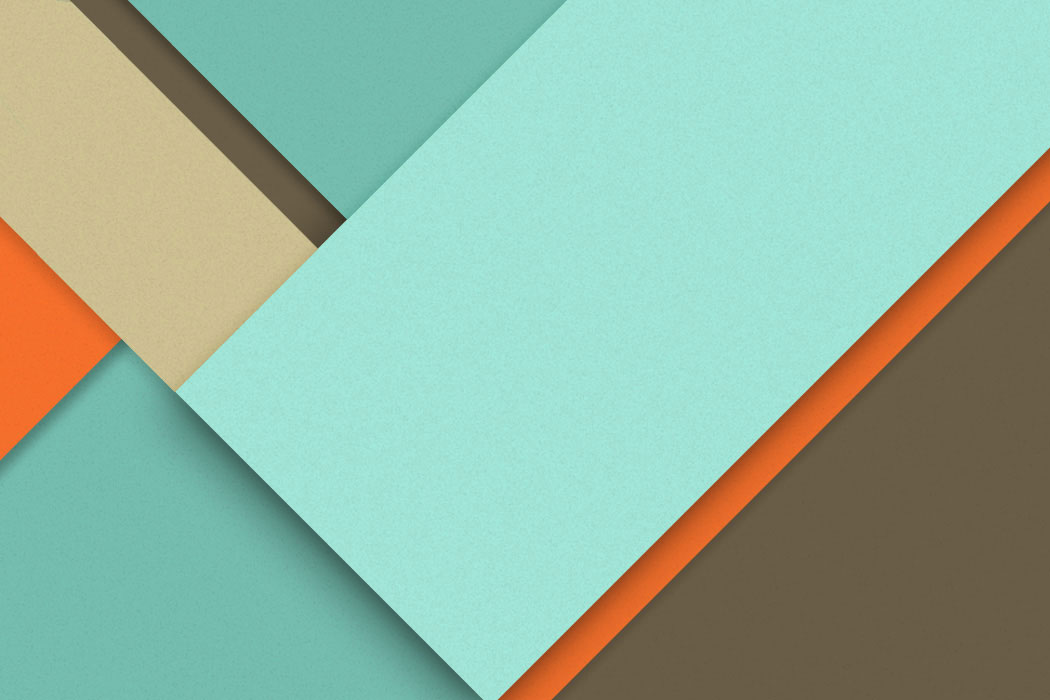
## 



**Fantasy Racing**

13/06/2022

**─**

Memoria Proyecto Final Desarrollo de Aplicaciones Multiplataforma

Antonio José de la Torre Pérez

2º DAM - 21/22

[0](#_Toc105403716)



[1. Introducción 3](#_Toc105403717)

[1.1 Motivación 3](#_Toc105403718)

[1.2 Objetivos 4](#_Toc105403719)

[1.3 Juegos de referencia 4](#_Toc105403720)

[2. Tecnologías usadas. Estado del Arte 5](#_Toc105403721)

[2.1 Motor de videojuegos 5](#_Toc105403722)

[2.2 Lenguaje de programación 5](#_Toc105403723)

[2.3 Herramienta de planificación 5](#_Toc105403724)

[3. Planificación 6](#_Toc105403725)

[3.1 Hitos 6](#_Toc105403726)

[3.2 Prototipos 7](#_Toc105403727)

[4. Presupuesto 7](#_Toc105403728)

[4.1 Recursos humanos 7](#_Toc105403729)

[4.2 Recursos materiales 7](#_Toc105403730)

[4.3 Presupuesto total 7](#_Toc105403731)

[5. Análisis Inicial 8](#_Toc105403732)

[5.1 Tipo de juego 8](#_Toc105403733)

[5.2 Resumen de la historia 8](#_Toc105403734)

[5.3 Características del personaje 8](#_Toc105403735)

[5.4 Equipo 8](#_Toc105403736)

[5.5 Objetos 8](#_Toc105403737)

[5.6 Escenarios 8](#_Toc105403738)

[5.7 Misiones 8](#_Toc105403739)

[5.8 Enemigos 8](#_Toc105403740)

[5.9 Interfaz de usuario 8](#_Toc105403741)

[5.10 Situaciones de juego 8](#_Toc105403742)

