



GRADO EN INGENIERÍA INFORMÁTICA

TRABAJO DE FINAL DE GRADO

**Panel de control para la gestión de
información relativa a apuestas deportivas**

Realizado por:
Marc PARDO FERNÁNDEZ

Supervisado por:
Juan Camilo GÓMEZ ESGUERRA
Tutorizado por:
Angel Pascual DEL POBIL FERRÉ

Fecha de lectura: 12 de Julio de 2023
Curso académico 2022/2023

Resumen

En este trabajo de fin de grado se muestra la planificación y desarrollo de un proyecto de desarrollo de software. Este proyecto se enfoca en crear una herramienta en la que apostadores deportivos profesionales pueden volcar sus datos para obtener estadísticas a partir de los mismos.

Para esto se plantea el desarrollo de un panel de control con tecnologías web y la gestión del proyecto mediante la metodología ágil Scrum. Para la vista web se emplea la biblioteca de JavaScript ReactJS junto a la herramienta Metronic, mientras que para el servidor y la base de datos se emplea el lenguaje de programación PHP y el sistema de gestión de bases de datos MySQL.

El producto a desarrollar ofrece cuatro grandes funcionalidades: gestión de usuarios, importación y consulta de datos, sistema de pagos y panel de control. Todas estas se presentan con una interfaz de usuario limpia y amigable.

Palabras clave

Panel de control, Apuestas deportivas, Estadísticas, Aplicación web

Keywords

Dashboard, Sports betting, Statistics, Web application

Índice general

1. Introducción	15
1.1. Contexto y motivación del proyecto	15
1.2. Objetivo y alcance del Proyecto	16
1.2.1. Objetivo del proyecto	16
1.2.2. Alcance del proyecto	16
1.3. Objetivo y alcance del Producto	17
1.3.1. Objetivo del producto	17
1.3.2. Alcance funcional	17
1.3.3. Alcance organizativo	18
1.3.4. Alcance informático	18
1.4. Descripción detallada del proyecto	20
1.4.1. Descripción del panel de control	21
1.4.2. Descripción del panel de datos	21
1.4.3. Descripción de la gestión de usuarios	22
1.4.4. Descripción de la gestión de pagos y suscripciones	22
1.4.5. Arquitectura y tecnologías utilizadas	23
1.5. Estructura de la memoria	24
2. Planificación del proyecto	27

2.1.	Metodología	27
2.2.	Planificación del proyecto	28
2.2.1.	Requisitos iniciales del proyecto	28
2.2.2.	Pila inicial del producto	29
2.2.3.	Priorización y estimación de la pila del producto	31
2.2.4.	Planificación de la pila del producto del primer sprint	35
2.2.5.	Mantenimiento del proyecto	37
2.3.	Costes	38
2.4.	Riesgos	40
2.4.1.	Identificación de Riesgos	40
2.4.2.	Análisis de Riesgos	40
2.4.3.	Planificación de respuesta	43
2.5.	Seguimiento del proyecto	44
2.5.1.	Sprint 1	44
2.5.2.	Sprint 2	44
2.5.3.	Sprint 3	46
2.5.4.	Sprint 4	49
2.5.5.	Sprint 5	51
2.5.6.	Sprint 6	52
3.	Análisis del sistema/producto	55
3.1.	Análisis de requisitos funcionales	55
3.2.	Análisis de requisitos no funcionales	60
4.	Diseño del sistema/producto	61
4.1.	Diseño de la base de datos	61

4.2.	Diseño de software	66
4.2.1.	Descripción general del diseño del sistema	66
4.2.2.	React hooks	66
4.2.3.	Composición	66
4.3.	Diseño de la arquitectura del sistema/producto	67
4.4.	Diseño del servidor	68
4.5.	Diseño de las interfaces	69
4.5.1.	<i>Sitemap</i>	72
5.	Implementación y pruebas	73
5.1.	Estructura del código	73
5.2.	Descripción técnica de la implementación	76
5.2.1.	Módulo de autenticación	76
5.2.2.	Módulo de importación y consulta de datos	79
5.2.3.	Página principal	84
5.2.4.	Página de suscripción	89
5.2.5.	Páginas de análisis	91
5.3.	Pruebas e implementación	91
5.3.1.	Pruebas unitarias y de la API	91
5.3.2.	Descripción técnica del despliegue	93
6.	Conclusiones	95
6.1.	Resultados del proyecto	95
6.2.	Posibles ampliaciones del proyecto	95
6.3.	Conclusiones personales	96

Índice de figuras

1.1. Logotipo de Ubik Geospatial S.L.	15
1.2. Logotipo de ReactJs	19
1.3. Logotipo de TypeScript	19
1.4. Logotipo de PHP	19
1.5. Logotipo de Bootstrap	20
1.6. Logotipos de Firebase y MySQL	20
1.7. Logotipo de PayPal	23
1.8. Logotipo de Visual Studio Code y GitHub	24
2.1. Logotipo de Taiga.io	27
2.2. Pila inicial del producto.	30
2.3. Pila del producto simplificada.	31
2.4. Épicas.	32
2.5. Pila del producto ordenada.	33
2.6. Pila del producto priorizada.	34
2.7. Resultados de la técnica Planning Poker.	35
2.8. Página de Taiga con la pila de Sprint 1.	36
2.9. Diagrama de tipo <i>Burn Down</i> del primer sprint.	44
2.10. Diagrama de tipo <i>Burn Down</i> del segundo sprint.	46

2.11. Diagrama de tipo <i>Burn Down</i> del tercer sprint.	48
2.12. Diagrama de tipo <i>Burn Down</i> del cuarto sprint.	51
2.13. Diagrama de tipo <i>Burn Down</i> del quinto sprint.	53
2.14. Diagrama de tipo <i>Burn Down</i> del sexto sprint.	54
4.1. Diseño conceptual inicial de la base de datos.	62
4.2. Diseño lógico inicial de la base de datos.	63
4.3. Diseño lógico final de la base de datos.	64
4.4. Diseño de la arquitectura del producto.	68
4.5. Plantilla base de Metronic.	69
4.6. Boceto de la página principal del producto.	70
4.7. Boceto de la tabla de datos.	70
4.8. Boceto de la página de análisis.	71
4.9. Boceto de la página de pagos.	71
4.10. <i>Sitemap</i> del proyecto.	72
5.1. Directorios del proyecto.	74
5.2. Página de recuperación de contraseña.	76
5.3. Página de inicio de sesión.	77
5.4. Página de registro.	78
5.5. Página de importación de datos.	80
5.6. Página de consulta de datos.	81
5.7. Página de importación de datos con filtros aplicados.	82
5.8. Página de importación de datos con filas seleccionadas.	83
5.9. Página principal del proyecto.	84
5.10. Fallo cargando datos en un componente.	85

5.11. Componente <i>Summary</i> de la página principal.	86
5.12. Componente <i>Markets</i> de la página principal.	86
5.13. Componente <i>Profit statistics</i> de la página principal.	87
5.14. Componente <i>Profit/Loss balance</i> de la página principal.	87
5.15. Componente <i>Markets performance</i> de la página principal.	88
5.16. Componente de filtrado presente en la página principal.	88
5.17. Campo de entrada de fechas presente en el filtro.	89
5.18. Página de suscripción del proyecto.	90
5.19. Botones de pago de Paypal.	90
5.20. Página de análisis de mercados del proyecto.	91
5.21. Logotipo de Postman	92
5.22. Llamada con token inválido al servidor utilizando Postman	92

Índice de cuadros

2.1. Salario de los implicados en el desarrollo.	38
2.2. Coste de las herramientas empleadas en el desarrollo.	39
2.3. Coste del equipo empleado en el desarrollo.	39
2.4. Identificación de riesgos del proyecto.	40
2.5. Análisis de riesgos 1	41
2.6. Análisis de riesgos 2	42
2.7. Planificación de respuesta a los riesgos del proyecto.	43
2.8. Riesgos identificados durante la planificación del segundo sprint.	45
2.9. Riesgos identificados durante la planificación del tercer sprint.	48
4.1. <i>Endpoints</i> personalizados añadidos a PHP-CRUD-API.	68

Listings

4.1. Diseño físico de la base de datos.	61
4.2. Disparador de la base de datos.	65
4.3. Composición con componentes genéricos en React.	67
4.4. Composición con componentes específicos en React.	67
5.1. Formato de los archivos .tsx.	74
5.2. Formato de los archivos .ts.	75
5.3. Fragmento de las pruebas unitarias para la función de transformación de datos. .	92
5.4. Archivo de configuración .htaccess.	93

Capítulo 1

Introducción

1.1. Contexto y motivación del proyecto

A continuación se presenta la memoria técnica del proyecto final de grado de Marc Pardo Fernández. El proyecto se llevó a cabo en Ubik Geospatial SL, ubicada en el Edificio ESPAITEC 2 de la Universidad Jaume I de Castellón. Esta empresa, fundada como *spin-off* universitario a finales de 2016, se dedica a crear herramientas para ayudar a personas y organizaciones a tomar decisiones más inteligentes. El logotipo de la empresa se puede observar en la figura 1.1.



Figura 1.1: Logotipo de Ubik Geospatial S.L.

Con el objetivo de brindar apoyo a apostadores deportivos profesionales y dar soporte a su toma de decisiones, Ubik propuso el desarrollo de una herramienta. El proyecto consistió en la creación de un panel de control que reflejará el rendimiento de los apostadores y proporcionará estadísticas basadas en sus resultados.

Este proyecto surgió a partir de las necesidades de uno de los clientes de la empresa, quien reconoció el gran potencial de un sistema de este tipo en el ámbito de las apuestas deportivas. La herramienta ofrecería una solución eficiente, efectiva y fácil de usar para abordar el problema común de la falta de seguimiento y control de los resultados en las apuestas deportivas.

En resumen, este proyecto desarrollado por Ubik Geospatial SL tiene un gran potencial y puede marcar la diferencia en la toma de decisiones de los apostadores deportivos profesionales, brindando soporte en el desarrollo de su actividad económica y alineándose perfectamente con la filosofía de la empresa.

1.2. Objetivo y alcance del Proyecto

1.2.1. Objetivo del proyecto

El objetivo principal de este proyecto es desarrollar un panel de control deportivo innovador y especializado para atender las necesidades de apostadores profesionales. Se busca aprovechar una oportunidad en el mercado al proporcionar una solución única y completa que permita a estos usuarios acceder a información deportiva relevante de manera intuitiva y eficiente. El enfoque se centra en desarrollar un servicio web que brinde herramientas avanzadas de análisis, estadísticas y otras características específicas para mejorar la toma de decisiones en el ámbito de las apuestas deportivas. El objetivo final es ayudar a los apostadores deportivos profesionales a maximizar sus oportunidades y lograr resultados exitosos en sus actividades de apuestas deportivas.

1.2.2. Alcance del proyecto

En cuanto al alcance del proyecto, se deben llevar a cabo las siguientes tareas para su ejecución:

- **Diseño de base de datos:** Esta tarea implicará diseñar la estructura de la base de datos que respaldará el panel de control deportivo. Se definirán las tablas, relaciones y campos necesarios para almacenar la información requerida.
- **Diseño de UI (Interfaz de Usuario):** En esta tarea se trabajará en el diseño visual y la usabilidad del panel de control. Se crearán bocetos y se definirá la disposición de los elementos en la interfaz para asegurar una experiencia de usuario intuitiva y atractiva.
- **Desarrollo Front-end:** Aquí se implementará la parte visual del panel de control utilizando tecnologías como HTML, CSS y TypeScript. Se traducirán los diseños de la interfaz en código funcional, asegurando una navegación fluida y una presentación visual coherente.
- **Desarrollo Back-end:** En esta tarea se construirá la lógica y funcionalidad del lado del servidor. Se utilizará PHP para desarrollar las API y procesar las solicitudes de los usuarios, así como para interactuar con la base de datos y proporcionar los datos necesarios al front-end.
- **Tests unitarios de funciones complejas:** Se realizarán pruebas unitarias exhaustivas para garantizar el correcto funcionamiento de las funciones complejas del panel de control. Esto permitirá detectar y corregir posibles errores y asegurar la calidad del software.
- **Documentación:** Esta tarea implica generar documentación detallada sobre el proyecto. Se incluirán descripciones de la estructura de la base de datos, explicaciones del código y funciones clave, instrucciones de instalación y uso, así como cualquier otra información relevante para futuros desarrolladores o usuarios del panel de control.

No se llevan a cabo pruebas en el servidor debido a que no se cuenta con las herramientas ni el entorno necesarios para su realización. Sin embargo, se realizan pruebas directamente en la

API, evaluando cada uno de los *endpoints* de manera exhaustiva. Este enfoque permite asegurar el correcto funcionamiento de la API y validar su integración con el front-end, garantizando así la calidad y la funcionalidad del sistema en su conjunto.

1.3. Objetivo y alcance del Producto

1.3.1. Objetivo del producto

El principal objetivo de este producto es ofrecer una solución rápida, sencilla y confiable que permita a los apostadores deportivos calcular sus estadísticas y beneficios de manera eficiente y precisa. Esta herramienta soluciona un problema muy presente en el campo de las apuestas deportivas, como es la complejidad en el cálculo de estadísticas y beneficios. En consecuencia, sirve de ayuda a los apostadores deportivos profesionales a desempeñar de forma más eficiente su profesión.

El objetivo principal se puede desglosar en los siguientes subobjetivos:

- Proporcionar una solución que sea accesible y asequible para los apostadores deportivos, independientemente de su nivel de experiencia.
- Diseñar un entorno amigable y fácil de usar que permita a los usuarios visualizar sus datos de manera clara y precisa.
- Automatizar todos los cálculos necesarios para ayudar a la toma de decisiones de los apostadores deportivos profesionales.
- Proporcionar soporte y actualizaciones regulares al usuario, garantizando así que la herramienta se mantenga actualizada y sea útil a largo plazo.

En conclusión, este producto busca desarrollar una herramienta eficiente y precisa que permita a los apostadores deportivos calcular sus estadísticas y beneficios de forma rápida y sencilla. Se enfoca en brindar accesibilidad, usabilidad y automatización, con el objetivo de apoyar a los apostadores deportivos profesionales en su desempeño y ofrecer un servicio útil y actualizado a largo plazo.

1.3.2. Alcance funcional

El alcance funcional de este producto se divide en varios aspectos clave: la gestión de cuentas de usuario, el registro y almacenamiento de datos de apuestas deportivas, la contratación de planes de pago y la consulta de datos y estadísticas a través de un panel de control personalizado.

En cuanto a la gestión de usuarios, se busca ofrecer una experiencia de uso sencilla y amigable. Los usuarios podrán iniciar sesión mediante cuentas de Google existentes o crear una cuenta específica para esta plataforma. El registro de usuario requerirá solo una dirección de

correo electrónico y contraseña, con requisitos de seguridad mínimos. Se habilitará una opción de recuperación de contraseña a través del correo electrónico y se permitirá la modificación de los datos del usuario.

El registro y almacenamiento de datos de apuestas deportivas se simplificará mediante una tabla que permitirá a los usuarios ingresar directamente los datos en un formato predefinido. Los usuarios podrán subir los datos y verlos actualizados en el panel de control. La aplicación también ofrecerá la consulta de información a través de la tabla, con opciones de filtrado para los campos relevantes de las apuestas deportivas.

La contratación de planes de pago permitirá a los usuarios ampliar los servicios de la aplicación. Se creará una cuenta gratuita por defecto y se podrán actualizar a planes de pago mediante el procesador de pago Paypal [13]. Se gestionará el control de los servicios contratados y se bloqueará el acceso en caso de falta de pago o cambio de plan.

El principal enfoque de la aplicación es la consulta de datos y estadísticas a través de un panel de control personalizado. Se mostrarán datos relevantes para los apostadores deportivos profesionales, como comisiones pagadas, depósitos, ganancias, rendimiento en competiciones, balance de beneficios y pérdidas, ROI, entre otros. El contenido exacto del panel de control se definirá en base a la retroalimentación recibida después de realizar pruebas con posibles usuarios finales.

1.3.3. Alcance organizativo

El alcance organizativo de este producto se limita a la empresa Ubik Geospatil SL y el cliente con el que se está tratando. Al tratarse de un desarrollo pequeño dentro de la empresa solo han trabajado en el mismo Marc Pardo Fernández y Jose Francisco Ramos Romero.

1.3.4. Alcance informático

El desarrollo del producto se ha llevado a cabo como una aplicación web reactiva utilizando las tecnologías ReactJS, PHP y TypeScript. Los logotipos de estas herramientas se encuentran en las figuras 1.2, 1.3 y 1.4.

ReactJS [1] es una biblioteca de JavaScript de código abierto desarrollada por Facebook, que se emplea para construir interfaces de usuario en aplicaciones web de una sola página. Su enfoque se basa en componentes, que son bloques de código reutilizables que se combinan para crear interfaces de usuario complejas y dinámicas.

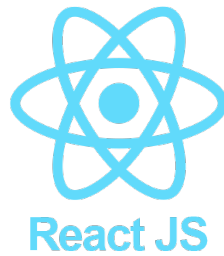


Figura 1.2: Logotipo de ReactJs

TypeScript [21] es una herramienta de JavaScript que añade características orientadas a objetos, tipado estático y otras funcionalidades a JavaScript. TypeScript es compatible con todas las bibliotecas y herramientas de JavaScript existentes.



Figura 1.3: Logotipo de TypeScript

Por último, PHP [8] es un lenguaje de programación de código abierto utilizado en el desarrollo web. Permite generar contenido dinámico en páginas web y se combina con HTML y CSS. PHP es compatible con diversas bases de datos y se utiliza para crear aplicaciones web interactivas. Es popular por su flexibilidad y facilidad de uso.



Figura 1.4: Logotipo de PHP

La combinación de estas tres herramientas, junto con otras como Bootstrap [2] (una biblioteca de código abierto diseñada para facilitar el desarrollo de interfaces de usuario adaptables), permite desarrollar una aplicación web sólida, escalable y de rápida implementación. Su logotipo se encuentra en el figura 1.5.

En cuanto a la persistencia de datos, se utilizan dos bases de datos: una para la gestión de usuarios y otra para almacenar los datos de las apuestas deportivas realizadas por los usuarios. Para la gestión de usuarios se empleará el servicio de autenticación de Google Firebase Auth [6], una solución fácil de usar y segura que permite a los usuarios iniciar sesión y acceder a los recursos de la aplicación. Para almacenar los datos de las apuestas deportivas, se utiliza una base de datos MySQL [12], un sistema de gestión de bases de datos relacional de código abierto que ofrece alta velocidad, confiabilidad y escalabilidad. Tanto el logotipo de MySQL como el de Firebase los podemos encontrar en la figura 1.6.



Figura 1.5: Logotipo de Bootstrap



Figura 1.6: Logotipos de Firebase y MySQL

1.4. Descripción detallada del proyecto

El proyecto tiene como objetivo abordar un problema existente en el ámbito de las apuestas deportivas. Actualmente, los apostadores profesionales se enfrentan a diversos desafíos para extraer información relevante de sus movimientos previos y tomar decisiones informadas para maximizar sus oportunidades de ganancias. Los apostadores requieren herramientas especializadas que les permitan analizar y procesar grandes volúmenes de datos en tiempo real, identificar patrones y tendencias, y realizar cálculos estadísticos complejos para evaluar el rendimiento de los equipos y eventos deportivos entre otros.

El proyecto busca cubrir este hueco en el mercado al proporcionar un panel de control deportivo integral y especializado. Este panel de control permitirá a los apostadores profesionales acceder a herramientas analíticas avanzadas para tomar decisiones informadas y estratégicas en sus apuestas. La solución ofrecerá una interfaz intuitiva que permitirá a los usuarios visualizar datos deportivos de manera eficiente y efectiva.

Con esta solución, se busca facilitar el proceso de toma de decisiones para los apostadores profesionales, brindándoles una herramienta poderosa y especializada que les permita evaluar las probabilidades, analizar los resultados pasados, identificar oportunidades y gestionar sus estrategias de apuestas. En última instancia, el proyecto tiene como objetivo proporcionar un valor agregado significativo a los apostadores profesionales, permitiéndoles mejorar su rendimiento y maximizar sus ganancias en el ámbito de las apuestas deportivas.

A continuación se va a describir con mayor detalle las funcionalidades que brinda el producto y las tecnologías empleadas para desarrollar el proyecto.

1.4.1. Descripción del panel de control

El panel de control es la parte central de la herramienta desarrollada en este proyecto. Proporciona una interfaz intuitiva y fácil de usar que permite a los apostadores deportivos visualizar y analizar sus datos de manera clara y precisa.

El panel de control muestra diferentes secciones y elementos que brindarán información relevante sobre el rendimiento de los apostadores y sus estadísticas. Algunos de los elementos que se incluirán en el panel de control son:

- **Resumen general:** Esta sección proporcionará un resumen de los datos clave, como el balance de beneficios y pérdidas, el ROI (*Return on Investment*), las ganancias totales, el número de apuestas realizadas, etc. Brindará una visión general del rendimiento general de los apostadores.
- **Gráficos y visualizaciones:** Se mostrarán gráficos y visualizaciones interactivas que permitirán a los usuarios analizar sus resultados de manera más detallada. Estos gráficos pueden incluir el rendimiento a lo largo del tiempo, la distribución de las apuestas por deporte, el porcentaje de aciertos, etc. Estas visualizaciones ayudarán a identificar patrones y tendencias en los resultados de las apuestas.

Esta sección del proyecto se presenta de manera menos detallada, ya que está sujeta a ajustes y modificaciones en función de los comentarios recibidos durante las pruebas de demostración de la aplicación. La intención es recopilar retroalimentación de los usuarios para definir de manera precisa la funcionalidad y características de la aplicación. A medida que se avance en el proceso de pruebas y se obtengan los comentarios pertinentes, esta sección se desarrollará con mayor detalle, brindando una visión completa del proyecto.

1.4.2. Descripción del panel de datos

En esta sección se proporciona una descripción detallada del panel de datos, el cual es una parte fundamental de la herramienta desarrollada. El panel de datos permite a los usuarios gestionar y visualizar la información relacionada con sus apuestas deportivas. A continuación, se presentan algunas de las funcionalidades clave del panel de datos:

- **Tabla de movimientos:** En esta sección se muestra una tabla que contiene información detallada sobre cada apuesta, comisión, ingreso y retirada de capital realizada por los usuarios. Los datos incluidos en la tabla pueden ser la fecha y hora de la apuesta, el deporte en el que se realizó, el resultado de la apuesta, la cuota, la cantidad apostada, entre otros. Los usuarios tienen la posibilidad de ordenar y filtrar la tabla según sus necesidades, lo que les permite acceder rápidamente a la información relevante.
- **Importación de datos:** El panel de datos permite a los usuarios importar sus datos de apuestas desde fuentes externas, como el mercado de intercambio Orbit. Se ofrece la opción de copiar y pegar directamente el texto de la sección de apuestas del mercado de

intercambio, lo que facilita la transferencia de datos sin necesidad de ingresar manualmente cada apuesta. Esta funcionalidad agiliza el proceso y asegura la precisión de la información importada.

En conclusión, el panel de datos proporciona a los usuarios la capacidad de gestionar y visualizar su historial de apuestas deportivas de manera eficiente. La tabla de apuestas permite un acceso rápido y ordenado a la información relevante, mientras que la funcionalidad de importación de datos desde el mercado de intercambio Orbit simplifica el proceso y mejora la precisión de los datos importados.

1.4.3. Descripción de la gestión de usuarios

A continuación hablaremos de la gestión de usuarios, uno de los aspectos fundamentales de la herramienta desarrollada. El objetivo principal es proporcionar a los usuarios un sistema seguro y confiable para acceder a la plataforma y gestionar sus cuentas. Para lograrlo, se utiliza Firebase Auth de Google, un servicio de autenticación ampliamente utilizado en el desarrollo de aplicaciones web.

Firebase Auth ofrece a los usuarios la posibilidad de iniciar sesión utilizando sus cuentas de Google existentes, lo que simplifica el proceso de autenticación y mejora la experiencia de usuario. Al utilizar esta tecnología, se garantiza un inicio de sesión seguro y protegido.

Es importante destacar que la herramienta ha sido diseñada teniendo en cuenta la privacidad y protección de datos de los usuarios. Se recopila y almacena la mínima información necesaria para la gestión de cuentas y la autenticación.

