

# **Gestor de videojuegos**

---

**Autor/es:** Álvaro Sebastián Acosta Cortizas

**Fecha:** 22-02-2025

**Convocatoria:** 2025

**Curso académico:** 2024-2025

# ÍNDICE

---

<b>1 INTRODUCCIÓN.....</b>	<b>5</b>
<b>2 ORIGEN Y CONTEXTUALIZACIÓN DEL PROYECTO.....</b>	<b>5</b>
2.1 Descripción de la necesidad a la que da respuesta el producto/servicio.....	5
<b>3 OBJETIVO GENERAL DEL PROYECTO.....</b>	<b>5</b>
<b>4 OBJETIVOS ESPECÍFICOS.....</b>	<b>6</b>
<b>5 CICLO DE DESARROLLO.....</b>	<b>6</b>
5.1 ANALISIS DE REQUISITOS.....	6
5.1.1 REQUISITOS FUNCIONALES.....	6
5.1.2 REQUISITOS NO FUNCIONALES.....	7
5.2 DISEÑO DEL SISTEMA.....	7
5.2.1 DIAGRAMA ENTIDAD RELACIÓN.....	7
5.2.2 DIAGRAMA DE CLASES.....	8
5.2.3 ESBOZO DE LA ESTÉTICA.....	9
5.3 Desarrollo e Implementación.....	9
5.3.1 CONFIGURACIÓN DEL ENTORNO.....	9
5.3.2 ESTRUCTURA.....	9
5.4 PRUEBAS Y VALIDACIÓN.....	10
<b>6 METODOLOGÍA DE DESARROLLO DEL PROYECTO.....</b>	<b>10</b>
6.1 METODOLOGÍA RUP.....	10
6.2 METODOLOGÍA AGILE.....	10
6.3 ¿CÚAL ESCOJO?.....	10
<b>7 MARCOS DE TRABAJO APLICADOS.....</b>	<b>10</b>
7.1 ¿POR QUÉ KANBAN?.....	10
7.2 APLICACIÓN.....	10
<b>8 TAREAS.....</b>	<b>11</b>
8.1 TAREA 1: CONFIGURACIÓN DEL ENTRONO DE DESARROLLO.....	11
8.1.1 Subtarea 1.1: Instalación de herramientas (backend).....	11
8.1.2 Subtarea 1.2: Instalación de herramientas (frontend).....	11
8.1.3 Subtarea 2.1 configurar el entorno (backend).....	11
8.1.4 Subtarea 2.2: Configuración del entorno de desarrollo para el frontend.....	11

<b>8.2 TAREA 2: PLANIFICACIÓN Y ANÁLISIS DE REQUISITOS.....</b>	<b>12</b>
8.2.1 Definición de requisitos funcionales.....	12
8.2.2 Requisitos no funcionales.....	12
8.2.3 Selección de tecnologías y herramientas.....	12
8.2.4 Planificación de plazos y tareas.....	12
<b>8.3 TAREA 3: DISEÑO DEL SISTEMA.....</b>	<b>13</b>
8.3.1 Diseño de la Base de Datos.....	13
8.3.2 Diseño de la Arquitectura del Sistema.....	13
8.3.3 Diseño de la Interfaz de Usuario (UI).....	13
<b>8.4 IMPLEMENTACIÓN DEL SISTEMA DE AUTENTICACIÓN.....</b>	<b>13</b>
8.4.1 Creación de los formularios de registro y login.....	13
8.4.2 Creación de perfiles de usuario.....	13
8.4.3 Pruebas de autenticación.....	13
<b>8.5 DESARROLLO DE LA BIBLIOTECA DE JUEGOS.....</b>	<b>14</b>
8.5.1 Estructuración de la base de datos para los juegos.....	14
8.5.2 Creación de formularios para agregar juegos.....	14
8.5.3 Implementación de la relación entre usuarios y juegos.....	14
8.5.4 Creación de vistas para mostrar los juegos en la biblioteca del usuario.....	14
8.5.5 Pruebas de la funcionalidad de la biblioteca.....	14
<b>8.6 DESARROLLO DEL DIARIO.....</b>	<b>15</b>
8.6.1 Diseño y creación de la base de datos del diario.....	15
8.6.2 Desarrollo de la funcionalidad para agregar entradas al diario.....	15
8.6.3 Implementación de la visualización de entradas anteriores.....	15
8.6.4 Pruebas del diario de juegos.....	15
<b>8.7 IMPLEMENTACIÓN DE UN BUSCADOR DE JUEGOS.....</b>	<b>15</b>
8.7.1 Diseño de la interfaz de búsqueda.....	15
8.7.2 Implementación de la lógica de búsqueda de juegos.....	15
8.7.3 Desarrollo de la página de resultados de búsqueda.....	15
8.7.4 Pruebas del buscador de juegos.....	15
<b>8.8 DESARROLLO DE UNA INTERFAZ DE USUARIO.....</b>	<b>16</b>
8.8.1 Diseño de la página de login y registro.....	16
8.8.2 Diseño de la página de perfil del usuario.....	16
8.8.3 Diseño de la página de la biblioteca de juegos.....	16
8.8.4 Diseño de la página del diario de juegos.....	16
8.8.5 Pruebas de la interfaz de usuario.....	16
<b>8.9 INTEGRACIÓN DE LAS APIs.....</b>	<b>17</b>
8.9.1 Investigación sobre las APIs.....	17
8.9.2 Integración de la API en Django.....	17
8.9.3 Visualización de datos obtenidos de la API en el frontend.....	17
8.9.4 Pruebas de integración con las APIs.....	17
<b>8.10 TIEMPO JUGADO Y GESTIÓN DEL DIARIO.....</b>	<b>18</b>
8.10.1 Diseño de la base de datos para almacenar el tiempo jugado.....	18
8.10.2 Implementación de la lógica para calcular el tiempo jugado por el usuario.....	18
8.10.3 Desarrollo de la visualización del tiempo jugado en el perfil del usuario.....	18
8.10.4 Pruebas de la gestión de tiempo jugado.....	18
<b>9 RECURSOS HUMANOS.....</b>	<b>19</b>
<b>9.1 Desarrollador Backend (Álvaro Acosta).....</b>	<b>19</b>

