



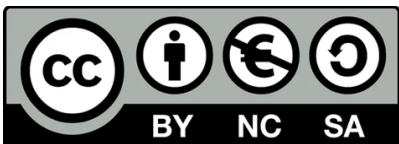
BlackTradingBot – Aplicación para la gestión de activos

Edgar Doutón Parra

Grado de Ingeniería Informática

Consultor: Antoni Oller Arcas

19-01-2024



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-CompartirIgual [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-sa/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	BlackTradingBot – Aplicación para la gestión de activos
Nombre del autor:	Edgar Doutón Parra
Nombre del consultor:	Antoni Oller Arcas
Fecha de entrega (mm/aaaa):	01/2024
Área del Trabajo Final:	Java EE
Titulación:	<i>Grado Ingeniería informática</i>

Resumen del Trabajo (máximo 250 palabras):

Este proyecto de trabajo final de grado presenta el desarrollo de una aplicación para la gestión de activos financieros en el sector bursátil, con un enfoque integral en la integración de tecnologías disruptivas como la inteligencia artificial y el social trading. A lo largo de cuatro años de estudios, se ha adquirido un sólido conocimiento para aplicarlo en este proyecto, que busca revolucionar la forma en que los inversores gestionan sus activos.

La plataforma propuesta permitirá a los usuarios invertir en diversos ámbitos financieros, desde acciones y forex hasta materias primas y criptomonedas. Destacando la inclusión de bots entrenados mediante inteligencia artificial generativa, se busca eficientizar la ejecución de operaciones y adaptar estrategias según las interacciones con usuarios y eventos globales.

Además, se abordará la necesidad de organizaciones y fondos privados de contar con una plataforma especializada para la gestión de activos, considerando las particularidades de este segmento en constante evolución. La implementación de foros y comunidades en la plataforma promoverá la colaboración entre brokers, permitiendo el intercambio de experiencias y conocimientos.

La función de social trading ofrecerá a los usuarios la posibilidad de seguir y replicar automáticamente las operaciones de brokers exitosos, aprovechando la sabiduría colectiva de la comunidad financiera. La plataforma brindará a los usuarios herramientas avanzadas para la toma de decisiones informadas y la simulación de estrategias. Este proyecto representa un esfuerzo significativo para crear una plataforma pionera que transforme la forma en que se abordan las inversiones, fusionando tecnología, colaboración y control personalizado.

Abstract (in English, 250 words or less):

This final year project presents the development of an application for financial asset management in the stock market sector, with a comprehensive focus on integrating disruptive technologies such as artificial intelligence and social trading. Over four years of study, a solid knowledge base has been acquired to apply it to this project, aiming to revolutionize how investors manage their assets.

The proposed platform will enable users to invest in various financial domains, ranging from stocks and forex to commodities and cryptocurrencies. Notably, the inclusion of bots trained through generative artificial intelligence seeks to streamline the execution of operations and adapt strategies based on user interactions and global events.

Furthermore, the project addresses the need for organizations and private funds to have a specialized platform for asset management, considering the specificities of this ever-evolving segment. The implementation of forums and communities on the platform will foster collaboration among brokers, facilitating the exchange of experiences and knowledge.

The social trading feature will empower users to automatically follow and replicate successful broker operations, leveraging the collective wisdom of the financial community. With real-time dashboards and a "sandbox" testing environment, the platform will provide users with advanced tools for making informed decisions and simulating strategies without financial risk. This project represents a significant effort to create a pioneering platform that transforms the approach to investments by merging technology, collaboration, and personalized control.

Palabras clave (entre 4 y 8):

Java EE, Spring, IA, Framework, Azure

Índice

1. Introducción	10
1.1. Contexto y justificación del Trabajo	10
1.2. Objetivos del Trabajo	11
1.3. Enfoque y método seguido.....	11
1.4. Planificación del Trabajo	12
1.5. Breve sumario de productos obtenidos	16
1.6. Breve descripción de los otros capítulos de la memoria.....	16
2. Requerimientos, Análisis y Diseño	17
2.1. Modelo de casos de uso	17
2.2. Fichas de caso de uso	20
2.3. Prototipo de pantallas.....	41
2.4. Modelo relacional de la base de datos	62
2.5. Diagrama de clases.....	64
2.6. Diagrama de arquitectura.....	67
3. Implementación	74
3.1. Tecnologías utilizadas.....	74
3.2. Repositorios del proyecto.....	78
3.3. Características de la aplicación	78
4. Conclusiones.....	83
5. Glosario.....	84
6. Bibliografía	86
7. Anexos	87

