Table of Contents

Design principles and client requirements

During the game's development, important decisions were made regarding the safety of players who have suffered a stroke. The first key decision concerns the camera movement. The camera is not controlled by the game during gameplay, its speed remains constant and it smoothly follows the natural movement of the head. However, the player has the option, if they wish, to turn their head and view the entire environment. In this way, we reduce nausea, disorientation, and loss of balance.

The game also avoids fast and sudden movements in the player's peripheral field of view. This is achieved by limiting the appearance of fish to a controlled area of the game in front of the user. The movement of the bucket that collects the fish is also restricted, following the boundary where the fish falls. This reduces the flow of images in the peripheral visual field and helps the player focus on the game without being distracted by extra visual information. This is important for players who have suffered a stroke, as excessive visual activity in virtual reality can increase fatigue, discomfort, and confusion.

The interaction system was designed to meet the client's requirement that it be accessible to players with mobility on the one hand. The goal was to enable players with upper limb impairments to choose whether to use their dominant hand (left or right) and have access to all game functions, such as moving the basket, navigating the menus, and using the virtual keyboard to enter their name and feedback.

The interaction system was designed to meet the client's requirement that it be accessible to players with mobility on the one hand. The goal was to enable players with upper limb impairments to choose whether to use their dominant hand (left or right) and have access to all game functions, such as moving the basket, navigating the menus, and using the virtual keyboard to enter their name and feedback.

The track is designed to provide a consistent environment, such as the lake, the waterfall, the two fixed TEXT elements for score and lives in fixed, equal positions on the right and left at the top of the screen. Additionally, the restriction on the basket's movement, the restriction on the appearance of the fish, and the fixed menu frames result in the player easily orienting themselves within the virtual environment and always having a clear visual reference point, which helps reduce discomfort and motion sickness, providing a sense of security, spatial stability, and understanding of the virtual environment.

The track is designed to provide a consistent environment, such as the lake, the waterfall, the two fixed TEXT elements for score and lives in fixed, equal positions on the right and left at the top of the screen. Additionally, the restriction on the basket's movement, the restriction on the appearance of the fish, and the fixed menu frames result in the player easily orienting themselves within the virtual environment and always having a clear visual reference point, which helps reduce discomfort and motion sickness, providing a sense of security, spatial stability, and understanding of the virtual environment.

All menus are easy to read and remain fixed in a specific location without major changes. All menus use the selected hand (left or right) and employ a laser pointer to target and select the desired button. This practice follows another principle of virtual reality, which states that all interactive objects must behave in the same way throughout the game. This reduces stress and increases users comfort.

Based on the client's description and requirements, a pause menu and a local feedback system have been created for text-entry tests. The pause menu was created so that the user can pause if they feel

tired or overwhelmed, continue when they want, or even start the game over from the beginning. The comments menu uses a custom keyboard that works the same way as in all other menus. There, the user can select letters to compose whatever they want to write. The selected letters appear above the keyboard.

The game's difficulty is adjusted so that it doesn't overwhelm the player from the start or cause sudden tension and anxiety. The gradual increase in difficulty gives them time to get familiar with the environment and the game mechanics. At first, they encounter simple mechanics and combinations. Gradually, the combinations, timing, and mechanics change, making the game increasingly difficult.

Considering all the points we explained above, Waterfall Fishing has been designed to meet the client's requirements and adhere to the principles of virtual reality design. It prioritizes comfort, safety, and accessibility, and offers the best experience for people with limited use of one hand.

Executive Summary

Waterfall Fishing is a stand-alone Meta Quest 3 VR prototype. It was designed and developed in Unreal Engine 5.7. The game was designed as an accessible one hand playable arcade. The player chooses to control the basket with left or right hand that is placed in front of a waterfall and tries to catch falling fishes to gain points. The player has 3 lives, can miss 3 fishes before the game ends but he must avoid the Killing fish that ends the game immediately . The game responds and follows a client brief focused on accessible gaming, satisfied and improved the demands that were given.

The current build includes a full menu flow, gameplay loop, scoring, lives, dynamic difficulty, player-name entry, Hall of Fame leaderboard, local feedback saving, haptic feedback, audio cues, and Quest 3 packaging support. The design prioritizes simple interaction, low physical demand, clear visual feedback, and readability in VR.

