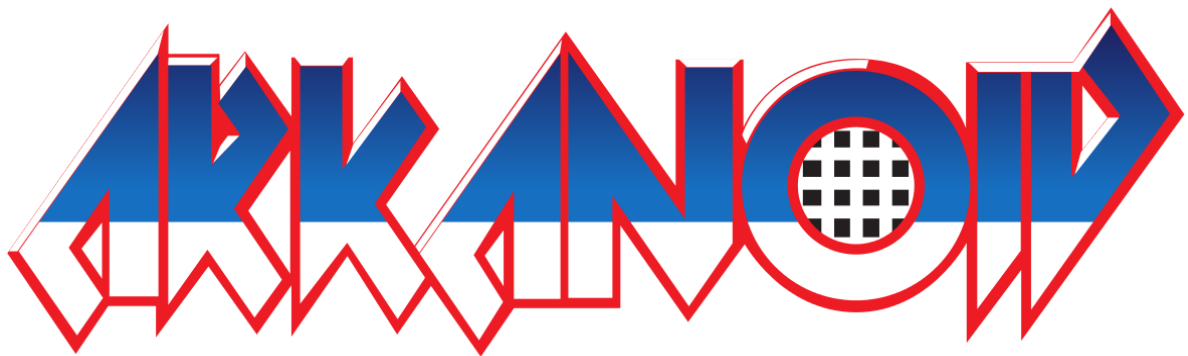


Manual Técnico

Arkanoïd



Realizado por:

Rodrigo Antonio Ungo Muñoz, 00075419

Mario Amadeo Urquiza Funes, 00223916

Contenido

1. Aspectos generales.....	3
1.1 Objetivos del documento.....	3
1.2 Descripción general.....	3
1.3 Software utilizado.....	3
2. Diagramas del programa.....	4
2.1 UML de clases.....	4
2.2 Diagrama Entidad Relación Extendido.....	6
2.3 Diagrama Relacional.....	7
3. Conceptos técnicos y distintos tipos de error....	8
3.1 Interfaz gráfica.....	8
3.2 Manejo de clases en modelo.....	9
3.3 Plataforma Base.....	9
4. Nomenclaturas.....	10
4.1 Abreviaturas.....	10
5. Eventos y Excepciones.....	11
5.1 Excepciones.....	11

Aspectos generales

Objetivos del documento

El objetivo de este documento pretende orientar y explicar el diseño del software creado, explicando las herramientas utilizadas para su creación.

Descripción general

El programa es un juego arcade que consiste en destruir los bloques que aparecen en el juego para obtener puntos, se gana cuando se destruyen todos los bloques. Para la creación del software se hizo uso del modelo – vista – controlador, sus siglas MVC.

Software utilizado

Para la creación del programa se utilizó **JetBrains Rider 2019**, en conjunto con **PostgreSQL 12** para la implementación de la base de datos. Complementos adicionales para Rider utilizados han sido **Npgsql** para la conexión del programa con la base de datos.

Diagramas del programa

UML de clases

El diseño arquitectónico del código está basado en el diagrama de clases siguiente (ver siguiente página):

Se deja un hipervínculo para una vista más detallada:

<https://drive.google.com/file/d/1lFDlGNBwy2ccAtxqQxYClej50lZnKSvo/view?usp=sharing>

