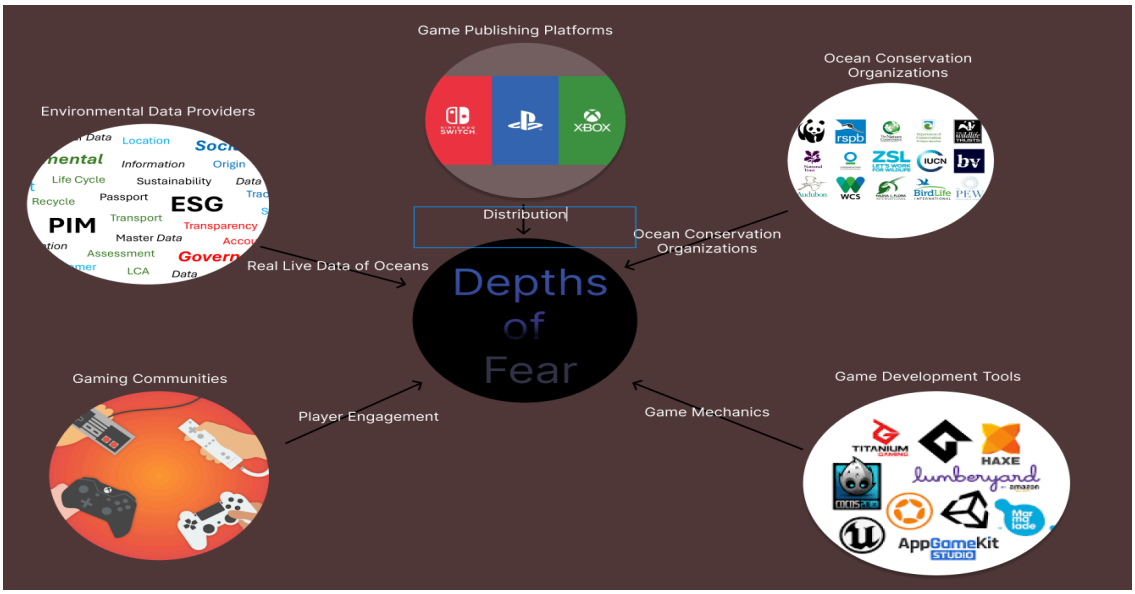


Figure 1 - Depths of Fear Context Diagram

3c Work Partitioning

Event Name	Input and Output	Summary
Player Starts a new game	Player interaction	Initializes game settings, generates game world, and begins introduction of story
Player reaches key conservation area	Player exploration	In-game information on ocean conservation and educational content triggered by proximity
In-game tutorial	Player completes tutorial actions	Game system unlocks core gameplay mechanics and conservation missions
Player completes conservation mission	Mission objectives achieved	Reward player with in-game currency or tolls and provides conservation-related facts or impact on story
Player completes game-ending	Final boss or mission completion	Game triggers final cutscene, ending, and conversation message related to the player choices.
Player encounters mutated marine life	Marine life encounter	Creature attacks or behaves acertain way, relevant educational information about the species displayed
Player encounters npcs on land	NPC interaction	NPC provides story clues, quests, or conservation-related dialogues
Player scans marine species	Scan trigger	Adds scanned species to player\s database with educational content about the species and conservation
time for environmental event	Game system time	Random or pre-programmed ocean

Event Name	Input and Output	Summary
Player Starts a new game	Player interaction	Initializes game settings, generates game world, and begins introduction of story
Player reaches key conservation area	Player exploration	In-game information on ocean conservation and educational content triggered by proximity
		events quests, or conservation-related dialogues
Time for game autosave	System time	Autosaves game state
Time for game update or DLC release	Game system update	Game applies new features, patches, or downloadable contenet
Time for oxygen depletion check	Player time underwater	Alert on oxygen depletion, prompting player to manage resources or return to sufface

3d Competing Products

Competitors and why the product is still needed:

- Lethal Company - Game about the effects of war.
- Similar gameplay with collecting/cleaning scrap in polluted environments
- This product is still needed, for Lethal Company highlights the effects of war instead of environmental consequences
- Subnautica Series - Game about exploring an alien ocean planet
- Similar gameplay with exploring ocean
- This product is still needed, for Subnautica does not highlight the effects of environmental consequences
- Informational websites on environmental protection
- This product is a gamified version, which will engage users more and the information will be given in a fun way.

4 The Scope of the Product

The proposed product will handle the subset of the work involving immersive gameplay and educational experiences related to ocean conservation. The product includes a variety of player actions, such as exploration, resource management, creature encounters, conservation missions, and narrative-driven interactions. It excludes non-interactive educational materials and direct real-world conservation initiatives.

4a Scenario Diagram(s)

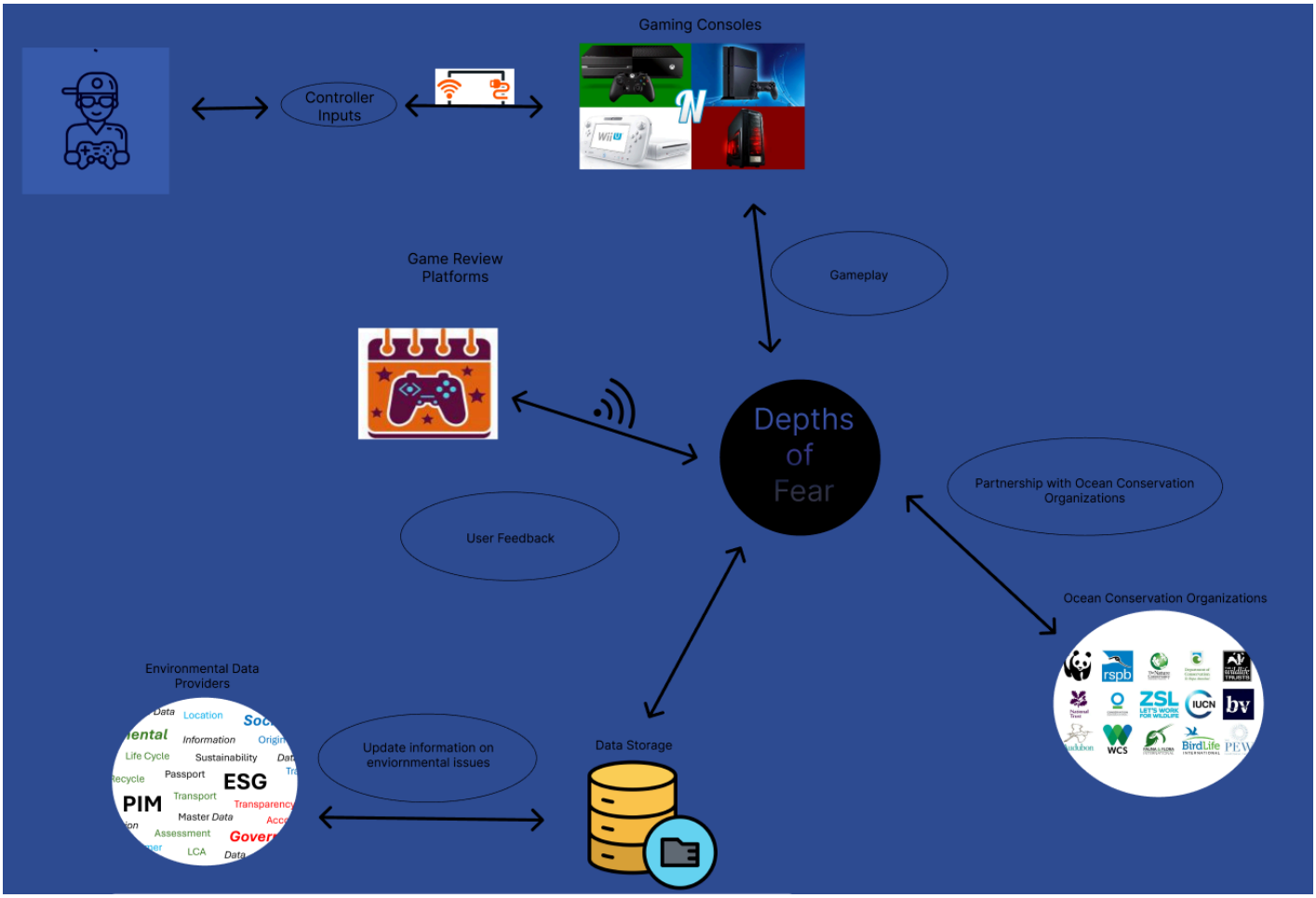


Figure 2 - Depths of Fear Scenario Diagram

4b Product Scenario List

Actor	Product Scenario	Summary
Player	Starting game	Player will be able to start the game and load their saved files from previous games. The game will then load saved chunks and the player can proceed to play.
Player	Manipulating inventory	“e” key on keyboard to access inventory or clicking inventory icon in top right. Can move, delete, and drop items into the environment.
Player	Identify monster underwater	Close proximity to monster to enrage monster, then monster information will show.
Environmental Conservation Organizations	Put in concise educational data pertaining to environmental conservation	Players will be able to access this data and learn off of it by reading it displayed on screen. Environmental conservation organizations will have to upload this data and clarify if it is correct.
SysAdmin	Deploy game system launches/updates	Upload it to a game website or Steam or both when capable. Primarily Steam. SysAdmin will make sure deployment is seamless and easily downloadable to players.

4c Individual Product Scenarios

A type of user input would be necessary to control the character's actions and interact with the gameplay. The gaming console would provide the visuals and sound effects to immerse the user in the environment. The game would allow the user to form partnerships with known conservation organizations, integrating real-world environmental efforts into the game. The game would access current, up-to-date information on ocean contamination and store that data in memory for use in the game environment. The user would receive updates and patches through regular feedback mechanisms, ensuring that gameplay and educational content are continuously improved based on user input.

5 Stakeholders

5a The Client

Our clients are gaming studio investors specializing in immersive horror experiences, seeking an innovative and terrifying game with high replayability to captivate and grow a loyal fanbase of horror enthusiasts.

5b The Customer

Our target customers are gamers drawn to horror exploration, who seek a thrilling and immersive experience and are fascinated by the mysteries and dangers lurking beneath the ocean's depths.

5c Hands-On Users of the Product

Our primary users will be gamers, ranging from casual players to horror game fans. Casual gamers seek immersive, straightforward gameplay, while horror fans are attracted to the game's intense atmosphere and tension.

5d Maintenance Users and Service Technicians

Maintenance users and service technicians for our oceanic horror game will primarily include developers, IT support staff and game platform technicians. These users will handle tasks such as installing updates, fixing bugs, and maintaining game servers. Their job is to ensure the smooth operation of the game for all users, with a focus on stability, performance, and timely updates to keep the gameplay experience immersive and functional

5e Other Stakeholders

1. **Sponsor:** The project's financial backer, providing the resources necessary to build and market the game. Their influence is high, and they must be regularly updated on the project's progress.
2. **Testers:** Both internal and external testers are needed to identify bugs, provide gameplay feedback, and ensure the game functions as intended. Their

involvement is essential during the development and beta phases, with moderate influence on gameplay refinement.

3. **Business Analysts:** They will evaluate the market potential, financial feasibility, and user demographics for the game, guiding decisions on feature prioritization and marketing strategies. Their influence is high as they align the game with market needs.
4. **Marketing Experts:** Responsible for branding, advertising, and launching campaigns to attract gamers, schools, and environmental organizations. Their influence is significant in shaping the game's public image and reaching the target audience.
5. **Legal Experts:** Needed to ensure the game complies with intellectual property laws, privacy policies, and any partnership agreements, such as those with environmental organizations. Their involvement is typically moderate but essential to avoid legal risks.
6. **Domain Experts (Marine Biologists, Environmentalists):** These experts provide critical insights into the game's educational content, ensuring that marine biology and environmental issues are accurately represented. Their influence is high, as they contribute to the game's educational credibility.

5f User Participation

1. Casual Gamers and Horror Fans: These users will participate primarily in playtesting phases and when the game releases, providing feedback on gameplay mechanics, pacing, and overall experience. Their involvement will be crucial for identifying usability issues and balancing the game's intensity. We expect periodic participation during the alpha and beta testing phases, requiring at least a few hours of gameplay and feedback sessions.
2. Environmental Enthusiasts and Domain Experts: These users will be involved early on, contributing business knowledge and subject matter expertise. Their feedback will help shape the accuracy of marine biology content and environmental messaging. Their participation will be ongoing, with a heavier focus during content development stages, and they may spend several days reviewing materials and attending design workshops.
3.

5g Priorities Assigned to Users

1. **Key User:**
   - **Casual Gamers and Horror Fans:** These are the primary target audience for the game and will directly influence its long-term success. Their feedback on gameplay mechanics, atmosphere, and user experience is critical. Their requirements will have the greatest influence on design, ensuring the game remains immersive, engaging, and enjoyable for its core audience.
2. **Secondary User:**
   - **Environmental Enthusiasts:** While important for promoting the educational and conservation aspects of the game, their feedback will be secondary to key users. Their input will influence the accuracy of marine biology content and environmental themes, but gameplay mechanics will take precedence over educational components.
3. **Unimportant Users:**

- **Infrequent users/biased users:** These may include users who interact with the game briefly or misuse its features. Their feedback will have little to no effect on the design, as they do not represent the target audience. Any specific requirements from this group will receive low priority and minimal consideration in design decisions.

6 Mandated Constraints

6a Solution Constraints

We will be utilizing Unity Engine for the development and production of Depths of Fear. This is because Unity offers cross-platform compatibility, which aligns with our goal of offering the game on multiple platforms. Unity also provides tools that are great for immersive environments, especially because it has a wide range of plugins suitable for implementing game mechanics such as underwater physics, and realistic lighting, which are crucial for maintaining the horror atmosphere. We will also be using Unity Engine due to its simplicity compared to other engines. Unity allows for an easier learning environment for the development team while offering the necessary tools and flexibility needed to create the game due to its user–friendly interface, as well as its scripting capabilities in C#. This will allow our development team to provide quicker and higher quality production, allowing the game to be more suitable and optimized for the systems it will be released. Thus, Depths of Fear shall utilize the many different features in Unity to manage the production and maintenance of the game.

6b Implementation Environment of the Current System

Depths of Fear will be developed and played on cross-platform environments, specifically Windows 10-11, PlayStation, and Xbox consoles. Development hardware includes gaming PCs with sufficient GPU capabilities for real-time 3D rendering and testing. The game must also support multiplayer connectivity, requiring different network protocols so that cross-platform gameplay is available. Our game needs to meet the expectations of the modern gaming community by providing good graphics and performance. By using these consoles, it allows the game to be more accessible to our target audience while ensuring quality gameplay. Thus, the game must pass platform requirements for PlayStation and Xbox, and on Windows, it must meet compatibility standards with DirectX 12.

6c Partner or Collaborative Applications

The game will collaborate with Discord for community interaction, multiplayer voice chat, and game-related communication. Discord is a convenient communication platform that is used for both in-game communication and community building. By providing our own official Discord server, we can help players find other players at different stages of the game, give and receive feedback from players and between

players, and provide routine updates on the state of the game and production to keep players motivated and engaged. Thus, the game will allow players to link their Discord accounts and join our official server, as well as have the option to provide rewards for those active in the game and community. We will also have to coordinate with NVIDIA's GeForce Experience and AMD's Radeon software to provide players with quality gameplay and performance. Both systems are the most common GPUs that are used by gamers, which allow players to optimize performance with any game as well as provide tools for content creation and streaming. As such, allowing our game to be optimized through those systems will allow players to have a better gameplay experience.

6d Off-the-Shelf Software

We will be utilizing Steam, Microsoft Store, and PlayStation store to distribute the game to Windows, Xbox, and PlayStation. These platforms are widely used among the gaming community and provide built-in infrastructure for game updates and community engagement. Thus, we must meet the requirements for publishing on Steam, Microsoft Store, and PlayStation, filing any required licenses and certifications.

6e Anticipated Workplace Environment

This game will be played in personal spaces containing a gaming console that we support. It is up to the user where they would like to play our game, whether it is at home or at a gaming cafe, as long as the console has our game on it, it can be played. The only constraint is if the user would like to play multiplayer, in which case they will need to be playing while connected to the internet.

6f Schedule Constraints

The development of Depths of Fear will follow an 18-month schedule where there will be specific milestones for ensuring the quality of the work. That will allow developers to not rush the final product, but also keep them within a time frame to boost productivity. This will also give us time to ensure that we adhere to any policies from third-party software that are necessary for the game's launch. Every two weeks, there should also be full-game testing to correct any issues in development up to that point. However, a free public demo or beta version of the game should be released a month or two before seasonal breaks, such as April or October, which should only be playable for about a week. This will give users a chance to try our game before it is fully released to give traction for the full release, which should be in June or December. Releasing a public demo/beta will also give our developer team to ensure all systems are working correctly by putting it all in production before a full release of the game. We chose June and December for the full release since this will be when most of our target audience will be free.

6g Budget Constraints

The budget for the development, marketing, and platform certifications for Depths of Fear is set at $1 million. This will include licensing for development tools such as

Unity engine as well as publishing costs and quality assurance. The game must be completed within this budget, and any additional funding will only be required post-release for something like game expansions/peripherals or server maintenance.

7 Naming Conventions and Definitions

7a Definitions of Key Terms

Terms	**Definition**
Downloadable Content (DLC)	Expansion packs to provide more content to the game
PC	Personal computer
DirectX 12	a Microsoft API that allows developers to add advanced graphics to games and applications
Non-Player Character (NPC)	characters controlled by the game's AI This can also be seen as: - Human NPC - HNPC - Monster NPC - MNPC
Mutations	Genetic alterations in marine life are caused by human interference.
Environmental Catastrophe	Harmful environmental changes in the ocean ecosystems, which is caused by human activities
Oxygen Meter	A survival mechanic in the game where the player must monitor and replenish their oxygen levels while exploring underwater environments.
ECO (Environmental Conservation Organization)	The organization partnered with the game
HUD (Heads Up Display	Shows play information such as health, score, inventory, etc.

7b UML and Other Notation Used in This Document

- Class Diagrams
  (i) Direct association →
  (ii) Inheritance ⇾
  (iii) Dependency ⇢
  (iv) Composition (arrow with filled in diamond head)
- Classes
  (i) Player - player object
  (ii) Monster - monster enemy
  (iii) Human - human NPC with quests
- Use case diagrams (representations)
  (i) Oval - Use cases
  (ii) Human figure - human figure, the actor (user)
  (iii) Lines (associations between each use case/actor)

7c Data Dictionary for Any Included Models

A server-side database will be needed to identify users with their online profiles for multiplayer.

- Objects
  (i) Player object and attributes:
    1. Health - Level meter, when attacked, decrease health meter
    2. Position - Based on coordinates
    3. Inventory - Composes of other item objects
    4. Oxygen levels - Level meter, time run out

  (ii) Enemy objects and attributes:
    1. Strength - Strength level of enemy
    2. Position - Based on coordinates
    3. AI (for detecting player, combat)

  (iii)Game states
    1. Menu
    2. Selections - Subclass of menu
    3. End game
    4. Current level

  (iv) Dictionary
    1. Each entity (key) will be connected to an information about it (value)
    2. Dictionary for enemy objects, key will be classification of type of enemy based on danger level, value will be the strength

8 Relevant Facts and Assumptions

8a Facts

The developer team will consist of developers split into six teams. The six teams will be responsible for the following:

- Core gameplay mechanics

- Player modeling and animation

- Enemy and NPC AI design

- World and level design and rendering

- Multiplayer networking

- UI/UX and sound design, game menus, HUD, music and sound

This will ensure all aspects of the game will be handled by groups based on specialty. At some point, there will be in-game purchases through DLC and expansion packs. Information about aquatic life and environments will be studied and provided through trusted sources, although some may be fictional to follow the horror theme of the game.

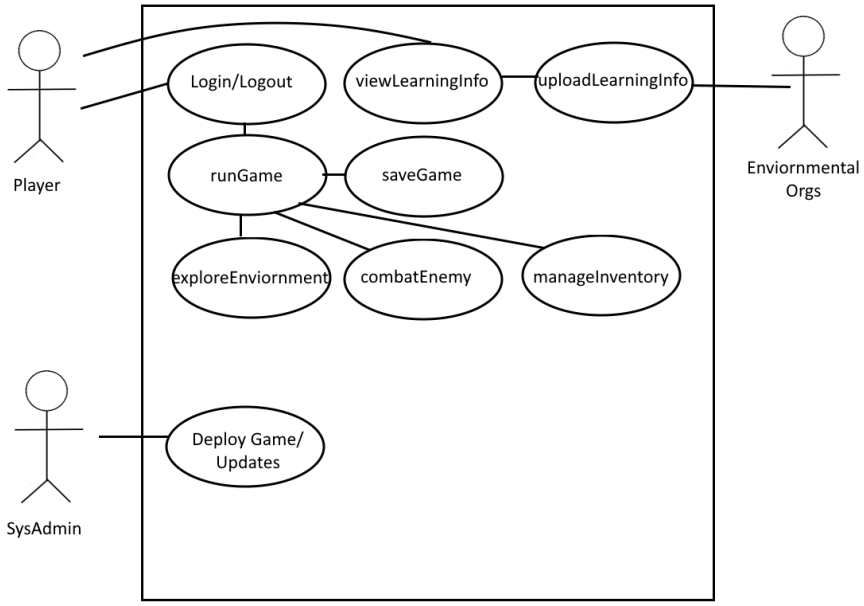
8b Assumptions

We will be assuming that Unity engine and other relevant developer tools will be available. We will also assume that we will acquire the necessary licenses and certifications for publishing by the end of production. Hardware requirements will be determined towards the end of the development phase, however, we expect that the game will be able to run smoothly on Xbox and PS5, while for PCs we expect it to run with 8GB+ RAM, GTX1060+ or AMD RX 580, and an Intel Core i5-4460 or AMD FX-8300.