<b>9.2 Desarrollador Frontend (Álvaro Acosta).....</b>	<b>19</b>
<b>9.3 Tester (Por determinar).....</b>	<b>19</b>
<b>9.4 Gestor de Proyecto (Álvaro Acosta).....</b>	<b>19</b>
<b>10 RECURSOS MATERIALES.....</b>	<b>20</b>
<b>11 CRONOGRAMA.....</b>	<b>20</b>
<b>12 PRESUPUESTO.....</b>	<b>21</b>
<b>13 POLÍTICA DE SEGUIMIENTO Y EVALUACIÓN.....</b>	<b>24</b>
<b>14 BIBLIOGRAFÍA - WEBGRAFÍA.....</b>	<b>24</b>

## **1 INTRODUCCIÓN**

El proyecto consiste en el desarrollo de una aplicación que sirva a modo de biblioteca y diario de videojuegos para sus usuarios. Donde estos puedan registrar cuanto han jugado a determinado juego, organizar su colección o biblioteca de videojuegos y recibir recomendaciones personalizadas. Además, incluirá una sección social, donde los usuarios podrán interactuar entre sí, compartiendo experiencias en los juegos, evaluándolos, haciendo reseñas de los mismos o incluso mostrándolos en el panel principal de su perfil.

## **2 ORIGEN Y CONTEXTUALIZACIÓN DEL PROYECTO**

### **2.1 Descripción de la necesidad a la que da respuesta el producto/servicio**

A día de hoy, con el auge de los videojuegos en formato digital, es normal encontrarse con usuarios que acumulen grandes bibliotecas de títulos, muchos de los cuales quedan en el olvido. Esta abundancia de opciones genera dos problemas principales: por un lado, jugadores que, a pesar de su extenso catálogo, terminan siempre jugando a los mismos títulos sin llegar a aprovechar su colección; por el otro lado, usuarios que, debido a la enorme oferta de videojuegos, se sienten abrumados y no encuentran títulos acordes a sus gustos, o se terminan desentendiendo de este pasatiempo.

Este problema se ve agravado por la falta de herramientas que ayuden a los jugadores a gestionar su colección de manera eficiente. Actualmente, muchas plataformas permiten la compra y almacenamiento de juegos digitales, pero pocas ofrecen funciones avanzadas para organizar, filtrar o planificar sesiones de juego según el tiempo disponible o las preferencias del usuario. Como consecuencia, muchos jugadores pierden el interés o terminan dedicando más tiempo a decidir qué jugar que a disfrutar realmente de los títulos que poseen.

Además, las plataformas actuales suelen ofrecer recomendaciones basadas en intereses comerciales o patrones de compra en lugar de enfocarse en preferencias específicas de los usuarios. Esto limita la capacidad de los jugadores para descubrir títulos que realmente encajen con sus gustos y experiencias previas. Del mismo modo, la falta de herramientas para realizar un seguimiento del tiempo de juego o establecer prioridades dentro de una biblioteca de videojuegos dificulta la toma de decisiones y la optimización del entretenimiento digital.

A esto se le suma el factor económico, ya que los precios de los videojuegos pueden variar considerablemente entre diferentes tiendas y plataformas. Sin una herramienta que permita comparar precios de manera rápida y eficiente, los jugadores pueden terminar pagando más de lo necesario o perdiendo oportunidades de descuento en títulos de su interés.

Este proyecto busca ofrecer una solución integral a todos estos problemas mediante una plataforma que permita a sus usuarios gestionar su biblioteca de videojuegos, registrar sus sesiones de juego, recibir recomendaciones personalizadas basadas en sus gustos reales y acceder a información actualizada sobre precios y disponibilidad de títulos en distintas plataformas. De esta forma, se pretende mejorar la experiencia de los jugadores, ayudándolos a optimizar su tiempo, descubrir nuevas experiencias y sacar el máximo provecho de su colección.

## **3 OBJETIVO GENERAL DEL PROYECTO**

El objetivo general del proyecto es desarrollar una aplicación que permita a los jugadores registrar su tiempo de juego, organizar su biblioteca, buscar recomendaciones personalizadas en base a sus gustos, y acceder a información relevante sobre los juegos que deseé.

## 4 OBJETIVOS ESPECÍFICOS

- Implementar un sistema de diario donde los usuarios puedan registrar sus horas de juego.
- Permitir la creación de listas personalizadas de videojuegos.
- Integrar APIs externas (SteamDB, IGDB, MobyGames) para obtener información de videojuegos.
- Desarrollar una función de comparación de precios de videojuegos mediante API o web scraping.
- Incluir una sección social para que los jugadores puedan interactuar y compartir experiencias.
- Diseñar diferentes perfiles de usuario con roles específicos (Administradores, Desarrolladores, Revisores, Jugadores).

