



ULPGC
Universidad de
Las Palmas de
Gran Canaria

Escuela de
Ingeniería Informática



MYGYMPLANNER

Grado en Ingeniería Informática

Autor: Geraldo Rodrigues Veloso



18 DE DICIEMBRE DE 2020

TUTORIZADO POR: ANA MARÍA PLÁCIDO CASTRO
Las Palmas de Gran Canaria

Agradecimientos

El Trabajo de Fin de Título, como su nombre indica, es la culminación de los estudios y vivencias a lo largo de la carrera. Por tanto, si tuviese que dar las gracias a todos los que han hecho posible que llegase a este punto, el documento se alargaría demasiado así que, haré un resumen de agradecimiento relacionado con la realización de este trabajo.

Primeramente, a mi familia, por todo el apoyo y amor incondicional que he recibido. Sobre todo, de mis hermanas, mi primo y mi madre. Especialmente mi madre, por toda su comprensión, por todo su cariño, por todas sus enseñanzas, por el buen ejemplo que es para mí y por ser padre y madre al mismo tiempo. No somos una familia que muestre el cariño abiertamente pero no podemos negar que no lo haya, siempre estamos para ayudarnos y para animarnos.

En segundo lugar, a mis amigos más cercanos, que son como una segunda familia para mí. Ellos me han tenido que aguantar en mis peores y mejores momentos. Han sido un gran apoyo y aunque no se lo digo mucho, estoy muy agradecido de que estén en mi vida. Me gustaría nombrar de manera especial a mi mejor amigo, él sí que ha tenido que aguantarme. Las charlas y discusiones que he tenido con él me han hecho abrir los ojos en varios aspectos de mi vida, dándome las fuerzas para mejorar como persona cada día.

Agradecer a los miembros de mi iglesia, por todas las oraciones, consejos y enseñanzas. Principalmente al grupo de jóvenes por toda la fuerza y amor que han demostrado, gracias a ellos no he dejado que Dios esté fuera de este camino, y eso ha hecho que no decaiga, por saber que siempre tendré las fuerzas para levantarme.

Y, por último, a mi tutora, Ana María Plácido Castro, por guiarme, aconsejarme y su dedicación para que este trabajo salga adelante. Realmente su ayuda en esta etapa ha sido primordial y le estoy muy agradecido por todo el tiempo que ha dedicado para perfeccionar el Trabajo de Fin de Título.

Muchas gracias, no podría haber terminado el proyecto sin todo el apoyo obtenido.

Resumen

El objetivo de este Trabajo Fin de Título es el desarrollo del prototipo de una aplicación cliente-servidor para usuarios y trabajadores de gimnasios (monitores, coordinadores, entrenadores personales, administradores), que ayudará a mejorar tanto la experiencia personal como profesional dentro de este tipo de negocios. Para la implementación del prototipo, se ha usado Angular para el desarrollo del lado del cliente y Spring para el lado del servidor.

El prototipo final ha permitido validar la solución propuesta tanto para los usuarios de un gimnasio, que podrán ver la información del gimnasio, guardar rutinas de ejercicios, reservar clases colectivas como Zumba, Pilates, etc., así como para los trabajadores, que podrán organizar las clases, ver sus horarios para organizar su trabajo diario, etc.

Abstract

The objective of this Final Degree Project is the development of the prototype of a client-server application for users and gym workers (monitors, coordinators, personal trainers, administrators), which will help improve both the personal and professional experience within this type of business. For the prototype implementation, Angular has been used for client-side development and Spring for the server-side.

The final prototype has made it possible to validate the proposed solution both for gym users, who will be able to view gym information, save exercise routines, book group classes such as Zumba, Pilates, etc., as well as for workers, who will be able to organize classes, see their schedules to organize your daily work, etc.

Índice

1. Introducción	8
1.1. Contexto.....	8
1.2. Motivación	10
1.3. Objetivos	12
1.4. Competencias específicas cubiertas	12
1.5. Aportaciones del proyecto.....	13
1.6. Estructura de la memoria	14
2. Metodología de desarrollo	16
2.1. Enfoque metodológico.....	16
2.2. Tecnologías de desarrollo	17
2.3. Proceso de desarrollo	20
2.4. Versiones del producto y entregables	22
3. Análisis preliminar	24
3.1. Descripción del problema	24
3.2. Estudio de aplicaciones similares	25
3.3. Fortalezas y debilidades principales	26
3.4. Solución propuesta	27
3.5. Plan de negocio.....	28
4. Modelado de negocio.....	30
4.1. Casos de uso de negocio.....	30
4.2. Procesos de negocio	31
5. Modelado del sistema	35
5.1. Requisitos de usuarios	35
5.2. Requisitos del sistema	37
5.3. Normativa y legislación.....	41
5.4. Diagramas de casos de uso	41

5.5. Diagramas de clases.....	50
5.6. Mockups.....	51
5.7. Arquitectura del prototipo.....	59
6. Desarrollo	64
6.1. Alcance del prototipo.....	64
6.2. Iteración 1	66
6.3. Iteración 2	67
6.4. Iteración 3	70
6.5. Iteración 4	71
6.6. Iteración 5	72
6.7. Iteración 6	73
7. Acceso al código y despliegue	74
8. Conclusiones y trabajo futuro	75
8.1. Resultados del proyecto	75
8.2. Posibles actuaciones futuras	76
9. Fuentes de información.....	78
10. Anexo. Manual de usuario	81
10.1. Acceso a la aplicación	81
10.2. Administrador	82
10.3. Socio.....	86
10.4. Monitor.....	90

Índice de figuras

Figura 1	9
Figura 2	16
Figura 3	28
Figura 4	32
Figura 5	33
Figura 6	42
Figura 7	44
Figura 8	45
Figura 9	46
Figura 10	47
Figura 11	48
Figura 12	50
Figura 13	51
Figura 14	52
Figura 15	52
Figura 16	53
Figura 17	53
Figura 18	54
Figura 19	54
Figura 20	55
Figura 21	55
Figura 22	56
Figura 23	56
Figura 24	57
Figura 25	57
Figura 26	58
Figura 27	58
Figura 28	59
Figura 29	60
Figura 30	63
Figura 31	81
Figura 32	82
Figura 33	83
Figura 34	83

Figura 35	84
Figura 36	85
Figura 37	85
Figura 38	86
Figura 39	87
Figura 40	87
Figura 41	88
Figura 42	89
Figura 43	89
Figura 44	90
Figura 45	91
Figura 46	91
Figura 47	92
Figura 48	93
Figura 49	93

1. Introducción

1.1. Contexto

Para poder entender el objetivo y la motivación de este trabajo de fin de título, es necesario explicar en primer lugar el mercado al que va dirigido: la industria fitness en España, más concretamente, los gimnasios low cost. A continuación, se definirán conceptos necesarios para entender lo que es la industria del fitness y el funcionamiento de los gimnasios low cost.

Hoy en día, podemos observar que cada vez son más las personas que quieren llevar una vida más saludable: “una vida fitness”.

¿Qué significa tener o llevar una vida fitness? Podemos encontrar dos definiciones para entender lo que es esta tendencia. La primera, sería llevar una vida de ejercicio diario y una dieta saludable para mejorar la salud física y el bienestar. La segunda, realización de un conjunto de ejercicios gimnásticos que se repiten varias veces por semana para obtener una buena forma física. Normalmente estos ejercicios se realizan en centros deportivos específicos como son los gimnasios. [1]

Durante años, el mercado del fitness ha estado en auge en EE. UU. Las cadenas de gimnasios de este país cerraron el 2018 con una facturación de 28.874 millones de euros [2], quedando en el número uno del top ten de facturación de este mercado, siguiéndole Alemania con 5.639 millones. Podemos observar que hay una gran diferencia entre los dos primeros puestos, por eso, a EE. UU. se le conoce como la potencia y cuna de este tipo de negocios. Aun así, este tipo de industria se ha extendido hasta Europa, incluyendo a España, que se encuentra en el top 10 de este ranking, cerrando el 2018 con 2.410 millones de euros y registrando 5,3 millones de socios, siendo el octavo país en volumen de abonados y el quinto mayor mercado del fitness en Europa, quedando detrás de Alemania, Reino Unido, Francia e Italia [3].

Por tanto, podemos ver que, en España, este estilo de vida también está en auge, aunque no es reciente. Desde el 2015, la facturación del fitness español ha aumentado un 7,5%. Y cada año va creciendo, aunque el ritmo ha ido decayendo los últimos años, según expertos es porque ya está alcanzando la madurez [4]. “El mundo del fitness se ha especializado y sofisticado. Esto se ve en todo: en la forma en la que se viste la

gente para hacer deporte, en el tipo de actividad que se practica y también en la difusión que le damos en las redes sociales, donde compartimos los entrenamientos. Esto lo hace más divertido”, explica el experto de Deloitte [5].

La industria en España se divide en gimnasios low cost, Premium, gestores de instalaciones municipales y centros boutiques.

Los gimnasios low cost son los que más crecieron en 2018 y se piensa que seguirán creciendo en números de abonados en los próximos años. En la siguiente figura se muestra la evolución del segmento de mercado Gimnasio low cost en España: el número de empresas privadas con gimnasios low cost y cantidad de gimnasios a fecha de 1 de enero de 2019, según informe realizado por Valgo Investment [36]

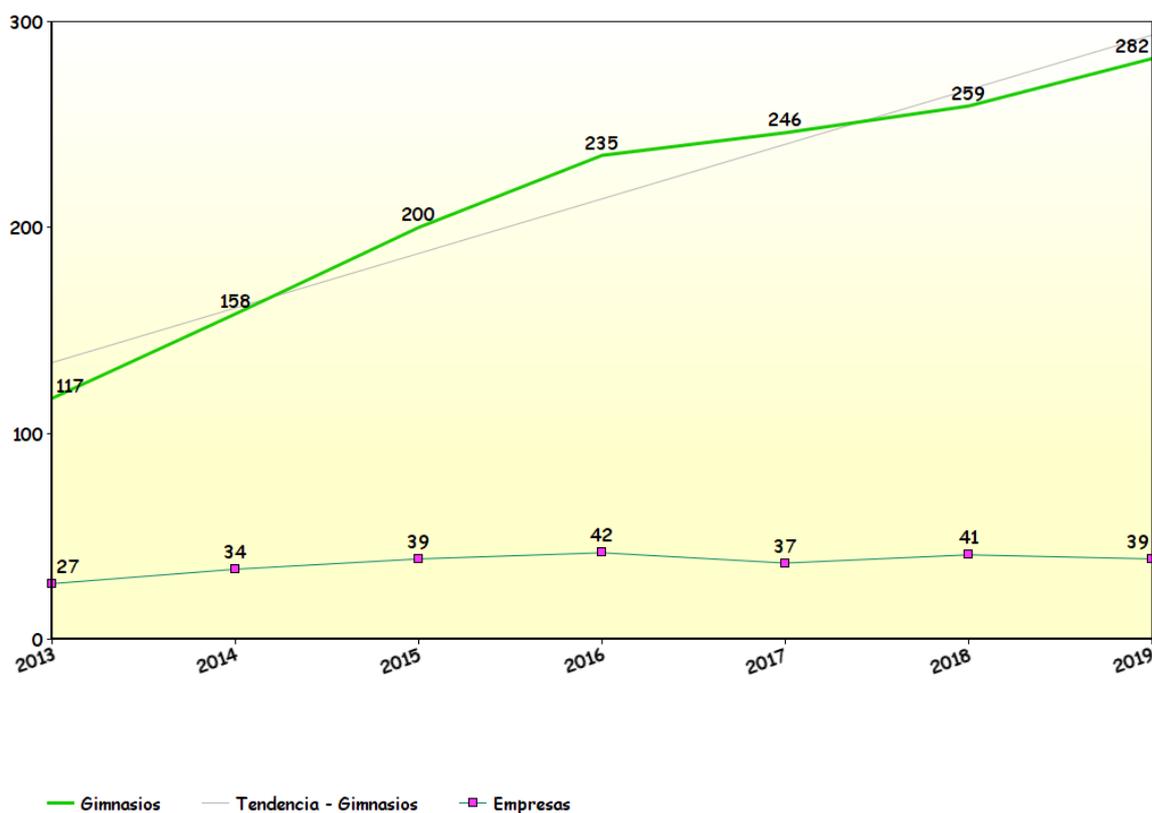


Figura 1

A continuación, se explica cuál es el modelo de negocio de los gimnasios low cost para entender por qué son los que más han crecido en los últimos años.

El modelo de negocio low cost, básicamente es bajar los precios para aumentar la demanda y generar beneficio. Este tipo de negocio solo triunfa si se tiene una gran cantidad de clientes e ingresos por otros servicios. El concepto de calidad es darle al cliente lo que quiere y un poco más, sin que este poco más sea un coste muy elevado [6].

A finales del siglo XX, las personas presumían de gastarse mucho dinero en productos de calidad, hoy presumen por saber encontrar buenos productos a precios más económicos. La mentalidad del cliente ha cambiado y los gimnasios se han adaptado. Hay expertos que dicen que los gimnasios Premium serán una minoría en unos años. El socio ya no pagará 50 euros mensuales, si puede ir a un gimnasio que le ofrece lo que quiere por 30 euros menos [7].

Este cambio de mentalidad ha surgido por la crisis económica que ha dejado huella en las familias españolas, disminuyendo el poder adquisitivo de las mismas y contribuyendo a que los socios de gimnasios Premium se cambien a gimnasios low cost.

Por otra parte, las personas fitness de España se caracterizan por el gran interés que tienen respecto a las clases dirigidas. En una encuesta realizada por LifeFitness respecto al mercado fitness español, casi el 60% de los encuestados asistía normalmente a entre 1 y 3 clases dirigidas distintas en su instalación deportiva [8]. Aunque muchos centros deportivos también ofrecen este tipo de servicios, los gimnasios de low cost, además de tener precios muy competitivos, se caracterizan por tener varias clases dirigidas incluidas en la cuota mensual, entre 20 y 30 e incluso más de 30 a la semana.

Con todo esto, se puede concluir que el sector al que va dirigido este trabajo de fin de título es un sector importante, que está en auge y que despierta mucho interés a nivel mundial.

1.2. Motivación

La motivación inicial para realizar una aplicación dentro del mundo fitness es que me interesa todo lo relacionado con la salud y el bienestar físico y, como usuario de este tipo de gimnasios, considero que las soluciones software que existen podrían mejorarse para adaptarse a las necesidades de este tipo de negocios y de sus usuarios.

Hemos visto en el apartado anterior, que el mundo del fitness está creciendo y evolucionando en España, pero, aun así, sigue habiendo un gran porcentaje de personas que no realizan ejercicio físico, poniendo en riesgo su salud y calidad de vida.

Según un estudio de la Organización Mundial de la Salud, el 22,9% de los hombres y el 30,5% de las mujeres no hacen el ejercicio suficiente recomendando, y un macro estudio realizado en el 2016, nos confirman que el 34,4% de la población no hace ninguna actividad física en su tiempo libre y el 38,9% lo hace de manera ocasional, es decir, que se concluye que el 73% de la población no hacía actividad física suficiente para mantener un buen estado de salud [9].

Gran parte de la población sabe que, para tener una vida saludable, es recomendable tener una buena dieta y buenos hábitos deportivos. El boom del mundo fitness ha hecho que los españoles estén más activos, ya sea, por querer cuidarse o publicar imágenes en las redes sociales. Un 46% de la población española hace deporte cada semana. 5,2 millones va al gimnasio y 3,7 millones optan por el deporte federado [10]. El ir al gimnasio, hacer ejercicio en casa o en el parque y el uso de aplicaciones de entrenamiento, ha aumentado la cifra de deportistas en el país.

Según un informe de Statista, las aplicaciones dedicadas al fitness generaron unos ingresos de 28,5 millones de euros en 2017. Por tanto, es una realidad que hay más gente que se interesa en el mundo fitness [11]. Y charlando con varios usuarios de gimnasios, afirman que usan aplicaciones deportivas, ya sean de progreso, apoyo nutricional o entrenamientos. Según el estudio de LifeFitness, 6 de cada 10 centros declaran que sus socios utilizan aplicaciones deportivas, 8 de cada 10 creen que sus socios valoran positivamente que el centro cuente con soluciones digitales, 7 de cada 10 ofrecen una solución tecnológica de fitness, pero solo el 39 % de los socios sabe que su centro usa una aplicación fitness y solo mitad la ha descargado.

La realidad es que este tipo de aplicaciones podría ayudar a mejorar la calidad del servicio que ofrecen los gimnasios, pero no se utilizan lo suficiente por parte de los socios. Podría ayudar a conseguir que más personas lleven una vida fitness y si a esto añadimos que los clientes satisfechos son la mejor publicidad para el negocio, también podría ayudar a aumentar el número de socios.

1.3. Objetivos

El objetivo del trabajo fin de título ha sido el desarrollo de un prototipo para gimnasios low cost: “MyGymPlanner”. Una aplicación que ayudará a mejorar tanto la experiencia personal de los clientes como el trabajo del personal en este tipo de negocios.

El prototipo implementa las funcionalidades más importantes que se han identificado en el análisis y se ha desarrollado con un diseño *responsive* [28], para que se pueda usar tanto desde un ordenador personal como desde un dispositivo móvil. El desarrollo del prototipo nos ha permitido comprobar la viabilidad del producto y futura evolución hasta completar todas las funcionalidades previstas en el análisis.

1.4. Competencias específicas cubiertas

IS01. Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la ingeniería del software.

Antes de desarrollar, se realizó un estudio previo de los requisitos que tendrían el proyecto, siendo estos asequibles de desarrollar y mantener siguiendo buenas prácticas de ingeniería del software, siguiendo los principios del Clean Code.

IS02. Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones.

En el estudio de los requisitos, se realizó un estudio de mercado y entrevistas para conocer las necesidades del cliente y poder especificar requisitos que satisfagan estas necesidades.

IS03. Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles.

Durante desarrollo se ha trabajado con dos módulos que se ejecutaban en diferentes puertos, por tanto, se siguió los estándares de Angular y Spring para poder realizar la comunicación entre los dos módulos.

ISO4. Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.

A lo largo del desarrollo, cada vez que surgía un problema, se solucionaba a partir de la documentación oficial de las tecnologías, haciendo uso de la metodología ágil iterativa e incremental, además de usar técnicas y herramientas actuales.

ISO6. Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos.

La aplicación busca que los gimnasios tengan soluciones para mejorar la calidad de los servicios ofrecidos con el objetivo de aumentar la fidelización de los socios, consiguiendo que una parte de la población se comprometa a llevar una vida más fitness, mejorando así su salud.

1.5. Aportaciones del proyecto

El proyecto realizado aportaría al mundo del fitness y, por tanto, a nuestro entorno socioeconómico, una aplicación útil y fácil de usar donde los usuarios de los gimnasios podrán gestionar su día de una manera más eficiente. Cuando nos referimos a usuarios, nos referimos principalmente a monitores, coordinadores y socios, además de los entrenadores personales que tengan socios en los gimnasios.

En el caso de los monitores y coordinadores, les aportaría una herramienta para gestionar las clases que pudiera llegar a tener el gimnasio, poder ver sus horarios y sus clientes y, en el caso que ofrezcan asesoramiento personal, un servicio que, no siendo un entrenador personal, ayude a los socios a alcanzar sus objetivos.

A los socios del gimnasio le aportaría un acercamiento al gimnasio, al tener una serie de servicios de su gimnasio en su dispositivo electrónico. Como hemos visto en los apartados anteriores, las personas que suelen ir al gimnasio usan aplicaciones externas para apoyar su entrenamiento y gestionar su progreso, así como dispositivos para el

conteo de calorías entre otros funcionamientos. La idea del proyecto es que gran parte de esas funcionalidades estén concentradas en una misma aplicación y así, fidelizar al socio, ya que, como hemos comprobado, los socios valoran positivamente el uso de aplicaciones en los gimnasios.

Y, por último, a los entrenadores personales, una gestión de sus clases de entrenamiento que les ayude en su día a día.