Project Overview

Category	Design Description
High Concept	A seated-or-standing VR fishing/catching game where the player uses one hand to move a basket and catch safe fish while avoiding a dangerous instant-loss fish.
Core Fantasy	The player stands or sits on a boat in front of stylized waterfall and reacts to falling objects using controlled upper-limb movement.
Main interaction	Hold trigger with the selected active hand to move the basket left and right.
Session Length	Short arcade sessions, suitable for testing and feedback collection.
Primary Goal	Achieve the highest possible score and enter the Hall of Fame leaderboard.
Failure Conditions	Lose all lives by missing safe fish or catching KillFish, which immediately ends the game.
Accessibility Goals	Make the game playable with one mobile arm and usable in seated or standing mode.

Client Brief Alignment

The client brief asks for a VR application linked to stroke awareness and upper-limb mobility issues, with a strong focus on accessibility, seated/standing support, single-arm play, local feedback collection, and Quest 3 standalone deployment.

Client Requirement	Waterfall Fishing Implementation	Status
Playable by target audience with one mobile arm	The basket can be controlled using only the selected active hand. Left and right hand options are available through the menu.	Done
Playable seated and standing	The player pawn estimates HMD height and adjusts the gameplay anchor / basket movement range for seated or standing use.	Done
Required interactions above waist height	The basket and UI panels are positioned in front of the player, avoiding low reach requirements.	Done
Interactions not blocked by chair	The basket interaction area is positioned in front of the body and can be controlled by controller movement rather than requiring broad full-body movement.	Done
Pause menu activated from left controller menu button	Pause menu support is implemented with menu toggle input; current build also supports additional right-side input for accessibility/testing.	Done
Feedback gathered locally on device	Feedback entries are saved using a local Save Game object and can include text input through the custom VR keyboard.	Done
Text input fallback	The prototype uses text input as the practical feedback method, avoiding reliance on microphone permissions.	Done
Playable on Meta Quest 3 standalone	The project is packaged for Android ASTC and installed on Quest 3 through ADB / Unknown Sources workflow.	Done / Tested
Performance, functional, and input guidance	The design uses simple mechanics, limited VFX, lightweight Niagara, clear UI states, and Quest-friendly lighting strategy.	Done

Design Pillars

Design Pillar	In the game
Accessible First	Every major interaction should be possible with one hand, limited motion, and seated posture.
Immediate Readability	Safe fish, high-value fish, and KillFish must be visually distinct in VR.
Short, playable sessions	Players should quickly understand the objective and retry after game over.
Comfortable VR Interaction	No forced locomotion, no artificial movement, and simple world-space UI interaction.
Lightweight Quest Performance	Avoid heavy lighting, excessive particles, and expensive post-process effects.
Testing Feedback	The pause menu includes a local text feedback system for user testing sessions.

Player Experience and User Flow

Phase	Player Action	System Response
Main Menu	Player views options and enters name.	Main Menu shows Start, How to Play, Accessibility, Feedback, Hall of Fame, Quit, and player-name input.
Accessibility Setup	Player choose active hand	Selected hand controls the basket and menu laser/haptic feedback.
Start Game	Player presses start button	Menus hide, gameplay un pauses, spawner activates, lives/score UI initialize.
Gameplay	Player holds trigger and moves selected hand left/right.	Basket follows clamped range; safe fish add score; missed safe fish remove lives.
Difficulty Progression	Player gain score.	Spawner unlocks more fish, increases fall speed, reduces spawn interval, and introduces KillFish.
Pause	Player opens pause menu.	Gameplay pauses and menu laser activates for UI interaction.
Game Over	Lives reach zero or KillFish is collected.	Game Over menu appears; score is saved to Hall of Fame using current player name.
Restart / Main Menu	Player restarts or returns to menu.	Restart skips Main Menu and starts instantly; Main Menu reload resets gameplay state.