## 5 CICLO DE DESARROLLO

### 5.1 ANALISIS DE REQUISITOS

#### 5.1.1 REQUISITOS FUNCIONALES

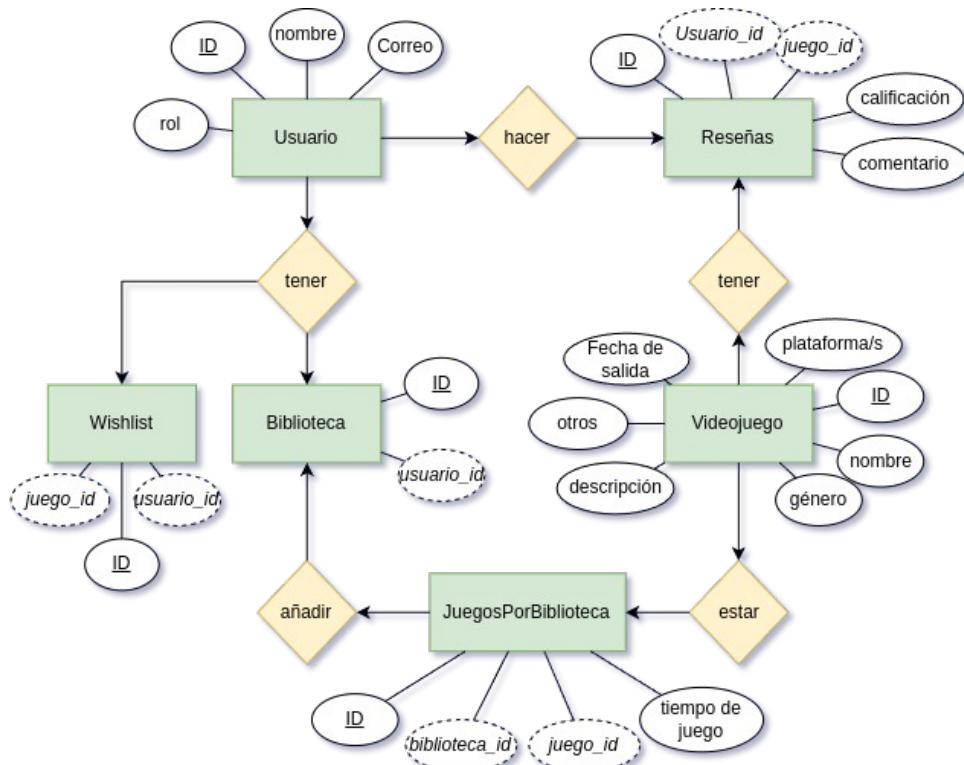
- **REGISTRO Y AUTENTICACIÓN DE USUARIOS:** La aplicación debe permitir a los usuarios crear cuentas y gestionar sus credenciales de forma segura.
- **GESTIÓN DE LA BIBLIOTECA DE VIDEOJUEGOS:** Los usuarios podrán agregar, eliminar y organizar sus títulos dentro de su colección personal.
- **REGISTRO DE TIEMPO DE JUEGO:** Los usuarios podrán registrar las horas que han jugado a cada videojuego, ya sea de forma manual o automática, con un cronómetro integrado en la plataforma.
- **COMPARACIÓN DE PRECIOS DE VIDEOJUEGOS:** Los usuarios podrán comparar los precios de un videojuego en diferentes tiendas y plataformas utilizando APIs como SteamDB, IGDB, MobyGames...
- **INTERACCIÓN SOCIAL ENTRE USUARIOS:** La aplicación incluirá una sección social, donde los usuarios puedan **compartir** experiencias, **reseñar** títulos, y **seguir** a otros usuarios para ver su biblioteca o actividad reciente.
- **PERFILES DE USUARIO CON ROLES DIFERENTES:** Los usuarios podrán ser:
  - **Administradores:** con control total sobre la plataforma
  - **Desarrolladores:** podrán modificar datos sobre sus videojuegos, así como añadir datos sobre cada versión del juego, una página de datos de curiosos, recibir feedback de como es acogido su videojuego en la comunidad...
  - **Revisores:** revisión del contenido y moderación
  - **Jugadores:** podrán interactuar con la plataforma creándose su perfil, siguiendo a otros usuarios y gestionando su biblioteca
- **VISUALIZACIÓN DE JUEGOS DESTACADOS O POPULARES:** En la página principal, cada usuario podrá ver una lista de juegos populares, o los destacados en función de la actividad de la comunidad.
- **SISTEMA DE NOTIFICACIONES:** Los usuarios recibirán notificaciones sobre nuevas recomendaciones, interacciones en su perfil y en caso de tener algún juego en la lista de deseados notificaciones de cambios de precio.

### 5.1.2 REQUISITOS NO FUNCIONALES

- **SEGURIDAD:** A pesar de que se usará HTTP, se implementará protección CSRF para evitar ataques.
  - El sistema utilizará cookies para almacenar el token CSRF.
  - Las contraseñas de los usuarios estarán cifradas con un algoritmo seguro.
- **INTERFAZ:** La interfaz deberá facilitar la navegación por la misma
- **ACCESIBILIDAD:** La aplicación permitirá ajustar entre temas, para facilitar su lectura, como por ejemplo un tema que contraste mucho entre las diferentes secciones de la app. Además de permitir algunos cambios extra por estética a gusto del usuario.
- **INTEGRACIÓN DE LAS APIs:** La aplicación se tendrá que integrar correctamente a las APIs, para obtener correctamente la información de cada juego.
- **COMPATIBILIDAD CON MÚLTIPLES PLATAFORMAS:** La aplicación deberá poder ser usada tanto en dispositivos móviles como en ordenadores de escritorio.

## 5.2 DISEÑO DEL SISTEMA

### 5.2.1 DIAGRAMA ENTIDAD RELACIÓN



#### • Usuario

Se almacenará la información básica del usuario, para su registro y su relación con otras tablas

#### • Biblioteca

Esta tabla se utilizará para relacionar cada juego que el usuario registre en su biblioteca. Se almacenarán también datos específicos de cada usuario, como su tiempo de juego

- **Reseñas**

Se almacenará cada reseña de cada jugador, para así mostrarlas en la pestaña principal de cada juego