1.6. Estructura de la memoria

En este apartado, se describe la estructura de la memoria del Trabajo de Fin de Título con un resumen de los temas que se tratan en los siguientes apartados.

Primeramente, el apartado a seguir es donde se detalla la metodología que se seleccionó para realizar el desarrollo del proyecto. Esta metodología es la metodología de desarrollo iterativo e incremental. Se da una introducción de cómo funciona y cómo ha sido implementada en el proyecto. Además, en este mismo apartado, se indican las diferentes tecnologías con las que se desarrolló la aplicación, explicando la elección de cada una de ellas.

En el apartado 3, se describe el análisis que se realizó para desarrollar la aplicación. Primero se trata la descripción del problema que se intenta solucionar. Después de este, se puede observar el análisis de aplicaciones similares. También se realizó un análisis de las fortalezas y debilidades principales del proyecto, para ello se realizó un análisis DAFO. A continuación, se describe la aportación de la solución propuesta.

En el apartado 4, se encuentra el modelado del negocio, donde se describe los casos de uso del negocio y los procesos en los que se enfoca la aplicación a desarrollar.

En el apartado 5 se describe el modelado del sistema. En este apartado se encuentran los requisitos de usuario, los funcionales y los no funcionales del proyecto, los diagramas de caso de uso, los diagramas de clases, el mockup y por último la arquitectura del prototipo.

En el apartado 6, se describe todo el proceso de desarrollo e implementación de la aplicación hasta llegar al entregable final.

Los últimos apartados, el 7, 8, 9 y 10 se encuentra como acceder al código y el despliegue de este, las conclusiones de la memoria, las fuentes de información y por último los anexos de la memoria.

2. Metodología de desarrollo

2.1. Enfoque metodológico

La metodología de desarrollo que se decidió usar para este proyecto ha sido la de iterativa e incremental [12]. Esta metodología es un proceso de desarrollo software creado en respuesta a las debilidades del modelo tradicional en cascada. Este modelo es uno de los más utilizados en los últimos tiempos y es una de las bases de un proyecto ágil.

En un desarrollo de iterativo e incremental, el proyecto se planifica en bloques temporales, llamados iteraciones. De manera general, las iteraciones suelen durar cortos periodos de tiempo, pocas semanas o un mes, luego, podemos decir que cada iteración se trata como un mini-proyecto. En cada iteración, se repite un proceso de trabajo similar que debe incluir revisión, pruebas y documentación del producto, mejorando así la calidad del software.

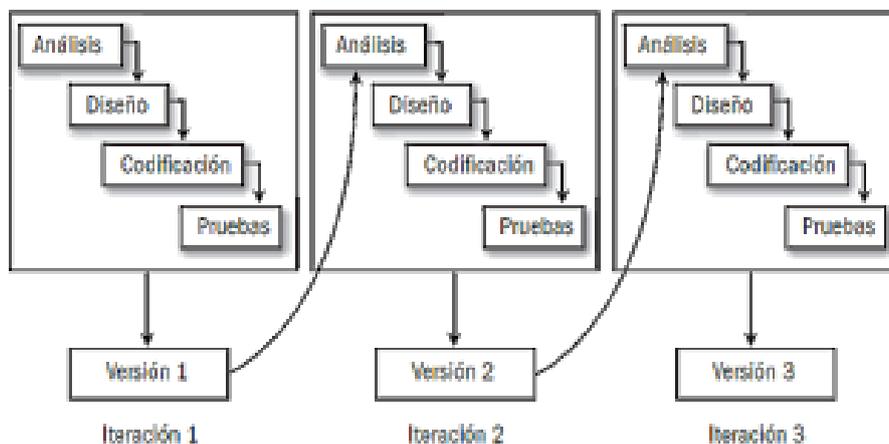


Figura 2

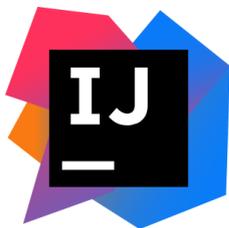
La parte incremental de la metodología viene de que, en cada iteración, el proyecto evoluciona a partir las iteraciones anteriores, añadiendo nuevos objetivos y/o requisitos o mejorando los que ya fueron completados. Es decir, que se le va agregando más funcionalidad al software en cada iteración.

Los beneficios más importantes de la metodología son los siguientes:

- En cada iteración, ya se puede obtener resultados importantes y utilizables, así gestionando las expectativas del cliente de manera regular.
- Se puede gestionar de manera natural los cambios, ya que, al final de cada iteración se realiza una revisión general del proyecto y, por tanto, se planifican las siguientes iteraciones de manera efectiva, mejorando la productividad y calidad.
- Se pueden reducir los riesgos, por el mismo motivo del punto anterior.
- Se gestiona la complejidad del proyecto, ya que, en una iteración, solo se trabaja los requisitos que aportan más valor en ella.

2.2. Tecnologías de desarrollo

2.2.1. IntelliJ IDEA



IntelliJ IDEA [13] es un entorno de desarrollo para el desarrollo de programas informático. Fue desarrollado por JetBrains y está disponible en dos ediciones: edición para la comunidad y edición comercial.

Este entorno de desarrollo ha sido diseñado para maximizar la productividad a la hora de codificar, teniendo en cuenta la ergonomía. Se basa en el principio de que cada minuto que el desarrollador se mantiene en el flujo es un buen minuto.

La razón por la cual se decidió usar IntelliJ como entorno de desarrollo fue precisamente por las características que presenta para poder centrarse en la codificación, como, por ejemplo, proporciona asistencia de codificación profunda, navegación rápida, análisis inteligente de errores y refactorizaciones, ya que analiza el código desarrollado, buscando conexiones en todos los archivos y lenguajes del proyecto.

2.2.2. Spring



Spring [14] es una plataforma para desarrollar en Java de código abierto. La filosofía de Spring se basa en la convención sobre configuración, reduciendo el número

de pasos que el desarrollador debe dar en la configuración inicial del proyecto, esto permite que el desarrollador pueda enfocarse en la lógica que requiere la aplicación.

Para la realización de la parte del servidor del proyecto se decidió usar Spring, principalmente porque está diseñado de manera modular, tiene varios paquetes y clases, pero solo se usan aquellos que son necesarios para el proyecto, al ser el framework más popular de java cuenta con una comunidad muy activa que genera una gran cantidad de documentación y probar una aplicación se vuelve un proceso simple, ya que, el código se traslada a este framework de manera transparente al usuario.

2.2.3. Angular



Angular [15] es una plataforma para desarrollar aplicaciones web y móvil en TypeScript de código abierto, para facilitar la creación de aplicaciones de una sola página, SPA.

Para el desarrollo de la parte del cliente se hizo uso de angular. Angular es un framework modular, basado en componentes reutilizables en el entorno de desarrollo, gracias a este diseño, las páginas web son más rápidas, ya que, no hace falta que se esté recargando la página en cada cambio de vista, solo es necesario que aparezcan los componentes usados para cada una de las vistas de la aplicación.

Esta es una de las razones por las cuales se decidió hacer uso de angular, además, cada componente contiene una serie de plantillas que mantienen por separado el código de la interfaz de usuario y el de la lógica de negocio. Esas dos características, más el conjunto de herramientas de líneas de comando, Angular CLI [16], permiten que el desarrollo sea más rápido y eficaz.

2.2.4. Trello



Trello[17] es una herramienta que organiza las tareas por tableros.

Se usó esta herramienta para determinar que se va a producir en este momento. Está basado en la metodología Kanban, una metodología ágil cuyo objetivo es gestionar de manera general como se van completando las tareas.

2.2.5. Git, GitHub, GitKraken



Git [18] es un software para el control de versiones de un proyecto, GitKraken [19] es una interfaz para Git que facilita la gestión de este y GitHub [20] es una plataforma que almacena proyectos de forma remota usando Git.

Por tanto, se usó Git para poder gestionar las versiones del proyecto. GitKraken para no tener que estar trabajando con las líneas de comando, si no, con una interfaz, ya que, al trabajar con una interfaz, facilita el uso de Git. Y GitHub, para tener un repositorio remoto que almacene el proyecto.

El entorno de desarrollo de IntelliJ cuenta con herramientas para la gestión de Git, por tanto, de cierta manera, se podría ver como innecesario el tener una interfaz como GitKraken para usar las herramientas de Git, pero al tenerlo separado, se trabaja de manera más eficaz, gracias al diseño de GitKraken, a la hora de visualizar las diferentes versiones y gestionar el repositorio local y remoto.

2.2.6. MySQL, MySQL WorkBench



MySQL [21] es un sistema de gestión de bases de datos relacional, esto se refiere que archiva datos en tablas separadas, y MySQL WorkBench [22] es una interfaz visual para MySQL.

Por tanto, para almacenar los datos que tendrá la aplicación se usó una base de datos en MySQL.

2.2.7. StarUml



StarUml [23] es una herramienta UML [27] (Lenguaje de modelado unificado, de sus siglas en inglés, unified modeling language), es decir, un software que permite modelar de manera gráfica un sistema usando las especificado de UML.

Por tanto, para la realización de los diferentes diagramas se hizo uso de esta herramienta, principalmente por la familiaridad que ya se tiene por su uso a lo largo de la carrera.

2.2.8. Balsamiq



Balsamiq [24] es un software para realizar prototipos estáticos de una manera efectiva y rápida.

Para la creación de los mockups, se decidió usar esta herramienta, por la facilidad y rapidez para realizarlos que ofrece Balsamiq.

2.3. Proceso de desarrollo

El proceso de desarrollo intentó ser fiel a la metodología de desarrollo ágil iterativa e incremental. Para la representación de los requisitos se hizo uso de las historias de usuarios, ampliamente utilizadas en las metodologías de desarrollo ágiles.

Una historia de usuario es una o dos frases utilizando el lenguaje común del usuario. Las historias de usuario deben responder a tres preguntas: ¿A quién va dirigido?, ¿Qué es lo que se quiere? y ¿Cómo se beneficia? El estilo de las historias suele ser libre, pero algunos autores recomiendan el siguiente formato, el cual se usó en este proyecto: “Como (rol del usuario) quiero (acción) para (beneficio)”.

Durante el proceso de desarrollo también se siguió los principios del manifiesto ágil [25]. A continuación, se describirá algunos puntos del manifiesto y su implantación en el proyecto según la metodología iterativa e incremental:

- Nuestra principal prioridad es satisfacer al cliente a través de la entrega temprana y continua de software de valor. Cada dos iteraciones, se intentaba contactar con algunos futuros usuarios finales para que pudiesen ver el progreso y conocer sus puntos de vista.
- Son bienvenidos los requisitos cambiantes, incluso si llegan tarde al desarrollo. Los procesos ágiles se dobligan al cambio como ventaja competitiva para el socio. A través de las opiniones de los usuarios, se intentaba identificar nuevos requisitos, o modificar o mejorar los que ya se tenían para poder trabajarlos en las siguientes iteraciones, siguiendo la parte incremental de la metodología que se estaba implementando.
- Entregar con frecuencia software que funcione, en periodos de un par de semanas hasta un par de meses, con preferencia en los periodos breves. Uno de los requisitos personales que se intentó llevar, es que, siempre se guardaba en Git, software que compilase.
- La simplicidad como arte de maximizar la cantidad de trabajo que se hace es esencial. Durante la implantación, se intentó seguir las cualidades del código limpio. Una de las cualidades sigue el principio KISS (*Keep It Simple, Stupid*), para evitar la complejidad innecesaria y enfocarse en las cosas más simples que hagan que el software funcione.
- En intervalos regulares, el equipo reflexiona sobre la forma de ser más efectivo y ajusta su conducta en consecuencia. Al finalizar cada iteración, se reflexionaba de cómo poder mejorar para la siguiente iteración.

A continuación, se describen las actividades del proceso de desarrollo que siguió cada iteración:

1. Identificar las historias de usuario que se desea implementar en la iteración, teniendo en cuenta los requisitos y/o las opiniones de usuario.
2. Describir en un Word las historias, las validaciones de estas, las horas estimadas para la realización de cada una, priorizarlas según el contenido y/o importancia cara a los diferentes roles.
3. Elegir una historia según prioridad y añadir en Trello las tareas para el desarrollo de esta.

4. Implementar los servicios y vistas necesarias para satisfacer las validaciones.
5. Una vez acabada, volver al punto 3.
6. Realizar y ejecutar pruebas unitarias sobre los servicios creados.
7. Realizar una prueba dinámica de las vistas creadas.
8. Ejecutar todas las pruebas unitarias y realizar una prueba dinámica sobre todo el proyecto.
9. Volver al punto 1 si las pruebas fueron satisfactorias, en caso contrario, corregir donde fuese necesario y volver al punto 8.

A lo largo del desarrollo del proyecto, la estructura del proceso ha ido evolucionando por cuestiones de calidad y optimización de este:

- En vez de realizar las pruebas unitarias y dinámicas al finalizar una iteración, se realizaban al acabar cada historia, así, se facilitaba las tareas de optimización y refactorización sobre el código, ya que, se realizaba conjuntamente al realizar las pruebas.
- En vez de colocar cada una de las tareas de una historia específica y su descripción en Trello, creando, por tanto, tantas tarjetas como tareas, se colocaba una tarjeta con la identificación numérica de la historia y el nombre, y en la descripción de la tarjeta se colocaba las tareas planteadas para la historia. Por tanto, facilitaba relacionar tareas con historias.

2.4. Versiones del producto y entregables

Durante el desarrollo del proyecto se obtuvieron las siguientes versiones del prototipo:

- Versión 1. Esta primera versión del prototipo implementa las vistas principales del socio y el administrador.
- Versión 2. La segunda versión del prototipo implementa las vistas de las rutinas y las clases dirigidas para el socio y la vista principal del monitor.

- Versión 3. La tercera versión del prototipo implementa la funcionalidad que permite al socio reservar clase y la que permite al monitor añadir socios a las clases.
- Versión 4. La cuarta versión del prototipo implementa la funcionalidad para la creación de equipos.
- Versión 5. La quinta versión del prototipo implementa la funcionalidad para distribuir clases y asignar monitores a las clases.
- Versión 6. La quinta versión del prototipo muestra una mejora visual de las funcionalidades anteriores.

Los entregables que se han obtenido en este trabajo de fin de título son los siguientes:

- Modelado de negocio. Descripción de los procesos de negocio.
- Modelo de requisitos. Especificación de requisitos de usuario y del sistema.
- Diseño de la UI. Modelado de la interfaz de usuario del prototipo.
- Arquitectura de la aplicación. Diseño arquitectónico del prototipo.
- Manual de usuario. Guía de usuario para el manejo de la aplicación.

3. Análisis preliminar

3.1. Descripción del problema

La aplicación final del proyecto dará solución a los problemas que experimenta cada usuario del gimnasio cuando se encuentra en el mismo, mejorando así la experiencia personal de cada uno.

Los problemas que la aplicación intenta abordar para los socios son:

- Creación de rutinas, es decir, una manera más fácil de poder generar y obtener una rutina dentro del gimnasio. Ya que, aunque el gimnasio tenga rutinas, muchas veces son limitadas, no modificables y normalmente las rutinas no están enfocadas en lo que el socio quiere conseguir, al ser genéricas.
- Reservar las clases colectivas, con el objetivo de facilitar la participación de estas.
- Visualización de los horarios del gimnasio y las diferentes clases.
- Solicitar un entrenador personal.

Los problemas que intenta abordar la aplicación respecto a los monitores son:

- Gestión de las clases colectivas, es decir, facilitar la visualización de las clases que tengan asignadas.
- En una misma clase, poder ver la lista de reservados y poder añadir a los socios que han asistido, pero no que hayan reservado.
- Repartir o asignar a monitores a las diferentes clases que tenga el gimnasio. En este caso, el problema a resolver se centra más en los monitores que tengan un rol de líder de equipo, es decir, monitores que tengan otros monitores a su cargo.

Por último, para los entrenadores personales, se centraría en solucionar los problemas de la gestión de los socios y sus entrenamientos.

3.2. Estudio de aplicaciones similares

Los puntos claves de la competencia, centrándonos primeramente en los socios de un gimnasio, serían aplicaciones que generen tablas de ejercicios que satisfagan las necesidades de este. En el caso de los empleados de un gimnasio, serían sistemas que ayudasen en sus tareas diarias, como es la gestión de socios o las reservas. A continuación, se presentan en una tabla algunas aplicaciones que se usan en la actualidad:

Aplicación	Tipo de socio	Servicios
Total Fitness	Usuarios independientes	<ul style="list-style-type: none"> • Más de 100 ejercicios • Descripción y videos de estos • Filtros por musculo • Creación de rutinas • Consejos nutricionales • Cálculo del índice de masa muscular • Retos
Jefit	Usuarios independientes	<ul style="list-style-type: none"> • Más de 1300 ejercicios • Creación de rutinas • Planes por músculo • Registrar cambios • Filtros por músculo
Gym Fitness & Workout: entrenador personal	Usuarios independientes	<ul style="list-style-type: none"> • Más de 300 ejercicios • Registrar progreso • Consejos nutricionales • Rutinas según el nivel
IsMyGym	Gimnasios	<ul style="list-style-type: none"> • Gestión de socios basado en la nube • Gestión de rutinas de entrenamiento • Grupos de comunicación • Gestión de reservas
Virtuagym	Gimnasios, entrenadores	<ul style="list-style-type: none"> • Gestión de socios basado en la nube • Gestión de rutinas de entrenamiento

Aplicación	Tipo de socio	Servicios
	personales, fisioterapeutas e instructores de clases	<ul style="list-style-type: none"> • Grupos de comunicación • Gestión de reservas • Gestión de facturación • Creación de planes de entrenamientos dirigido a los monitores y entrenadores personales • Programación de cursos de formación

Como podemos observar los servicios ofrecidos por cada aplicación para usuarios independientes, son similares en comportamiento y funciones. Lo mismo ocurre con las aplicaciones dirigidas para gimnasios.

3.3. Fortalezas y debilidades principales

Para poder destacar las fortalezas y debilidades para la realización del proyecto, se realizó un análisis DAFO o FODA [26], una metodología de estudio de la situación de una empresa o proyecto, con el objetivo de analizar las características internas y la situación externa:

	Fortalezas	Debilidades
Interno	Fortalezas <ul style="list-style-type: none"> • Formación universitaria • Gran motivación por la temática que ofrece el proyecto • Alto compromiso por el proyecto • Buen ambiente para el desarrollo. 	Debilidades <ul style="list-style-type: none"> • Sin experiencia en la creación de proyectos del sector • No haber un equipo de trabajo • Muchos roles en una sola persona
Externo	Oportunidades <ul style="list-style-type: none"> • Gran cantidad de usuarios interesados en la temática y, por tanto, oportunidad en el mercado. 	Amenazas <ul style="list-style-type: none"> • Gran cantidad de plataformas que ofrecen un servicio similar, es decir, existe mucha competencia en el mercado.

	Fortalezas	Debilidades
	<ul style="list-style-type: none"> Pocos usuarios de gimnasios que usen las aplicaciones de estos. 	<ul style="list-style-type: none"> Presión en la obtención de éxito en el mercado debido a la competencia.

Del análisis, se ha concluido que, aunque es verdad que existen muchas aplicaciones similares, la aplicación a desarrollar no está dirigida a todo el mundo fitness, si no, a las personas que hacen uso de los gimnasios para llevar ese tipo de vida. Por tanto, se puede tener en cuenta el conocimiento que se tiene de aplicaciones propias de los gimnasios que puedan tener interés para el socio y el beneficio que tendría para los gimnasios una aplicación que ayude a aumentar el nivel de satisfacción de los socios con los servicios ofrecidos.

3.4. Solución propuesta

El principal objetivo de MyGymPlanner es ayudar a mejorar la experiencia de las personas que se relacionan con los gimnasios low cost, concretamente: socios y empleados (coordinadores y monitores), proporcionando a cada uno de ellos una serie de servicios, a través de nuestra aplicación, que les facilitará la consecución de sus objetivos y la realización de sus tareas diarias en este tipo de negocios.

