

Trabajo Fin de Grado

“Aplicación web para gestión de clubes de pádel”

Andrés Sánchez Tamargo

Tutor

Alejandro Rey López

Colmenarejo, julio 2022

AGRADECIMIENTOS

Habiendo terminado mi Trabajo de fin de grado, quiero dedicar unos agradecimientos a las personas que me han acompañado durante todo este viaje.

En primer lugar, a mi familia por haberme acompañado y apoyado desde el primer día para poder llegar hasta donde he llegado hoy.

En segundo lugar a todos mis compañeros a los que ya puedo llamar amigos que he conocido a lo largo de mi etapa en la universidad. Aquí he conocido a muchas de las personas que sé que me acompañarán durante el resto de mi vida. Me gustaría hacer una mención especial a mi compañero y amigo Fran, con el que, desde primero de carrera, he trabajado conjuntamente en la gran mayoría de proyectos de la carrera, nos hemos convertido a lo largo de estos años en el mejor equipo que jamás habría imaginado.

Me gustaría mencionar a mi pareja, Leonor, quien me ha apoyado siempre desde el primer momento y me ha dado esa motivación para seguir siempre que lo he necesitado, es algo que nunca olvidaré.

A los profesores y a la Universidad Carlos III de Madrid por haberme hecho disfrutar de estos años, y por haberme transmitido esa pasión por la informática.

Y por último, a mi tutor Alejandro Rey López por haberme permitido realizar este proyecto, por su tiempo y por haberme apoyado desde el primer día, muchas gracias a ti también.

Índice general

1. Introducción.....	13
1.1. Planteamiento del problema	13
1.2. Objetivos	14
1.3. Estructura del documento.....	15
2. Gestión de proyecto	17
2.1. Metodología y ciclo de vida	17
2.2. Planificación temporal.....	20
2.3. Análisis de costes	22
3. Marco regulador	25
4. Estado del arte	26
4.1. Estado actual del pádel.	26
4.2. Tecnologías Front-end.....	29
4.2.1. HTML	29
4.2.2. CSS	29
4.2.3. JavaScript.....	30
4.2.4. jQuery.....	30
4.2.5. Ajax.....	30
4.3. Tecnologías Back-end.....	30
4.3.1. Java EE	31
4.3.2. PHP	31
4.3.3. Python	31
4.4. Bases de datos.....	32
4.4.1. Bases de datos relacionales	32
4.4.2. Bases de datos no relacionales	33
4.5. Tecnologías escogidas	34
4.6. Aplicaciones web similares.....	35
4.6.1. Playtomic.....	35
4.6.2. Otras aplicaciones	36
5. Análisis del sistema	38
5.1. Descripción del problema	38
5.2. Especificación de requisitos	38
5.2.1. Requisitos funcionales.....	41
5.2.2. Requisitos no funcionales	59

5.3. Matriz de trazabilidad	62
6. Diseño del sistema	66
6.1. Arquitectura de la aplicación.	66
6.2. Diseño de la base de datos.....	68
6.2.1. Tabla administrador	69
6.2.2. Tabla usuario	69
6.2.3. Tabla clase	70
6.2.4. Tabla particular	70
6.2.5. Tabla producto	71
6.2.6. Tabla tarifa	71
6.2.7. Tabla pago	72
7. Implementación del sistema	73
8. Evaluación del sistema	80
9. Conclusiones y trabajos futuros	112
9.1. Conclusiones personales	112
9.2. Conclusiones sobre la realización del proyecto	113
9.3. Trabajos futuros	114
10. Glosario	115
11. Bibliografía	116
Anexo 2: Manual de usuario	119
Anexo 2: Summary	141

Índice de tablas

Tabla 1: Coste de personal	22
Tabla 2: Costes hardware	22
Tabla 3: Costes software	23
Tabla 4: Costes indirectos	23
Tabla 5: Costes totales del proyecto	24
Tabla 6: Comparativa entre servicios ofrecidos por aplicaciones de pádel.....	37
Tabla 7: Plantilla de tabla de requisitos	39
Tabla 8: Requisito funcional 1.	41
Tabla 9: Requisito funcional 2.	42
Tabla 10: Requisito funcional 3.	42
Tabla 11: Requisito funcional 4.	43
Tabla 12: Requisito funcional 5.	44
Tabla 13: Requisito funcional 6.	45
Tabla 14: Requisito funcional 7.	46
Tabla 15: Requisito funcional 8.	46
Tabla 16: Requisito funcional 9.	47
Tabla 17: Requisito funcional 10.	47
Tabla 18: Requisito funcional 11.	48
Tabla 19: Requisito funcional 12.	48
Tabla 20: Requisito funcional 13.	49
Tabla 21: Requisito funcional 14.	50
Tabla 22: Requisito funcional 15.	51
Tabla 23: Requisito funcional 16.	51
Tabla 24: Requisito funcional 17.	52
Tabla 25: Requisito funcional 18.	52
Tabla 26: Requisito funcional 19.	53
Tabla 27: Requisito funcional 20.	54
Tabla 28: Requisito funcional 21.	55
Tabla 29: Requisito funcional 22.	56
Tabla 30: Requisito funcional 23.	56
Tabla 31: Requisito funcional 24.	57
Tabla 32: Requisito funcional 25.	57
Tabla 33: Requisito funcional 26.	58
Tabla 34: Requisito funcional 27.	58
Tabla 35: Requisito no funcional 1	59
Tabla 36: Requisito no funcional 2	59
Tabla 37: Requisito no funcional 3	60
Tabla 38: : Requisito no funcional 4	60
Tabla 39: Requisito no funcional 5	61
Tabla 40: Requisito no funcional 6	61
Tabla 41: Requisito no funcional 7	62
Tabla 42: Matriz de trazabilidad de requisitos 1.....	63
Tabla 43: Matriz de trazabilidad de requisitos 2.....	64
Tabla 44: Matriz de trazabilidad de requisitos.....	65
Tabla 45: Descripción Handlers 1	74

Tabla 46: Descripción Handlers 2	75
Tabla 47: Descripción Handlers 3	76
Tabla 48: Descripción Handlers 4	77
Tabla 49: Descripción Handlers 5	78
Tabla 50: Descripción Handlers 6	79
Tabla 51: Plantilla tabla de pruebas	80
Tabla 52: Prueba 1.....	81
Tabla 53: Prueba 2.....	82
Tabla 54: Prueba 3.....	82
Tabla 55: Prueba 4.....	83
Tabla 56: Prueba 5.....	83
Tabla 57: Prueba 6.....	84
Tabla 58: Prueba 7.....	84
Tabla 59: Prueba 8.....	85
Tabla 60: Prueba 9.....	85
Tabla 61: Prueba 10.....	86
Tabla 62: Prueba 11.....	86
Tabla 63: Prueba 12.....	87
Tabla 64: Prueba 13.....	87
Tabla 65: Prueba 14.....	88
Tabla 66: Prueba 15.....	88
Tabla 67: Prueba 16.....	89
Tabla 68: Prueba 17.....	89
Tabla 69: Prueba 18.....	90
Tabla 70: Prueba 19.....	90
Tabla 71: Prueba 20.....	91
Tabla 72: Prueba 21.....	91
Tabla 73: Prueba 22.....	92
Tabla 74: Prueba 23.....	92
Tabla 75: Prueba 24.....	93
Tabla 76: Prueba 25.....	93
Tabla 77: Prueba 26.....	94
Tabla 78: Prueba 27.....	94
Tabla 79: Prueba 28.....	95
Tabla 80: Prueba 29.....	95
Tabla 81: Prueba 30.....	96
Tabla 82: Prueba 31.....	96
Tabla 83: Prueba 32.....	97
Tabla 84: Prueba 33.....	97
Tabla 85: Prueba 34.....	98
Tabla 86: Prueba 35.....	98
Tabla 87: Prueba 36.....	99
Tabla 88: Prueba 37.....	99
Tabla 89: Prueba 38.....	100
Tabla 90: Prueba 39.....	100
Tabla 91: Prueba 40.....	101
Tabla 92: Prueba 41.....	101
Tabla 93: Prueba 42.....	102

Tabla 94: Prueba 43.....	102
Tabla 95: Prueba 44.....	103
Tabla 96: Prueba 45.....	103
Tabla 97: Prueba 46.....	104
Tabla 98: Prueba 47.....	104
Tabla 99: Prueba 48.....	105
Tabla 100: Prueba 49.....	105
Tabla 101: Prueba 50.....	106
Tabla 102: Prueba 51.....	106
Tabla 103: Prueba 52.....	107
Tabla 104: Prueba 53.....	107
Tabla 105: Prueba 54.....	108
Tabla 106: Prueba 55.....	108
Tabla 107: Prueba 56.....	109
Tabla 108: Prueba 57.....	109
Tabla 109: Prueba 58.....	110
Tabla 110: Prueba 59.....	110
Tabla 111: Prueba 60.....	111

Índice de figuras

Figura 1: Visualización general Trello.....	18
Figura 2: Visualización tarjeta Trello	19
Figura 3: Diagrama de Gantt simplificado.....	20
Figura 4: Diagrama de Gantt expandido	21
Figura 5: Evolución clubes de pádel en Europa.....	26
Figura 6: Evolución pistas de pádel en Europa.....	27
Figura 7: Inversión en pádel	27
Figura 8: Países donde se juega a pádel en el mundo	28
Figura 9: Jugadores de pádel por país.....	28
Figura 10: Visualización interna del club con Playtomic	35
Figura 11: Arquitectura del sistema	66
Figura 12: Diseño de la base de datos.....	68
Figura 13: Funcionamiento interno de la aplicación.....	73
Figura 14: Inicio sesión Adminclub.....	120
Figura 15: Crear producto 1	121
Figura 16: Crear producto 2	121
Figura 17: Modificar producto	122
Figura 18: Modificar producto 2	122
Figura 19: Modificar producto 3	123
Figura 20: Vender producto	123
Figura 21: Vender producto 2	124
Figura 22: Vender producto 3	124
Figura 23: Crear usuario	125
Figura 24: Crear usuario 2	125
Figura 25: Detalles usuario.....	126
Figura 26: Detalles usuario 2.....	126
Figura 27: Detalles usuario 3.....	127
Figura 28: Modificar usuario	127
Figura 29: Buscar grupo.....	128
Figura 30: Buscar grupo 2.....	128
Figura 31: Crear grupo.....	129
Figura 32: Crear grupo 2.....	129
Figura 33: Crear grupo 3.....	130
Figura 34: Alumnos en grupo	130
Figura 35: Alumnos en grupo 2	131
Figura 36: Alumnos en grupo 3	131
Figura 37: Alumnos en grupo 4	132
Figura 38: Alumnos en grupo 5	132
Figura 39: Eliminar grupo	133
Figura 40: Particulares.....	133
Figura 41: Particulares 2.....	134
Figura 42: Crear particular	134
Figura 43: Crear particular 2.....	135
Figura 44: Confirmar clase particular	135
Figura 45: Confirmar clase particular 2	136

Figura 46: Confirmar clase particular 3	136
Figura 47: Tarifas	137
Figura 48: Tarifas 2	137
Figura 49: Crear tarifa	138
Figura 50: Crear tarifa 2	138
Figura 51: Modificar tarifa.....	139
Figura 52: Modificar tarifa 2.....	139
Figura 53: Cuentas.....	140
Figura 54: Cuentas 2.....	140
Figura 55: System architecture	145
Figura 56: Database design	147
Figura 57: Inners workings of the application.....	149

1. Introducción

En esta sección se explicará la motivación del Trabajo de Fin de Grado (TFG), su propósito y los objetivos que se quieren cubrir con el mismo. Por último, se describirán brevemente cada uno de los apartados de la estructura del documento.

1.1. Planteamiento del problema

El pádel es un deporte relativamente nuevo, que se encuentra en auge, tanto es así que, actualmente, todavía faltan herramientas adaptadas y diseñadas para la gestión interna y monitorización de los clubes de pádel.

Como profesor de pádel, ejerciendo desde hace ya más de 9 años, y habiendo trabajado en distintos clubes de pádel, he visto a este deporte crecer en todos los aspectos salvo en uno. No existen herramientas de gestión interna que permitan llevar un control de las clases impartidas o de los alumnos, así como de la tienda o diversas actividades de interés internas del club, que actualmente, se siguen llevando en papel.

Las grandes empresas del pádel se han centrado en el desarrollo de aplicaciones o plataformas de gestión de reservas de pistas de pádel dejando de lado otras muchas funciones importantes a la hora de gestionar internamente un club.

En este Trabajo de Fin de Grado (TFG), se propone el desarrollo de una aplicación web que cubra todas las necesidades relacionadas con la gestión interna de un club de pádel que actualmente no se encuentran cubiertas.

El proyecto se limita al diseño y desarrollo de una aplicación que permita la gestión de alumnos, grupos de pádel, clases particulares, la tienda y un control de las finanzas del club, monitorizando los cobros y las ventas realizadas. La gestión de las reservas de pistas no se encuentra dentro del alcance del proyecto ya que esta funcionalidad está más que cubierta por muchas otras aplicaciones ya existentes.

1.2. Objetivos

En esta sección se describirán los objetivos que se pretenden cumplir en el desarrollo de este proyecto, detallando las necesidades que se quieren cubrir.

El objetivo general de este proyecto es la realización de una aplicación web que permita la gestión interna de un club de pádel. Esta aplicación gestionará 4 áreas básicas a la hora de gestionar internamente un club:

- **Gestión de Usuarios:** Se podrán gestionar tanto los alumnos como los profesores del club, pudiendo dar de alta o baja a un usuario, así como llevar un control de los pagos realizados, de los productos adquiridos o de las clases en las que está inscrito un usuario.
- **Gestión de Productos:** Se podrán dar de alta o dar de baja nuevos productos, modificarlos, cambiar el stock y gestionar la venta de dichos productos.
- **Gestión de Clases:** Las clases están divididas en dos tipos:
 - **Grupos:** Se podrá dar de alta o dar de baja a un grupo, inscribir alumnos y asignar el grupo a un profesor.
 - **Particulares:** Se podrán crear distintos bonos de clases particulares (dependiendo del número de clases) asociados a un usuario.
- **Gestión de Finanzas:** Se podrá llevar un control de los cobros realizados ya sea por cobro de clases o de productos de la tienda del club, también se podrán modificar las tarifas de los distintos servicios ofrecidos.

1.3. Estructura del documento

En esta sección se hará una breve descripción de cada apartado en el que está dividido estructuralmente el proyecto.

- **Introducción:** En esta sección se explicará la motivación del Trabajo de Fin de Grado (TFG), su propósito y los objetivos que se quieren cubrir con el mismo.
- **Gestión del proyecto:** En esta sección se explicará la metodología que ha sido utilizada en la realización de este proyecto. También se explicará la planificación diseñada y los costes finales asociados al proyecto.
- **Marco regulador:** En esta sección se hará un estudio de la legislación aplicable a este proyecto.
- **Estado del arte:** En esta sección se llevará a cabo una investigación sobre todo el conocimiento acumulado del tema desarrollado en específico, cuál es la situación actual del tema del que se parte. Esta investigación nos permite tener un conocimiento crítico sobre el tema para poder crear nuevos conocimientos. Se hablará sobre las diferentes tecnologías existentes y sobre qué tecnologías se han escogido para la implementación del proyecto.
- **Análisis del sistema:** Se describirá el problema y se hará una detallada especificación de requisitos que se deben cumplir.
- **Diseño del sistema:** Se hablará sobre la estructura interna del sistema así como del diseño de la base de datos escogida.
- **Implementación del sistema:** Se hablará sobre el funcionamiento interno del sistema y sobre cómo ha sido implementado.
- **Evaluación del sistema:** Se elaborará un plan de pruebas para comprobar el correcto funcionamiento del sistema y el cumplimiento de todos los requisitos.
- **Conclusiones y Trabajos Futuros:** Se hablará sobre conclusiones personales como de la realización del trabajo y de posibles trabajos futuros que complementarían la aplicación desarrollada.
- **Anexo 1: Manual de usuario:** Se hará un manual de usuario explicando cómo realizar cada una de las funciones de la aplicación.

- **Anexo 2: Summary:** Breve resumen de todo el trabajo escrito en inglés.

2. Gestión de proyecto

2.1. Metodología y ciclo de vida

En esta sección se explicará la metodología ágil que ha sido utilizada en la realización de este proyecto.

La elección de una buena metodología de trabajo es algo crucial a la hora de poder desarrollar óptimamente un proyecto y que repercute directamente en el resultado final. Durante la carrera, las metodologías ágiles estudiadas estaban basadas principalmente en el trabajo en equipo. Al tratarse este proyecto de un proyecto individual, la metodología que se ha seguido ha sido una metodología propia adaptada a las necesidades que este proyecto requería.

Una vez realizado el análisis y el diseño del proyecto, se decidió que el método de trabajo a utilizar iba a ser un método de trabajo basado en ciclos siguiendo el patrón: front-end, back-end y pruebas.

El proyecto, por lo tanto, contará con 4 ciclos, uno por cada área de gestión: productos, usuarios, grupos y finanzas.

Para llevar un control del estado del proyecto, se ha utilizado la herramienta Trello. Trello es una herramienta web dedicada a la gestión de trabajos donde, mediante el uso de tarjetas, se puede llevar un control visual del estado general del proyecto, partes en las que se está trabajando, objetivos ya cumplidos o partes que todavía faltan por desarrollar.

Las tareas del proyecto están divididas por diferentes listas:

- Front-end: Todas aquellas tareas basadas en la programación del entorno visual del proyecto, es decir, de cada una de las páginas JSP.
- Back-end: Todas aquellas tareas basadas en la programación del funcionamiento interno de la aplicación, es decir, cada una de las clases java dedicadas al correcto funcionamiento del proyecto, llamadas handlers y controller.
- BBDD: Todas aquellas tareas basadas en el diseño y la programación de la base de datos.
- Working on: Tareas en las que se está trabajando en un momento dado.
- Bugs: Errores que han sido encontrados ya sea realizando otra tarea o mediante la utilización de la aplicación.
- Testing: Todas aquellas tareas basadas en la realización de pruebas de cada uno de los módulos de la aplicación.

En la figura 1 se mostrará una imagen en la que se podrá observar el estado del proyecto a día 16 de abril de 2022:

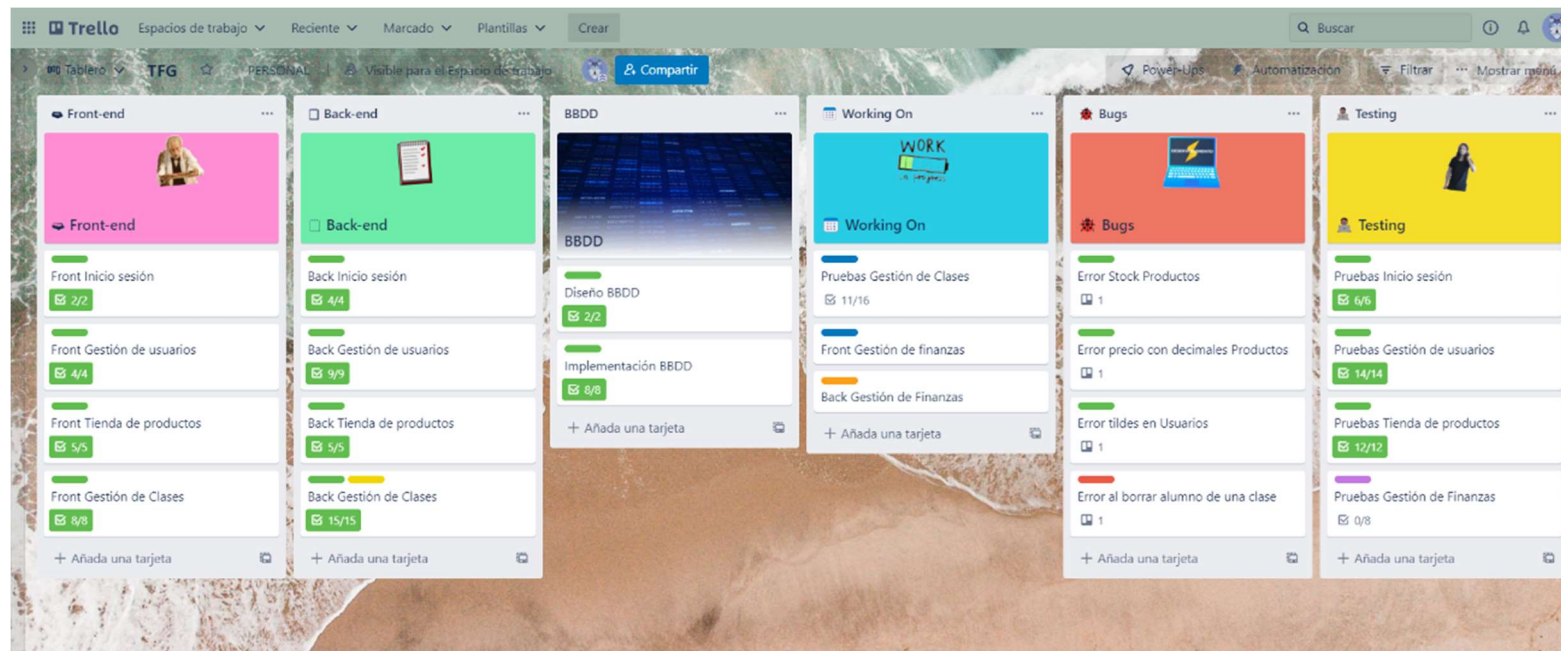


Figura 1: Visualización general Trello

Para poder distinguir el estado de cada una de las tareas, se asignaron diferentes colores a los diferentes estados posibles:

- Verde: Tarea completada. Al haber completado una tarea, se marca en verde, se devuelve a la lista de tal forma que a primera vista se pueda ver que esa tarea está completada.
- Azul: Tarea en proceso. Cuando se está desarrollando una tarea, se marca en azul y se incluye en la lista “Working on”
- Amarillo: Es necesaria una revisión de la tarea.
- Rojo: Precaución, hay un error por corregir.
- Naranja: Tarea bloqueada. Es necesaria la realización de otra tarea para poder seguir.
- Lila: Tareas pendientes.

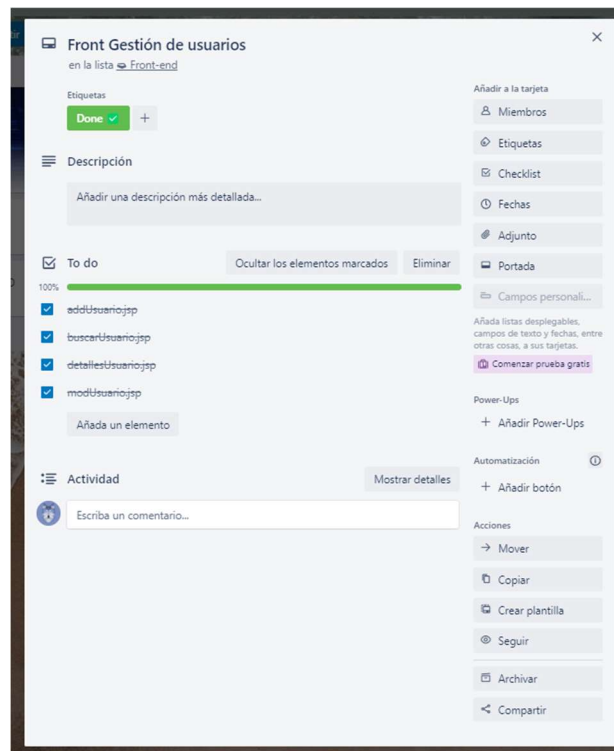


Figura 2: Visualización tarjeta Trello

En la figura 2 se muestra cómo es internamente una tarea. En la parte superior se puede observar el nombre “Front Gestión de usuarios”, esto nos indica que van a ser pequeñas tareas relacionadas con la parte visual del área de gestión de usuarios. Más abajo, está incluida una lista “To do” con las distintas pequeñas tareas a realizar, en este caso, cuatro tareas. Cada una de estas tareas se corresponde con una página JSP que se debe programar. Una vez terminada la página, se marca como realizada y queda tachada automáticamente de la lista. Cuando todas las tareas están acabadas, se marca con una etiqueta verde “Done”.

2.2. Planificación temporal

En esta sección se explicará la planificación temporal inicial diseñada para la realización de este proyecto. La realización del proyecto empezó el día 24 de enero de 2022 y la fecha final aproximada sería el 10 de junio de 2022.

A continuación se puede observar el diagrama de Gantt inicial del proyecto:

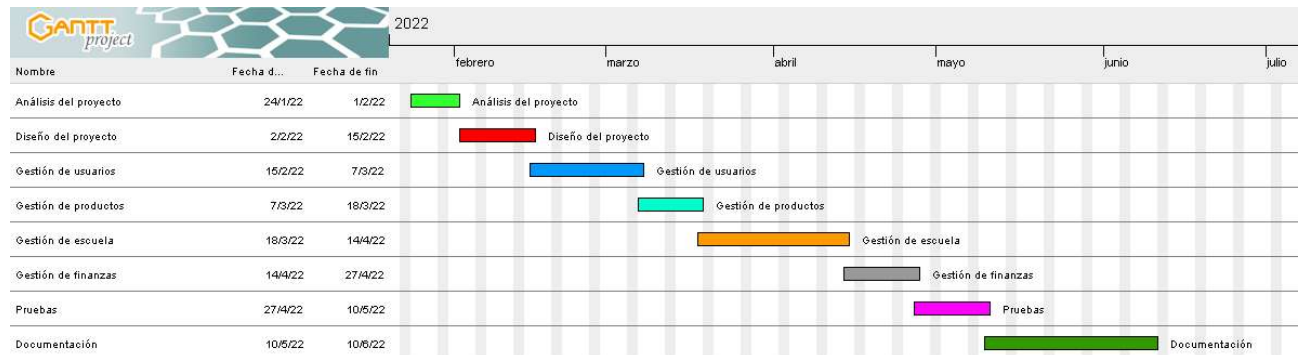


Figura 3: Diagrama de Gantt simplificado

Una vez realizados tanto el análisis del proyecto como el diseño, se estableció el tiempo de duración aproximado para realizar cada una de las tareas del proyecto, estando marcadas cada una de ellas de un color diferente.

En un principio, este diagrama de Gantt está diseñado teniendo en cuenta que se dedicarían 2h al día de lunes a viernes ininterrumpidamente desde el día de inicio del proyecto al día final.

Teniendo en cuenta lo anteriormente mencionado, el tiempo aproximado estimado para la realización de este proyecto fue de 200 horas.

A continuación se mostrará el diagrama de Gantt del proyecto dividiendo cada bloque en las distintas fases cíclicas que atraviesa, como ya se comentó anteriormente en el apartado 2.1 (Metodología y ciclo de vida):



Figura 4: Diagrama de Gantt expandido

En la realización del proyecto, se intentó ceñirse a la planificación temporal por lo que el tiempo de dedicación medio diario estimado que fue de 2 horas pasó a ser realmente de 2,5 horas dando como resultado un tiempo total de realización del proyecto de 250 horas.

