



Universidad de Valladolid

Escuela de Ingeniería Informática

TRABAJO DE FIN DE GRADO

Grado en Ingeniería Informática
Mención Ingeniería de Software

**Aplicación web para gestionar
y compartir ejercicios de
baloncesto**

Autor:

D. Rubén González Ruiz

Tutor:

D.^a Yania Crespo González-Carvajal

Agradecimientos

Quisiera dedicar unas líneas para recordar y agradecer a todas esas personas que han hecho posible el que hoy esté en el punto que estoy.

En primer lugar, a mis padres, Carmen y José Antonio, por haber estado ahí cuando más cuesta arriba se hacía la carrera, los que más me han apoyado en los momentos difíciles y los primeros en celebrar conmigo los buenos. En segundo lugar, me gustaría agradecer también a mis amigos y compañeros entrenadores su colaboración en este proyecto. Por último, dar las gracias también a todos los profesores que han participado en mi formación, de los que de todos me llevo algo bueno, especialmente a Yania, mi tutora, por la formación que nos ha dado a mis compañeros y a mí y por haberme posibilitado llevar a cabo este proyecto

Resumen

En el presente documento se recoge el desarrollo de una web para crear y compartir ejercicios de baloncesto. Dicha aplicación web está orientada a entrenadores de baloncesto para que puedan tener un lugar donde almacenar y compartir con otros los ejercicios que habitualmente utilizan en sus entrenamientos y poder reutilizarlos sin tener que describirlos o dibujarlos de continuo.

El proyecto ha seguido la metodología del Proceso Unificado orientado a la Educación, con las fases que esto conlleva: inicio, elaboración, construcción y transición.

Antes de comenzar el desarrollo, se ha realizado un boceto de la misma en papel y ha sido mostrada a entrenadores reales para que estos dieran una primera opinión y añadir alguna modificación durante las fases de análisis y diseño. Una vez terminada la aplicación, ha sido desplegada en un servidor online en su versión gratuita y compartida con los usuarios que vieron el boceto inicial, donde han podido realizar pruebas y dar su visión de la aplicación.

Abstract

This document includes the developmental process of a website to create and share basketball exercises. In this sense, this web application is aimed at basketball coaches so they can have a place to store and share with others the exercises they usually use in their workouts and be able to reuse them without having to describe or draw them continuously.

The project has followed the methodology of the Unified Process for Education, with the phases that this entails: Inception, Elaboration, Construction and Transition

Before starting the development, a paper sketch was made and it was shown afterwards to real coaches for them to give a first opinion and add some modifications during the analysis and design phases. Once the application was finished, it was deployed on an online server in its free version and shared with the users who saw the initial sketch, where they were able to carry out tests and give their vision of the application.

Tabla de contenidos

Capítulo 1. Introducción	9
1.1 Contexto	9
1.2 Motivación	9
1.3 Objetivos	9
1.3.1 Objetivos de la aplicación	10
1.3.2 Objetivos de formación	10
1.4 Estado de la cuestión	10
Capítulo 2. Requisitos	11
2.1 Especificación de requisitos	11
2.1.1 Tabla de requisitos funcionales:	11
2.1.4 Tabla de requisitos de información	13
2.1.5 Tabla de requisitos no funcionales:	14
2.1.6 Restricciones:	15
2.1.7 Diccionario de Datos	16
2.2 Casos de uso	18
2.2.1 Tabla de casos de uso	18
2.2.2 Descripción de los casos de uso	19
2.2.3 Matriz de trazabilidad:	40
2.2.3.1 Ejercicios	40
2.2.3.2 Sesión de entrenamiento	41
2.2.3.3 Usuarios	41
Capítulo 3. Plan de proyecto	43
3.1 Resumen del proyecto	43
3.1.1 Propósito, alcance y objetivos	43
3.1.2 Definiciones y acrónimos	43
3.1.3 Artefactos del proyecto	43
3.1.4 Evolución del Plan	44
3.2 Plan del proyecto	44
3.2.1 Ciclo de vida del proyecto	44
3.2.2 Plan de puesta en marcha	45
3.2.3 Plan de trabajo	46
3.2.3.1 Fase Inicio	46
3.2.3.2 Fase Elaboración	46
3.2.3.3 Fase Construcción	47
3.2.3.4 Fase Transición	47
3.2.4 Análisis de riesgos	48
3.2.5 Presupuesto	50
3.2.5.1 Estimación de costes del proyecto	51
3.2.5.2 Simulación de un proyecto real.	51

3.3 Seguimiento del plan de trabajo	53
3.3.1 Seguimiento de las Fases	53
3.3.1.1 Seguimiento de la Fase de Inicio	53
3.3.1.2 Seguimiento de la Fase de Elaboración:	53
3.3.1.3 Seguimiento de la fase de Construcción:	54
3.3.1.4 Seguimiento de la fase de Transición:	55
3.3.2 Seguimiento de riesgos	55
3.3.3 Seguimiento de presupuesto y costes reales del proyecto	55
Capítulo 4. Análisis	57
4.1 Descripción del modelo de usuario	57
4.2 Diagrama de casos de uso	57
4.2.1 Ejercicios	58
4.2.2 Sesión de entrenamiento	59
4.2.3 Usuarios	60
4.3 Modelado de dominio	61
Capítulo 5. Tecnologías utilizadas	62
5.1 Tecnologías utilizadas para el frontend	62
5.2 Tecnologías utilizadas para el backend	62
Capítulo 6. Diseño	65
6.1 Diseño centrado en el usuario	65
6.1.1 Bocetaje	65
6.1.2 Retroalimentación obtenida de los usuarios	65
6.1.3 Actuación acorde a la retroalimentación	66
6.2 Decisiones de diseño	66
6.3 Diseño del almacenamiento persistente	66
6.4 Diseño arquitectónico	67
6.4.1 Arquitectura propuesta	67
6.4.1.1 Arquitectura lógica	67
6.4.1.2 Despliegue	68
6.4.2 Arquitectura del cliente	69
6.4.3 Arquitectura del servidor	71
6.5 Diseño del API REST	72
6.6 Realización en diseño de casos de uso	73
6.6.1 Crear Ejercicio	73
6.6.2 Eliminar Ejercicio	75
6.7 Diseño detallado	76
6.7.1 Paquetes del cliente	76
6.7.2 Paquetes del Servidor	78
Capítulo 7. Implementación	83
7.1 Entorno de desarrollo	83

7.2 Implementación de la Base de Datos	83
7.3 Control de versiones	84
7.4 Organización del código	84
7.4.1 Parte del servidor:	84
7.4.2 Parte del cliente:	84
7.5 Solución de problemas	85
7.6 CI/CD	85
7.7 Patrones utilizados	86
7.7.1 Patrón MVC	86
7.7.2 Patrón DAO	86
7.7.3 Patrón ID	86
7.7.4 Patrón GRASP Alta Cohesión - Bajo Acoplamiento	87
Capítulo 8. Plan de pruebas y evaluación	89
8.1 Casos de prueba	89
8.1.1 Casos de uso de los ejercicios	89
8.1.2 Casos de uso de buscar ejercicio	90
8.1.3 Casos de uso de la sesión de entrenamiento	91
8.1.4 Casos de uso del usuario	93
8.2 Resultado de las pruebas	94
8.3 Pruebas con usuarios reales	96
Capítulo 9. Conclusiones y trabajo futuro	97
9.1 Conclusiones	97
9.1.1 Objetivos conseguidos:	97
9.1.2 Objetivos no conseguidos:	97
9.2 Trabajo futuro	97
9.3 Valoración personal	97
Capítulo 10. Manual de usuario	99
10.1 Primer contacto	100
10.1.1 Inicio	100
10.1.2 Registro	100
10.1.3 Acceso	101
10.2 Perfil y configuración del usuario	102
10.2.1 Perfil del usuario	102
10.2.2 Configuración del usuario	102
10.3 Crear ejercicio	103
10.3.1 Dibujar el ejercicio	103
10.3.2 Añadir información del ejercicio	104
10.4 Gestión de nuestros ejercicios	105
10.4.1 Trabajando con los ejercicios	106
10.4.2 Usando la herramienta de búsqueda	107

10.5	Uso de la sesión de entrenamiento	107
10.5.1	Añadir ejercicios a la sesión de entrenamiento	107
10.5.2	Definir el orden, la duración y las observaciones de los ejercicios	107
10.5.3	Añadir pausa	108
10.5.4	Exportar la sesión de entrenamiento	108
10.5.5	Resultado	109
10.6	Otros usuarios	109
10.6.1	Búsqueda de usuarios	109
10.6.2	Perfil de otro usuario	110
10.6.3	Ejercicios de otro usuario	112
10.7	Gestión de los usuarios seguidores y de los usuarios seguidos	112
10.7.1	Gestión de solicitudes	112
10.7.2	Gestión de listas de seguidores/seguídos	114
	Capítulo 11. Manual de Mantenimiento	117
11.1	Modificación del contenido	118
11.1.1	Modificación del frontend	118
11.1.2	Modificación del backend	118
11.2	Creación de contenido	118
11.2.1	Modificación del frontend	118
11.2.2	Modificación del backend	118
	Capítulo 12. Manual de despliegue de la aplicación	119
12.1	Despliegue en local	120
12.1.1	Herramientas necesarias	120
12.1.2	Configuración del proyecto	120
12.1.2.1	Instalaciones iniciales	120
12.1.2.2	Configuración del perfil de desarrollo	120
12.1.2.3	Configuración para la base de datos	120
12.1.3	Despliegue de la aplicación	121
12.1.3.1	Arranque de la aplicación	121
12.1.3.2	Poblar la base de datos	121
12.1.3.3	Acceso a la aplicación	121
12.2	Despliegue en producción	121
12.2.1	Herramientas y servicios necesarios	121
12.2.1.1	Crear aplicación en Heroku	121
12.2.1.2	Añadir el plugin de JawsDB	123
12.2.2	Configuración del proyecto	125
12.2.2.1	Configuración del perfil de producción	125
12.2.2.2	Configuración para la base de datos	126
12.2.2.3	Configuración para las llamadas del frontend	126
12.2.3	Despliegue del proyecto	126

12.2.3.1 Iniciar repositorio Git.	126
12.2.3.2 Desplegar la aplicación en el servidor	126
12.2.3.3 Poblar la base de datos	126
Bibliografía	127
Anexos	129

Anexos

Anexo I. Clasificación completa de las etiquetas	129
Anexo II. Simbología de las figuras de los ejercicios	130
Anexo III. Boceto en papel presentado a los usuarios	132
Anexo IV. Contenido del CD	144

Capítulo 1. Introducción

1.1 Contexto

Hoy en día, la tecnología ha logrado inmiscuirse en todos y cada uno de los aspectos de nuestra vida y en el deporte no iba a ser menos. Desde los avanzados sistemas de monitorización cardiaca de los deportistas profesionales, que les permiten explotar su cuerpo al límite, hasta los aficionados a correr por los parques, que pueden medir sus pulsaciones con una sencilla aplicación para el teléfono móvil.

Hoy vamos a hablar de la tecnología aplicada al deporte, pero no para los deportistas, sino para a una pieza fundamental que muchas veces queda en un segundo plano, pero es tan importante como ellos. No estamos hablando sino de los entrenadores, concretamente para este proyecto, de los entrenadores de baloncesto.

Los entrenadores tienen en sus despachos (o en las estanterías de sus casas quienes no sean profesionales) libros, revistas, artículos, cd's, etc., con ejercicios, jugadas y anotaciones que les ayudan y les permiten hacer mejor su trabajo. Si bien es verdad que internet en sí ha ayudado mucho a la difusión y compartición de este conocimiento, especialmente mediante blogs y foros, se echa en falta un lugar o alguna manera donde compartir fácilmente y de manera directa esta información. Así es como nace la idea de este proyecto.

El contexto de este proyecto tiene por finalidad servir como Trabajo de Fin de Grado de la Mención de Ingeniería del Software del Grado de Ingeniería Informática. Al tratarse de un proyecto software ha de seguir una serie de pasos estipulados para alcanzar el producto final, empezando por la planificación y el análisis y terminando por el diseño, la implementación y las pruebas, así como el seguimiento de cada una de ellas. Este documento tiene como fin recoger la información pertinente en relación a todas estas fases.

1.2 Motivación

Todos hemos dicho alguna vez ‘ojalá hubiera una app que...’, ‘sería genial que esta página web además permitiese...’, etc. Una de las principales ventajas de la carrera que hemos elegido es que nos permite tener los conocimientos, o al menos las nociones, para que esas frases dejen ser castillos en el aire y comienzan a ser líneas de código. Esta es precisamente la motivación de este proyecto.

En mi caso particular, llevo vinculado al mundo del baloncesto desde los ocho años hasta hoy, como jugador y los últimos, además, como entrenador y siempre he echado en falta aplicaciones que me permitieran gestionar las sesiones de entrenamiento que realizaba con los equipos que entrenaba. Estaba cansado de reescribir, los ejercicios que había usado en sesiones anteriores, o de buscar entre documentos dispersos las actividades que poder realizar. Había llegado el momento de dejarlo todo atrás, ponerse manos a la obra y hacer algo, no solo que sé que para mí será útil, sino algo que podría ayudar a muchos entrenadores y conseguir una formación baloncestística mejor para los futuros jugadores.

1.3 Objetivos

El objetivo principal de este proyecto es conseguir una aplicación web social que permita recoger ejercicios que practicar en las sesiones de baloncesto, dotarlos de ciertas etiquetas que permitan su filtrado y la generación de documentos externos donde exponerlos conjuntamente para conformar sesiones de entrenamiento.

1.3.1 Objetivos de la aplicación

- Permitir crear y almacenar ejercicios
- Permitir recuperar ejercicios almacenados utilizando filtros de búsqueda
- Agrupar ejercicios conformando una sesión de entrenamiento
- Exportar la sesión de entrenamiento a un documento externo
- Interacción entre usuarios
 - Los usuarios podrán seguirse unos a otros, siempre y cuando el seguido acepte la petición
 - Los usuarios dotarán de visibilidad a sus ejercicios, de tal manera que estos podrán ser accesibles para sus seguidores, para todos o para nadie más que él
- Desarrollar la aplicación compatible con los principales navegadores: Firefox y Chrome.
- Dotar la aplicación de una interfaz agradable a la vista y que la permita ser sencilla de entender y utilizar.

1.3.2 Objetivos de formación

- Aprendizaje y consolidación del framework Spring Boot para la creación de un API REST que actuará como backend.
- Estudio del framework Spring Security para dotar de seguridad el backend.
- Aprendizaje e investigación de JavaScript a través de la librería de jQuery para mostrar el frontend y realizar las llamadas al API.
- Llevar a cabo las fases y documentación del proyecto para habituarse al mundo laboral.

1.4 Estado de la cuestión

Esta aplicación jamás habría visto la luz de no haber existido otras anteriores que abrieran el camino, aunque a día de hoy hayan quedado obsoletas o sus interfaces no llegasen a ser lo suficientemente buenas para ser utilizadas fácilmente.

Algunos ejemplos de ellas son los que aparecen en la Tabla 1:

Nombre	Descripción
Basketball Playbook	Aplicación de escritorio de pago que permite dibujar y almacenar ejercicios y jugadas de baloncesto. Actualmente se encuentra en la versión 12. Puede usarse durante un periodo de 30 días de prueba.
BSM 2003	Aplicación de escritorio que permite dibujar jugadas y ejercicios y grabar los movimientos de los jugadores para recrear después la jugada. Obsoleta a día de hoy.
Basketball Coaching	Aplicación móvil para dibujar ejercicios y jugadas. Oficial de la FIBA.
FastDraw	Aplicación web más similar a la que queremos diseñar. Permite crear ejercicios y jugadas, recuperarlos y compartirlos con usuarios. Aplicación de pago con una cuota de suscripción anual considerablemente elevada

Tabla 1: Aplicaciones predecesoras

Capítulo 2. Requisitos

2.1 Especificación de requisitos

2.1.1 Tabla de requisitos funcionales:

ID	Nombre	Descripción	Prioridad
RF1	Gestión de ejercicios	El sistema deberá permitir gestionar los ejercicios.	ALTA
RF1.1	Crear ejercicio	El sistema deberá permitir que el usuario diseñe ejercicios y los almacene.	ALTA
RF1.2	Ver mis ejercicios	El sistema deberá permitir al usuario consultar los ejercicios que ha añadido.	ALTA
RF1.2.1	Modificar ejercicio	El sistema deberá permitir que el usuario trabaje sobre un ejercicio creado por él previamente, modificando su información y almacenando los cambios.	ALTA
RF1.3	Eliminar ejercicio	El sistema deberá permitir que el usuario elimine ejercicios que él mismo hubiera creado previamente.	ALTA
RF1.4	Buscar ejercicio	El sistema deberá permitir que el usuario encuentre ejercicios utilizando para ello una serie de parámetros de búsqueda preestablecidos (Ver Diccionario de datos: Parámetro de búsqueda).	ALTA
RF1.5	Exportar ejercicio	El sistema deberá permitir al usuario generar un documento que incluya la descripción del ejercicio. (ver restricción R4)	ALTA
RF1.6	Visibilizar ejercicio	El usuario deberá establecer una visibilidad al ejercicio. (Ver Diccionario de datos: Visibilidad de ejercicio).	ALTA
RF2	Gestión de la sesión de entrenamiento	El sistema deberá permitir que el usuario sea capaz de componer una sesión de entrenamiento que incluya los ejercicios que seleccione.	ALTA
RF2.1	Crear sesión de entrenamiento	El sistema deberá permitir la creación de una sesión de entrenamiento donde ir incluyendo ejercicios	ALTA
RF2.2	Añadir ejercicio a la sesión	El usuario deberá ser capaz de añadir ejercicios a la sesión de entrenamiento incluyendo su duración y una posible observación sobre el mismo.	ALTA
RF2.3	Eliminar ejercicio de la sesión	El usuario deberá ser capaz de eliminar un ejercicio que previamente haya incluido en la sesión de entrenamiento.	ALTA
RF2.4	Modificar orden de los ejercicios	El usuario deberá ser capaz de cambiar el orden de los ejercicios de la sesión.	ALTA
RF2.5	Añadir pausas a la sesión	El usuario deberá poder introducir un periodo de pausa (descanso) con una duración entre ejercicios.	ALTA
RF2.6	Exportar sesión de entrenamiento	El usuario deberá poder crear un documento que incluya los ejercicios especificados en la sesión.	ALTA
RF3	Gestión de relaciones entre usuarios	El sistema deberá permitir la gestión de relaciones entre usuarios	MEDIA

RF3.1	Buscar usuarios	El sistema deberá permitir que un usuario encuentre otros usuarios a través de su identificador.	MEDIA
RF3.2	Ver perfil de otros usuarios	El sistema permitirá a un usuario ver el perfil de otro, así como los ejercicios que éste decida mostrar.	MEDIA
RF3.3	Seguir a un usuario	El sistema permitirá que un usuario pueda ser seguido por otros de tal manera que éstos puedan ver los ejercicios los ejercicios protegidos. (Ver Diccionario de datos: Visibilidad de ejercicio) (Ver Diccionario de datos: Relación entre usuarios)	MEDIA
RF3.4	Dejar de seguir a un usuario	El sistema permitirá que un usuario deje de seguir a otro, perdiendo así la posibilidad de ver los ejercicios protegidos. (Ver Diccionario de datos: Visibilidad de ejercicio) (Ver Diccionario de datos: Relación entre usuarios)	MEDIA
RF3.5	Eliminar un seguidor	El sistema permitirá que un usuario saque de sus seguidores a un usuario concreto, de tal manera que este no pueda ver sus ejercicios protegidos salvo que se le vuelva a dar acceso para ello. (Ver Diccionario de datos: Visibilidad de ejercicio) (Ver Diccionario de datos: Relación entre usuarios)	MEDIA
RF3.6	Tomar prestados ejercicios de otros usuarios	El usuario podrá añadir a sus ejercicios los ejercicios que pueda ver de otros usuarios.	MEDIA
RF4	Gestión del propio usuario	El sistema deberá permitir la gestión del usuario	ALTA
RF4.1	Registrar usuario	El sistema deberá permitir que un usuario se registre.	ALTA
RF4.2	Iniciar sesión	El sistema deberá permitir que un usuario registrado inicie sesión.	ALTA
RF4.3	Modificar datos del usuario	El sistema deberá permitir que un usuario registrado modifique sus datos personales.	ALTA
RF4.4	Cerrar sesión	El sistema deberá permitir que un usuario cierre sesión una vez haya terminado de utilizar la aplicación.	ALTA

Tabla 2: Requisitos funcionales

2.1.4 Tabla de requisitos de información

ID	Nombre	Descripción	Prioridad
RI1	Ejercicio	El sistema deberá almacenar información sobre cada ejercicio, en particular: título, descripción, figura, número mínimo y máximo de jugadores, enlace(url) a descripción externa del ejercicio, autor, etiquetas	ALTA
RI2	Etiqueta	El sistema deberá almacenar información sobre cada tipo de etiqueta. (Ver Diccionario de datos: Etiqueta)	ALTA
RI3	Categorías de visibilidad	El sistema deberá almacenar información sobre la visibilidad de cada ejercicio, en particular, si se trata de un ejercicio público, privado o protegido. (Ver Diccionario de datos: Visibilidad de ejercicio)	ALTA
RI4	Relaciones entre usuario	El sistema deberá almacenar información sobre las relaciones entre usuarios, concretamente si se siguen o no. (Ver Diccionario de datos: Relación entre usuarios)	MEDIA
RI5	Sesión de entrenamiento	El sistema deberá almacenar información relativa a la sesión de entrenamiento, en particular: la fecha, la hora, la duración, el lugar, los objetivos y los ejercicios del entrenamiento. (Ver Diccionario de datos: Sesión de entrenamiento)	ALTA
RI6	Usuario	El sistema deberá almacenar información relativa a un usuario, en particular: alias, nombre y sus ejercicios. (Ver Diccionario de datos: Alias) (Ver Diccionario de datos: Nombre) (Ver Diccionario de datos: Ejercicio)	ALTA

Tabla 3: Requisitos de información

2.1.5 Tabla de requisitos no funcionales:

Se utilizará el modelo FURPS+ para definir los requisitos no funcionales. FURPS+ es el acrónimo de Functionality, Usability, Reliability, Performance y Support:

- **Funcionalidad** (Functionality): descritos en el apartado de requisitos funcionales.
- **Usability** (Usabilidad): características relacionadas con todo aquello que tenga que ver con la estética y la interfaz de usuario.
- **Reliability** (Confiabilidad): características correspondientes al comportamiento de la aplicación con respecto a la disponibilidad y a la recuperación frente a fallos.
- **Performance** (Rendimiento): características que determinan la productividad, como, por ejemplo, el tiempo de respuesta.
- **Support** (Soporte): características que tienen que ver con la mantenibilidad, escalabilidad, instalabilidad, etc., de la aplicación.

ID	Nombre	Descripción
RNF1	Uso de la aplicación	La aplicación será utilizada a través de un navegador web.
RNF2	Formato UTF8	El sistema deberá utilizar el formato de codificación de caracteres UTF-8
RNF3	Dispositivos soportados	El sistema será utilizado sobre ordenadores, quedando los dispositivos móviles y las tabletas para un futuro.
RNF4	Navegadores soportados	El sistema se podrá utilizar en los navegadores Google Chrome y Mozilla Firefox
RNF5	Usabilidad (Facilidad de uso)	La aplicación será fácil de utilizar, asegurando que el 90% de los usuarios sepan utilizarla completa y correctamente en su tercer inicio de sesión
RNF6	Confiabilidad (Disponibilidad)	La aplicación estará disponible todos los días de 8.00 a 24.00.
RNF7	Rendimiento (Velocidad Tiempo de respuesta)	La aplicación tardará menos de 5 segundos en responder cada interacción del usuario.
RNF8	Soporte (Instalabilidad)	La aplicación no requerirá de ninguna instalación por parte del usuario, salvo la del propio navegador.
RNF9	Idioma	La aplicación estará disponible en castellano, pudiendo ser traducida a otros idiomas en el futuro.
RNF10	Seguridad	La aplicación contará con un sistema simple de acceso mediante usuario y contraseña
RNF11	Privacidad	La aplicación se compromete a no exportar datos de usuario a terceros.

Tabla 4: Requisitos no funcionales

2.1.6 Restricciones:

ID	Nombre	Descripción
R1	Acceso a la aplicación	No se podrá acceder a la aplicación si no se está registrado.
R2	Búsqueda usuario	Cualquier usuario podrá buscar a otro a través de su identificador. Podrán verse todos los ejercicios públicos de ese usuario, pero sólo podrán verse los ejercicios protegidos si el usuario buscado ha aceptado como seguidor al usuario buscador.
R3	Visibilidad de ejercicios	En el momento que un usuario cambie la visibilidad de un ejercicio a otra más restrictiva, los usuarios que se hubieran copiado el ejercicio dejarán de tener acceso a él y no aparecerá en sus búsquedas. Si el usuario autor decide ponerlo de nuevo con una visibilidad accesible para el usuario que quiere copiar el ejercicio, entonces podrá copiarlo de nuevo.
R4	Exportación de ejercicios y sesiones de entrenamiento a documentos	Todo aquello que se permita exportar se hará en formato PDF.

Tabla 5: Restricciones

2.1.7 Diccionario de Datos

En esta parte del documento, entre la Tabla 6 y la Tabla 13, se especificarán los conceptos que necesiten de un mayor nivel de comprensión del que se puede ofrecer en las tablas de requisitos.

Nombre	Ejercicio
Descripción	Cada una de las partes en las que está compuesta una sesión real de un entrenamiento de un deporte específico (en este caso baloncesto).
Campos	Un ejercicio debe especificar los siguientes campos: <u>Título</u> : nombre que permita identificar el ejercicio fácilmente <u>Descripción</u> : descripción e instrucciones del propósito del ejercicio <u>Número de participantes</u> : número de jugadores que desarrollarán el ejercicio. <u>mínimo</u> : número mínimo necesario para realizar el ejercicio. <u>máximo</u> : número máximo recomendado para realizar el ejercicio <u>Figura</u> : imagen conformada por nodos y flechas que refleja el comportamiento especificado en la descripción

Tabla 6: Definición de Ejercicio

Nombre	Sesión de entrenamiento
Descripción	Conjunto de ejercicios que se realizarán durante un entrenamiento.
Campos	Fecha y hora del entrenamiento Objetivo del entrenamiento: propósito que persigue la realización específica de los ejercicios incluidos. Lista de ejercicios: ejercicios a realizar (ver definición de ejercicio) y, además: Duración: tiempo estipulado para la realización Observaciones: detalles a mayores que quiera incluir el usuario para la realización del objetivo

Tabla 7: Definición Sesión de entrenamiento

Nombre	Etiqueta
Descripción	Propósito general que perseguirá el ejercicio
Posibles valores	Ver anexo I

Tabla 8: Definición de etiqueta

Nombre	Visibilidad de un ejercicio
Descripción	El autor de un ejercicio puede definir la visibilidad que le da al mismo.
Posibles valores	Valores ordenados de menor a mayor visibilidad: <u>Privado</u> : nadie, salvo su autor, podrá tener acceso al ejercicio <u>Protegido</u> : sólo el autor y los usuarios que le sigan podrán tener acceso al ejercicio. <u>Público</u> : todo usuario podrá tener acceso al ejercicio.

Tabla 9: Definición de visibilidad

Nombre	Seguir a un usuario
Descripción	El usuario seguidor podrá ver los ejercicios no privados del usuario seguido. Misma filosofía que los usuarios protegidos de Twitter, donde sólo pueden ver las publicaciones si se es aceptado. Seguir no es una operación bidireccional, un usuario puede seguir a otro sin que el otro le siga a él.

Tabla 10: Definición de seguir a un usuario

Nombre	Relación entre usuarios
Descripción	Un usuario respecto a otro podrá seguirle o no.
Posibles valores	<u>Siguiendo</u> : el usuario, si es aceptado, podrá ver los ejercicios protegidos y públicos del usuario seguido. <u>No siguiendo</u> : el usuario, podrá ver únicamente los ejercicios públicos del otro usuario.

Tabla 11: Definición de relación entre usuarios

Nombre	Ejercicio tomado prestado
Descripción	Aquel ejercicio que habiendo sido creado por un usuario es utilizado por otro. El usuario que lo toma prestado puede utilizarlo para sus sesiones, pero no puede modificarlo ni borrarlo del sistema, quedando esas operaciones exclusivamente para el autor. El autor se reserva el derecho de dejar de compartir el ejercicio. Si un usuario elimina de entre sus seguidores a un usuario seguidor, este dejará de tener acceso a los ejercicios que pudiera haber tomado prestados del seguidor. Así mismo, si el usuario cambia la visibilidad a una más restrictiva los usuarios con ese ejercicio tomado como prestado podrían perder el acceso a él si no cumplen con las restricciones de la nueva visibilidad.

Tabla 12: Definición de ejercicio prestado

Nombre	Solicitud de seguimiento
Descripción	Solicitud enviada por un usuario emisor a un usuario receptor con el fin de ser admitido entre sus seguidores.
Posibles valores	<u>Pendiente</u> : estado inicial. No ha sido ni aceptada, ni rechazada <u>Aceptada</u> : estado final. El usuario receptor acepta al emisor entre sus seguidores <u>Rechazada</u> : estado final. El usuario receptor no admite al emisor como seguidor.

Tabla 13: Definición de solicitud de seguimiento

2.2 Casos de uso

Se recoge un listado de los casos de uso existentes, en la Tabla 14, y la descripción de los mismos entre la Tabla 15 y la Tabla 45.

2.2.1 Tabla de casos de uso

Identificador	Nombre del caso de uso	Prioridad
CU-E01	Ver mis ejercicios	ALTA
CU-E02	Crear un ejercicio	ALTA
CU-E03	Seleccionar ejercicio	ALTA
CU-E04	Modificar Ejercicio	ALTA
CU-E05	Modificar Visibilidad	ALTA
CU-E06	Eliminar ejercicio	ALTA
CU-E07	Exportar ejercicio	ALTA
CU-E08	Buscar ejercicio	ALTA
CU-S01	Ver la sesión de entrenamiento actual	ALTA
CU-S02	Añadir ejercicio a la sesión entrenamiento	ALTA
CU-S03	Añadir pausa	ALTA
CU-S04	Modificar orden ejercicios sesión	ALTA
CU-S05	Eliminar ejercicio de la sesión de entrenamiento	ALTA
CU-S06	Añadir duración y observaciones a un ejercicio de la sesión	ALTA
CU-S07	Exportar sesión de entrenamiento	ALTA
CU-U01	Iniciar sesión	ALTA
CU-U02	Cerrar sesión	ALTA
CU-U03	Registrarse	ALTA
CU-U04	Modificar datos personales	ALTA
CU-U05	Ver perfil propio	ALTA
CU-U06	Buscar Usuario	MEDIA
CU-U07	Ver perfil de otro usuario	MEDIA
CU-U08	Aceptar solicitud de seguimiento	MEDIA
CU-U09	Rechazar solicitud de seguimiento	MEDIA
CU-U10	Cancelar solicitud de seguimiento	MEDIA
CU-U11	Enviar solicitud de seguimiento a otro usuario	MEDIA
CU-U12	Dejar de seguir usuario	MEDIA
CU-U13	Eliminar de mis seguidores	MEDIA
CU-U14	Ver ejercicios de otro usuario	MEDIA
CU-U15	Tomar prestado ejercicio	MEDIA

2.2.2 Descripción de los casos de uso

Ejercicios:

CU-E01	Ver mis ejercicios	
Versión	1.0	
Descripción	El usuario obtendrá una lista con los ejercicios que él haya creado o tomado prestados de otro usuario.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' selecciona la opción “Ver mis ejercicios”
	2	El sistema muestra una lista con los ejercicios que él mismo ha desarrollado y aquellos que ha tomado prestado.
Postcondición	El usuario obtendrá una lista con los ejercicios que él haya creado	
Excepciones	Paso	Acción
	2a	Si el usuario no ha creado (ni tomado prestado) ningún ejercicio, el sistema mostrará un mensaje indicando que aún no hay ejercicios añadidos.
Comentarios		

Tabla 15: Caso de uso “Ver mis ejercicios”

CU-E02	Crear un ejercicio	
Versión	1.0	
Descripción	El usuario tendrá la posibilidad de añadir ejercicios a su base de datos personal. Los ejercicios podrán contarán con una descripción, gráficos ilustrativos, número mínimo y máximo de participantes y etiquetas.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	

Secuencia normal	Paso	Acción
	1	El actor 'Usuario' selecciona la opción de "Crear nuevo ejercicio"
	2	El sistema muestra una nueva pantalla con las herramientas necesarias para crear un nuevo ejercicio.
	3	El actor 'Usuario' diseña el ejercicio incluyendo todos los campos necesarios para ello e indica la opción de guardar. Los campos son: Título Descripción Número mínimo de participantes Número máximo recomendado de participantes Etiquetas (seleccionar al menos una) Visibilidad (por defecto Privado)
4	El sistema comprueba que el usuario ha añadido todos los campos obligatorios y almacena el ejercicio en la base de datos. El sistema muestra la posibilidad de registrar otro ejercicio.	
Postcondición	El ejercicio quedará almacenado y asignado el usuario como autor.	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	4a	El sistema comprueba que el usuario no ha incluido alguno de los campos obligatorios, se lo notifica al usuario y el paso de uso continúa en el paso 3
Comentarios		

Tabla 16: Caso de uso "Crear un ejercicio"

CU-E03	Seleccionar ejercicio	
Versión	1.0	
Descripción	El usuario escoge un ejercicio entre los presentados para realizar una opción con él.	
Actores	Usuario	
Dependencias	CU-E08, CU-E01	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El sistema ha presentado una lista de ejercicios	
Secuencia normal	1	El actor 'Usuario' selecciona uno de los ejercicios presentados
	2	El ejercicio queda seleccionado para poder realizar distintas operaciones
Postcondición	El ejercicio queda seleccionado para poder realizar alguna opción con él.	

Excepciones	Paso	Acción
Comentarios		

Tabla 17: Caso de uso "Seleccionar ejercicio"

CU-E04	Modificar Ejercicio	
Versión	2.0	
Descripción	El usuario tendrá la posibilidad de modificar los ejercicios que haya diseñado previamente.	
Actores	Usuario	
Dependencias	CU-E03	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario debe haber accedido a sus ejercicios	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Seleccionar ejercicio"
	2	El actor 'Usuario' selecciona la opción "Modificar ejercicio" sobre uno de ellos
	3	El sistema muestra una nueva pantalla con las herramientas necesarias para modificar el ejercicio.
	4	El actor 'Usuario' modifica el ejercicio incluyéndole todos los campos necesarios para ello y selecciona la acción de guardar.
	5	El sistema comprueba que el usuario ha añadido todos los campos obligatorios (mismos que en el caso de uso CU-E02). El sistema muestra la opción de seguir modificando.
	6	El actor 'Usuario' descarta seguir modificando el ejercicio.
Postcondición	El ejercicio quedará modificado y almacenado en la base de datos.	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	5a	El sistema comprueba que el usuario no ha incluido alguno de los campos obligatorios, se lo notifica al usuario y el paso de uso continúa en el paso 3
	6a	El usuario confirma y el caso de uso continúa en el paso 3.
Comentarios		

Tabla 18: Caso de uso "Modificar ejercicio"

CU-E05	Modificar Visibilidad	
Versión	2.0	
Descripción	El usuario podrá cambiar la visibilidad entre los tipos posibles para un ejercicio	
Actores	Usuario	
Dependencias	CU-E03	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario está viendo sus ejercicios.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Modificar ejercicio"
	2	El actor 'Usuario' cambia su visibilidad y selecciona la acción guardar.
	3	El sistema almacena el cambio de visibilidad
Postcondición	La visibilidad del ejercicio cambia, afectando a los usuarios que hubieran copiado el ejercicio impidiéndoles verlo en caso de que la nueva visibilidad sea más restrictiva.	
Excepciones	Paso	Acción
Comentarios		

Tabla 19: Caso de uso "Modificar visibilidad"

CU-E06	Eliminar ejercicio	
Versión	1.0	
Descripción	El usuario tendrá la posibilidad de eliminar los ejercicios que haya diseñado previamente.	
Actores	Usuario	
Dependencias	CU-E03	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario está viendo sus ejercicios.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Seleccionar ejercicio"

	2	El actor 'Usuario' selecciona la opción "Eliminar ejercicio"
	3	El sistema muestra un mensaje de confirmación advirtiendo de que no podrá recuperarse el ejercicio si sigue adelante con el caso de uso.
	4	El actor 'Usuario' confirma que desea realizar la eliminación del ejercicio.
	5	El sistema elimina el ejercicio.
	6	
	7	
Postcondición	El ejercicio quedará eliminado del sistema, perdiendo acceso al mismo el autor y todos aquellos que pudieran verlo.	
Excepciones	Paso	Acción
	4a	El actor usuario no confirma y el caso de uso queda sin efecto.
Comentarios		

Tabla 20: Caso de uso "Eliminar ejercicio"

CU-E07	Exportar ejercicio	
Versión	1.0	
Descripción	El usuario generará un documento con la descripción del ejercicio.	
Actores	Usuario	
Dependencias	CU-E03	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario está viendo sus ejercicios	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Seleccionar ejercicio"
	2	El actor 'Usuario' selecciona un ejercicio y elige la acción: "exportar a PDF".
	3	El sistema abre una ventana para que el usuario elija la ruta destino.
	4	El actor 'Usuario' selecciona la ruta y la acción de guardar
	5	El sistema genera el PDF y lo guarda en la ruta especificada.

Postcondición	Se genera un documento con la información del ejercicio	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
Comentarios		

Tabla 21: Caso de uso "Exportar ejercicio"

CU-E08	Buscar ejercicio	
Versión	1.0	
Descripción	El usuario debe poder obtener una lista de ejercicios en función de una serie de parámetros de búsqueda que haya introducido.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' selecciona la opción de "Buscar ejercicios"
	2	El sistema muestra una nueva ventana con campos necesarios para que el usuario introduzca los parámetros de búsqueda. Podrá buscarse por: Título Cualquiera de las etiquetas Número de jugadores Espacio utilizado
	3	El actor 'Usuario' introduce los parámetros de búsqueda que considere oportuno y pulsa sobre el botón de buscar.
	4	El sistema realiza el filtrado de ejercicios en función de los parámetros introducidos por el usuario y muestra una lista con el título y la descripción de los ejercicios.
Postcondición	El usuario obtiene una lista de ejercicios acordes a su búsqueda.	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	4a	Si no hay ejercicios acordes a los parámetros introducidos, el sistema mostrará un mensaje indicando que no hay ejercicios acordes a la búsqueda realizada.

Comentarios	
-------------	--

Tabla 22: Caso de uso "Buscar ejercicio"

Sesión Entrenamiento

CU-S01	Ver la sesión de entrenamiento actual	
Versión	1.0	
Descripción	El usuario podrá ver los ejercicios incluidos a la sesión de entrenamiento	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la opción "Sesión de entrenamiento"
	2	El sistema muestra una lista con los ejercicios que el usuario ha ido incluyendo y otras opciones.
Postcondición	El sistema muestra el estado de la sesión de entrenamiento	
Excepciones	Paso	Acción
	2a	Si no hay ejercicios introducidos, el sistema mostrará un mensaje indicando que no hay ejercicios añadidos a la sesión
Comentarios		

Tabla 23: Caso de uso "Ver la sesión de entrenamiento actual"

CU-S02	Añadir ejercicio a la sesión entrenamiento	
Versión	1.0	
Descripción	El usuario podrá incluir ejercicios a una sesión de entrenamiento, pudiendo generar un documento con los ejercicios que haya decidido incluir	
Actores	Usuario	
Dependencias	CU-E03	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario debe haber buscado previamente el ejercicio.	

Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Seleccionar ejercicio"
	2	El actor 'Usuario' elige la acción "Añadir a la sesión de entrenamiento"
	3	El sistema añade el ejercicio a la sesión de entrenamiento.
Postcondición	El ejercicio queda añadido a la sesión de entrenamiento	
Excepciones	Paso	Acción
Comentarios		

Tabla 24: Caso de uso "Añadir ejercicio a la sesión de entrenamiento"

CU-S03	Añadir pausa	
Versión	2.0	
Descripción	El usuario añade una pausa de duración limitada junto a los ejercicios.	
Actores	Usuario	
Dependencias	CU-S01	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Ver la sesión de entrenamiento actual"
	2	El actor 'Usuario' elige la acción "Añadir pausa".
	3	El sistema añade una pausa al final de la lista de ejercicios.
Postcondición	Queda situada una pausa en el lugar elegido	
Excepciones	Paso	Acción
Comentarios		

Tabla 25: Caso de uso "Añadir pausa"

CU-S04	Modificar orden ejercicios sesión	
Versión	1.0	
Descripción	El usuario altera la disposición inicial de los ejercicios (y pausas).	
Actores	Usuario	
Dependencias	CU-S01	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. Existen ejercicios introducidos en la sesión.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Ver la sesión de entrenamiento actual"
	2	El actor 'Usuario' elige un ejercicio y la posición en la que quiera colocarlo.
	3	El sistema recoge el cambio de orden de los ejercicios.
Postcondición	El orden de los ejercicios habrá cambiado	
Excepciones	Paso	Acción
Comentarios		

Tabla 26: Caso de uso "Modificar orden ejercicio sesión"

CU-S05	Eliminar ejercicio de la sesión de entrenamiento	
Versión	1.0	
Descripción	El usuario elimina un ejercicio que había introducido previamente.	
Actores	Usuario	
Dependencias	CU-S01	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. Existen ejercicios introducidos en la sesión.	

Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Ver la sesión de entrenamiento actual".
	2	El actor 'Usuario' selecciona y elimina un ejercicio de la lista.
	3	El sistema elimina el ejercicio sin pedir confirmación el ejercicio.
Postcondición	El ejercicio queda eliminado y la sesión reduce el número de ejercicios en uno.	
Excepciones	Paso	Acción
Comentarios		

Tabla 27: Caso de uso "Eliminar ejercicio de la sesión de entrenamiento"

CU-S06	Añadir duración y observaciones a un ejercicio de la sesión	
Versión	1.0	
Descripción	El usuario podrá establecer una duración en minutos a cada uno de los ejercicios introducidos en la sesión de entrenamiento y, si lo desea, una observación que permita entender el desarrollo del ejercicio más claramente.	
Actores	Usuario	
Dependencias	CU-S01	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario debe haber introducido al menos un ejercicio en la sesión de entrenamiento	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Ver la sesión de entrenamiento actual".
	2	El actor 'Usuario' selecciona un ejercicio de entre los de la sesión y cambia el valor del campo duración de cero a otro valor superior.
	3	El sistema almacena el cambio en la duración
	4	El actor 'Usuario' selecciona un ejercicio de entre los de la sesión y cambia el valor del campo observación introduciendo una explicación de hasta 200 caracteres.
	5	El sistema almacena el cambio en la duración
Postcondición	Se obtiene un documento con la sesión del entrenamiento.	
Excepciones	Paso	Acción

	2a	Si el actor 'Usuario' introduce un valor inferior o mantiene el cero, el sistema muestra un mensaje de error y el caso de uso continúa en el paso 2.
Comentarios		

Tabla 28: Caso de uso "Añadir duración y observaciones a un ejercicio de la sesión"

CU-S07	Exportar sesión de entrenamiento	
Versión	2.0	
Descripción	El usuario podrá generar un documento con los ejercicios de la sesión e información sobre la misma	
Actores	Usuario	
Dependencias	CU-S01	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Ver la sesión de entrenamiento actual".
	2	El actor 'Usuario' selecciona la opción "Exportar sesión de entrenamiento".
	3	El sistema comprueba que todos los ejercicios tienen duración y solicita una fecha y un inicio de la hora de sesión, así como el nombre del equipo que llevará a cabo el entrenamiento.
	4	El actor 'Usuario' introduce la información solicitada y pulsa sobre el botón generar.
	5	El sistema comprueba que la información ha sido introducida y abre una ventana para que el usuario elija la ruta destino.
	6	El actor 'Usuario' selecciona la ruta y la acción de guardar
	7	El sistema genera el PDF y lo guarda en la ruta especificada. El sistema muestra un mensaje ofreciendo al usuario la posibilidad de eliminar la información de la sesión que acaba de ser generada.
	8	El actor 'Usuario' confirma
9	El sistema elimina todos los ejercicios de la sesión mostrando la lista vacía.	
Postcondición	Se obtiene un documento con la sesión del entrenamiento.	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.