[5.11 Evolución del juego 8](#_Toc105403743)

[6. Diseño e implementación 8](#_Toc105403744)

[6.1 Sistema de objetos 8](#_Toc105403745)

[6.2 Gestión del personaje 8](#_Toc105403746)

[6.3 Sistema de habilidades 9](#_Toc105403747)

[6.4 Inventario y mochila 9](#_Toc105403748)

[6.5 Equipamiento 9](#_Toc105403749)

[6.6 Sistema de misiones 9](#_Toc105403750)

[6.7 Elementos de la interfaz 9](#_Toc105403751)

[6.8 Controlador de audio 9](#_Toc105403752)

[6.9 Guardado y carga de datos 9](#_Toc105403753)

[7. Pruebas 9](#_Toc105403754)

[8. Manual de usuario 9](#_Toc105403755)

[8.1 Menú principal 9](#_Toc105403756)

[8.2 Pantalla de juego 9](#_Toc105403757)

[8.3 Controles 9](#_Toc105403758)

[8.4 Consejos 9](#_Toc105403759)

[9. Recursos utilizados 9](#_Toc105403760)

[10. Conclusiones 9](#_Toc105403761)

[10.1 Conocimientos aplicados 9](#_Toc105403762)

[10.2 Dificultades encontradas 10](#_Toc105403763)

[10.3 Resultados obtenidos 10](#_Toc105403764)

[11. Trabajo futuro 10](#_Toc105403765)

[12. Bibliografía 10](#_Toc105403766)

# Introducción

El uso de las tecnologías de la información en la época actual está creciendo a pasos agigantados. Actualmente cualquier persona tiene acceso a un ordenador e incluso tienen un mini ordenador en su bolsillo, su smartphone.

A su vez la industria de los videojuegos no deja de evolucionar, con el paso del tiempo, el número de jugadores aumenta de manera notable. En los últimos cinco años este incremento de jugadores se ha visto disparado. El valor de la industria mundial de los videojuegos supera los 250 mil millones de euros.

Actualmente los videojuegos son claramente una de las principales formas de entretenimiento entre el público joven. Cabe destacar las competiciones profesionales de videojuegos, conocidas como e-Sports, seguidas mundialmente por millones de usuarios.

La evolución de los videojuegos ha sido rápida, muy ligada a los avances tecnológicos del momento. El lanzamiento de nuevos componentes de ordenador o consolar de nueva generación viene de la mano con lanzamientos de nuevos videojuegos con cambios en gráficos impactantes. Podemos ver notables diferencias entre los videojuegos de hace tan solo 5 o 10 años y los actuales, desde la implementación de Ray Tracing[[1]](#footnote-1) hasta herramientas de creación de personajes ultra realistas[[2]](#footnote-2)

## 1.1 Motivación

Desde pequeño los videojuegos han sido mi principal fuente de entretenimiento. El mundo de los videojuegos me interesa hasta el punto de estar al tanto de las noticias de los últimos lanzamientos y seguir los avances del desarrollo de nuevas entregas.

Mi proyecto se relaciona con este campo debido a la necesidad de comprobar si sería capaz de conseguir entretener e impresionar a la gente con algo creado por mí.

En este proyecto he querido plasmar los conocimientos adquiridos en programación durante estos dos últimos años, así como mi creatividad personal. Los retos son algo que me motiva y me mantiene activo, y considero que este proyecto se presentaba como uno de los más grandes.

## 1.2 Objetivos

El objetivo principal de este proyecto es desarrollar un videojuego desde cero. La idea es crear una primera versión totalmente jugable de un juego de carreras online. Por una parte, implementaremos las características de un juego de carreras y por otro lado le sumaremos lo más icónico de los juegos MOBA[[3]](#footnote-3), como las habilidades disponibles para los jugadores.

Este videojuego será desarrollado para PC. El motor de videojuegos nos permitirá exportar este juego para plataformas como Windows, Mac OS, WEB GL o consolas. Para esta primera versión del juego se exportará para Windows.

## 1.3 Juegos de referencia

En la industria de los videojuegos existe una amplia variedad de juegos de conducción, desde juegos arcade hasta juegos de conducción realista. En mi opinión los mejores son aquellos que juegas en familia de manera casual y crean momentos de diversión.