2.3. Análisis de costes

En esta sección se detallarán los costes tanto de personal, hardware, software e indirectos, así como el coste final de la realización de este proyecto.

En cuanto al coste de personal, este proyecto ha sido realizado por dos individuos, un jefe de proyecto cuyas funciones principales serán coordinar y determinar los objetivos del proyecto. Y un analista, diseñador y programador que se encargará del desarrollo y la implementación de la aplicación, así como de llevar a cabo las pruebas correspondientes que comprueben el correcto funcionamiento del sistema.

El coste de personal quedaría de la siguiente manera:

	Rol	Precio	Horas	Sueldo
Alejandro Rey López	Jefe de proyecto	25 €/h	40	1.000 €
Andrés Sánchez Tamargo	Analista Diseñador Programador	20 €/h	250	5.000 €
Coste total del personal				6.000 €

Tabla 1: Coste de personal

Dentro de los costes materiales, los costes hardware comprenden aquellos costes en material informático físico que han sido requeridos para la realización del proyecto. Este proyecto ha sido realizado en un ordenador Honor MagicBook 14, por lo que los costes hardware quedarían de la siguiente manera:

Costes Hardware				
Producto	Coste total	Uso (en semanas)	Vida útil (en semanas)	Coste en proyecto
Honor MagicBook 14	750 €	20	208	72,12 €
Total costes hardware				72,12 €

Tabla 2: Costes hardware

Los costes software comprenden todos los costes en programas u aplicaciones que han sido necesarias para la realización de este proyecto, en este caso, no se ha requerido invertir dinero en software para la realización de la aplicación, quedando la tabla de costes software de la siguiente manera:

Costes Software			
Producto	Coste total	Uso (en semanas)	Coste en proyecto
Microsoft Windows 11	0 €	20	0,00 €
Trello	0 €	20	0,00 €
MySQL Workbench	0 €	20	0,00 €
Servidor Payara Server 5	0 €	20	0,00 €
Plantilla front-end	0 €	20	0,00 €
Eclipse	0 €	20	0,00 €
Gantt Project	0 €	20	0,00 €
Total costes software			0,00 €

Tabla 3: Costes software

Los costes indirectos comprenden el coste de la luz o el agua que han sido indirectamente necesarios para la realización de la aplicación. La tabla de costes indirectos quedaría de la siguiente forma:

Costes Indirectos			
Producto	Coste total (en semanas)	Uso (en semanas)	Coste en proyecto
Electricidad	3 €	20	60,00 €
Internet	4 €	20	80,00 €
Total costes software			140,00 €

Tabla 4: Costes indirectos

Una vez calculados cada uno de los costes del proyecto, se calculará el coste final de la realización del mismo. Para calcular el coste final se tendrán en cuenta los costes tanto de personal, hardware, software e indirectos añadiéndole a estos el margen de riesgo que será de un 15%, un margen de beneficio de un 20% y un IVA de un 21%. Tanto el margen de riesgo como el margen de beneficio han sido calculados teniendo en cuenta únicamente los costes directos. Posteriormente, se aplicó el IVA del 21% dando como resultado el coste final del proyecto.

Tipo de coste	Coste
Coste total de personal	6.000 €
Coste total hardware	72,12 €
Coste total software	0 €
Costes indirectos	140 €
Margen de riesgo (15%)	910,82 €
Beneficios (20%)	1.214,42 €
Base imponible	8.337,36 €
IVA (21%)	1.750,85 €
Coste total del proyecto	10.088,21 €

Tabla 5: Costes totales del proyecto

3. Marco regulador

En esta sección se hará un estudio de la legislación aplicable a este proyecto. Es necesario destacar que la aplicación actualmente no se encuentra accesible a través de internet, sin embargo, está previsto que lo esté por lo que debe cumplir la siguiente legislación:

- **Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales. (LOPDGDD).** Esta ley tiene como objetivo garantizar la confidencialidad, el buen tratamiento y la protección de los datos personales manejados. Los datos recogidos pueden recogerse en tres niveles teniendo en cuenta la sensibilidad de los mismos:
 - **Básico:** NIF, nombre, apellidos, teléfono, dirección, número de tarjeta...
 - **Medio:** gustos, costumbres, crédito o solvencia, datos tributarios...
 - **Alto:** salud, origen racial, ideología, sexualidad...

En cuanto a nuestra aplicación, todos los datos recogidos serán de nivel básico.

El responsable de la recogida de los datos también es el responsable del cumplimiento de esta ley.

Entre los deberes del responsable están el deber de informar al usuario para obtener el consentimiento de la manipulación de sus datos personales, el asegurarse que los datos obtenidos son veraces y adecuados, y la obligación de guardar secreto y seguridad.

En nuestro proyecto, deberá de ser el responsable de cada club, el que se encargue de notificar al usuario y de obtener su consentimiento para el tratamiento de sus datos personales [1, 2].

Licencias

Para la realización del proyecto ha sido usada una plantilla front-end llamada AdminLTE donde se especifica lo siguiente [3]:

Qué puede hacer con AdminLTE:

- Utilizarlo en proyectos comerciales.
- Usar en proyectos personales/privados.
- Modificar y cambiar el trabajo.
- Distribuir el código.
- Sublicenciar: incorporar la obra en algo que tenga una licencia más restrictiva.

4. Estado del arte

En esta sección estudiaremos el estado del arte del proyecto. El estado del arte hace referencia a la investigación sobre todo el conocimiento acumulado de un tema en específico, cuál es la situación actual del tema del que se parte. Esta investigación nos permite tener un conocimiento crítico sobre el tema para poder crear nuevos conocimientos.

4.1. Estado actual del pádel.

Según el estudio “Global Padel Report” [4] realizado por la consultora Monitor Deloitte en colaboración con la empresa Playtomic, el pádel es, a día de hoy, uno de los deportes con mayor crecimiento a nivel mundial.

Tanto es así que tanto el número de clubes en Europa como el número de pistas construidas han tenido un crecimiento exponencial en los últimos años. Tanto es así que el número de clubes ha crecido un 181% y el número de pistas un 159% desde 2016 hasta 2021.

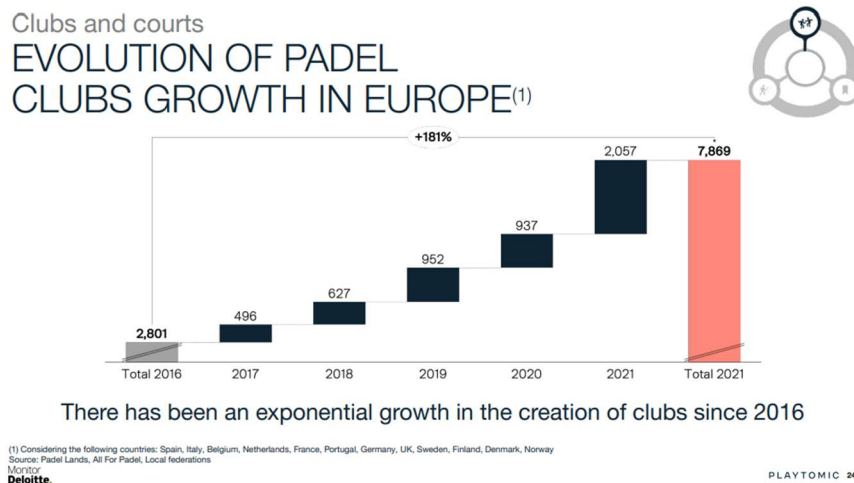
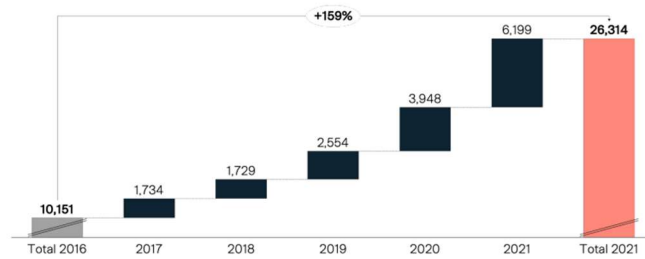


Figura 5: Evolución clubes de pádel en Europa

Clubs and courts EVOLUTION OF PADEL COURTS GROWTH IN EUROPE⁽¹⁾



Almost 25% of total courts in selected countries have been constructed in 2021

(1) Considering the following countries: Spain, Italy, Belgium, Netherlands, France, Portugal, Germany, UK, Sweden, Finland, Denmark, Norway
Source: Padel Lands, All For Padel, Local federations

Monitor
Deloitte.

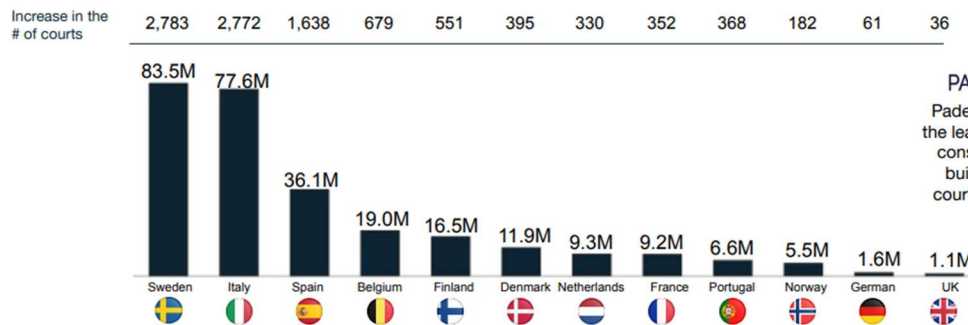
PLAYTOMIC 25

Figura 6: Evolución pistas de pádel en Europa

El ranking en estos dos ámbitos (creación de clubes de pádel y creación de pistas de pádel) los lidera por amplia diferencia España, sin embargo, cada vez son más los países que están invirtiendo dinero en este deporte.

Dentro de Europa podemos destacar países como Italia o Suecia, siendo Suecia el país de Europa que más ha invertido en pádel en los últimos años.

Clubs and courts INVESTMENTS MADE IN THE LAST 2 YEARS



PADEL GALIS®
Padel Galis, one of the leading European constructors, has built **3,000** new courts in the last 2 years

The total investment in padel courts construction in the last 2 years was **279M€** ⁽¹⁾

(1) Considering the following countries: Spain, Italy, Belgium, Netherlands, France, Portugal, Germany, UK, Sweden, Finland, Denmark, Norway (2) Considering the following court construction prices (€): Portugal (18K), Spain (22K) France and Germany (26K), Italy, Netherlands, Belgium (28K), Sweden, Finland, Norway, Denmark and UK (30K)
Source: Padel Lands, All For Padel, Padel Galis, Market experts interviews

Monitor
Deloitte.

PLAYTOMIC 28

Figura 7: Inversión en pádel

Sin embargo, el pádel no está localizado únicamente en Europa. Cada vez más países del mundo están invirtiendo en este deporte, destacando Sudamérica donde el pádel ya es un deporte más que consolidado puesto que nació allí o los Emiratos Árabes Unidos, que en este 2022 han invertido y creado un nuevo circuito profesional de pádel llamado Premier Padel, llevando a cifras de récord los premios repartidos a los campeones de cada torneo.

Worldwide Padel Phenomenon PADEL IN NUMBERS

IN THE PRESENT

+90 Countries

Padel Courts & Players

51
National Federations in FIP

+18 M
Active Padel Players

300 K
Federated Players

Monitor
Deloitte.

IN THE FUTURE 

PADEL strives to become WORLDWIDE & an EXHIBITION SPORT in the Olympics of 2024



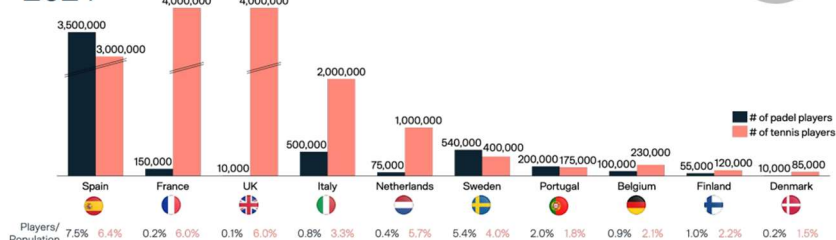
Figura 8: Países donde se juega a pádel en el mundo

A día de hoy, el pádel, es jugado en más de 90 países alrededor del mundo, teniendo ya más de 51 federaciones de pádel, más de 18 millones de jugadores en activo y más de 300 mil jugadores federados, teniendo en cuenta que este estudio se hizo en 2021.

En España, Suecia y Portugal, el número de jugadores de pádel ha superado ya a el número de jugadores de tenis, según los estudios, con el paso de los años, si se sigue con esta tendencia, el pádel acabará superando al tenis a nivel mundial, sin embargo, todavía queda mucho camino para llegar a ese punto.

Competitions and federations

TOTAL NUMBER OF PLAYERS BY COUNTRY 2021



In Spain, Sweden and Portugal the number of padel players has surpassed that of tennis players. In other countries like France, UK, Italy or the Netherlands there are considerably less padel players than tennis players, leaving room for a huge growth potential

Monitor
Deloitte.

Source: Playtomic, International Padel Federation (FIP), International Tennis Federations (ITF)

PLAYTOMIC 55

Figura 9: Jugadores de pádel por país

4.2. Tecnologías Front-end.

En esta sección se hará un estudio sobre algunas de las diferentes tecnologías front-end existentes para poder argumentar finalmente la elección de las tecnologías usadas en este proyecto.

Las tecnologías front-end son las utilizadas en el lado del cliente, son las tecnologías que se usan tanto para diseñar la interfaz de la aplicación, es decir, la parte visual como para conectarnos con el servidor a través de internet. Estas tecnologías y lenguajes vienen normalmente implementadas en los navegadores como podrían ser Google Chrome o Mozilla Firefox [5].

4.2.1. HTML

HTML o *HyperText Markup Language* es un lenguaje de marcado para la elaboración de páginas web con el que, a través de etiquetas, se puede definir la estructura y semántica de su contenido.

Estas etiquetas pueden ser de múltiples tipos como podría ser una etiqueta de párrafo, un título, una imagen o un encabezado. Actualmente existen 142 etiquetas HTML disponibles [6].

Ventajas.

Entre sus principales ventajas destaca su sencillez con una curva de aprendizaje pronunciada pues se alcanza un alto nivel de destreza rápidamente.

Desventajas.

Entre sus principales desventajas destaca el tener que utilizar otro lenguaje de programación como podría serlo JavaScript para poder otorgarle una funcionalidad dinámica. Las páginas HTML son páginas estáticas.

4.2.2. CSS

CSS o *Cascading Style Sheets*, se trata de una tecnología front-end utilizada para dotar de cualidades visuales y estéticas a una página web[7].

Con CSS podemos, como su propio nombre indica, otorgarle de estilo a un documento HTML. Para ello, tanto HTML como CSS trabajan conjuntamente, siendo el documento HTML el encargado de portar la información de la estructura y el contenido de la página web y el documento CSS el estilo.

4.2.3. JavaScript

JavaScript es el lenguaje de programación encargado de proporcionar un mayor dinamismo e interactividad a una página web.

Con este lenguaje de programación se pueden crear animaciones, efectos, controlar las entradas en los formularios o modificar elementos de una página web de forma sencilla.

Ahora mismo, JavaScript es el lenguaje más popular [8] y es reconocido como uno de los tres lenguajes nativos siendo estos (HTML, CSS y JavaScript), pudiendo trabajar los tres conjuntamente.

Desde hace unos años, se ha creado una versión capaz de ejecutarse también en el lado del servidor compitiendo en igualdad de condiciones con tecnologías back-end como PHP [9].

4.2.4. jQuery

jQuery es una librería de JavaScript [10], es de código abierto y, mediante plugins creados por desarrolladores de todo el mundo, permite agregar interactividad a un sitio web.

Estos plugins pueden ir desde una galería de imágenes a un menú desplegable con animaciones y su instalación es sencilla. Esto nos permite realizar acciones que se podrían hacer con JavaScript pero con muchas menos líneas de código [11].

4.2.5. Ajax

Ajax o *Asynchronous JavaScript and XML*, permite que las aplicaciones web trabajen de forma asíncrona realizando operaciones en segundo plano. Una aplicación que use Ajax podrá enviar y recuperar datos de un servidor sin la necesidad de tener que volver a cargar la página.

Estas funcionalidades otorgadas por Ajax pueden ser útiles para cosas como el autocompletado de una barra de búsqueda[12].

4.3. Tecnologías Back-end

En esta sección se hará un estudio sobre algunas de las diferentes tecnologías back-end existentes para poder argumentar finalmente la elección de las tecnologías usadas en este proyecto.

Las tecnologías back-end son las utilizadas en el lado del servidor, son las tecnologías que se usan para gestionar las diferentes peticiones realizadas por el usuario, operar con la base de datos y poder devolver la información al dispositivo para que, posteriormente, pueda verse gracias a las tecnologías front-end [5].

4.3.1. Java EE

Java EE o *Java Platform, Enterprise Edition*, es una plataforma de programación donde poder desarrollar aplicaciones programadas en lenguaje Java y que nos permite ejecutarlas sobre un servidor de aplicaciones.

Java EE tiene varias API como podrían ser JDBC (Java Database Connectivity), JPA (Java Persistence API) o JMS (Java Message Service). Además también dispone de tecnologías de servicios web como podrían ser JavaServer Pages, que nos permiten incrustar código Java en páginas escritas en HTML para crear páginas dinámicas, o servlets que nos ayudan a gestionar las solicitudes del cliente con el servidor [13,14].

4.3.2. PHP

PHP o *Hypertext Preprocessor* es un lenguaje de programación destinado a la programación y desarrollo de páginas o aplicaciones web dinámicas, favoreciendo la comunicación entre el servidor y la interfaz del usuario. Es un lenguaje de código abierto y, gracias a su comunidad, está en constante perfeccionamiento [15].

Algunas de sus ventajas son:

- Admite una gran cantidad de datos
- Es compatible con las principales bases de datos (MySQL, Oracle)
- Disminuye los tiempos de carga al ser un lenguaje más dinámico que la gran mayoría de las demás opciones posibles.

4.3.3. Python

Python es un lenguaje de programación que, al igual que Java, es utilizado para desarrollar aplicaciones de todo tipo. Sin embargo, al contrario que Java, se trata de un lenguaje interpretado por lo que no es necesario compilarlo antes de ejecutarlo, va siendo ejecutado a medida que va siendo interpretado [16].

Grandes compañías como Google, Spotify o Instagram usan Python para ofrecer sus servicios por varias de sus principales ventajas:

- Fácil de usar y leer: es un lenguaje relativamente sencillo, en el que se usa el inglés en gran parte, tiene una buena visualización y es muy fácil de leer.
- Gran biblioteca estándar: cuenta con una gran biblioteca de código que facilita la productividad en el desarrollo.
- Desarrollo rápido: es uno de los lenguajes más rápidos para el desarrollo.
- Codificación asíncrona: al haber puntos de contención, cada unidad de código se ejecuta por separado, por lo que te permite gestionar y resolver problemas de una manera mucho más sencilla y rápida.

4.4. Bases de datos

En esta sección se hará un estudio sobre algunas de las diferentes bases de datos existentes para poder argumentar finalmente la elección de base de datos utilizada en el desarrollo de este proyecto.

Una base de datos es una colección de información almacenada electrónicamente en un sistema informático [17].

4.4.1. Bases de datos relacionales

Las bases de datos relacionales son aquellas que relacionan otorga acceso a puntos de datos con relaciones entre sí. Esta información queda almacenada en tablas con filas y columnas, siendo cada columna un campo con un tipo de dato y cada fila un objeto de una determinada tabla.

Las tablas están unidas entre sí con distintos tipos de relación, ya sea de uno a uno, de uno a “n” o de “n” a “n” [18].

4.4.1.1. Oracle

Oracle es una base de datos relacional basada en el lenguaje SQL. Cuenta con varias versiones:

- Express Edition
- Standard Edition
- Enterprise Edition

Siendo la versión Express edition la única versión gratuita y por lo tanto, a pesar de tener menos prestaciones, teniendo en cuenta las dimensiones del proyecto, la única versión a contemplar.

Entre sus principales ventajas destaca su alta compatibilidad con aplicaciones y plataformas, su gran popularidad entre las grandes empresas y su gran comunidad [19].

4.4.1.2. MySQL

MySQL es una base de datos relacional de código abierto basada en el lenguaje SQL. También cuenta con una versión comercial gestionada por Oracle.

Entre sus principales ventajas destacan los desencadenantes, que viene a ser la posibilidad de automatizar acciones dentro de la base de datos, es decir, en el momento en el que se produce un evento, una acción es ejecutada. El hecho de ser una aplicación de código abierto también es una de sus mayores ventajas [20].

4.4.2. Bases de datos no relacionales

Una base de datos no relacional es aquella base de datos que no usa el esquema de tablas con filas y columnas y que a su vez están relacionadas entre sí. Los datos pueden estar almacenados en pares clave/valor, en documentos JSON o en grafos con bordes y vértices. Además son bases de datos NoSQL, lo que significa que no usan SQL como lenguaje de programación [21].

4.4.2.1. MongoDB

MongoDB es una base de datos no relacional, por lo tanto, NoSQL orientada a documentos de código abierto y escrita en C++. Los datos no son almacenados en tablas sino que son almacenados en documentos BSON, similares a los documentos JSON con un esquema dinámico [22].

4.4.2.2. Redis

Redis es un almacén de estructura de datos de valores de clave en memoria rápido y de código abierto.

Una de sus mayores ventajas es su facilidad de uso y su velocidad, es por ello por lo que es un tipo de base de datos elegida popularmente para juegos, aplicaciones web o aplicaciones móviles. Además es compatible con muchos lenguajes de programación como podrían ser Java, Python o C++ entre otros [23].

4.5. Tecnologías escogidas

En cuanto al front-end, la estructura de las páginas está hecha con HTML, siendo complementadas con CSS para otorgarles de diseño y estilo y de JavaScript para otorgarles interactividad. Además, se han usado plugins de JQuery para las tablas de datos de toda la aplicación otorgándole a estas la función de buscador, o la opción de poderlas descargar entre otras. También se ha usado Ajax para el autocompletado de formularios.

En cuanto al back-end, para el desarrollo de este proyecto se ha utilizado la plataforma Java EE (Java Enterprise Edition). Por lo tanto, el lenguaje de programación escogido ha sido Java, utilizando páginas JSP para poder crear páginas dinámicas incrustando código Java en ellas, servlets para gestionar las solicitudes del cliente con el servidor, y JPA para interactuar con la base de datos.

Además se ha utilizado Payara Server como servidor de aplicaciones, siendo este un servidor de aplicaciones de código abierto derivado de GlashFish una vez Oracle anunciara que suspendería su soporte comercial [24].

La base de datos escogida para la realización de este proyecto ha sido MySQL ya que es la que más se ajusta a las necesidades del proyecto. Esto se debe a que nuestra aplicación web tiene una estructura bien definida, no va a tener que soportar registros de nuevos usuarios, y a la hora de realizar operaciones con la base de datos que relacionen varias tablas o filas, las bases de datos relacionales son la mejor opción.

4.6. Aplicaciones web similares.

En esta sección vamos a estudiar las diferentes alternativas existentes en el mercado a la aplicación web que se ha desarrollado en este proyecto.

4.6.1. Playtomic

Playtomic [25] es, a día de hoy, la aplicación más importante a nivel de gestión de club en el mundo del pádel, podría decirse que se trata de un monopolio ya que se ha convertido en el único método por el cual ya muchos de los clubes de pádel del mundo permiten reservar sus pistas de pádel.

En cuanto al usuario de pádel, con Playtomic puedes reservar pistas en todos los clubes que estén trabajando con la aplicación, que actualmente son la inmensa mayoría, también puedes organizar partidos y, dependiendo del resultado vas obteniendo una valoración aproximada en cuanto a nivel de pádel, por último cabe destacar que tiene incorporado también funciones de red social donde poder hablar por chat con otros usuarios, poder subir fotos o comentar en publicaciones.

En cuanto a nivel de club, que es lo que interesa en este proyecto, Playtomic está principalmente centrado en la gestión de los alquileres del club, dejando un poco de lado lo que implica a las clases impartidas en el club o la tienda.

En la figura 10, se mostrará una imagen de la aplicación Playtomic a nivel de club.

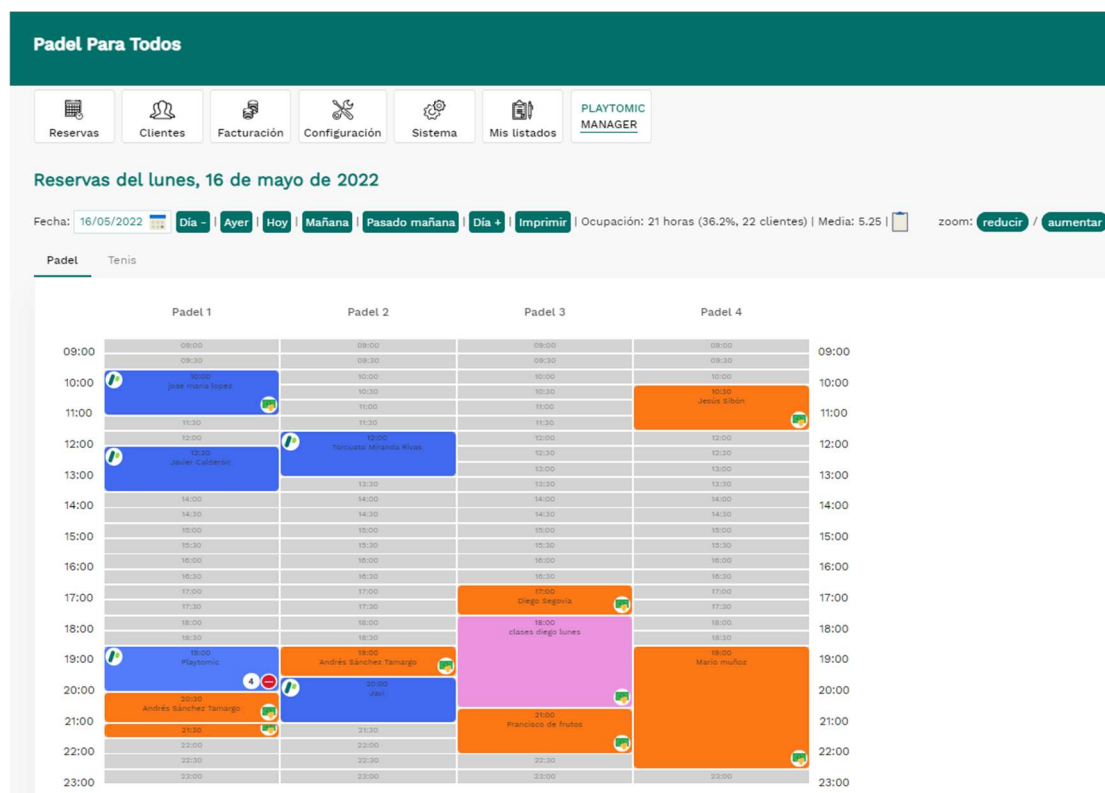


Figura 10: Visualización interna del club con Playtomic

En la página principal podemos observar el listado de reservas del club siendo las reservas de color azul las realizadas a través de la aplicación, las reservas naranjas las realizadas manualmente y las de color rosa las reservas automatizadas.