1.4.4. Descripción de la gestión de pagos y suscripciones

La aplicación proporciona una variedad de planes de pago diseñados para satisfacer las necesidades individuales de los usuarios. Estos planes permiten aumentar la cantidad de registros que los usuarios pueden agregar en la plataforma, brindándoles una mayor flexibilidad y capacidad de almacenamiento de datos.

Al acceder a la sección de contratación de planes de pago, los usuarios tienen la opción de seleccionar entre diferentes alternativas de suscripción. Cada plan ofrece un límite de registros superior al del plan gratuito, lo que les permite administrar un mayor volumen de datos relacionados con las apuestas deportivas.

El proceso de pago se realiza a través de PayPal, lo que brinda a los usuarios la posibilidad de efectuar el pago utilizando una tarjeta de crédito o su cuenta de PayPal existente. Una vez que se haya completado exitosamente el proceso de pago, la cuenta del usuario se actualizará automáticamente, otorgándole acceso a los servicios adicionales correspondientes al plan de pago seleccionado. Podemos encontrar el logotipo de Paypal en la figura 1.7.



Figura 1.7: Logotipo de PayPal

1.4.5. Arquitectura y tecnologías utilizadas

La arquitectura de la aplicación se basa en una estructura cliente-servidor, donde el cliente es el panel de control que interactúa con el servidor para obtener y actualizar los datos. Se utilizarán tecnologías web modernas y ampliamente utilizadas para desarrollar tanto el cliente como el servidor.

En el lado del cliente, se utiliza ReactJS como biblioteca principal para construir la interfaz de usuario del panel de control. ReactJS es conocido por permitir crear componentes reutilizables y dinámicos, facilitando así el desarrollo de interfaces interactivas y responsivas. Para el diseño y la estilización de la interfaz, se emplea Metronic [10], una poderosa herramienta construida sobre Bootstrap 5. Metronic ofrece un conjunto de estilos predefinidos y componentes altamente personalizables, lo que permitirá desarrollar proyectos de manera eficiente y adaptar la interfaz del panel de control de acuerdo a las necesidades específicas.

En el servidor del proyecto se utiliza PHP, un lenguaje de programación de código abierto ampliamente utilizado en el desarrollo web. Para simplificar la creación de una API CRUD (Create, Read, Update, Delete) en PHP, se aprovecha una herramienta llamada PHP-CRUD-API [22], desarrollada por Maurits van der Schee (mevdschee en GitHub). Esta herramienta ofrece una configuración automática y rápida de una API que maneja las operaciones básicas en una base de datos. Con PHP-CRUD-API, podrás definir *endpoints* personalizados para obtener datos estadísticos directamente desde el servidor, evitando la necesidad de calcularlos en el cliente de la aplicación. Esta solución eficiente y sencilla agilizará el desarrollo del servidor del proyecto, permitiéndote implementar una API CRUD en PHP sin tener que escribir todo el código desde cero.

Para la comunicación entre el cliente y el servidor, se utiliza una API RESTful. Estas API permiten establecer una comunicación segura y eficiente entre el cliente y el servidor a través de solicitudes HTTP. El servidor proporciona *endpoints* (puntos finales) que el cliente podrá utilizar para enviar solicitudes y recibir respuestas con los datos necesarios.

En cuanto al almacenamiento de datos, se utilizan dos bases de datos. Para la gestión de usuarios y la autenticación, se empleará Google Firebase Auth, un servicio seguro y fácil de usar que permite a los usuarios iniciar sesión y acceder a los recursos de la aplicación. Para almacenar los datos de las apuestas deportivas, se utiliza una base de datos MySQL. MySQL es un sistema de gestión de bases de datos relacional de código abierto, ampliamente utilizado

y confiable.

Por otra parte se ha optado por utilizar GitHub [7] como plataforma para alojar el proyecto utilizando el sistema de gestión de versiones Git y Visual Studio Code [11] como editor de código principal. Visual Studio Code se ha seleccionado por la experiencia previa del equipo de desarrollo con el mismo y por ser gratuito, ofrecer soporte multilenguaje, su gran variedad de extensiones y por su integración con Git, entre otras ventajas. Los logotipos de ambas herramientas se encuentra en la figura 1.8.



Figura 1.8: Logotipo de Visual Studio Code y GitHub

1.5. Estructura de la memoria

A continuación se proporcionará una breve explicación de cada sección de la memoria y su contenido.

- **Sección 2: Planificación del proyecto**

En esta sección se abordará el tema de las metodologías ágiles y su utilización en el proyecto, la planificación del proyecto utilizando estas metodologías, el seguimiento que se llevará a cabo para asegurar su progreso, los costos estimados y los posibles riesgos que puedan surgir.

- **Sección 3: Análisis del producto**

En esta sección se establecerán los requisitos del producto a desarrollar. Se examinarán las necesidades de los apostadores profesionales y se identificarán las funcionalidades y características que deben incorporarse en la herramienta.

- **Sección 4: Diseño del producto**

En esta sección se llevará a cabo el diseño del producto. Se diseñará la base de datos que contendrá la información necesaria, se diseñará el software que conformará el panel deportivo y se establecerá la estructura general del sistema o producto. Además, se diseñarán las interfaces que permitirán a los usuarios interactuar con la herramienta.

- **Sección 5: Implementación y pruebas**

En esta sección se llevará a cabo la implementación del producto, es decir, se escribirá el código necesario para que funcione correctamente. Se describirá la estructura del código y se proporcionará una descripción técnica de la implementación. También se realizarán pruebas para verificar y validar que el sistema o producto cumple con los requisitos establecidos.

- **Sección 6: Conclusiones**

En esta sección se presentarán las conclusiones obtenidas a lo largo del proyecto. Se analizará el cumplimiento de los objetivos establecidos y se evaluará el valor agregado proporcionado a los apostadores profesionales. Se destacarán los logros alcanzados y se ofrecerán posibles recomendaciones para futuras mejoras.

Capítulo 2

Planificación del proyecto

2.1. Metodología

Este proyecto cuenta inicialmente con cuatro requisitos generales, cada uno con una definición proporcionada por el cliente, que varía en nivel de detalle según el requisito en cuestión. A lo largo del desarrollo, estos requisitos se irán refinando mediante reuniones con el cliente. Dado que la mayoría de los requisitos carecen de especificaciones detalladas se ha optado por utilizar una metodología ágil similar a Scrum [15], adaptándola según las necesidades específicas del proyecto y del equipo de desarrollo.

Con el objetivo de garantizar una gestión eficiente del progreso del desarrollo y la documentación, se utilizará la plataforma de gestión de proyectos Taiga [19]. Taiga es una herramienta de código abierto especialmente diseñada para equipos que trabajan con metodologías ágiles. Su funcionalidad y características se adaptan de manera óptima a las necesidades del proyecto, permitiendo una colaboración efectiva y un seguimiento adecuado de las tareas y el avance del equipo. Se ha elegido esta herramienta porque ofrece vistas tanto de tableros Kanban para el seguimiento visual de las tareas, como de gestión de épicas y sprints para una planificación más estructurada. Además, cabe destacar que ya estoy familiarizado con la herramienta, lo que facilitará su implementación y uso efectivo en el proyecto. El logotipo de taiga se puede observar en la figura 2.1.



Figura 2.1: Logotipo de Taiga.io

El equipo de desarrollo está formado por solo dos personas (Marc Pardo Fernández y Jose Francisco Ramos Romero), teniendo Jose Francisco Ramos Romero el rol de dueño del producto y el de líder del equipo de Scrum (*Scrum Master*) mientras Marc Pardo Fernández posee

únicamente el rol de desarrollador.

Para que todo el equipo se mantenga informado se realizarán tres tipos de reuniones: reuniones diarias, reuniones de inicio de sprint y reuniones de cierre de sprint. Tanto las de inicio como las de cierre de sprint contarán de forma obligatoria con el dueño del producto y se redactará o actualizará su correspondiente documentación al finalizar las mismas (minutas, decisiones de diseño o nuevas historias). Por otro lado, las reuniones diarias no requieren la participación del dueño del producto y se contará con un único documento en el que registrar lo comentado en estas (qué trabajo se ha realizado desde la última reunión, qué trabajo se planea realizar y qué problemas se han encontrado hasta el momento).

En la próxima sección definiremos la pila del producto (estimada y priorizada), se estimará la velocidad de los sprints, se definirá de forma más detallada como se trabajará en el proyecto y se realizará la planificación del primer sprint.

2.2. Planificación del proyecto

2.2.1. Requisitos iniciales del proyecto

Los requisitos iniciales del proyecto proporcionados por el cliente son los siguientes:

■ Gestión de usuarios

El sistema de gestión de usuarios contempla las siguientes funcionalidades:

1. Registro: Los usuarios podrán registrarse utilizando su propio correo electrónico o mediante su cuenta de Google.
2. Recuperación de contraseña: Se implementará un mecanismo para permitir a los usuarios restablecer sus contraseñas en caso de olvido.
3. Control de inicio de sesión: Se garantizará el control del inicio de sesión en distintas partes de la web, asegurando la seguridad y la protección de la cuenta del usuario.
4. Página de gestión de la cuenta: Se creará una página dedicada para que los usuarios administren su cuenta, donde podrán realizar cambios en su información personal y configuraciones relacionadas.

■ Módulo de Importación

El módulo de importación permitirá a los usuarios realizar la importación de datos. Además, se proporcionará documentación escrita y un video explicativo para guiar a los usuarios a través del proceso de importación. Dado que no existe una API para leer los datos, el usuario deberá realizar un proceso de copiar y pegar en el panel de control.

■ Pagos

Se requerirá que los usuarios realicen un pago para acceder al panel de control. Se creará una página de precios que describa los diferentes planes disponibles. Las plataformas de pago utilizadas serán Paypal o Stripe. El precio dependerá del número de registros importados. Además, se ofrecerá una opción gratuita con un límite máximo de registros.

- **Dashboard**

Una vez que los usuarios hayan realizado el pago y tengan acceso al panel de control, se les proporcionarán estadísticas relacionadas con sus inversiones deportivas basadas en los datos importados. Estas estadísticas estarán disponibles en el panel de control para que los usuarios puedan consultar y analizar su rendimiento en el ámbito deportivo.

2.2.2. Pila inicial del producto

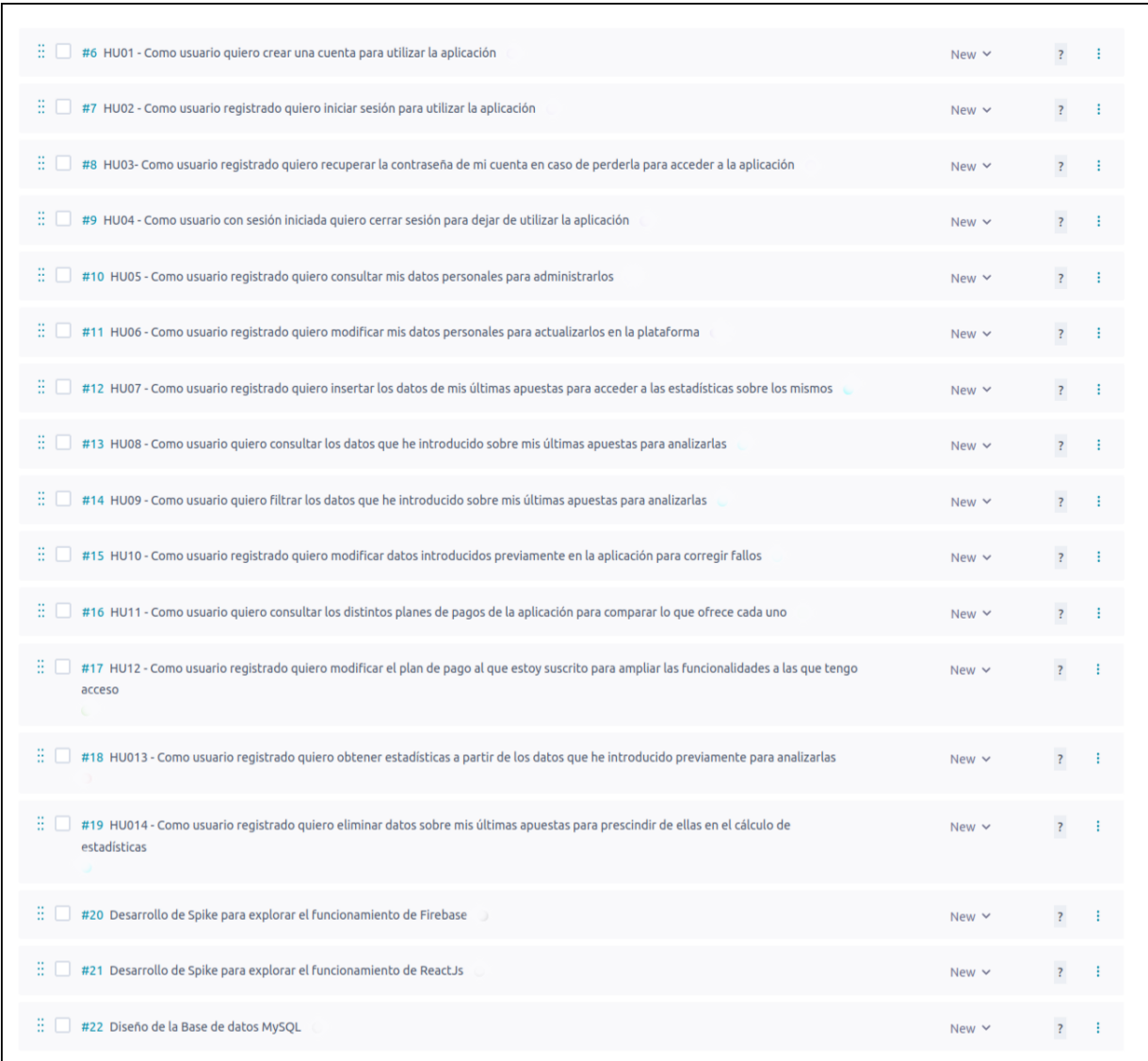
Tras analizar los requisitos del proyecto se ha desarrollado una versión inicial de la pila del producto.

- **HU01** - Como usuario quiero crear una cuenta para utilizar la aplicación.
- **HU02** - Como usuario registrado quiero iniciar sesión para utilizar la aplicación.
- **HU03** - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación.
- **HU04** - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación.
- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos.
- **HU06** - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma.
- **HU07** - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos.
- **HU08** - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas.
- **HU09** - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas.
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos.
- **HU11** - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno.
- **HU12** - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso.
- **HU013** - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas.
- **HU014** - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas.

A esta pila del producto inicial se han añadido las siguientes tres tareas sin historia asociada. Estas son tareas necesarias para que los desarrolladores lleven a cabo las historias.

- Desarrollo de *Spike* para explorar el funcionamiento de Firebase.
- Desarrollo de *Spike* para explorar el funcionamiento de ReactJs.
- Diseño de la base de datos MySQL.

Se puede observar una imagen de la pila del producto con el formato de la herramienta de gestión que se está utilizando en la figura 2.2.



#6 HU01 - Como usuario quiero crear una cuenta para utilizar la aplicación	New	?	⋮
#7 HU02 - Como usuario registrado quiero iniciar sesión para utilizar la aplicación	New	?	⋮
#8 HU03 - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación	New	?	⋮
#9 HU04 - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación	New	?	⋮
#10 HU05 - Como usuario registrado quiero consultar mis datos personales para administrarlos	New	?	⋮
#11 HU06 - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma	New	?	⋮
#12 HU07 - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos	New	?	⋮
#13 HU08 - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas	New	?	⋮
#14 HU09 - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas	New	?	⋮
#15 HU10 - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos	New	?	⋮
#16 HU11 - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno	New	?	⋮
#17 HU12 - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso	New	?	⋮
#18 HU013 - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas	New	?	⋮
#19 HU014 - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas	New	?	⋮
#20 Desarrollo de Spike para explorar el funcionamiento de Firebase	New	?	⋮
#21 Desarrollo de Spike para explorar el funcionamiento de ReactJs	New	?	⋮
#22 Diseño de la Base de datos MySQL	New	?	⋮

Figura 2.2: Pila inicial del producto.

Una vez se ha definido la pila del producto se ha decidido prescindir de las tareas sin historia durante las siguientes fases de la planificación, dejando la pila con los elementos más relevantes

para poder gestionar las tareas de forma sencilla. La pila actualizada se puede observar en la figura 2.3.

<input type="checkbox"/> #6 HU01 - Como usuario quiero crear una cuenta para utilizar la aplicación	New ▾	? ⋮
<input type="checkbox"/> #7 HU02 - Como usuario registrado quiero iniciar sesión para utilizar la aplicación	New ▾	? ⋮
<input type="checkbox"/> #8 HU03- Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación	New ▾	? ⋮
<input type="checkbox"/> #9 HU04 - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación	New ▾	? ⋮
<input type="checkbox"/> #10 HU05 - Como usuario registrado quiero consultar mis datos personales para administrarlos	New ▾	? ⋮
<input type="checkbox"/> #11 HU06 - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma	New ▾	? ⋮
<input type="checkbox"/> #12 HU07 - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos	New ▾	? ⋮
<input type="checkbox"/> #13 HU08 - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas	New ▾	? ⋮
<input type="checkbox"/> #14 HU09 - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas	New ▾	? ⋮
<input type="checkbox"/> #15 HU10 - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos	New ▾	? ⋮
<input type="checkbox"/> #16 HU11 - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno	New ▾	? ⋮
<input type="checkbox"/> #17 HU12 - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso	New ▾	? ⋮
<input type="checkbox"/> #18 HU013 - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas	New ▾	? ⋮
<input type="checkbox"/> #19 HU014 - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas	New ▾	? ⋮

Figura 2.3: Pila del producto simplificada.

2.2.3. Priorización y estimación de la pila del producto

Con la pila del producto simplificada se ha buscado agrupar las historias según su funcionalidad y priorizarlas- Para esto se han creado las épicas que encontramos en la figura 2.4 (se ha añadido la épica *Storyless* para agrupar las tareas sin historia que surjan durante el desarrollo) y se ha asignado una a cada historias como se puede observar en la figura 2.5.

A su vez se priorizaron las historias en función de los siguientes criterios:

- Conocimientos sobre las tecnologías a utilizar.
- Exigencia del cliente.
- Complejidad del cálculo de estadísticas.

- Desarrollo en vista y/o servidor.
- Uso de servicios externos.

Con estos criterios en mente y utilizando la metodología Moscow (*Must-have*, *Should-have*, *Could-have*, y *Won't-have*) se ha priorizado las historias quedando estas como se puede observar en la figura 2.6

Una vez ordenada y priorizada la pila del producto se ha realizado la técnica *Planning Poker* para estimar los puntos de historia de toda la pila del producto. Para esto se ha utilizado la plataforma Planning Poker. La figura 2.7 muestra como han quedado repartidos los 84 puntos de historia.

Con los puntos de historia definidos se ha decidido tener una velocidad de 25 puntos de historia por sprint y por lo tanto finalizar el proyecto en 4 sprints de dos semanas. Se va a trabajar en el proyecto durante 23 horas semanales, dotando al proyecto de una duración de 184 horas en caso de cumplir la predicción. Teniendo en cuenta que la estancia en las prácticas debe ser de 300 horas y ya se han realizado 46 horas, se deja un margen de 70 horas (tres semanas) para posibles retrasos y ampliaciones de la pila del producto.

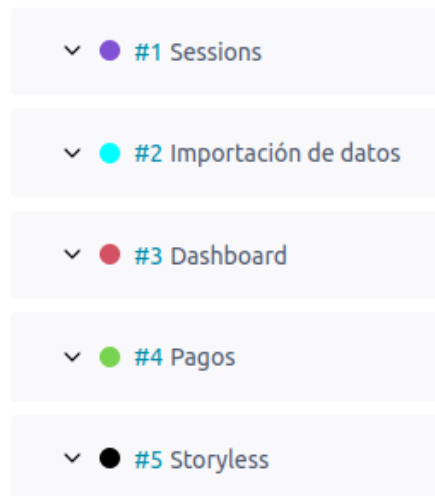


Figura 2.4: Épicas.

☐ #6 HU01 - Como usuario quiero crear una cuenta para utilizar la aplicación ●	New ▾	? ⋮
☐ #7 HU02 - Como usuario registrado quiero iniciar sesión para utilizar la aplicación ●	New ▾	? ⋮
☐ #8 HU03 - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación ●	New ▾	? ⋮
☐ #9 HU04 - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación ●	New ▾	? ⋮
☐ #10 HU05 - Como usuario registrado quiero consultar mis datos personales para administrarlos ●	New ▾	? ⋮
☐ #11 HU06 - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma ●	New ▾	? ⋮
☐ #12 HU07 - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos ●	New ▾	? ⋮
☐ #13 HU08 - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas ●	New ▾	? ⋮
☐ #14 HU09 - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas ●	New ▾	? ⋮
☐ #15 HU10 - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos ●	New ▾	? ⋮
☐ #16 HU11 - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno ●	New ▾	? ⋮
☐ #17 HU12 - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso ●	New ▾	? ⋮
☐ #18 HU013 - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas ●	New ▾	? ⋮
☐ #19 HU014 - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas ●	New ▾	? ⋮

Figura 2.5: Pila del producto ordenada.

USER STORY	STATUS	POINTS
#6 HU01 - Como usuario quiero crear una cuenta para utilizar la aplicación m ●	New ▾	? ?
#7 HU02 - Como usuario registrado quiero iniciar sesión para utilizar la aplicación m ●	New ▾	? ?
#8 HU03 - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación s ●	New ▾	? ?
#9 HU04 - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación m ●	New ▾	? ?
#10 HU05 - Como usuario registrado quiero consultar mis datos personales para administrarlos c ●	New ▾	? ?
#11 HU06 - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma w ●	New ▾	? ?
#12 HU07 - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos m ●	New ▾	? ?
#13 HU08 - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas m ●	New ▾	? ?
#14 HU09 - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas s ●	New ▾	? ?
#15 HU10 - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos w ●	New ▾	? ?
#16 HU11 - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno s ●	New ▾	? ?
#17 HU12 - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso c ●	New ▾	? ?
#18 HU013 - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas m ●	New ▾	? ?
#19 HU014 - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas w ●	New ▾	? ?

Figura 2.6: Pila del producto priorizada.

Your Planning Poker® Game Summary

TFG Backlog

Story	Story Title	Score
1	HU01 - Como usuario quiero crear una cuenta para utilizar la aplicación.	5
2	HU02 - Como usuario registrado quiero iniciar sesión para utilizar la aplicación.	5
3	HU03 - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación.	3
4	HU04 - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación.	1
5	HU05 - Como usuario registrado quiero consultar mis datos personales para administrarlos.	2
6	HU06 - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma.	3
7	HU07 - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos.	13
8	HU08 - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas.	5
9	HU09 - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas.	8
10	HU10 - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos .	5
11	HU11 - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno.	1
12	HU12 - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso.	8
13	HU013 - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas.	20
14	HU014 - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas.	5

Figura 2.7: Resultados de la técnica Planning Poker.

2.2.4. Planificación de la pila del producto del primer sprint

Teniendo en cuenta la priorización y estimación de la pila del producto se ha configurado la pila del primer sprint. Se han añadido a esta las historias HU01, HU02, HU04, HU07 y una tarea sin historia con el nombre *Diseño inicial del proyecto*. Esta cuenta con 5 puntos de historia y su objetivo es diseñar la arquitectura del sistema y la base de datos. Se puede ver como queda la pila del sprint en la figura 2.8.

Para estas historias de usuario se han definido las siguientes tareas:

- **HU01** - Como usuario quiero crear una cuenta para utilizar la aplicación.
 1. Conectar Firebase con el front-end. (30 minutos)
 2. Desarrollar la vista. (2 horas)
 3. Actualizar el estado de la sesión una vez se inicie sesión. (30 minutos)

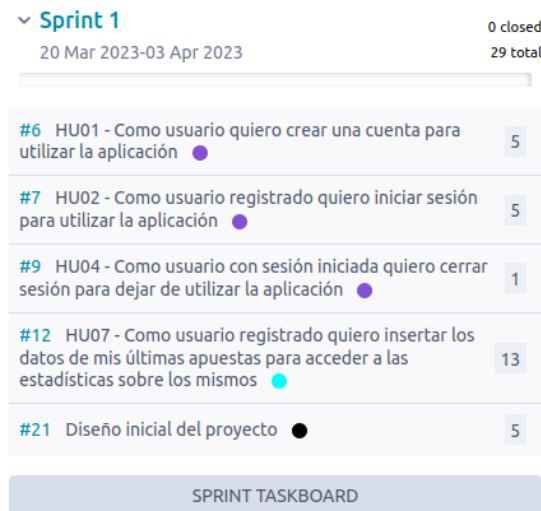


Figura 2.8: Página de Taiga con la pila de Sprint 1.