Lista de figuras

<i>Ilustración 1 - Vista general Instagantt</i>	13
<i>Ilustración 2 - Planificación PEC 1 Instagantt</i>	14
<i>Ilustración 3 - Planificación PEC 2 Instagantt</i>	14
<i>Ilustración 4 - Planificación PEC 3 Instagantt</i>	15
<i>Ilustración 5 - Planificación PEC 4 Instagantt</i>	15
<i>Ilustración 6 - Modelo de caso de uso usuario</i>	17
<i>Ilustración 7 - Modelo de caso de uso organización</i>	17
<i>Ilustración 8 - Modelo de caso de uso administrador</i>	18
<i>Ilustración 9 - Modelo de caso de uso administrador grupo</i>	18
<i>Ilustración 10 - Modelo de caso de uso bróker I</i>	19
<i>Ilustración 11 - Modelo de caso de uso bróker II</i>	19
<i>Ilustración 12 - Prototipado CU-01 Registro de brokers</i>	41
<i>Ilustración 13 - Prototipado CU-02 Registro de organizaciones</i>	41
<i>Ilustración 14 - Prototipado CU-03 Gestión de perfil bróker</i>	42
<i>Ilustración 15 - Prototipado CU-04 Gestión de organización</i>	42
<i>Ilustración 16 - Prototipado CU-05 Gestión de usuarios de la organización</i>	43
<i>Ilustración 17 - Prototipado CU-06 Inicio de sesión seguro</i>	43
<i>Ilustración 18 - Prototipado CU-07 Búsqueda de brokers</i>	44
<i>Ilustración 19 - Prototipado CU-08 Enviar mensaje a bróker</i>	44
<i>Ilustración 20 - Prototipado CU-09 Creación de portafolio</i>	45
<i>Ilustración 21 - Prototipado CU-10 Gestión de portafolio</i>	45
<i>Ilustración 22 - Prototipado CU-11 Listar portafolios</i>	46
<i>Ilustración 23 - Prototipado CU-12 Búsqueda de activos por categoría</i>	46
<i>Ilustración 24 - Prototipado CU-13 Búsqueda de activos por nombre</i>	47
<i>Ilustración 25 - Prototipado CU-14 Seguimiento de activos por nombre</i>	47
<i>Ilustración 26 - Prototipado CU-15 Consultar sentimiento del mercado</i>	48
<i>Ilustración 27 - Prototipado CU-16 Compra de activos</i>	48
<i>Ilustración 28 - Prototipado CU-17 Venta de activos</i>	49
<i>Ilustración 29 - Prototipado CU-18 Ejecución de órdenes avanzadas</i>	49
<i>Ilustración 30 - Prototipado CU-19 Capacidad de recomendación de activos personalizados (IA)</i>	50
<i>Ilustración 31 - Prototipado CU-20 Capacidad de realizar trading automático</i>	50

Ilustración 32 - Prototipado CU-21 Herramientas de análisis técnico	51
Ilustración 33 - Prototipado CU-22 Gestión de impuestos automatizados	51
Ilustración 34 - Prototipado CU-23 Monitoreo y seguimiento de portafolio.....	52
Ilustración 35 - Prototipado CU-24 Notificaciones y alertas personalizadas	52
Ilustración 36 - Prototipado CU-25 Listar portafolios de social trading.....	53
Ilustración 37 - Prototipado CU-26 Seguimiento de portafolio de social trading.....	53
Ilustración 38 - Prototipado CU-27 Copiar portafolio de social trading.....	54
Ilustración 39 - Prototipado CU-28 Desvincular portafolio de social trading.....	54
Ilustración 40 - Prototipado CU-29 Pruebas de estrategias trading en entorno sandbox	55
Ilustración 41 - Prototipado CU-30 Capacidad de exportación sobre los datos	55
Ilustración 42 - Prototipado CU-31 Creación de grupos.....	56
Ilustración 43 - Prototipado CU-32 Gestión de grupos.....	56
Ilustración 44 - Prototipado CU-33 Listar brokers.....	57
Ilustración 45 - Prototipado CU-34 Listar grupos.....	57
Ilustración 46 - Prototipado CU-35 Creación de canales.....	58
Ilustración 47 - Prototipado CU-36 Gestión de canales.....	58
Ilustración 48 - Prototipado CU-37 Listar canales.....	59
Ilustración 49 - Prototipado CU-38 Enviar invitación de grupo	59
Ilustración 50 - Prototipado CU-39 Aceptar invitación de grupo.....	60
Ilustración 51 - Prototipado CU-40 Rechazar invitación de grupo.....	60
Ilustración 52 - Prototipado CU-41 Bloquear usuarios.....	61
Ilustración 53 - Modelo relacional de la base de datos	62
Ilustración 54 - Diagrama de clases.....	64
Ilustración 55 - Diagrama de arquitectura de microservicios	72
Ilustración 56 - Diagrama de arquitectura on-premise	72
Ilustración 57 - Diagrama de arquitectura en Azure Cloud.....	73
Ilustración 58 - Logotipo de spring	74
Ilustración 59 - Logotipo de spring boot	74
Ilustración 60 - Logotipo de MyBatis	75
Ilustración 61 - Logotipo de java melody	75
Ilustración 62 - Logotipo de swagger.....	76
Ilustración 63 - Logotipo de project lombok.....	76
Ilustración 64 - Logotipo de MySql	76

<i>Ilustración 65 - Registro de usuarios Bróker/Organización.....</i>	<i>79</i>
<i>Ilustración 66 - Inicio de sesión.....</i>	<i>79</i>
<i>Ilustración 67 - Creación de chat con otro usuario</i>	<i>79</i>
<i>Ilustración 68 - Visualizar chats</i>	<i>80</i>
<i>Ilustración 69 - Ajustes de la cuenta</i>	<i>80</i>
<i>Ilustración 70 - Visualizar grupos.....</i>	<i>80</i>
<i>Ilustración 71 - Crear grupo.....</i>	<i>81</i>
<i>Ilustración 72 - Añadir usuarios a organización</i>	<i>81</i>
<i>Ilustración 73 - Ajustes de la cuenta</i>	<i>81</i>
<i>Ilustración 74 - Eliminar la cuenta</i>	<i>81</i>
<i>Ilustración 75 - Bloquear usuario.....</i>	<i>82</i>
<i>Ilustración 76 - Confirmación de bloqueo</i>	<i>82</i>

1. Introducción

1.1. Contexto y justificación del Trabajo

En la sociedad actual, que se caracteriza por un progreso tecnológico continuo y cambios rápidos en los mercados financieros globales, la necesidad de una educación económica sólida se ha vuelto más importante que nunca. Sin embargo, a pesar de que vivimos en una época en la que las decisiones financieras tienen un gran impacto en la vida cotidiana, existen serios problemas con la educación económica. La falta de acceso a información oportuna y comprensible, así como la falta de enfoques prácticos y tecnológicos, crea una brecha entre el cada vez más complejo mundo de las finanzas y la capacidad de los individuos para comprender y tomar decisiones informadas.

