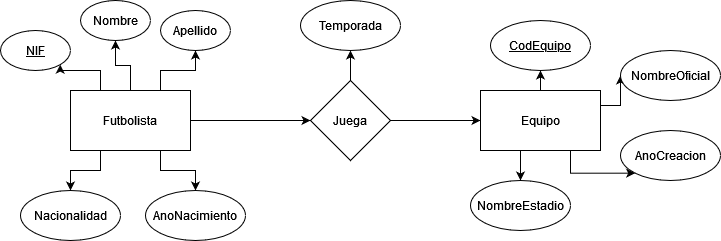
ADA-ITS

CICLO FORMATIVO DE GRADO SUPERIOR DESARROLLO DE APLICACIONES WEB

FCB FEMENINO

Jesús Carrillo González

SEVILLA, 2023

***APARTADO A***

1:N

1:N

## APARTADO B

*B.1 Tabla Futbolista*

CREATE TABLE **Futbolista**(

    NIF VARCHAR(9) PRIMARY KEY,

    Nombre VARCHAR (20),

    Apellido VARCHAR(20),

    AnoNacimiento SMALLINT(4) UNSIGNED NOT NULL,

    Nacionalidad VARCHAR(10) NOT NULL

);

*B.2 Tabla Equipo*

CREATE TABLE **Equipo**(

    CodigoEquipo SMALLINT(4) PRIMARY KEY,

    NombreOficial VARCHAR(15) UNIQUE,

    AnoCreacion SMALLINT(4) UNSIGNED NOT NULL,

    NombreEstadio VARCHAR(30) NOT NULL

);

*B.3 Tabla Juega*

CREATE TABLE **Juega**(

    Temporada SMALLINT(4),

    NIF VARCHAR(9),

    CodigoEquipo SMALLINT(4),

    PRIMARY KEY(Temporada, NIF, CodigoEquipo),

    CONSTRAINT fk\_Juega\_Futoblista FOREIGN KEY(NIF) REFERENCES Futbolista(NIF),

    CONSTRAINT fk\_Juega\_Equipo FOREIGN KEY(CodigoEquipo) REFERENCES Equipo(CodigoEquipo)

);

## APARTADO C

*C.1. Añade un atributo en el que se recoja si la jugadora llegó de los escalafones inferiores del FCB. Este atributo solo puede tomar los valores “S” o “N”.*

ALTER TABLE Futbolista ADD COLUMN EscalafonesInferiores CHAR(1) NOT NULL CHECK(EscalafonesInferiores IN ('S', 'N'));

*C.2. Añade otro atributo que contenga el e-mail de la jugadora. El valor de este*

*atributo debe ser único para cada jugadora y se debe comprobar que contiene la*

*arroba (@) como parte de su validez.*

ALTER TABLE Futbolista ADD COLUMN Email VARCHAR(30) CHECK(Email LIKE '%@%');

*C.3. Escribe la sentencia que eliminaría el atributo nacionalidad de la jugadora.*

ALTER TABLE Futbolista DROP COLUMN Nacionalidad;

*C.4. Escribe la sentencia que añadiría un atributo que contenga la capacidad de* *espectadores que tiene el estadio. Debe ser un número positivo no nulo y menor de* *99.999, y por supuesto mayor de cero.*

ALTER TABLE Equipo ADD COLUMN Capacidad INT(5) UNSIGNED NOT NULL CHECK(Capacidad > 0 AND Capacidad < 99999);

*C.5. Escribe la sentencia que cree un índice con el campo nombre y apellidos de las jugadoras.*

CREATE INDEX IndiceNombreApellido ON Futbolista (Nombre, Apellido);

*C.6. Escribe la sentencia que cree un índice con el campo año de creación del equipo.*

CREATE INDEX IndiceFechaIngreso ON Equipo (AnoCreacion);

*C.7. Si por ejemplo has llamado a la tabla contiene a las jugadoras como* *“FUTBOLISTAS”, escribe la sentencia que renombraría esa tabla a “JUGADORAS”.*

ALTER TABLE Futbolista RENAME TO Jugadora;

*C.8. Escribe la sentencia que eliminaría esta última tabla.*

DROP TABLE Jugadora;

## APARTADO D

*D.1. PROCEDIMIENTO ALMACENADO que agregue una JUGADORA. Deberá recibir todos los* *atributos de la jugadora, que se pasarán como parámetros de entrada.*

DELIMITER $$

DROP PROCEDURE IF EXISTS AddJugadora;

CREATE PROCEDURE AddJugadora(IN NIF VARCHAR(9), IN Nombre VARCHAR (20), IN Apellido VARCHAR(20), IN AnoNacimiento DATE, IN Nacionalidad VARCHAR(10))

BEGIN

    INSERT INTO Futbolista VALUES(NIF, Nombre, Apellido, AnoNacimiento, Nacionalidad);

END $$

DELIMITER ;

*D.2. PROCEDIMIENTO ALMACENADO que agregue un EQUIPO. Deberá recibir todos los atributos del equipo, que se pasarán como parámetros de entrada.*

