**UNIVERSIDAD POLITÉCNICA SALESIANA**

Carrera de Computación

**PROYECTO DE VINCULACIÓN EMBLEMÁTICO E INTERDISCIPLINAR**   
“Aplicación de herramientas tecnológicas para la innovación en unidades educativas”

**TIC-InnovaEdu**



**Flappy Math: Aprender Multiplicación Jugando**

Diseño de Detallado del Software

Versión 1.0.0



Diciembre, 2024

Guayaquil, Ecuador

# Historial de Versionamiento

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Responsable** |
|  |  |  |  |
|  |  |  |  |

Contenido

[Historial de Versionamiento 2](#_Toc190075137)

[Listado de tablas 4](#_Toc190075138)

[Listado de gráficos 5](#_Toc190075139)

[Introducción 6](#_Toc190075140)

[Stakeholder 6](#_Toc190075141)

[Diagrama de Clases 6](#_Toc190075142)

[Diagrama de Interacción de Objetos 8](#_Toc190075143)

[Lista de Casos de Uso 8](#_Toc190075144)

[Descripción de Casos de Uso 9](#_Toc190075145)

[Módulo de Seguridad (MS) 9](#_Toc190075146)

[Módulo Académico (MA) 10](#_Toc190075147)

[Módulo de Juego Interactivo (MJI) 11](#_Toc190075148)

# Listado de tablas

[Tabla 1. Listado de los stakeholders. 6](#_Toc183102753)

# Listado de gráficos

[Grafico 1. Diagrama de Clases 7](file:///C:\Users\ASUS%20TUF%20DASH\Downloads\TIC-InnovaEdu_DDS-NOMBRE_PROYECTO-v1.0.0.docx#_Toc190075126)

# Introducción

El **juego de multiplicaciones** es una aplicación educativa diseñada para reforzar el aprendizaje de las tablas de multiplicar a través de una mecánica interactiva basada en **Flappy Bird**. Su arquitectura modular permite una separación clara entre **seguridad, gestión de preguntas y la dinámica del juego**, garantizandouna experiencia fluida y escalable.

Este documento detalla los principales **diagramas y casos de uso** implementados en el sistema, asegurando una descripción clara de la funcionalidad de cada módulo.

# Stakeholder

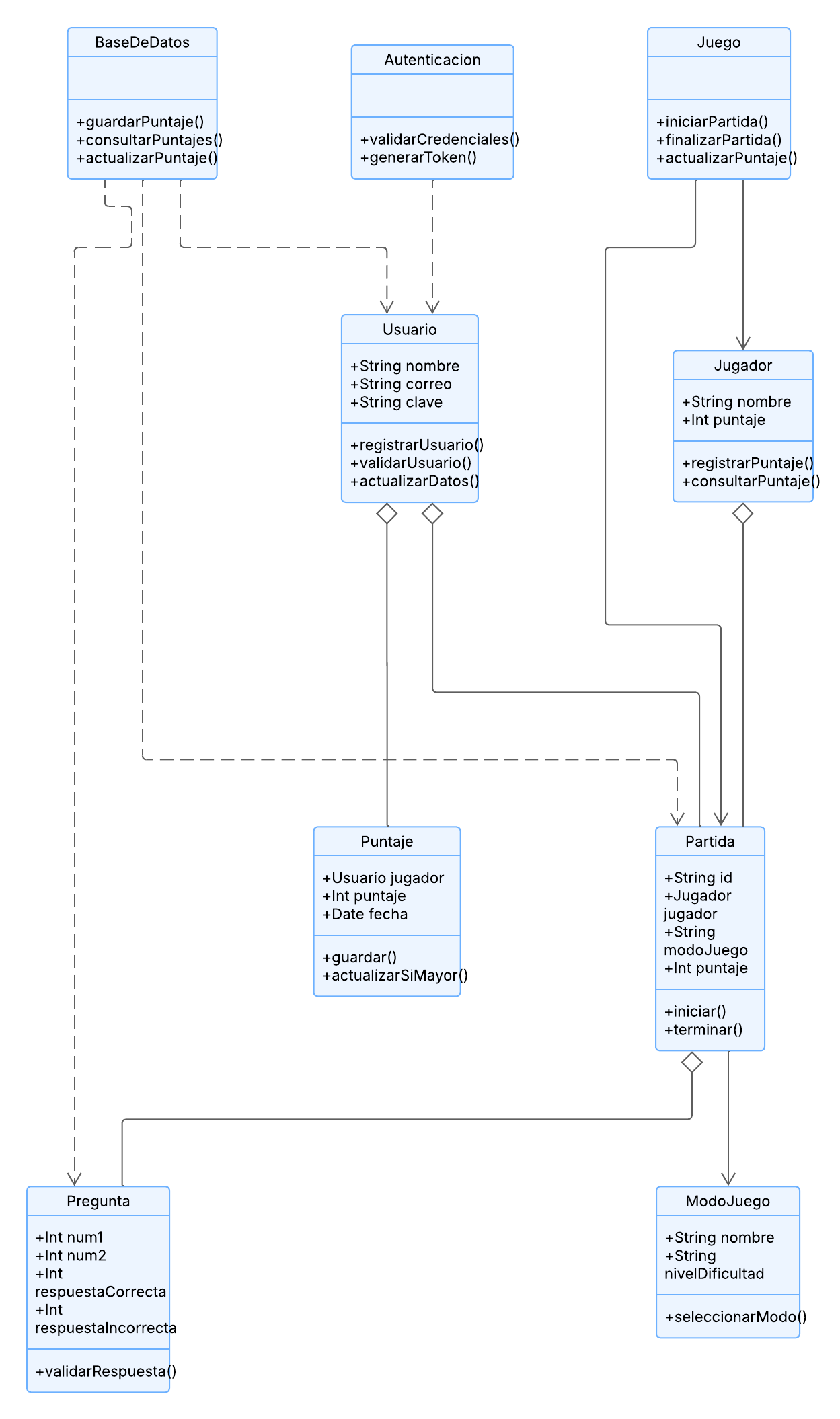
|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Administrador | Gestiona el sistema, administra usuarios y revisa los puntajes almacenados. |
| Jugador(Estudiante) | Usuario principal del juego. Responde preguntas de multiplicación y acumula puntaje. |
| Docente | Supervisa el rendimiento de los estudiantes, consulta puntajes y analiza estadísticas. |
| Escuela | Escuela o entidad que utiliza el juego como herramienta de aprendizaje. |
| Desarrolladores | Equipo encargado del diseño, implementación y mantenimiento del juego. |

