



**CFP/UAX**

Centro de Formación Profesional

Proyecto de fin de ciclo

Desarrollo de aplicaciones multiplataforma

Curso 2022-2023



**Nombre del Proyecto: Game Haven**

**Alumno: Aitor Nain Mendoza Vallejo**

**Tutor: Germán Fernández**



## Game Haven

Abstract .....	4
Resumen y objetivos .....	5
Antecedentes del proyecto .....	6
Análisis y especificación de requisitos: .....	8
Propuesta de solución: .....	12
Plan de trabajo.....	21
Presupuesto Detallado.....	24
Desarrollo de la solución .....	27
Despliegue e instalación .....	29
Interfaz de usuario y capturas de pantalla .....	38
Pruebas y validación .....	42
Estrategia de marketing y monetización .....	44
Evolución y trabajo futuro .....	45
Conclusiones .....	46
Bibliografía .....	48

## Abstract

Welcome to GameHaven, your ultimate indie game hub. Think of it as a digital playground packed with unique games, made possible with the cutting-edge C#, Windows Presentation Foundation (.NET 4.7.2), and SQL Server 2022. These tech terms might seem complex, but they simply make GameHaven run like a dream.

And let's not forget the Hamburger Menu, our secret sauce for a sleek front-end design. It makes browsing our game trove as easy as pie. More than just a desktop application, GameHaven is a vibrant marketplace that champions creativity and indie game developers.

But our mission doesn't stop at providing a platform. We're on a quest to reshape the indie gaming landscape, delivering games that aren't just engaging but tell a unique story. GameHaven offers indie developers the spotlight, a stage to showcase their innovative concepts and skills.

At the heart of it all, GameHaven is your sanctuary for indie gaming. It's a place where unconventional gaming experiences aren't the exception; they're the rule. This is GameHaven, where indie gaming dreams take flight. Come join us on this exciting journey!

## Resumen y objetivos

Te presentamos GameHaven, plataforma para los fanáticos de los videojuegos indie. Es un proyecto hecho con mucho mimo, con C#, Windows Presentation Foundation (.NET 4.7.2) y SQL Server 2022. Puede sonar a puro tecnicismo, pero en esencia, estas son las herramientas que hacen que GameHaven sea súper rápido y amigable en tu ordenador.

Y no podemos olvidarnos del HamburgerMenu (framework front-end en wpf). Es nuestro pequeño truco para que navegar por la plataforma sea tan fácil como contar hasta tres. Pero GameHaven es mucho más que una simple aplicación. Es un lugar lleno de vida, donde damos voz a los desarrolladores indie para que enseñen al mundo sus videojuegos únicos.

Nuestro sueño va más allá de ser una plataforma más. Queremos transformar el universo indie, llevándote juegos que no solo te mantengan pegado a la pantalla, sino que te hagan vivir historias únicas e inolvidables. Aquí, los desarrolladores indie tienen un escaparate para mostrar todo su talento.

Así que, en resumen, GameHaven es tu refugio, tu paraíso de los videojuegos indie. Es un espacio donde las experiencias de juego extraordinarias son el pan de cada día. Bienvenido a GameHaven, el lugar donde los sueños de los videojuegos indie se hacen realidad. ¡Vamos, únete a nosotros en esta emocionante aventura!

## Antecedentes del proyecto

En el sector de los videojuegos, plataformas como Steam, Epic Games Store, GOG, Itch.io y Humble Bundle han jugado un papel crucial en la revolución de cómo los videojuegos son distribuidos y consumidos. Estas plataformas ofrecen una amplia variedad de juegos, incluyendo títulos de desarrolladores indies, estamos hablando del sector de pc gaming.

No obstante, a pesar de la inclusión de juegos indie, el dominio de títulos de grandes estudios a menudo ensombrece a los creadores indie en estas plataformas. Las limitaciones en la visibilidad y el descubrimiento de juegos indie representan un desafío para los desarrolladores que buscan conectar con su público objetivo.

Aquí es donde GameHaven interviene. Nuestra plataforma está exclusivamente dedicada a los juegos indie, ofreciendo un espacio en el que los desarrolladores pueden mostrar sus creaciones sin ser opacados por grandes títulos. Al centrarnos en una experiencia de usuario intuitiva y eficiente, facilitamos el descubrimiento de nuevos juegos para nuestros usuarios.

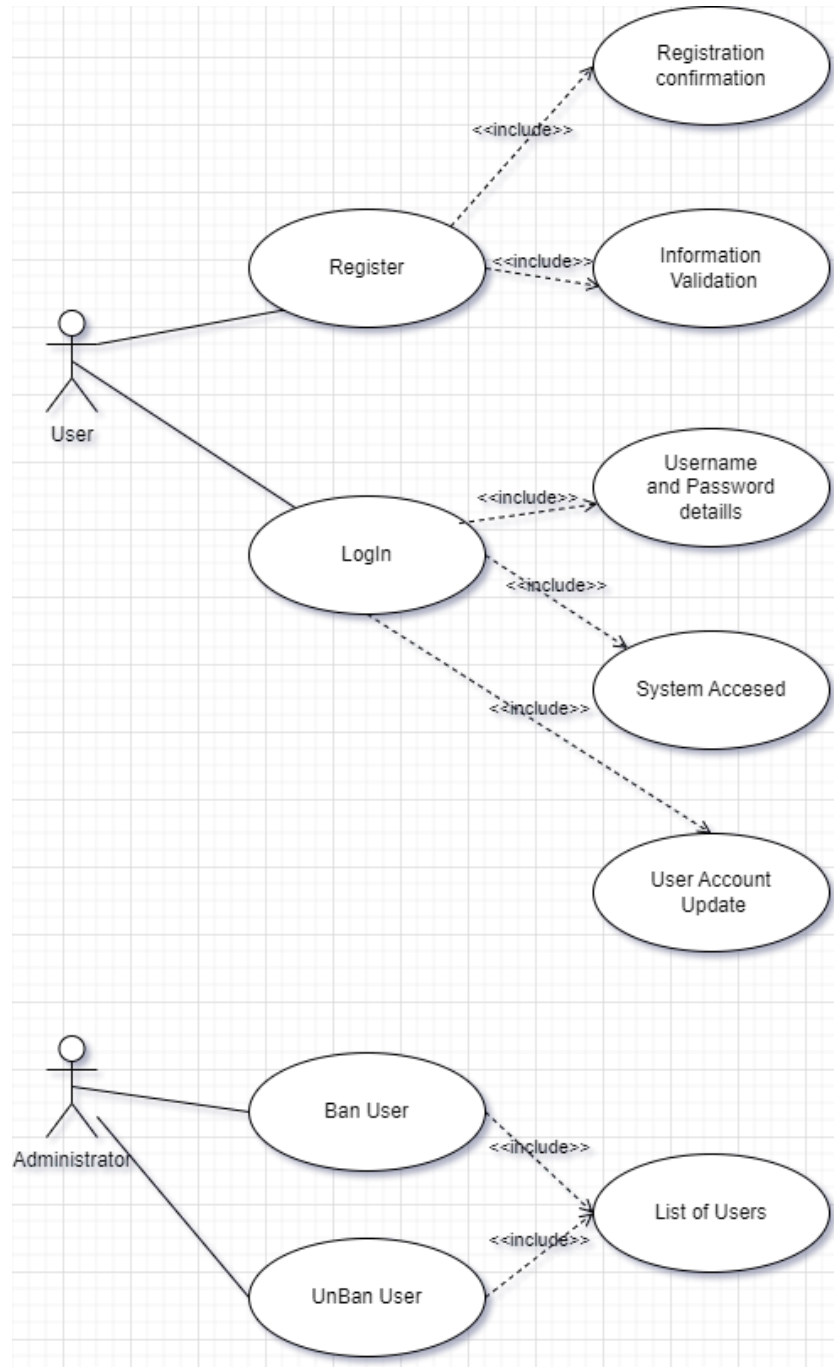
Aquí presentamos una tabla de comparación:

<b>Característica</b>	<b>Steam</b>	<b>Epic Games Store</b>	<b>GOG</b>	<b>Itch.io</b>	<b>Humble Bundle</b>	<b>GameHaven</b>
Amplia variedad de juegos	Sí	Sí	Sí	Sí	Sí	Sí
Juegos de grandes estudios	Sí	Sí	Sí	Limitado	Sí	No
Juegos indie	Sí	Sí	Sí	Sí	Sí	Sí
Enfoque en juegos indie	No	No	No	Sí	No	Sí
Experiencia de usuario intuitiva	Sí	Sí	Sí	Sí	Sí	Sí
Facilita el descubrimiento de juegos indie	No	No	No	Sí	No	Sí

GameHaven se presenta como una solución que aborda las deficiencias en las plataformas existentes, proporcionando un enfoque exclusivo en juegos indie y facilitando su descubrimiento.

## Análisis y especificación de requisitos

Para llevar a cabo el proyecto GameHaven, es crucial definir y entender los requisitos generales de la plataforma. A continuación, se presentan los requisitos y casos de uso claves:





### **Registro de usuarios:**

La plataforma deberá permitir a los usuarios crear una cuenta proporcionando detalles relevantes como nombre y contraseña.

### **Inicio de sesión y cierre de sesión:**

Los usuarios registrados deben poder iniciar sesión.

La aplicación iniciará pidiendo iniciar sesión, aunque tienes la opción de avanzar y navegar por la aplicación sin tener cuenta, pero lo más recomendable es tener cuenta para poder acceder a todas las funcionalidades.

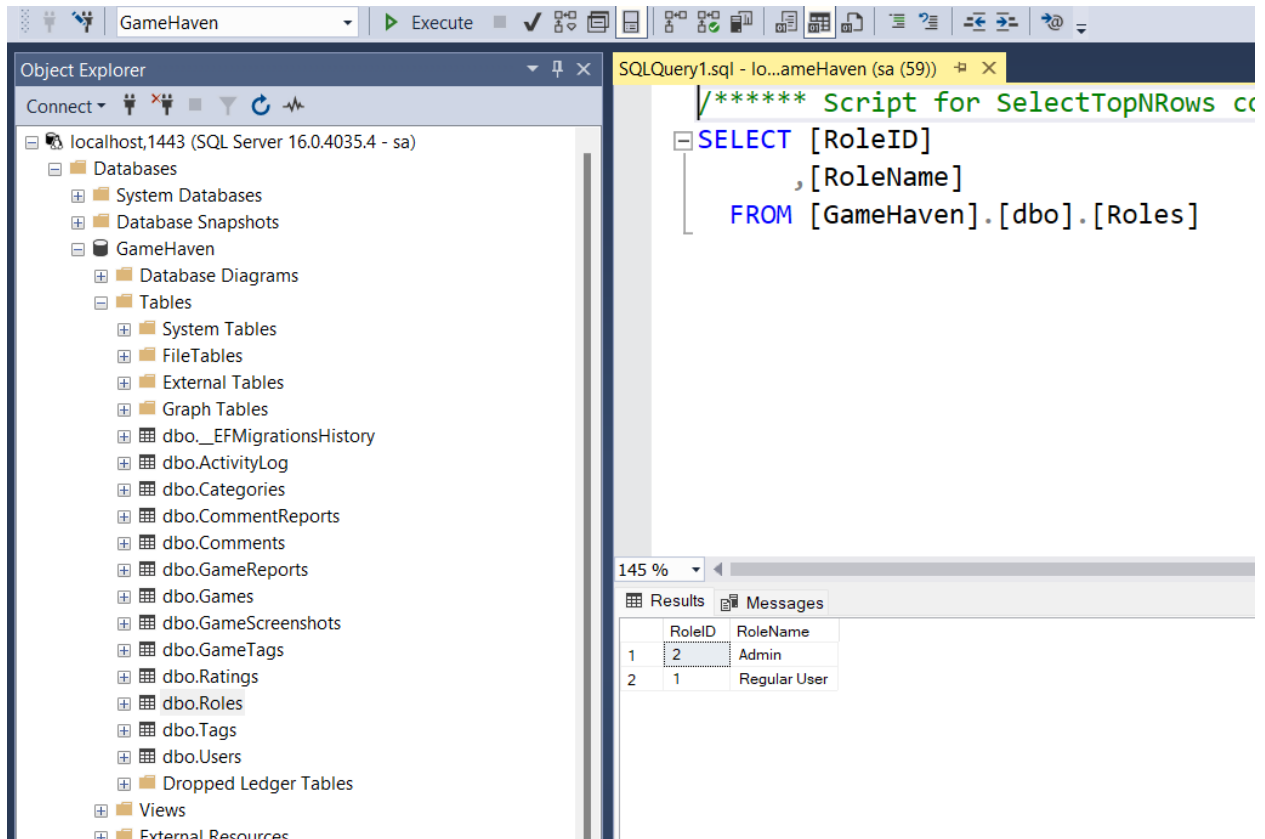
### **Subida de juegos:**

Los desarrolladores deben poder subir sus enlaces hacia la descarga de juegos a la plataforma o en caso del que el juego sea jugable online a través de un navegador Chrome, por ejemplo, su link respectivo, proporcionando detalles como el título del juego, una breve descripción, capturas de pantalla.