En la parte superior podemos observar las funciones que ofrece Playtomic:

- Reservas: se puede volver a la pantalla principal donde se visualizan las reservas y programar reservas automáticas.
- Clientes: se puede gestionar el listado de clientes y enviar notificaciones por SMS o email.
- Facturación: se puede llevar un control de los cobros de las pistas de pádel.
- Configuración: se pueden modificar las horas activas del club o los festivos.
- Sistema: se pueden dar de alta a los usuarios con privilegios para gestionar la aplicación interna de Playtomic.
- Mis listados: actualmente esta función está inutilizada, no tiene ninguna función.

Como se puede observar, no cubre las principales áreas que cubre nuestra proyecto, es decir, la gestión de las clases y la tienda del club.

4.6.2. Otras aplicaciones

Existen otras muchas aplicaciones capaces de gestionar un club, pero no son aplicaciones destinadas a las necesidades de un club de pádel sino aplicaciones dedicadas a una gestión más general de un centro deportivo como un gimnasio, una piscina municipal o un polideportivo.

Alguna de estas aplicaciones pueden ser:

- Nubapp: empresa con distintas aplicaciones para gestionar distintos ámbitos de un club deportivo. La única aplicación aplicable a un club de pádel ofrecida por Nubapp es la aplicación Resasports, aplicación centrada principalmente en la gestión de las pistas de pádel [26].
- Padel Manager [27]: aplicación similar a Nubapp y a Playtomic, destinada principalmente a la gestión de las pistas de pádel, incluyendo además herramientas para la creación de torneos o ligas internas en el club.

4.6.3. Comparación entre aplicaciones para gestionar un club

	ADMINCLUB (nuestro proyecto)	PLAYTOMIC	RESASPORTS	PADEL MANAGER
PISTAS	✗	✓	✓	✓
CLIENTES	✓	✓	✓	✓
TIENDA DE PRODUCTOS	✓	✗	✗	✓
FINANZAS	✓	✓	✓	✓
CLASES	✓	✗	✗	✗
TORNEOS	✗	✗	✗	✓

Tabla 6: Comparativa entre servicios ofrecidos por aplicaciones de pádel

Como se puede observar, nuestra aplicación AdminClub es la única aplicación que permite llevar una gestión de las clases de pádel de un club. Además es de las pocas que permiten gestionar la tienda, que es una parte indispensable en las ganancias de un club.

Cabe destacar que a la hora del análisis y diseño del proyecto no se incluyó la gestión de pistas de pádel ya que actualmente existe un monopolio dominado por la aplicación Playtomic.

5. Análisis del sistema

5.1. Descripción del problema

El objetivo de este proyecto es el diseño y la implementación de una aplicación web que permita gestionar internamente un club de pádel.

Para concretar el alcance y los objetivos a cumplir por esta aplicación se han estudiado las necesidades del club “Pádel para todos” en Guadarrama, dando como resultado unas necesidades de gestión principalmente de: productos, alumnos, clases y finanzas.

Una vez concretados los objetivos del proyecto, se ha realizado una especificación de requisitos que la aplicación debe cumplir para el correcto funcionamiento de la misma y para poder concluir con un resultado satisfactorio en la realización del proyecto.

A continuación se detallará esta especificación de requisitos:

5.2. Especificación de requisitos

En este apartado se detallará la especificación de requisitos, tanto funcionales como no funcionales, que son de necesario cumplimiento y nos ayudarán al diseño e implementación del proyecto.

Esta especificación de requisitos ha sido desarrollada a partir de las entrevistas previas con el cliente, donde se han establecidos las necesidades y alcance del proyecto.

IDENTIFICADOR: TR-AC-XX			
NOMBRE			
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN			
PRUEBAS ASOCIADAS			

Tabla 7: Plantilla de tabla de requisitos

La tabla anteriormente mostrada es la plantilla de especificación de requisitos. Esta plantilla está compuesta por los siguientes apartados:

- **Identificador:** Este campo es único e identifica a cada requisito. Su formato es TR-AC-XX, teniendo las diferentes siglas el siguiente significado:
 - TR: Tipo de requisito, pudiendo ser RF si se trata de un requisito funcional o RNF si se trata de un requisito no funcional.
 - AC: AdminClub, estas siglas no cambiarán en ninguno de los requisitos, hacen referencia al propio nombre de la aplicación desarrollada.
 - XX: Número de requisito.
- **Nombre:** Descripción breve y concisa del requisito.
- **Necesidad:** Determinará la importancia del requisito para el completo desarrollo del proyecto. Si un requisito tiene una necesidad esencial, esto significará que sin el correcto cumplimiento del mismo en el proyecto, el proyecto sería insatisfactorio. Sin embargo, un requisito con una necesidad opcional querría decir que, aunque este requisito no se hubiera podido cumplir, no afectaría al correcto funcionamiento del sistema.
- **Prioridad:** Determina, a grandes rasgos, el orden en el que deben desarrollarse los requisitos, siendo los de prioridad alta, los primeros en tener que desarrollarse. Esto se debe a que existen requisitos que pueden depender de otros para su desarrollo, por lo tanto, estos otros requisitos son requisitos de mayor prioridad.

- Complejidad: Valoración de la dificultad del desarrollo de un requisito.
- Descripción: Explicación detallada del requisito, explicando qué se debe cumplir y cómo debe cumplirse. Debe de estar explicado de la mejor y más clara manera posible.
- Pruebas asociadas: Listado de pruebas realizadas que confirman la satisfacción en el cumplimiento de un requisito. Las pruebas serán identificadas como PR-XX:
 - PR: Siglas de “prueba”.
 - XX: Número de la prueba.

El nivel de necesidad, prioridad o complejidad, será representado en la tabla de color lila.

5.2.1. Requisitos funcionales

Los requisitos funcionales, como su propio nombre indica, son todas aquellas funciones que debe de ser capaz de satisfacer la aplicación.

IDENTIFICADOR: RF-AC-01			
NOMBRE	Crear usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder crear un usuario, y debe seguir la siguiente estructura: • Nombre • Apellidos • Fecha de nacimiento • Género • Teléfono • Correo electrónico • Categoría (Alumno / Profesor) • Redes sociales (opcional)		
PRUEBAS ASOCIADAS	PR-AC-01, PR-AC-02, PR-AC-03, PR-AC-04, PR-AC-05, PR-AC-06, PR-AC-07, PR-AC-08		

Tabla 8: Requisito funcional 1.

IDENTIFICADOR: RF-AC-02			
NOMBRE	Modificar usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se deben poder modificar todos los datos de un usuario.		
PRUEBAS ASOCIADAS	PR-AC-09, PR-AC-10, PR-AC-11, PR-AC-12, PR-AC-13, PR-AC-14, PR-AC-15, PR-AC-16		

Tabla 9: Requisito funcional 2.

IDENTIFICADOR: RF-AC-03			
NOMBRE	Eliminar Usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder eliminar un usuario.		
PRUEBAS ASOCIADAS	PR-AC-17		

Tabla 10: Requisito funcional 3.

IDENTIFICADOR: RF-AC-04			
NOMBRE	Listado de usuarios		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se podrán consultar los usuarios en un listado con las siguientes columnas: • Nombre • Apellidos • Teléfono • Correo electrónico • Fecha de nacimiento • Género • Categoría El listado contará con las siguientes funciones: • Cuadro de búsqueda • Imprimir el listado • Exportar el listado a Excel • Elección de las columnas mostradas		
PRUEBAS ASOCIADAS	PR-AC-18		

Tabla 11: Requisito funcional 4.

IDENTIFICADOR: RF-AC-05			
NOMBRE	Crear producto		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder crear un producto, y debe seguir la siguiente estructura: • Nombre • Categoría: ○ Bebida / comida ○ Bolas ○ Palas ○ Paleteros ○ Ropa ○ Otros • Precio • Stock inicial		
PRUEBAS ASOCIADAS	PR-AC-19, PR-AC-20, PR-AC-21, PR-AC-22, PR-AC-23		

Tabla 12: Requisito funcional 5.

IDENTIFICADOR: RF-AC-06			
NOMBRE	Ver listado de productos		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se podrán vender y consultar los productos mediante un listado con las siguientes columnas: • Nombre • Categoría • Stock • Precio • Cantidad de venta • Género • Categoría El listado contará con las siguientes funciones: • Cuadro de búsqueda • Imprimir el listado • Exportar el listado a Excel • Elección de las columnas mostradas		
PRUEBAS ASOCIADAS	PR-AC-24		

Tabla 13: Requisito funcional 6.

IDENTIFICADOR: RF-AC-07			
NOMBRE	Eliminar producto		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder eliminar un producto.		
PRUEBAS ASOCIADAS	PR-AC-25		

Tabla 14: Requisito funcional 7.

IDENTIFICADOR: RF-AC-08			
NOMBRE	Modificar producto		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se deben poder modificar todos los campos de un producto.		
PRUEBAS ASOCIADAS	PR-AC-26, PR-AC-27, PR-AC-28, PR-AC-29, PR-AC-30		

Tabla 15: Requisito funcional 8.

IDENTIFICADOR: RF-AC-09			
NOMBRE	Vender producto		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder vender un producto introduciendo la cantidad de dicho producto y mostrándose un resumen de la venta con el precio final.		
PRUEBAS ASOCIADAS	PR-AC-31, PR-AC-32, PR-AC-33		

Tabla 16: Requisito funcional 9.

IDENTIFICADOR: RF-AC-10			
NOMBRE	Crear grupo de pádel		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder crear un grupo de pádel y debe incluir: • Profesor • Día • Hora		
PRUEBAS ASOCIADAS	PR-AC-34, PR-AC-35, PR-AC-36, PR-AC-37		

Tabla 17: Requisito funcional 10.

IDENTIFICADOR: RF-AC-11			
NOMBRE	Eliminar grupo de pádel		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder eliminar un grupo de pádel.		
PRUEBAS ASOCIADAS	PR-AC-38		

Tabla 18: Requisito funcional 11.

IDENTIFICADOR: RF-AC-12			
NOMBRE	Añadir alumno a grupo de pádel		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder inscribir a un alumno a un grupo de pádel previamente creado.		
PRUEBAS ASOCIADAS	PR-AC-39		

Tabla 19: Requisito funcional 12.

IDENTIFICADOR: RF-AC-13			
NOMBRE	Eliminar alumno de grupo de pádel		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder eliminar a un alumno de un grupo de pádel.		
PRUEBAS ASOCIADAS	PR-AC-40		

Tabla 20: Requisito funcional 13.

IDENTIFICADOR: RF-AC-14			
NOMBRE	Listado de grupos de pádel.		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se podrán consultar los grupos de pádel creados mediante un listado con las siguientes columnas: • Profesor • Día • Hora • Número de alumnos El listado contará con las siguientes funciones: • Cuadro de búsqueda • Imprimir el listado • Exportar el listado a Excel • Elección de las columnas mostradas		
PRUEBAS ASOCIADAS	PR-AC-41		

Tabla 21: Requisito funcional 14.

IDENTIFICADOR: RF-AC-15			
NOMBRE	Crear bono de clase particular		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder crear un bono de clases particulares indicando: • Cliente • Tipo de bono		
PRUEBAS ASOCIADAS	PR-AC-42		

Tabla 22: Requisito funcional 15.

IDENTIFICADOR: RF-AC-16			
NOMBRE	Marcar clase particular		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se deberá poder confirmar que se ha impartido una clase particular quedando registrada la fecha en la que se ha impartido la clase.		
PRUEBAS ASOCIADAS	PR-AC-43		

Tabla 23: Requisito funcional 16.

IDENTIFICADOR: RF-AC-17			
NOMBRE	Crear tarifa		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se podrán crear, a modo de notas, tarifas que ayuden al personal de recepción del club por si tuviesen alguna duda.		
PRUEBAS ASOCIADAS	PR-AC-44, PR-AC-45, PR-AC-46		

Tabla 24: Requisito funcional 17.

IDENTIFICADOR: RF-AC-18			
NOMBRE	Modificar tarifa		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder modificar una tarifa		
PRUEBAS ASOCIADAS	PR-AC-47, PR-AC-48, PR-AC-49		

Tabla 25: Requisito funcional 18.

IDENTIFICADOR: RF-AC-19			
NOMBRE	Eliminar tarifa		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder eliminar una tarifa		
PRUEBAS ASOCIADAS	PR-AC-50		

Tabla 26: Requisito funcional 19.

IDENTIFICADOR: RF-AC-20			
NOMBRE	Listado de tarifas		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se podrán consultar las tarifas mediante un listado con las siguientes columnas: • Nombre del producto • Precio El listado contará con las siguientes funciones: • Cuadro de búsqueda • Imprimir el listado • Exportar el listado a Excel • Elección de las columnas mostradas		
PRUEBAS ASOCIADAS	PR-AC-51		

Tabla 27: Requisito funcional 20.

IDENTIFICADOR: RF-AC-21			
NOMBRE	Listado de pagos		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se podrán consultar los pagos mediante un listado con las siguientes columnas: • Nombre del producto • Cliente • Precio • Fecha El listado contará con las siguientes funciones: • Cuadro de búsqueda • Imprimir el listado • Exportar el listado a Excel • Elección de las columnas mostradas		
PRUEBAS ASOCIADAS	PR-AC-52		

Tabla 28: Requisito funcional 21.

IDENTIFICADOR: RF-AC-22			
NOMBRE	Detalles de usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Deberá existir una página detallada con todos los datos del usuario: • Información personal • Grupos a los que pertenece • Bonos particulares a su nombre • Pagos realizados		
PRUEBAS ASOCIADAS	PR-AC-53		

Tabla 29: Requisito funcional 22.

IDENTIFICADOR: RF-AC-23			
NOMBRE	Listado de grupos asociados a un usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se deben poder consultar los grupos a los que pertenece un usuario.		
PRUEBAS ASOCIADAS	PR-AC-54		

Tabla 30: Requisito funcional 23.

IDENTIFICADOR: RF-AC-24			
NOMBRE	Listado de bonos asociados a un usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder consultar los bonos de particulares asociados a un usuario.		
PRUEBAS ASOCIADAS	PR-AC-55		

Tabla 31: Requisito funcional 24.

IDENTIFICADOR: RF-AC-25			
NOMBRE	Crear pago asociado a un usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se debe poder crear un pago asociado a un usuario y debe incluir: • Usuario • Asunto • Precio		
PRUEBAS ASOCIADAS	PR-AC-56		

Tabla 32: Requisito funcional 25.

IDENTIFICADOR: RF-AC-26			
NOMBRE	Listado de pagos asociados a un usuario		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Se deben poder consultar los pagos asociados a un usuario.		
PRUEBAS ASOCIADAS	PR-AC-57		

Tabla 33: Requisito funcional 26.

IDENTIFICADOR: RF-AC-27			
NOMBRE	Inicio de sesión		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Para acceder a la aplicación será necesario iniciar sesión previamente.		
PRUEBAS ASOCIADAS	PR-AC-58, PR-AC-59, PR-AC-60		

Tabla 34: Requisito funcional 27.

5.2.2. Requisitos no funcionales

Los requisitos no funcionales, a diferencia de los requisitos funcionales, no se basa en las funcionalidades que debe satisfacer la aplicación sino en las condiciones y restricciones que debe cumplir el sistema.

IDENTIFICADOR: RNF-AC-01			
NOMBRE	Registro		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Será necesario estar registrado y haber iniciado sesión para poder acceder a la aplicación.		

Tabla 35: Requisito no funcional 1

IDENTIFICADOR: RNF-AC-02			
NOMBRE	Compatibilidad en navegadores		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	El proyecto debe de ser compatible con los navegadores Mozilla Firefox y Google Chrome.		

Tabla 36: Requisito no funcional 2

IDENTIFICADOR: RNF-AC-03			
NOMBRE	El proyecto estará desarrollado en Java		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Al tratarse de una aplicación web, además de usar HTML o CSS, el proyecto estará desarrollado principalmente en lenguaje Java.		

Tabla 37: Requisito no funcional 3

IDENTIFICADOR: RNF-AC-04			
NOMBRE	Consistencia con la base de datos.		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	La información mostrada por la interfaz de usuario será consistente con la información existente en la base de datos.		

Tabla 38: : Requisito no funcional 4

IDENTIFICADOR: RNF-AC-05			
NOMBRE	Gestor de base de datos MySQL.		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	La base de datos a utilizar será una base de datos relacional utilizando el gestor MySQL.		

Tabla 39: Requisito no funcional 5

IDENTIFICADOR: RNF-AC-06			
NOMBRE	Permisos de acceso a la aplicación.		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	Los permisos de acceso a la aplicación podrán ser solamente otorgados por el administración de la aplicación, no existirá opción de registrarse manualmente.		

Tabla 40: Requisito no funcional 6

IDENTIFICADOR: RNF-AC-07			
NOMBRE	Diseño responsive		
NECESIDAD	Esencial	Deseable	Opcional
PRIORIDAD	Alta	Media	Baja
COMPLEJIDAD	Alta	Media	Baja
DESCRIPCIÓN	La aplicación debe poseer un diseño responsive con el objetivo de poder ser visualizada correctamente en cualquier dispositivo.		

Tabla 41: Requisito no funcional 7

5.3. Matriz de trazabilidad

En esta sección se va a mostrar la matriz de trazabilidad entre los requisitos y las pruebas realizadas de tal forma que se pueda comprobar fácilmente que todos los requerimientos propuestos para el programa han sido comprobados y cumplidos.

	PR-AC-01	PR-AC-02	PR-AC-03	PR-AC-04	PR-AC-05	PR-AC-06	PR-AC-07	PR-AC-08	PR-AC-09	PR-AC-10	PR-AC-11	PR-AC-12	PR-AC-13	PR-AC-14	PR-AC-15	PR-AC-16	PR-AC-17	PR-AC-18	PR-AC-19	PR-AC-20
RF-AC-01																				
RF-AC-02																				
RF-AC-03																				
RF-AC-04																				
RF-AC-05																				
RF-AC-06																				
RF-AC-07																				
RF-AC-08																				
RF-AC-09																				
RF-AC-10																				
RF-AC-11																				
RF-AC-12																				
RF-AC-13																				
RF-AC-14																				
RF-AC-15																				
RF-AC-16																				
RF-AC-17																				
RF-AC-18																				
RF-AC-19																				
RF-AC-20																				
RF-AC-21																				
RF-AC-22																				
RF-AC-23																				
RF-AC-24																				
RF-AC-25																				
RF-AC-26																				
RF-AC-27																				

Tabla 42: Matriz de trazabilidad de requisitos 1

	PR-AC-21	PR-AC-22	PR-AC-23	PR-AC-24	PR-AC-25	PR-AC-26	PR-AC-27	PR-AC-28	PR-AC-29	PR-AC-30	PR-AC-31	PR-AC-32	PR-AC-33	PR-AC-34	PR-AC-35	PR-AC-36	PR-AC-37	PR-AC-38	PR-AC-39	PR-AC-40
RF-AC-01																				
RF-AC-02																				
RF-AC-03																				
RF-AC-04																				
RF-AC-05																				
RF-AC-06																				
RF-AC-07																				
RF-AC-08																				
RF-AC-09																				
RF-AC-10																				
RF-AC-11																				
RF-AC-12																				
RF-AC-13																				
RF-AC-14																				
RF-AC-15																				
RF-AC-16																				
RF-AC-17																				
RF-AC-18																				
RF-AC-19																				
RF-AC-20																				
RF-AC-21																				
RF-AC-22																				
RF-AC-23																				
RF-AC-24																				
RF-AC-25																				
RF-AC-26																				
RF-AC-27																				

Tabla 43: Matriz de trazabilidad de requisitos 2

	PR-AC-41	PR-AC-42	PR-AC-43	PR-AC-44	PR-AC-45	PR-AC-46	PR-AC-47	PR-AC-48	PR-AC-49	PR-AC-50	PR-AC-51	PR-AC-52	PR-AC-53	PR-AC-54	PR-AC-55	PR-AC-56	PR-AC-57	PR-AC-58	PR-AC-59	PR-AC-60
RF-AC-01																				
RF-AC-02																				
RF-AC-03																				
RF-AC-04																				
RF-AC-05																				
RF-AC-06																				
RF-AC-07																				
RF-AC-08																				
RF-AC-09																				
RF-AC-10																				
RF-AC-11																				
RF-AC-12																				
RF-AC-13																				
RF-AC-14																				
RF-AC-15																				
RF-AC-16																				
RF-AC-17																				
RF-AC-18																				
RF-AC-19																				
RF-AC-20																				
RF-AC-21																				
RF-AC-22																				
RF-AC-23																				
RF-AC-24																				
RF-AC-25																				
RF-AC-26																				
RF-AC-27																				

Tabla 44: Matriz de trazabilidad de requisitos

6. Diseño del sistema

En esta sección se hará un análisis sobre el diseño final del sistema, tanto a nivel de arquitectura interna de la aplicación como del diseño de la base de datos.

6.1. Arquitectura de la aplicación.

En esta sección se explicará cuál ha sido la arquitectura interna de la aplicación.

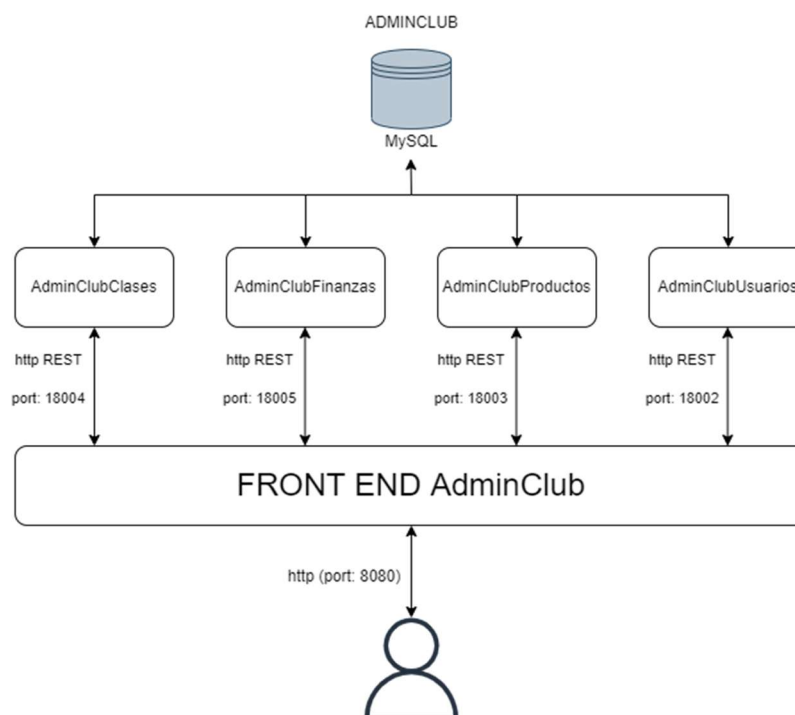


Figura 11: Arquitectura del sistema

Como se puede observar en la figura 11, la arquitectura elegida para el desarrollo de la aplicación es una arquitectura basada en microservicios.

Un microservicio es un estilo o modo de programar que consiste en dividir las distintas partes del proyecto en bloques, según las funcionalidades de las que se encargan. La idea es que estos microservicios puedan trabajar conjuntamente para el completo funcionamiento de la aplicación pero que, a su vez, si un microservicio falla, el resto de la aplicación siga funcionando correctamente.

La aplicación consta de un front-end, es decir, una interfaz de usuario a la que se accede por el puerto local 8080, cuatro microservicios comunicados por diferentes puertos a este front-end y una base de datos que se comunica también con estos cuatro microservicios.

A continuación se explicará de que se ocupa cada microservicio:

- **AdminClubClases:** el microservicio AdminClubClases, se comunica con el front-end por el puerto 18004 y se encarga de gestionar tanto las clases semanales como los bonos particulares del club. Entre sus funciones se pueden encontrar la creación de una clase, la creación de un bono o sus modificaciones y la eliminación de las mismas.
- **AdminClubFinanzas:** el microservicio AdminClubFinanzas se comunica con el front-end por el puerto 18005 y se encarga de gestionar todos los pagos realizados en el club, ya sean pagos de clases, de bonos particulares, de productos o de cualquier otro servicio ofrecido por el club.
- **AdminClubProductos:** el microservicio AdminClubProductos se comunica con el front-end por el puerto 18003 y se encarga de gestionar los productos de la tienda del club. Entre sus funciones se pueden encontrar la creación de un producto, la modificación del mismo o de su stock y su eliminación.
- **AdminClubUsuarios:** el microservicio AdminClubUsuarios se comunica con el front-end por el puerto 18002 y se encarga de gestionar a los usuarios, tanto alumnos como profesores, del club. Entre sus funciones podemos destacar la creación, modificación o eliminación de los usuarios o la recopilación de toda la información del usuario, clases en las que está inscrito o pagos que ha realizado.

6.2. Diseño de la base de datos.

A continuación se va a realizar un análisis del diseño de la base de datos, hablando de los problemas que se han presentado a lo largo del diseño y de las decisiones finales tomadas para conseguir un correcto funcionamiento de la aplicación pudiendo cumplir cada uno de los requisitos propuestos en la especificación de requisitos.

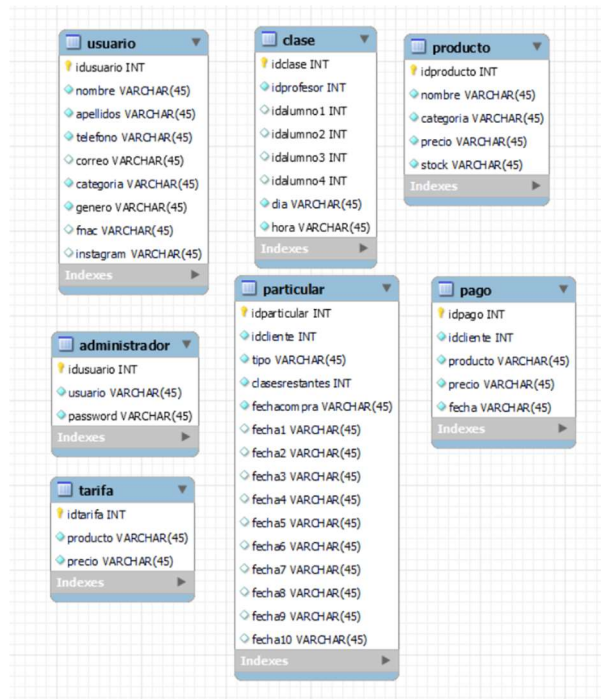


Figura 12: Diseño de la base de datos

En la figura 12, se puede observar el diseño final de la base de datos.