II. Requirements

9 Product Use Cases

9a Use Case Diagrams



9b Product Use Case List

Login/Logout (UC-1)	Actors: Player Followed By: N/A
viewLearningInfo (UC-2)	Actors: Player Followed By: uploadLearningInfo (UC-3)
uploadLearningInfo (UC-3)	Actors: Environmental Orgs Followed By: N/A
runGame (UC-4)	Actors: Player Followed By: N/A
saveGame (UC-5)	Actors: Player Followed By: runGame (UC-5)
exploreEnvironment (UC-6)	Actors: Player Followed By: runGame (UC-5)
combatEnemy	Actors: Player

(UC-7)	Followed By: runGame (UC-5)
manageInventory (UC-8)	Actors: Player Followed By: runGame (UC-5)
Deploy Game/Updates	Actors: SysAdmin Followed By: N/A

9c Individual Product Use Cases

Use case ID: UC-1 Name: Login/Logout pre-conditions: User has valid account post-conditions: User is logged in and directed to the main dashboard Initiated by: Player Triggering Event: Player enters their credentials and clicks the "Login"/"Logout" button Additional Actors: N/A
Sequence of Events: 1 User clicks the "Login" button 2 Game enters player data for player to play with their saved data 3 User clicks the "Logout" button 4 System ends the user session and clears session data. 5 System redirects the user to the login page.
● Alternatives: ○ Session timeout: If the user session times out due to inactivity, the system automatically logs out the user and displays a timeout message on the login page. ● Exceptions:

- System error during logout: If the logout process encounters an error, the system displays an error message and prompts the user to attempt logging out again.
- System error during login: If the login process encounters an error, the system displays an error message and prompts the user to attempt logging in again.

Use case ID: UC-2 Name: viewLearningInfo

pre-conditions: Learning info must be uploaded by Environmental Orgs

post-conditions: Player views educational information

Initiated by: Player

Triggering Event: Player clicks the (?) info symbol to learn more

Additional Actors: Environmental Orgs

Sequence of Events:

1 User clicks the (?) info symbol to learn more

2 Data of educational information will load to player screen

3 User clicks the (x) symbol to exit out

- Alternatives:
  - Session timeout: If the user session times out due to inactivity on viewing the information, the system automatically logs out the user and displays a timeout message on the login page.
- Exceptions:
  - System error during viewing information: If the viewing information process encounters an error, the system displays an error message and prompts the user to attempt viewing it again.
  - System error during closing information: If the viewing information process encounters an error, the system displays an error message and prompts the user to attempt closing out again.

Use case ID: UC-3 Name: uploadLearningInfo

pre-conditions: Learning info must be fact checked by environmental orgs, environmental orgs must agree to partner with game

post-conditions: Learning info uploaded to game

Initiated by: Environmental Orgs

Triggering Event: Upload the information to educational information database

Additional Actors: N/A

Sequence of Events:

1 Environmental organization loads the educational information to game database

2 Environmental organization uploads the educational information to game database

- Alternatives:
  - Redundant information: If the information uploaded to educational informational database is redundant/copy, then error message prompts to the screen saying it is an exact copy and load another
- Exceptions:
  - System error during uploading information: If there is an error uploading information, system prompts environmental organization to reupload again

Use case ID: UC-4 Name: runGame pre-conditions: Must have a valid login post-conditions: Game is loading textures/chunks and prompts player to the game Initiated by: Player Triggering Event: Player hits “Play Game” Additional Actors: N/A
Sequence of Events: 1 Player clicks “Play Game” 2 Game runs textures/chunks and loads saves 3 Game is loaded
● Alternatives: ○ Network delay: If the system experiences a delay retrieving the profile, a loading indicator appears until the information loads. ● Exceptions: ○ Data retrieval error: If there’s an error in loading chunks, the system displays an error message and suggests retrying

Use case ID: UC-5 Name: saveGame pre-conditions: Must be in the current game post-conditions: Game says “Game Saved” and game is saved onto data Initiated by: Player Triggering Event: Player hits “Save Game” Additional Actors: N/A

Sequence of Events:

1 Player clicks “Save Game”

2 Game processes save with loading icon, not interfering with gameplay

3 Game data is saved into data and can be loaded again upon playing again

- Alternatives:
  - Data save delay: If the system experiences a delay uploading the data, a loading indicator appears until the information uploads.
- Exceptions:
  - Data entry error: If there’s an error in uploading data, the system displays an error message and suggests retrying

Use case ID: UC-6 Name: exploreEnvironment

pre-conditions: Must be in the current game

post-conditions: Player is able to move around and navigate game

Initiated by: Player

Triggering Event: Player moves in game with WASD keys

Additional Actors: N/A

Sequence of Events:

1 Player hits WASD keys

2 Player moves

3 Player can keep moving until end of map

4 If player hits end of map, the game character falls off and dies

- Alternatives:
    - Player lag: If the player freezes, display that the ping/network of the game in red instead of green and have an icon “high ping”
- Exceptions:
    - Move error: If there’s an error in moving/no detection in moving, game runs as it normally does

Use case ID: UC-7 Name: combatEnemy

pre-conditions: Must be in the current game and encounter an enemy

post-conditions: Player battles enemy with controls

Initiated by: Player

Triggering Event: Player is in close proximity of enemy using WASD keys and enemy is triggered

Additional Actors: N/A

Sequence of Events:

1 Player hits WASD keys

2 Player moves to close proximity of enemy

3 Enemy is triggered, and a small symbol is put up

4 Enemy uses battle mechanics (clicking left mouse button) to battle

- Alternatives:
    - Player surrender: Player can surrender in boss fights, and the player is in a wounded state afterwards. Boss fight can be restarted.
- Exceptions:
    - Enemy proximity detection error: If there is an error entering a fight, prompt the user to retry.

Use case ID: UC-8 Name: manageInventory

pre-conditions: Must be in the current game and can be able to enter inventory (not in boss fights, in neutral environment)

post-conditions: Player is able to view and modify inventory

Initiated by: Player

Triggering Event: Player enters the “e” key on keyboard to enter inventory

Additional Actors: N/A

Sequence of Events:

1 Player hits “e” key on keyboard

2 The inventory contents are opened up as a display of player

3 If player wants to delete, click object and hit “delete” key on keyboard

4 If player wants to rearrange contents of objects, click and drag object around in the boxes of inventory

5 If player wants to drop an item outside of inventory and into environment, click the object and drag outside of inventory screen

6 Player exits inventory by hitting “e” key again

- Alternatives:
  - Player can enter inventory by left clicking the inventory icon on top right of screen
- Exceptions:
  - Inventory open error: If there is an error opening inventory, prompt user to open again on display screen
  - Inventory close error: If there is an error closing inventory, prompt user to close again on display screen
  - Inventory delete error: If there is an error deleting object from inventory, prompt user to delete object again on display screen
  - Inventory open error: If there is an error dropping object from inventory, prompt user to drop object again on display screen

Use case ID: UC-9 Name: Deploy Game/Updates pre-conditions: Development of gameplay must be quality before uploading game downloads to server post-conditions: Game download is available to player download in server Initiated by: SysAdmin Triggering Event: SysAdmin deploys game onto downloads server Additional Actors: N/A
Sequence of Events: 1 Game is polished 2 SysAdmin is prompted to upload game into game server 3 SysAdmin processes the game uploads to server 4 Game is successfully uploaded to server for players to download
● Alternatives: ○ Game uploaded on Steam rather than game website, SysAdmin has to successfully upload on Steam ● Exceptions: ○ Game Upload Error: If uploading the game has an error, a display message saying game upload error has occurred to the SysAdmin's screen.

10 Functional Requirements

ID: F-1 - Player Login

Description: The system must allow players to log into their accounts by entering a username and password.

Rationale: To ensure that only authorized players can access their progress and saved game data.

Fit Criterion: The player must be able to log in successfully with valid credentials and receive an error message when entering invalid credentials.

Acceptance Tests: *Valid Login Test**,** Invalid Login Test*

ID: F-2 - Save Game Progress

Description: The game must provide a mechanism to save player progress, including current level, health, and inventory.

Rationale: Allowing players to save progress enables them to continue from where they left off in future sessions.

Fit Criterion: Players can save their progress at any point, and the game can retrieve the exact state when reloaded.

Acceptance Tests: *Save Progress Test, Load Saved Game Test*

ID: F-3 - View Player Stats

Description: Players should be able to view their current stats, including health, stamina, and inventory, through an in-game menu.

Rationale: Providing players with information about their status helps them make strategic decisions.

Fit Criterion: The player stats must accurately reflect the current game state, and the menu should update dynamically based on in-game events.

Acceptance Tests: *View Stats Test*

ID: F-4 - Enemy Interaction

Description: The game must support interactions with enemy characters, including attacking and avoiding.

Rationale: Player-enemy interactions are central to gameplay, as they create challenges and objectives.

Fit Criterion: The player must be able to initiate attacks and experience appropriate responses from enemies based on game rules.

Acceptance Tests: *Enemy Attack Test, Enemy Evasion Test*

ID: F-5 - Health Regeneration

Description: The game must provide a mechanism for players to regenerate health over time or through consumables.

Rationale: Health regeneration adds a layer of strategy to survival, allowing players to recover during extended gameplay.

Fit Criterion: The player's health bar should accurately reflect regeneration effects, and consumable items should have a visible effect when used.

Acceptance Tests: *Health Regeneration Test, Consumable Use Test*

ID: F-6 - Fear Level Indicator

Description: A fear level indicator must be present, showing the player's current fear level based on encounters and environment.

Rationale: The fear level impacts gameplay mechanics, such as enemy behavior and player visibility, adding depth to the horror experience.

Fit Criterion: The fear level indicator should change dynamically based on gameplay events and visibly influence player abilities.

Acceptance Tests: *Fear Level Increase Test, Fear Level Decrease Test*

ID: F-7 - Map Exploration

Description: The game must allow players to explore a map, with areas unlocking based on progression or specific in-game actions.

Rationale: Map exploration is a core aspect of gameplay, encouraging players to explore and discover new challenges.

Fit Criterion: The player must be able to unlock and access new areas upon meeting certain requirements or finding specific items.

Acceptance Tests: *Area Unlock Test, Restricted Access Test*

ID: F-8 - Audio Cues for Proximity

Description: Audio cues must alert players to the proximity of enemies or other game-relevant entities.

Rationale: Audio cues enhance immersion and provide players with a non-visual indication of nearby threats or objectives.

Fit Criterion: The system must play distinct sounds that vary in intensity based on the proximity of an enemy or important object.

Acceptance Tests: *Enemy Proximity Cue Test, Objective Proximity Cue Test*

ID: F-9 - Inventory Management

Description: Players must be able to manage their inventory, adding and removing items as needed.

Rationale: Effective inventory management allows players to strategize and adapt to various in-game challenges.

Fit Criterion: Players should be able to view, add, and remove items in their inventory, with changes reflected immediately.

Acceptance Tests: *Add Item Test, Remove Item Test, View Inventory Test*

ID: F-10 - Combat Mechanics

Description: The game must implement combat mechanics allowing players to attack, defend, and use weapons effectively.

Rationale: Combat mechanics are essential to gameplay, providing challenges and goals for the player.

Fit Criterion: Players should be able to initiate attacks, defend against enemies, and use weapons with varied effects.

Acceptance Tests: *Attack Test, Defense Test, Weapon Use Test*

ID: F-12 - Health Indicator

Description: The game must display a health indicator showing the player's current health status.

Rationale: The health indicator informs the player of their survival status, helping them make informed gameplay decisions.

Fit Criterion: The health indicator must accurately display health status and respond to damage or healing effects.

Acceptance Tests: *Health Reduction Test, Health Increase Test*

ID: F-13 - Educational Information Read

Description: The game must display the educational information showing to the player when prompted and the player can exit out of viewing the educational information.

Rationale: The educational information will help players learn about environmental conservation

Fit Criterion: The information must be accurate and approved by environmental organizations.

Acceptance Tests: *Information display test, Information go away display test*

11 Data Requirements

ID: D-1 - Player Profile

Description: The Player Profile stores data related to individual players, including username, password, current level, health status, and inventory.

Rationale: Player profiles allow users to access their saved progress and personalize their gaming experience.

Fit Criterion: Each player must have a unique profile that can be created, updated, and deleted, with secure access to sensitive data like passwords.

Acceptance Tests: *Profile Creation Test, Profile Update Test*

ID: D-2 - Enemy Entity

Description: The Enemy Entity data structure contains information about different enemy types, including health, attack power, movement patterns, and fear level impact.

Rationale: Defining enemy attributes and behaviors allows the game to create diverse challenges and adjust difficulty levels.

Fit Criterion: The system must support the instantiation of various enemy entities with unique attributes that affect gameplay dynamics.

Acceptance Tests: *Enemy Instantiation Test, Enemy Attribute Verification Test*

ID: D-3 - Item Inventory

Description: The Item Inventory maintains a list of items that players can collect, use, or store, including item names, effects, and usage limitations.

Rationale: Managing items is essential for gameplay, allowing players to use resources strategically for survival and progression.

Fit Criterion: The inventory must track each item accurately, support adding and removing items, and handle item-specific effects.

Acceptance Tests: *Add Item Test, Remove Item Test, Item Effect Test*

ID: D-4 - Health and Stamina Status

Description: Health and Stamina Status data represent the player's current health points (HP) and stamina levels, which are affected by actions and enemy encounters.

Rationale: Health and stamina are key gameplay metrics that influence player survival and performance in combat or exploration.

Fit Criterion: Health and stamina values should update in real-time based on player actions and interactions with the environment.

Acceptance Tests: *Health Update Test, Stamina Update Test*

ID: D-5 - Game Map

Description: The Game Map data structure holds information about the layout, rooms, unlocked/locked areas, and points of interest within the game world.

Rationale: The game map structure provides a navigational framework, enabling the game to manage area accessibility and exploration.

Fit Criterion: The map must support dynamic changes, like unlocking new areas, and allow the system to store and retrieve map states accurately.

Acceptance Tests: *Map Access Test, Area Unlock Test*

ID: D-7 - Fear Level Indicator

Description: The Fear Level Indicator data tracks the player's current fear level, which influences enemy behavior and player abilities.

Rationale: Fear level is a core mechanic affecting the game's intensity and player immersion.

Fit Criterion: The system must adjust the fear level based on encounters and environmental factors, and it must dynamically impact gameplay.

Acceptance Tests: *Fear Increase Test, Fear Decrease Test*

ID: D-8 - Audio Cues

Description: Audio Cue data includes proximity-based sounds, such as enemy approach alerts, environmental sounds, and background music.

Rationale: Audio cues enhance player immersion and provide non-visual feedback that informs gameplay.

Fit Criterion: Audio cues should play in response to specific events, with distinct sounds associated with various in-game conditions.

Acceptance Tests: *Proximity Cue Test), Event-Based Cue Test*

ID: D-9 - Game Settings

Description: The Game Settings data contains player preferences, including audio levels, display options, and control configurations.

Rationale: Allowing players to customize game settings enhances the user experience and makes gameplay more accessible.

Fit Criterion: Players must be able to adjust settings at any time, with changes applied immediately and saved for future sessions.

Acceptance Tests: *Settings Adjustment Test, Settings Persistence Test*

ID: D-10 - Save State

Description: Save State data stores the player's current game progress, including position, health, inventory, and unlocked areas.

Rationale: Saving progress allows players to continue their game from the same state at a later time, enhancing replayability.

Fit Criterion: The save state must capture all essential data points and allow the game to restore the exact conditions upon loading.

Acceptance Tests: *Save State Test, Load State Test*

ID: D-11 - Educational Information

Description: Educational information must be stored for players to read from

Rationale: Players must be able to view educational information stored as data in order to learn about environmental conservation.

Fit Criterion: The data must be accurate and approved by environmental organizations, and be able to be retrieved accurately from data with each specific information attached to the game.

Acceptance Tests: *Educational information retrieval test*

12 Performance Requirements

12a Speed and Latency Requirements

ID: PR-01 - Response Time for User Interface

Description: The game's interface should respond within 1 second for all user interactions to maintain immersion in a suspenseful atmosphere.

Rationale: Players need fast feedback on actions (e.g., switching tools, inventory access, or checking oxygen levels) to avoid frustration and break in engagement, especially when facing threats.

Fit Criterion: The game shall respond within 1 second for 90% of interactions, with no action taking longer than 2.5 seconds.

Acceptance Tests: *UI Response Tests*

ID: PR-02 - Environmental Loading Time

Description: Underwater and on-land environments should load within 5 seconds to avoid disrupting gameplay.

Rationale: Players' sense of immersion and the horror effect depend on smooth transitions, especially in environments meant to feel isolated and mysterious.

Fit Criterion: Environmental transitions shall load in under 5 seconds.

Acceptance Tests: *Gameplay Timing Test*

12b Precision or Accuracy Requirements

ID: AR-01 - Accuracy of Oxygen and Health Levels

Description: Oxygen and health levels must be updated accurately and reflect real-time changes as the player moves or performs actions.

Rationale: Accurate tracking of resources is crucial for survival elements. Minor inaccuracies can undermine the player's strategy and immersion in the survival experience.

Fit Criterion: The game shall display oxygen and health levels with updates occurring every 0.1 seconds.

Acceptance Tests: *Oxygen/Health Test*

ID: AR-02 - Environmental Hazard Precision

Description: Environmental hazards, such as predator proximity and resource locations, should be tracked accurately within a ±1-meter range.

Rationale: Accurate representation of hazards and resources is essential for strategic gameplay. Misrepresenting distances could frustrate players or make the game too easy.

Fit Criterion: Hazard tracking accuracy within ±1 meter.

Acceptance Tests: AT-04, accuracy tests for proximity and distance measurement during gameplay.

12c Capacity Requirements

ID: CR-01 - Maximum Simultaneous Users

Description: The server should handle up to 200 simultaneous players during peak hours.

Rationale: To support online play and a steady user base, server infrastructure must handle a significant number of simultaneous users without lag or downtime.

Fit Criterion: The server shall support 200 simultaneous users during peak hours.

Acceptance Tests: *Max User Stress Test*

ID: CR-02 - Max Resource Allocation for Deep-Sea Maps

Description: Each map should allow up to 100 resources (e.g., oxygen tanks, habitat zones) to be loaded concurrently without affecting performance.

Rationale: Players need consistent access to game resources to plan and execute survival strategies effectively.

Fit Criterion: Maps can load and track 100 concurrent resources.

Acceptance Tests: AT-06, in-game performance tests with maximum resources loaded.

13 Dependability Requirements

13a Reliability Requirements

ID: RR-01 - Gameplay Continuity

Description: The game shall not fail more than once every 10 hours of continuous gameplay, maintaining a stable experience without unexpected shutdowns or critical errors.

Rationale: Reliability is essential to keep players immersed in the game, as unexpected errors can disrupt the horror experience and reduce player engagement.

Fit Criterion: The game shall experience no more than one unexpected shutdown or critical error per 10 hours of gameplay.

Acceptance Tests: *Gameplay Stability Test*

ID: RR-02 - Data Integrity During Save Operations

Description: Game save operations shall not corrupt data, ensuring that all saved files remain intact and accessible after saving.

Rationale: Protecting saved data from corruption is essential to maintain player progress and trust in the game's reliability.

Fit Criterion: Game saves shall retain data integrity in 99.9% of save operations, with no corrupted data in 100 test cases.

Acceptance Tests: *Data Integrity Test*

13b Availability Requirements

ID: AV-01 - Game Server Uptime

Description: The game servers shall maintain an uptime of 99%, allowing access to players 24/7, with minimal downtime due to server maintenance or unexpected issues.

Rationale: High availability is necessary to provide players with the flexibility to play at any time, enhancing accessibility and ensuring that the game is always ready for players.

Fit Criterion: Game servers shall maintain 99% uptime, with any downtime averaging no more than 15 minutes per day.

Acceptance Tests: *Server Uptime Monitoring*, *Downtime Tracking Test*

ID: AV-02 - Load Handling During Peak Hours

Description: The game shall support up to 10,000 simultaneous players during peak times, ensuring access without connection drops or excessive wait times.

Rationale: Players are more likely to log in during peak hours; the game must be able to accommodate these times to maintain accessibility.

Fit Criterion: Game servers shall support 10,000 simultaneous players during peak times with no significant connection delays (average response time under 2 seconds).

Acceptance Tests: *Peak Load Capacity Test*, *Connection Time Test*

13c Robustness or Fault-Tolerance Requirements

ID: ROB-01 - Offline Mode Activation

Description: If the player's internet connection is lost, the game shall automatically switch to offline mode, allowing continued gameplay without online features.

Rationale: To avoid breaking immersion due to connectivity issues, the game should handle network disruptions smoothly by providing an offline mode and resuming online functions once connectivity is restored.

Fit Criterion: The game shall enter offline mode within 5 seconds of connection loss and restore online functionality within 10 seconds of reconnecting.

Acceptance Tests: *Connection Loss Handling Test*, *Reconnect Restoration Test*

ID: ROB-02 - Graceful Failure Handling

Description: If critical components (e.g., graphics engine) fail, the game shall enter a safe mode that displays an error message and saves the current progress automatically.

Rationale: Ensuring a graceful failure response is essential to prevent data loss or crashes that might affect the overall user experience.

Fit Criterion: The game shall enter safe mode and save the player's current progress within 3 seconds of a critical failure.

Acceptance Tests: *Graceful Failure Test*, *Safe Mode Activation Test*

13d Safety-Critical Requirements

ID: SC-01 - Visual Comfort and Health Protection

Description: The game shall provide settings for visual adjustments (e.g., brightness, motion blur) and display a warning for disturbing imagery to protect player comfort and health.