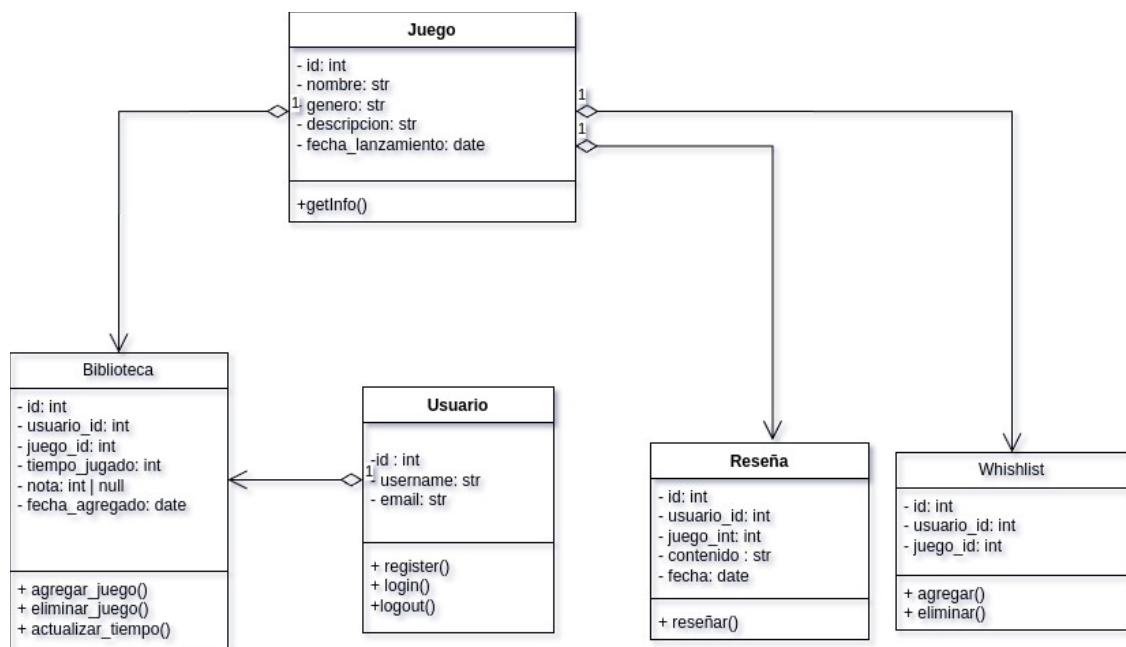
- **Videojuego**

La información de cada juego, esta provendrá de una API

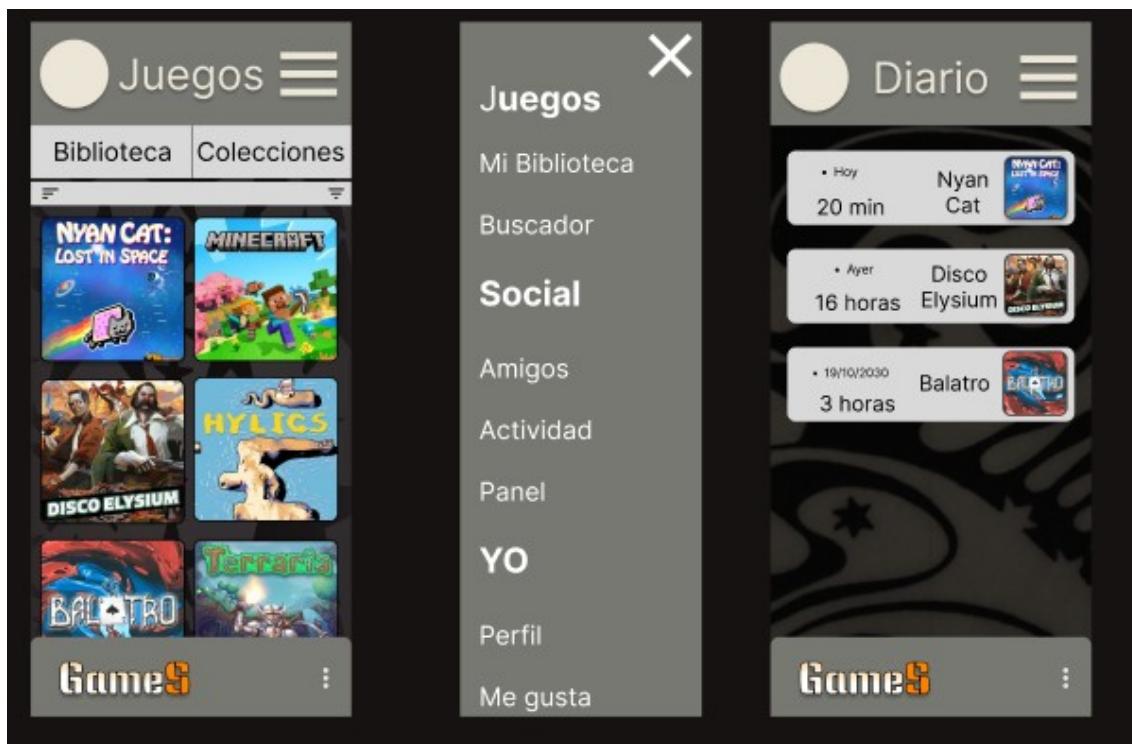
- **Wishlist**

Cada usuario puede tener su propia lista de deseos, esto vendría a formar parte de el sistema de recomendaciones y notificaciones. Probablemente se requiera otra tabla para almacenar las preferencias generales de cada jugador (sus géneros favoritos, franquicias, o desarrolladoras)

### 5.2.2 DIAGRAMA DE CLASES



### 5.2.3 ESBOZO DE LA ESTÉTICA



En el primer panel se observa como cada usuario puede ver su panel de videojuegos. Pudiendo filtrar, ordenar, y cambiar el tamaño o vista que se muestra. En el header de la aplicación siempre se podrá acceder a un menú para desplazarse por las principales rutas de la app, y también cada usuario podrá ver su diario de actividad.

Estos tres componentes son la parte más esencial de la aplicación, El núcleo(la gestión de nuestra biblioteca), la navegación por la app, la organización de nuestra actividad.

## 5.3 Desarrollo e Implementación

A continuación se detallará el proceso de desarrollo del proyecto. Lo cual comienza desde la creación desde cero utilizando las herramientas anteriormente mencionadas.

### 5.3.1 CONFIGURACIÓN DEL ENTORNO

Primero se instalarán las dependencias y herramientas necesarias. Deberé de tener instalado **Python** en primer lugar, y luego las dependencias del proyecto, **Django** y **Django REST Framework**.

Para Angular será necesario **Node.js** y **NPM** para poder crear el proyecto.

Y finalmente puede que se opte por **PostgreSQL** o **MySQL** para gestionar la base de datos.

### 5.3.2 ESTRUCTURA

El proyecto deberá detener una estructuración limpia y cómoda para facilitar lo mejor posible la ubicación de cada parte del código y los elementos que contenga.

## 5.4 PRUEBAS Y VALIDACIÓN

- **Pruebas de integración:** Se validará la comunicación entre el backend en Django y el frontend en Angular.
- **Pruebas de usabilidad:** Se evaluará que la interfaz sea clara y fácil de usar.
- **Pruebas de seguridad:** Se verificarán los **CSRF tokens**, el almacenamiento seguro de contraseñas y posibles vulnerabilidades.
- **Pruebas funcionales:** Comprobar que las funciones principales (registro, biblioteca, listas, interacciones) operen bien.
- **Retroalimentación:** Se recogerán sugerencias.