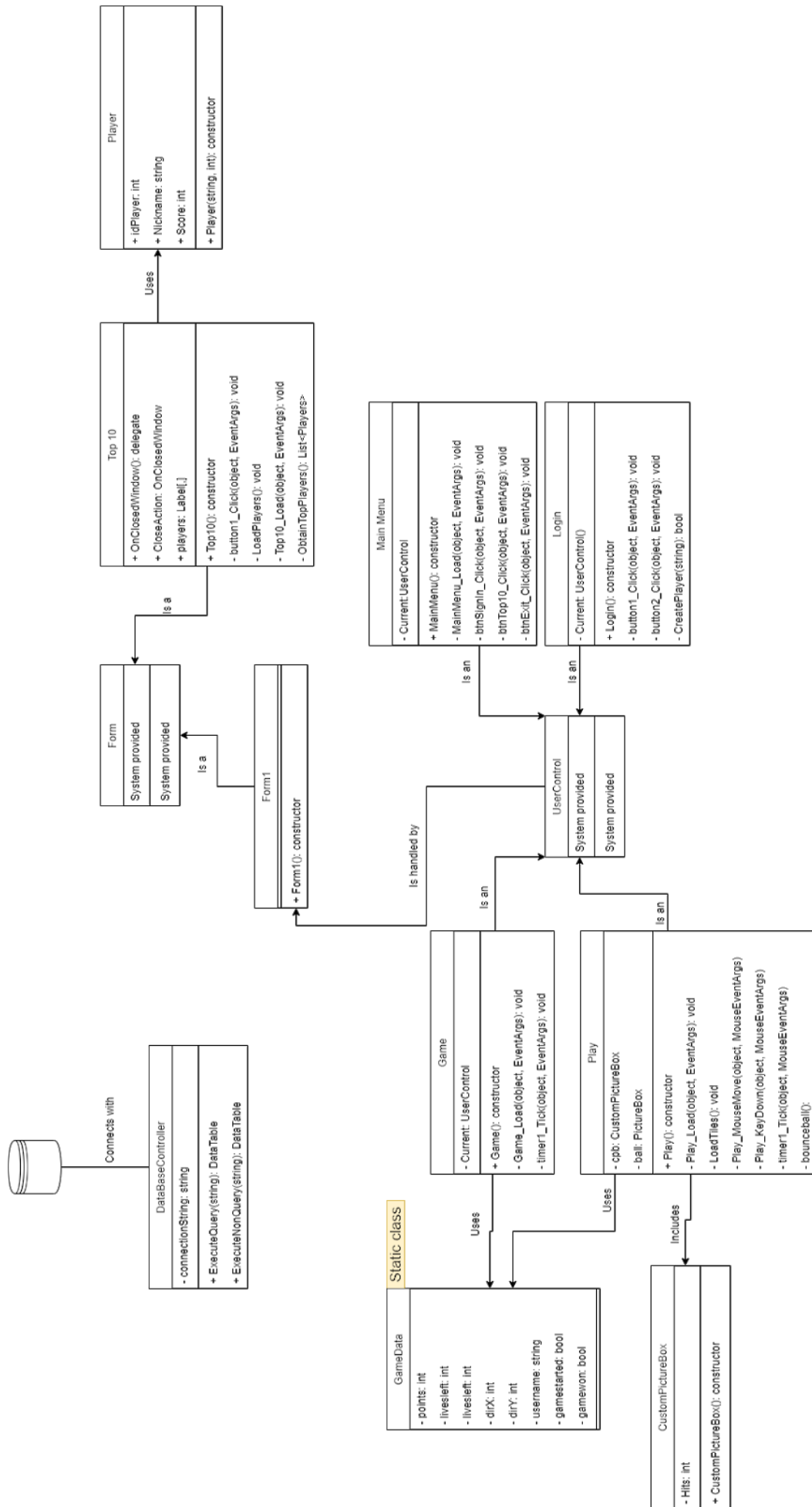


Diagrama Entidad Relación Extendido

El diseño de la base de datos está basado en el siguiente diagrama entidad relación.

Se deja un hipervínculo para una vista más detallada:

<https://drive.google.com/file/d/1KKAAsmDT5rn-3cUJMdSTeCmK6xmITfjK/view?usp=sharing>

