



Universidad Nacional del Nordeste



Facultad de Ciencias Exactas y Naturales y Agrimensura

Lic. en Sistemas de Información

Base de datos 1

Grupo N° 39

Integrantes:

<u>Apellido y Nombre</u>	<u>Número de DNI</u>
Orban, Tobias Naim	46.385.637
Firmapaz, Gabriel Andres	43.205.785
Gomez Hertler, Lisandro Leonel	46.461.232

Indice

Contenido:

Capítulo I — Introducción.....	4
1.1 Tema y contexto.....	4
1.2 Definición del problema.....	4
1.3 Objetivos.....	4
1.3.1 Objetivo general.....	4
1.3.2 Objetivos específicos.....	4
1.4 Alcances y limitaciones.....	5
Capítulo II — Marco conceptual o referencial.....	6
2.1. Enfoque y delimitación del problema.....	6
2.2. Conceptos clave.....	6
1. Plataforma social (social network).....	6
2. Contenido generado por usuarios (UGC).....	6
3. Experiencia de visualización.....	6
4. Calificación (rating).....	7
5. Opinión / Reseña.....	7
6. Spoilers.....	7
7. Privacidad.....	7
8. Seguimiento y favoritos.....	7
9. Curaduría / Partidos destacados.....	7
10. Recordatorios.....	8
11. Reputación y confianza.....	8
12. Integridad e interoperabilidad de datos.....	8
13. Autenticación y Seguridad.....	8
2.3. Terminología de datos.....	8
Capítulo III — Metodología.....	10
3.1 Diseño metodológico.....	10
3.2 Proceso de modelado.....	10
3.3 Estrategia de implementación.....	10
3.4 Herramientas y control de calidad.....	10
3.5 Esquema relacional.....	11
1. Entidades principales:.....	11
2. Highlights de diseño:.....	12
Capítulo IV — Resultados.....	13
4.1 Modelo de datos consolidado.....	13
4.2 Esquema relacional y DDL.....	13
4.3 Carga representativa y consultas.....	13
4.4 Documentación complementaria.....	14
Capítulo V — Conclusiones.....	15
5.1 Síntesis de hallazgos.....	15

5.2 Líneas futuras.....	15
Capítulo VI — Bibliografía.....	16
6.1 Referencias citadas.....	16
6.2 Recursos complementarios.....	16

Índice de figuras:

Figura 1: Diagrama Entidad Relación.....	11
--	----

Capítulo I — Introducción

1.1 Tema y contexto

Tribuneros es una red social del fútbol orientada a que las personas aficionadas puedan registrar, puntuar y comentar los partidos que ven. El proyecto propone el diseño e implementación de un modelo de datos relacional que sirva de base para futuras aplicaciones web y móviles centradas en la experiencia pospartido.

El trabajo se enmarca en la asignatura Bases de Datos I (FaCENA–UNNE) y se desarrolla bajo una modalidad iterativa que combina investigación, diseño conceptual y validación técnica mediante scripts en ANSI SQL. La documentación se organiza en capítulos para reflejar el recorrido académico solicitado por la cátedra.

1.2 Definición del problema

En la actualidad los hinchas gestionan su historial futbolero de forma dispersa (aplicaciones de resultados, redes sociales generalistas y notas personales). Esto dificulta:

- Consolidar estadísticas personales y comparativas.
- Compartir opiniones con control de spoilers y privacidad.
- Recibir recordatorios y recomendaciones personalizadas.
- Identificar partidos destacados curados por la comunidad.

Tribuneros busca unificar estas necesidades en una plataforma que capture la trazabilidad completa de las interacciones con cada partido, desde la visualización hasta la publicación de reseñas. Este requerimiento se traduce en el modelado de entidades, relaciones y reglas de negocio que garanticen integridad y escalabilidad.

1.3 Objetivos

1.3.1 Objetivo general

Diseñar, documentar e implementar el modelo de datos de Tribuneros en SQL Server, asegurando coherencia con los requisitos funcionales de la plataforma social propuesta.

1.3.2 Objetivos específicos

- Identificar actores, eventos y catálogos propios del dominio futbolero.
- Elaborar un esquema relacional normalizado con claves y restricciones explícitas.
- Implementar la base mediante scripts `script/creacion.sql` y poblarla con datos representativos desde `script/carga_inicial.sql`.
- Registrar la lógica de validación y métricas básicas en `script/verificacion.sql` y `script/conteo.sql`.

- Documentar decisiones de diseño, terminología y procesos para facilitar revisiones académicas.