Entre los juegos que me han inspirado encontramos:

* Mario Kart 8 Deluxe
* Crash Tag Team Racing
* La saga Need For Speed
* Hot Wheels Unleashed
* La saga Forza



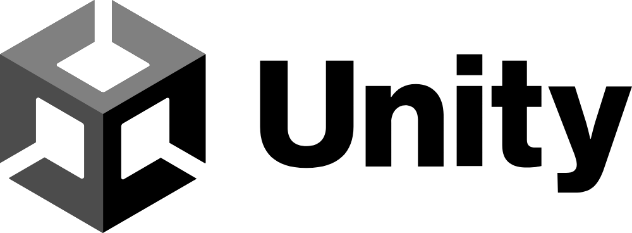
# Tecnologías usadas. Estado del Arte

## 2.1 Motor de videojuegos

Para este proyecto me decanté por utilizar el motor de videojuegos Unity. Ente las alternativas disponibles encontramos Unreal Engine, CryEngine o Amazon Lumberyard.

Entre otras razones elegí Unity por:

* La gran cantidad de información, documentación, tutoriales y componentes que la comunidad mantiene en internet.
* Su curva de aprendizaje. Para un proyecto de estas dimensiones su complejidad es la adecuada, pudiendo hacer cosas interesantes con conocimientos de principiante.
* Su lenguaje de programación.
* El hecho de haberlo usado anteriormente, tanto en clase como en antiguos proyectos personales.



## 2.2 Lenguaje de programación

En Unity se puede crear un videojuego casi totalmente desde la interfaz visual, pero sin un script de programación no podríamos hacer que el juego tuviera vida. Para crear la lógica del juego es necesario utilizar Scripts.

Para crear estos scripts utilicé C# ya que, pese a tener otras opciones, ya lo conocía con anterioridad y es el principal lenguaje soportado. La mayoría de información de scripting de Unity en internet es sobre C#.

C# es un lenguaje de programación orientado a objetos desarrollado por Microsoft como parte de su plataforma .NET. Su sintaxis básica deriva de C/C++ y utiliza un modelo de objetos similar al de Java, aunque incluye mejoras derivadas de otros lenguajes.

## 2.3 Software adicional

### Photon 2

PUN (Photon Unity Networking) es un paquete de Unity para videojuegos multijugador. Los jugadores se conectan a un servidor central y se reúnen en salas dónde los objetos pueden ser sincronizados a través de la red.

### Playfab

Playfab es un potente y reciente servicio online de Microsoft que nos permite gestionar los datos de los usuarios registrados. Utilicé esta forma de almacenamiento de datos por su sinergia con los videojuegos. Este servicio se asemeja a Firebase de Google, pero su especialidad son los videojuegos.

