**Especificación de Requerimientos**

Sistema: Base de Datos para la Selección Argentina de futbol

Fecha: 28/11/2024

Versión: 1.0

Elaborado por: Cristian Ramirez, Lucas Romano, Luciano Varela, y Salvador Merayo

**1.1. Introducción**

La selección Argentina necesita una base de datos que tenga la información de algunos jugadores que pasaron por la selección, incluyendo su fecha de nacimiento y debut, en que equipo esta y en qué equipo debuto, estado físico, directo técnico que lo hizo debutar, su posición, que gano en la selección, su estado actual y cómo fue su paso en la selección.

**1.2. Objetivos del Proyecto**

1.Facilitar la información de algunos jugadores que jugador en la selección Argentina.

2. Tener registros de cómo le fue en su carrera futbolista tanto en clubes como en la selección.

**1.3. Alcance del Sistema**

La base de datos tiene los siguientes módulos principales:

1. Información del jugador y que numero tenía en la selección.
2. En cuantos equipos a jugado, en que equipo está actualmente o si ya está retirado, y en qué equipo debuto y en qué país fue su debut.
3. Su posición en la cacha.
4. Que peso tenía (cuando era jugador) o tiene.
5. Que altura tenía (cuando era jugador) o tiene.
6. El estado actual del jugador, si está jugando o esta retirado.
7. Cuantos títulos gano en la selección.
8. Si el jugador a jugado un mundial.
9. Los datos del director técnico que hizo debutar al jugador en la selección.

**1.4. Reglas de Negocio**

**RN-1:** Cada jugador debe tener un número de de ID para ser identificado.

**RN-2:** Todas las tablas están relacionada.

**RN-3:** Cada dato de las diferentes tablas deben estar asociados al id de jugador , en la tabla jugador y registrado .

**RN-4:** Cada técnico tiene que estar asignado a un único jugador a la vez, aunque varios técnicos puedan estar asociados a varios jugadores.

**1.5. Restricciones y Suposiciones**

**RS-1:** La base de datos deberá integrarse con un sistema de gestión de membresías en el futuro.

**RS-2:** No se considerará en esta fase la posibilidad de gestionar merchandising o ventas de boletos, ya que estos módulos se desarrollarán posteriormente.

**RS-3:** Cada jugador es único y no se repite el jugador.

**1.6. Aprobación y Validación**

Este documento de requisitos será revisado y aprobado por la selección Argentina . Las validaciones se realizan en cada etapa del proyecto para asegurar que el diseño y la implementación cumplen con los requisitos especificados.

**2. Diseño conceptual**

**Objetivo:** Crear una representación abstracta de los datos, que incluya entidades principales y sus relaciones.

**Actividades:**

1. Identificar las entidades.
2. Definir los atributos de cada entidad.
3. Establecer las relaciones entre las entidades.
4. Representar el diseño mediante un diagrama de diseño conceptual.

**Resultado:** Diagrama de diseño que muestra las entidades, sus atributos y relaciones.

**3. Diseño lógico**

**Objetivo:** Convertir el modelo conceptual en un modelo lógico específico para un sistema de bases de datos relacional.

**Actividades:**

1. Mapear las entidades y relaciones del modelo conceptual en tablas (en caso de bases de datos relacionales).
2. Definir las llaves primarias (para identificar de forma única cada registro) y llaves foráneas (para relacionar tablas).
3. Normalización: aplicar las reglas de normalización (hasta la 3NF o más) para reducir la redundancia y mejorar la integridad de los datos.
4. Crear un diagrama lógico que muestre las tablas, los campos y las relaciones entre ellas.
5. Crear tres vistas.

**Resultado:** Un esquema lógico de la base de datos que detalla las tablas, campos, llaves primarias y foráneas, y las relaciones.

**4. Diseño físico**

**Objetivo:** Transformar el modelo lógico en un diseño específico que optimice el rendimiento y considere las características físicas del sistema de base de datos (DBMS) que se usará.

**Actividades:**

1. Definir la estructura de almacenamiento en el DBMS elegido (incluye elección de tipos de datos, índices, particionamiento de tablas, etc.).
2. Determinar los espacios de almacenamiento y asignaciones de memoria según el volumen de datos esperado.

**Resultado:** Un diseño físico que especifica cómo se implementarán las tablas y sus relaciones en el DBMS elegido.

**5. Implementación**

**Objetivo:** Crear la base de datos real en el DBMS usando el diseño físico.

**Actividades:**

1. Escribir scripts SQL para crear tablas, índices, vistas y otros elementos definidos en el diseño físico.
2. Ejecutar los scripts en el DBMS para crear las estructuras de la base de datos.
3. Poblar la base de datos con datos iniciales (si es necesario).

**Resultado:** Una base de datos operativa que sigue el esquema diseñado y que está lista para ser usada.

**6. Pruebas**

**Objetivo:** Verificar que la base de datos funciona correctamente y cumple con los requisitos establecidos.

**Actividades:**

1. **Pruebas de integridad:** asegurarse de que las restricciones y relaciones funcionan correctamente.
2. **Pruebas de rendimiento:** optimizar consultas en caso de ser necesario.
3. **Pruebas de casos de uso:** validar que las consultas y las operaciones CRUD (crear, leer, actualizar, eliminar) funcionan correctamente.

**Resultado:** Una base de datos probada y validada, con un rendimiento aceptable y sin errores de diseño.

**7. Mantenimiento y optimización continua**

**Objetivo:** Asegurar que la base de datos continúe funcionando bien con el tiempo y se ajuste a nuevos requisitos.

**Actividades:**

1. Monitoreo del rendimiento y uso de la base de datos.
2. Optimización de consultas.
3. Realizar tareas de recuperación en caso de fallo.

**Resultado:** Una base de datos que evoluciona con las necesidades de la empresa y se mantiene eficiente y segura.

**Resumen de los pasos**

1. Recopilación y análisis de requisitos (documento de especificación de requerimientos).

2. Diseño conceptual (diagrama conceptual)

3. Diseño lógico (normalización y modelo de tablas)

4. Diseño físico (optimización específica del DBMS)

5. Implementación (creación de la BD en el DBMS)

6. Pruebas (verificación de integridad, rendimiento y seguridad)

7. Mantenimiento y optimización continua

Cada uno de estos pasos es esencial para el éxito de la base de datos. Un diseño bien estructurado permite un mejor rendimiento, facilita el mantenimiento y asegura la integridad de los datos. Además, una base de datos bien diseñada puede escalar y adaptarse más fácilmente a cambios futuros.