Este proyecto se centra en el sector financiero y tiene como objetivo crear una plataforma innovadora de gestión de activos bursátiles que admita tecnologías disruptivas como la inteligencia artificial (IA) y el *social trading*. Este proyecto tiene como objetivo llenar el vacío percibido en la sociedad con respecto a la integración de la tecnología en el sector financiero. El objetivo es diseñar, desarrollar y evaluar una plataforma que permita a los usuarios invertir en diversas áreas financieras, incluidas acciones, divisas, materias primas y criptomonedas. La diferencia clave es la integración de robots de entrenamiento de IA generativa que pueden aprender de las interacciones y noticias de los usuarios para ajustar estrategias y optimizar el retorno de la inversión.

El trabajo también se centra en la necesidad de una plataforma profesional de gestión de activos para organizaciones y *brokers*, teniendo en cuenta las necesidades especiales de este sector y su importancia en los mercados financieros. La plataforma propuesta fomenta la interacción entre *brokers* a través de foros y comunidades, creando un espacio para el aprendizaje colaborativo. Se ha introducido una función de *social trading* que permite a los usuarios seguir automáticamente las operaciones de *brokers* exitosos. Se simplifica el control y supervisión de activos mediante *dashboards* que ofrecen información en tiempo real y análisis detallado del rendimiento del capital.

Además, se incluye un entorno de prueba "*sandbox*" donde los *brokers* pueden simular diversas situaciones financieras, permitiendo a los usuarios experimentar y mejorar sus estrategias sin arriesgar ningún capital real. En conclusión, el TFG representa un esfuerzo significativo para crear una plataforma dinámica que integre tecnología, colaboración y control personal en la gestión de activos financieros. El proyecto se centra en las necesidades cambiantes de los *brokers* y las organizaciones en el siglo XXI y tiene la misión de sincronizar varios microservicios y marcos en tiempo real para transformar los enfoques de inversión y proporcionar una visión interesante para el futuro de las finanzas.

1.2. Objetivos del Trabajo

1. Desarrollar una plataforma de gestión de activos bursátiles respaldada por tecnologías disruptivas, como la inteligencia artificial y el social trading, con el propósito de revolucionar la forma en que los inversores gestionan sus activos.
2. Implementar *bots* entrenados mediante inteligencia artificial generativa que no solo faciliten la ejecución eficiente de operaciones, sino que también aprendan de las interacciones con usuarios y noticias mundiales, adaptando así sus estrategias para optimizar el rendimiento del capital.
3. Abordar las necesidades específicas de organizaciones y fondos privados al proporcionar una plataforma especializada para la gestión de activos, ofreciendo herramientas avanzadas para la administración de inversiones y la supervisión de equipos de trabajo.
4. Promover la interacción entre *brokers* mediante la creación de foros y comunidades donde puedan compartir experiencias, estrategias y conocimientos financieros, creando así un ambiente de aprendizaje colaborativo.
5. Desarrollar una función de social trading que permita a los usuarios seguir y replicar automáticamente las operaciones de *brokers* exitosos que comparten sus estrategias, aprovechando la sabiduría colectiva de la comunidad de *brokers*.
6. Permitir la compra y venta de acciones a terceros servicios de forma rápida y sencilla mediante nuestra plataforma centralizada.
7. Establecer un sistema de notificaciones por correo electrónico para informar a los usuarios sobre diferentes acontecimientos sucedidos dentro de la plataforma y acciones que requieren de su revisión.

1.3. Enfoque y método seguido

El plan de trabajo seguido ha sido realizar un producto a partir de cero. Por ello, primero se ha realizado un plan de trabajo donde se han ido estableciendo las fechas para las entregas de los principales hitos del proyecto.

Además, se han definido las bases y los principales requisitos del proyecto. Con ello, se ha realizado un análisis de diseño en base a una combinación de investigación e innovación donde se ha establecido que actores se encuentran dentro de la aplicación, los principales casos de uso, diagramas de clases, modelo relacional, diagrama de arquitectura y para finalizar el prototipado de la página web.

Posteriormente, se ha continuado con la fase de implementación de la aplicación, mediante una forma iterativa, implementado cada caso de uso establecido en la fase de análisis y diseño.

Finalmente, se ha realizado los documentos de la memoria del proyecto y su presentación.

El método seguido para llevar a cabo el proyecto es el modelo en cascada, donde se encuentran establecidas las diferentes fases del proyecto, ejecutándolas en su respectivo orden.

1.4. Planificación del Trabajo

Se procede a presentar una tabla con las fechas de inicio y final sobre los principales hitos que se han llevado a cabo.

	Hitos	Fecha inicio	Fecha final
PEC1	Plan de trabajo	27/09/2023	11/10/2023
PEC2	Análisis y Diseño	12/10/2023	11/11/2023
PEC3	Implementación	12/11/2023	02/01/2024
PEC4	Memoria y Presentación	03/01/2024	19/01/2024

A continuación, se presenta la planificación detallada con un diagrama Gantt. Cabe destacar que las horas de trabajo de cada tarea es estimativo, ya que se ha calculado en jornadas de 8 horas. Como se puede apreciar en el Gantt posterior, se ha valorado incluyendo los fines de semana y festivos, ya que se ha sacado provecho de ellos.

