

MANUAL TÉCNICO

Arkanoid



Elaborado por

Josué Humberto Montecinos Torres.....	00103013
Stanley Alexander Humberto Bonilla.....	00016717
Zair Alexander Ortíz Flores.....	00355519
Mario Orlando Moisa Candray.....	00135115



Contenido

Manual técnico de Arkanoid.....	1
Aspectos Generales.....	3
• Objetivo de documento.....	3
• Descripción general	3
• Software utilizado	3
Modelo Utilizado.....	4
• Diagrama UML de clases.....	4
• Diagrama UML de casos.....	5
• Diagrama entidad relación.....	5
• Diagrama relación de la base de datos.....	5
Conceptos Técnicos.....	6
• Interfaz gráfica	7
• Manejo de clase modelo	7
• Plataforma bases	7
• Nomenclatura.....	7
Eventos y Excepciones.....	8
• Eventos.....	8
• Excepciones.....	8



Aspectos Generales

Objetivos del documento

El objetivo de este documento pretende orientar y explicar el diseño del software, explicando las herramientas empleadas para la creación de este, y de futuras versiones.

Descripción general

Para la creación de este del software hizo uso del MCV (Modelo-Controlador-Vista). El programa presenta su principal funcionalidad en el manejo de implementación del juego Arkanoid en el cual consiste en la destrucción de bloques a través de una pelota la cual rebota sobre una plataforma, y el juego termina; cuando se destruyen todos los bloques o se pierde todas las vidas. Al final te muestra el marcador de cuantos puntos se ha adquirido o si pierdes te da la opción de volver a jugar.

Software Utilizado

Para la creación de la aplicación se utilizó JetBrains Rider 2019.3.4 con el conjunto de PgAdmin 4.20 para la creación de la Base de Datos y a través de componentes Nuget de Rider para la conexión de la base de datos a través Npgsql.