Aunque es cierto que existen muchas aplicaciones que dan solución al mundo del fitness, no hay herramientas que hagan posible la coordinación y comunicación entre las personas involucradas: los gimnasios usan unas y los socios otras. Con nuestra propuesta, se pretende dar una solución integral que ayude a optimizar el servicio que ofrecen los gimnasios low cost, mejorando la experiencia de los usuarios de este tipo de negocios y, por tanto, los servicios que ofrecen.

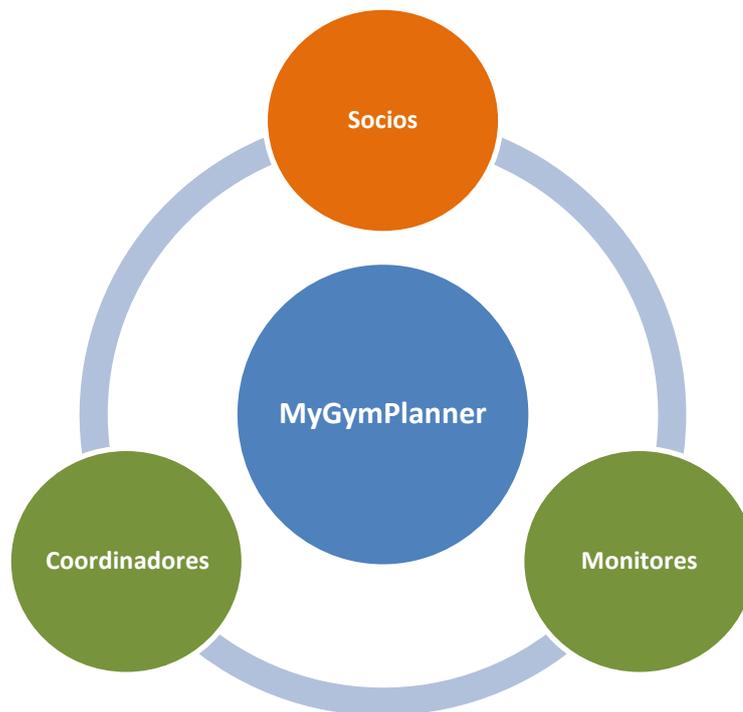


Figura 3

Por tanto, la solución propuesta sería un software fácil de usar, que se adapta a diferentes tipos de dispositivos y que engloba soluciones personalizadas para diferentes tipos de usuarios denominados roles. Estas funcionalidades por roles de usuarios, que podrán evolucionar según sus necesidades, es el principal enfoque que se ha usado para el desarrollo de nuestra aplicación.

3.5. Plan de negocio

En este punto se tratará el cuales serían las opciones para poder financiar el desarrollo del proyecto.

La primera opción que se podría plantear sería pedir un préstamo al banco, ya que, es la opción más fácil y común utilizada. Y presentar el proyecto en alguna plataforma de “crowdfunding” [30].

También se podría considerar la figura del Bussines Angel [31], un inversor que además de aportar financiación propia, aporta su experiencia y contactos. InnoBan [32] es una red de Bussines Angels que facilita la búsqueda de un inversor interesado, adhiriendo el proyecto y ayudando en la presentación.

Otra opción sería solicitar las ayudas y subvenciones que pone en disposición la Administración Pública para financiar negocios. Una ayuda interesante sería la ayuda para jóvenes emprendedores que ofrece ENISA [33]. Si se cumple los requisitos necesarios, la ayuda daría un importe fijo entre 25000 y 75000 € para la adquisición de activos fijos y circulantes necesarios para el desarrollo.

Si ninguna de las opciones tuviera éxito, se empezaría con el FFF's [34], que sería financiar el proyecto con el dinero del círculo cercano, amigos, familia y "locos", y el bootstrapping [35], que es financiar el proyecto con finanzas personales. La mayor desventaja de estos dos métodos es que el capital aportado no sería elevado.

Para financiar la evolución del proyecto, se podría hacer publicidad de productos y servicios a través de nuestro software. Para ello, se podría contactar con empresas de venta de material de gimnasio o empresas con programas fitness que pudieran estar interesadas en la promoción de sus negocios.

Respecto al flujo de ingresos, en primer lugar, se debería realizar un estudio más profundo sobre los ingresos y gastos de un gimnasio. Se podría plantear que el modelo del flujo estuviera basado en una licencia por suscripción. El gimnasio pagaría por una licencia que cubriría funcionalidades específicas para su negocio y se cobraría una tarifa mensual o anual por cada usuario con el rol de socio, lo que le permitiría al gimnasio utilizar el software durante el periodo de suscripción.

El software contaría con un período de prueba gratis para que el gimnasio pueda usar la aplicación, tener el feedback de los usuarios y luego decidir si desean pagar por la suscripción.

4. Modelado de negocio

En el apartado actual se realizará un modelado de negocio para describir el funcionamiento interno de un gimnasio. Primeramente, se explicará de qué trata el modelado de negocio y la importancia de realizar uno.

El modelado de negocio es un proceso de representación gráfica de uno o más aspectos del negocio con el fin de conocer todos los elementos que la componen, su interacción, recursos, metas, procesos, la comunicación y relaciones que existen.

La importancia de realizar este proceso se basa en que se llega a conocer el ámbito de información, así como las actividades y procesos dentro del negocio, con el objetivo de tener una buena comprensión de este a la hora de crear soluciones softwares que se ajusten a medida a una organización.

Ahora que se tiene una pincelada sobre lo que es y la importancia del modelado de negocio, se continuará describiendo los casos de uso de los actores del negocio que se llegaron a identificar. Un actor del negocio es un individuo, grupo, organización o máquina que interactúa con la organización. Un trabajador del negocio representa un rol que juega una persona dentro del negocio. Un trabajador representa un rol, no una posición, es decir, una persona puede tener varios roles dentro de una organización.

Una vez identificados los principales casos de uso del negocio, se describirá los procesos de negocio relacionados con estos casos de usos.

4.1. Casos de uso de negocio

Un caso de uso de negocio describe una secuencia de acciones realizadas en el negocio que produce un resultado de valor observable para un actor individual del negocio.

Los casos de usos principales son:

1. Registrar socio. En este caso de uso identificamos al actor socio, individuo que solicita unirse al gimnasio, y al trabajador administrador, que se encargaría de registrarlo.

2. Monitorear sala. Se identifica al actor socio, individuo o grupo de individuos que hacen uso de una sala dentro del gimnasio y al trabajador monitor, persona que supervisa la sala.
3. Asesorar socio. En este caso de uso, podemos identificar al actor socio, que es el individuo que solicita el servicio de asesoramiento personal, y al trabajador asesor, que es la persona que se encarga de asesorarlo.
4. Dar clase colectiva. Se identifica al actor socio, individuo o grupo de individuos, que asisten a una clase colectiva, y al trabajador monitor, persona que se encarga de dar la clase.
5. Asignar clases. En este caso de uso solo identificamos a trabajadores; el monitor, persona o grupo de personas al que se le asigna las clases y el coordinador, persona que se encarga de asignar las clases.

Por supuesto, hay más casos de usos, como sería “Cobrar cuota”, pero se decidió solo centrarse en aquellos en los que el proyecto trata de automatizar.

4.2. Procesos de negocio

Para la realización de este punto, se estarán realizando diagramas de actividades para la representación gráfica de los procesos. Los diagramas de actividades representan lo que ocurre en los procesos de negocios, simbolizadas por los trabajadores y actividades que ellos realizan.

1. El primer proceso de negocio que se tratará será el que está relacionado con el caso de uso “Registrar socio”, el cuál es, “Registrar el socio”. En este caso, simplemente se listará las actividades, ya que, es un proceso que solo relaciona al administrador y al socio:
 - El socio solicita apuntarse en el gimnasio con un plan de pago. El administrador solicita los datos para poder asociarlo. El socio paga la cuota establecida y la matricula. El administrador le da la factura, numero de socio y le explica la forma de acceso.

2. El siguiente proceso, relacionado con el caso de uso “Monitorear sala”, sería “Monitorear la sala”. Al igual que el anterior, solo se listará las actividades, por la simplicidad del proceso:

- El monitor camina por la sala, observando como los socios hacen uso de ella. Si un socio necesita o solicita ayuda, el monitor se la ofrece. Una vez la ayuda haya sido atendida, el monitor sigue caminando por la sala.

3. El caso de uso “Asesorar socio”, está relacionado con dos procesos:

- “Atender solicitud de asesoramiento”:

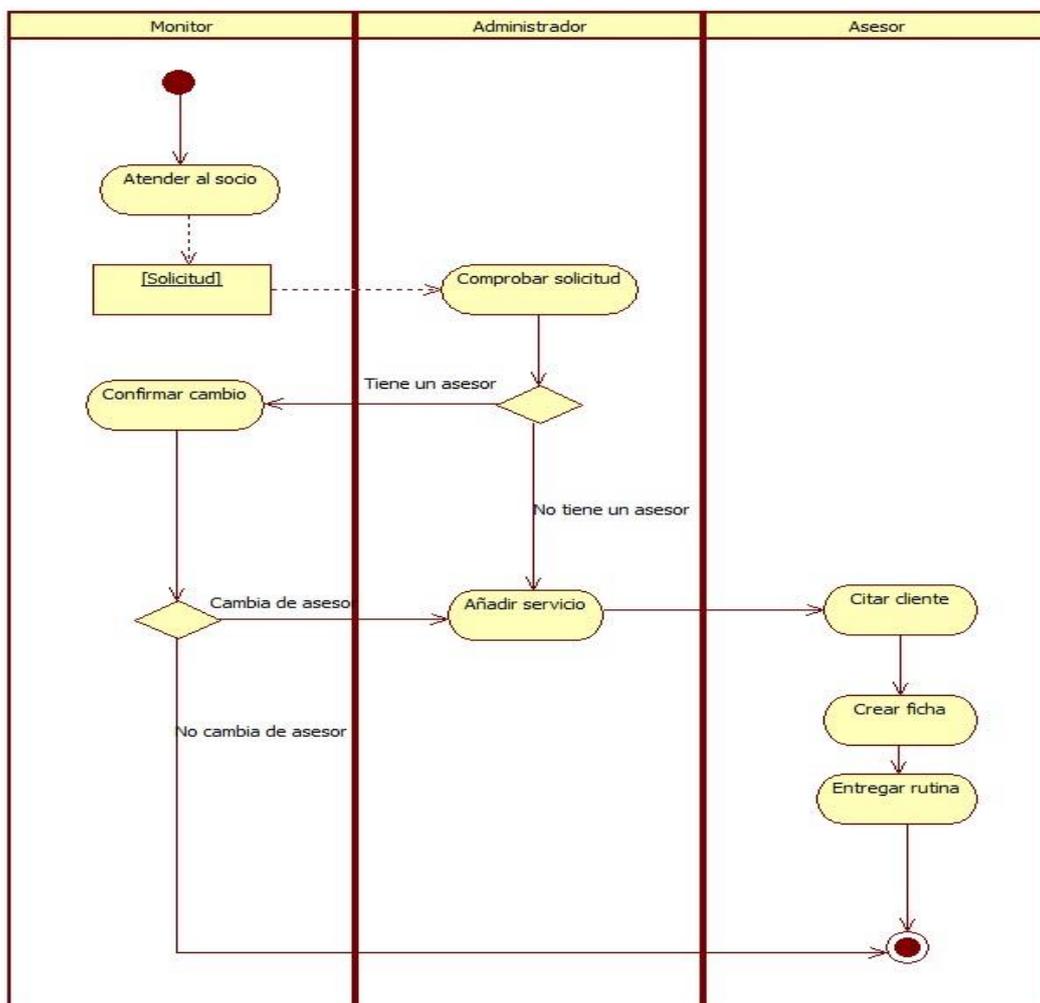


Figura 4

- Y “Asesorar el socio”. El asesor cita al socio en periodos de tiempos acordados con el socio para hacerle un seguimiento.
4. Con el caso de uso “Dar clase colectiva”, también está relacionado con dos procesos de negocio:
- “Atender solicitud de reserva”: El socio solicita poder participar a una clase. El administrador comprueba el aforo de la clase, en el caso de que este completa, deniega la solicitud, caso contrario, apunta al socio a la clase.
 - “Dar la clase colectiva”:

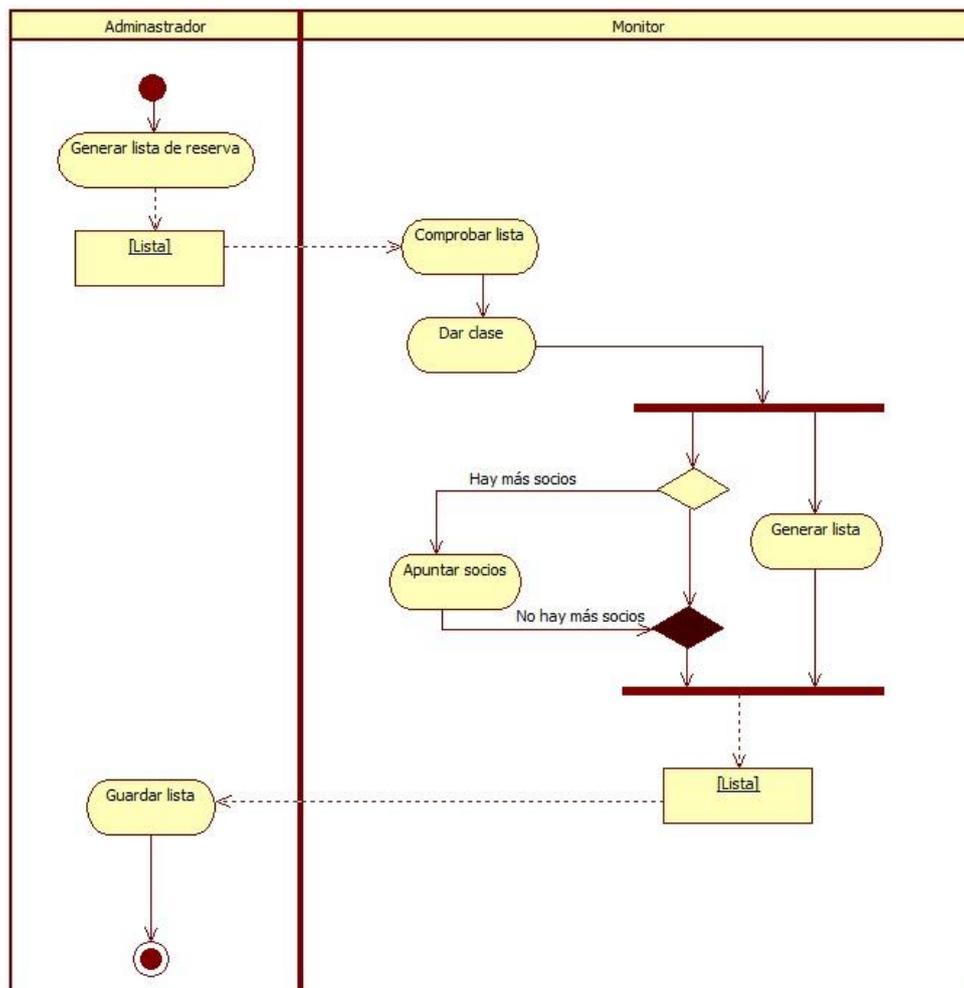


Figura 5

5. Por último, el caso de uso “Asignar clases” está relacionado con el proceso de negocio “Asignar clases colectivas”. En este proceso de negocio, solo participa el coordinador:

- El coordinador comprueba el horario de las clases de la semana. El coordinador comprueba los horarios de los monitores disponibles. El coordinador asigna a cada monitor las clases según los horarios y especialidades de cada uno.

5. Modelado del sistema

5.1. Requisitos de usuarios

Para identificar los requisitos de usuario, se realizaron entrevistas con los diferentes usuarios del gimnasio. Para conocer las necesidades de los socios, se hicieron entrevistas a personas de entre 20 y 37 años, que habían estado en uno o más gimnasios. Y para conocer las necesidades de los monitores, se eligió una muestra de monitores de las cadenas de gimnasio AltaFit y MacroFit.

5.1.1. Necesidades de los socios

Antes de entrar en los problemas que intentaba abordar el proyecto, se preguntó por las carencias del gimnasio o necesidades según su punto de vista, y las más destacables fueron las siguientes:

- Más rutinas de entrenamiento con diferentes niveles y con consejos para poder cambiar ejercicios en la rutina, ya sea, para que este más dirigida a su nivel físico o porque ese día quieren realizar otro ejercicio en el que se trabajen los mismos músculos.
- Explicaciones de cómo realizar los ejercicios correctamente para evitar lesiones y saber qué músculos se están usando.
- Alguna forma de controlar el progreso.
- Que el gimnasio tuviese algún apartado para consejos de nutrición.
- Que avisaran de los pagos.
- Poder reservar las clases colectivas sin necesidad de estar en el gimnasio. En este caso, dentro de los gimnasios hay un dispositivo para poder reservar o los usuarios van directamente.

Los usuarios hacían usos de aplicaciones externas para cubrir las necesidades que fueron descritas, como, por ejemplo, TotalFitness para la realización de las rutinas de entrenamiento más personalizadas, ya que, no tenían conocimiento de si el gimnasio contaba con una aplicación que resolviese las necesidades expuestas.

Una vez conocidas las necesidades de los socios, se planteó los problemas que intentaría resolver el proyecto y de cómo se solucionarían desde la aplicación para el rol del socio siendo confirmadas por los mismos.

5.1.2. Necesidades de los monitores

En el caso de los monitores, al ser empleados y entender mejor el funcionamiento de los gimnasios, se aprovechó para analizar cómo se trabaja en los gimnasios low cost y poder conocer mejor el negocio.

Se les planteó los problemas y necesidades que intentaba solucionar la aplicación a desarrollar, no solo la parte que les afectaba directamente sino también la de los socios. Los problemas planteados inicialmente por estos usuarios se centraban en las clases colectivas, pero al realizar las entrevistas se fueron descubriendo más necesidades:

- Primeramente, se plantearon los problemas y necesidades que tenían los socios y fueron confirmados.
- A continuación, se plantearon los problemas que afectaban a los monitores y fueron comentando nuevas necesidades:
 - Respecto a los horarios de las clases, no veían tanto la necesidad de saber qué clases tenían asignadas, sino, poder ver y resaltar las propias y poder ver las de sus compañeros para conocer su ubicación o por si tenían que sustituirlos.
 - Sobre la lista de reservados, se comentó que era necesario saber quién se apuntaba a cada clase y que se pudiera añadir a aquellos socios que no hubieran reservado (con su nombre, número de socio, DNI o teléfono).
- Se le plantearon los problemas del rol de líder de equipo. Al parecer este rol coincide con el de coordinador o encargado, por tanto, se optó por cambiar a rol de coordinador:

- Al plantear el problema de asignación de clases, se confirmó que esta necesidad era muy importante, ya que, era algo que se hacía de manera semanal o quincenal y, en la mayoría de los casos, no se usaba una aplicación específica que facilitara el trabajo, se hacía uso de un Excel para la asignación, consultando los horarios de los monitores y las clases de gimnasio en la base de datos interna.
- Por último, los monitores, además de sus labores en el gimnasio, suelen dar un servicio de asesoramiento personalizado. Para ello, charlan con los socios interesados para saber sus objetivos y así poder realizarles tablas, hacerle una ficha para medir su progreso, apuntar sus medidas, etc. Para realizar este trabajo suelen usar dos de las aplicaciones de la suite de Microsoft Office: Word y Excel.

A continuación, y para finalizar este apartado, se deberían haber descrito el rol de entrenador personal, pero se decidió descartar este rol del proyecto por dos motivos:

- Por la similitud con el servicio de asesoramiento personal.
- Porque los entrenadores personales, según los monitores, no son empleados del gimnasio, pueden usar las instalaciones de estos, pero son ajenos al gimnasio.