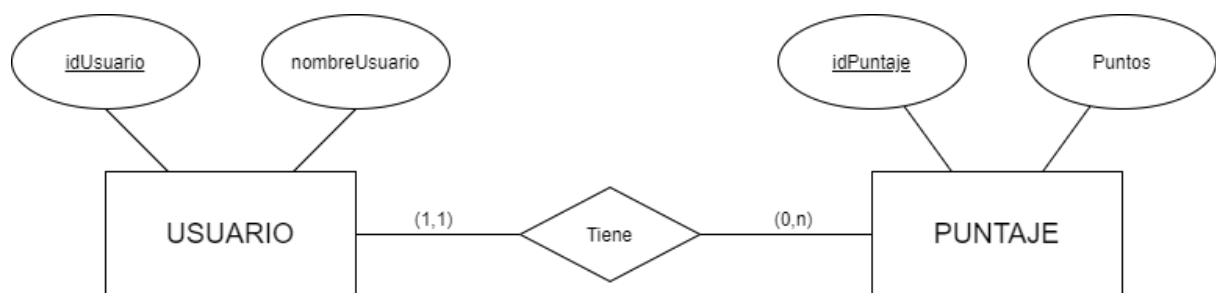
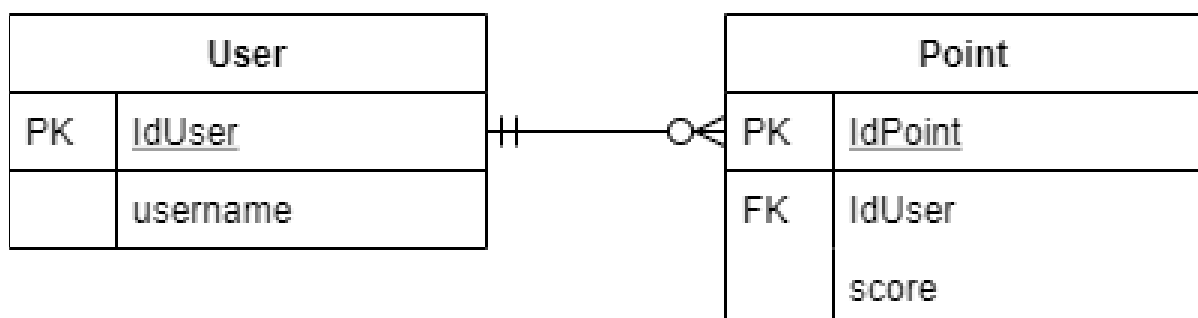


Diagrama Relacional

El diseño de las tablas de la base de datos está basado en el siguiente diagrama relacional.

Se deja un hipervínculo para una vista más detallada:

https://drive.google.com/file/d/1gLgjiBuo2_UQeXclpTTaB3tylfDr3tut/view?usp=sharing



Conceptos técnicos y distintos tipos de error

Interfaz gráfica

La interfaz gráfica del programa consiste en un único formulario o ventana, se navega a través del programa con distintos botones distribuidos en varios controles de usuario o UserControl en inglés, cada user control posee botones para cambiar a otro user control.

Cada user control ejecuta un acción propia y específica del programa, los controles de usuario existentes son:

MainMenu.cs	Game.cs
Login.cs	Play.cs

El programa también posee un formulario o ventana extra para el listado de los 10 mejores puntajes llamado **Top10.cs**. Se utilizó una ventana debido a que se consideró más apropiado y fácil de implementar.

Manejo de clases en modelo

Para manejar la parte fundamental del modelo del programa, se cuenta con las siguientes clases:

CustomPictureBox.cs	GameData.cs
DataBaseController.cs	Player.cs

Plataforma Base

Sistema operativo	Multiplataforma
Tecnologías	JetBrains Rider
Lenguaje	C#
Gestor de DB	PostgreSQL

Nomenclaturas

Abreviaturas

Para los elementos del entorno gráfico se implementa la siguiente normativa de nombramiento:

<Abreviatura de tipo>_descripción

Las abreviaturas se presentan a continuación:

Label	lbl
TextBox	tex
PictureBox	ptb
Button	btn

Eventos y Excepciones

Excepciones

Las excepciones, debido a los nombres asignados, son autoexplicativas e intuitivas.

Constan de un constructor que recibe un string con el mensaje de error o advertencia. Se cuentan con las siguientes:

GameOverException.cs

OutOfBoundsException.cs

EmptyFieldException.cs

ExceededMaxCharactersException.cs

GameWinException.cs