Core game play

Core loop

1. A fish or dangerous object spawns above the play area.
2. The object moves toward a drop point and then falls toward the basket zone.
3. The player holds the trigger to move the basket horizontally using the active controller.
4. If the basket catches a safe fish, the player gains score.
5. If a safe fish is missed, the player loses one life.
6. If the player catches KillFish, the game ends immediately.
7. The game becomes faster and more dangerous as score increases.

Player Objective

The player has two objectives. The immediate and the long term. The immediate is to catch as many fish as possible and avoid the KillFish. The long term is to achieve the best score and be on the list Hall of Fame

Win / Lose Conditions

Condition	Result
Catch safe fish	Score increases by the fish point value.
Miss Fish	Lives Decrease by 1.
Lives reach 0	Game Over.
Catch KillFish	Immediate Game Over.
Restart from Game Over	Level reloads and gameplay begins immediately without showing Main Menu.

Game Mechanics

Basket Movement

The basket is the player's primary interaction object. It is spawned relative to a gameplay anchor and moves horizontally while the player holds the trigger. Movement is clamped to prevent the basket from leaving the intended play area.

Variable / Concept	Purpose
Active Hand	Determines whether the left or right controller drives basket movement and haptics.
bBasketControlHeld	True while the trigger is held during active gameplay.
BasketMoveRange	Horizontal movement range adjusted for seated or standing mode.
Gameplay Anchor	Reference point for basket spawning and movement calibration.

Falling Object System

Falling objects are spawned by BP_ObjectSpawnerBase. Each object moves toward a drop point on the X axis, rotates into falling orientation, and then falls down toward the basket area. The falling object base supports active/inactive state, collision, collect behavior, miss behavior, and kill fish logic.

Object Type	Behaviour	Gameplay Role
Fish 1	Basic safe fish with low or standard score value.	Early game learning.
Fish 2	Unlocked at higher score threshold.	Adds variety and score progression.
Fish 3	Unlocked at higher score threshold.	Add challenge and variety
KillFish	Instant game over if collected visually distinguished using emissive material and Niagara ribbon trail.	High-risk object that forces attention and avoidance.

Score & Lives

System	Design
Starting Lives	3 lives at the beginning of gameplay.
Safe fish catch	Adds score based on the fish child blueprint value.
Safe fish miss	Removes 1 life
KillFish collection	Triggers game over
Final score	Captured by Game Mode and sent to Game Over / Hall of Fame systems.

Difficult Scaling

Difficulty increases based on score. The spawner selects different fish classes, adds fall-speed bonus, adjusts spawn interval, and introduces KillFish chance. This creates a gradual learning curve: the early game is readable and comfortable, while later score ranges require faster reactions.

Score Range	Difficulty change
0-19	Only the basic fish appears slow spawn interval.
20+	Fish 2 Spawns & spawn interval begins to decrease.
30+	Fish 3 Spawns
40+	KillFish becomes possible and danger management begins.
80+	Fall-speed bonus and spawn pacing become noticeably more demanding.
150+	Higher-pressure gameplay; shorter spawn interval and stronger speed bonus.
250-350+	High score challenge phase with fast spawn pacing and higher reaction demand.

Accessibility Design

Accessibility is the main goal of the game. The game is designed around low complexity, minimal locomotion, one-hand operation, readable UI, and seated/standing support.

Accessibility Feature	Design Implementation
One hand play	Only one active controller is required for basket movement. The player can select left or right hand.
Seated and standing mode	The player pawn estimates HMD height and adjusts basket range / gameplay anchor.
Above waist interaction	Basket and UI are placed in front of the player at comfortable reach height.
No locomotion	The player does not need to walk or teleport.
Menu laser interaction	World-space UI can be targeted with a visible menu laser.
Text feedback option	Feedback can be entered using a custom VR keyboard without microphone reliance.
Clear danger cues	KillFish uses visual differentiation , emissive color, ribbon trail, audio, and haptic feedback.
Haptic feedback	Short controller vibrations reinforce menu clicks, fish collection, damage, and game over events.