### **Banear usuarios:**

Los Admin podrán banear y desbanear usuarios

## Autenticación de usuario: Tenemos Admin y Regular User



The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure for 'GameHaven'. The main window shows a SQL query being executed. The query is a SELECT statement that retrieves RoleID and RoleName from the Roles table in the GameHaven database. The results pane at the bottom shows two rows of data: RoleID 2 for 'Admin' and RoleID 1 for 'Regular User'.

```
SQLQuery1.sql - lo...ameHaven (sa (59))
```

```
/* Script for SelectTopNRows of Table Roles */
SELECT [RoleID]
       , [RoleName]
FROM [GameHaven].[dbo].[Roles]
```

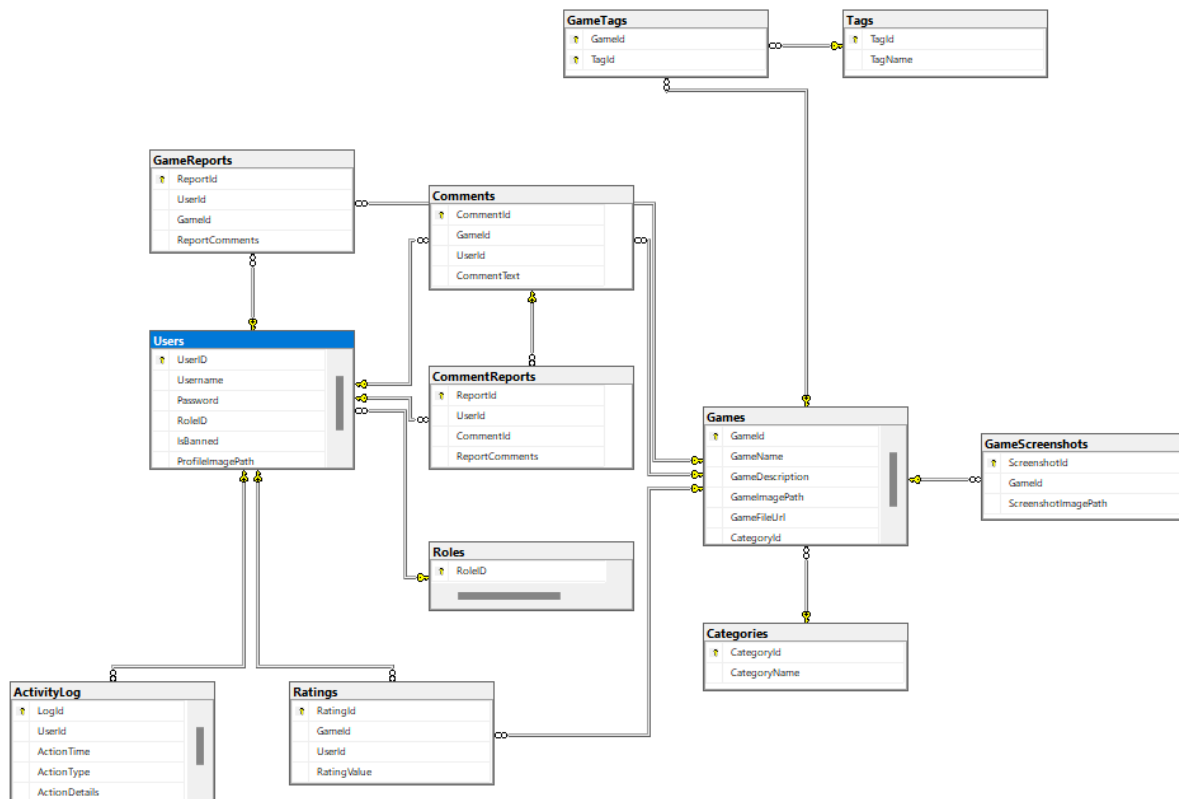
	RoleID	RoleName
1	2	Admin
2	1	Regular User

El modelo de desarrollo MVVM será aplicado en la implementación de estos casos de uso, garantizando una separación clara entre la lógica de la aplicación y la interfaz de usuario. Esto facilitará el desarrollo y mantenimiento de la plataforma, permitiendo que cada parte de la aplicación se desarrolle y se pruebe de forma independiente.

## Propuesta de solución:

La solución propuesta para GameHaven involucra la implementación de una arquitectura de sistema que se adhiere al patrón Model-View-ViewModel (MVVM) y utiliza SQL Server para la gestión de datos.

### Diagrama de mis modelos de base de datos GameHaven en sql server management studio



### **Mi base de datos con sus tablas:**

```
CREATE TABLE [dbo]. [ActivityLog] (  
  
    [LogId] INT IDENTITY (1,1) NOT NULL PRIMARY KEY,  
  
    [UserId] INT NOT NULL,  
  
    [ActionTime] DATETIME NULL DEFAULT (GETDATE ()),  
  
    [ActionType] NVARCHAR (100) NOT NULL,  
  
    [ActionDetails] NVARCHAR(MAX) NULL,  
  
    CONSTRAINT FK_ActivityLog_UserId FOREIGN KEY([UserId])  
  
    REFERENCES [dbo]. [Users]([UserID])  
  
);
```

```
CREATE TABLE [dbo]. [Categories] (  
  
    [CategoryId] INT IDENTITY (1,1) NOT NULL PRIMARY KEY,  
  
    [CategoryName] NVARCHAR (100) NOT NULL  
  
);
```

```
CREATE TABLE [dbo]. [CommentReports] (  
  
    [ReportId] INT IDENTITY (1,1) NOT NULL PRIMARY KEY,  
  
    [UserId] INT NULL,  
  
    [CommentId] INT NULL,  
  
    [ReportComments] NVARCHAR(MAX) NULL,  
  
    CONSTRAINT FK_CommentReports_CommentId FOREIGN KEY([CommentId])
```

```

REFERENCES [dbo]. [Comments]([CommentId]),

CONSTRAINT FK_CommentReports_UserId FOREIGN KEY([UserId])

REFERENCES [dbo]. [Users]([UserID])

);

CREATE TABLE [dbo]. [Comments] (

    [CommentId] INT IDENTITY (1,1) NOT NULL,

    [GameId] INT NOT NULL,

    [UserId] INT NOT NULL,

    [CommentText] NVARCHAR(MAX) NULL,

    PRIMARY KEY ([CommentId]),

    FOREIGN KEY ([GameId]) REFERENCES [dbo]. [Games] ([GameId]),

    FOREIGN KEY ([UserId]) REFERENCES [dbo]. [Users] ([UserID])

);

CREATE TABLE [dbo]. [Games] (

    [GameId] INT IDENTITY (1,1) NOT NULL,

    [GameName] VARCHAR (255) NOT NULL,

    [GameDescription] TEXT NOT NULL,

    [GameImagePath] VARCHAR (255) NULL,

    [GameFileUrl] NVARCHAR(MAX) NULL,

    [CategoryId] INT NOT NULL,

    PRIMARY KEY ([GameId]),

    CONSTRAINT FK_Games_CategoryId FOREIGN KEY([CategoryId]) REFERENCES

[dbo]. [Categories] ([CategoryId])
    
```

);

```
CREATE TABLE [dbo]. [GameReports] (

    [ReportId] INT IDENTITY (1,1) NOT NULL,

    [UserId] INT NOT NULL,

    [GameId] INT NOT NULL,

    [ReportComments] NVARCHAR(MAX) NULL,

    PRIMARY KEY ([ReportId]),

    FOREIGN KEY ([GameId]) REFERENCES [dbo]. [Games] ([GameId]),

    FOREIGN KEY ([UserId]) REFERENCES [dbo]. [Users] ([UserID])

);
```

```
CREATE TABLE [dbo]. [GameScreenshots] (

    [ScreenshotId] INT IDENTITY (1,1) NOT NULL PRIMARY KEY,

    [GameId] INT NULL,

    [ScreenshotImagePath] NVARCHAR(MAX) NULL,

    CONSTRAINT FK_GameScreenshots_GameId FOREIGN KEY([GameId])

    REFERENCES [dbo]. [Games]([GameId])

);
```

```
CREATE TABLE [dbo]. [GameTags] (

    [GameId] INT NOT NULL,

    [TagId] INT NOT NULL,

    CONSTRAINT PK_GameTags PRIMARY KEY ([GameId], [TagId]),

    CONSTRAINT FK_GameTags_GameId FOREIGN KEY ([GameId]) REFERENCES [dbo].

[Games] ([GameId]),
```

```
CONSTRAINT FK_GameTags_TagId FOREIGN KEY ([TagId]) REFERENCES [dbo].
[Tags] ([TagId])
```

```
);
```

```
CREATE TABLE [dbo]. [Ratings] (
    [RatingId] INT IDENTITY (1,1) NOT NULL,
    [GameId] INT NOT NULL,
    [UserId] INT NOT NULL,
    [RatingValue] INT NULL,
    PRIMARY KEY ([RatingId]),
    FOREIGN KEY ([GameId]) REFERENCES [dbo]. [Games] ([GameId]),
    FOREIGN KEY ([UserId]) REFERENCES [dbo]. [Users] ([UserID]),
    CHECK ([RatingValue] >= 1 AND [RatingValue] <= 5)
);
```

```
CREATE TABLE [dbo]. [Roles] (
    [RoleID] INT NOT NULL,
    [RoleName] NVARCHAR (100) NOT NULL,
    PRIMARY KEY ([RoleID])
);
```

```
CREATE TABLE [dbo]. [Tags] (
    [TagId] INT IDENTITY (1,1) NOT NULL PRIMARY KEY,
    [TagName] NVARCHAR (100) NOT NULL,
    CONSTRAINT UQ_Tags_TagName UNIQUE ([TagName])
);
```



```
CREATE TABLE [dbo]. [Users] (  
    [UserID] INT IDENTITY (1,1) NOT NULL,  
    [Username] NVARCHAR (100) NOT NULL,  
    [Password] NVARCHAR (100) NOT NULL,  
    [RoleID] INT NOT NULL,  
    [IsBanned] BIT NULL DEFAULT ((0)),  
    [ProfileImagePath] VARCHAR (255) NULL,  
    PRIMARY KEY ([UserID]),  
    CONSTRAINT FK_Users_RoleID FOREIGN KEY([RoleID]) REFERENCES [dbo]. [Roles]  
    ([RoleID])  
)
```

### Descripción de la arquitectura de la aplicación de escritorio:

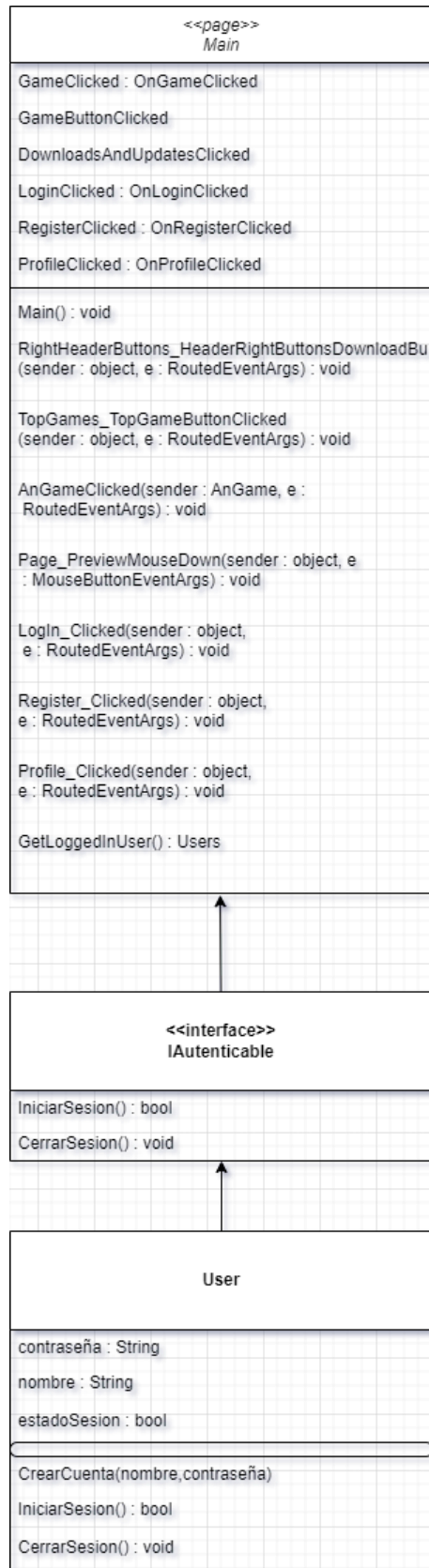
**Modelo:** El modelo es la representación de los datos y la lógica de negocio. En el caso de GameHaven, los modelos incluirán clases como Usuario, Juego, Comentario, entre otras. Cada una de estas clases se conectará a una tabla correspondiente en la base de datos SQL Server.