Rationale: As a horror game, it is important to offer customization for players with visual sensitivities, helping them adjust the visuals to reduce discomfort and avoid potential health issues.

Fit Criterion: Visual adjustment settings shall be accessible from the main menu, and a warning regarding disturbing imagery shall be displayed when the game starts.

Acceptance Tests: *Visual Settings Test*, *Imagery Warning Display Test*

ID: SC-02 - Audio Alert Levels

Description: The game shall provide an option to adjust jump-scare sound levels and display a volume warning, allowing players with auditory sensitivities to customize their experience.

Rationale: High-volume audio can be distressing, especially in a horror setting; allowing players to adjust these settings can protect against auditory strain.

Fit Criterion: Jump-scare volume settings shall be adjustable from the settings menu, with a volume warning shown if levels exceed a set threshold.

Acceptance Tests: *Audio Sensitivity Test*, *Volume Warning Display Test*

14 Maintainability and Supportability Requirements

14a Maintenance Requirements

ID: MSR-01 - Timely Report Generation

Description: The system shall generate new Management Information System (MIS) reports within one working week after the requirements are agreed upon.

Rationale: Quick turnaround for report generation is essential to meet business needs and decision-making processes.

Fit Criterion: 95% of MIS report requests shall be completed within one working week of requirements finalization.

Acceptance Tests: *Report Generation Time Test*

14b Supportability Requirements

ID: SUP-01 - Self-Help Support Features

Description: The system shall include a comprehensive self-help feature, allowing users to access troubleshooting guides and FAQs without needing to contact support.

Rationale: Providing self-help resources empowers users to resolve minor issues independently, reducing support load and improving user experience.

Fit Criterion: 80% of user-reported issues should be resolved through self-help features without needing direct support interaction.

Acceptance Tests: *Self-Help Resource Usage Test*

14c Adaptability Requirements

ID: ADR-01 - Cross-Platform Functionality

Description: The system shall operate on Windows, Playstation, and Xbox platforms without loss of functionality or performance.

Rationale: Ensuring cross-platform compatibility maximizes user accessibility and broadens the market reach of the product.

Fit Criterion: The system shall function equivalently on both operating systems with less than 5% performance variance.

Acceptance Tests: *Cross-Platform Functionality Test*

14d Scalability or Extensibility Requirements

ID: SER-01 - Customer Capacity Expansion

Description: The system shall be capable of processing existing customer data for 100,000 customers, with scalability to support up to 500,000 customers within three years.

Rationale: Planning for growth in user capacity is essential to maintain system performance and meet future demand.

Fit Criterion: The system shall handle increased capacity without performance degradation for 95% of operational tasks.

Acceptance Tests: *Customer Capacity Test*

14e Longevity Requirements

ID: LOR-01 - Expected Operational Lifetime

Description: The system shall be designed to operate effectively within the maximum maintenance budget for a minimum of five years.

Rationale: Understanding the expected lifetime of the product helps in budgeting and planning for future upgrades and support.

Fit Criterion: The system shall maintain operational functionality with minimal disruptions and costs for five years under normal operating conditions.

Acceptance Tests: *Operational Longevity Test*

15 Security Requirements

15a Access Requirements

ID: SR-01 - Gameplay Data Access Controls

Description: Specify access permissions for different types of user data, ensuring that only authorized players can access their game profiles and progress. Developers and testers have access to limited game analytics data to improve gameplay but cannot view personal user data or saved game content without consent.

Rationale: To protect players' saved progress and profiles, and to restrict data access solely to authorized individuals for debugging or enhancing gameplay, preventing unauthorized changes or data misuse.

Fit Criterion: Only the individual player can access, modify, or delete their own saved game data. Game analytics data is accessible to the development team only for improving gameplay and enhancing user experience without breaching player confidentiality.

Acceptance Tests: Account Creation/Deletion, User Authentication

15b Integrity Requirements

ID: SR-02 - Game Progress Data Integrity

Description: The game shall ensure that players' saved progress is protected against accidental loss or corruption. Save files are automatically backed up periodically, with data checks in place to prevent any incomplete or corrupted saves from being loaded.

Rationale: To provide players with a reliable gaming experience by safeguarding their progress, preventing the frustration and negative impact associated with lost or corrupted saves.

Fit Criterion: The game automatically backs up save data at regular intervals. On startup, the game checks the integrity of the saved data and, if corrupted, loads the last intact backup.

Acceptance Tests: *Account Creation/Deletion, Conservation Awareness Impact*

ID: SR-03 - Conservation Content Integrity

Description: The game shall protect conservation and educational content from accidental modification or deletion, ensuring that all information presented remains accurate and consistent with conservation themes.

Rationale: To maintain the educational integrity of the game by ensuring that conservation information remains accurate and accessible throughout gameplay.

Fit Criterion: Conservation content is stored separately and flagged to prevent unauthorized modifications. Content changes are only allowed through verified updates reviewed by conservation experts.

Acceptance Tests: *Conservation Content Approval, Marine Biology Content Review*

ID: SR-04 - Player Feedback Data Integrity

Description: Ensure the integrity of player feedback data by securely storing feedback submissions to avoid accidental data loss or tampering. Aggregated feedback remains accurate and free from corruption to guide development improvements.

Rationale: To maintain reliable data that reflects players' opinions and experiences, ensuring genuine improvements to gameplay and educational content.

Fit Criterion: Feedback data is backed up and stored in a secure, dedicated database, protected from accidental modification.

Acceptance Tests: *Conservation Awareness Impact*

15c Privacy Requirements

ID: SR-05 - Player Data Collection and Consent

Description: Before collecting any personal data (e.g., for account creation, feedback, or in-game purchases), the game shall inform players of its data collection practices and obtain explicit consent. Players must be made aware of what data is being collected, its purpose, and how it will be used.

Rationale: To comply with privacy laws and build player trust by providing transparency in data collection, ensuring that players are fully informed and can make an informed choice regarding their data.

Fit Criterion: Players are presented with a data collection and usage policy prior to data submission, with an option to consent or decline.Any changes in the data collection policy prompt users to review and consent again.

Acceptance Tests: *Account Creation/Deletion, Conservation Awareness Impact*

ID SR-06 - Player Data Access and Control

Description: Players should have the right to view, update, or delete their personal information stored within the game. The game must provide a straightforward process for players to access their data, request corrections, or request deletion of their account and associated data.

Rationale: To empower players with control over their data and meet data privacy standards that require user access to and control over their personal information.

Fit Criterion: Players can access and modify their account data at any time through in-game options. Players have an option to permanently delete their data, after which a confirmation email is sent for verification.

Acceptance Tests: *Account Creation/Deletion, User Authentication*

ID: SR-07 - Data Storage and Retention

Description: The game shall store player data only as long as necessary for gameplay and game improvement purposes. Upon account deletion, all associated data shall be permanently removed from the system within a defined retention period.

Rationale: To prevent the storage of unnecessary data and to comply with privacy laws by ensuring data retention policies align with player expectations and legal standards.

Fit Criterion: Deleted accounts and their data are fully removed from the system within 30 days of the deletion request. Only aggregated, anonymized data remains in the system for analytics or game improvement.

Acceptance Tests: *Account Creation/Deletion*

ID: SR-08 - Data Sharing and Privacy Policy Transparency

Description: The game shall disclose any data-sharing practices with third parties in a clear and accessible privacy policy, specifying how and with whom data is shared (if applicable). Players must be notified of any updates to the privacy policy that affect data sharing.

Rationale: To maintain transparency and meet privacy law obligations, ensuring players are aware of and can consent to any data sharing practices.

Fit Criterion: Privacy policy is easily accessible and clearly specifies data-sharing practices. Players are notified of any privacy policy updates and asked to reconfirm consent.

Acceptance Tests: *Conservation Awareness Impact*

ID: SR-09 - Data Protection Measures

Description: Implement safeguards to protect player data from unauthorized access, including encryption of sensitive information and regular data security checks to prevent breaches.

Rationale: To ensure compliance with data protection standards and secure player data against accidental or malicious access, enhancing trust and data integrity.

Fit Criterion: Sensitive data (e.g., account credentials) is encrypted and stored securely. Regular security assessments are performed to identify and resolve vulnerabilities.

Acceptance Tests: *Quality Assurance Standards Compliance*

15d Audit Requirements

ID: SR-10 - Conservation Interaction Log

Description: The game shall retain records of player interactions with conservation content, including missions completed, educational material accessed, and any conservation-themed side quests finished. This data will be used to evaluate player engagement with conservation themes and assess the impact of the game's educational goals.

Rationale: To gather insights on how effectively the game promotes ocean conservation awareness and to identify areas for improvement in engaging players with conservation content.

Fit Criterion: Records are kept for all conservation-related interactions, including timestamps, content accessed, and mission outcomes. Data is anonymized and stored securely to respect player privacy while providing insights into educational impact.

Acceptance Tests: *Conservation Awareness Impact*

ID: SR-11 - Gameplay Activity Log

Description: The game shall retain anonymized records of significant gameplay events, such as resource depletion, survival challenges, and creature encounters. This information will support analytics on gameplay balance and difficulty, assisting in continuous improvement of the player experience.

Rationale: To identify trends in player behavior and gameplay challenges, ensuring the game provides an enjoyable and appropriately challenging experience.

Fit Criterion: Logs include events such as oxygen depletion, encounters with marine creatures, and puzzle completions. Data is retained for a defined period (e.g., 90 days) and analyzed periodically to refine gameplay balance.

Acceptance Tests: *Quality Assurance Standards Compliance*

ID: SR-12 - User Feedback Audit Trail

Description: The game shall maintain records of player feedback submitted in-game, including timestamped entries of feedback and any developer responses. This trail will support auditing for continuous improvement and ensure transparency in addressing player concerns.

Rationale: To provide accountability in how player feedback is handled and to ensure all feedback is considered in future updates and improvements.

Fit Criterion: All feedback entries are timestamped, categorized by type (e.g., gameplay, conservation content), and stored for audit purposes. Developer responses, if provided, are also recorded alongside the original feedback.

Acceptance Tests: *Conservation Awareness Impact*

ID: SR-13 - Content Update and Patch Audit Log

Description: The game shall retain a log of all content updates, patches, and downloadable content releases, detailing the changes made, purpose, and date of each update. This audit log ensures transparency in game modifications and allows players to track changes to game content over time.

Rationale: To ensure players are informed of content updates and to maintain a history of changes for accountability, especially regarding educational content or gameplay balance.

Fit Criterion: Each update is logged with details such as date, purpose, and affected game elements. Players can access a version history or update log to view past changes.

Acceptance Tests: *ESRB Content Rating, Quality Assurance Standards Compliance*

15e Immunity Requirements

ID: SR-14 - Secure Game Installation and Update Process

Description:
The game shall verify the integrity and authenticity of files during installation and updates. All game files must come from verified sources, and digital signatures should be checked to ensure they are untampered.

Rationale:
To prevent unauthorized software from being introduced during installation or updates, ensuring players download only secure and verified files, protecting the game from potential malware.

Fit Criterion: The game performs a digital signature check on all files during installation and each update. Files failing verification are flagged, and the installation/update is paused with a warning to the user.

Acceptance Tests: *Quality Assurance Standards Compliance*

ID: SR-15 - Runtime Integrity Check

Description:
The game shall perform periodic runtime integrity checks to detect and respond to unauthorized modifications or injected code. If a violation is detected, the game should prompt the player to restart and log the event for further review.

Rationale:
To ensure that the game environment remains free from external interference during gameplay, protecting the player's experience and preventing malicious code from altering gameplay.

Fit Criterion: Integrity checks are performed at regular intervals during gameplay, checking for unauthorized modifications. Any detected modifications result in a warning message to the player and a prompt to restart the game.

Acceptance Tests: *Quality Assurance Standards Compliance*

ID: SR-16 - Safe Asset Download Mechanism

Description: All downloadable content (DLC) or additional assets accessed by the game shall be scanned for known vulnerabilities and come from a trusted source. The game should ensure that no external code is embedded in these assets unless verified as safe.

Rationale: To protect the game from unauthorized access points that could be introduced through additional downloadable content or external assets.

Fit Criterion: All DLC and assets are validated against a trusted source before being available for download. Assets flagged as unsafe or unverified are blocked, and the player is informed.

Acceptance Tests: *Third-Party Asset Licensing Clearance*

ID: SR-17 - Game Session Malware Detection

Description: The game shall integrate with the player's antivirus or security software, enabling scans on running game sessions to detect any unauthorized processes or injected code that could disrupt gameplay or introduce malware.

Rationale: To ensure that the game does not inadvertently open channels for malware and works alongside security software to provide additional protection.

Fit Criterion: The game checks compatibility with the player's security software, allowing periodic scans without disrupting gameplay. If a threat is detected, the game alerts the player and suggests a system scan before continuing.

Acceptance Tests: *Quality Assurance Standards Compliance*

16 Usability and Humanity Requirements

16a Ease of Use Requirements

ID: UH-01 - Efficiency of Use

Description: The game interface and controls should allow users to efficiently navigate both on land and underwater, perform essential survival tasks, and interact with key elements (oxygen refills, tools, etc.) with minimal delay.

Rationale: Efficient interactions ensure players stay engaged and focused on the immersive experience without being hindered by complex controls or slow mechanics.

Fit Criterion: A test group of players should be able to locate and interact with essential resources (oxygen stations, tools) within three seconds of initial approach in 85% of test scenarios.

Acceptance Tests: *Speed of Essential Interaction, Navigation and Task Completion Time*

ID: UH-02 - Ease of Remembering

Description: Players should be able to easily remember the main controls and essential actions (e.g., recharging oxygen, creature avoidance) after initial gameplay.

Rationale: Ensuring players retain critical gameplay knowledge allows them to focus on the immersive and exploratory aspects rather than relearning controls, enhancing overall enjoyment.

Fit Criterion: After a two-week hiatus, 75% of a player test group should successfully recall and execute primary controls and survival actions without prompts.

Acceptance Tests: *Memory Retention Test*

ID: UH-03 - Error Rates

Description: The game should be designed to minimize user errors in critical situations, such as during stealth or resource management. Unintended errors (e.g., using a resource when it is unnecessary) should be rare and not game-breaking.

Rationale: Lower error rates enhance user satisfaction and immersion, especially in high-stress gameplay scenarios common in survival horror.

Fit Criterion: A test panel should complete at least 90% of stealth sequences and resource management tasks on their second attempt without critical errors.

Acceptance Tests: *Error Reduction Test, Resource Management Error Rate*

ID: UH-04 - Overall Satisfaction

Description: Players should experience high satisfaction with the gameplay, feeling rewarded and immersed. This satisfaction reflects positively on the game's overall design and encourages replayability.

Rationale: High satisfaction ensures players are motivated to continue, revisit, and recommend the game, which is essential for competitive success in the gaming market.

Fit Criterion: A survey should show that 80% of players report a positive gameplay experience after 30 minutes of play.

Acceptance Tests: *Satisfaction Survey, Replayability Intention*

ID: UH-05 - Feedback

Description: The game should provide clear and immediate feedback for user actions, particularly in survival and combat scenarios, so players feel confident and in control.

Rationale: Effective feedback reassures players of the impact of their actions and enhances the horror experience by emphasizing consequences and urgency in survival situations.

Fit Criterion: In post-gameplay surveys, 85% of players should indicate they understood the feedback provided for their in-game actions.

Acceptance Tests: *Player Perception of Action Feedback*

16b Personalization and Internationalization Requirements

ID: UH-06 - Language Selection

Description: The game should support multiple languages, allowing users to select their preferred language for all in-game text, subtitles, menus, and interface elements.

Rationale: Supporting multiple languages ensures accessibility to a global audience and enhances player immersion by allowing users to engage with the game in their native language.

Fit Criterion: The language menu shall allow users to select from a list of at least five major languages, with full in-game translations available for each language.

Acceptance Tests: *Language Selection Menu and Text Translation Accuracy, Subtitles and Interface Localization*

ID: UH-07 - Regional Spelling and Idioms

Description: The game should adapt spelling and idioms to align with the region selected by the user, particularly for English variations (e.g., American English vs. British English).

Rationale: Using regionally appropriate language enhances immersion and prevents confusion, allowing users to feel more connected to the game's environment and story.

Fit Criterion: In testing, users shall confirm that regional language differences (spelling and idioms) are accurately represented based on their selected region in 95% of cases.

Acceptance Tests: *Regional Language Test, Spelling and Idiom Accuracy Across Regions*

ID: UH-08 - Unit Customization

Description: Players should be able to customize measurement units (e.g., depth in meters or feet) based on their regional preferences.

Rationale: Allowing users to select familiar units (such as metric or imperial) enhances accessibility, making gameplay metrics like depth and pressure more intuitive.

Fit Criterion: Players shall be able to select between metric and imperial units from the settings menu, with the chosen units displayed consistently throughout gameplay.

Acceptance Tests: *Unit Configuration Test, Unit Selection Consistency, Measurement Display Accuracy*

ID: UH-09 - Personal Configuration Options

Description: Players should have options to personalize certain gameplay settings, such as HUD visibility, sound levels, and visual effects to optimize for personal preferences and hardware capabilities.

Rationale: Personalization options allow players to tailor their experience to their preferences and ensure compatibility with a range of devices, enhancing both immersion and accessibility.

Fit Criterion: In usability testing, 90% of users should successfully configure at least three personalization options (e.g., HUD elements, sound levels, and visual effects) within 5 minutes of accessing the settings menu.

Acceptance Tests: Personalization Test ID#009A – "HUD Customization," Test ID#009B – "Sound and Visual Effect Adjustments"

ID: UH-10 - Currency for In-Game Purchases

Description: If the game includes an in-game store for additional content, it should support multiple currencies and display symbols according to user region.

Rationale: Using the correct currency symbols and conventions prevents confusion, especially in regions with different decimal or currency formats, enhancing user trust in purchase accuracy.

Fit Criterion: The in-game store shall display currency symbols and decimal conventions according to the user's selected region in 98% of test cases.

Acceptance Tests: Currency Display Test ID#010A – "Accuracy of Regional Currency Symbols and Decimal Placement"

16c Learning Requirements

ID: UH-11 - Basic Gameplay Familiarization

Description: New players should be able to learn essential gameplay mechanics (movement, interaction, and basic survival elements) within the first few minutes of play without requiring additional guidance or tutorials.

Rationale: Given that this game targets a broad audience, it’s crucial for players to grasp basic mechanics quickly to enjoy the horror and exploration without frustration.

Fit Criterion: Ninety percent of a test panel should complete a tutorial scenario covering movement, resource collection, and basic survival tasks within 10 minutes of first gameplay.

Acceptance Tests: Learning Test ID#011A – "Tutorial Completion Time," Test ID#011B – "Unassisted Basic Task Execution"

ID: UH-12 - Advanced Survival Skills Introduction

Description: Players should learn advanced survival skills, such as oxygen management, stealth, and avoiding dangerous sea creatures, within the first 30 minutes of gameplay.

Rationale: The game's survival mechanics are core to the experience, and players need to be proficient with these to succeed. A gradual learning curve for more complex skills will ensure they feel prepared for later challenges.

Fit Criterion: Eighty percent of a test group should successfully demonstrate understanding of advanced survival mechanics (such as stealth and oxygen management) after 30 minutes of gameplay.

Acceptance Tests: Advanced Skills Test ID#012A – "Successful Execution of Survival Tasks," Test ID#012B – "Stealth and Oxygen Management Proficiency"

ID: UH-13 - In-Game Help and Hints

Description: The game should provide optional, context-sensitive hints and help sections that players can access to better understand tasks or navigate challenging areas.

Rationale: Context-sensitive hints allow players to seek help as needed without disrupting immersion, particularly beneficial for players who may not be familiar with survival horror mechanics.

Fit Criterion: Ninety percent of players accessing the help feature should report successfully completing the relevant task after using the in-game hints.

Acceptance Tests: *Help Functionality Test, Feedback Survey*

ID: UH-14 - Self-Guided Journal Exploration

Description: Players should be able to navigate and understand the in-game journal, which includes marine biology information, game lore, and environmental data, without extensive tutorial guidance.

Rationale: An easily navigable journal enhances immersion and learning, allowing players to explore educational content independently while reinforcing game lore.

Fit Criterion: Eighty-five percent of a test panel should be able to locate specific entries in the journal and retrieve relevant information within three minutes of accessing it.

Acceptance Tests: *Journal Navigation Test, Information Retrieval Efficiency, User Self-Guidance in Journal Use*

ID: UH-15- Interactive Tutorial for Dual Environments

Description: The game should include an interactive tutorial that introduces players to both land and underwater controls and mechanics.

Rationale: A tailored tutorial for each environment ensures that players are comfortable with the unique challenges and mechanics specific to each setting, improving their gameplay experience.

Fit Criterion: Seventy-five percent of players should complete the interactive tutorial covering both environments within 20 minutes, demonstrating confidence in basic mechanics.

Acceptance Tests: *Dual Environment Test, Tutorial Completion for Land and Underwater, Environment-Specific Task Success Rate*

16d Understandability and Politeness Requirements

ID: UH-16 - Intuitive Language and Symbol Use

Description: The game should use language and symbols that are immediately recognizable and clear to players, avoiding technical jargon or overly complex terms in UI elements, instructions, and feedback messages.

Rationale: Using intuitive language and symbols ensures players understand instructions and game elements without requiring prior knowledge of technical or scientific terms, improving immersion and reducing frustration.

Fit Criterion: Ninety percent of players should correctly interpret icons, prompts, and symbols on their first encounter without additional explanation.

Acceptance Tests: Symbol Interpretation Test, Language Clarity Test, Prompt and Message Comprehension

ID: UH-17 - Context-Specific Notifications

Description: Messages and notifications should provide clear, context-specific guidance, such as explaining why a player cannot perform an action or providing subtle hints when players seem lost.