## 2.4 Herramienta de planificación

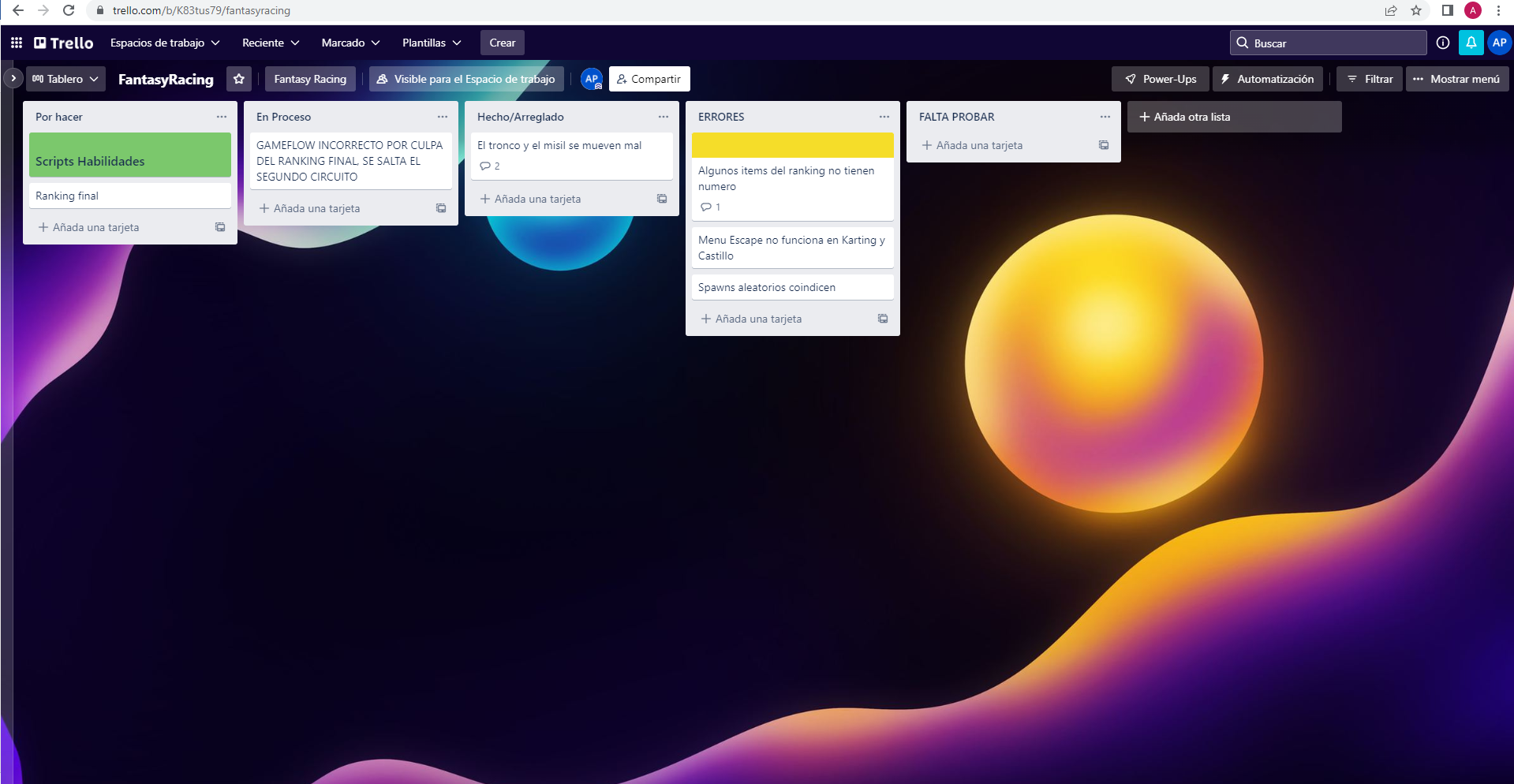
Para organizar el proyecto he utilizado la herramienta Trello.

Trello es un organizador de tareas tanto personales como grupales. Gracias a su gran variedad de opciones es posible manejar la información fácilmente. La aplicación se basa en el método Kanban, organiza la información por tableros, listas y tarjetas personalizables.

En mi caso dividí el proyecto en tareas pequeñas que asigné a tarjetas individuales. Creé 5 listas denominadas “Por hacer”, “En proceso”, “Hecho”, “Errores encontrados” y “Errores arreglados”.

A medida que iba comprobando que las tarjetas completadas o arregladas funcionaban correctamente archivaba esa tarjeta para despejar el tablero teniendo sólo las tarjetas relevantes a la vista.

Gracias a esta herramienta he podido organizar el proyecto fácilmente, sabiendo en todo momento las tareas que aún tenía que hacer, qué fue lo último que estaba haciendo o apuntando errores cuando surgían.



# Planificación

Inicialmente dediqué la primera semana a identificar los objetivos principales del proyecto, seguidamente dividí el proyecto en hitos u objetivos a cumplir semanalmente para tener las distintas partes del proyecto totalmente identificadas y avanzar de manera que las partes iniciales fueran la base del juego y las partes posteriores fuesen mejoras sobre esa base.