1.4 Alcances y limitaciones

- Alcance funcional: se cubren usuarios, perfiles, partidos, ligas, equipos, interacciones (visualizaciones, calificaciones, opiniones), curaduría de destacados, favoritos, seguimientos y recordatorios.
- Alcance técnico: la implementación apunta a SQL Server 2019+ utilizando T-SQL. Se contemplan claves primarias, foráneas, restricciones CHECK y valores predeterminados.
- Fuera de alcance: algoritmos de recomendación avanzados, integración con APIs externas y análisis estadístico profundo quedan como trabajo futuro.
- Suposiciones: los datos de ejemplo son ilustrativos y pueden ampliarse en iteraciones posteriores; la autenticación y la gestión de multimedia se delegan a capas superiores de la aplicación.

Capítulo II — Marco conceptual o referencial

2.1. Enfoque y delimitación del problema

- Tema: registro social de experiencias de visionado de fútbol (partidos vistos) y producción de reseñas/calificaciones con control de spoilers.
- Objeto de estudio: las interacciones entre usuarios, partidos y contenido (visualizaciones, calificaciones, opiniones, favoritos, seguimientos, recordatorios y curaduría de destacados).
- Unidad de análisis: usuario, partido y publicación (opinión/calificación) como entidades primarias.
- Ámbito: ligas y equipos profesionales (extensible a selecciones y copas), con foco en el consumo pos-partido y la conversación segura (sin spoilers).

2.2. Conceptos clave

1. Plataforma social (social network)

- Definición operativa: sistema multiusuario con perfiles, publicaciones y vínculos (seguir equipos, marcar favoritos), persistidos en tablas usuarios, perfiles, opiniones, favoritos y seguidos.
- Rol: infraestructura sociotécnica que habilita la interacción, el efecto red y la gestión segura de acceso.

2. Contenido generado por usuarios (UGC)

- Definición operativa: opiniones, calificaciones y reseñas creadas por personas usuarias respecto de un partido (opiniones, calificaciones).
- Rol: insumo central para la experiencia, la reputación y la curaduría.

3. Experiencia de visualización

- Definición operativa: acto de ver un partido con metadatos de medio y duración (visualizaciones: medio, visto_en, minutos_vistos).

- Rol: fuente primaria de verdad para habilitar calificar/comentar y para métricas de consumo.

4. Calificación (rating)

- Definición operativa: valoración numérica discreta del partido (p.ej., 1–5) registrada en calificaciones.puntaje.
- Rol: variable cuantitativa para rankings, promedios y recomendaciones básicas.

5. Opinión / Reseña

- Definición operativa: publicación textual con banderas publica y tiene_spoilers en opiniones, con marca temporal de creación/edición.
- Rol: canal cualitativo para discusión y memoria del partido; su visibilidad depende de privacidad y spoilers.

6. Spoilers

- Definición operativa: revelación de eventos clave (marcador, goles decisivos) marcada por tiene_spoilers = 1.
- Rol: mecanismo de protección de experiencia que condiciona el feed y las reglas de visibilidad.

7. Privacidad

- Definición operativa: ámbito de publicación publica (público/privado) en opiniones y controles de perfil.
- Rol: determinante del alcance del contenido y del cumplimiento normativo.

8. Seguimiento y favoritos

- Definición operativa: relación de interés del usuario con equipos (seguidos) y marcación de partidos como favoritos (favoritos).
- Rol: personalización del feed, descubrimiento de contenido y notificaciones.

9. Curaduría / Partidos destacados

- Definición operativa: selección editorial o algorítmica registrada en partidos_destacados (destacado_en, nota).

- Rol: aumentar descubrimiento y conversación de calidad.

10. Recordatorios

- Definición operativa: programación de alertas para un partido (recordatorios.recordar_en, estado).
- Rol: elevar intención de visualización y completar el ciclo ver → calificar → opinar.

11. Reputación y confianza

- Definición operativa: señales derivadas del historial (antigüedad de cuenta, proporción de opiniones públicas, consistencia de calificaciones). En esta versión se infiere a partir de actividad; queda abierta la futura métrica explícita (p. ej., karma).
- Rol: ordenar el feed y priorizar contenido de calidad.

12. Integridad e interoperabilidad de datos

- Definición operativa: claves foráneas y estados válidos (estado en partidos, medio en visualizaciones, estado en recordatorios) que garantizan consistencia entre ligas, equipos y partidos.
- Rol: base para reportes confiables y evolución del modelo.

13. Autenticación y Seguridad

- Definición operativa: Uso de hashes criptográficos (SHA2_512) para almacenar contraseñas de usuarios en usuarios.password_hash y procedimientos almacenados para la gestión (sp_usuario_set_password_simple) y verificación (sp_usuario_login_simple) de credenciales.
- Rol: Proteger el acceso a las cuentas de usuario y asegurar la confidencialidad de las contraseñas.
- Definición operativa: claves foráneas y estados válidos (estado en partidos, medio en visualizaciones, estado en recordatorios) que garantizan consistencia entre ligas, equipos y partidos.
- Rol: base para reportes confiables y evolución del modelo.