Rationale: Context-sensitive messages help players understand game mechanics and restrictions without breaking immersion, making the experience feel natural and supportive rather than obstructive.

Fit Criterion: In usability testing, 80% of players should understand the reason behind restricted actions or receive helpful hints when idle in challenging areas within five seconds of notification.

Acceptance Tests: *Contextual Help Test, Message Relevance and Comprehensibility, Restriction Clarity Test, Understanding of Action Limitations*

ID: UH-18 - Consistent Tone and Language

Description: The game should maintain a consistent tone in its language, reflecting the survival horror genre, and avoiding overly complex or academic language that might disrupt the immersion.

Rationale: A consistent tone reinforces the game's atmosphere and keeps players engaged in the story, helping them understand the stakes and feel connected to the narrative.

Fit Criterion: In surveys, 85% of players should report that they felt the language and tone aligned with the game's atmosphere and did not interfere with their experience.

Acceptance Tests: *Tone Consistency Survey, "Language Impact on Atmosphere*

ID: UH-19 - Minimized Technical Terminology

Description: All in-game instructions, tooltips, and notifications should avoid technical terminology that players are unlikely to know, unless explained in the context (e.g., defining "bioluminescence" within the narrative if referenced).

Rationale: Minimizing technical terms prevents players from feeling alienated by language they don't understand, ensuring the game is accessible to a wide audience and keeps them focused on gameplay.

Fit Criterion: Eighty-five percent of players should successfully interpret gameplay instructions and notifications without needing to look up terms outside the game.

Acceptance Tests: *Terminology Test, Clarity Test, Player Feedback on Language Accessibility*

ID: UH-20 - "Polite" System Feedback

Description: System feedback, including error messages or guidance prompts, should be polite and supportive, avoiding abrupt or overly negative language that could frustrate or demotivate the player.

Rationale: Friendly and encouraging feedback maintains player engagement and reduces frustration, especially important in high-stress survival horror settings where players may encounter frequent challenges.

Fit Criterion: Ninety percent of players in surveys should report that they felt the game's feedback messages were clear, constructive, and supportive.

Acceptance Tests: *Politeness Survey, User Reaction to Error and Guidance Prompts*

16e Accessibility Requirements

ID: UH-21 - Visual Accessibility Options

Description: The game should include options for players with visual impairments, such as adjustable text size, high-contrast UI modes, and a colorblind mode to address common forms of color vision deficiency (e.g., red-green).

Rationale: Providing visual accessibility options ensures that visually impaired players, including those with colorblindness, can navigate menus, read text, and interact with in-game elements effectively.

Fit Criterion: Ninety percent of visually impaired players in accessibility testing should be able to adjust settings and interact with all essential gameplay elements without difficulty.

Acceptance Tests: *Visual Accessibility Test, Colorblind Accessibility Test*

ID: UH-22 - Subtitles and Audio Descriptions

Description: The game should offer customizable subtitles (adjustable size, background opacity, and font) and audio descriptions for important environmental cues, such as creature sounds or spatial directions.

Rationale: Customizable subtitles and audio descriptions ensure players with hearing impairments can fully engage with the storyline and are aware of critical sounds that contribute to survival.

Fit Criterion: Eighty-five percent of hearing-impaired players should report that they understood narrative and gameplay audio cues via subtitles or audio descriptions.

Acceptance Tests: *Subtitle Test, Audio Description Test*

ID: UH-23 - Motor Accessibility Options

Description: The game should offer remappable controls and an option for simplified input to assist players with limited mobility or motor impairments.

Rationale: Remapping and simplified input options make the game more accessible to players with limited dexterity, allowing them to customize controls to suit their needs.

Fit Criterion: Ninety percent of players in motor accessibility testing should successfully complete movement and interaction tasks after adjusting controls to their preferred configuration.

Acceptance Tests: *Control Customization Test, Simplified Input Test, Accessibility of Simplified Control Options*

ID: UH-24 - Cognitive Accessibility and Assistance Mode

Description: The game should include an optional assistance mode that offers simplified puzzles, additional hints, and slower oxygen depletion rates to support players with cognitive impairments or those who may struggle with complex mechanics.

Rationale: Cognitive accessibility options allow players to enjoy the game at a manageable pace and avoid frustration with challenging mechanics, making the game enjoyable for a wider audience.

Fit Criterion: Eighty percent of players in cognitive accessibility testing should report being able to complete gameplay tasks with reduced difficulty and increased satisfaction.

Acceptance Tests: Assistance Mode Test ID#024A – "Effectiveness of Simplified Mechanics," Cognitive Load Test ID#024B – "User Satisfaction with Additional Hints and Simplified Tasks"

ID: UH-25 - Screen Reader Compatibility for Menus and Text

Description: Menus, text, and in-game journals should be compatible with screen readers to aid players with significant visual impairments.

Rationale: Screen reader compatibility ensures visually impaired players can navigate menus and access text-based game elements, such as the in-game journal, enhancing their experience.

Fit Criterion: Ninety percent of screen-reader users should be able to successfully navigate menus and read text elements during accessibility testing.

Acceptance Tests: *Screen Reader Test, Journal Accessibility Test, Screen Reader Compatibility with In-Game Text*

16f User Documentation Requirements

ID: UH-26 - User Manual

Description: A comprehensive user manual will be provided, covering game installation, basic controls, gameplay mechanics, and frequently asked questions. This manual will be available digitally within the game and as a downloadable PDF.

Rationale: The user manual will provide players with essential information to get started, troubleshoot basic issues, and understand core mechanics, enhancing the user experience.

Fit Criterion: Ninety percent of players should be able to locate and understand basic gameplay instructions from the user manual within five minutes.

Acceptance Tests: *Documentation Test, User Manual Accessibility, Content Comprehension Test*

ID: UH-27 - Quick Start Guide

Description: A concise quick start guide will be included to help new players get started quickly, covering installation, setup, and the first gameplay steps. This will be provided as a digital document accessible from the game launcher.

Rationale: The quick start guide allows players to begin gameplay without needing to read the entire manual, making it easier for casual players to get into the game.

Fit Criterion: Eighty percent of new users should successfully complete installation and initial gameplay setup within ten minutes using the quick start guide.

Acceptance Tests: *Quick Start Test, Installation and Setup Success Rate, Initial Gameplay Test, Ease of First-Time Use with Guide*

ID: UH-28- In-Game Help System

Description: An in-game help system will provide context-sensitive tips, definitions of key terms, and navigation help within menus. This will be accessible at any time via the pause menu.

Rationale: An in-game help system reduces the need for players to exit gameplay to reference external documentation, keeping them immersed while still receiving guidance as needed.

Fit Criterion: Ninety percent of players should report that the in-game help system provided useful information in a timely manner without needing to exit the game.

Acceptance Tests: *Help System Test, Player Satisfaction with In-Game Tips, Context-Sensitivity Test, Relevance of Contextual Help*

ID: UH-29 - Accessibility Guide

Description: An accessibility guide will detail all customization options available for users with disabilities, explaining settings like colorblind mode, subtitle customization, and control remapping.

Rationale: The accessibility guide ensures that players with disabilities can quickly identify and configure settings to improve their gameplay experience.

Fit Criterion: In accessibility testing, 90% of users with disabilities should be able to locate and apply relevant settings using the guide within five minutes.

Acceptance Tests: *Accessibility Documentation Test, Ease of Finding Accessibility Options, Configuration Success Test, User Satisfaction with Accessibility Settings*

ID: UH-30 - Troubleshooting and Support Guide

Description: A troubleshooting guide will include solutions for common technical issues, such as installation problems, performance optimization, and connection issues. It will also provide support contact information.

Rationale: Providing a detailed troubleshooting guide reduces frustration by helping players resolve common issues on their own, enhancing user satisfaction.

Fit Criterion: Eighty-five percent of players should be able to resolve basic technical issues within 10 minutes using the troubleshooting guide.

Acceptance Tests: Troubleshooting Test, Issue Resolution Rate, Support Access Test, Ease of Finding Support Information

ID: UH-31 - Patch Notes and Update Documentation

Description: Patch notes and update logs will be provided with each game update, detailing new features, bug fixes, and gameplay adjustments.

Rationale: Clear patch notes keep players informed of game changes and allow them to stay up-to-date with the latest features, building trust and transparency.

Fit Criterion: Patch notes should be available on the game's main menu and official website within 24 hours of any major update release.

Acceptance Tests: *Patch Note Visibility Test, Content Clarity Test*

16g Training Requirements

ID: UH-32 - Basic Gameplay Training Module

Description: An optional, interactive training module will guide players through basic gameplay mechanics, such as movement, interaction, resource collection, and managing survival elements like oxygen levels.

Rationale: Introducing players to core gameplay mechanics in a structured way ensures they can start playing confidently without frustration, enhancing their immersion and enjoyment.

Fit Criterion: Ninety percent of players should complete the training module within 15 minutes, demonstrating proficiency in basic controls and survival tasks.

Acceptance Tests: *Training Module Test, User Proficiency Test*

ID: UH-33 - Advanced Survival Skills Training

Description: An optional advanced training module will cover skills like stealth, managing threats from underwater creatures, and using specialized tools (e.g., flares or repurposed lab equipment) to evade or distract enemies.

Rationale: Advanced skills are critical for survival in the game, and this training will allow players to practice them in a controlled environment, building confidence before they face these challenges in the main gameplay.

Fit Criterion: Eighty percent of players should complete the advanced training module within 20 minutes, successfully demonstrating stealth and threat management skills.

Acceptance Tests: *Survival Skills Training, Stealth Success Rate Test*

ID: UH-34 - Accessibility Settings Training

Description: A brief training session will guide players through the various accessibility settings, showing them how to adjust features like colorblind mode, subtitles, remappable controls, and visual contrasts.

Rationale: This training ensures that players with disabilities or specific needs can customize the game for optimal accessibility, enhancing their experience and comfort.

Fit Criterion: Ninety percent of players in accessibility testing should be able to locate and apply at least three accessibility settings within five minutes of starting the training.

Acceptance Tests: *Accessibility Training Test, User Satisfaction Test*

ID: UH-35 - In-Game Journal and Educational Content Tutorial

Description: A short tutorial will introduce players to the in-game journal, guiding them on how to access and understand educational content related to marine biology and environmental themes.

Rationale: Helping players navigate the journal and educational content enriches their gameplay experience and promotes awareness of the game's underlying environmental themes.

Fit Criterion: Eighty-five percent of players should be able to locate and access educational entries in the journal within five minutes after completing the tutorial.

Acceptance Tests: *Journal Navigation Test, Educational Content Test, Comprehension and Engagement with Journal Entries*

ID: UH-36 - Post-Launch Updates Training

Description: In case of major gameplay changes or added features post-launch, short tutorials or tooltips will guide returning players through any significant updates.

Rationale: Updating players on new features or mechanics ensures that returning players can seamlessly adapt to changes without feeling lost or frustrated by unfamiliar elements.

Fit Criterion: After major updates, 80% of returning players should be able to identify and understand new features or changes within five minutes of first encounter.

Acceptance Tests: *Update Familiarization Test, User Adaptability to New Features, Post-Update Satisfaction Survey, Ease of Transition After Updates*

17 Look and Feel Requirements

17a Appearance Requirements

ID: LF-01 - Oceanic Color Palette

Description: The game's color scheme will primarily use deep blues, greens, and dark tones to reflect the oceanic environment, with bioluminescent highlights in creatures and plants for contrast and visual interest.

Rationale: A color palette inspired by the ocean enhances immersion and aligns with the underwater theme, evoking the vast and mysterious nature of the deep sea.

Fit Criterion: Ninety percent of players in a test group should describe the color palette as "immersive" and "reflective of the oceanic environment" after 10 minutes of gameplay.

Acceptance Tests: *Color Scheme Test, User Feedback on Color Palette, Visual Consistency Test, Consistency of Oceanic Theme in Color Use*

ID: LF-02 - Horror-Inspired Font and UI Style

Description: Fonts and UI elements should have a slightly distressed or eerie style while remaining legible. Titles, menus, and journal entries should evoke the horror theme without sacrificing readability.

Rationale: A font and UI style that reflects the horror genre enhances the game's atmosphere and reinforces the unsettling, mysterious tone of the game.

Fit Criterion: Eighty-five percent of players should report that the font and UI style enhance the horror theme without impacting readability or usability.

Acceptance Tests: *Font Style Test, User Feedback on Font Appropriateness and Readability, UI Atmosphere Test, Perceived Consistency with Horror Theme*

ID: LF-03 - Bioluminescent Accents for Underwater Elements

Description: Underwater creatures and plants should include bioluminescent accents to create striking visuals, aiding player navigation and enhancing the eerie beauty of the underwater world.

Rationale: Bioluminescence not only adds to the atmosphere but also serves a practical purpose, guiding players through dark, underwater environments and contributing to the horror theme by illuminating the unknown.

Fit Criterion: Ninety percent of players in a visual test group should rate the bioluminescent accents as visually appealing and thematically fitting for an underwater horror setting.

Acceptance Tests: *Bioluminescent Effect Test, Player Response to Visual Appeal, Navigation Aid Test, Usefulness of Bioluminescent Accents for Navigation*

ID: LF-04 - Branding and Logo Design

Description: The game's logo and branding materials should be designed to evoke themes of ocean exploration and horror, using colors and fonts consistent with the game's overall aesthetic.

Rationale: A thematic logo and branding reinforce the game's identity and help attract players who are interested in horror and oceanic themes.

Fit Criterion: The branding department should certify that the logo and branding materials align with the game's aesthetic and themes.

Acceptance Tests: *Branding Compliance Test, Branding Department Certification, Player Attraction Test, Initial Reaction to Logo and Branding*

ID: LF-05 - Dynamic Lighting Effects

Description: The game should include dynamic lighting effects, especially underwater, to simulate light refraction, shadows, and darkness for a realistic, immersive experience.

Rationale: Dynamic lighting enhances immersion and the horror atmosphere, as the interplay of light and shadow can heighten fear and suspense.

Fit Criterion: In a survey, 85% of players should report that the lighting effects significantly contribute to the immersion and horror elements of the game.

Acceptance Tests: *Lighting Test, Player Immersion with Lighting Effects, Atmosphere Test, Impact of Lighting on Horror Experience*

17b Style Requirements

ID: LF-06 - Suspenseful and Foreboding Atmosphere

Description: The game should evoke a constant sense of suspense and dread, emphasizing the vast, unknown nature of the deep sea and the lurking dangers within it. Sound design, visuals, and pacing should contribute to this feeling.

Rationale: An atmosphere that instills suspense and foreboding aligns with the horror genre and immerses players in the game's unsettling underwater environment, enhancing the emotional impact.

Fit Criterion: In a post-game survey, 85% of players should report feeling suspense or unease throughout gameplay, particularly during underwater exploration.

Acceptance Tests: *Atmosphere Test, Player Reaction to Suspense Elements, Emotional Impact Test, Sense of Dread in Underwater Sections*

ID: LF-07 - Mysterious and Intriguing Visual Style

Description: The visual style should blend realistic oceanic visuals with eerie, mysterious elements, such as dark, uncharted waters and decayed underwater structures. This style should captivate players while keeping them on edge.

Rationale: A mysterious visual style draws players into the story, prompting them to explore deeper while enhancing the horror experience through atmospheric elements that suggest an unknown history.

Fit Criterion: Seventy-five percent of a test panel should describe the visual style as "mysterious" or "intriguing" after 15 minutes of gameplay.

Acceptance Tests: *Visual Style Test, User Perception of Visual Intrigue, Environmental Detail Test, Consistency with Mysterious Atmosphere*

ID: LF-08 - Isolation and Vulnerability

Description: The gameplay experience should create a sense of isolation and vulnerability, especially in deep-sea environments, through limited lighting, minimal NPC interactions, and challenging survival mechanics.

Rationale: Emphasizing isolation enhances the horror experience, making players feel more vulnerable in the vast ocean and heightening the stakes in survival situations.

Fit Criterion: Eighty percent of players in immersion testing should report feeling isolated and vulnerable during underwater gameplay sections.

Acceptance Tests: *Immersion Test, User Experience of Isolation, Vulnerability Test, Effectiveness of Survival Mechanics in Enhancing Vulnerability*

ID: LF-09 - Realism with Subtle Horror

Description: The game's style should emphasize realism in its depiction of ocean life and environments but introduce subtle horror elements, such as mutated creatures or decayed facilities, to keep players on edge.

Rationale: A realistic yet subtly terrifying style ensures players are immersed in the ocean environment while also primed for the horror elements, making encounters feel unpredictable and unsettling.

Fit Criterion: Eighty-five percent of players should report that the realistic visuals and subtle horror elements enhanced the sense of immersion and fear.

Acceptance Tests: *Realism Test, User Perception of Ocean Realism, Subtle Horror Test, Effectiveness of Mutated Creature Design*

ID: LF-10 - Minimalist and Intuitive HUD

Description: The HUD (heads-up display) should be minimalist, displaying only essential information like oxygen levels, health, and limited inventory to avoid disrupting the immersive experience.

Rationale: A minimalist HUD minimizes distractions, allowing players to focus on the environment and heightening the horror experience by limiting visible information.

Fit Criterion: Ninety percent of players should find the HUD clear, unobtrusive, and informative during gameplay.

Acceptance Tests: *HUD Usability Test, Clarity and Usefulness of HUD, Minimalism Test, Impact of HUD on Immersion*

18 Operational and Environmental Requirements

18a Expected Physical Environment

ID: OE-01 - Usability in Low-Light Conditions

Description: The game interface should be fully usable in dim lighting, with an optional dark mode for menus and an interface that remains legible without excessive brightness.

Rationale: Many players prefer to play horror games in low-light environments to enhance immersion, so the game must remain visually clear in such settings.

Fit Criterion: Ninety percent of players should report no difficulty in reading text or navigating menus when playing in low-light environments.

Acceptance Tests: *Low-Light Usability Test, Menu and Interface Legibility in Dark Mode," Visual Accessibility Test, User Experience in Dim Lighting*

ID: OE-02 - Audio Balance for Quiet and Noisy Environments

Description: The game's audio settings should allow players to clearly hear essential sounds and cues, even in environments with varying noise levels. Adjustable volume and sound balance settings will be provided.

Rationale: Players may play the game in both quiet and noisy environments, and critical sound cues (e.g., creature proximity) must be audible to maintain the horror experience.

Fit Criterion: Eighty-five percent of players should report that they could hear and identify essential sounds, such as cues and dialogue, in both quiet and noisy environments after adjusting settings.

Acceptance Tests: *Audio Balance Test, Audibility of Key Sounds in Different Noise Levels, Sound Cue Clarity Test, Player Awareness of Critical Audio Cues*

ID: OE-03 - Compatible with Multiple Display Sizes

Description: The game should be fully functional and visually optimized for a range of display sizes, from standard monitors to widescreen and ultra-widescreen setups.

Rationale: Ensuring compatibility with different screen sizes allows more players to enjoy the game as intended, regardless of their monitor setup.

Fit Criterion: Ninety percent of players using various display sizes should report a consistent and visually pleasing experience.

Acceptance Tests: *Display Compatibility Test, Visual Consistency Across Display Sizes, Widescreen Functionality Test, Game Performance and Visuals on Ultra-Widescreen Monitors*

ID: OE-04 - Performance Optimization for Indoor and Outdoor Settings

Description: The game should maintain stable performance and visual clarity in both indoor (home) and outdoor (mobile/laptop) settings, where lighting and noise levels can vary.

Rationale: Many players may choose to play on laptops in diverse locations, so the game must function smoothly without requiring ideal conditions for immersion and performance.

Fit Criterion: In testing, 90% of users playing on laptops in various environments should experience consistent frame rates and visuals.

Acceptance Tests: *Performance Test, Frame Rate Stability in Variable Lighting and Noise Conditions, Visual Quality Test, Clarity and Consistency in Indoor vs. Outdoor Settings*

ID: OE-05 - VR Mode Suitability for Seated and Standing Play

Description: If the game includes a VR mode, it should be optimized for both seated and standing play, allowing players to adjust comfort settings for either option.

Rationale: VR users may prefer different setups based on their comfort and space constraints, so the game should accommodate both seated and standing gameplay.

Fit Criterion: Eighty percent of VR players should report a comfortable and immersive experience in both seated and standing positions.

Acceptance Tests: *VR Mode Test, User Comfort in Seated Play, Standing Mode Test, Immersion and Stability in Standing Play*

18b Requirements for Interfacing with Adjacent Systems

ID: OE-06- Compatibility with VR Headsets

Description: The game must be compatible with popular VR headsets, including the latest versions of Oculus, HTC Vive, and PlayStation VR, supporting their standard interface and control systems.

Rationale: VR compatibility offers players a more immersive experience and allows them to use their preferred VR devices for the game, broadening the user base.

Fit Criterion: Ninety percent of VR test cases should show stable functionality with Oculus, HTC Vive, and PlayStation VR systems, including compatibility with each device's control scheme and standard VR features.

Acceptance Tests: *VR Compatibility Test, Connection Stability with VR Headsets, Control Interface Test, Device-Specific Control Functionality*

ID: OE-07 - Integration with Game Streaming Platforms

Description: The game should integrate with game streaming platforms, such as Twitch and YouTube, enabling features like real-time broadcasting, capturing gameplay highlights, and streaming-specific HUD options.

Rationale: Streaming support allows players to share their experience with an audience, increasing game visibility and engagement within the gaming community.

Fit Criterion: The game should allow users to connect and broadcast to Twitch, YouTube, and other major streaming platforms within 60 seconds of setup.

Acceptance Tests: *Streaming Integration Test, Connectivity with Streaming Platforms, Broadcast Functionality Test, Stability During Live Streaming*

ID: OE-08 - Cloud Save Compatibility

Description: The game should support cloud saves through platforms like Steam, PlayStation Network, and Xbox Live, allowing players to save and resume gameplay across different devices.