Como se puede ver, las tablas no están relacionadas entre sí. Esto se debe a la gran cantidad de errores y problemas que estaban surgiendo a la hora de actualizar tablas que dependían de otras, a la hora de realizar consultas complejas o a la hora de eliminar filas.

Debido a estos problemas, se tomó la decisión de no relacionar las tablas y suplir estas relaciones mediante el uso del front-end y el back-end de la aplicación.

Para controlar los campos obligatorios u opcionales se usó el atributo “required” en los formularios de tal forma que, antes de poder insertar una entidad en la base de datos, se comprobase, a nivel de front-end, que todos los campos obligatorios estaban correctamente introducidos.

Para poder controlar consultas que involucraran varias tablas, actualizaciones o borrados en cascada, se usó el back-end, programando en Java, para suplir la lógica de las relaciones.

A continuación se explicarán cada una de las tablas de la aplicación.

6.2.1. Tabla administrador

La tabla administrador recoge los datos de acceso de un administrador de un club. Estas cuentas de acceso serán entregadas por el administración de la aplicación una vez se hayan contratado los servicios de nuestra aplicación AdminClub. Consta de lo siguiente:

- **Idusuario:** número identificador de un administrador, es su clave primaria.
- **Usuario:** nombre de usuario a introducir para poder acceder a la aplicación.
- **Password:** contraseña a introducir para poder acceder a la aplicación.

6.2.2. Tabla usuario

La tabla usuario recoge todos los datos de los alumnos y profesores del club. Estos datos serán introducidos por el administrador usando los formularios de creación de usuarios. Consta de lo siguiente:

- **Idusuario:** número identificador de un usuario, es su clave primaria.
- **Nombre:** nombre de un usuario.
- **Apellidos:** apellidos de un usuario.
- **Teléfono:** teléfono de un usuario.
- **Correo:** correo electrónico de un usuario, se comprobará el formato del correo mediante el front-end.
- **Categoría:** categoría del usuario pudiendo ser alumno o profesor.
- **Género:** género del usuario pudiendo ser hombre o mujer.
- **Fnac:** fecha de nacimiento del usuario (opcional).
- **Instagram:** Instagram del usuario (opcional).

6.2.3. Tabla clase

La tabla clase recoge los datos necesarios para poder gestionar una clase de pádel, siendo una clase de pádel una clase que se repite semanalmente. Se compone de lo siguiente:

- **Idclase:** número identificador de una clase, es su clave primaria.
- **Idprofesor:** número identificador del profesor que imparte la clase, se corresponde con el “idusuario” de un usuario con “categoría: profesor”.
- **Idalumno(1-4):** número identificador de un alumno inscrito a una clase, el número de alumnos puede variar entre cero y cuatro, se corresponde con el “idusuario” de un usuario con “categoría: alumno”.
- **Día:** día de la semana en la que se imparte la clase, pudiendo ser de lunes a domingo.
- **Hora:** hora a la que se imparte la clase pudiendo ser desde las 9:00 hasta las 23:00.

6.2.4. Tabla particular

La tabla particular recoge los datos necesarios para poder gestionar un bono de clases particulares, estas clases pueden ser impartidas en cualquier momento sin tener un día y hora preestablecidos. Se compone de lo siguiente:

- **Idparticular:** número identificador de una particular, es su clave primaria.
- **Idcliente:** número identificador del cliente que ha adquirido el bono, se corresponde con el “idusuario” de un usuario.
- **Tipo:** tipo de bono, pudiendo ser cuatro o diez.
- **Clasesrestantes:** número de clases restantes del bono, siendo cuatro o diez al momento de adquirir el bono y yéndose restando de una en una cada vez que se haya impartido una clase.
- **Fecha compra:** fecha de adquisición del bono.
- **Fecha(1-10):** fecha en la que se ha impartido una clase, será controlado si el máximo de fechas rellenables son cuatro o diez dependiendo del tipo de bono.

6.2.5. Tabla producto

La tabla producto recoge los datos necesarios para poder gestionar los productos del club y su posterior posible venta. Se compone de lo siguiente:

- **Idproducto:** número identificador de un producto, es su clave primaria.
- **Nombre:** nombre del producto.
- **Categoría:** categoría del producto, pudiendo ser:
 - Bebida / comida
 - Bolas
 - Palas
 - Paleteros
 - Ropa
 - Otros
- **Precio:** valor numérico del precio del producto, puede contener decimales.
- **Stock:** cantidad del producto disponible por el club, debe de ser un número mayor a cero.

6.2.6. Tabla tarifa

La tabla tarifa recoge los datos de precios de distintos servicios que ofrece el club. Consta de lo siguiente:

- **Idtarifa:** número identificador de una tarifa, es su clave primaria.
- **Idcliente:** nombre del producto o servicio.
- **Producto:** valor numérico del precio del producto o servicio, puede contener decimales.

6.2.7. Tabla pago

La tabla pago recoge todos los datos que tienen que ver con los pagos que se han realizado en el club ya sean de productos, clases, particulares o servicios ofrecidos. Consta de lo siguiente:

- **Idpago:** número identificador de un pago, es su clave primaria.
- **Idcliente:** nombre del cliente que ha hecho la compra si se trata de una compra de clases.
- **Producto:** nombre del producto o servicio.
- **Precio:** valor numérico del precio del producto o servicio pagado, puede contener decimales.
- **Fecha:** fecha de compra del producto o servicio.

7. Implementación del sistema

En esta sección se explicará cómo se ha implementado el sistema y, por lo tanto, cada uno de sus handlers. Para ello se ha elaborado una tabla con cada uno de los handlers implementados explicando con qué microservicio o microservicios trabaja, qué funciones realiza, su JSP de entrada y de salida y una pequeña descripción explicando de qué se encarga cada handler.

Los handlers o *manejadores*, son los encargados de gestionar la información recibida y llamar a cada una de las funciones necesarias que se encarguen de realizar las operaciones que son pedidas por el usuario.

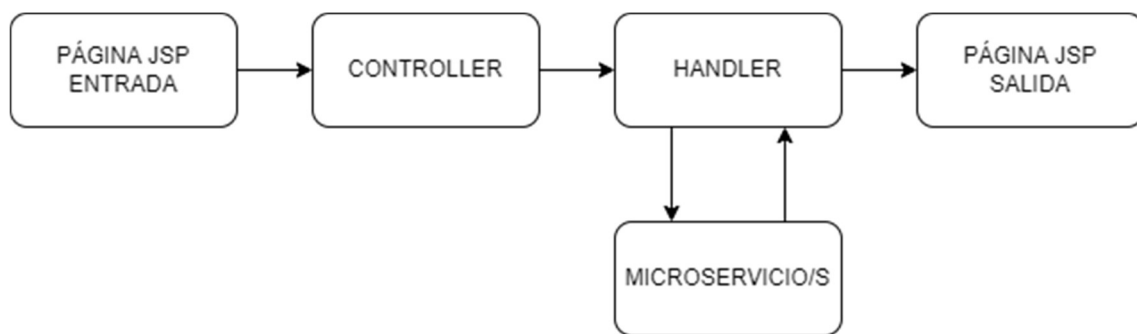


Figura 13: Funcionamiento interno de la aplicación

En la figura 13 se puede observar cómo funcionaría la aplicación a la hora de satisfacer una petición de un usuario:

1. El usuario, desde una página JSP de entrada, realizaría una petición. Esa petición estaría compuesta por el nombre del handler que debe satisfacerla y los datos necesarios para ello.
2. El controller recibiría la petición y se encargaría de redireccionar todos esos datos al handler correspondiente.
3. El handler rescata los datos que le han sido proporcionados y realiza las operaciones previas pertinentes en caso de ser necesario. Una vez hecho esto, llama a los microservicios correspondientes especificando las funciones necesarias para que el microservicio se encargue de realizar las operaciones con la base de datos.
4. El microservicio, una vez realizadas las operaciones pedidas, devuelve una respuesta de confirmación de la operación y, en caso de haberlos, unos datos como resultado de la operación realizada, al handler.
5. Por último, en caso de haberlos, el handler se encarga de devolver los datos que le ha proporcionado el microservicio a una nueva página JSP de salida.

HANDLER	MICROSERVICIO/S (FUNCION/ES)	JSP ENTRADA	JSP SALIDA
ActualizarProductoHandler	AdminClubProductos (/producto, /productos)	modProductoForm.jsp	home.jsp
	<u>DESCRIPCIÓN</u> Actualiza un producto al haber sido modificado y devuelve una lista de los productos actualizada.		
AddAlGrupoFormHandler	- AdminClubClases (/grupo) - AdminClubUsuarios (/alumnos)	detallesgrupo.jsp	addalgrupoform.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de alumnos al formulario que se encarga de añadir un alumno a un grupo.		
AddAlGrupoHandler	- AdminClubClases (/grupo, /addalgrupo) - AdminClubUsuarios (/profesores, /usuarios)	addalgrupoform.jsp	detallesgrupo.jsp
	<u>DESCRIPCIÓN</u> Se encarga de inscribir un usuario a un grupo.		
AddFechaClaseHandler	- AdminClubClases (/particular, /addfechaparticular) - AdminClubUsuarios (/profesores, /usuarios)	detallesparticular.jsp	detallesparticular.jsp
	<u>DESCRIPCIÓN</u> Agrega a un bono de clases particulares la fecha en la que ha sido dada una clase particular y actualiza las clases particulares restantes de ese bono.		
addGrupoHandler	- AdminClubClases (/creargrupo, /grupos) - AdminClubUsuarios (/profesores)	addgrupo.jsp	grupos.jsp
	<u>DESCRIPCIÓN</u> Crea un grupo y devuelve la lista de todos los grupos actualizada		

Tabla 45: Descripción Handlers 1

HANDLER	MICROSERVICIO/S (FUNCION/ES)	JSP ENTRADA	JSP SALIDA
ConfirmarVentaHandler	- AdminClubProductos (/producto, /productos) - AdminClubFinanzas (/crearPago)	venderProducto.jsp venderParticular.jsp	home.jsp
	<u>DESCRIPCIÓN</u> Confirma cualquier venta de la aplicación y la deja registrada en la base de datos.		
CrearGrupoHandler	AdminClubUsuarios (/profesores)	grupos.jsp	addGrupo.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de profesores al formulario que se encarga de crear un nuevo grupo de clases.		
CrearParticularFormHandler	AdminClubUsuarios (/usuarios)	particulares.jsp	addParticular.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de clientes al formulario que se encarga de crear una nueva clase particular.		
CrearParticularHandler	- AdminClubClases (/crearparticular, /particulares) - AdminClubFinanzas (/crearPago) - AdminClubUsuarios (/usuarios)	addParticular.jsp	particulares.jsp
	<u>DESCRIPCIÓN</u> Crea un bono de clases particulares, registra el pago y devuelve una lista actualizada de los bonos particulares.		
CrearProductoHandler	AdminClubProductos (/crearProducto, /productos)	crearProducto.jsp	home.jsp
	<u>DESCRIPCIÓN</u> Crea un producto y devuelve la lista de productos actualizada.		
CrearTarifaHandler	AdminClubFinanzas (/crearTarifa, /tarifas)	creartarifa.jsp	tarifas.jsp
	<u>DESCRIPCIÓN</u> Crea una tarifa y devuelve la lista de tarifas actualizada.		

Tabla 46: Descripción Handlers 2

HANDLER	MICROSERVICIO/S (FUNCION/ES)	JSP ENTRADA	JSP SALIDA
DetallesGrupoHandler	- AdminClubClases (/grupo) - AdminClubUsuarios (/profesores, /usuarios)	detallesUsuario.jsp grupos.jsp	destallesGrupo.jsp
	<u>DESCRIPCIÓN</u> Proporciona toda la información referente a un grupo de clases específico.		
DetallesParticularHandler	- AdminClubClases (/particular) - AdminClubUsuarios (/usuario)	detallesUsuario.jsp particulares.jsp	detallesParticular.jsp
	<u>DESCRIPCIÓN</u> Proporciona toda la información referente a una clase particular específica.		
DetallesUsuarioHandler	- AdminClubUsuarios (/usuario, /usuarios, /profesores) - AdminClubClases (/particularId, /grupoid) - AdminClubFinanzas (/pagos)	buscarUsuario.jsp	detallesUsuario.jsp
	<u>DESCRIPCIÓN</u> Proporciona toda la información relevante respecto a un usuario: información personal, grupos en los que está inscrito, bonos de clases particulares a su nombre y pagos realizados por el usuario.		
EliminarDelGrupoHandler	- AdminClubClases (/grupo, /eliminarDelGrupo) - AdminClubUsuarios (/usuarios, /profesores)	detallesGrupo.jsp	detallesGrupo.jsp
	<u>DESCRIPCIÓN</u> Elimina a un alumno de un grupo y actualiza la vista de los detalles del grupo.		

Tabla 47: Descripción Handlers 3

HANDLER	MICROSERVICIO/S (FUNCION/ES)	JSP ENTRADA	JSP SALIDA
EliminarGrupoHandler	- AdminClubClases (/grupos, /eliminarGrupo) - AdminClubUsuarios (/profesores)	detallesGrupo.jsp	grupos.jsp
	<u>DESCRIPCIÓN</u> Elimina un grupo y devuelve la lista de grupos actualizada.		
EliminarProductoHandler	AdminClubProductos (/eliminarProducto, /productos)	modProductoForm.jsp	home.jsp
	<u>DESCRIPCIÓN</u> Elimina un producto y devuelve la lista de productos actualizada.		
EliminarTarifaHandler	AdminClubFinanzas (/eliminarTarifa, /tarifas)	modTarifa.jsp	tarifas.jsp
	<u>DESCRIPCIÓN</u> Elimina una tarifa y devuelve la lista de tarifas actualizada.		
EliminarUsuarioHandler	AdminClubUsuarios (/deleteUsuario, /usuarios)	detallesUsuario.jsp	buscarUsuario.jsp
	<u>DESCRIPCIÓN</u> Elimina un usuario y devuelve la lista de usuarios actualizada.		
GuardarUsuarioHandler	AdminClubUsuarios (/guardarUsuario, /usuarios)	addUsuario.jsp	buscarUsuario.jsp
	<u>DESCRIPCIÓN</u> Crea un producto y devuelve la lista de productos actualizada.		
LoginHandler	- AdminClubUsuarios (/login) - AdminClubProductos (/productos)	start.jsp	home.jsp
	<u>DESCRIPCIÓN</u> Inicia sesión en la aplicación y devuelve la lista de productos que se cargan en la página inicial.		

Tabla 48: Descripción Handlers 4

HANDLER	MICROSERVICIO/S (FUNCION/ES)	JSP ENTRADA	JSP SALIDA
ModProductoFormHandler	AdminClubProductos (/producto)	modProducto.jsp	modProductoForm.jsp
	<u>DESCRIPCIÓN</u> Proporciona los detalles de un producto para poder tener una vista previa, sobre los datos actuales, antes de modificarlos.		
ModProductoHandler	AdminClubProductos (/producto)	TODOS	modProducto.jsp
	<u>DESCRIPCIÓN</u> Proporciona una lista de productos para poder ser modificados.		
ModTarifaFormHandler	AdminClubFinanzas (/tarifa)	tarifas.jsp	modTarifa.jsp
	<u>DESCRIPCIÓN</u> Proporciona los detalles de una tarifa para poder tener una vista previa, sobre los datos actuales, antes de modificarlos.		
ModTarifaHandler	AdminClubFinanzas (/tarifa, /tarifas)	modTarifa.jsp	tarifas.jsp
	<u>DESCRIPCIÓN</u> Modifica una tarifa y devuelve la lista de tarifas actualizada.		
ModUsuarioHandler	AdminClubUsuarios (/usuario, /usuarios)	detallesUsuario.jsp	buscarUsuario.jsp
	<u>DESCRIPCIÓN</u> Modifica un usuario y devuelve la lista de usuarios actualizada.		
VenderProductoHandler	AdminClubProductos (/productos)	TODOS	home.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de productos para que se muestren y se puedan vender.		

Tabla 49: Descripción Handlers 5

HANDLER	MICROSERVICIO/S (FUNCION/ES)	JSP ENTRADA	JSP SALIDA
VentaProductoHandler	AdminClubProductos (/producto)	home.jsp	venderproducto.jsp
	<u>DESCRIPCIÓN</u> Obtiene el producto, la cantidad y el precio y proporciona los datos a una página resumen de la venta.		
verGruposHandler	- AdminClubClases (/grupos) - AdminClubUsuarios (/profesores)	TODOS	grupos.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de grupos.		
VerPagosHandler	- AdminClubFinanzas (/tarifa) - AdminClubUsuarios (/usuarios)	TODOS	cuentas.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de pagos.		
VerParticularesHandler	- AdminClubClases (/particulares) - AdminClubUsuarios (/usuarios)	TODOS	particulares.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de bonos particulares.		
VerTarifasHandler	AdminClubFinanzas (/tarifas)	TODOS	tarifas.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de tarifas.		
VerUsuariosHandler	AdminClubUsuarios (/usuarios)	TODOS	buscarUsuario.jsp
	<u>DESCRIPCIÓN</u> Proporciona la lista de usuarios.		

Tabla 50: Descripción Handlers 6

8. Evaluación del sistema

En esta sección se va a evaluar el ajuste del sistema a la especificación de requisitos propuesta inicialmente. Para llevar a cabo esta evaluación se han desarrollado una serie de pruebas que comprueben cada uno de estos requisitos.

La plantilla de cada una de estas pruebas es la siguiente:

IDENTIFICADOR: PR-AC-XX	
NOMBRE	
DESCRIPCIÓN	
RESULTADO ESPERADO	
RESULTADO OBTENIDO	
EVALUACIÓN	
REQUISITOS ASOCIADOS	

Tabla 51: Plantilla tabla de pruebas

La tabla anteriormente mostrada es la plantilla de pruebas. Esta plantilla está compuesta por los siguientes apartados:

- Identificador: Este campo es único e identifica a cada requisito. Su formato es PR-AC-XX, teniendo las diferentes siglas el siguiente significado:
 - PR: Indicando que se trata de una prueba.
 - AC: AdminClub, estas siglas no cambiarán en ninguno de los requisitos, hacen referencia al propio nombre de la aplicación desarrollada.
 - XX: Número de prueba.
- Nombre: Descripción breve y concisa de la prueba.
- Descripción: Explicación detallada de la prueba.

- Resultado esperado: Resultado que se espera obtener tras la realización de la prueba.
- Resultado obtenido: Resultado obtenido tras la realización de la prueba.
- Evaluación: Tras comparar el resultado esperado y el resultado obtenido, se llega a la conclusión de si la prueba ha sido superada con éxito o ha sido insatisfactoria.

Las pruebas realizadas para comprobar el correcto funcionamiento del sistema son las siguientes:

IDENTIFICADOR: PR-AC-01	
NOMBRE	Crear un usuario correctamente.
DESCRIPCIÓN	Se debe poder crear un usuario rellenando todos los datos solicitados en el formulario y que este usuario quede registrado en la base de datos.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario y pulsar en “Crear Usuario”, el usuario es creado e insertado en la base de datos.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario y pulsar en “Crear Usuario”, el usuario es creado e insertado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 52: Prueba 1

IDENTIFICADOR: PR-AC-02	
NOMBRE	Crear un usuario con el campo “nombre” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “nombre” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “nombre” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “nombre”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “nombre” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “nombre”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 53: Prueba 2

IDENTIFICADOR: PR-AC-03	
NOMBRE	Crear un usuario con el campo “apellidos” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “apellidos” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “apellidos” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “apellidos”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “apellidos” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “apellidos”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 54: Prueba 3

IDENTIFICADOR: PR-AC-04	
NOMBRE	Crear un usuario con el campo “teléfono” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “teléfono” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “teléfono” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “teléfono”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “teléfono” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “teléfono”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 55: Prueba 4

IDENTIFICADOR: PR-AC-05	
NOMBRE	Crear un usuario con el campo “género” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “género” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “género” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “género”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “género” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “género”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 56: Prueba 5

IDENTIFICADOR: PR-AC-06	
NOMBRE	Crear un usuario con el campo “correo electrónico” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “correo electrónico” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “correo electrónico” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “correo electrónico”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “correo electrónico” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “correo electrónico”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 57: Prueba 6

IDENTIFICADOR: PR-AC-07	
NOMBRE	Crear un usuario con el campo “categoría” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “categoría” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “categoría” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “categoría”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “categoría” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “categoría”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 58: Prueba 7

IDENTIFICADOR: PR-AC-08	
NOMBRE	Crear un usuario con el campo “fecha de nacimiento” vacío.
DESCRIPCIÓN	Crear un usuario con el campo “fecha de nacimiento” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario, dejando el campo “fecha de nacimiento” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “fecha de nacimiento”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario, dejando el campo “fecha de nacimiento” vacío y pulsar en “Crear Usuario”, el usuario no puede ser creado y se nos pide rellenar el campo “fecha de nacimiento”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-01

Tabla 59: Prueba 8

IDENTIFICADOR: PR-AC-09	
NOMBRE	Modificar un usuario correctamente.
DESCRIPCIÓN	Se debe poder modificar un usuario modificando cada uno de los datos solicitados en el formulario y que este usuario quede actualizado en la base de datos.
RESULTADO ESPERADO	Tras modificarse los datos de un usuario y pulsar en “Modificar Usuario”, el usuario es modificado en la base de datos.
RESULTADO OBTENIDO	Tras modificarse los datos de un usuario y pulsar en “Modificar Usuario”, el usuario es modificado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 60: Prueba 9

IDENTIFICADOR: PR-AC-10	
NOMBRE	Modificar un usuario con el campo “nombre” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “nombre” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “nombre” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “nombre”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “nombre” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “nombre”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 61: Prueba 10

IDENTIFICADOR: PR-AC-11	
NOMBRE	Modificar un usuario con el campo “apellidos” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “apellidos” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “apellidos” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “apellidos”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “apellidos” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “apellidos”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 62: Prueba 11

IDENTIFICADOR: PR-AC-12	
NOMBRE	Modificar un usuario con el campo “teléfono” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “teléfono” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “teléfono” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “teléfono”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “teléfono” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “teléfono”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 63: Prueba 12

IDENTIFICADOR: PR-AC-13	
NOMBRE	Modificar un usuario con el campo “género” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “género” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “género” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “género”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “género” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “género”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 64: Prueba 13

IDENTIFICADOR: PR-AC-14	
NOMBRE	Modificar un usuario con el campo “correo electrónico” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “correo electrónico” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “correo electrónico” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “correo electrónico”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “correo electrónico” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “correo electrónico”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 65: Prueba 14

IDENTIFICADOR: PR-AC-15	
NOMBRE	Modificar un usuario con el campo “categoría” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “categoría” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “categoría” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “categoría”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “categoría” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “categoría”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 66: Prueba 15

IDENTIFICADOR: PR-AC-16	
NOMBRE	Modificar un usuario con el campo “fecha de nacimiento” vacío.
DESCRIPCIÓN	Modificar un usuario con el campo “fecha de nacimiento” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un usuario, dejando el campo “fecha de nacimiento” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “fecha de nacimiento”.
RESULTADO OBTENIDO	Al intentar modificar un usuario, dejando el campo “categoría” vacío y pulsar en “Actualizar Usuario”, el usuario no puede ser modificado y se nos pide rellenar el campo “categoría”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-02

Tabla 67: Prueba 16

IDENTIFICADOR: PR-AC-17	
NOMBRE	Eliminar un usuario
DESCRIPCIÓN	Buscar un usuario ya creado y eliminarlo
RESULTADO ESPERADO	Al hacer pulsar en “Eliminar usuario” el usuario es eliminado de la base de datos y, por lo tanto, eliminado de la lista de usuarios de la aplicación.
RESULTADO OBTENIDO	Al hacer pulsar en “Eliminar usuario” el usuario es eliminado de la base de datos y, por lo tanto, eliminado de la lista de usuarios de la aplicación.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-03

Tabla 68: Prueba 17

IDENTIFICADOR: PR-AC-18	
NOMBRE	Ver listado de usuarios
DESCRIPCIÓN	Comprobar que el listado de usuarios corresponde con los usuarios que hay dados de alta en la base de datos.
RESULTADO ESPERADO	Al pulsar en “Buscar usuario”, el listado de usuarios corresponde con el listado de usuarios que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-03, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
RESULTADO OBTENIDO	Al pulsar en “Buscar usuario”, el listado de usuarios corresponde con el listado de usuarios que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-03, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-04

Tabla 69: Prueba 18

IDENTIFICADOR: PR-AC-19	
NOMBRE	Crear un producto correctamente.
DESCRIPCIÓN	Se debe poder crear un producto rellenando todos los datos solicitados en el formulario y que este producto quede registrado en la base de datos.
RESULTADO ESPERADO	Tras rellenarse los datos de un usuario y pulsar en “Subir producto”, el producto es creado e insertado en la base de datos.
RESULTADO OBTENIDO	Tras rellenarse los datos de un usuario y pulsar en “Subir producto”, el producto es creado e insertado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-05

Tabla 70: Prueba 19

IDENTIFICADOR: PR-AC-20	
NOMBRE	Crear un producto con el campo “nombre” vacío.
DESCRIPCIÓN	Crear un producto con el campo “nombre” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un producto, dejando el campo “nombre” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “nombre”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un producto, dejando el campo “nombre” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “nombre”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-05

Tabla 71: Prueba 20

IDENTIFICADOR: PR-AC-21	
NOMBRE	Crear un producto con el campo “categoría” vacío.
DESCRIPCIÓN	Crear un producto con el campo “categoría” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un producto, dejando el campo “categoría” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “categoría”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un producto, dejando el campo “categoría” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “categoría”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-05

Tabla 72: Prueba 21

IDENTIFICADOR: PR-AC-22	
NOMBRE	Crear un producto con el campo “precio” vacío.
DESCRIPCIÓN	Crear un producto con el campo “precio” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un producto, dejando el campo “precio” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “precio”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un producto, dejando el campo “precio” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “precio”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-05

Tabla 73: Prueba 22

IDENTIFICADOR: PR-AC-23	
NOMBRE	Crear un producto con el campo “stock inicial” vacío.
DESCRIPCIÓN	Crear un producto con el campo “stock inicial” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de un producto, dejando el campo “stock inicial” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “stock inicial”.
RESULTADO OBTENIDO	Tras rellenarse los datos de un producto, dejando el campo “precio” vacío y pulsar en “Subir producto”, el producto no puede ser creado y se nos pide rellenar el campo “precio”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-05

Tabla 74: Prueba 23

IDENTIFICADOR: PR-AC-24	
NOMBRE	Ver listado de productos.
DESCRIPCIÓN	Comprobar que el listado de productos corresponde con los productos que hay dados de alta en la base de datos.
RESULTADO ESPERADO	Al pulsar en “Vender producto”, el listado de productos corresponde con el listado de productos que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-06, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
RESULTADO OBTENIDO	Al pulsar en “Vender producto”, el listado de productos corresponde con el listado de productos que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-06, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-06