	3b	El sistema comprueba que no todos los ejercicios tienen una duración asignada e informa al usuario. El caso de uso continúa en el paso 1.
	4b	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	5a	El sistema detecta que no todos los campos obligatorios han sido rellenados y el caso de uso continúa en el paso 3.
	8a	El actor 'Usuario' descarta la opción y no se elimina la sesión pudiéndose exportar de nuevo un documento.
Comentarios		

Tabla 29: Caso de uso "Exportar sesión entrenamiento"

Usuarios:

CU-U01	Iniciar sesión	
Versión	1.0	
Descripción	El usuario inicia sesión en la aplicación	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige "Iniciar sesión"
	2	El sistema solicita el identificador y la contraseña del usuario
	3	El actor 'Usuario' introduce ambos campos
	4	El sistema comprueba que los campos han sido rellenados correctamente y muestra la página de inicio del usuario en la aplicación.
Postcondición	El usuario ha iniciado sesión, ve su página principal y se ha creado una sesión de entrenamiento.	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	4a	El sistema detecta que los campos no se han introducido correctamente, muestra un mensaje de error y el caso de uso continúa en el paso 3.
Comentarios		

Tabla 30: Caso de uso "Iniciar sesión"

CU-U02	Cerrar sesión	
Versión	2.0	
Descripción	El usuario crea una cuenta en la aplicación	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. La sesión está en vigor.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la opción "Cerrar sesión"
	2	El sistema desconecta al actor 'Usuario', haciendo que ya no esté autenticado e imposibilitando de su acceso a las páginas restringidas si no inicia sesión de nuevo.
Postcondición	La sesión queda cerrada.	
Excepciones	Paso	Acción
Comentarios		

Tabla 31: Caso de uso "Cerrar sesión"

CU-U03	Registrarse	
Versión	1.0	
Descripción	El usuario crea una cuenta en la aplicación	
Actores	Usuario	
Dependencias		
Precondición		
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la opción "Registrarse"
	2	El sistema muestra un formulario con los campos necesarios para registrarse. Dichos campos son:

		Nombre Alias Email Contraseña
	3	El actor 'Usuario' rellena los campos
	4	El sistema comprueba que los campos han sido rellenos correctamente y registra al usuario en la aplicación.
	5	
	6	
Postcondición	El usuario queda registrado en el sistema.	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	4a	El sistema detecta que los campos no se han introducido correctamente, muestra un mensaje de error y el caso de uso continúa en el paso 3.
	4b	El sistema detecta que el alias utilizado para registrarse ya ha sido utilizado. Muestra un mensaje informando al usuario y el caso de uso continúa en el paso 3.
Comentarios		

Tabla 32: Caso de uso "Registrarse"

CU-U04	Modificar datos personales	
Versión	1.0	
Descripción	El usuario desea cambiar los datos que tiene el sistema sobre él.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la opción de "Modificar datos personales"
	2	El sistema muestra un formulario con los datos del usuario y solicita la modificación. Datos personales del usuario: Nombre Tiempo entrenando Nivel del entrenador Club donde entrena Frase sobre el entrenador

	3	El actor 'Usuario' modifica los campos que desee.
	4	El sistema comprueba que los campos han sido rellenos correctamente y modifica los datos del usuario en la aplicación.
Postcondición	Los campos modificados se registran en el sistema	
Excepciones	Paso	Acción
	3a	El actor 'Usuario' decide no continuar y el caso de uso queda sin efecto.
	4a	El sistema detecta que los campos no se han introducido correctamente, muestra un mensaje de error y el caso de uso continúa en el paso 3.
Comentarios		

Tabla 33: Caso de uso "Modificar datos personales"

CU-U05	Ver perfil propio	
Versión	1.0	
Descripción	El usuario accede a su propio perfil donde puede ver su información, sus ejercicios y los ajustes de configuración.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' pulsa sobre "Inicio"
	2	El sistema muestra el perfil del usuario. El perfil mostrará: Datos personales del usuario (ver CU-U04) Acceso a los ejercicios Acceso a los usuarios seguidores Acceso a los usuarios seguidos
Postcondición	El sistema muestra el perfil del usuario	
Excepciones	Paso	Acción
Comentarios		

Tabla 34: Caso de uso "Ver perfil propio"

CU-U06	Buscar Usuario	
Versión	2.0	
Descripción	El usuario podrá buscar otros usuarios a través de su identificador de usuario.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' introduce el identificador del usuario que quiere buscar
	2	El sistema muestra una lista de posibles candidatos.
Postcondición	El usuario accede al perfil del usuario que busca	
Excepciones	Paso	Acción
	2a	El sistema no encuentra candidatos y el caso de uso continúa en el paso 1.
Comentarios		

Tabla 35: Caso de uso “Buscar usuario”

CU-U07	Ver perfil de otro usuario	
Versión	1.0	
Descripción	El usuario accede al perfil de otro usuario donde podrá ver su información y los ejercicios que comparta.	
Actores	Usuario	
Dependencias	CU-U06	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso “Buscar usuario”
	2	El actor 'Usuario' elige la acción “Ver perfil” del usuario a quien quiere visitar

	3	El sistema muestra el perfil del usuario. (ver CU-U05)
Postcondición	El sistema muestra el perfil del usuario al que se ha accedido	
Excepciones	Paso	Acción
Comentarios		

Tabla 36: Caso de uso “Ver perfil de otro usuario”

CU-U08	Aceptar solicitud de seguimiento	
Versión	1.0	
Descripción	El usuario ha recibido una solicitud de otro usuario para poder ver sus ejercicios protegidos y la acepta.	
Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario ha recibido una solicitud	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la acción “Aceptar solicitud”
	2	El sistema añade al solicitante a la lista de seguidores del solicitado. La solicitud cambia su estado a “Aceptada”.
Postcondición	El usuario que mandó la solicitud se añade a la lista de seguidores del usuario. La solicitud cambia su estado a “Aceptada”.	
Excepciones	Paso	Acción
Comentarios		

Tabla 37: Caso de uso “Aceptar solicitud de seguimiento”

CU-U09	Rechazar solicitud de seguimiento	
Versión	1.0	
Descripción	El usuario ha recibido una solicitud de otro usuario para poder ver sus ejercicios protegidos y la rechaza.	

Actores	Usuario	
Dependencias		
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario ha recibido una solicitud	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la acción "Rechazar solicitud"
	2	El sistema no añade al solicitante a la lista de seguidores del solicitado. La solicitud cambia su estado a "Rechazada".
Postcondición	El usuario que mandó la solicitud no se añade a la lista de seguidores del usuario. La solicitud cambia su estado a "Rechazada".	
Excepciones	Paso	Acción
Comentarios		

Tabla 38: Caso de uso "Rechazar solicitud de seguimiento"

CU-U10	Cancelar solicitud de seguimiento	
Versión	1.0	
Descripción	El usuario ha enviado una solicitud a otro usuario para poder ver sus ejercicios protegidos y quiere descartar esa solicitud.	
Actores	Usuario	
Dependencias	CU-U11	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario ha enviado una solicitud	
Secuencia normal	Paso	Acción
	1	El actor 'Usuario' elige la acción "Cancelar solicitud"
	2	El sistema registra el cambio y la solicitud cambia su estado a "Rechazada".
Postcondición	El usuario que mandó la solicitud no podrá ser aceptado como seguidor. Podrá enviar otra más adelante si lo desea.	
Excepciones	Paso	Acción

Comentarios	
-------------	--

Tabla 39: Caso de uso “Cancelar solicitud de seguimiento”

CU-U11	Enviar solicitud de seguimiento a otro usuario	
Versión	1.0	
Descripción	El usuario envía una solicitud de seguimiento a otro para poder ver los ejercicios protegidos que publique.	
Actores	Usuario	
Dependencias	CU-U07	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario accede al perfil de otro usuario	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso “Ver perfil de otro usuario”
	2	El actor 'Usuario' selecciona la acción “Seguir”
	3	El sistema envía una petición al usuario solicitado.
Postcondición	El sistema registra la solicitud y la almacena como pendiente hasta que el usuario solicitado la acepte o la rechace.	
Excepciones	Paso	Acción
	2a	Si el usuario ya ha mandado la petición, aparecerá en su lugar la acción “Cancelar solicitud”
Comentarios		

Tabla 40: Caso de uso “Enviar solicitud de seguimiento a otro usuario”

CU-U12	Dejar de seguir usuario	
Versión	1.0	
Descripción	El usuario dejar de seguir a otro, no viendo más sus ejercicios protegidos.	
Actores	Usuario	
Dependencias	CU-U07	

Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario accede al perfil de otro usuario	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso “Ver perfil de otro usuario”
	2	El actor 'Usuario' elige la acción “Dejar de seguir”
	3	El sistema solicita confirmación.
	4	El actor 'Usuario' confirma la operación.
	5	El sistema elimina al usuario seguidor de la lista de seguidores del usuario.
6		
Postcondición	El usuario buscado queda excluido de la lista de seguidos del usuario buscador	
Excepciones	Paso	Acción
	4a.	El actor ‘Usuario’ no confirma y la operación queda sin efecto.
Comentarios		

Tabla 41: Caso de uso “Dejar de seguir usuario”

CU-U13	Eliminar de mis seguidores	
Versión	1.0	
Descripción	El usuario elimina a uno de los usuarios que le siguen.	
Actores	Usuario	
Dependencias	CU-U07	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario a eliminar debe ser seguidor. El usuario accede al perfil del usuario a eliminar.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso “Ver perfil de otro usuario”
	2	El actor 'Usuario' elige la acción "Eliminar de mis seguidores"
	3	El sistema solicita confirmación.
4	El actor 'Usuario' confirma la operación.	

	5	El sistema elimina al usuario seguidor de la lista de seguidores del usuario.
Postcondición	El usuario seguidor ya no sigue al usuario y no puede ver sus ejercicios protegidos.	
Excepciones	Paso	Acción
	4a.	El actor 'Usuario' no confirma y la operación queda sin efecto.
Comentarios		

Tabla 42: Caso de uso "Eliminar de mis seguidores"

CU-U14	Ver ejercicios de otro usuario	
Versión	1.0	
Descripción	El usuario accede a los ejercicios creados por otro usuario.	
Actores	Usuario	
Dependencias	CU-U07	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario accede al perfil del usuario a eliminar.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso "Ver perfil de otro usuario"
	2	El actor 'Usuario' elige la acción "Ver ejercicios"
	3	El sistema muestra una lista con los ejercicios públicos (y protegidos si el usuario del que se quiere ver los ejercicios ha aceptado la petición de seguimiento) del usuario.
Postcondición	El usuario accede a los ejercicios del otro usuario	
Excepciones	Paso	Acción
	3a	Si el usuario no hubiera creado ejercicios, se mostraría un mensaje indicando que el usuario visitado no tiene ejercicios disponibles.
Comentarios		

Tabla 43: Caso de uso "Ver ejercicios de otro usuario"

CU-U15	Tomar prestado ejercicio	
Versión	1.0	

Descripción	El usuario toma prestado un ejercicio de creado por otro usuario de tal manera que aparecerá en sus búsquedas y podrá utilizarlo en sus sesiones de entrenamiento.	
Actores	Usuario	
Dependencias	CU-E03, CU-U14	
Precondición	El usuario deberá estar previamente registrado y haber accedido a la aplicación. El usuario accede al perfil de otro usuario y puede ver sus ejercicios, bien porque sean públicos o protegidos.	
Secuencia normal	Paso	Acción
	1	Se ejecuta el caso de uso “Ver ejercicios de otro usuario”
	2	Se ejecuta el caso de uso “Seleccionar ejercicio”
	3	El actor 'Usuario' elige la acción “Tomar prestado”
	4	El sistema almacena la referencia al ejercicio entre los ejercicios a los que tiene disposición el usuario.
Postcondición	El ejercicio queda almacenado entre los ejercicios del usuario, estando disponible para sus búsquedas.	
Excepciones	Paso	Acción
Comentarios		

Tabla 44: Caso de uso “Tomar prestado ejercicio”

2.2.3 Matriz de trazabilidad:

A continuación, se muestra una relación entre los casos de uso y los requisitos funcionales. Se han separado por bloques para facilitar la visibilidad.

2.2.3.1 Ejercicios

La relación entre el bloque de requisitos funcionales y los casos de uso para el apartado de “Ejercicios” se describe en la Tabla 46.

CU/RF	RF1.1	RF1.2	RF1.2.1	RF1.3	RF1.4	RF1.5	RF1.6
CU-E01							
CU-E02							
CU-E03							
CU-E04							
CU-E05							
CU-E06							
CU-E07							
CU-E08							

Tabla 45: Matriz de trazabilidad de los casos de uso relativos a ejercicios

2.2.3.2 Sesión de entrenamiento

La relación entre el bloque de requisitos funcionales y los casos de uso para el apartado de “Sesión de entrenamiento” se describe en la Tabla 47.

CU/RF	RF2.1	RF2.2	RF2.3	RF2.4	RF2.5	RF2.6
CU-S01						
CU-S02						
CU-S03						
CU-S04						
CU-S05						
CU-S06						
CU-S07						

Tabla 46: Matriz de trazabilidad de los casos de uso relativos a la sesión de entrenamiento

2.2.3.3 Usuarios

La relación entre el bloque de requisitos funcionales y los casos de uso para el apartado de “Usuarios” se describe en la Tabla 48.

CU/RF	RF1.2	RF3.1	RF3.2	RF3.3	RF3.4	RF3.5	RF3.6	RF4.1	RF4.2	RF4.3	RF4.4
CU-U01											
CU-U02											
CU-U03											
CU-U04											
CU-U05											
CU-U06											
CU-U07											
CU-U08											
CU-U09											
CU-U10											
CU-U11											
CU-U12											
CU-U13											
CU-U14											
CU-U15											

Tabla 47: Matriz de trazabilidad de los casos de uso relativos a los usuarios

Capítulo 3. Plan de proyecto

3.1 Resumen del proyecto

3.1.1 Propósito, alcance y objetivos

El propósito es conseguir una aplicación web dirigida a entrenadores de baloncesto donde éstos puedan almacenar y compartir los ejercicios que elaboren.

El objetivo principal es que con los ejercicios que el entrenador registre, y aquellos que tome prestados, pueda seleccionar los que desee conformando así una “sesión de entrenamiento” que podrá ser exportada a un documento externo para llevar a la sesión de entrenamiento sin depender de la aplicación.

La aplicación también permitirá, como se ha mencionado antes, compartir ejercicios entre los usuarios, pudiendo un usuario tomar prestado un ejercicio de otro siempre y cuando la visibilidad del ejercicio lo permita. Un usuario puede dar establecer la visibilidad de sus ejercicios como pública, protegida o privada. Los ejercicios públicos serán accesibles para todo usuario, los protegidos sólo para los seguidores del usuario y los privados para el autor únicamente.

3.1.2 Definiciones y acrónimos

Acrónimo	Significado
UPEDU	Unified Process for EDUcation. (Proceso Unificado orientado al entorno educativo)
HTML5	HyperText Markup Language en su versión 5. Lenguaje para desarrollar el entorno web
CSS3	Cascade Sheet Style en su versión 3. Lenguaje para dotar de estilo y aumentar la visibilidad del entorno web, complementando al HTML5.
API	Aplication Programming Interface. Conjunto de funciones que un servicio ofrece para que sea utilizado por otro
IDE	Integrated Development Enviroment. Programa que permite desarrollar el código del proyecto
SGBD	Sistema gestor de bases de datos
SQL	Structured Query Language. Lenguaje utilizado para realizar consultas a la base de datos
MariaDB	Es el SGBD que se utilizará para el proyecto. Basado en SQL.
UVa	Universidad de Valladolid
TFG	Trabajo de fin de grado
MVC	Patrón de diseño Modelo Vista Controlador
PDF	Portable Document Format (Formato de documento portátil)

Tabla 48: Definiciones y acrónimos

3.1.3 Artefactos del proyecto

Más adelante se describe la metodología UPEDU, la cual necesita de los siguientes artefactos a desarrollar:

- Plan de Desarrollo de Software
- requisitos
- Análisis

- Diseño
- Implementación
- Test

3.1.4 Evolución del Plan

La presente memoria contendrá el plan de desarrollo del proyecto. El plan se irá actualizando fase por fase incluyendo las tareas que se lleven a cabo, así como una calendarización y duración de ellas.

El proyecto contará también con un seguimiento de cada una de las tareas y el tiempo de trabajo de las mismas, así como un seguimiento de los riesgos y el plan de actuación en consecuencia.

3.2 Plan del proyecto

En función de la duración y la fecha de realización del proyecto se ha decidido apostar por la metodología UPEDU.

La UPEDU [1] es una simplificación de la metodología del Proceso Unificado de Rational (RUP) desarrollado por IMB, que traza las líneas a seguir para llevar a cabo el RUP en un proyecto software de carácter académico.

Se definen cuatro fases (Inicio, Elaboración, Construcción y Transición), las cuales se desarrollan llevando a cabo diferentes iteraciones.

Al final de cada fase se obtienen distintos artefactos.

3.2.1 Ciclo de vida del proyecto

Seguidamente, se describen las fases que va a seguir el proyecto:

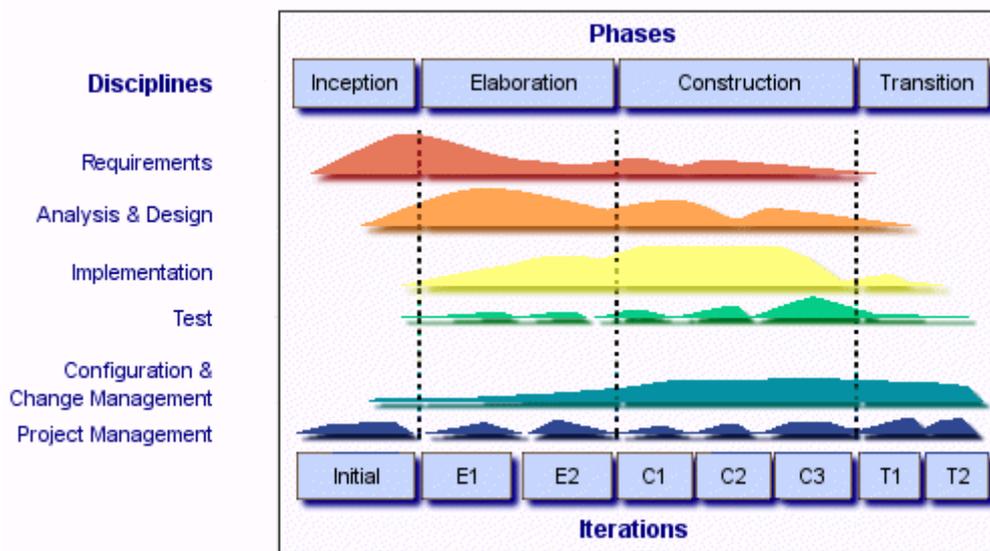


Figura 1: fases de la metodología UPEDU

Fase de inicio:

Fase de orientada de cara al cliente donde se especifica en qué consiste el proyecto y donde se definen el grueso de los requisitos que se deben llevar a cabo para que los desarrolladores puedan trabajar en una posterior implementación.

Fase de Elaboración:

Los desarrolladores cobran importancia en esta fase, teniendo que manejar una descripción detallada de los requisitos para poder combinar el deseo del cliente con la solución informática.

Fase de Construcción:

Los desarrolladores llevan a cabo la solución aproximada a la conseguida en la fase de elaboración, ya que siempre puede haber tenido que introducir algún cambio o requisito durante el proceso.

Fase de Transición:

Fase utilizada para para probar y desplegar la solución implementada.

3.2.2 Plan de puesta en marcha

Es habitual en proyectos de este carácter tomar como referencia proyectos realizados anteriormente que sean similares a él. En todo proyecto, además de la experiencia, se necesitan adquirir nuevas habilidades que permitan llevarlo a cabo.

- Conocimientos de tecnologías de la parte Frontend: HTML5, CSS3, JavaScript, jQuery, FabricJs.
- Conocimientos de tecnologías de la parte Backend: Java, JPA, Hibernate, Spring Boot.
- Configuración de los entornos Frontend y Backend, en especial la seguridad, utilizando Spring Security.
- Conocimientos de despliegue, en local y en producción.

Pese a ser un proyecto cuyo objetivo es ser presentado como TFG, aparte de funcionar en local se desea que sea desplegado en servidores gratuitos para que pueda ser probado por usuarios reales.

3.2.3 Plan de trabajo

A continuación, se describirán fases y cada una de las tareas que se desarrollarán en ellas además de su calendarización.

Se toma el 'día' como unidad de tiempo para establecer la duración

3.2.3.1 Fase Inicio

Nombre de la tarea	Duración	Comienzo	Fin
FASE INICIO	18	2019/07/01	2019/07/26
Inicio de la fase de inicio	0	2019/07/01	2019/07/01
Búsqueda de información sobre la UPEDU	1	2019/07/01	2019/07/01
Análisis de requisitos iniciales	3	2019/07/03	2019/07/05
Definición de las tareas de las fases	4	2019/07/05	2019/07/08
Calendarización	2	2019/07/07	2019/07/08
Gestión de riesgos	3	2019/07/13	2019/07/15
Documento seguimiento	1	2019/07/15	2019/07/15
Calcular presupuestos	1	2019/07/17	2019/07/17
Documento costes	1	2019/07/19	2019/07/19
Preparar entorno y herramientas	2	2019/07/25	2019/07/26
Fin de la fase de inicio	0	2019/07/26	2019/07/26

Tabla 49: Fase de inicio

3.2.3.2 Fase Elaboración

Nombre de la tarea	Duración	Comienzo	Fin
FASE ELABORACIÓN	42	2019/08/01	2019/09/11
Inicio de la fase de elaboración	0	2019/08/01	2019/08/01
Especificación completa de requisitos	3	2019/08/01	2019/08/03
Modelo de caso de uso	1	2019/08/05	2019/08/05
Especificación completa de los casos de uso	4	2019/08/06	2019/08/09
Modelo de dominio	2	2019/08/10	2019/08/11
Diseño de la base de datos	2	2019/08/12	2019/08/13
Investigación de jQuery	3	2019/08/19	2019/08/21
Investigación de FabricJs	5	2019/08/21	2019/08/25
Investigación de iText	5	2019/08/26	2019/08/30

Investigación de Spring Security	3	2019/08/31	2019/09/02
Diseño de las vistas de la interfaz de usuario	4	2019/09/01	2019/09/04
Arquitectura de la aplicación	1	2019/09/05	2019/09/05
Diagrama de clases de diseño	3	2019/09/05	2019/09/07
Diagrama de realización de los casos de uso	6	2019/09/06	2019/09/11
Fin de la fase de elaboración	0	2019/09/11	2019/09/11

Tabla 50: Fase de elaboración

3.2.3.3 Fase Construcción

Nombre de la tarea	Duración	Comienzo	Fin
FASE CONSTRUCCIÓN	41	2019/09/10	2019/10/20
Inicio de la fase de construcción	0	2019/09/10	2019/09/10
Configuración del cliente	3	2019/09/10	2019/09/12
Configuración del servidor	3	2019/09/13	2019/09/15
Implementar casos de uso de los ejercicios	15	2019/09/16	2019/09/30
Implementar casos de uso de la sesión de entrenamiento	10	2019/09/26	2019/10/05
Implementar casos de uso de los usuarios	10	2019/10/11	2019/10/20
Fin de la fase de construcción	0	2019/10/20	2019/10/20

Tabla 51: Fase de construcción

3.2.3.4 Fase Transición

Nombre de la tarea	Duración	Comienzo	Fin
FASE TRANSICIÓN	11	2019/10/28	2019/11/04
Inicio de la fase de transición	0	2019/10/28	2019/10/28
Pruebas de la aplicación	5	2019/10/28	2019/11/01
Elaborar manual de usuario	2	2019/10/31	2019/11/01
Elaborar manual de instalación	2	2019/11/02	2019/11/03
Elaborar manual del programador	2	2019/11/03	2019/11/04

Tabla 52: Fase de transición

3.2.4 Análisis de riesgos

A continuación, se detallan los riesgos principales del proyecto y el plan de contingencia elegido para sobrellevarlos en el caso de que sucedan.

La Tabla 54 muestra la exposición al riesgo que existir en función de su impacto (catastrófico, crítico, marginal, despreciable) y su probabilidad de aparición (0-100%)

Impacto/Probabilidad	100 - 81 %	80 - 61 %	60 - 41 %	40 - 21 %	20 - 0 %
Catastrófico	Alto	Alto	Moderado	Moderado	Bajo
Crítico	Alto	Alto	Moderado	Bajo	Ninguno
Marginal	Moderado	Moderado	Bajo	Ninguno	Ninguno
Despreciable	Moderado	Bajo	Bajo	Ninguno	Ninguno

Tabla 53: Definición de exposición al riesgo

El conjunto de tablas entre la Tabla 55 y la Tabla 61 recogen los riesgos existentes para el proyecto.

Identificador	R01 - Fallo en la planificación
Descripción	Los tiempos estimados para la planificación son insuficientes para llevar a cabo el desarrollo de lo planificado.
Impacto	Crítico
Probabilidad	70%
Exposición	Alto
Plan de mitigación	Realizar las tareas de modo que se cumpla con los tiempos establecidos.
Plan de contingencia	Priorizar las tareas a desarrollar y hacer los cambios pertinentes en la planificación.

Tabla 54: Riesgo de fallo en la planificación

Identificador	R02 - Indisponibilidad del desarrollador
Descripción	El desarrollador no puede dedicar el tiempo suficiente a su labor provocando así una demora en los tiempos planificados.
Impacto	Crítico
Probabilidad	50%
Exposición	Moderado
Plan de mitigación	Cumplir los tiempos planificados e incluir un colchón de seguridad.
Plan de contingencia	Replanificar las tareas, solapando si se pudiera algunas de ellas, y dedicar tiempo a mayores que inicialmente no estaba previsto.

Tabla 55: Riesgo de indisponibilidad del desarrollador

Identificador	R03 - Indisponibilidad del tutor
Descripción	El tutor no puede dedicar el tiempo suficiente a su labor provocando así una demora en los tiempos planificados.

Impacto	Crítico
Probabilidad	25 %
Exposición	Bajo
Plan de mitigación	Cumplir los tiempos planificados e incluir un colchón de seguridad.
Plan de contingencia	Replanificar las tareas, solapando si se pudiera algunas de ellas.

Tabla 56: Riesgo de indisponibilidad del tutor

Identificador	R04 - Enfermedad
Descripción	El desarrollador no puede dedicar el tiempo suficiente o avanza más despacio debido a una enfermedad o motivos personales en su labor, provocando así una demora en los tiempos planificados.
Impacto	Marginal
Probabilidad	25%
Exposición	Ninguno
Plan de mitigación	Cumplir los tiempos planificados e incluir un colchón de seguridad.
Plan de contingencia	Replanificar las tareas, solapando si se pudiera algunas de ellas, y dedicar tiempo a mayores que inicialmente no estaba previsto.

Tabla 57: Riesgo de enfermedad

Identificador	R05 - Curva de aprendizaje demasiado larga
Descripción	La curva de aprendizaje de las nuevas tecnologías a utilizar, por parte del desarrollador, supone una demora importante en el tiempo planificado para ello.
Impacto	Crítico
Probabilidad	80%
Exposición	Alta
Plan de mitigación	Cumplir los tiempos planificados e incluir un colchón de seguridad. Ir adelantando la investigación.
Plan de contingencia	Replanificar las tareas, solapando si se pudiera algunas de ellas.

Tabla 58: Riesgo de curva de aprendizaje demasiado lenta

Identificador	R06 - Tecnologías escogidas inadecuadamente
Descripción	Las tecnologías escogidas no permiten llevar a cabo los requisitos especificados.
Impacto	Catastrófico
Probabilidad	20%
Exposición	Bajo
Plan de mitigación	Estudiar minuciosamente las posibilidades que permiten las tecnologías escogidas.
Plan de contingencia	Buscar una tecnología sustitutiva que permita desarrollar los objetivos y readaptar la calendarización en base a ello.

Tabla 59: Riesgo de tecnologías escogidas inadecuadamente

Identificador	R07 - Ausencia de comunicación
Descripción	Comunicación escasa o nula entre el alumno y el tutor, provocando retraso en el avance del proyecto.
Impacto	Crítico
Probabilidad	35%
Exposición	Bajo
Plan de mitigación	Hablar diaria/semanalmente y planificar antes de algún periodo en el que se sepa de antemano que la comunicación es imposible.
Plan de contingencia	Replanificar las tareas, solapando si se pudiera algunas de ellas, y dedicar tiempo a mayores que inicialmente no estaba previsto.

Tabla 60: Riesgo de ausencia de comunicación

3.2.5 Presupuesto

A continuación, se detalla el presupuesto estimado del proyecto, considerando los costes materiales:

- Recursos personales: El desarrollo del proyecto lo llevará a cabo una única persona, el alumno. Desempeñará los siguientes roles:
 - Cliente
 - Jefe de Proyecto
 - Diseñador
 - Programador
 - Probador
- Recursos materiales: El desarrollo del proyecto es imposible que se lleve a cabo sin determinados costes materiales:
 - Ordenador personal
 - PC sobremesa
 - Portátil
 - Servidor para el despliegue
 - Heroku
 - JawsDB
 - Herramientas
 - Astah UML
 - Balsamiq
 - IDE Spring Tools
 - IDE Microsoft Visual Studio Code
 - Google Drive
 - Postman
 - Spring Boot
 - MariaDB
 - DataGrip
 - GitLab
 - Material oficina
 - Papel
 - Material de escritura
 - Lugar de trabajo
 - Residencia del alumno
 - Servicios
 - Conexión a internet
 - Conexión a red eléctrica

3.2.5.1 Estimación de costes del proyecto

Los costes para este proyecto van a ser nulos puesto que se trata de un proyecto estudiantil y no hay intención de invertir dinero en él. Se utilizarán siempre recursos de coste cero.

Recurso	Descripción	Coste
Ordenador Personal	Tanto el PC de sobremesa como el ordenador portátil han sido utilizados y amortizados en proyectos anteriores.	0€
Servidor	El coste del despliegue del proyecto será nulo puesto que se utilizarán servicios gratuitos pese a que puedan tener limitaciones.	0€
Herramientas	Todas las herramientas utilizadas son gratuitas a excepción de Astah UML, Balsamiq, DataGrip y GitLab que son de pago, pero tienen versión gratuita para estudiantes.	0€
Material de oficina	Se dispone de papel y herramientas de escritura comprados anteriormente.	0€
Lugar de trabajo	El alumno no paga por su residencia.	0€
Servicios	El alumno no paga por el internet ni la luz consumidos en su residencia.	0€

Tabla 61: Estimación de los costes materiales del proyecto

Rol	Coste
Jefe Proyecto	0€
Diseñador	0€
Programador	0€
Probador	0€

Tabla 62: Estimación de los costes humanos del proyecto

3.2.5.2 Simulación de un proyecto real.

No obstante, vamos a plantear una simulación donde los costes no tuvieran valor nulo. Vamos a simular que este proyecto no fuera un TFG sino un proyecto real que quisiera llevar a cabo una empresa y calcular unos costes acordes a este contexto.

Recurso	Descripción	Coste
Ordenador Personal	Ya amortizado	0€
Servidor	Heroku [2] - mensual	55€*
	JawsDB [3] - mensual	110€*
Herramientas	Astah UML [4] - paquete 6 meses	45€
	Balsamiq [5] - durante 6 meses	64€

	DataGrip [6] - paquete 12 meses	199€
	GitLab [7] - durante 6 meses	36€
Material de oficina	Papel y herramientas de escritura	10€
Lugar de trabajo	Ya amortizado / Pagado a parte	0€
Servicios	Ya amortizado / Pagado a parte	0€
	TOTAL	519€

Tabla 63: Estimación de los costes materiales de un proyecto real

*Durante el desarrollo del proyecto no es lógico estar pagando los servicios del servidor. Asumiremos que pagamos un único mes, en lo que configuramos el despliegue y el resto ya será coste del mantenimiento del proyecto.

Suponiendo un salario anual 36.164€ al año para el jefe de proyecto en España [8], lo que supone 2.356€ mensuales; y un salario anual de 21.890€ al año para el resto de roles en Castilla y León [9], lo que supone 1.564€ euros mensuales.

Supongamos también una media de 20 días laborables al mes.

Rol	Tiempo	Coste
Jefe Proyecto	18 días	2120.4 €
Diseñador	94 días	7350.8 €
Programador		
Probador		
TOTAL	112 días	9471.2 €

Tabla 64: Estimación de los costes humanos de un proyecto real

Suma total de los recursos materiales: 519+9471.2 = 9990.2 €

3.3 Seguimiento del plan de trabajo

3.3.1 Seguimiento de las Fases

3.3.1.1 Seguimiento de la Fase de Inicio

Nombre	Duración prevista	Duración real	Comienzo previsto	Fin previsto	Comienzo Real	Fin Real
Inicio de la fase de inicio	0	0	2019/07/01	2019/07/01	2019/07/01	2019/07/01
Búsqueda de información sobre la UPEDU	1	1	2019/07/01	2019/07/01	2019/07/01	2019/07/01
Análisis de requisitos iniciales	3	5	2019/07/03	2019/07/05	2019/07/01	2019/07/05
Definición de las tareas de las fases	4	4	2019/07/05	2019/07/08	2019/07/05	2019/07/08
Calendarización	2	2	2019/07/07	2019/07/08	2019/07/07	2019/07/08
Gestión de riesgos	3	4	2019/07/13	2019/07/15	2019/07/13	2019/07/16
Documento seguimiento	1	1	2019/07/15	2019/07/15	2019/07/16	2019/07/16
Calcular presupuestos	1	1	2019/07/17	2019/07/17	2019/07/17	2019/07/17
Documento costes	1	1	2019/07/19	2019/07/19	2019/07/19	2019/07/19
Preparar entorno y herramientas	2	3	2019/07/25	2019/07/26	2019/07/25	2019/07/27
Fin de la fase de inicio	0	0	2019/07/26	2019/07/26	2019/07/27	2019/07/27

Tabla 65: Seguimiento de la fase de inicio

En esta fase no se encuentran demasiados problemas y solamente hubo un pequeño percance a la hora de conectar el IDE STS con la base de datos MariaDB.

3.3.1.2 Seguimiento de la Fase de Elaboración:

Nombre	Duración prevista	Duración real	Comienzo previsto	Fin previsto	Comienzo Real	Fin Real
Inicio de la fase de elaboración	0	0	2019/08/01	2019/08/01	2019/08/01	2019/08/01
Especificación completa de requisitos	3	3	2019/08/01	2019/08/03	2019/08/01	2019/08/03
Modelo de caso de uso	1	1	2019/08/05	2019/08/05	2019/08/05	2019/08/05
Especificación completa de los casos de uso	4	4	2019/08/06	2019/08/09	2019/08/08	2019/08/11
Modelo de dominio	2	3	2019/08/10	2019/08/11	2019/08/12	2019/08/14
Diseño de la base de datos	2	3	2019/08/12	2019/08/13	2019/08/14	2019/08/16
Investigación de jQuery	3	5	2019/08/19	2019/08/21	2019/08/18	2019/08/22
Investigación de FabricJs	5	8	2019/08/21	2019/08/25	2019/08/22	2019/08/29
Investigación de iText	5	5	2019/08/26	2019/08/30	2019/08/30	2019/09/03
Investigación de Spring Security	3	7	2019/08/31	2019/09/02	2019/09/05	2019/09/11

Diseño de las vistas de la interfaz de usuario	4	8	2019/09/01	2019/09/04	2019/09/12	2019/09/19
Arquitectura de la aplicación	1	2	2019/09/05	2019/09/05	2019/09/20	2019/09/21
Diagrama de clases de diseño	3	2	2019/09/05	2019/09/07	2019/09/22	2019/09/23
Diagrama de realización de los casos de uso	6	6	2019/09/06	2019/09/11	2019/09/24	2019/09/29
Fin de la fase de elaboración	0	0	2019/09/11	2019/09/11	2019/09/29	2019/09/29

Tabla 66: Seguimiento de la fase de elaboración

En esta ocasión sí ha habido problemas para llevar a cabo el proyecto y seguir la planificación del proyecto. Como se describe en el riesgo R01, el tiempo estimado para la tarea a realizar ha sido insuficiente, lo que ha supuesto un incremento en cuanto al tiempo utilizado para desempeñar las labores y, con ello, su consecuente desfase en las fechas de comienzo y fin de las mismas.

3.3.1.3 Seguimiento de la fase de Construcción:

Esta fase comienza con un desfase de más de dos semanas, motivo por el que es difícil comenzar y terminar en los plazos preestablecidos.

Nombre	Duración prevista	Duración real	Comienzo previsto	Fin previsto	Comienzo Real	Fin Real
Inicio de la fase de construcción	0	0	2019/09/10	2019/09/10	2019/10/6	2019/10/6
Configuración del cliente	3	3	2019/09/10	2019/09/12	2019/10/6	2019/10/8
Configuración del servidor	3	3	2019/09/13	2019/09/15	2019/10/8	2019/10/10
Implementar casos de uso de los ejercicios	15	41	2019/09/16	2019/09/30	2019/10/10	2019/11/19
Implementar casos de uso de la sesión de entrenamiento	10	24	2019/09/26	2019/10/05	2019/11/19	2019/12/12
Implementar casos de uso de los usuarios	10	16	2019/10/11	2019/10/20	2019/12/12	2019/12/27
Fin de la fase de construcción	0	0	2019/10/20	2019/10/20	2019/12/27	2019/12/27

Tabla 67: Seguimiento de la fase de construcción

Esta fase estaba planificada para realizarse en 31 días, pero, por motivos laborales, aparece el riesgo R02 en diversas ocasiones. Además, el R01 reaparece de nuevo y, como puede apreciarse, sólo el desarrollo de los casos de uso de los ejercicios termina durando 10 días más de los estimado debido a que las nuevas tecnologías utilizadas tienen una curva de aprendizaje demasiado costosa (R05). Tras el desfase al implementar los casos de uso de los ejercicios, las dos tareas siguientes se demoran, pero en menor escala, según se va obteniendo dominio de las herramientas.

Al final de la fase se obtiene un desfase de dos meses con respecto a la planificación original.

3.3.1.4 Seguimiento de la fase de Transición:

Nombre	Duración prevista	Duración real	Comienzo previsto	Fin previsto	Comienzo Real	Fin Real
Inicio de la fase de transición	0	0	2019/10/28	2019/10/28	2020/01/07	2020/01/07
Pruebas de la aplicación	5	5	2019/10/28	2019/11/01	2020/01/07	2020/01/11
Elaborar manual de usuario	2	2	2019/10/31	2019/11/01	2020/01/11	2020/01/12
Elaborar manual de instalación	2	2	2019/11/02	2019/11/03	2020/01/12	2020/01/13
Elaborar manual del programador	2	2	2019/11/03	2019/11/04	2020/01/13	2020/01/14

Tabla 68: Seguimiento de la fase de transición

En esta ocasión, la fase está bien estimada de tiempo, pero se sigue arrastrando el desfase de las dos fases anteriores.

3.3.2 Seguimiento de riesgos

De todos los riesgos descritos, como ya se ha comentado antes, solo han aparecido durante el proyecto tres:

- **R01:** Tiene lugar durante las fases de elaboración y construcción. La estimación de la planificación ha sido insuficiente, requiriéndose más tiempo del que se pensó inicialmente. No es raro que haya sucedido este riesgo, su probabilidad era bastante alta.
- **R02:** Tiene lugar durante la fase de construcción. Es un riesgo que inicialmente no se contaba con que apareciese, pero debido a requerimientos laborales, el desarrollador tuvo que hacer horas extra durante algunos días imposibilitando trabajar en el proyecto.
- **R05:** Tiene lugar durante la fase de construcción, especialmente en la implementación de los casos de uso de los ejercicios. Pese haber investigado el uso de la librería JavaScript FabricJs, su uso no es tan sencillo como los ejemplos vistos en la investigación, lo que hace que se tarde mucho más tiempo en tener los resultados esperados.
- El resto de riesgos no aparecen.

3.3.3 Seguimiento de presupuesto y costes reales del proyecto

Las Tabla 70 y la Tabla 71 representan, respectivamente, el coste material y humano si aplicamos el coste real en función del tiempo final empleado:

Recurso	Descripción	Coste
Ordenador Personal	Ya amortizado	0€
Servidor	Heroku - mensual	55€*

	JawsDB - mensual	110€*
Herramientas	Astah UML - paquete 6 meses	45€
	Balsamiq - durante 6 meses	64€
	DataGrip - paquete 12 meses	199€
	GitLab - durante 6 meses	36€
Material de oficina	Papel y herramientas de escritura	10€
Lugar de trabajo	Ya amortizado / Pagado a parte	0€
Servicios	Ya amortizado / Pagado a parte	0€
	TOTAL	519€

Tabla 69: Seguimiento de los costes materiales del proyecto

Rol	Tiempo	Coste
Jefe Proyecto	25 días	2.945 €
Diseñador	177 días	13.841,4 €
Programador		
Probador		
TOTAL	112 días	16.786,4 €

Tabla 70: Seguimiento de los costes humanos del proyecto

Suma total de los recursos materiales: 519 + 16.786,4 = 17.305,4 €

Capítulo 4. Análisis

4.1 Descripción del modelo de usuario

El sistema está pensado para ser utilizado por un grupo concreto de personas. Estas serían personas pertenecientes al mundo del baloncesto, concretamente los entrenadores.

Los usuarios (entrenadores) se definen como personas, independientemente de su género, con una edad a partir de los 16 años, que es la edad legal a la que se puede empezar a entrenar.

El rango de edad estimado para el modelo de usuario es algo difuso, puesto que las personas cuyo empleo principal no sea ser entrenador dejan de entrenar cuando encuentran un trabajo acorde a sus estudios o cuando deciden formar una familia. Es decir, es un hobby para ellos. Por otro lado, si el usuario sí se dedica exclusivamente a las labores de entrenador, podríamos encontrar usuarios con edades próximas a la jubilación. Se estimará que las edades de los usuarios estarán comprendidas en el rango [16,70].

Por la época actual en la que vivimos y el rango establecido de edades, damos por sentado que los usuarios están familiarizados con un aprendizaje informático estándar y que saben utilizar internet de manera sencilla.

El baloncesto, como todo, sigue unos estándares de definición y existe un lenguaje y unos símbolos concretos para poder comunicarse. Se da por sentado que el usuario que utilice esta aplicación conoce dichos términos lingüísticos y símbolos (ver anexo II) ya que forman parte del día a día de un entrenador.

4.2 Diagrama de casos de uso

A continuación, se muestra el diagrama de casos de uso de la aplicación. Se ha dividido en tres agrupaciones diferentes en función del contexto de cada caso de uso.

Las agrupaciones definidas son:

1. Ejercicios: todos aquellos casos de uso que relacionados con la creación y operaciones relacionadas con los ejercicios.
2. Sesión de entrenamiento: todos aquellos casos de uso relacionados con la gestión de la sesión de entrenamiento.
3. Usuarios: todos aquellos casos de uso relacionados con la interacción entre usuarios.