**Views:** Las Views son las representaciones visuales de los modelos. Por ejemplo, la vista de un juego mostrará la información de ese juego en la interfaz de usuario. Las Views en GameHaven estarán implementadas en XAML.

**ViewModel:** Los ViewModel actuarán como mediadores entre las Views y los Modelos. Ellos contienen la lógica para cómo mostrar los modelos en las Views y para manejar la interacción del usuario

En cuanto a la interfaz de usuario, se planea que sea intuitiva y fácil de navegar. Contará con una pantalla de inicio de sesión/registro, una página principal con una lista de juegos y categorías, páginas individuales de detalles de juegos, y un panel de administración para la gestión de la plataforma.

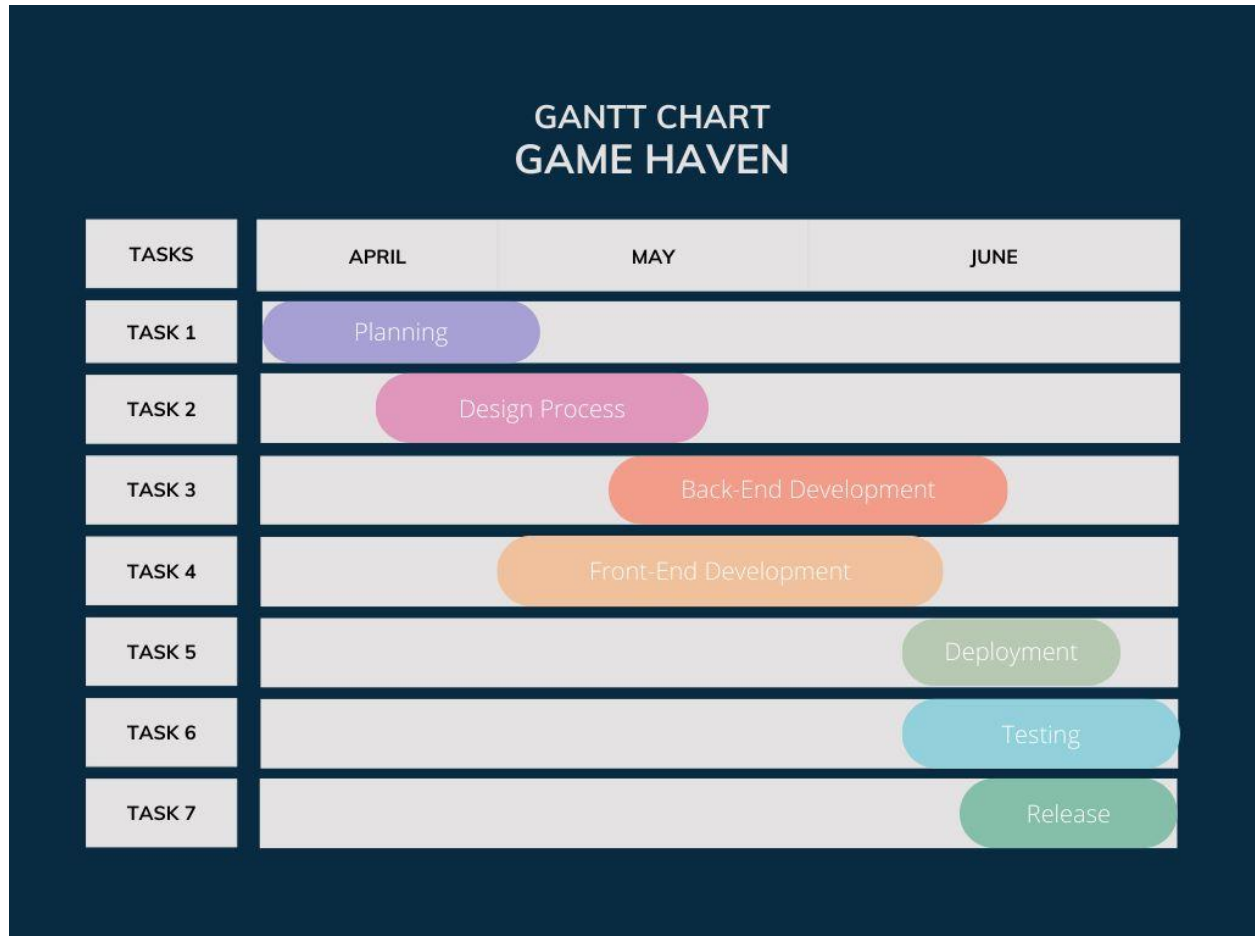
## Diagrama de clase de Main



Esta propuesta de solución permite un desarrollo de software modular y limpio, facilitando la capacidad de realizar cambios y pruebas en partes individuales de la aplicación sin afectar al resto. La elección de SQL Server como sistema de gestión de base de datos garantiza un manejo eficiente y seguro de los datos, lo cual es crucial para el éxito de GameHaven.

## Plan de trabajo

### Diagrama de Gantt



### **Sprint 1 (16 de abril - 29 de abril):**

Objetivo del sprint: Configurar el entorno y establecer la arquitectura del proyecto y preparar la interfaz.

#### **Tareas:**

Establecer el entorno de desarrollo y las herramientas necesarias.

Crear la base de datos y las entidades básicas (usuarios, juegos indie, categorías, etc.), se especificará en la sección de desarrollo de la solución con sus tablas y constraints.

Desarrollar la funcionalidad de registro y autenticación de usuarios, se mostrará un listado en la barra de arriba de la ventana.

Crear la interfaz básica de la tienda par luego desarrollarla usando framework de Hamburger Menu.

### **Sprint 2 (30 de abril - 13 de mayo):**

Objetivo del sprint: Crear una interfaz de usuario atractiva y funcional para la tienda de videojuegos indie.

#### **Tareas:**

Diseñar e implementar la interfaz de usuario para la navegación de la tienda y la visualización de los videojuegos indie.

Crear la funcionalidad para que los desarrolladores puedan subir sus videojuegos indie, incluyendo detalles del juego, imágenes y archivos de juego.

Desarrollar la funcionalidad de búsqueda de juegos y filtrado por categorías.

### **Sprint 3 (14 de mayo - 27 de mayo):**

Objetivo del sprint: Desarrollar la funcionalidad de la lista de videojuegos

#### **Tareas:**

Crear la funcionalidad de lista de videojuegos que permita a los usuarios explorar los juegos disponibles en la tienda.

### **Sprint 4 (28 de mayo - 1 de junio):**

Objetivo del sprint: Realizar pruebas, correcciones y preparar la plataforma para el lanzamiento.

#### **Tareas:**

Realizar pruebas exhaustivas de todas las funcionalidades.

Corregir cualquier bug o problema encontrado durante las pruebas.

Realizar mejoras de interfaz de usuario y experiencia de usuario basadas en la retroalimentación de las pruebas.

Preparar la plataforma para el lanzamiento y hacer las actividades de promoción correspondientes.

## Presupuesto Detallado

\*Este presupuesto es una estimación de las horas invertidas en el proyecto.

### **Investigación y análisis de mercado**

Investigación de plataformas existentes: 10 horas

Identificación de características y funcionalidades clave: 10 horas

Análisis de la competencia: 5 horas

Estudio de mercado: 10 horas

Total: 35 horas

### **Diseño y planificación**

Definición de requisitos: 15 horas

Diseño de la arquitectura: 10 horas

Diseño de la interfaz de usuario: 20 horas

Diseño de la base de datos: 10 horas

Planificación del desarrollo: 10 horas

Total: 65 horas



## **Desarrollo**

Implementación de la capa de presentación: 40 horas

Implementación de la capa de negocio: 60 horas

Implementación de la capa de datos: 30 horas

Integración de minijuegos: 20 horas

Integración con tiendas: 15 horas

Total: 175 horas

## **Pruebas y depuración**

Pruebas unitarias: 30 horas

Pruebas de integración: 20 horas

Pruebas de rendimiento: 10 horas

Pruebas de usabilidad: 10 horas

**Depuración y corrección de errores: 20 horas**

Total: 90 horas

## **Implementación y mantenimiento**

Implementación en servidores: 10 horas

Documentación: 20 horas

Mantenimiento y actualizaciones: 30 horas

Total: 60 horas

Horas totales: 425 horas

Total: 425 horas \* 20 EUR= 8,500 EUR

## Desarrollo de la solución

El desarrollo de la solución GameHaven ha sido un proceso de múltiples etapas que ha implicado una cuidadosa implementación de la solución propuesta en el diseño.

### **Configuración del entorno de desarrollo:**

El primer paso fue la configuración del entorno de desarrollo. Esto incluyó la instalación y configuración de las herramientas y tecnologías necesarias, como el IDE de Visual Studio, .NET 4.7.2, y SQL Server 2022(la última hasta ahora), mediante el uso de Docker en su última versión actual Docker Desktop 4.19.0.

### **Diseño de la base de datos:**

Se diseñó y creó la base de datos en SQL Server, creando tablas como Usuarios, Juegos, Comentarios y Valoraciones. Se establecieron relaciones entre las tablas según las necesidades del sistema.

### **Implementación del Modelo:**

Se implementaron las clases correspondientes a los modelos de datos, que representan las entidades de la aplicación. Estas clases interactúan directamente con la base de datos, permitiendo las operaciones de CRUD (Crear, Leer, Actualizar, Eliminar).

### **Implementación del ViewModel:**

Se crearon los ViewModel para cada uno de los modelos. Los ViewModel contienen la lógica de negocio y sirven como enlace entre los modelos y las Views. Utilizan la notificación de cambio de propiedad (INotifyPropertyChanged) y los comandos (ICommand) para facilitar la comunicación con las Views.

**Creación de las Views:** Se crearon las Views usando XAML. Estas Views se enlazan a los ViewModel correspondientes, lo que permite una separación clara entre la lógica de presentación y la lógica de negocio (la que hace el funcionamiento de la app con la base de datos).

**Pruebas y ajustes:** A medida que se iban desarrollando los componentes, se llevaban a cabo pruebas unitarias y de integración para asegurar la calidad y funcionalidad del software. Los errores y problemas encontrados durante las pruebas se corrigieron a medida que se presentaban.

**Despliegue e instalación:** Una vez que todos los componentes estuvieron funcionando correctamente en el entorno de desarrollo, se llevó a cabo el despliegue de la aplicación en un entorno de producción.