Tabla 75: Prueba 24

IDENTIFICADOR: PR-AC-25	
NOMBRE	Eliminar un producto.
DESCRIPCIÓN	Buscar un producto ya creado y eliminarlo.
RESULTADO ESPERADO	Al hacer pulsar en “Eliminar producto” el producto es eliminado de la base de datos y, por lo tanto, eliminado de la lista de productos de la aplicación.
RESULTADO OBTENIDO	Al hacer pulsar en “Eliminar producto” el producto es eliminado de la base de datos y, por lo tanto, eliminado de la lista de productos de la aplicación.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-07

Tabla 76: Prueba 25

IDENTIFICADOR: PR-AC-26	
NOMBRE	Modificar un producto correctamente.
DESCRIPCIÓN	Se debe poder modificar un producto modificando cada uno de los datos solicitados en el formulario y que este producto quede actualizado en la base de datos.
RESULTADO ESPERADO	Tras modificarse los datos de un producto y pulsar en “Actualizar producto”, el producto es modificado en la base de datos.
RESULTADO OBTENIDO	Tras modificarse los datos de un producto y pulsar en “Actualizar producto”, el producto es modificado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-08

Tabla 77: Prueba 26

IDENTIFICADOR: PR-AC-27	
NOMBRE	Modificar un producto con el campo “nombre” vacío.
DESCRIPCIÓN	Modificar un producto con el campo “nombre” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un producto, dejando el campo “nombre” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “nombre”.
RESULTADO OBTENIDO	Al intentar modificar un producto, dejando el campo “nombre” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “nombre”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-08

Tabla 78: Prueba 27

IDENTIFICADOR: PR-AC-28	
NOMBRE	Modificar un producto con el campo “categoría” vacío.
DESCRIPCIÓN	Modificar un producto con el campo “categoría” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un producto, dejando el campo “categoría” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “categoría”.
RESULTADO OBTENIDO	Al intentar modificar un producto, dejando el campo “categoría” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “categoría”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-08

Tabla 79: Prueba 28

IDENTIFICADOR: PR-AC-29	
NOMBRE	Modificar un producto con el campo “precio” vacío.
DESCRIPCIÓN	Modificar un producto con el campo “precio” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un producto, dejando el campo “precio” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “precio”.
RESULTADO OBTENIDO	Al intentar modificar un producto, dejando el campo “precio” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “precio”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-08

Tabla 80: Prueba 29

IDENTIFICADOR: PR-AC-30	
NOMBRE	Modificar un producto con el campo “stock” vacío.
DESCRIPCIÓN	Modificar un producto con el campo “stock” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar un producto, dejando el campo “stock” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “stock”.
RESULTADO OBTENIDO	Al intentar modificar un producto, dejando el campo “stock” vacío y pulsar en “Actualizar producto”, el producto no puede ser modificado y se nos pide rellenar el campo “stock”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-08

Tabla 81: Prueba 30

IDENTIFICADOR: PR-AC-31	
NOMBRE	Vender un producto correctamente.
DESCRIPCIÓN	Se debe poder vender un producto indicando el número de unidades.
RESULTADO ESPERADO	Tras modificarse los datos de un producto y pulsar en “Actualizar producto”, el producto es modificado en la base de datos.
RESULTADO OBTENIDO	Tras modificarse los datos de un producto y pulsar en “Actualizar producto”, el producto es modificado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-09

Tabla 82: Prueba 31

IDENTIFICADOR: PR-AC-32	
NOMBRE	Vender un producto con “cantidad” negativa.
DESCRIPCIÓN	Al intentar vender un producto con el apartado “cantidad” negativo, debe saltar un error.
RESULTADO ESPERADO	Al intentar vender un producto con el apartado “cantidad” negativo, salta un error indicando que el valor de cantidad debe de ser mayor que cero.
RESULTADO OBTENIDO	Al intentar vender un producto con el apartado “cantidad” negativo, salta un error indicando que el valor de cantidad debe de ser mayor que cero.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-09

Tabla 83: Prueba 32

IDENTIFICADOR: PR-AC-33	
NOMBRE	Vender un producto con “cantidad” superior al stock existente.
DESCRIPCIÓN	Al intentar vender un producto con el apartado “cantidad” superior al stock existente, debe saltar un error.
RESULTADO ESPERADO	Al intentar vender un producto con el apartado “cantidad” superior al stock existente, salta un error indicando que el valor de cantidad debe de ser menor que el stock del producto.
RESULTADO OBTENIDO	Al intentar vender un producto con el apartado “cantidad” superior al stock existente, salta un error indicando que el valor de cantidad debe de ser menor que el stock del producto.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-09

Tabla 84: Prueba 33

IDENTIFICADOR: PR-AC-34	
NOMBRE	Crear una clase correctamente.
DESCRIPCIÓN	Se debe poder crear una clase rellenando todos los datos solicitados en el formulario y que esta clase quede registrada en la base de datos.
RESULTADO ESPERADO	Tras rellenarse los datos de una clase y pulsar en “Crear Grupo”, la clase es creada e insertada en la base de datos.
RESULTADO OBTENIDO	Tras rellenarse los datos de una clase y pulsar en “Crear Grupo”, la clase es creada e insertada en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-10

Tabla 85: Prueba 34

IDENTIFICADOR: PR-AC-35	
NOMBRE	Crear una clase con el campo “profesor” vacío.
DESCRIPCIÓN	Crear una clase con el campo “profesor” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de una clase, dejando el campo “profesor” vacío y pulsar en “Crear Grupo”, la clase no puede ser creada y se nos pide rellenar el campo “profesor”.
RESULTADO OBTENIDO	Tras rellenarse los datos de una clase, dejando el campo “profesor” vacío y pulsar en “Crear Grupo”, la clase no puede ser creada y se nos pide rellenar el campo “profesor”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-10

Tabla 86: Prueba 35

IDENTIFICADOR: PR-AC-36	
NOMBRE	Crear una clase con el campo “día” vacío.
DESCRIPCIÓN	Crear una clase con el campo “día” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de una clase, dejando el campo “día” vacío y pulsar en “Crear Grupo”, la clase no puede ser creada y se nos pide rellenar el campo “día”.
RESULTADO OBTENIDO	Tras rellenarse los datos de una clase, dejando el campo “día” vacío y pulsar en “Crear Grupo”, la clase no puede ser creada y se nos pide rellenar el campo “día”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-10

Tabla 87: Prueba 36

IDENTIFICADOR: PR-AC-37	
NOMBRE	Crear una clase con el campo “hora” vacío.
DESCRIPCIÓN	Crear una clase con el campo “hora” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de una clase, dejando el campo “hora” vacío y pulsar en “Crear Grupo”, la clase no puede ser creada y se nos pide rellenar el campo “hora”.
RESULTADO OBTENIDO	Tras rellenarse los datos de una clase, dejando el campo “hora” vacío y pulsar en “Crear Grupo”, la clase no puede ser creada y se nos pide rellenar el campo “hora”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-10

Tabla 88: Prueba 37

IDENTIFICADOR: PR-AC-38	
NOMBRE	Eliminar una clase.
DESCRIPCIÓN	Buscar una clase ya creada y eliminarla.
RESULTADO ESPERADO	Al hacer pulsar en “Eliminar grupo” el grupo es eliminado de la base de datos y, por lo tanto, eliminado de la lista de grupos de la aplicación.
RESULTADO OBTENIDO	Al hacer pulsar en “Eliminar grupo” el grupo es eliminado de la base de datos y, por lo tanto, eliminado de la lista de grupos de la aplicación.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-11

Tabla 89: Prueba 38

IDENTIFICADOR: PR-AC-39	
NOMBRE	Añadir un alumno a un grupo
DESCRIPCIÓN	Buscar una clase ya creada y añadir un alumno.
RESULTADO ESPERADO	Al elegir un alumno y pulsar en “Añadir al grupo” el alumno es añadido al grupo y, por lo tanto, el grupo es actualizado en la base de datos.
RESULTADO OBTENIDO	Al elegir un alumno y pulsar en “Añadir al grupo” el alumno es añadido al grupo y, por lo tanto, el grupo es actualizado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-12

Tabla 90: Prueba 39

IDENTIFICADOR: PR-AC-40	
NOMBRE	Eliminar un alumno de una clase.
DESCRIPCIÓN	Buscar una clase ya creada y eliminar un alumno.
RESULTADO ESPERADO	Al abrir un grupo, buscar un alumno y pulsar en “Eliminar” el alumno es eliminado del grupo y, por lo tanto, el grupo es actualizado en la base de datos.
RESULTADO OBTENIDO	Al abrir un grupo, buscar un alumno y pulsar en “Eliminar” el alumno es eliminado del grupo y, por lo tanto, el grupo es actualizado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-13

Tabla 91: Prueba 40

IDENTIFICADOR: PR-AC-41	
NOMBRE	Ver listado de grupos.
DESCRIPCIÓN	Comprobar que el listado de grupos corresponde con los grupos que hay dados de alta en la base de datos.
RESULTADO ESPERADO	Al pulsar en “Grupos” el listado de grupos corresponde con el listado de grupos que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-14, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
RESULTADO OBTENIDO	Al pulsar en “Grupos” el listado de grupos corresponde con el listado de grupos que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-14, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-14

Tabla 92: Prueba 41

IDENTIFICADOR: PR-AC-42	
NOMBRE	Crear un bono de clases correctamente.
DESCRIPCIÓN	Se debe poder crear un bono de clases rellenando todos los datos solicitados en el formulario y que este bono de clases quede registrado en la base de datos.
RESULTADO ESPERADO	Tras rellenarse los datos de un bono de clases y pulsar en “Confirmar Venta”, el bono de clases es creado e insertado en la base de datos.
RESULTADO OBTENIDO	Tras rellenarse los datos de un bono de clases y pulsar en “Confirmar Venta”, el bono de clases es creado e insertado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-15

Tabla 93: Prueba 42

IDENTIFICADOR: PR-AC-43	
NOMBRE	Marcar clase de bono correctamente.
DESCRIPCIÓN	Se debe poder marcar una clase como impartida.
RESULTADO ESPERADO	Tras entrar a los detalles de un bono particular ya creado y pulsar en “Confirmar clase”, la clase queda marcada especificando la fecha en la que se ha impartido y queda reflejado en la base de datos.
RESULTADO OBTENIDO	Tras entrar a los detalles de un bono particular ya creado y pulsar en “Confirmar clase”, la clase queda marcada especificando la fecha en la que se ha impartido y queda reflejado en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-16

Tabla 94: Prueba 43

IDENTIFICADOR: PR-AC-44	
NOMBRE	Crear una tarifa correctamente.
DESCRIPCIÓN	Se debe poder crear una tarifa rellenando todos los datos solicitados en el formulario y que esta tarifa quede registrada en la base de datos.
RESULTADO ESPERADO	Tras rellenarse los datos de una tarifa y pulsar en “Subir Tarifa”, la tarifa es creada e insertada en la base de datos.
RESULTADO OBTENIDO	Tras rellenarse los datos de una tarifa y pulsar en “Subir Tarifa”, la tarifa es creada e insertada en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-17

Tabla 95: Prueba 44

IDENTIFICADOR: PR-AC-45	
NOMBRE	Crear una tarifa con el campo “nombre del producto” vacío.
DESCRIPCIÓN	Crear una tarifa con el campo “nombre del producto” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de una tarifa, dejando el campo “nombre del producto” vacío y pulsar en “Subir Tarifa”, la clase no puede ser creada y se nos pide rellenar el campo “nombre del producto”.
RESULTADO OBTENIDO	Tras rellenarse los datos de una tarifa, dejando el campo “nombre del producto” vacío y pulsar en “Subir Tarifa”, la clase no puede ser creada y se nos pide rellenar el campo “nombre del producto”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-17

Tabla 96: Prueba 45

IDENTIFICADOR: PR-AC-46	
NOMBRE	Crear una tarifa con el campo “precio del producto” vacío.
DESCRIPCIÓN	Crear una tarifa con el campo “precio del producto” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Tras rellenarse los datos de una tarifa, dejando el campo “precio del producto” vacío y pulsar en “Subir Tarifa”, la clase no puede ser creada y se nos pide rellenar el campo “precio del producto”.
RESULTADO OBTENIDO	Tras rellenarse los datos de una tarifa, dejando el campo “precio del producto” vacío y pulsar en “Subir Tarifa”, la clase no puede ser creada y se nos pide rellenar el campo “precio del producto”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-17

Tabla 97: Prueba 46

IDENTIFICADOR: PR-AC-47	
NOMBRE	Modificar una tarifa correctamente.
DESCRIPCIÓN	Se debe poder modificar una tarifa modificando cada uno de los datos solicitados en el formulario y que esta tarifa quede actualizada en la base de datos.
RESULTADO ESPERADO	Tras modificarse los datos de una tarifa y pulsar en “Actualizar tarifa”, la tarifa es modificada en la base de datos.
RESULTADO OBTENIDO	Tras modificarse los datos de una tarifa y pulsar en “Actualizar tarifa”, la tarifa es modificada en la base de datos.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-18

Tabla 98: Prueba 47

IDENTIFICADOR: PR-AC-48	
NOMBRE	Modificar una tarifa con el campo “nombre” vacío.
DESCRIPCIÓN	Modificar una tarifa con el campo “nombre” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar una tarifa, dejando el campo “nombre” vacío y pulsar en “Actualizar Tarifa”, la tarifa no puede ser modificada y se nos pide rellenar el campo “nombre”.
RESULTADO OBTENIDO	Al intentar modificar una tarifa, dejando el campo “nombre” vacío y pulsar en “Actualizar Tarifa”, la tarifa no puede ser modificada y se nos pide rellenar el campo “nombre”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-18

Tabla 99: Prueba 48

IDENTIFICADOR: PR-AC-49	
NOMBRE	Modificar una tarifa con el campo “precio” vacío.
DESCRIPCIÓN	Modificar una tarifa con el campo “precio” vacío siendo este campo un campo obligatorio.
RESULTADO ESPERADO	Al intentar modificar una tarifa, dejando el campo “precio” vacío y pulsar en “Actualizar Tarifa”, la tarifa no puede ser modificada y se nos pide rellenar el campo “precio”.
RESULTADO OBTENIDO	Al intentar modificar una tarifa, dejando el campo “precio” vacío y pulsar en “Actualizar Tarifa”, la tarifa no puede ser modificada y se nos pide rellenar el campo “precio”.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-18

Tabla 100: Prueba 49

IDENTIFICADOR: PR-AC-50	
NOMBRE	Eliminar una tarifa.
DESCRIPCIÓN	Buscar una tarifa ya creada y eliminarla.
RESULTADO ESPERADO	Al hacer pulsar en “Eliminar Tarifa” la tarifa es eliminada de la base de datos y, por lo tanto, eliminada de la lista de tarifas de la aplicación.
RESULTADO OBTENIDO	Al hacer pulsar en “Eliminar Tarifa” la tarifa es eliminada de la base de datos y, por lo tanto, eliminada de la lista de tarifas de la aplicación.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-19

Tabla 101: Prueba 50

IDENTIFICADOR: PR-AC-51	
NOMBRE	Ver listado de tarifas.
DESCRIPCIÓN	Comprobar que el listado de tarifas corresponde con las tarifas que hay dadas de alta en la base de datos.
RESULTADO ESPERADO	Al pulsar en “Tarifas” el listado de tarifas corresponde con el listado de tarifas que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-20, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
RESULTADO OBTENIDO	Al pulsar en “Tarifas” el listado de tarifas corresponde con el listado de tarifas que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-20, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-20

Tabla 102: Prueba 51

IDENTIFICADOR: PR-AC-52	
NOMBRE	Ver listado de pagos.
DESCRIPCIÓN	Comprobar que el listado de pagos corresponde con los grupos que hay dados de alta en la base de datos.
RESULTADO ESPERADO	Al pulsar en “Comprobar cuentas” el listado de pagos corresponde con el listado de pagos que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-21, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
RESULTADO OBTENIDO	Al pulsar en “Comprobar cuentas” el listado de pagos corresponde con el listado de pagos que hay dados de alta en la base de datos, aparecen todas las columnas requeridas en el requisito RF-AC-21, se puede exportar en los formatos pedidos en el mismo requisito y el buscador funciona.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-21

Tabla 103: Prueba 52

IDENTIFICADOR: PR-AC-53	
NOMBRE	Ver detalles del usuario
DESCRIPCIÓN	Comprobar que se pueden consultar todos los datos personales de un usuario
RESULTADO ESPERADO	Al buscar un usuario y pulsar en “VER” podemos acceder a todos los datos personales de ese usuario.
RESULTADO OBTENIDO	Al buscar un usuario y pulsar en “VER” podemos acceder a todos los datos personales de ese usuario.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-22

Tabla 104: Prueba 53

IDENTIFICADOR: PR-AC-54	
NOMBRE	Ver listado de grupos asociados a un usuario
DESCRIPCIÓN	Comprobar que se pueden consultar todos los grupos de clases en los que está inscrito un usuario
RESULTADO ESPERADO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Clases”, se ven los grupos a los que un usuario está inscrito.
RESULTADO OBTENIDO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Clases”, se ven los grupos a los que un usuario está inscrito.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-23

Tabla 105: Prueba 54

IDENTIFICADOR: PR-AC-55	
NOMBRE	Ver listado de bonos asociados a un usuario.
DESCRIPCIÓN	Comprobar que se pueden consultar todos los bonos de clases adquiridos por un usuario.
RESULTADO ESPERADO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Bonos”, se ven los bonos que ha adquirido un usuario.
RESULTADO OBTENIDO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Bonos”, se ven los bonos que ha adquirido un usuario.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-24

Tabla 106: Prueba 55

IDENTIFICADOR: PR-AC-56	
NOMBRE	Ver listado de pagos asociados a un usuario.
DESCRIPCIÓN	Comprobar que se pueden consultar todos los pagos realizados por un usuario.
RESULTADO ESPERADO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Pagos”, se ven los pagos que ha realizado un usuario.
RESULTADO OBTENIDO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Pagos”, se ven los pagos que ha realizado un usuario.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-26

Tabla 107: Prueba 56

IDENTIFICADOR: PR-AC-57	
NOMBRE	Crear pago asociado a un usuario.
DESCRIPCIÓN	Comprobar que se puede crear un pago asociado a un usuario.
RESULTADO ESPERADO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Pagos”, se puede crear un pago introduciendo un asunto y una cantidad de dinero.
RESULTADO OBTENIDO	Al buscar un usuario y pulsar en “VER”, entrando en la pestaña “Pagos”, se puede crear un pago introduciendo un asunto y una cantidad de dinero.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-25

Tabla 108: Prueba 57

IDENTIFICADOR: PR-AC-58	
NOMBRE	Comprobar inicio de sesión.
DESCRIPCIÓN	Comprobar que se puede iniciar sesión para entrar en la aplicación.
RESULTADO ESPERADO	Al entrar en la aplicación e introducir los datos de inicio de sesión correctamente, se puede acceder a la aplicación.
RESULTADO OBTENIDO	Al entrar en la aplicación e introducir los datos de inicio de sesión correctamente, se puede acceder a la aplicación.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-27

Tabla 109: Prueba 58

IDENTIFICADOR: PR-AC-59	
NOMBRE	Inicio de sesión con nombre de usuario incorrecto
DESCRIPCIÓN	Comprobar que salta error al introducir mal el nombre de usuario.
RESULTADO ESPERADO	Al entrar en la aplicación e introducir mal el nombre de usuario, salta un error indicando que el nombre de usuario es incorrecto.
RESULTADO OBTENIDO	Al entrar en la aplicación e introducir mal el nombre de usuario, salta un error indicando que el nombre de usuario es incorrecto.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-27

Tabla 110: Prueba 59

IDENTIFICADOR: PR-AC-60	
NOMBRE	Inicio de sesión con contraseña incorrecta.
DESCRIPCIÓN	Comprobar que salta error al introducir mal la contraseña.
RESULTADO ESPERADO	Al entrar en la aplicación e introducir mal la contraseña, salta un error indicando que la contraseña es incorrecta.
RESULTADO OBTENIDO	Al entrar en la aplicación e introducir mal la contraseña, salta un error indicando que la contraseña es incorrecta.
EVALUACIÓN	Al coincidir tanto el resultado esperado como el resultado obtenido, esta prueba queda calificada como exitosa.
REQUISITOS ASOCIADOS	RF-AC-27

Tabla 111: Prueba 60

9. Conclusiones y trabajos futuros

En esta sección se hablará sobre tanto las conclusiones a nivel personal como a nivel de la realización del trabajo una vez terminado el proyecto como de posibles trabajos futuros que se podrían implementar en la aplicación.

9.1. Conclusiones personales

Este trabajo puedo considerarlo como el mayor desafío al que me he enfrentado hasta la fecha. Es mi primera experiencia más cercana a lo que sería el mundo laboral y me he tenido que enfrentar a ella en solitario. Eso contrasta mucho con el resto de trabajos que he ido realizando en el grado, ya que en los anteriores trabajos a los que me había enfrentado siempre había tenido un grupo de compañeros con los que poder apoyarme e incluso dividir tareas.

Cuando realizas un proyecto de esta magnitud en solitario te das cuenta de la importancia de una buena planificación y de un buen control del estado del proyecto. El hecho de saber organizarte, establecer unos horarios e intentar cumplir con los objetivos diarios es una tarea muy importante, herramientas como Trello [28] me han sido de gran ayuda para la realización de este proyecto.

El hecho de haber diseñado e implementado una aplicación web, me ha hecho darme cuenta que es de las áreas del conocimiento en informática que más me atraen y me gustan. A pesar de tratarse de un trabajo con muchas horas de dedicación y haberlo hecho de forma autónoma, no se me ha hecho pesado pasar tiempo desarrollándolo.

Por último quiero destacar que me siento muy orgulloso de haber realizado este proyecto ya que la propuesta del proyecto fue una iniciativa personal. Trabajo en un club de pádel y pude darme cuenta de la necesidad que teníamos en el club de una aplicación como la que he desarrollado a lo largo de este proyecto y que se espera poder ser usada.

9.2. Conclusiones sobre la realización del proyecto

La finalidad de este proyecto era el diseño e implementación de una aplicación web capaz de gestionar un club de pádel. Para ello, debía cumplir cuatro objetivos:

- El primer objetivo se basaba en la gestión de usuarios. En esta aplicación web se han implementado funciones capaces de registrar nuevos usuarios recogiendo sus datos personales y de contacto, modificarlos y eliminarlos, además de poder llevar un control sobre los pagos realizados, clases en las que están inscritos o servicios adquiridos por cada usuario.
- El segundo objetivo se basaba en la gestión de una tienda. En esta aplicación web se han implementado funciones capaces de llevar un control del inventario de la tienda del club, pudiendo registrar productos, modificar su stock o incluso venderlos.
- El tercer objetivo se basaba en la gestión de clases. En esta aplicación web se han implementado funciones capaces de crear nuevos grupos de pádel, estando estos grupos asociados a un profesor, día de la semana y hora, y pudiendo inscribir hasta a cuatro alumnos por grupo. Además también se ha implementado funciones capaz de gestionar los bonos de clases particulares, estando estos bonos asociados a un cliente y pudiendo llevar un control de las clases restantes del bono.
- El cuarto objetivo se basaba en la gestión de las finanzas. En esta aplicación se han implementado funciones capaces de generar listas y reportes de los cobros realizados en el club además de poder tener un listado de las tarifas aplicadas a los distintos servicios ofrecidos.

Es necesario destacar que la realización de esta aplicación viene dada por las necesidades de un club de pádel real y que se espera poder ser usada para la gestión interna del mismo.

En conclusión, el trabajo desarrollado cumple con todos los objetivos inicialmente propuestos por lo que se puede afirmar que el resultado final es satisfactorio. Se ha desarrollado una aplicación web capaz de gestionar tanto a usuarios como a clases, tienda y finanzas.

9.3. Trabajos futuros

En esta sección se comentarán diferentes trabajos futuros que podrían implementarse para poder crear una aplicación más completa en el futuro:

- **Versión en inglés:** debido a la expansión del pádel por el mundo, un trabajo futuro a tener en cuenta debería de ser la posibilidad de poder elegir si usar la aplicación en su versión en español o en una versión en inglés.
- **Creación automática de torneos:** implementación de una herramienta de ayuda para la creación de torneos. La herramienta funcionaría de forma que, inscribiendo las parejas apuntadas a un torneo, se generaran automáticamente los emparejamientos.
- **Calendario:** implementación de un calendario donde poder visualizar y apuntar los eventos futuros del club.
- **Mejorar base de datos:** creación de una base de datos con tablas relacionadas entre sí y así, no depender del front-end y el back-end para el correcto funcionamiento de la gestión de los datos dentro de la aplicación.
- **Aplicación móvil:** desarrollo de una aplicación móvil similar a la aplicación web desarrollada en el proyecto.
- **Versión alumno/profesor:** crear una versión para alumnos y profesores con grupos de chat por cada clase que se esté inscrito o se imparta de tal forma que exista una comunicación entre los alumnos y profesores. Además, tanto alumnos como profesores, podrían llevar un control de los servicios contratados o pagos realizados en el club.

10. Glosario

- **Hardware:** es el conjunto de partes físicas o tangibles que constituyen un sistema informático.
- **Software:** es el conjunto de programas o aplicaciones que pueden ser usados en un equipo informático.
- **Servidor:** es un hardware, es decir una máquina física integrada en una red informática que nos permite procesar solicitudes y entregar solicitudes a los clientes.
- **Aplicación web:** son aquellas herramientas a las que el usuario puede utilizar accediendo a un servidor web mediante internet o en local.
- **Diagrama de Gantt:** es una herramienta de planificación que nos ayuda a tener una visión global de la planificación temporal de un proyecto.
- **Java:** lenguaje de programación de alto nivel más popular, usado por las empresas más grandes del mundo.
- **Python:** lenguaje de programación de alto nivel que se caracteriza por ser sencillo y fácil de leer.
- **Puerto:** interfaz mediante la cual se pueden recibir y enviar datos.
- **Plugin:** pequeños programas ya prediseñados que añaden funcionalidad a aplicaciones web.
- **Relaciones entre tablas:** es una asociación que permite relacionar tablas y sus datos de tal forma que se logre una consistencia entre ellos.
- **Función:** algoritmo que realiza una secuencia de operaciones.
- **JSON:** JavaScript Object Notation, es un formato diseñado para el intercambio de datos.
- **Interfaz:** es la estructura visual diseñada para que el usuario pueda acceder a los contenidos de una aplicación web y realizar solicitudes.
- **Margen de riesgo:** margen económico dedicado a posibles situaciones adversas que pudieses suceder a lo largo de un proyecto.
- **Navegador:** programa que permite visualizar el contenido de una aplicación o página web. Normalmente interpretan contenido HTML.