4.2.1 Ejercicios

EJERCICIOS

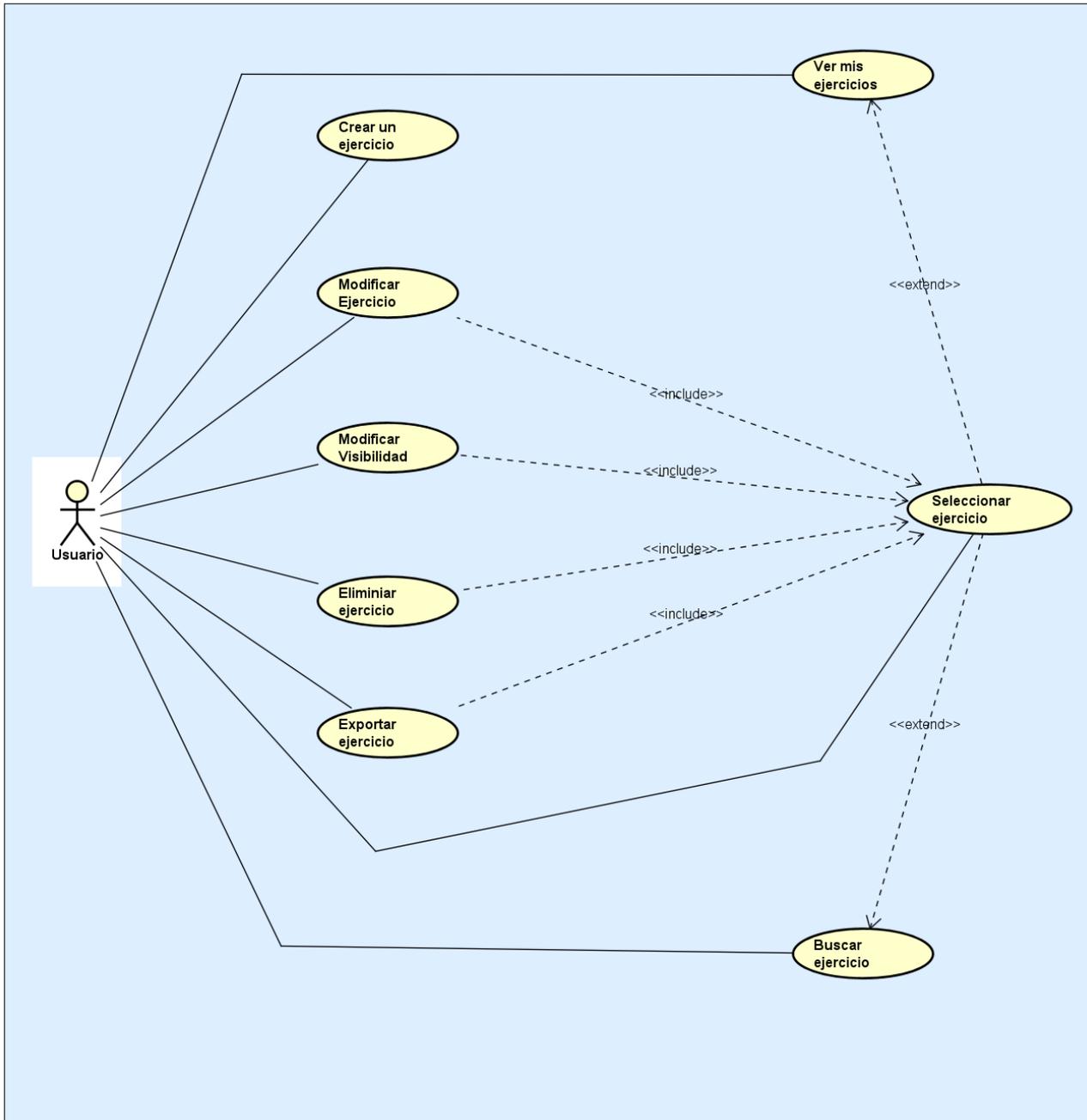


Figura 2: Diagrama de casos de uso relacionados con los ejercicios

4.2.2 Sesión de entrenamiento

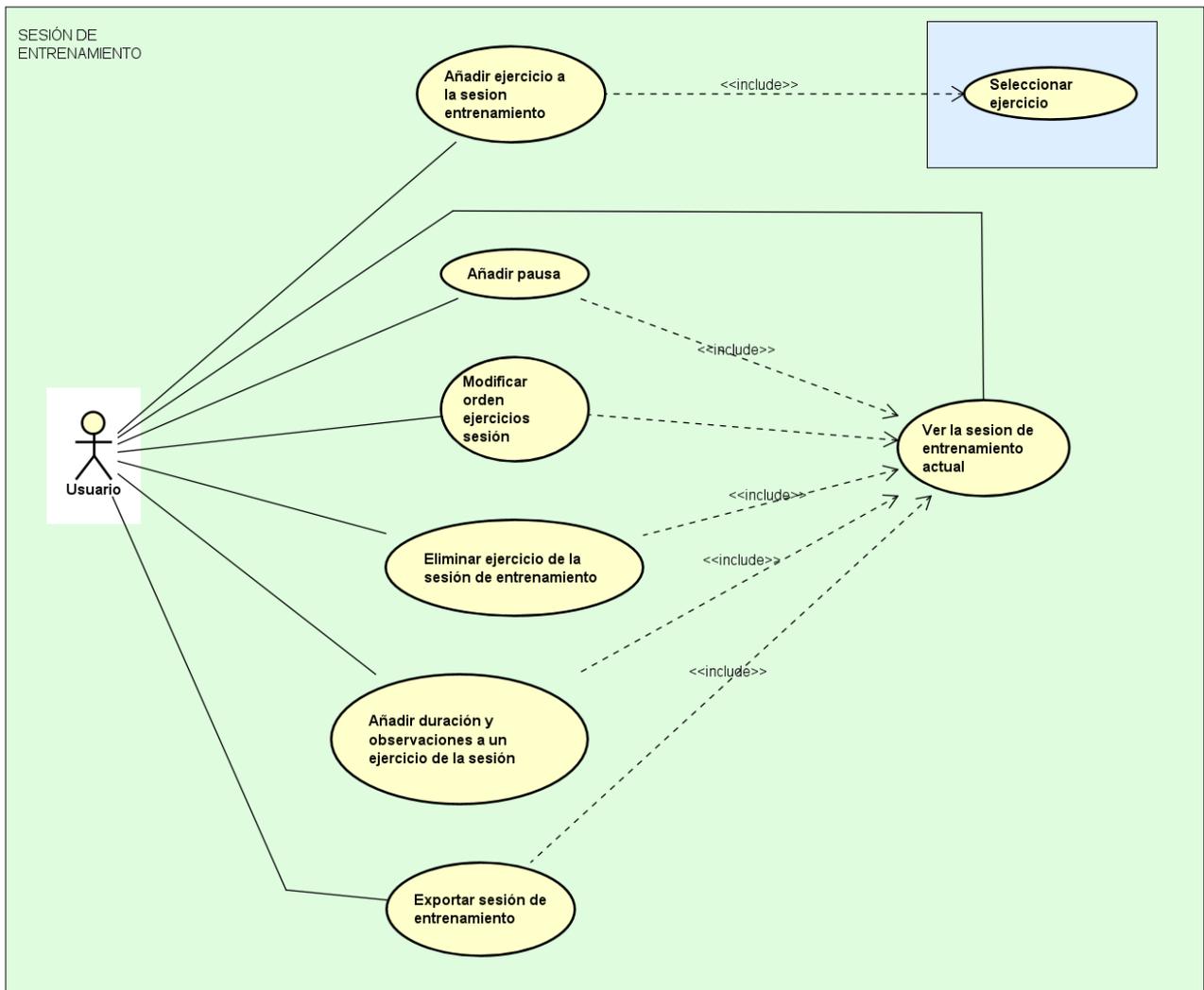


Figura 3: Diagrama de casos de uso relacionados con la sesión de entrenamiento

4.2.3 Usuarios

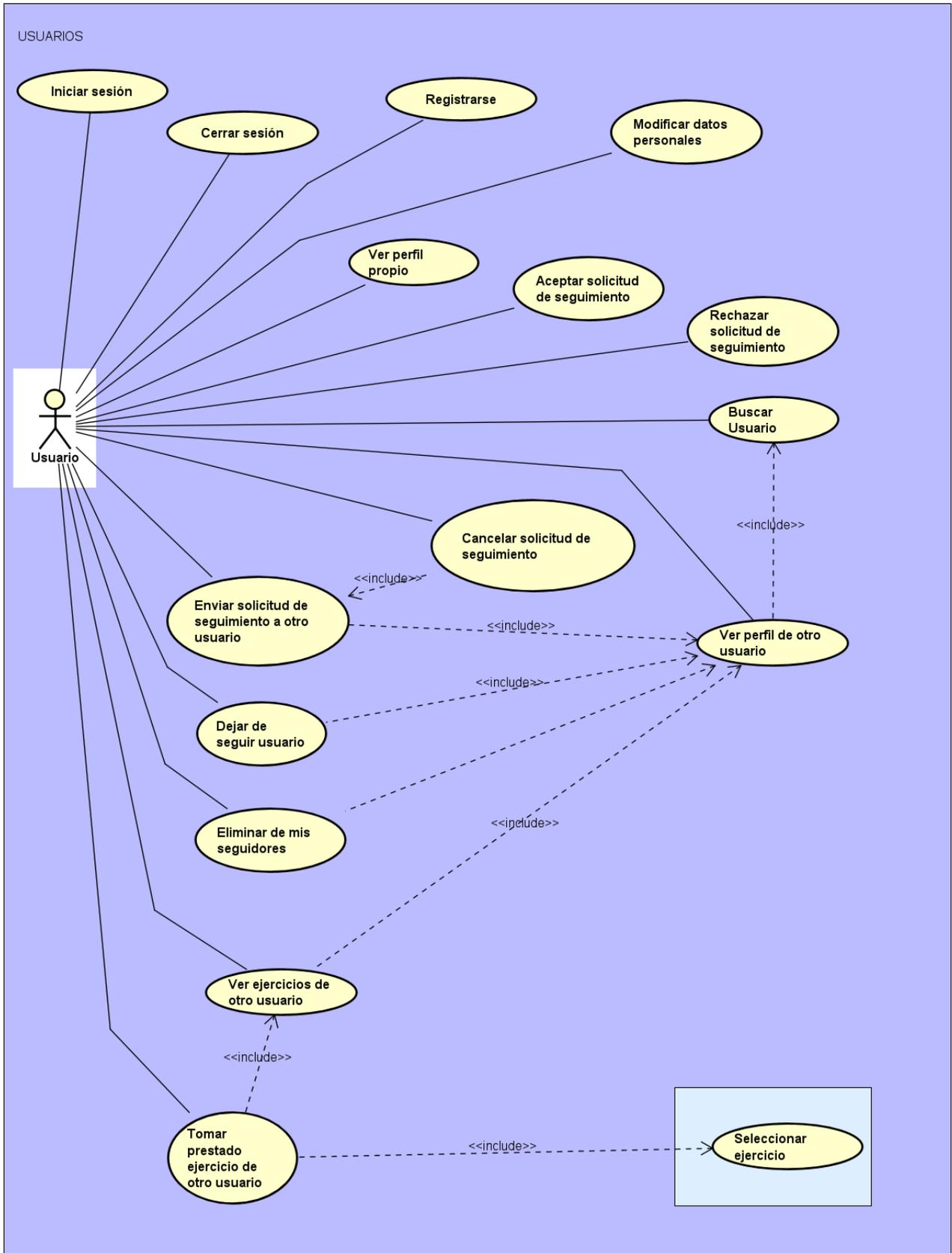


Figura 4: Diagrama de casos de uso relacionados con los usuarios

4.3 Modelado de dominio

La Figura 5 muestra el diagrama de clases que representa el modelado conceptual resultante de la fase de análisis. En él se recogen las clases principales que representarán las entidades que se diseñarán más adelante, y las relaciones que hay entre ellas.

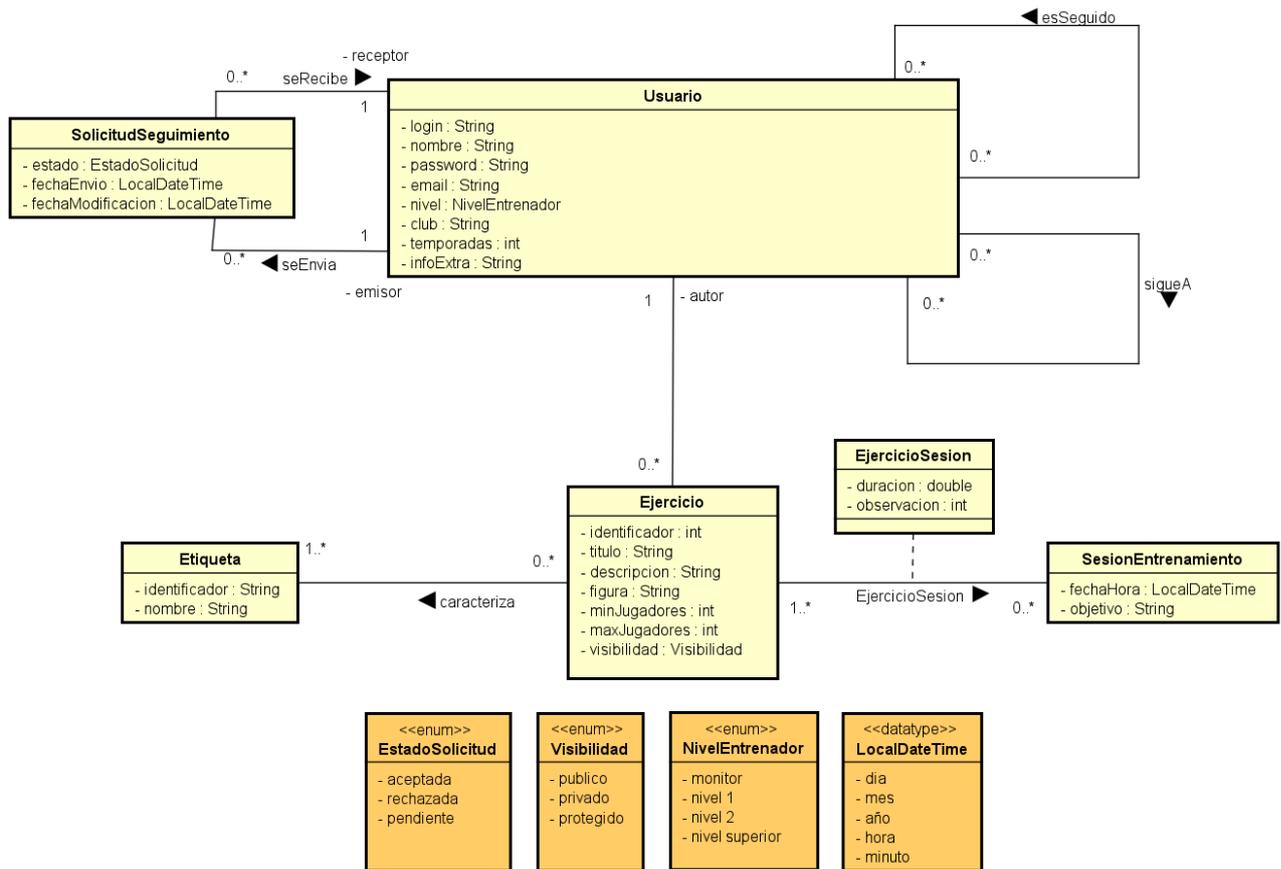


Figura 5: Diagrama del modelo de dominio

Capítulo 5. Tecnologías utilizadas

5.1 Tecnologías utilizadas para el frontend

Se recoge en la Tabla 72 la lista de las tecnologías utilizadas y su utilidad dentro del desarrollo de la parte frontend:

Nombre	Utilidad
HTML5	Estructura de la página web
CSS3	Estilo de la página web
JavaScript	Lenguaje de programación empotrado en el navegador para añadir funcionalidad y estilo a la página web
jQuery	Biblioteca JavaScript que permite simplificar la interacción con los elementos HTML y realizar llamadas contra el backend.
Bootstrap	Biblioteca basada en JavaScript y jQuery que aporta un conjunto de plantillas para dotar de estilo la aplicación.
FabricJs	Biblioteca necesaria para dibujar las figuras de los ejercicios.
Bootpag	Biblioteca para mostrar paginación.
Bootstrap-notify	Biblioteca utilizada para mostrar mensajes temporales de retroalimentación al usuario en determinadas ocasiones.
jQuery Session	Biblioteca que complementa a jQuery para manejar de manera más sencilla la sesión del navegador.
jQuery User Interface	Biblioteca que complementa a jQuery en la interfaz de usuario. Permite ordenar elementos arrastrándolos. Utilizado para ordenar los ejercicios de la sesión.
jQuery Validate	Biblioteca que utiliza jQuery para validar de forma sencilla formularios y en el lado del cliente, mostrando un mensaje de error en caso de que no se cumpla la validación.
Spectrum	Biblioteca utilizada como color-picker (selector de color) para dibujar los ejercicios.
Bootstrap4-toogle	Biblioteca JavaScript que permite convertir un checkbox en un selector deslizable. Utilizado para cambiar entre el campo entero y el medio campo al crear un ejercicio.
Bootbox	Biblioteca JavaScript que se utiliza para mostrar todas las alertas y mensajes de confirmación.
DateTimePicker	Biblioteca JavaScript que se utiliza para permitir seleccionar una fecha y una hora de manera sencilla y visual. Utilizada para seleccionar la fecha y la hora del entrenamiento.

Tabla 71: Tecnologías utilizadas en el frontend

5.2 Tecnologías utilizadas para el backend

Se recoge en la Tabla 73 la lista de las tecnologías utilizadas y su utilidad dentro del desarrollo de la parte backend:

Nombre	Utilidad
Spring Boot	Framework para crear el API-REST y proporcionar servicio al frontend.
JPA (Java Persistence API) + Hibernate	Framework que conecta contra una base de datos para manejar datos relacionales sin necesidad de utilizar consultas SQL continuamente.
Spring Security	Framework que complementa a Spring Boot y dota de seguridad al backend.
iText PDF	Biblioteca Java que permite crear documentos en formato PDF.
Lombok	Biblioteca Java que permite evitar tener que generar los getters y setters de las clases (resources y beans), así como dotarlas de un logger.
Thymeleaf	Biblioteca Java que sirve como motor de plantillas para abastecer al frontend.

Tabla 72: Tecnologías utilizadas en el backend

Capítulo 6. Diseño

6.1 Diseño centrado en el usuario

Antes de comenzar a tomar decisiones de diseño, se ha optado por realizar un boceto en papel para mostrárselo a los usuarios objetivo y recabar información de la usabilidad de la aplicación, con el fin de cambiar algunos aspectos pensados inicialmente si los usuarios encuentran dificultades para utilizarla.

Es preferible tener una idea final y preparar el diseño en base a esa idea a tener que cambiar y corregir aspectos una vez se haya hecho.

6.1.1 Bocetaje

Se ha utilizado la herramienta Balsamiq para generar un boceto en papel de la aplicación web. El boceto se recoge en el “Anexo III”.

Este ha sido enviado a varios entrenadores reales para que den su opinión y algunas sugerencias de mejora.

6.1.2 Retroalimentación obtenida de los usuarios

De los usuarios se ha obtenido la siguiente retroalimentación tras enviarles un boceto dinámico en papel de la interfaz de usuario.

	Edad	Observaciones, dudas y sugerencias
Entrenador 1	23	Todo claro. Que los ejercicios se carguen “por páginas”, es decir ver 5, dar a cargar más, ver otros 5.
Entrenador 2	29	Tiene dudas en un primer momento. Afirma que ya existen algunas aplicaciones similares. No obstante, cree que puede ser una buena idea.
Entrenador 3	22	Le gusta la idea. La compara con una red social estilo Twitter y WhatsApp.
Entrenador 4	30	La transición entre páginas le parece buena. Añadiría la opción de crear sistemas de juego y de ellos poder generar ejercicios. También la posibilidad de incluir ejercicios meramente físicos. Ha tenido que fijarse bien para diferenciar entre las búsquedas normales y la avanzada.
Entrenador 5	30	Le ha gustado mucho la idea. Añadiría la posibilidad de importar figuras en lugar de pintarlas. Buscar ejercicios por el nombre también.
Entrenador 6	42	Muy completa y accesible.
Entrenador 7	57	Todo muy bien y sencillo de utilizar.
Entrenador 8	21	La usabilidad es buena y se entiende todo bien. Añadiría que se pudieran importar figuras en lugar de tener que pintarlas online, así como subir videos de la realización del ejercicio. También incluir la posibilidad de registrar el material necesario en el ejercicio.
Entrenador 9	19	Le gusta la idea. Cree conveniente añadir etiquetas para medio campo y campo completo.
Entrenador 10	65	Le parece una buena idea, pero se muestra de primeras reacio a utilizarla. Se orienta mejor a mano.

Tabla 73: Retroalimentación obtenida de los usuarios del boceto

6.1.3 Actuación acorde a la retroalimentación

A la mayoría de los entrenadores les ha encantado la idea de poder registrar y compartir ejercicios, pero también buscan una funcionalidad que les permita llevar una gestión de su equipo. Este, por el momento, no es el objetivo de la aplicación.

Hay algunas ideas que podrían ser interesantes, como la de subir vídeos de la realización del ejercicio. El problema de esta idea, por ejemplo, es que probablemente los vídeos contendrían imágenes de menores de edad, pudiendo entrar en conflicto con el Reglamento General de Protección de Datos, así que de momento no se va a gestionar ese aspecto.

Sí se va a incluir dos buenas ideas:

- a) el recuperar los ejercicios utilizando una paginación de ejercicios de cinco en cinco,
- b) crear etiquetas para permitir la búsqueda de ejercicios en función del espacio de la pista que se necesite.

6.2 Decisiones de diseño

A continuación, se recogen las decisiones de diseño aplicadas:

- Se utilizará una arquitectura Cliente-Servidor. El cliente será un cliente “thin”, por el contrario, el servidor será el “thick”. Esto significa que el cliente contendrá la lógica justa y necesaria para representar la información proveniente del servidor.
- Se utilizará el framework JPA como especificación de las entidades para el mapeo de las entidades con la base de datos.
- Se usará el framework Hibernate como implementación de la especificación de JPA, para la persistencia de los datos. Será el encargado de realizar las operaciones CRUD contra la base de datos.
- Se persistirá la información en una base de datos relacional MariaDB.
- El desarrollo será con el framework Spring y Java 8.
- Se usará un servidor Tomcat 8 para correr la aplicación en el entorno de desarrollo (local).
- Se utilizará el servidor Tomcat proporcionado por Heroku para correr la aplicación en el entorno de producción.
- El desarrollo de la parte cliente se realizará con HTML5+CSS3+jQuery principalmente (ver apartado anterior).
- Se utilizará el framework Bootstrap para dotar de estilo a las páginas

6.3 Diseño del almacenamiento persistente

La Figura 6 recoge el modelo relacional que se ha obtenido para la base de datos. El esquema utilizado para representar las claves foráneas es el siguiente:

```
<nombre campo de la tabla>:<nombre del campo de la tabla a la que referencia>
```

Este esquema es producto de haber utilizado Hibernate. Se ha generado automáticamente. Por esa razón, algunas tablas o campos pueden tener nombres extraños. Por ejemplo: la tabla “ejercicio_etiqueta” se genera automáticamente a partir de la relación “muchos a muchos” (Many-to-many) que existen entre la entidad ejercicio y la entidad etiqueta.

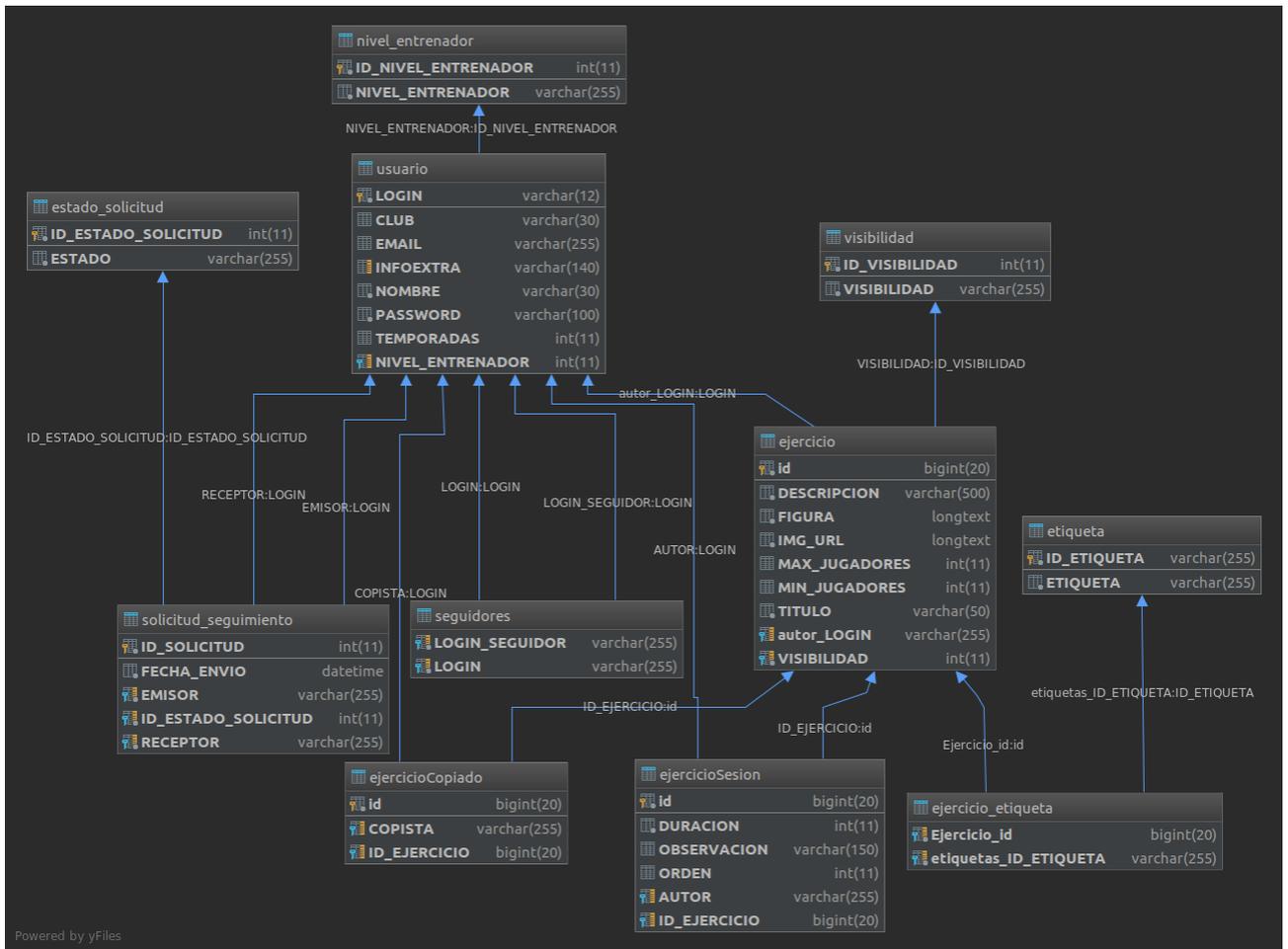


Figura 6: Diagrama relacional de la base de datos

6.4 Diseño arquitectónico

6.4.1 Arquitectura propuesta

6.4.1.1 Arquitectura lógica

Puesto que se ha elegido una arquitectura de tipo cliente-servidor, se puede afirmar que el lado del cliente contendrá la interfaz de la aplicación (es decir, la capa de presentación) que se ve alimentada mediante peticiones al lado del servidor (capa de negocio).

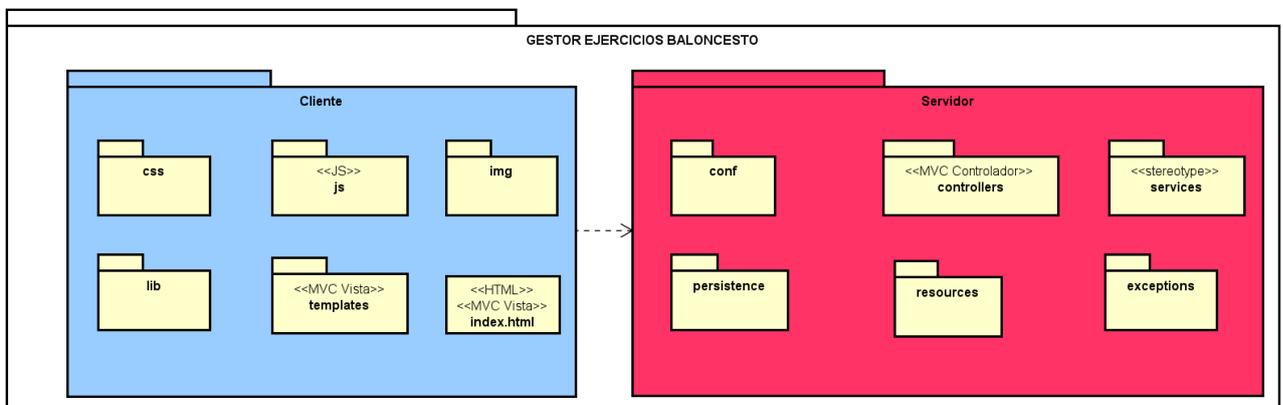


Figura 7: Arquitectura de la aplicación

6.4.1.2 Despliegue

En cuanto a lo que al despliegue se refiere, se han realizado dos despliegues idénticos conceptualmente, pero con distinta utilidad. Para ambos lados el lado cliente es igual y se resume en un navegador realizando peticiones contra el servidor. En cambio, el lado servidor sí que varía.

Por un lado, para el despliegue de desarrollo se utiliza el servidor Tomcat que trae por defecto que conecta contra un SGBD MariaDB desplegado también en local.

En cambio, para el despliegue en producción se utilizan los servicios proporcionados por Heroku. Heroku permite, de manera gratuita, pero con alguna limitación, desplegar aplicaciones Java en sus servidores, utilizando también Tomcat para este fin. Heroku también proporciona un servicio de persistencia de datos a través de JawsDB, un SGBD que se ha configurado para correr MariaDB.

Como se ha mencionado antes, los servicios tienen limitaciones si son gratuitos: si la aplicación web no es utilizada durante 30 min se detiene el servicio, siendo arrancado cuando alguien vuelva a acceder al dominio. JawsDB limita el número de conexiones simultáneas a diez y un almacenamiento máximo de 0.005 GB. Condiciones suficientes para que el pequeño grupo de usuarios que probarán la aplicación pueda hacerlo.

Diagrama de despliegue durante el desarrollo:

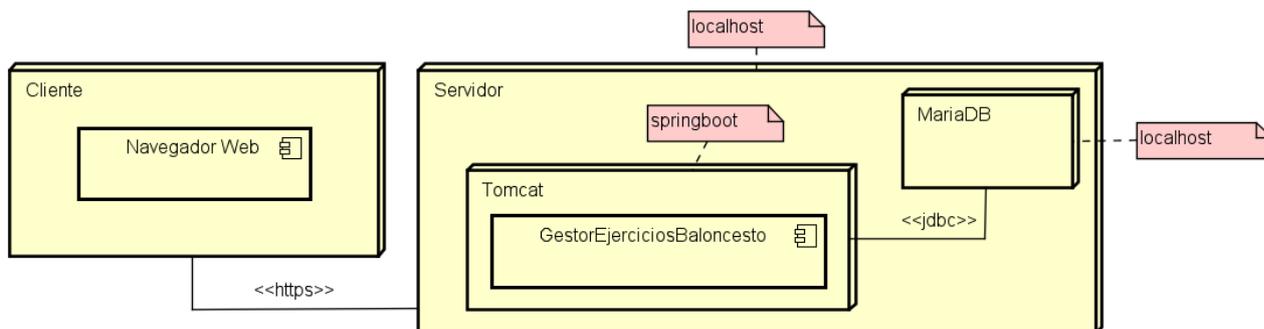


Figura 8: Diagrama de despliegue en local

Diagrama de despliegue en producción:

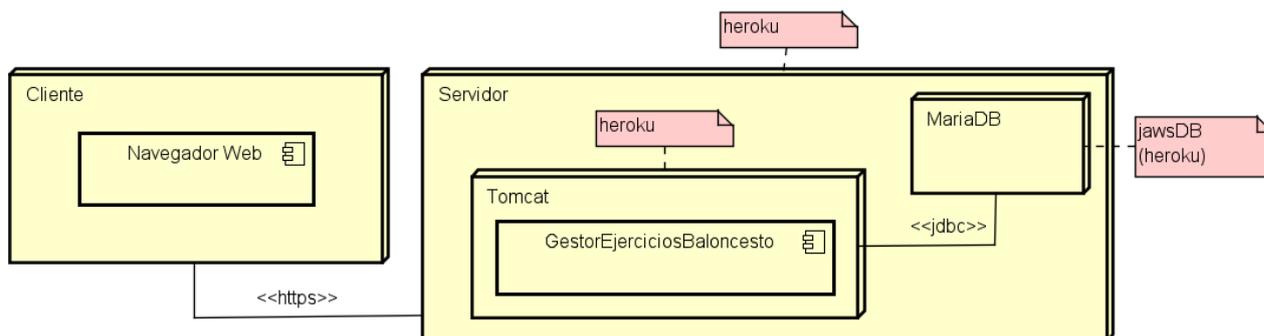


Figura 9: Diagrama de despliegue en producción

6.4.2 Arquitectura del cliente

El cliente se ha diseñado de manera muy sencilla. Al no utilizar un framework que marque una determinada manera de actuación, como en el caso del servidor, el cliente está dividido en los siguientes paquetes y las dependencias mostradas (Figura 10). En el paquete templates encontramos todas las vistas. Para cada vista, existe un archivo JavaScript que contendrá los métodos encargados de realizar las llamadas al servidor para mostrar/almacenar los datos pertinentes, y las funciones que modifican las vistas, como se puede apreciar en la Figura 11.

Los paquetes css y lib contienen los estilos y las librerías necesarias para que las vistas se muestren y funcionen correctamente

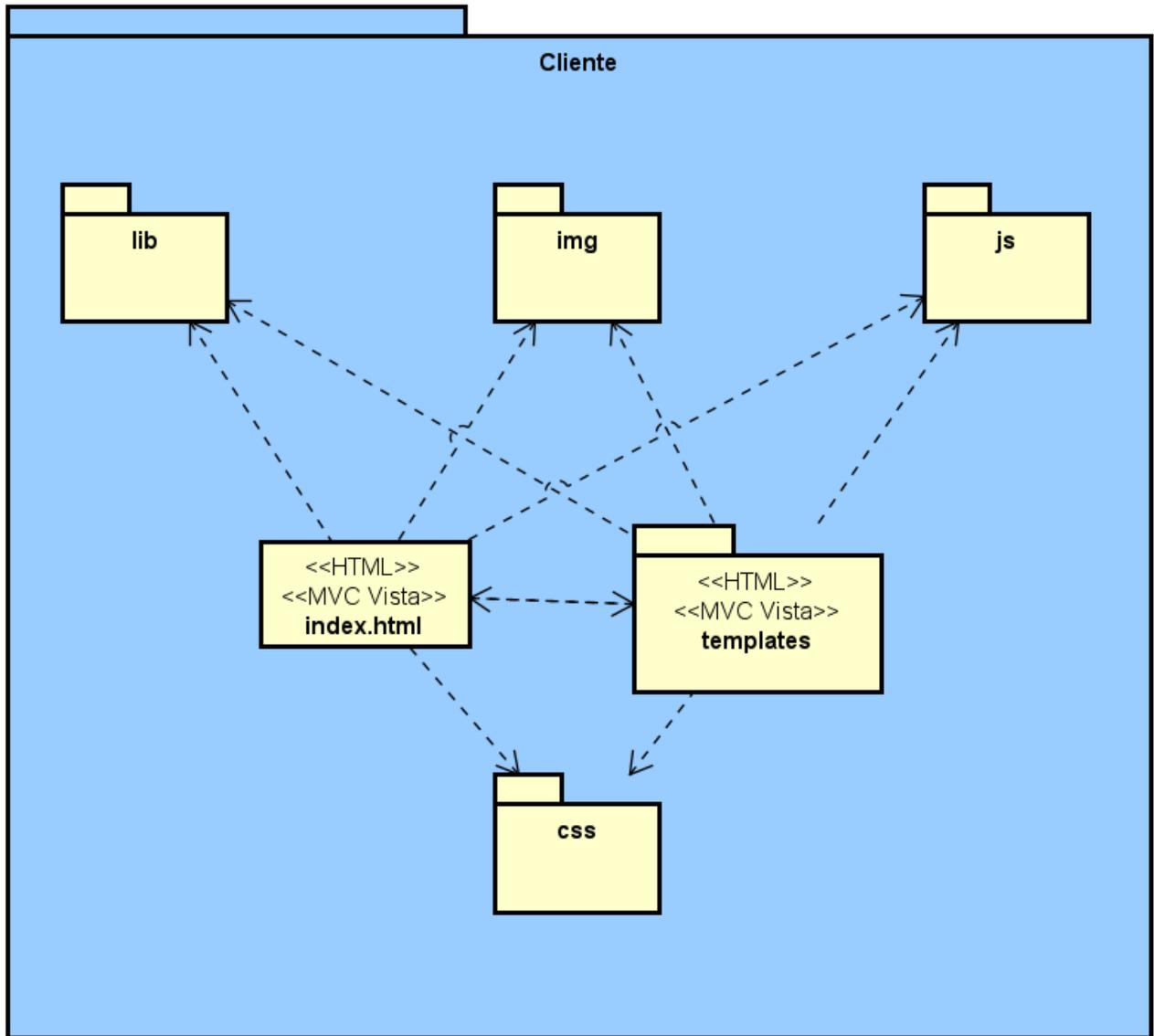


Figura 10: Dependencias en el lado del cliente

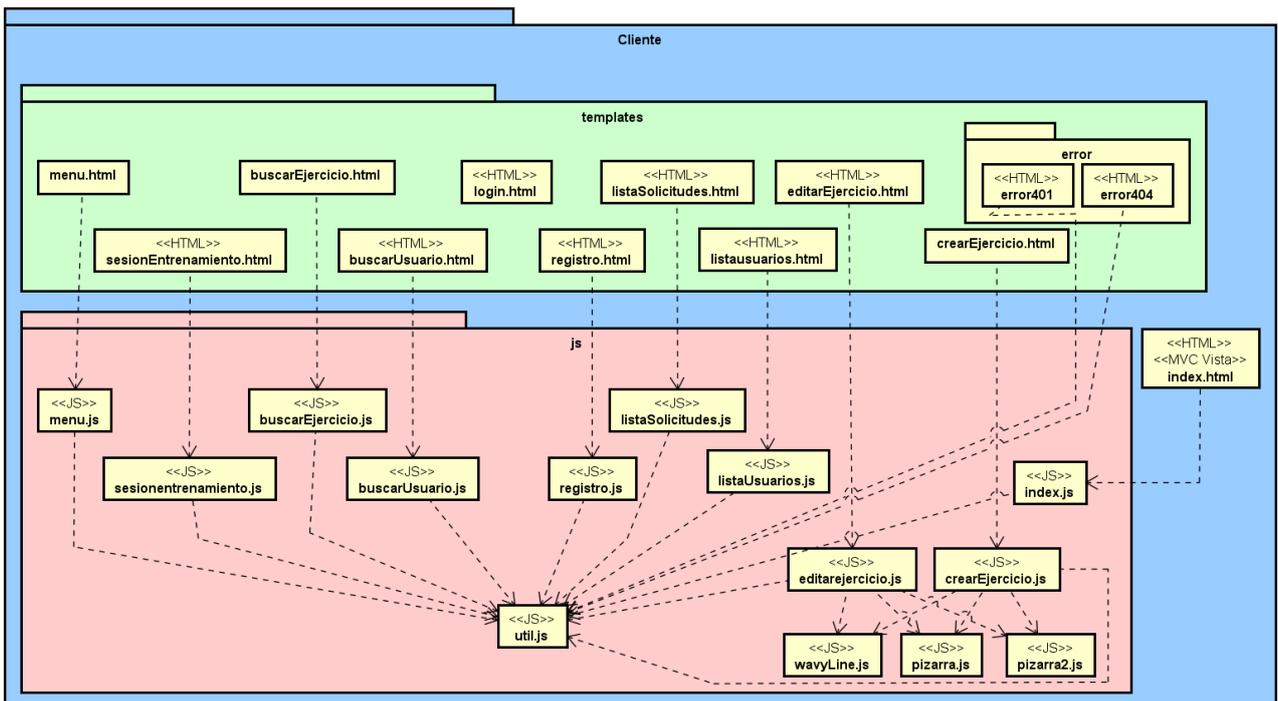


Figura 11: Dependencias entre las vistas y los archivos JavaScript

6.4.3 Arquitectura del servidor

Como se ha mencionado, para el servidor (backend) se utiliza el framework Spring que está pensado para implementar un API REST utilizando el patrón MVC.

Los paquetes pertenecientes al modelo son todas aquellos que podemos ver en la imagen a excepción del paquete Controllers que, como su propio nombre indica, contiene los controladores. Los controladores reciben las peticiones enviadas por el cliente (frontend), y delegan en las clases del modelo la lógica para obtener la solicitud recibida. El modelo conecta con la base de datos a través del paquete repositorio y trabaja con las entidades definidas en el paquete resources. Todos los paquetes tienen dependencias hacia las excepciones, ya que en el momento en el que se detecta un error, se lanza una excepción que será recogida y traducida en una respuesta HTTP que contenga información sobre la causa del error.

El paquete conf contiene las clases de configuración y seguridad del proyecto.

La Figura 12 representa visualmente los paquetes y las dependencias descritas.

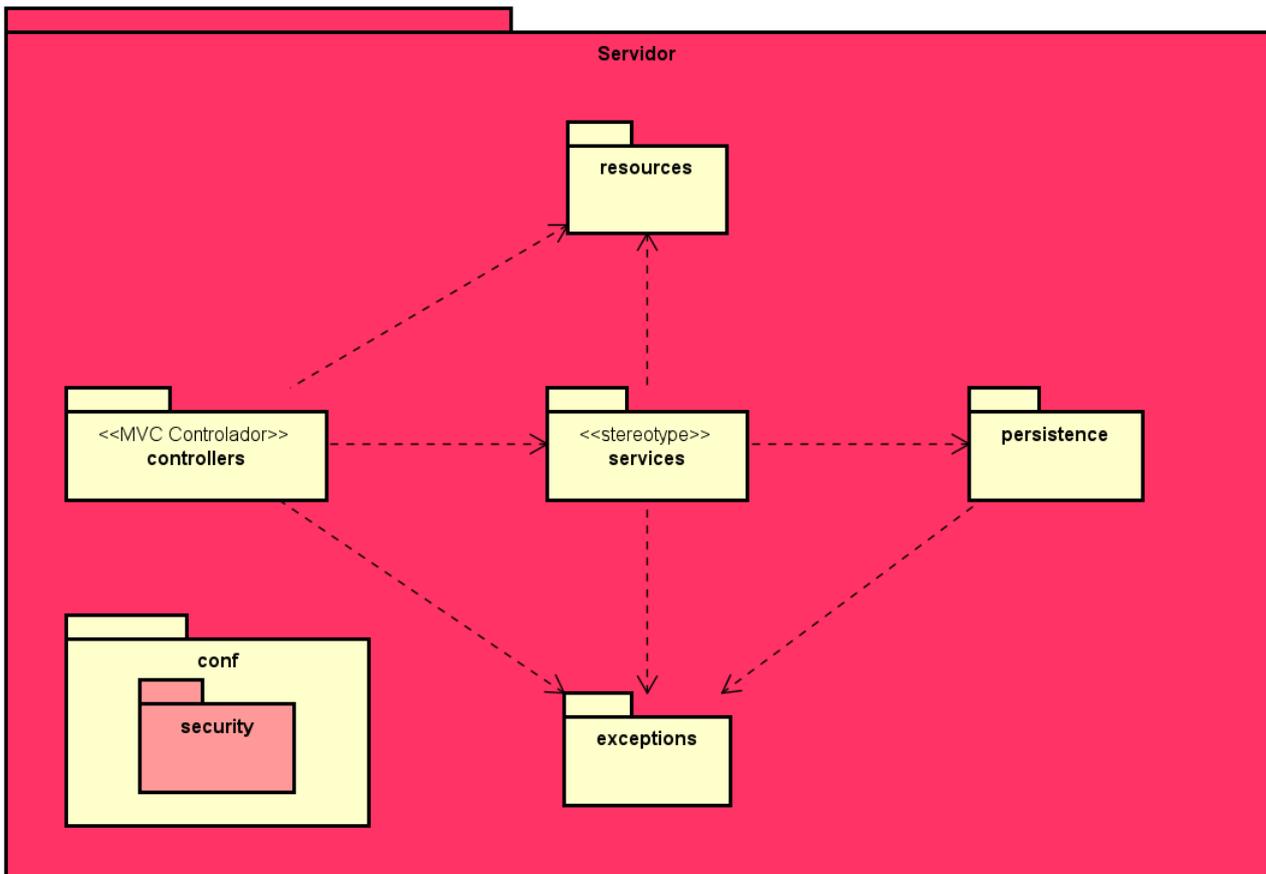


Figura 12: Dependencias del lado del servidor

6.6 Realización en diseño de casos de uso

Puesto que Spring Boot utiliza el patrón MVC, prácticamente todos los casos de uso son conceptualmente iguales. Se han escogido tres como los más representativos: “Crear Ejercicio”, “Exportar Sesión de entrenamiento” y “Eliminar Ejercicio”.