GameHaven es una aplicación de escritorio, Por ejemplo, la solicitud de servicios y el almacenamiento de datos son consideraciones importantes en el desarrollo de la solución. Utilizamos SQL Server para el almacenamiento de datos, lo que permite un manejo eficiente y seguro de los datos de los usuarios y de los juegos. Las solicitudes de servicios, como la autenticación de usuarios y la carga de datos de juegos, se manejan a través del patrón MVVM y los modelos de datos correspondientes.

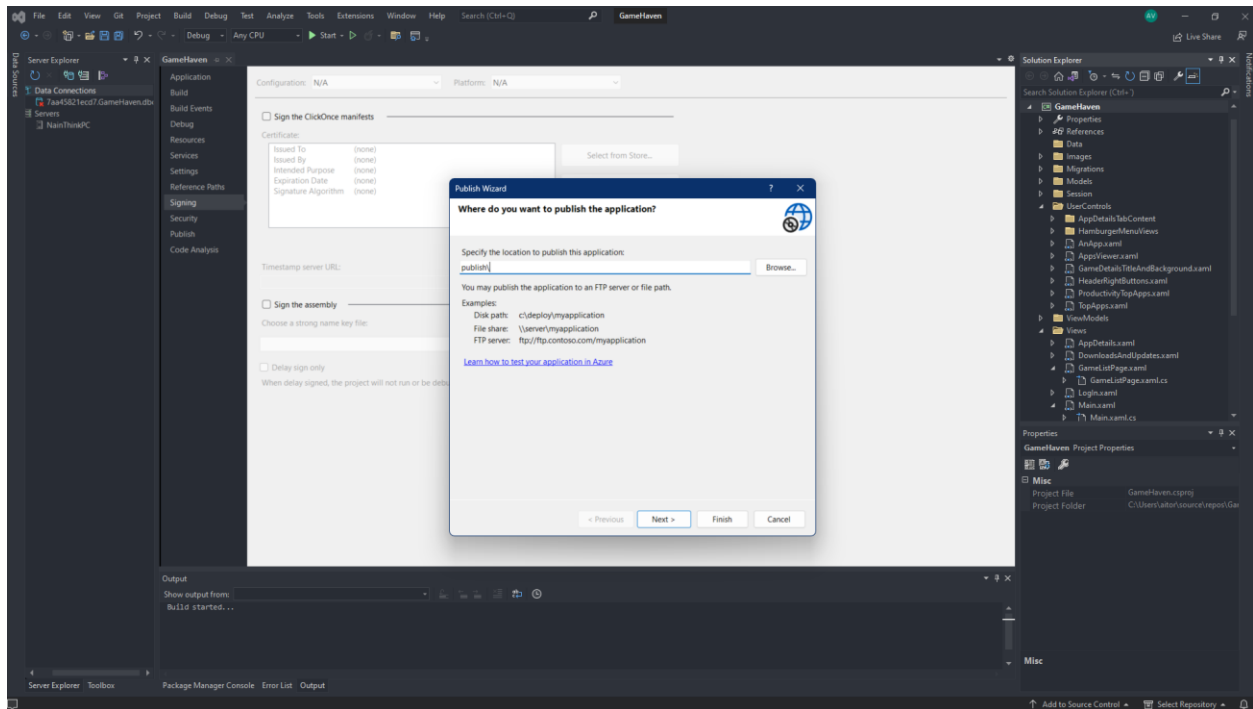
Este enfoque de desarrollo ha permitido un flujo de trabajo fluido y eficiente, asegurando que cada parte de GameHaven funcione como se espera y ofrezca una excelente experiencia al usuario.

## Despliegue e instalación

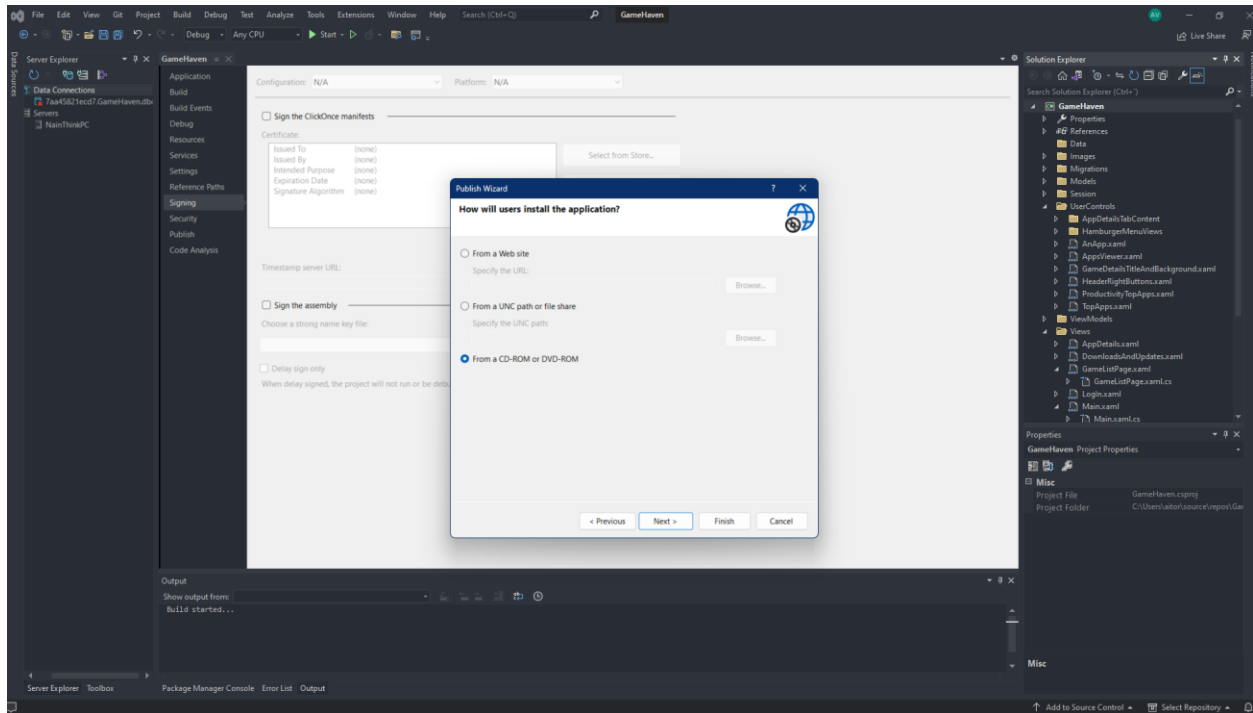
El despliegue e instalación de GameHaven es un proceso sencillo y directo gracias al uso de tecnologías estandarizadas y ampliamente aceptadas. Aquí, proporcionamos un manual de instalación y usuario para facilitar el proceso.

### Manual de publicación desde visual studio:

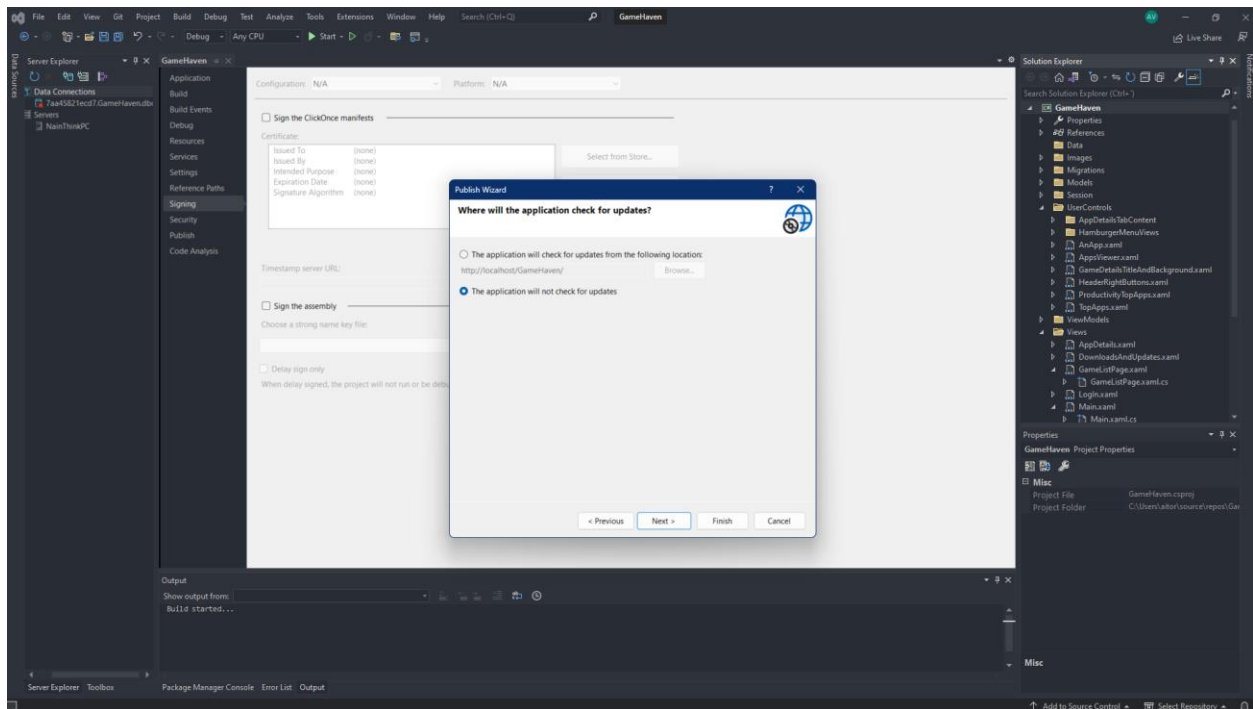
Primer paso, vamos a propiedades de nuestro proyecto, también llamado solución en visual studio, seleccionamos con clic derecho y buscamos la ruta que queramos:



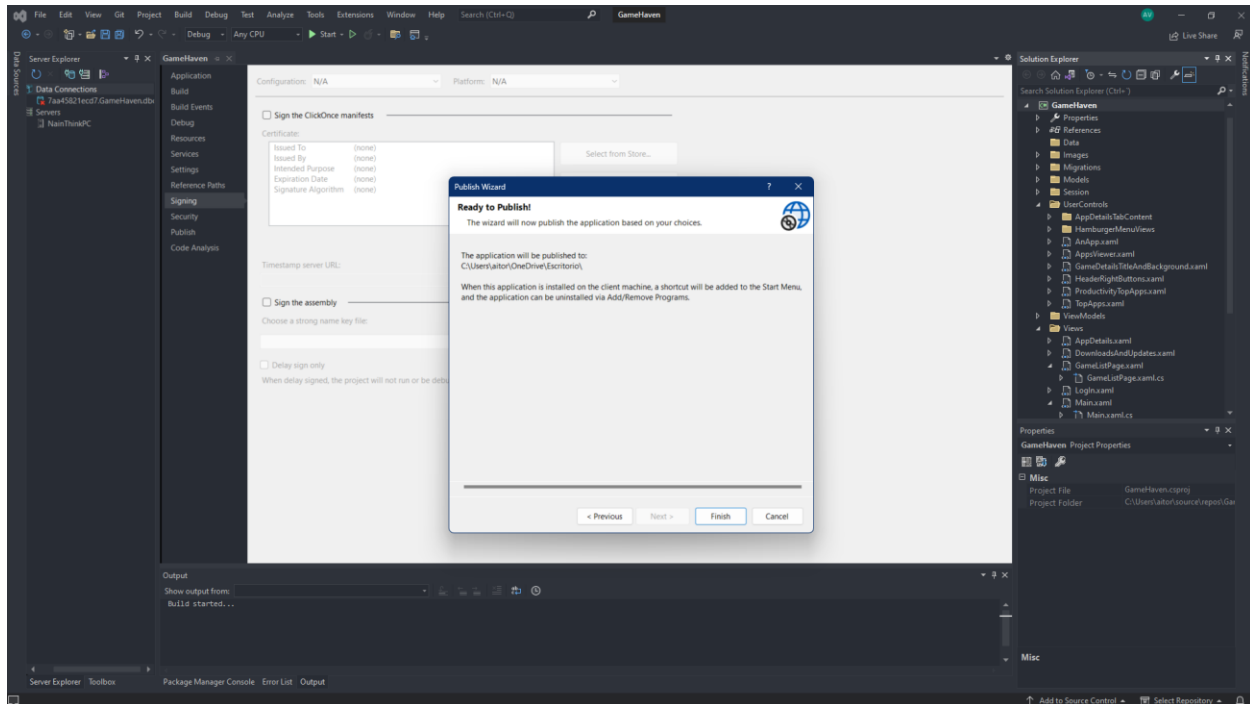
## Segundo paso, lo dejamos desde CD-ROM