11. Bibliografía

- [1] BOE, «Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos,» 6 diciembre 2018. [En línea]. Available: <https://www.boe.es/buscar/pdf/2018/BOE-A-2018-16673-consolidado.pdf>.
- [2] Infoautónomos, [En línea]. Available: <https://reduce.es/aspectos-mas-importantes-de-la-ley-de-proteccion-de-datos/>. [Último acceso: 18 mayo 2022].
- [3] AdminLTE, 10 febrero 2022. [En línea]. Available: <https://adminlte.io/docs/3.2/license.html>.
- [4] Monitor Deloitte, «Global Padel Report,» 2021. [En línea]. Available: [https://8258038.fs1.hubspotusercontent-na1.net/hubfs/8258038/Playtomic_Global%20Padel%20Report_Final%20Version%20\(0x\).pdf?utm_medium=email&_hsmi=213465606&_hsenc=p2ANqtz--NlznDZZKu6OguQaslxtqUZ-zBKQ5yHqo7CaeRgjMq1fucLmYarsgzjbcKhKQEAKDYnTDYUqVO_WTplwIVKc_](https://8258038.fs1.hubspotusercontent-na1.net/hubfs/8258038/Playtomic_Global%20Padel%20Report_Final%20Version%20(0x).pdf?utm_medium=email&_hsmi=213465606&_hsenc=p2ANqtz--NlznDZZKu6OguQaslxtqUZ-zBKQ5yHqo7CaeRgjMq1fucLmYarsgzjbcKhKQEAKDYnTDYUqVO_WTplwIVKc_).
- [5] J. Ribas, «Tecnologías FrontEnd y BackEnd en el Desarrollo Web,» [En línea]. Available: <https://dissenyproducte.blogspot.com/2017/10/tecnologias-frontend-y-backend-en-el.html?m=1>. [Último acceso: 20 mayo 2022].
- [6] X. Idevik, «¿Qué es una etiqueta HTML y cuáles son las más importantes?,» 14 julio 2021. [En línea]. Available: <https://www.iebschool.com/blog/que-es-etiqueta-html-analitica-usabilidad/#:~:text=En%20la%20actualidad%20existen%20142,ido%20evolucionando%20los%20%C3%BAltimos%20a%C3%B1os.>
- [7] Manz, «¿Qué es CSS?,» [En línea]. Available: <https://lenguajecss.com/css/introduccion/que-es-css/>. [Último acceso: 20 mayo 2022].
- [8] J. R. González, «enter.co,» 5 noviembre 2021. [En línea]. Available: <https://www.enter.co/especiales/dev/herramientas-dev/estos-son-los-10-lenguajes-de-programacion-mas-populares-en-2021/#:~:text=JavaScript%20se%20mantiene%20como%20el,de%20desarrolladores%20en%20el%20mundo..>
- [9] R. Ramos, «¿Qué es JavaScript y para qué sirve?,» [En línea]. Available: <https://soyrafamos.com/que-es-javascript-para-que-sirve/>. [Último acceso: 20 mayo 2022].
- [10] jQuery, «jQuery,» [En línea]. Available: <https://jquery.com/>. [Último acceso: 20 mayo 2022].
- [11] L. Chuburu, «Qué es JQuery y cómo implementarlo,» [En línea]. Available: <https://www.laurachuburu.com.ar/tutoriales/que-es-jquery-y-como-implementarlo.php>.

[Último acceso: 20 mayo 2022].

- [12] G. B. «¿Qué es AJAX y cómo funciona?,» 2 marzo 2022. [En línea]. Available: <https://www.hostinger.es/tutoriales/que-es-ajax>.
- [13] Wikipedia, la enciclopedia libre, «Java EE,» [En línea]. Available: https://es.wikipedia.org/wiki/Java_EE. [Último acceso: 20 mayo 2022].
- [14] Oracle, «Java EE at a Glance,» [En línea]. Available: <https://www.oracle.com/es/java/technologies/java-ee-glance.html>. [Último acceso: 20 mayo 2022].
- [15] I. d. Souza, «Descubre qué es el lenguaje de programación PHP y en qué situaciones se hace útil,» 9 marzo 2020. [En línea]. Available: <https://rockcontent.com/es/blog/php/>.
- [16] Keep Coding, «7 Ventajas del desarrollo Web con Python,» 21 enero 2022. [En línea]. Available: <https://keepcoding.io/blog/desarrollo-web-con-python/>.
- [17] Oracle, «Base de datos definida,» [En línea]. Available: <https://www.oracle.com/es/database/what-is-database/>. [Último acceso: 20 mayo 2022].
- [18] Oracle, «¿Qué es una base de datos relacional (RDBMS)?,» [En línea]. Available: <https://www.oracle.com/es/database/what-is-a-relational-database/>. [Último acceso: 21 mayo 2022].
- [19] IONOS, «Oracle Database: definición y funcionamiento,» 16 febrero 2022. [En línea]. Available: <https://www.ionos.es/digitalguide/hosting/cuestiones-tecnicas/oracle-database/>.
- [20] Á. Robledano, «Qué es MySQL: Características y ventajas,» 24 septiembre 2019. [En línea]. Available: <https://openwebinars.net/blog/que-es-mysql/>.
- [21] Microsoft, «Datos no relacionales y NoSQL,» [En línea]. Available: <https://docs.microsoft.com/es-es/azure/architecture/data-guide/big-data/non-relational-data>. [Último acceso: 21 mayo 2022].
- [22] Á. Robledano, «Qué es MongoDB,» 28 octubre 2019. [En línea]. Available: <https://openwebinars.net/blog/que-es-mongodb/>.
- [23] Amazon, «¿Qué es Redis?,» [En línea]. Available: <https://aws.amazon.com/es/elasticache/what-is-redis/>. [Último acceso: 21 mayo 2022].
- [24] Wikipedia, la enciclopedia libre, «Payara Server,» [En línea]. Available: https://en.wikipedia.org/wiki/Payara_Server. [Último acceso: 20 mayo 2022].
- [25] Playtomic, [En línea]. Available: <https://blog.playtomic.io/lo-que-debes-saber-sobre-playtomic/>. [Último acceso: 24 julio 2017].
- [26] Nubapp, «Especialistas en software de gestión deportiva,» [En línea]. Available: <https://www.nubapp.com/es/>. [Último acceso: 22 mayo 2022].

[27] Padel Manager, [En línea]. Available: <https://padelmanager.com/>. [Último acceso: 22 mayo 2022].

[28] Trello, [En línea]. Available: <https://trello.com/es>. [Último acceso: 10 febrero 2022].

Anexo 2: Manual de usuario

En esta sección se elaborará a modo de manual, un tutorial explicando cómo usar cada una de las funciones disponibles en la aplicación.

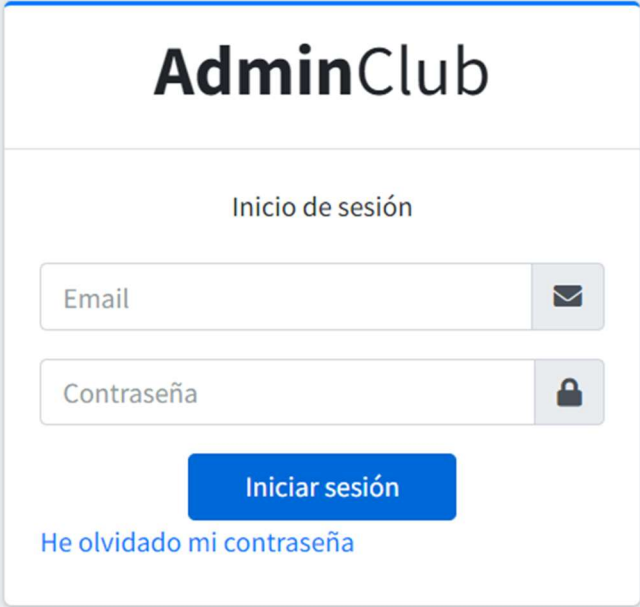
Índice del manual de usuario

1. Inicio de la aplicación.	120
1.1. Iniciar sesión.....	120
2. Gestión de productos.	121
2.1. Crear producto.	121
2.2. Modificar/Eliminar producto.....	122
2.3. Vender producto.	123
3. Gestión de usuarios.....	125
3.1. Crear usuario.....	125
3.2. Modificar/Eliminar usuario.	127
4. Gestión de clases.....	128
4.1. Buscar grupo.	128
4.2. Crear grupo.	129
4.3. Añadir/Eliminar alumnos de un grupo.....	130
4.4. Eliminar grupo.....	133
4.5. Buscar bono de clases particulares.....	133
4.6. Crear bono de clases particulares.....	134
4.7. Marcar clase particular como impartida.....	135
5. Gestión de finanzas.....	137
5.1. Buscar tarifa.	137
5.2. Crear tarifa.	138
5.3. Modificar/Eliminar tarifa.....	139
5.4. Comprobar cuentas.....	140

1. Inicio de la aplicación.

1.1. Iniciar sesión.

Una vez accedas a la aplicación, se deben introducir las credenciales email y contraseña que han sido otorgadas por la compañía AdminClub.



The image shows a login form for AdminClub. The form is centered on a light gray background. It has a white header with the text "AdminClub" in bold black font. Below the header, the text "Inicio de sesión" is centered. There are two input fields: "Email" with an envelope icon and "Contraseña" with a lock icon. Below these fields is a blue button labeled "Iniciar sesión". At the bottom of the form, there is a link that says "He olvidado mi contraseña" in blue text.

Figura 14: Inicio sesión Adminclub

2. Gestión de productos.

2.1. Crear producto.

Para crear un producto, accederemos, usando la barra de navegación lateral, a la sección Tienda > Crear producto.

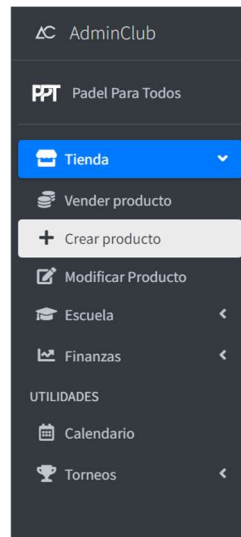


Figura 15: Crear producto 1

Una vez dentro de la sección, accederemos a un formulario que tendremos que rellenar y, posteriormente, tendremos que pulsar en “Subir producto”.

Crear producto	
Nombre del producto	Categoría
Sudadera pádel rosa	Ropa
Precio del producto	Stock inicial
\$ 18	30
Subir producto	

Figura 16: Crear producto 2

2.2. Modificar/Eliminar producto.

Para modificar o eliminar un producto, accederemos, usando la barra de navegación lateral, a la sección Tienda > Modificar producto.

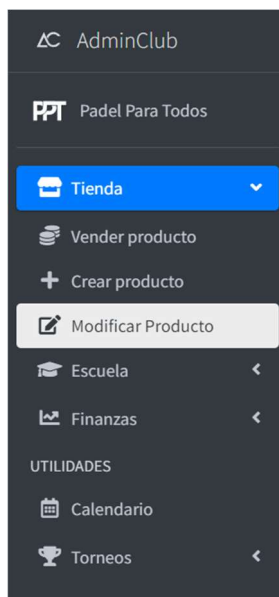


Figura 17: Modificar producto

Una vez dentro de la sección, accederemos a una lista con los productos subidos hasta la fecha, buscaremos el producto que queremos modificar o eliminar y pulsaremos en “Modificar”.

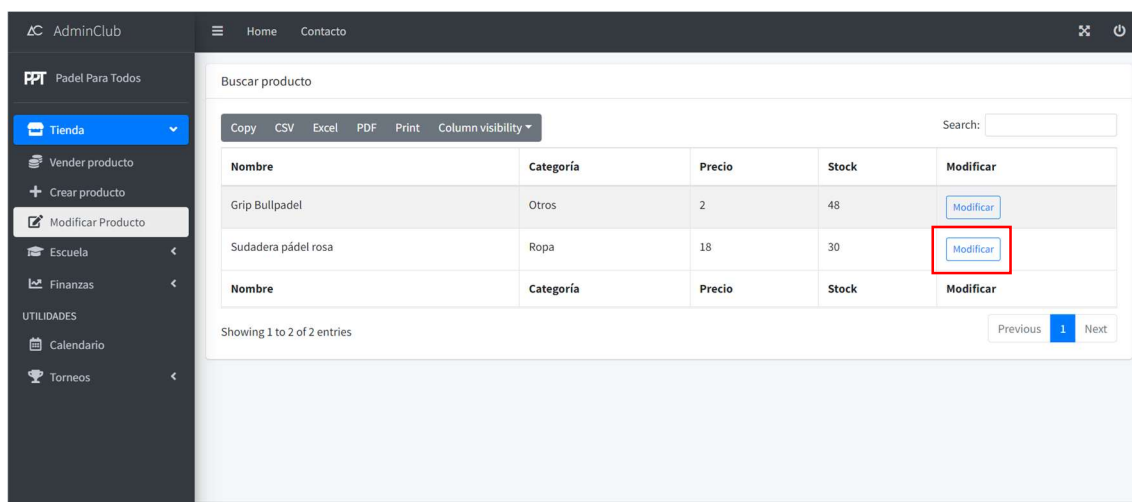


Figura 18: Modificar producto 2

Una vez dentro, accederemos a un formulario donde poder o bien modificar los datos cambiando los campos y pulsando en “Actualizar producto” o bien eliminar el producto pulsando en “Eliminar producto”.

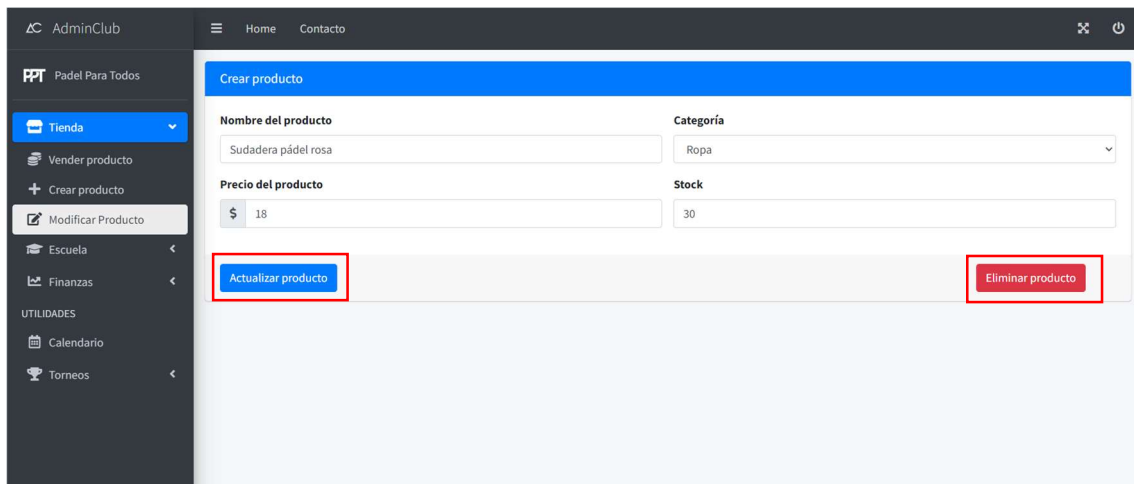


Figura 19: Modificar producto 3

2.3. Vender producto.

Para vender un producto, accederemos, usando la barra de navegación lateral, a la sección Tienda > Vender producto.

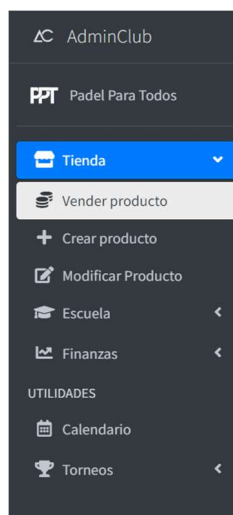


Figura 20: Vender producto

Una vez dentro de la sección, accederemos a una lista con los productos subidos hasta la fecha, buscaremos el producto que queremos vender, introduciremos una cantidad y pulsaremos en “Vender”.

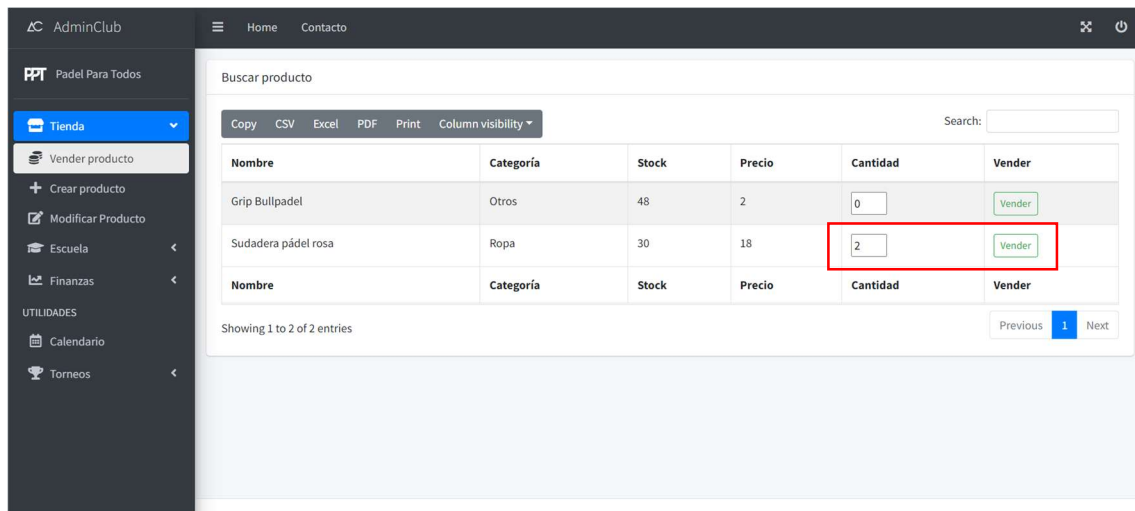


Figura 21: Vender producto 2

Haciendo clic en “Vender” se generará un resumen de la venta y, pulsando en “Confirmar Venta” esa venta será confirmada guardando esa venta en la base de datos y modificando el stock del producto vendido.

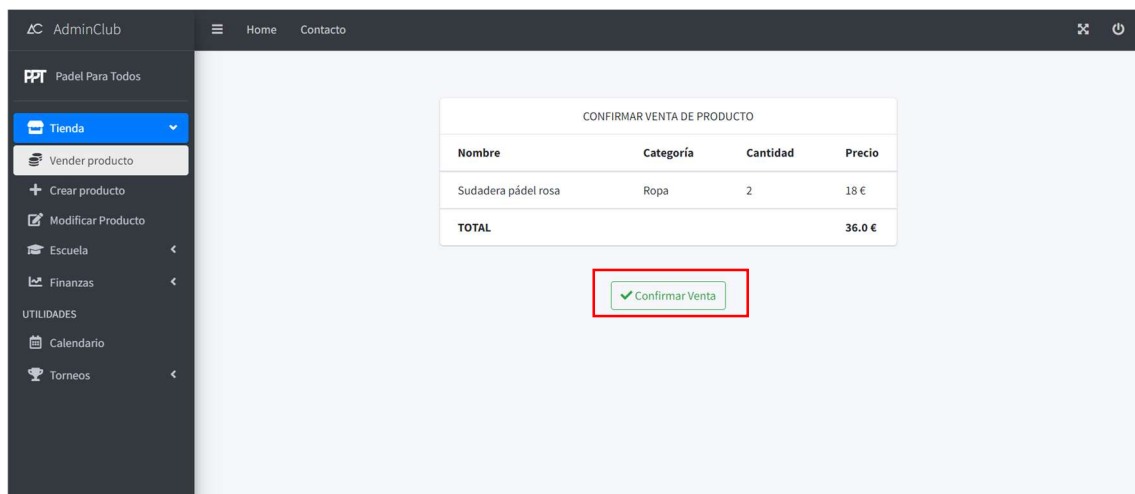


Figura 22: Vender producto 3

3. Gestión de usuarios.

3.1. Crear usuario.

Para crear un usuario, accederemos, usando la barra de navegación lateral, a la sección Escuela > Añadir usuario.

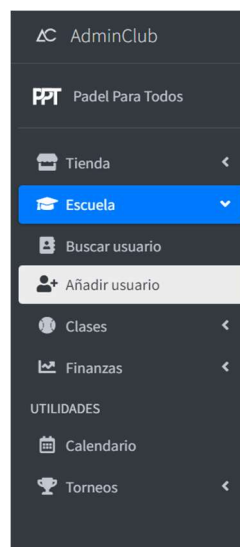


Figura 23: Crear usuario

Una vez dentro de la sección, accederemos a un formulario que tendremos que rellenar y, posteriormente, tendremos que pulsar en “Crear usuario”.

Esta imagen muestra el formulario 'Añadir Usuario' dentro de la interfaz de AdminClub. El formulario está dividido en varias secciones con campos de entrada: 'Nombre' (Rafael), 'Apellidos' (Nadal), 'Teléfono' (123456789), 'Correo electrónico' (rafanadal@gmail.com), 'Género' (Hombre), 'Categoría' (Alumno), 'Fecha de nacimiento' (03/06/1986) y 'Instagram (opcional)' (rafaelnadal). En la parte inferior del formulario, hay un botón azul con el texto 'Crear Usuario', que está rodeado por un recuadro rojo. A la izquierda del formulario, se repite la barra de navegación lateral con 'Escuela' seleccionada y 'Añadir usuario' como la opción activa.

Figura 24: Crear usuario 2

Detalles usuario.

Para poder consultar los detalles de un usuario, accederemos, usando la barra de navegación lateral, a la sección Escuela > Buscar usuario.

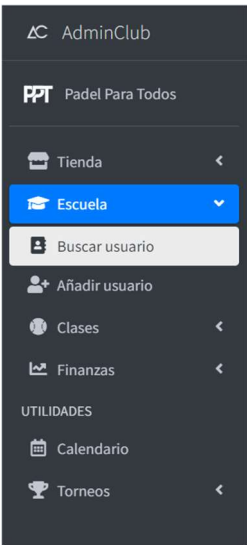


Figura 25: Detalles usuario

Una vez dentro de la sección, accederemos a una lista con los usuarios creados hasta la fecha, buscaremos el usuario que queremos consultar y pulsaremos en “Ver”.

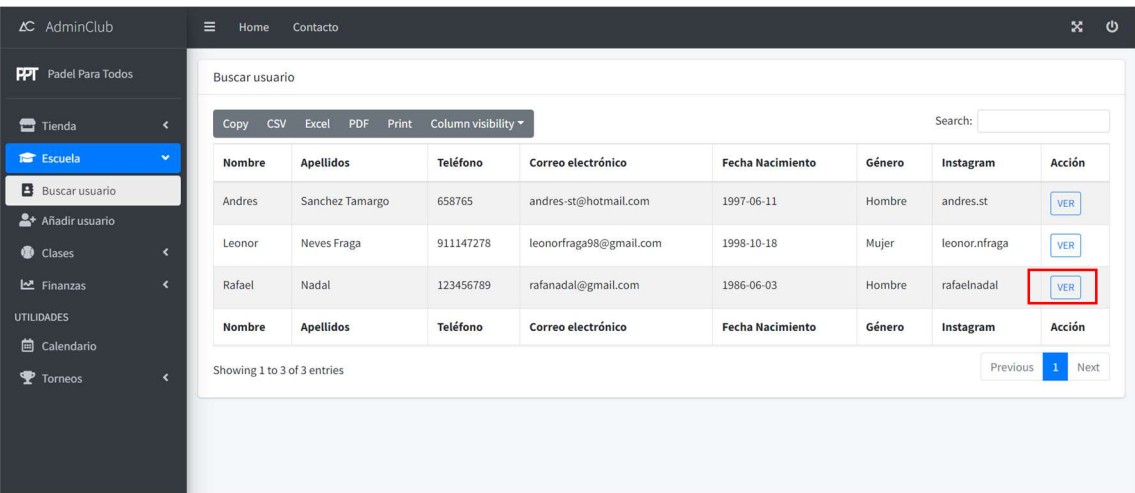


Figura 26: Detalles usuario 2

Una vez dentro de esta sección podremos consultar toda su información personal así como consultar las clases en las que está inscrito, bonos adquiridos o pagos realizados por el usuario usando la barra de navegación superior.

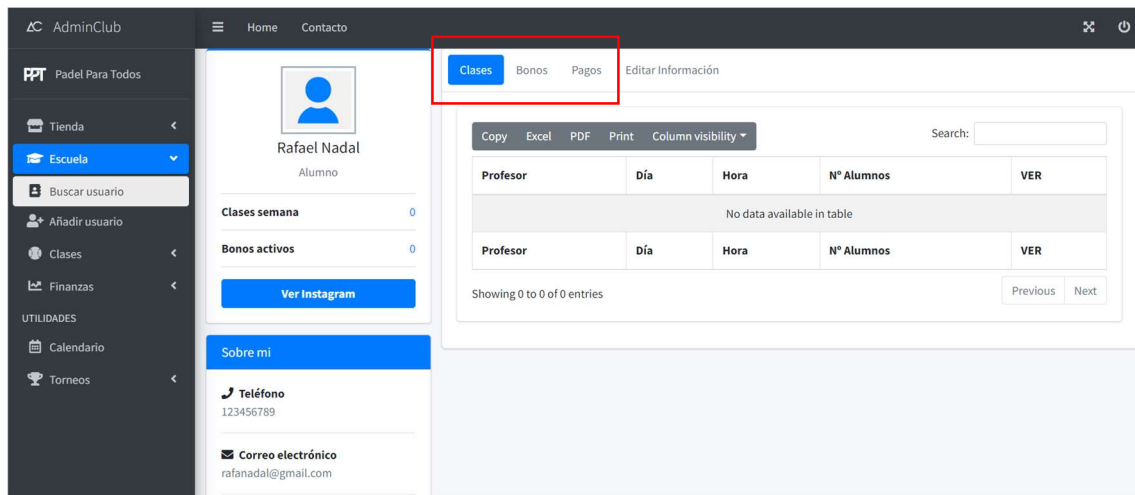


Figura 27: Detalles usuario 3

3.2. Modificar/Eliminar usuario.

Una vez dentro de los detalles de un usuario, usando la barra de navegación superior, accederemos a “Editar información”. En esta sección tendremos acceso a un formulario donde poder modificar los datos del usuario pulsando en “Actualizar usuario” o eliminar el usuario pulsando en “Eliminar usuario”.