# Diagrama de Clases

Las clases que van a intervenir en el desarrollo del juego de multiplicaciones son las siguientes:

* **Juego**
* **Jugador**
* **Partida**
* **Pregunta**
* **ModoJuego**
* **Usuario**
* **Autenticación**
* **BaseDeDatos**
* **Puntaje**

A continuación, se presenta el **Modelo de Clases** del sistema, que refleja la organización del código y la interacción entre los diferentes elementos del juego.



##### Grafico 1. Diagrama de Clases

# Diagrama de Interacción de Objetos

## Lista de Casos de Uso

A continuación, se listan los casos de uso a implementarse:

**Módulo de Seguridad**

|  |  |
| --- | --- |
| **Identificador** | **Caso de Uso** |
| CU-MS-001 | Ingresar al Sistema |
| CU-MS-002 | Consultar Tabla de Puntajes |

**Módulo Académico**

|  |  |
| --- | --- |
| **Identificador** | **Caso de Uso** |
| CU-MA-001 | Registrar pregunta en la base de datos |
| CU-MA-002 | Modificar pregunta en la base de datos |

**Módulo de Juego Interactivo**

|  |  |
| --- | --- |
| **Identificador** | **Caso de Uso** |
| CU-MJI-001 | Iniciar Partida |
| CU-MJI-002 | Seleccionar Modo de Juego |
| CU-MJI-003 | Registrar Sesión de Juego |
| CU-MJI-004 | Registrar Puntaje en la Base de Datos |

## Descripción de Casos de Uso

### Módulo de Seguridad (MS)

#### CU-MS-001: Ingresar al Sistema

|  |  |
| --- | --- |
| **Código** | CU-MS-001 |
| **Nombre** | Ingresar al Sistema |
| **Flujo de eventos:** | * El usuario ingresa su nombre. * El sistema verifica si el usuario ya existe en la base de datos. * Si el usuario es nuevo, se registra en la base de datos. * El usuario accede al menú principal. |
| **Condición de entrada:** | El usuario debe ingresar un nombre válido. |
| **Condición de salida:** | El usuario es registrado o autenticado correctamente en el sistema. |
| **Requerimientos de calidad:** | El proceso debe completarse en menos de 2 segundos. |