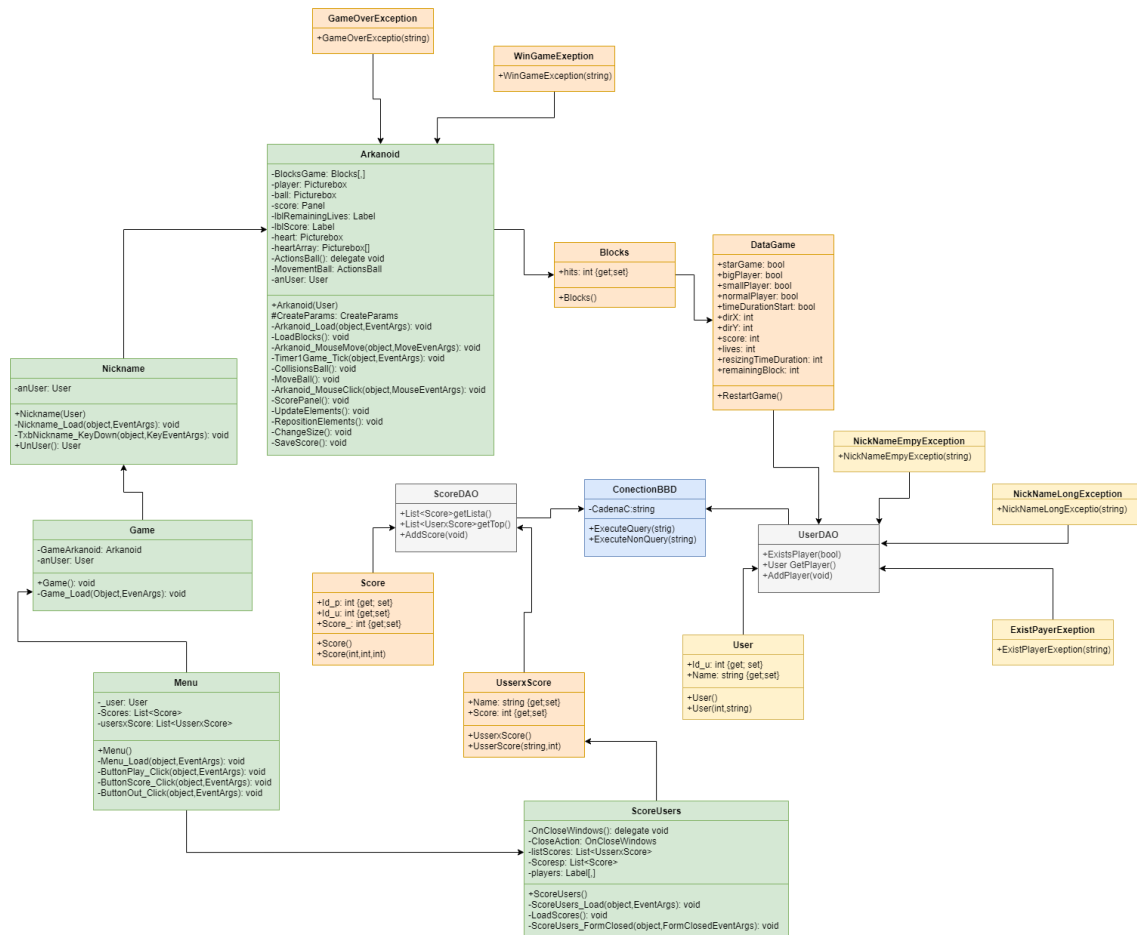
Modelo Utilizado

Diagrama UML de clases

En él muestra el funcionamiento del programa a través de este diagrama.

En el siguiente apartado se podrá tener una conexión al apartado de funcionamiento del programa.

<https://app.diagrams.net/#G15hGRZrEDD3uzCtvoNZUFohfsLpCLXgA3>



Descripción del Funcionamiento del Programa

En principio se tendrá en cuenta que en diagrama de clases se tendrá en cuenta el funcionamiento de un programa en la cual un usuario a través de la clase Nickname que a través de Game, que a su vez se tiene acceso a través de Menu, a clase Arkanoïd la cual tendrá el funcionamiento de todos los aspectos del juego de la cual esta heredará la clase Blocks la cual se vincula a la clase DataGame.

La clase DataGame está vinculada a UserDAO que se tendrá como precedente heredar, la cual actuará como controlador de la función User la cual se podrá acceder a la base de datos de donde tendrá la conexión query de los usuarios a través de la función User la cual se registrará el marcador a través de la conexión de la clase controlador ScoreDAO la cual se encargará del funcionamiento de la clase Score la cual se encarga del funcionamiento del marcador del juego y UserScore que heredará la cual tendrá la información de puntos de los Usuarios a Través de un UserControl ScoreUsers en la cual se anidaran las puntuaciones de los jugadores.

La cual también hereda a la función Menú los lineamientos de la función Game la cual actúa hereda de la función Game la cual está vinculada a la función NickName que como se menciona anteriormente esta estará vinculada al funcionamiento del Arkanoïd.

Diagrama UML de casos de uso

En este apartado se nos muestra un esquema del funcionamiento del uso que un jugador podrá realizar al acceder al juego.