## 6 METODOLOGÍA DE DESARROLLO DEL PROYECTO

### 6.1 METODOLOGÍA RUP

**RUP** es un proceso de desarrollo estructurado en cuatro fases: Inicio, Elaboración, Construcción y Transición. Su enfoque es más rígido que Agile y requiere una planificación detallada antes de comenzar la implementación. Se basa en la gestión de riesgos y en la documentación extensa.

### 6.2 METODOLOGÍA AGILE

**Agile** es una metodología ágil que divide el desarrollo en ciclos cortos llamados **sprints**, permitiendo entregas frecuentes de nuevas funcionalidades y adaptaciones constantes según las necesidades del usuario.

### 6.3 ¿CÚAL ESCOJO?

Optaré por la metodología **Agile**, ya que ofrece una mayor flexibilidad y se adapta mejor a un trabajo individual. Aunque **RUP** puede ser útil para proyectos más grandes con equipos diversos, su enfoque más estructurado y la necesidad de una planificación detallada no se ajustan a las características de este proyecto. Dado que es un trabajo personal con un alcance moderado, **Agile** proporciona una estructura más adecuada para gestionar de manera eficiente el desarrollo.

## 7 MARCOS DE TRABAJO APLICADOS

En este proyecto, se ha optado por implementar el marco de trabajo **Kanban**, el cual es ampliamente utilizado en entornos de desarrollo ágil debido a su flexibilidad y facilidad de adaptación.

### 7.1 ¿POR QUÉ KANBAN?

El marco **Kanban** se caracteriza por su **enfoque visual** y su sistema de gestión de tareas mediante tableros y tarjetas. Esta metodología permite una **organización continua** de tareas sin la necesidad de establecer ciclos o sprints definidos, lo que resulta ideal para un proyecto individual con un alcance moderado como este.

### 7.2 APLICACIÓN

Durante el desarrollo se organizarán las tareas en función de como vayan avanzando. Colocaré todas las tareas en una sección de tareas pendientes, para luego mover aquellas que comiencen a progresar a una sección donde se refleje que están progresando.

Finalmente aquellas tareas que han sido finalizadas y hayan pasado las pruebas correspondientes se moverán a las tareas completadas.

## 8 TAREAS

### 8.1 TAREA 1: CONFIGURACIÓN DEL ENTRONO DE DESARROLLO

La primera tarea consistirá en configurar el entorno, dejarlo todo preparado para comenzar a programar, esto consistirá en configurar las herramientas necesarias para el frontend y el backend. Y la instalación de los programas necesarios en el sistema

#### 8.1.1 *Subtarea 1.1: Instalación de herramientas (backend)*

**Se instalarán todas las herramientas necesarias y se asegurará que el sistema operativo tenga los programas adecuados.**

- Instalar Python en el sistema operativo.
- Crear un entorno virtual en Python utilizando venv para manejar las dependencias del proyecto de manera aislada.
- Instalar Django usando pip (gestor de paquetes de Python).
- Instalar MySQL como el sistema de gestión de bases de datos.
- Instalar MySQL Workbench para administrar la base de datos de manera visual.

#### 8.1.2 *Subtarea 1.2: Instalación de herramientas (frontend)*

- Instalar Angular CLI globalmente para gestionar el desarrollo del frontend.
- Crear el proyecto inicial en Angular y realizar la configuración básica.

#### 8.1.3 *Subtarea 2.1 configurar el entorno (backend)*

- Integrar las rutas de Angular para que puedan acceder a las APIs proporcionadas por el backend de Django.
- Asegurarse de que el frontend pueda realizar solicitudes a la API y que los datos se reciban correctamente.

#### 8.1.4 *Subtarea 2.2: Configuración del entorno de desarrollo para el frontend*

- Configurar el archivo angular.json y los entornos de desarrollo y producción en Angular.
- Integrar las rutas de Angular para que puedan acceder a las APIs proporcionadas por el backend de Django.
- Asegurarse de que el frontend pueda realizar solicitudes a la API y que los datos se reciban correctamente.

	Duración (horas)	Fecha inicio	Fecha fin	Responsable
CONFIGURACIÓN DEL ENTRONO DE DESARROLLO	15	01-04-2025	07-04-2025	Álvaro Acosta
Instalación de herramientas	6	01-04-2025	04-04-2025	Álvaro Acosta
Configuración del entorno	9	04-04-2025	07-04-2025	Álvaro Acosta

## **8.2 TAREA 2: PLANIFICACIÓN Y ANÁLISIS DE REQUISITOS**

Esta tarea se encargará de recopilar y definir todos los requisitos del sistema. Se tratarán tanto los requisitos funcionales como no funcionales.

### **8.2.1     *Definición de requisitos funcionales***

Descripción de las funciones que debe realizar el sistema, como la gestión de usuarios la biblioteca de juegos, y la interacción con APIs.

### **8.2.2     *Requisitos no funcionales***

Se identificarán los requisitos no funcionales del sistema, como el rendimiento, la seguridad y la usabilidad. Esto ayudará a asegurar que el sistema cumpla con las expectativas de los usuarios y funcione de manera eficiente.

### **8.2.3     *Selección de tecnologías y herramientas***

Se decidirán las tecnologías a utilizar para el desarrollo del proyecto. También se seleccionarán otras herramientas necesarias para el desarrollo, como editores de código y sistemas de control de versiones.

### **8.2.4     *Planificación de plazos y tareas***

División de trabajo para que el proyecto se desarrolle de manera ordenada y eficiente.

	Duración (horas)	Fecha inicio	Fecha fin	Responsable
PLANIFICACIÓN Y ANÁLISIS DE REQUISITOS	20	08-04-2025	14-04-2025	Álvaro Acosta
Requisitos	10	08-04-2025	10-04-2025	Álvaro Acosta
Planificación de plazos y tareas	5	11-04-2025	12-04-2025	Álvaro Acosta
Selección de tecnologías	5	13-04-2025	14-04-2025	Álvaro Acosta

### **8.3 TAREA 3: DISEÑO DEL SISTEMA**

Se realizará el diseño de la base de datos, la arquitectura del sistema y la interfaz de usuario.