## 3.1 Hitos

* Semana 1: Game Design Document  
  El Documento de Diseño del Videojuego debe de ser lo primero para establecer los objetivos y límites del proyecto, en ese documento se recoge la primera idea de lo que será nuestro juego.
* Semana 2: Mapas  
  Utilizando el editor visual de Unity modelé cada uno de los mapas utilizando formas simples y modelos 3D descargados de internet.
* Semana 3: Movimiento Básico del coche  
  Este primer script de movimiento nos permitía acelerar, frenar, andar marcha atrás y girar.
* Semana 4: HUD[[4]](#footnote-4) y Puntos de control  
  Se diseñó el aspecto visual de lo que el jugador vería durante la partida, así como el sistema de puntos de control a lo largo del circuito.
* Semana 5: Online  
  Implementé la posibilidad de ver a otros jugadores en el mismo mapa.
* Semana 6: Menús y Gameflow[[5]](#footnote-5)  
  Se crearon las ventanas de inicio del juego y selección de sala y se comprobó que los jugadores acababan la carrera para pasar al siguiente mapa.
* Semana 7: Habilidades, gestión de usuarios  
  Se implementaron las habilidades de cada uno de los coches y el acceso de los usuarios utilizando email y contraseña.
* Semana 8: Guardado de datos y sonido  
  Añadí sonido a las habilidades, impactos de objetos y puntos de control. También se almacenarían los mejores tiempos de los circuitos en nuestra base de datos.
* Semana 9: Desarrollo de esta memoria

# Presupuesto

## 4.1 Recursos humanos

Para poder estimar los costes del trabajo personal he supuesto una jornada de 8 horas diarias 5 días por semana, obteniendo así los siguientes resultados:

|  |  |  |  |
| --- | --- | --- | --- |
| Tarea | Horas dedicadas | Precio por hora | Coste |
| Planificación | 32 | 6€ | 192€ |
| Diseño | 15 | 8€ | 120€ |
| Modelado 3D | 45 | 8€ | 360€ |
| Implementación | 153 | 8€ | 1224€ |
| Sonido | 12 | 6€ | 72€ |
| Pruebas | 62 | 6€ | 372€ |
| Documentación | 41 | 6€ | 246€ |
|  | **360** |  | **2586€** |

## 4.2 Recursos materiales

* Unity: El motor de videojuegos dispone de una versión para uso personal. Esta versión es gratuita si nosotros o nuestra compañía ha facturado menos de cien mil dólares durante los últimos 12 meses. Al iniciar el juego aparecerá el logo de Unity por usar esta versión gratuita.
* Modelos 3D: He utilizado los modelos 3D gratuitos que nos ofrece la página web [Kenney](https://www.kenney.nl/assets/racing-kit). Estos assets[[6]](#footnote-6) son completamente gratuitos tanto para uso personal como comercial.
* Audio: Los audios utilizados para este juego han sido descargados de la página web [Kenney](https://www.kenney.nl/assets?q=audio). Son completamente gratuitos para uso personal y comercial.
* Música: La música del juego proviene de la web del autor [Patrick de Arteaga](https://patrickdearteaga.com/es/musica-libre-derechos-gratis/). Esta música es gratuita y puede utilizarse en proyectos comerciales o personales, el único requisito que pide el autor es mencionar su nombre o la web en los créditos del proyecto.
* Photon 2: El framework de conexión en tiempo real para juegos multijugador tiene un plan básico para desarrollo y evaluación que resulta suficiente para esta primera versión del juego. Permite la conexión de hasta 20 jugadores concurrentes.
* Playfab: El servicio online de almacenamiento de datos de usuarios tiene un plan inicial gratuito disponible para juegos en desarrollo. Nos permite almacenar hasta 10 juegos diferentes con un volumen de cien mil jugadores por título.
* Ordenador: Para desarrollar este proyecto se ha utilizado un ordenador sobremesa compuesto por un procesador i5 8600K y una tarjeta gráfica GTX 1060 6GB valorado en 1200€. Se estima un coste de amortización de 150€.

## Presupuesto total

Gracias al uso de recursos con licencias de uso gratuitas el coste total del proyecto se limita a los recursos humanos y la amortización del ordenador utilizado.

|  |  |
| --- | --- |
| Recursos | Coste |
| Humanos | 2586€ |
| Materiales | 150€ |
|  | **2736€** |

# Análisis Inicial

En este análisis inicial hablaremos sobre los aspectos relacionados con la jugabilidad, escenarios, personajes, objetos, etc.

## 5.1 Tipo de juego

El juego mezcla dos géneros de juego diferentes. Por un lado, incluye las características de un juego de carreras.

