Institut Puig Castellar
Projecte / Crèdit de síntesi
Document funcional

DOCUMENTO FUNCIONAL DEL PROYECTO: RAMPTRUCK

PROYECTO	RampTruck (Conducción y Precisión 2D)
AUTORES	Rayan El Maazouzi Raho / Abdelkader Hamza Mostefi
PLATAFORMA	PC
MOTOR	Godot Engine 4.6.2

Índice

1. Introducción y Contexto

1.1 Resumen del Proyecto

RampTruck es una experiencia de conducción 2D centrada en la precisión y la habilidad. El jugador controla un vehículo que debe atravesar niveles complejos repletos de rampas, plataformas elevadas y obstáculos. El objetivo técnico es alcanzar la meta manteniendo la integridad del vehículo, evitando volcar o quedar atrapado, utilizando para ello una combinación de gestión de aceleración, equilibrio aéreo y, como novedad, defensa mediante misiles.

1.2 Evolución del Concepto (Justificación)

Inicialmente, el proyecto se planteó como un *Escape Room* de terror en 3D. Sin embargo, tras un análisis de viabilidad técnica y pedagógica, decidimos pivotar hacia un juego de plataformas y físicas en 2D. Esta decisión permitió profundizar en el control de físicas reales de vehículos, optimizar el rendimiento y garantizar un acabado profesional dentro del motor Godot 4.

1.3 Objetivos

- **General:** Crear una actividad interactiva que ponga a prueba los reflejos y la gestión del estrés del jugador.

- **Específicos:**
  - Implementar un sistema de físicas de suspensión real (amortiguación).
  - Diseñar niveles con dificultad progresiva y diseño técnico de colisiones.
  - Integrar mecánicas de combate (misiles) y recolección (diamantes).

2. Análisis de Requisitos

2.1 Requisitos Funcionales (RF)

Código	Descripción
RF1	El juego debe presentar un menú principal con opciones de "Jugar" y "Salir".
RF2	El vehículo responderá a las teclas de dirección para acelerar, frenar y equilibrarse en el aire.
RF3	Mecánica de Ataque: Al pulsar la tecla de espacio el coche disparará misiles para despejar el camino.
RF4	El sistema detectará automáticamente si el vehículo se vuelca (colisión del techo con el suelo) provocando que tengas que pulsar la tecla R.
RF5	El contador debe actualizarse en tiempo real al recoger diamantes.
RF6	Al alcanzar el nodo de "Meta", pasarás a la siguiente interfaz/nivel.

2.2 Requisitos No Funcionales (RNF)

- **Puntuación:** Cuando el jugador pase al siguiente nivel debería salir la puntuación que ganó haciendo acrobacias y recogiendo diamantes.
- **Pantalla de victoria:** Cuando el jugador llegue a la meta debería de salir una pantalla mostrando la puntuación y los diamantes.
- **Botón de volver a empezar**: Cuando el jugador tenga un bug no hace falta que tenga que cerrar el juego con volver a reaparecer sería muy útil.

3. Análisis de Usuarios y Roles

En esta sección se definen las entidades que interactúan con el software y sus capacidades dentro del entorno de ejecución de RampTruck.

Rol	Descripción	Permisos y Atribuciones Principales
Usuario (Jugador)	Persona física que interactúa con el teclado.	Control de inputs (aceleración, balance, disparo de misiles), navegación por menús y uso de portales.
Motor de Juego (Sistema)	Entidad lógica que procesa las reglas del mundo.	Cálculo de físicas (suspensión), detección de señales (vuelo/vuelco), gestión de teletransporte y ejecución de la lógica de muerte.

4. Casos de Uso / Escenarios de Uso

Los casos de uso describen las interacciones clave entre el jugador y el sistema.

