PROJECT ABYSS: GAME DESIGN DOCUMENT & TECHNICAL BIBLE

1. Executive Summary

Title: Project Abyss (Working Title) **Genre:** Co-op Extraction Survival / Physics Sandbox **Core Loop:** Sail -> Scan -> Dive -> Extract -> Survive. **Hook:** *Sea of Thieves* sailing physics meets *Subnautica* depth and horror, wrapped in an extraction shooter loop (without the guns). **Engine:** Unreal Engine 5.7 **Target Platform:** PC (Steam) first.

2. Gameplay Mechanics

A. The "Extraction" Loop

1. **Preparation:** Players spawn on a "Safe Island." They stock the ship with fuel, oxygen tanks, and repair wood.
2. **The Voyage:** Players physically sail (manage sails/wheel) to a generated coordinate.
   - *Challenge:* Dynamic waves, storms, and navigation.
3. **The Descent:** Players deploy the **Diving Bell** from the ship.
   - *Challenge:* Pressure limits. Players cannot swim deep without the Bell as a safe zone.
4. **The Heist:** Players leave the Bell to explore a procedural "Rift" (Cave/Ruin).
   - *Challenge:* Limited O2, Bioluminescent darkness, predators.
   - *Task:* Find heavy "Artifacts" (Physical objects).
5. **The Retrieval:** Artifacts are too heavy to swim with easily. Players must hook them to the Bell's winch.
6. **The Ascent:** One player operates the winch on the ship; the other rides the Bell up, defending the loot from sharks.
7. **The Return:** Sail back to port with the loot while a storm chases you.

B. Two-Layer Physics System

- **Layer 1: The Surface (Newtonian Physics)**
  - The Ship (`AMasterShip`) is a rigid body controlled by `OceanSolver` (Gerstner Waves).
  - It rocks, tilts, and drifts based on wave height and wind.
  - Players walk on deck using "Velocity Injection" to stay attached without sliding.
- **Layer 2: The Abyss (6DOF Movement)**
  - Below sea level, gravity is disabled.

- Movement becomes 6-Degrees-of-Freedom (Swimming).
- Lighting shifts to "Thalasophobia Mode" (Pitch black, only flashlight/bioluminescence).

3. Technical Architecture (C++)

A. Core Classes

1. `FOceanSolver` **(Struct)**
   - **Responsibility:** Calculates wave height at any X/Y location on the CPU.
   - **Why:** Ensures server-side physics matches client-side shaders without expensive GPU readbacks.
   - **Math:** Sum of 3-5 Gerstner Waves.
2. `AMasterShip` **(Pawn + Interface)**
   - **Components:** `UStaticMesh` (Hull), 4x `USceneComponent` (Pontoons), `USpringArm` (Camera).
   - **Input:** Enhanced Input (Throttle, Rudder).
   - **Physics:** 4-Point Buoyancy logic in `Tick()`.
   - **Interface:** Implements `IInteractableInterface` to allow possession.
3. `AMarineCharacter` **(Character)**
   - **Movement:** `UCharacterMovementComponent` tuned for moving floors (ImpartVelocity = true).
   - **Input:** WASD (Walking), Mouse (Look), E (Interact).
   - **Interaction:** Raycasts forward. If it hits an Interface, executes it.
4. `IInteractableInterface` **(Interface)**
   - **Function:** `Interact(APawn* Instigator)`.
   - **Use Case:** Standardizes how players use Ships, Winches, Cannons, and Loot.

B. The "Safe Winch" Architecture

- **Problem:** Physics constraints (ropes) explode over long distances (500m).
- **Solution:**
  - **Visual:** `UCableComponent` connects Ship to Bell. (No collision).
  - **Logic:** The Bell is *Kinematic*. It moves up/down via a Timeline or InterpTo logic based on the Winch state.
  - **Physics:** Only enabled for the short distance between Bell and Loot.

4. Visual Direction (UE 5.7 Features)

A. Lighting Strategy

- **Surface:** Lumen Global Illumination + Volumetric Clouds.
- **The Deep: MegaLights** (UE 5.7).
  - Allows 1000+ shadow-casting lights.
  - Every glowing fish and flare casts real shadows, creating dynamic horror.
- **Absorption:** Custom material parameter collection fades colors by depth (Red disappears at 10m, Blue at 100m).

B. Biome Design

1. **The Shallows (0-50m):**
   - Bright, colorful, safe.
   - Assets: Nanite Coral, White Sand.
2. **The Kelp Forest (50-150m):**
   - Murky green, low visibility.
   - Assets: Tall Nanite Kelp strands (Move with currents).
3. **The Midnight Zone (150m+):**
   - Pitch black.
   - Assets: Basalt rocks, "Snow" particles, Bioluminescent flora.

5. Development Roadmap (Solo Dev)

Phase 1: The Foundation (Current)

- [x] **Physics Engine:** `OceanSolver + MasterShip` Buoyancy.
- [x] **Controls:** Enhanced Input for Ship (WASD).
- [x] **Character:** `MarineCharacter` setup.
- [x] **Interaction:** `Interface` system to swap between walking/sailing.

Phase 2: The Mechanic (Next)

- [ ] **The Winch:** Build the visual cable and Bell actor.
- [ ] **The Loot:** Create `BP_Artifact` with `UPhysicsHandle`.
- [ ] **Inventory:** Simple slot system for small items.

Phase 3: The World

- [ ] **Landmass:** Sculpt "Spawn Island" (Safe Zone).
- [ ] **PCG:** Create a "Scatter Graph" to populate the ocean floor with rocks/coral automatically.
- [ ] **Atmosphere:** Setup Volumetric Fog and Post-Process depth fading.

Phase 4: The Threat

- [ ] **AI:** Simple Boids (Fish) and one Hostile Shark (NavMesh Invoker).
- [ ] **Damage:** Ship Hull Health + Repair Hammer mechanic.

6. Asset Pipeline (AI Assisted)

- **Models:** Use **Rodin / Meshy** to generate base meshes for organic props (coral, rocks).
- **Textures:** Use **Quixel Megascans** (Built-in) for realistic surfaces.
- **Code:** Use **Cursor + Rider** (Mac) for generation and debugging.
- **Audio:** Use **ElevenLabs** for voice logs found in wrecks.