## 5.2 Resumen de la historia

## 5.3 Características del personaje

## 5.4 Equipo

## 5.5 Objetos

## 5.6 Escenarios

## 5.7 Misiones

## 5.8 Enemigos

## 5.9 Interfaz de usuario

## 5.10 Situaciones de juego

## 5.11 Evolución del juego

# Diseño e implementación

## 6.1 Sistema de objetos

## 6.2 Gestión del personaje

## 6.3 Sistema de habilidades

## 6.4 Inventario y mochila

## 6.5 Equipamiento

## 6.6 Sistema de misiones

## 6.7 Elementos de la interfaz

## 6.8 Controlador de audio

## 6.9 Guardado y carga de datos

# Pruebas

# Manual de usuario

## 8.1 Menú principal

## 8.2 Pantalla de juego

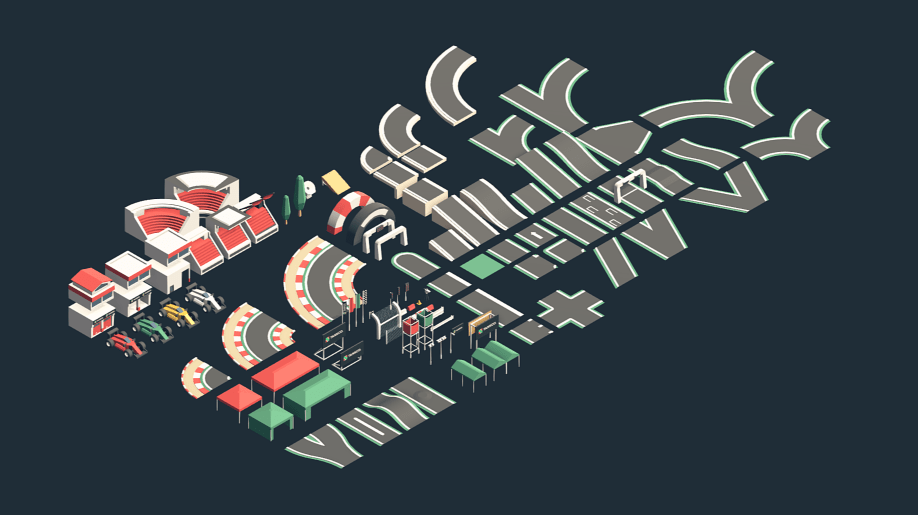
## 8.3 Controles

## 8.4 Consejos

# Recursos utilizados

## 9.1 Modelos 3D

Para construir el diseño de los circuitos, el trazado de la calzada y la decoración del escenario he utilizado modelos del paquete “[Racing Kit](https://www.kenney.nl/assets/racing-kit)” de la compañía Kenney. Este paquete es bastante completo y su diseño Low Poly[[7]](#footnote-7) hace que no sean pesados en almacenamiento ni en rendimiento del juego.