- **HU02** - Como usuario registrado quiero iniciar sesión para utilizar la aplicación.
 1. Desarrollar la vista. (1 hora)
 2. Actualizar el estado de la sesión una vez se inicie sesión. (30 minutos)
- **HU04** - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación.
 1. Desarrollar la vista. (30 minutos)
 2. Actualizar el estado de la sesión una vez se inicie sesión. (30 minutos)
- **HU07** - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos.
 1. Desarrollar la vista. (2 horas)
 2. Verificación del formato de los datos introducidos. (2 horas)
 3. Desarrollo de la inserción de datos en la base de datos. (1 hora)
 4. Desarrollo de pruebas unitarias para el formateo de datos. (30 minutos)

El primer sprint supera en 4 puntos de historia la velocidad establecida, pero se ha configurado de esta forma porque durante las semanas previas al comienzo del proyecto se desarrollaron diversas pruebas con las tecnologías a utilizar durante el proyecto, y en el desarrollo de estas se implementó todo lo relacionado con las historias HU01, HU02 y HU04. Por lo tanto, el trabajo a realizar en estas historias sería adaptar lo desarrollado antes de iniciar el proyecto.

Este primer sprint iniciará el 20/03/2023 y finalizará el 03/04/2023. El tema del mismo será *Acceso a la aplicación e introducción de datos*.

Para registrar las tareas realizadas antes del inicio del proyecto se ha creado en Taiga un *sprint 0* con tareas para cada prueba de tecnologías, redacción de documentación y el estudio de plantillas para creación de paneles de control.

2.2.5. Mantenimiento del proyecto

Una vez se ha planificado el primer sprint y ordenado, estimado y priorizado la pila inicial del producto se tomarán las siguientes medidas para mantener el código y la documentación del proyecto ordenados, organizados y estandarizados:

- Uso del formato '*Como [tipo de usuario], quiero [realizar esta acción] para [conseguir un objetivo].*' para la creación de cada nueva historia de usuario.
- Uso del estándar *Camel* para nombrar las clases presentes en el código del proyecto.
- Uso del estándar '*// Comentario*' con un máximo de una línea para comentarios puntuales.
- Uso de bloques descriptivos que indiquen para que sirven, que reciben y que devuelven métodos y funciones al principio de los mismos.
- Uso de bloques descriptivos que indiquen descripción, autor y dependencias de cada clase al principio de las mismas.
- Diseño de todas las parte implicadas para llevar a cabo una historia de usuario antes de comenzar el desarrollo de la misma.
- Documentación de todas las decisiones de diseño en nuevas páginas de la sección de documentación de Taiga.
- Desarrollo de escenarios variados para cada historia de usuario en la documentación de Taiga.
- Asignación de una o más épicas a cada historia de usuario en Taiga.
- Asignación de prioridad a cada nueva historia de usuario en Taiga.
- Desarrollo de pruebas unitarias para cada función compleja presente en el proyecto.
- Desarrollo de pruebas de integración y aceptación para cada historia de usuario a partir de los escenarios creados previamente.
- Uso del siguiente estándar '*git branch categoría/referencia/decripción-en-estándar-kebab*' para nombrar las ramas del repositorio Git del proyecto.
- Uso del siguiente estándar '*git commit -m 'categoría: hacer algo; hacer otra cosa''*' para nombrar de los *commits* del repositorio Git del proyecto.

Para mantener en el tiempo lo conseguido con estas medidas se realizarán periódicamente las siguientes tareas:

- Actualización del estado de las historias de usuario y sus respectivas tareas en los tableros disponibles en Taiga.
- Revisión de la pila del producto en cada reunión de inicio de sprint.

- Revisión y corrección de la pila del producto en cada reunión de cierre de sprint.
- Revisión y corrección de la documentación generada en cada reunión de cierre de sprint.
- Revisión y corrección de las tareas en cada reunión diaria.
- Revisión y corrección del nombre de las clases creadas al finalizar cada tarea.
- Revisión y corrección del nombre de las ramas creadas al finalizar cada historia de usuario.
- Búsqueda y corrección de redundancias en la documentación al acabar cada sprint.
- Revisión y corrección de comentarios presentes en el código al finalizar cada tarea.

2.3. Costes

En lo relativo a los costes, este proyecto tiene como principal gasto el sueldo de los dos desarrolladores. Para calcular los mismos se ha tenido en cuenta que el desarrollador junior ha trabajado 300 horas, mientras que el líder del equipo de Scrum ha trabajado menos en el mismo (al estar involucrado en más proyectos) con salarios aproximados obtenidos en talent.com [20]. Se estima que el líder del equipo de Scrum ha trabajado en el proyecto un 30 % de lo trabajado por el desarrollador.

Teniendo en cuenta los sueldos mostrados en la tabla 2.1 (esta se ha elaborado con los sueldos medios obtenidos de talent.com), un 20 % de coste adicional por contratación en cada uno contamos con el coste en recursos humanos que muestra la ecuación 2.1.

Puesto	Sueldo mensual	Sueldo por hora
Ingeniero de software junior	1.958€	12,05€
Líder del equipo de Scrum	3.625€	22,31€

Cuadro 2.1: Salario de los implicados en el desarrollo.

$$1,2 \times 12,05 \times 300 + 22,31 \times 300 \times 0,3 \times 1,2 = 6747,48 \quad (2.1)$$

No debemos olvidar los costes indirectos, que son aquellos costes que la empresa debe asumir pero que no se relacionan directamente con el proyecto. Dado que no conocemos el coste de mantenimiento de las oficinas y todos los aspectos relacionados, se añadirá al coste de recursos humanos del proyecto un porcentaje 20 % adicional. Este cálculo lo podemos ver en la ecuación 2.2.

$$6747,48 \times 1,2 = 8096,98 \quad (2.2)$$

Por otra parte contamos con los costes asociados al software que se va a utilizar. Contamos con una gran cantidad de herramientas gratuitas, pero tanto el servidor web, el dominio y la herramienta Metronic tiene un costo asociado. Se contará el coste del primer año de vida del

producto. En la tabla 2.2 se muestran todas las herramientas y sus costes, siendo el coste total el reflejado en la ecuación 2.3.

Herramienta	Coste periódico	Coste fijo
ReactJS	0€	0€
Metronic	0€	49€
PHP	0€	0€
PHP-CRUD-API	0€	0€
Google Firebase Auth Plan Spark	0€	0€
Hostinger Hosting Empresarial	3,99€/mes	0€
Hostinger Dominio .com	9,99€/año	0€
GitHub	0€	0€
Visual Studio Code	0€	0€
Taiga.io Basic Plan	0€	0€

Cuadro 2.2: Coste de las herramientas empleadas en el desarrollo.

$$49 + 9,99 + 3,99 \times 12 = 106,87 \quad (2.3)$$

El proyecto se desarrolló utilizando el ordenador personal del desarrollador, pero se proporcionó tanto un teclado como un monitor auxiliar para facilitar el desarrollo. Estos costes se pueden consultar en la tabla 2.3. Teniendo en cuenta que estos equipos se buscan amortizar en 4 años y solo se les ha dado uso durante 3 meses, el coste asociado al equipo utilizado durante el desarrollo sería el que muestra la ecuación 2.4

Producto	Coste total	Coste proporcional
Monitor 1080p 24 pulgadas	110€	6,88€
Teclado	15€	0,94€

Cuadro 2.3: Coste del equipo empleado en el desarrollo.

$$6,88 + 0,94 = 7,82 \quad (2.4)$$

Teniendo en cuenta todo lo mencionado el coste final del proyecto sería 8.211,67 como muestra la ecuación 2.5.

$$8096,98 + 106,87 + 7,82 = 8211,67 \quad (2.5)$$

2.4. Riesgos

En esta sección se van a identificar los riesgos del proyecto. Para cada riesgo observado se valorarán sus efectos y contexto de aparición, permitiendo definir estrategias para reducir su probabilidad de aparición o controlarlo si aparece. Este análisis cubre toda la extensión del proyecto desde su fase inicial.

2.4.1. Identificación de Riesgos

A continuación se van a mostrar los riesgos identificados en la tabla 2.4.

ID	Descripción del riesgo	Tipo del riesgo
R01	Falta de experiencia en tareas de planificación	Riesgo del proyecto
R02	Diseño erróneo	Riesgo del producto
R03	Falta de experiencia con las herramientas utilizadas	Riesgo del producto/proyecto
R04	Entendimiento errado de los requerimientos	Riesgo del proyecto/producto
R05	Estimación de costo temporal fuera del alcance de la realidad	Riesgo del proyecto
R06	Falta de comunicación entre los miembros del equipo	Riesgo del proyecto

Cuadro 2.4: Identificación de riesgos del proyecto.

2.4.2. Análisis de Riesgos

El análisis de los riesgos identificados en la sección anterior se muestra en las tablas 2.5 y 2.6.

ID	Magnitud	Descripción	Impacto	Probabilidad de aparición
R01	Media	El equipo tiene poca experiencia en el desarrollo de software siguiendo una metodología ágil.	La planificación guía todo el desarrollo del proyecto. Un error en la misma puede incidir directamente en sus resultados. No obstante, la división en iteraciones reduce el posible impacto de los errores, permitiendo que estos puedan ser corregidos o absorbidos en iteraciones posteriores a la de su aparición.	Alta
R02	Media	El diseño del sistema resulta inadecuado. Al realizar actividades de implementación puede encontrarse que el diseño carece del suficiente nivel de detalle o está mal enfocado, bien por la naturaleza del problema, o bien por restricciones de uso impuestas por tecnologías de terceros.	Puede introducir retrasos en el proyecto ante la necesidad de volver a considerar el diseño trazado. Requiere la actualización o modificación de los artefactos de diseño.	Baja
R03	Alta	El equipo tiene dificultades a la hora de realizar sus objetivos (tanto de documentación como de implementación) por su inexperiencia con las herramientas disponibles para el mismo.	Puede suponer retrasos.	Alta

Cuadro 2.5: Análisis de riesgos 1

ID	Magnitud	Descripción	Impacto	Probabilidad de aparición
R04	Alta	No se entienden correctamente todos los requisitos del proyecto que se va a desarrollar.	En función de la cantidad e importancia de los requisitos que no se han entendido se podría necesitar desde unas ligeras modificaciones a un rediseño completo del software.	Baja
R05	Media	Se sobrestima o subestima el costo temporal del desarrollo del producto de software.	Puede generar que el equipo entre en períodos de sobrecarga de trabajo o períodos de ausencia del mismo, lo que a su vez puede conllevar a un deterioro en la calidad.	Alta
R06	Baja	Falta de intercambio efectivo de información y colaboración entre los miembros de un equipo.	Puede suponer retrasos, conflicto y duplicación de esfuerzos (desarrollando varias personas la misma parte del proyecto).	Baja

Cuadro 2.6: Análisis de riesgos 2

2.4.3. Planificación de respuesta

Por último se ha creado la tabla 2.7 con la planificación de respuesta a los riesgos analizados previamente.

ID	Prevención del riesgo	Respuesta al riesgo
R01	Realización de reuniones entre los miembros del proyecto para la evaluación de la marcha del proyecto y consultas al tutor.	Se analizará la planificación comparando con los informes de seguimiento de su ejecución para tratar de corregir errores en la planificación de las siguientes fases del proyecto.
R02	Verificación y validación del diseño cada vez que una parte del mismo esté finalizada.	Análisis de los errores del software y rediseño del mismo para corregirlos. En caso de tener demasiados habría que hacer un diseño nuevo.
R03	Se formará al equipo en el uso de estas herramientas para que puedan desempeñar sus tareas correctamente.	Se formará sobre la marcha al equipo y en caso de necesidad se haría una redistribución de tareas.
R04	Verificación y validación de los requisitos con ayuda del cliente una vez se han definido.	Modificación de los requisitos erróneos y rediseño de las partes implicadas. En caso de ser necesario redimensionar el proyecto.
R05	Realización de varias estimaciones con metodologías diferentes.	Redimensionar el proyecto conforme se ejecuta.
R06	Establecer roles, reuniones periódicas y canales de comunicación claros.	Identificar el origen del problema y mejorar los canales de comunicación.

Cuadro 2.7: Planificación de respuesta a los riesgos del proyecto.

2.5. Seguimiento del proyecto

2.5.1. Sprint 1

En el primer sprint del proyecto, se trabajó en la implementación de las funcionalidades relacionadas con el inicio de sesión y la introducción de datos. El sprint se inició con un total de 29 puntos de historias. Durante el mismo se logró completar exitosamente 11 puntos de historias. Durante el sprint se priorizó la finalización de las siguientes historias:

- **HU01** - Como usuario quiero crear una cuenta para utilizar la aplicación.
- **HU02** - Como usuario registrado quiero iniciar sesión para utilizar la aplicación.
- **HU04** - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación.

Estas historias contaron exclusivamente con desarrollo Front-end gracias a el uso de Firebase para la gestión de usuarios. Además, se realizó el diseño de la base de datos y se desarrolló el módulo de importación de datos en el front-end. Para garantizar la correcta transformación de los datos introducidos en el objeto necesario para mostrarlos en la tabla, se crearon distintos pruebas unitarias.

Quedó pendiente las historia **HU07** (Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos) y la tarea **Diseño inicial del proyecto** para ser abordadas en futuros sprints.

Se puede ver en la figura 2.9 un diagrama de tipo *Burn Down* en el que se muestra la evolución de los puntos de historia durante este sprint.

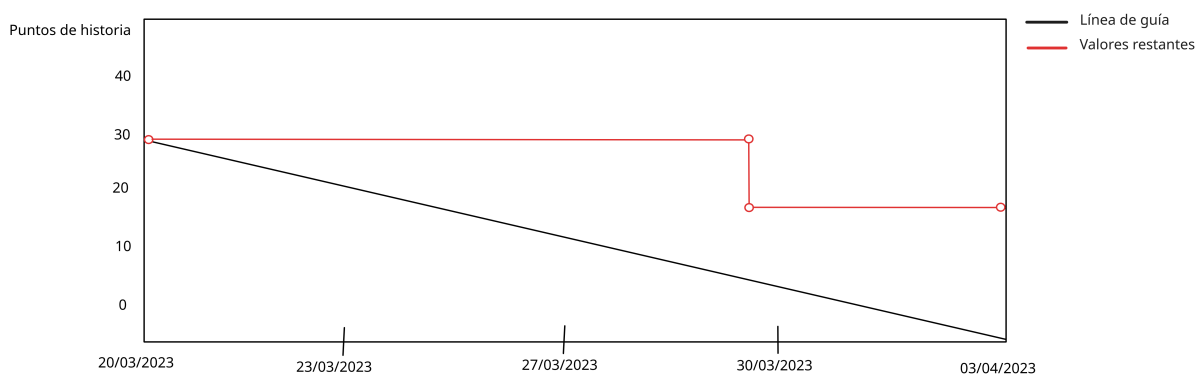


Figura 2.9: Diagrama de tipo *Burn Down* del primer sprint.

2.5.2. Sprint 2

Durante el segundo sprint del proyecto, nos centramos en la configuración del sprint para lograr el objetivo de obtener un producto mínimo viable que permitiera añadir movimientos a

la plataforma y consultar las estadísticas correspondientes utilizando una cuenta de usuario.

En este sprint continuamos con el desarrollo de las historias de usuario HU07 y Diseño inicial del proyecto. Además, se añadieron las historias HU03 y HU08, quedando la pila del sprint de la siguiente manera:

- **HU07** - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos (13 puntos).
- Diseño inicial del proyecto (5 puntos).
- **HU03** - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación (3 puntos).
- **HU08** - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas (5 puntos).

El sprint se inició con un total de 26 puntos de historia.

Durante la planificación del mismo también se identificaron posibles riesgos y se planificaron medidas para mitigarlos. Se pueden consultar en la tabla 2.8.

Riesgo	Mitigación
Fallos en el diseño de la base de datos.	Se realizará un re-diseño de la base de datos en caso de surgir fallos.
Problemas con las tecnologías utilizadas debido a falta de conocimiento.	Se buscarán tutoriales y se consultará al equipo para solucionar cualquier problema relacionado con las tecnologías utilizadas.
Errores de formato en el módulo de importación.	Se corregirán los errores y se actualizarán las pruebas diseñadas para adaptarse al nuevo formato.

Cuadro 2.8: Riesgos identificados durante la planificación del segundo sprint.

Al finalizar el segundo sprint se lograron completar todas las historias de usuario planificadas sin ningún problema, finalizando los 26 puntos de historias previstos.

Durante el sprint, se realizó la corrección e implementación de la base de datos. Además, se desarrolló un servicio CRUD, utilizando la herramienta PHP-CRUD-API, para leer y escribir en la base de datos. Por otra parte se añadieron las siguientes historias de usuario a la pila del producto:

- **HU15** - Como usuario quiero que mi suscripción se renueve de forma automática para mantener mi plan de pago sin interrupciones (10 puntos).
- **HU16** - Como usuario quiero poder crear paneles de control personalizados para consultar la información más relevante para mi situación (20 puntos).

- **HU17** - Como usuario quiero poder exportar mis datos para poder utilizarlos fuera de la aplicación (8 puntos).

Si bien se planteó la posibilidad de añadir una historia adicional a la iteración, no se pudo comenzar su desarrollo durante este sprint al invertir demasiado tiempo en solucionar algunos problemas con los estilos y detalles visuales de las historias ya implementadas.

Durante el sprint aparecieron dos riesgos, fallos en el diseño de la base de datos y errores de formato en el módulo de importación. Al tener planificada la respuesta a estos riesgos se solucionaron sin demasiado problema.

El cierre del segundo sprint se llevó a cabo sin contratiempos graves, gracias a la mitigación efectiva de los riesgos identificados. Se puede ver en la figura 2.10 un diagrama de tipo *Burn Down* en el que se muestra la evolución de los puntos de historia durante este sprint.

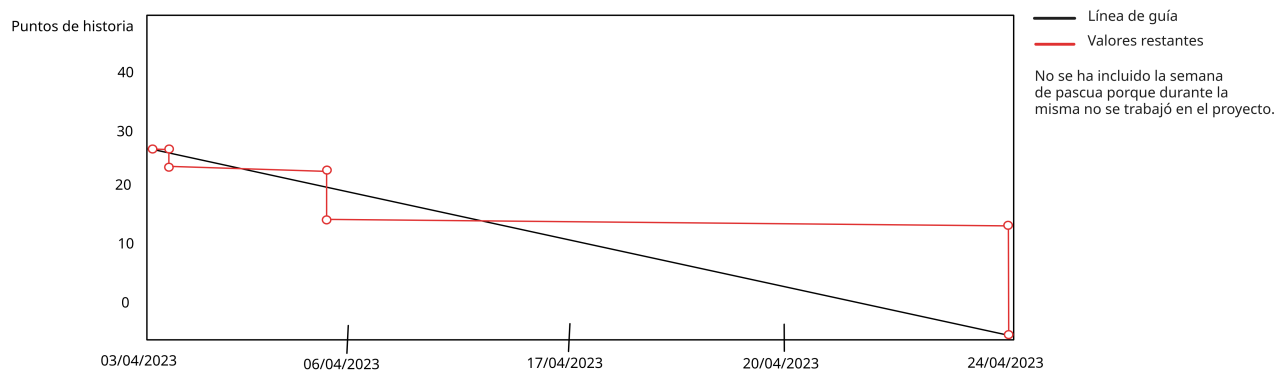


Figura 2.10: Diagrama de tipo *Burn Down* del segundo sprint.

2.5.3. Sprint 3

Durante el tercer sprint el objetivo principal fue obtener estadísticas en base a los datos introducidos en la plataforma.

Antes de configurar el sprint, se dividió la historia de usuario HU13 en varias historias más pequeñas para una mejor organización. En total, las historias incluidas en la pila del sprint fueron:

- **HU13** - Como usuario registrado quiero acceder a la sección panel de control para consultar las estadísticas disponibles (3 puntos).
- **HU18** - Como usuario registrado quiero consultar el número de apuestas realizadas para controlar mi ritmo de trabajo (3 puntos).
- **HU19** - Como usuario registrado quiero consultar mi balance de ganancias y pérdidas para conocer que tipo de apuesta me genera más beneficio (5 puntos).

- **HU20** - Como usuario registrado quiero consultar cuánto he pagado de comisiones para evaluar el impacto de las mismas en mi balance (5 puntos).
- **HU21** - Como usuario quiero conocer la cantidad de dinero depositado y retirado para calcular mis beneficios (5 puntos).
- **HU22** - Como usuario quiero conocer mi balance de beneficios y pérdidas para tener clara mi situación actual (5 puntos).
- **HU23** - Como usuario quiero consultar mi rendimiento en las distintas competiciones en las que apuesto para tomar mejores decisiones en futuras apuestas (5 puntos).
- **HU24** - Como usuario quiero consultar mi rendimiento en los distintos mercados en los que apuesto para tomar mejores decisiones en futuras apuestas (5 puntos).
- **HU25** - Como usuario quiero conocer mi balance entre apuestas finalizadas con *CashOut* y las finalizadas por el final del evento para tomar mejores decisiones en futuras apuestas (5 puntos).
- **HU26** - Como usuario quiero consultar mi ROI para analizar mi situación actual (5 puntos).
- **HU27** - Como usuario quiero conocer el porcentaje máximo de capital en riesgo en una apuesta para analizarlo (5 puntos).

Una vez realizada la ampliación de la pila del producto, se decidió configurar de la siguiente forma la pila del sprint para cumplir con el objetivo del mismo:

- **HU09** - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas (8 puntos).
- **HU13** - Como usuario registrado quiero acceder a la sección panel de control para consultar las estadísticas disponibles (3 puntos).
- **HU18** - Como usuario registrado quiero consultar el número de apuestas realizadas para controlar mi ritmo de trabajo (3 puntos).
- **HU19** - Como usuario registrado quiero consultar mi balance de ganancias y pérdidas para conocer que tipo de apuesta me genera más beneficio (5 puntos).
- **HU20** - Como usuario registrado quiero consultar cuánto he pagado de comisiones para evaluar el impacto de las mismas en mi balance (5 puntos).
- **HU21** - Como usuario quiero conocer la cantidad de dinero depositado y retirado para calcular mis beneficios (5 puntos).

El sprint se inició con un total de 29 puntos de historia.

Antes de iniciar el sprint se añadieron identificaron nuevos riesgos y se planificó su respuesta como se puede observar en la tabla 2.9.

Riesgo	Mitigación
Fallos en el diseño de la base de datos.	Se realizará un re-diseño de la base de datos en caso de surgir fallos.
Problemas con las tecnologías utilizadas debido a falta de conocimiento.	Se buscarán tutoriales, se leerá la documentación y se consultará al equipo para solucionar cualquier problema relacionado con las tecnologías utilizadas.
Errores de formato en el módulo de importación.	Se corregirán los errores y se actualizarán las pruebas diseñadas para adaptarse al nuevo formato.
Encontrar problemas en la plantilla de Metronic por falta de conocimiento.	Se buscarán tutoriales, se leerá la documentación y se preguntarán las dudas a los miembros del equipo por si alguien sabe cómo solucionar el problema.
Fallos en el cálculo de los datos necesarios para el panel de control.	Se realizará una revisión de los datos necesarios y se corregirá el cálculo de los mismos.

Cuadro 2.9: Riesgos identificados durante la planificación del tercer sprint.

Durante el desarrollo, se enfrentaron varios problemas relacionados con los estilos del panel de control, los cuales se abordaron y resolvieron siguiendo lo decidido durante la planificación del sprint. La mayoría de las historias se completaron exitosamente, a excepción de la historia número 21, la cual quedó con una tarea pendiente sin finalizar.

