

Práctica 3:
Diseño, desarrollo e implantación de
un sistema de información Web:
diseño e implementación de la capa
de persistencia de datos



Andrea Ferradas Agudo
Jorge Hernández Aznar
Mario Clavero Artal
Sistemas de Información 2024/2025

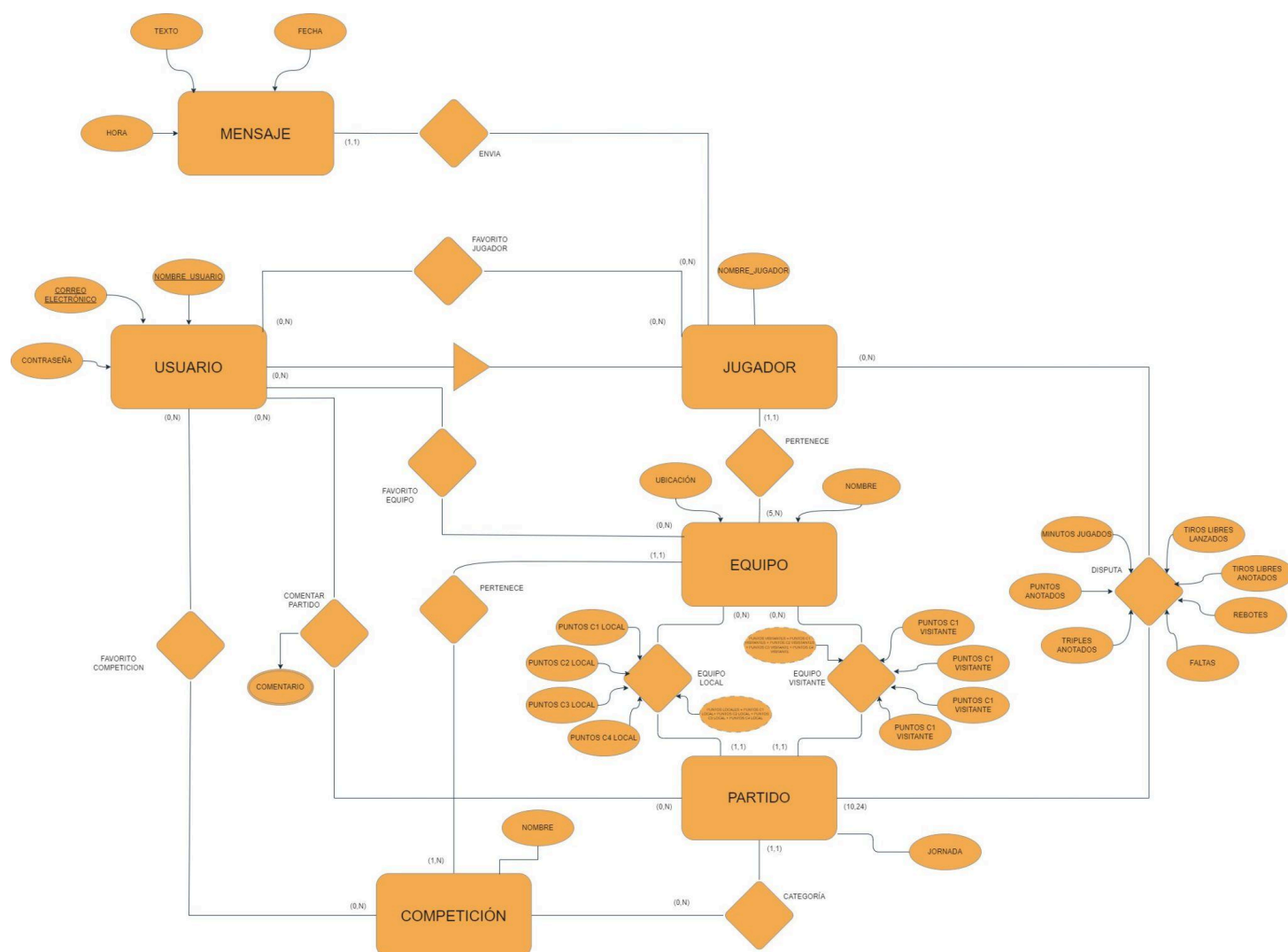
Resumen

Hemos decidido desarrollar un sistema que guarde la información de los partidos de baloncesto que se lleven a cabo a través de la Federación Española de Baloncesto. Se gestionará información de las competiciones, equipos y jugadores que participen.