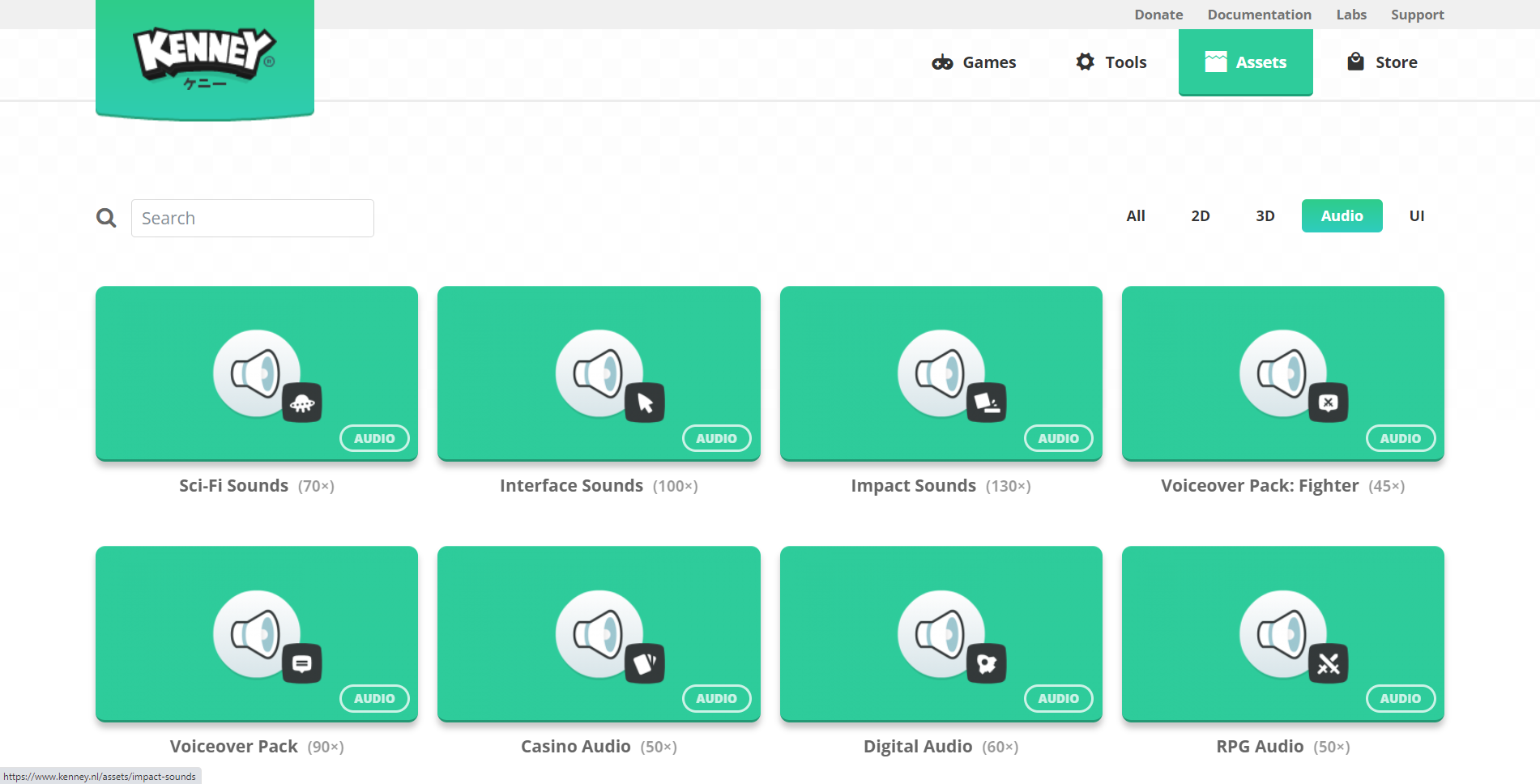
Para los modelos de los coches he utilizado el paquete “[Car Kit](https://www.kenney.nl/assets/car-kit)” también de la compañía Kenney. Al ser de la misma compañía tienen un diseño semejante que encaja a la perfección con los modelos 3D del circuito.