A continuación, se adjunta el Gantt (Se puede consultar de forma on-line en el siguiente enlace: <https://app.instagantt.com/shared/6526c778b679e550ffadfaeb>)

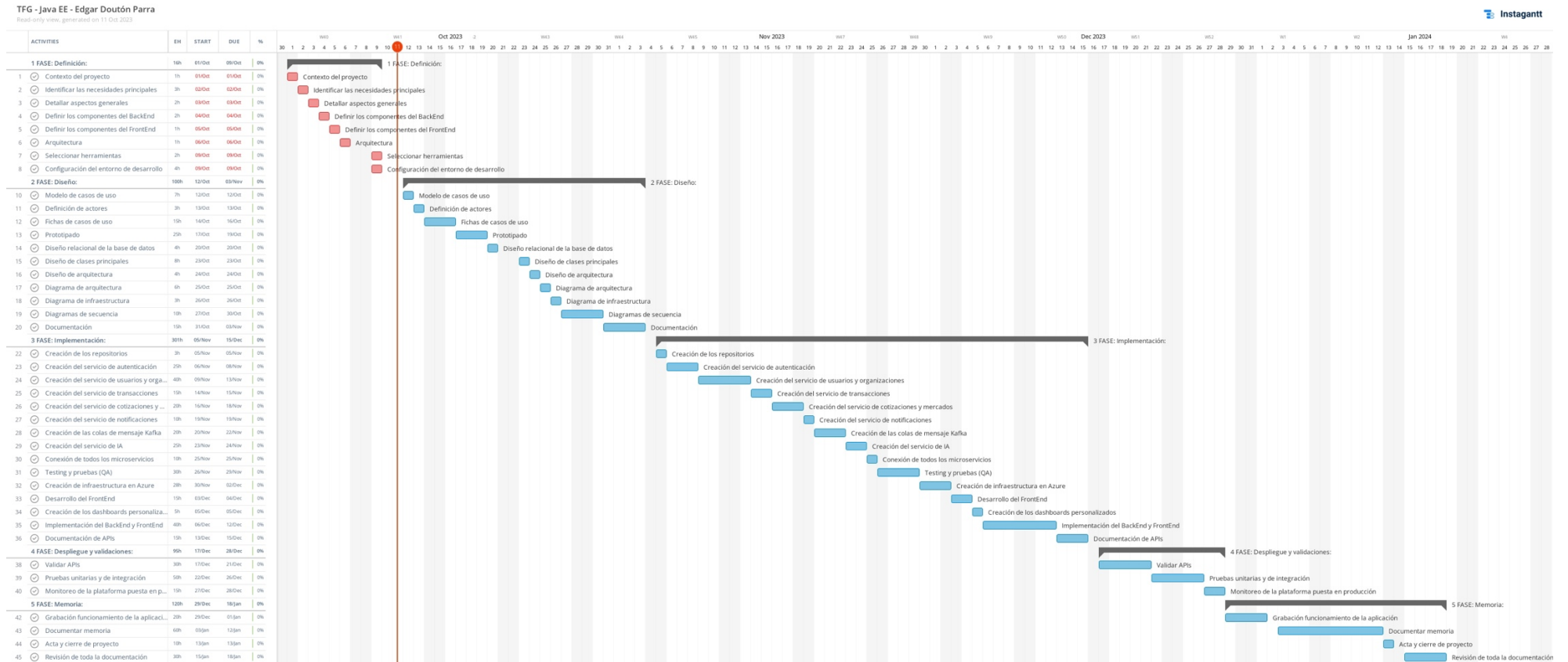


Ilustración 1 - Vista general Instagantt

PEC 1 – Plan de Trabajo

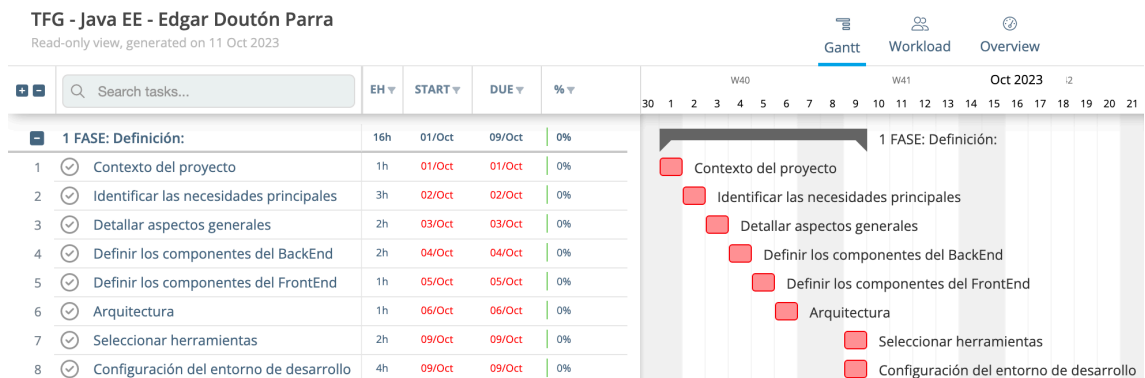


Ilustración 2 - Planificación PEC 1 Instagantt

En esta etapa, se procede a establecer una toma inicial con el proyecto definido y se reflexionará sobre la idea del trabajo, así como cuáles serán las tecnologías necesarias. En resumen, en esta etapa se realizará el contexto y justificación del trabajo, sus objetivos y la planificación inicial.

PEC 2 – Análisis y Diseño

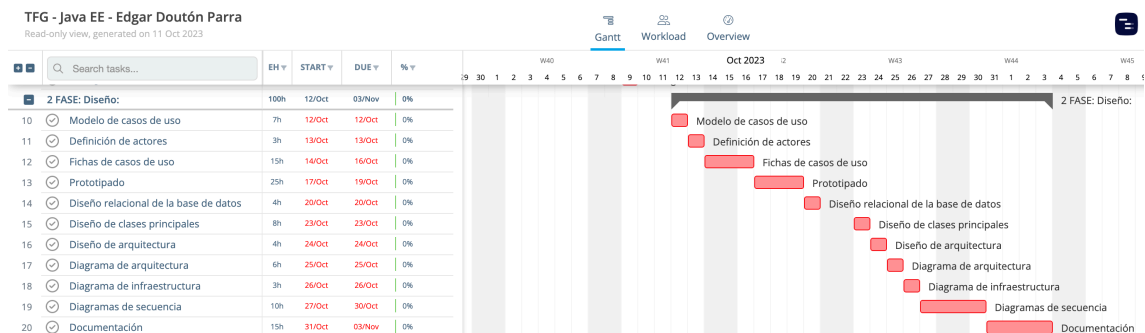


Ilustración 3 - Planificación PEC 2 Instagantt

Esta segunda etapa, se definen los requerimientos del proyecto con los casos de uso, así como las fichas de caso de uso para los más complejos. Una vez tenemos esta parte lista, se procede a elaborar un modelado de pantallas y las referencias con sus casos de uso.

