

The background is an abstract composition of flowing, wavy lines in various shades of blue, creating a sense of movement and depth. In the upper center, a faint, hazy silhouette of a city skyline with several tall buildings is visible against a lighter blue sky.

Manual Técnico: Juego de Combate Naval en Realidad Virtual

Contenido

- Objetivo del juego5
 - Sumérgete en la acción5
 - Demuestra tu habilidad.....5
- Requisitos técnicos.....6
- Diseño del juego.....7
 - Entorno visual impresionante7
 - Interfaz intuitiva7
- Implementación del juego7
- Arquitectura del Sistema.....8
 - Visión General:8

Componentes del Sistema:	10
Diseño del Sistema	15
Diseño de Clases/Scripts:	15
Implementación	20
Frameworks y Bibliotecas	20
Sonidos Realistas.....	20
Ayuda en Pantalla	21
Modos de Juego y Niveles.....	21

An abstract, flowing, wavy background in shades of blue and white, resembling liquid or smoke, occupies the left side of the page.

INTRODUCCIÓN

Bienvenido al manual técnico del juego de combate naval en realidad virtual. En esta sección, te introduciremos al fascinante mundo de la realidad virtual

Objetivo del juego

Sumérgete en la acción

El objetivo del juego es brindar a los jugadores la sensación de estar en medio de un combate naval. A través de la realidad virtual, podrán vivir la emoción de dirigir un barco y participar en batallas estratégicas contra otros jugadores.

Demuestra tu habilidad

El juego también busca poner a prueba las habilidades estratégicas y de liderazgo de los jugadores. Deberán planear sus movimientos.



Requisitos técnicos

1

Tecnología de realidad virtual

El juego requerirá el uso de un dispositivo de realidad virtual, como Oculus Rift para una experiencia inmersiva completa.

2

Controladores de movimiento

Para una experiencia más realista, se recomienda utilizar controladores de movimiento que permitan a los jugadores interactuar y controlar el barco de forma física.



Diseño del juego

Entorno visual impresionante

El juego contará con un entorno visualmente impresionante, recreando detalladamente el océano y los barcos de guerra. Los jugadores podrán disfrutar de paisajes realistas y efectos especiales impresionantes.

Interfaz intuitiva

Se diseñará una interfaz intuitiva y fácil de usar para garantizar que los jugadores puedan acceder a todas las funciones del juego de manera sencilla y sin problemas durante la batalla.

Implementación del juego

La implementación del juego se realizará utilizando un motor de desarrollo de juegos en realidad virtual, como Unity. Se utilizarán técnicas de programación y diseño innovadoras para asegurar una experiencia fluida y realista.

Arquitectura del Sistema

Visión General:

Módulos Principales:

1. Controlador de Torreta:
 - a. Responsable de gestionar la entrada del controlador Oculus para apuntar y disparar desde la torreta principal del destructor.
 - b. Comunica la dirección y velocidad de los proyectiles al sistema de juego.
2. Sistema de Juego:
 - a. Orquesta el flujo general del juego, incluyendo la gestión del temporizador, el objetivo del juego y la transición entre modos y niveles.
 - b. Gestiona la lógica central del juego, como el conteo de disparos acertados y la victoria/derrota del jugador.
3. Sistema de Sonido:
 - a. Controla la reproducción de efectos de sonido realistas, como el sonido de los cañones, las olas rompiendo y otros efectos ambientales.
 - b. Sincroniza el audio con eventos clave del juego.

4. Modos de Juego y Niveles:

- a. Define y controla los parámetros específicos para diferentes modos de juego (fácil, medio, difícil).
- b. Determina la aparición y comportamiento de tiempo según la dificultad seleccionada.

Conexiones y Comunicación:

- Eventos y Mensajes:
 - c. Se utiliza un sistema de eventos y mensajes para la comunicación entre los diferentes módulos.
 - d. Por ejemplo, el Controlador de Torreta puede enviar eventos de disparo al Sistema de Juego, y este puede notificar a la Personalización del Destructor sobre cambios de modelo.

Interfaz de Usuario (UI):

- La UI se actualiza dinámicamente para mostrar información relevante, como el temporizador, el número de disparos acertados y mensajes de estado del juego.

Integración de Recursos:

- Modelos 3D:
 - e. Los modelos 3D de destructores y objetos del entorno se integran en el juego para crear un entorno realista y detallado.