6.6.1 Crear Ejercicio

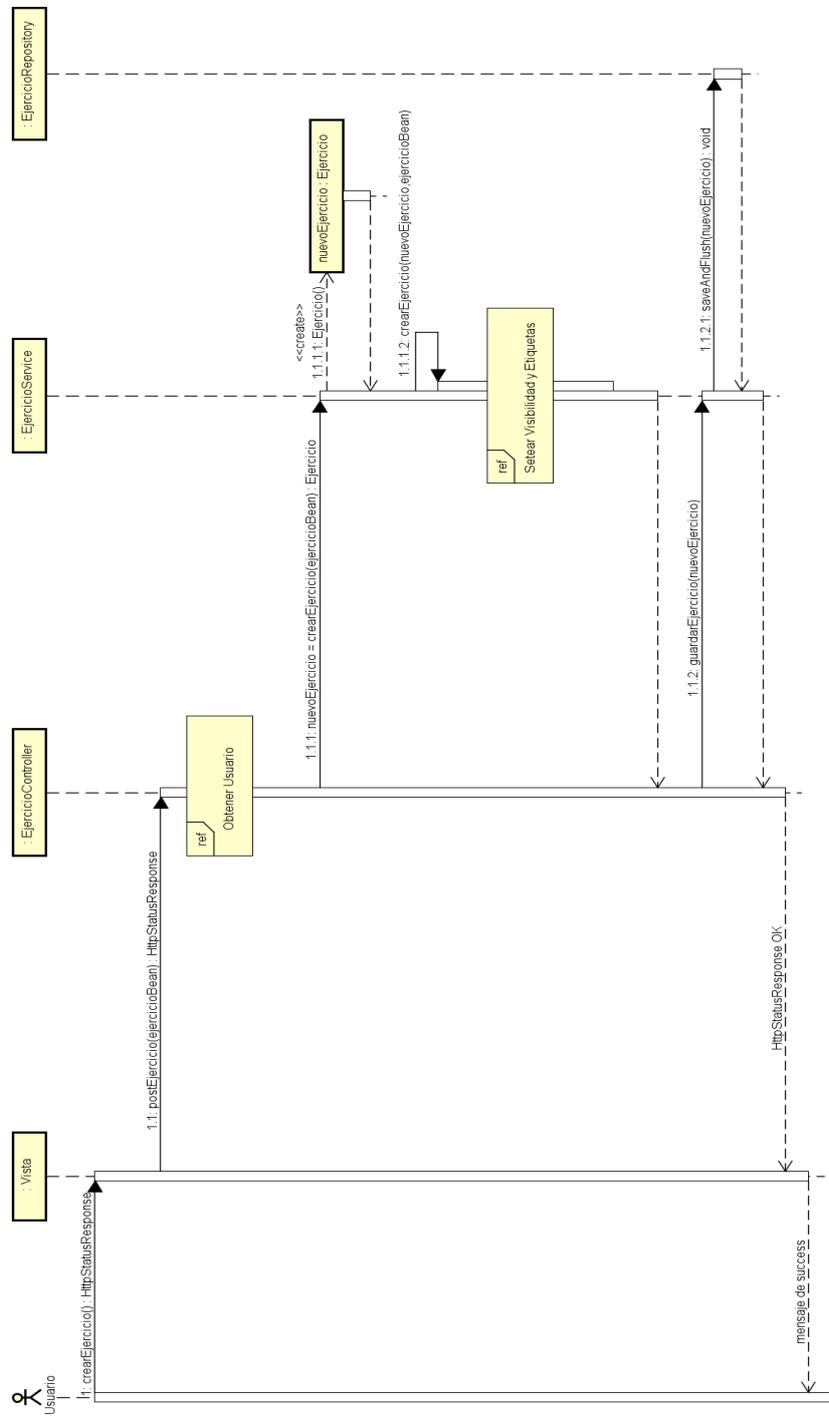


Figura 14: Diagrama de secuencia de “Crear Ejercicio”

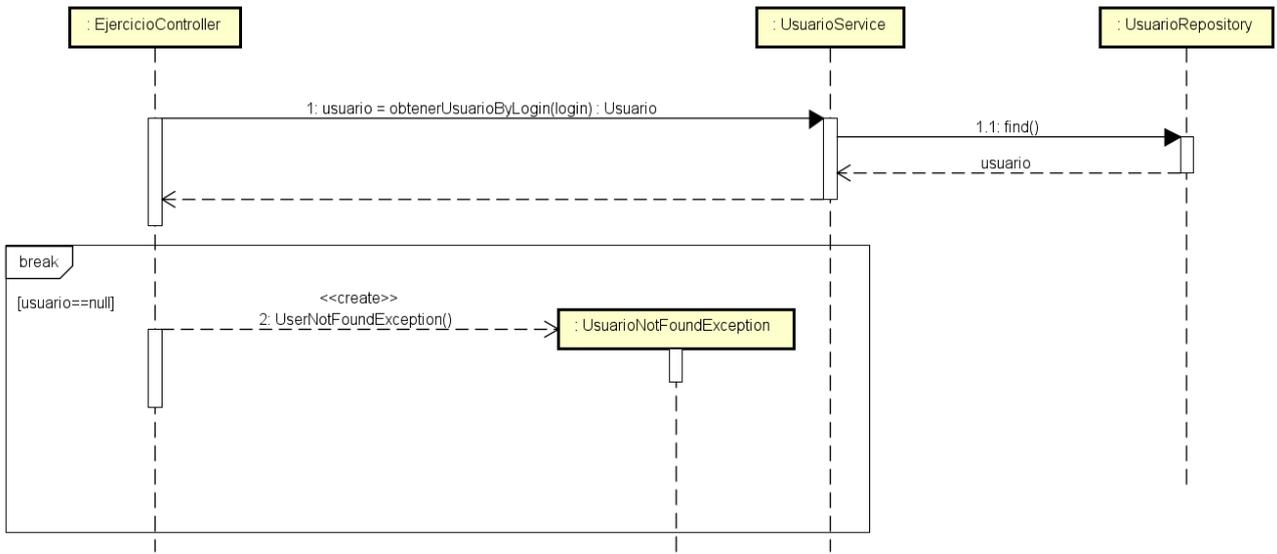


Figura 15: Diagrama de secuencia de "Obtener usuario"

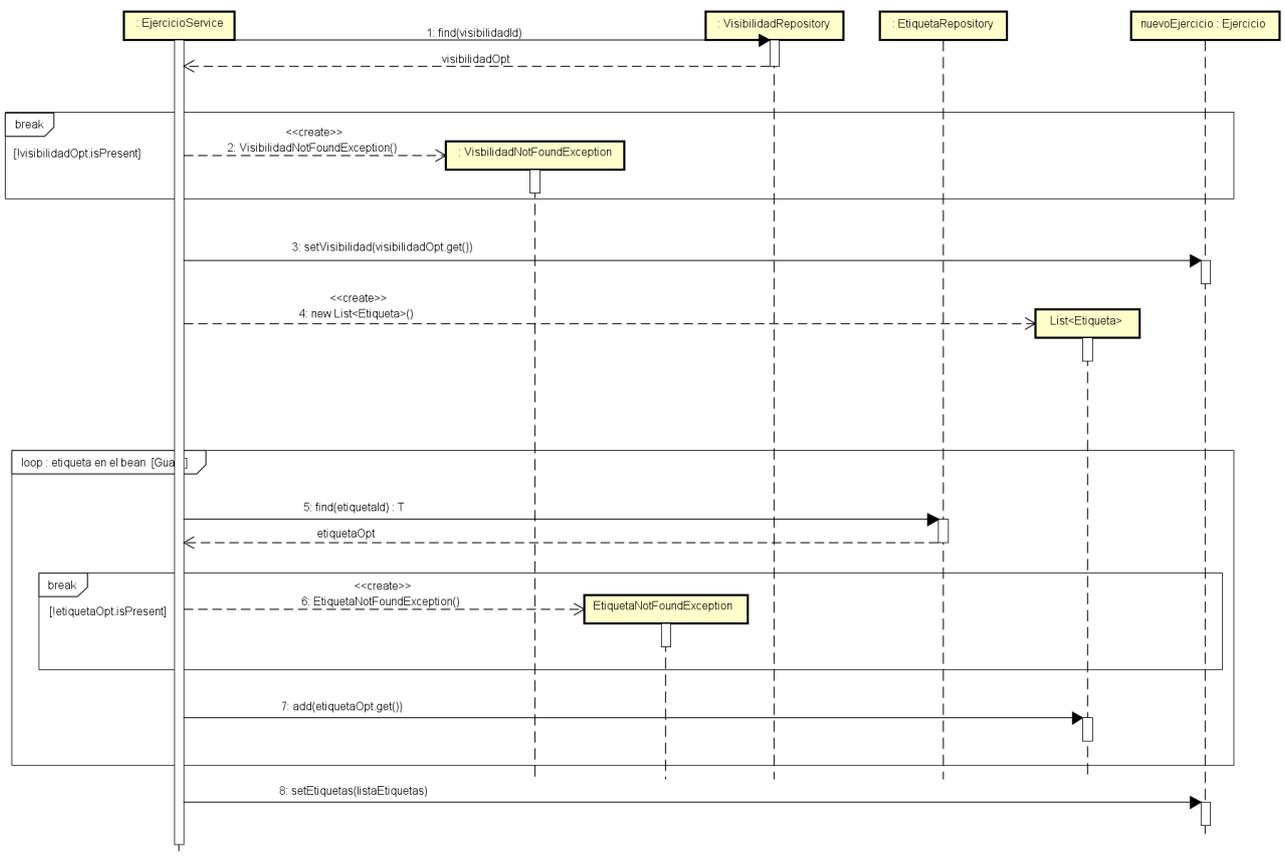


Figura 16: Diagrama de secuencia de "Setear visibilidad y etiquetas"

6.6.2 Eliminar Ejercicio

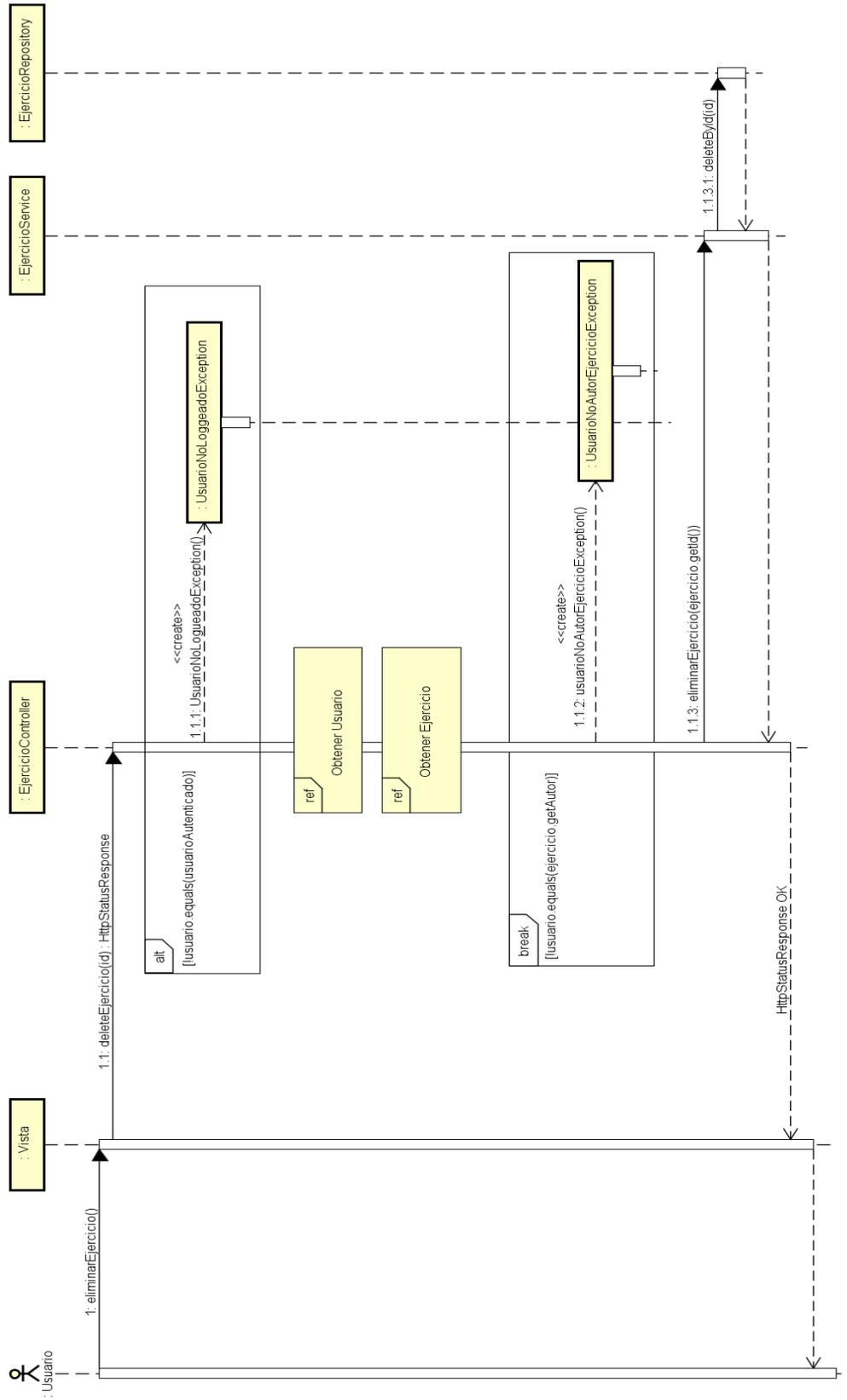


Figura 17: Diagrama de secuencia de “Eliminar ejercicio”

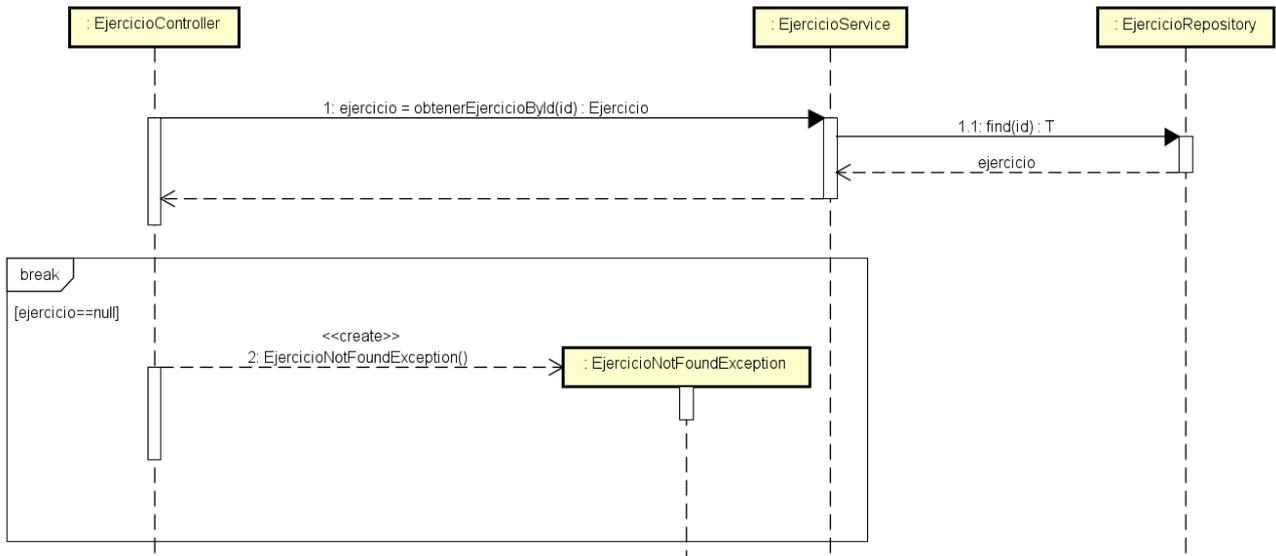


Figura 18: Diagrama de secuencia de “Obtener ejercicio”

6.7 Diseño detallado

Las figuras comprendidas entre Figura 19 y Figura 27 detallan el contenido de los paquetes que no han sido descritos previamente.

6.7.1 Paquetes del cliente

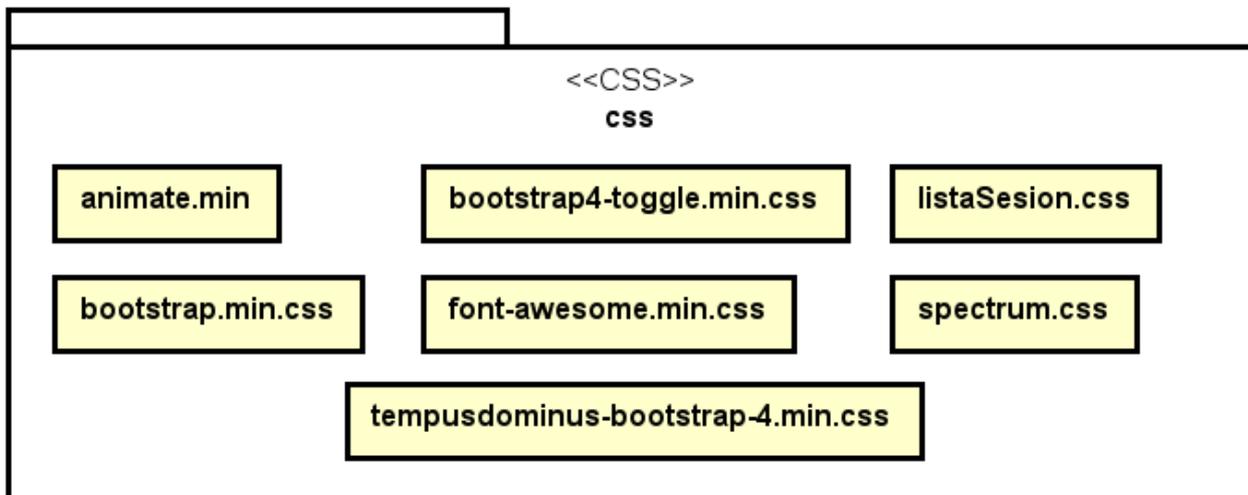


Figura 19: Archivos de estilo existentes

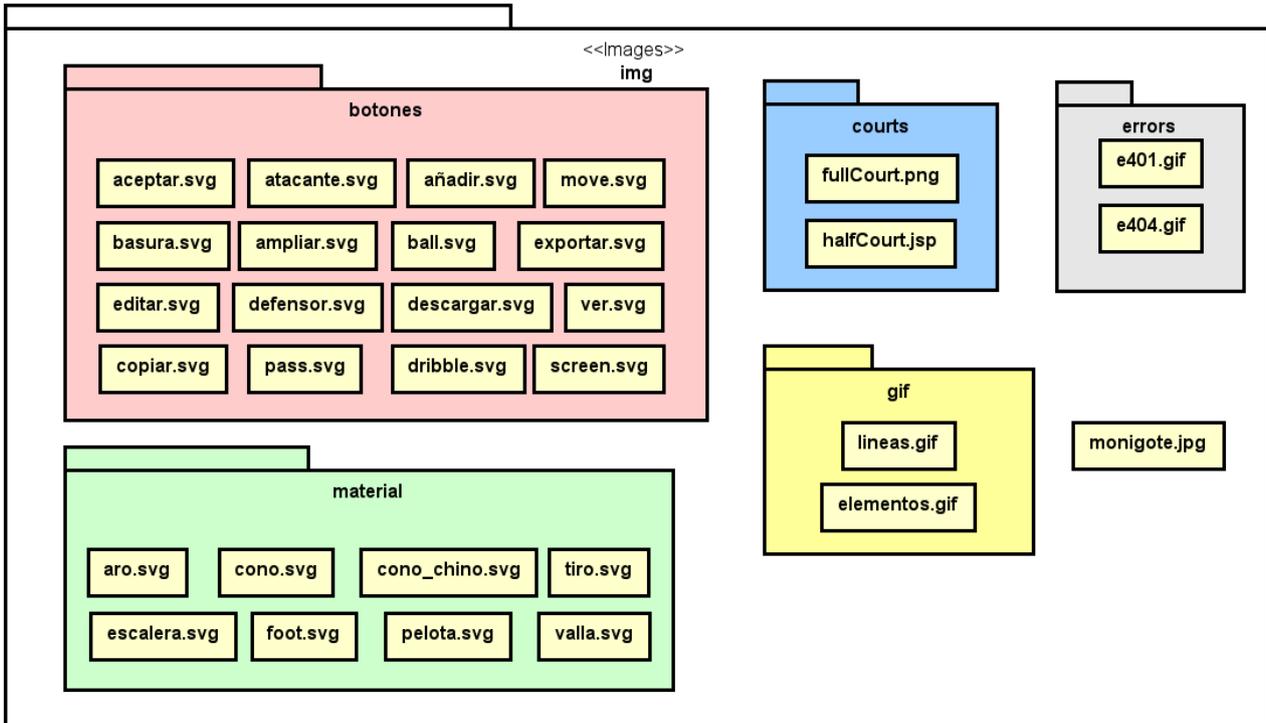


Figura 20: Archivos de imagen existentes

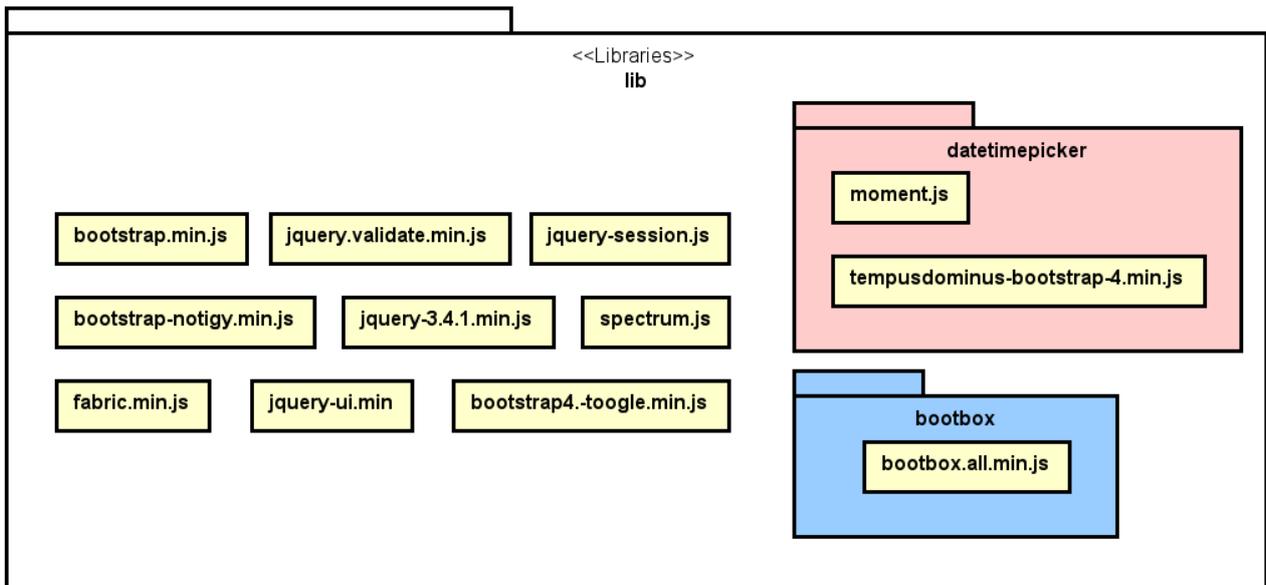


Figura 21: Librerías utilizadas

6.7.2 Paquetes del Servidor

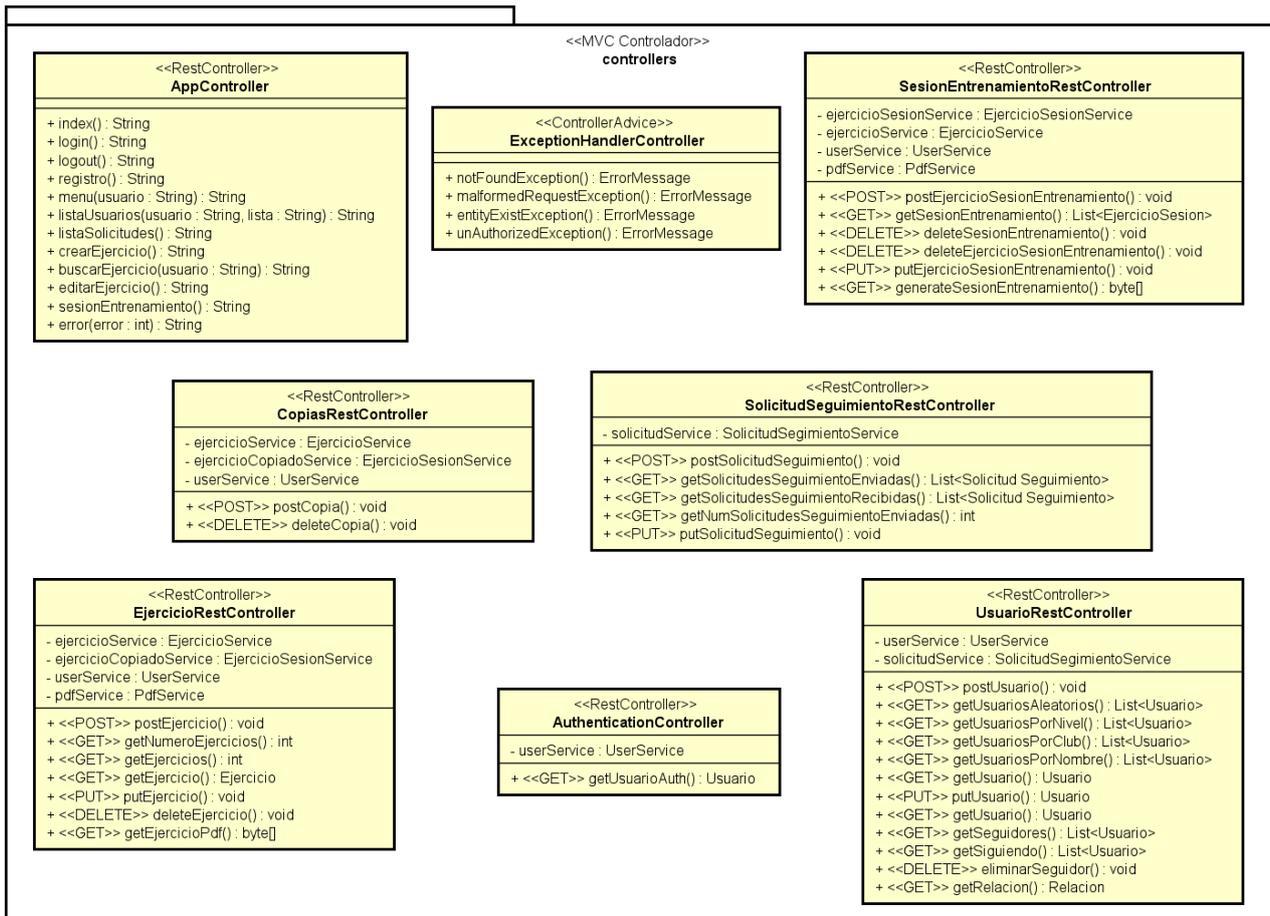


Figura 22: Controladores existentes

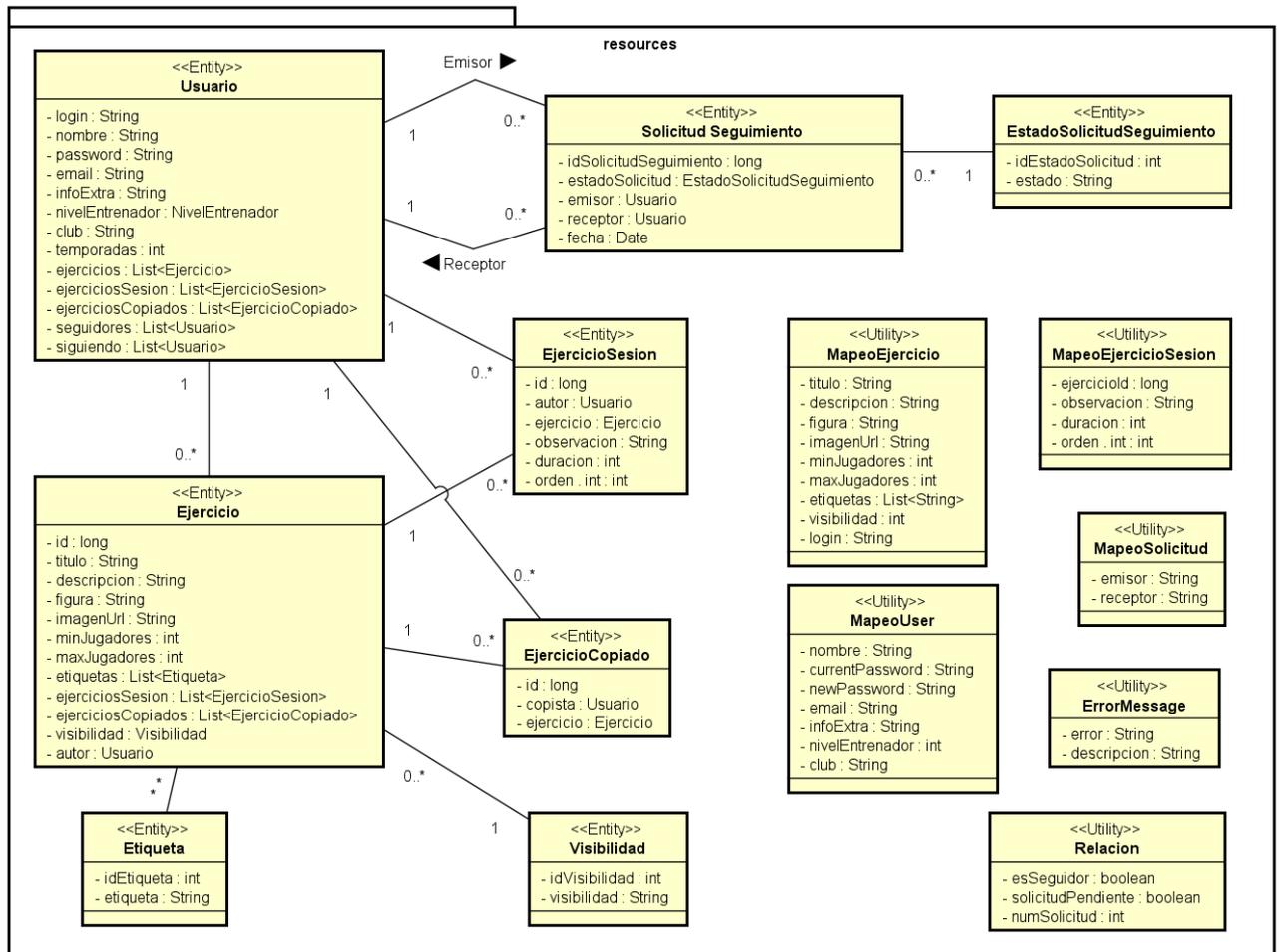


Figura 23: Entidades existentes

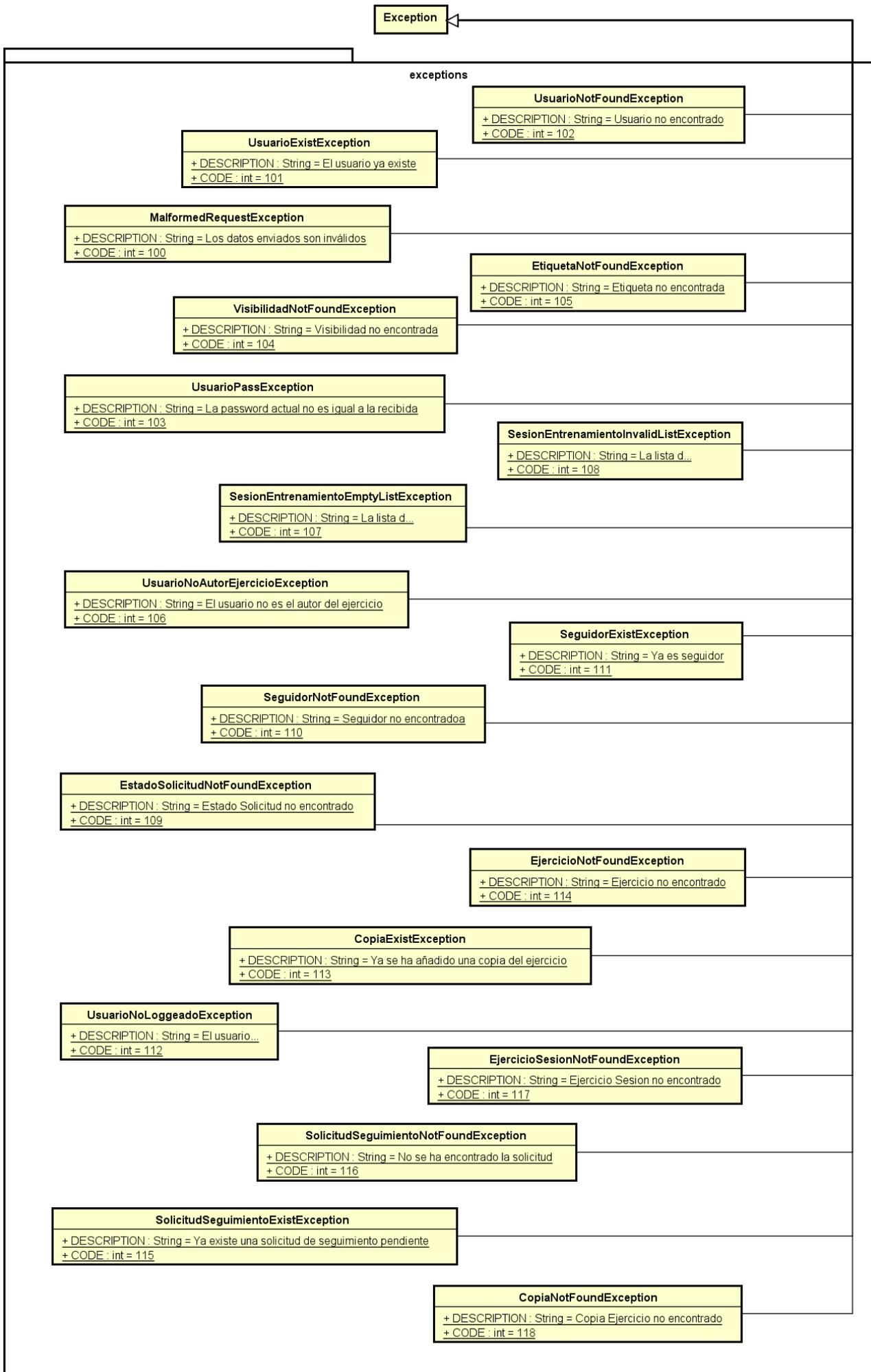


Figura 24: Excepciones utilizadas

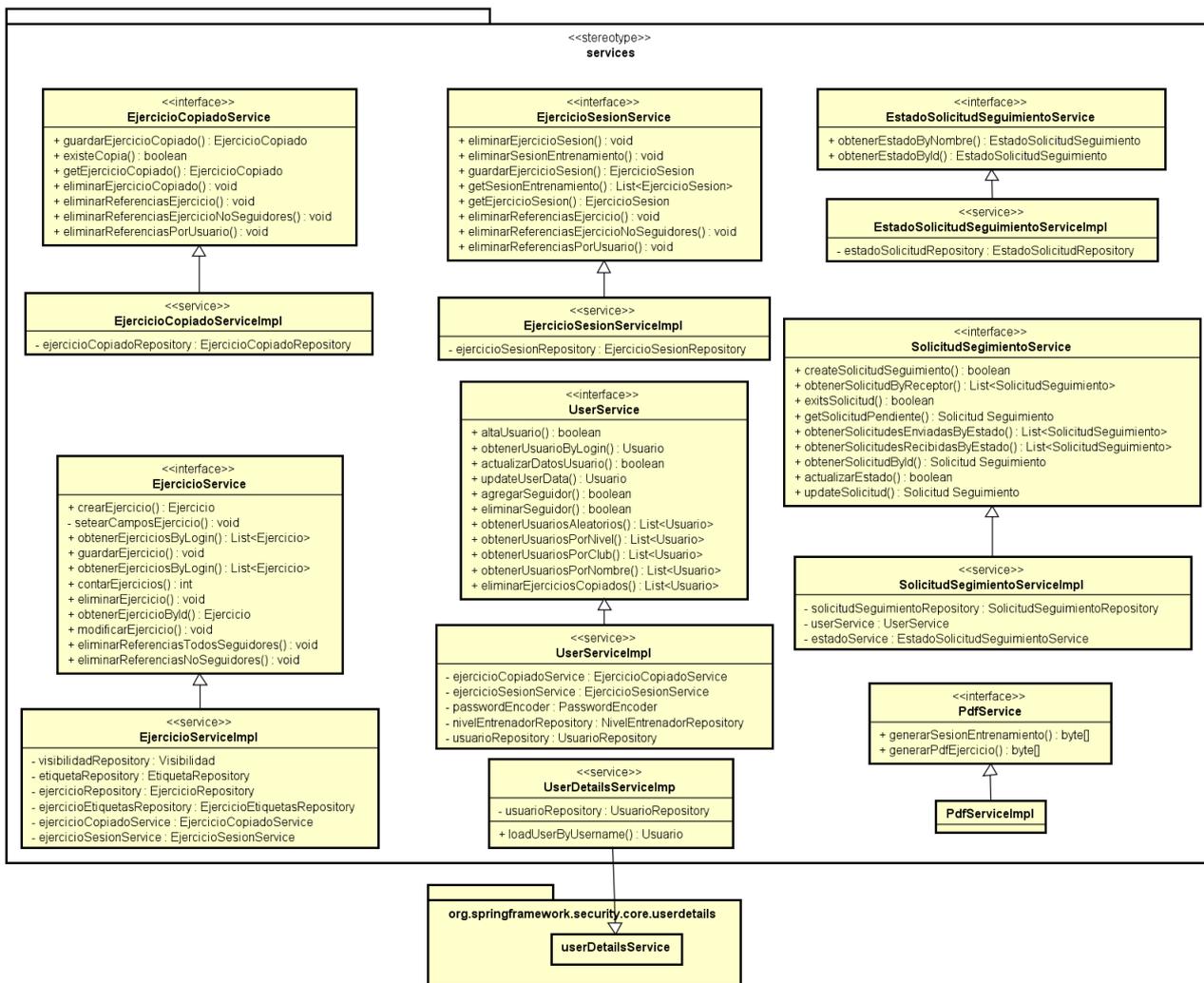


Figura 25: Servicios existentes

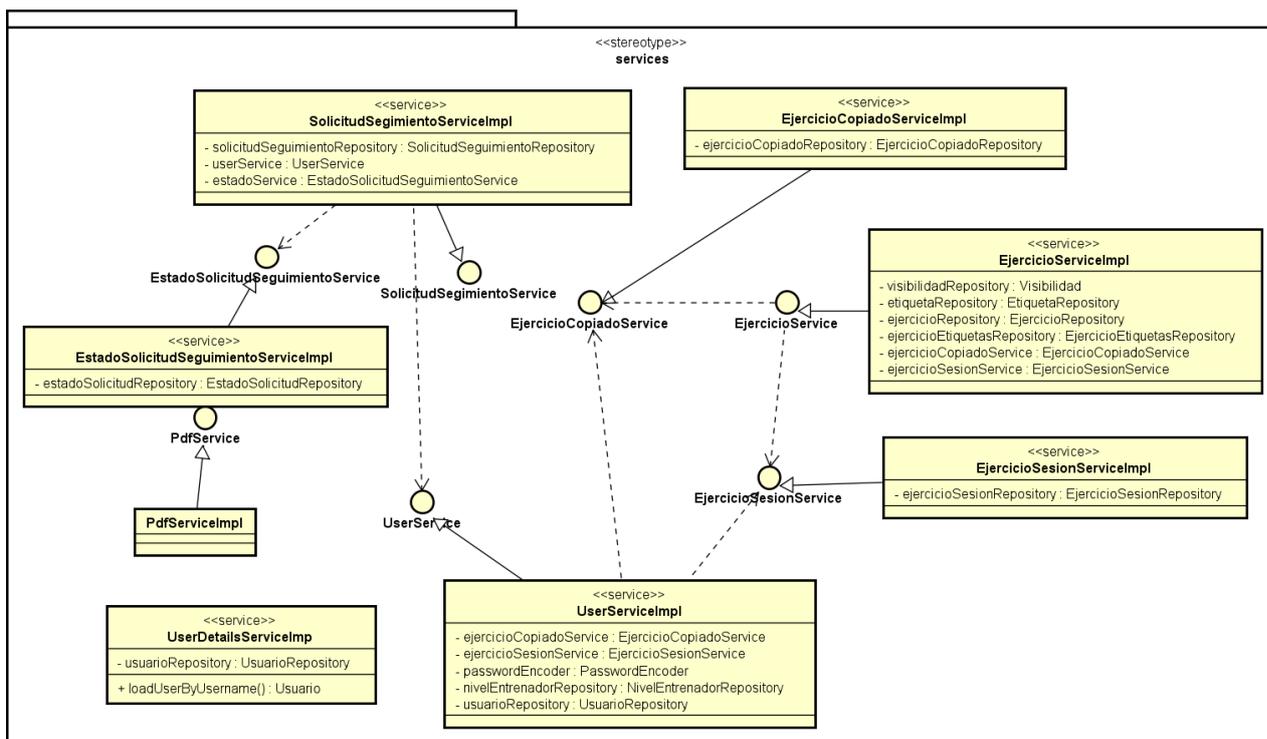


Figura 26: Relación entre servicios existentes

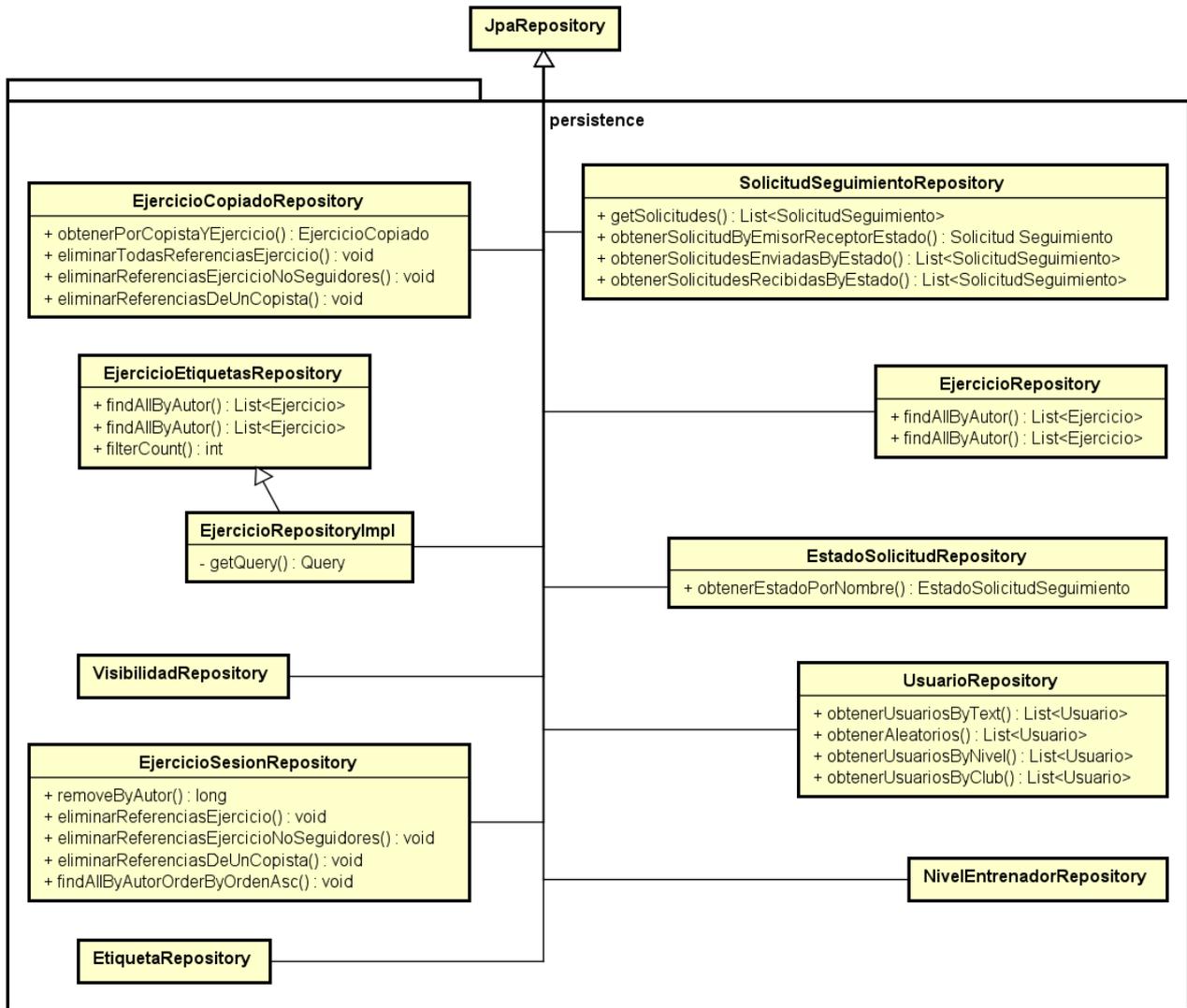


Figura 27: Repositorios existentes

Capítulo 7. Implementación

A continuación, se describe el proceso de implementación del proyecto. En primer lugar, se ha de hablar de la elección de la licencia del proyecto. Se ha escogido utilizar la Licencia Pública General de GNU [10], ya que controla los derechos de autor del proyecto y es muy utilizada en los proyectos de software libre como este. Esta licencia permite que cualquier persona utilice, comparta y modifique el proyecto siempre y cuando mantenga sus libertades originales.

El objetivo es que cualquier persona que quiera pueda utilizar este proyecto, ampliándolo si cabe, para poder facilitar aún más las labores a los entrenadores.

7.1 Entorno de desarrollo

Se ha optado por utilizar el IDE SpringBoot Tools Suite (STS) para el desarrollo del proyecto. Este IDE es gratuito y se puede instalar de manera sencilla en los principales sistemas operativos. Está basado en el IDE de Eclipse, uno de los que hemos aprendido a utilizar durante la carrera. Además, STS es el IDE que utilizo en la empresa.

En cuanto al sistema operativo, la mayor parte del proyecto ha sido desarrollado en un sistema operativo Linux básico como es Ubuntu, que trae por defecto instalado el sistema de control de versiones que se ha utilizado: GIT.

Para la parte de documentación, se han utilizado indistintamente Windows y Linux ya que se ha utilizado un editor online y el programa Astah, para los diagramas, es fácilmente instalable en ambos.

7.2 Implementación de la Base de Datos

Como SGBD se ha utilizado MariaDB. Inicialmente se pensó en utilizar MySQL, pero esto se descartó ya que Oracle lo compró y dejó de ser un proyecto OpenSource. MariaDB es la bifurcación de MySQL de código abierto que mantiene a los desarrolladores originales. Por ello, ambos SGBD son muy similares y muchas veces la documentación de uno sirve para el otro.

MariaDB sigue un esquema relacional, que para una aplicación como esta es lo que se debe utilizar. Se descartaron opciones como MongoDB, que se basa en colecciones, ya que esta aplicación no va a albergar una cantidad desmesurada de datos.

Las queries que se han utilizado en el proyecto son muy sencillitas y muchas vienen encapsuladas dentro del framework de JPA + Hibernate. La query que más complicación ha tenido ha sido la que permite buscar los ejercicios mediante filtros, ya que se ha tenido que desarrollar de forma tradicional en lugar de utilizar las ventajas que proporcionan estos frameworks. La dificultad de esta query reside en que es variable en cuanto al número de filtros que el usuario haya decidido introducir. La query queda de la siguiente manera:

```
select distinct ejercicio.* from (select * from ejercicio as e where
e.autor_LOGIN=':login' union select ej.* from ejercicioCopiado ec, ejercicio ej
where ec.COPISTA=':login' and ej.id=ec.ID_EJERCICIO) as ejercicio where
ejercicio.id>0
and ejercicio.TITULO collate UTF8_GENERAL_CI like '%%s%'
and ejercicio.MIN_JUGADORES<=:min
and ejercicio.MAX_JUGADORES<=:max
and ejercicio.VISIBILIDAD>=1
and ejercicio.VISIBILIDAD=2
and ejercicio.id in (select ejercicioSub.id as id from ejercicio ejercicioSub,
ejercicio_etiqueta ejercicio_etiquetaSub where ejercicioSub.id =
ejercicio_etiquetaSub.Ejercicio_id and
ejercicio_etiquetaSub.etiquetas_ID_ETIQUETA=':etiqueta')
```

Siendo el último “and” repetido tantas veces como etiquetas contenga la consulta.

7.3 Control de versiones

Se ha optado por utilizar Git para llevar a cabo el control de versiones de la aplicación ya que es el que hemos venido utilizando durante la carrera. El repositorio donde alojamos el código del proyecto es el proporcionado por la escuela: GitLab.

La url del proyecto es:

https://gitlab.inf.uva.es/rubgonz/gestor_ejercicios_tfg

La ventaja del GitLab de la escuela es que la tutora puede ir viendo el desarrollo del proyecto en caso de ser necesario. Haber utilizado GitHub hubiera supuesto un despliegue en Heroku mucho más sencillo. Se tratará el tema del despliegue más adelante.

La filosofía para el control de versiones ha sido “Rama por tarea”. Partiendo de la rama inicial máster, se ha creado una rama de desarrollo “develop” de donde se han ido sacando e integrando ramas según se iban empezando y terminando las tareas. Finalmente se ha volcado todo sobre la rama máster una vez terminado el proyecto.

7.4 Organización del código

En este apartado se explica cómo está organizado el código. Para ello, se ha tomado como referencia la organización del código de otro TFG anterior, que utilizaba las mismas tecnologías (especialmente para el backend). Esta práctica es habitualmente utilizada no sólo en el ámbito educativo, sino también en el empresarial. Los nuevos proyectos tienden a reflejarse sobre proyectos anteriores. La autora del proyecto mencionado es Cristina Gómez [11], y se puede acceder a su proyecto en el apartado de Bibliografía.

7.4.1 Parte del servidor:

Almacenado en *src/main/java/es/uva/tfg/gestorejerciciosbaloncesto*. Aquí reside todo el código del servidor almacenado en los paquetes que se han visto en el apartado de diseño. Dentro del directorio mencionado, se han creado los siguientes paquetes que contienen clases homogéneas.

- *conf*: Contiene las clases de configuración del proyecto y de seguridad.
- *controllers*: Contienen todos los controladores de la aplicación que definen la API REST del proyecto. Se encargan de recibir las peticiones del cliente.
- *services*: Se encuentran los servicios a los que llamarán los controladores para desarrollar la lógica de negocio. Contiene tanto las interfaces como sus implementaciones.
- *resources*: Almacena todas las entidades del dominio y otras clases de utilidad utilizadas de manera auxiliar.
- *persistence*: Se encuentran las interfaces que contienen todas las operaciones necesarias para realizar contra la base de datos.
- *exceptions*: Se encuentran todas las excepciones definidas.

7.4.2 Parte del cliente:

En *src/main/resources/static* encontramos la parte del código correspondiente a la capa de presentación del proyecto. En esta ruta existen diferentes directorios separados por su funcionalidad.

- *css*: Contiene los archivos de estilos,
- *img*: Contiene las imágenes y dibujos vectoriales utilizados en el frontend, tanto como elementos decorativos como iconos para los botones.
- *js*: contiene los archivos JavaScript, que se utilizan tanto para realizar las llamadas como para dotar de funcionalidad a las vistas.
- *lib*: Contiene las bibliotecas utilizadas en el cliente.

- templates: Encontramos todos los archivos HTML, salvo el index, y representan las vistas de la aplicación.
- index: Archivo HTML. Es la página landing de la aplicación.

7.5 Solución de problemas

Durante el desarrollo de un proyecto siempre aparecen pequeños contratiempos y problemas que hacen que el tiempo empleado acabe resultando mayor del estimado. Este proyecto no iba a ser distinto en este aspecto, pero a continuación comentaré dos problemas importantes que se han tenido que superar, el primero de carácter crítico y el segundo con cierta relevancia, pero no tan importante. Ambos problemas tienen lugar en la implementación de los casos de uso de los ejercicios, razón principal de la excesiva duración de ese apartado.

El primer problema tiene que ver con la necesidad de dibujar una línea ondulada que representa el desplazamiento de un jugador con balón, algo necesario y vital para este proyecto. Buscando en internet, como caída del cielo, aparece una solución muy sencilla de adaptar con lo que se estaba desarrollando para poder hacerlo. Consistía crear una especie de “clase” que heredaba de la clase utilizada para construir líneas. Con ello fue muy rápido pintar esa línea y almacenar los ejercicios.

El problema llega cuando, tras almacenar el dibujo, al intentar recuperarlo para implementar el caso de uso “Editar ejercicio”, la librería muestra un error al cargar de nuevo esta línea que había sido creada a mano. Faltaba un método de recarga que el autor no puso inicialmente. La solución pasó por crear una pregunta en la web stackoverflow.com referenciando al post original. Días después el propio autor del post original añadió la función que faltaba y se solucionó el problema. Url del post:

<https://stackoverflow.com/questions/58614245/fabricjs-loadfromjson-subclass-of-line>

El segundo problema consistía en que la biblioteca utilizada, al recuperar el JSON que representa al ejercicio, lo hace utilizando una llamada asíncrona que se llamaba al cargar la página. Esto suponía que la página se cargaba antes de recibir la respuesta y el ejercicio no mostraba más que un fondo blanco. Una primera solución fue hacer dormir a la página medio segundo tras lanzar la función asíncrona. De esta manera, cuando seguía con la ejecución, habitualmente ya se había recibido de vuelta la respuesta y el ejercicio quedaba dibujado.