Adicionalmente, se implementó una restricción de acceso al servicio CRUD mediante el uso de tokens, lo que fortaleció la seguridad del sistema. En términos de puntos de historia, se logró completar un total de 24 puntos.

Por otra, parte se encontraron dificultades al intentar crear un controlador personalizado para el back-end. Esto se dejó para iteraciones posteriores. Se puede ver en la figura 2.11 un diagrama de tipo *Burn Down* en el que se muestra la evolución de los puntos de historia durante este sprint.

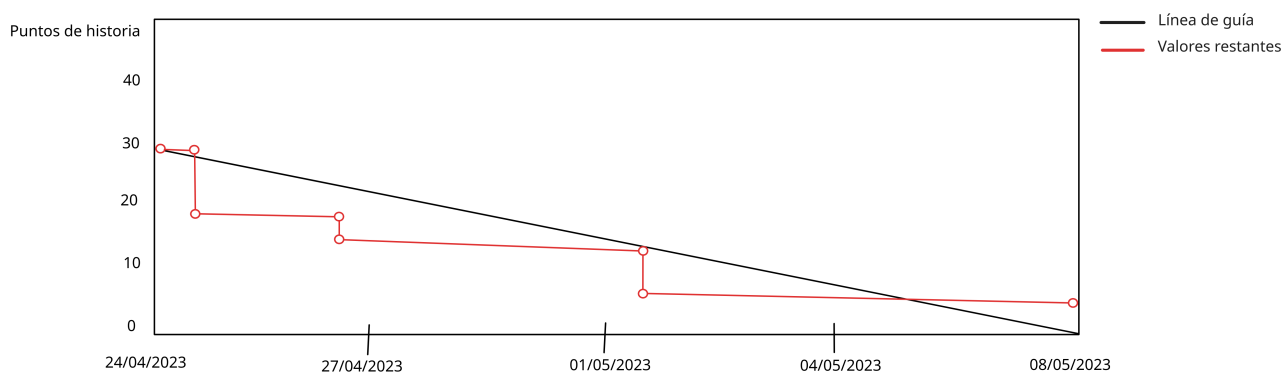


Figura 2.11: Diagrama de tipo *Burn Down* del tercer sprint.

2.5.4. Sprint 4

Como objetivo del cuarto sprint del proyecto buscamos finalizar la épica de usuarios y añadir funcionalidades extra a las tablas.

El objetivo principal de este sprint fue lograr un prototipo 100% funcional que permitiera mostrar datos específicos dependiendo del usuario con sesión iniciada. Además, se planificó añadir la modificación y borrado de datos de la tabla *My Data* y, en caso de tener tiempo, comenzar el desarrollo del sistema de cobros de la aplicación.

En este sprint, se mantuvo la historia de usuario HU21, la cual consiste en *Como usuario quiero conocer la cantidad de dinero depositado y retirado para calcular mis beneficios* con un total de 5 puntos.

Además, se añadieron las siguientes historias de usuario para completar la pila del sprint:

- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos (8 puntos).
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos (5 puntos).
- **HU11** - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno (1 punto).
- **HU14** - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas (5 puntos).

Este sprint inició con un total de 24 puntos de historia.

Para este sprint solo se identificó un riesgo nuevo, *Aparición de errores al refactorizar el código*. Para mitigar este problema, se planteó la creación de pruebas unitarias para las funciones más complejas y la revisión exhaustiva del código antes de realizar modificaciones, con el objetivo de asegurar que el desarrollador comprendiera adecuadamente el comportamiento del mismo.

A pesar de la planificación inicial del sprint, poco después de empezar el mismo se decidió replantear todo el diseño de la aplicación. Evitando así los problemas de implementación en el lado del servidor y satisfaciendo las necesidades del cliente en el diseño de la interfaz.

En relación al servidor, consideramos la posibilidad de migrar todo el back-end a Firebase Cloud Functions como un entorno *serverless*. Sin embargo, debido a los altos costos asociados, se decidió desarrollar un servicio de autenticación externo al servicio CRUD existente, manteniendo nuestra infraestructura actual.

En cuanto a la vista de la aplicación, se llevó a cabo un rediseño de los componentes existentes, enfocándonos en mejorar la presentación de los datos sin modificar la lógica subyacente. De esta manera, logramos optimizar la experiencia del usuario al mostrar la información de forma más clara y atractiva.

Los cambios se vieron reflejados en las historias de usuario, añadiendo distintas historias y modificando las existentes para satisfacer las nuevas necesidades del producto. Las historias HU18, HU19 y HU20 se unificaron en un único componente. Además, se decidió mostrar algunos datos de HU18 junto con HU24. Se mantuvieron las historias ya completadas y las que estaban en progreso, pero se modificaron su definición y sus escenarios.

Por otra parte, HU26 cambió su nombre a *Como usuario quiero consultar mis estadísticas sobre mis beneficios para analizar mi situación actual*. Esta nueva definición amplía el alcance de la historia al incluir más datos y se adapta al nuevo diseño, sin perder su condición de historia de usuario de tamaño reducido.

La pila del producto quedó de la siguiente forma tras añadir las nuevas historias y modificar las existentes:

- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos.
- **HU06** - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma.
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos.
- **HU11** - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno.
- **HU12** - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso.
- **HU14** - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas.
- **HU16** - Como usuario quiero poder crear paneles de control personalizados para consultar la información más relevante para mi situación.
- **HU17** - Como usuario quiero poder exportar mis datos para poder utilizarlos fuera de la aplicación.
- **HU21** - Como usuario quiero conocer la cantidad de dinero depositado y retirado para calcular mis beneficios.
- **HU22** - Como usuario quiero conocer mi balance de beneficios y pérdidas para tener clara mi situación actual.
- **HU23** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada mercado en el que opero para conocer mi trayectoria.
- **HU24** - Como usuario quiero consultar mi rendimiento en los distintos mercados en los que apuesto para tomar mejores decisiones en futuras apuestas.
- **HU25** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada competición en la que opero para conocer mi trayectoria.

- **HU26** - Como usuario quiero consultar mis estadísticas sobre mis beneficios para analizar mi situación actual.
- **HU27** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada tipo de operación en la que opero para conocer mi trayectoria.
- **HU28** - Como usuario quiero poder filtrar el contenido del panel de control por periodos para consultar mi evolución.

Durante este sprint se ha enfrentado el riesgo R02 *Diseño erróneo*, pero se mitigó sin demasiados problemas siguiendo lo indicado en la gestión de riesgos (tabla 2.7).

A esta iteración se añadieron las historias HU24 y HU25, pero finalizó sin completar ninguna historia de usuario con 34 puntos de historia pendientes, por lo que todas pasaron al siguiente sprint o a la pila del producto. Se puede ver en la figura 2.12 un diagrama de tipo *Burn Down* en el que se muestra la evolución de los puntos de historia durante este sprint.

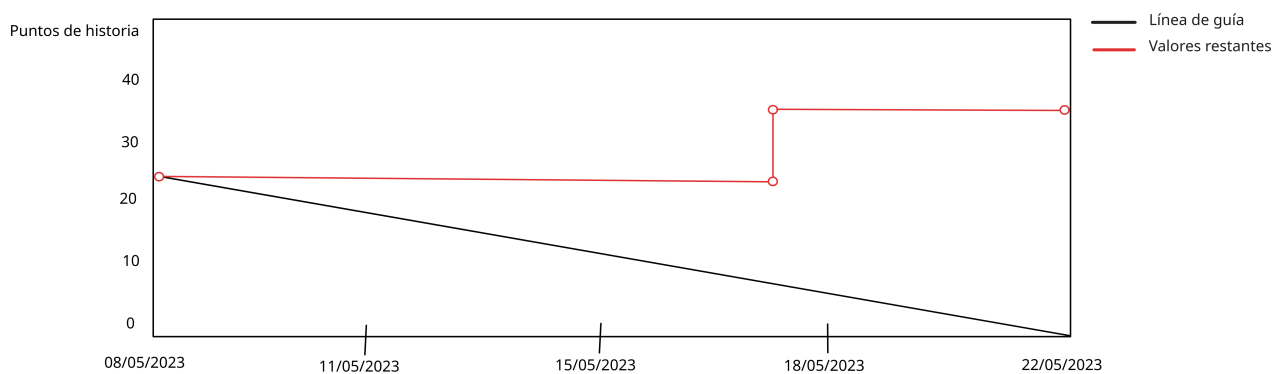


Figura 2.12: Diagrama de tipo *Burn Down* del cuarto sprint.

2.5.5. Sprint 5

Durante el inicio del sprint, se buscó solventar los fallos existentes y adaptar el trabajo desarrollado hasta el momento al nuevo diseño propuesto durante el sprint anterior. El objetivo principal fue finalizar todas las historias relacionadas con el panel de control que quedaron pendientes en el sprint anterior.

Además, se planificó la implementación de una opción de filtrado por fecha en todo el panel de control, con el fin de permitir a los usuarios visualizar la información según sus necesidades temporales. También se resolvió un problema que venía presentándose desde la segunda iteración, asegurando que solo los usuarios autorizados tuvieran acceso a sus propios datos y no a los datos de otros usuarios.

Se mantuvieron de la pila del sprint de la iteración anterior las siguientes historias de usuario:

- **HU21** - Como usuario quiero conocer la cantidad de dinero depositado y retirado para calcular mis beneficios (5 puntos).

- **HU24** - Como usuario quiero consultar mi rendimiento en los distintos mercados en los que apuesto para tomar mejores decisiones en futuras apuestas (5 puntos).
- **HU26** - Como usuario quiero consultar mis estadísticas sobre mis beneficios para analizar mi situación actual (5 puntos).

A estas se añadió la siguiente:

- **HU28** - Como usuario quiero poder filtrar el contenido del panel de control por periodos para consultar mi evolución (15 puntos).

Por último, se devolvieron a la pila del producto las siguientes historias, buscando realizar únicamente historias de prioridad alta durante esta iteración:

- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos (8 puntos).
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos (5 puntos).
- **HU11** - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno (1 punto).
- **HU14** - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas (5 puntos).

Este sprint inició con 30 puntos de historia.

Durante el mismo se añadió la historia HU22 cuando quedó pendiente solo una historia por completar, pero no se llegó a completar y quedó pendiente junto a la HU24. A pesar de esto el resto de historias se completaron de forma satisfactoria, completando 25 puntos de historia de los 30 iniciales. Se puede ver en la figura 2.13 un diagrama de tipo *Burn Down* en el que se muestra la evolución de los puntos de historia durante este sprint.

2.5.6. Sprint 6

El último sprint buscaba finalizar las tareas pendientes del sprint anterior para acabar con una versión básica pero funcional del panel de control. Con esto hecho se planificó añadir la sección de pagos y la posibilidad de pagar por uno que no sea el predeterminado (gratuito).

Después de examinar el funcionamiento de PayPal, se ha observado que al activar un plan de pago como suscripción en PayPal este se encarga automáticamente de la renovación. Esto hace que con la historia de usuario HU12 se realiza de forma simultanea la HU15, por lo tanto se ha decidido eliminar la historia de usuario HU15.

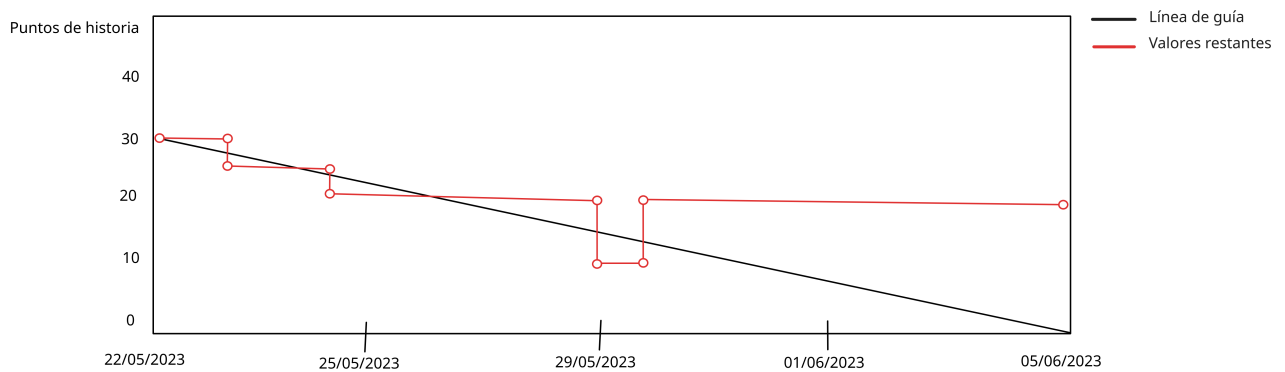


Figura 2.13: Diagrama de tipo *Burn Down* del quinto sprint.

Se mantuvieron en la pila del producto las historias que no se finalizaron durante la iteración previa (HU22 y HU24) y se añadieron HU11, HU12 y HU14, quedando la pila del sprint como se muestra a continuación:

- **HU11** - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno (1 punto).
- **HU12** - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso (10 puntos).
- **HU14** - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas (5 puntos).
- **HU22** - Como usuario quiero conocer mi balance de beneficios y pérdidas para tener clara mi situación actual (5 puntos).
- **HU24** - Como usuario quiero consultar mi rendimiento en los distintos mercados en los que apuesto para tomar mejores decisiones en futuras apuestas (10 puntos).

Este sprint inició con 31 puntos de historia.

Durante la primera semana del sprint se finalizaron todas las historias, por lo que se agregaron las siguientes historias:

- **HU23** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada mercado en el que opero para conocer mi trayectoria (13 puntos).
- **HU25** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada competición en la que opero para conocer mi trayectoria (13 puntos).
- **HU27** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada tipo de operación en la que opero para conocer mi trayectoria (13 puntos).

El sprint acabó el día 14 de junio con todas las historias de la pila del sprint implementadas y con una primer versión producto desplegada.

En la pila del producto quedaron las siguientes historias sin completar (siendo estas menos prioritarias para el cliente):

- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos.
- **HU06** - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma.
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos.
- **HU16** - Como usuario quiero poder crear paneles de control personalizados para consultar la información más relevante para mi situación.
- **HU17** - Como usuario quiero poder exportar mis datos para poder utilizarlos fuera de la aplicación.

Se puede ver en la figura 2.14 un diagrama de tipo *Burn Down* en el que se muestra la evolución de los puntos de historia durante este sprint.

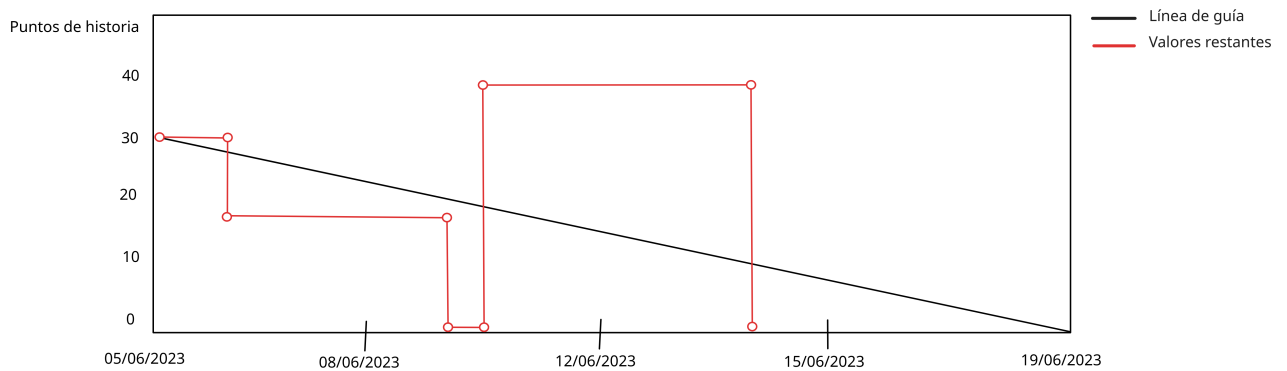


Figura 2.14: Diagrama de tipo *Burn Down* del sexto sprint.

Capítulo 3

Análisis del sistema/producto

Al emplear una metodología ágil, específicamente Scrum, se han establecido la mayoría de los requisitos funcionales durante la etapa de planificación mediante el uso de historias de usuario. En este capítulo, se llevará a cabo un refinamiento de los mismos (trabajando sobre la pila final del producto), proporcionando una descripción más detallada y estableciendo los criterios necesarios para considerar que las historias de usuario se han completado (conocido como *Definition Of Done* en inglés).

3.1. Análisis de requisitos funcionales

- **HU01** - Como usuario quiero crear una cuenta para utilizar la aplicación.
 - Para poder utilizar la aplicación se necesita crear una cuenta de usuario o utilizar una cuenta de Google existente.
 - Al crear una cuenta de usuario, esta cuenta cuenta con el plan de pago gratuito. Este se podrá actualizar a otro superior en cualquier momento.
 - Para crear una cuenta de usuario solo se necesitará un correo electrónico y una contraseña. En caso de querer añadir campos adicionales al perfil, estos se deberán añadir tras el registro. Esto es así para dar el máximo anonimato posible a los usuarios.
 - Tras crear la cuenta de usuario se inicia sesión de forma automática.
- **HU02** - Como usuario registrado quiero iniciar sesión para utilizar la aplicación.
 - Para poder utilizar la aplicación se necesita iniciar sesión, ya sea utilizando una cuenta de Google o una registrada en la aplicación.
 - Cada usuario registrado tiene acceso únicamente a las funcionalidades que ofrece el plan al que se encuentra suscrito.
 - Cada usuario registrado tiene acceso únicamente a sus datos.
- **HU03** - Como usuario registrado quiero recuperar la contraseña de mi cuenta en caso de perderla para acceder a la aplicación.

- Solo se permite recuperar la contraseña a usuarios registrados mediante el método de autenticación correo electrónico y contraseña.
 - Cuando se solicita la recuperación, se envía un correo electrónico al usuario para iniciar el proceso de recuperación.
- **HU04** - Como usuario con sesión iniciada quiero cerrar sesión para dejar de utilizar la aplicación.
 - Para cambiar de cuenta de usuario o salir de la aplicación, se debe permitir a los usuarios con sesión iniciada cerrar sesión.
 - La opción de cerrar sesión solo debe ser visible para los usuarios con sesión iniciada.
- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos.
 - Debo tener acceso a una página en la que poder consultar mis datos personales.
 - Mis datos personales se dividen en dos secciones, obligatorios (correo electrónico y contraseña) y opcionales(avatar y nombre de usuario entre otros).
- **HU06** - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma.
 - Desde la página en la que consulto mis datos personales debo tener la opción de modificar mis datos en todo momento.
 - El correo electrónico no debe poderse modificar.
 - Los datos opcionales pueden eliminarse.
- **HU07** - Como usuario registrado quiero insertar los datos de mis últimas apuestas para acceder a las estadísticas sobre los mismos.
 - Para obtener datos sobre las apuesta deportivas realizadas se necesita ofrecer una herramienta para ingresar los datos de las mismas.
 - Esta herramienta debe ser una tabla que permita introducir datos en la que los campos coincidan con el formato ofrecido en la consulta de movimientos por el mercado de intercambio Orbit.
 - La herramienta debe permitir el ingreso de datos con solo copiar los datos y presionar un botón.
 - El formato de la tabla será el siguiente: [Settled (Fecha finalización) / Ref ID (ID (comission, apuesta, deposito/withdrawal)) / Placed (Fecha de realización) / Description(Competition / Home / Away / Market / Result / Type) / Avg. Odds (Cuota de entrada) / Stake (Importe de apuesta) / Status (Estado (perdida, ganada, o -)) / Debit (Lo perdido si procede) / Credit (Lo ganada si procede) / Balance (Saldo de la cuenta)]
- **HU08** - Como usuario quiero consultar los datos que he introducido sobre mis últimas apuestas para analizarlas.
 - Para la consulta de datos, se dispone de una tabla que refleja todo lo previamente introducido por el usuario.

- La tabla permite la ordenación de las filas por campos, tanto de forma ascendente como descendente.
 - Se puede consultar diferentes cantidades de datos de forma simultánea y seleccionar diversas filas.
 - La tabla se actualiza automáticamente después de cada introducción, modificación o eliminación de los datos que refleja.
- **HU09** - Como usuario quiero filtrar los datos que he introducido sobre mis últimas apuestas para analizarlas.
 - Se permite filtrar los datos de la tabla de datos ya importados.
 - No se permite filtrar los datos de la tabla de importación.
 - Se permite filtrar a partir de cualquier dato presente en la tabla.
 - Se puede filtrar utilizando texto o por tipo de movimiento utilizando un selector.
 - Se puede limpiar el filtro con un botón.
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos.
 - Solo se podrán modificar movimientos que pertenezcan al usuario que los quiere modificar, mediante verificación del token.
 - Se utilizará un modal para modificar los campos de un movimiento.
 - Se mostrará un mensaje de confirmación en la vista al realizar modificaciones para evitar modificaciones involuntarias.
 - No se permite modificar varios movimientos de forma simultánea.
- **HU11** - Como usuario quiero consultar los distintos planes de pagos de la aplicación para comparar lo que ofrece cada uno.
 - En una sección del panel de control se deben mostrar todos los planes de pago disponibles junto con sus características.
 - Esta página enlazará con Paypal para realizar el pago al plan seleccionado.
 - Se debe indicar qué plan de pago tiene seleccionado actualmente.
- **HU12** - Como usuario registrado quiero modificar el plan de pago al que estoy suscrito para ampliar las funcionalidades a las que tengo acceso.
 - El usuario tiene la opción de cancelar la transacción en cualquier momento.
 - El resultado de esta operación debe actualizar el plan de pago del cliente.
 - Es necesario estar registrado para realizar un pago.
 - Una vez que el usuario se suscribe a un plan de pago, este se renueva automáticamente a través de PayPal.
- **HU013** - Como usuario registrado quiero obtener estadísticas a partir de los datos que he introducido previamente para analizarlas.
 - Una vez se inicia sesión, se permite el acceso al panel de control.
 - Se puede acceder al panel de control incluso sin tener registros importados.

- En caso de no tener registros importados, se mostrarán los campos de estadísticas vacíos o con valor 0 en caso de ser campos numéricos.
- **HU014** - Como usuario registrado quiero eliminar datos sobre mis últimas apuestas para prescindir de ellas en el cálculo de estadísticas.
 - Para eliminar una entrada, el usuario debe estar registrado.
 - Solo se podrán eliminar movimientos que pertenezcan al usuario que los quiere borrar.
 - Se debe mostrar un mensaje de confirmación en la vista al realizar la eliminación para evitar borrados involuntarios.
 - Se permite borrar varios movimientos de forma simultánea.
 - Se muestra un mensaje indicando si se han eliminado todos los elementos, en caso de no ser así se indica el número de movimientos eliminados.
- **HU16** - Como usuario quiero poder crear paneles de control personalizados para consultar la información más relevante para mi situación.
 - Debe existir una sección de la web desde la que acceder a estos paneles de control personalizados.
 - Los paneles de control personalizados se crean a partir de componentes presentes en el resto de páginas de la herramienta.
 - Cada plan de pago tiene un número máximo de paneles de control personalizados.
- **HU17** - Como usuario quiero poder exportar mis datos para poder utilizarlos fuera de la aplicación.
 - Desde la tabla de movimientos se pueden exportar todos los datos en formato CSV.
 - Esta opción solo está disponible para algunos planes de pago.
 - A pesar de que no se muestren todos movimientos en la tabla se exportarán todos.
- **HU18** - Como usuario registrado quiero consultar el número de apuestas realizadas para controlar mi ritmo de trabajo.
 - En caso de no tener registros importados, se mostrará un 0.
 - Se mostrará el número total de apuestas, depósitos, retiradas y comisiones.
 - Estos datos se encuentran en la página principal del producto.
- **HU19** - Como usuario registrado quiero consultar mi balance de ganancias y pérdidas para conocer que tipo de apuesta me genera más beneficio.
 - Se mostrará el balance actual de la cuenta.
 - En caso de no tener registros importados, se mostrará un 0.
 - Estos datos se encuentran en la página principal del producto.
- **HU20** - Como usuario registrado quiero consultar cuánto he pagado de comisiones para evaluar el impacto de las mismas en mi balance.
 - Se mostrará la cantidad pagada al mercado de intercambio por comisiones.
 - En caso de no tener registros importados, se mostrará un 0.