Las principales funcionalidades en las que el usuario más interactúa serán realizar y publicar comentarios sobre partidos concretos y además los propios jugadores tendrán un chat propio para mandar mensajes a sus seguidores. A parte de estas funcionalidades, el usuario podrá añadir a favoritos sus jugadores, equipos y competiciones de las cuales podrá obtener información detallada.

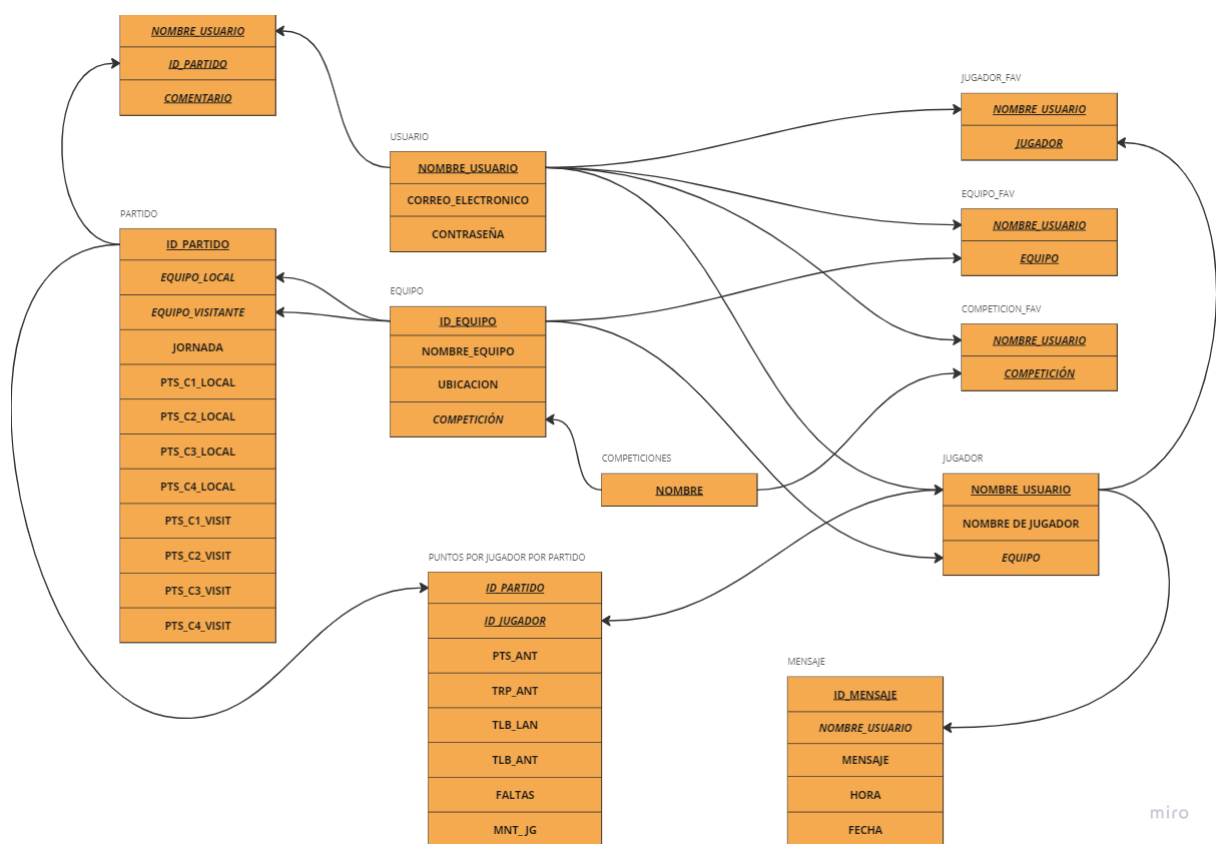
Los principales perfiles de usuario serán el público objetivo y el administrador. En el público objetivo se pueden distinguir dos subtipos, los jugadores que tendrán su propio chat para mandar mensajes, y los seguidores que son simplemente aficionados que podrán comentar los partidos a su vez. La figura del administrador se encargará de introducir la información necesaria sobre jugadores, equipos, competiciones y partidos.

Esquema Entidad-Relación



En nuestro modelo entidad-relación gestionamos un sistema deportivo donde podemos enviar **MENSAJE**, cada jugador puede enviar los mensajes por su canal de difusión y se necesita guardar cada mensaje; hacer **COMENTARIOS**, cada usuario puede realizar comentarios en cada uno de los partidos y marcar como favoritos a **JUGADORES**, **EQUIPOS** y **COMPETICIONES**. Los **JUGADORES** forman parte de **EQUIPOS** y participan en **PARTIDOS**, donde registramos sus estadísticas, como puntos anotados, rebotes y faltas. A su vez, los **EQUIPOS** compiten como local o visitante en los **PARTIDOS**, que pertenecen a una **COMPETICIÓN** específica. Cada **PARTIDO** detalla los puntos por cuarto para ambos equipos.