Aquí se colocará un enlace de dónde provino el esquema

<https://app.diagrams.net/#G12OHf4jt7antz4dNqDvySgm6Zq33MwhcL>

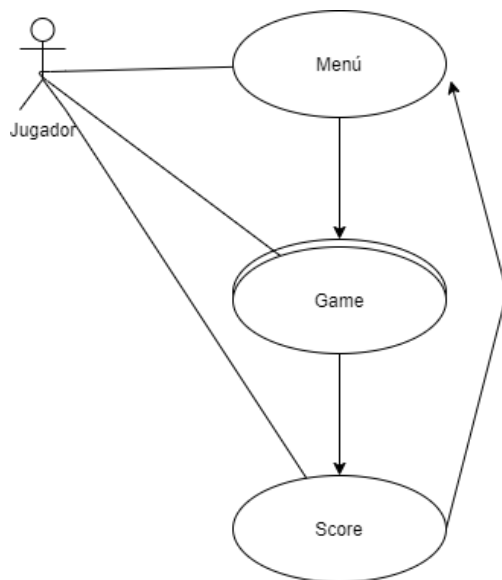


Diagrama entidad relación de la base de datos

Enlace a la zona de edición de esquema planteado

<https://app.diagrams.net/#G1mYqjVwEu5VdLJKA2gV5JjWFMgvF1lwvV>

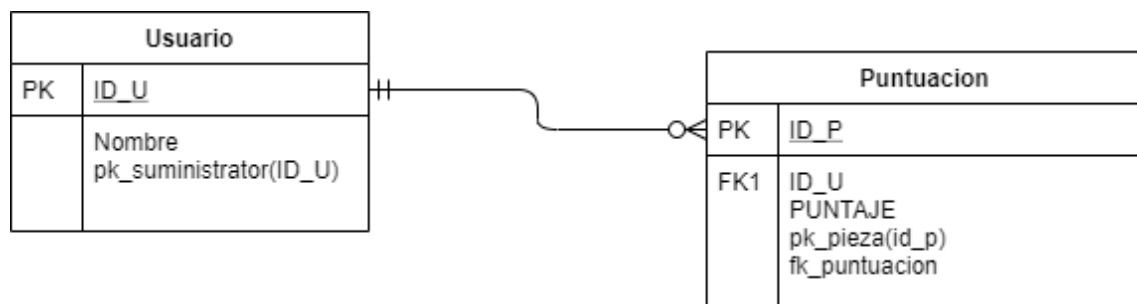


Diagrama Relacional de la Base de Datos

En este diagrama se muestra la relación que hay en la base las cuales en cierto punto estarán relacionadas en el funcionamiento del marcador y del almacenaje de información del usuario.

<https://app.diagrams.net/#G1-hfDv0ApsdCipwCUp6mc7OMgxP3hr2nO>



Conceptos Técnicos

Implementación de la Interfaz Gráfica

En lo que se refiere a la implementación de la interfaz gráfica, se refiere a la implementación de una serie de herramientas las cuales fueron la implementación de sistemas Form y Usercontrols los cuales ayudaron para el funcionamiento coherente del juego. El nombre de las interfaces fue.

Arkanoir.cs

Menu.cs

Nickname.cs

Game.cs

ScoreUssers.cs

Manejo de la clase modelo

Para el manejo de las clases en modelo se encuentran las siguientes clases:

Blocks.cs

Score.cs



User.cs

Users.cs

UsersxScore.cs

Plataforma Base

Sistema Operativo	Multiplataforma
Tecnología	Jetbrains Rider 2019.3.4
Lenguaje	C#
Gestor de BDD	pgAdmin 4

Nomenclaturas

En lo que se refiere a las nomenclaturas, se refiere a las propiedades de las diferentes herramientas


Label	Lb
Picturebox	pic
Buttom	bt
Load	ld
DateTimerPicket	dtp
Panel	pn

Eventos y Excepciones

Eventos

Para poder pasar una acción reflejada en un control de usuario al formulario principal fue necesario en la cual para el funcionamiento se emplearon.

1. MoveEvenArgs.cs



Este evento se utilizó para la animación de los picturebox para en movimiento de la pelota y la plataforma y la animación de ruptura de los bloques

2. **MouseEventArgs.cs**

Este evento se utilizó para la para los eventos de ratón que corresponde a los funcionamientos que tendrá el mouse en la aplicación como mover la plataforma.

3. **KeyEventArgs.cs**

Se refiere al evento que se tendrá al presionar botones de control este caso los de control defunciones o dar un enter.

4. **FormClosedEventArgs.cs**

Como lo indica el nombre es un evento cuya función es cerrar el forms de la aplicación al momento de finalización.

Excepciones

Las excepciones se refieren a una cadena tipo string que se refiere a un error

1. **NickNameEmptyException.cs**

2. **NickNameLongException.cs**

3. **ExistPayerExeption.cs**

4. **WinGameExeption.cs**

5. **GameOverException.cs**