- Estos datos se encuentran en la página principal del producto.
- **HU21** - Como usuario quiero conocer la cantidad de dinero depositado y retirado para calcular mis beneficios.
 - Se mostrará la cantidad de dinero ingresado y retirado del mercado de intercambio.
 - En caso de no tener registros importados, se mostrará un 0.
 - Estos datos se encuentran en la página principal del producto.
- **HU22** - Como usuario quiero conocer mi balance de beneficios y pérdidas para tener clara mi situación actual.
 - Se mostrará la evolución de los beneficios y las perdidas del usuario en una gráfica.
 - En caso de no tener registros importados, la gráfica se mostrará vacía.
 - Estos datos se encuentran en la página principal del producto.
- **HU23** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada mercado en el que opero para conocer mi trayectoria.
 - El sistema contará con una página exclusiva para mostrar datos sobre los mercados.
 - Esta página debe mostrar el beneficio obtenido en un mercado durante un periodo.
 - Cuenta con un selector de mercado y otro de periodo para filtrar los datos.
 - Muestra una gráfica con la evolución del beneficio en dicho mercado.
 - Esta tabla contiene a su vez un filtro para modificar la forma en la que muestran los datos. Este nos permitirá consultar la evolución los datos en distintas unidades (horas, días, semanas, meses y años).
- **HU24** - Como usuario quiero consultar mi rendimiento en los distintos mercados en los que apuesto para tomar mejores decisiones en futuras apuestas.
 - Se debe mostrar una lista con los resultados en distintos mercados ordenada por rendimiento y una gráfica con la evolución de sus beneficios en el tiempo.
 - En caso de no tener registros importados, se mostrará tanto la lista como la gráfica vacías.
- **HU25** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada competición en la que opero para conocer mi trayectoria.
 - El sistema contará con una página exclusiva para mostrar datos sobre las competiciones.
 - Esta página debe mostrar el beneficio obtenido en una competición durante un periodo.
 - Cuenta con un selector de competición y otro de periodo para filtrar los datos.
 - Muestra una gráfica con la evolución del beneficio en dicha competición.
 - Esta tabla contiene a su vez un filtro para modificar la forma en la que muestran los datos. Este nos permitirá consultar la evolución los datos en distintas unidades (horas, días, semanas, meses y años).
- **HU26** - Como usuario quiero consultar mis estadísticas sobre mis beneficios para analizar mi situación actual.

- El objetivo es consultar el ROI (Return on Investment), ROC (Return on Capital), porcentaje de movimientos de tipo apuesta, beneficio medio y una tabla que muestre la evolución del beneficio en el tiempo.
 - En caso de no tener registros importados, se mostrará un 0.
 - Estos datos se encuentran en la página principal del producto.
- **HU27** - Como usuario quiero acceder a una página en la que analizar los datos específicos de cada tipo de operación en la que opero para conocer mi trayectoria.
 - El sistema contará con una página exclusiva para mostrar datos sobre las operaciones.
 - Esta página debe mostrar el beneficio obtenido en una operación durante un periodo.
 - Cuenta con un selector de operación y otro de periodo para filtrar los datos.
 - Muestra una gráfica con la evolución del beneficio en dicha operación.
 - Esta tabla contiene a su vez un filtro para modificar la forma en la que muestran los datos. Este nos permitirá consultar la evolución los datos en distintas unidades (horas, días, semanas, meses y años).
 - **HU28** - Como usuario quiero poder filtrar el contenido del panel de control por periodos para consultar mi evolución.
 - Desde el panel de control se podrá filtrar los datos en todo momento.
 - Se ofrece la opción de filtrado de datos diarios, semanales, mensuales o de un periodo personalizado.
 - También se podrá aplicar un filtro a las gráficas de forma independiente para modificar la forma en la que muestran los datos. Este nos permitirá consultar la evolución los datos en distintas unidades (horas, días, semanas, meses y años).

3.2. Análisis de requisitos no funcionales

- **Autenticación:** Solo se permite acceder al servidor proporcionando un token válido de un usuario registrado. El token no solo verifica la identidad del usuario, sino que también se utiliza para filtrar los datos que se devuelven.
- **Optimización de consultas:** Se debe minimizar el número de consultas a la base de datos para evitar grandes tiempos de espera en la aplicación. Se deben realizar consultas más complejas que devuelvan conjuntos de datos completos en lugar de realizar múltiples consultas sencillas.
- **Navegación intuitiva:** La interfaz de usuario debe contar con una barra lateral que facilite la navegación por la aplicación. Esto permitirá a los usuarios acceder rápidamente a las diferentes secciones de la web.
- **Diseño sencillo:** La interfaz debe ser simple y fácil de usar, asegurando que los usuarios comprendan rápidamente cómo interactuar con la aplicación.

Capítulo 4

Diseño del sistema/producto

4.1. Diseño de la base de datos

En esta sección se indica como se diseñó la base de datos. Se va a trabajar con una base de datos relacional MySQL. Al inicio del proyecto se realizó un diseño inicial de la base de datos, pero esta fue evolucionando durante el proyecto en función de las necesidades del producto.

Inicialmente, se plantearon los diseños conceptual y lógico presentes en las figuras 4.1 y 4.2 respectivamente, pero la empresa decidió simplificar el esquema para permitir consultas más sencillas a través de la herramienta PHP-CRUD-API (siendo esta utilizada para relaizar consultas sencillas sin necesidad de implementar desde cero un servicio). En esta nueva versión se simplificó la tabla *Movement* y se añadió una tabla por cada elemento cuyos posibles valores conocíamos de antemano. También se añadió la tabla *User* cuando se implementó el sistema de pago. Esta nueva tabla se añadió debido a la necesidad de almacenar el plan de pago al que se encuentra suscrito cada usuario. Junto a esta modificación se añadió un disparador (*trigger* en inglés) para verificar que el plan de pago se encuentra entre los disponibles.

En la figura 4.3 podemos observar el diseño lógico final de la base de datos. Por otra parte podemos observar el diseño físico y del disparador en los fragmentos de código 4.1 y 4.2.

```
1 -- tables
2 -- Table: BetType
3 CREATE TABLE BetType (
4     Name varchar(100) NOT NULL,
5     Id int(11) NOT NULL AUTO_INCREMENT,
6     CONSTRAINT BetType_pk PRIMARY KEY (Id)
7 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
8
9 -- Table: Competition
10 CREATE TABLE Competition (
11     Name varchar(100) NOT NULL,
12     Id int(11) NOT NULL AUTO_INCREMENT,
13     UNIQUE INDEX Name (Name),
14     CONSTRAINT Competition_pk PRIMARY KEY (Id)
15 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

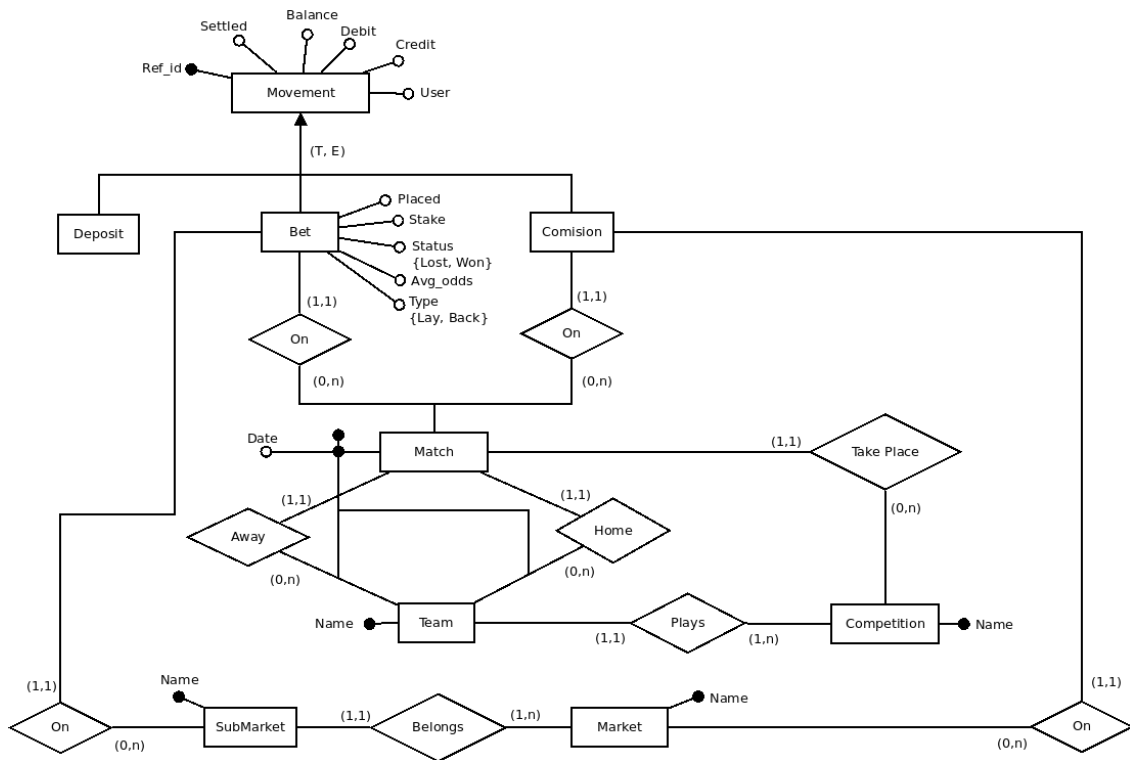


Figura 4.1: Diseño conceptual inicial de la base de datos.

```

16
17 -- Table: Market
18 CREATE TABLE Market (
19     Name varchar(100) NOT NULL,
20     Id int(11) NOT NULL AUTO_INCREMENT,
21     UNIQUE INDEX Name (Name),
22     CONSTRAINT Market_pk PRIMARY KEY (Id)
23 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
24
25 -- Table: Movement
26 CREATE TABLE Movement (
27     User varchar(100) NOT NULL,
28     Ref_id varchar(50) NOT NULL,
29     Type int(11) NOT NULL,
30     Settled datetime NOT NULL,
31     Placed datetime NULL DEFAULT null,
32     Stake float NULL DEFAULT null,
33     Competition_id int(11) NULL DEFAULT null,
34     Home_id int(11) NULL DEFAULT null,
35     Away_id int(11) NULL DEFAULT null,
36     SubMarket_id int(11) NULL DEFAULT null,
37     BetType_id int(11) NULL DEFAULT null,
38     Avg_odds float NULL DEFAULT null,
39     CashOut tinyint(1) NULL DEFAULT null,
40     Status tinyint(1) NULL DEFAULT null,
41     Debit float NOT NULL,
42     Credit float NOT NULL,
43     Balance float NOT NULL,
44     Id int(11) NOT NULL AUTO_INCREMENT,
45     UNIQUE INDEX Ref_id (Ref_id),

```

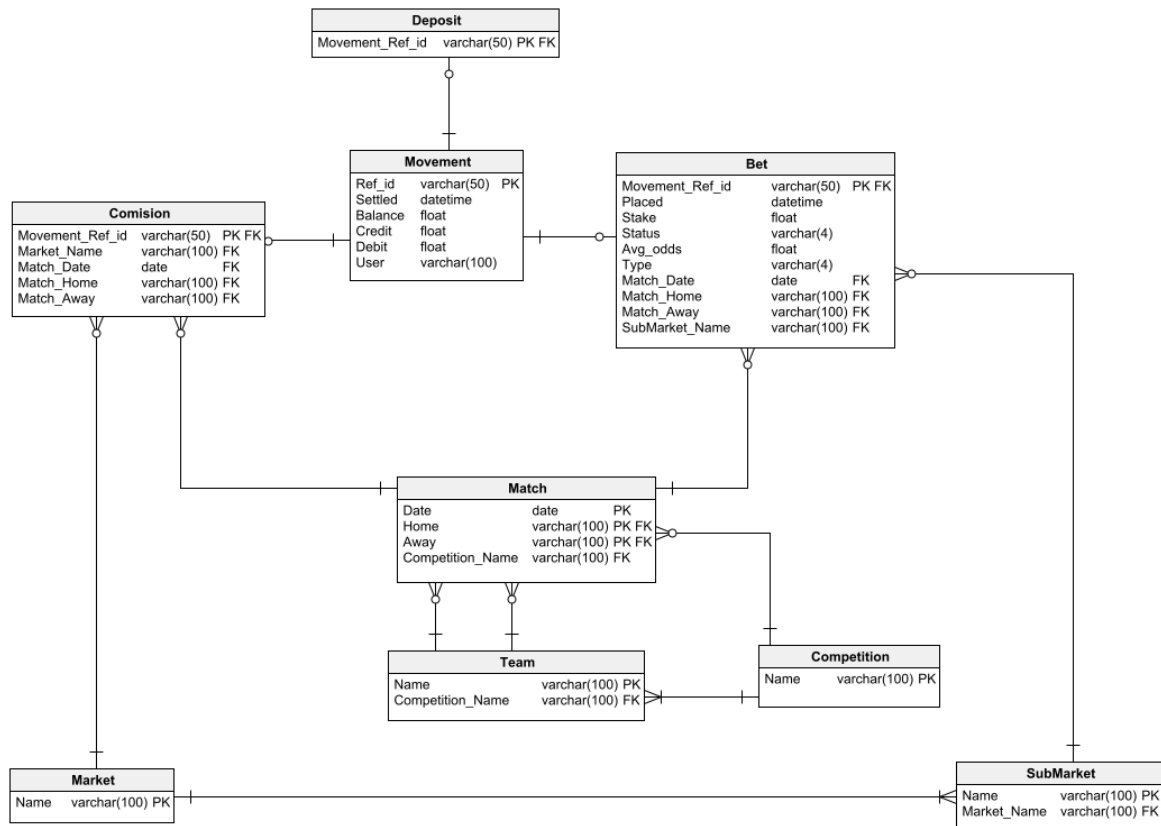


Figura 4.2: Diseño lógico inicial de la base de datos.

```

46     CONSTRAINT Movement_pk PRIMARY KEY (Id)
47 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
48
49 CREATE INDEX Movement_SubMarket ON Movement (SubMarket_id);
50
51 CREATE INDEX Movement_Competition ON Movement (Competition_id);
52
53 CREATE INDEX Movement_Home_Team ON Movement (Home_id);
54
55 CREATE INDEX Movement_Away_Team ON Movement (Away_id);
56
57 CREATE INDEX Movement_Type ON Movement (Type);
58
59 CREATE INDEX Movement_BetType ON Movement (BetType_id);
60
61 CREATE INDEX Movement_User ON Movement (User);
62
63 -- Table: SubMarket
64 CREATE TABLE SubMarket (
65     Name varchar(100) NOT NULL,
66     Market_id int(11) NOT NULL,
67     Id int(11) NOT NULL AUTO_INCREMENT,
68     UNIQUE INDEX Name (Name),
69     CONSTRAINT SubMarket_pk PRIMARY KEY (Id)
70 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;

```

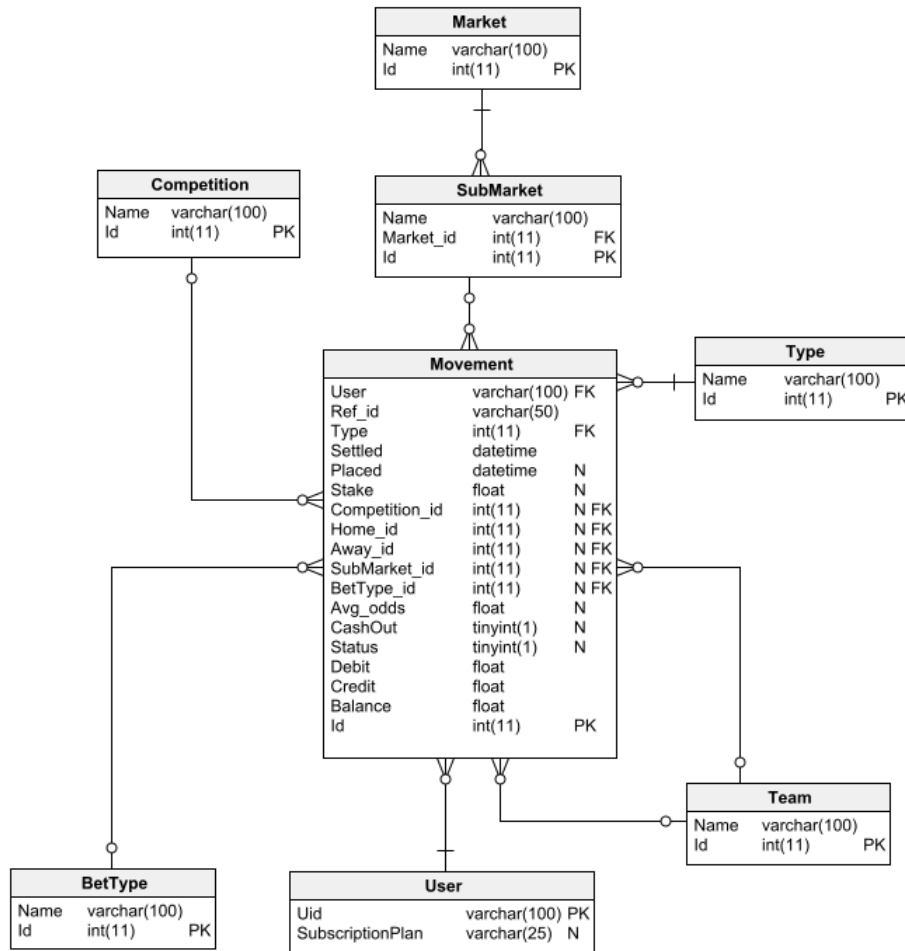


Figura 4.3: Diseño lógico final de la base de datos.

```

71
72 CREATE INDEX SubMarket_Market ON SubMarket (Market_id);
73
74 -- Table: Team
75 CREATE TABLE Team (
76     Name varchar(100) NOT NULL,
77     Id int(11) NOT NULL AUTO_INCREMENT,
78     UNIQUE INDEX Name (Name),
79     CONSTRAINT Team_pk PRIMARY KEY (Id)
80 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
81
82 -- Table: Type
83 CREATE TABLE Type (
84     Name varchar(100) NOT NULL,
85     Id int(11) NOT NULL AUTO_INCREMENT,
86     CONSTRAINT Type_pk PRIMARY KEY (Id)
87 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
88
89 -- Table: User
90 CREATE TABLE User (

```



```

91     Uid varchar(100) NOT NULL,
92     SubscriptionPlan varchar(25) NULL DEFAULT null,
93     CONSTRAINT User_pk PRIMARY KEY (Uid)
94 ) ENGINE InnoDB CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
95
96 -- foreign keys
97 -- Reference: Movement_Away_Team (table: Movement)
98 ALTER TABLE Movement ADD CONSTRAINT
99 Movement_Away_Team FOREIGN KEY Movement_Away_Team (Away_id)
100     REFERENCES Team (Id);
101
102 -- Reference: Movement_BetType (table: Movement)
103 ALTER TABLE Movement ADD CONSTRAINT
104 Movement_BetType FOREIGN KEY Movement_BetType (BetType_id)
105     REFERENCES BetType (Id);
106
107 -- Reference: Movement_Competition (table: Movement)
108 ALTER TABLE Movement ADD CONSTRAINT
109 Movement_Competition FOREIGN KEY Movement_Competition (Competition_id)
110     REFERENCES Competition (Id);
111
112 -- Reference: Movement_Home_Team (table: Movement)
113 ALTER TABLE Movement ADD CONSTRAINT
114 Movement_Home_Team FOREIGN KEY Movement_Home_Team (Home_id)
115     REFERENCES Team (Id);
116
117 -- Reference: Movement_SubMarket (table: Movement)
118 ALTER TABLE Movement ADD CONSTRAINT
119 Movement_SubMarket FOREIGN KEY Movement_SubMarket (SubMarket_id)
120     REFERENCES SubMarket (Id);
121
122 -- Reference: Movement_Type (table: Movement)
123 ALTER TABLE Movement ADD CONSTRAINT
124 Movement_Type FOREIGN KEY Movement_Type (Type)
125     REFERENCES Type (Id);
126
127 -- Reference: Movement_User (table: Movement)
128 ALTER TABLE Movement ADD CONSTRAINT
129 Movement_User FOREIGN KEY Movement_User (User)
130     REFERENCES User (Uid)
131     ON UPDATE CASCADE;
132
133 -- Reference: SubMarket_Market (table: SubMarket)
134 ALTER TABLE SubMarket ADD CONSTRAINT
135 SubMarket_Market FOREIGN KEY SubMarket_Market (Market_id)
136     REFERENCES Market (Id);

```

Listing 4.1: Diseño físico de la base de datos.

```

1 CREATE TRIGGER 'trg_ValidateSubscriptionPlan'
2 BEFORE INSERT ON 'User' FOR EACH ROW BEGIN
3     DECLARE InvalidSubscriptionPlanError CONDITION FOR SQLSTATE '45000';
4
5     IF NEW.SubscriptionPlan NOT IN ('Free-Month', 'Free-Year',
6     'Premium-Month', 'Premium-Year', 'Pro-Month', 'Pro-Year') THEN
7         SIGNAL InvalidSubscriptionPlanError
8         SET MESSAGE_TEXT = "Invalid SubscriptionPlan value.";
9     END IF;
10 END

```

Listing 4.2: Disparador de la base de datos.

4.2. Diseño de software

4.2.1. Descripción general del diseño del sistema

Este proyecto presenta un diseño bastante sencillo debido a las herramientas utilizadas en su desarrollo. Se han empleado PHP-CRUD-API en el servidor y ReactJS junto a Metronic en la vista, lo que ha permitido reducir el coste de desarrollo en ambas partes de la aplicación. Estas dos herramientas priorizan el uso de funciones, en lugar de clases y objetos.

ReactJS encapsula el código de la vista en funciones que devuelven código HTML, las cuales son llamadas componentes. Estas funciones también contienen la lógica encargada de actualizar la vista utilizando el patrón *Observer* mediante *hooks*, y las llamadas a otras funciones responsables de la transformación de datos y la comunicación con servicios externos a la web.

En lo que a estilos se refiere, Metronic clases propias combinadas con las que ofrece Bootstrap. Esto permite a los desarrolladores la posibilidad de crear diseños visualmente agradables y consistentes, manteniendo una estructura modular y escalable.

Por otro lado, PHP-CRUD-API nos brinda un servicio CRUD (Crear, Leer, Actualizar, Eliminar en inglés) que nos permite comunicarnos con nuestra base de datos simplemente añadiendo los datos de la misma a la herramienta. Además, ofrece la posibilidad de crear puntos finales (*endpoints* en inglés) personalizados mediante el uso de funciones.

En conclusión, este proyecto destaca por su diseño sencillo y eficiente, gracias a las herramientas utilizadas. El enfoque en el uso de funciones en lugar de clases y objetos ha facilitado la implementación y el mantenimiento del proyecto, a pesar de que ha limitado las opciones de diseño.

4.2.2. React hooks

React en su versión 16.8.0 incorporó los *hooks*. Estos ofrecen una alternativa al patrón de diseño *Observer* [18] [5] para los desarrolladores que utilicen un enfoque funcional en el desarrollo de vistas web reactivas, dando lugar a código más simple y fácil de leer. En el caso de este proyecto se ha empleado este enfoque, por lo que se han utilizado los *hooks* para la gestión de estados.

4.2.3. Composición

Al optar por un enfoque funcional en React se debe trabajar empleando composición [17] [16] [4]. La composición es el acto de combinar distintos componentes para formar uno más grande y complejo. Esta forma de trabajar permite tanto el desarrollo de componentes genéricos como el de componentes especializados.

Los componentes genéricos nos permiten crear componentes que realizan funciones de forma

independiente a sus hijos. La creación de estos componentes genéricos la facilita React al permitir trabajar con componentes hijos sin conocer la identidad de los mismos hasta el momento de la implementación. Se puede ver un ejemplo sencillo de composición en el fragmento de código 4.3.

```
1 const TextRed = (props) => {
2   return (
3     <div className='text-red'>
4       {props.children}
5     </div>
6   );
7 }
8
9 const NewText = () => {
10  return (
11    <TextRed>
12      <p className="Dialog-message">Hello world!</p>
13    </TextRed>
14  )
15 }
```

Listing 4.3: Composición con componentes genéricos en React.

Por otra parte, permite la creación de componentes específicos, permitiendo la creación de estos a partir de componentes más genéricos. En el fragmento de código 4.4 se puede ver un ejemplo de este tipo de composición.

