**Documento de Especificación de Base de Datos**

**Proyecto de Rector Simulator**

**Versión 1.0**

**HISTORIAL DE CAMBIOS**

| **Versión** | **Autores** | **Descripción** | **Fecha** |
| --- | --- | --- | --- |
| **1.0** | **Karlo Andrés Toro Vargas** | 1. **Introducción** 2. **Especificación técnica**   **2.2 Explicación de las entidades** | **27/09/2024** |
|  |  |  |  |

1. **Introducción**

**1.1.**  **Propósito de este documento**

**El presente documento proporciona toda la información relativa a la base de datos, desde aspectos técnicos como el DBMS a utilizar, la definición de los atributos de cada entidad, así como sus límites de tamaño, el diseño de la base de datos y una explicación lógica de las funciones y relaciones de cada entidad.**

1. **Especificaciones técnicas de la Base de Datos**

**2.1.** **Sistema Gestor de Base de Datos**

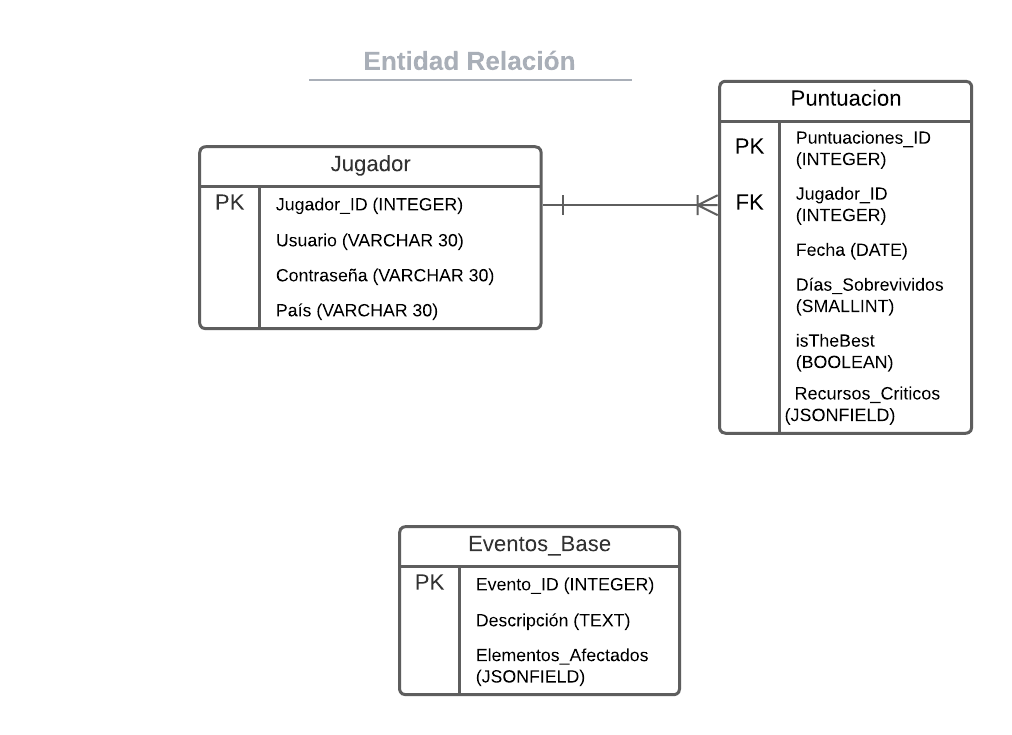
**Como sistema gestor de Base de Datos se usará MySQL, esto por los beneficios que se explicarán a continuación; todo relacionado con las necesidades de la aplicación que como proyecto está en desarrollo:**

**·** **Es gratuito y de código abierto, lo que significa que no se debe pagar una licencia para utilizarlo, sin restricciones del tipo de proyecto como con otras licencias, pudiendo aplicarse tanto en proyectos no comerciales como comerciales.**

**·** **Familiaridad del equipo de desarrollo: El equipo de desarrollo ha trabajado con anterioridad en sistemas que utilizan como DBMS a MySQL, lo que permite una mayor rapidez al momento de implementar el proyecto, gestionar cambios y solucionar problemas.**

**·** **Amplia cantidad de recursos: MySQL es una plataforma con casi 30 años en el mercado, lo que ha generado una gran y activa comunidad en torno a ella, por lo que se cuentan con amplios recursos como foros, tutoriales y libros que facilitan la resolución de problemas y agilizan el flujo de trabajo.**

**2.2.** **Entidad Relación**

****

**2.3.** **Explicación de las Entidades**

**2.3.1.** **Tabla: Jugador**

**El jugador cuenta con los elementos de identificación necesarios para su acceso al sistema, donde a su vez se muestra su relación 1 a n con “Puntuación”, ya que un jugador en el transcurso de sus partidas obtendrá distintos puntajes que formaran parte de su historial.**

**2.3.2.** **Tabla: Puntuaciones**

**Cada puntuación está asociada a un “Jugador” y este puede tener n puntuaciones, debido a esto es que Jugador\_ID es una llave foránea. El resto de campos resultan autoexplicativos si los vemos en el contexto de un ranking, sin embargo, “Recursos\_Criticos” requiere más atención, pues es un tipo de dato poco común en este caso JSON el cual nos va a permitir simular un array en MySQL, de esta forma reducimos el número de campos independientes de la entidad al mantener consistencia entre el valor esperado en cada posición del array implementado como JSON.**

**2.3.2.** **Tabla: Eventos Base**

**En el caso de esta entidad no mantiene relación con las otras, pues representa los casos iniciales y comunes a partir de los cuales se va a generar la historia mediante IA, indicándole los efectos que debe tener determinado escenario en la universidad así como una descripción para contextualizarla, a partir de la cual generara escenarios no previstos por el equipo de desarrollo, pero que mantengan coherencia histórica y lógica con la trama. El manejo del campo JSONFIELD cumplirá una función similar a “Recursos\_Criticos” en la entidad “Puntuaciones” donde se mantendrá la consistencia para cada índice del array.**