#### **8.3.1      *Diseño de la Base de Datos***

Se realizará el esquema de la base de datos utilizando diagramas de entidad-relación.

#### **8.3.2      *Diseño de la Arquitectura del Sistema***

Se planteará la estructura de la aplicación, cómo se manejarán las interacciones entre el frontend y el backend.

#### **8.3.3      *Diseño de la Interfaz de Usuario (UI)***

Se hará un esbozo del diseño visual de las pantallas principales de la aplicación, como las páginas de inicio, perfil de usuario, biblioteca de juegos, diario y las interacciones básicas entre el usuario y el sistema.

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
DISEÑO DEL SISTEMA	40	15-04-2025	22-04-2025	Álvaro Acosta
Diseño de la Base de Datos	18	15-04-2025	17-04-2025	Álvaro Acosta
Diseño de la Arquitectura del Sistema	10	18-04-2025	19-04-2025	Álvaro Acosta
Diseño de la Interfaz de Usuario (UI)	12	20-04-2025	22-04-2025	Álvaro Acosta

### **8.4 IMPLEMENTACIÓN DEL SISTEMA DE AUTENTICACIÓN**

#### **8.4.1      *Creación de los formularios de registro y login***

Crear formularios de registro e inicio de sesión.

#### **8.4.2      *Creación de perfiles de usuario***

Crear perfiles vinculados a los usuarios.

#### **8.4.3      *Pruebas de autenticación***

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
IMPLEMENTACIÓN DEL SISTEMA DE AUTENTICACIÓN	25	23-04-2025	29-04-2025	Álvaro Acosta
Creación de formularios de registro y login	10	23-04-2025	24-04-2025	Álvaro Acosta
Creación de perfiles de usuario	8	25-04-2025	26-04-2025	Álvaro Acosta
Pruebas de autenticación	7	28-04-2025	29-04-2025	Álvaro Acosta

## **8.5 DESARROLLO DE LA BIBLIOTECA DE JUEGOS**

### **8.5.1      *Estructuración de la base de datos para los juegos***

Definir la estructura de datos para los juegos. Básicamente se estructurará que datos se recogerán de la API (fotos, descripciones, plataformas, fechas de salida...)

### **8.5.2      *Creación de formularios para agregar juegos***

Crear formularios para añadir juegos a la biblioteca. En principio tan solo con un botón en cada juego bastaría, también se podrán añadir colecciones enteras (Por ejemplo si has jugado la saga entera de Crash Bandicoot podrás añadirla toda a tu biblioteca con un solo clic)

### **8.5.3      *Implementación de la relación entre usuarios y juegos***

Configurar la relación entre los usuarios y los juegos. Juegos en wishlist, juegos gustados, juegos dentro de la biblioteca...

### **8.5.4      *Creación de vistas para mostrar los juegos en la biblioteca del usuario***

Mostrar los juegos de cada usuario en su biblioteca. Y permitir modificar esta vista, por ejemplo en cuantos juegos se muestran por página, o el filtrado ya sea por género, nombre fecha de salida...

### **8.5.5      *Pruebas de la funcionalidad de la biblioteca***

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
DESARROLLO DE LA BIBLIOTECA DE JUEGOS	50	30-04-2025	08-05-2025	Álvaro Acosta
Estructuración de la base de datos para los juegos	12	30-04-2025	01-05-2025	Álvaro Acosta
Creación de formularios para agregar juegos	12	01-05-2025	03-05-2025	Álvaro Acosta
Implementación de la relación entre usuarios y juegos	10	04-05-2025	05-05-2025	Álvaro Acosta
Creación de vistas para mostrar los juegos en la biblioteca del usuario	10	06-05-2025	07-05-2025	Álvaro Acosta
Pruebas de la funcionalidad de la biblioteca	6	08-05-2025	08-05-2025	Álvaro Acosta

## **8.6 DESARROLLO DEL DIARIO**

### **8.6.1      *Diseño y creación de la base de datos del diario***

Crear la base de datos para el diario.

### **8.6.2      *Desarrollo de la funcionalidad para agregar entradas al diario***

Implementar la opción de agregar entradas al diario. Con sus notas, y checks (por ejemplo si completas o abandonas algún juego poder resaltar aquella sesión en la que se hizo).

### **8.6.3      *Implementación de la visualización de entradas anteriores***

Mostrar las entradas previas en el diario. Permitir ver detalles acerca de las sesiones y añadir notas y verlas.

### **8.6.4      *Pruebas del diario de juegos***

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
DESARROLLO DEL DIARIO	40	09-05-2025	15-05-2025	Álvaro Acosta
Diseño y creación de la base de datos del diario	10	09-05-2025	10-05-2025	Álvaro Acosta
Desarrollo de la funcionalidad para agregar entradas al diario	10	11-05-2025	12-05-2025	Álvaro Acosta
Implementación de la visualización de entradas anteriores	10	13-05-2025	14-05-2025	Álvaro Acosta
Pruebas del diario de juegos	10	15-05-2025	15-05-2025	Álvaro Acosta

## **8.7 IMPLEMENTACIÓN DE UN BUSCADOR DE JUEGOS**

### **8.7.1      *Diseño de la interfaz de búsqueda***

Crear la interfaz para buscar juegos. Un input text en la pagina principal bastaría para comenzar a filtrar.

### **8.7.2      *Implementación de la lógica de búsqueda de juegos***

Desarrollar la lógica de búsqueda de juegos. En principio se conectara con la API y comenzará a recorrer coincidencias de búsqueda

### **8.7.3      *Desarrollo de la página de resultados de búsqueda***

Mostrar los resultados de búsqueda. Se deberán mostrar en páginas capadas, los resultados serán demasiados de no hacerlo.

### **8.7.4      *Pruebas del buscador de juegos***

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
IMPLEMENTACIÓN DE UN BUSCADOR DE JUEGOS	35	16-05-2025	22-05-2025	Álvaro Acosta
Diseño de la interfaz de búsqueda	10	16-05-2025	17-05-2025	Álvaro Acosta
Implementación de la lógica de búsqueda de juegos	10	18-05-2025	19-05-2025	Álvaro Acosta
Desarrollo de la página de resultados de búsqueda	8	20-05-2025	21-05-2025	Álvaro Acosta
Pruebas del buscador de juegos	7	22-05-2025	22-05-2025	Álvaro Acosta

## **8.8 DESARROLLO DE UNA INTERFAZ DE USUARIO**

### **8.8.1      *Diseño de la página de login y registro***

Estilizar las páginas de login y registro.