DELIMITER $$

DROP PROCEDURE IF EXISTS AddEquipo;

CREATE PROCEDURE AddEquipo(IN NombreOficial VARCHAR(15), IN AnoCreacion SMALLINT(4), IN NombreEstadio VARCHAR(20))

BEGIN

    INSERT INTO Equipo VALUES(NombreOficial, AnoCreacion, NombreEstadio);

END $$

DELIMITER ;

*D.3. PROCEDIMIENTO ALMACENADO que modifique año de nacimiento de una JUGADORA. Recibirá como parámetro de entrada la clave de la jugadora y el nuevo valor del año de nacimiento.*

DELIMITER $$

DROP PROCEDURE IF EXISTS ModificarJugadora;

CREATE PROCEDURE ModificarJugadora(IN NIFIntroducido VARCHAR(9), IN NuevoAnoNac SMALLINT(4))

BEGIN

    UPDATE FUTBOLISTA SET ANONACIMIENTO = NuevoAnoNac WHERE NIF = NIFIntroducido;

END $$

DELIMITER ;

*D.4. PROCEDIMIENTO ALMACENADO que obtiene la lista de JUGADORAS jugaron en una determinada temporada en un determinado equipo. Como parámetros de entrada, el procedimiento recibirá un identificador de equipo y la temporada para la que deseamos sacar el listado. La salida contendrá el título de la temporada, el nombre del equipo, y el nombre/apellidos de las jugadoras.*

DELIMITER $$

DROP PROCEDURE IF EXISTS JugadorasPorTemporada;

CREATE PROCEDURE JugadorasPorTemporada(IN CodigoEquipoEntrada SMALLINT(4), IN TemporadaEntrada SMALLINT(4))

BEGIN

    SELECT JUEGA.TEMPORADA, EQUIPO.NOMBREOFICIAL, CONCAT(FUTBOLISTA.NOMBRE, " ",FUTBOLISTA.APELLIDO) AS "Nombre/Apellido" FROM

    FUTBOLISTA NATURAL JOIN JUEGA NATURAL JOIN EQUIPO WHERE EQUIPO.CODIGOEQUIPO=CodigoEquipoEntrada AND JUEGA.TEMPORADA=TemporadaEntrada;

END $$

DELIMITER ;

*D.5. PROCEDIMIENTO ALMACENADO que lista los equipos en los que ha militado una jugadora. El identificador de la jugadora será el que se le pasa al PROCEDIMIENTO como parámetro.*

DELIMITER $$

DROP PROCEDURE IF EXISTS EquiposDeJugadora;

CREATE PROCEDURE EquiposDeJugadora(IN NIFIntroducido VARCHAR(9))

BEGIN

    SELECT DISTINCT EQUIPO.NOMBREOFICIAL AS "Nombre-Equipo" FROM EQUIPO NATURAL JOIN JUEGA NATURAL JOIN FUTBOLISTA WHERE FUTBOLISTA.NIF=NIFIntroducido;

END $$

DELIMITER ;

*D.6. FUNCION ESCALAR que recibe como parámetro el identificador de la jugadora y devuelva el número (cantidad) de equipos en los que ha militado.*

DROP FUNCTION IF EXISTS Escalar;

DELIMITER $$

CREATE FUNCTION **Escalar**(NIFIntroducido VARCHAR(9))

    RETURNS INT

    BEGIN

        DECLARE numeroEquipos INT;

        SELECT COUNT(DISTINCT EQUIPO.NOMBREOFICIAL) INTO numeroEquipos FROM EQUIPO NATURAL JOIN JUEGA NATURAL JOIN FUTBOLISTA WHERE FUTBOLISTA.NIF="66666666F";

    RETURN numeroEquipos;

END $$

DELIMITER ;

*D.7. Vista (VIEW) que liste temporada, nombre de equipo y nombre/apellido de jugadora. Deberá estar ordenado por temporada-equipo-jugadora.*

CREATE OR REPLACE VIEW **listarTemporadaEquipoJugadora** AS SELECT JUEGA.TEMPORADA, EQUIPO.NOMBREOFICIAL, CONCAT(FUTBOLISTA.NOMBRE, " ", FUTBOLISTA.APELLIDO) FROM FUTBOLISTA NATURAL JOIN JUEGA NATURAL JOIN EQUIPO ORDER BY JUEGA.TEMPORADA, EQUIPO.NOMBREOFICIAL, FUTBOLISTA.NOMBRE;

*D.8. Vista (VIEW) que lista todos las jugadoras y equipos en los que ha militado.*

CREATE OR REPLACE VIEW **ListarJugadoraEquipo** AS SELECT DISTINCT EQUIPO.NOMBREOFICIAL, CONCAT(FUTBOLISTA.NOMBRE, " ", FUTBOLISTA.APELLIDO) FROM FUTBOLISTA NATURAL JOIN JUEGA NATURAL JOIN EQUIPO ORDER BY FUTBOLISTA.NOMBRE;