Rationale: Cloud save compatibility enhances player convenience by allowing users to seamlessly continue their progress on different devices or after reinstalling the game.

Fit Criterion: Ninety-five percent of cloud save test cases should verify that saved data can be accessed across multiple platforms without data loss or corruption.

Acceptance Tests: *Cloud Save Test, Cross-Device Save Access, Data Integrity Test*

ID: OE-09 - Support for Game Controllers and Keyboard/Mouse

Description: The game should be fully compatible with popular game controllers (Xbox, PlayStation, etc.) and standard keyboard/mouse setups, allowing users to choose their preferred control method.

Rationale: Supporting both controllers and keyboard/mouse setups increases accessibility and allows players to use their preferred control schemes.

Fit Criterion: Ninety percent of players should be able to connect and play with an Xbox or PlayStation controller, as well as with a keyboard and mouse, with no loss of functionality.

Acceptance Tests: *Controller Compatibility Test, Functionality with Game Controllers, Keyboard/Mouse Compatibility Test, Full Control Functionality with Keyboard/Mouse*

ID: OE-10 - Cross-Platform Multiplayer Support

Description: If the game includes multiplayer functionality, it should support cross-platform play across PC and major consoles, including PlayStation, Xbox, and Nintendo Switch.

Rationale: Cross-platform multiplayer increases the player base and makes it easier for friends on different systems to play together, enhancing the game's social aspect.

Fit Criterion: Ninety percent of multiplayer test sessions should successfully connect players across different platforms without compatibility issues or excessive latency.

Acceptance Tests: *Cross-Platform Test, Successful Cross-Platform Connections, Latency Test, Performance Stability Across Platforms*

ID: OE-11 - Data Export for Environmental Awareness Campaigns

Description: The game should support exporting player progress and completion data for integration with partner environmental awareness campaigns, enabling players to track and contribute to real-world conservation efforts.

Rationale: Allowing data export for environmental partnerships enhances the game's educational impact and enables players to engage with real-world conservation actions.

Fit Criterion: Ninety percent of test cases should verify accurate data export (e.g., completion milestones) compatible with external systems used by environmental organizations.

Acceptance Tests: *Data Export Test, Accuracy of Exported Player Data, Compatibility Test, Integration with Environmental Partner Systems*

18c Productization Requirements

ID: OE-12 - Digital Distribution Format

Description: The game shall be distributed as a downloadable digital product through major online platforms, including Steam, PlayStation Store, Xbox Live, and Nintendo eShop.

Rationale: Digital distribution allows for broad access to the game across various platforms and simplifies the installation process for users.

Fit Criterion: The game must be available for download on the specified platforms within one day of the official release date.

Acceptance Tests: *Distribution Test, Availability on Major Platforms, Download Test, Successful Download and Installation from Each Platform*

ID: OE-13 - Installation Process for Untrained Users

Description: The installation process should be straightforward, allowing users to install the game without needing technical knowledge or printed instructions. The process should include an automatic setup wizard for PC and standard installation methods for consoles.

Rationale: A simple installation process makes the game accessible to a wider audience, preventing technical barriers that could deter potential players.

Fit Criterion: Ninety percent of users in usability testing should be able to complete installation without external assistance within five minutes.

Acceptance Tests: *Installation Test, Installation Success Rate for Untrained Users, Setup Wizard Test, Usability and Simplicity of Installation Process*

ID: OE-14 - Platform-Specific Storage Requirements

Description: The game should be optimized to fit within storage limitations typical for each platform (e.g., less than 40GB for PC, PlayStation, and Xbox, and less than 20GB for Nintendo Switch).

Rationale: Optimized storage requirements allow the game to be accessible on devices with limited storage capacity, enhancing the user experience across all platforms.

Fit Criterion: The game's file size should meet the platform-specific storage limits in 100% of testing scenarios across supported devices.

Acceptance Tests: *Storage Requirement Test, File Size Compliance Across Platforms, Installation Success Test, Compatibility with Platform Storage Limits*

ID: OE-15 - Automatic Updates

Description: The game should support automatic updates, ensuring players receive the latest patches, features, and bug fixes without manual intervention.

Rationale: Automatic updates ensure that all players have the most current version of the game, reducing the potential for gameplay issues and improving overall stability and performance.

Fit Criterion: Ninety percent of users should report that updates are installed automatically without impacting gameplay or requiring manual intervention.

Acceptance Tests: *Update Test, Automatic Update Functionality, User Feedback Test, Satisfaction with Update Process*

ID: OE-16 - Digital Rights Management (DRM) Protection

Description: The game should include DRM protection to prevent unauthorized access and distribution, using industry-standard methods compatible with each distribution platform.

Rationale: DRM protection helps protect the game's intellectual property, ensuring that only paying customers can access it and reducing potential revenue loss from piracy.

Fit Criterion: DRM checks should pass in 95% of test scenarios across supported platforms, preventing unauthorized access while remaining unobtrusive to legitimate users.

Acceptance Tests: *DRM Functionality Test, Unauthorized Access Prevention Rate, User Experience Test, Impact of DRM on Gameplay Experience*

ID: OE-17 - Cross-Platform Save Transfer

Description: If feasible, the game should allow players to transfer saved data between platforms (e.g., from PC to console), enabling flexibility for players who wish to switch devices.

Rationale: Cross-platform save transfer enhances convenience, allowing players to continue their game progress across devices without needing to restart.

Fit Criterion: In testing, 85% of users should be able to transfer saves between two or more supported platforms without data loss.

Acceptance Tests: *Save Transfer Test, Data Integrity During Save Transfer, Cross-Platform Compatibility Test, User Experience with Cross-Platform Saves*

18d Release Requirements

ID: OE-18 - Initial Release and Major Updates

Description: The game will have an initial release followed by two major content updates within the first year. These updates may include additional story content, new gameplay mechanics, or environmental expansions.

Rationale: Major updates keep the game fresh, encouraging continued player engagement by adding new features and content over time.

Fit Criterion: The initial release should be followed by the first major update within six months, with the second update within the following six months.

Acceptance Tests: *Update Schedule Test, Adherence to Major Update Timeline, Content Expansion Test, Successful Integration of New Features*

ID: OE-19 - Quarterly Maintenance Patches

Description: Quarterly maintenance patches will be provided to address any bugs, optimize performance, and make minor adjustments to game balance as needed.

Rationale: Regular maintenance ensures the game remains stable, bug-free, and enjoyable for players by promptly addressing any technical issues or imbalances.

Fit Criterion: A maintenance patch should be released every three months, and each patch should address at least 90% of reported issues from the previous quarter.

Acceptance Tests: *Patch Release Test, Timeliness of Quarterly Patches, Bug Fix Effectiveness Test, Resolution Rate of Reported Issues*

ID: OE-20 - Emergency Hotfixes for Critical Issues

Description: Emergency hotfixes will be deployed as needed for any critical issues that severely impact gameplay, stability, or user security, within 48 hours of identifying the issue.

Rationale: Quickly addressing critical issues preserves player trust and prevents disruptions in gameplay, especially for issues that impact large portions of the player base.

Fit Criterion: Ninety percent of critical issues should be resolved with a hotfix within 48 hours of discovery.

Acceptance Tests: *Hotfix Deployment Test, Time to Resolution for Critical Issues, Stability Test, Impact of Hotfixes on Gameplay and Stability*

ID: OE-21 - Backward Compatibility for Updates

Description: Each update or patch should be backward compatible, ensuring that new releases do not cause previous game features to fail or corrupt saved game data.

Rationale: Backward compatibility prevents disruptions in gameplay, ensuring players retain their progress and enjoy consistent functionality across updates.

Fit Criterion: Ninety-five percent of users should report no issues with previous features or saved data following an update.

Acceptance Tests: Compatibility Test, Feature Integrity Post-Update, Save Data Stability Test, Data Preservation Across Updates

ID: OE-22 - Annual Performance Review and Optimization

Description: An annual review of the game's performance will be conducted to identify areas for optimization, such as load times, graphics performance, and overall system stability. An optimization update will be provided if necessary.

Rationale: Regular performance reviews ensure the game remains smooth and responsive, especially as updates accumulate and platform requirements evolve.

Fit Criterion: Ninety percent of players should report improved or consistent performance following the annual optimization update.

Acceptance Tests: *Performance Review Test, Annual Performance Assessment Results, User Satisfaction Test, Impact of Optimization on Gameplay Experience*

19 Cultural and Political Requirements

19a Cultural Requirements

ID: CP-01 - Culturally Neutral Content and Imagery

Description: The game should avoid content, symbols, or imagery that may be offensive or inappropriate for players from diverse cultural or religious backgrounds.

Rationale: Ensuring the game remains culturally neutral avoids alienating or offending players, making the game accessible and respectful to a global audience.

Fit Criterion: In cultural sensitivity testing, 95% of testers from various cultural and religious backgrounds should report no offensive or inappropriate content.

Acceptance Tests: *Cultural Sensitivity Test, Perception of Content and Imagery, International Feedback Test, Acceptability Across Cultural Contexts*

ID: CP-02 - Localization of Language and Idioms

Description: The game should localize in-game text, instructions, and dialogue to match cultural norms and language idioms appropriate for each region, especially for non-English-speaking markets.

Rationale: Proper localization enhances accessibility, allowing players to experience the game in a culturally relevant way that feels natural and immersive.

Fit Criterion: Ninety percent of players in localization testing should report that language and idioms in their region's version feel accurate and culturally relevant.

Acceptance Tests: *Localization Accuracy Test, Language and Idiom Relevance, Regional Playability Test, User Experience in Localized Versions*

ID: CP-03 - Color and Symbol Sensitivity

Description: The game should avoid colors, symbols, or visual elements that have culturally sensitive or negative meanings in certain regions. For instance, red may signify danger or warnings, but has other connotations in various cultures.

Rationale: Being mindful of color and symbol meanings prevents unintended messages and makes the game universally acceptable without cultural misinterpretations.

Fit Criterion: In testing, 90% of players across cultural regions should report no issues with color and symbol use within the game.

Acceptance Tests: *Visual Sensitivity Test, Interpretation of Color and Symbol Use, Regional Symbol Testing, Feedback on Cultural Relevance of Symbols*

ID: CP-04 - Respect for Environmental and Marine Conservation Beliefs

Description: The game should align with and promote global environmental values, particularly regarding marine conservation, avoiding content that could be seen as harmful to environmental causes.

Rationale: Many cultures place importance on environmental conservation, especially regarding ocean ecosystems. Emphasizing sustainable and respectful environmental themes will align with these values.

Fit Criterion: In testing, 85% of users across different cultures should report that the game's environmental themes align with positive conservation values and show respect for marine ecosystems.

Acceptance Tests: *Environmental Theme Test, Alignment with Marine Conservation Values, User Feedback Test, Reception of Environmental Message*

ID: CP-05 - Adaptation to Culturally Relevant In-Game Holidays and Events

Description: The game should recognize and adapt to relevant in-game holiday themes or events if possible, such as seasonal changes that might align with major global holidays or events.

Rationale: Adapting to significant cultural events helps increase engagement and relatability, making the game feel more connected to players' lives and traditions.

Fit Criterion: For each regional version, 80% of players should recognize and appreciate culturally relevant holiday-themed events or seasonal changes.

Acceptance Tests: *Cultural Relevance Test, Recognition of Regional Holidays, Engagement Test, User Participation in Holiday Events*

19b Political Requirements

ID: CP-06- Compliance with Domestic Development Standards

Description: The game's development and production should prioritize the use of domestic resources, including American-made software tools and components, to meet any internal organizational or contractual requirements.

Rationale: Meeting domestic production standards may be necessary to align with company policy, external partner expectations, or government contracts.

Fit Criterion: Ninety-five percent of tools, software, and components used should be sourced from domestic providers or comply with specified standards.

Acceptance Tests: *Domestic Compliance Test, Percentage of American-Made Components Used, Procurement Review Test, Verification of Domestic Tool and Component Use*

ID: CP-07 - Accessibility for Senior Management

Description: All core game features, development progress tracking, and reports should be accessible to senior management through a designated interface or reporting system.

Rationale: Ensuring senior management has easy access to game progress and performance data enhances transparency and satisfies internal requirements for oversight.

Fit Criterion: Ninety percent of requested game development reports should be accessible to senior management within 24 hours of request.

Acceptance Tests: *Access Test, Functionality of Management Access to Reports, Report Delivery Test, Timeliness of Data Access for Senior Management*

ID: CP-08 - Environmental Partnership Acknowledgment

Description: If partnering with environmental organizations for conservation campaigns, the game should prominently acknowledge these partners in the game's credits and promotional materials.

Rationale: Acknowledging environmental partners helps reinforce the game's commitment to conservation and satisfies partnership requirements for brand visibility and collaboration.

Fit Criterion: Partner logos and acknowledgments should appear in the credits and on promotional materials in 100% of distributed copies and marketing assets.

Acceptance Tests: *Partner Acknowledgment Test, Presence of Partner Logos in Credits, Marketing Material Review Test, Inclusion of Partner Branding in Promotional Content*

ID: CP-09 - Adherence to International Sanctions and Export Restrictions

Description: The game should comply with all relevant international sanctions and export restrictions, ensuring it is not distributed in restricted countries or regions.

Rationale: Compliance with sanctions and export laws is necessary to avoid legal issues and maintain positive standing with regulatory bodies.

Fit Criterion: The game should be restricted from download or purchase in regions under sanctions, confirmed by platform-specific regional access controls.

Acceptance Tests: *Export Compliance Test, Regional Access Restrictions, Legal Review Test, Compliance with International Trade Laws*

ID: CP-10 - Inclusion of Company Branding and Logos

Description: The game should include company branding and logos on the main menu, loading screens, and within promotional materials to align with internal marketing directives.

Rationale: Including company branding reinforces brand identity and satisfies internal marketing requirements for product association.

Fit Criterion: The company logo should appear in at least three distinct in-game locations and in 100% of promotional materials.

Acceptance Tests: *Branding Placement Test, Visibility of Company Logos in Game, Marketing Material Review Test, Brand Presence in Promotional Content*

20 Legal Requirements

20a Compliance Requirements

ID: LR-01 - Legal Rights

Description: The application must comply with relevant data protection laws, including ensuring the security and privacy of user data in alignment with the Data Protection Act and applicable international regulations (e.g., GDPR, CCPA).

Rationale: To prevent legal issues such as lawsuits, regulatory fines, and delays in launch due to non-compliance with data protection laws. Compliance with data protection regulations is essential as it reinforces player trust and aligns with company policy.

Fit Criterion: Legal counsel certifies that the game fully complies with all applicable data protection laws, with no infringement of user privacy rights.

Acceptance Tests: *Review by legal experts on adherence to data protection laws. Documentation proving data encryption and secure storage practices are implemented.*

20b Standards Requirements

ID: LR-02 - Company Standards

Description: The game must be developed in accordance with recognized game development standards, including SSADM (Structured Systems Analysis and Design Method) and internal company standards for quality, testing, and software lifecycle management.

Rationale: Adherence to standards ensures a consistent development process, meeting the industry's best practices, avoiding delays, and enhancing quality and usability.

Fit Criterion: The project undergoes regular audits and reviews to verify that all stages meet SSADM and other internal standards, with certification from the standards officer.

Acceptance Tests: *Regular documentation and review sessions with a standards compliance officer. QA testing according to SSADM methodology.*

21 Requirements Acceptance Tests

21a Requirements – Test Correspondence Summary

Requirement ID	Req1	Req2	Req3	Req4	Req5	Req6	Req7	Req8	Req9
Test 1	**X**								
Test 2		**X**							
Test 3			**X**						
Test 4			**X**						
Test 5		**X**		**X**					
Test 6									
Test 7			**X**						
Test 8	**X**								
Test 9		**X**							

Table 1 - Requirements - Acceptance Tests Correspondence

21b Acceptance Test Descriptions

Account Creation/Deletion

Description: Allow players to create, modify, and delete game profiles, supporting personalized settings and tracking progress throughout the game.

Tutorial and Onboarding

Description: Test the initial onboarding experience to ensure players understand basic controls, resource management (oxygen), and survival strategies, helping them navigate both underwater and on-land environments effectively

Conservation Mission Completion

Description: Verify that conservation missions are engaging, and that players receive appropriate rewards, such as in-game currency or tools, while gaining knowledge on ocean conservation themes

Creature Interaction and Survival Mechanics:

Description: Add a note on testing how these interactions impact the player's resource management to highlight the connection between creature encounters and survival elements, such as oxygen levels

Puzzle Mechanics and Environmental Challenges

Description: Test the puzzle mechanics to ensure they are challenging but achievable, requiring players to explore and interact with the environment in ways that enhance immersion and storytelling

Conservation Content Accuracy

Description: Conservation organizations review the educational content to ensure accuracy, providing players with real-world conservation information seamlessly integrated into the storyline

Exploration and Resource Management

Description: Test that exploration is intuitive and that resource management (e.g., oxygen levels) adds meaningful challenges, encouraging careful planning and resourceful play

Marine Biology Content Review

Description: Marine biology experts verify the accuracy of marine life portrayals and the underwater environment, ensuring educational content is accurate and enhances immersion

Atmosphere and Horror Element Effectiveness

Description: Assess the effectiveness of the game's horror elements, ensuring a balance between suspense and player engagement without overwhelming the player.

III Design

22 Design Goals

Because Depths of Fear is a horror exploration game, our goal is to create an environment where the player can feel the true horrors that humanity can bring to our current world. So, for our design, we strive to have the following elements:

- **Immersive Atmosphere:** Strive for an engaging and terrifying underwater environment through detailed graphics, dynamic lighting, and eerie soundscapes to heighten the sense of isolation and fear.
- **Tension and Pacing:** Optimize gameplay flow to balance moments of exploration and suspense, building tension without overwhelming the player.
- **Realistic Physics:** Ensure underwater movement and interactions feel natural while maintaining playability, even if it requires simplifying some physics to prioritize responsiveness.
- **Performance Optimization:** Achieve smooth gameplay on a wide range of hardware without compromising on the graphical and auditory fidelity essential to the horror experience.
- **AI Behavior:** Design intelligent and unpredictable AI for underwater threats to maximize suspense and replayability, ensuring encounters feel fresh and challenging.
- **Player Engagement:** Maximize player immersion through adaptive storytelling, compelling mysteries, and meaningful player choices that influence the narrative and environment.
- **Accessibility:** Strive for inclusivity by designing adjustable difficulty levels and controls to ensure all players can experience the terror and thrill of Depths of Fear.

23 Current System Design

Currently, we don't have a pre-existing system design, but any implementation of a new system should be attempting to follow the elements of the game listed within our design goals.

