

FC Barcelona Online: Fase 01

1 Descripción del negocio

1.1 Entidades de negocio

El **FC BARCELONA** ha decidido crear una aplicación web administrativa.

El **FC BARCELONA** tiene personal contratado (*empleados*) y *directivos*. Todos ellos tienen en común características tales como *nombre*, *apellido*, *fecha de nacimiento* y *nacionalidad*. Dentro del personal contratado se pueden distinguir varios tipos: *jugadores*, *técnicos*, *doctores* y *fisioterapeutas*.

Los jugadores pueden clasificarse en *arqueros*, *defensas*, *mediocampistas* y *delanteros*. Los jugadores pertenecen a los activos del club, por lo que además tienen otras características adicionales:

- Un *valor base*, y
- un *valor de mercado* que se calcula utilizando el *valor base* y un *coeficiente* que depende del tipo de jugador. Para un arquero el coeficiente es 1, para un defensa 1,2, para un mediocampista 1,5 y para un delantero 2.

Por ejemplo: Si Leo Messi tiene un valor base de 60 millones de euros, y es delantero, entonces su valor de mercado es $60 \times 2 = 120$ millones de euros.

El **FC BARCELONA** tiene además una masa social importante. Los socios del club pueden ser de 2 tipos: socios con derecho a asiento o simpatizantes (sin derecho a asiento). Los socios tienen datos personales (nombre, apellido, fecha de nacimiento y nacionalidad), además de datos de contacto que pueden ser de varios tipos (dirección, teléfono o email).

Todo personal contratado (empleados) tiene un contrato vigente con el club. Todo socio tiene un contrato vigente con el club. Un contrato tiene atributos como fecha de inicio, fecha de expiración, y mensualidad.

El **FC BARCELONA** posee también otro tipo de activos (por ejemplo: el estadio, bienes raíces diversos, campo de entrenamiento, etc.), los cuales tienen un valor económico. El Club también tiene pasivos, los cuales también tiene un valor económico además de un estado (el cual puede ser pendiente o pagado).

Para finalizar, la aplicación tiene usuarios registrados, los cuales se caracterizan por su *nombre de usuario* y *contraseña*. El nombre de usuario debe ser único.

1.2 Procesos de negocio

1.2.1 Administración de usuarios

- Agregar y eliminar usuarios.

1.2.2 Administración de contratos

- Agregar, eliminar o actualizar contratos de personal contratado: Los contratos siempre están relacionados con un empleado del club. Si un contrato es eliminado, se debe eliminar el empleado relacionado con dicho contrato. La actualización de un contrato se realiza sobre atributos como la fecha de inicio, fecha de expiración y mensualidad.
- Agregar, eliminar o actualizar contratos de socios: idem.

Por simplicidad no es necesario preocuparse de actualizar los datos una vez que los contratos llegan a su fecha de expiración. En otras palabras, si hay un contrato en la base de datos que ya ha expirado, no hace falta eliminarlo del sistema.

1.2.3 Administración de personal

- Agregar, eliminar y actualizar empleados: al agregar un empleado debe existir un contrato relacionado con él. De la misma forma, si se elimina un empleado el contrato correspondiente también debe ser eliminado.

1.2.4 Administración de finanzas del club

- Agregar, eliminar y actualizar activos
- Agregar, eliminar y actualizar pasivos

Para tener un control de las finanzas del club, se pide también proveer lo siguiente:

- Cálculo del valor total de activos: considerando los jugadores y los otros activos
- Cálculo del valor total de pasivos: sólo considerar los pasivos con estado "pendiente"
- Balance total (activos - pasivos "pendientes")

2 Entregable Fase 1

Para esta primera fase, se solicita que construya la totalidad de la capa de negocios de la aplicación. Puede usar el diseño tradicional de **3 (o N) capas** o bien un enfoque **DDD (Domain-driven design)**. En concreto se le pide:

1. Escribir el código de la capa de negocios, usando session beans y entidades.
2. Escribir tests unitarios y tests de integración para validar funcionalmente la lógica de su código. Indique claramente cuales son test unitarios y cuales de integración.

3. Un documento con las decisiones de diseños tomadas donde explique las decisiones más importantes que ha tenido que tomar, con un análisis de las distintas alternativas y su debida justificación.

3 Requerimientos no funcionales de la Aplicación Web

El desarrollo del sistema debe respetar un conjunto de requerimientos no funcionales que se describen a continuación:

1. Utilización de JPA para el mapping de entidades.
2. Usar las buenas prácticas de programación y diseño orientado a objetos.
3. Cobertura de código (por los tests) de al menos **70%** de la capa de negocios.

4 Entregable

El proyecto debe ser entregado a través de U-Cursos a más tardar el viernes 3 de mayo a las 23:59. **NO se aceptan atrasos.** El entregable debe contener:

- El proyecto que debe contener todas las fuentes y librerías utilizadas.
- Un pequeño documento de diseño que explique¹:
 - ✓ Las decisiones de diseño tomadas.
 - ✓ El modelo de clases.

El proyecto debe realizarse en grupos de a 2 personas. El trabajo estimado para esta tarea es de 15UD por persona. Se elegirá una mejor tarea considerando la calidad del diseño, la calidad técnica del código fuente (apego a buenas prácticas de programación orientado a objetos), funcionalidades adicionales no solicitadas etc.

¹ El documento no debe poseer faltas de ortografía ni de gramática. Estas serán penalizadas duramente.