Tras la buena experiencia con el problema anterior, se publicó otra pregunta en StackOverflow y, afortunadamente, la misma persona que ayudó previamente lo hizo de nuevo en esta ocasión. Explicó cómo hacer introducir el resultado en una llamada que se ejecutase al terminar la llamada asíncrona, así se pintaba el dibujo cuando se recibía la respuesta. Url del post:

<https://stackoverflow.com/questions/58635365/from-json-to-image-using-fabricjs>

Por desgracia, tras haber dado con la solución, finalmente es descartada y sustituida por almacenar en la base de datos la imagen del ejercicio, ya que sería necesaria posteriormente para generar los archivos PDF. Una vez que ya se almacena la imagen no es necesario cargar el dibujo y generar la imagen después en el frontend. La razón de esta primera idea era evitar almacenar la imagen en la base de datos, pero ha sido necesario ya que no hay manera de generarla en el lado del servidor.

7.6 CI/CD

Como se ha mencionado anteriormente, el proyecto está desplegado en un servidor gratuito de tal manera que pudiera ser probado por usuarios reales.

Se decidió utilizar Heroku, en su versión gratuita ya que, aunque cuenta limitaciones, nos permite ver desplegado el proyecto de manera muy sencilla y en un dominio elegido por el usuario (casi íntegramente).

El proyecto está desplegado en la siguiente dirección:

<https://gestor-ejercicios-baloncesto.herokuapp.com/>

Existen dos limitaciones principales por parte de Heroku.

- 1) Si la aplicación no es usada durante 30 minutos por nadie, Heroku mata el hilo de la aplicación y no crea uno nuevo hasta que alguien solicita los servicios de nuevo. Esto supone que, si un usuario es el primero que accede en un periodo largo de tiempo, la carga de la página puede rondar los 20 segundos hasta que se arranca todo de cero.
- 2) La base de datos proporcionada por Heroku, JawsDB, tiene una limitación de 10 conexiones simultáneas y una capacidad máxima de 0.005 GB.

Estas limitaciones hacen que la aplicación no tenga escalabilidad de usuarios, pero es suficiente para permitir a un grupo de 10-15 usuarios utilizarla sin problemas.

7.7 Patrones utilizados

En este apartado se describirán los patrones utilizados para implementar el proyecto. Se puede encontrar más información sobre ellos en los libros de Larman [12], Fowler [13] y Gamma [14].

7.7.1 Patrón MVC

Como se ha mencionado varias veces a lo largo de la memoria, Spring Boot es un framework que utiliza de manera constante el patrón Modelo-Vista-Controlador.

El MVC aboga por separar la lógica de negocio y los datos de la presentación de los mismos.

La vista, también llamada capa de presentación o interfaz de usuario, debe transmitir todas las peticiones a la capa de dominio (lógica de negocio) que será la encargada de gestionarlas. La capa de presentación puede ser conocedora de los objetos del dominio, pero nunca al revés.

Aplicamos el patrón MVC al utilizar la aplicación completa. El cliente representaría la capa de presentación. En el servidor, el paquete de controladores haría las funciones de Controlador (como su propio nombre indica), dejando a los servicios, entidades y repositorios la función de modelo.

7.7.2 Patrón DAO

El patrón Data Access Object es un patrón de acceso a datos que desacopla el negocio de la fuente de datos utilizada, haciendo así equivalente que recuperemos los datos de una base de datos, de archivos XML, etc.

DAO se encarga de gestionar la conexión con la fuente de datos, recuperando y almacenando la información. Habitualmente, DAO implementa una interfaz permitiendo así que todas las clases tengan una forma unificada de acceder contra ellos sin conocer la fuente ni cómo acceder a ella.

DAO facilita la transparencia y migración de los componentes centralizando todo en una sola capa. En el proyecto, esta capa está representada por el paquete “persistence” del lado del servidor. En ella podemos encontrar una clase DAO por cada una de las entidades.

Spring y JPA se encargan de llevar a cabo el patrón DAO. Todas las clases del paquete extienden de una clase JPA que cuenta con las operaciones CRUD básicas, las cuales simplifican bastante la gestión de queries ya que se llevan a cabo de forma transparente, dejando como trabajo al desarrollador únicamente las queries más elaboradas.

7.7.3 Patrón ID

Spring utiliza de manera constante la Inyección de Dependencias en su manera de trabajar. En el momento que se arranca, el contenedor de Spring crea y conecta objetos entre sí. Estos objetos se denominan Beans.

Los Beans se crean y conecta conforme a las anotaciones que se les haya indicado en código, y que se comprobarán durante el despliegue de la aplicación.

Se puede apreciar un ejemplo observando los controladores y los servicios del servidor. Cada controlador tiene dependencias hacia unos determinados servicios que son inyectados en los beans durante el arranque de Spring.

7.7.4 Patrón GRASP Alta Cohesión - Bajo Acoplamiento

Se puede definir la cohesión en términos software como las responsabilidades que posee un módulo. Un módulo con alta cohesión realizará una sola responsabilidad, teniendo sus funcionalidades mucha relación entre sí.

En términos software el acoplamiento puede definirse como la relación existente entre los distintos módulos de una aplicación. Bajo acoplamiento significa que los módulos son independientes entre sí y que un módulo no conoce cómo funcionan de manera interna los demás.

Es muy buena práctica que las aplicaciones que desarrollemos se vean provistas de una alta cohesión y un bajo acoplamiento ya que genera un código mucho más mantenible y que puede ser probado de manera más sencilla.

En el proyecto, podemos encontrar ejemplos de este principio en diversos lugares. Los servicios, por ejemplo, se encargan de llevar a cabo las funciones específicas que tienen que ver con entidades concretas. También, estos servicios no se entrelazan y mantienen su funcionalidad aislada unos de otros.

Capítulo 8. Plan de pruebas y evaluación

En este apartado se muestra la batería de pruebas realizada. Se ha probado la aplicación al completo, pero sólo se describen algunas de las pruebas de caja negra sobre los casos de uso más importantes.

También se recoge la retroalimentación obtenida tras las pruebas de usuario.

8.1 Casos de prueba

Las Tablas 75 – Tabla 104 recogen las pruebas de caja negra realizadas relativas a los casos de uso antes mencionadas.

8.1.1 Casos de uso de los ejercicios

P-1	Crear un nuevo ejercicio
Descripción	El usuario crea un nuevo ejercicio para añadirlo junto con el resto de sus ejercicios.
Entrada	El usuario dibuja la figura del ejercicio y rellena todos los campos del formulario. El usuario incluye también las etiquetas pertinentes. Finalmente pulsa sobre guardar.
Resultado esperado	El ejercicio es guardado en el almacenamiento persistente y se muestra un mensaje de retroalimentación informando al usuario.

Tabla 74: Caso de prueba “Crear un nuevo ejercicio”

P-2	Crear un ejercicio sin añadir etiquetas
Descripción	El usuario completa los pasos para crear un ejercicio a excepción de incluir las etiquetas descriptivas.
Entrada	El usuario dibuja la figura del ejercicio y rellena todos los campos del formulario. El usuario no incluye las etiquetas pertinentes. Finalmente pulsa sobre guardar.
Resultado esperado	El formulario muestra un error e indica que se introduzca al menos una etiqueta

Tabla 75: Caso de prueba “Crear un ejercicio sin añadir etiquetas”

P-3	Abandonar la creación de un ejercicio
Descripción	El usuario decide no continuar con el proceso de creación de un ejercicio.
Entrada	El usuario se encuentra en la pantalla de creación y pulsa sobre cualquier enlace para salir.
Resultado esperado	El sistema muestra un cuadro de diálogo avisando de que se perderán los progresos conseguidos hasta el momento.

Tabla 76: Caso de prueba “Abandonar la creación de un ejercicio”

P-4	Cambiar la visibilidad de un ejercicio por una más restrictiva 1
Descripción	El usuario cambia la visibilidad de un ejercicio de público a protegido
Entrada	El usuario modifica un ejercicio y cambia la visibilidad actual de “Público” a “Protegido”.
Resultado esperado	El ejercicio cambia la visibilidad y desaparece de entre los ejercicios tomados prestados de los usuarios que no sean seguidores de él

Tabla 77: Caso de prueba “Cambiar la visibilidad de un ejercicio por una más restrictiva 1”

P-5	Cambiar la visibilidad de un ejercicio por una más restrictiva 2
Descripción	El usuario cambia la visibilidad de un ejercicio de protegido a privado

Entrada	El usuario modifica un ejercicio y cambia la visibilidad actual de “Protegido” a “Privado”.
Resultado esperado	El ejercicio cambia la visibilidad y desaparece de entre los ejercicios tomados prestados de todos los usuarios que lo tomasen.

Tabla 78: Caso de prueba “Cambiar la visibilidad de un ejercicio por una más restrictiva 2”

P-6	Eliminar ejercicio propio
Descripción	El usuario elimina uno de los ejercicios que ha creado
Entrada	El usuario selecciona un ejercicio que eliminar y confirma su eliminación.
Resultado esperado	El ejercicio desaparece de los ejercicios del usuario, y si hubiere sido tomado prestado, desaparece también de las listas de los otros usuarios.

Tabla 79: Caso de prueba “Eliminar ejercicio propio”

P-7	Eliminar ejercicio tomado prestado
Descripción	Eliminar un ejercicio de la lista de ejercicios del usuario que ha sido tomado prestado.
Entrada	El usuario selecciona el ejercicio a eliminar y confirma su eliminación.
Resultado esperado	El ejercicio desaparece la lista de ejercicios del usuario

Tabla 80: Caso de prueba “Eliminar ejercicio tomado prestado”

P-8	Editar ejercicio propio
Descripción	Al pulsar editar debe abrirse la página de edición
Entrada	El usuario selecciona entre sus ejercicios uno y pulsa editar
Resultado esperado	El sistema abre la página de edición con el ejercicio cargado, tanto la figura como el formulario

Tabla 81: Caso de prueba “Editar ejercicio propio”

P-9	Editar ejercicio ajeno (o tomado prestado)
Descripción	El usuario desea editar un ejercicio que no es suyo
Entrada	El usuario escribe la dirección url de edición escribiendo un id de un ejercicio que no es suyo
Resultado esperado	El sistema debe redirigir a una página de error 401

Tabla 82: Caso de prueba “Editar ejercicio ajeno”

8.1.2 Casos de uso de buscar ejercicio

P-10	Buscar ejercicio filtrando por etiquetas
Descripción	El usuario decide filtrar entre sus ejercicios utilizando el panel de búsqueda.
Entrada	El usuario accede a la página de búsqueda de ejercicios y selecciona etiquetas entre las mostradas. El usuario pulsa sobre “Aplicar filtros”
Resultado esperado	El sistema devuelve una lista con los ejercicios que hayan sido etiquetados con las etiquetas que el usuario ha seleccionado en el panel de búsqueda.

Tabla 83: Caso de prueba “Caso de uso buscar ejercicio”

P-11	Buscar por título del ejercicio
Descripción	El usuario desea filtrar entre sus ejercicios buscando parte del nombre del ejercicio.
Entrada	El usuario introduce una cadena de caracteres que formen parte de los ejercicios a buscar. El usuario pulsa sobre “Aplicar filtros”
Resultado esperado	El sistema devuelve un listado de ejercicios, los cuales contienen la cadena introducida en su título.

Tabla 84: Caso de prueba “Buscar por título del ejercicio”

P-12	Buscar por número mínimo de participantes
Descripción	El usuario desea filtrar entre sus ejercicios buscando por número de participantes
Entrada	El usuario introduce un número mínimo y da a buscar.
Resultado esperado	El sistema devuelve un listado de ejercicios, los cuales contienen como número máximo de participantes el número introducido o uno menor.

Tabla 85: Caso de prueba “Buscar por número mínimo de participantes”

P-13	Buscar por una combinación de etiquetas inexistentes
Descripción	El usuario filtra entre los ejercicios utilizando una combinación de etiquetas no registrada en ninguno.
Entrada	El usuario introduce una/s etiquetas en el buscador que no han sido aplicadas a ningún ejercicio
Resultado esperado	El sistema muestra una retroalimentación indicando que no se han encontrado ejercicios

Tabla 86: Caso de prueba “Buscar por una combinación de etiquetas inexistentes”

8.1.3 Casos de uso de la sesión de entrenamiento

P-14	Añadir un ejercicio a la sesión de entrenamiento
Descripción	El usuario añade un ejercicio a la sesión de entrenamiento
Entrada	El usuario selecciona un ejercicio y lo añade a la sesión.
Resultado esperado	El usuario recibe una retroalimentación indicando que el ejercicio ha sido añadido y el ejercicio aparece al final de la lista de ejercicios de la sesión.

Tabla 87: Caso de prueba “Añadir un ejercicio a la sesión de entrenamiento”

P-15	Añadir un ejercicio de nuevo a la sesión
Descripción	El usuario añade un ejercicio a la sesión que ya estaba añadido
Entrada	El usuario selecciona un ejercicio introducido previamente y lo añade a la sesión
Resultado esperado	El usuario recibe una retroalimentación indicando que el ejercicio ha sido añadido y el ejercicio aparece al final de la lista de ejercicios de la sesión.

Tabla 88: Caso de prueba “Añadir un ejercicio de nuevo a la sesión”

P-16	Añadir una pausa a la sesión de entrenamiento
------	---

Descripción	El usuario añade una pausa a la sesión de entrenamiento
Entrada	El usuario pulsa sobre añadir pausa
Resultado esperado	La final de la lista de ejercicios se añade uno nuevo con el título Pausa y sin hacer referencia a ningún ejercicio

Tabla 89: Caso de prueba “Añadir una pausa a la sesión de entrenamiento”

P-17	Eliminar ejercicio de la sesión de entrenamiento
Descripción	El usuario elimina un ejercicio de la sesión de entrenamiento
Entrada	El usuario pulsa sobre el botón eliminar del ejercicio que ha seleccionado
Resultado esperado	El ejercicio es eliminado de la lista de ejercicios de la sesión sin pedir confirmación

Tabla 90: Caso de prueba “Eliminar ejercicio de la sesión de entrenamiento”

P-18	Descartar la sesión de entrenamiento
Descripción	Eliminar todos los ejercicios de la sesión de entrenamiento
Entrada	El usuario pulsa sobre el botón descartar y confirma.
Resultado esperado	La lista de ejercicios queda vacía.

Tabla 91: Caso de prueba “Descartar la sesión de entrenamiento”

P-19	Exportar la sesión de entrenamiento
Descripción	Exportar a un documento externo PDF la sesión con la lista de ejercicios tras haberles añadido al menos una duración a cada uno y haber escrito el nombre del equipo
Entrada	El usuario ha rellenado los campos de duración correctamente y pulsa sobre exportar. Después escribe el nombre del equipo y confirma.
Resultado esperado	se genera un archivo externo en formato PDF y se pregunta al usuario si desea descartar la sesión.

Tabla 92: Caso de prueba “Exportar la sesión de entrenamiento”

P-20	Modificar orden de un ejercicio de la sesión
Descripción	El usuario cambia el orden de un ejercicio situándolo en otro punto de la lista.
Entrada	El usuario clicca sobre uno de los ejercicios incluidos y sin soltar el ratón lo arrastra sobre la posición que desee.
Resultado esperado	El orden de la lista de ejercicios cambia.

Tabla 93: Caso de prueba “Modificar orden de un ejercicio de la sesión”

P-21	Exportar la sesión de entrenamiento con algún ejercicio con duración cero
Descripción	Exportar a un documento externo PDF la sesión con la lista de ejercicios tras haber dejado al menos uno con una duración de cero y haber escrito el nombre del equipo
Entrada	El usuario ha rellenado los campos de duración correctamente excepto en al menos uno y pulsa sobre exportar. Después escribe el nombre del equipo y confirma.

Resultado esperado	El sistema muestra una lista con los ejercicios que tienen una duración de cero minutos.
--------------------	--

Tabla 94 Caso de prueba “Exportar la sesión de entrenamiento con algún ejercicio con duración cero”

P-22	Exportar la sesión de entrenamiento sin incluir el nombre del equipo
Descripción	El usuario exporta la sesión con los ejercicios correctos, pero sin incluir el nombre del equipo.
Entrada	El usuario ha rellenado los campos de duración correctamente y pulsa sobre exportar. Después no escribe el nombre del equipo y confirma.
Resultado esperado	El sistema muestra un mensaje solicitando que se incluya un nombre para el equipo.

Tabla 95: Caso de prueba “Exporta la sesión de entrenamiento sin incluir el nombre del equipo”

8.1.4 Casos de uso del usuario

P-23	Modificar el nombre
Descripción	El usuario introduce un nuevo nombre
Entrada	Desde el Pop-Up de configuración-perfil se introduce como nuevo nombre “Rubén González” en lugar del actual “Rubén”
Resultado esperado	El sistema muestra un mensaje de retroalimentación de que el nombre ha sido cambiado.

Tabla 96: Caso de prueba “Modificar el nombre”

P-24	Modificar el nombre añadiendo caracteres que no sean letras o espacios
Descripción	El usuario introduce elementos no válidos en el nombre como números o caracteres especiales como el asterisco *
Entrada	El usuario añade un número en su nombre al final del nombre: Rubén Gonzalez82
Resultado esperado	El sistema muestra una retroalimentación indicando que solo puede incluir letras y espacios.

Tabla 97: Caso de prueba “Modificar el nombre añadiendo caracteres que no sean letras o espacios”

P-25	Modificar el email
Descripción	El usuario introduce un nuevo email.
Entrada	Desde el Pop-Up de configuración-seguridad se modifica el email actual por “ruben@gmail.com” y se introduce la contraseña actual.
Resultado esperado	El sistema muestra un mensaje de retroalimentación de que el email ha sido cambiado.

Tabla 98: Caso de prueba “Modificar el email”

P-26	Modificar el email con uno no válido.
Descripción	El usuario introduce un email que no coincide con la expresión regular. $([a-z][A-Z][0-9][\-_\.\,])+\@([a-z][A-Z][0-9][\-_\.\,])+[a-z][a-z][a-z]?$
Entrada	El usuario introduce como nuevo email “ejemplo.com”
Resultado esperado	El sistema muestra una retroalimentación indicando que el email no es válido.

Tabla 99: Caso de prueba “Modificar el email con uno no válido”

P-27	Añadir los datos informativos del usuario
Descripción	El usuario introduce datos sobre su experiencia como entrenador
Entrada	Desde el Pop-Up de configuración-perfil el usuario añade el tiempo que lleva entrenando, el club donde entrena, su nivel y una reflexión.
Resultado esperado	El sistema muestra un mensaje de retroalimentación de que la información ha sido almacenada.

Tabla 100: Caso de prueba “Añadir los datos informativos del usuario”

P-28	Buscar usuario por nombre
Descripción	Realiza una búsqueda de usuarios en función de su nombre o de su alias
Entrada	El usuario introduce una cadena de caracteres “ruben” para buscar los usuarios que la contengan en su alias o en su nombre.
Resultado esperado	Lista de usuarios con dos usuarios. rubén y ruben2

Tabla 101: Caso de prueba “Buscar usuario por nombre”

P-29	Buscar usuario por nivel
Descripción	Realiza una búsqueda de usuarios en función de su nivel
Entrada	El usuario introduce “Segundo nivel” para buscar los usuarios de nivel 2.
Resultado esperado	Lista de usuarios con un usuario. luisblanco11

Tabla 102: Caso de prueba “Buscar usuario por nivel”

P-30	Buscar usuario por club
Descripción	Realiza una búsqueda de usuarios en función del club que han introducido.
Entrada	El usuario introduce “palencia” para buscar clubes cuyo nombre contenga esa cadena
Resultado esperado	Lista de usuarios con tres usuarios. luisblanco, juanki, egm,

Tabla 103: Caso de prueba “Buscar usuario por club”

8.2 Resultado de las pruebas

En la Tabla 105, se detallan los resultados obtenidos en las pruebas. Se ha comprobado el funcionamiento desde los navegadores Firefox y Chrome. Las bibliotecas utilizadas son válidas para ambos navegadores por lo que los posibles fallos que se dieron para uno se dieron también para el otro.

En su mayoría son fallos provenientes de descuidos o despistes por parte del desarrollador al implementar la lógica.

Prueba	Resultado	Solución
P-1	Ok	-
P-2	Ok	-
P-3	Ok	-

P-4	Fallo: la visibilidad cambia, pero sigue apareciendo para los usuarios que lo habían tomado prestado	Se añade el código necesario para cumplir esta condición. Al probarlo de nuevo se obtiene el resultado esperado
P-5	Fallo: la visibilidad cambia, pero sigue apareciendo para los usuarios que lo habían tomado prestado	Se añade el código necesario para cumplir esta condición. Al probarlo de nuevo se obtiene el resultado esperado
P-6	Ok	-
P-7	Ok	-
P-8	Ok	-
P-9	Fallo: se mostraba un error que impedía cargar el ejercicio del otro usuario, pero no redirigía a la página de error	Se añade la redirección automática al recibir ese error en esta situación. Al probarlo de nuevo se obtiene el resultado esperado
P-10	Ok	-
P-11	Ok	-
P-12	Fallo: se mostraban los ejercicios con número mínimo de participantes \geq al introducido	Se modifica la query. Al probarlo de nuevo se obtiene el resultado esperado
P-13	Ok	-
P-14	Ok	-
P-15	Ok	-
P-16	Ok	-
P-17	Ok	-
P-18	Ok	-
P-19	Ok	-
P-20	Ok	-
P-21	Fallo: en producción, el mensaje de retroalimentación desaparecía de inmediato impidiendo ver el error sucedido	Se sustituye el mensaje embebido en el HTML por un Pop-Up. Al probarlo de nuevo se obtiene el resultado esperado
P-22	Fallo: no había retroalimentación y el sistema parecía quedarse colgado	Se añade un mensaje de retroalimentación. Al probarlo de nuevo se obtiene el resultado esperado.
P-23	Fallo: la retroalimentación mostraba un mensaje de error genérico impidiendo saber al usuario por qué el nombre introducido era incorrecto.	Se cambia el mensaje por uno más explicativo
P-24	Ok	-
P-25	Ok	-
P-26	Fallo: la retroalimentación mostraba un mensaje de error genérico impidiendo saber al usuario por qué el email introducido era incorrecto.	Se cambia el mensaje por uno más explicativo
P-27	Ok	-
P-28	Ok	-
P-29	Ok	-
P-30	Ok	-

Tabla 104: Resultados de las pruebas realizadas

8.3 Pruebas con usuarios reales

En este apartado se recoge la retroalimentación obtenida por parte de los usuarios a los que se les mostró el boceto en papel. Contiene tanto errores identificados, como sugerencias de mejora de la aplicación y opiniones en cuanto a la usabilidad.

	Edad	Retroalimentación
Entrenador 1	23	Sugiere que, tras registrarse, la aplicación entre directamente sin tener que acceder. Detecta que cuando sucede un error durante el registro no se indica el porqué del error. Piensa que debería pedirse una contraseña más elaborada. No le queda claro cómo añadir un ejercicio a la sesión de entrenamiento. Le resulta sencillo crear los ejercicios tras leer la pestaña de ayuda.
Entrenador 2	29	Le gusta el comportamiento y la sencillez de la aplicación. La considera intuitiva y muy útil. Alaba la parte de poder compartir ejercicios y exportarlos a formato PDF. Pide permiso para utilizarla con otros compañeros suyos.
Entrenador 3	22	Le gusta, pero echa de menos poder cambiar el alias. Se equivocó en el formulario y no tiene manera de cambiar el nombre de usuario después. Tiene dudas de qué diferencias hay entre los tipos de visibilidad. No tiene problemas para dibujar ejercicios.
Entrenador 4	30	No le gusta el hecho de tener que registrarse. Le gusta mucho la idea de “tomar ejercicios prestados”. Tiene problemas para pintar las flechas al dibujar los ejercicios. Las acaba sacando tras practicar. Echa en falta la posibilidad de dibujar los materiales y las fechas de otros colores para facilitar los ejercicios, especialmente la rotación. Indica que la aplicación es intuitiva. Experimentó problemas para descargar el PDF, no se le bajaba con la extensión del archivo, pero al escribirla se visualizaba perfectamente.
Entrenador 5	30	Le gusta la aplicación. Entiende el funcionamiento y es capaz de utilizarla sin problemas. No le gusta el flujo de que para pintar otra línea haya que deshabilitar primero la que se está utilizando. Sugiere la posibilidad de añadir más pizarras para un mismo ejercicio, los ejercicios con varias oleadas pueden necesitar más de una.
Entrenador 6	42	Alaba la aplicación y le gusta mucho el resultado. Le resulta fácil de usar e intuitiva.
Entrenador 7	57	No prueba la aplicación
Entrenador 8	21	Le gusta la velocidad con la que funciona la aplicación y su funcionalidad. No consigue dibujar ejercicios desde el teléfono móvil. También sugiere la posibilidad de añadir más pizarras para un mismo ejercicio.
Entrenador 9	19	Hace un uso básico de la aplicación. No encuentra problemas en utilizarla.
Entrenador 10	65	Le parece sencilla de utilizar, consigue crear un ejercicio sin problemas. Sugiere que los ejercicios podrían ser etiquetados también en función de la categoría.

Tabla 105 :Retroalimentación obtenida de los usuarios tras el uso de la web.

Tras recoger las opiniones de los diez usuarios que han probado la aplicación, por un lado, se han solucionado los errores que se detectaron, especialmente por el Entrenador 1, mostrando el motivo por el que fallaba la aplicación a la hora de hacer el registro.

En cuanto a las sugerencias aportadas, todas han sido valoradas, pero no han sido incluidas, anotándose muchas de ellas como trabajo futuro

La aplicación ha tenido buena acogida entre los usuarios, tanto que al final se ha llegado al límite de almacenamiento de la base de datos. Esto supone que, actualmente, la aplicación no funciona

correctamente ya que no es capaz de registrar nuevos usuarios, ejercicios, etc., pudiendo únicamente trabajar con los datos que se tienen almacenados.

Capítulo 9. Conclusiones y trabajo futuro

9.1 Conclusiones

9.1.1 Objetivos conseguidos:

- Se ha conseguido cumplir todos los requisitos planteados inicialmente.
- Se ha conseguido aprender las tecnologías nuevas que suponían un reto al principio del proyecto.
- Se ha conseguido afianzar el conocimiento de las tecnologías ya conocidas utilizadas durante el proyecto.
- Se ha conseguido desplegar la aplicación en un servidor externo y que otros usuarios la prueben.
- Se ha conseguido que estos usuarios queden satisfechos tras la prueba.

Estos objetivos suponen un éxito ya que en un proyecto real en el mundo laboral es importante que lleguen a conseguirse.

9.1.2 Objetivos no conseguidos:

- No se ha conseguido cumplir con los plazos estimados del proyecto.

Aunque los plazos marcados en el plan de proyecto han sido rebasados ampliamente, y el tiempo requerido ha sido mucho mayor que el tiempo estimado, el objetivo final era conseguir terminar y presentar el trabajo antes de marzo de 2020. Los plazos iniciales, aunque optimistas, estaban resguardados con un colchón amplio de tiempo para poder terminar antes de esta fecha.

9.2 Trabajo futuro

Existen varios aspectos que se han quedado en el tintero por falta de tiempo:

- Mejorar el aspecto de las alertas mostradas por la aplicación para que sean todas uniformes.
- Desarrollar más funcionalidad en el apartado social de la aplicación, creando un nuevo apartado “Grupo” donde varios usuarios puedan compartir sus ejercicios.
- Dotar de un servicio de mensajería a la aplicación.
- Establecer un servicio de recuperación de contraseña mediante correo electrónico.
- Crear un apartado de comentarios para los ejercicios donde el propio usuario u otros usuarios puedan añadir sus opiniones de los ejercicios.
- Acceso automático tras el registro a la aplicación.
- Añadir nuevas herramientas de diseño en los ejercicios que permitan cambiar los colores de las flechas y el material e incluir las nuevas etiquetas sugeridas por los usuarios.
- Implementar un servicio de votación de ejercicios que permita priorizarlos en las búsquedas.
- Posibilidad de incluir un ejercicio volátil en la sesión, que desaparezca tras exportarla.
- Ampliar el concepto de la aplicación para que no sólo sea un gestor de ejercicios, sino que también abarque el campo de la gestión de equipos, sus jugadores, etc., e incluso los clubes.

9.3 Valoración personal

El desarrollo de este proyecto supone poner fin a una etapa importante en la vida como es la formación universitaria. Este proyecto recoge todo lo que hemos aprendido durante estos años, desde los primeros pasos en la programación hasta los últimos relacionados con la gestión de proyectos.

Si hay algo que destacar de esta preparación que hemos recibido, no es el mayor conocimiento en unas u otras materias o tecnologías, sino la capacidad de resolución y reacción ante los posibles problemas que puedan surgir. Como decía Darwin, sobrevive quien mejor se adapta, y en ese aspecto, con el tiempo, nos hemos convertido en unos maestros. Este proyecto es prueba de ello.

Por último, me siento muy satisfecho de haber conseguido llevarlo a cabo y sobre todo del resultado final. Este proyecto era una espinita clavada que tenía desde hace mucho tiempo y haberlo podido realizar como trabajo de fin de grado supone para mí una doble satisfacción.

Capítulo 10. Manual de usuario

Guía básica de cómo utilizar la aplicación <https://gestor-ejercicios-baloncesto.herokuapp.com/>

10.1 Primer contacto

10.1.1 Inicio

Al acceder a la url encontraremos una página que nos introduce la aplicación y que nos permite efectuar dos opciones: acceder o registrarse.

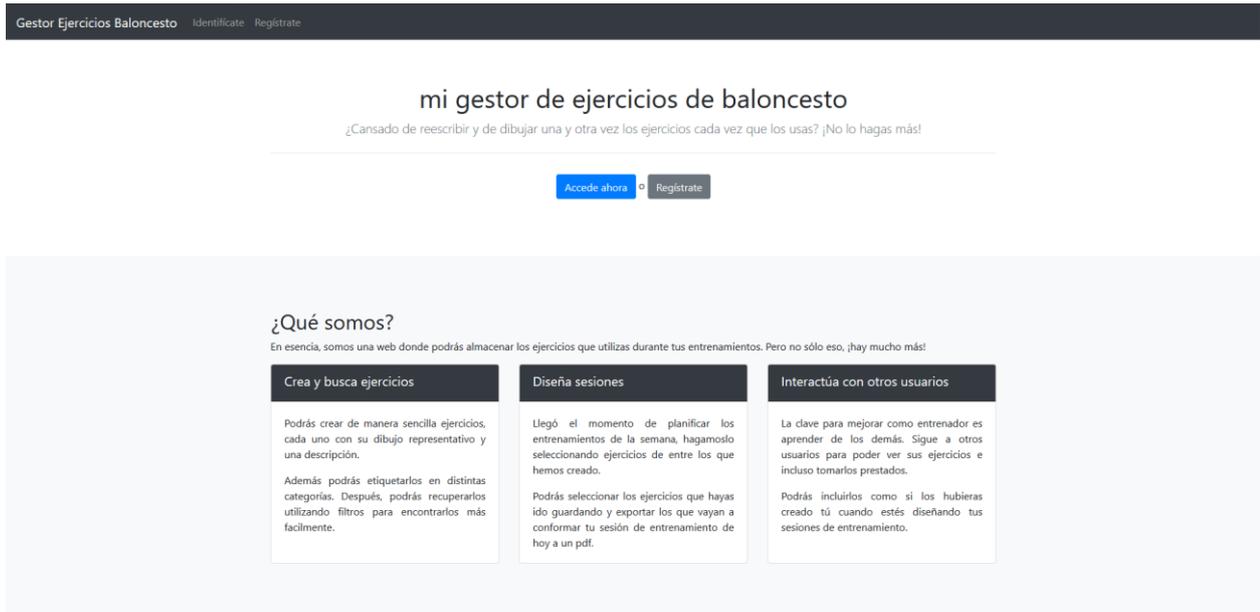


Figura 28

10.1.2 Registro

Para registrarse en la aplicación hay que rellenar un sencillo formulario.

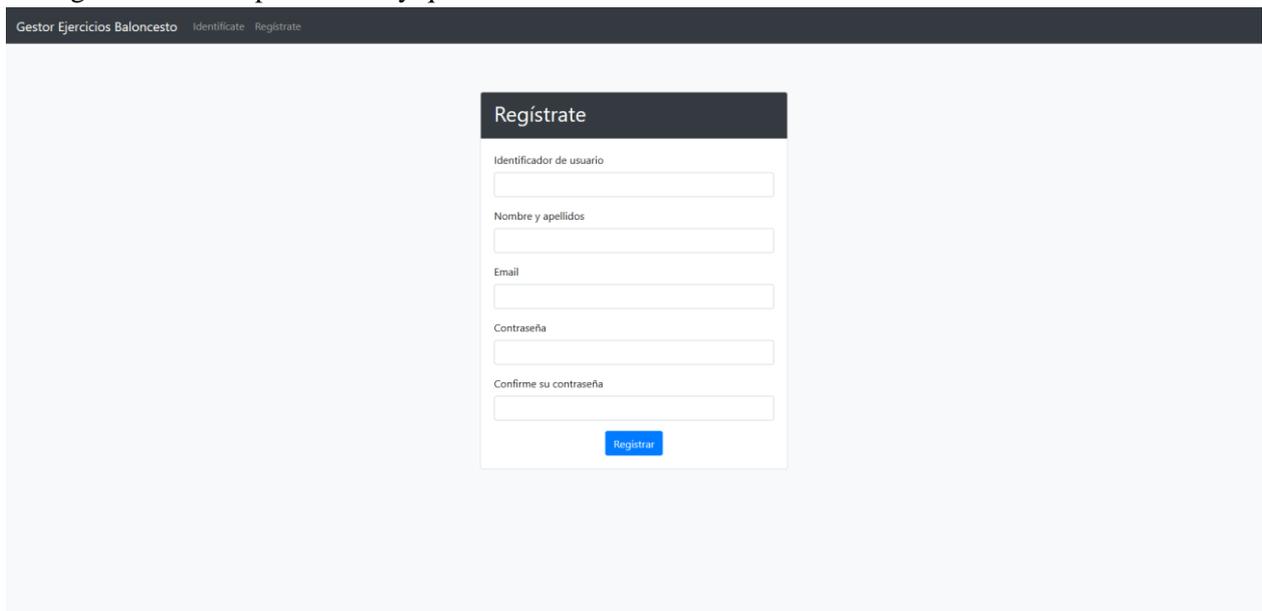
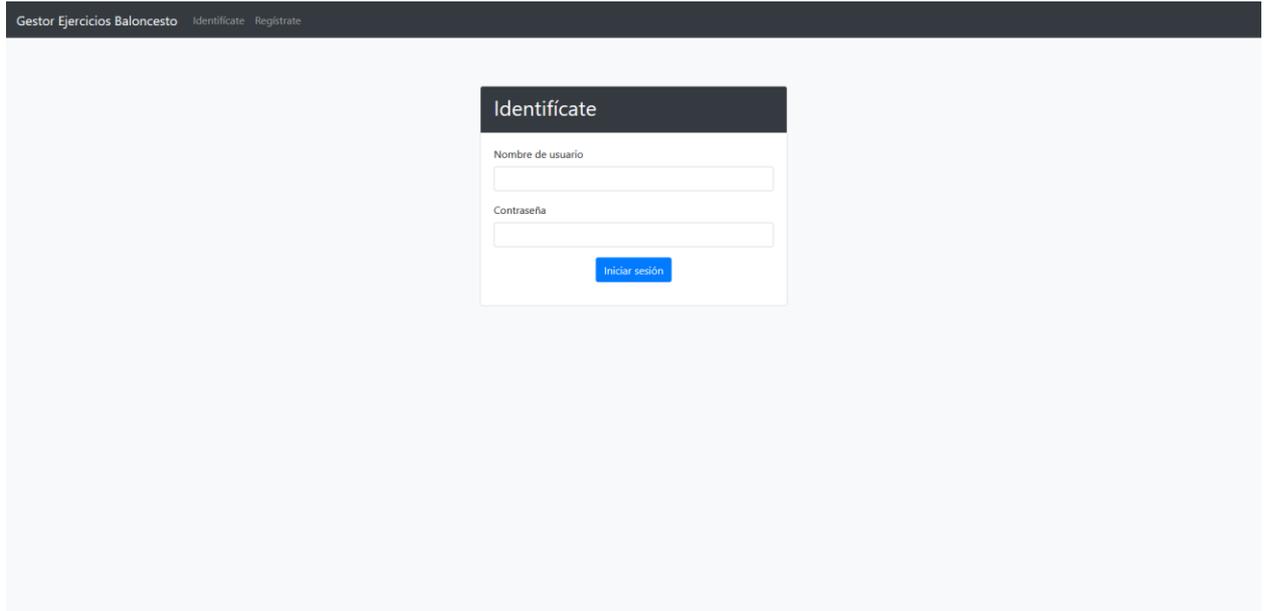


Figura 29

10.1.3 Acceso

Una vez registrados podremos acceder con nuestro usuario y nuestra clave.



The image shows a web application interface for user authentication. At the top, a dark navigation bar contains the text "Gestor Ejercicios Baloncesto" followed by "Identificate" and "Regístrate" in smaller font. Below this, the main content area features a central white box with a dark header titled "Identificate". Inside this box, there are two input fields: "Nombre de usuario" and "Contraseña". Below the "Contraseña" field is a blue button labeled "Iniciar sesión".

Figura 30

10.2 Perfil y configuración del usuario

10.2.1 Perfil del usuario

Siempre que nos identifiquemos en la aplicación veremos nuestro perfil de usuario. Desde aquí podremos acceder a toda la funcionalidad de la aplicación.

La herramienta principal para llevar a cabo las acciones que queramos desempeñar será la barra de navegación. En la parte izquierda veremos lo relacionado con los ejercicios. A la derecha, lo relacionado con los usuarios.

En nuestro perfil de usuario encontramos otras opciones como la configuración (que veremos más adelante) o la información que hayamos introducido. Debajo del nombre, podemos ver la cantidad de seguidores que tenemos, a cuántos usuarios seguimos, y un indicador de solicitudes pendientes.

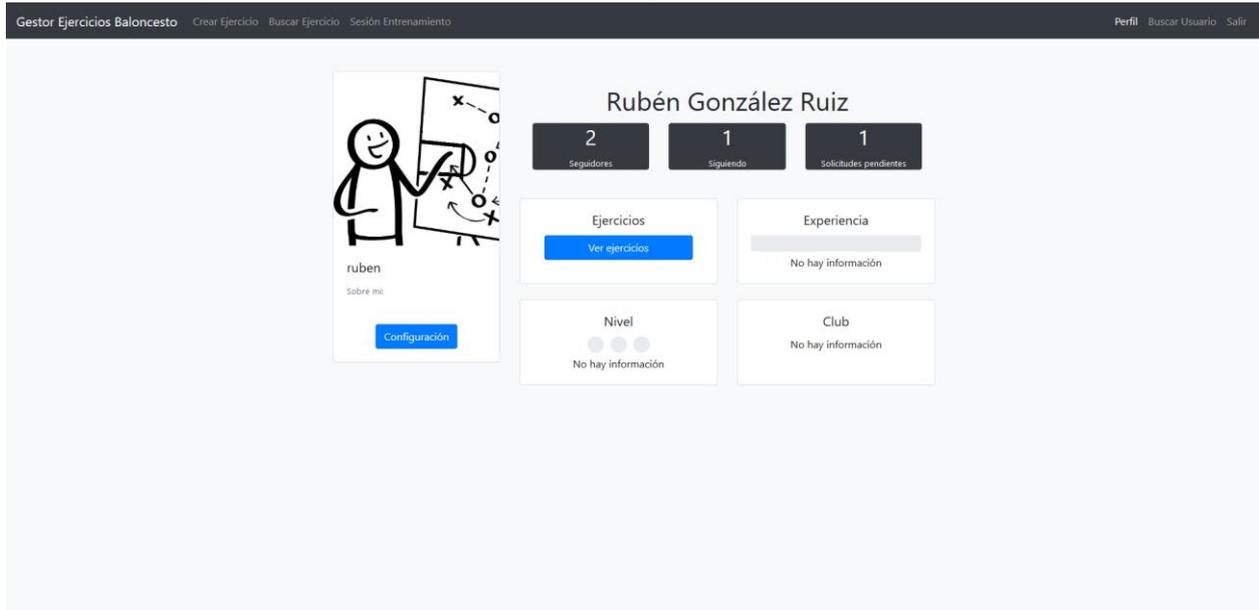


Figura 31

10.2.2 Configuración del usuario

Con respecto a la configuración del usuario aparecen dos partes.

En la primera podremos editar la información personal que queremos mostrar, no siendo obligatorio ningún campo salvo el nombre. Esta información es útil no solo para que otros usuarios puedan saber de nosotros, sino que también ayudará en los filtros de búsqueda del apartado "Buscar usuario".

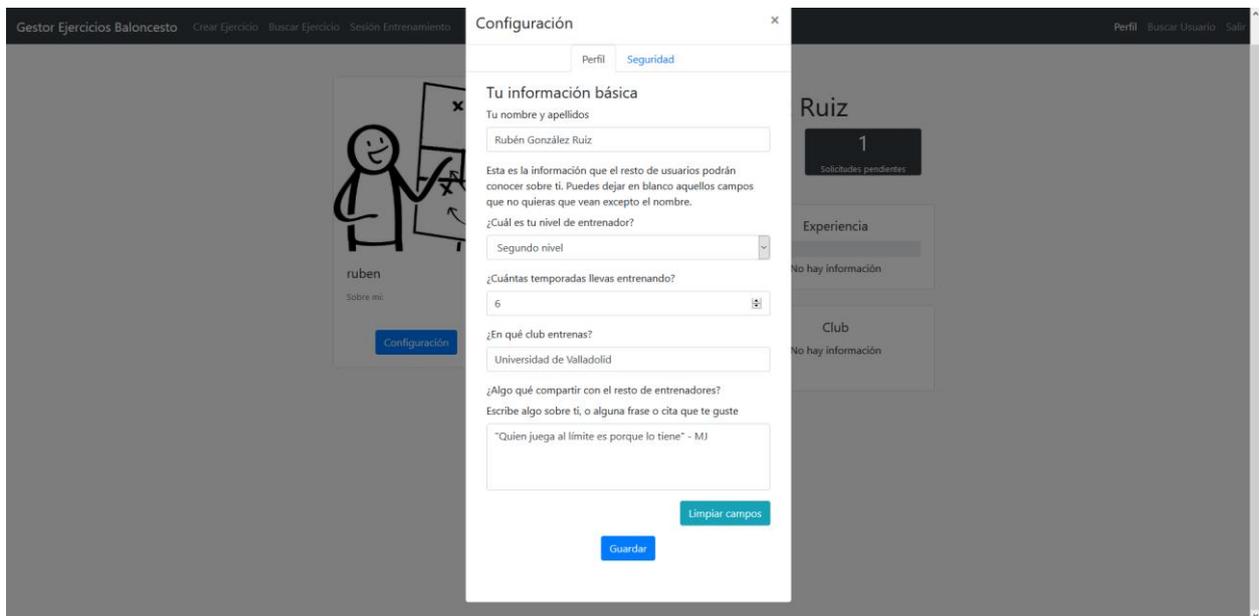


Figura 32

En segundo lugar, nos encontramos con las opciones de seguridad. Desde aquí podremos cambiar la dirección de correo electrónico introducida y cambiar la contraseña de la cuenta.

Al tratarse de un apartado de seguridad deberemos introducir la contraseña actual para poder registrar los cambios.



Figura 33

10.3 Crear ejercicio

10.3.1 Dibujar el ejercicio

Para crear un ejercicio, deberemos utilizar la pestaña “Crear ejercicio” de la barra de navegación. La aplicación nos mostrará una pantalla similar a la que encontramos en la siguiente imagen, pero solamente con el fondo. Podremos utilizar el toggle lateral para cambiar entre la pizarra de medio campo y de campo entero.

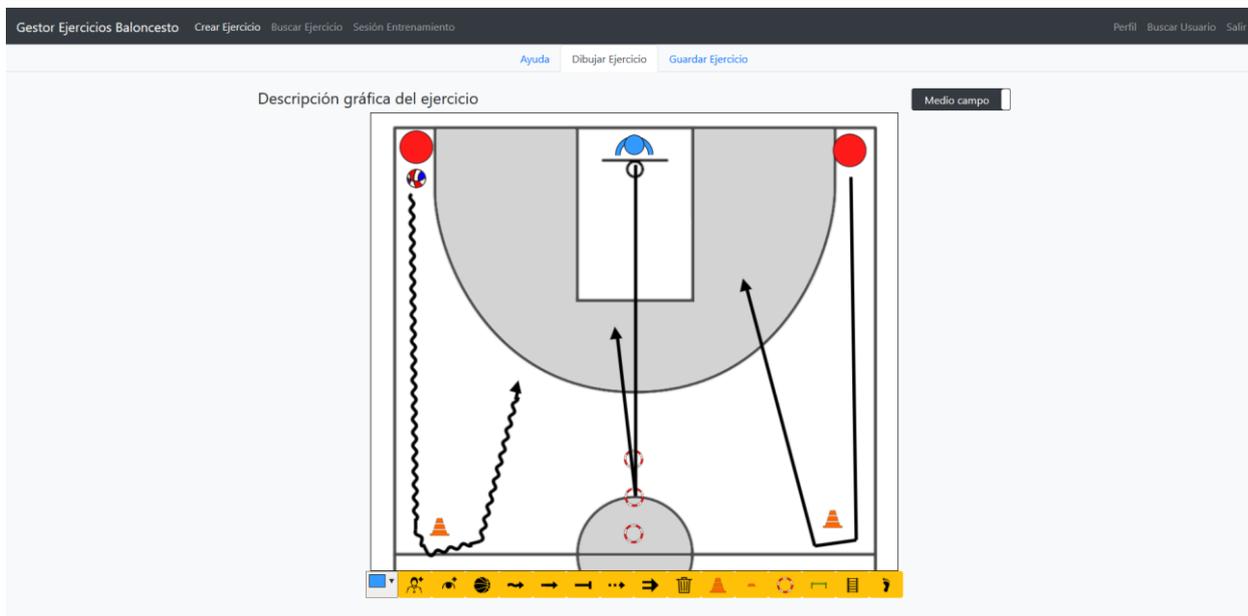


Figura 34

Debajo de la pizarra, encontramos los botones para interactuar con ella.