- Efectos Visuales:

- f. Se implementan efectos visuales, como el movimiento de las olas y el viento, para aumentar la inmersión del jugador.

Escalabilidad:

Modularidad:

- g. La arquitectura está diseñada con un enfoque modular para facilitar la expansión y la adición de nuevas características en el futuro.

Extensiones y Actualizaciones:

- h. La arquitectura permite la fácil incorporación de nuevos modos de juego, niveles, destructores y características sin afectar la estabilidad del sistema principal.

Componentes del Sistema:

1. Controlador de Torreta:

Descripción:

Gestiona la entrada del controlador Oculus para controlar la torreta del destructor.

Calcula la dirección y velocidad de los proyectiles en función de la entrada del jugador.

Responsabilidades:

Interpretación de la entrada del controlador Oculus.

Control del movimiento de la torreta.

Disparo de proyectiles.

2. Sistema de Juego:

Descripción:

Orquesta el flujo general del juego, desde el inicio hasta la conclusión.

Gestiona el temporizador del juego, el objetivo de hundir el barco enemigo y la lógica central del juego.

Responsabilidades:

Control del temporizador y manejo de eventos temporales.

Seguimiento de disparos acertados y determinación de la victoria o derrota del jugador.

Transición entre modos de juego y niveles.

3. Sistema de Sonido:

Descripción:

Controla la reproducción de efectos de sonido realistas para mejorar la inmersión.

Sincroniza el audio con eventos clave del juego.

Responsabilidades:

Integración de efectos de sonido, como el estruendo de los cañones y el sonido de las olas.

Ajuste dinámico de niveles de audio.

4. Modos de Juego y Niveles:

Descripción:

Define y controla los parámetros específicos para diferentes modos de juego (fácil, medio, difícil) y niveles.

Determina la aparición y comportamiento de los barcos enemigos según la dificultad seleccionada.

Responsabilidades:

Configuración de parámetros específicos para cada modo de juego.

Gestión de la aparición y comportamiento de los enemigos.

5. Interfaz de Usuario (UI):

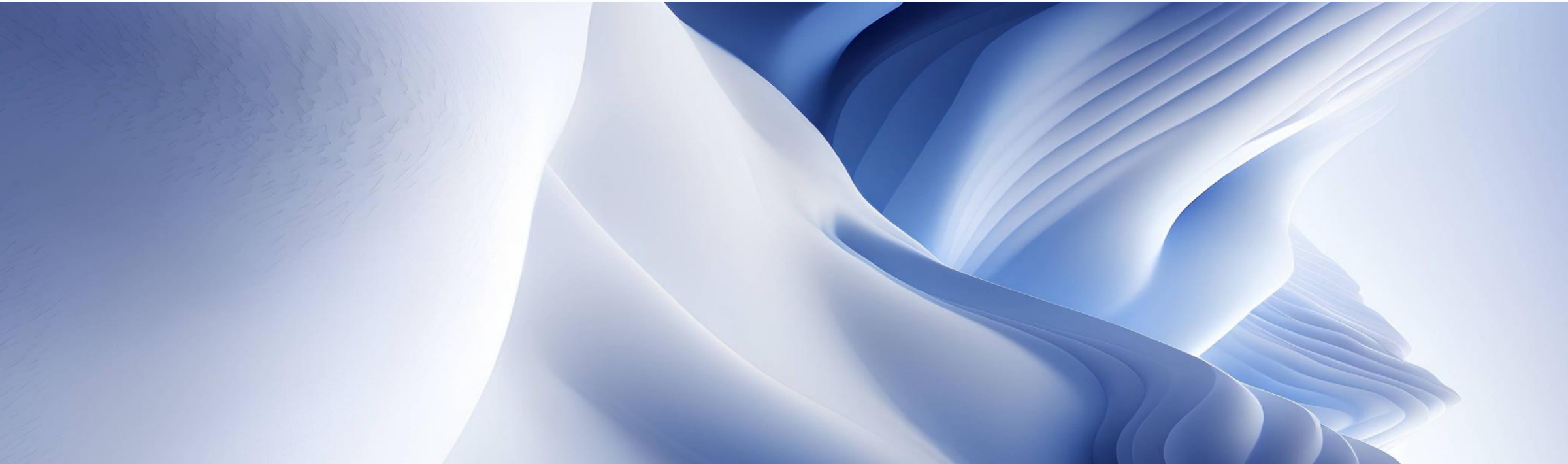
Descripción:

Proporciona información visual al jugador sobre el estado del juego, como el temporizador y el número de disparos acertados.