```
1 const Title = (props) => {
2   return (
3     <h1>
4       {props.title}
5     </h1>
6     <h2>
7       {props.message}
8     </h2>
9   );
10 }
11
12 const LoginTitle = () => {
13   return (
14     <Title title="Login" message="Now you can login with Google!"/>
15   )
16 }
```

Listing 4.4: Composición con componentes específicos en React.

4.3. Diseño de la arquitectura del sistema/producto

La arquitectura del panel de control es una arquitectura cliente/servidor de tres capas. Esta fue decidida por la empresa al optar por el uso de la herramienta PHP-CRUD-API. Se ha alojado tanto el servidor como la vista en un servidor web alojado por la empresa Hostinger.

Esta arquitectura es la más utilizada en la mayoría de aplicaciones web cliente/servidor clásicas, permite separar la vista de la lógica de la aplicación como muestra la figura 4.4.



Figura 4.4: Diseño de la arquitectura del producto.

4.4. Diseño del servidor

Como se ha mencionado anteriormente, a la hora de trabajar el servidor del proyecto se utilizó la herramienta PHP-CRUD-API. Al trabajar con esta herramienta, el diseño en el servidor se limitó a la ampliación de *endpoints* utilizando funciones personalizadas. Los *endpoints* personalizados desarrollados se pueden consultar en la tabla 4.1.

Método	Nombre	Descripción
GET	/get-summary	Este <i>endpoint</i> se encarga de devolver el balance, el beneficio, las apuestas ganadas, las comisiones, los depósitos y los retiros del usuario en el periodo de tiempo indicado.
GET	/get-best-subMarkets	Este <i>endpoint</i> devuelve un listado de los 6 submercados en los que el usuario tiene mejor rendimiento junto al beneficio obtenido en los mismos.
GET	/get-profit-stats	Este <i>endpoint</i> se encarga de devolver para un periodo de tiempo determinado el balance en distintos puntos del periodo, ROC (<i>Return On Capital</i>), ROI (<i>Return On Investment</i>), el número de apuestas y el rendimiento diario por apuesta.
GET	/get-balance-chart-data	Este <i>endpoint</i> devuelve los datos necesarios para mostrar la evolución de beneficios y pérdidas en un periodo determinado.
GET	/get-markets-chart-data	Este <i>endpoint</i> devuelve los datos necesarios para mostrar la evolución del rendimiento de los seis mercados con más beneficio en un periodo determinado.
GET	/get-analysis-chart-data	Este <i>endpoint</i> devuelve los datos necesarios para mostrar la evolución del rendimiento de el elemento seleccionado para analizar en un periodo determinado.
GET	/get-analysis-profit	Este <i>endpoint</i> devuelve el beneficio y las comisiones de el elemento seleccionado para analizar en un periodo determinado.
DELETE	/delete-table-element	Este <i>endpoint</i> elimina todos los elementos de la tabla <i>Movement</i> en los que el valor RefId coincide con el proporcionado como parámetro.

Cuadro 4.1: *Endpoints* personalizados añadidos a PHP-CRUD-API.

4.5. Diseño de las interfaces

En el diseño de la interfaz, se ha utilizado como punto de partida las plantillas, componentes y estilos proporcionados por Metronic. La plantilla mostrada en la figura 4.5 ha servido como base inicial. Además, se han creado varios bocetos para mostrar cómo debería lucir la aplicación, dedicando uno para cada sección del proyecto. Estos bocetos han sido desarrollados teniendo en cuenta la plantilla y los componentes de Metronic, lo que garantiza que reflejen fielmente el diseño final del sitio web.

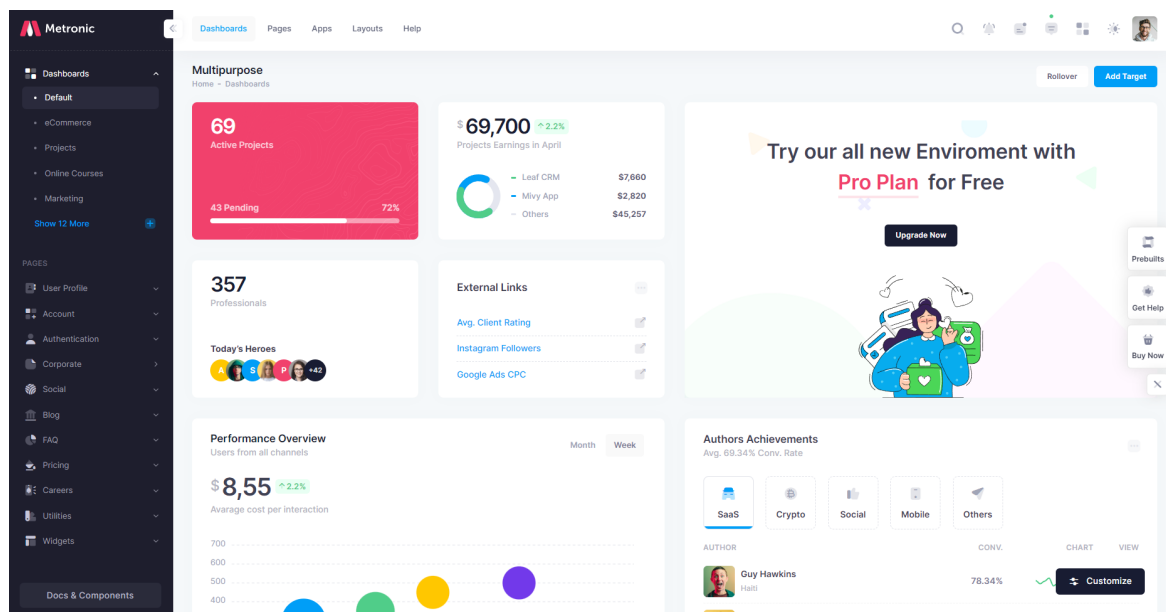


Figura 4.5: Plantilla base de Metronic.

El primero muestra la página principal del proyecto, en esta se pueden consultar datos como el balance, beneficio, número de apuestas y el rendimiento en distintos mercados entre otros datos relevante. Esta página es de las más importantes de la aplicación, ya que permite tener una imagen global de la situación en la que se encuentra el usuario en lo relacionado con apuestas deportivas. Este boceto se puede observar en la figura 4.6

También se diseñó un boceto para mostrar como debían verse las tablas que muestran los datos importados y permiten importar. El boceto muestra como la tabla simplemente debe mostrar los datos y dar opción de filtrar y ordenar los mismos. Podemos observar esto en la figura 4.7.

Por otra parte se realizó el diseño de una de las páginas de análisis, estas debían permitir consultar distinta información (como el beneficio o las comisiones) de los movimientos en función del mercado, la competición o el tipo de apuesta. En el boceto presente en la figura 4.8 nos centramos en la página de análisis por tipo de apuesta.

Por último se creó el boceto para la página de pagos, esta contiene los distintos planes de suscripción, un selector para cambiar entre planes mensuales y anuales, una descripción del plan de pago seleccionado y el botón para pagara desde PayPal. Esto se puede ve en la figura 4.9.

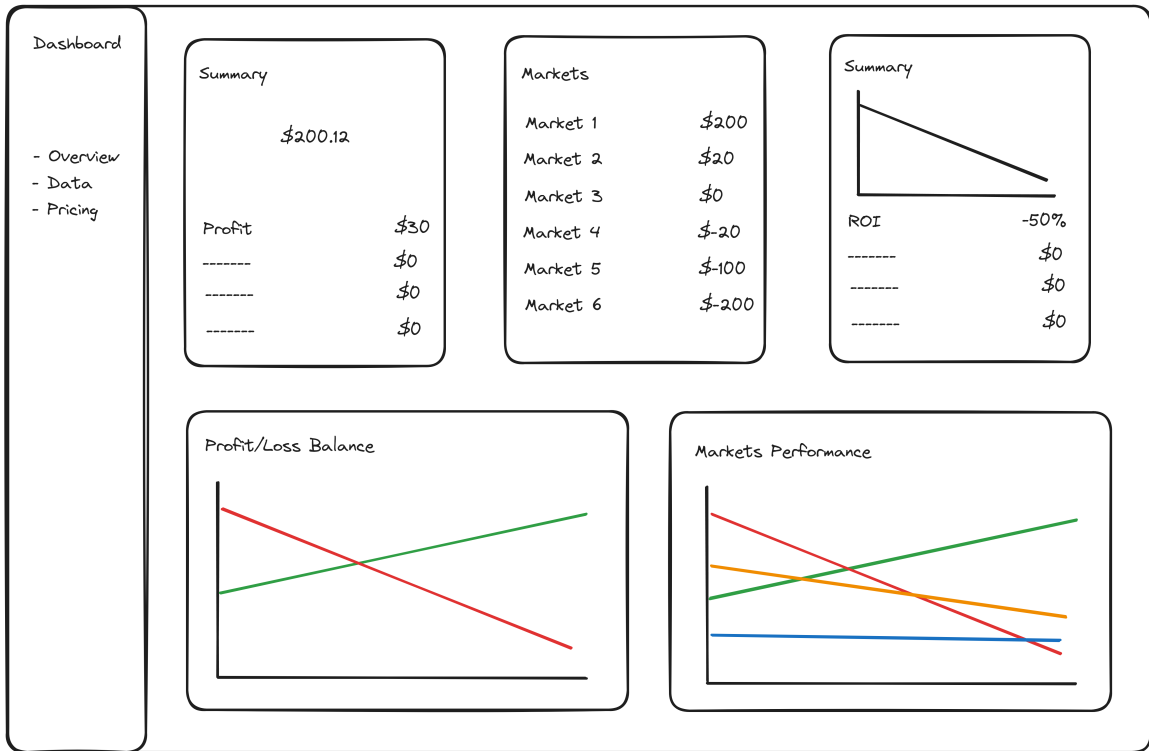


Figura 4.6: Boceto de la página principal del producto.

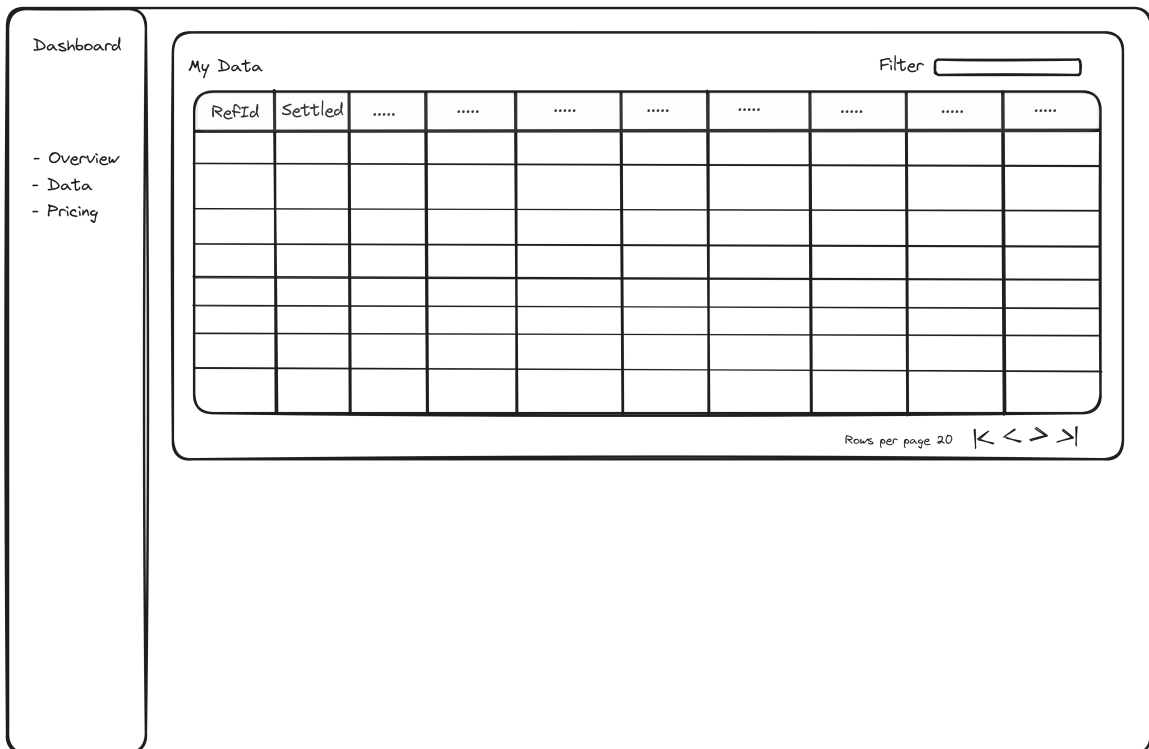


Figura 4.7: Boceto de la tabla de datos.

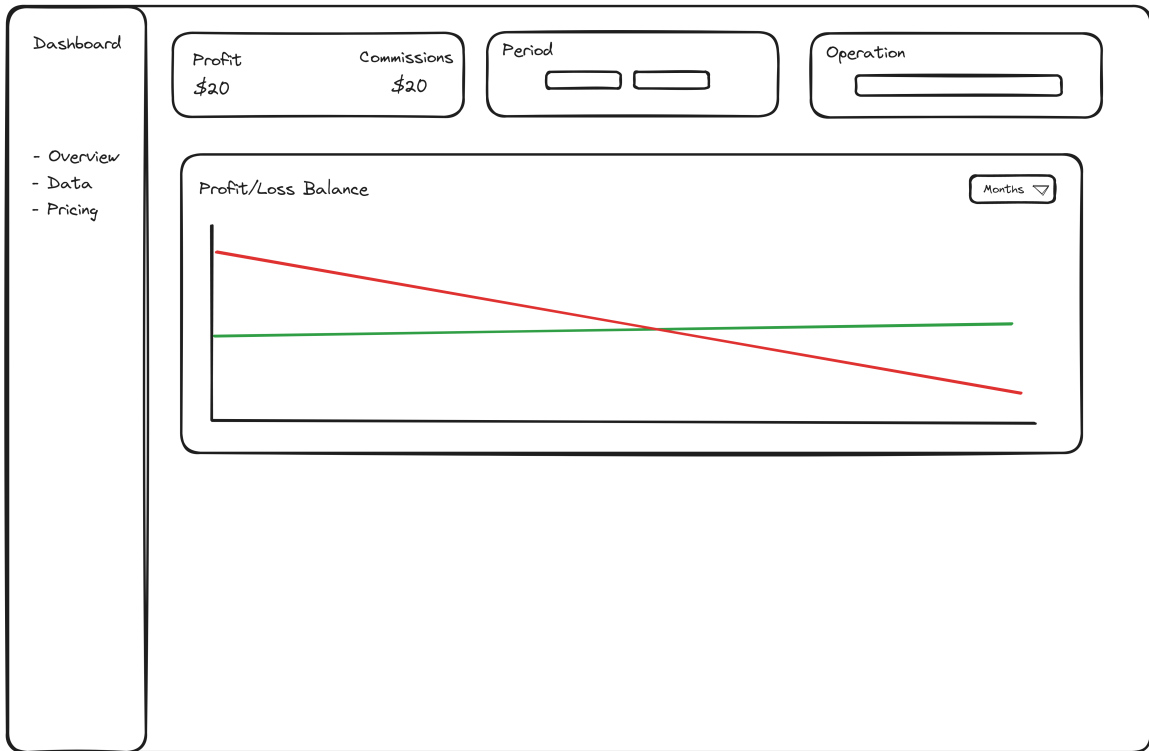


Figura 4.8: Boceto de la página de análisis.



Figura 4.9: Boceto de la página de pagos.

4.5.1. Sitemap

Se ha desarrollado un *sitemap* para mostrar las páginas presentes en el proyecto y la forma de navegar entre las mismas. En este se muestra una página principal externa a lo desarrollado durante el proyecto que conecta con el mismo. Se puede observar el *sitemap* en la figura 4.10.

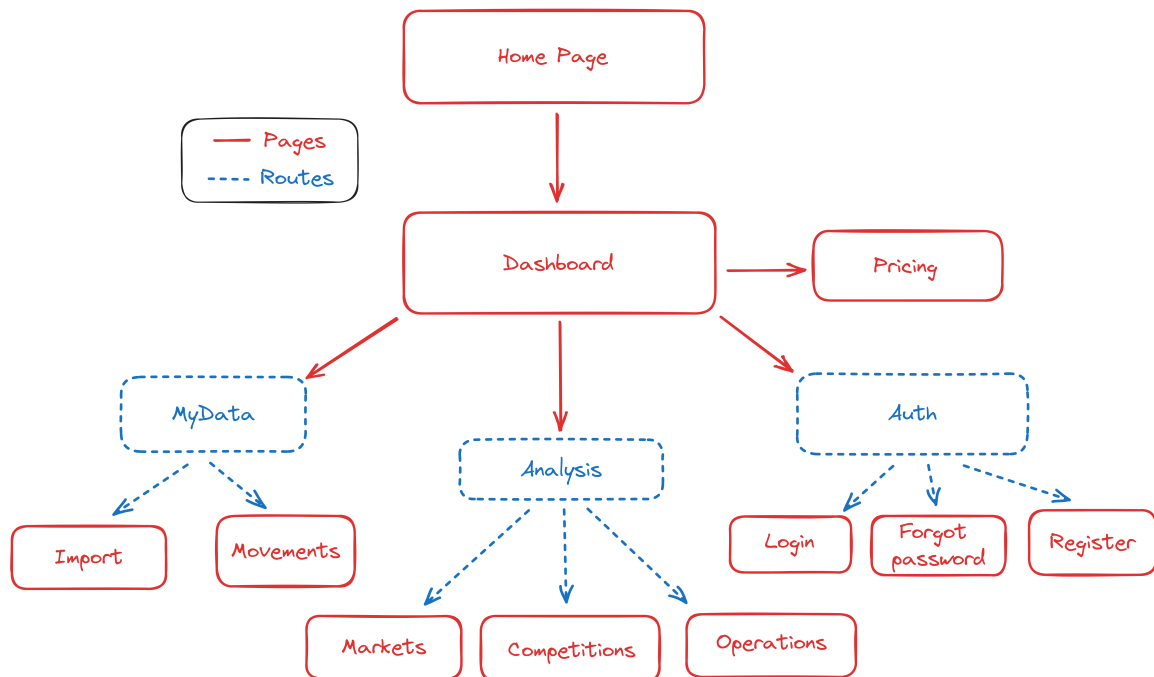


Figura 4.10: Sitemap del proyecto.

Capítulo 5

Implementación y pruebas

5.1. Estructura del código

En la vista del proyecto (presente en el directorio `src/app`) se ha seguido la estructura de directorios propuesta por Metronic, la cual consiste en separar las páginas principales de su contenido. Como se puede observar en en la figura 5.1 contamos con un directorio llamada *dashboardPages* en la que se encuentran las páginas y sus rutas, en este caso contamos con una única página principal en la que se mostrarán todas las demás a excepción de las relacionadas con autenticación.

Por otra parte el contenido de esta página se encuentra en el directorio *modules*. Este directorio contiene subdirectorios que contienen de forma independiente cada funcionalidad de la web. Estos a su vez contienen los archivos con el contenido a mostrar y dos directorios, *core* y *components*. En *core* se almacenan todas las funciones encargadas de la transformación de datos y comunicación con servicios externos, mientras que en *components* encontramos distintos elementos de menor tamaño presentes en la interfaz como botones, filtros o tablas.

Se ha añadido un directorio llamado *model* en la que se almacena la interfaz *DataTableInterface* y las enumeraciones *BetType*, *StatusType* y *MovementType*. Estos elementos se emplean para trabajar con los datos presentes en las tablas.

En el proyecto, nos encontramos con dos tipos de archivos: `.tsx` y `.ts`. Los archivos `.tsx` contienen una combinación de HTML y TypeScript, y son responsables de la parte visual y del manejo de los estados. La estructura de estos archivos se muestra en el fragmento de código 5.1. Al principio del documento, se define una interfaz que especifica los objetos que recibe como parámetro la función que crea el componente. Dentro de la función que crea el componente, se encuentra la lógica encargada de manejar los estados (la sección ubicada debajo del comentario *ALERT* en ese caso). Por último, encontramos el código que define la vista del componente.

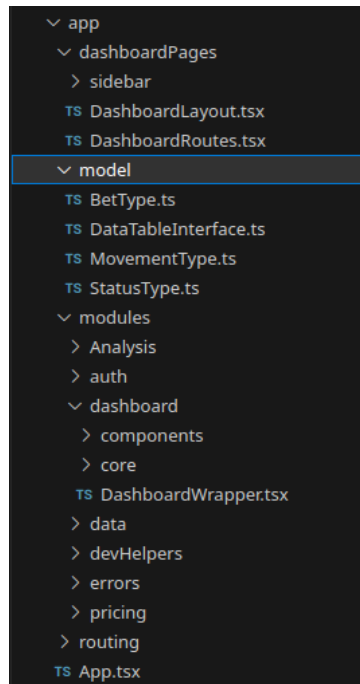


Figura 5.1: Directorios del proyecto.

```

1 interface Props {
2   show: boolean
3   message: string
4 }
5
6 const ImportAlert =(props: Props) =>{
7
8 // ----- ALERT -----
9
10  const [alert, setAlert] = useState(false);
11
12  useEffect(() => {
13    if(!props.show){
14      const timeout = setTimeout(() => {
15        setAlert(false);
16      }, 200);
17      return () => clearTimeout(timeout);
18    }else{
19      setAlert(true);
20    }
21  }, [alert, props.show]);
22
23  return(
24    <div
25      className="alert alert-dismissible bg-light-danger
26      d-flex align-items-center p-5 mb-10">
27      <KTSVG path="/media/icons/duotone/general/gen040.svg"
28      className="svg-icon svg-icon-2hx svg-icon-danger me-4 mb-5 mb-
29      sm-0"/>
30      <div className="d-flex flex-column">
31        <h5 className="mb-1">Error</h5>
32        <span>{props.message}</span>

```

```

32         </div>
33     </div>
34 )
35 }
36
37 export { ImportAlert }

```

Listing 5.1: Formato de los archivos .tsx.

A diferencia de los archivos .tsx, los archivos .ts contienen una función a la que los anteriores pueden llamar en caso de necesidad. Además, los archivos .ts pueden incluir funciones adicionales para brindar soporte a la función principal, lo que permite separar el código y hacerlo más legible. Sin embargo, es importante destacar que estas funciones adicionales no se exportan del archivo. Las funciones presentes en estos documentos suelen emplearse para llamar al servidor o formatear datos. Un ejemplo de este tipo de funciones se puede encontrar en el fragmento de código 5.2.