Systems & Blueprint Architecture

Blueprint / Asset	Use
BP_PlayerPawn	VR pawn, input mapping, active hand control, basket spawning, widget components, menu state management, haptics, accessibility calibration.
BP_Basket	Collects falling objects through collision and acts as the physical gameplay target.
BP_ObjectSpawnerBase	Spawns falling objects, select fish class, manages spawn interval and difficulty scaling.
BP_FallingObjectBase	Base behavior for fish movement, falling, collection, miss handling, and kill fish logic.
BP_WaterfallGameMode	Stores score, lives, final score, game over logic, and delegates for UI updates.
BP_WaterfallGameInstance	Stores persistent runtime values such as player name and restart/menu flow flags; handles Hall of Fame save/load helpers.
BP_FeedbackSave	Local Save Game object for user feedback entries.
BP_HallOfFameSave	Local Save Game object for leaderboard entries.
WBP_MainMenu	Main menu UI, player-name entry, How to Play, Accessibility, Hall of Fame, Quit, and keyboard panels.
WBP_PauseMenu	Runtime pause UI with resume, restart, main menu, accessibility, and feedback.
WBP_GameOver	Game Over UI with final score, restart, and quit options.
WBP_HallOfFameRow	Reusable row widget for leaderboard entries.

User Interface & Menus

UI Screen	Purpose	Key Buttons / Elements
Main Menu	Entry point for the game.	Start Game, How To Play, Accessibility, Hall of Fame, Feedback, Quit, Player Name.
How To Play	Explains core controls and danger rules.	Back
Accessibility	Allows hand selection and communicates seated/standing support.	Select Left Hand, Select Right Hand, back.
Player Name Input	Allows name entry for Hall of Fame.	Keyboard letters, backspace, space, submit, back.
Pause Menu	Allows control during gameplay without leaving VR.	Resume, Restart, Main Menu, Accessibility, Feedback, Quit.
Feedback Panel	Records local user feedback.	Text input / keyboard, submit, back.
Game Over Menu	Shows end state and allows restart.	Restart, Quit / Main Menu as implemented.
Hall of Fame	Shows top scores in descending order.	Rows with player name and score, back.

Feedback & Local Storage

The project includes local feedback storage for testing sessions. The feedback system supports text entry, stores entries locally, and records information such as selected hand and seated mode state where available.

Feedback Data	Purpose
Feedback text	Captures the player/tester comment.
Submitted date / session text	Records session marker
Score at submit	Optional context for when feedback was given.
Active hand at submit	Helps identify left/right hand usability patterns.
Was seated mode	Help evaluate seated/standing accessibility.

Hall of Fame / High Score System

The Hall of Fame system records the final score after game over using the player name currently stored in the GameInstance. Entries are saved locally using a SaveGame object. The leaderboard sorts entries from highest to lowest score and keeps the top 10.

System	Description
Player enters name	The main menu keyboard updates CurrentPlayerName in BP_WaterfallGameInstance.
Game Over	GameMode sends the final score through the game over delegate.
Save score	GameInstance loads or creates BP_HallOfFameSave and adds the new score entry.
Sort and trim	Entries are sorted by score descending and limited to top 10.
Display	WBP_MainMenu creates one row widget per entry and shows the leaderboard.

Audio & Haptic Feedback

Feedback Type	Purpose	Example Use
Waterfall ambient sound	Creates environment presence.	Looping background ambience.
Water splash	Confirms fish/basket interaction.	Fish collection or falling into water.
Cling / collect sound	Score feedback	Caught fish
Glass shatter / damage sound	Negative feedback.	Life lost.
Game Over sound	Communicates failure state.	Lives reach zero or KillFish caught.
Menu click sound	Improves UI response feeling	Buttons in main/pause/game over menus.
Haptic effect	Physical confirmation	Collect, damage, game over, menu click.

Visual style & Art

The visual style is stylized and readable rather than realistic. The environment uses a waterfall and lake scene, while gameplay objects use clear colors and silhouettes to support fast decision-making in VR.

Visual Element	Direction
Environment	Stylized waterfall / lake scene with simple lighting and comfortable visual density.
Fish	Distinct colors / materials for different point values.
KillFish	Danger identity using different mesh/material, emissive color, ribbon Niagara trail, and stronger audio/haptic cue.
UI	Large world-space panels, high contrast, readable button size, laser targeting.
Lighting	Dynamic or no-precomputed lighting strategy recommended for stable Quest brightness.