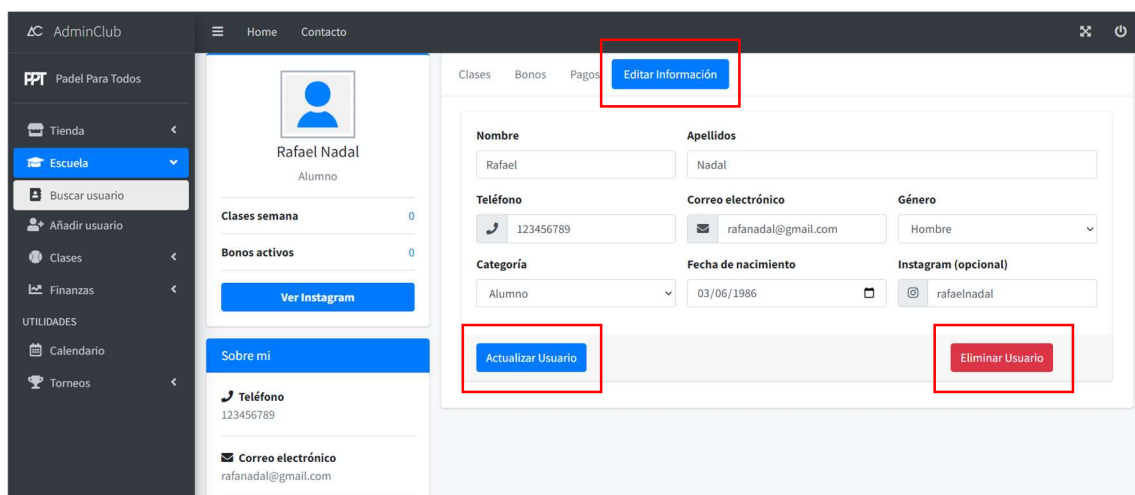


Figura 28: Modificar usuario

4. Gestión de clases.

4.1. Buscar grupo.

Para poder buscar un grupo, accederemos, usando la barra de navegación lateral, a la sección Escuela > Clases > Grupos.

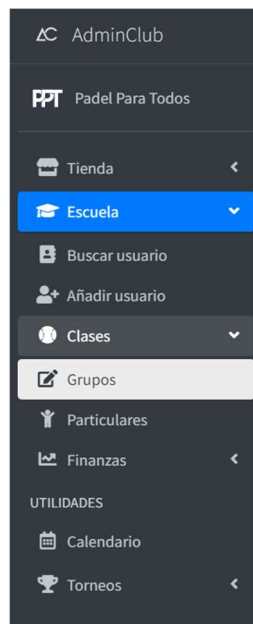


Figura 29: Buscar grupo

Una vez dentro de la sección, accederemos a una lista con los grupos creados hasta la fecha, podremos usar la barra de búsqueda para encontrar el grupo deseado.

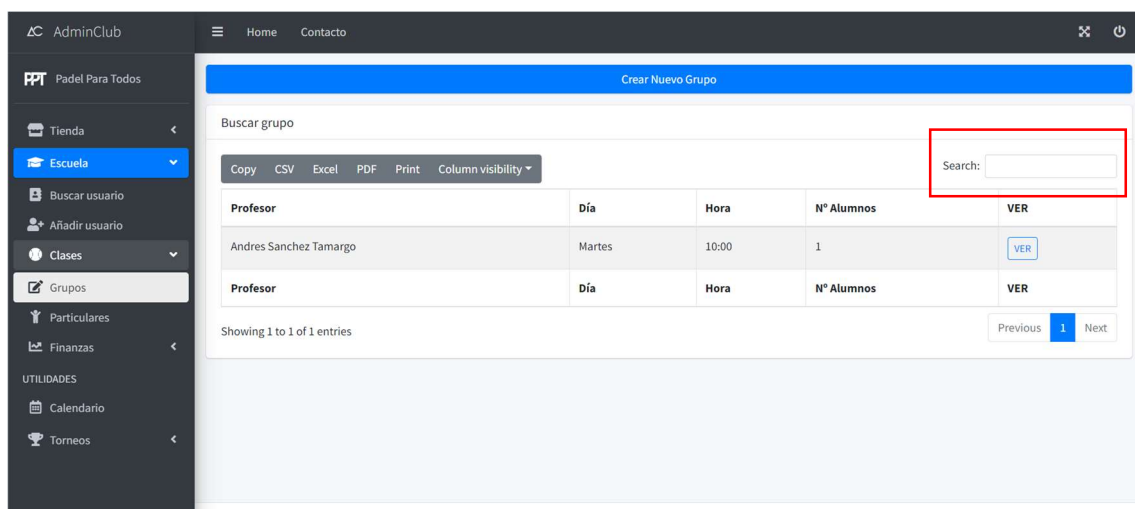


Figura 30: Buscar grupo 2

4.2. Crear grupo.

Para poder buscar un grupo, accederemos, usando la barra de navegación lateral, a la sección Escuela > Clases > Grupos.

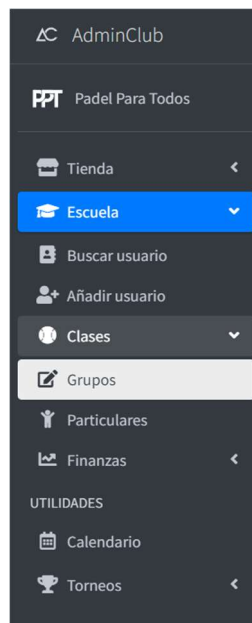


Figura 31: Crear grupo

Una vez dentro de la sección, accederemos a una lista con los grupos creados hasta la fecha, y pulsaremos en “Crear nuevo grupo”.

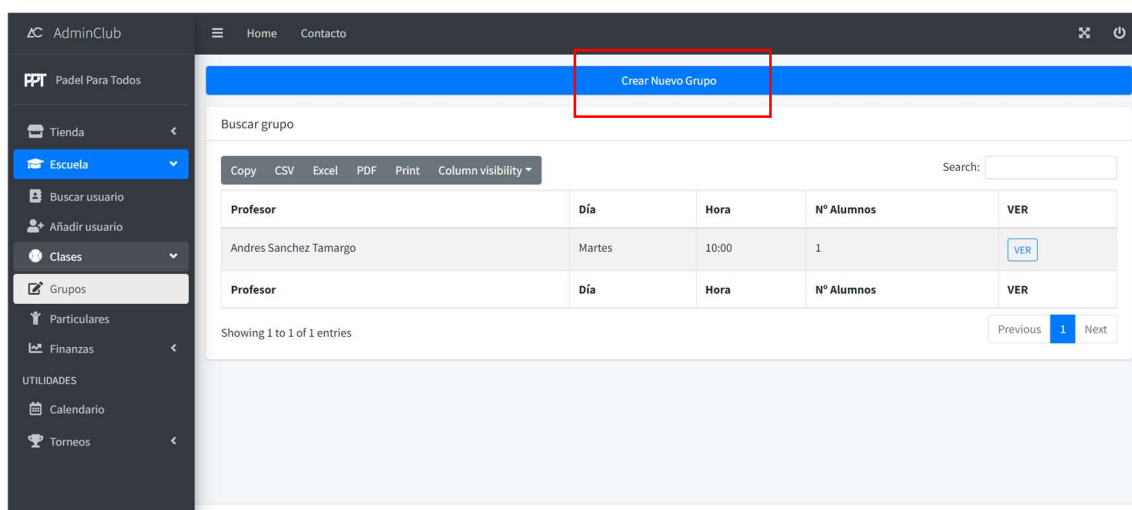


Figura 32: Crear grupo 2

Una vez dentro de la sección, accederemos a un formulario que tendremos que rellenar y, posteriormente, tendremos que pulsar en “Crear grupo”.

The screenshot shows the 'AdminClub' interface with a sidebar on the left containing navigation links like 'Tienda', 'Escuela', 'Buscar usuario', 'Añadir usuario', 'Clases', 'Grupos', 'Particulares', 'Finanzas', 'UTILIDADES', 'Calendario', and 'Torneos'. The main content area is titled 'Crear Grupo' and contains three dropdown menus: 'Profesor' (selected: Andres Sanchez Tamargo), 'Día' (selected: Domingo), and 'Hora' (selected: 12:00). Below these fields is a blue button labeled 'Crear Grupo', which is highlighted with a red rectangle.

Figura 33: Crear grupo 3

4.3. Añadir/Eliminar alumnos de un grupo.

En la lista de grupos, deberemos buscar el grupo que queremos modificar como se comentaba en el apartado 4.1. y pulsar en “Ver”.

The screenshot shows the 'AdminClub' interface with the sidebar on the left. The main content area is titled 'Crear Nuevo Grupo' and contains a search bar labeled 'Buscar grupo'. Below the search bar is a table with columns: 'Profesor', 'Día', 'Hora', 'Nº Alumnos', and 'VER'. The table contains two entries. The first entry has 'Andres Sanchez Tamargo' as the professor, 'Martes' as the day, '10:00' as the time, and '1' as the number of students. The second entry has 'Andres Sanchez Tamargo' as the professor, 'Domingo' as the day, '12:00' as the time, and '0' as the number of students. The 'VER' button for the second entry is highlighted with a red rectangle. At the bottom of the table, it says 'Showing 1 to 2 of 2 entries' and there are navigation buttons: 'Previous', '1', and 'Next'.

Profesor	Día	Hora	Nº Alumnos	VER
Andres Sanchez Tamargo	Martes	10:00	1	VER
Andres Sanchez Tamargo	Domingo	12:00	0	VER

Figura 34: Alumnos en grupo

Una vez dentro de la sección, accederemos a los detalles del grupo donde podremos consultar los alumnos que hay inscritos, inscribir a un nuevo alumno si todavía existe algún hueco en el grupo o eliminar un alumno ya inscrito.

Para añadir un alumno a un grupo pulsaremos en “Añadir” en un hueco libre.

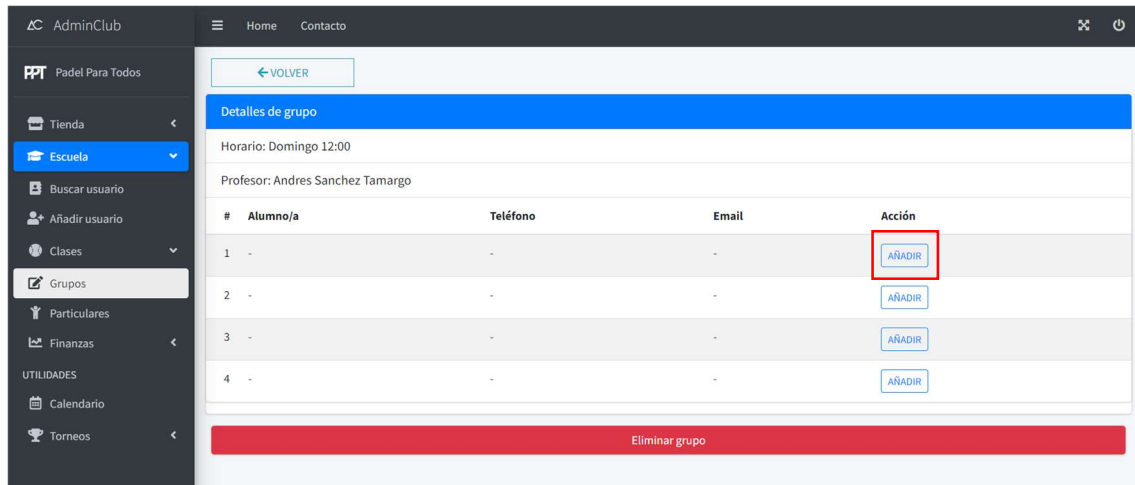


Figura 35: Alumnos en grupo 2

Una vez pulsado en “Añadir” accederemos a un menú desplegable donde podremos elegir a un usuario de categoría “alumno” para inscribirlo al grupo, se dispone de un buscador para poder encontrar de una forma más rápida al alumno a inscribir.

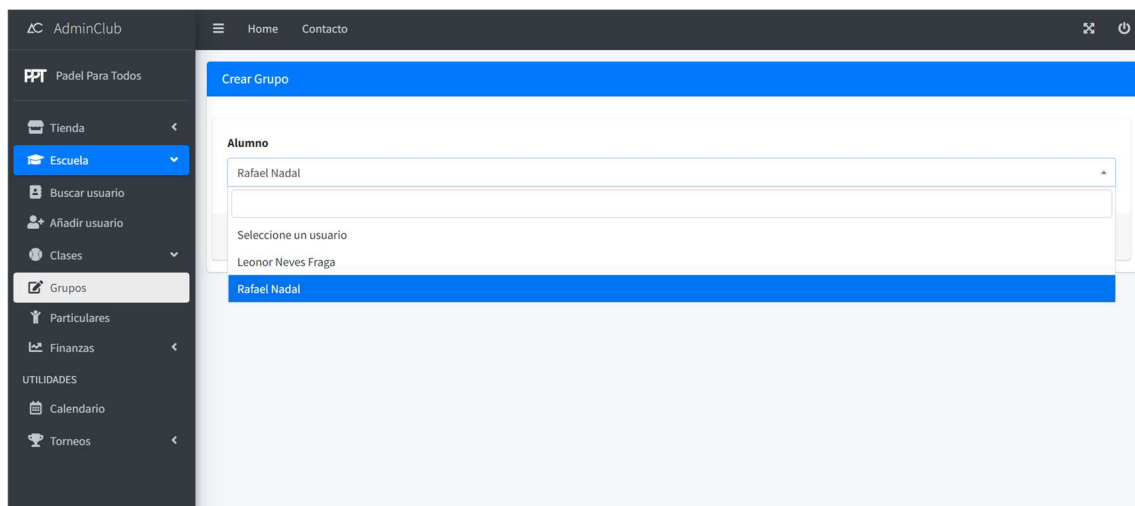


Figura 36: Alumnos en grupo 3

Una vez seleccionado el alumno a inscribir en el grupo, pulsaremos en “Añadir al grupo”.

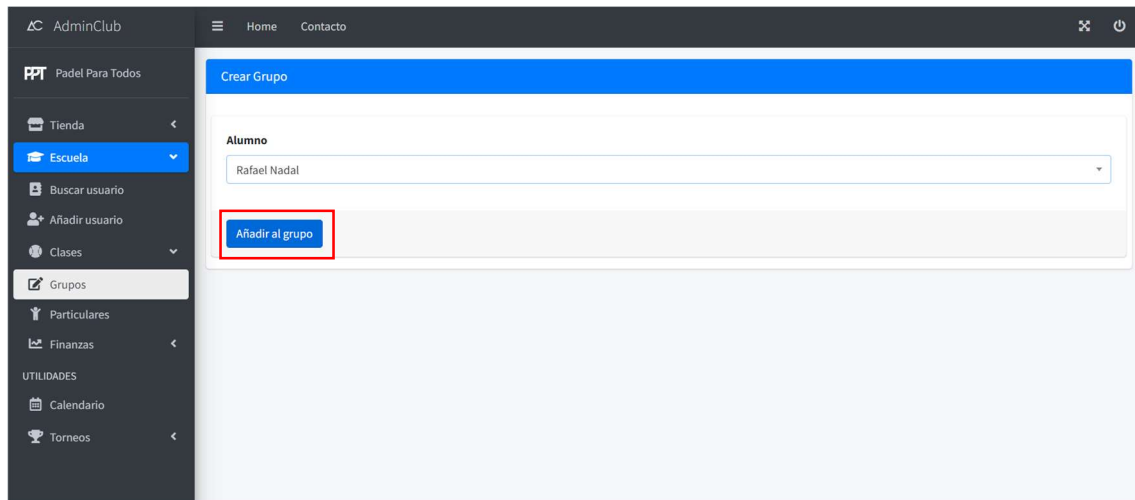


Figura 37: Alumnos en grupo 4

Para eliminar a un alumno de un grupo, entraremos en los detalles del grupo, buscaremos el alumno que deseamos eliminar y pulsaremos en “Eliminar”.

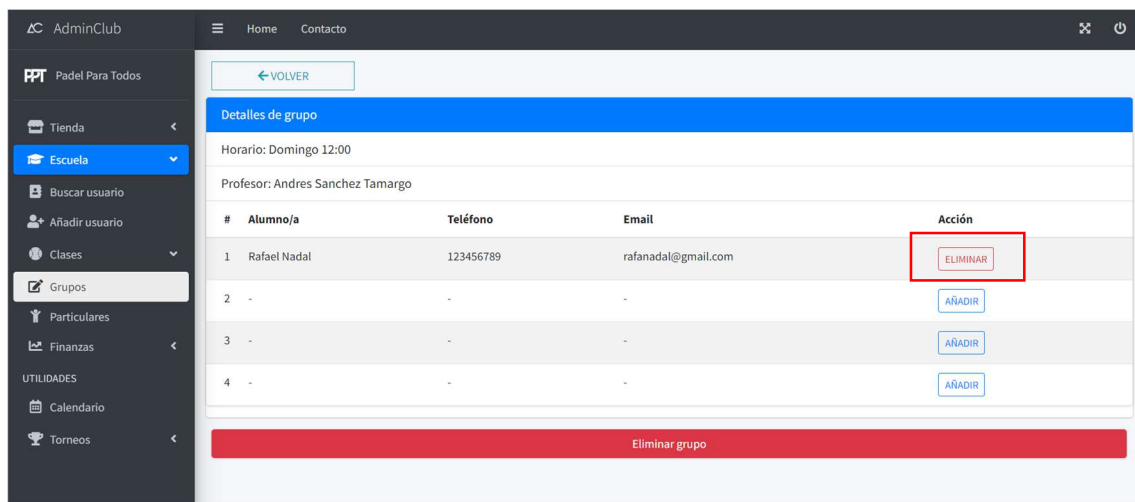


Figura 38: Alumnos en grupo 5

4.4. Eliminar grupo.

Para eliminar un grupo, accederemos a los detalles de un grupo como se especificaba en el apartado 4.3. Una vez dentro, podremos eliminar el grupo pulsando en “Eliminar grupo”.

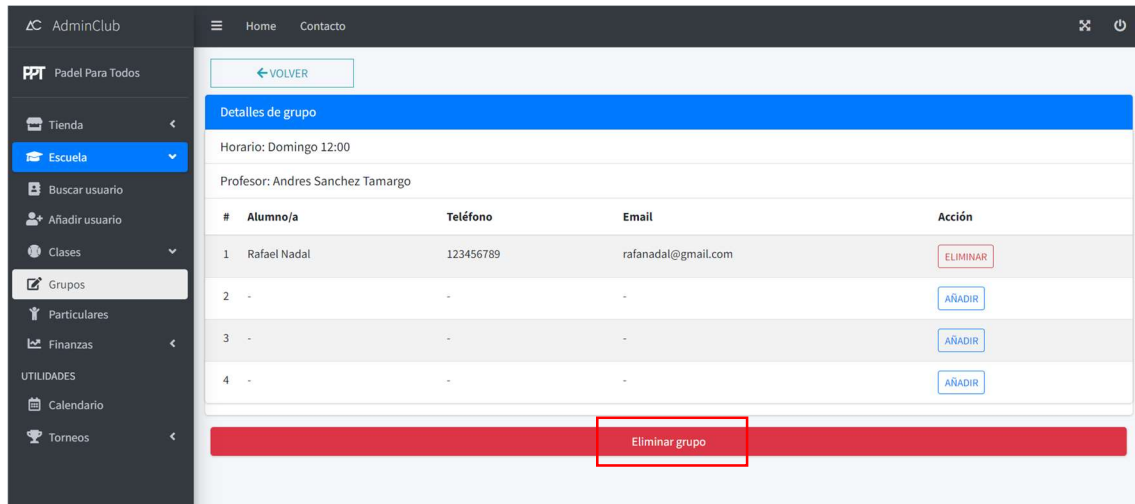


Figura 39: Eliminar grupo

4.5. Buscar bono de clases particulares.

Para poder buscar un grupo, accederemos, usando la barra de navegación lateral, a la sección Escuela > Clases > Particulares.

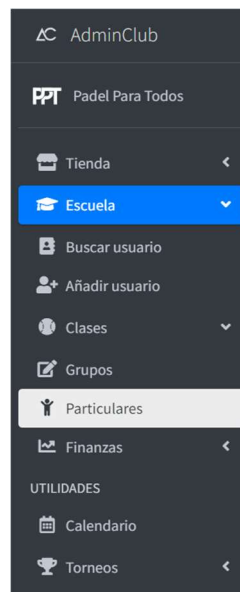


Figura 40: Particulares

Una vez dentro de la sección, accederemos a una lista con los bonos creados hasta la fecha, podremos usar la barra de búsqueda para encontrar el bono deseado.

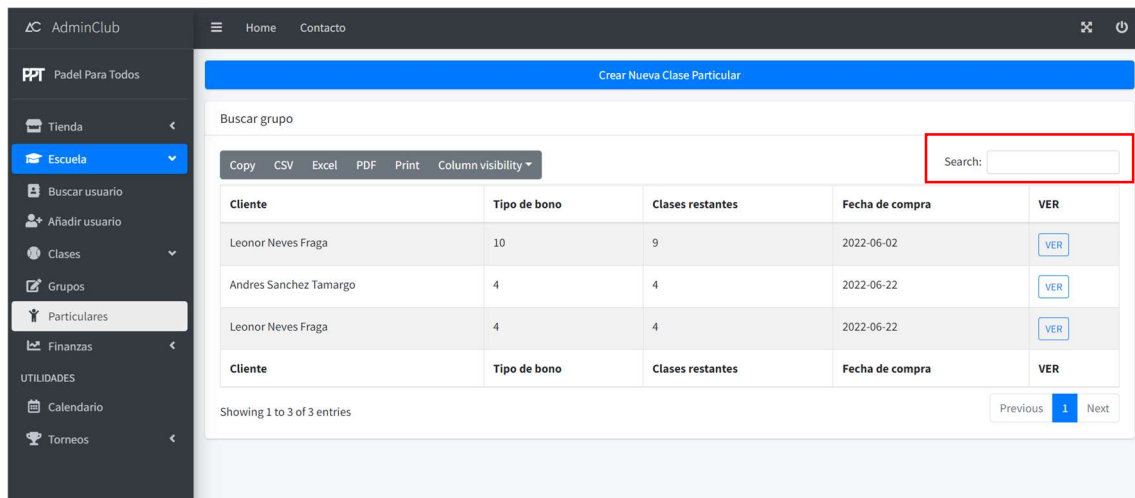


Figura 41: Particulares 2

4.6. Crear bono de clases particulares.

Para crear un bono de clases particulares debemos entrar a la lista de bonos creados como se comentaba en el apartado 4.5. y pulsar en “Crear Nueva Clase Particular”.

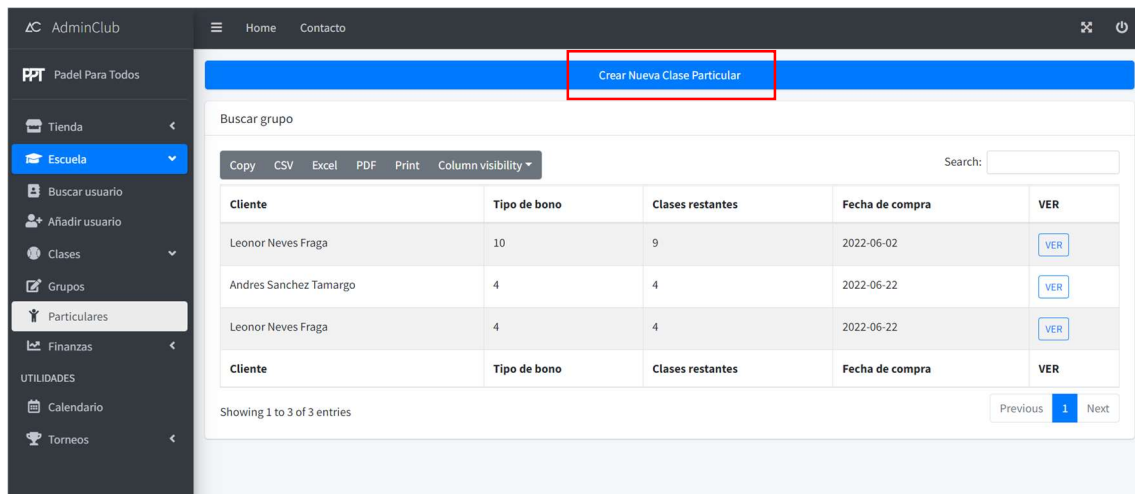


Figura 42: Crear particular

Una vez dentro de la sección, accederemos a un formulario que tendremos que rellenar y, posteriormente, tendremos que pulsar en “Confirmar venta”. La venta de ese bono, quedará registrada en la base de datos de la aplicación.

AdminClub

Home Contacto

Padel Para Todos

Tienda

Escuela

Buscar usuario

Añadir usuario

Clases

Grupos

Particulares

Finanzas

UTILIDADES

Calendario

Torneos

Crear Clase Particular

Cliente: Rafael Nadal

Tipo de bono: Bono de 10 clases (240€)

Confirmar venta

Figura 43: Crear particular 2

4.7. Marcar clase particular como impartida.

Para confirmar una clase particular como impartida, deberemos acceder a la lista de bonos como se explicaba en el apartado 4.5. y pulsar en “Ver” una vez encontrado el bono buscado.

AdminClub

Home Contacto

Padel Para Todos

Tienda

Escuela

Buscar usuario

Añadir usuario

Clases

Grupos

Particulares

Finanzas

UTILIDADES

Calendario

Torneos

Crear Nueva Clase Particular

Buscar grupo

Copy CSV Excel PDF Print Column visibility

Search:

Cliente	Tipo de bono	Clases restantes	Fecha de compra	VER
Leonor Neves Fraga	10	9	2022-06-02	VER
Andres Sanchez Tamargo	4	4	2022-06-22	VER
Leonor Neves Fraga	4	4	2022-06-22	VER
Rafael Nadal	10	10	2022-06-22	VER

Showing 1 to 4 of 4 entries

Previous 1 Next

Figura 44: Confirmar clase particular

Una vez dentro de esta sección accederemos a los detalles del bono pudiendo confirmar una clase pulsando en “Confirmar clase”. Esta clase quedará reflejada en los detalles del bono con la fecha en la que se ha impartido la clase particular y serán actualizadas las clases restantes del bono.