5.2. Requisitos del sistema

La solución que se propone resolverá los problemas que se describieron en apartados anteriores. Como se comentó anteriormente, el proyecto consistirá en un software que aportará a los usuarios de los gimnasios una sensación de control de su actividad diaria, mejorando su experiencia personal.

La idea sería un software base que se pueda adaptar a las necesidades de los gimnasios, ya que no todos los gimnasios ofrecen los mismos servicios o no los gestionan de la misma manera.

5.2.1. Requisitos funcionales

En el siguiente punto se detallan los principales requisitos funcionales que tendrá la aplicación a desarrollar. Como se comentó en apartados anteriores, el proyecto tendrá un sistema de roles, permitiendo a los usuarios iniciar y cerrar sesión con su rol correspondiente. A continuación, se detallan los requisitos clasificados por tipo de rol.

Rol de administrador	
RF_AD001	El sistema permitirá al usuario administrador añadir y eliminar usuarios de la aplicación.
RF_AD002	El sistema permitirá al usuario administrador dar roles a los usuarios.
RF_AD003	El sistema permitirá al usuario administrador ver todos los usuarios.
RF_AD004	El sistema deberá permitir al usuario administrador filtrar los usuarios por roles.
RF_AD005	El sistema deberá permitir al usuario administrador ver el horario del gimnasio y sus clases.
RF_AD006	El sistema permitirá al usuario administrador asignar usuarios coordinadores a los diferentes horarios.
RF_AD007	El sistema permitirá al usuario administrador ver las solicitudes de asesoramiento personal.
RF_AD008	El sistema permitirá al usuario administrador asignar socios a los usuarios monitores.
RF_AD009	El sistema permitirá al usuario administrador crear equipos de trabajo y asignar coordinador en ellos.

Rol de monitor	
RF_MO001	El sistema permitirá al usuario coordinador ver a los usuarios monitores asignados en su equipo.
RF_MO002	El sistema mostrará al usuario coordinador el horario y clases que tenga asignado para el equipo.
RF_MO003	El sistema permitirá al usuario coordinador repartir el horario y clases que tenga asignado con el equipo.
RF_MO004	El sistema permitirá al usuario monitor ver los horarios de las clases del gimnasio.
RF_MO005	El sistema permitirá al usuario monitor ver el horario que tiene asignado y clases que tendrá que monitorizar.
RF_MO006	El sistema permitirá al usuario monitor ver la lista de usuarios socios de una clase que tenga asignada.
RF_MO007	El sistema permitirá al usuario monitor añadir usuarios socios a lista de

Rol de monitor	
	una clase.
RF_MO008	El sistema permitirá al usuario monitor ver los socios que tenga asignados.
RF_MO009	El sistema permitirá al usuario monitor crear fichas a usuarios socios.
RF_MO010	El sistema permitirá al usuario monitor crear rutinas a los usuarios socios.
RF_MO011	El sistema permitirá al usuario monitor modificar rutinas a los usuarios socios.
RF_MO012	El sistema permitirá al usuario monitor eliminar rutinas a los usuarios socios.

Rol de socio	
RF_CL001	El sistema permitirá al usuario socio ver las clases del gimnasio y sus horarios.
RF_CL002	El sistema permitirá al usuario socio reservar alguna clase.
RF_CL003	El sistema permitirá al usuario socio agregar sus datos de progreso.
RF_CL004	El sistema permitirá al usuario socio ver las rutinas de ejercicios del gimnasio.
RF_CL005	El sistema permitirá al usuario socio elegir y guardar rutinas de ejercicios.
RF_CL006	El sistema permitirá al usuario socio modificar los ejercicios de una rutina guardada.
RF_CL007	El sistema permitirá al usuario socio eliminar rutinas guardadas.
RF_CL008	El sistema permitirá al usuario socio crear rutinas de ejercicios y guardarlas.
RF_CL009	El sistema permitirá al usuario socio solicitar el servicio de asesoramiento personal.

5.2.2. Requisitos no funcionales

A continuación, se muestran algunos requisitos no funcionales necesarios para la aplicación a desarrollar.

Requisitos de fiabilidad	
RNF001	El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente accederlo.
RNF002	El promedio de duración de fallas no podrá ser mayor a 15 minutos.

Requisitos de usabilidad	
RNF003	El tiempo de aprendizaje del sistema deberá ser menor a 1 hora.
RNF004	El sistema debe poseer interfaces gráficas intuitivas para el usuario final.
RNF005	El sistema debe proporcionar mensajes de error que sean informativos y orientados al usuario final.

Requisitos de eficiencia	
RNF006	Toda funcionalidad del sistema debe responder al usuario en menos de 5 segundos.
RNF007	El sistema debe ser capaz de operar adecuadamente con hasta 100.000 usuarios con sesiones concurrentes.
RNF008	Los datos modificados en la base de datos deben ser actualizados para todos los usuarios que acceden en menos de 2 segundos.
RNF009	El sistema debe permitir una navegación fluida por todos los componentes de la aplicación.

Requisitos de portabilidad	
RNF010	El sistema deberá desarrollarse tanto para dispositivos Android como para escritorio.
RNF011	La aplicación debe ser compatible con todas las versiones de Android, desde la versión 5.0 (API 22, Lollipop).
RNF012	La aplicación debe ser compatible con todas las versiones de Windows, desde Windows 95.

Requisitos de mantenibilidad	
RNF013	El código fuente que se implemente en JAVA deberá cumplir las recomendaciones de Clean Code

Requisitos de seguridad	
RNF014	El sistema debe asegurar que los datos estén protegidos del acceso no autorizado.
RNF015	La aplicación deberá instalarse antes de usarse.
RNF016	El sistema necesitará acceso a internet para poder hacer uso de ella.

5.3. Normativa y legislación

Entre la normativa y legislación vinculada al uso de las aplicaciones informáticas cabe citar la Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos de Personales y garantía de los derechos digitales.

La citada Ley fue aprobada con el fin de proteger y garantizar el tratamiento de datos personales, las libertades públicas y los derechos fundamentales de las personas; especialmente los que atañen a la intimidad personal y familiar de los ciudadanos.

Esta ley supone nuevas obligaciones para las empresas que deben actuar correctamente frente a los afectados, personas titulares de los datos.

El objeto de la Ley es garantizar y proteger el tratamiento y almacenamiento de datos informatizados por parte de terceros, así como las libertades y derechos fundamentales de las personas físicas, especialmente su honor e intimidad. Su aplicación es a todas las empresas, tanto ficheros privados como públicos.

5.4. Diagramas de casos de uso

Para representar la funcionalidad del sistema se ha usado UML [27]. A continuación, se muestran los diagramas de casos de uso de la aplicación, así como las especificaciones de los casos de uso principales para los diferentes actores del sistema:

- Administrador
- Socio
- Monitor
- Coordinador
- Asesor personal

En primer lugar, se representan los casos de uso del Administrador.

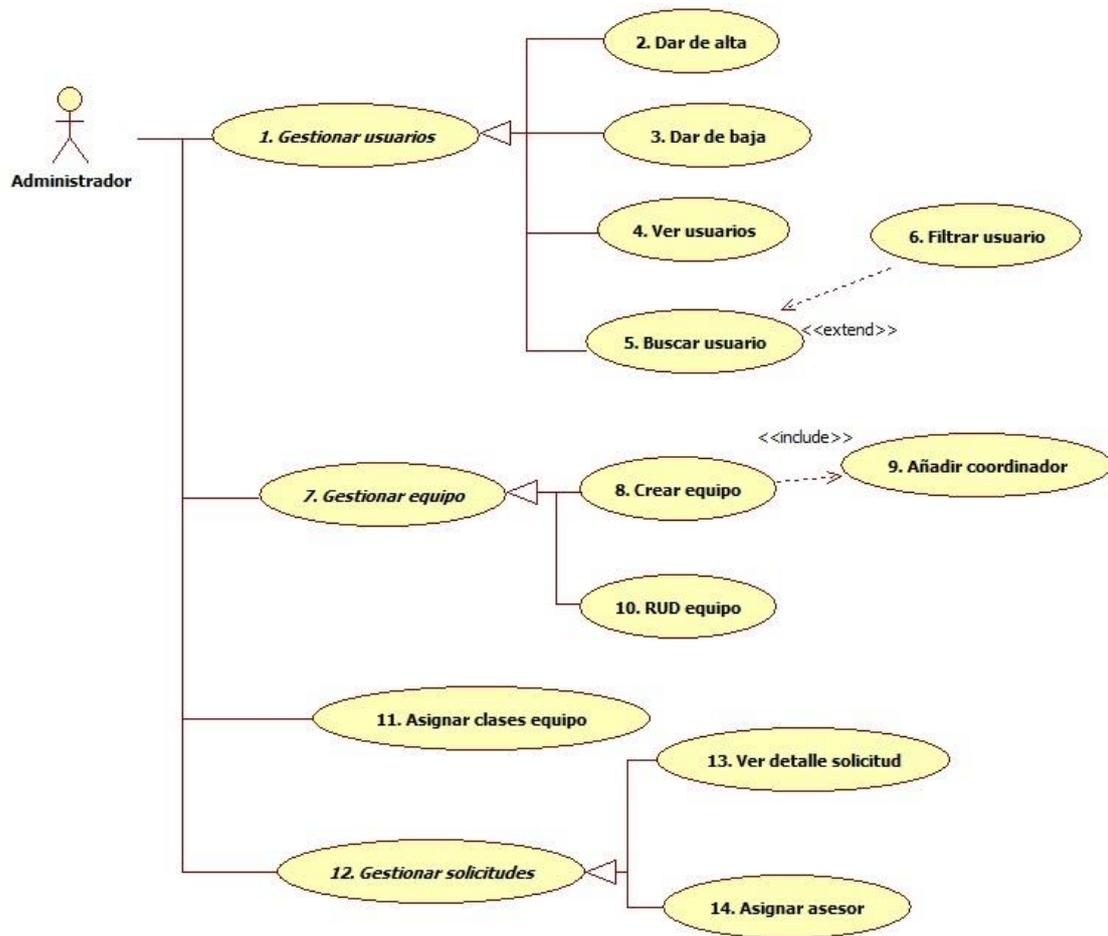


Figura 6

CASO DE USO	CU02	Dar de alta	
Descripción	El administrador da de alta a un usuario		
Actores	Administrador		
Flujo normal	Paso	Acción	
	1	El sistema muestra los campos a rellenar	
	2	El administrador añadirá los datos del usuario	
Postcondiciones	El usuario se encuentra creado		
Excepciones			

CASO DE USO	CU08	Crear equipo	
Descripción	El administrador crea un equipo		
Actores	Administrador		
Flujo normal	Paso	Acción	
	1	El administrador añade un nombre al equipo	
	2	El sistema muestra los monitores disponibles	
	3	El administrador selecciona los monitores que estarán en el equipo	
	4	El administrador selecciona al monitor que será coordinador (CU: Asignar coordinador)	
	5	El administrador crea el equipo	
Postcondiciones	El equipo se encuentra creado		
Excepciones			

CASO DE USO	CU11	Asignar clases equipo	
Descripción	El administrador asigna un tipo de clases a un equipo		
Actores	Administrador		
Flujo normal	Paso	Acción	
	1	El sistema muestra la lista de clases no asignadas	
	2	El administrador selecciona una de las clases	
	3	El sistema muestra los equipos del gimnasio y las clases que tienen asignadas	
	4	El administrador asigna la clase a un equipo	
Postcondiciones	El equipo es asignado		
Excepciones			

A continuación, se presentan los casos de uso del socio:

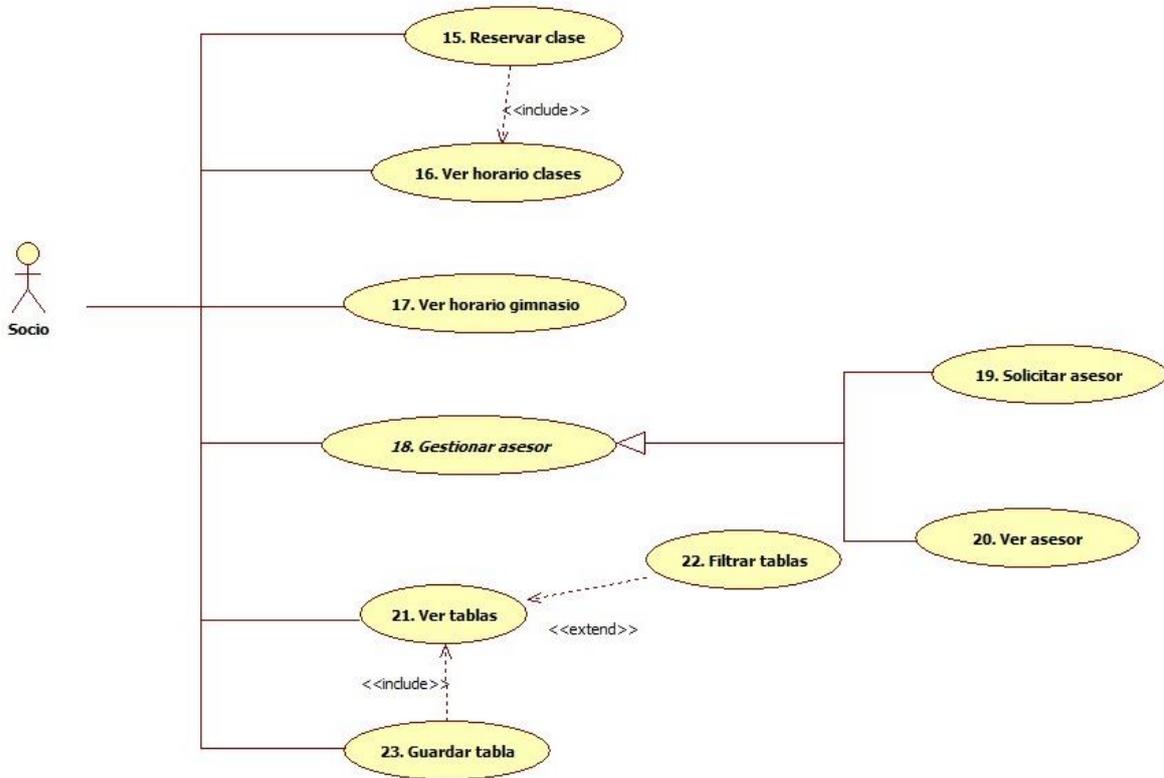


Figura 7

CASO DE USO	CU15	Reservar clase	
Descripción	El socio reserva una clase colectiva		
Actores	Socio		
Flujo normal	Paso	Acción	
	1	El sistema muestra los horarios de las clases (CU16: Ver horario clases)	
	2	El socio selecciona y reserva una clase	
Postcondiciones	El socio se encuentra en lista de reserva		
Excepciones			

CASO DE USO	CU23	Guardar tabla	
Descripción	El socio guarda una tabla de entrenamiento		
Actores	Socio		
Flujo normal	Paso	Acción	
	1	El sistema muestra las tablas de entrenamiento (CU21: Ver tablas)	
	2	El socio selecciona y guarda una tabla	
Postcondiciones	La tabla se encuentra en las tablas del socio		
Excepciones			

A continuación, los casos de uso del monitor. En este caso hay que tener en cuenta que tanto el coordinador como el asesor personal, son también monitores.

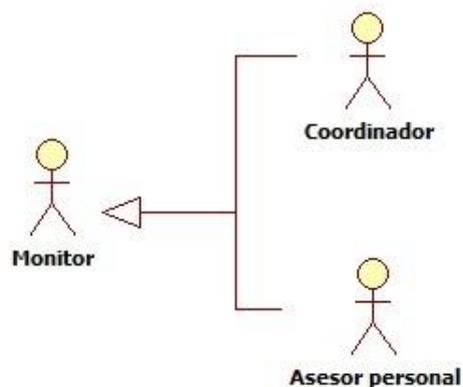


Figura 8

Tanto el coordinador como el asesor personal podrán tener acceso a los casos de uso del monitor.

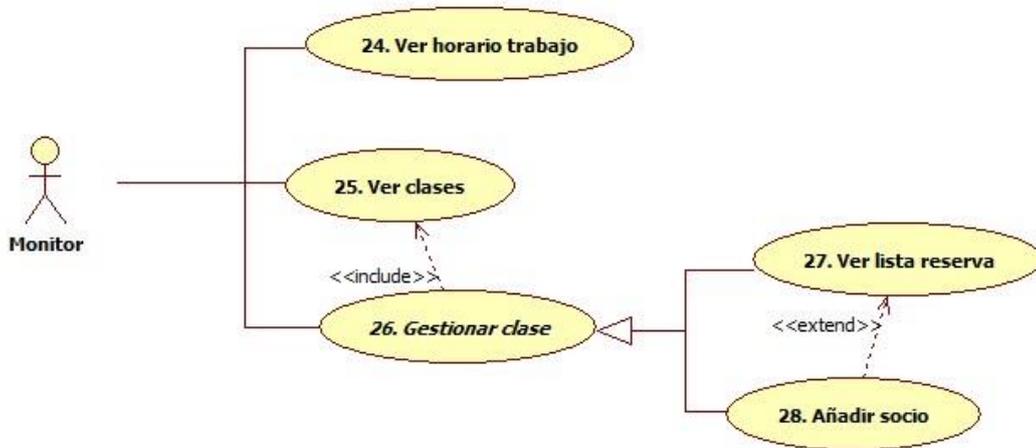


Figura 9

CASO DE USO	CU27	Ver lista de reserva	
Descripción	El monitor ve la lista de reserva de una clase		
Actores	Monitor		
Flujo normal	Paso	Acción	
	1	El sistema muestra las clases colectivas (CU25: Ver clases)	
	2	El monitor ve la lista de reserva de una clase	
Postcondiciones	El socio		
Extensiones	Paso	Condición	Caso de uso
	2	El monitor añade un socio que no esté en la lista	CU28: Añadir socio
Excepciones			

CASO DE USO	CU28	Añadir socio	
Descripción	El monitor añade un socio a la lista de la clase		
Actores	Socio		
Flujo normal	Paso	Acción	
	1	El sistema muestra los campos por los cuales buscar	
	2	El monitor busca a un socio por uno de los campos	
	3	El sistema muestra a los socios con ese nombre	
	4	El monitor selecciona y añade al socio	
Postcondiciones	El socio se encuentra en la lista de la clase		
Variaciones	Paso	Acción	
	2A	El monitor busca por otro campo	
Excepciones	4	El socio ya se encuentra en la lista	

A continuación, se muestran los casos de uso a los que solo podrá acceder el coordinador.

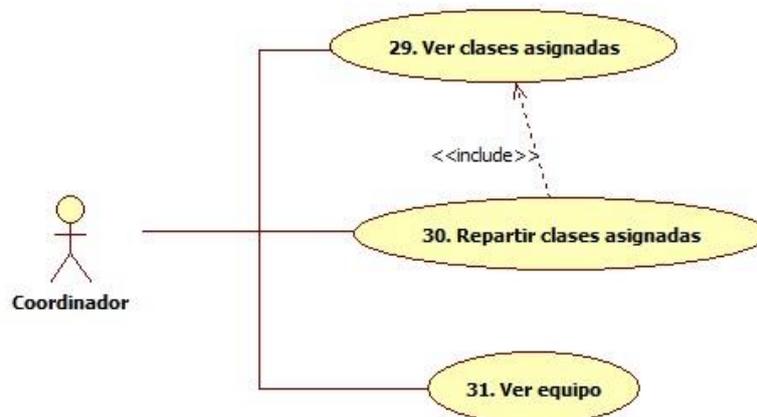


Figura 10

CASO DE USO	CU30	Repartir clases asignadas	
Descripción	El coordinador reparte las clases colectivas a los miembros del equipo		
Actores	Coordinador		
Flujo normal	Paso	Acción	
	1	El sistema muestra las clases colectivas asignadas (CU29: Ver clases asignadas)	
	2	El coordinador asigna las clases a los miembros disponibles	
Postcondiciones	Las clases se encuentran asignadas		
Excepciones			

A continuación, se muestran los casos de uso exclusivos para el asesor personal.