Figura 35

Todos los botones, salvo los de las flechas, pintan el objeto o persona a la que representan en el medio de la pantalla. Desde ese lugar podremos arrastrarlo hasta la posición de la pizarra que queramos. En algunos objetos, se permite la rotación de los mismos, pero nunca el cambio de tamaño. Para eliminar un objeto, simplemente hay que seleccionarlo y pulsar sobre el botón de eliminar (papelera).

El primero es un selector de color que nos permitirá modificar el color de los atacantes y de los defensores.

Los tres siguientes sirven para añadir atacantes, defensores y balones respectivamente. Los atacantes y defensores incluidos tomarán el color que muestre el selector en el momento que se incluyen.

A continuación, aparecen cinco iconos de flechas. Las cuatro primeras representan distintos tipos de desplazamiento que pueden realizar los jugadores: desplazamiento botando el balón, desplazamiento corriendo sin balón, desplazamiento que termina en un bloqueo y desplazamiento del balón por el aire (pase). Para utilizar una, la seleccionamos, quedando el resto de botones desactivados hasta que vuelva a pulsarse de nuevo. Haremos doble click sobre el punto donde queramos iniciar la flecha e iremos dando clicks simples por la pizarra para ir pintando los cambios de dirección que queramos reflejar. Para pintar la punta de la flecha, terminaremos con un doble click de nuevo.

La quinta flecha, que representa un lanzamiento a canasta, y los objetos que le siguen se pintan en el medio de la pantalla y tal como se muestran en el botón. Después pueden ser arrastrados y rotados según se necesite. Los objetos representan: cono alto, cono bajo, aro (Hula-Hoop), valla, escalera de coordinación y pisadas.

10.3.2 Añadir información del ejercicio

Una vez que el ejercicio se ha pintado, debe ser descrito. Habrá que darle un título, un mínimo de jugadores y el resto de campos que se solicitan. La visibilidad es un desplegable que tiene tres valores:

- Privado: solo yo puedo verlo y acceder a él
- Protegido: solo yo y mis seguidores podemos verlo y acceder a él.

- Público: todo usuario registrado puede verlo y acceder a él.

Figura 36

Por último, hay que añadir las etiquetas correspondientes al ejercicio. Es importante seleccionar bien las etiquetas que añadimos ya que luego podrán ser utilizadas en las opciones de búsqueda del apartado “Buscar ejercicio”. A continuación, se muestra una imagen con el desplegable para seleccionar etiquetas.

Figura 37

Una vez terminado el dibujo y la descripción del ejercicio pulsamos sobre el botón de guardado.

10.4 Gestión de nuestros ejercicios

Para poder acceder a nuestros ejercicios disponemos de dos opciones:

- Acceso a través del perfil: pulsando sobre el botón “Ver ejercicios”.
- Acceso a través de la barra de navegación: pulsando sobre la pestaña “Buscar ejercicios”

Se presentará ante nosotros una página con nuestros ejercicios (y los que hayamos tomado prestados) y una herramienta para el filtrado de ejercicios. Los ejercicios se van mostrando de cinco en cinco, pudiendo ver los siguientes utilizando el índice de paginación.

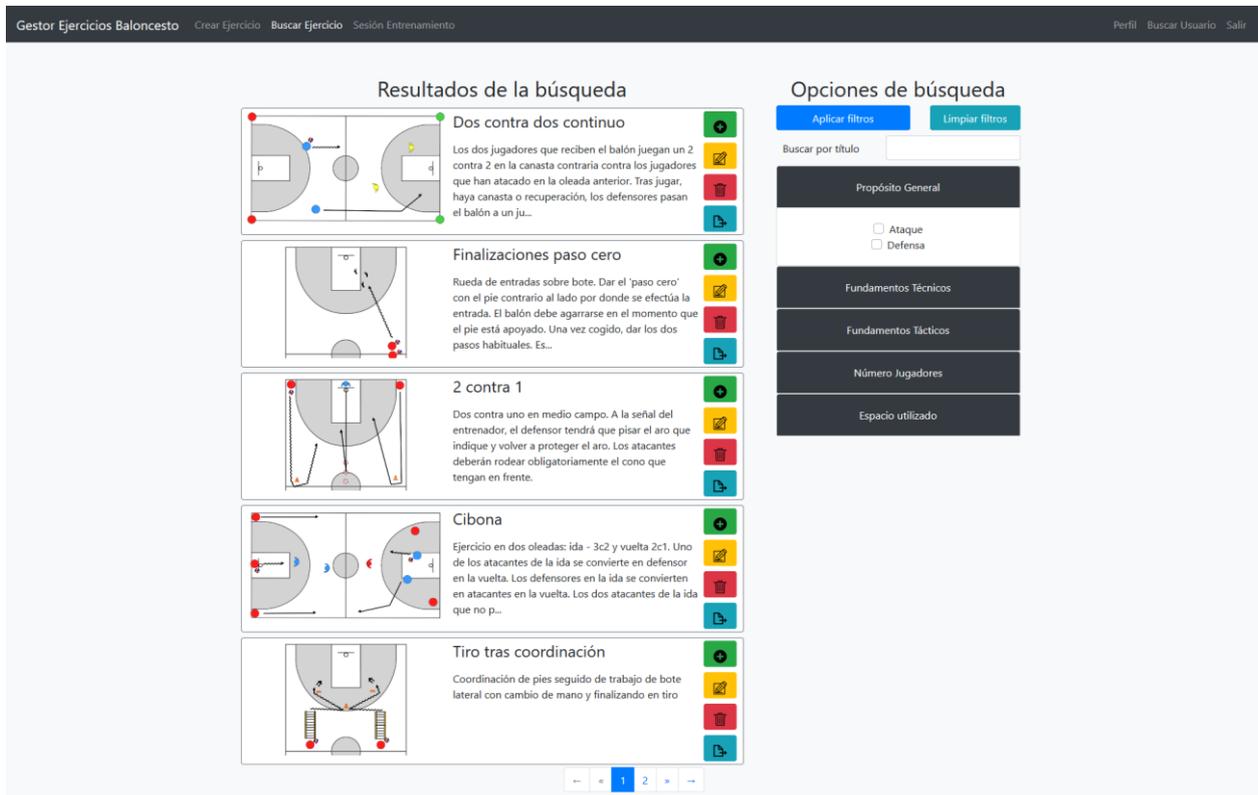


Figura 38

10.4.1 Trabajando con los ejercicios

Al hacer click sobre uno de los ejercicios, se amplía la descripción y la imagen del mismo. De esta manera podremos ver también el número mínimo y máximo de jugadores indicados y las etiquetas

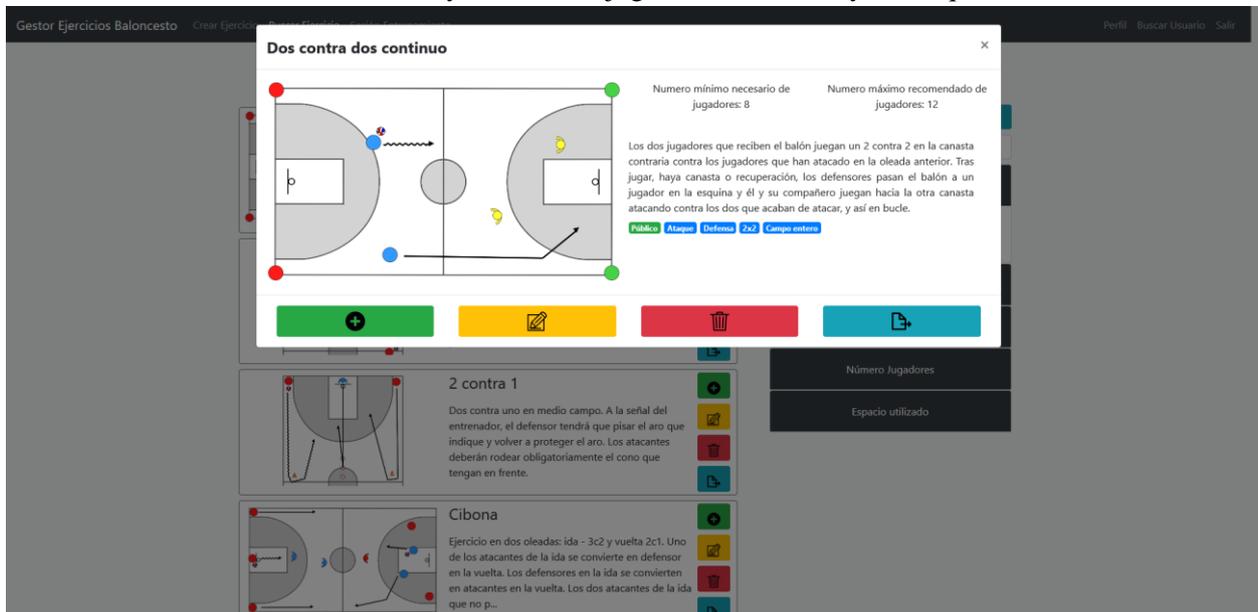


Figura 39

En cuanto a los botones:

- Botón verde: añadir el ejercicio a la sesión de entrenamiento
- Botón amarillo: editar el ejercicio
- Botón rojo: eliminar el ejercicio
- Botón azul: exportar el ejercicio a un documento externo.



Figura 40

10.4.2 Usando la herramienta de búsqueda

Los ejercicios tienen una herramienta de búsqueda que permite buscar ejercicios a través de

- Su título, o parte de su título
- Las etiquetas introducidas al crear el ejercicio
- El número de jugadores mínimo y máximo
- El espacio de campo utilizado para desarrollarse

Cabe destacar que cuantas más opciones introduzcamos, más concreta y precisa será la búsqueda. Si marcamos tres etiquetas, por ejemplo, obtendremos una lista de los ejercicios que contengan, como mínimo, esas tres etiquetas.

Por último, una vez seleccionadas las opciones de búsqueda, pulsar sobre “Aplicar filtros”.

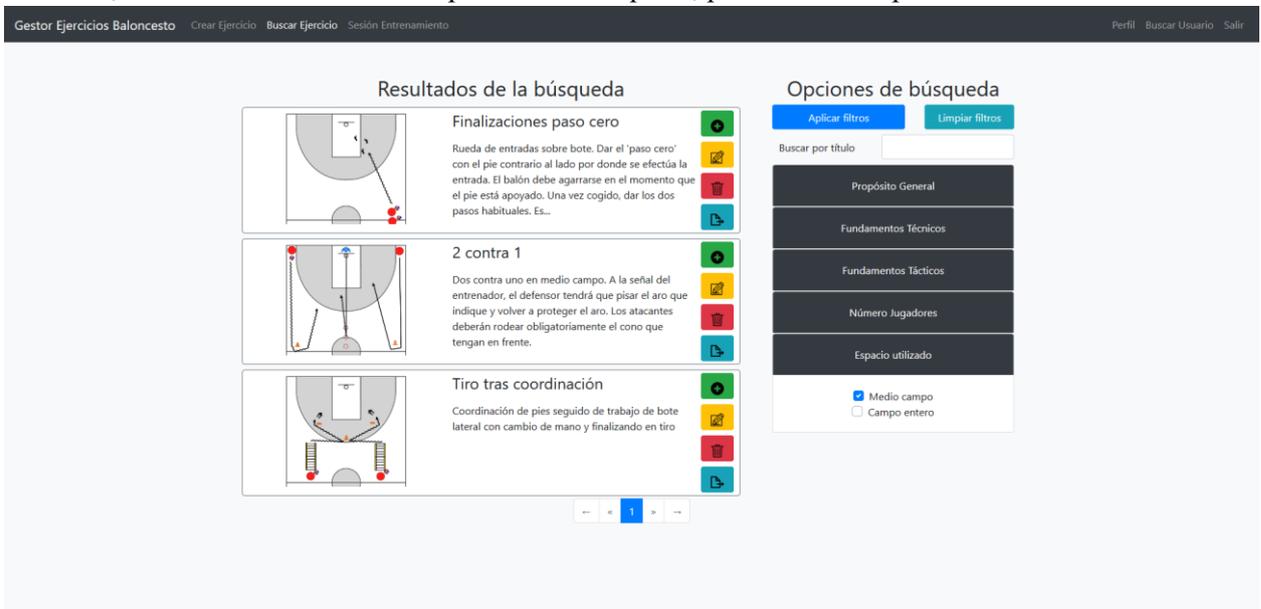


Figura 41

10.5 Uso de la sesión de entrenamiento

10.5.1 Añadir ejercicios a la sesión de entrenamiento

Desde la lista de nuestros ejercicios, una de las opciones que aparece en color verde es la de “Añadir a la sesión de entrenamiento”.

Al pulsar sobre esa opción, añadiremos el ejercicio a la sesión de entrenamiento que vayamos a realizar. Los ejercicios se añaden por orden de introducción, pudiendo ser modificado su orden después.

Para ver los ejercicios introducidos, pulsamos sobre la pestaña “Sesión entrenamiento” de la barra de navegación.

10.5.2 Definir el orden, la duración y las observaciones de los ejercicios

Una vez que hemos introducido ejercicios en la sesión de entrenamiento debemos establecer al menos la duración de los mismos. No puede quedar ningún ejercicio con una duración de cero minutos.

Es recomendable añadir alguna observación a mayores de la explicación que contiene la descripción del ejercicio, aunque no es obligatorio.

Para modificar el orden de los ejercicios es tan sencillo como arrastrar el ejercicio a la posición que queramos que ocupe.

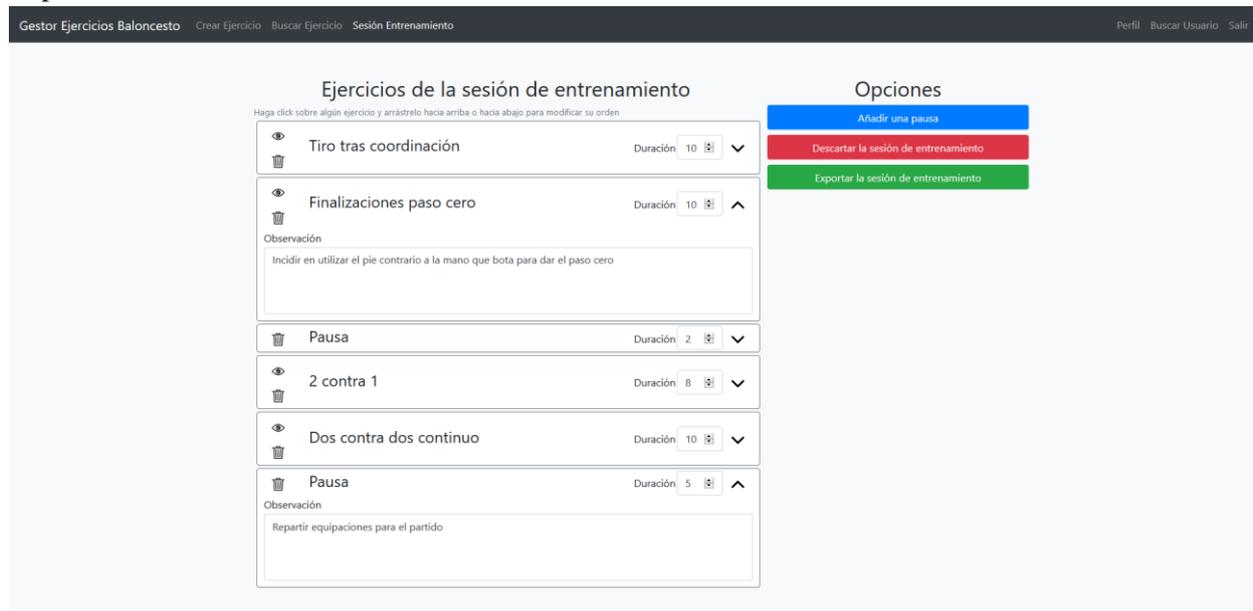


Figura 42

10.5.3 Añadir pausa

Las pausas son actividades con una determinada duración que no hacen referencia a ningún ejercicio de los que hayamos podido crear o tomar prestado. Son útiles para marcar, por ejemplo, los descansos para beber agua u otras actividades como la del ejemplo.

Al igual que los ejercicios deben tener una duración superior a cero.

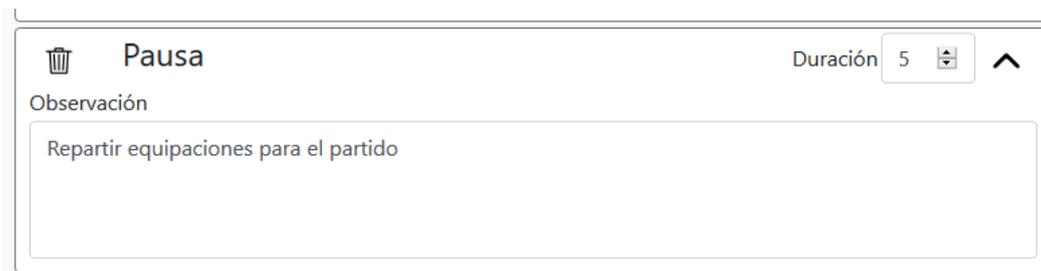


Figura 43

Para añadir una pausa debe pulsarse sobre el botón “Añadir pausa” de la derecha. Esta será introducida al final de la lista de ejercicios y se podrá arrastrar para cambiar su posición.

10.5.4 Exportar la sesión de entrenamiento

Una vez ordenados y especificadas las duraciones de los ejercicios exportamos la sesión de entrenamiento a un documento externo en formato PDF.

Se debe introducir el nombre del equipo al que pertenece la sesión que vamos a exportar. Es conveniente introducir también cuál es el objetivo concreto del entrenamiento.

Por último, introducir la fecha y la hora del entrenamiento. Si no se introduce, el sistema tomará por defecto la fecha y hora actual.

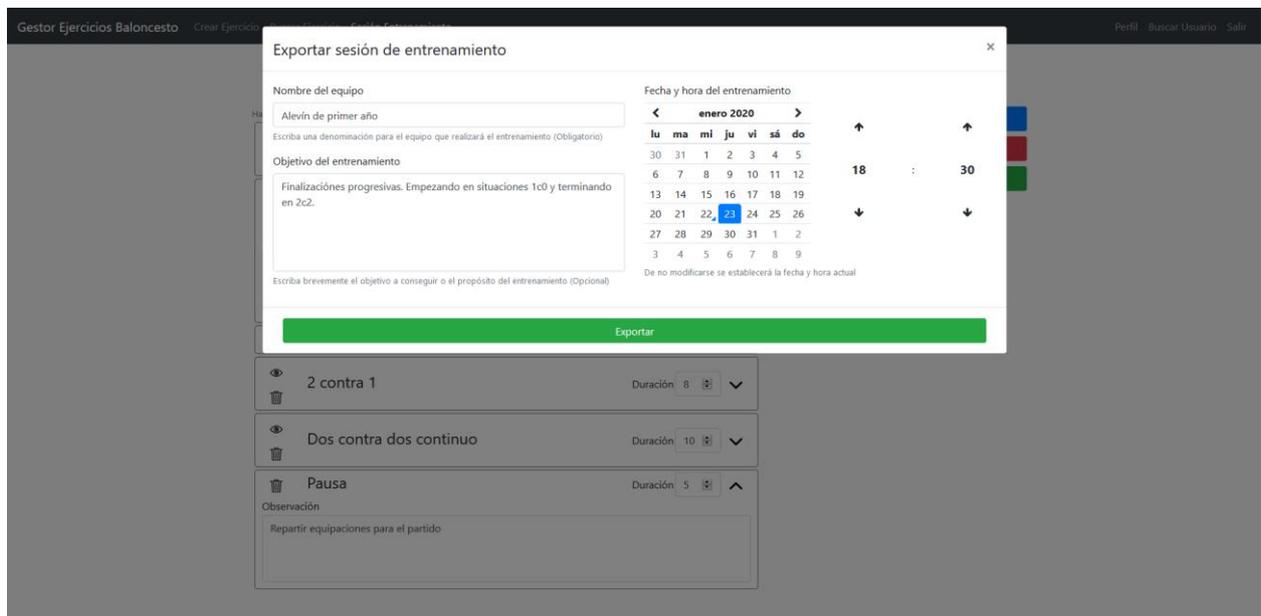


Figura 44

10.5.5 Resultado

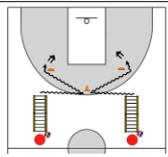
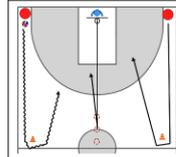
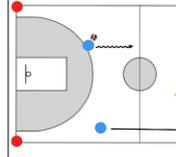
ALEVÍN DE PRIMER AÑO		2020-01-23	
Objetivo del entrenamiento: Finalizaciones progresivas. Empezando en situaciones 1c0 y terminando en 2c2.		18:30	
Tiro tras coordinación	10'	18:30	18:40
	Descripción: Coordinación de pies seguido de trabajo de bote lateral con cambio de mano y finalizando en tiro	Autor: nbees	
Finalizaciones paso cero	10'	18:40	18:50
	Descripción: Rueda de entradas sobre bote. Dar el 'paso cero' con el pie contrario al lado por donde se efectúa la entrada. El balón debe agarrarse en el momento que el pie está apoyado. Una vez cogido, dar los dos pasos habituales. Es importante que el paso cero se dé de esta manera para finalizar la entrada con equilibrio	Autor: nbees	
Pausa	2'	18:50	18:52
Agua			
2 contra 1	8'	18:52	19:00
	Descripción: Dos contra uno en medio campo. A la señal del entrenador, el defensor tendrá que pisar el aro que indique y volver a proteger el aro. Los atacantes deberán rodear obligatoriamente el cono que tengan en frente.	Autor: nbees	
Dos contra dos continuo	10'	19:00	19:10
	Descripción: Los dos jugadores que reciben el balón juegan un 2 contra 2 en la canasta contraria contra los jugadores que han atacado en la oleada anterior. Tras jugar, haya canasta o recuperación, los defensores pasan el balón a un jugador en la esquina y él y su compañero juegan hacia la otra canasta atacando contra los dos que acaban de atacar, y así en bucle.	Autor: nbees	
Pausa	5'	19:10	19:15
Repartir equipaciones para el partido			

Figura 45

10.6 Otros usuarios

10.6.1 Búsqueda de usuarios

La aplicación cuenta con un servicio de búsqueda de usuarios. Para acceder al mismo, solamente hay que pulsar sobre la pestaña "Buscar usuario" de la barra de navegación.

Una vez que se muestra la página, podemos buscar usuarios de tres maneras distintas:

- A través de su nombre de usuario o de su alias
- A través de su nivel de entrenador
- A través de su club

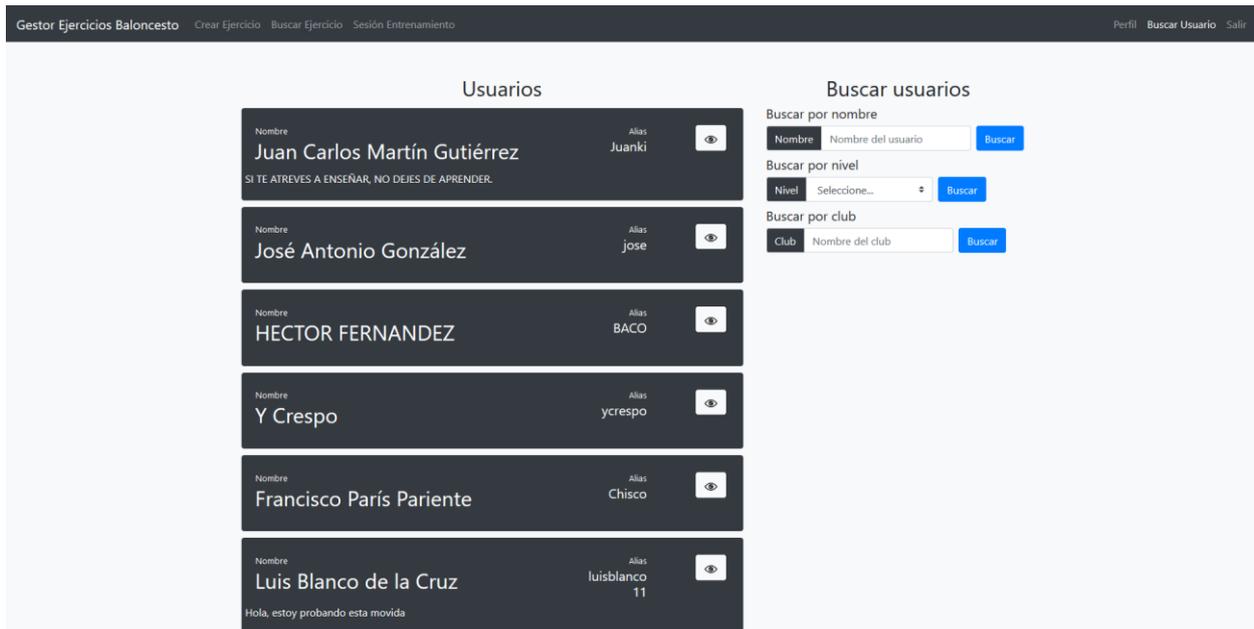


Figura 46

Si introducimos, por ejemplo, una “r” en el campo de búsqueda por nombre obtenemos el siguiente resultado:

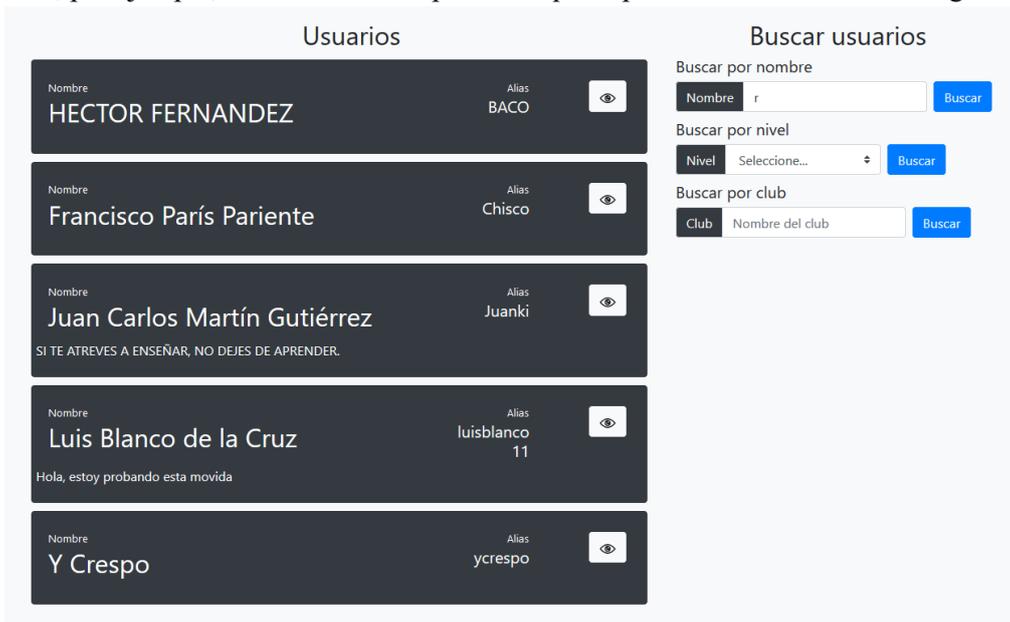


Figura 47

El resultado de la búsqueda muestra todos los usuarios que tienen al menos una “r” en su nombre o en su alias. Pulsando sobre el botón de cada usuario accederemos a ver su perfil.

10.6.2 Perfil de otro usuario

Una vez que accedemos al perfil de otro usuario podremos interactuar con él. Debajo de su alias y su información personal se encuentra el botón “Seguir”.

Al hacer click sobre el botón se enviará una solicitud al usuario, que si es aceptada nos añadirá a la lista de seguidores de ese usuario, permitiéndonos ver sus ejercicios protegidos.

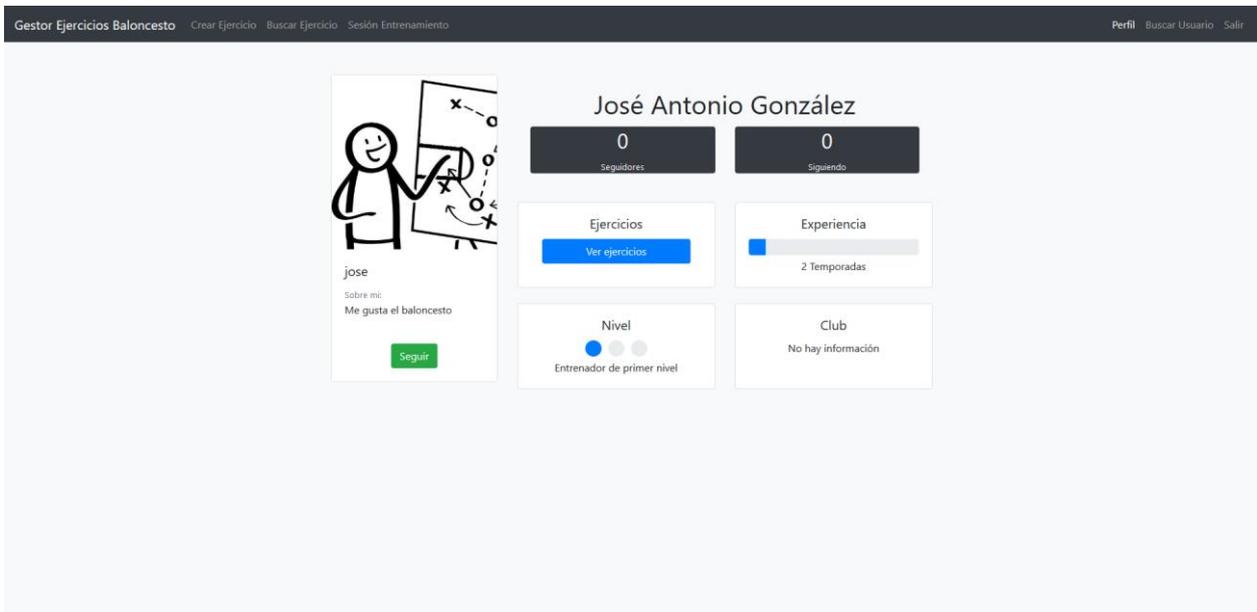


Figura 48

Una vez que el botón es pulsado y la solicitud enviada, quedamos a la espera de ser aceptados o rechazados por el usuario. En este momento, si la solicitud no ha sido resuelta, podremos cancelarla.



Figura 49

Si el usuario acepta la solicitud, el perfil del usuario quedará de la siguiente manera, pudiendo dejar de seguirle cuando se desee:



Figura 50

Si el usuario, por el contrario, rechaza la solicitud volveremos al estado inicial y podremos enviar otra solicitud.

10.6.3 Ejercicios de otro usuario

Una vez que se accede al perfil de otro usuario, pulsando sobre el botón “Ver ejercicios” podremos acceder a los ejercicios que haya creado. Los usuarios elegimos la visibilidad que queremos dar a los ejercicios. Podremos ver los ejercicios que el usuario haya marcado como públicos o, en caso de que seamos seguidores, los que haya marcado como protegidos.

En esta ocasión también podremos utilizar la herramienta de búsqueda para filtrar entre los ejercicios del otro usuario.

Cuando estemos visualizando ejercicios que no sean creados por nosotros aparecerá el autor debajo del título. Con ellos podremos:

- **Tomar como prestado** un ejercicio: El ejercicio aparecerá en nuestra lista, como si lo hubiéramos creado nosotros. Se consigue pulsando el botón verde.
- **Exportar ejercicio**: podremos exportar el PDF de un ejercicio en concreto, pero previamente habremos tenido que tomarlo como prestado. Se consigue al pulsar el botón azul.

Hay que tener en cuenta que cuando tomamos un ejercicio prestado no deja de ser un préstamo. El autor decide cuando deja de dártelo. Si un usuario decide cambiar la visibilidad de público a protegido, si no somos seguidores del usuario dejaremos de verlo y perderemos el acceso.

Sucede algo similar cuando un usuario elimina a otro de su lista de seguidores, o bien un usuario deja de seguir por voluntad propia a otro. Si tenemos tomado prestado un ejercicio de un usuario del que dejamos de ser seguidor, perdemos el acceso a dicho ejercicio.

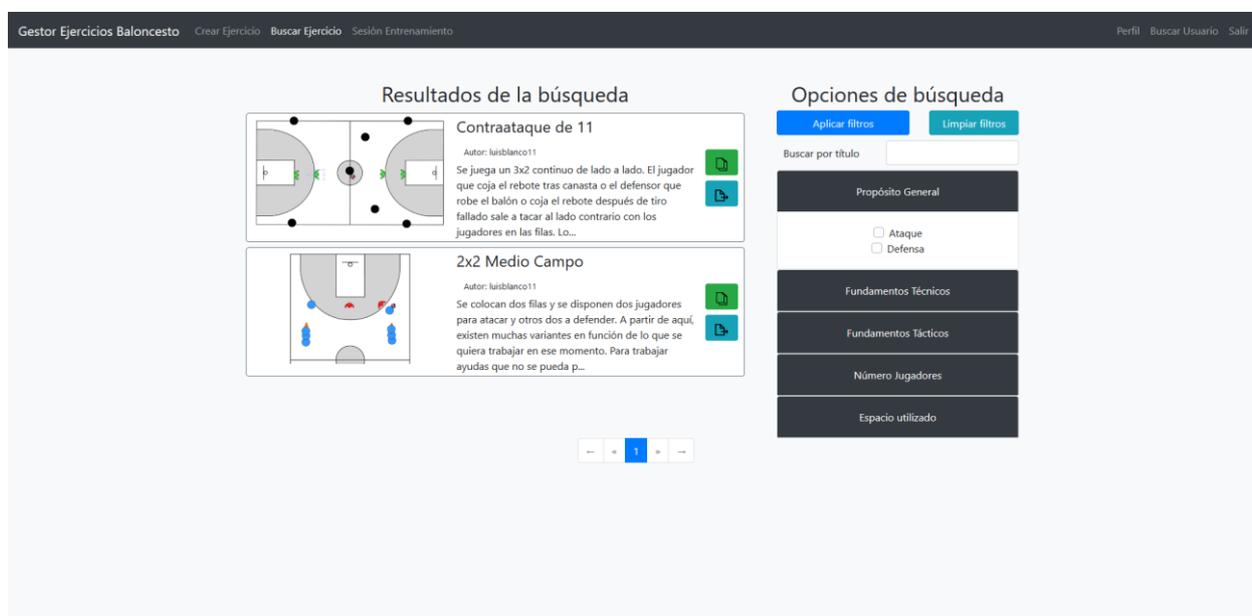


Figura 51

10.7 Gestión de los usuarios seguidores y de los usuarios seguidos

10.7.1 Gestión de solicitudes

Al acceder al perfil del usuario podemos ver si existe algún usuario que quiera convertirse en seguidor nuestro.

Al pulsar sobre el apartado “Solicitudes pendientes” accederemos a la página de gestión de solicitudes.

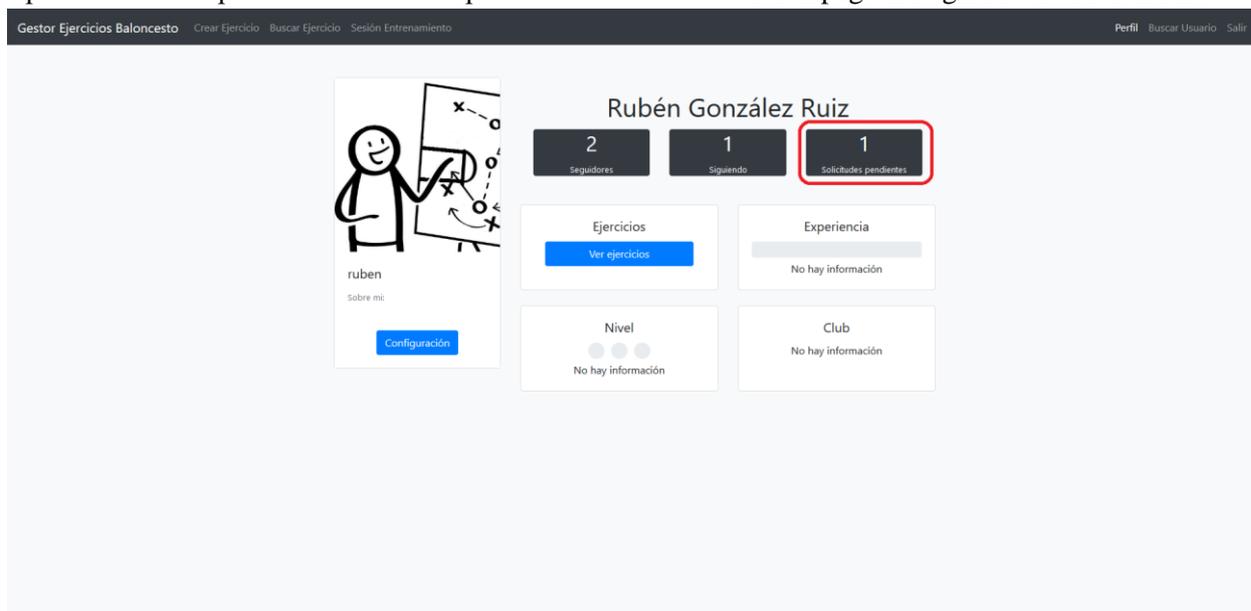


Figura 52

Una vez dentro, accederemos por defecto al apartado de las solicitudes recibidas. Nos mostrará qué usuarios desean seguirnos y nos permitirá rechazar o aceptar la solicitud.

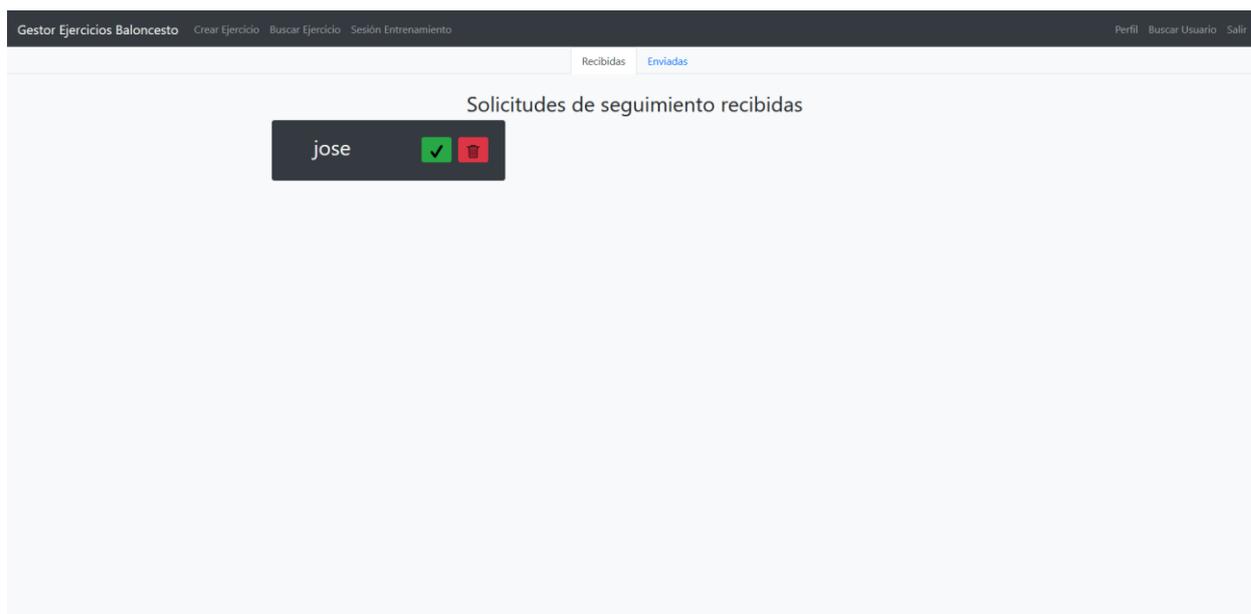


Figura 53

Si cambiamos de pestaña, podremos ver las solicitudes que hemos enviado y aún no han sido contestadas. Desde ahí podremos cancelar la petición si así lo deseamos.

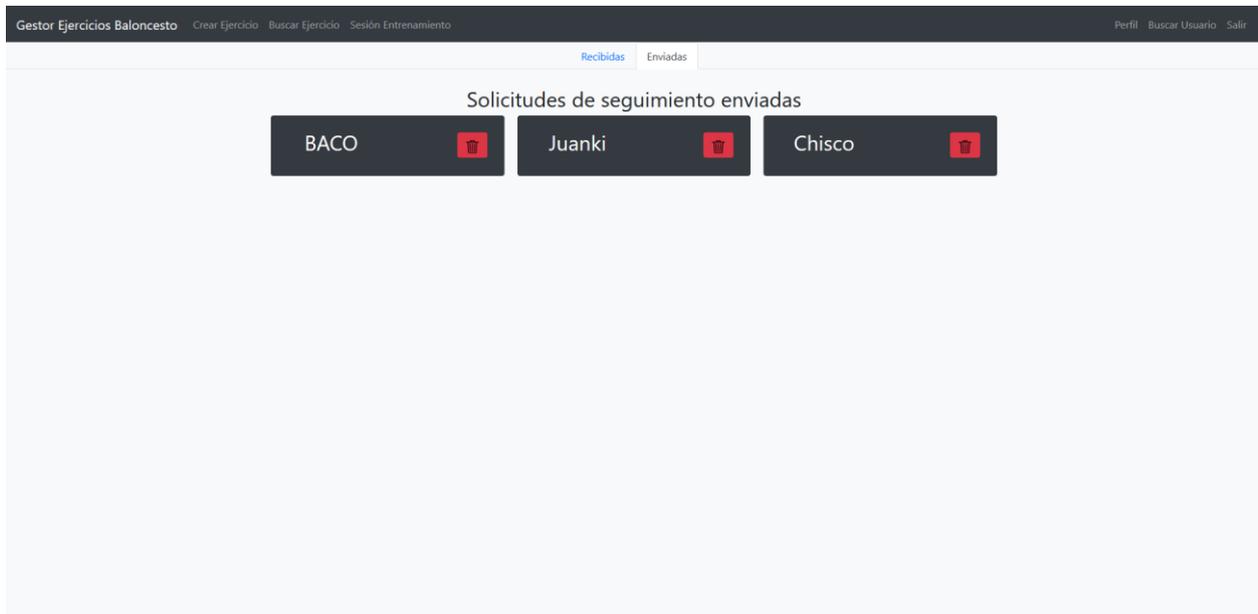


Figura 54

10.7.2 Gestión de listas de seguidores/seguidos

Una vez que hemos enviado y recibido solicitudes, y estas han sido aceptadas, empezaremos a ver cómo nuestros seguidores aumentan, y también aumentarán el número de personas a las que seguimos.

Pulsando sobre el apartado “Seguidores” y “Siguiendo” del perfil accederemos a las listas de usuarios correspondientes.

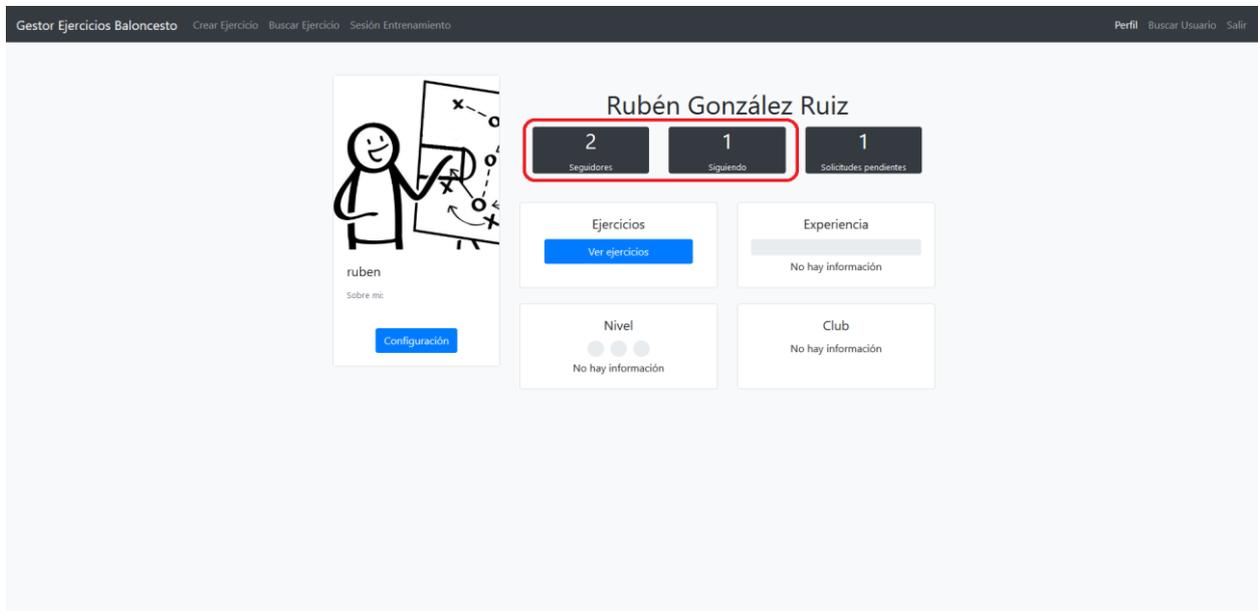


Figura 55

Desde la lista de usuarios a los que seguimos podremos acceder al perfil de los mismos o dejar de seguirlos.

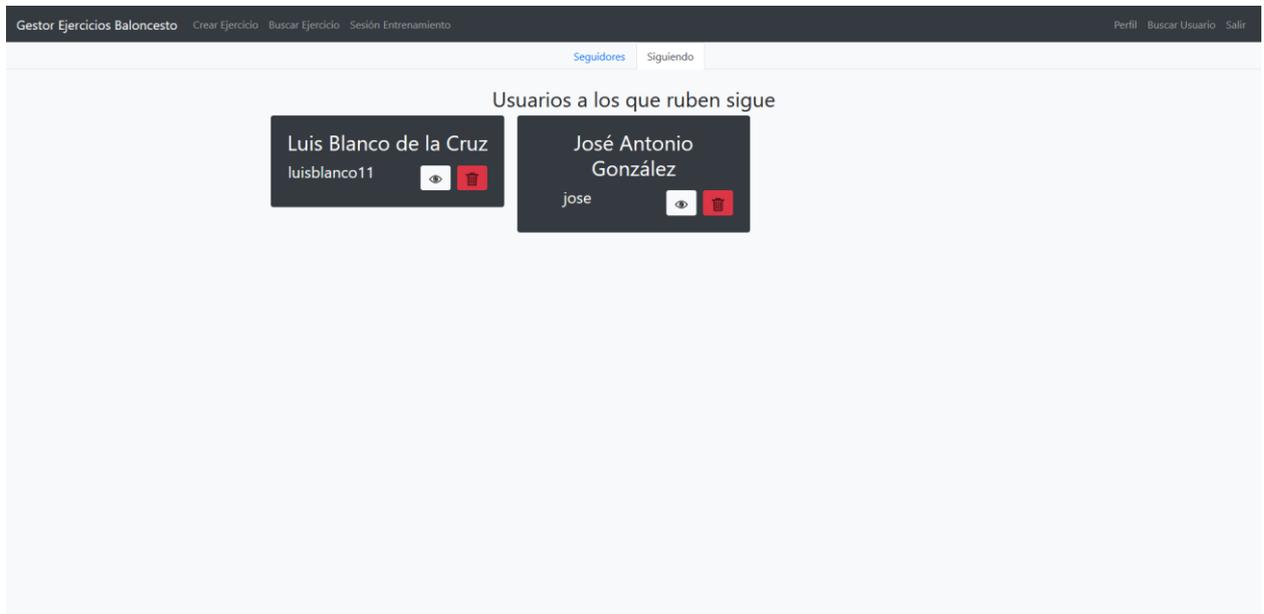


Figura 56

De forma análoga, desde la lista de personas que nos siguen podremos acceder a su perfil o hacer que dejen de seguirnos.