Posteriormente, se procede a la parte de análisis y diseño con el diseño relacional de la base de datos, los diagramas de clases con las clases más importantes, y el diagrama de arquitectura, el cual será un diagrama con los componentes más importantes de nuestro proyecto.

PEC 3 - Implementación



Ilustración 4 - Planificación PEC 3 Instagantt

En esta penúltima etapa, se comenzará con la implementación del proyecto. Principalmente, se implementará más del 80% del proyecto para posteriormente hacerle una entrega parcial al tutor para su validación. Después ya se procede a implementar todo lo restante y realizar las pruebas necesarias.

PEC 4 – Memoria y Presentación

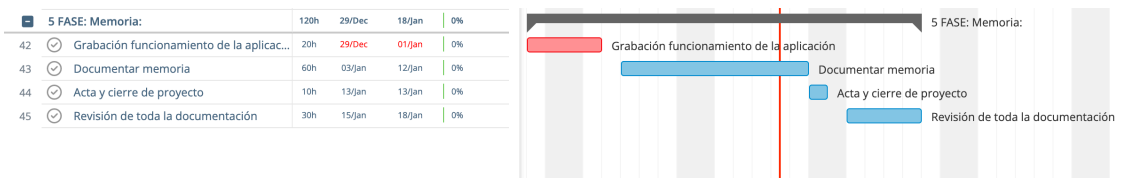


Ilustración 5 - Planificación PEC 4 Instagantt

Finalmente, en esta última etapa, se recopila toda la documentación elaborada en las anteriores etapas para la realización de la memoria. Por otro lado, se hará una presentación virtual consistiendo en un vídeo de la presentación oral del trabajo realizado, así como los resultados obtenidos.

1.5. Breve resumen de productos obtenidos

El proyecto consiste en una aplicación de tipo Single Page Application, una aplicación web para la gestión de diferentes acciones bursátiles como acciones, criptomonedas y materias primas.

La memoria del proyecto se ha realizado, con información de las diferentes fases del proyecto y con un anexo explicando los pasos para la instalación del producto.

El documento de presentación del producto, así como el vídeo demostrativo de la operativa básica de la aplicación.

1.6. Breve descripción de los otros capítulos de la memoria

En la segunda parte del documento, se habla sobre el análisis y diseño del proyecto. En concreto, se muestra el modelo de casos de uso de los diferentes roles de la aplicación, posteriormente se mostrará las fichas de cada caso de uso, adjuntando su debido prototipo de pantalla de la aplicación. Seguiremos con el modelo relacional de la base de datos y finalizamos con los diagramas de clases y de arquitectura.

En la tercera parte, nos centramos en la etapa de implementación donde se explican las diversas tecnologías empujadas en el proyecto, y las diferentes características de la aplicación donde también se muestran diversas capturas de pantalla de la aplicación en funcionamiento. Finalmente, se comentarán varios aspectos pendientes de implementar y posibles mejoras a realizar.

En el cuarto capítulo, se reflexionará sobre las conclusiones obtenidas sobre el proyecto. La quinta parte del documento se mostrará un glosario sobre algunos términos importantes empleados en este documento. En la sexta parte se menciona la bibliografía utilizada en el transcurso del proyecto y finalmente, en el séptimo capítulo se explica los pasos para levantar el proyecto.