Responsabilidades:

Actualización dinámica de la interfaz de usuario según los eventos del juego.

Presentación de mensajes de estado y tutorial.



Diseño del Sistema

Diseño de Clases/Scripts:

1. Timer:

Descripción:

Clase encargada de gestionar el temporizador del juego.

Métodos:

StartTimer(): Inicia el temporizador.

StopTimer(): Detiene el temporizador.

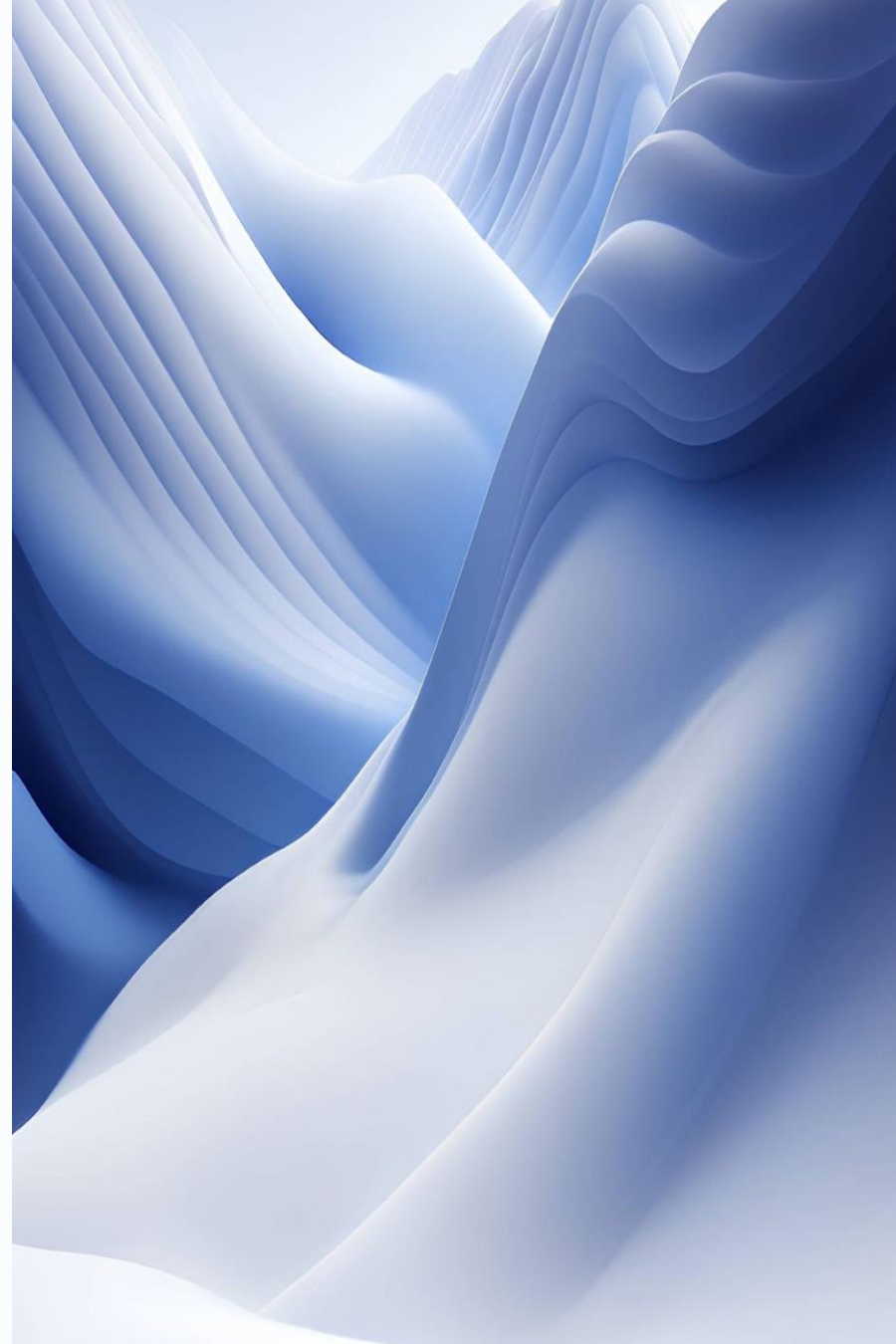
GetTimeRemaining(): Obtiene el tiempo restante.