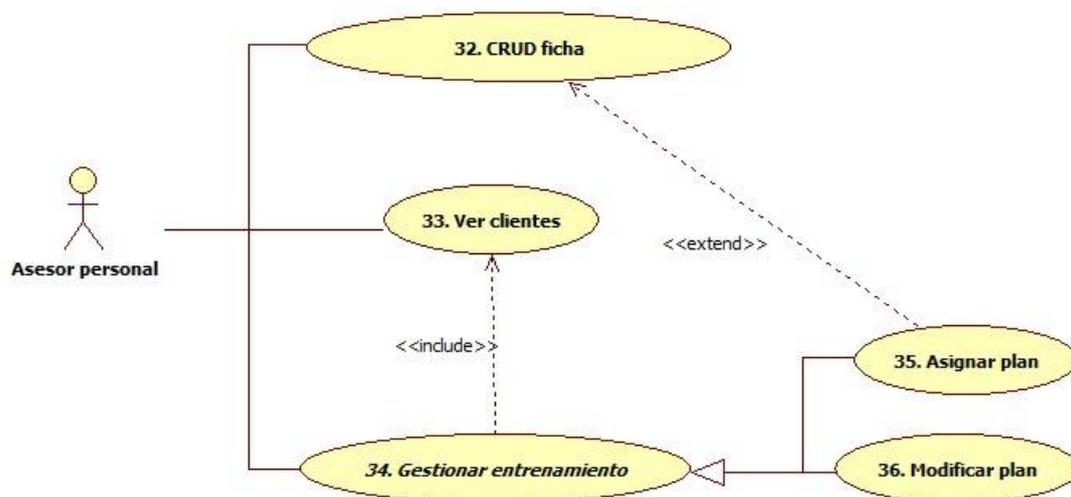


Figura 11

CASO DE USO	CU32.C	Crear ficha	
Descripción	El asesor personal crea una ficha a un cliente del servicio		
Actores	Asesor personal		
Flujo normal	Paso	Acción	
	1	El sistema muestra los campos a rellenar	
	2	El asesor asigna a un cliente y añade los datos faltantes	
	3	El asesor guarda la ficha	
Postcondiciones	La ficha queda creada		
Extensiones	Paso	Condición	Caso de uso
	2	El asesor asigna un plan de entrenamiento al cliente	CU35: Asignar plan
Excepciones			

CASO DE USO	CU35	Asignar plan	
Descripción	El asesor asigna un plan de entrenamiento a un socio		
Actores	Asesor		
Flujo normal	Paso	Acción	
	1	El sistema muestra los objetivos disponibles	
	2	El asesor selecciona un objetivo	
	3	El sistema muestra las rutinas y días en ese objetivo	
	4	El asesor selecciona las rutinas y los días	
	5	El asesor guarda el plan	
Postcondiciones	El plan se adhiere a la ficha del socio		
Excepciones			

5.5. Diagramas de clases

Para representar la estructura de la aplicación se decidió hacer uso de los diagramas de clases, que representen las relaciones entre los diferentes objetos y sistemas del negocio.

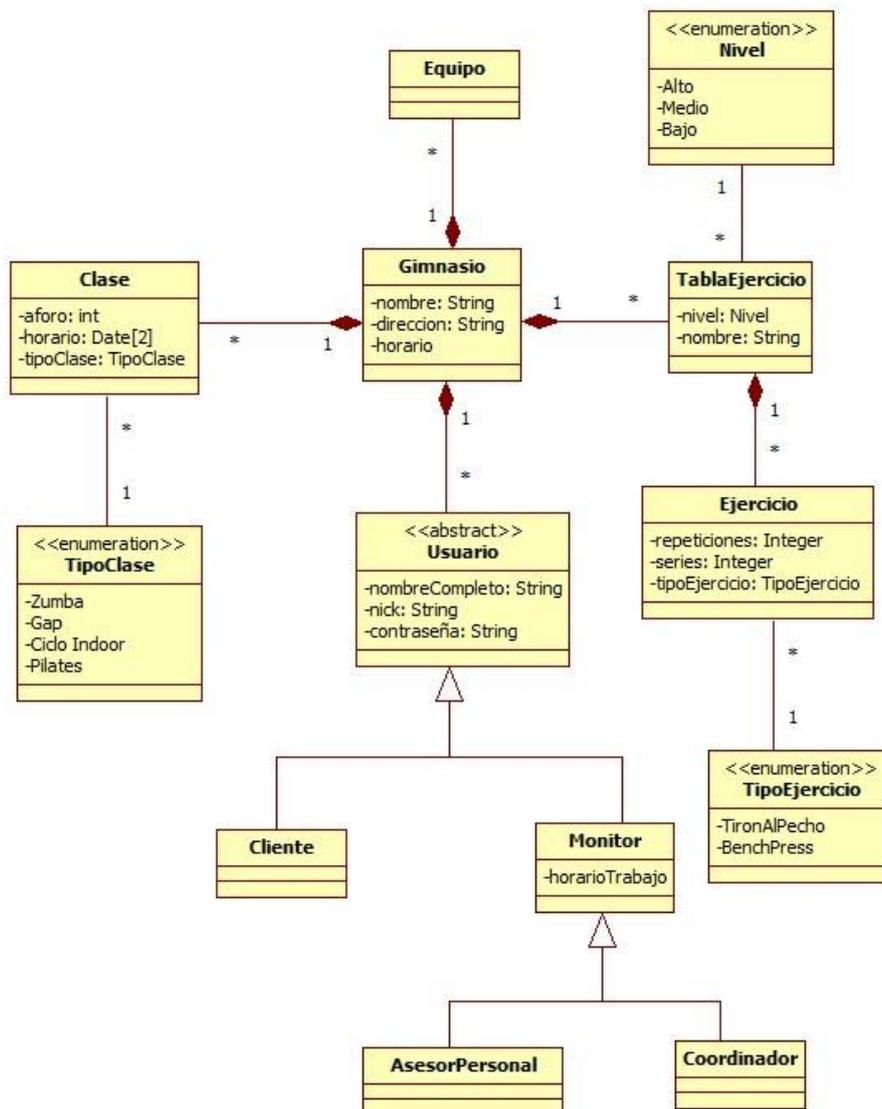


Figura 12

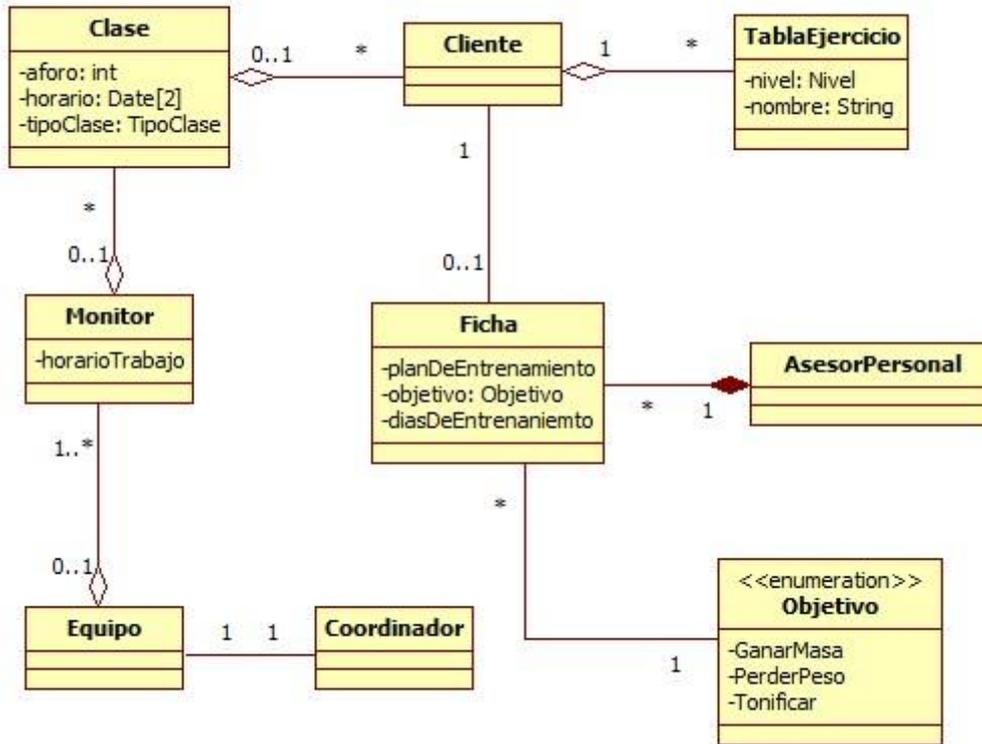


Figura 13

5.6. Mockups

Los mockups que se realizaron se hicieron con vistas al prototipo. Esta separado en los mockups de la aplicación web y móvil. Aunque la aplicación es *responsive*, se decidió mostrar los mockups como lo vería mayoritariamente cada rol.

5.6.1. Mockups de la aplicación web

Primeramente, la pantalla de inicio de sesión.

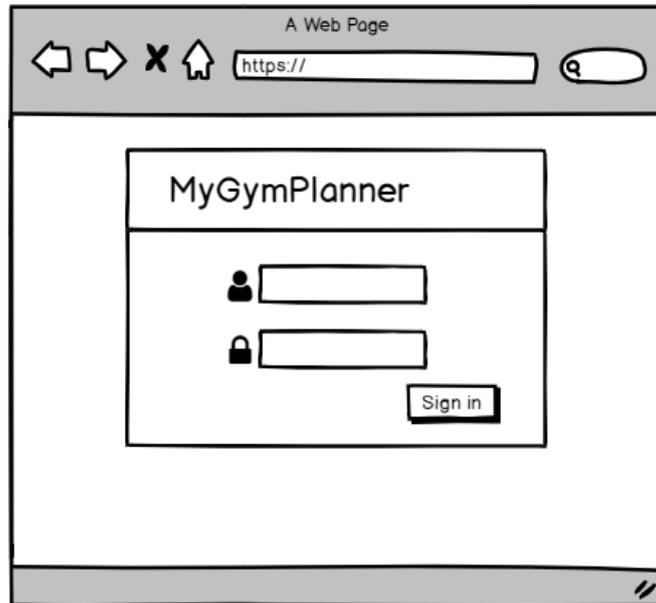


Figura 14

Se empezó con el rol de administrador:

- La página principal, donde se vería la lista de usuarios de la aplicación:

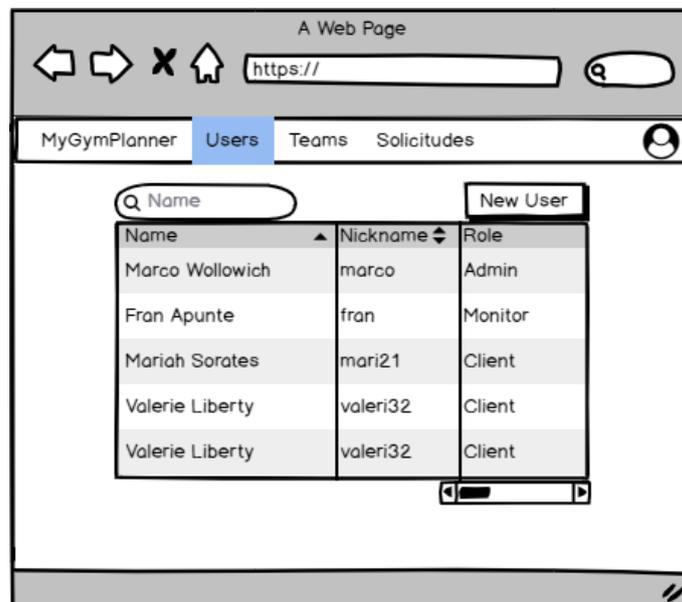


Figura 15

- La página para dar de alta a un usuario:

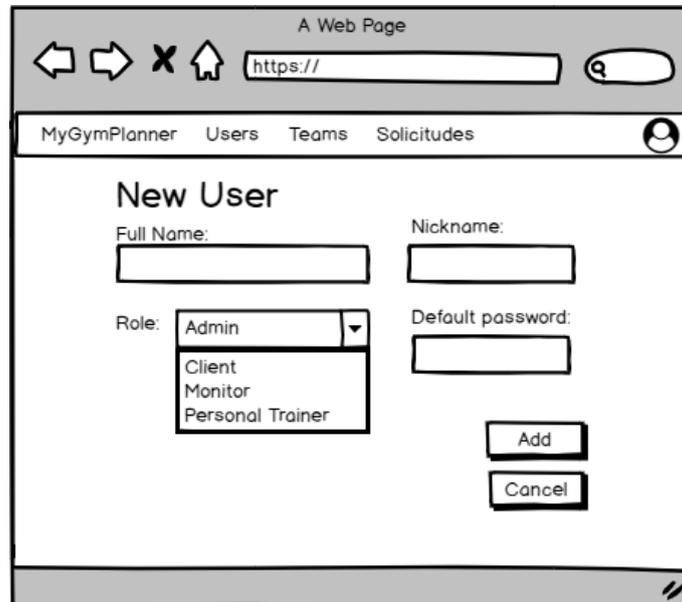


Figura 16

- La página de los equipos:

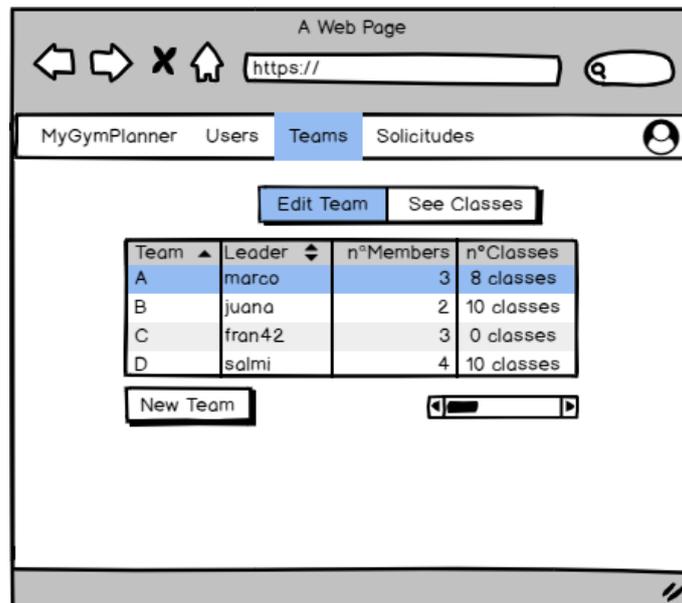


Figura 17

- La página para crear o editar un equipo:

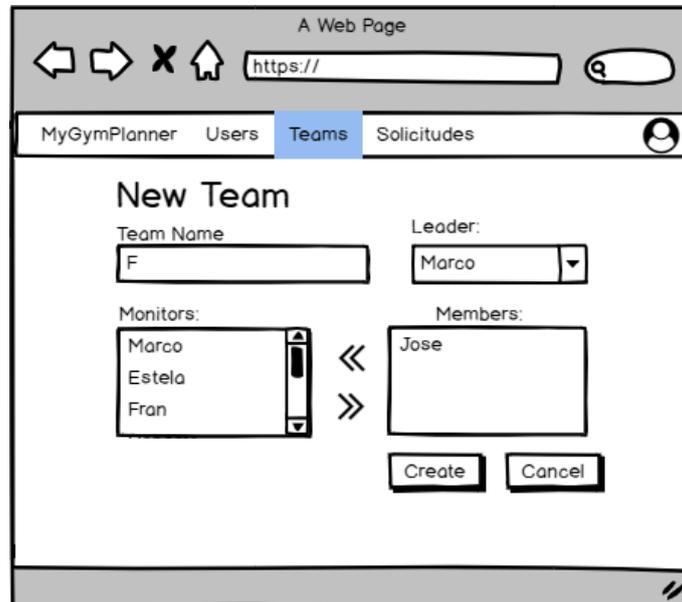


Figura 18

- La página de ver una clase:

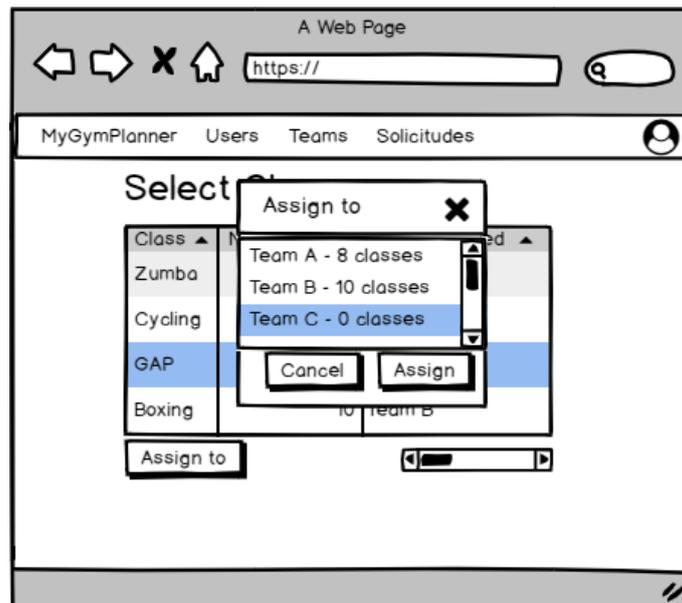


Figura 19

- La página de asignar una clase:

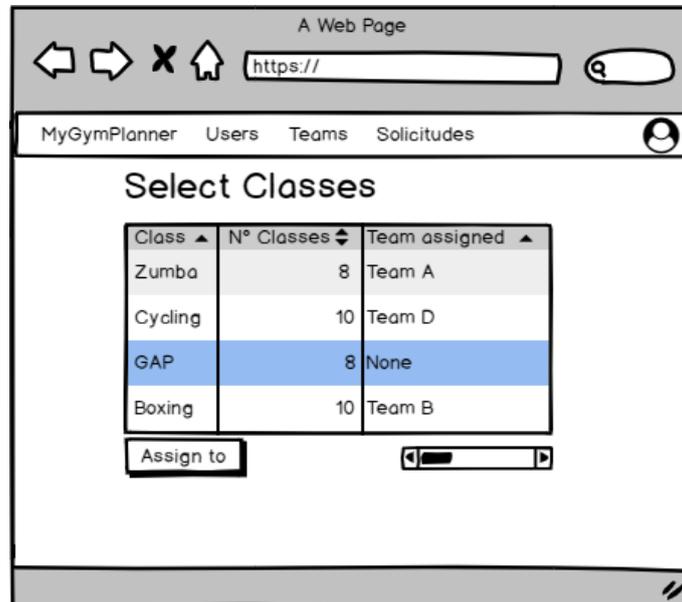


Figura 20

5.6.2. Mockups de la aplicación móvil

Los siguientes mockups representan al rol de socio. La idea es que el socio pueda usar la aplicación en su dispositivo móvil:

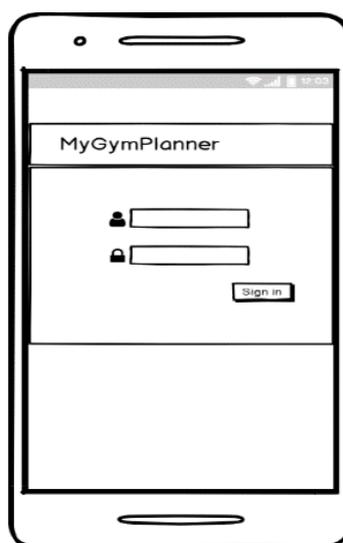


Figura 21

- La página de inicio y menú:

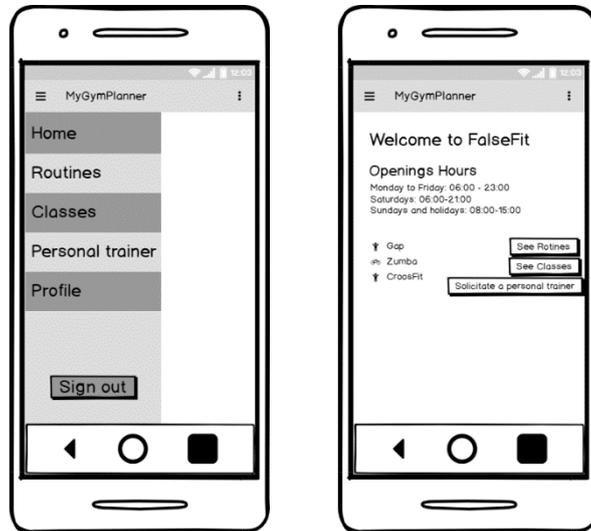


Figura 22

- La página de rutinas

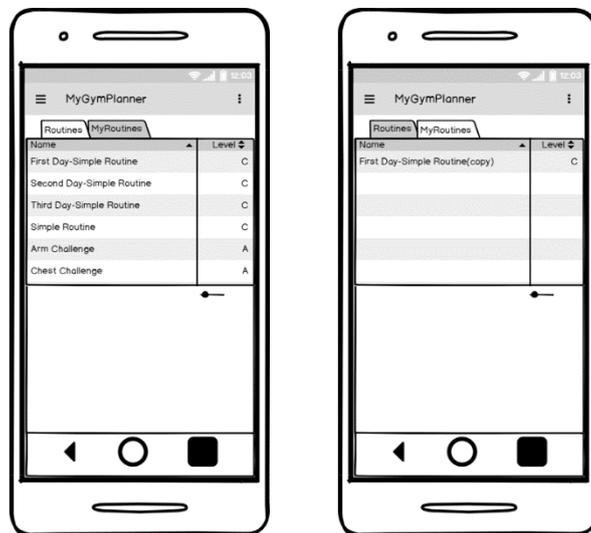


Figura 23

- Una rutina individual y un ejercicio

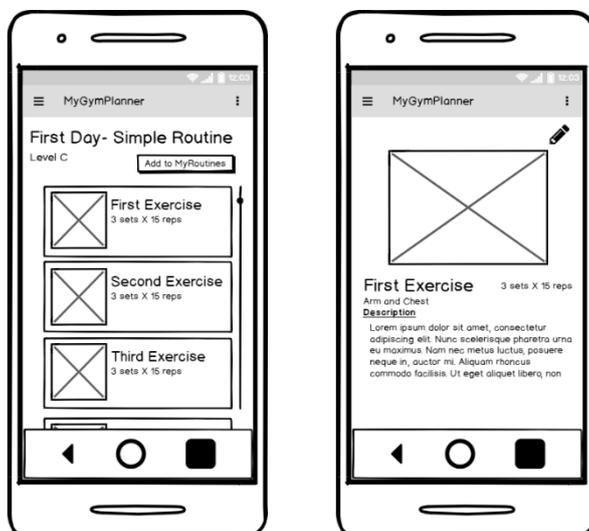


Figura 24

- Las clases colectivas

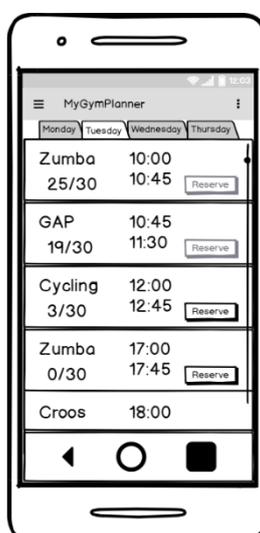


Figura 25

Y por último el rol del monitor:

- El menú y la pantalla principal

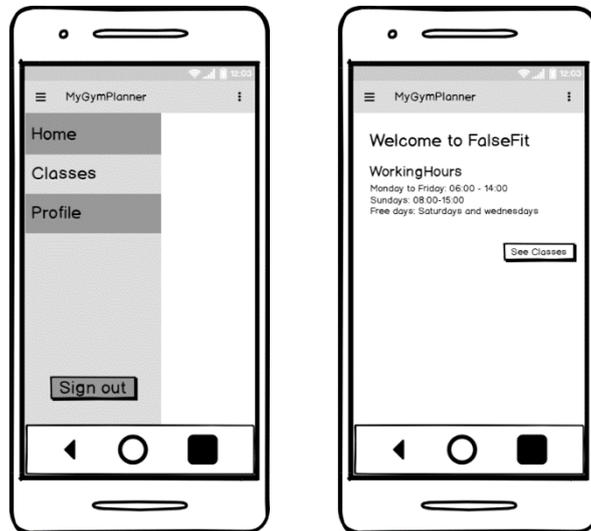


Figura 26

- Las siguientes pantallas representan las pantallas de las clases colectivas:

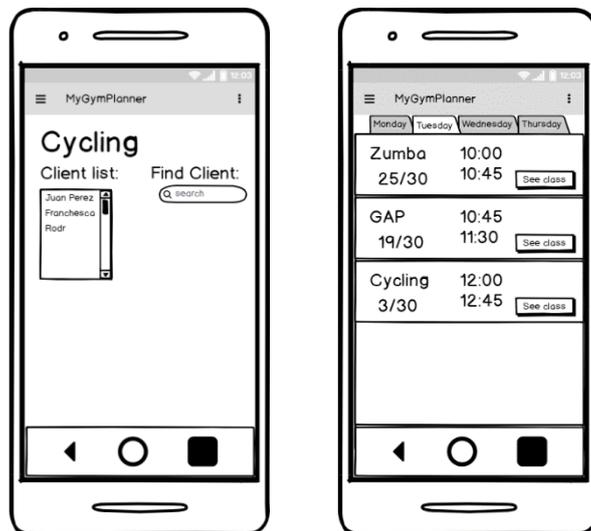


Figura 27

A continuación, las pantallas de un coordinador de equipo, que sería similar a las del monitor, por tanto solo se muestra las diferencias que sería la asignación de un monitor a una clase:

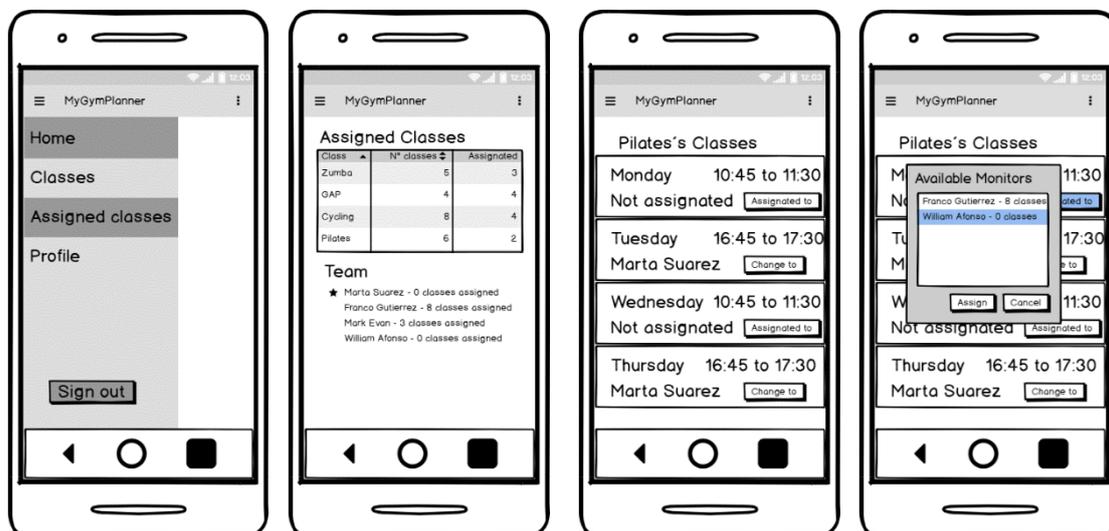


Figura 28

5.7. Arquitectura del prototipo

Para el desarrollo de la aplicación se decidió usar el estilo arquitectónico cliente-servidor [29]. La arquitectura tiene dos módulos principales que son independientes: el frontend que representa el módulo del cliente y el módulo backend que representa el módulo del servidor.

5.7.1. Diseño arquitectónico

El módulo backend se divide a su vez en varios módulos para facilitar el entendimiento de la arquitectura interna de la aplicación y evitar acoplamiento al tener módulos con responsabilidades únicas:

- Controller. En este módulo se encuentra las clases que representan las operaciones HTTP, que sería las clases públicas que puede visualizar el lado del cliente.

- DAO (Data Access Object) o repositorio. Es el encargado de las comunicaciones con la base de datos.
- Service. Aquí se encuentra toda la lógica de negocio.
- Model o entidades. Representa los diferentes objetos de la aplicación.
- DTO (Data Transfer Object). La responsabilidad de este módulo es de crear objetos planos con una serie de atributos que puedan ser enviados o recuperados del servidor. Esto permite que un objeto pueda contener información de múltiples tablas, con la finalidad de enviar al lado del socio la información que necesita.
- Converter. Este módulo se encarga de convertir una entidad en un DTO y viceversa.
- Exception. La responsabilidad de este módulo es de crear excepciones propias del prototipo.

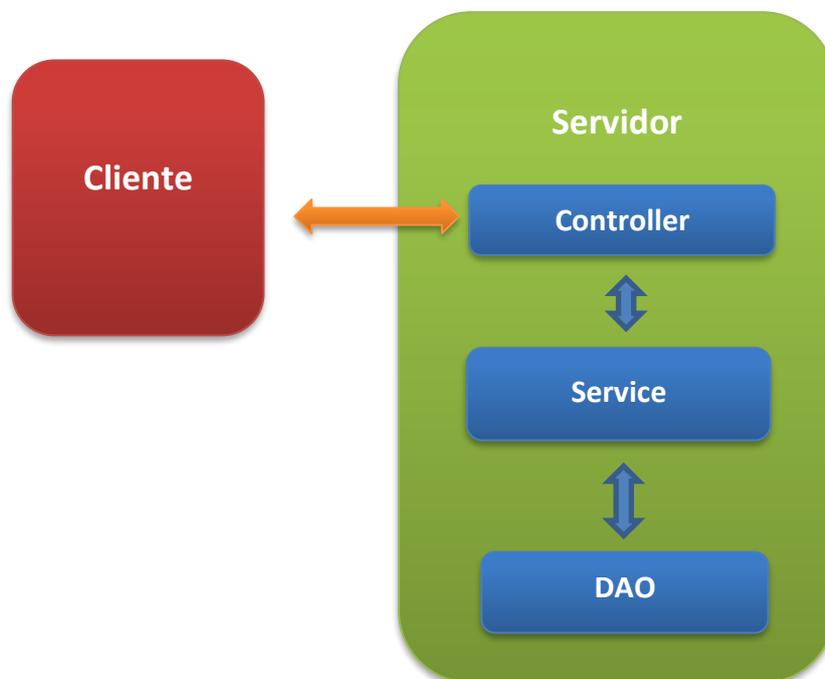


Figura 29

Cuando el lado del cliente realiza una consulta al servidor, los pasos para resolverla es la siguiente: el controller se encarga de recibir la consulta y se la encarga al service. Este realiza las operaciones necesarias y solicita los datos necesarios al DAO o pide que modifique alguna tabla y devuelve los datos al controller y el controller se lo devuelve al socio según como lo haya solicitado.

El módulo frontend también contiene varias capas que siguen las recomendaciones de Angular para reducir la dependencia entre componentes:

- Components. En este módulo se encuentra las vistas del socio.
- DTO. Los objetos planos que el socio necesita.
- Model. El modelo de la base de datos.
- Servicios. Es el encargado de realizar las consultas al lado del servidor y los servicios del lado del socio.
- Utilities. Aquí se encuentra servicios o clases de utilidades para los componentes.

5.7.2. Diseño *responsive*

Para la realización de la aplicación/del lado del cliente se decidió usar un diseño web *responsive*.

El diseño web *responsive* [28] o diseño web adaptativo es una técnica de diseño que busca la correcta visualización de una página web en distintos dispositivos, desde ordenadores de escritorio a dispositivos móviles y tabletas. Esto se consigue al redimensionar los elementos de la web y colocarlos de manera que se adapten al tamaño de cada dispositivo.

La razón de la elección de hacer o usar el diseño *responsive*, fue que la aplicación final será usada tanto en dispositivos móviles como en ordenadores, la idea sería que los socios usaran la aplicación móvil mayoritariamente. Por tanto, el hacer el prototipo con este tipo de diseño, evitaría tener que desarrollar varias aplicaciones para cada tipo dispositivo en la fase del prototipo. Se tiene una sola página web que se adapta a las diferentes resoluciones de pantalla.

5.7.3. Arquitectura de la base de datos

La base de datos que se diseñó para la realización del prototipo es una base de datos MySQL con doce tablas:

- Gym. Contiene a los gimnasios.
- User. Contiene a los usuarios. La distinción de usuarios se realizó por el campo 'role', es decir, rol.
- Class_directed. Contiene las clases colectivas diarias.
- Type_class. Contiene los tipos de las clases colectivas: GAP, Zumba, Pilates...
- Class_directed_client_list. Representa la relación entre una clase y los socios que asisten.
- Workout_table. Contiene a las rutinas.
- Exercise. Contiene a los ejercicios.
- Exercise_type. Contiene a los tipos de ejercicio.
- Exercise_category. Contiene a la categoría de los ejercicios o zona: Brazos, pecho, espalda...
- Workout_table_exercice_list. Representa la relación entre una rutina y su lista de ejercicio.
- Team. Contiene a los equipos.
- Team_members. Representa la relación entre un equipo y sus miembros.

La siguiente figura muestra el diagrama ER [37] de la base de datos.

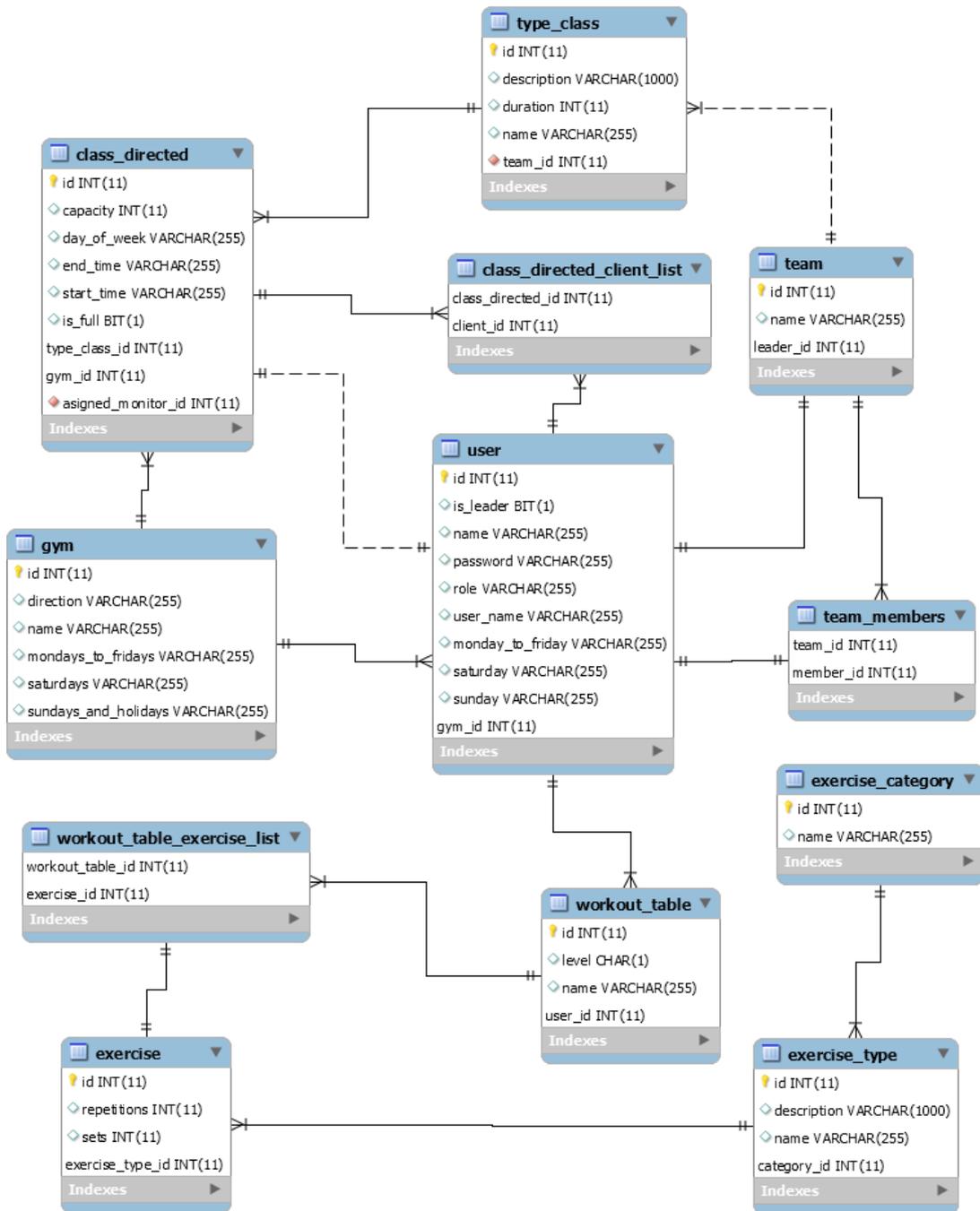


Figura 30

6. Desarrollo

6.1. Alcance del prototipo

En este primer punto se estará describiendo que funcionalidades tendrá el prototipo que se entregará junto a la actual memoria, que será muy similar a los mockups que se mostraron en el apartado 4.

El prototipo implementa las funcionalidades más interesantes para los gimnasios, que son:

- Primeramente, un sistema de inicio de sesión para mostrar el sistema de roles que tiene la aplicación.
- Las funcionalidades que tendría el usuario con rol administrador son:
 - Dar de alta a un usuario en la aplicación.
 - Crear y modificar equipos.
 - Ver las clases del gimnasio y poder asignarlas a un equipo.
- Las funcionalidades que tendría el usuario con el rol de monitor son:
 - Ver todas las clases colectivas del gimnasio, destacando en las que está asignado.
 - Ver la lista de reservados y añadir a los que no hayan reservado
- Las funcionalidades que tendría el usuario con el rol de coordinador:
 - Todas las funcionalidades que tiene el de rol monitor.
 - Ver a los monitores que están bajo su cargo.
 - Ver las clases colectivas que tiene asignadas y poder elegir que monitor se encarga de cada una.

- Las funcionalidades que tendría el usuario con el rol de socio:
- Ver las clases colectivas del gimnasio.
- Reservar en aquellas que vaya a asistir.
- Ver las rutinas de entrenamiento del gimnasio.
- Modificar las rutinas de entrenamiento del gimnasio.

En los siguientes apartados, se explica el proceso de desarrollo. Como se comentó anteriormente, cada iteración, tiene unas series de historias, las cuales tienen la siguiente estructura:

- ID: Identificador numérico.
- Nombre: Nombre de la historia
- Descripción: La historia con el formato “Como (rol del usuario) quiero (acción) para (beneficio)”
- Rol: Rol del usuario.
- Prioridad: Prioridad de realización de la historia, en la escala 0 a 100, siendo 100 la prioridad máxima.
- Validación: Criterios de aceptación de la historia.
- Estimación: Estimación de cuanto se tardaría en realizar la historia en horas.
- Velocidad real: Velocidad real de lo que se tardó en hacer la historia en horas.
- Notas: Notas sobre la implementación.

Se pueden ver los resultados en el Anexo.

6.2. Iteración 1

En la iteración 1, los objetivos alcanzados han sido inicializar el proyecto y ya tener una primera versión con operaciones sencillas para el socio.

Los roles que se trabajaron fueron el rol de administrador, ya que, era necesario para poder crear usuarios, y el rol de socio con las funcionalidades que se muestran en las historias.

Se crearon las clases necesarias para cada funcionalidad y se añadieron cuatro tablas a la base de datos.