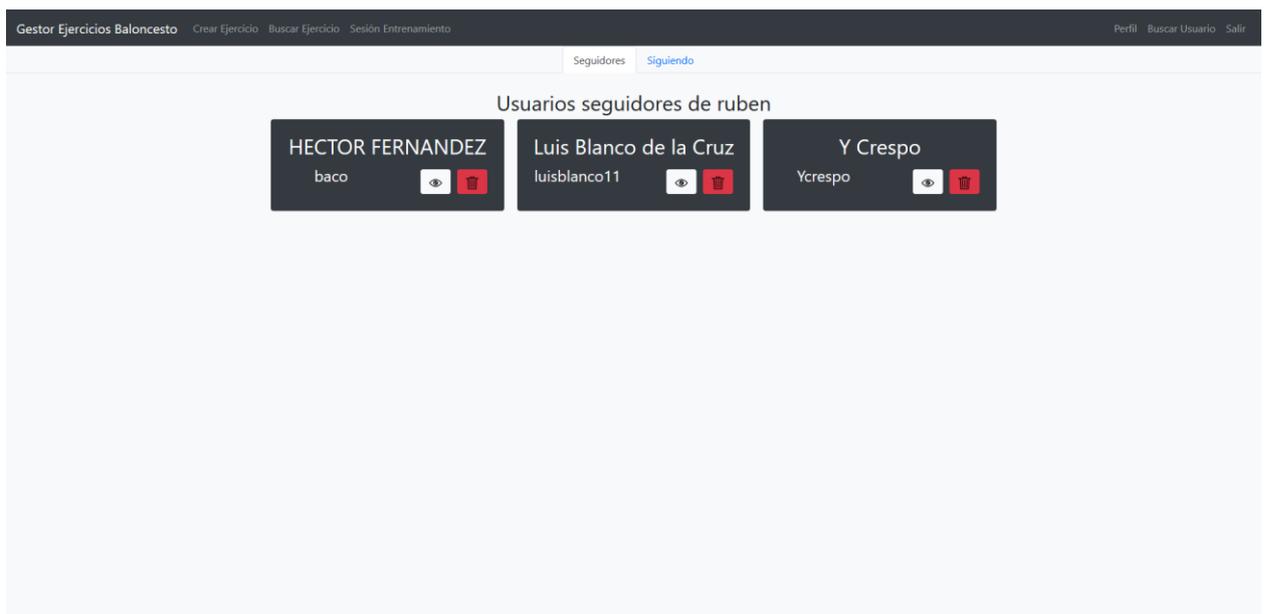


Figura 57

Capítulo 11. Manual de Mantenimiento

Guía de cómo aplicar cambios en el contenido y cómo de añadir nueva funcionalidad

11.1 Modificación del contenido

Es posible que se quiera modificar la funcionalidad que tiene actualmente la aplicación.

11.1.1 Modificación del frontend

Como se aprecia en el apartado de la organización del código, para cada una de las vistas existe un archivo JavaScript que contiene las llamadas pertinentes para obtener el contenido y las funciones para mostrarlo.

Si se requiriera modificar la estructura de la vista bastaría con modificar el archivo HTML de la vista, si se quiere cambiar alguna llamada o la lógica para mostrar los datos sólo habría que modificar el archivo JavaScript.

11.1.2 Modificación del backend

En el apartado de organización del código se recoge cómo están organizados los paquetes, agrupando controladores, servicios, entidades y repositorios. Probablemente, los cambios más comunes habrá que realizarlos en el paquete de los servicios ya que los cambios que afectan a la funcionalidad deben hacerse ahí.

Si se necesitase hacer cambios en los atributos de las entidades habría que modificar la clase que fuera dentro de ese paquete y si se necesitase realizar nuevas consultas con la base de datos habría que incluir cambios en el paquete de persistencia.

11.2 Creación de contenido

Es posible que se quiera añadir nueva funcionalidad a la que tiene actualmente la aplicación.

11.2.1 Modificación del frontend

Al igual que para la modificación de contenido, si quisiéramos realizar una nueva vista deberíamos seguir la misma idea y crear un nuevo archivo HTML y un nuevo archivo JavaScript donde añadir la estructura de la vista y su funcionalidad, respectivamente.

Es importante recordar que se debe siempre importar el archivo “util.js” que es el que contiene la variable con el endpoint común deben utilizar todas las llamadas.

11.2.2 Modificación del backend

Para la creación de una nueva funcionalidad:

O bien, consistirá en añadir una función a un controlador existente: lo que supone nuevas funciones en los servicios a los que llame, que quizás tengan necesitan de nuevos métodos en sus repositorios.

O bien, habrá que crear un nuevo controlador que probablemente requiera de su propio servicio donde implementar la lógica y que en el caso de que incluya nuevas entidades es posible que se tengan que crear nuevos archivos en el repositorio también. Todas estas nuevas clases deberán almacenadas ir en su paquete correspondiente.

Capítulo 12. Manual de despliegue de la aplicación

Guía para desplegar la aplicación tanto en local como en producción.

12.1 Despliegue en local

Se entiende despliegue en local como el despliegue de la aplicación en el ordenador donde se realiza el desarrollo del proyecto.

Para todo el desarrollo se ha utilizado un ordenador con sistema operativo Ubuntu 18.01 - Linux. Aunque es equivalente si se realiza desde un sistema operativo Windows.

12.1.1 Herramientas necesarias

A continuación, se recoge una lista de las herramientas necesarias para desplegar el proyecto:

- Java Development Kit - JDK (versión 8)
- Spring Boot Tools Suite (versión 4.1.1)
- Lombok (1.16.16)
- MariaDB Server
- DataGrip

12.1.2 Configuración del proyecto

12.1.2.1 Instalaciones iniciales

Lombok es un plugin que debe instalarse sobre el IDE y añadirse además como dependencia en el POM del proyecto. También debe especificarse en el POM la versión de Java utilizada.

Se debe instalar el servidor MariaDB y crear manualmente en él una base de datos donde albergar la información persistente del proyecto. Para este proyecto el nombre de la base de datos local será “gestorDatabase” y como usuario se utilizará el usuario “root”.

12.1.2.2 Configuración del perfil de desarrollo

En el archivo “~/gestor_ejercicios_tfg/src/main/resources/application.properties” se debe indicar que el perfil activo para desplegar el proyecto debe ser el de “desarrollo”:

```
spring.profiles.active=desarrollo
```

12.1.2.3 Configuración para la base de datos

Para el perfil de desarrollo se debe especificar a Spring Boot en el archivo “~/gestor_ejercicios_tfg/src/main/resources/application-desarrollo.properties” las variables para la base de datos. Para la conexión en local:

```
database.url=jdbc:mariadb://localhost:3306/gestorDatabase
database.username=root
database.password=cssisawesom3!
```

La clase java **GestorEjerciciosConfiguration** se encarga de los aspectos de configuración del proyecto, entre ellos de la conexión con la base de datos. Spring leerá las tres variables anteriores y las utilizará para realizar la conexión.

Al arrancar Spring se encargará de conectar con la base de datos utilizando la dirección, el usuario y la contraseña. También lee las entidades especificadas en las clases java y crea automáticamente las tablas correspondientes en la base de datos.

12.1.3 Despliegue de la aplicación

12.1.3.1 Arranque de la aplicación

Con el IDE STS abierto y el proyecto importado, buscar el apartado “Boot dashboard”, seleccionar el proyecto y pulsar el botón de arranque (el primero con el cuadro rojo)

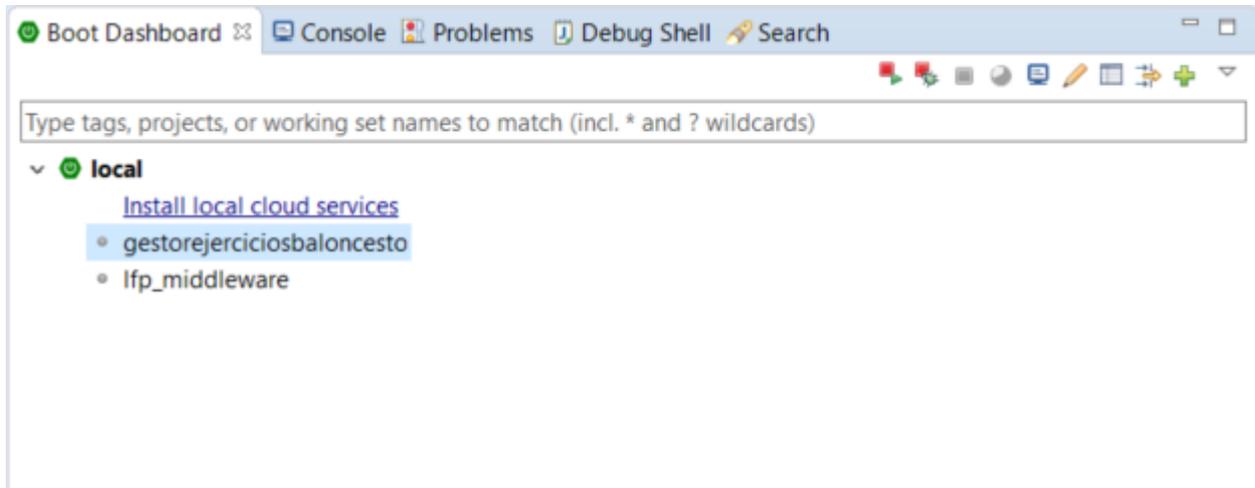


Figura 58

12.1.3.2 Poblar la base de datos

Una vez que se han creado las tablas, utilizar el gestor de bases de datos DataGrip (donde se debe configurar la conexión utilizando los mismos valores que los definidos en el archivo de propiedades) para cargar un script con valores iniciales necesarios para los enumerados.

12.1.3.3 Acceso a la aplicación

Accederemos a la aplicación a través de introduciendo “localhost:8080” en el navegador que elijamos. La aplicación ha sido probada con los navegadores Chrome y Firefox.

12.2 Despliegue en producción

Se entiende despliegue en producción al despliegue realizado sobre un servidor online gratuito, en este caso Heroku.

12.2.1 Herramientas y servicios necesarios

A continuación, se detallan los servicios necesarios para desplegar el proyecto:

- Heroku (Aplicación de servicios en la nube)
- JawsDB (Servicio online de bases de datos)
- DataGrip
- Git
- Heroku CLI (Consola de Heroku que necesita de Git)

12.2.1.1 Crear aplicación en Heroku

En primer lugar, es imprescindible registrarse y crear una cuenta en Heroku. Una vez realizado hay que acceder y crear un nuevo proyecto:

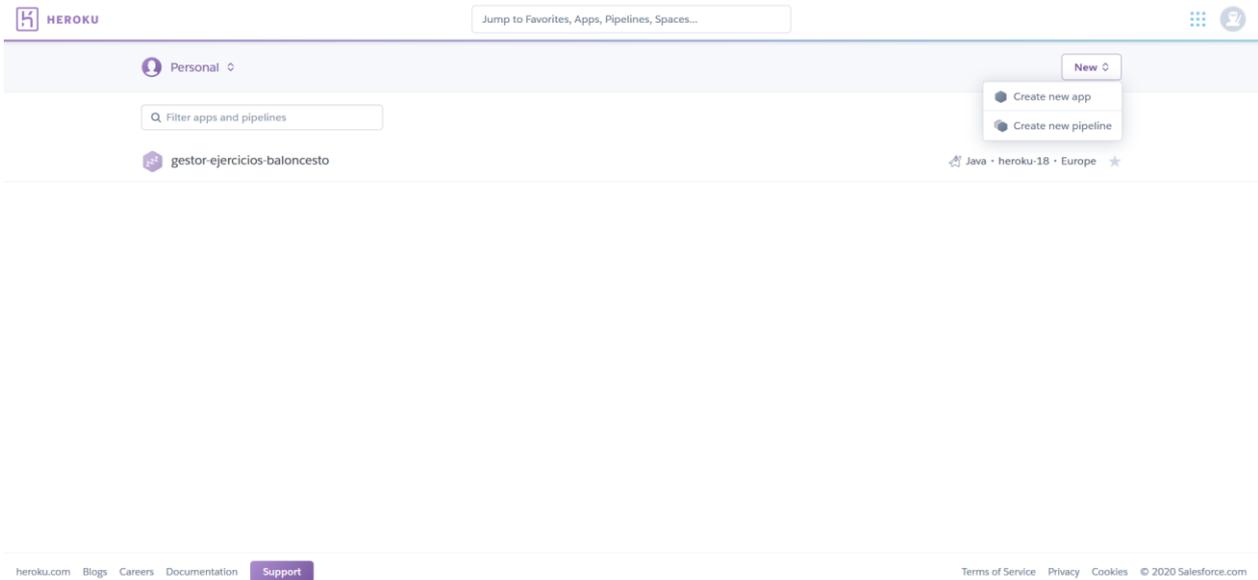


Figura 59

Después se elige el nombre del proyecto y se crea la aplicación:

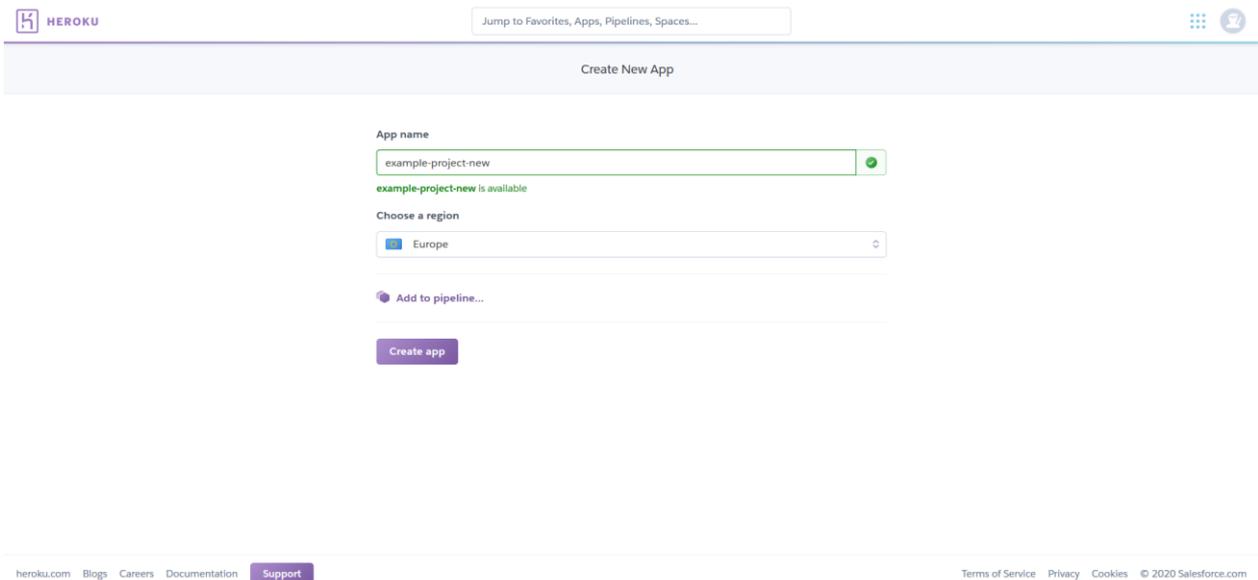


Figura 60

Con estos pasos ya se ha conseguido crear una nueva aplicación:

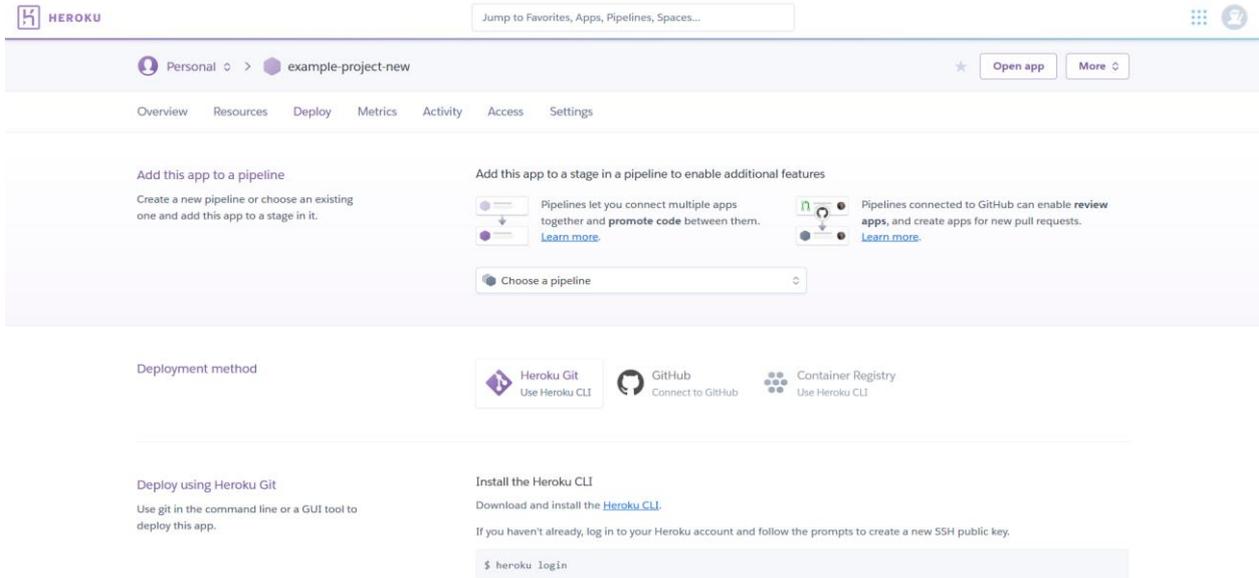


Figura 61

Si se pulsa sobre el botón “Open app” veremos el contenido por defecto de la aplicación:



Figura 62

12.2.1.2 Añadir el plugin de JawsDB

En el apartado Resources de la aplicación hay que añadir un add-on de JawsDB para crear una base de datos. Seleccionar el de MariaDB.

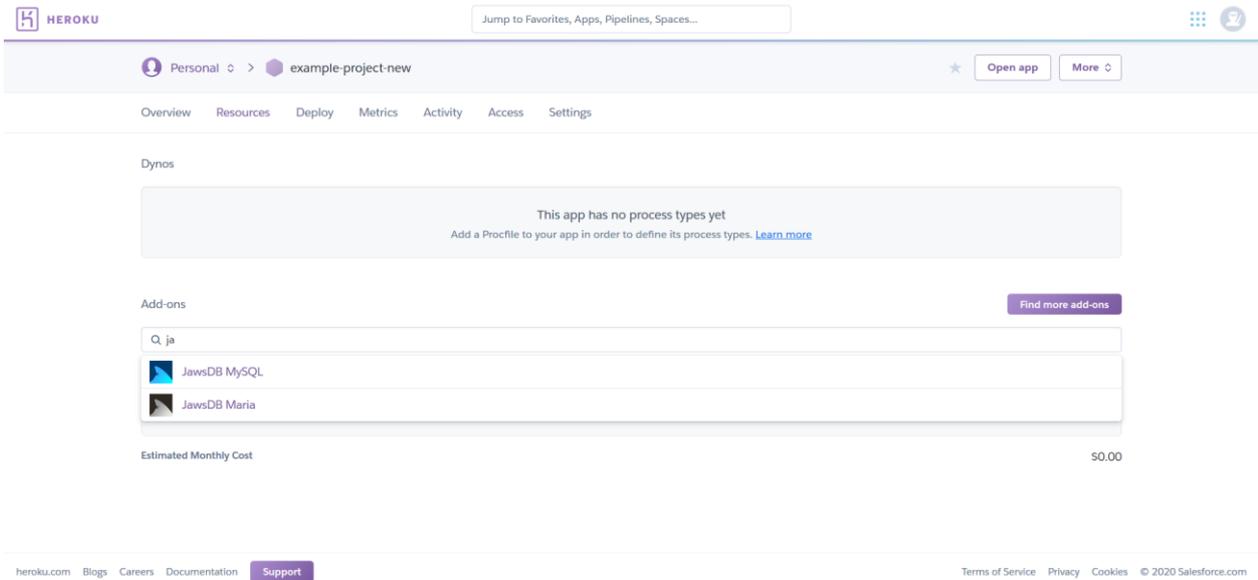


Figura 63

Como versión de JawsDB, seleccionar la versión Kitefin gratuita.

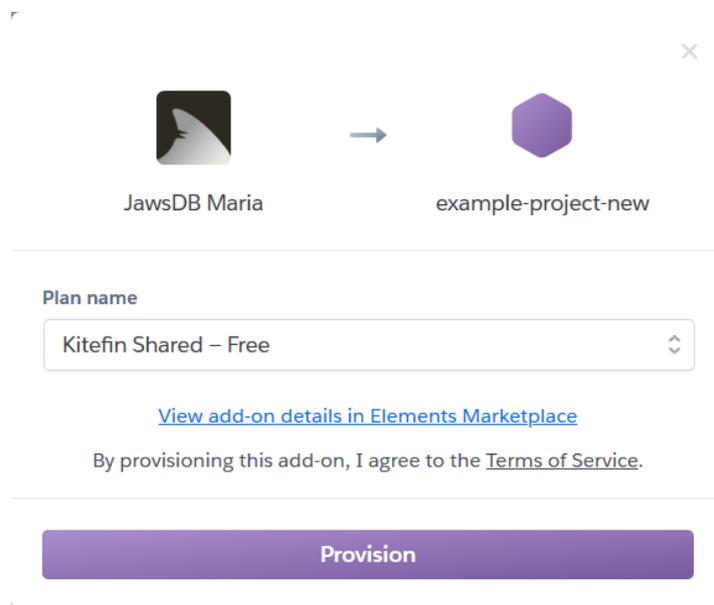


Figura 64

Al pulsar sobre “Provision” quedará añadido el plugin a la aplicación.

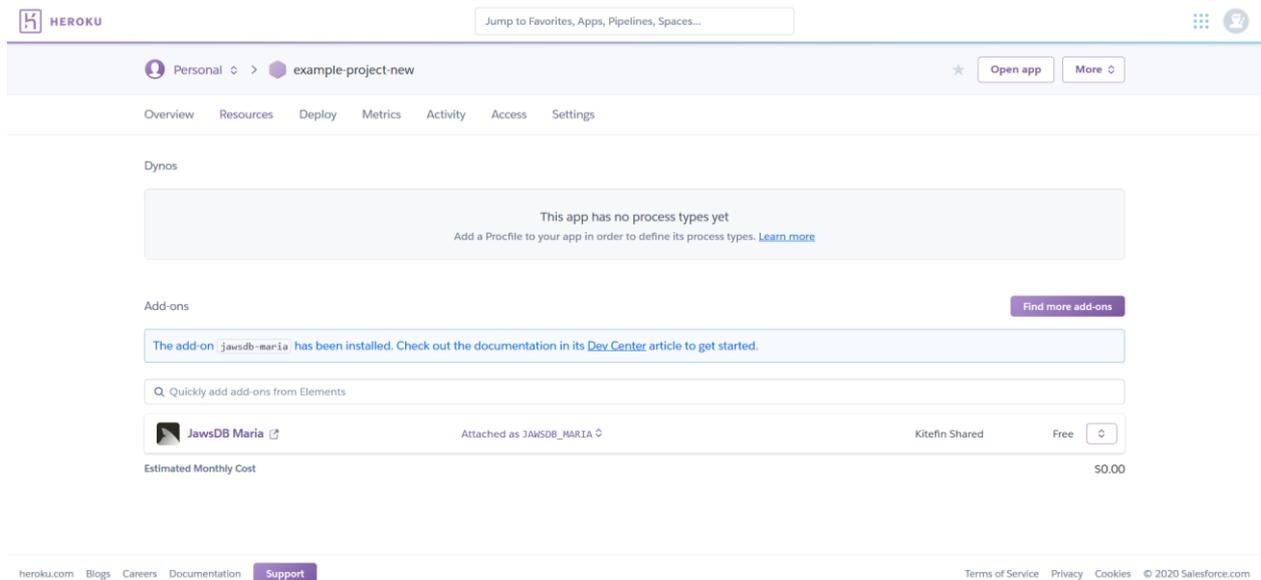


Figura 65

Para ver los detalles de la conexión de la base de datos que se acaba de agregar, hay que pulsar sobre “JawsDB Maria”. Se abre una pestaña nueva con la siguiente información:

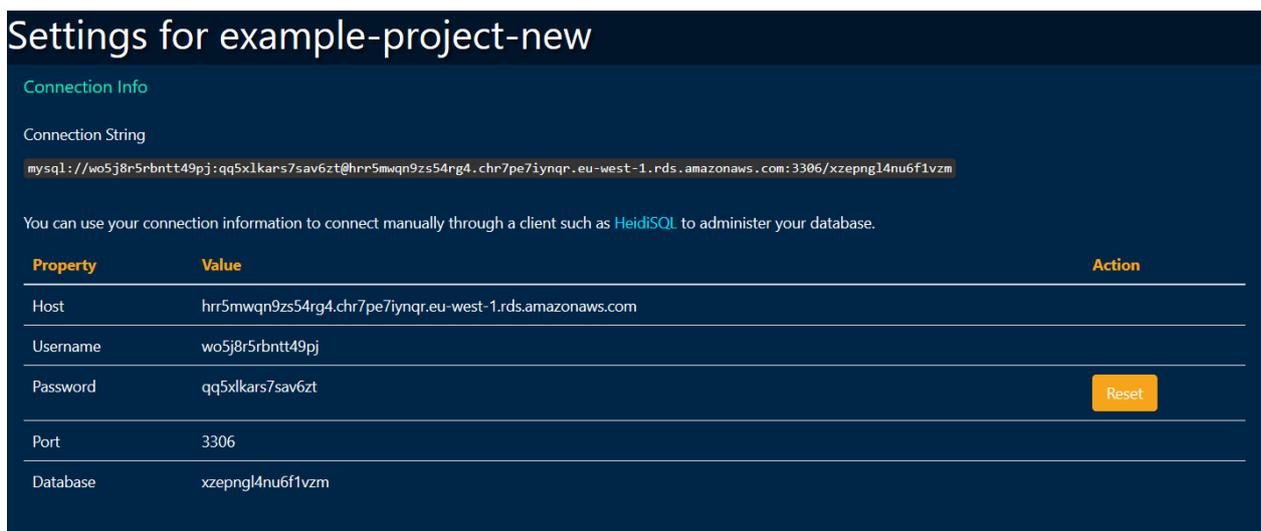


Figura 66

Con estos datos, conectar con la base de datos a través de DataGrip.

12.2.2 Configuración del proyecto

12.2.2.1 Configuración del perfil de producción

En el archivo “~/gestor_ejercicios_tfg/src/main/resources/application.properties” se debe indicar que el perfil activo para desplegar el proyecto debe ser el de “producción”:

```
spring.profiles.active=produccion
```

12.2.2.2 Configuración para la base de datos

En el archivo “~/gestor_ejercicios_tfg/src/main/resources/application-desarrollo.properties” hay que indicar que las variables de configuración de la base de datos alberguen la información referente a la base de datos que proporciona JawsDB. En este ejemplo:

```
database.url=jdbc:mariadb://hrr5mwqn9zs54rg4.chr7pe7iynqr.eu-west-1.rds.amazonaws.com:3306/xzepngl4nu6flvzm
database.username=wo5j8r5rbntt49pj
database.password=qq5xlkars7sav6zt
```

Nota: darse cuenta que la url sigue el siguiente formato:

```
jdbc:mariadb://<dirección host>:<puerto>/<base de datos>
```

12.2.2.3 Configuración para las llamadas del frontend

Todas las vistas del frontend tienen acceso a un archivo JavaScript común, denominado `utils.js`, donde existe una variable denominada “`backEndpoint`”.

A la hora de desplegar hay que hacer que esta variable indique la dirección donde va a estar desplegada la aplicación.

```
//var backEndpoint = "http://localhost:8080/";
var backEndpoint = "https://gestor-ejercicios-baloncesto.herokuapp.com/";
```

Los métodos que efectúan las llamadas contra el api están preparados para añadir el resto del path necesario en la petición. Tan solo hay que indicar cuál es el endpoint correspondiente comentando el endpoint que queremos evitar y descomentando el endpoint al que se desea acceder.

12.2.3 Despliegue del proyecto

Para desplegar el proyecto que se ha desarrollado en local sobre la aplicación que se ha creado en Heroku se debe acceder al directorio donde está el proyecto con los cambios explicados en el apartado anterior.

12.2.3.1 Iniciar repositorio Git.

Utilizando la terminal, acceder al cliente de Heroku (Heroku CLI) y conectar el repositorio remoto de Heroku con el repositorio local. Solo hay que indicar el nombre de la aplicación:

```
heroku git:remote -a example-project-new
```

12.2.3.2 Desplegar la aplicación en el servidor

Desplegar la aplicación haciendo un push sobre la rama master del repositorio de Heroku:

```
git push heroku master
```

Heroku dispone de todas las herramientas necesarias para desplegar la aplicación y lo hará de manera transparente al usuario, informando de los logs en la terminal. Heroku está preparado automáticamente para que al hacer un push sobre la rama máster despliegue el proyecto.

12.2.3.3 Poblar la base de datos

Poblar los enumerados almacenados en la base de datos con el script de poblado utilizando DataGrip.

Bibliografía

- [1] P. N. Robillard, P. Kruchten, and P. D'Astous, *YOOPEEDOO (UPEDU): A process for teaching software process*. 2001.
- [2] "Pricing | Heroku." [Online]. Available: <https://www.heroku.com/pricing>. [Accessed: 02-Feb-2020].
- [3] "JawsDB Database-as-a-Service." [Online]. Available: <https://www.jawsdb.com/#start>. [Accessed: 02-Feb-2020].
- [4] "Astah Online Store." [Online]. Available: <https://sites.fastspring.com/astah/product/online-store>. [Accessed: 02-Feb-2020].
- [5] "Balsamiq Store | Balsamiq." [Online]. Available: <https://balsamiq.com/buy/#cloud>. [Accessed: 02-Feb-2020].
- [6] "Buy DataGrip: Pricing and Licensing, Discounts - JetBrains Toolbox Subscription." [Online]. Available: <https://www.jetbrains.com/datagrip/buy/#commercial?billing=yearly>. [Accessed: 02-Feb-2020].
- [7] "GitLab Pricing | GitLab." [Online]. Available: <https://about.gitlab.com/pricing/>. [Accessed: 02-Feb-2020].
- [8] "Salarios para empleos de Jefe de proyecto en España | Indeed.es." [Online]. Available: <https://www.indeed.es/salaries/jefe-de-proyecto-Salaries>. [Accessed: 02-Feb-2020].
- [9] "Salarios para empleos de Desarrollador/a de software en España | Indeed.es." [Online]. Available: <https://www.indeed.es/salaries/desarrollador-de-software-Salaries>. [Accessed: 02-Feb-2020].
- [10] "A Quick Guide to GPLv3 - GNU Project - Free Software Foundation." [Online]. Available: <http://www.gnu.org/licenses/quick-guide-gplv3.html>. [Accessed: 03-Feb-2020].
- [11] D. Cristina and G. Escribano, "Escuela de Ingeniería Informática Desarrollo de aplicación para la generación de juegos geográficos a medida." [Online]. Available: <http://uvadoc.uva.es/handle/10324/38747>. [Accessed: 02-Feb-2020].
- [12] C. Larman, *UML y patrones : una introducción al análisis y diseño orientado a objetos y al proceso unificado*, 2nd ed. Prentice Hall, 2002.
- [13] M. Fowler, *Patterns of Enterprise Application Architecture*, 1st ed. Addison Wesley, 2002.
- [14] V. Gamma, Helm, Johnson, *Patrones de Diseño*. Addison Wesley.

Anexos

Anexo I. Clasificación completa de las etiquetas

Se dividen las etiquetas que podemos aplicar a los en los siguientes bloques:

- Propósito general
 - Ataque
 - Defensa
- Fundamentos técnicos
 - Condición física
 - Paradas
 - Arrancadas
 - Pase
 - Bote (dribbling)
 - Tiro
 - Mecánica de tiro
 - Finalizaciones
 - Fintas
 - Bloqueos
 - Rebote
- Fundamentos tácticos
 - Superioridades
 - 1x0
 - 2x1
 - 3x1
 - 3x2
 - 4x1
 - 4x2
 - 4x3
 - 5x1
 - 5x2
 - 5x3
 - 5x4
 - Juego estático
 - 1x1
 - 2x2
 - 3x3
 - 4x4
 - 5x5

Anexo II. Simbología de las figuras de los ejercicios

A continuación, se recogen los símbolos básicos y más difundidos con los que se puede dibujar cualquier ejercicio de baloncesto.

Símbolo	Descripción
	Jugador atacante.
	Jugador defensor. Los brazos orientan la posición del cuerpo del defensor.
	Balón.
	Desplazamiento de un jugador botando el balón.
	Desplazamiento de un jugador sin balón.
	Pase.
	Desplazamiento de un jugador que termina con un bloqueo/pantalla.
	Tiro.

Tabla 106: Simbología

En Figura 67 se muestra un ejemplo de cómo dos jugadores atacan contra dos defensores. El que tiene el balón avanza botando mientras que el otro se separa y corre por la banda. Es habitual utilizar colores distintos para representar a los diferentes equipos, parejas, tríos, etc., que se enfrentan unos contra otros.

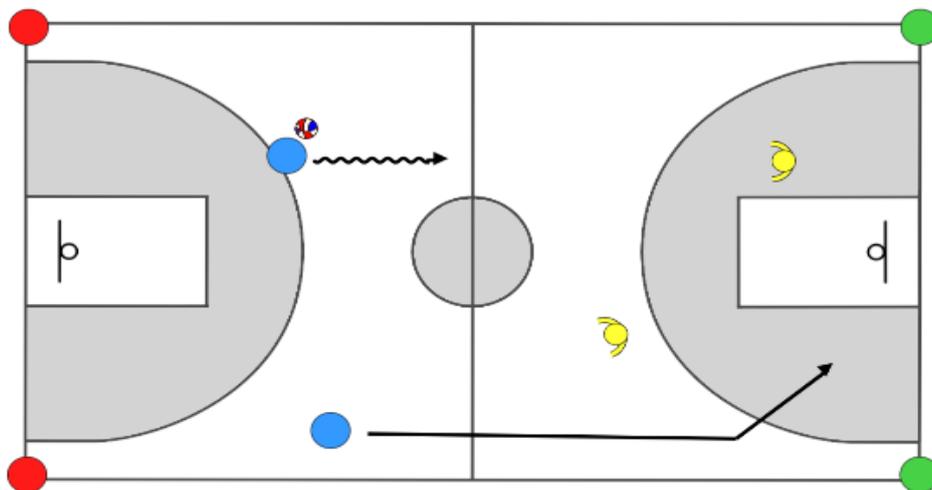


Figura 67

También se han incluido otros símbolos no tan estandarizados, pero fácilmente reconocibles como material de entrenamiento.

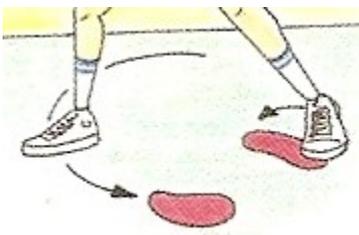
Símbolo	Imagen real	Descripción
		Cono. Útil para marcar posiciones en el campo, como el lugar donde se encontraría el defensor o el sitio hasta el que hay que desplazarse durante el ejercicio.
		Cono plano (chino). Misma utilidad que los conos altos.
		Aro. Sirve para delimitar zonas del campo, pero pudiendo situarse sobre ellas.
		Huellas de pies. Sirven para marcar movimientos específicos que deben realizar los jugadores con los pies.
		Valla. Útil para ejercicios de condición física.
		Escalera de coordinación. Útil para ejercicios de condición física.

Tabla 107: Simbología extra (materiales)

Anexo III. Boceto en papel presentado a los usuarios

Las figuras comprendidas entre Figura 68 y Figura 90 muestran el boceto de la interfaz de usuario que se les enseñó a los usuarios.

A Web Page
http://misejerciciosdebasket.com/index

MisEjerciciosDeBasket.com

Usuario

Contraseña

¿Aun no tienes cuenta? Regístrate, es gratis

Figura 68

A Web Page
http://misejerciciosdebasket.com/registro

Introduzca los siguientes datos para el registro:

Nombre Completo	<input type="text"/>	Contraseña	<input type="password"/>
Alias	<input type="text"/>	Confirmar Contraseña	<input type="password"/>
Email	<input type="text"/>	<input type="button" value="¡Regístrame!"/>	