2. SceneLoaderGO:

Descripción:

Clase para cargar escenas del juego.

Métodos:



`LoadScene(string sceneName)`: Carga una escena específica.

3. Rayolzquierdo:

Descripción:

Controla el rayo desde el controlador izquierdo de Oculus para interactuar con elementos del juego.

Métodos:

`CheckForColliders()`: Realiza un raycast y comprueba colisiones.

4. RayoDerecho:

Descripción:

Similar a Rayolzquierdo pero para el controlador derecho de Oculus.



5. MainMenu:

Descripción:

Gestiona la lógica del menú principal del juego.

Métodos:

FacilBt(): Inicia el nivel fácil

MedioBt(): Inicia el nivel medio

DifícilBt(): Inicia el nivel difícil

6. JoystickLocomotion:

Descripción:

Controla el movimiento del jugador utilizando el joystick de Oculus.

Métodos:

Update(): Captura la entrada y mueve al jugador.



7. EnemyHealth:

Descripción:

Gestiona la salud de los enemigos.

Métodos:

TakeDamage(int damage): Reduce la salud del enemigo.

8. DrawPosition:

Descripción:

Determina la posición del jugador en el espacio.

Métodos:

UpdatePlayerPosition(): Actualiza la posición del jugador.

9. DestroyOnCollisionProjectile:

Descripción:

Destruye el objeto al colisionar con otro objeto.



Métodos:

OnCollisionEnter(): Detecta colisiones y destruye el objeto.

OnTriggerExit(): Detecta colisiones y destruye el objeto.

10. CannonController:

Descripción:

Controla el cañón del jugador.

Métodos:

Update(): Maneja la entrada del jugador y dispara el cañón.

FireCannon(): Dispara el cañón.

11. Bullet:

Descripción:

Representa el proyectil disparado por el cañón.



Métodos:

Update(): Mueve el proyectil.

Implementación

Lenguajes de Programación:

Lenguaje de programación utilizados, con un enfoque en C#.

Frameworks y Bibliotecas

Sonidos Realistas

Implementación de Efectos de Sonido:

Descripción detallada de cómo se incorporan efectos de sonido realistas, como el estruendo de los cañones y el sonido de las olas.

Ayuda en Pantalla

Tutorial en Pantalla:

Implementación de un tutorial o guía en pantalla para ayudar a los jugadores a comprender las mecánicas del juego y la interfaz de control.

Modos de Juego y Niveles

Implementación de Modos de Juego:

Los modos de juego y niveles se implementan mediante un sistema de gestión que ajusta diversos parámetros según la dificultad seleccionada. La implementación se divide en tres niveles: fácil, medio y difícil.

Ajustes Específicos para Cada Dificultad:

1. Fácil:

Descripción:

En el modo fácil, se ofrece una experiencia más relajada para los jugadores principiantes.

Ajustes:

Tiempo Límite: Generoso para completar la misión.

Aparición de Enemigos: Solo 1 barco enemigo en pantalla.

2. Medio:

Descripción:

Ofrece un equilibrio moderado entre desafío y accesibilidad.

Ajustes:



Tiempo Límite: Moderado para completar la misión.

Aparición de Enemigos: 2 barcos enemigos en pantalla.

3. Difícil:

Descripción:

Para jugadores experimentados que buscan un desafío significativo.

Ajustes:

Tiempo Límite: Limitado para aumentar el desafío.

Aparición de Enemigos: 5 barcos enemigos en pantalla.

Implementación General:

Sistema de Gestión de Niveles:

Un componente central que determina la configuración específica del nivel.

Se comunica con el sistema de juego para establecer parámetros clave.



Configuración Dinámica:

Los ajustes específicos para cada dificultad se aplican dinámicamente durante la carga del nivel.

Se establece el tiempo límite y se determina la cantidad de barcos enemigos.

Aparición de Enemigos:

Un sistema de aparición de enemigos gestiona la creación y posicionamiento de barcos enemigos.

La cantidad y posición se ajustan según la dificultad seleccionada.