HU001	Inicializar el proyecto
Descripción	Estructurar el frontend y el backend, inicializar los repositorios y preparar la BBDD
Rol	Administrador de la aplicación
Prioridad	100
Validación	Todo se encuentra subido en el repositorio remoto
Estimación	2 horas
Velocidad real	2 horas

HU002	Iniciar sesión
Descripción	Como usuario registrado quiero poder identificarme usando una contraseña para poder acceder a la aplicación.
Rol	Usuario
Prioridad	90
Validación	Se recibe al usuario
Estimación	3 horas
Velocidad real	6 horas y 50 minutos

HU003	Crear usuario
Descripción	Como administrador quiero poder crear usuarios para poder registrarlos en la aplicación
Rol	Administrador
Prioridad	80
Validación	Se crea en la base de datos
Estimación	4 horas
Velocidad real	6 horas y 45 minutos

HU004	Ver usuario
Descripción	Como administrador quiero poder ver los usuarios registrados para poder modificarlos.
Rol	Administrador
Prioridad	60
Validación	Se recibe a los usuarios registrados
Estimación	4 horas
Velocidad real	5 horas

HU005	Ver gimnasio
Descripción	Como socio quiero poder ver los datos del gimnasio para estar informando.
Rol	Socio
Prioridad	70
Validación	Se recibe el gimnasio en el que esta apuntado
Estimación	3 horas
Velocidad real	4 horas

HU006	Ver tablas
Descripción	Como socio quiero poder ver las tablas para poder prepararme una rutina.
Rol	Socio
Prioridad	70
Validación	Se ven todas las tablas del gimnasio
Estimación	6 horas
Velocidad real	6 horas y 35 minutos

6.3. Iteración 2

En esta segunda iteración, el resultado obtenido ha sido poder guardar rutinas por parte del socio y mostrar la página principal del monitor.

Además de las historias trabajadas en esta iteración, se agregaron algunas de la iteración anterior para poder mejorarlas, principalmente en visualización, como fue la HU005.

Se añadieron en esta iteración las tablas correspondientes para poder realizar las funcionalidades que se muestran a continuación.

HU007	Cerrar sesión
Descripción	Como usuario registrado quiero poder cerrar mi sesión en cualquier momento para salir de la aplicación.
Rol	Usuario
Prioridad	40
Validación	Se vuelve a la pantalla de inicio de sesión
Estimación	3 horas
Velocidad real	20 minutos

HU008	Ver tabla
Descripción	Como socio quiero poder seleccionar alguna tabla para poder ver los ejercicios en la misma.
Rol	Socio
Prioridad	70
Validación	Se recibe una tabla en concreto
Estimación	4 horas
Velocidad real	3 horas y 40 minutos

HU009	Guardar tabla
Descripción	Como socio quiero poder guardar una tabla para facilitar la búsqueda cada que acceda a la aplicación.
Rol	Socio
Prioridad	60
Validación	Se crea una copia de una tabla para el socio
Estimación	3 horas
Velocidad real	7 horas

HU010	Modificar tabla
Descripción	Como socio quiero poder modificar una tabla para cambiar algún ejercicio.
Rol	Socio
Prioridad	80
Validación	Se crea una copia de una tabla con modificaciones
Estimación	4 horas
Velocidad real	9 horas y 30 minutos

HU011	Ver detalles de un ejercicio
Descripción	Como socio quiero poder los detalles de los ejercicios para saber cómo realizarlo.
Rol	Socio
Prioridad	60
Validación	Se ve la descripción del ejercicio
Estimación	2 horas
Velocidad real	3 horas

HU012	Ver clases dirigidas
Descripción	Como socio quiero poder ver el horario de las clases para poder asistir a alguna.
Rol	Socio
Prioridad	90
Validación	Se reciben todas las clases dirigidas del gimnasio.
Estimación	5 horas
Velocidad real	6 horas y 30 minutos

HU013	Ver horario de trabajo
Descripción	Como monitor quiero poder ver el horario de trabajo para saber cuándo y dónde me toca trabajar.
Rol	Monitor
Prioridad	90
Validación	Se ve su horario de trabajo
Estimación	3 horas
Velocidad real	4 horas

HU014	Ver clases asignadas
Descripción	Como monitor quiero poder ver las clases que tengo asignadas para poder prepararlas.
Rol	Monitor
Prioridad	60
Validación	Se ven las clases que tiene asignadas
Estimación	3 horas
Velocidad real	3 horas y 10 minutos

6.4. Iteración 3

En la iteración 3, se ha obtenido la funcionalidad para poder reservar una clase por parte del socio y añadir socios a una clase dirigida por parte del monitor.

En esta iteración, se mostró la aplicación a ciertos usuarios del gimnasio y se volvió a abrir algunas historias anteriores para poder trabajar mejor las funcionalidades que se identificaron.

Se trabajó más en el diseño y colores que tendría la aplicación. Se decidió usar una paleta de tonalidades verdes por la sensación de salud que se quiere lograr.

HU015	Reservar clase
Descripción	Como socio quiero poder reservar una de las clases para poder asistir sin tener que preocuparme del aforo
Rol	Socio
Prioridad	90
Validación	Se encontrará en la lista de socios de la clase
Estimación	6 horas
Velocidad real	7 horas y 40 minutos

HU016	Añadir socio
Descripción	Como monitor quiero poder añadir socios a las clases para poder registrar a aquellos que no hayan reservado.
Rol	Monitor
Prioridad	70
Validación	Se añade a la lista de socios
Estimación	4 horas
Velocidad real	2 horas y 50 minutos

HU017	Eliminar rutina
Descripción	Como socio quiero poder eliminar una de las rutinas guardadas para poder quitar las que ya no necesito.
Rol	Socio
Prioridad	60
Validación	Se elimina la rutina de la base de datos
Estimación	2 horas
Velocidad real	1 hora

HU018	Ver lista de socios
Descripción	Como monitor quiero poder ver la lista de socios de una clase para poder luego comprobar y/o poder añadir socios.
Rol	Monitor
Prioridad	80
Validación	Se ve la lista de socios
Estimación	4 horas
Velocidad real	2 horas

6.5. Iteración 4

En esta iteración, los resultados fueron la creación de equipos de trabajo y el poder visualizarlo por parte del coordinador.

En la parte de análisis de esta iteración se decidió mostrar la aplicación a algunos monitores y poder implementar bien las historias.

Además, se tuvieron en cuenta algunos detalles para la mejora de algunas páginas, por tanto, se estuvieron trabajando también en algunas historias anteriores.

HU019	Crear equipo
Descripción	Como administrador quiero poder crear equipos para poder registrar equipos de trabajo en la aplicación.
Rol	Administrador
Prioridad	90
Validación	Se crea el equipo en la base datos
Estimación	6 horas
Velocidad real	6 horas y 45 minutos

HU020	Ver equipos
Descripción	Como administrador quiero poder ver los equipos que están formados para poder modificarlos.
Rol	Administrador
Prioridad	50
Validación	Se ve la lista de equipos
Estimación	2 horas
Velocidad real	2 horas y 10 minutos

HU021	Asignar coordinador
Descripción	Como administrador quiero poder asignar coordinadores a los equipos para facilitar el trabajo de asignar las clases.
Rol	Administrador
Prioridad	70
Validación	Se asigna el coordinador en la base de datos.
Estimación	2 horas
Velocidad real	1 hora

HU022	Ver miembros de equipo
Descripción	Como coordinador quiero poder ver a los miembros de equipo que tengo asignado para poder repartir las clases asignadas.
Rol	Monitor
Prioridad	60
Validación	Se ve a todos los miembros del equipo
Estimación	2 horas
Velocidad real	3 horas

6.6. Iteración 5

En la iteración 5, se consiguió obtener la funcionalidad para el reparto o asignación de clases a los monitores.

En esta iteración se terminó de realizar las funcionalidades que tendría el prototipo.

HU023	Ver clases impartidas
Descripción	Como administrador quiero poder ver todas las clases que imparten el gimnasio para poder saber cuáles de estas no están asignadas.
Rol	Administrador
Prioridad	90
Validación	Se ve todas las clases
Estimación	4 horas
Velocidad real	4 horas y 25 minutos

HU024	Ver clases no asignadas
Descripción	Como administrador quiero poder ver las clases que no están asignadas para poder asignarlas.
Rol	Administrador

HU024	Ver clases no asignadas
Prioridad	80
Validación	Se ve todas las clases no asignadas
Estimación	4 horas
Velocidad real	2 horas

HU025	Asignar equipo a la clase
Descripción	Como administrador quiero poder asignar clases a un equipo para que el coordinador las pueda repartir.
Rol	Administrador
Prioridad	70
Validación	Se asigna la clase a un equipo.
Estimación	6 horas
Velocidad real	6 horas y 15 minutos

HU026	Asignar monitor a la clase
Descripción	Como coordinador quiero poder repartir las clases a los miembros para que sepan donde tienen que estar.
Rol	Monitor
Prioridad	60
Validación	Se asigna el miembro a la clase
Estimación	6 horas
Velocidad real	10 horas

6.7. Iteración 6

En esta iteración, se replanteó el diseño y estructura de cada pantalla del prototipo. Por tanto, no se creó ninguna nueva historia, si no, que se trabajó sobre cada una de ellas, sin añadir o quitar ninguna funcionalidad, simplemente mejorándola, quedando el prototipo como se ve en el manual de usuario.

7. Acceso al código y despliegue

El código de este trabajo está disponible en el siguiente enlace, donde se encuentra el repositorio, <https://github.com/GeraldoRV/MyGymPlanner>. Se puede descargar o clonar el código accediendo al enlace.

Como se ha comentado, se tiene dos módulos, el frontend y el backend.

El frontend ha sido desarrollado con Angular CLI, para poder ejecutarlo es necesario tener instalado en el sistema operativo NodeJS y NPM, que se puede descargar fácilmente en la página oficial de NodeJS.

Una vez instalado, se debe abrir una terminal en el directorio del frontend para ejecutar el comando 'npm install' y así instalar las dependencias.

El backend ha sido desarrollado con Spring, para poder ejecutarlo es necesario tener una versión superior de JDK 1.7 y Maven 3.2.

Para poder desplegar la aplicación, es necesario que desde una terminal en el directorio del frontend se ejecute el comando 'ng build' y a continuación en el directorio del backend el comando 'mvn clean install'. Se generará un fichero jar que se encuentra en '\MyGymPlanner\backend\target'.

La base de datos es local. Por tanto, para hacer uso de la base de datos, es necesario que la configure antes de desplegar la aplicación en el fichero '\MyGymPlanner\backend\src\main\resources\application.properties' y '\MyGymPlanner\backend\src\test\resources\application-test.properties' para la base de datos de las pruebas.

Cada vez que se ejecute la aplicación se recreará la base de datos con los datos que se encuentran en '\MyGymPlanner\backend\src\main\java\main\dao\DBSeeder.java'.

8. Conclusiones y trabajo futuro

8.1. Resultados del proyecto

Con el proyecto desarrollado se ha conseguido el prototipo de una aplicación que ayude a cumplir con los objetivos de los usuarios de un gimnasio low cost. La aplicación del prototipo contiene un diseño web *responsive*, siendo posible poder acceder a él desde cualquier tipo de dispositivo, simplemente conociendo el enlace y estando en la misma red.

Uno de los objetivos de realizar un prototipo es poder previsualizar cómo va a quedar la aplicación a desarrollar, así como conocer de una manera más clara las funcionalidades de la aplicación final. Con el prototipo desarrollado, podemos ver que se cumplen estos objetivos.

El prototipo contiene las funcionalidades más importantes de los usuarios de la aplicación. Estos usuarios son los socios y los monitores. El usuario monitor se ha dividido en dos roles, monitor y coordinador.

Empezando con los monitores, podemos decir que hay una funcionalidad que se destaca sobre las demás, ya que, afecta a una parte del funcionamiento de los gimnasios low cost, la cual es, la funcionalidad de asignar monitores a las clases colectivas. Un gimnasio low cost tiene como 20 o 30 clases colectivas a la semana, por tanto, el prototipo muestra cómo sería esta funcionalidad que va dirigida a los monitores que se encargan de esta tarea, es decir, los coordinadores.

Continuando con los monitores, el prototipo ha logrado mostrar las funcionalidades que más le ayudasen y redujesen la carga de trabajo diaria como es la muestra de las clases colectivas y los monitores asignados en cada una, mejorando así su experiencia en el trabajo.

Con respecto a los socios, las funcionalidades más importantes son aquellas que acerquen el gimnasio a ellos, por tanto, con el prototipo desarrollado se ha logrado mostrar cómo se verían las rutinas que ofrece el gimnasio, así como la posibilidad de modificarlas, para poder acercarla al nivel del socio. Además, al poder ver las

descripciones de cómo hacer los ejercicios, facilita al socio el realizarlo, disminuyendo futuras lesiones por ejecutarlos de manera errónea.

Además, el prototipo muestra la funcionalidad de ver poder y reservar las clases colectivas.

Por tanto, se puede concluir que con las funcionalidades que se han implementado, la experiencia personal del socio mejora, ya que, facilita el logro de sus objetivos.

Respecto al diseño de la interfaz de usuario, se ha logrado que sea simple, intuitiva y cómoda para los usuarios, logrando así que los usuarios puedan usarla fácilmente.

8.2. Posibles actuaciones futuras

Lo que se propone en primer lugar sería terminar de implementar el producto final a partir del prototipo validado y, posteriormente, continuar con la implementación de las funcionalidades identificadas en el análisis:

- La implementación del servicio asesoramiento personalizado, y todas las características que engloba al usuario con el rol de asesor y el rol del socio con este servicio:
 - La creación y modificación de rutinas por parte de los monitores, tanto si tuviesen el rol de asesor como si no.
 - Poder solicitar el servicio por parte del socio.
 - Creación y modificación de fichas de socios dentro del servicio para esta funcionalidad.
 - Poder visualizar la rutina o rutinas que el asesor asignase al socio.
- Que el socio pueda visualizar los datos que el gimnasio tenga de este, además de poder registrar aquellos que sean necesarios para poder medir el progreso, como sería el peso o la altura. Sería interesante que el asesor tuviera acceso para visualizar y modificar ciertos datos para la realización de la ficha, en el caso de que el socio tuviese contratado el servicio de asesoramiento.

- Que los ejercicios, además de la explicación, tuviesen vídeos sobre la realización correcta de estos.
- Notificaciones del pago de la cuota contratada por parte del socio.

También se espera que se pueda seguir analizando en profundidad este tipo de negocios, a través de entrevistas con monitores de diferentes gimnasios, con el objetivo de profundizar en su funcionamiento y conocer otros roles interesantes que se pudiesen implementar en la aplicación, como el rol del entrenador personal, que, se descartó en este prototipo por las razones descritas en el apartado del modelado del sistema.

Respecto al desarrollo, se espera que se siga usando la misma metodología que se usó para la implementación del prototipo. El prototipo fue desarrollado para una aplicación web con Angular en la parte del cliente, pero, como se puede ver en los requisitos no funcionales, se espera que la aplicación se implemente para Android y Windows, por lo que habría que analizar y decidir qué lenguaje utilizar.

En todo caso, el código a desarrollar debe ser modular y seguir los principios de Clean Code para poder mantener un código limpio y que sea fácil de entender y adaptar a las futuras necesidades de los gimnasios.

La base de datos del prototipo es local, pero para la aplicación debería ser una base de datos con gestión en la nube para poder gestionar los usuarios, las clases colectivas y las rutinas de ejercicios. Para el caso de los ejercicios, el servidor consumiría un servicio web de ejercicios.

Por último, habría que optimizaría el diseño de la interfaz de usuario para que fuera más atractiva y cómoda para los usuarios.

9. Fuentes de información

[1] Definición fitness: <https://cuidateplus.marca.com/ejercicio-fisico/diccionario/fitness.html>

[2] P. López. “El ‘fitness’ de EEUU aumenta su ticket medio y eleva su facturación hasta 28.600 millones en 2018”. Palco23. <https://www.palco23.com/fitness/el-fitness-de-eeuu-aumenta-su-ticket-medio-y-eleva-su-facturacion-hasta-28600-millones-en-2018.html>

[3] P. López. “El ‘fitness’ español escala al top ten del ranking mundial por facturación con 2.700 millones de dólares”. Palco23. <https://www.palco23.com/fitness/el-fitness-espanol-escala-al-top-ten-del-ranking-mundial-por-facturacion-con-2700-millones-de-dolares.html>

[4] P. López. “El negocio de los gimnasios en España crece un 2,5% en 2018, hasta 2.291 millones de euros”. Palco23 <https://www.palco23.com/fitness/el-negocio-de-los-gimnasios-en-espana-crece-un-25-en-2018-hasta-2291-millones-de-euros.html>

[5] R. Villaécija. “España infla la burbuja del fitness”. El mundo. <https://www.elmundo.es/economia/ahorro-y-consumo/2017/09/13/59b132cf22601d39768b4611.html>

[6] D. Gómez. “Introducción al modelo de negocio de los gimnasios Low Cost”. La jugada financiera. <http://lajugadafinanciera.com/gimnasios-low-cost-modelo-de-negocio/>

[7] D. Gómez. “¿Cómo está evolucionando el sector de los gimnasios?”. La jugada financiera. <http://lajugadafinanciera.com/gimnasios-evolucion-sector/>

[8] El mercado del fitness en España 2017. http://www.gepacv.org/wp-content/uploads/2017/05/Zoom-Mercado-2017_LifeFitness.pdf

[9] J. Salas. “La OMS alerta de la caída de la actividad física en el siglo XXI”. El País. https://elpais.com/elpais/2018/09/04/ciencia/1536054340_198371.html

[10] P. López. “2018: la práctica deportiva en España acelera al calor del ‘running’ y el ‘fitness’”. Palco23 <https://www.palco23.com/entorno/2018-la-practica-deportiva-en-espana-acelera-al-calor-del-running-y-el-fitness.html>

- [11] S. García. “8fit y el auge de las 'apps' de fitness que mueven 1.800 millones de euros en todo el mundo”. Bolsa Manía. <https://www.bolsamania.com/noticias/empresas/8fit-y-el-auge-de-las-apps-de-fitness-que-mueven-1800-millones-de-euros-en-todo-el-mundo--3207408.html>
- [12] Desarrollo iterativo e incremental: https://es.wikipedia.org/wiki/Desarrollo_iterativo_y_creciente
- [13] IntelliJ Idea: <https://www.jetbrains.com/idea/>
- [14] Spring: <https://spring.io/>
- [15] Angular: <https://angular.io/>
- [16] Angular CLI: <https://cli.angular.io/>
- [17] Trello: <https://trello.com/>
- [18] Git: <https://git-scm.com/>
- [19] GitKraken: <https://www.gitkraken.com/git-client>
- [20] GitHub: <https://github.com/>
- [21] MySQL: <https://www.mysql.com/>
- [22] Workbench: <https://www.mysql.com/products/workbench/>
- [23] StartUml: <http://staruml.io/>
- [24] Balsamiq: <https://balsamiq.com/>
- [25] Principios del manifiesto ágil: <https://agilemanifesto.org/iso/es/principles.html>
- [26] Definición de análisis DAFO: https://es.wikipedia.org/wiki/An%C3%A1lisis_FODA
- [27] UML: https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado
- [28] Diseño *responsive*: https://es.wikipedia.org/wiki/Dise%C3%B1o_web_adaptable
- [29] Arquitectura Cliente-Servidor: <https://es.wikipedia.org/wiki/Cliente-servidor>
- [30] Crowdfunding: <https://es.wikipedia.org/wiki/Micromecenazgo>
- [31] Business Angels: <https://www.business-angel.es/business-angels.html>

- [32] InnoBan: <https://www.businessangelsinnoban.es/>
- [33] ENISA: <https://www.emprenderencanarias.es/servicios-de-apoyo/creacion-de-empresas-3/financiacion/enisa-jovenes-emprendedores/>
- [34] FFF's: <https://www.muypymes.com/2015/08/31/friends-family-and-fools-el-primer-recurso-del-emprendedor-que-busca-financiacion>
- [35] Bootstrapping: <https://estrategafinanciero.com/bootstrapping-que-es/>
- [36] Informe Valgo: <https://www.valgo.es/7o-informe-gimnasios-low-cost-espana-2019>
- [37] Diagrama Entidad Relación: https://es.wikipedia.org/wiki/Modelo_entidad-relaci%C3%B3n

10. Anexo. Manual de usuario

10.1. Acceso a la aplicación

Para hacer uso de la aplicación, es necesario que el administrador conceda el acceso y las credenciales de usuario.