2.3. Terminología de datos

Modelo lógico: se describe en detalle en el Capítulo III y su materialización se implementa en [script/creacion.sql](#).

Datos de ejemplo: la carga inicial que acompaña los casos de uso se encuentra en [script/carga_inicial.sql](#).

Diccionario de datos: referencia tabular completa disponible en [docs/diccionario_datos.md](#), imprescindible para interpretar columnas y restricciones.

Capítulo III — Metodología

3.1 Diseño metodológico

El trabajo adopta un enfoque iterativo-incremental. Cada iteración combina análisis del dominio, diseño conceptual y verificación técnica. Las actividades se organizaron en sprints cortos que produjeron artefactos verificables (diagramas, scripts SQL y documentación). Este enfoque facilita recibir retroalimentación temprana y ajustar el modelo antes de comprometer la implementación definitiva.

Se parte de los requisitos funcionales definidos en el Capítulo I y las definiciones operativas del Capítulo II. El resultado esperado es un repositorio reproducible que permita a la cátedra evaluar decisiones de diseño y calidad de datos.

3.2 Proceso de modelado

1. **Levantamiento del dominio:** identificación de entidades núcleo (usuarios, partidos, equipos) y eventos de interacción. Se elaboró un glosario compartido para evitar ambigüedades terminológicas.
2. **Modelado conceptual:** construcción del diagrama entidad-relación (ver [assets/der-tribuneros.png](#)) priorizando claridad de relaciones cardinales y atributos clave.
3. **Normalización:** revisión de dependencias funcionales y aplicación de 3FN para minimizar redundancias.
4. **Modelado lógico:** traducción del modelo conceptual a tablas, claves primarias/foráneas, restricciones y valores predeterminados.
5. **Validación cruzada:** contraste del modelo con casos de uso narrativos y con la carga de datos de prueba para detectar huecos.

3.3 Estrategia de implementación

- **Definición del esquema:** el script [script/creacion.sql](#) crea la base `tribuneros_bdi`, define tablas y restricciones y aplica índices de apoyo.
- **Poblado inicial:** [script/carga_inicial.sql](#) inserta un conjunto representativo de ligas, equipos, partidos y actividades de usuarios para ejercitar las restricciones.
- **Validaciones:** [script/verificacion.sql](#) incluye consultas de consistencia y verificaciones funcionales; `script/conteo.sql` provee métricas rápidas para auditoría.
- **Procedimientos almacenados:** se incluyen procedimientos para la gestión segura de contraseñas (`sp_usuario_set_password_simple`) y la verificación de credenciales (`sp_usuario_login_simple`).
- **Orden de ejecución:** se recomienda el flujo indicado en el README utilizando `:r` desde SQL Server Management Studio o `sqlcmd`.

3.4 Herramientas y control de calidad

- **Gestión de versiones:** GitHub para trazabilidad y revisión de cambios.

- **SQL Server:** motor objetivo, probado localmente con SQL Server 2019.
- **Documentación colaborativa:** Markdown para capítulos y diccionario de datos, siguiendo el orden propuesto en docs/indice.md.
- **Revisión:** controles manuales tras cada iteración y verificación automática mediante las consultas incluidas en los scripts de validación.

3.5 Esquema relacional



Figura 1: Diagrama Entidad Relación

1. Entidades principales:

- 1.1. Usuarios (usuarios, perfiles)
- 1.2. Catálogos (ligas, equipos)
- 1.3. Partidos (partidos)
- 1.4. Interacciones (calificaciones, opiniones, favoritos, visualizaciones)
- 1.5. Social (seguimiento_equipos, seguimiento_ligas, seguimiento_usuarios)
- 1.6. Curaduría y extras (partidos_destacados, recordatorios)

2. Highlights de diseño:

- PK/FK y **UNIQUE** en combinaciones clave (partido_id,usuario_id).
- **CHECK** para estados (partidos.estado, visualizaciones.medio, recordatorios.estado).
- **ON DELETE CASCADE** en relaciones de usuario; **RESTRICT/SET NULL** en catálogos.
- Índices por partidos.fecha_utc, partidos.liga_id y usuario_id en tablas de interacciones para consultas frecuentes.

Capítulo IV — Resultados

4.1 Modelo de datos consolidado

El modelo consolidado cubre el ciclo completo de interacción en Tribuneros: identificación de usuarios y perfiles, catalogación de ligas/equipos, programación y registro de partidos, acciones sociales (calificaciones, opiniones, favoritos, seguimientos) y mecanismos de curaduría y recordatorios. El diagrama entidad-relación actualizado se encuentra en `assets/der-tribuneros.png` y sintetiza las cardinalidades y claves primarias/foráneas definidas.