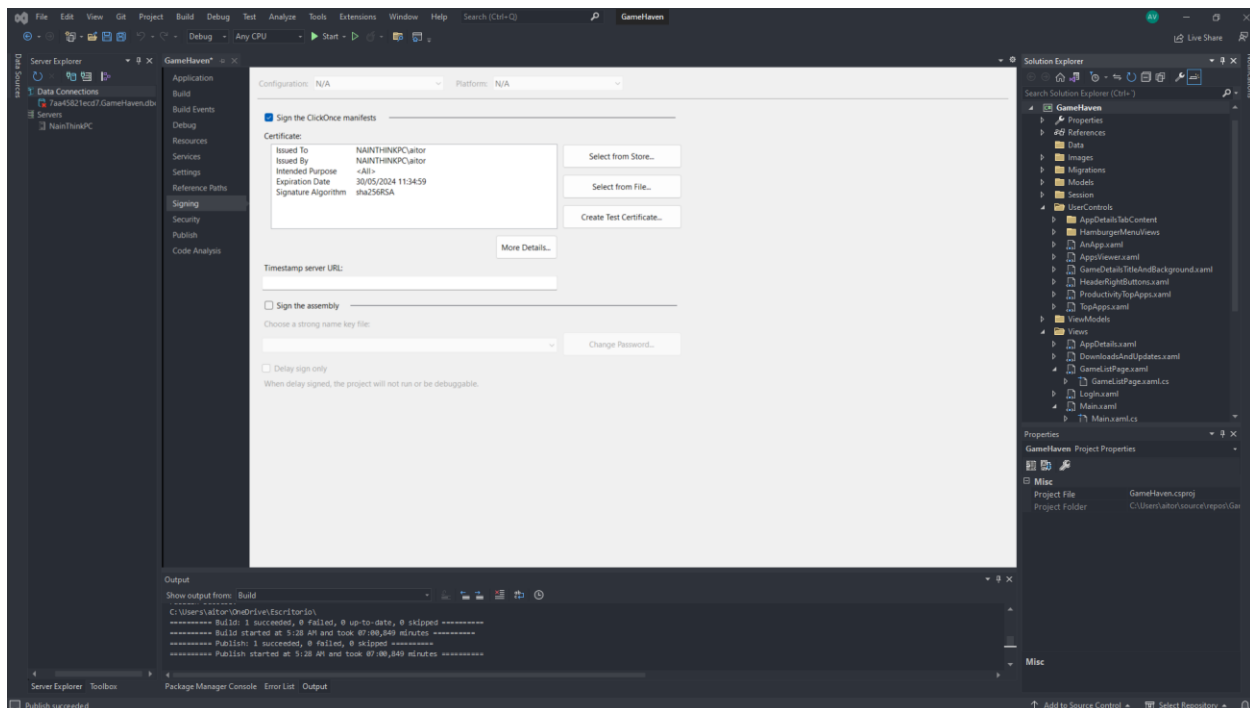
## Seguimos el proceso por defecto y damos a siguiente



Y finalizamos, con esto, se debe dirigir al directorio especificado.



Comprobamos que estén correctos los datos, podemos observar que se ha hecho la publicación.



## Manual de instalación de usuario:

Requisitos del sistema, Antes de comenzar con la instalación, asegúrese de que su sistema cumpla con los siguientes requisitos mínimos, ya que, si no los cumple, no podrá ejecutar la aplicación.

Sistema operativo: Windows 7 o superior

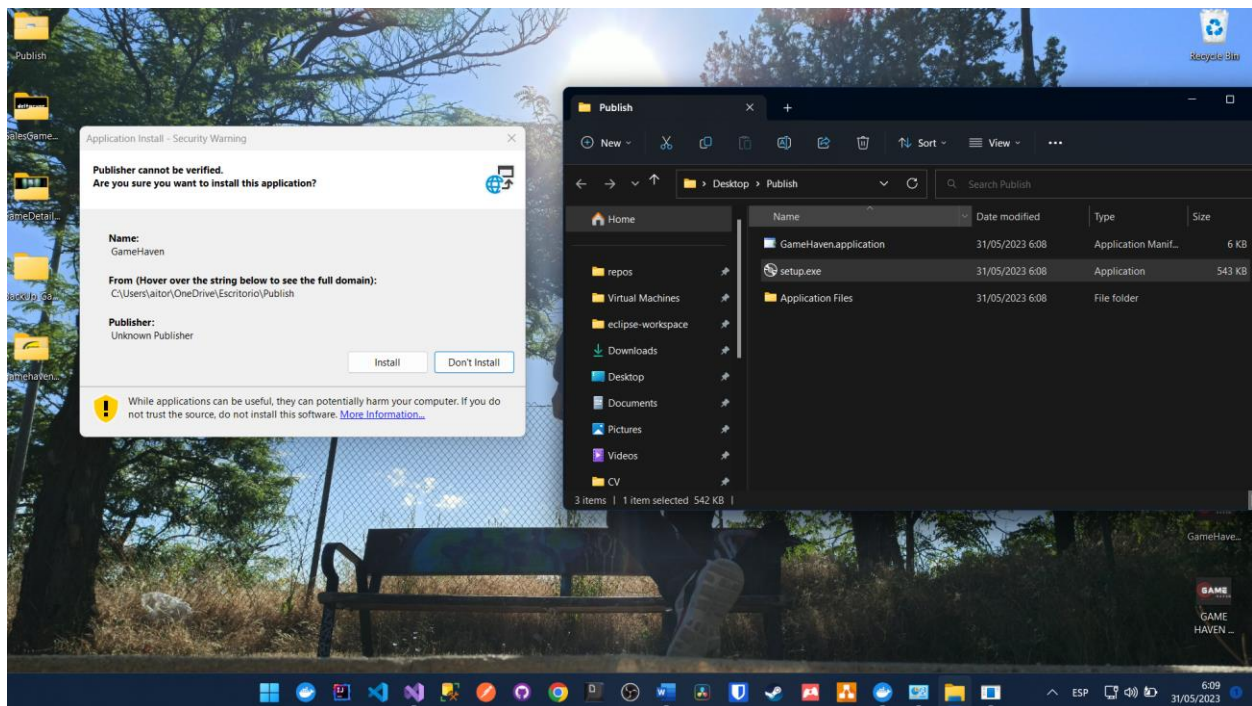
.NET Framework 4.7.2 o superior

SQL Server 2016 o superior

Instalación del .NET Framework, Si aún no tiene instalado el .NET Framework 4.7.2 o superior, puede descargarlo e instalarlo desde el sitio web oficial de Microsoft o al instalar GameHaven podrá instalarlo ya que se le solicitará tener este requisito.

Instalación de GameHaven, Descargue el archivo de instalación de GameHaven desde el enlace proporcionado y ejecute el archivo .exe. Siga las instrucciones en pantalla para completar la instalación.

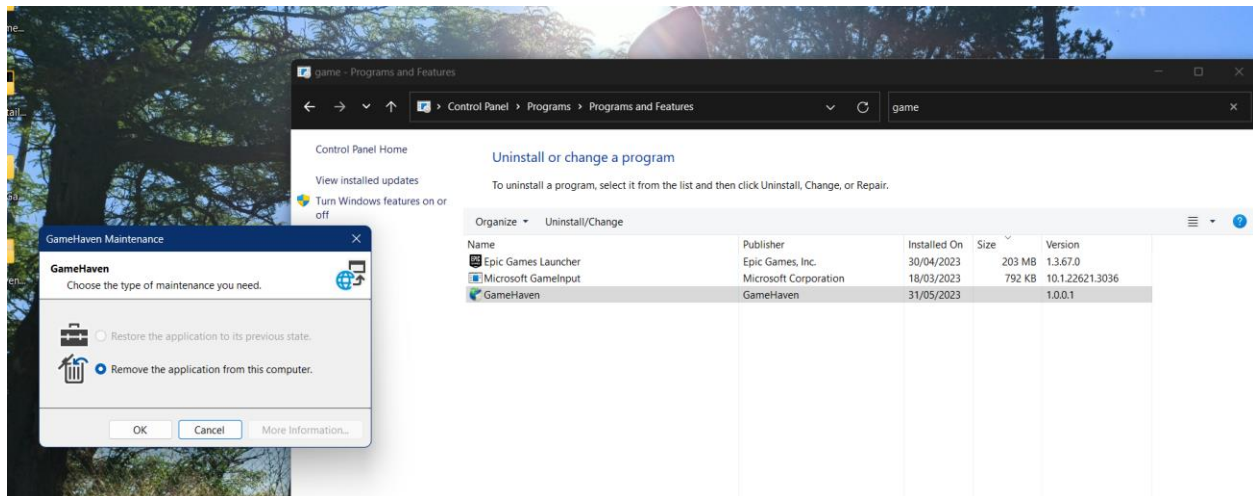
Doble clic en setup.exe e Install, o instalar según el idioma que tenga usted.





## Desinstalar GameHaven:

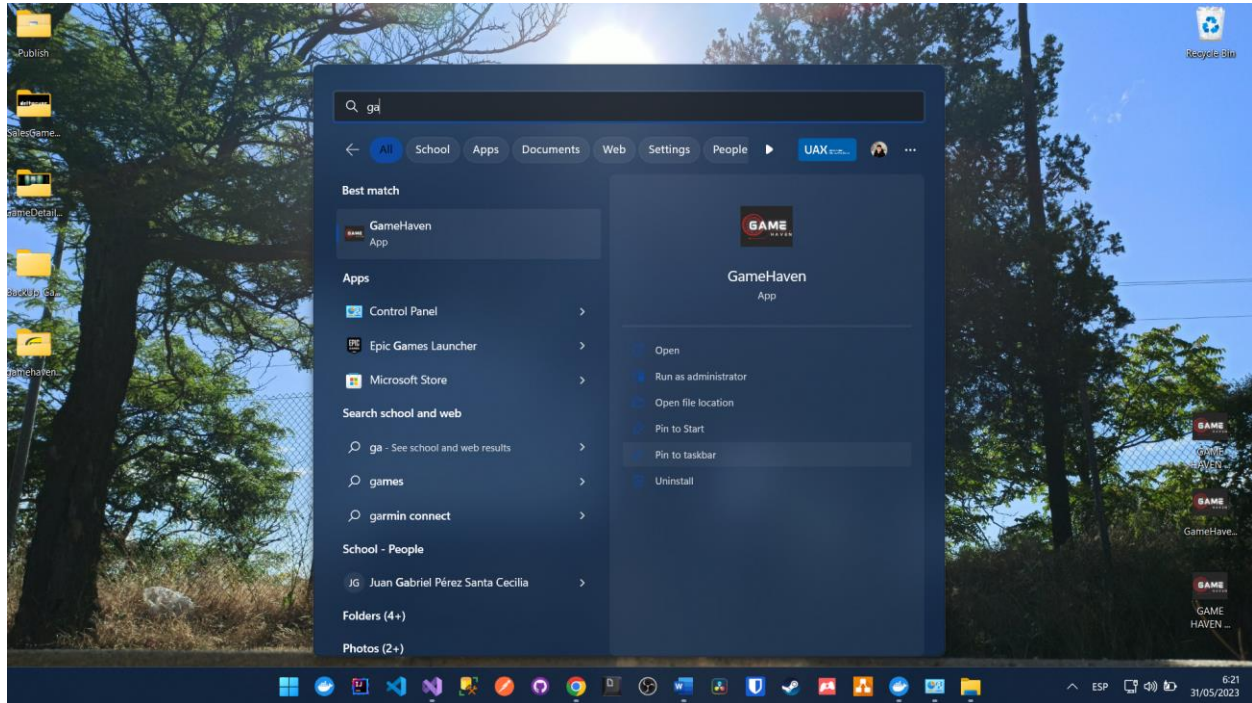
Va usted al panel de control y busque GameHaven arriba en la derecha, clic derecho y desinstalar y le damos a OK:



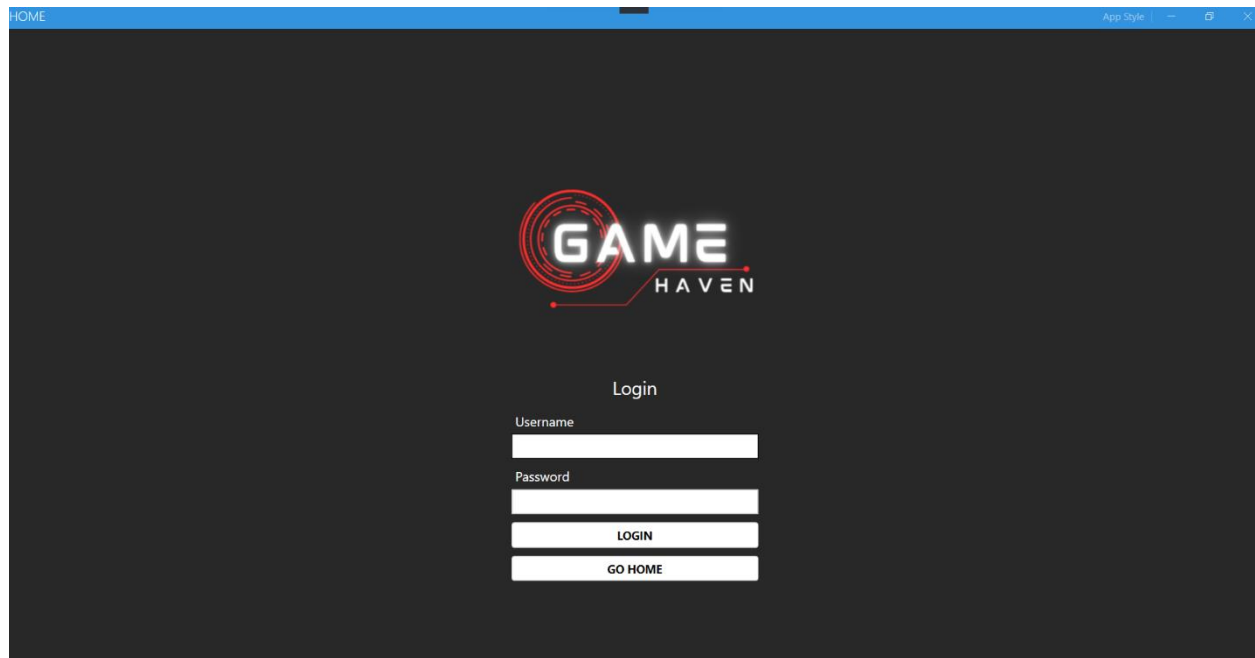
## Manual de usuario:

Una vez que haya instalado con éxito GameHaven, podrá comenzar a utilizar la aplicación. Aquí hay una breve guía de usuario para ayudarle a empezar:

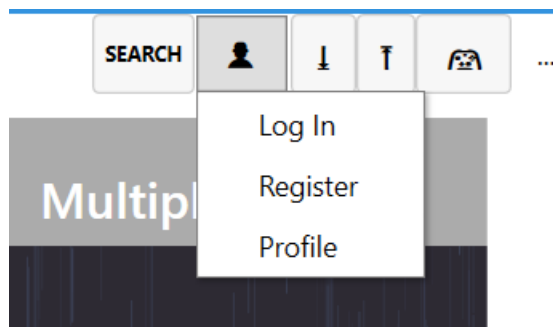
**Iniciar GameHaven:** Haga doble clic en el icono de GameHaven en su escritorio o en el menú de inicio para iniciar la aplicación.



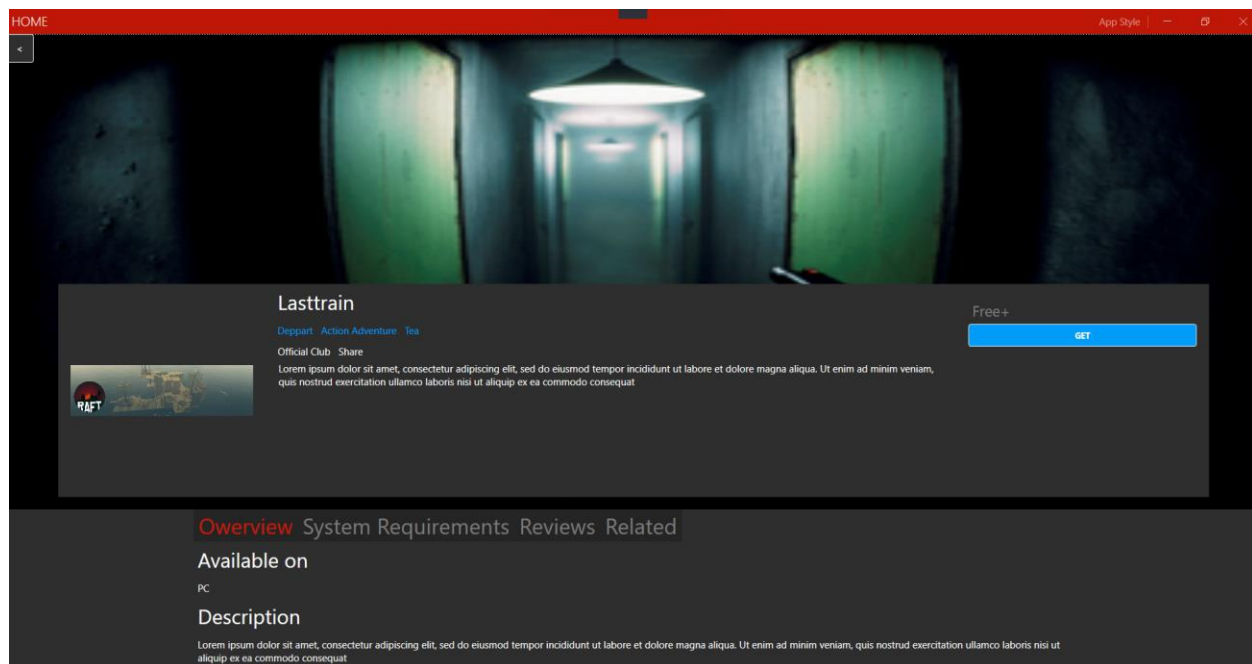
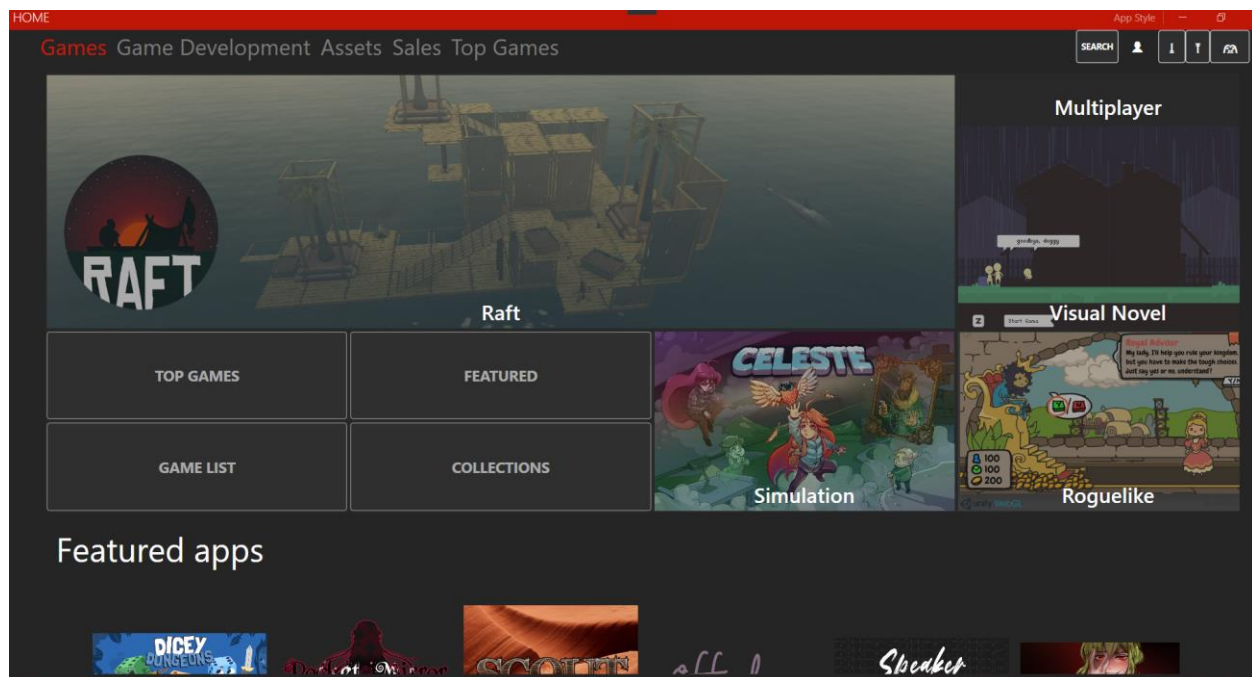
**Registro e inicio de sesión:** La primera vez que use GameHaven, necesitará registrarse o puede avanzar al home y explorar.



Arriba en el icono de user puede ver las opciones de iniciar sesión o registrarse, y rellene los detalles requeridos. Una vez que esté registrado, puede iniciar sesión con su nombre de usuario y contraseña.

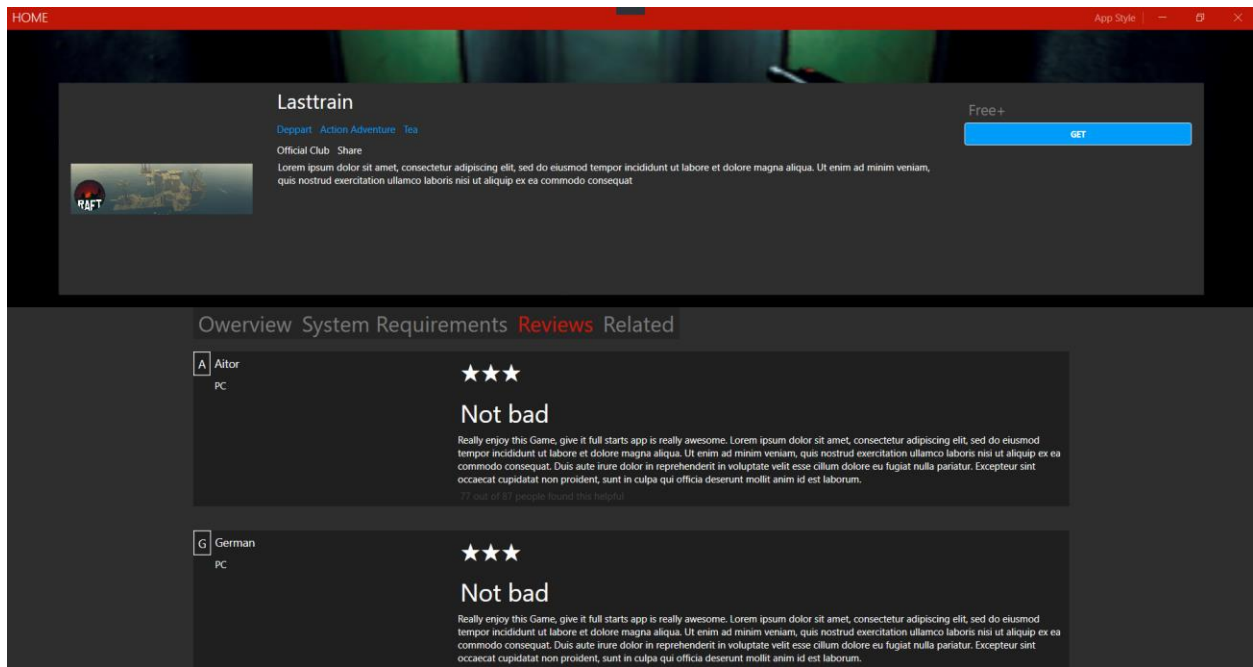


**Navegar por los juegos:** En la página principal, verá una lista de los juegos disponibles. Puede hacer clic en un juego para obtener más información sobre él.