2. Análisis y Diseño

2.1. Modelo de casos de uso

Rol - Usuario

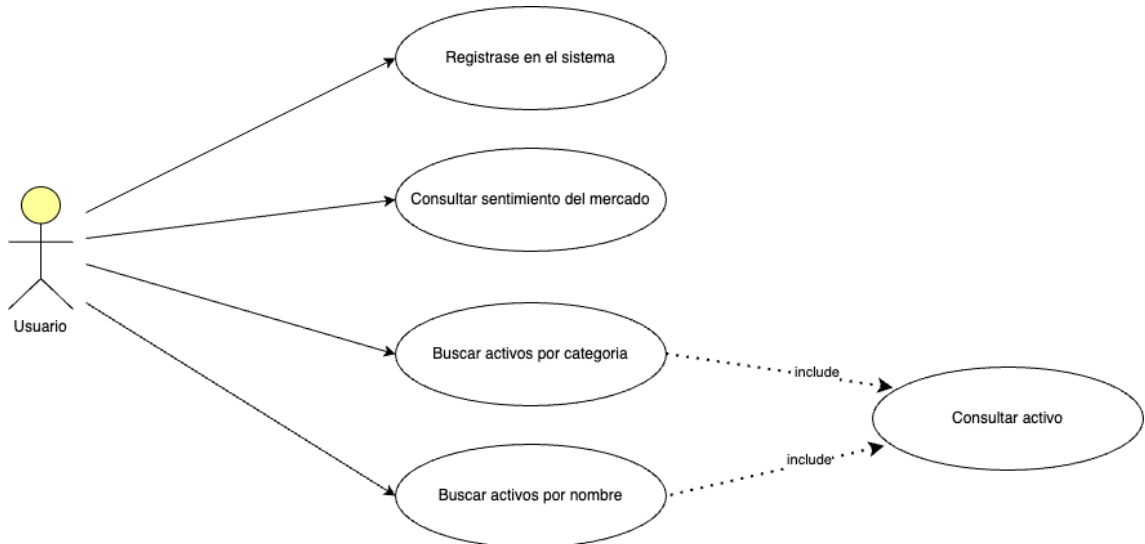


Ilustración 6 - Modelo de caso de uso usuario

Rol - Organización

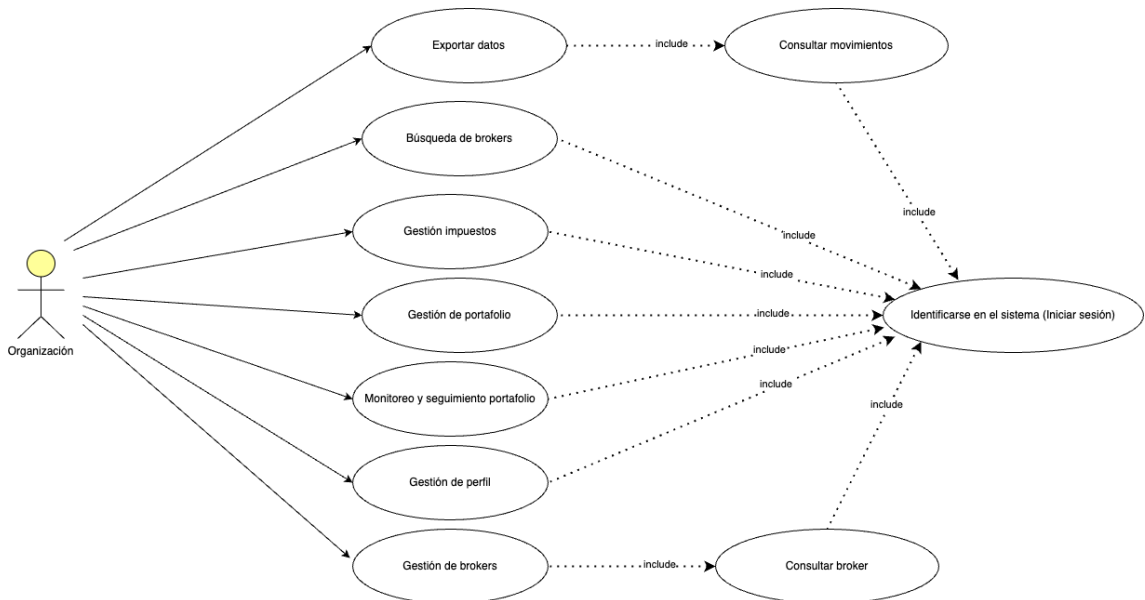


Ilustración 7 - Modelo de caso de uso organización

Rol - Administrador

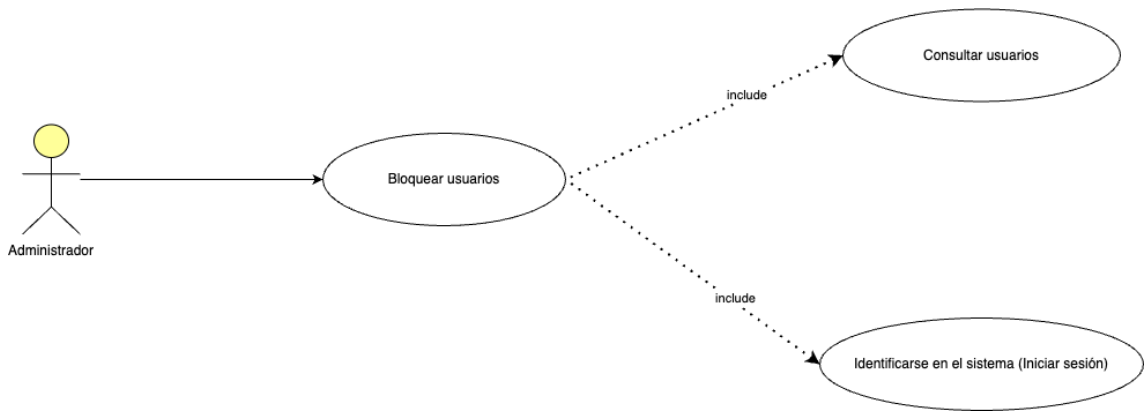


Ilustración 8 - Modelo de caso de uso administrador

Rol – Administrador Grupo

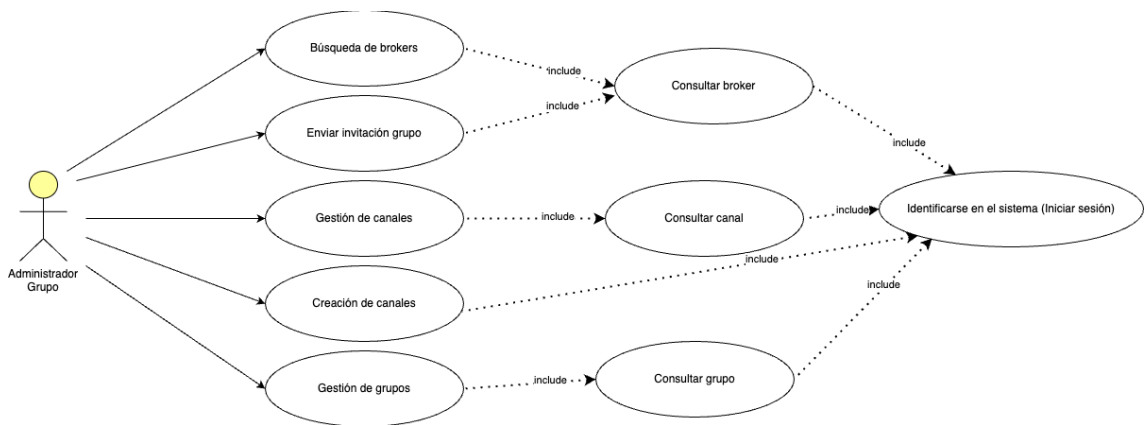


Ilustración 9 - Modelo de caso de uso administrador grupo

Rol – Bróker

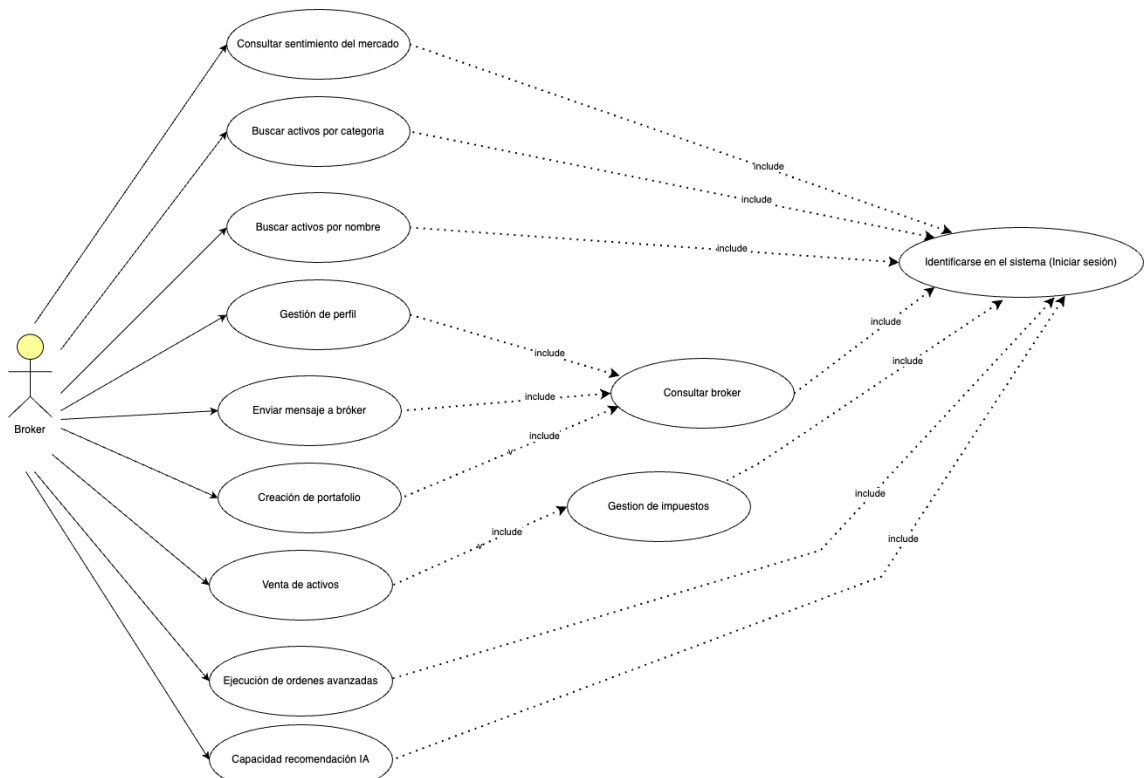


Ilustración 10 - Modelo de caso de uso bróker I

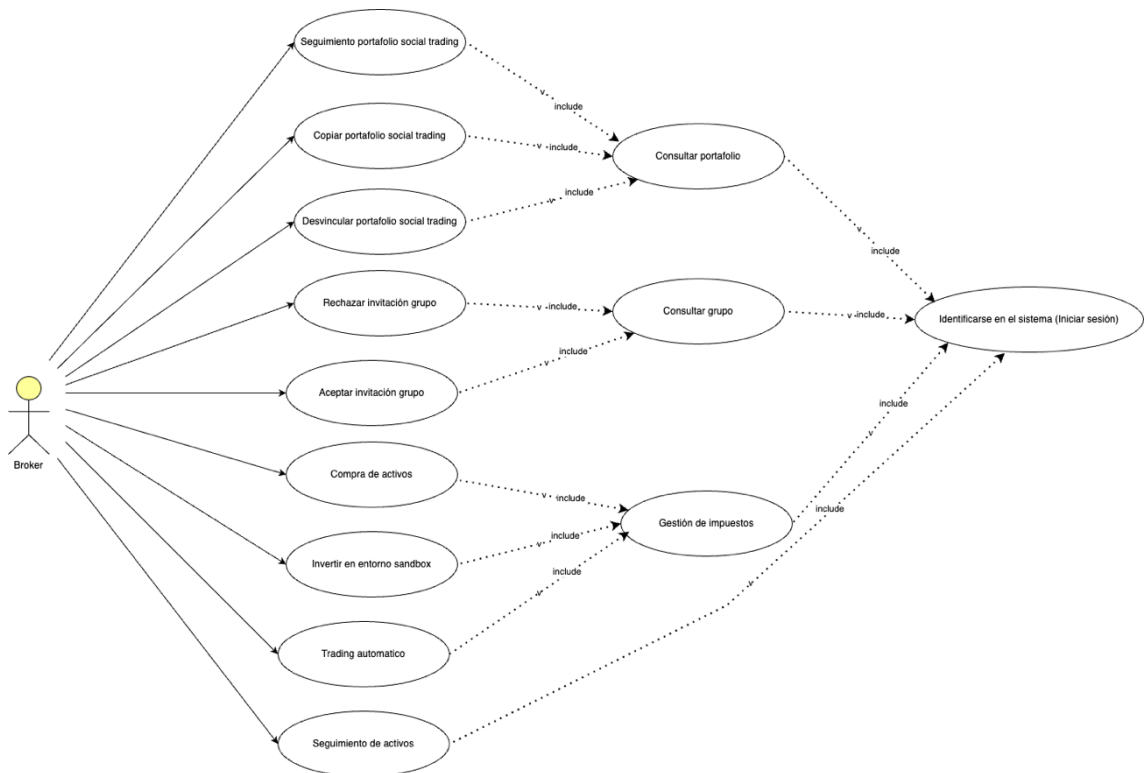


Ilustración 11 - Modelo de caso de uso bróker II

2.2. Fichas de caso de uso

Lista de casos de uso

Código	Caso de Uso	Actor
CU-01	Registro de brokers	Usuario
CU-02	Registro de organizaciones	Usuario
CU-03	Gestión de perfil bróker	Bróker
CU-04	Gestión de organización	Organización
CU-05	Gestión de usuarios de la organización	Organización
CU-06	Inicio de sesión seguro	Bróker, Organización
CU-07	Búsqueda de brokers	Bróker, Organización, Administrador, Administrador Grupo
CU-08	Enviar mensaje a bróker	Bróker
CU-09	Creación de portafolio	Bróker
CU-10	Gestión de portafolio	Bróker
CU-11	Listar portafolios	Bróker
CU-12	Búsqueda de activos por categoría	Usuario, Bróker
CU-13	Búsqueda de activos por nombre	Usuario, Bróker
CU-14	Seguimiento de activos por nombre	Bróker
CU-15	Consultar sentimiento del mercado	Usuario, Bróker
CU-16	Compra de activos	Bróker
CU-17	Venta de activos	Bróker
CU-18	Ejecución de órdenes avanzadas	Bróker
CU-19	Capacidad de recomendación de activos personalizados (IA)	Bróker
CU-20	Capacidad de realizar trading automático	Bróker
CU-21	Herramientas de análisis técnico	Bróker
CU-22	Gestión de impuestos automatizada	Bróker, Organización
CU-23	Monitoreo y seguimiento de portafolio	Bróker, Organización
CU-24	Notificaciones y alertas personalizadas	Bróker
CU-25	Listar portafolios de social trading	Bróker