24 Proposed System Design

24a Initial System Analysis and Class Identification

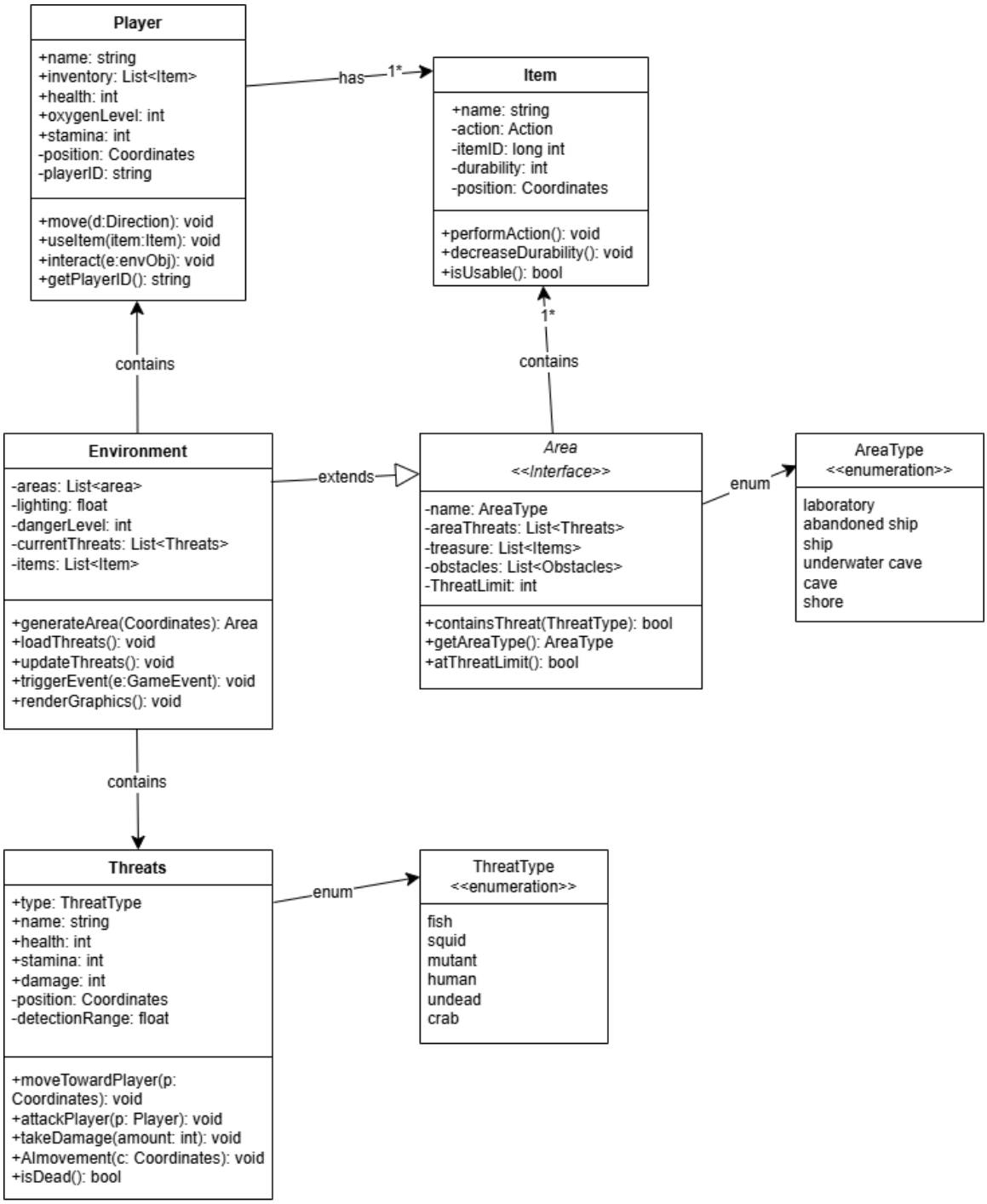


Diagram 1 - Depths of Fear Class Model

24b Dynamic Modelling of Use-Cases

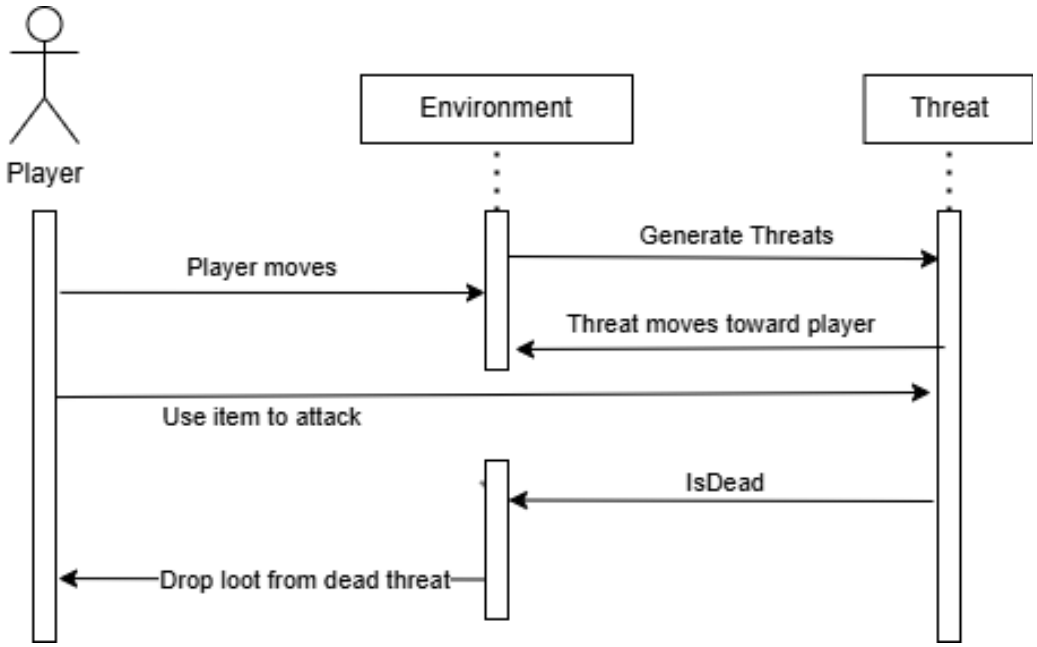


Diagram 2 - Player killing Threat Dynamic Model

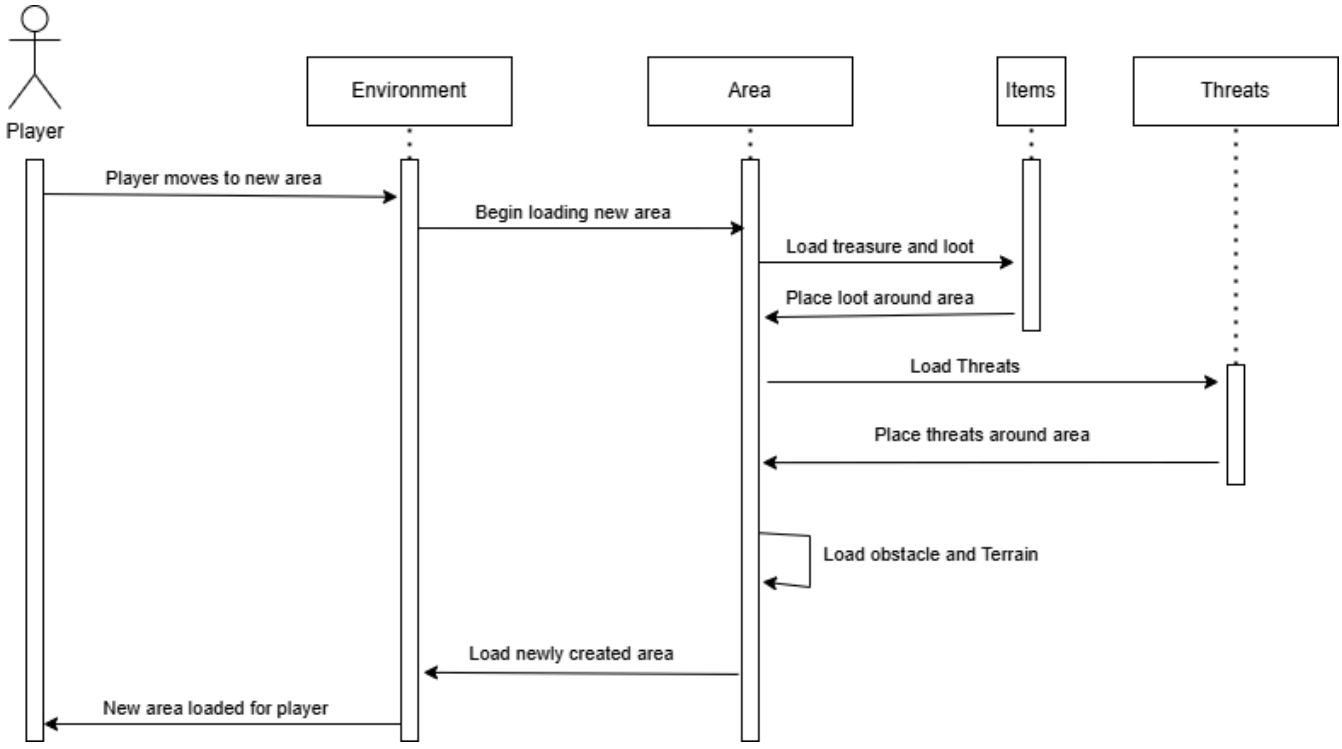


Diagram 3 - Player Discovers New Area

24c Proposed System Architecture

Our proposed system architecture will consist of a Model-View-Controller (MVC), which will have the responsibility of handling game logic, graphics/UI, and user input processing. Our MVC will have the main three responsibilities:

- ***Model:*** This represents the game logic and data, ensuring that the game's rules, player stats, environment variables, and threats are independent of user interactions or display logic. This allows core game mechanics outside of the player's control to stay separate from the player's actions, allowing for smoother gameplay and an easier way to debug later on.
- ***View:*** This represents the visuals of the game and the user interface, focusing on creating the immersive atmosphere and experience for the player. This allows for any graphical representations within the game to be dynamic, which will boost efficiency of gameplay and resources.
- ***Controller:*** This acts as the intermediary between the user's inputs and the game's logic. It is responsible for processing player actions, and updating the ***Model*** and the ***View*** accordingly.

We will have three main subsystems to handle these responsibilities: Gameplay Subsystem, Rendering Subsystem, and Interaction Subsystem. All in all, the MVC is meant to ensure that the different components of the game will continue to coincide with the user's actions, and update the gameplay as the user progresses throughout the game.

24d Initial Subsystem Decomposition

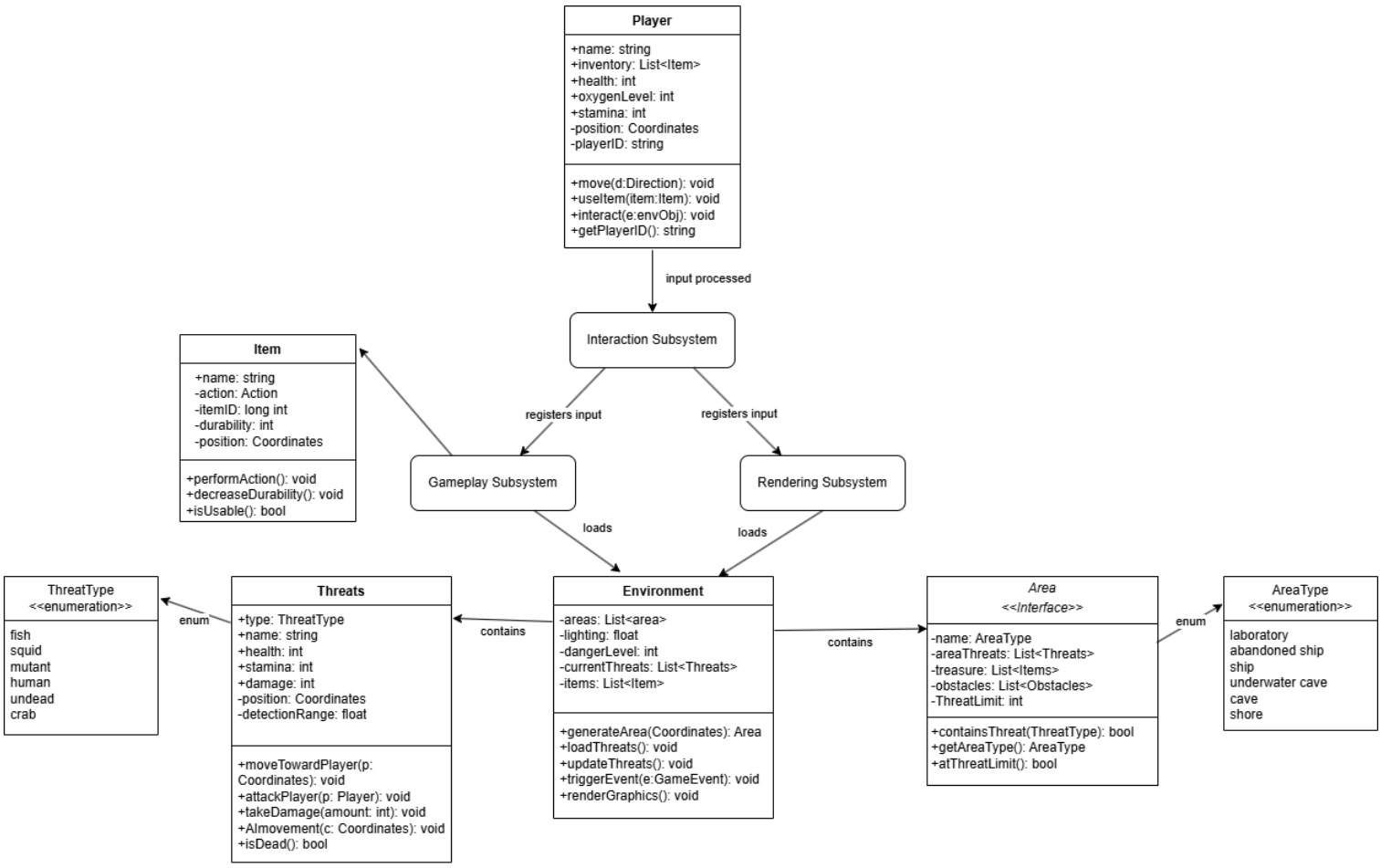


Diagram 4 - Depths of Fear MVC Interaction with Classes

25 Additional Design Considerations

25a Hardware / Software Mapping

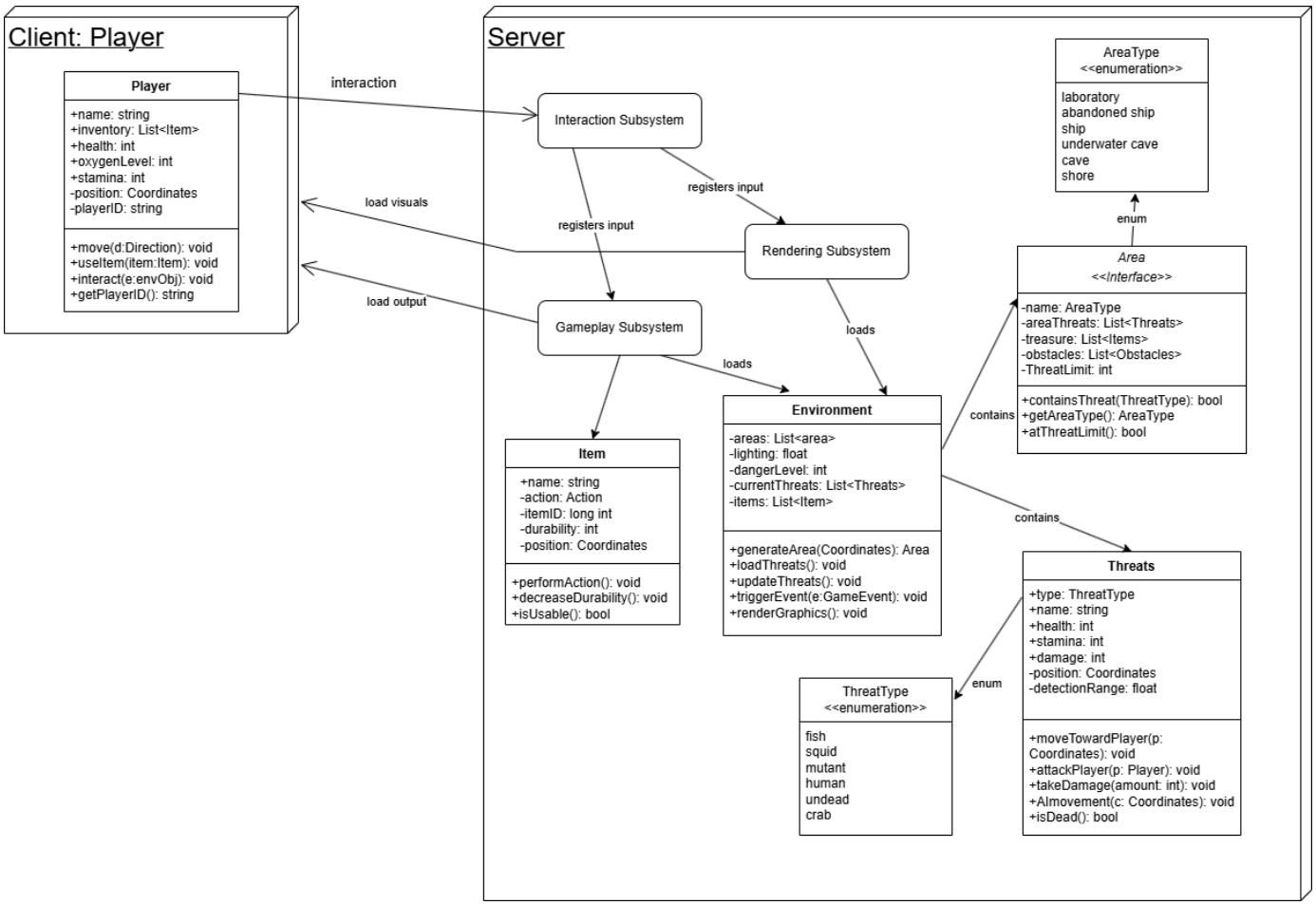


Diagram 5 - Depths of Fear Client-Server Connection

Because we expect to implement multiplayer for Depths of Fear, we expect to have the majority of the game's logic and mechanics be handled on the server-side, while the server reads the client's inputs and changes the gameplay according to our MVC. The server will then return to the client any game changes. This will allow for all players to be interacting with the same world at the same time, and thus achieves the goal of having the players be able to interact with each other.

25b Persistent Data Management

We will be having a Save Manager Subsystem to be handling the user's progress within the game. The Save Manager will utilize a database to save all current game states, and determine what will be saved and what should be deleted so that only the necessary components of the game will need to be loaded when the user returns to the game.

25c Access Control and Security

We will have security checks on all subsystems to protect and secure gameplay elements from outside resources. This will especially be in our MVC, which already handles most aspects of gameplay to ensure that no anomalies are present while playing. We will also need to ensure that client-server communications are secure, ensuring that no private information is leaked when playing with other players.

25d Global Software Control

Our MVC generally handles all runtime states, and ensures that all aspects of the game are handled properly according to the game logic implemented.

25e Boundary Conditions

Startups and shutdowns will be handled by our Save Manager Subsystem and Menu Manager to ensure that all progress is loaded and saved successfully, as well as all necessary objects are loaded at runtime.

25f User Interface

Diagram 6 - Depths of Fear Initial Screen

This is an example of what the game would look like at startup.

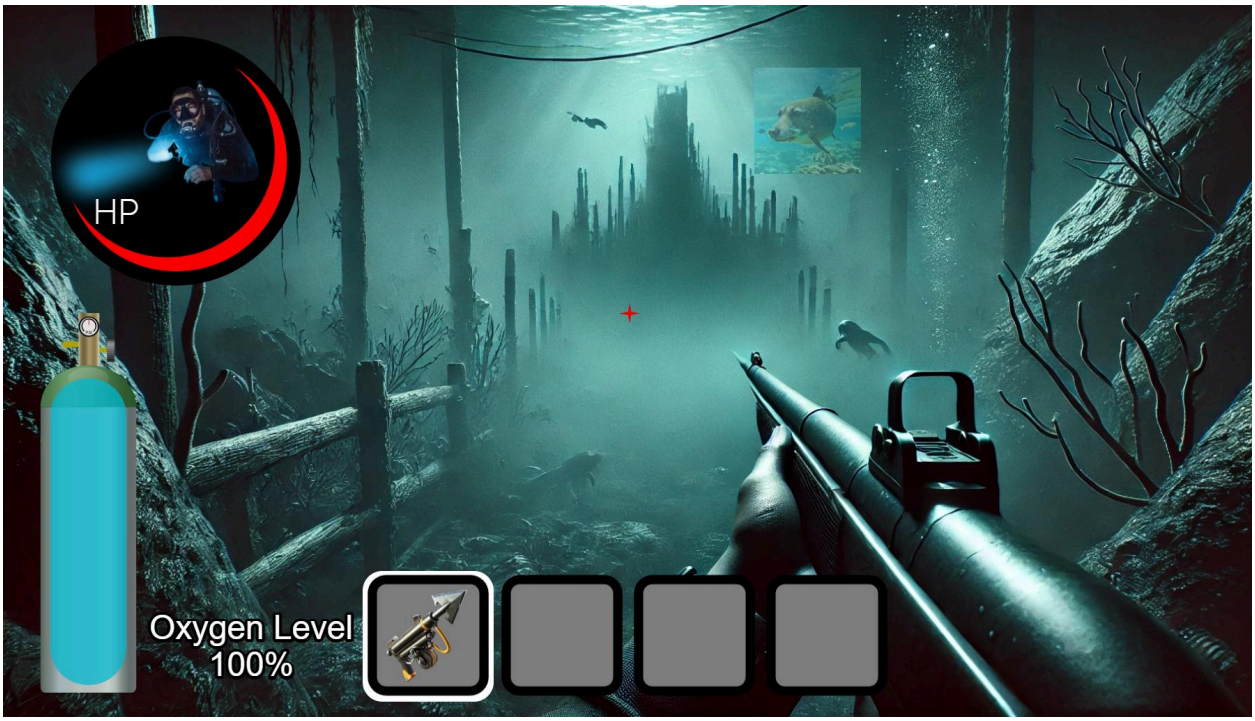


Diagram 7 - Depths of Fear HUD

This is an example of what our gameplay would look like underwater. It will display the health bar, oxygen levels, four items in the player's inventory for quick access, and a crosshair to show where the player is aiming.

25g Application of Design Patterns

Singleton: Player (when in single player), MVC, All subsystems

Factory Method: Threat creation, Area

Abstract Factory: Biomes, Tools, Equipment

Prototype: Item, Threats

Builder: Threat customization, Research Facilities, Abandoned ships

25h Final System Design

Refer to Diagram 4 and 5 for our final system design. We expect our systems to revolve around the player choosing single player, where Diagram 4 best represents that gameplay, and multiplayer, where Diagram 5 best represents that gameplay.

26 Object Design

26a Packages

There will be two packages, the client package (consisting of elements of single player gameplay), and the server package (consisting of elements of multiplayer gameplay). This is so that single player and multiplayer can both be played at any time since both types of packages are different in nature.

26b Subsystem I

The Model-View-Controller (MVC) is our most important system, which will have the responsibility of handling game logic, graphics/UI, and user input processing. Our MVC will have the main three responsibilities which are split into three subsystems:

- ***Gameplay Subsystem:*** This represents the game logic and data, ensuring that the game's rules, player stats, environment variables, and threats are independent of user interactions or display logic.
- ***Rendering Subsystem:*** This represents the visuals of the game and the user interface, focusing on creating the immersive atmosphere and experience for the player.
- ***Interaction Subsystem:*** This acts as the intermediary between the user's inputs and the game's logic. It is responsible for processing player actions, and updating the ***Gameplay*** and the ***Rendering Subsystems*** accordingly.

26c Subsystem II

Save Manager Subsystem: This handles the user's progress within the game. It will utilize a database to save all current game states, and determine what will be saved and what should be deleted so that only the necessary components of the game will need to be loaded when the user returns to the game.

Audio Subsystem: This handles playing certain noises and music at any given part of the game.

Authentication Manager: This handles authenticating user information for multiplayer, as well as ensuring game files are not corrupted or have been tampered with.

26d etc.

Networking Module: This handles multiplayer server-client connections and communications. It enables security and speed to ensure gameplay is smooth and secure.

Graphic Engine: This works closely with the Rendering Subsystem, ensuring that all visuals are loaded correctly based on the user's settings.

Database Manager: This works closely with the Save Manager Subsystem, ensuring that servers are able to handle individual player worlds and progress when being played online.

State Manager: This works closely with the Gameplay Subsystem, assisting in checking game states and game progress to ensure that correct core gameplay mechanics are presented at the correct times.