Figura 69



Figura 70

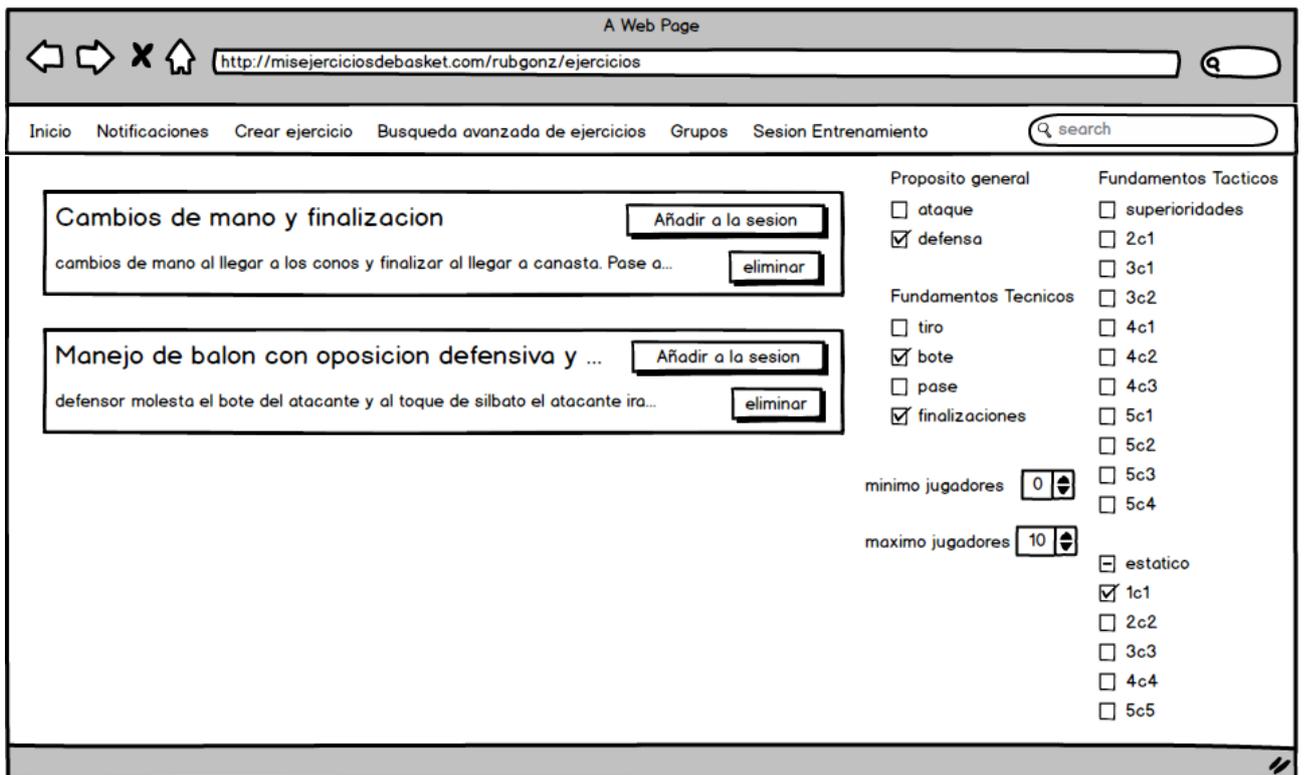


Figura 71

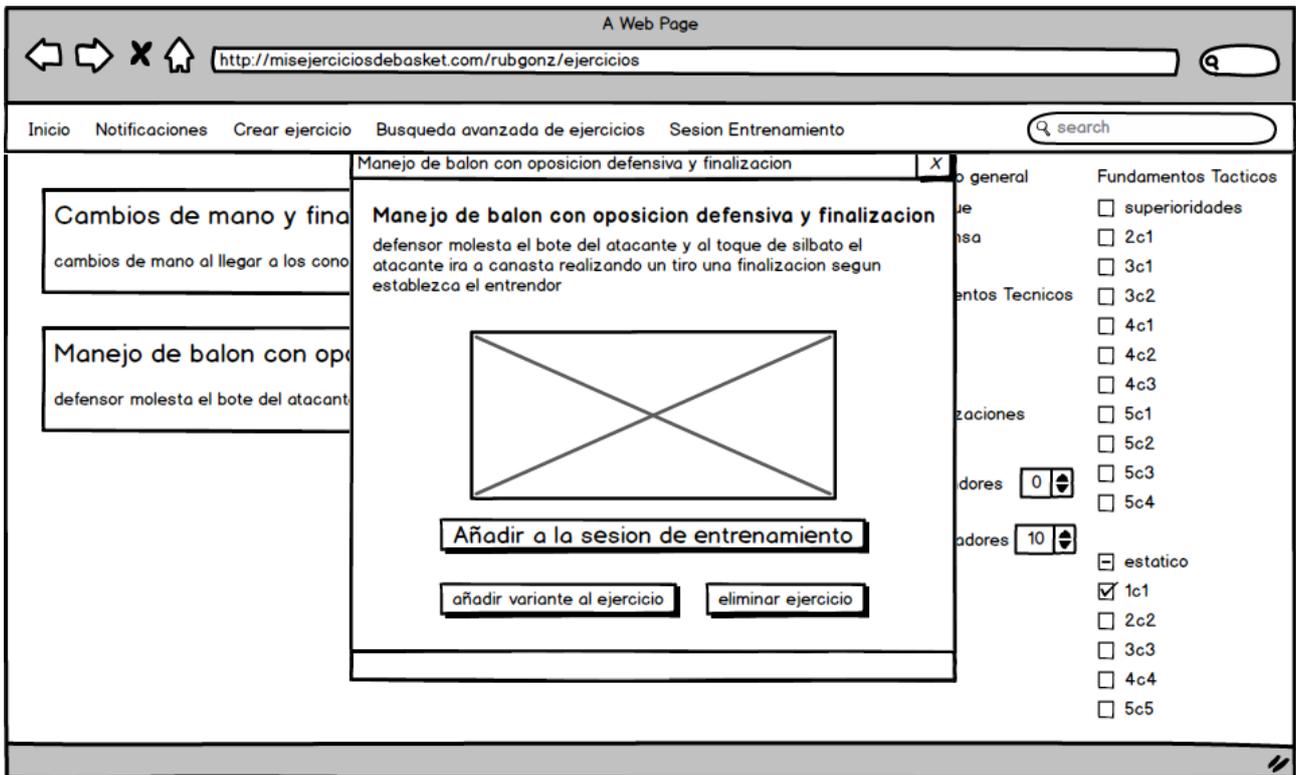


Figura 72

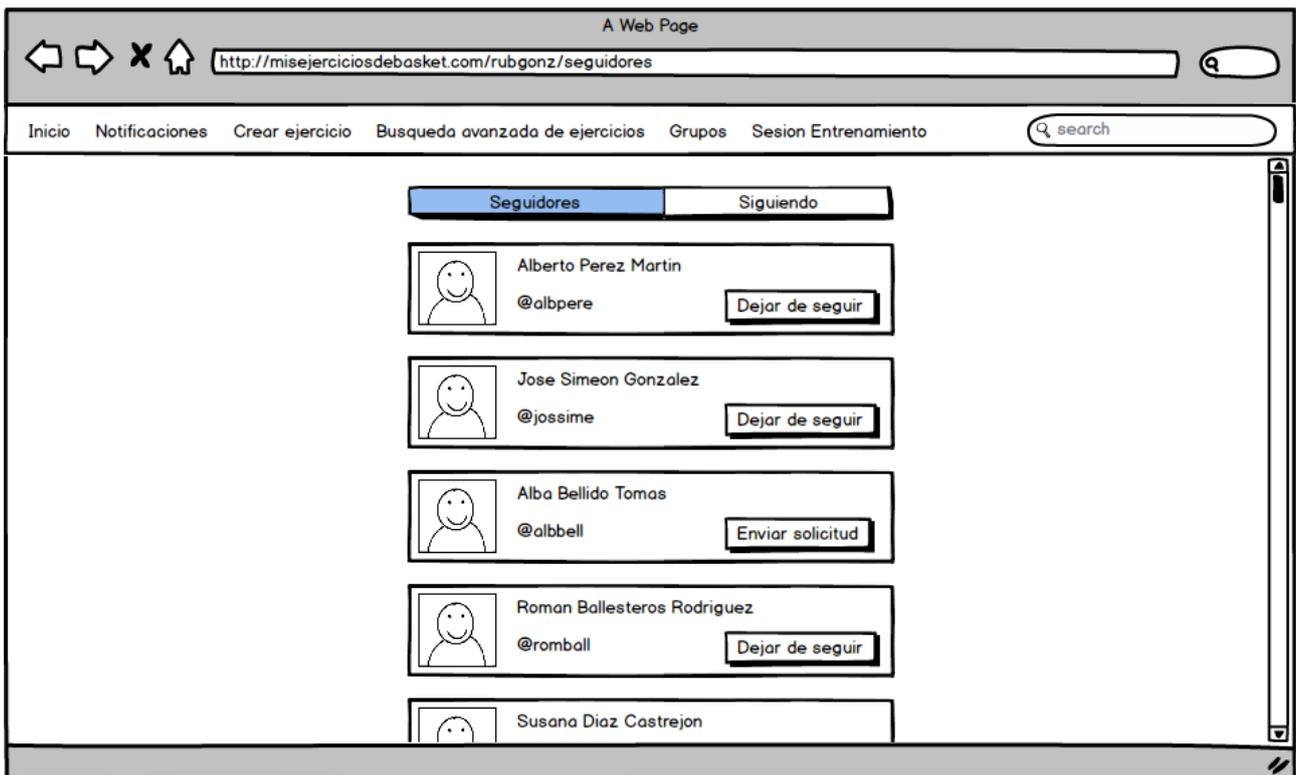


Figura 73

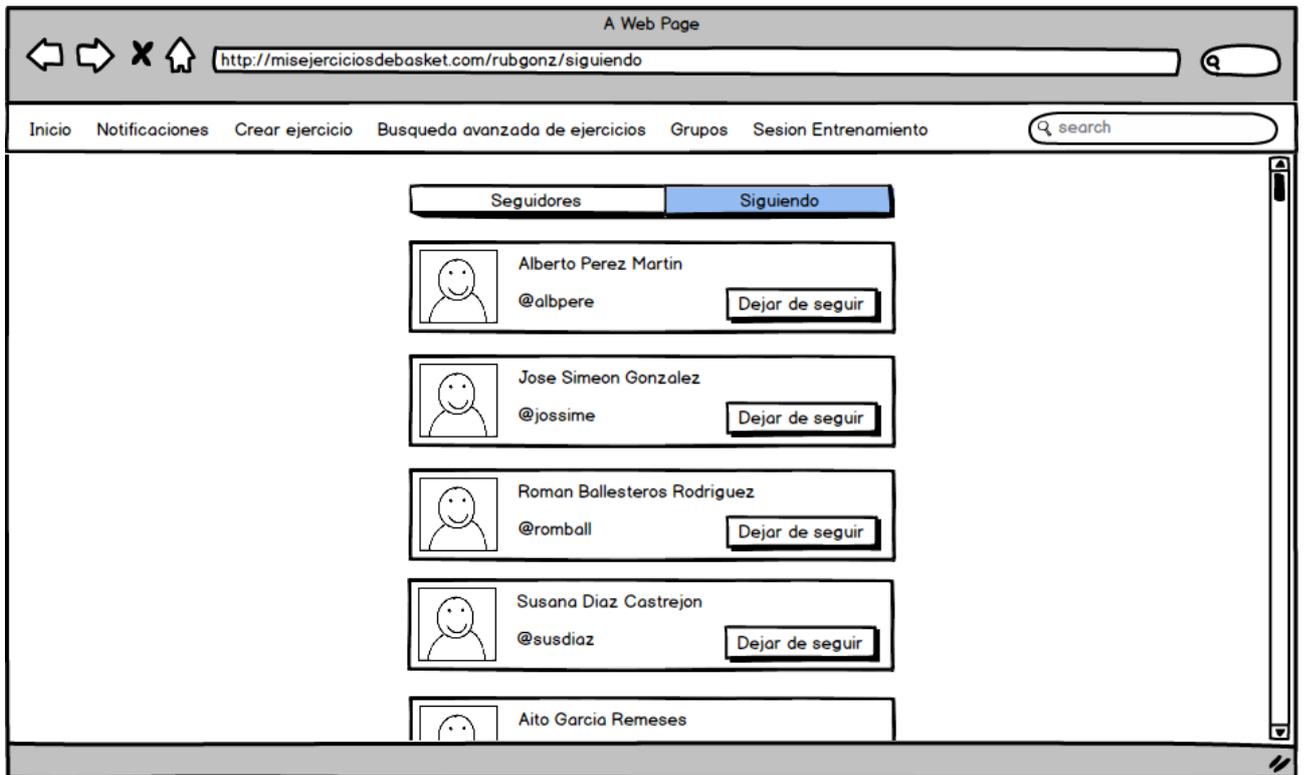


Figura 74

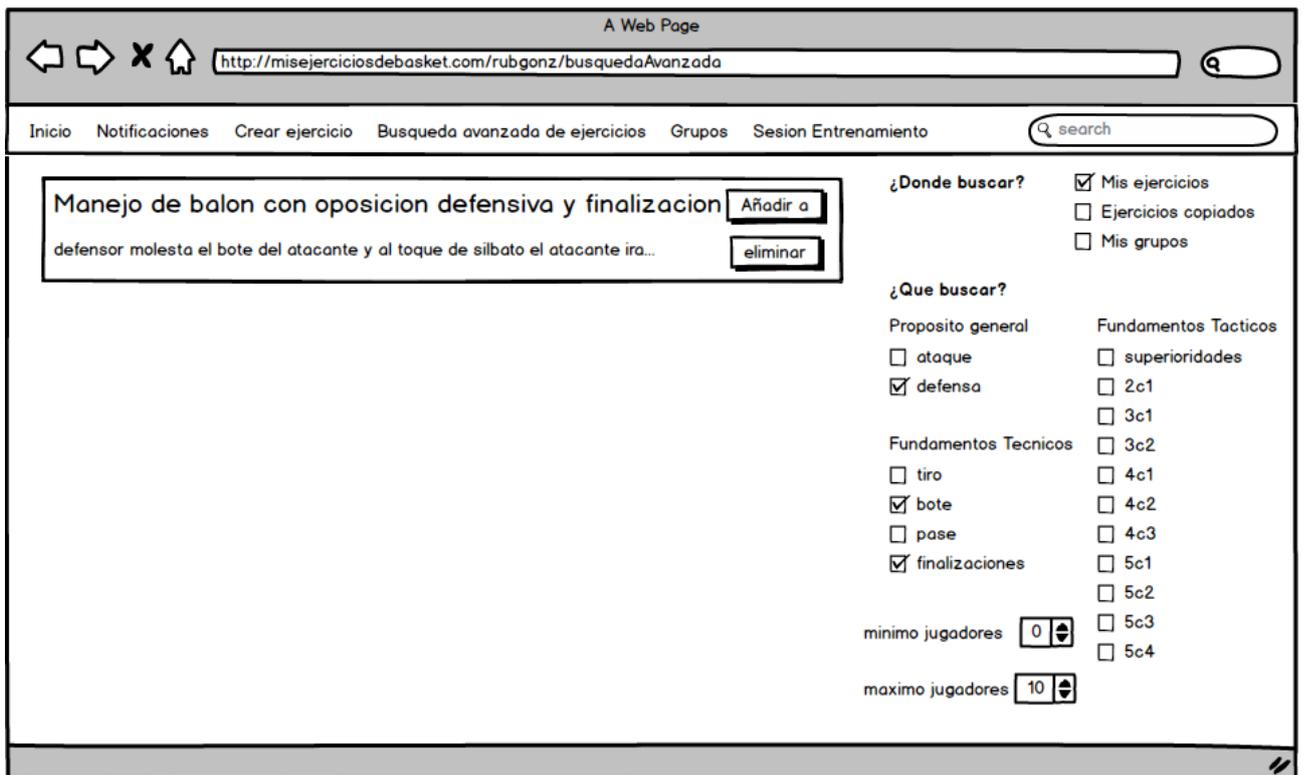


Figura 75

A Web Page

http://misejerciciosdebasket.com/rubgonz/configuracion

Inicio Notificaciones Crear ejercicio Busqueda avanzada de ejercicios Grupos Sesion Entrenamiento search

Basica Avanzada

Tu nombre Ruben Gonzalez Ruiz

¿Cual es tu nivel de entrarond? Nivel 2 ▾

¿En que club entrenas? CB Villaconejos

¿Cuantos años llevas entrenando? 3 ▾

Guardar informacion

Figura 76

A Web Page

http://misejerciciosdebasket.com/rubgonz/configuracion

Inicio Notificaciones Crear ejercicio Busqueda avanzada de ejercicios Grupos Sesion Entrenamiento search

Basica Avanzada

Cambiar nombre usuario

nuevo nombre de usuario

Cambiar direccion de correo electronico

nueva direccion correo electronico

Cambiar contraseña

Contraseña actual

Nueva contraseña

Confirmar contraseña

Guardar

Figura 77

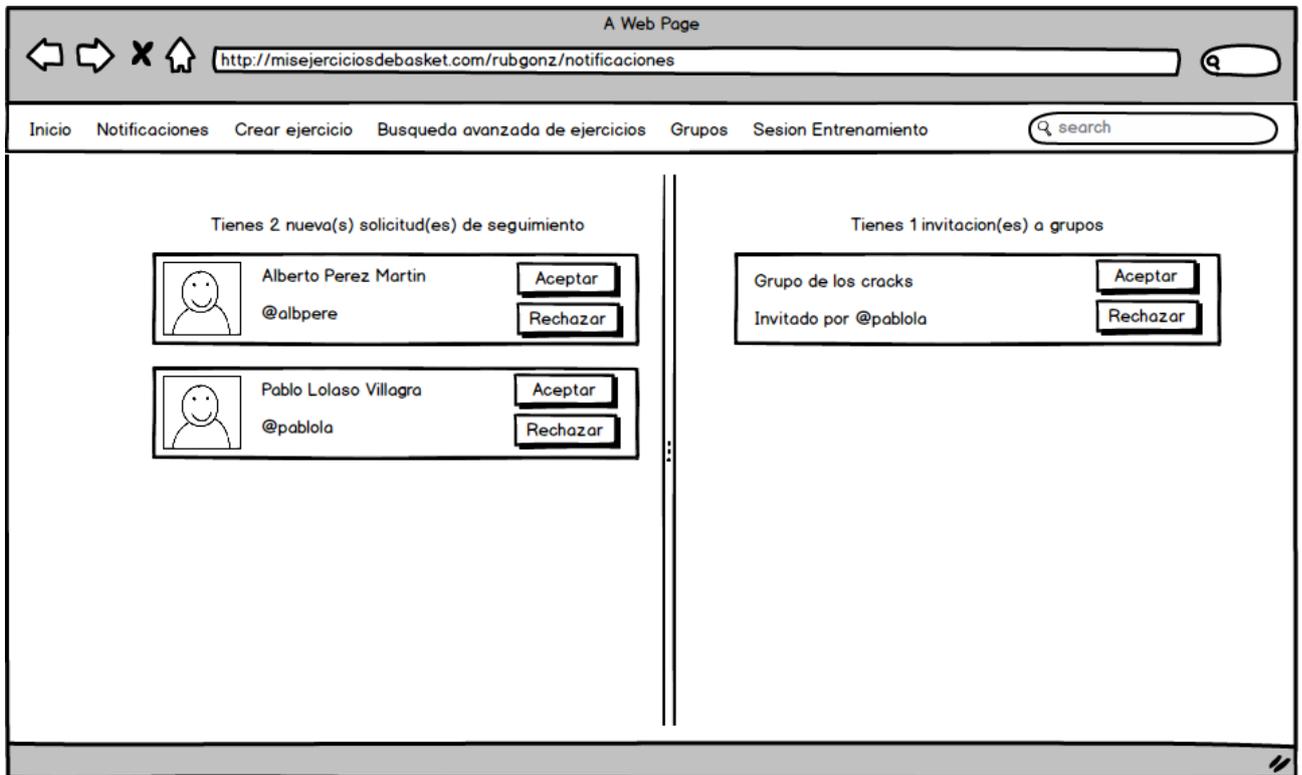


Figura 78

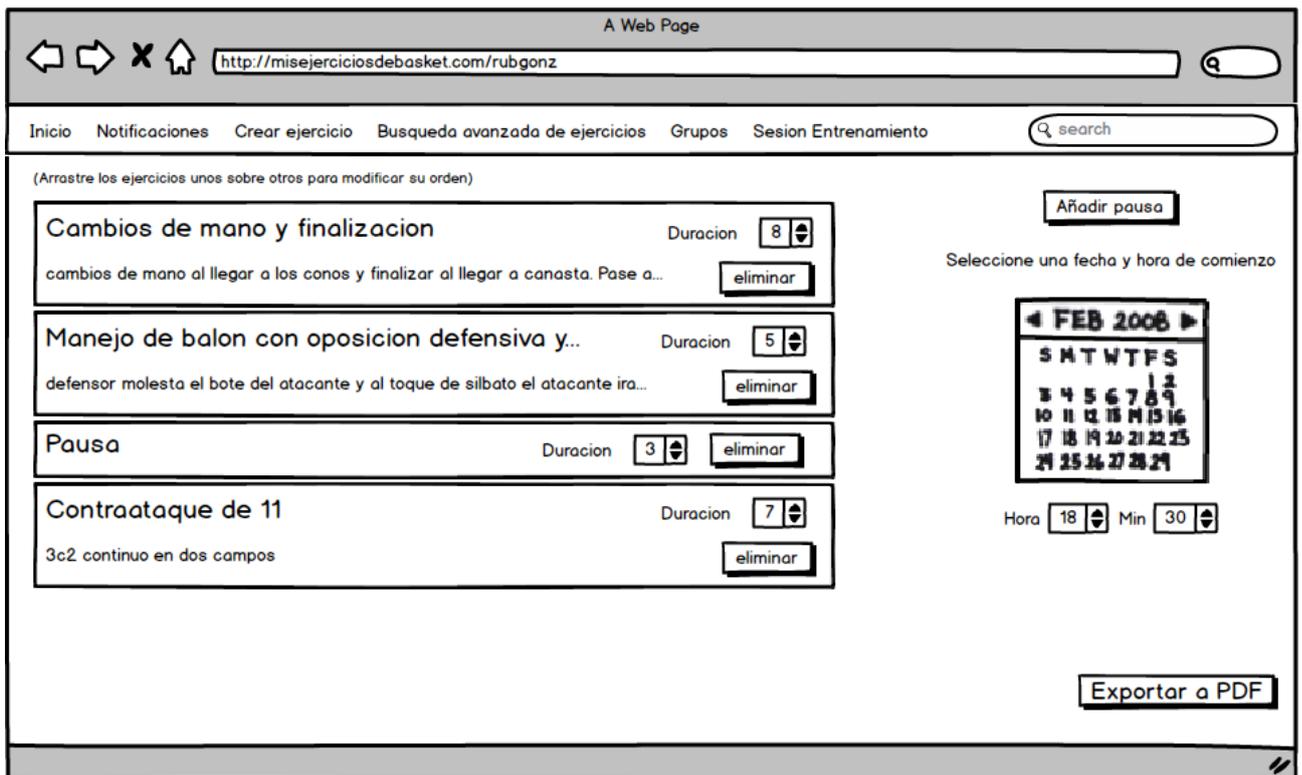


Figura 79



Figura 80

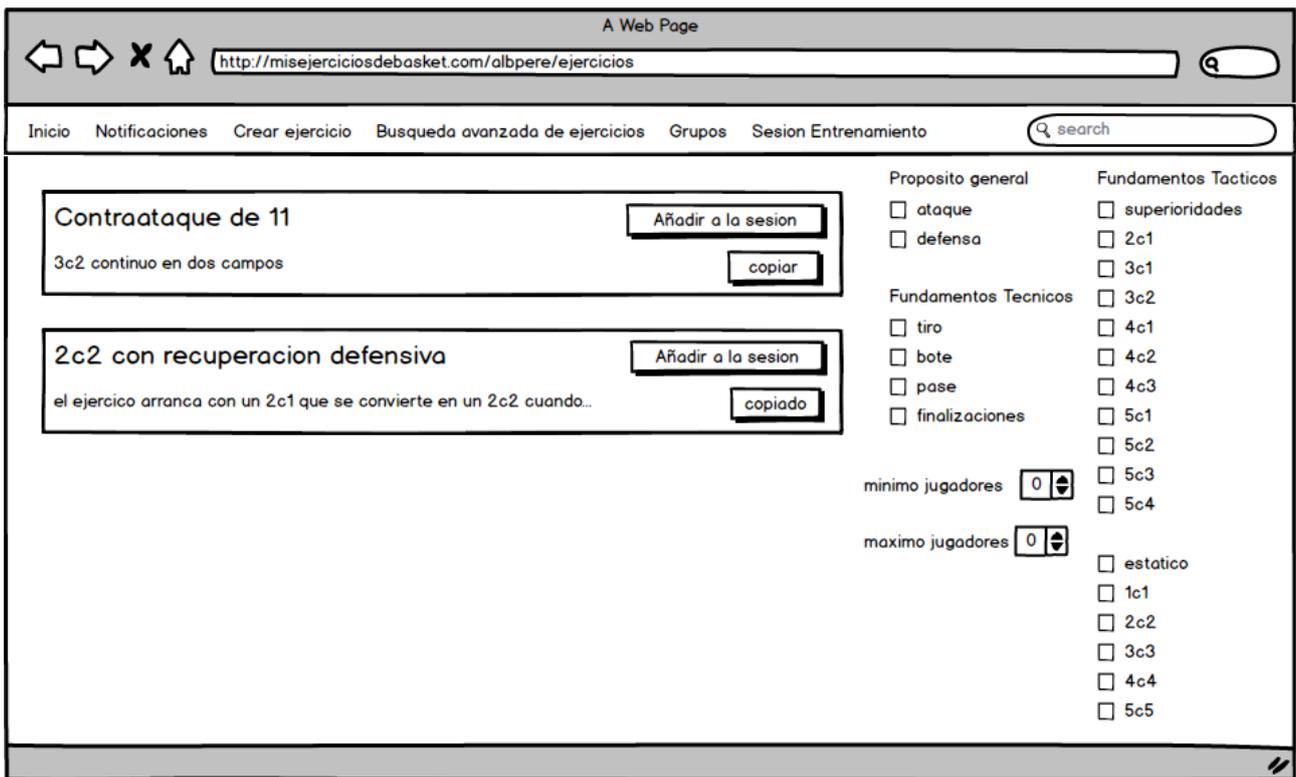


Figura 81

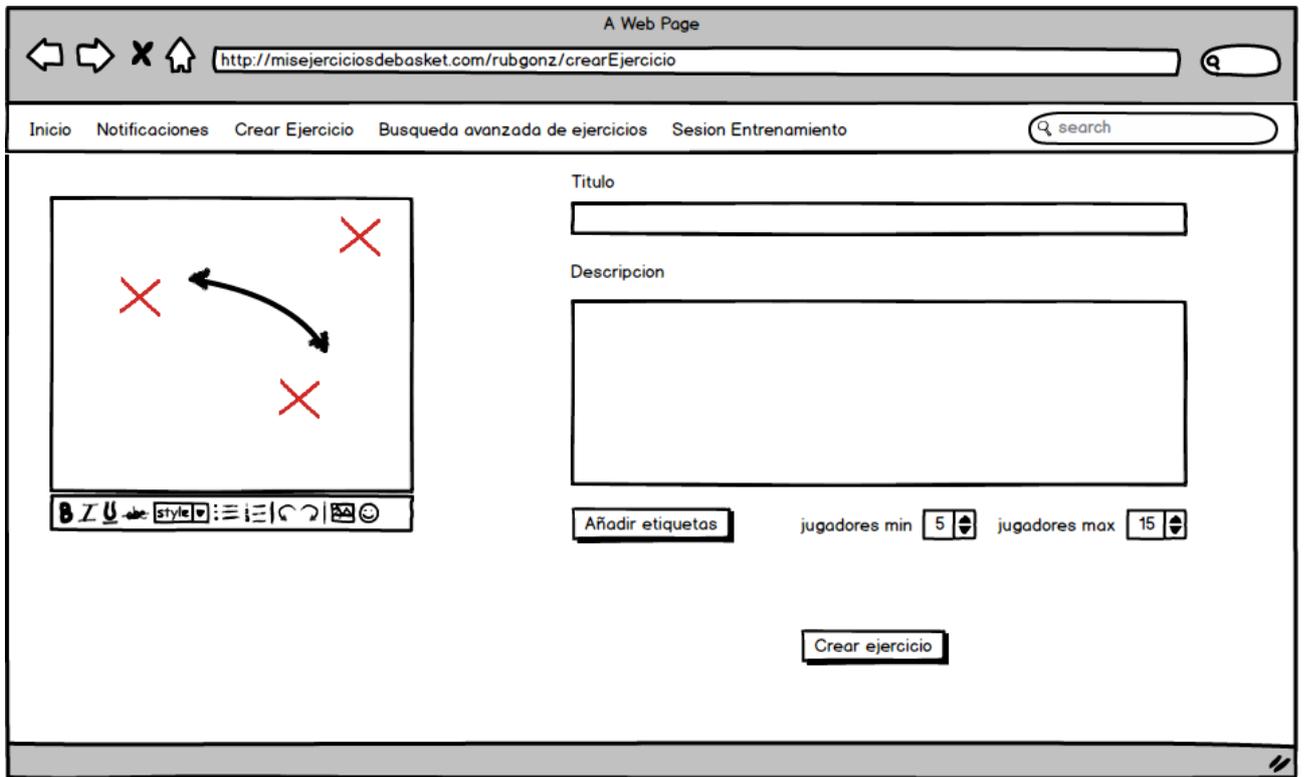


Figura 82

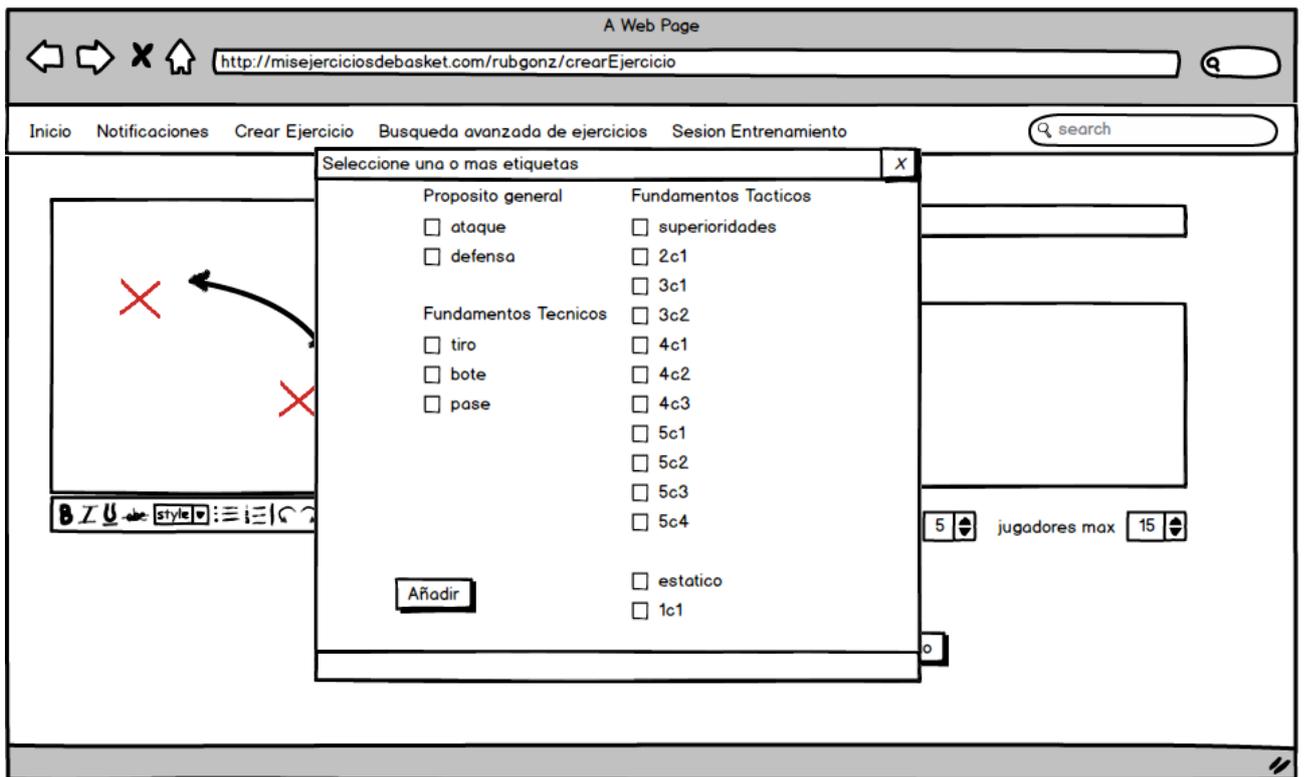


Figura 83

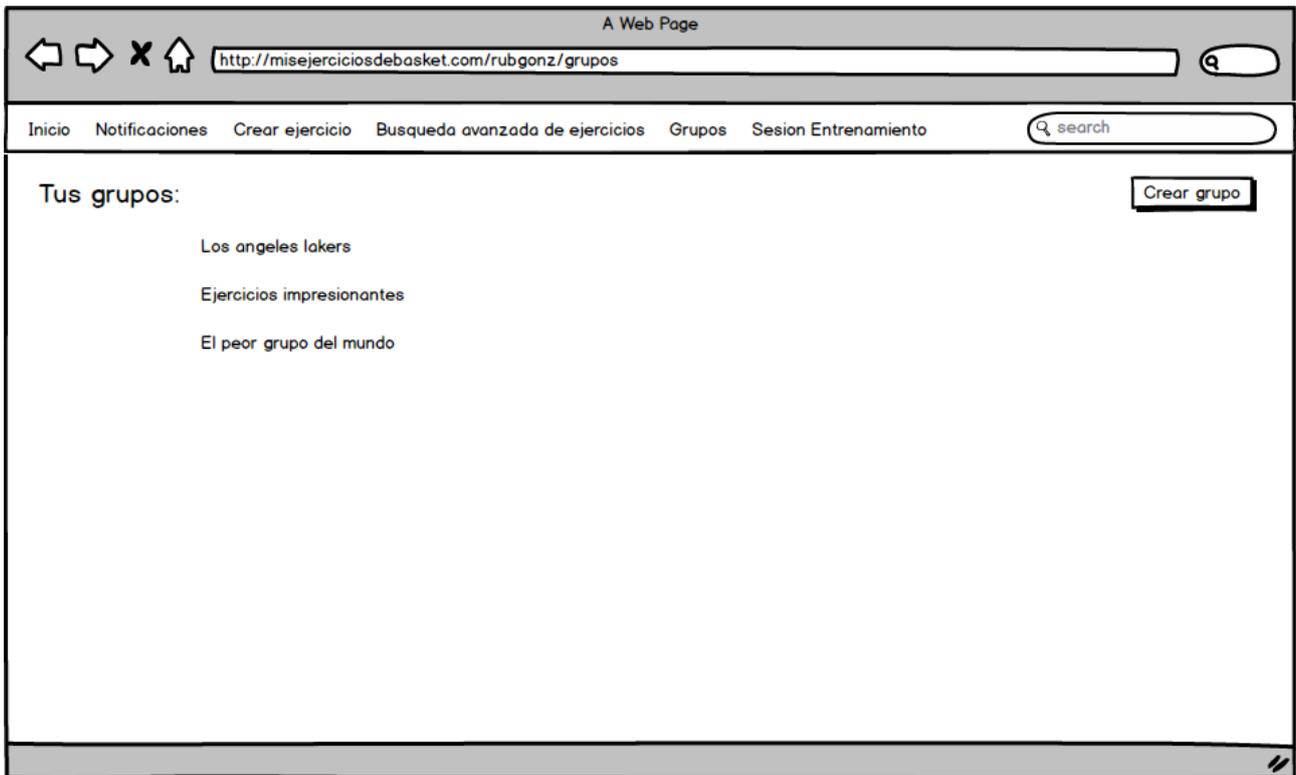


Figura 84

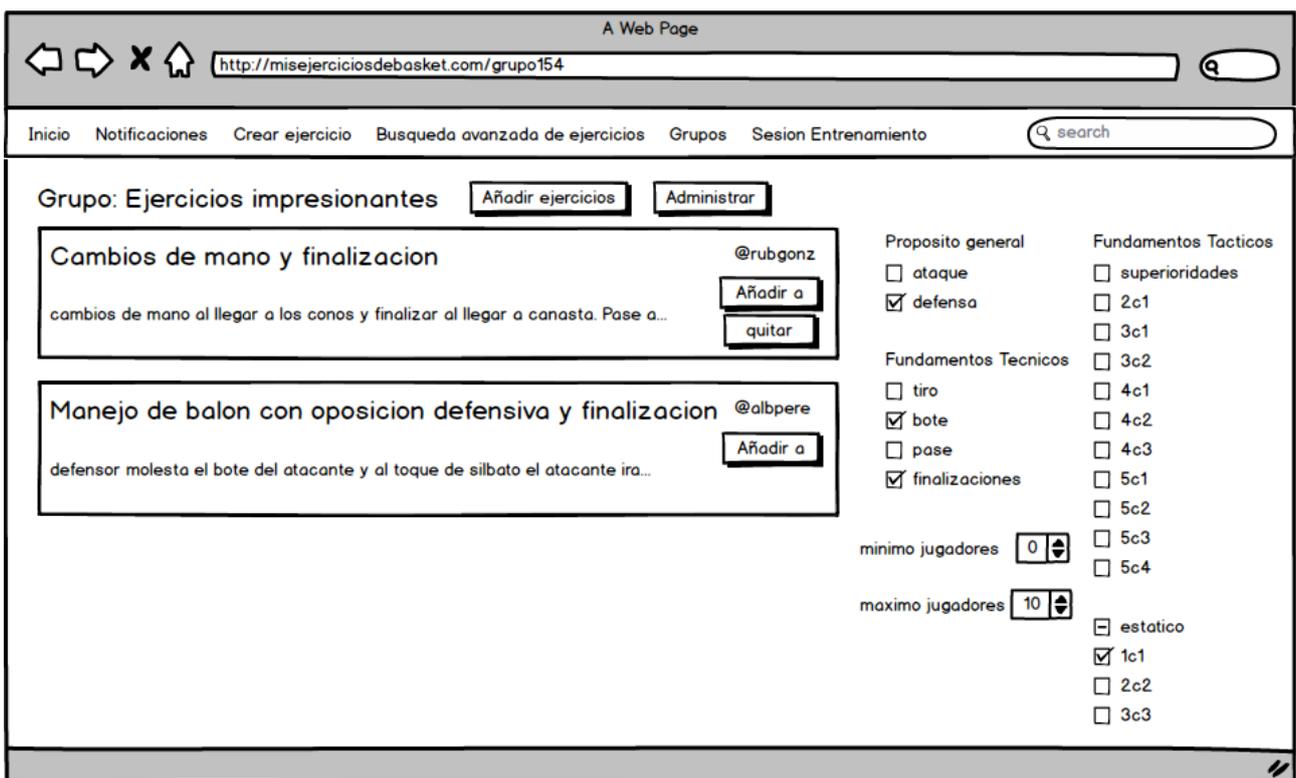


Figura 85

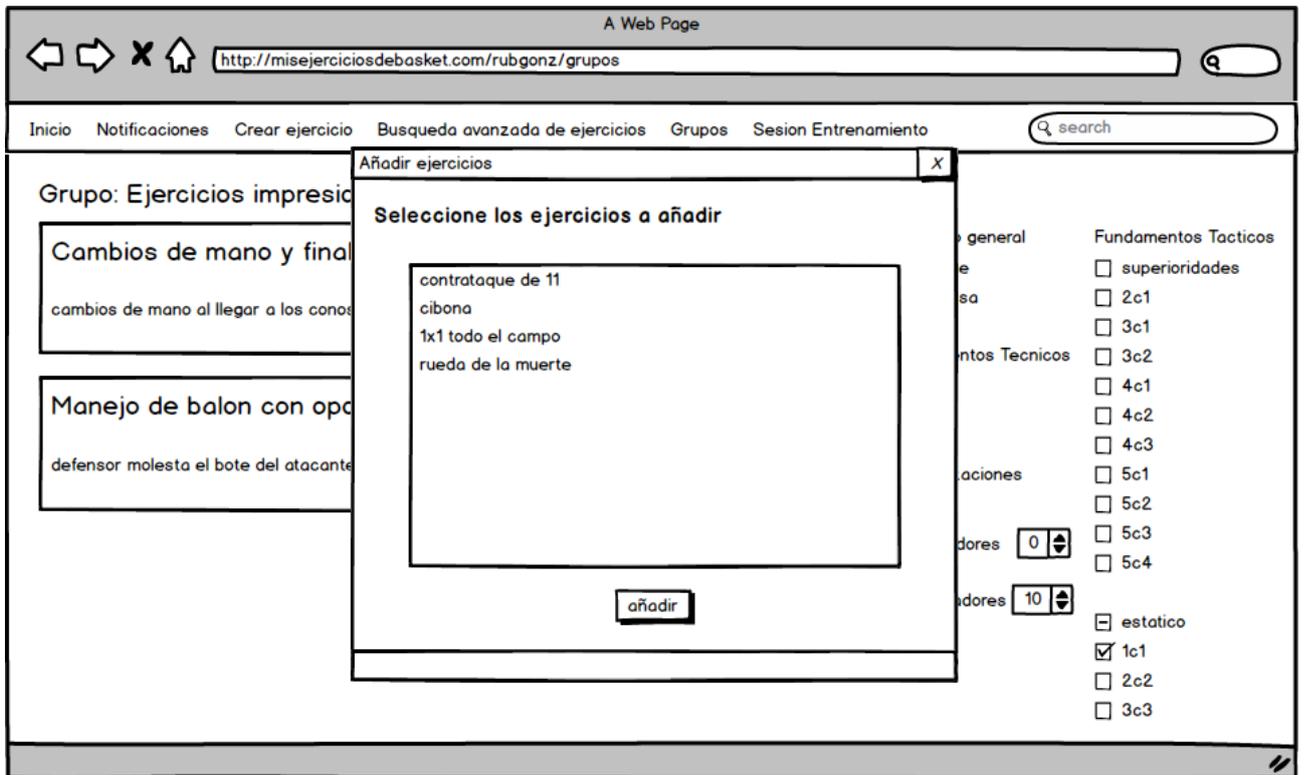


Figura 86

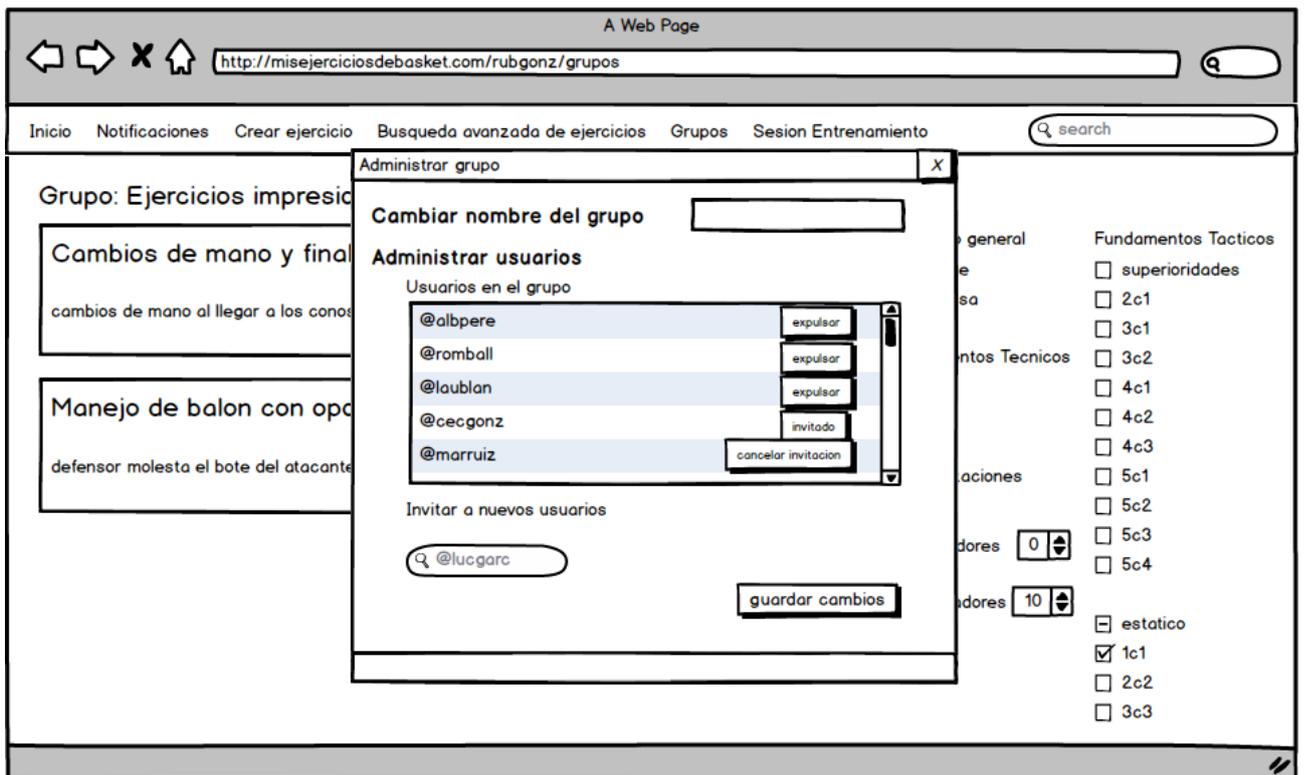


Figura 87

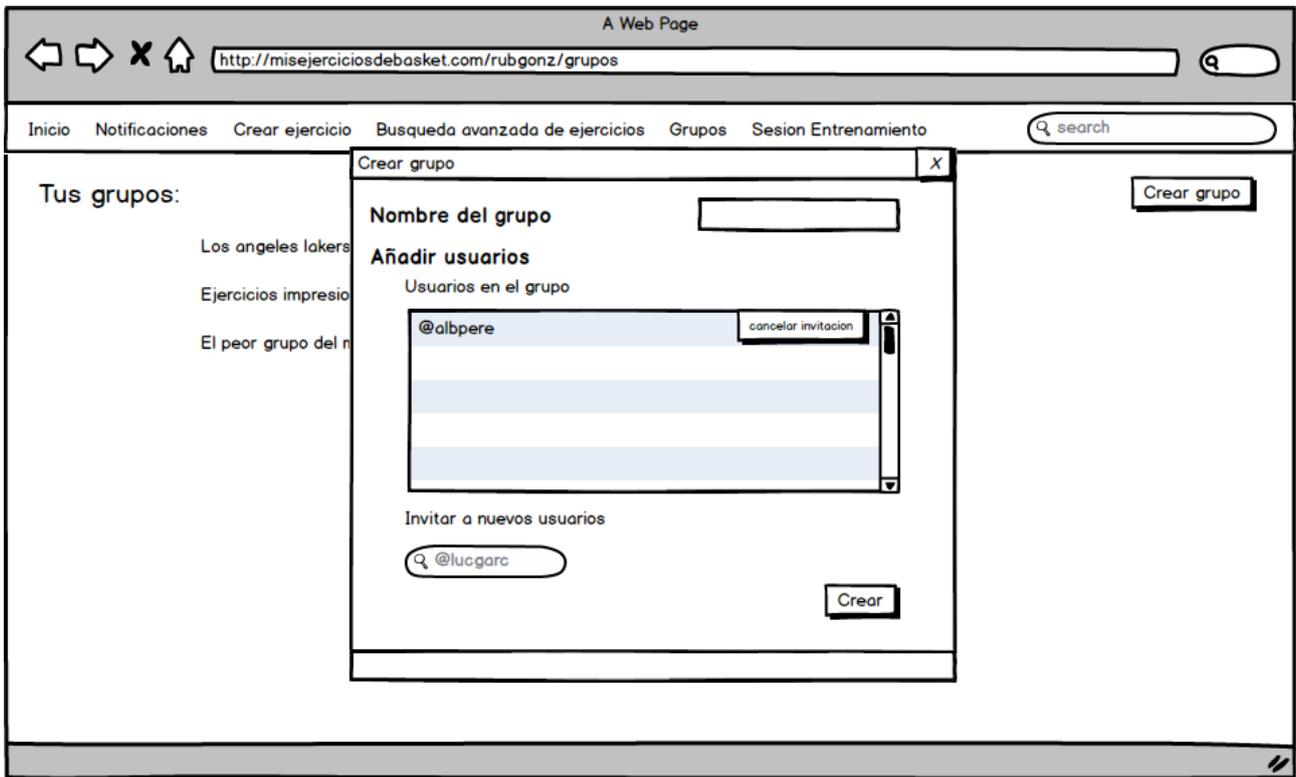


Figura 88

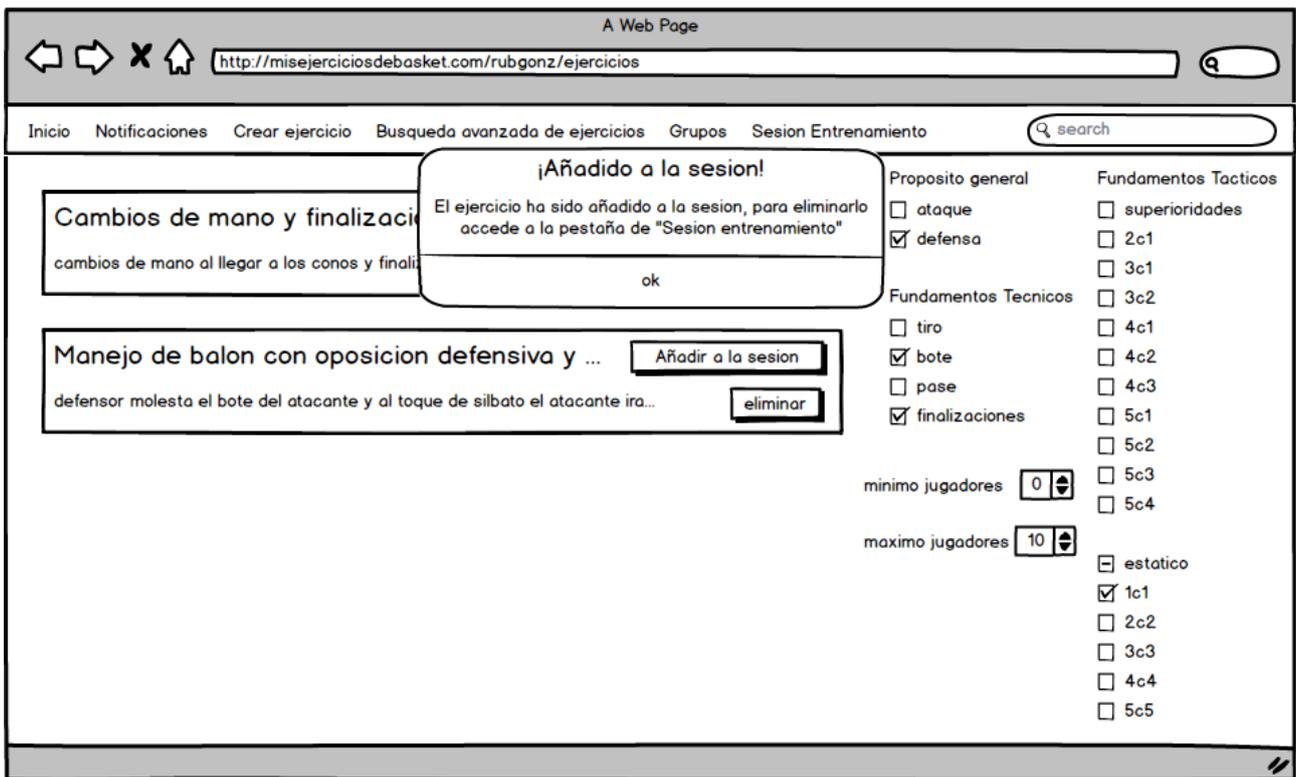


Figura 89

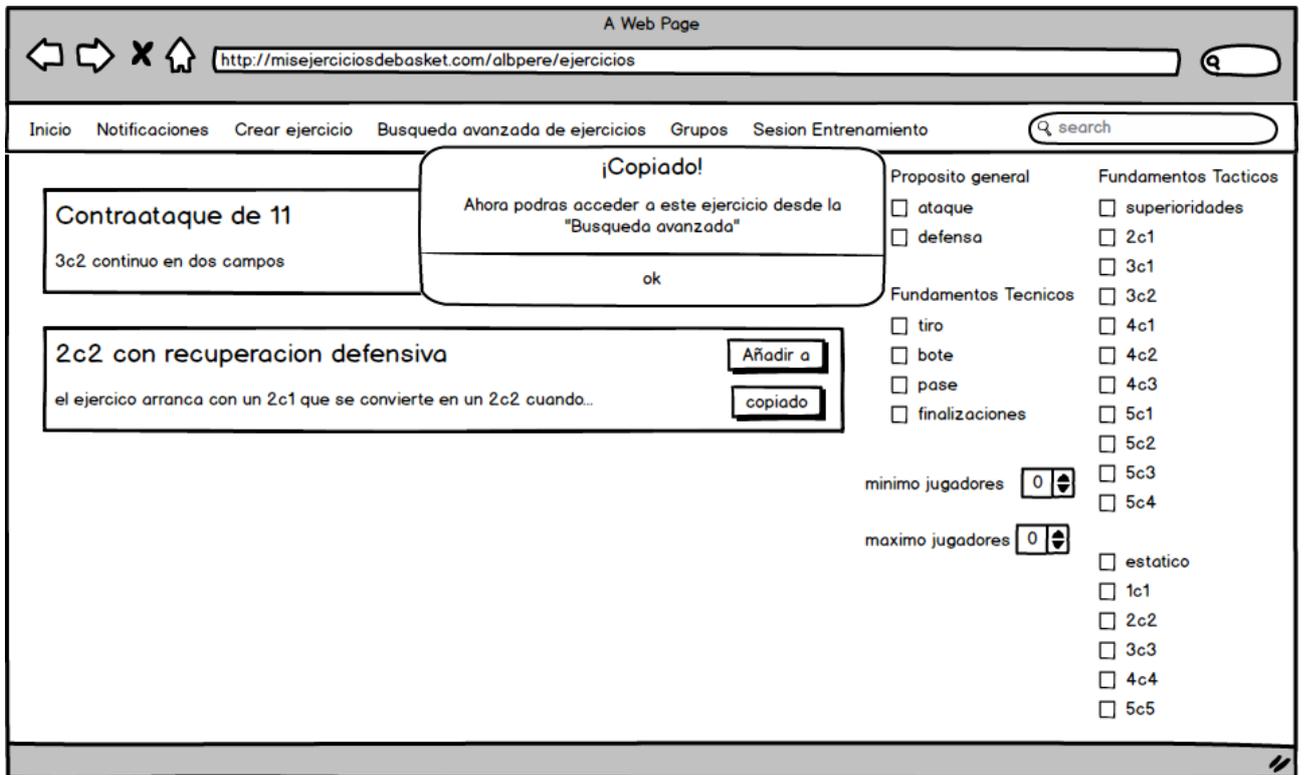


Figura 90

Anexo IV. Contenido del CD

El CD contendrá la siguiente estructura:

- Software
 - Código
 - gestorEjerciciosBaloncesto.zip
 - Scripts
 - scriptEnum.sql
- Documentación
 - Memoria
 - memoria.pdf
 - Manuales
 - manualUsuario.pdf
 - manualMantenimiento.pdf
 - manualDespliegue.pdf
 - Boceto
 - boceto.pdf