4.2 Esquema relacional y DDL

El script [script/creacion.sql](#) materializa el modelo lógico en SQL Server. Entre los puntos más relevantes se destacan:

- Declaración explícita de claves primarias (**PK_***) y foráneas con reglas de eliminación (**ON DELETE**).
- Inclusión de la columna **password_hash** en la tabla **usuarios** para el almacenamiento seguro de contraseñas.
- Restricciones de dominio (**CHECK**) para estados de partidos (**estado**), medios de visualización (**medio**), rangos de puntaje y estados de recordatorios.
- Índices de apoyo (**IX_***) para optimizar consultas por usuario, partido y fechas clave.
- Valores predeterminados de fechas mediante **CURRENT_TIMESTAMP** para trazabilidad temporal.
- Implementación de procedimientos almacenados (**sp_usuario_set_password_simple**, **sp_usuario_login_simple**) para la gestión de autenticación.

La ejecución del script crea la base **tribuneros_bdi** y garantiza un estado limpio al reiniciar el entorno (drop condicional de tablas).

4.3 Carga representativa y consultas

- **Datos de ejemplo:** [script/carga_inicial.sql](#) inserta usuarios con contraseñas iniciales, ligas, equipos y un partido emblemático (River vs Boca), junto con interacciones asociadas (visualización, calificación, opinión, favorito y recordatorio).

- **Validación operativa:** [script/verificacion.sql](#) ofrece consultas para comprobar integridad referencial (joins entre partidos y equipos, conteo de calificaciones/opiniones) y escenarios de negocio (partidos destacados vigentes).
- **Métricas rápidas:** [script/conteo.sql](#) resume cantidades por entidad para monitorear la carga.

Estos artefactos permiten reproducir la evidencia de funcionamiento y sirven de base para ensayos adicionales.

4.4 Documentación complementaria

- **Diccionario de datos:** el detalle por tabla, campo y restricción se encuentra en [docs/diccionario_datos.md](#).
- **Guía de ejecución:** el [README](#) explica el procedimiento recomendado para correr los scripts en SQL Server Management Studio o sqlcmd.
- **Material visual:** recursos gráficos en [assets/](#) (banner, badge e imagen DER) acompañan presentaciones y entregas.

Capítulo V — Conclusiones

5.1 Síntesis de hallazgos

- El modelo de datos desarrollado satisface los requerimientos principales de la plataforma Tribuneros, permitiendo almacenar interacciones complejas entre usuarios, partidos y contenido generado por la comunidad.
- Las restricciones implementadas en [script/creacion.sql](#) y validadas con [script/verificacion.sql](#) aseguran integridad referencial y control de dominios, reduciendo errores durante la carga.
- La documentación generada (capítulos, índice y diccionario) consolida un circuito claro para revisión académica y facilita la trazabilidad entre narrativa, decisiones técnicas y scripts ejecutables.

5.2 Líneas futuras

1. Ampliar la carga de datos con casos que representen distintas competencias, fases del torneo y diversidad de usuarios, extendiendo `script/carga_inicial.sql`.
2. Incorporar métricas avanzadas de reputación y recomendación, apoyándose en vistas o procedimientos almacenados que se sumen a los scripts existentes.
3. Integrar automatizaciones (tests unitarios SQL, pipelines CI) que ejecuten `creacion.sql`, `carga_inicial.sql` y `verificacion.sql` en ambientes controlados.
4. Diseñar APIs y prototipos de interfaz que consuman la base, reutilizando el diccionario y los diagramas para alinear implementaciones futuras.

Capítulo VI — Bibliografía

6.1 Referencias citadas

- Elmasri, R., & Navathe, S. (2016). Fundamentals of Database Systems (7th ed.). Pearson.
- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2020). Database System Concepts (7th ed.). McGraw-Hill.
- Fowler, M. (2003). Patterns of Enterprise Application Architecture. Addison-Wesley.
- ISO/IEC. (2016). ISO/IEC 9075 — Information technology — Database languages — SQL. International Organization for Standardization.
- Microsoft. (2024). Transact-SQL Reference (Database Engine). Recuperado de <https://learn.microsoft.com/sql/t-sql/language-reference>

6.2 Recursos complementarios

- Diagrama entidad-relación: [Figura 1: Diagrama Entidad Relación](#) .
- Diccionario de datos: [docs/diccionario_datos.md](#).
- Scripts del proyecto: [script/](#).
- Repositorio en GitHub: [BD1-proyecto](#).