### **8.8.2      *Diseño de la página de perfil del usuario***

Estilizar la página del perfil de usuario.

### **8.8.3      *Diseño de la página de la biblioteca de juegos***

Estilizar la página para la biblioteca de juegos.

### **8.8.4      *Diseño de la página del diario de juegos***

Estilizar la página del diario de juegos.

### **8.8.5      *Pruebas de la interfaz de usuario***

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
DESARROLLO DE UNA INTERFAZ DE USUARIO	40	23-05-2025	30-05-2025	Álvaro Acosta
Diseño de la página de login y registro	10	23-05-2025	24-05-2025	Álvaro Acosta
Diseño de la página de perfil del usuario	10	25-05-2025	26-05-2025	Álvaro Acosta
Diseño de la página de la biblioteca de juegos	10	27-05-2025	28-05-2025	Álvaro Acosta
Diseño de la página del diario de juegos	5	29-05-2025	29-05-2025	Álvaro Acosta
Pruebas de la interfaz de usuario	5	30-05-2025	30-05-2025	Álvaro Acosta

## **8.9 INTEGRACIÓN DE LAS APIs**

### **8.9.1      *Investigación sobre las APIs***

Seleccionar las APIs necesarias para el proyecto.

### **8.9.2      *Integración de la API en Django***

Conectar la API con Django.

### **8.9.3      *Visualización de datos obtenidos de la API en el frontend***

Mostrar los datos de la API en la interfaz de usuario.

### **8.9.4      *Pruebas de integración con las APIs***

	Duración (horas)	Fecha inicio	Fecha fin	Responsable
INTEGRACIÓN DE LAS APIs	40	31-05-2025	08-06-2025	Álvaro Acosta
Investigación sobre las APIs	10	31-05-2025	01-06-2025	Álvaro Acosta
Integración de la API en Django	10	02-06-2025	03-06-2025	Álvaro Acosta
Visualización de datos obtenidos de la API en el frontend	10	04-05-2025	05-05-2025	Álvaro Acosta
Pruebas de integración con las APIs	10	07-06-2025	08-06-2025	Álvaro Acosta

## 8.10 TIEMPO JUGADO Y GESTION DEL DIARIO

### 8.10.1 *Diseño de la base de datos para almacenar el tiempo jugado*

Crear la base de datos para almacenar el tiempo jugado. Se podrá añadir manualmente o dejar un cronometro.

### 8.10.2 *Implementación de la lógica para calcular el tiempo jugado por el usuario*

Desarrollar la lógica para calcular el tiempo jugado. Los datos de tiempo de juego pueden ser relevantes para conocer al usuario así que se guardarán para mejorar las recomendaciones (Por ejemplo, si un usuario juega 2 horas semanales, no es muy coherente recomendarle juegos que consuman un montón de tiempo)

### 8.10.3 *Desarrollo de la visualización del tiempo jugado en el perfil del usuario*

Mostrar el tiempo jugado en el perfil del usuario, en cada juego al que haya jugado. Posiblemente mostrar todas sus horas en general, aunque puede verse excesivo.

### 8.10.4 *Pruebas de la gestión de tiempo jugado*

	<b>Duración (horas)</b>	<b>Fecha inicio</b>	<b>Fecha fin</b>	<b>Responsable</b>
TIEMPO JUGADO Y GESTION DEL DIARIO	45	08-06-2025	16-06-2025	Álvaro Acosta
Diseño de la base de datos para almacenar el tiempo jugado	10	08-06-2025	09-06-2025	Álvaro Acosta
Implementación de la lógica para calcular el tiempo jugado por el usuario	15	10-06-2025	12-06-2025	Álvaro Acosta
Desarrollo de la visualización del tiempo jugado en el perfil del usuario	10	13-06-2025	14-06-2025	Álvaro Acosta
Pruebas de la gestión de tiempo jugado	10	15-06-2025	16-06-2025	Álvaro Acosta

## 9 RECURSOS HUMANOS

### 9.1 Desarrollador Backend (Álvaro Acosta)

El encargado de la Implementación de la **lógica del servidor** utilizando Django y la **gestión de la base de datos**.

### 9.2 Desarrollador Frontend (Álvaro Acosta)

Desarrollo de la interfaz de usuario. Encargado de la Integración con el backend para obtener y mostrar datos. Deberá de asegurarse de la accesibilidad y facilidad en su uso.

### 9.3 Tester (Por determinar)

Los testers se encargarán de utilizar la aplicación y buscar posibles fallas, serán usuarios que trasteen en busca de mejorar la aplicación. Luego el equipo podrá encontrar más fallas en el proceso de desarrollo, pudiendo pulir de forma más eficiente el resultado final.

### 9.4 Gestor de Proyecto (Álvaro Acosta)

Planificación y seguimiento del progreso del proyecto. Coordinador general

## 10 RECURSOS MATERIALES

Procesador Intel Core i7-13700H (8C/OctaCore 4.7GHz, 24MB)