```

1 const getCurrentPlan = async (): Promise<string> => {
2     const token = await getAuth().currentUser?.getIdToken()
3     var newUserReq = {
4         method: 'GET',
5         headers: {
6             'Content-Type': 'application/json',
7             Authorization: token!,
8         },
9         mode: 'cors' as RequestMode,
10    }
11    try {
12        const response = await fetch(
13            'https://mybettingdata.com/apiorbit/api.php/records/User?include=
SubscriptionPlan&filter=Uid,eq,' +
14            getAuth().currentUser?.uid,
15            newUserReq
16        )
17        if (!response.ok) {
18            throw new Error(response.status + ' (' + response.statusText + ')')
19        }
20        const json = await response.json()
21        return json.records[0].SubscriptionPlan
22    } catch (error) {
23        console.log(GetErrorMessage(error))
24        throw error
25    }
26 }
27
28 export {getCurrentPlan}

```

Listing 5.2: Formato de los archivos .ts.

Por otro lado, en la parte del servidor, se creó un archivo PHP que desempeña un papel importante en la definición de los *endpoints* y las funciones asociadas a ellos. Este archivo se ubica en el mismo directorio que el servicio PHP-CRUD-API, y en la configuración de dicho servicio se debe especificar el nombre del nuevo archivo. A su vez, se ha añadido una función que se encarga de llamar a la API de Firebase para comprobar que el Token recibido es válido y obtener del mismo el identificador del usuario.

5.2. Descripción técnica de la implementación

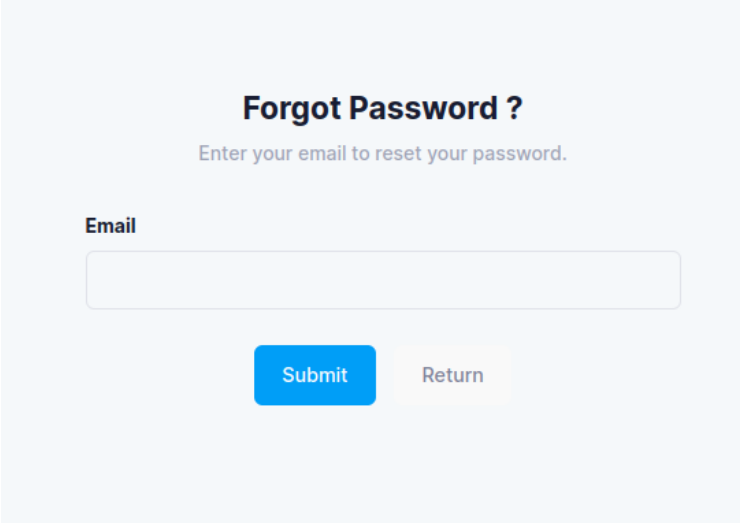
5.2.1. Módulo de autenticación

La implementación del proyecto se inició con el desarrollo del módulo de autenticación. Para ello, se utilizó como punto de partida las plantillas proporcionadas por Metronic, aprovechando los estilos y componentes predefinidos, como botones y formularios. Estas plantillas fueron ajustadas y personalizadas según las necesidades específicas del proyecto.

Posteriormente, se procedió a crear un proyecto en Firebase y se estableció la conexión entre este proyecto y la vista del proyecto utilizando el módulo de Firebase para JavaScript. Esta integración permitió utilizar las funcionalidades y servicios de autenticación proporcionados por Firebase de manera sencilla y segura.

Una vez que la configuración inicial estaba lista, se implementaron diferentes métodos relacionados con el proceso de autenticación. Esto incluyó métodos para el inicio de sesión utilizando correo y contraseña, inicio de sesión con Google, cierre de sesión y recuperación de contraseña.

Las distintas páginas desarrolladas para este módulo se pueden observar en las figuras 5.3, 5.2 y 5.4.




Forgot Password ?
Enter your email to reset your password.

Email

Submit **Return**

Figura 5.2: Página de recuperación de contraseña.

Sign In
Your Personal Dashboard

 Sign in with Google

Or with email

Email

Email

Password

Forgot Password ?

Continue


Not a Member yet? [Sign up](#)

[Terms](#) [Plans](#) [Contact Us](#)

Figura 5.3: Página de inicio de sesión.

Sign Up

Your Social Campaigns

 Sign in with Google

Or with email

Email

Email

Password

Password

Use 8 or more characters with a mix of letters, numbers & symbols.

Confirm Password

Password confirmation

I Accept the [Terms](#).

Submit

Return

[Terms](#) [Plans](#) [Contact Us](#)

Figura 5.4: Página de registro.

5.2.2. Módulo de importación y consulta de datos

Una vez implementado el módulo de autenticación, se dio inicio al desarrollo del módulo de importación y consulta de datos. Este módulo permite al usuario agregar nueva información sobre apuestas deportivas en la aplicación y posteriormente obtener datos a partir de ellos.

El proceso de desarrollo se inició con la creación de la tabla destinada a mostrar los datos. Sin embargo, surgieron algunos desafíos debido a la incompatibilidad entre las tablas proporcionadas por Metronic y React. Para superar este obstáculo, se realizaron pruebas con diferentes bibliotecas de tablas de datos y se optó finalmente por utilizar **react-data-table-component** [3]. Esta biblioteca ofrece una tabla altamente personalizable que incluye opciones para seleccionar elementos, paginar resultados y mostrar componentes adicionales en la cabecera. Una vez seleccionada la herramienta para crear las tablas, se procedió al desarrollo de dos páginas: una para la importación de datos y otra para la consulta de datos.

La página de importación de datos consta de una tabla vacía y dos botones. El primer botón, llamado *Paste Data*, se encarga de leer el último mensaje copiado en el portapapeles y procesarlo para convertir los datos provenientes del mercado de intercambio Orbit al formato requerido para su visualización en la tabla. En caso de que el formato no coincida, se muestra un mensaje de error al usuario. Una vez copiados los datos, se presentan en la tabla y, al hacer clic en el botón *Import Data*, se inicia el proceso de importación de los datos en la base de datos. Si los datos ya existen en la base de datos, se notifica al usuario mediante un mensaje y se mantienen los registros repetidos en la tabla. En caso de que la importación sea exitosa, se eliminan las entradas de la tabla y se notifica al usuario con un mensaje que indica el éxito de la operación de importación. En caso de fallar también se muestra un mensaje indicando que no se han podido importar los datos, manteniendo los mismos en la tabla para poder intentar insertar los datos de nuevo.

La página de consulta de datos consta de una tabla que muestra todos los datos importados por el usuario, así como una herramienta de filtrado. Esta herramienta de filtrado permite realizar dos tipos de filtrado: por tipo de movimiento utilizando selectores y por texto, donde se realiza una búsqueda en todos los campos no numéricos de la tabla para encontrar coincidencias con el texto ingresado. Además, el filtro cuenta con un botón que permite borrar los parámetros del filtro y mostrar nuevamente todos los datos sin filtrar.

Esta tabla también ofrece la funcionalidad de seleccionar elementos. Al seleccionar uno o varios elementos de la tabla, se muestra un botón que permite eliminar los movimientos seleccionados. Este borrado se realiza utilizando el *endpoint* personalizado `/delete-table-element`.

Para asegurar el correcto funcionamiento de la función que transforma los datos del portapapeles a una lista de objetos del tipo *DataTableInterface* se realizaron una serie de test unitarios.

Los distintos elementos desarrollados para este módulo se pueden observar en las figuras 5.5, 5.6, 5.7 y 5.8.

Import Data

Copy your data from Orbit and press paste

[Paste Data](#) [Import Data](#)

<input type="checkbox"/>	Settled	Refid	Type	Placed	Home	Away	Competi...	Market	SubMar...	Bet type	Triggere...	Average ...	Stake	Status	Debit	Credit	Balance
<input type="checkbox"/>	2023-02-17 21:01	1,209976073	Commissio n	--	Tondela	Academico de Viseu	Portuguese Segunda Liga	Match Odds	--	--	--	0	0	--	3.96	0	1312.23
<input type="checkbox"/>	2023-02-17 21:01	100688336	Bet	2023-02-17 12:37	Tondela	Academico de Viseu	Portuguese Segunda Liga	Match Odds	The Draw	Back	false	3.2	60	Won	0	132	1316.19
<input type="checkbox"/>	2023-02-17 11:38	100655738	Bet	2023-02-16 23:26	Western United	Wellington Phoenix	Australian A-League Men	Match Odds	The Draw	Back	false	3.75	60	Lost	60	0	1184.19
<input type="checkbox"/>	2023-02-15 19:55	1,210032431	Commissio n	--	Pesaro	Pontedera	Italian Serie C	Match Odds	--	--	--	0	0	--	3.32	0	1244.19
<input type="checkbox"/>	2023-02-15 19:54	100521666	Bet	2023-02-15 17:49	Pesaro	Pontedera	Italian Serie C	Match Odds	The Draw	Back	false	3.05	54	Won	0	110.7	1247.51
<input type="checkbox"/>	2023-02-11 23:01	1,209766258	Commissio n	--	San Telmo	Defensores de Belgrano	Argentinian Primera B Nacional	Match Odds	--	--	--	0	0	--	3	0	1136.81
<input type="checkbox"/>	2023-02-11 23:01	100145259	Bet	2023-02-11 20:48	San Telmo	Defensores de Belgrano	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	3	50	Won	0	100	1139.81
<input type="checkbox"/>	2023-02-11 02:58	1,209766708	Commissio n	--	Gimnasia Jujuy	Ferro Carril Oeste	Argentinian Primera B Nacional	Match Odds	--	--	--	0	0	--	7.7	0	1039.81
<input type="checkbox"/>	2023-02-11 02:57	100021978	Bet	2023-02-10 22:54	Gimnasia Jujuy	Ferro Carril Oeste	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	2.82	141	Won	0	256.62	1047.51
<input type="checkbox"/>	2023-02-10 22:57	99999295	Bet	2023-02-10 20:23	Cadiz	Girona	Spanish La Liga	Match Odds	The Draw	Back	false	3.2	94	Lost	94	0	790.89

Rows per page: 10 ▾ 1-10 of 14 |< < > >|

✓ **Success**
 Data readed without errors.

Figura 5.5: Página de importación de datos.

My Movements

Query, filter and search your movements

Filter Table X

Bet
 Commission
 Deposit

<input type="checkbox"/>	Settled	Refid	Type	Placed	Home	Away	Competi...	Market	SubMar...	Bet type	Triggere...	Average ...	Stake	Status	Debit	Credit	Balance
<input type="checkbox"/>	2023-03-12 16:56:00	103179327	Bet	2023-03-12 14:37:00	Mafra	Academico de Viseu	Portuguese Segunda Liga	Match Odds	The Draw	Back	false	3.3	210	Lost	210	0	904.51
<input type="checkbox"/>	2023-03-12 13:56:00	103148956	Bet	2023-03-12 09:30:00	Covilha	Benfica B	Portuguese Segunda Liga	Match Odds	The Draw	Back	false	3.55	140	Lost	140	0	1114.51
<input type="checkbox"/>	2023-03-12 01:00:00	103121252	Bet	2023-03-11 23:04:00	Alvarado	CA Guemes	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	2.8	70	Lost	70	0	1254.51
<input type="checkbox"/>	2023-03-11 22:56:00	103095057	Bet	2023-03-11 19:47:00	Deportivo Madryn	Deportivo Riestra	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	2.94	70	Lost	70	0	1324.51
<input type="checkbox"/>	2023-03-08 02:04:00	102620110	Bet	2023-03-07 10:55:00	Brasil de Pelotas	Ponte Preta	Brazilian Cup	Match Odds	The Draw	Back	false	3.05	6.91	Lost	6.91	0	1394.51
<input type="checkbox"/>	2023-03-08 02:04:00	102620097	Bet	2023-03-07 10:55:00	Brasil de Pelotas	Ponte Preta	Brazilian Cup	Match Odds	The Draw	Back	false	3.1	323.09	Lost	323.09	0	1401.42
<input type="checkbox"/>	2023-03-07 20:01:00	102667467	Bet	2023-03-07 19:49:00	Altinordu	Goztepe	Turkish 1 Lig	Match Odds	Goztepe	Lay	false	1.18	330	Lost	59.4	0	1724.51
<input type="checkbox"/>	2023-03-06 22:53:00	102584315	Bet	2023-03-06 20:22:00	Defensores de Belgrano	Guillermo Brown	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	3.05	220	Lost	220	0	1783.91
<input type="checkbox"/>	2023-03-06 20:31:00	102567122	Bet	2023-03-06 17:59:00	Martigues	Red Star	French National	Match Odds	The Draw	Back	false	3.15	110	Lost	110	0	2003.91
<input type="checkbox"/>	2023-03-06 18:01:00	102551989	Bet	2023-03-06 15:18:00	Al-Masry	El Geish	Egyptian Premier	Correct Score	1 - 1	Back	false	6.4	56	Lost	56	0	2113.91

Rows per page: 10 1-10 of 65 |< < > >|

Figura 5.6: Página de consulta de datos.

My Movements
Query, filter and search your movements

Colombia

Bet
 Commission
 Deposit

<input type="checkbox"/>	Settled	RefId	Type	Placed	Home	Away	Competi...	Market	SubMar...	Bet type	Triggere...	Average ...	Stake	Status	Debit	Credit	Balance
<input type="checkbox"/>	2023-03-05 00:12:00	1.210591634	Commissio n	--	Envigado	Santa Fe	Colombian Primera A	--	--	--	--	0	0	--	5.59	0	1903.6
<input type="checkbox"/>	2023-02-24 01:28:00	1.210369669	Commissio n	--	Real Cartagena	Boyaca Chico	Colombian Cup	--	--	--	--	0	0	--	3.63	0	1538.14

Rows per page: 10 1-2 of 2

Figura 5.7: Página de importación de datos con filtros aplicados.

4 items selected Delete

<input type="checkbox"/>	Settled	Refid	Type	Placed	Home	Away	Compet...	Market	SubMar...	Bet type	Triggere...	Average ...	Stake	Status	Debit	Credit	Balance
<input checked="" type="checkbox"/>	2023-03-12 16:56:00	103179327	Bet	2023-03-12 14:37:00	Mafra	Academico de Viseu	Portuguese Segunda Liga	Match Odds	The Draw	Back	false	3.3	210	Lost	210	0	904.51
<input checked="" type="checkbox"/>	2023-03-12 13:56:00	103148956	Bet	2023-03-12 09:30:00	Covilha	Benfica B	Portuguese Segunda Liga	Match Odds	The Draw	Back	false	3.55	140	Lost	140	0	1114.51
<input checked="" type="checkbox"/>	2023-03-12 01:00:00	103121252	Bet	2023-03-11 23:04:00	Alvarado	CA Guemes	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	2.8	70	Lost	70	0	1254.51
<input checked="" type="checkbox"/>	2023-03-11 22:56:00	103095057	Bet	2023-03-11 19:47:00	Deportivo Madryn	Deportivo Riestra	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	2.94	70	Lost	70	0	1324.51
<input type="checkbox"/>	2023-03-08 02:04:00	102620110	Bet	2023-03-07 10:55:00	Brasil de Pelotas	Ponte Preta	Brazilian Cup	Match Odds	The Draw	Back	false	3.05	6.91	Lost	6.91	0	1394.51
<input type="checkbox"/>	2023-03-08 02:04:00	102620097	Bet	2023-03-07 10:55:00	Brasil de Pelotas	Ponte Preta	Brazilian Cup	Match Odds	The Draw	Back	false	3.1	323.09	Lost	323.09	0	1401.42
<input type="checkbox"/>	2023-03-07 20:01:00	102667467	Bet	2023-03-07 19:49:00	Altinordu	Goztepe	Turkish 1 Lig	Match Odds	Goztepe	Lay	false	1.18	330	Lost	59.4	0	1724.51
<input type="checkbox"/>	2023-03-06 22:53:00	102584315	Bet	2023-03-06 20:22:00	Defensores de Belgrano	Guillermo Brown	Argentinian Primera B Nacional	Match Odds	The Draw	Back	false	3.05	220	Lost	220	0	1783.91
<input type="checkbox"/>	2023-03-06 20:31:00	102567122	Bet	2023-03-06 17:59:00	Martigues	Red Star	French National	Match Odds	The Draw	Back	false	3.15	110	Lost	110	0	2003.91
<input type="checkbox"/>	2023-03-06 18:01:00	102551989	Bet	2023-03-06 15:18:00	Al-Masry	El Geish	Egyptian Premier	Correct Score	1 - 1	Back	false	6.4	56	Lost	56	0	2113.91

Rows per page: 10 ▾ 1-10 of 65 |< < > >|

Figura 5.8: Página de importación de datos con filas seleccionadas.

5.2.3. Página principal

Una vez que los usuarios podían acceder a la aplicación y realizar la importación de datos, se procedió al desarrollo de la página principal del sitio web. En esta página se muestra un resumen general de la situación del usuario, basado en los datos previamente importados. El resumen abarca tanto datos concretos, como el balance, beneficio o número de apuestas, como también diversas gráficas que permiten visualizar la evolución de algunos valores a lo largo del tiempo. Además, se incluye un componente que permite filtrar los datos en función del periodo de tiempo seleccionado. Todos los componentes presentes en la página principal se presentan inicialmente sombreados con un icono indicando que están cargando los datos. En paralelo se realizan las llamadas necesarias al servidor, si estas fallan se mantiene el sombreado y se muestra un mensaje de error (figura 5.10), mientras que si funcionan correctamente se quita el sombreado y se muestran los datos. Se puede observar el resultado de la implementación en la figura 5.9.

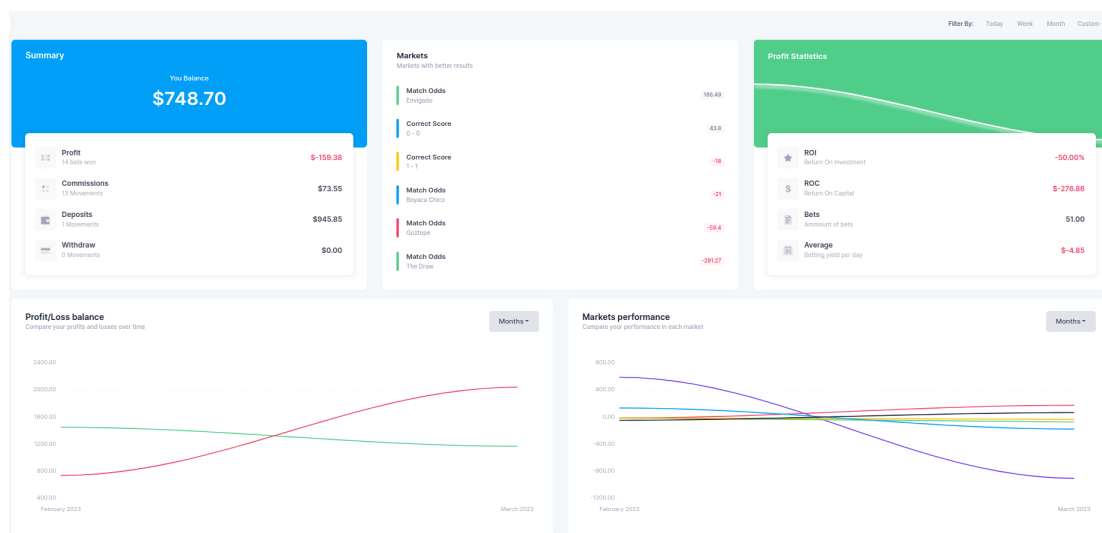


Figura 5.9: Página principal del proyecto.

En esta página encontramos seis componentes principales, siendo estos *Summary*, *Markets*, *Profit statistics*, *Profit/Loss balance*, *Markets performance* y el filtro de periodo.

El componente *Summary* muestra en primer lugar el balance del último movimiento realizado en el periodo seleccionado. En la parte inferior se muestra el beneficio y el número de apuestas, seguido de la suma y cantidad de comisiones, ingresos y retiradas. Este componente obtiene los datos utilizando el *endpoint* personalizado `/get-summary`. Este componente se puede ver en la figura 5.11.

A este primer componente le sigue el componente *Markets*, que consiste en una lista de los seis mercados en los que el usuario ha obtenido el mejor rendimiento durante el periodo seleccionado. Estos mercados se encuentran ordenados y se muestra el nombre del submercado seguido de un subtítulo que indica el mercado al que pertenece, a la derecha del nombre se muestra el beneficio obtenido en dicho mercado. El componente obtiene sus datos llamando al *endpoint* `/get-best-subMarkets`. Se puede observar el componente en la figura 5.12.

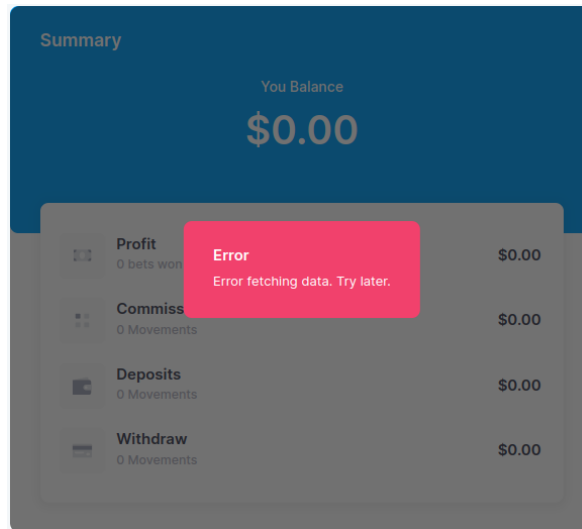


Figura 5.10: Fallo cargando datos en un componente.

Para finalizar este primer grupo de componentes presentes en esta página, contamos con el componente *Profit statistics*. Este componente sigue el estilo del componente *Summary*, pero en lugar de mostrar el balance, presenta una gráfica en la que se aprecia la evolución del balance durante el periodo de tiempo seleccionado.

En la parte inferior del componente se muestran diversos datos relevantes para comprender los beneficios o pérdidas que está experimentando el usuario. Estos datos son el ROC (*Return On Capital*), ROI (*Return On Investment*), el número de apuestas y el rendimiento diario por apuesta. Estos datos son obtenidos a través del *endpoint* `/get-profit-stats`. La figura 5.13 muestra el componente en cuestión.

En la sección inferior de la página se pueden ver dos tablas. Ambas tablas muestran una cantidad de dinero en el eje Y y unidades de tiempo en el eje X. La selección de las unidades de tiempo se puede modificar utilizando un selector ubicado en la parte superior izquierda del componente, lo que permite visualizar desde horas hasta años. La primera tabla muestra la evolución de los beneficios, representados por una línea verde, y las pérdidas, indicadas por una línea roja, a lo largo del tiempo. Por otro lado, la segunda tabla muestra la evolución de los beneficios de los seis mercados con mejor rendimiento. Los componentes llaman a los *endpoints* `/get-markets-chart-data` y `/get-balance-chart-data` respectivamente.

Ambas tablas se encuentran en las figuras 5.14 y 5.15.

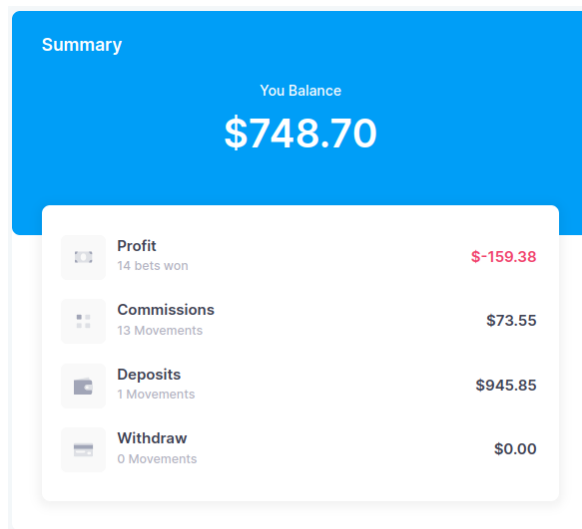


Figura 5.11: Componente *Summary* de la página principal.

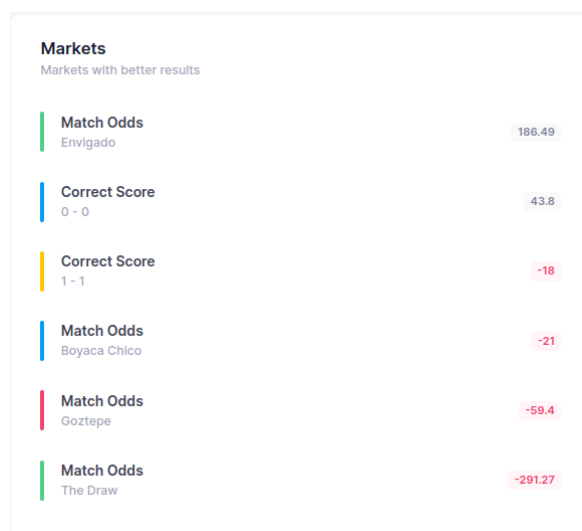


Figura 5.12: Componente *Markets* de la página principal.

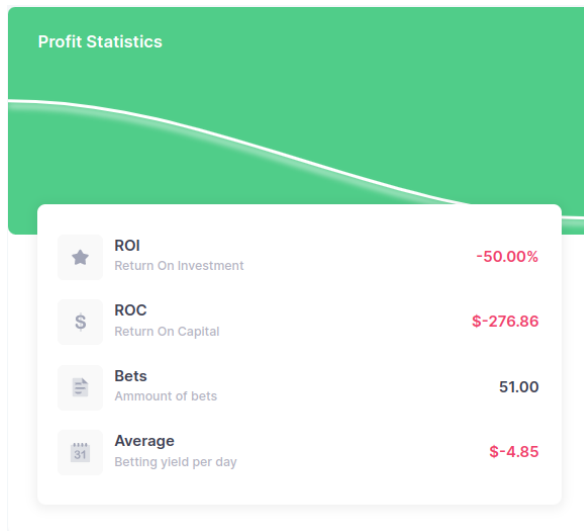


Figura 5.13: Componente *Profit statistics* de la página principal.



Figura 5.14: Componente *Profit/Loss balance* de la página principal.

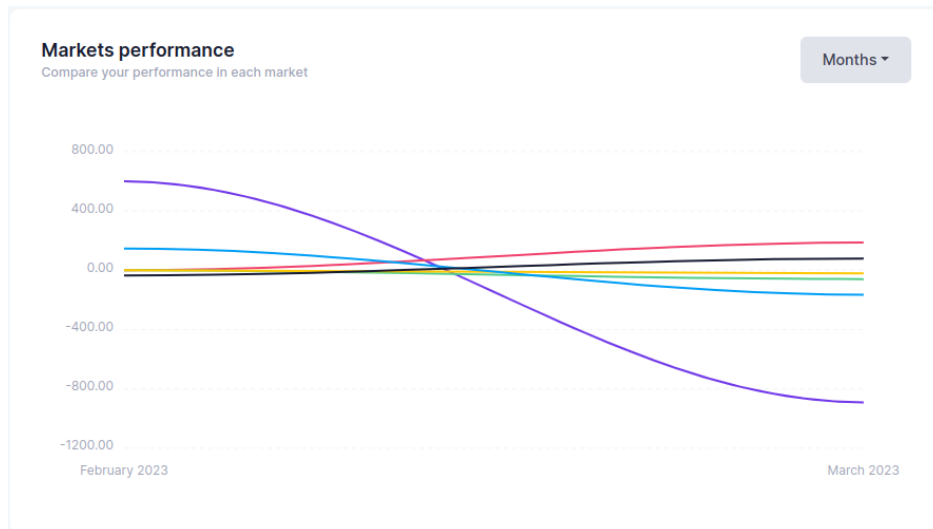


Figura 5.15: Componente *Markets performance* de la página principal.

Todos los elementos mencionados anteriormente desempeñan un papel fundamental en el filtrado de datos, en base al período seleccionado. La selección de dicho período se realiza mediante el componente que se muestra en la figura 5.16. Este componente ofrece varias opciones: *Today* (Hoy), *Week* (Semana), *Month* (Mes) y *Custom* (Personalizado).

Las tres primeras opciones aplican períodos en relación al día actual. En primer lugar, la opción *Today* muestra los datos de las últimas 24 horas. En segundo lugar, la opción *Week* muestra los datos de los últimos 7 días. Y en tercer lugar, la opción *Month* muestra los datos de los últimos 30 días.

Por otro lado, contamos con la opción *Custom* que permite al usuario seleccionar un período personalizado. Al seleccionar esta opción, se muestran los campos de entrada de fecha presentes en la figura 5.17. Estos campos permiten al usuario elegir libremente el rango de fechas que desea utilizar como filtro.

Cabe destacar que todos estos elementos se activan automáticamente al ser seleccionados y se desactivan al seleccionar otro período o incluso al seleccionar el mismo período una vez que ya está activo. Para proporcionar estos datos a los otros componentes se han creado dos *React hooks* en el componente que contiene todos los presentes en la página principal para almacenar la fecha inicial y final del período seleccionado. El componente encargado de filtrar actualiza estos valores mientras que los demás los reciben y actualizan sus datos en caso de que estos cambien.

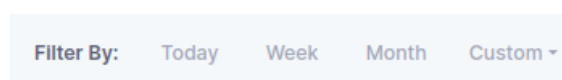


Figura 5.16: Componente de filtrado presente en la página principal.

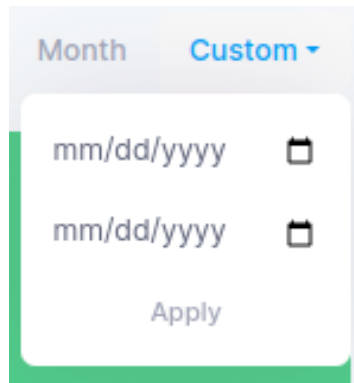


Figura 5.17: Campo de entrada de fechas presente en el filtro.

5.2.4. Página de suscripción

Una vez obtenido el producto mínimo viable, se desarrolló la página de suscripción. Esta página, presenta las suscripciones disponibles en nuestra aplicación, tanto en versiones mensuales como anuales. Cada una de ellas viene acompañada de una descripción detallada de los beneficios que ofrece y su respectivo precio.

Cuando un usuario elige un plan de suscripción, se le muestra un botón que le permite proceder al pago correspondiente. En el caso de las suscripciones gratuitas, simplemente se actualiza la información de suscripción del usuario en nuestra base de datos. Sin embargo, en el caso de las suscripciones de pago, se muestra un botón de PayPal que dirige al usuario a una página de pago donde se realiza la transacción asociada al plan elegido. Una vez completada la suscripción, se actualiza automáticamente el plan de suscripción del usuario en nuestra base de datos.

La página de suscripción utiliza la misma técnica para mostrar la carga datos que los componentes presentes en la página principal. Mientras llama a la base de datos para obtener el plan de suscripción del usuario registrado, se sombrea el componente principal de la página y muestra un icono indicando que está cargando. Una vez cargado el plan de suscripción, se muestra la página dando acceso a la misma. En caso de fallar muestra el mensaje de error visto en la figura 5.10.

Sin embargo, falta implementar tanto la redirección a la página de pago real (actualmente redirige a una página de pago de pruebas ofrecida por PayPal) como la verificación de pago mensual para asegurar que las suscripciones se mantengan activas a lo largo del tiempo. También se busca desarrollar un sistema que permita a los usuarios cambiar de plan de forma automática, sin necesidad de desactivar manualmente su suscripción actual.

La página de suscripción y los botones de pago de PayPal se puede ver en las figuras 5.18 y 5.19.

The screenshot shows a subscription selection interface. At the top left, the heading "Choose a Plan" is followed by the instruction "Select the plan that best suits your needs". In the top right corner, there are two tabs: "Monthly" (active) and "Annual".

Three plan options are listed on the left:

- Free plan**: "Best for testing", priced at \$0/month. This plan is highlighted with a blue background and a green dot.
- Premium plan**: "Best for regular users", priced at \$10/month. It is marked as "Most popular" with a green tag.
- Pro plan**: "Best for advanced users", priced at \$50/month.

Each plan has a corresponding radio button. A green "Subscribe" button is located at the bottom center of the plan selection area.

On the right side, a box titled "What's in Free plan?" lists features with green checkmarks:

- Optimal to test the application
- Up to 100 Movements
- Dashboard Module
- Data Module
- Analysis Module

A red asterisk note at the bottom right states: "* if you want to upgrade or downgrade your payment plan, make sure that you are unsubscribed from your previous plan. [More Info.](#)"

Figura 5.18: Página de suscripción del proyecto.



Figura 5.19: Botones de pago de Paypal.

5.2.5. Páginas de análisis

Por último, se implementaron tres páginas que utilizan los mismos componentes para filtrar datos específicos de distintos grupos de apuestas: competiciones, mercados y tipo de apuesta. Al ser las tres páginas idénticas, se va a emplear la página de análisis de mercados para explicar el funcionamiento de las misma.

Esta página tiene un selector que permite filtrar los datos por mercado seleccionado. Una vez seleccionado el mercado específico, se muestra el balance de las apuestas realizadas en ese mercado, las comisiones en caso de ser posible obtener estos datos y una tabla similar a la vista en la página principal en la que se ve la evolución de los beneficios y pérdidas en este mercado. Estos datos se pueden filtrar por periodo utilizando el componente *Period* con dos campos de entrada de fecha. Todo lo mencionado se puede observar en la figura 5.20.

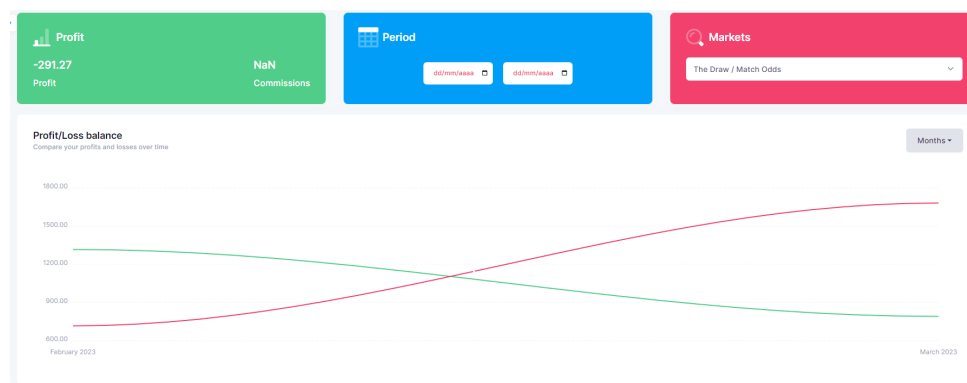


Figura 5.20: Página de análisis de mercados del proyecto.

Mientras que el componente que permite seleccionar mercado emplea los servicios ofrecidos por PHP-CRUD-API, tanto el componente que muestra el beneficio como la tabla utilizan los *endpoints* personalizados `/get-analysis-profit` y `/get-analysis-chart`.

5.3. Pruebas e implementación

5.3.1. Pruebas unitarias y de la API

Como se indica al principio del documento, no se han realizado ni pruebas de integración ni de aceptación dado que el grueso del desarrollo ha sido la vista y que lo único que realiza el servidor es comunicarse con la base de datos. Ante esta situación se ha empleado la herramienta Postman [14] para probar de forma manual los *endpoints* personalizados y algunos de los que ofrece PHP-CRUD-API, verificando tanto que los datos son correctos como que el formato de los mismos es el deseado. En la figura 5.21 se puede ver el logotipo de Postman, mientras que en la figura 5.22 se puede ver un ejemplo de llamada a la api utilizándolo.



Figura 5.21: Logotipo de Postman

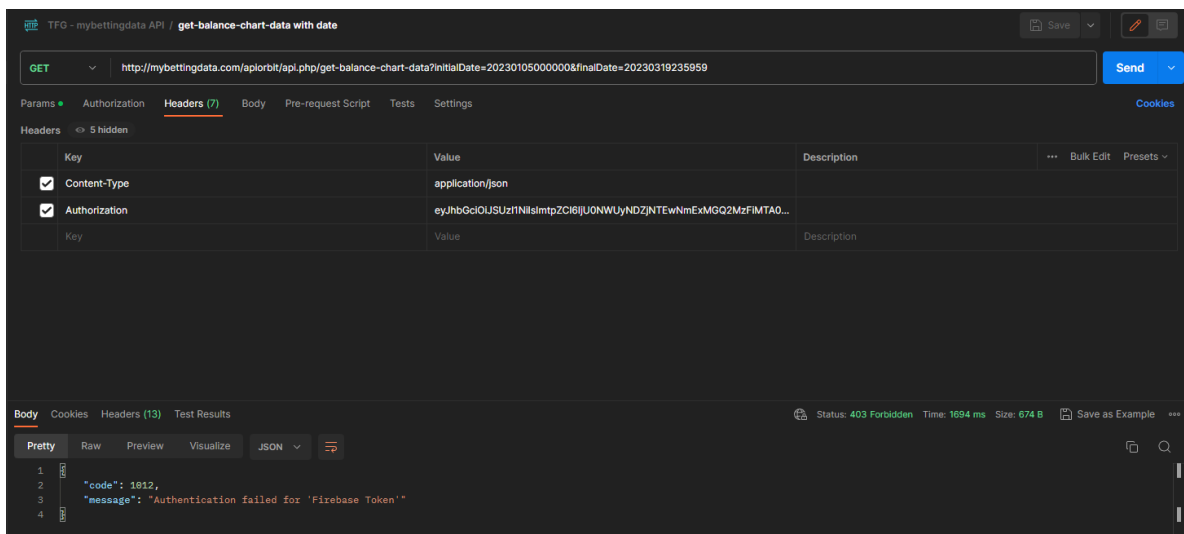


Figura 5.22: Llamada con token inválido al servidor utilizando Postman

Por otra parte, para la función de transformación de datos presente en el módulo de importación de datos, llamada *DataAdapter*, se han realizado distintas pruebas unitarias para verificar su correcto funcionamiento. Este módulo es el único de la vista con pruebas unitarias debido a su complejidad y su importancia para el correcto funcionamiento del sistema, ya que sin el mismo los usuarios no podrían ingresar sus datos y en consecuencia la aplicación quedaría inservible. En estas pruebas se pasan distintos textos para formatear, en el caso de las pruebas válidas se pasan con el formato que debería de utilizar el usuario, mientras que para las inválidas se modifica el formato de algunos campos como las fechas o directamente se borran columnas. Para estas pruebas se ha empleado la herramienta JestJs [9]. Esta permite realizar pruebas en cualquier entorno de desarrollo en el que se utilice JavaScript. Un ejemplo de estas pruebas se pueden ver en el fragmento de código 5.3.