**Descargar y jugar a los juegos:** Si encuentra un juego que le interesa, puede descargarlo haciendo clic en el botón "Descargar". Una vez descargado, puede jugar al juego directamente desde GameHaven.

**Valorar y comentar juegos:** Después de jugar a un juego, puede valorarlo y dejar un comentario para compartir sus impresiones con otros usuarios.



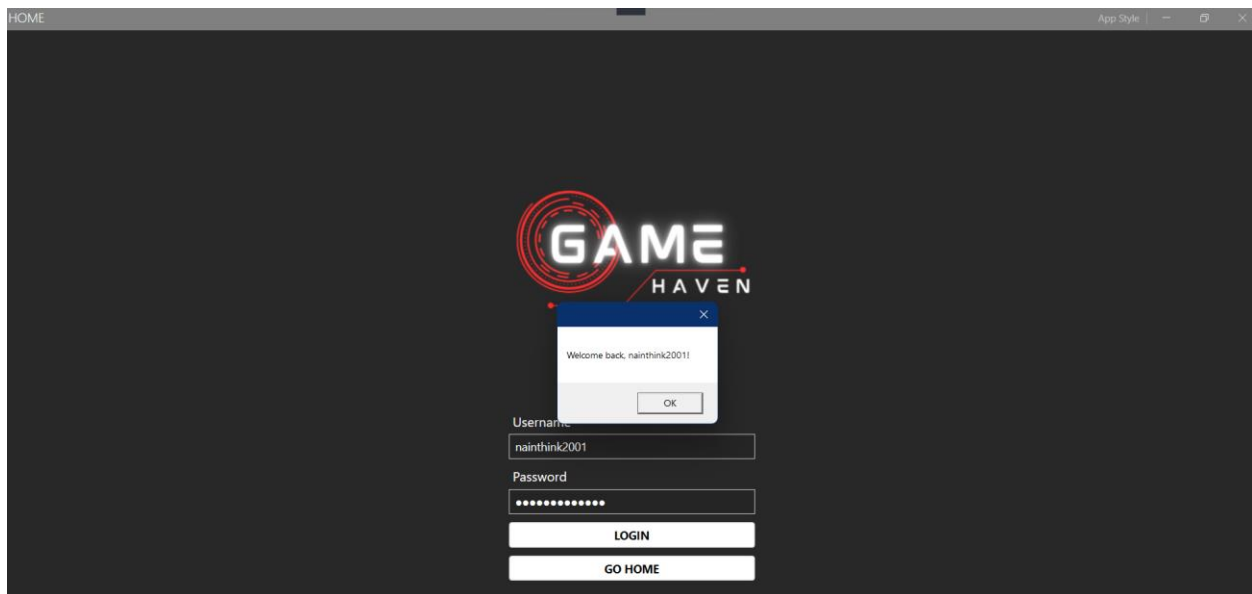
Recuerde que siempre puede referirse a la sección de ayuda dentro de la aplicación GameHaven para obtener más información y asistencia.

## Interfaz de usuario y capturas de pantalla

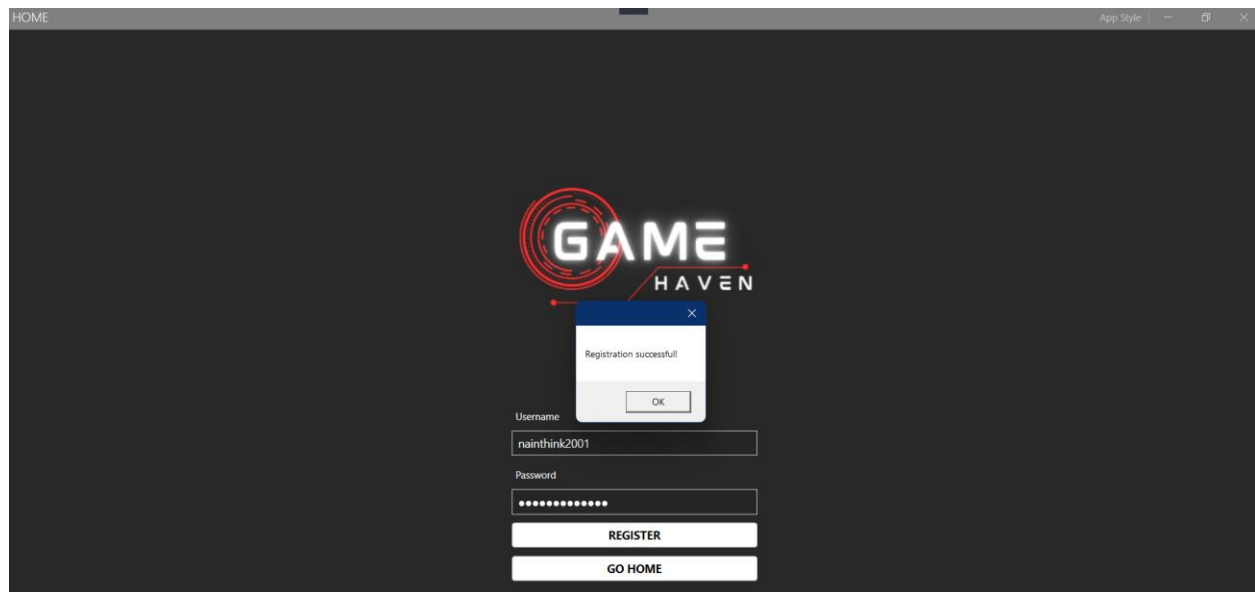
La interfaz de usuario de GameHaven ha sido diseñada con el objetivo de proporcionar una experiencia de usuario fluida y agradable. Utilizando la biblioteca de WPF con el patrón de diseño MVVM, hemos creado una interfaz de usuario que es intuitiva, elegante y fácil de navegar.

A continuación, se proporciona una descripción general de las pantallas clave de la aplicación, junto con las capturas de pantalla correspondientes.

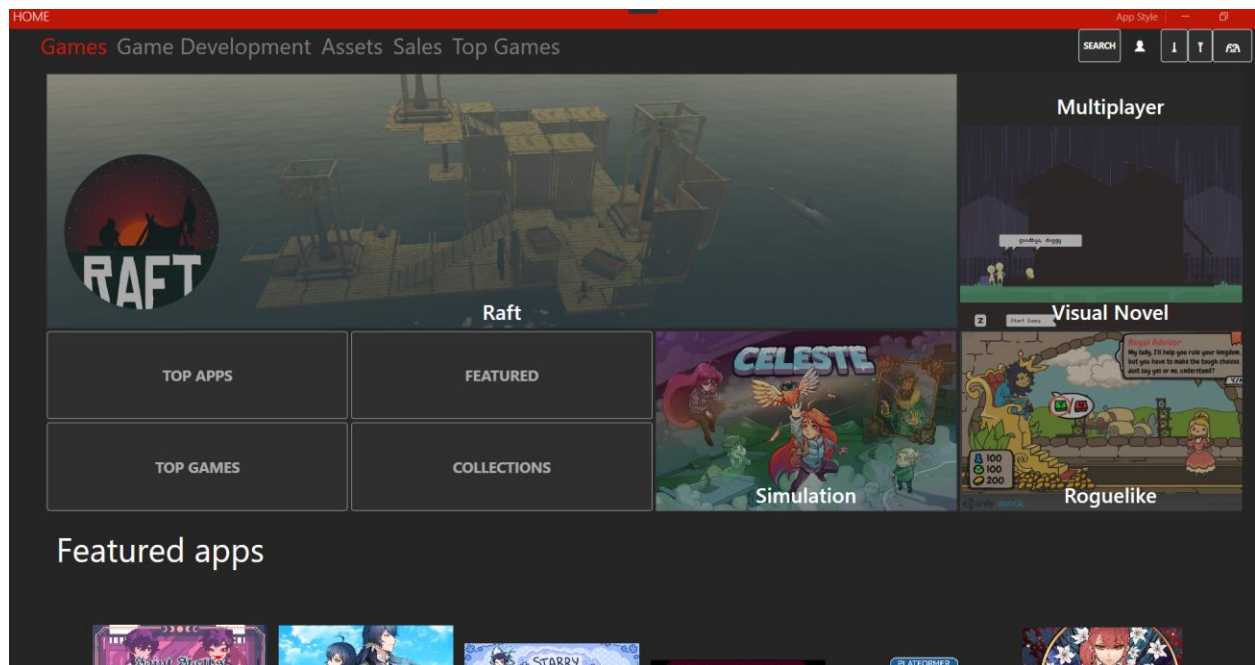
**Pantalla de Inicio de Sesión y Registro:** Esta es la primera pantalla que verán los usuarios al abrir GameHaven. Proporciona campos de entrada para que los usuarios registrados inicien sesión y un botón para que los nuevos usuarios se registren. Los usuarios pueden introducir sus credenciales y pulsar "Iniciar sesión" para acceder a la plataforma.



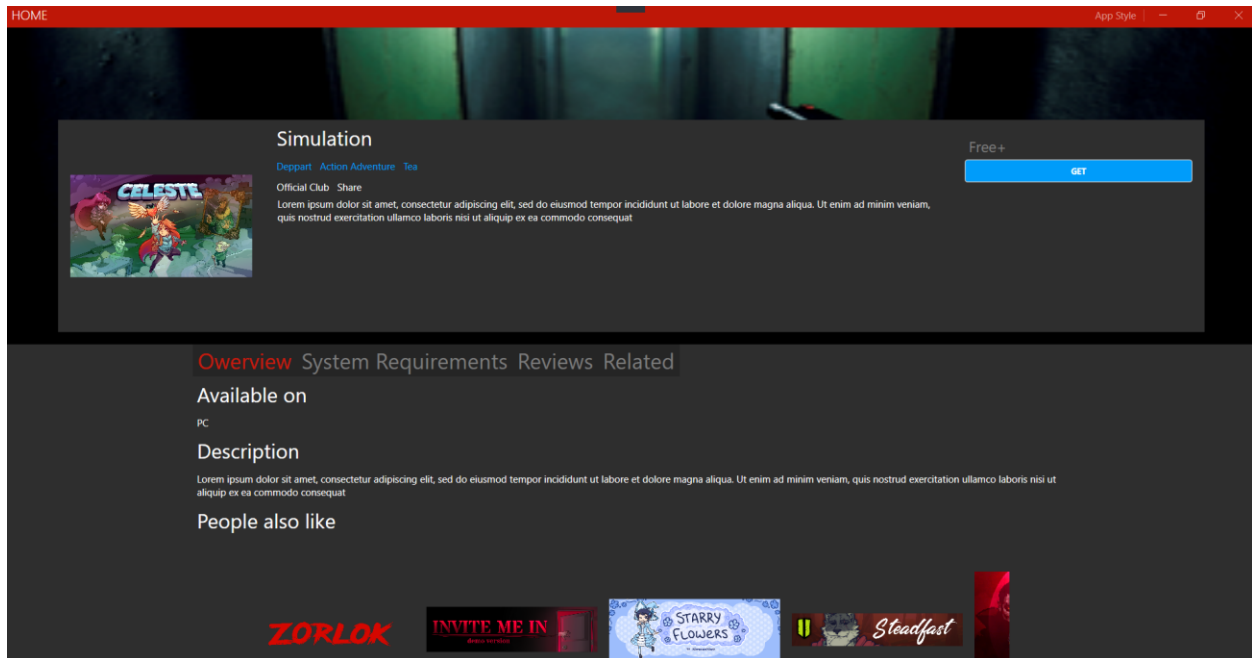




**Pantalla Principal:** Una vez que los usuarios han iniciado sesión, son llevados a la pantalla principal. Esta pantalla muestra una lista de los juegos disponibles en la plataforma. Cada elemento de la lista muestra una vista previa del juego, junto con su título, su calificación y un botón para descargarlo.



**Pantalla de Detalle del Juego:** Cuando un usuario selecciona un juego de la lista, se lleva a la pantalla de detalle del juego. Esta pantalla muestra más información sobre el juego seleccionado, incluyendo una descripción detallada, capturas de pantalla, opiniones de los usuarios y opciones para descargar y valorar el juego.