CU-26	Seguimiento de portafolio de social trading	Bróker
CU-27	Copiar portafolio de social trading	Bróker
CU-28	Desvincular portafolio de social trading	Bróker
CU-29	Pruebas de estrategias trading en entorno sandbox	Bróker
CU-30	Capacidad de exportación sobre los datos	Bróker, Organización
CU-31	Creación de grupos	Bróker
CU-32	Gestión de grupos	Administrador grupo
CU-33	Listar brokers	Bróker
CU-34	Listar grupos	Bróker
CU-35	Creación de canales	Administrador grupo

CU-36	Gestión de canales	Administrador grupo
CU-37	Listar Canales	Bróker, Administrador grupo
CU-38	Enviar invitación de grupo	Administrador grupo
CU-39	Aceptar invitación de grupo	Bróker
CU-40	Rechazar invitación de grupo	Bróker
CU-41	Bloquear usuarios	Administrador

Fichas de caso de uso

CU-01 Registro de brokers	
Actores	Usuario
Breve descripción	En este caso se describe el proceso de registro de brokers en la plataforma de gestión de activos.
Precondición	El usuario puede acceder a la página web y no tiene ninguna cuenta creada.
Postcondición	El bróker queda registrado en la plataforma y puede comenzar a utilizar sus funcionalidades.
Escenario básico	<ol style="list-style-type: none"> 1. El usuario accede a la página web. 2. El usuario accede a la sección de registrarse como bróker. 3. El usuario proporciona la información requerida. 4. El usuario adjunta documentación necesaria para verificar la autenticidad. 5. El usuario pulsa completar. 6. El sistema crea un usuario de tipo bróker.
Escenario alternativo	<p>3a. El usuario no introduce ninguna información y pulsa completar.</p> <p>3a1. El sistema muestra un mensaje de error indicando que no se ha introducido ningún dato.</p> <p>4a. El usuario no realiza la verificación de documentación.</p> <p>4a1. El sistema muestra un mensaje de error indicando que no se ha realizado la verificación.</p>

CU-02 Registro de organizaciones	
Actores	Usuario
Breve descripción	En este caso se describe el proceso de registro de organización en la plataforma de gestión de activos.