```
1 import { DataAdapter } from '../app/modules/data/core/DataAdapter'
2
3 const FormatError = new Error('Invalid input format');
4
5 describe('DataAdapter Unit Tests', () => {
6   describe('Valid Data', () => {
7     test('Prueba unitaria 1: Formateas una linea de apuesta', async () => {
8       var value: string = "29-03 2023 03:10 104770852 29-03 2023 02:29
9         Colombian Primera A / Boyaca Chico v La Equidad / Match Odds / Boyaca Chico\
nLay\n1.07 300.00 Lost 21.00 0.00 748.70";
10      var result = DataAdapter(value);
```

```

10     expect(result).toEqual([
11         "settled": "2023-03-29 03:10",
12         "refId": "104770852",
13         "type": "Bet",
14         "placed": "2023-03-29 02:29",
15         "home": "Boyaca Chico",
16         "away": "La Equidad",
17         "competition": "Colombian Primera A",
18         "market": "Match Odds",
19         "subMarket": "Boyaca Chico",
20         "cashOut": "false",
21         "betType": "Lay",
22         "avgOdds": 1.07,
23         "stake": 300.00,
24         "status": "Lost",
25         "debit": 21.00,
26         "credit": 0.00,
27         "balance": 748.70
28     ]]);
29 });
30 });
31
32 describe('Invalid Data', () => {
33     test('Prueba unitaria 1: Formateas una linea de apuesta sin el tipo de
34     apuesta', async () => {
35         var value: string = "29-03 2023 03:10 104770852 2023-03-29 02:29
36     Colombian Primera A / Boyaca Chico v La Equidad / Match Odds / Boyaca Chico\
37     n1.07 300.00 Lost 21.00 0.00 748.70";
38         expect(() => {DataAdapter(value)}).toThrowError(FormatError);
39     });
40 });
41 });

```

Listing 5.3: Fragmento de las pruebas unitarias para la función de transformación de datos.

5.3.2. Descripción técnica del despliegue

Durante la finalización del último sprint, se desplegó una versión de demostración del producto para permitir el acceso a un grupo reducido de usuarios con el fin de obtener retroalimentación de los mismos y localizar posibles fallos en la aplicación.

Para realizar el despliegue únicamente se tuvo que configurar el proyecto indicando la dirección de la web en la que se iba a alojar y compilar el proyecto (<https://mybettingdata.com/dashboard>). El proyecto compilado se subió al servidor web proporcionado por Hostinger y se configuró el mismo utilizando el archivo `.htaccess` para lanzar la página cuando se accediera a la ruta correspondiente. Esta configuración se puede ver en el fragmento de código 5.4.

```

1 Options -MultiViews
2 RewriteEngine On
3 RewriteCond %{REQUEST_FILENAME} !-f
4 RewriteRule ^ index.html [QSA,L]

```

Listing 5.4: Archivo de configuración `.htaccess`.

Una vez lanzada la aplicación, se procedió a corregir varios errores relacionados con el formato de los datos provenientes del servidor. Estos errores no se solucionaron antes del despliegue de la aplicación debido a la falta de pruebas previas de las llamadas al servidor desde la aplicación web, ocasionada por un problema de CORS (Cross-Origin Resource Sharing).

Capítulo 6

Conclusiones

6.1. Resultados del proyecto

Una vez finalizado el proyecto, se ha obtenido una versión de demostración del producto. Esta versión, como se indica en la introducción de este documento, se ofrecerá a distintos apostadores deportivos profesionales para obtener retroalimentación tras su uso del producto. Con esta información, se busca tanto corregir fallos, como añadir funcionalidades. Gracias a las herramientas utilizadas y la sencillez buscada al diseñar la experiencia de usuarios, el producto es agradable a nivel visual y evita que el usuario se pueda confundir utilizándolo.

Por otra parte, las consultas a la base de datos presentes en el servidor se han optimizado lo máximo posible, obteniendo de cada consulta el máximo número de datos posible. De esta manera, se evita la necesidad de realizar consultas múltiples, reduciendo así el tiempo de espera causado por la latencia al conectar con el servidor de la base de datos.

A pesar de los avances y la satisfacción general con el proyecto se han quedado por implementar algunas historias de usuario de baja prioridad y quedan algunas cuestiones, como añadir las suscripciones reales a la sección de pagos, por finalizar. Sin embargo se ha obtenido una versión funcional del producto, cumpliendo en gran medida los objetivos del proyecto.

6.2. Posibles ampliaciones del proyecto

En caso de continuar con el proyecto, se priorizaría la corrección de los elementos existentes tras recibir la retroalimentación por parte de los usuarios de la versión de demostración. Una vez corregidos los posibles errores que nos indiquen estos usuarios, se lanzará una versión final del producto y se tendrán en cuenta sus propuestas para mejorar el servicio en futuras versiones del mismo.

Por otra parte contamos con las historias de usuario que no se finalizaron, estas se pueden ver en la siguiente lista.

- **HU05** - Como usuario registrado quiero consultar mis datos personales para administrarlos.
- **HU06** - Como usuario registrado quiero modificar mis datos personales para actualizarlos en la plataforma.
- **HU10** - Como usuario registrado quiero modificar datos introducidos previamente en la aplicación para corregir fallos.
- **HU16** - Como usuario quiero poder crear paneles de control personalizados para consultar la información más relevante para mi situación.
- **HU17** - Como usuario quiero poder exportar mis datos para poder utilizarlos fuera de la aplicación.

Para finalizar, se crearían historias de usuario para llevar a cabo las ampliaciones propuestas por los usuario de la versión de demostración, y junto a las historias pendientes se volvería a priorizar la pila del producto. A su vez, se dividiría la historia HU16 en diversas historias de menor tamaño convirtiendo esta en una épica. Con esta nueva pila del producto se iniciaría un nuevo periodo de desarrollo para ampliar la versión actual del producto, teniendo en cuenta la corrección de los posibles fallos que aparecieran tras su despliegue final.

6.3. Conclusiones personales

Este proyecto me ha ayudado a crecer tanto de forma profesional como personal, permitiéndome trabajar en un entorno real con cierto grado de responsabilidad y poder de decisión, además de aprender la importancia de las relaciones interpersonales en el entorno de laboral. Al trabajar en un entorno con más desarrolladores he aprendido a ayudar y pedir ayuda a mis compañeros de oficina (a pesar de no trabajar en el mismo proyecto). Por otro lado, he descubierto que soy capaz de buscar información y formarme por mi cuenta utilizando como base los conocimientos técnicos obtenidos durante el grado. Esto me ha ayudado a enfrentarme sin problemas al reto que supone empezar a trabajar con nuevas tecnologías desde cero.

En lo relacionado con conocimientos técnicos he aprendido a desarrollar aplicaciones web utilizando ReactJs y he obtenido conocimientos básicos de HTML y CSS. También he realizado por primera vez un despliegue real y he visto todo lo que esto implica y los fallos que pueden surgir durante el mismo. A su vez he afianzado los conocimientos en diseño de base de datos obtenidos durante el grado, diseñando una base de datos siguiendo los requisitos y adaptando la misma a las necesidades de la empresa posteriormente.

He pasado tres meses trabajando con la metodología ágil scrum, utilizando tableros kanban, historias de usuario y tareas. Esto me ha permitido entender como trabajar en un equipo ágil y la importancia de las reuniones en los mismos. A su vez, he conocido a mis compañeros de oficina y me he apoyado en ellos cuando lo he necesitado. Ellos en ocasiones también se han apoyado en mi, dejándome ver que aun estando de prácticas soy un desarrollador más. Con esto he visto lo necesario que son las habilidades sociales en un desarrollador y que muchas veces es mejor ser capaz de trabajar en equipo que poseer mayor conocimiento técnico. Por otra parte

he visto lo importante que es reconocer tus errores y aceptar las críticas. Ser capaz de aceptar que el trabajo que has realizado tiene fallos o no es lo que necesita el proyecto es vital para obtener el resultado posible en un proyecto.

En conclusión con este proyecto he comprendido como funciona un desarrollo real, he podido utilizar los conocimientos adquiridos durante el grado para desarrollar el proyecto y he sido consciente de la importancia de la interacción con otras personas a la hora de desarrollar un proyecto de forma satisfactoria.

Bibliografía

- [1] Deyimar A. Qué es react: definición, características y funcionamiento - hostinger. <https://www.hostinger.es/tutoriales/que-es-react>. [Consulta: 31 de Mayo de 2023].
- [2] Deyimar A. ¿qué es bootstrap? - una guía para principiantes - hostinger. <https://www.hostinger.es/tutoriales/que-es-bootstrap>. [Consulta: 31 de Mayo de 2023].
- [3] John Betancur. React data table component. <https://react-data-table-component.netlify.app>. [Consulta: 17 de Junio de 2023].
- [4] CloudAppi. Patrones de diseño: Composición en react. <https://cloudappi.net/patrones-de-diseno-composicion-en-react/>. [Consulta: 17 de Junio de 2023].
- [5] Gerardo Fernández. React hooks and the “observer” pattern. <https://plainenglish.io/blog/react-hooks-and-the-observer-pattern-1e4274f0e5f5>. [Consulta: 17 de Junio de 2023].
- [6] firebase.google.com. Firebase authentication - firebase. <https://firebase.google.com/products/auth>. [Consulta: 31 de Mayo de 2023].
- [7] GitHub. Hello world - github docs. <https://docs.github.com/es/get-started/quickstart/hello-world>. [Consulta: 6 de Junio de 2023].
- [8] The PHP Group. ¿qué es php? - php. <https://www.php.net/manual/es/intro-what-is.php>. [Consulta: 31 de Mayo de 2023].
- [9] JestJs. Jestjs home page. <https://jestjs.io>. [Consulta: 22 de Junio de 2023].
- [10] Keentthemes. Metronic - keentthemes. <https://keentthemes.com/metronic>. [Consulta: 31 de Mayo de 2023].
- [11] Microsoft. Visual studio - microsoft. <https://visualstudio.microsoft.com/es/>. [Consulta: 5 de Junio de 2023].
- [12] oracle.com. What is mysql? - oracle. <https://www.oracle.com/mysql/what-is-mysql/>. [Consulta: 31 de Mayo de 2023].
- [13] PayPal. Paypal subscriptions - paypal developer documentation. <https://developer.paypal.com/docs/subscriptions/>. [Consulta: 5 de Junio de 2023].
- [14] Postman. What is postman? <https://www.postman.com/product/what-is-postman/>. [Consulta: 22 de Junio de 2023].

- [15] proyectosagiles.org. Qué es scrum. <https://proyectosagiles.org/que-es-scrum/>. [Consulta: 22 de Junio de 2023].
- [16] React. Composición vs. herencia. <https://es.legacy.reactjs.org/docs/composition-vs-inheritance.html>. [Consulta: 17 de Junio de 2023].
- [17] refactoring.guru. Composite. <https://refactoring.guru/design-patterns/composite>. [Consulta: 17 de Junio de 2023].
- [18] refactoring.guru. Observer. <https://refactoring.guru/es/design-patterns/observer>. [Consulta: 17 de Junio de 2023].
- [19] Taiga.io. Taiga. <https://taiga.io/es>. [Consulta: 6 de Junio de 2023].
- [20] talent.com. Salario en españa 2023. <https://es.talent.com/salary>. [Consulta: 10 de Junio de 2023].
- [21] typescriptlang.org. Why create typescript - typescript. <https://www.typescriptlang.org/why-create-typescript>. [Consulta: 31 de Mayo de 2023].
- [22] Maurits van der Schee. Php-crud-api - github. <https://github.com/mevdschee/php-crud-api>. [Consulta: 31 de Mayo de 2023].