Menu Manager: This handles the start-up screen, pause menu, inventory menu, store menu, etc..

IV. Project Issues

27 Open Issues

1. VR Mode Implementation

Uncertainty: Determining whether integrating an optional VR mode is feasible within our current budget and timeline.

Impact: VR support could enhance immersion but may require additional resources, specialized development, and testing efforts.

Considerations:

- Assessing the technical challenges of implementing VR, such as motion sickness prevention and performance optimization.
- Evaluating the potential return on investment given the current market share of VR users.
- Deciding if VR development should proceed concurrently with the main game or as a post-launch update.

2. Partnerships with Environmental Organizations

Uncertainty: Securing collaborations with organizations like Greenpeace or the World Wildlife Fund.

Impact: Partnerships could enhance the game's credibility and educational value but may involve strict content guidelines or financial commitments.

Considerations:

- Understanding the requirements and expectations of potential partners.
- Evaluating how partnership constraints might influence game content or development processes.
- Planning for contingencies if partnerships cannot be established.

3. Realism vs. Performance in Graphics

Uncertainty: Balancing high-quality, realistic underwater graphics with performance across various hardware configurations.

Impact: Striving for graphical excellence may limit the game's accessibility to players with lower-end systems.

Considerations:

- Optimizing graphics to maintain performance without compromising the horror atmosphere.
- Implementing scalable graphic settings to accommodate a wider range of hardware.
- Allocating sufficient resources for performance testing and optimization.

4. Artificial Intelligence Complexity

Uncertainty: Developing advanced AI behaviors for creatures sensitive to sound and movement.

Impact: Complex AI is crucial for gameplay but may extend development time and require specialized expertise.

Considerations:

- Determining the scope of AI behaviors necessary for an engaging experience.
- Assessing whether existing AI frameworks can be adapted or if custom solutions are needed.
- Prioritizing AI features based on their impact on gameplay and resource availability.

5. Educational Content Accuracy

Uncertainty: Ensuring that in-game educational content is accurate, up-to-date, and engaging.

Impact: Inaccurate information could damage the game's credibility and educational goals.

Considerations:

- Collaborating with marine biologists and environmental experts to validate content.
- Establishing processes for regular content reviews, especially if updates are planned.
- Balancing educational content with gameplay to maintain player engagement.

6. Market Trends and Player Expectations

Uncertainty: Adapting to rapidly changing market trends in the gaming industry.

Impact: Shifts in player preferences could affect the game's reception and commercial success.

Considerations:

- Conducting market research to understand current and emerging player interests.
- Monitoring competitor releases and industry developments that could influence player expectations.
- Remaining flexible in design to incorporate features that align with market demands.

7. Game Engine Limitations and Dependencies

Uncertainty: Potential limitations or changes in the third-party game engine used for development.

Impact: Engine constraints could affect game features, performance, or delay the development schedule.

Considerations:

- Evaluating the chosen engine's capabilities in handling complex underwater environments and VR.
- Staying informed about upcoming engine updates that might impact the project.
- Considering backup plans or alternative engines if significant issues arise.

8. Hardware Platform Changes

Uncertainty: Emerging new consoles or VR platforms during the development cycle.

Impact: New platforms could offer opportunities but might necessitate additional development work.

Considerations:

- Monitoring hardware trends and announcements from major manufacturers.
- Designing the game architecture to be adaptable to new platforms.
- Weighing the benefits of early adoption against the risks of spreading resources too thin.

9. Team Expertise and Resource Allocation

Uncertainty: Ensuring the development team has the necessary expertise, particularly in specialized areas like VR and AI.

Impact: Skills gaps could lead to delays, increased costs, or compromised quality.

Considerations:

- Assessing the current team's skills and identifying areas needing reinforcement.
- Planning for training, hiring specialists, or outsourcing certain development aspects.
- Balancing the project scope with available human resources.

10. Funding and Budget Constraints

Uncertainty: Securing sufficient funding to cover all aspects of development, especially if unforeseen expenses arise.

Impact: Budget shortfalls could force feature cuts, lower quality, or halt development.

Considerations:

- Creating a detailed budget with contingencies for unexpected costs.
- Exploring additional funding sources like grants, investors, or crowdfunding.
- Prioritizing features and components essential to the core experience.

28 Off-the-Shelf Solutions

In developing Depths of Fear, it's essential to consider existing products and components that could be integrated into the game or used instead of developing certain parts from scratch. This approach can save time, reduce costs, and leverage proven technologies.

28a Ready-Made Products

1. Game Engines

Unreal Engine 5

Description: A cutting-edge game engine known for its photorealistic graphics and advanced features like Nanite and Lumen.

Considerations:

- Ideal for creating realistic underwater environments with dynamic lighting.
- Supports complex physics simulations necessary for underwater mechanics.

Potential Use: Serve as the primary development platform for the game.

Unity

Description: A versatile game engine suitable for both 2D and 3D game development, with extensive VR support.

Considerations:

- Large asset store with pre-made models and scripts.
- Strong community support and documentation.

Potential Use: An alternative to Unreal Engine if team expertise favors Unity.

2. Middleware for AI and Physics

Havok Physics Engine

Description: Industry-standard physics engine used in many AAA titles.

Considerations:

- Capable of simulating complex underwater physics, such as buoyancy and fluid dynamics.

Potential Use: Integrate to enhance the realism of underwater movement and interactions.

Kynapse AI Middleware

Description: Provides advanced AI solutions for non-player characters and creatures.

Considerations:

- Can simplify the development of sophisticated AI behaviors for marine life.

Potential Use: Implement to handle creature and NPC AI.

3. VR Development Kits

Oculus SDK

Description: Software development kit for creating VR experiences on Oculus devices.

Considerations:

- Essential for developing the optional VR mode.

Potential Use: Utilize to ensure compatibility and performance on Oculus platforms.

SteamVR

Description: A VR platform that supports a variety of VR hardware.

Considerations:

- Allows the game to reach a broader VR audience.

Potential Use: Implement for wider VR compatibility.

4. Educational Content Platforms

EcoLearn

Description: An educational platform focused on ecology and environmental science.

Considerations:

- Offers content that could be integrated into the game's educational aspects.

Potential Use: Use to enhance the in-game journal with accurate environmental information.

5. Sound and Music Libraries

Boom Library – Underwater Sound Effects

Description: A collection of high-quality underwater sound effects.

Considerations:

- Can provide immersive audio for the underwater environment.

Potential Use: Incorporate into the game's sound design.

Dynamic Music Solutions

Description: Middleware for adaptive music that responds to gameplay.

Considerations:

- Enhances the horror atmosphere through dynamic soundtracks.

Potential Use: Implement to adapt music based on player actions and environments.

Motivation:

Time and Cost Efficiency: Utilizing ready-made products can significantly reduce development time and costs.

Quality and Reliability: Established products have been tested and refined, offering reliability.

Considerations:

Licensing and Costs: Evaluate the financial implications and licensing agreements.

Integration Effort: Assess the compatibility with our existing systems and potential integration challenges.

Restrictions: Be aware of any limitations or constraints imposed by using these products.

28b Reusable Components

1. Graphics and Rendering

Substance Suite by Adobe

Description: Tools for creating realistic textures and materials.

Considerations:

- Essential for achieving high-quality visuals, especially for underwater environments.

Potential Use: Create detailed textures for sea creatures, plants, and environments.

SpeedTree

Description: A tool for modeling and animating vegetation.

Considerations:

- Useful for generating realistic underwater flora like seaweed and coral.

Potential Use: Populate the underwater world with dynamic plant life.

2. AI Libraries

Behavior Tree Libraries

Description: Frameworks for implementing AI behaviors.

Considerations:

- Simplify the development of complex creature and NPC behaviors.

Potential Use: Structure AI decision-making processes.

3. Physics and Simulation

PhysX by NVIDIA

Description: A physics engine that can simulate real-time fluid dynamics.

Considerations:

- Can enhance underwater physics for more realistic movement.

Potential Use: Simulate water currents and buoyancy effects.

4. Audio Middleware

FMOD Studio

Description: A sound engine for creating adaptive audio.

Considerations:

- Allows for dynamic soundscapes that react to gameplay.

Potential Use: Implement immersive audio that enhances the horror atmosphere.

Wwise

Description: An audio middleware solution for interactive sound design.

Considerations:

- Offers advanced features for spatial audio and sound integration.

Potential Use: Enhance the game's auditory experience.

5. Networking and Multiplayer

Photon Networking

Description: A networking framework for real-time multiplayer games.

Considerations:

Useful if multiplayer or cooperative modes are planned.

Potential Use: Facilitate online interactions between players.

6. User Interface Components

NoesisGUI

Description: A UI framework for game development.

Considerations:

- Enables the creation of high-performance, interactive UIs.

Potential Use: Develop the game's menus, HUD, and in-game journal interfaces.

Motivation:

Efficiency: Reusing components accelerates development.

Proven Solutions: Libraries and toolkits have established reliability.

Focus on Core Features: Allows the team to concentrate on unique aspects of the game.

28c Products That Can Be Copied

Open-Source Projects and Assets

1. Open-Source Game Templates and Demos

Godot Engine Demos

Description: Godot Engine provides a range of open-source demo projects showcasing various game mechanics.

Legal Status: Released under the MIT License, allowing free use, modification, and distribution.

Potential Use:

- Adapt code for player movement, environmental interactions, and UI elements.
- Utilize 3D rendering techniques suitable for underwater environments.

Unity Asset Store Free Assets

Description: Offers a variety of free assets, including scripts, shaders, and models.

Legal Status: Assets are available under Unity's standard license, permitting use in both personal and commercial projects.

Potential Use:

- Incorporate environmental shaders for water effects.
- Use scripts for character controllers and camera systems.

2. Open-Source AI Frameworks

ORCA (Optimal Reciprocal Collision Avoidance)

Description: An open-source library for simulating realistic agent movements.

Legal Status: Released under the BSD License.

Potential Use:

- Implement AI behaviors for creatures navigating in the environment.
- Simulate realistic movement patterns for schools of fish or other marine life.

3. Public Domain and Creative Commons Assets

OpenGameArt.org

Description: A repository of freely available game assets, including 3D models, textures, sound effects, and music.

Legal Status: Assets are available under various licenses like CC0 (Public Domain), CC-BY, which allow copying and modification.

Potential Use:

- Source models for sea creatures, underwater plants, and environmental props.
- Incorporate ambient sounds and music to enhance the game's atmosphere.

NASA's Visible Earth

Description: A catalog of NASA images and animations of our planet.

Legal Status: Most content is in the public domain.

Potential Use:

- Use satellite imagery for creating realistic textures and backgrounds.

Educational Content and Databases

1. Marine Biology Databases

Encyclopedia of Life (EOL)

Description: A free, online collaborative encyclopedia intended to document all living species.

Legal Status: Content is available under Creative Commons licenses.

Potential Use:

- Integrate accurate information about marine species into the in-game journal.
- Use images and descriptions to enhance educational aspects.

NOAA Public Data

Description: The National Oceanic and Atmospheric Administration provides extensive public data on marine life and ecosystems.

Legal Status: Data is in the public domain.

Potential Use:

- Utilize real-world data for environmental storytelling and to inform game mechanics related to ocean currents and weather patterns.

Open-Source Middleware and Tools

1. Physics Engines

Bullet Physics Library

Description: An open-source real-time physics simulation library.

Legal Status: Released under the Zlib License.

Potential Use:

- Implement realistic underwater physics, such as buoyancy and fluid dynamics.

2. Audio Libraries

OpenAL (Open Audio Library)

Description: A cross-platform 3D audio API.

Legal Status: Open-source and freely available.

Potential Use:

- Handle spatial audio to create immersive underwater soundscapes.

3. AI and Pathfinding Libraries

PolyNav (for Unity)

Description: A 2D pathfinding solution suitable for certain aspects of AI navigation.

Legal Status: Available under permissive licenses or free versions.

Potential Use:

- Adapt for AI movement in complex underwater environments.

Legal Considerations

Licensing Compliance:

- Ensure all assets and code are used in accordance with their respective licenses.
- Some licenses may require attribution or restrict commercial use.

Quality Assurance:

- Verify the quality and compatibility of open-source assets with our game's requirements.

- Modify and optimize assets as necessary to fit the game's aesthetic and technical needs.

Motivation:

Reuse Rather Than Reinvention:

- By leveraging existing resources, we can focus on developing unique features that set Depths of Fear apart.

Cost and Time Efficiency:

- Reduces development time and costs associated with creating assets and systems from scratch.

Community Support:

- Open-source projects often have active communities that provide support, updates, and improvements.

Considerations

Customization Needs:

- While existing products can be copied or modified, they may require significant customization to meet our game's specific requirements and style.

Potential Risks:

- Relying on external resources may introduce dependencies that could affect the project if those resources become unavailable or unsupported.

Ethical Use:

- Even when legally permissible, we should ensure that the use of external assets aligns with ethical standards and contributes positively to the gaming community.

29 New Problems

29a Effects on the Current Environment

1. Workflow Disruptions

Change in Development Processes:

Effect: Implementing new technologies like advanced AI systems, underwater physics, and VR support may require changes to our existing development workflows.

Considerations:

- Team members may need to learn new tools or adapt to different programming paradigms.
- Existing processes may need to be re-evaluated to accommodate the complexity of the new systems.

2. Resource Allocation

Shift in Focus from Other Projects:

Effect: Allocating significant resources to Depths of Fear could detract from other ongoing or planned projects.

Considerations:

- Potential delays or deprioritization of other products.
- Strain on team members juggling multiple responsibilities.

3. Team Skill Gaps

Need for Specialized Expertise:

Effect: Current staff may lack experience in areas critical to the project, such as underwater environment modeling, VR development, or AI for complex creature behaviors.

Considerations:

- May require hiring new personnel or investing in training programs.
- Could lead to temporary decreases in productivity during the learning curve.

4. Cultural and Ethical Challenges

Environmental Messaging:

Effect: The game's focus on environmental issues may require sensitivity training to ensure accurate and respectful representation.

Considerations:

- Team members may need to become more knowledgeable about environmental science and ethics.
- Potential conflicts if staff have differing views on environmental topics.

5. Workplace Stress

Increased Pressure:

Effect: The ambitious scope of the project could lead to increased stress and potential burnout among team members.

Considerations:

- Need for effective project management and support systems.
- Importance of promoting a healthy work-life balance.

29b Effects on the Installed Systems

1. Hardware Constraints

Increased Demands on Infrastructure:

Effect: Developing high-fidelity graphics and complex simulations may strain our current hardware capabilities.

Considerations:

- Potential need for hardware upgrades, such as more powerful servers or development machines.
- Downtime during installation and configuration of new hardware.

2. Software Compatibility Issues

Integration with New Tools:

Effect: Introducing new software components, like updated game engines or middleware, may conflict with existing systems.

Considerations:

- Compatibility issues with current software versions.
- Necessity to update or patch existing systems, leading to potential disruptions.

3. Network and Storage Impact

Higher Data Requirements:

Effect: Large asset files and version control operations could tax our network bandwidth and storage solutions.

Considerations:

- May require enhancements to network infrastructure.
- Need for expanded storage capacity and robust backup solutions.

4. Security Vulnerabilities

Increased Exposure:

Effect: New systems and external integrations might introduce security risks.

Considerations:

- Implementation of additional security measures.
- Regular security assessments and monitoring.

29c Potential User Problems

1. Motion Sickness in VR

Effect: The optional VR mode could induce motion sickness or discomfort in some players.

Considerations:

- Implement comfort settings, such as adjustable field of view and motion blur reduction.
- Provide clear warnings and recommendations for VR use.

2. Accessibility Barriers

Effect: Complex controls and lack of accessibility features may exclude players with disabilities.

Considerations:

- Incorporate accessibility options like remappable controls, subtitles, and colorblind modes.
- Ensure compliance with accessibility standards (e.g., WCAG).

3. Psychological Impact

Effect: The horror elements and themes of isolation might negatively affect players sensitive to such content.

Considerations:

- Offer content warnings regarding intense horror scenes and themes.
- Provide options to adjust the intensity of horror elements.

4. Performance Issues on Lower-End Systems

Effect: High system requirements may prevent users with older hardware from playing the game effectively.

Considerations:

- Optimize the game to run on a range of hardware configurations.
- Include scalable graphics settings to accommodate lower-end systems.

5. Cultural Sensitivities

Effect: The depiction of environmental disasters and mutated creatures might be disturbing or offensive to some cultural groups.

Considerations:

- Conduct cultural sensitivity reviews.
- Adapt content as necessary for different regions.

29d Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

1. Technological Constraints

Hardware Limitations:

Effect: Our development environment may lack the advanced hardware needed for testing VR and high-fidelity graphics.

Considerations:

- Need to invest in new development kits and testing platforms.
- Potential delays while acquiring necessary equipment.

2. Internet Connectivity

Bandwidth Restrictions:

Effect: Remote team members may face slow internet speeds, hindering collaboration and access to large files.

Considerations:

- Implement efficient file-sharing protocols.
- Consider cloud-based development environments to mitigate issues.

3. Environmental Factors

Power Stability:

Effect: In regions with unreliable power supplies, development could be interrupted.

Considerations:

- Use uninterruptible power supplies (UPS) and backup generators.
- Schedule critical tasks during periods of expected stability.

4. Regulatory and Compliance Limitations

Legal Restrictions:

Effect: Laws in certain countries may limit the depiction of certain content, such as horror elements or environmental disasters.

Considerations:

- Research regional regulations to ensure compliance.
- Prepare region-specific versions of the game if necessary.

5. Team Collaboration Challenges

Remote Work Limitations:

Effect: If team members are distributed globally, time zone differences and communication barriers may impede progress.

Considerations:

- Establish clear communication protocols.
- Utilize collaboration tools to facilitate teamwork.

29e Follow-Up Problems

1. Post-Launch Support Overload

Effect: A high volume of user feedback and bug reports post-launch may overwhelm the support team.

Considerations:

- Plan for sufficient post-launch support staff.
- Implement efficient ticketing and issue-tracking systems.

2. Negative Publicity

Effect: Misinterpretation of the game's environmental themes could lead to public relations challenges.

Considerations:

- Develop a clear communication strategy highlighting the game's educational intent.
- Engage with the community to address concerns proactively.

3. Legal Challenges

Effect: Potential infringement on intellectual property if not all assets are properly licensed.

Considerations:

- Ensure thorough legal review of all assets and content.
- Maintain meticulous records of licenses and permissions.

4. Scalability Issues

Effect: If the game is more successful than anticipated, our infrastructure may not handle the increased demand.

Considerations:

- Design scalable backend systems, especially if online features are included.
- Monitor performance and be prepared to upscale services quickly.

5. Financial Constraints

Effect: Unforeseen expenses could strain the project's budget, affecting completion and quality.

Considerations:

- Establish a contingency fund within the budget.
- Regularly review expenses and adjust plans accordingly.

6. Dependency on Third-Party Services

Effect: Reliance on external services (e.g., middleware, APIs) may pose risks if those services change or become unavailable.

Considerations:

- Have backup plans or alternative solutions.
- Avoid over-reliance on any single third-party provider.

7. Market Competition

Effect: Similar games released by competitors could overshadow Depths of Fear.

Considerations:

- Continuously monitor the market for emerging trends and competitors.
- Highlight unique selling points in marketing efforts.

30 Migration to the New Product

Not Applicable

31 Risks

1. Technical Challenges

Risk: Implementation of Advanced Technologies

Description: The game requires sophisticated technologies, such as advanced AI for creature behaviors, realistic underwater physics, and optional VR support.

Probability: High

Impact: High

Potential Effects:

- Delays in development due to unforeseen technical hurdles.
- Increased costs from additional development time or need for specialized expertise.
- Compromises in game quality if technical issues cannot be fully resolved.

Mitigation Strategies:

- Allocate additional time in the schedule for research and development.
- Hire or consult with experts in AI, physics simulation, and VR.
- Develop prototypes early to identify and address technical challenges.

2. Resource Constraints

Risk: Inadequate Staffing and Expertise

Description: The team may lack sufficient personnel with the necessary skills, leading to overworked staff and skill gaps.

Probability: Medium

Impact: High

Potential Effects:

- Reduced productivity due to burnout.
- Delays in project milestones.
- Lower quality output from inexperienced team members.

Mitigation Strategies:

- Assess current team capabilities and identify skill gaps.
- Recruit additional staff or provide training to existing team members.
- Prioritize tasks to focus on critical components first.

3. Inaccurate Cost and Schedule Estimations

Risk: Underestimating Project Complexity

Description: The project's scope and complexity may lead to underestimation of time and budget required.

Probability: High

Impact: High

Potential Effects:

- Budget overruns.
- Inability to meet deadlines.
- Need to cut features or reduce quality to stay within constraints.

Mitigation Strategies:

- Conduct a detailed project planning and estimation exercise.
- Include contingency buffers in the budget and schedule.
- Regularly review and adjust estimates as the project progresses.

4. Dependency on Third-Party Tools and Assets

Risk: Reliance on External Software and Assets

Description: The project depends on third-party game engines, middleware, and assets, which may have licensing issues, compatibility problems, or become unsupported.

Probability: Medium

Impact: Medium

Potential Effects:

- Unexpected costs due to licensing fees.
- Development delays if tools are discontinued or incompatible.
- Legal issues arising from improper use of assets.

Mitigation Strategies:

- Carefully review licenses and terms of use for all third-party components.
- Maintain good relationships with vendors and stay updated on their product roadmaps.
- Have backup options or plan for in-house development if necessary.

5. Market Competition and Changing Trends

Risk: Competitive Releases and Shifting Player Preferences

Description: Similar games may be released by competitors, or player interests may shift, reducing the game's market appeal.

Probability: Medium

Impact: High

Potential Effects:

- Reduced sales and revenue.
- Difficulty in capturing market share.
- Pressure to add features or change the game design mid-development.