Código	Nombre	Actor	Descripción	Resultado Esperado
CU1	Iniciar Partida	Jugador	Selecciona "Jugar" en el menú.	Carga del nivel y liberación de la física del coche.
CU2	Control Físico	Jugador	Usa flechas/WASD para moverse.	El coche aplica torque y tracción real en las ruedas.
CU3	Ataque (Misil)	Jugador	Pulsa la tecla Espacio.	Se dispara un misil que destruye obstáculos en su trayectoria.
CU4	Teletransporte	Jugador	Entra en un área de Portal.	El coche cambia su posición al portal de destino vinculado.
CU5	Impacto y Boost	Sistema	El coche choca o acelera usando el boost.	Se activa una sacudida de cámara para inmersión.
CU6	Muerte Modular	Jugador	El coche toca un objeto con nombre "pincho/prensa".	El sistema detecta el nombre del área y ejecuta la explosión del coche.
CU7	Victoria	Jugador	El coche cruza la meta.	Se detiene el juego y aparece la siguiente pantalla.

5. Modelo de Datos y Estructura

El juego organiza la información de forma lógica para que el motor de físicas y el jugador interactúen sin errores.

5.1 Elementos Principales (Entidades)

- **Vehículo (Coche):** Gestiona los datos de posición, velocidad, daño acumulado y su estado actual (si está activo o ha explotado).

- **Mapa:** Es el contenedor que agrupa todos los obstáculos, plataformas, rampas y trampas del circuito.

- **Meta:** Es un objeto especial que marca el punto final del nivel y activa la condición de victoria.

5.2 Relaciones y Funcionamiento

- **Interacción Coche-Mapa:** El vehículo reacciona físicamente a cada elemento del escenario (rebota en rampas, se desliza en plataformas).

- **Gestión del Sistema:** El motor del juego vigila constantemente las colisiones. Si detecta un impacto crítico o una trampa, cambia el estado del coche a "explotado".

- **Contenedor de Nivel:** El mapa tiene la responsabilidad de cargar y posicionar todos los elementos para que el jugador pueda interactuar con ellos desde el inicio.

6. Planificación Técnica

6.1 Herramientas y Tecnologías

- **Motor**: Godot Engine 4.6.2
- **Lenguaje**: GDScript
- **Diseño**: Google (En imágenes para encontrar texturas), Youtube (Efectos de sonido de motor y explosión).

6.2 Reparto de Tareas

- **Rayan**: Programación del núcleo físico del coche (suspensión), interfaces y botones y parte del diseño del mapa y creación de colisiones manuales (CollisionPolygon2D).
- **Abdelkader**: Sistema de disparo de misiles y lógica de teletransporte cada uno con su programación, también parte del diseño de niveles y efectos de sacudida de cámara (*Screen Shake*).

7. Análisis de Riesgos

Riesgo	Probabilidad de fallo	Impacto	Solución Aplicada
Bucle de Teletransporte	Alta	Alto	Implementación de un retardo (Cooldown) tras el uso del portal.
Atascamiento en rampas	Media	Medio	Sustitución de colisiones cuadradas por polígonos manuales suaves.
Complejidad de Físicas	Media	Medio	Script organizado e ir probando hasta encontrar la solución.

8. Validación y Criterios de Éxito

8.1. Criterios de aceptación

- El juego se inicia sin errores
- El coche se controla correctamente
- El coche no traspasa colisiones por la alta velocidad
- Existen pantallas las cuales vas cambiando
- El rendimiento es estable

8.2. Pruebas

- Pruebas de control y colisiones
- Pruebas de jugabilidad
- Pruebas de rendimiento

9. Conclusión

RampTruck ha pasado de ser un concepto de terror a un videojuego de precisión 2D con una arquitectura técnica avanzada. Gracias al uso de lógica modular (detección por nombres) y sistemas vinculados (portales mediante variables exportadas), el proyecto es fácilmente escalable. Al añadir mecánicas como el disparo de misiles y el Screen Shake garantiza que la experiencia no solo sea técnicamente sólida, sino también altamente inmersiva y satisfactoria para el jugador.