Level & Environment Design

Level Area	Purpose
Player position	Stable VR viewpoint avoids artificial locomotion.
Waterfall / Lake area	Thematic background and falling-object context.
Spawn area	Objects spawn above and forward of the basket, using random Y-axis variation.
Basket collection zone	Primary interaction area placed comfortably in front of the player.
World space UI zone	Main, pause, game over, and feedback widgets appear in reachable view.

Technical Implementation

Technical Area	Implementation Notes
Engine	Unreal Engine 5 with Blueprint-based VR prototype systems.
Platform	Meta Quest 3 standalone build through Android ASTC packaging.
Input	Enhanced Input mapping contexts for VR hands, grab actions, menu toggle, and UI interaction.
UI Interaction	Widget Interaction components and custom menu laser for world-space widget targeting.
Data Persistence	SaveGame objects for feedback and Hall of Fame.
Restart flow	GameInstance flag controls whether level restarts skipping the Main Menu and begins gameplay immediately.
Lighting	Movable Directional Light and Sky Light recommend Force No Precomputed Lighting to prevent baked-lighting mismatch.
VFX	Simple Niagara effects used sparingly for KillFish readability.

Performance & Quest 3 Considerations

The design avoids full locomotion, heavy physics, and excessive VFX. Gameplay actors are simple and short-lived. The current recommendation is to use dynamic/no-precomputed lighting with stable manual exposure, lightweight UI, and limited Niagara particles for Quest 3 consistency.

Risk Area	Mitigation
Lighting appears darker in packaged Quest build	Use manual exposure, movable lights, and avoid problematic baked lightmaps.
Unbuilt Lighting / reflection warnings	Use Force No Precomputed Lighting or rebuild and save lighting data if static lighting is retained.
Too many Particles	Keep KillFish trail spawn rate, lifetime, and alpha low.
UI interactions conflicts	Ensure only the active widget has collisions and hide/collapse inactive menus.
Input complexity	Keep one-hand active control and allow hand switching from menus.

Testing Plan

Test Category	Test Case	Expected Results
Main Menu	Start Game from Main Menu.	Menu hides, spawner starts, basket control works, lives show 3.
Accessibility	Switch active hand in Main Menu.	Selected controller controls basket and laser/haptic feedback.
Seated / Standing	Run while sitting and standing.	Basket is reachable and not blocked by chair position.
Gameplay	Catch fish	Score increases, collect audio/haptic feedback plays.
Gameplay	Miss fish	Lives decrease by 1, damage feedback plays.
KillFish	Catch KillFish.	Immediate Game Over, strong feedback, final score saved.
Pause	Open Pause Menu during gameplay.	Gameplay pauses, laser appears, buttons work.
Game Over	Press Restart	Level reloads and starts gameplay immediately without Main Menu.
Hall of Fame	Complete multiple games with different scores.	Top 10 scores show with the player names
Feedback	Submit feedback text	Feedback saved locally
Quest Build	Package and install on Quest 3.	Game launches from Unknown Sources and runs standalone.

Know Risks & Limitations

Risk / Limitation	Current Mitigation or Note
Speech feedback not implemented	Text feedback system is implemented as the reliable fallback.
Lighting mismatch between editor and Quest	Use manual exposure and movable/no-precomputed lighting workflow.
Baked lightmap UV warnings from imported meshes	Fix UV1 lightmap channel in Blender or use movable/no-precomputed lighting.
Hall of Fame stores local data only	Suitable for prototype; cloud/online leaderboard not planned.
No formal clinical validation	Prototype supports accessible interaction but is not a medical assessment tool.
Short tutorial is menu-based, not cinematic	How To Play Panel currently covers tutorial information.

Future Development

- Add a short in-world tutorial cinematic showing how to move the basket and avoid KillFish.
- Add optional voice feedback if microphone permission is granted, while keeping text input as fallback.
- Add adjustable difficulty presets for different user ability levels.
- Add comfort calibration screen before gameplay begins.
- Expand fish behaviors and introduce optional therapy-inspired movement challenges.
- Add analytics/testing export for feedback entries and session score data.
- Polish environment lighting and water material specifically for standalone Quest performance.

Assets & Media

Game Play

Menus

FeedBack