Una vez que tenga el acceso, se podrá acceder conociendo la URL, desde cualquier dispositivo. El manual de usuario mostrará la vista en ordenador en el caso del administrador y vista móvil en los demás casos. Las dos vistas son similares, lo único que cambiaría sería las posiciones de los objetos.



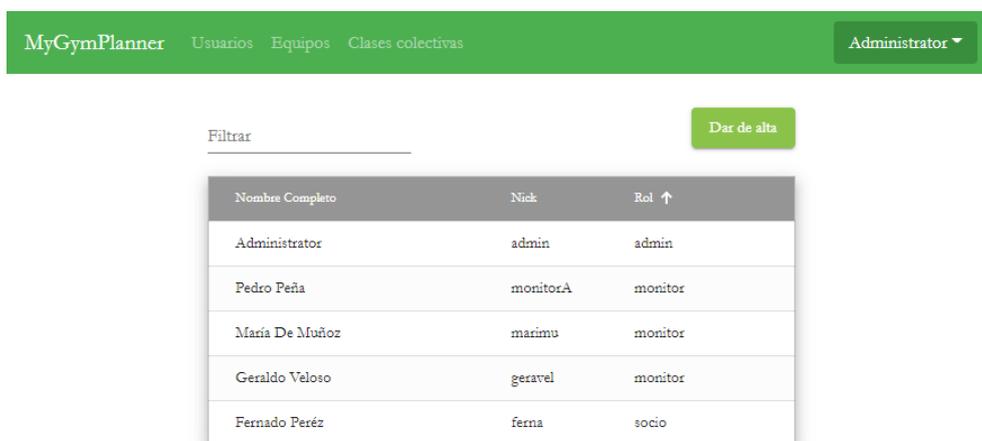
The image shows a login interface for 'MYGYMPLANNER'. At the top, there is a green banner with a white icon of a clipboard and the text 'MYGYMPLANNER' and 'Tu gimnasio, tu control'. Below this, the text 'Bienvenido!' is displayed. There are two input fields: the first is labeled 'Nick de usuario...' and the second is labeled 'Contraseña...'. Below the password field is a green button with the text 'Iniciar sesión'.

Figura 31

10.2. Administrador

10.2.1. Lista de usuarios

La primera pantalla una vez que se hayan insertado las credenciales de usuario, es la lista de usuarios de la aplicación. Donde se podrá filtrar según las columnas que se muestran, además de ordenar las columnas.



Nombre Completo	Nick	Rol ↑
Administrator	admin	admin
Pedro Peña	monitor.A	monitor
María De Muñoz	marimu	monitor
Geraldo Veloso	geravel	monitor
Fernado Pérez	ferna	socio

Figura 32

10.2.2. Dar de alta

Se podrá dar de alta a un nuevo usuario pulsando el botón dar de alta que mostrará la pantalla con el formulario para realizar la acción.

MyGymPlanner Usuarios Equipos Clases colectivas Administrator ▾

Nuevo usuario

Nombre completo

Nick de usuario

Rol

Contraseña (opcional)

Figura 33

En el formulario habrá que rellenar los campos de nombre completo, Nick de usuario y rol ('admin', 'monitor' y 'socio') de manera obligatoria y una contraseña de manera opcional, en el caso de que no se añada la contraseña será el Nick de usuario.

10.2.3. Equipos

Para acceder a los equipos se deberá pulsar 'Equipos' en la barra de navegación.

MyGymPlanner Usuarios Equipos Clases colectivas Administrator ▾

Equipo	Coordinador	Tamaño del equipo
A	Pedro Peña	3
B	Marta Hernández	3

Figura 34

10.2.4. Crear nuevo equipo

Al pulsar sobre el botón 'Crear nuevo equipo' se accederá a la página con el formulario para crear equipo.



The screenshot shows the 'Nuevo equipo' (New team) form in the MyGymPlanner application. The form is titled 'Nuevo equipo' and is located within a light blue container. At the top of the page, there is a green navigation bar with the text 'MyGymPlanner' and links for 'Usuarios', 'Equipos', and 'Clases colectivas'. On the right side of the navigation bar, there is a dropdown menu labeled 'Administrator'. The form itself contains the following elements:

- Nombre del equipo:** A text input field.
- Asignar coordinador:** A dropdown menu for selecting a coordinator.
- Monitores:** A list of available monitors: Pedro Peña, María De Muñoz, Geraldo Veloso, and Marta Hernández.
- Monitores en el equipo:** A list for the team's monitors, currently empty.
- Buttons:** A green button with a right-pointing arrow (>) to add a monitor, a green button with a left-pointing arrow (<) to remove a monitor, a green 'Crear equipo' button, and a red 'Cancelar' button.

Figura 35

En el formulario los campos obligatorios son el nombre del equipo y el coordinador. En el campo del coordinador se deberá elegir un monitor de la lista que se muestre.

Para asignar monitores en el equipo se deberá seleccionar a los monitores que están en el campo de la izquierda y pulsar sobre el botón '>'. Para desasignarlos se deberá hacer el mismo proceso con los monitores de la derecha y el botón '<'.

Se deberá pulsar sobre crear equipos para guardar los equipos.

10.2.5. Clases colectivas

Para acceder a las clases colectivas se deberá pulsar 'Clases colectivas' en la barra de navegación.

Clases

Clase	Número de clases	Equipo Asignado	
GAP	2	A	Modificar
Cross Fit	1	B	Modificar
Pilates	1	Aún no asignado	Añadir
Indoor Cycle	3	Aún no asignado	Añadir
Boxeo	3	Aún no asignado	Añadir
Zumba	4	Aún no asignado	Añadir

Figura 36

10.2.6. Asignar equipo

Para asignar un conjunto de clases semanales a un equipo, se deberá pulsar sobre el botón de 'Añadir' y si ya está asignado, pero se desea cambiar sobre 'Modificar'. En los dos casos se abrirá un dialogo para la asignación.

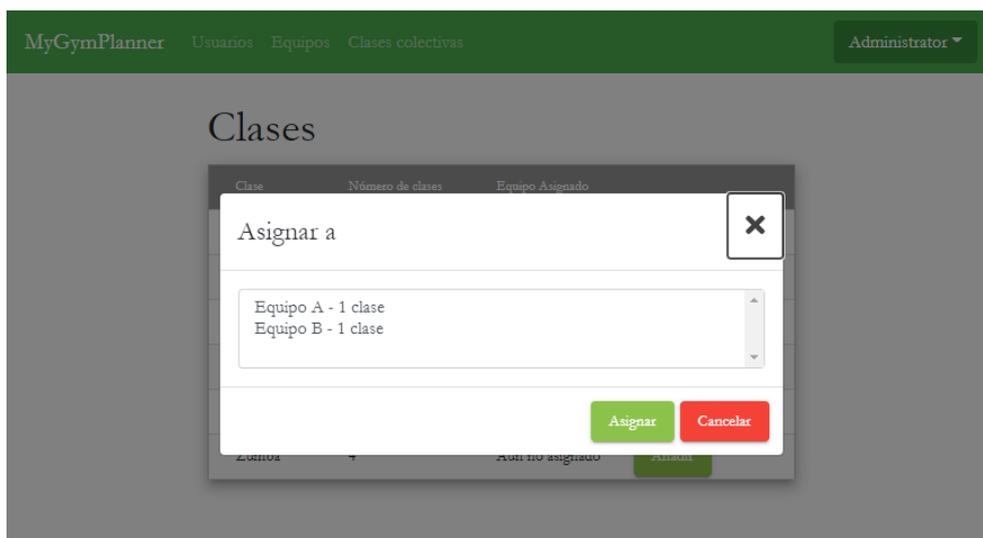


Figura 37

Lo que se muestra en el campo, es el nombre del equipo y el número del conjunto de clases asignadas. En el caso del equipo A, tiene asignado todas las clases semanales de GAP, Cross Fit y Zumba.

Se deberá seleccionar un equipo para asignarlo.

10.3. Socio

10.3.1. Inicio

La primera pantalla que se mostrará una vez que se hayan insertado las credenciales de usuario, es la página de inicio. En esta pantalla se muestra la información de gimnasio, las clases que imparte y los servicios.



Figura 38

Al pulsar sobre el nombre una de las clases se mostrará un dialogo con la descripción y duración de la clase.

Al pulsar sobre el menú, se desplegará lo que sería la barra de navegación.



Figura 39

10.3.2. Rutinas

Para acceder a las rutinas, hay que ir al menú o a pulsando en el botón 'Prepare su rutina' si se está en la pantalla de inicio.



Figura 40

En la pantalla se podrá ver las rutinas de gimnasio y las rutinas guardadas, 'rutinas' y 'mis rutinas'.

10.3.3. Ver y editar una rutina

Solo se podrá eliminar las rutinas que estén en 'Mis rutinas', pulsando sobre el icono 'basura'. Para poder ver los ejercicios de una rutina se deberá pulsar sobre el icono 'ojo'.



Figura 41

La primera pantalla muestra una rutina del gimnasio y en la segunda una rutina propia. Se puede modificar los ejercicios, añadiendo al pulsar añadir y eliminando al pulsar el icono de basura.

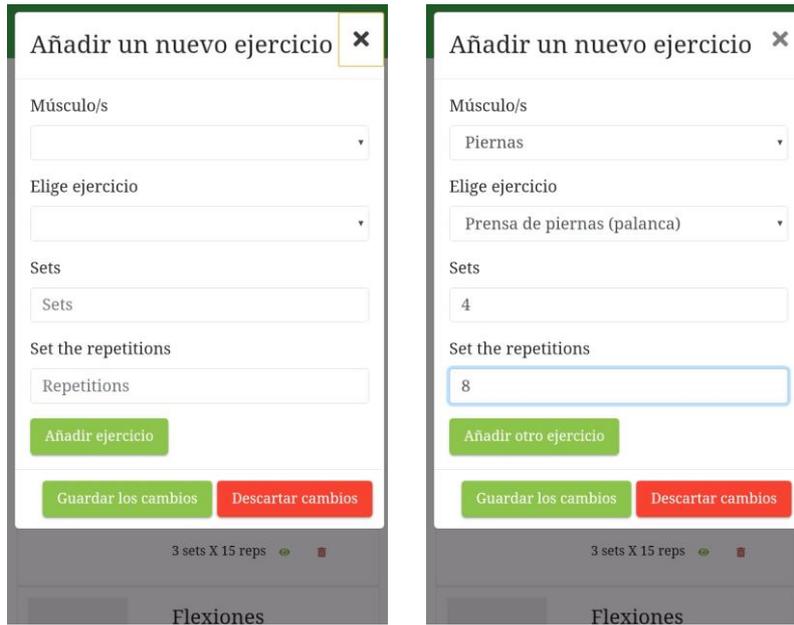


Figura 42

Hasta no pulsar guardar cambios, se podrá añadir tantos ejercicios como se desee.

Para ver un ejercicio específico se debe pulsar sobre el icono del ojo, en la pantalla que se muestra, se podrá editar un ejercicio pulsando sobre el lápiz.

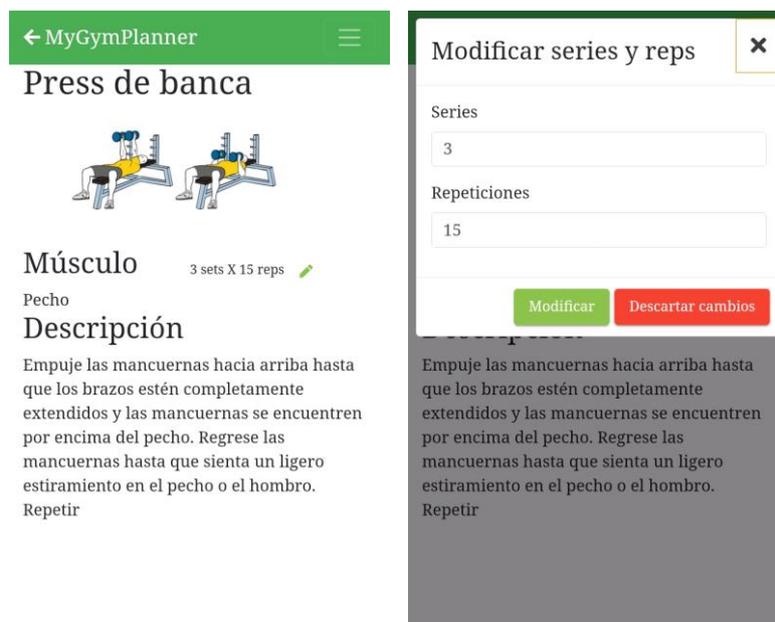


Figura 43

10.3.4. Ver y reservar clases colectivas

Para acceder a las clases, hay que ir al menú o pulsando en el botón 'Reserve su plaza en una clase' si se está en la pantalla de inicio.

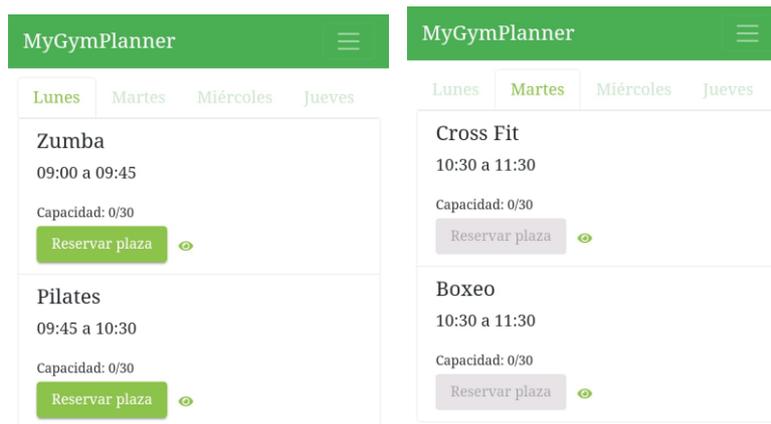


Figura 44

Al pulsar sobre el icono del ojo se podrá ver un modal con la descripción de la clase y al pulsar sobre el botón se realiza la reserva.

10.4. Monitor

10.4.1. Horario

La primera pantalla que se mostrará una vez que se hayan insertado las credenciales del monitor, es la página de inicio. En esta pantalla se muestra las clases del gimnasio y el horario de trabajo del monitor.



Figura 45

Las clases que el monitor tenga que impartir aparecerá el botón de 'Ver lista de socios'. En el caso de que no sea sus clases, aparecerá el nombre del monitor que tiene asignada la clase.

Al pulsar sobre el menú, se desplegará lo que sería la barra de navegación.



Figura 46

En el caso de que el monitor no sea coordinador, no se mostrará 'Clases colectivas'.

10.4.2. Ver lista de socios

Al pulsar sobre 'Ver lista de socios', se mostrará los socios apuntados en esa clase.

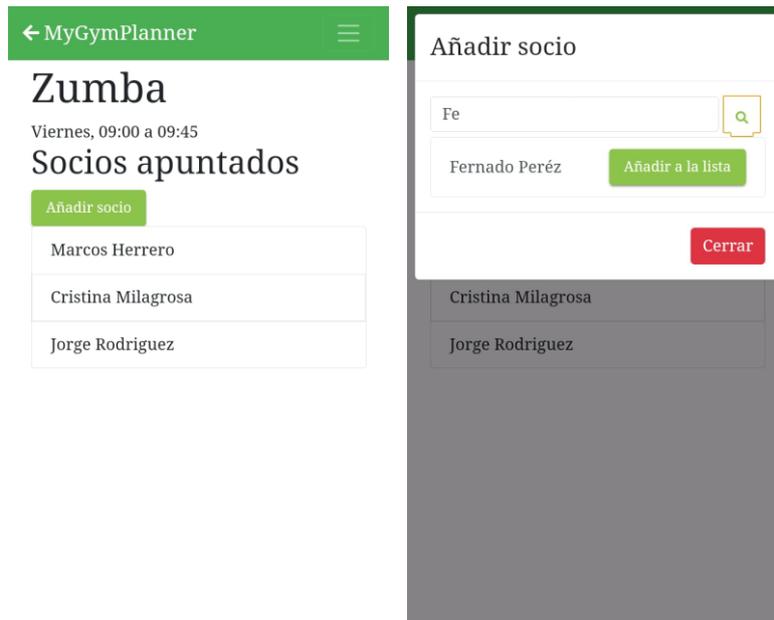


Figura 47

Se podrá añadir socios que no esten apuntados pero que hayan asistidos a traves del dialogo que se muestra al pulsar sobre el boton 'Añadir socio'.

10.4.3. Asignar monitores

Para asignar monitores, el monitor tiene que ser un coordinador y acceder a 'Clases colectivas' desde el menú.

The screenshot shows the 'MyGymPlanner' app interface. At the top, there is a green header with the app name and a menu icon. Below the header, the title 'Clases' is displayed. Underneath, there is a table with three columns: 'Nombre', 'Nº Clases', and 'Asignadas'. The table lists three classes: GAP (2 total, 1 assigned), Pilates (1 total, 0 assigned), and Zumba (4 total, 4 assigned). Each row has a small green eye icon to the right. Below the 'Clases' table, the title 'Monitores' is displayed. Underneath, there is another table with three columns: 'Nombre', 'Nick', and 'Nº clases asignadas'. The table lists three monitors: María De Muñoz (nick: marimu, 0 assigned), Geraldo Veloso (nick: geravel, 4 assigned), and Pedro Peña (nick: monitorA, 1 assigned).

Nombre	Nº Clases	Asignadas
GAP	2	1
Pilates	1	0
Zumba	4	4

Nombre	Nick	Nº clases asignadas
María De Muñoz	marimu	0
Geraldo Veloso	geravel	4
Pedro Peña	monitorA	1

Figura 48

En esta página se ve la tabla de las clases, donde se puede ver el número que hay y cuántas están asignadas, y la tabla de los monitores en el equipo, donde se puede ver su nombre y número de clases asignadas.

Para poder asignar, hay que pulsar sobre el icono del ojo.

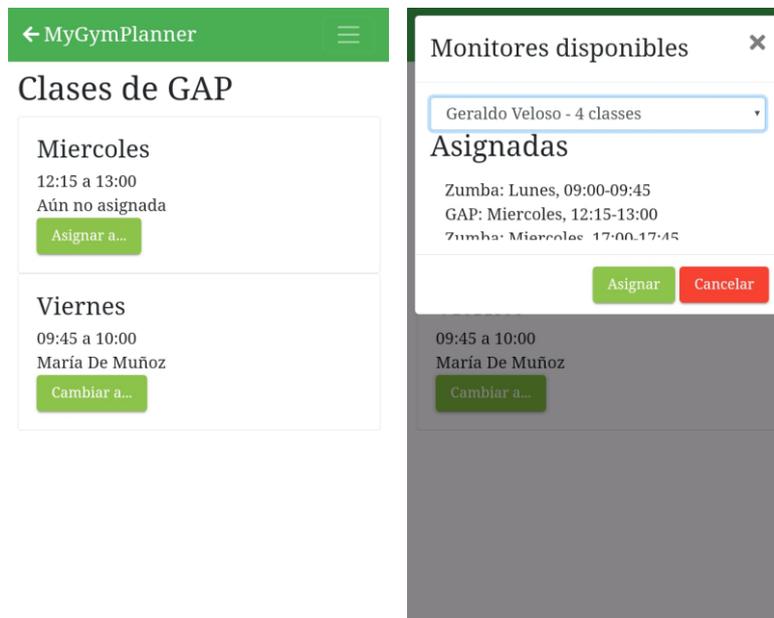


Figura 49

En esta página, se ve las clases semanales de cierta clase, en esta caso GAP. Al pulsar sobre el botón añadir se abría un diálogo donde se podrá elegir a un monitor para asignarlo. Cuando se selecciona a un monitor y antes de asignarlo, se muestra las clases que tiene este.