Mitigation Strategies:

- Conduct market research to understand trends and gaps.
- Highlight unique selling points in marketing efforts.
- Stay flexible to adjust the game design if necessary.

6. Scope Creep

Risk: Uncontrolled Expansion of Project Scope

Description: Adding new features and requirements during development without proper assessment.

Probability: High

Impact: High

Potential Effects:

- Project delays and increased costs.
- Overcomplication of the game, potentially reducing quality.
- Team frustration due to changing objectives.

Mitigation Strategies:

- Implement strict change management processes.
- Prioritize features based on their impact and feasibility.
- Regularly communicate with stakeholders to manage expectations.

7. Quality Assurance Issues

Risk: Insufficient Testing Leading to Low-Quality Product

Description: Not allocating enough time or resources for thorough testing could result in a buggy or unpolished game.

Probability: Medium

Impact: High

Potential Effects:

- Negative reviews and player dissatisfaction.
- Increased post-launch support costs.
- Damage to the company's reputation.

Mitigation Strategies:

- Establish a comprehensive QA plan early in the project.
- Allocate sufficient time in the schedule for testing phases.
- Use automated testing tools where appropriate.

8. Legal and Regulatory Risks

Risk: Non-Compliance with Laws and Regulations

Description: Potential legal issues related to content, licensing, and data protection.

Probability: Low

Impact: High

Potential Effects:

- Fines and legal penalties.
- Forced changes to the game content.
- Delays in release or market restrictions.

Mitigation Strategies:

- Consult legal experts to ensure compliance.
- Review all content for potential legal issues.
- Secure proper licenses for all third-party assets.

9. Negative Public Reception

Risk: Controversy Over Game Content

Description: Themes in the game may be perceived negatively, leading to backlash.

Probability: Low

Impact: Medium

Potential Effects:

- Harm to brand image.
- Reduced sales due to boycotts or negative publicity.

Mitigation Strategies:

- Conduct sensitivity reviews of game content.
- Engage with the community to gather feedback.
- Be prepared with a PR strategy to address concerns.

10. Financial Constraints

Risk: Insufficient Funding

Description: The project may run out of funds before completion.

Probability: Low

Impact: High

Potential Effects:

- Inability to complete the project.
- Need to secure additional funding under unfavorable terms.
- Potential project cancellation.

Mitigation Strategies:

- Create a detailed budget with contingency funds.
- Monitor expenditures closely.
- Explore alternative funding options early, such as investors or crowdfunding.

11. Team Collaboration Challenges

Risk: Communication Breakdown in the Team

Description: Poor communication can lead to misunderstandings, duplicated efforts, or overlooked tasks.

Probability: Medium

Impact: Medium

Potential Effects:

- Decreased efficiency and productivity.
- Lower team morale.
- Errors and oversights in the final product.

Mitigation Strategies:

- Establish clear communication channels and protocols.
- Use collaboration tools to facilitate teamwork.
- Hold regular meetings to ensure alignment.

12. Technology Changes During Development

Risk: Emerging Technologies or Platform Changes

Description: New technologies or updates may render current development approaches obsolete.

Probability: Low

Impact: Medium

Potential Effects:

- Need to rework parts of the game to stay current.
- Additional costs and time required to adapt.

Mitigation Strategies:

- Stay informed about industry trends.
- Design the game architecture to be flexible.
- Avoid over-commitment to unproven technologies.

32 Costs

Summary of Estimated Costs

- Total Development Cost: $10,000,000 - $20,000,000
- Development Timeframe: 24 - 30 months
- Team Size: 35 - 45 full-time staff
- Opportunity Costs: Potential delay or reprioritization of other projects due to resource allocation

1. Personnel Costs

Estimated Cost: $4,200,000 - $5,400,000

Development Team (25 Developers):

- Annual Salary per Developer: $80,000 - $120,000
- Total Cost for Developers: $4,000,000 - $6,000,000 over 2 years

Artists and Designers (10 Staff):

- Annual Salary per Artist/Designer: $70,000 - $100,000
- Total Cost for Artists/Designers: $1,400,000 - $2,000,000 over 2 years

Project Management and Support Staff (5 Staff):

- Annual Salary per Staff: $90,000 - $130,000
- Total Cost for Management/Support: $900,000 - $1,300,000 over 2 years

QA Testers (5 Staff):

- Annual Salary per Tester: $50,000 - $70,000
- Total Cost for QA Testers: $500,000 - $700,000 over 2 years

2. Technology and Tools

Estimated Cost: $600,000 - $800,000

Software Licenses:

- Game Engine (e.g., Unreal Engine, Unity): Royalty-based fees or subscription costs
- Middleware and Plugins: $150,000 - $200,000

Hardware Costs:

- Development Workstations: $2,500 per unit x 40 units = $100,000
- Testing Devices (PCs, Consoles, VR Equipment): $150,000

Server and Infrastructure Costs:

- Version Control Systems, Build Servers, Cloud Services: $100,000 - $150,000

Asset Acquisition:

- Purchase of Assets, Asset Packs, Sound Libraries: $50,000 - $100,000

3. Training and Hiring Costs

Estimated Cost: $250,000 - $350,000

Recruitment Expenses:

- Job Postings, Recruitment Agency Fees: $50,000 - $70,000

Training Programs:

- Workshops on VR Development, Advanced AI, Underwater Physics: $200,000 - $280,000

4. Marketing and Promotion

Estimated Cost: $500,000 - $700,000

Pre-Launch Marketing:

- Teaser Trailers, Demo Releases, Social Media Campaigns: $250,000 - $350,000

Post-Launch Marketing:

- Advertisements, Influencer Partnerships, Press Events: $250,000 - $350,000

5. Testing and Quality Assurance

Estimated Cost: $300,000 - $400,000 (In addition to QA staff salaries)

External Testing Services:

- Compatibility Testing, Compliance Testing: $150,000 - $200,000

User Testing Sessions:

- Beta Testing Management, User Feedback Analysis: $150,000 - $200,000

6. Legal and Compliance Costs

Estimated Cost: $150,000 - $200,000

Legal Consultation:

- Contract Negotiations, Licensing Agreements, Intellectual Property Rights: $75,000 - $100,000

Compliance and Certification:

- Age Ratings (ESRB, PEGI), Accessibility Compliance: $75,000 - $100,000

7. Overhead Costs

Estimated Cost: $400,000 - $500,000

Office Space Rent and Utilities:

- $12,000 per month x 24 months: $288,000

Operational Expenses:

- Insurance, Office Supplies, Maintenance: $112,000 - $212,000

8. Contingency Fund

Estimated Cost: $600,000

Reserved for Unforeseen Expenses:

- Technical Challenges, Schedule Overruns, Emergency Funding

Function Points Calculation

- Number of Input and Output Flows: 60
- Number of Business Events: 35
- Number of Product Use Cases: 30
- Number of Functional Requirements: 120
- Number of Nonfunctional Requirements: 60
- Number of Requirements Constraints: 25
- Total Function Points (Estimated): 250

Cost per Function Point

Industry Average for Game Development:

- Cost per Function Point: $5,000 - $10,000

Estimated Development Cost Based on Function Points:

- 250 Function Points x $5,000 - $10,000 = $1,250,000 - $2,500,000

Time Resources

Estimated Development Duration: 24 - 30 months

Working Days: Approximately 480 - 600 days

Total Man-Hours:

- For a team of 35 staff: 480 days x 8 hours/day x 35 staff = 134,400 man-hours

Potential Delays:

- Risk Factors: Technical challenges, scope changes, resource availability
- Possible Extension: 6 - 12 months, adding $1,500,000 - $2,500,000 to costs

Opportunity Costs

Reallocation of Resources:

- Effect on Other Projects: Delay or cancellation of other potential projects

Estimated Opportunity Cost: $1,000,000 - $2,000,000

- Lost Revenue: From projects not pursued
- Market Positioning: Potential loss in market share in other genres or platforms

33 Waiting Room

1. Expanded VR Support with Haptic Feedback

Description: Enhance the VR mode by integrating haptic feedback devices for a more immersive experience.

Potential Benefits:

- Deepens player immersion through tactile sensations.
- Sets the game apart in the VR market.

Considerations:

- Requires partnerships with hardware manufacturers.
- Limited user base with necessary equipment.

2. Modding Support and Tools

Description: Provide players with tools to create and share their own content, such as new missions, creatures, or environments.

Potential Benefits:

- Fosters a community around the game.
- Extends the game's lifespan through user-generated content.

Considerations:

- Additional resources needed to develop and support modding tools.
- Potential for unmoderated content affecting game balance or experience.

3. Additional Story DLCs and Expansions

Description: Release downloadable content packs that expand the game's story, introduce new characters, or explore different regions of the ocean.

Potential Benefits:

- Provides ongoing revenue streams.
- Keeps the player base engaged over time.

Considerations:

- Requires a separate development cycle.
- Must ensure new content integrates seamlessly with the base game.

4. Advanced Accessibility Features

Description: Enhance accessibility options, including support for screen readers, customizable control schemes, and options for players with sensory impairments.

Potential Benefits:

- Makes the game accessible to a wider audience.
- Demonstrates commitment to inclusivity.

Considerations:

- Additional development and testing resources needed.
- Ensuring features do not compromise gameplay for other users.

5. Integration with Real-Time Environmental Data

Description: Incorporate live data feeds of real-world ocean conditions, such as tides, currents, or weather events, to influence in-game environments.

Potential Benefits:

- Offers a unique and ever-changing gameplay experience.
- Raises awareness of real-time environmental issues.

Considerations:

- Reliance on external data sources and APIs.
- Need to handle scenarios where data is unavailable or inconsistent.

6. Haptic Suit Compatibility

Description: Support for full-body haptic suits to simulate physical sensations like water currents or creature interactions.

Potential Benefits:

- Provides an unparalleled level of immersion.
- Appeals to enthusiasts of cutting-edge gaming technology.

Considerations:

- Extremely niche market with limited accessibility.
- High development costs for a small user base.

34 Ideas for Solutions

Programming Languages and Game Engine

1. Game Engine: Unreal Engine 5

Justification:

- High-Fidelity Graphics: Unreal Engine 5 offers cutting-edge rendering capabilities suitable for creating realistic underwater environments.
- Nanite Virtualized Geometry: Allows for highly detailed models without significant performance loss, essential for complex underwater flora and fauna.
- Lumen Global Illumination: Provides dynamic lighting that can enhance the horror atmosphere with realistic shadows and reflections.

- Blueprint Visual Scripting: Enables rapid prototyping and development without extensive coding, beneficial for designers and artists.
- VR Support: Built-in support for virtual reality simplifies the implementation of the optional VR mode.

Programming Language:

- C++: The primary language for Unreal Engine, offering performance and control for complex game mechanics and systems.

2. Alternative Game Engine: Unity

Justification:

- Versatility: Unity is known for its flexibility and a large community, with extensive documentation and tutorials.
- C# Language: Easier learning curve compared to C++, which might be advantageous if the team is more familiar with C#.
- Asset Store: Offers a vast array of assets and plugins that can accelerate development.
- VR and Cross-Platform Support: Strong capabilities for VR and deployment across multiple platforms, including consoles and mobile devices.

Development Tools and IDEs

1. Integrated Development Environments (IDEs)

Visual Studio (for C++ with Unreal Engine):

- Features: Robust debugging, IntelliSense code completion, and performance profiling tools.
- Integration: Seamless integration with Unreal Engine for efficient development workflows.

Rider for Unreal Engine:

- Description: An IDE by JetBrains tailored for Unreal Engine development.
- Benefits: Advanced code analysis, refactoring tools, and a user-friendly interface.

` **Visual Studio Code (for C# with Unity):**

- Features: Lightweight, customizable with extensions, and integrated debugging support.

- Benefits: Ideal for developers who prefer a simpler interface or are working on multiple platforms.

2. Version Control Systems

Git with Git LFS (Large File Storage):

- Justification: Manages source code and large binary assets effectively.
- Tools: GitKraken, SourceTree, or command-line tools for repository management.

Perforce Helix Core:

- Justification: Widely used in game development for handling large files and binary assets.
- Benefits: Offers robust version control capabilities suitable for large teams.

Libraries and Middleware

1. Artificial Intelligence

Unreal Engine AI Tools:

- Behavior Trees and Blackboards: For designing complex AI behaviors for creatures and NPCs.
- Environment Query System (EQS): Helps AI agents make context-aware decisions.

Unity AI Libraries:

- NavMesh Components: For pathfinding and navigation.
- Third-Party Assets: Assets like Behavior Designer can provide advanced AI functionalities.

2. Physics Simulation

PhysX by NVIDIA (Integrated in Unreal Engine and Unity):

- Use Case: Simulating realistic underwater physics, including buoyancy, water currents, and pressure effects.

Project Chrono:

- Description: An open-source multi-physics simulation engine.
- Use Case: Could be integrated for advanced physics requirements beyond standard engine capabilities.

3. Audio Systems

FMOD Studio or Wwise:

- Description: Professional audio middleware for interactive and adaptive sound design.
- Benefits: Facilitates the creation of immersive soundscapes, crucial for horror and underwater environments.

Asset Creation Tools

1. 3D Modeling and Animation

Autodesk Maya or 3ds Max:

- Use Case: Industry-standard tools for creating high-quality 3D models and animations.

Blender:

- Use Case: Open-source alternative for modeling, rigging, and animating, reducing licensing costs.

ZBrush:

- Use Case: Digital sculpting tool ideal for creating detailed creature models and textures.

2. Texturing and Materials

Substance Painter and Substance Designer:

- Use Case: Creating realistic textures and materials with PBR (Physically Based Rendering) workflows.

Quixel Megascans:

- Use Case: Access to a vast library of high-quality scanned assets and surfaces for environmental details.

Testing Strategies and Tools

1. Automated Testing

Unreal Engine Automation System:

- Features: Allows for automated functional testing, performance testing, and regression testing.

Unity Test Framework:

- Features: Provides unit testing and integration testing capabilities within the Unity environment.

2. Performance Profiling

Unreal Insights:

- Use Case: Profiling CPU, GPU, and memory usage to optimize performance, crucial for VR and complex scenes.

Unity Profiler:

- Use Case: Identifying bottlenecks and optimizing resource usage in Unity projects.

3. Continuous Integration and Continuous Deployment (CI/CD)

Jenkins or TeamCity:

- Use Case: Automating builds, tests, and deployments to streamline the development process.

GitLab CI/CD:

- Use Case: Integrated with version control for seamless continuous integration workflows.

Project Management and Collaboration

1. Agile Methodology

Scrum Framework:

- Justification: Promotes iterative development, adaptability, and continuous feedback.

Kanban Boards:

- Tools: Jira, Trello, or Azure DevOps for visualizing workflows and managing tasks.

2. Communication Tools

Slack or Microsoft Teams:

- Use Case: Facilitating real-time communication, file sharing, and collaboration among team members.

Confluence or Notion:

- Use Case: Documenting project knowledge, design decisions, and maintaining a centralized knowledge base.

Potential Challenges and Mitigation Strategies

1. Technical Complexity

Challenge: Advanced features like realistic underwater physics and AI may increase development time.

Mitigation:

- Prototype Early: Develop prototypes to test complex systems before full-scale implementation.
- Use Middleware: Leverage existing solutions to reduce development effort.

2. Performance Optimization for VR

Challenge: Maintaining high frame rates necessary for VR to prevent motion sickness.

Mitigation:

- Optimize Assets: Use performance-friendly models and textures.
- Efficient Coding Practices: Optimize code to reduce CPU and GPU load.

3. Team Skill Development

Challenge: Team may need to acquire new skills for unfamiliar technologies.

Mitigation:

- Training Programs: Invest in workshops and courses.
- Hiring Specialists: Bring in experienced developers for critical areas.

35 Project Retrospective

What Worked Well:

Weekly Group Meetings

Description: We met as a team every week to discuss progress, assign tasks, and set priorities.

Outcome:

- Enhanced Communication: Regular meetings ensured everyone was on the same page, minimizing misunderstandings.
- Effective Collaboration: Facilitated teamwork and allowed for real-time problem-solving.
- Accountability: Team members were aware of their responsibilities and deadlines.

Effective Use of Communication Tools (Discord)

Description: We utilized Discord for daily communication, sharing updates, and quick consultations.

Outcome:

- Timely Assistance: Immediate support was available when someone faced challenges.
- Flexibility: Enabled communication regardless of location or time constraints.
- Continuous Engagement: Maintained team cohesion outside of scheduled meetings.

Group Review and Feedback Sessions

Description: We regularly reviewed each other's work, providing constructive feedback and suggestions.

Outcome:

- Improved Quality: Collaborative reviews led to more polished and error-free deliverables.
- Knowledge Sharing: Team members learned from each other's expertise and perspectives.
- Consistency: Ensured that all parts of the project aligned with overall objectives and standards.

What Didn’t Work Well:

Time Management Challenges

Description: Due to busy personal and academic schedules, some tasks were postponed until close to deadlines.

Outcome:

- Increased Stress: Last-minute work sessions were stressful for team members.
- Risk to Quality: Rushed tasks may not have received the attention needed for optimal quality.
- Limited Review Time: Less opportunity for thorough proofreading and revisions.

Improvement Idea:

Implement Structured Time Management Techniques

- Suggestion: Adopt methods like the Pomodoro Technique or set personal deadlines ahead of the actual due dates.
- Expected Outcome: Better individual time management, leading to reduced last-minute work and stress.

IV Glossary

AI (Artificial Intelligence): The simulation of human intelligence by computers, enabling machines to perform tasks that typically require human thought, such as learning and problem-solving.

Asset: Any resource used in game development, like images, sounds, 3D models, animations, or scripts.

Combat Mechanics: The rules and systems that govern fighting and battles within a game.

Marine Biologist: A scientist who studies life in the oceans and other saltwater environments.

Creature (Game Term): Any living being in a game, ranging from animals to fantastical monsters, that the player can interact with.

Scrum: A framework used in project management, especially in software development, focusing on teamwork, regular progress, and adaptability.

Harpoon: A long spear-like tool used for hunting or fishing, often featured in games for underwater exploration or combat.

Obstacle (Game Term): Anything in a game that challenges the player or blocks their progress, like puzzles, traps, or enemies.

Genetic Modification: The process of changing the genes of an organism to alter its characteristics, which can occur naturally or be done by humans.

Environmental Consequences: The effects that actions (usually human) have on the natural world, which can be either positive or negative.

Survival Elements: Game mechanics that require players to manage basic needs like health, hunger, thirst, or oxygen to stay alive.

Medicinal Herbs: Plants used for their healing properties, which in games can restore health or provide other benefits.

Environmental Issues: Problems in the environment, such as pollution, deforestation, or climate change, that can harm ecosystems and living organisms.

Non-Player Character (NPC): Characters in a game that are not controlled by the player but by the game's artificial intelligence, often providing information or quests.

Journal (Game Term): An in-game book or log where players can collect information, notes, or records of their journey and discoveries.

Sound Design: The art of creating audio elements for a game, including music, sound effects, and ambient sounds, to enhance the player's experience.

Isolation: The state of being alone or separated from others; in gaming, it can enhance feelings of suspense or fear.

Coral Reef Destruction: Damage to coral reefs caused by factors like pollution, overfishing, and climate change, leading to the loss of important marine habitats.

Immersion: The feeling of being deeply engaged or involved in something, like a game, to the point of forgetting about the outside world.

Environmental Storytelling: A way of telling a story through the game's setting and environment, using visual details to convey background and plot without words.

Pressure (Underwater): The force exerted by the weight of water above, increasing with depth and affecting divers and equipment.

VR (Virtual Reality): A computer-generated simulation of a three-dimensional environment that can be interacted with using special equipment like headsets, giving the user a sense of being inside the game world.

Environmental Catastrophe: A major disaster resulting from significant damage to the environment, often caused by human activities.

Resource (Game Term): Items collected by players that can be used for crafting, trading, or survival, such as materials or currency.

Technology (Game Development): The tools, software, and hardware used to create and run games.

Repository Architecture: A software design where a central data storage is accessed by various components, allowing for data sharing and communication.

Stealth Mechanics: Game features that allow players to avoid detection by enemies, often by hiding, moving quietly, or using disguises.

Client-Server Architecture: A network design where multiple computers (clients) request and receive services from a central computer (server).

Equipment (Game Term): Tools or gear that a player uses within a game to help them progress, like diving suits or harpoons.

Currents (Ocean): The continuous movement of seawater in a particular direction, influenced by wind, temperature, and Earth's rotation.

Oxygen Levels (Game Mechanic): A measure of how much breathable air a player has left while underwater, requiring management to avoid suffocation.

Weather Patterns: The state of the atmosphere at a place and time, including factors like wind, temperature, and precipitation, which can affect gameplay in certain games.

Mutated Creatures: Animals or organisms that have undergone changes in their DNA, leading to new traits or appearances, often featured as enemies in games.

V References / Bibliography

National Oceanic and Atmospheric Administration (NOAA). (2023). *Marine Life and Ocean Ecosystems*. Retrieved from https://www.noaa.gov

World Wildlife Fund (WWF). (2023). *Marine Conservation Initiatives*. Retrieved from https://www.worldwildlife.org/initiatives/oceans

Unreal Engine Documentation. (2023). *Unreal Engine 5 Documentation*. Retrieved from https://docs.unrealengine.com

International Game Developers Association (IGDA). (2014). *Game Design Document Template*. Retrieved from https://www.igda.org

Open Source Initiative. (2023). *Open Source Licenses*. Retrieved from https://opensource.org/licenses

Unity Technologies. (2023). *Unity Manual and Scripting API*. Retrieved from https://docs.unity3d.com

Greenpeace. (2023). *Ocean Plastics Pollution*. Retrieved from https://www.greenpeace.org/usa/oceans/plastics

VI Index