#### CU-MS-002: Consultar Tabla de Puntajes

|  |  |
| --- | --- |
| **Código** | CU-MS-002 |
| **Nombre** | Consultar Tabla de Puntajes |
| **Flujo de eventos:** | * Un usuario solicita ver la tabla de clasificación. * El sistema recupera los datos de la base de datos. * Se muestran los jugadores ordenados por puntaje. |
| **Condición de entrada:** | Debe haber al menos un usuario registrado con un puntaje. |
| **Condición de salida:** | Se muestra la tabla de clasificación actualizada. |
| **Requerimientos de calidad:** | La consulta debe completarse en menos de 1 segundo. |

### Módulo Académico (MA)

#### CU-MA-001: Registrar pregunta en la base de datos

|  |  |
| --- | --- |
| **Código** | CU-MA-001 |
| **Nombre** | Registrar pregunta en la base de datos |
| **Flujo de eventos:** | * Un administrador ingresa una nueva pregunta. * Se añaden las opciones de respuesta. * La pregunta se almacena en la base de datos. |
| **Condición de entrada:** | El usuario debe tener permisos de administrador. |
| **Condición de salida:** | La pregunta queda disponible en el sistema. |
| **Requerimientos de calidad:** | Validación de datos antes del almacenamiento. |

#### CU-MA-002: Modificar pregunta en la base de datos

|  |  |
| --- | --- |
| **Código** | CU-MA-002 |
| **Nombre** | Modificar pregunta en la base de datos |
| **Flujo de eventos:** | * Un administrador busca la pregunta en la base de datos. * Se editan sus valores (texto, respuestas, dificultad). * Se actualiza la información en la base de datos. |
| **Condición de entrada:** | La pregunta debe existir en la base de datos. |
| **Condición de salida:** | La pregunta se actualiza correctamente. |
| **Requerimientos de calidad:** | Solo los administradores pueden modificar preguntas. |

### 

### Módulo de Juego Interactivo (MJI)

#### CU-MJI-001: Iniciar Partida

|  |  |
| --- | --- |
| **Código** | CU-MJI-001 |
| **Nombre** | Iniciar Partida |
| **Flujo de eventos:** | * El jugador selecciona su nombre y el modo de juego. * Se genera una nueva partida con preguntas aleatorias. * El juego inicia y se presentan los primeros desafíos. |
| **Condición de entrada:** | El usuario debe haber ingresado su nombre y seleccionado un modo de juego. |
| **Condición de salida:** | La partida comienza y el jugador ve la primera pregunta. |
| **Requerimientos de calidad:** | La carga del juego debe realizarse en menos de 2 segundos. |

#### CU-MJI-002: Seleccionar Modo de Juego

|  |  |
| --- | --- |
| **Código** | CU-MJI-002 |
| **Nombre** | Seleccionar Modo de Juego |
| **Flujo de eventos:** | * El jugador elige entre Modo Fácil, Difícil o Tabla Personalizada. * Se configura la partida según la opción seleccionada. * El juego inicia con preguntas adaptadas al modo. |
| **Condición de entrada:** | El usuario debe haber seleccionado una opción válida. |
| **Condición de salida:** | La partida se configura correctamente según el modo elegido. |
| **Requerimientos de calidad:** | La selección del modo debe reflejarse de inmediato. |

#### CU-MJI-003: Registrar Sesión de Juego

|  |  |
| --- | --- |
| **Código** | CU-MJI-003 |
| **Nombre** | Registrar Sesión de Juego |
| **Flujo de eventos:** | * Durante la partida, el sistema registra respuestas correctas e incorrectas. * Se calcula el puntaje final cuando el jugador pierde o termina el juego. * Se almacena la sesión de juego en la base de datos. |
| **Condición de entrada:** | El usuario debe haber iniciado una partida. |
| **Condición de salida:** | Se almacena la sesión y se actualiza la tabla de puntajes. |
| **Requerimientos de calidad:** | El registro debe realizarse en tiempo real y sin retrasos. |

#### CU-MJI-004: Registrar Puntaje en la Base de Datos

|  |  |
| --- | --- |
| **Código** | CU-MJI-004 |
| **Nombre** | Registrar Puntaje en la Base de Datos |
| **Flujo de eventos:** | * Al finalizar una partida, se calcula el puntaje del jugador. * Se compara con el puntaje anterior almacenado. * Si el nuevo puntaje es mayor, se actualiza en la base de datos. |
| **Condición de entrada:** | El usuario debe haber finalizado una partida. |
| **Condición de salida:** | El puntaje queda almacenado correctamente si es mayor que el anterior. |
| **Requerimientos de calidad:** | No se deben registrar puntajes menores al anterior registrado. |