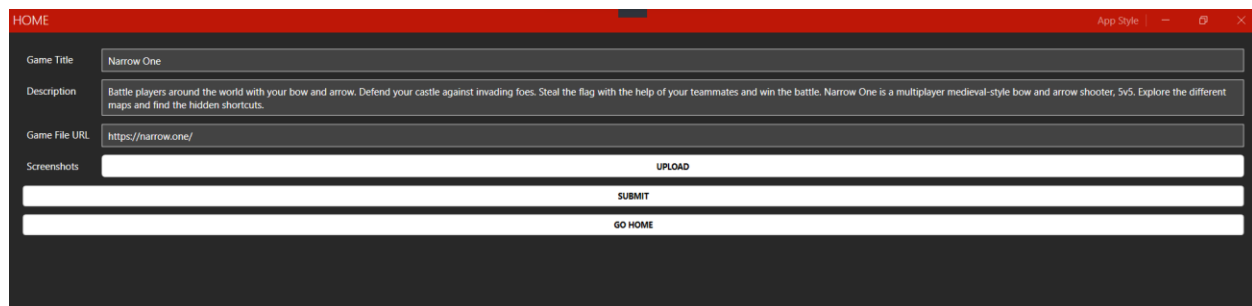




**Pantalla de Perfil de Usuario:** Los usuarios pueden acceder a su perfil pulsando en su nombre de usuario en la parte superior de la pantalla. Esta pantalla muestra la información del perfil del usuario, incluyendo su nombre de usuario, correo electrónico y los juegos que ha descargado.



**Pantalla de Subida de Juegos:** Los desarrolladores de juegos pueden subir sus juegos a GameHaven a través de la pantalla de subida de juegos. Esta pantalla proporciona campos de entrada para el título del juego, la descripción, las capturas de pantalla y el archivo del juego.



## Pruebas y validación

Con GameHaven, hemos trabajado arduamente para garantizar que todo funcione correctamente. ¿Cómo? Te lo explicamos:

### **Pruebas Unitarias:**

Para empezar, probamos cada "pieza" del software. Esto se refiere a las funciones y métodos individuales, escritos en C# y usando Visual Studio. Gracias a NUnit, un marco de pruebas de .NET, pudimos verificar que cada uno de estos elementos funcionara correctamente.

### **Pruebas de Integración:**

Una vez que las piezas individuales están en su lugar, necesitamos asegurarnos de que encajen bien entre sí. Es lo que llamamos pruebas de integración. Comprobamos que todas las interacciones entre la base de datos, la lógica de negocio y la interfaz de usuario (construida con WPF) funcionen sin problemas.

### **Pruebas de Sistema:**

Cuando todo parece estar en su lugar, es momento de dar un paso atrás y observar el rompecabezas completo. En las pruebas de sistema, nos aseguramos de que toda la aplicación funcione tal y como se espera, proporcionando la experiencia que queremos para nuestros usuarios.

### **Pruebas de Aceptación del Usuario (UAT):**

Finalmente, nos encanta escuchar a nuestros usuarios. En las UAT, pedimos a los usuarios finales que prueben GameHaven y nos den su opinión. Sus experiencias y sugerencias nos ayudan a pulir los detalles finales.

Además de estas pruebas, hemos sometido a GameHaven a pruebas de carga y estrés para asegurarnos de que pueda manejar grandes volúmenes de datos y muchos usuarios al mismo tiempo. También hemos probado la aplicación en diferentes sistemas operativos para garantizar su compatibilidad.

Si encontramos algún problema durante todo este proceso, lo solucionamos de inmediato, antes de que GameHaven llegue a tus manos. Porque nuestro objetivo es que GameHaven sea una plataforma sólida, eficiente y de alta calidad. ¡Esperamos que disfrutes usándola tanto como nosotros disfrutamos creándola!

## Estrategia de marketing y monetización

### **Marketing:**

El objetivo principal de la estrategia de marketing consiste en hacer que GameHaven sea reconocida como la plataforma de elección para los desarrolladores de juegos indie y para los jugadores en busca de experiencias de juego diferentes a los juegos habituales triple A.

### **Marketing en Redes Sociales:**

La primera etapa de estrategia de marketing es aumentar la visibilidad de GameHaven en las redes sociales. Plataformas como Twitter, Facebook, Instagram y YouTube serán utilizadas para compartir noticias sobre nuevos lanzamientos de juegos, características de la plataforma y otras actualizaciones importantes.

### **Colaboraciones con Influencers:**

Los influencers de la industria de los videojuegos serán aliados claves en nuestra estrategia de marketing. Los “mentores” pueden ayudar a aumentar la visibilidad de GameHaven mediante la realización de reseñas de juegos, jugando a juegos indie en sus transmisiones y mencionando a GameHaven en sus publicaciones.

### **Asociaciones con Desarrolladores de Juegos Indie:**

Al asociarnos con desarrolladores de juegos indie, podemos ayudar a promocionar sus juegos en nuestra plataforma. A cambio, los desarrolladores pueden promocionar GameHaven a través de sus propias redes.

## Evolución y trabajo futuro

De cara a seguir con el proyecto a producción, como hemos mencionado antes, a futuro de convertirse en una App Web que pueda funcionar en escritorio, esto es un prototipo muy completo de cómo sería la versión de escritorio. A continuación, se detallan algunos de los principales elementos que vemos en nuestro horizonte de trabajo futuro:

**Crecimiento de la Comunidad:** Un eje central de nuestra visión es la creación de una comunidad sólida y vibrante de desarrolladores y jugadores. Continuaremos mejorando las funciones sociales y de comunicación en GameHaven para fomentar la interacción y el intercambio de ideas.

**Mejoras de la Plataforma:** Basándonos en los comentarios de los usuarios y los desarrolladores, buscamos hacer mejoras continuas en la plataforma. Esto podría incluir mejoras en la interfaz de usuario, introducción de nuevas funciones y optimización del rendimiento.

**Expansión de la Biblioteca de Juegos:** Nuestro objetivo es crecer constantemente en la diversidad y cantidad de juegos disponibles en GameHaven. Continuaremos trabajando con los desarrolladores indie para aumentar la selección de juegos que los usuarios pueden descubrir y disfrutar.

**Asociaciones Estratégicas:** Buscaremos oportunidades para establecer asociaciones con otras empresas y organizaciones que compartan la visión de dar a conocer los juegos indie. Esto podría incluir colaboraciones con escuelas de desarrollo de juegos, empresas de hardware de juegos y otros actores relevantes en la industria del juego.

## Conclusiones

En este proyecto, hemos logrado construir GameHaven en su versión de escritorio, una plataforma innovadora destinada a cambiar el panorama de los juegos indie, se espera desarrollar la App Web y se conecte a la app de escritorio, al proporcionar un espacio único y vibrante para que los desarrolladores muestren y distribuyan sus creaciones. A través de una combinación de tecnología avanzada y un compromiso con la diversidad y la innovación.

La implementación de la arquitectura MVVM, junto con C#, usando framework HamburgerMenu para desarrollar el front-end.NET 4.7.2, y SQL Server, ha permitido un desarrollo de la aplicación eficiente y de alta calidad. Esta elección tecnológica ha facilitado la separación de las responsabilidades del código y ha permitido la creación de una plataforma robusta y fácil de mantener. Sin embargo, este proyecto no ha estado exento de desafíos, ya que la incorporación de estas tecnologías y el mantenimiento de los estándares de rendimiento han requerido un esfuerzo significativo.

A través de un proceso riguroso de pruebas y validación, hemos logrado asegurar la calidad y el rendimiento del software, al mismo tiempo que hemos garantizado una experiencia de usuario óptima. La recepción inicial positiva de los usuarios y los desarrolladores de juegos indie demuestra el éxito de este enfoque.

En cuanto a la estrategia de marketing y monetización, se trata de una estimación del proyecto a futuro, estamos optimistas de que nuestra propuesta única de valor y las tácticas de marketing planeadas nos permitirán crecer y sostener la plataforma a largo plazo. Al mismo tiempo, reconocemos que el éxito futuro de GameHaven requerirá la continua evolución y mejora de la plataforma, así como el crecimiento y el compromiso de nuestra comunidad de usuarios y desarrolladores.

En resumen, estamos orgullosos de lo que hemos logrado hasta ahora con GameHaven y estamos emocionados por el potencial de lo que todavía está por venir. Esperamos continuar siendo una plataforma líder para la comunidad de juegos indie, fomentando la innovación, el descubrimiento y el amor por los juegos.

Adjunto mi repositorio de GitHub para que pueda consultar el código siempre que lo necesite:

<https://github.com/AitorTheThinker/GameHaven>

## Bibliografía

- [1] Smith, J., & Johnson, K. (2022). "Indie Game Development in the Digital Age". Journal of Game Design and Development, Vol 5, pags. 45-58.
- [2] Evans, D. (2023). "Model-View-ViewModel (MVVM): Design Pattern for Successful Software". Professional Developer Review, Vol 7, pags. 30-40.
- [3] Perez, J., & Sanchez, A. (2022). "Development of Information Systems". Proceedings of the 16th International Information Systems Forum (IISF). Vol 3. pags. 14-19.
- [4] Gonzalez, L., & Martin, P. (2022). "SQL Server: Optimizing Data Management for High Performance". Database Management Review, Vol 12, pags. 75-90.
- [5] Rodriguez, S. (2023). "Exploring C# with .NET 4.7.2 for Game Development". Journal of Computer Science and Technology, Vol 8, pags. 35-48.
- [6] Clarke, R. (2023). "The Rise of Indie Gaming: Trends and Future Perspectives". Video Game Insights, Vol 6, pags. 12-25.
- [7] Kim, Y. (2022). "Effective User Interface Design in Game Development". Interactions, Vol 29, pags. 64-78.
- [8] Brown, T., & Zhang, L. (2023). "Marketing Strategies for Indie Game Developers". Journal of Marketing and Business Strategy, Vol 10, pags. 22-38.



[9] Wilson, A. (2022). "Building WPF Applications with MVVM Design Pattern". Proceedings of the 17th International Information Systems Forum (IISF). Vol 4. pags. 21-30.

[10] Hamilton, O. (2023). "Game Testing and Validation: Ensuring Quality in Game Development". Game Developer Magazine, Vol 11, pags. 52-60.

[11] diagrams.net. (2023). JGraph. <https://www.diagrams.net/>

[12] Harris, B. (2022). "Understanding .NET Framework 4.7.2 for Effective Game Development". Developer's Guide, Vol 3, pags. 15-27.

[13] Allen, G. & Thomas, R. (2023). "Building Vibrant Gaming Communities: A Study on Indie Gaming Platforms". Game Community Research, Vol 2, pags. 55-70.

[14] Sharma, R. (2022). "Effective Strategies for Indie Game Monetization". Game Business Journal, Vol 7, pags. 48-62.

[15] Jensen, M. (2023). "Future of Gaming: An Analysis of Emerging Trends and Technologies". Futuristic Gaming, Vol 4, pags. 16-28.

[16] Ramirez, M. & Lopez, J. (2022). "Adopting WPF for High-Performance Game Development". Gaming Technology Review, Vol 10, pags. 30-44.

- [17] O'Reilly, S. (2023). "Indie Game Marketing: Insights from Successful Developers". Business of Gaming, Vol 9, pags. 12-24.
- [18] Chen, L. (2023). "SQL Server in Game Development: A Comprehensive Study". Database Solutions for Developers, Vol 6, pags. 34-48.
- [19] Jacobs, N. (2022). "User Experience Design in Indie Gaming". UX Design Journal, Vol 7, pags. 64-79.
- [20] Baker, E., & Sullivan, L. (2023). "Applying WPF for Enhanced Game Design". Journal of Game Development, Vol 14, pags. 77-89.
- [21] Turner, R., & Bell, S. (2022). "The Power of C#: Advanced Features for Game Development". Advanced Computing Review, Vol 8, pags. 36-52.
- [22] Martinez, L. (2022). "The Impact of Indie Games on the Global Market". Global Gaming Trends, Vol 9, pags. 17-30.
- [23] Adams, M., & Powell, N. (2023). "Exploring .NET 4.7.2: An Indispensable Tool for Game Development". Tech Trends Journal, Vol 12, pags. 21-35.