Memoria RAM 32GB DDR4 3200MHz

Tarjeta gráfica NVIDIA GeForce RTX 4060 8GB GDDR6

Sistema operativo Linux Mint

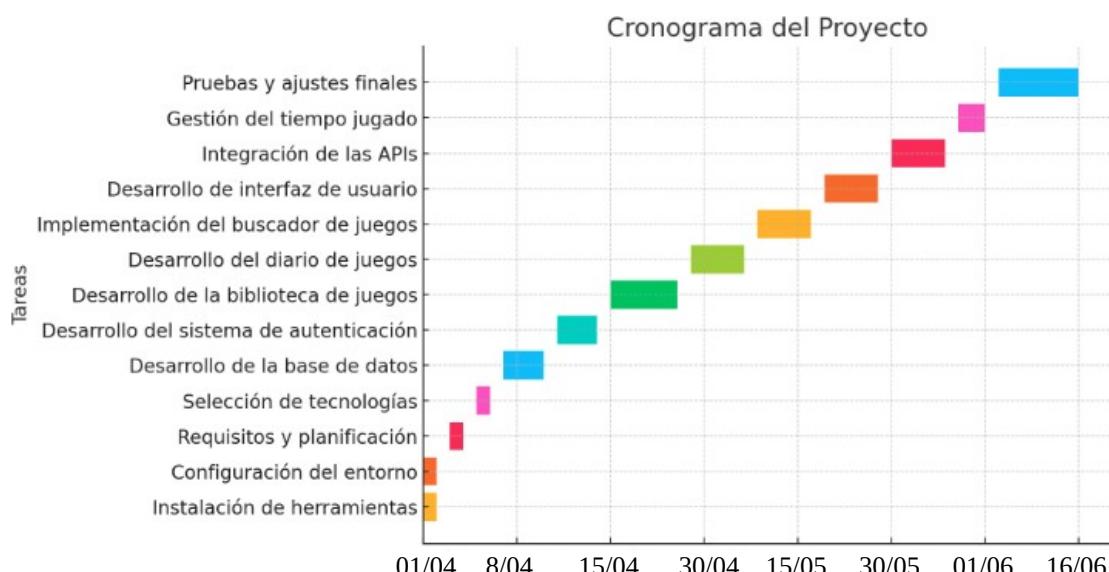
Almacenamiento de 1TB SSD

Memoria adicional de 2TB HDD

Internet (fibra óptica): 25€

Electricidad general del hogar: 50€

## 11 CRONOGRAMA



## 12 PRESUPUESTO

Tarea	Subtarea	Duración (horas)	Recursos humanos	Recursos materiales	Total
<b>Configuración del Entorno de Desarrollo</b>	Instalación de herramientas (backend)	6	90	0 €	315
	Configuración del entorno (backend)	9	135	0 €	
	Instalación de herramientas (frontend)	3	45	0 €	
	Configuración del entorno (frontend)	3	45	0 €	
<hr/>					
<b>Planificación y Análisis de Requisitos</b>	Requisitos funcionales	10	150	0 €	300
	Planificación de plazos y tareas	5	75	0 €	
	Selección de tecnologías	5	75	0 €	
<hr/>					
<b>Diseño del Sistema</b>	Diseño de la base de datos	18	270	0 €	600
	Diseño de la arquitectura del sistema	10	150	0 €	
	Diseño de la interfaz de usuario (UI)	12	180	0 €	
<hr/>					
<b>Implementación del Sistema de Autenticación</b>	Creación de los formularios de registro y login	10	150	0 €	375
	Creación de perfiles de usuario	8	120	0 €	
	Pruebas de autenticación	7	105	0 €	
<hr/>					

<b>Desarrollo de la Biblioteca de Juegos</b>	Estructuración de la base de datos para los juegos	12	180	0 €	750
	Creación de formularios para agregar juegos	12	180	0 €	
	Implementación de la relación entre usuarios y juegos	10	150	0 €	
	Creación de vistas para mostrar los juegos	10	150	0 €	
	Pruebas de la funcionalidad de la biblioteca	6	90	0 €	
<hr/>					
<b>Desarrollo del Diario</b>	Diseño y creación de la base de datos del diario	10	150	0 €	600
	Desarrollo de la funcionalidad para agregar entradas	10	150	0 €	
	Implementación de la visualización de entradas anteriores	10	150	0 €	
	Pruebas del diario de juegos	10	150	0 €	
<hr/>					
<b>Implementación de un Buscador de Juegos</b>	Diseño de la interfaz de búsqueda	10	150	0 €	525
	Implementación de la lógica de búsqueda de juegos	10	150	0 €	
	Desarrollo de la página de resultados de búsqueda	8	120	0 €	

	Pruebas del buscador de juegos	7	105	0 €	
<b>Desarrollo de una Interfaz de Usuario</b>	Diseño de la página de login y registro	10	150	0 €	525
	Diseño de la página de perfil del usuario	10	75	0 €	
	Diseño de la página de la biblioteca de juegos	10	150	0 €	
	Diseño de la página del diario de juegos	5	75	0 €	
	Pruebas de la interfaz de usuario	5	75	0 €	
<b>Integración de las APIs</b>	Investigación sobre las APIs	10	75	0 €	330
	Integración de la API en Django	10	90	0 €	
	Visualización de datos obtenidos de la API en frontend	10	90	0 €	
	Pruebas de integración con las APIs	10	75	0 €	
<b>Tiempo Jugado y Gestión del Diario</b>	Diseño de la base de datos para almacenar el tiempo jugado	10	90	0 €	390
	Implementación de la lógica para calcular el tiempo jugado	15	120	0 €	
	Desarrollo de la visualización del tiempo jugado	10	90	0 €	

	Pruebas de la gestión de tiempo jugado	10	90	0 €	
<hr/>					
<b>Energía y otros:</b>			300		
<b>TOTAL:</b>	356		6706		

## 13 POLÍTICA DE SEGUIMIENTO Y EVALUACIÓN

Durante el desarrollo del proyecto, se implementarán diversas acciones para asegurar que se cumplan los objetivos establecidos y se eviten desviaciones. Se realizará un seguimiento continuo de las tareas y plazos. Además se llevará un control detallado de las horas invertidas en cada tarea.

Además, se llevará a cabo una evaluación constante de las funcionalidades implementadas en cada fase, para verificar que el desarrollo se ajusta a los requisitos definidos. Este proceso de evaluación garantizará que el proyecto esté alineado con los objetivos y las expectativas establecidas desde su inicio.

## 14 BIBLIOGRAFÍA - WEBGRAFÍA

- **Xataka:** ["Muchos compran en Steam, pero pocos juegan: ya sabemos cuánto de tu biblioteca tienes sin usar"](#)
- **Vandal:** ["El valor de los juegos no jugados en Steam asciende a 19.000 millones de dólares."](#)
- **Hubspot:** [Metodología Agile: qué es y cómo aplicarla a tu proyecto](#)
- **Kanbantool:** [¿Por qué utilizar la metodología Kanban?](#)