Esquema Relacional



En nuestro modelo relacional, tenemos varias tablas que almacenan información clave del sistema deportivo. La tabla **USUARIO** guarda los datos de los usuarios, como nombre, correo y contraseña; la tabla **PARTIDO** registra los detalles de los encuentros, como equipos, puntos por cuarto y jornada. **EQUIPO** almacena el nombre, ubicación y competición de cada equipo, mientras que **JUGADOR** guarda el nombre de los jugadores y el equipo al que pertenecen. **COMENTARIOS** conecta a los usuarios con los comentarios que hacen sobre los partidos, y **PTS_JUG_PAR** registra las estadísticas de los jugadores, como puntos, triples, faltas y minutos jugados. Además, **MENSAJE** almacena los mensajes enviados por los usuarios, mientras que las tablas **JUGADORES_FAV**, **EQUIPO_FAV** y **COMPETICION_FAV** relacionan a los usuarios con sus jugadores, equipos y

competiciones favoritas, respectivamente. Finalmente, la tabla **COMPETICIONES** guarda el nombre de cada competición.

Cada tabla tiene subrayada su clave primaria o clave compuesta, del mismo modo que cada una tiene referenciada con una flecha en cursiva su clave extranjera.

Creación de tablas

Para la creación de las tablas sql nos hemos basado en el esquema relacional, hemos asignado tipos de variables, claves primarias, extranjeras y sobretodo algunas restricciones, ya sean atributos como en la tabla de usuario con atributos que no pueden ser nulos. Por otro lado la restricción puede ser que el número de puntos en partido sea negativo.

```
CREATE TABLE sisinf_db.USUARIO (  
    nombre_usuario TEXT PRIMARY KEY,  
    correo_elec TEXT NOT NULL,  
    password TEXT NOT NULL  
);  
  
CREATE TABLE sisinf_db.PARTIDO (  
    id_partido INTEGER PRIMARY KEY,  
    equipo_local INTEGER REFERENCES EQUIPO(id_equipo) ON DELETE CASCADE,  
    equipo_visitante INTEGER REFERENCES EQUIPO(id_equipo) ON DELETE CASCADE,  
    jornada INTEGER,  
    pts_c1_local INTEGER CHECK (pts_c1_local >= 0),  
    pts_c2_local INTEGER CHECK (pts_c2_local >= 0),  
    pts_c3_local INTEGER CHECK (pts_c3_local >= 0),  
    pts_c4_local INTEGER CHECK (pts_c4_local >= 0),  
    pts_c1_visit INTEGER CHECK (pts_c1_visit >= 0),  
    pts_c2_visit INTEGER CHECK (pts_c2_visit >= 0),  
    pts_c3_visit INTEGER CHECK (pts_c3_visit >= 0),  
    pts_c4_visit INTEGER CHECK (pts_c4_visit >= 0),  
    CHECK (equipo_local != equipo_visitante)  
);
```

Creación de la Base de Datos

Para crear la base de datos en nuestro entorno de trabajo eclipse hemos decidido utilizar una Base de Datos **PostgreSQL**, para gestionarla con más facilidad usamos **DBeaver**.

Inicialmente dentro de la perspectiva de DBeaver, seleccionamos crear una Conexión a una base de datos, seleccionamos el tipo de BD PostgreSQL, asociamos el usuario y contraseña pertinentes al igual que los drivers. Una vez creada la conexión nos conectamos y creamos un nuevo esquema en la BD llamado **sisinf_db**. Aquí es donde crearemos todas las tablas necesarias. Para ello usamos el script *crearTablas.sql* adjuntado en los ficheros fuentes. Se podría matizar que para que se incluyan las tablas en el esquema correcto hay que usar la siguiente sintaxis.

```
CREATE TABLE sisinf_db.USUARIO (  
    nombre_usuario TEXT PRIMARY KEY,  
    correo_elec TEXT NOT NULL,  
    password TEXT NOT NULL  
);
```

A continuación, creamos varios inserts para introducir el usuario administrador, tres jugadores, un equipo y una competición.

Finalmente creamos una clase de java para cada una de las tablas, definiendo sus atributos privados, constructor y métodos para establecer y obtener los valores de las variables.