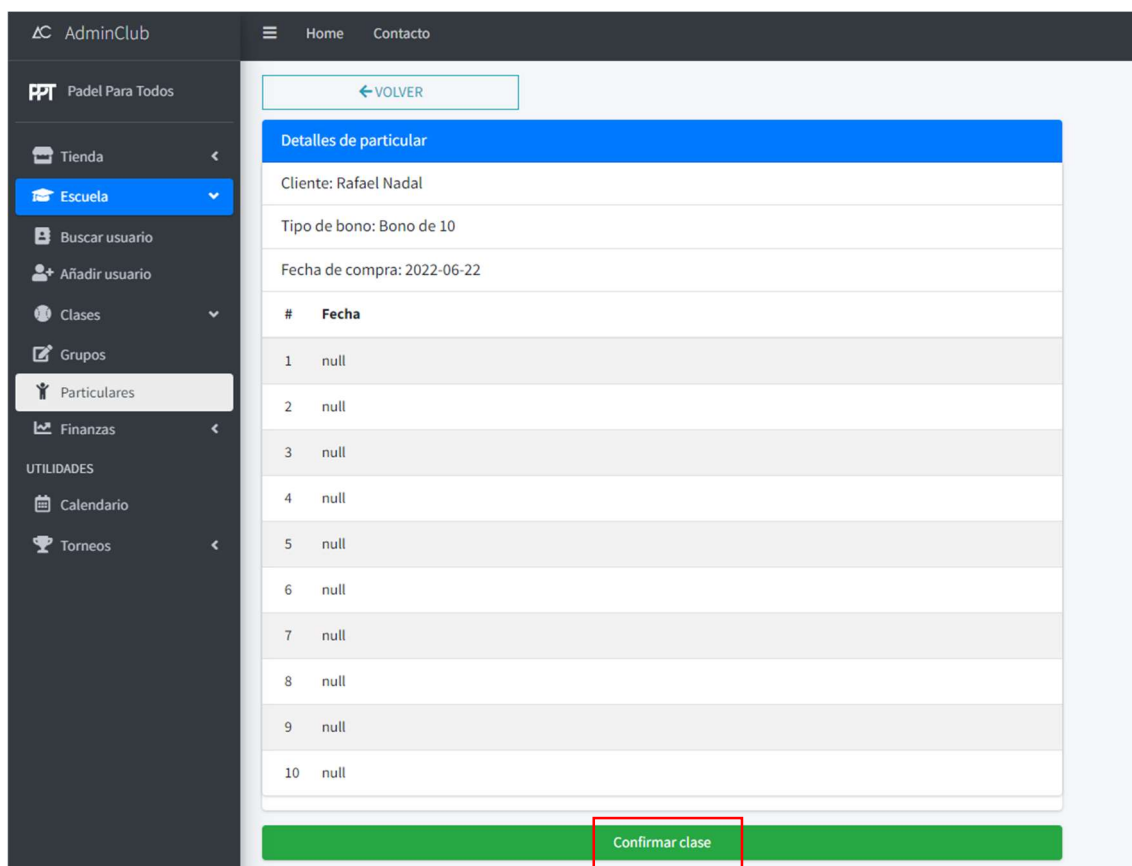


Figura 45: Confirmar clase particular 2

Detalles de particular	
Cliente: Rafael Nadal	
Tipo de bono: Bono de 10	
Fecha de compra: 2022-06-22	
#	Fecha
1	2022-06-22
2	null
3	null
4	null

Figura 46: Confirmar clase particular 3

5. Gestión de finanzas.

5.1. Buscar tarifa.

Para poder buscar un grupo, accederemos, usando la barra de navegación lateral, a la sección Finanzas > Tarifas.

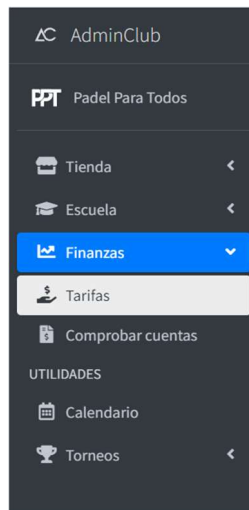


Figura 47: Tarifas

Una vez dentro de la sección, accederemos a una lista con las tarifas creadas hasta la fecha, podremos usar la barra de búsqueda para encontrar la tarifa deseada.

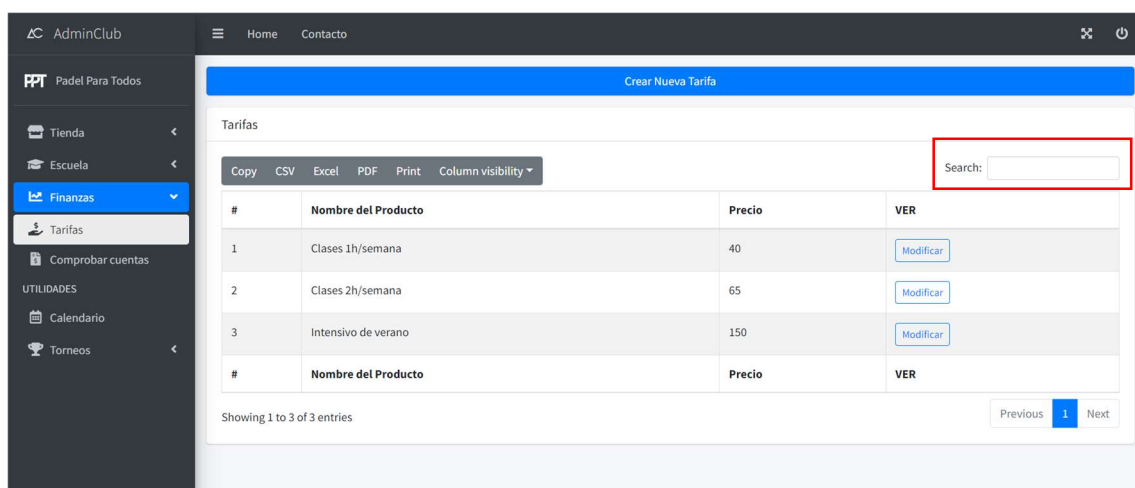


Figura 48: Tarifas 2

5.2. Crear tarifa.

Para crear una tarifa deberemos acceder al listado de tarifas como se explica en el punto 5.1. y pulsar en “Crear Nueva Tarifa”.

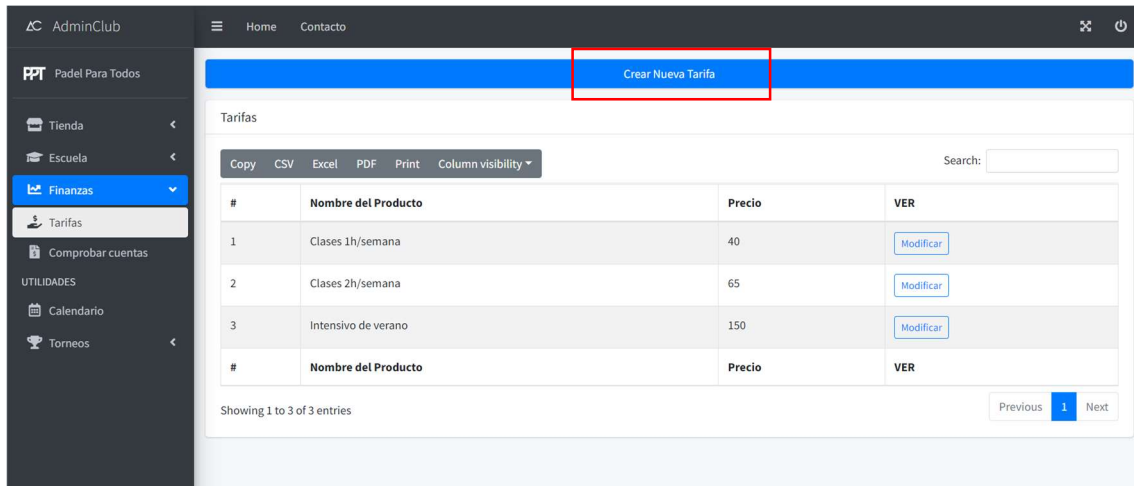


Figura 49: Crear tarifa

Una vez dentro de la sección, accederemos a un formulario que tendremos que rellenar y, posteriormente, tendremos que pulsar en “Subir Tarifa”.

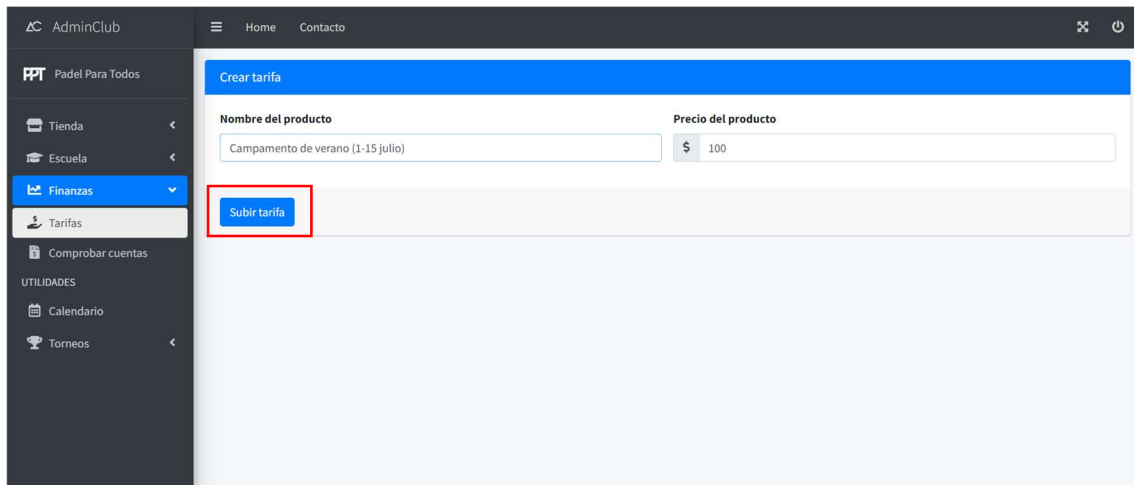


Figura 50: Crear tarifa 2

5.3. Modificar/Eliminar tarifa.

Para modificar o eliminar una tarifa deberemos buscar la tarifa sobre la que queremos trabajar como se explicaba en el apartado 5.1, y pulsaremos en “Modificar”.

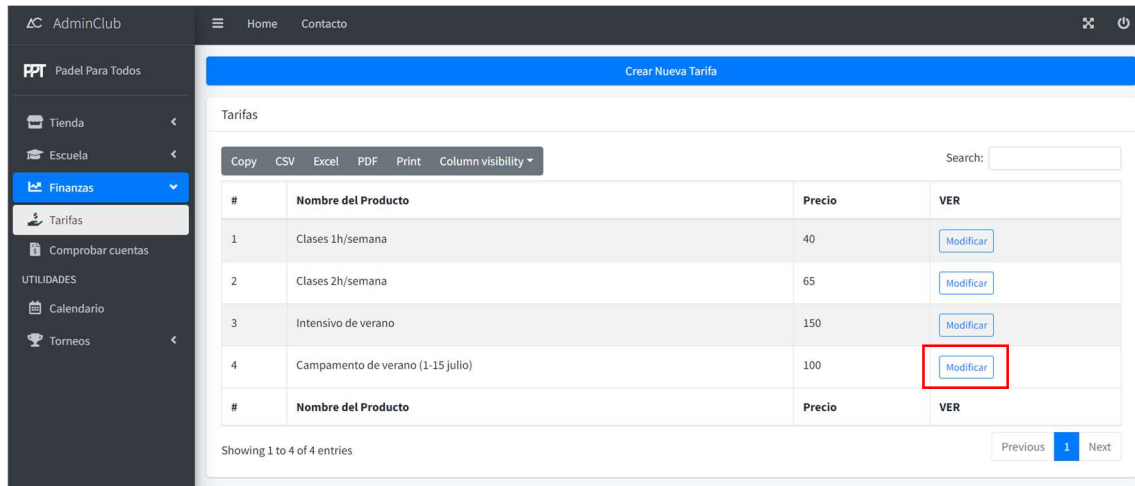


Figura 51: Modificar tarifa

Una vez dentro, accederemos a un formulario donde poder o bien modificar los datos cambiando los campos y pulsando en “Actualizar tarifa” o bien eliminar la tarifa pulsando en “Eliminar tarifa”.

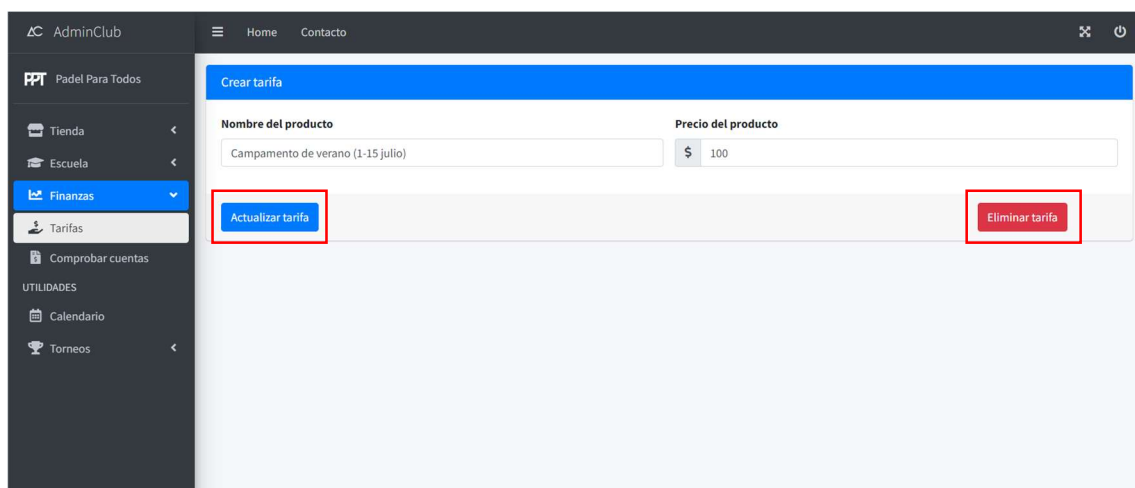


Figura 52: Modificar tarifa 2

5.4. Comprobar cuentas.

Para comprobar las cuentas del club, es decir, las ventas y cobros realizados y gestionados por la aplicación deberemos acceder, usando el menú lateral de navegación a la sección Finanzas >Comprobar cuentas.

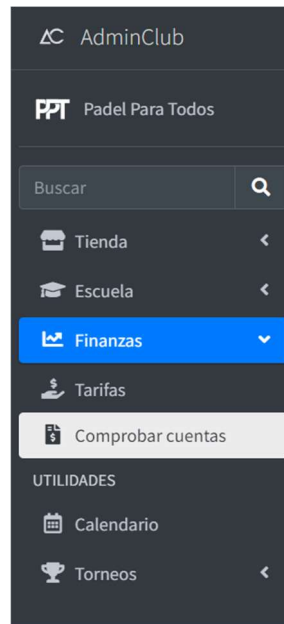


Figura 53: Cuentas

Una vez dentro de la sección, accederemos a una lista con los pagos y cobros registrados hasta la fecha, podremos usar la barra de búsqueda para encontrar el pago o cobro deseado.

The screenshot shows the AdminClub web application interface. The left sidebar contains the same navigation menu as the mobile app, with 'Finanzas' highlighted. The main content area displays a table with the following data:

Id	Producto	Cliente	Precio	Fecha
6	Sudadera pádel rosa		18	2022-06-22
5	Bono de 10 clases	Rafael Nadal	240	2022-06-22
4	Bono de 4 clases	Leonor Neves Fraga	100	2022-06-22
3	Bono de 4 clases	Andres Sanchez Tamargo	100	2022-06-22
2	Grip Bullpadel		4	2022-06-02
1	Bono de 10 clases	Leonor Neves Fraga	240	2022-06-02

At the bottom of the table, it says 'Showing 1 to 6 of 6 entries'. There is a search bar at the top right and a pagination bar at the bottom with 'Previous', '1', and 'Next' buttons.

Figura 54: Cuentas 2

Anexo 2: Summary

Index

- 1. Introduction..... 142
- 2. Objectives..... 143
- 3. Methodology 144
- 4. Chosen technologies for the development of the Project..... 144
 - 4.1. Front-end..... 144
 - 4.2. Back-end..... 144
 - 4.3. Database..... 145
- 5. System design 145
 - 5.1. System architecture 145
 - 5.2. Database design 147
- 6. System implementation 149
- 7. Future work 150

1. Introduction

Padel is a relatively new sport, which is booming, so, currently, there is still a lack of tools adapted and designed for the internal management and monitoring of padel clubs.

As a padel teacher, having worked in different padel clubs for more than 9 years, I have seen this sport grow in all aspects except in one. There are no internal management tools that allow you to keep track of the classes taught or clients, as well as the shop or various activities of interest internal to the club, which currently, are still on paper.

Large padel companies have focused on the development of applications or platforms for managing padel court reservations, leaving aside many other important functions when it comes to managing a club internally.

In this Final Degree Project (TFG), we propose the development of a web application that covers all the needs related to the internal management of a padel club that are currently not covered.

The project is limited to the design and development of an application that allows the management of clients, padel groups, private lessons, the shop and a control of the club's finances, monitoring the collections and sales made. The management of court reservations is not within the scope of the project as this functionality is covered by many other existing applications.

2. Objectives

This section will describe the objectives to be met in the development of this project, detailing the needs to be covered.

The general objective of this project is the creation of a web application that allows the internal management of a padel club. This application will manage four basic areas for the internal management of a club:

- **User Management:** You can manage both clients and teachers of the club, being able to register or withdraw a user, as well as keep track of payments made, products purchased or classes in which a user is enrolled.
- **Product Management:** It will be possible to register or withdraw new products, modify them, change the stock and manage the sale of these products.
- **Class Management:** The classes are divided into two types:
 - **Groups:** It will be possible to register or withdraw a group, register clients and assign the group to a teacher.
 - **Individuals:** It is possible to create different vouchers for individual classes (depending on the number of classes) associated with a user.
- **Financial Management:** It will be possible to keep track of the payments made either by payment of classes or for products from the club shop, as well as modifying the rates for the different services offered.

3. Methodology

This section will explain the agile methodology that has been used in the realisation of this project.

The choice of a good working methodology is crucial for the optimal development of a project and has a direct impact on the final result. During the degree, the agile methodologies studied were mainly based on teamwork. As this was an individual project, the methodology followed was a methodology adapted to the needs of this project.

Once the analysis and design of the project had been carried out, it was decided that the work method to be used would be a cycle-based work method following the pattern: front-end, back-end and testing.

The project, therefore, will have 4 cycles, one for each management area: products, users, groups and finances.

4. Chosen technologies for the development of the Project

4.1. Front-end

The structure of the pages is made with HTML, being complemented with CSS to give them design and style and JavaScript to give them interactivity. In addition, JQuery plugins have been used for the data tables throughout the application, giving them the function of a search engine, or the option of being able to download them, among others. Ajax has also been used for form autocomplete.

4.2. Back-end

The Java EE (Java Enterprise Edition) platform has been used for the development of this project. Therefore, the programming language chosen was Java, using JSP pages to create dynamic pages by embedding Java code in them, servlets to manage client requests to the server, and JPA to interact with the database.

Payara Server has also been used as an application server, being an open source application server derived from GlashFish after Oracle announced that it would discontinue commercial support.

4.3. Database

The database chosen for this project was MySQL as it is the one that best suits the needs of the project. This is because our web application has a well-defined structure, it will not have to support new user registrations, and to carry out operations with the database that relate several tables or rows, relational databases are the best option.

5. System design

5.1. System architecture

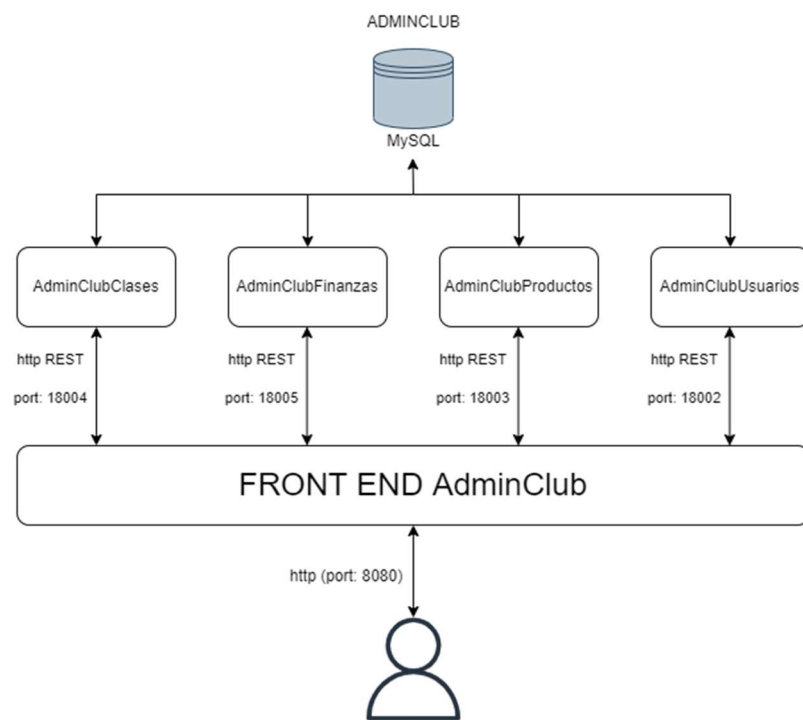


Figura 55: System architecture

As can be seen in Figura 55, the architecture chosen for the development of the application is an architecture based on microservices.

A microservice is a style or way of programming that consists of dividing the different parts of the project into blocks, according to the functionalities they are responsible for. The idea is that these microservices can work together for the complete functioning of the application but that, at the same time, if a microservice fails, the rest of the application continues to work correctly.

The application consists of a front-end, in other words, a user interface accessed through local port 8080, four microservices communicating through different ports to this front-end and a database that also communicates with these four microservices.

In the following we will explain what each microservice does:

- **AdminClubClases:** the microservice AdminClubClases, communicates with the front-end through port 18004 and is in charge of managing both the weekly classes and the club's individual vouchers. Its functions include the creation of a class, the creation of a voucher or its modifications and its deletion.
- **AdminClubFinanzas:** the AdminClubFinanzas microservice communicates with the front-end through port 18005 and is in charge of managing all the payments made in the club, whether they are payments for classes, vouchers, products or any other service offered by the club.
- **AdminClubProductos:** the AdminClubProductos microservice communicates with the front-end through port 18003 and is in charge of managing the products in the club's shop. Its functions include creating a product, modifying it or its stock and deleting it.
- **AdminClubUsuarios:** the AdminClubUsuarios microservice communicates with the front-end through port 18002 and is in charge of managing the club's users, both clients and teachers. Among its functions we can highlight the creation, modification or deletion of users or the collection of all user information, classes in which they are enrolled or payments they have made.

5.2. Database design

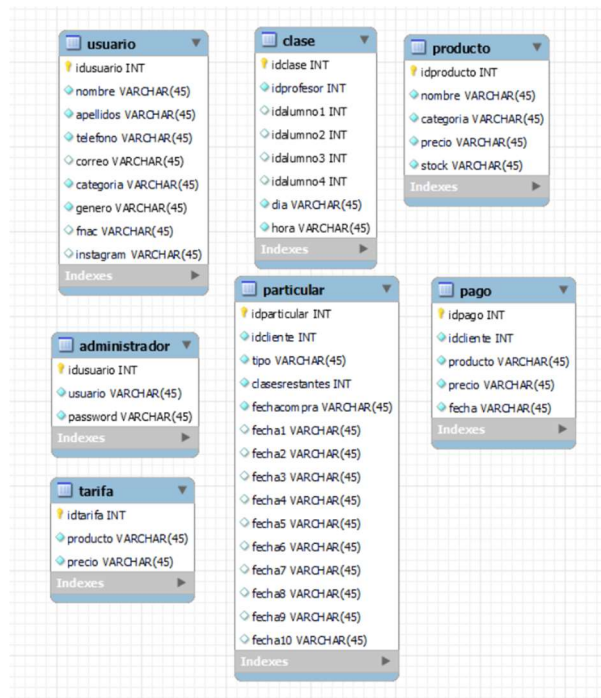


Figura 56: Database design

Figure 56 shows the final design of the database.

As can be seen, the tables are not related to each other. This is due to the large number of errors and problems that were arising when updating tables that depended on others, when performing complex queries or when deleting rows.

Due to these problems, the decision was taken not to relate the tables and to supply these relationships through the use of the front-end and back-end of the application.

To control the mandatory or optional fields, the "required" attribute was used in the forms so that, before inserting an entity in the database, it was checked, at the front-end level, that all the mandatory fields were correctly entered.

To be able to control queries involving several tables, updates or cascading deletions, the back-end was used, programming in Java, to supply the logic of the relationships.

Each of the application's tables will be explained below.

- Table “Usuarios”: collects all the data of the clients and teachers of the club. This data will be entered by the administrator using the user creation forms.
- Table “Administrador”: collects the access data of a club administrator. These access accounts will be delivered by the administration of the application once the services of our AdminClub application have been contracted.
- Table “Producto”: contains the data necessary to manage the club's products and their possible sale.
- Table “Clase”: collects the necessary data to be able to manage a paddle tennis class, a paddle tennis class being a class that is repeated weekly.
- Table “Particular”: collects the necessary data to be able to manage a voucher for private classes, these classes can be given at any time without having a pre-established date and time.
- Table “Pago”: collects all data related to payments that have been made in the club for products, classes, individuals or services offered.
- Table “Tarifa”: collects price data for various services offered by the club.

6. System implementation

This section will explain the inner workings of the application when responding to a user request.

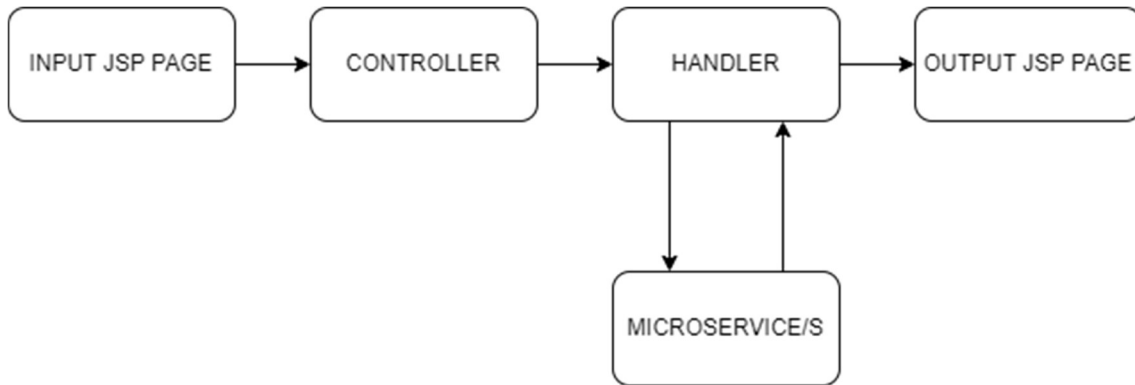


Figura 57: Inners workings of the application

Figure 57 shows how the application would work when satisfying a user request:

1. The user, from an input JSP page, would make a request. This request would be composed of the name of the handler that must satisfy it and the data necessary.
2. The controller would receive the request and would be in charge of redirecting all the data to the corresponding handler.
3. The handler retrieves the data provided to it and performs the previous operations if necessary. Once this is done, it calls the corresponding microservices specifying the necessary functions so that the microservice is in charge of performing the operations with the database.
4. The microservice, once the requested operations have been performed, returns a response confirming the operation and, if any, some data as a result of the operation performed, to the handler.
5. Finally, if any, the handler returns the data provided by the microservice to a new output JSP page.

7. Future work

In this section we will discuss different future works that could be implemented in order to create a more complete application in the future:

- English version: due to the expansion of padel around the world, a future work to be taken into account should be the possibility of being able to choose whether to use the application in its Spanish version or in an English version.
- Automatic creation of tournaments: implementation of a help tool for the creation of tournaments. The tool would work in such a way that, by registering the couples registered for a tournament, the pairings would be automatically generated.
- Calendar: implementation of a calendar where the club's future events can be visualised and noted.
- Improve database: creation of a database with interrelated tables so as not to depend on the front-end and back-end for the correct functioning of the data management within the application.
- Mobile application: development of a mobile application similar to the web application developed in the project.
- Client/teacher version: create a version for clients and teachers with chat groups for each class that is enrolled or taught so that there is communication between client and teachers. In addition, both client and teachers could keep track of the services contracted or payments made in the club.