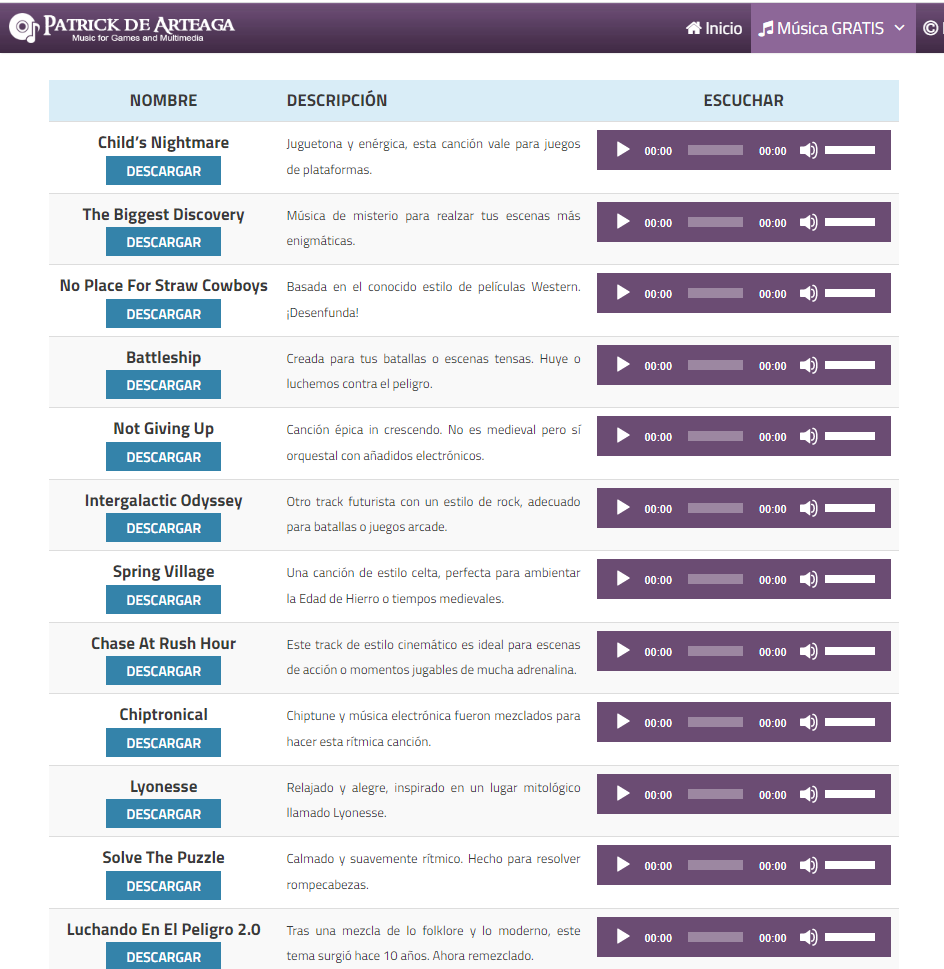
En referencia a los objetos utilizables por los jugadores como la bomba, el fuego o la basura he utilizado modelos procedentes de la página web [Sketchfab](https://sketchfab.com/).

## 9.2 Audio

Para los efectos de sonido del videojuego he utilizado los archivos de audio descargables de la compañía [Kenney](https://www.kenney.nl/assets?q=audio). He utilizado los paquetes “Sci-Fi Sounds”, “Digital Audio” e “Impact Sounds”.



La música del videojuego proviene del autor [Patrick de Arteaga](https://patrickdearteaga.com/es/musica-libre-derechos-gratis/) que en su página web nos ofrece una amplia variedad de temas. Concretamente he utilizado las canciones “Chiptonical”, “Electronic Fantasy” e “Intergalactic Odyssey”.



# Conclusiones

## 10.1 Conocimientos aplicados

## 10.2 Dificultades encontradas

## 10.3 Resultados obtenidos

# Trabajo futuro

# Bibliografía

## [1] Unity: <https://docs.unity.com/>

## [2] C#: <https://docs.microsoft.com/es-es/dotnet/csharp/>

## [3] Photon: <https://doc.photonengine.com/en-us/fusion/current/getting-started/fusion-intro>

## [4] Playfab: <https://docs.microsoft.com/en-us/gaming/playfab/>

## [5] Kenney: <https://www.kenney.nl/>

## [6] Sketchfab: <https://sketchfab.com/>

## [7] Música: <https://patrickdearteaga.com/>

1. **Ray Tracing**: Técnica de renderizado basada en el trazado de una gran cantidad de rayos desde distintas fuentes de luz, calculando y analizando los rebotes de cada rayo sobre los distintos objetos. [↑](#footnote-ref-1)
2. [**MetaHuman Creator**:](https://www.unrealengine.com/en-US/metahuman-creator) Aplicación basada en la nube que permite a cualquier persona crear seres humanos digitales fotorrealistas [↑](#footnote-ref-2)
3. **MOBA**: Género de videojuegos multijugador de arena de batalla en línea [↑](#footnote-ref-3)
4. **Head-Up Display**: Información que se muestra en pantalla durante la partida. [↑](#footnote-ref-4)
5. **Gameflow**: Organización y cambio de una pantalla a otra durante el juego. [↑](#footnote-ref-5)
6. **Assets**: Recursos multimedia que utiliza el juego [↑](#footnote-ref-6)
7. Low Poly: Malla poligonal en gráficos 3D por computadora que tiene un número relativamente pequeño de polígonos. [↑](#footnote-ref-7)