Precondición	El usuario puede acceder a la página web y no tiene ninguna cuenta creada.
Postcondición	La organización queda registrada en la plataforma y puede comenzar a utilizar sus funcionalidades.
Escenario básico	<ol style="list-style-type: none"> 1. El usuario accede a la página web. 2. El usuario accede a la sección de registrarse como organización. 3. El usuario proporciona la información requerida.

	<ol style="list-style-type: none"> 4. El usuario adjunta documentación necesaria para verificar la autenticidad. 5. El usuario pulsa completar. <p>El sistema crea un usuario de tipo Organización.</p>
Escenario alternativo	<p>3a. El usuario no introduce ninguna información y pulsa completar.</p> <p>3a1. El sistema muestra un mensaje de error indicando que no se ha introducido ningún dato.</p> <p>4a. El usuario no realiza la verificación de documentación.</p> <p>4a1. El sistema muestra un mensaje de error indicando que no se ha realizado la verificación.</p>

CU-03 Gestión de perfil bróker	
Actores	Bróker
Breve descripción	Este caso de uso describe el proceso de gestión del perfil de un bróker en la plataforma, permitiéndole actualizar y personalizar la información asociada con su cuenta.
Precondición	El bróker ha iniciado sesión en la plataforma y tiene acceso a su perfil.
Postcondición	Los cambios realizados en el perfil del bróker se guardan y reflejan en el sistema.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker inicia sesión en la plataforma. 2. El bróker accede a la sección de gestión de perfil. 3. El bróker visualiza la información actual de su perfil. 4. El bróker modifica la información que desea actualizar. 5. El bróker guarda los cambios realizados. 6. El sistema procesa los cambios.
Escenario alternativo	<p>4a. El bróker no introduce toda la información necesaria para actualizar el perfil.</p> <p>4a1. El sistema muestra un mensaje de error indicando que faltan datos.</p>

CU-04 Gestión de organización	
Actores	Organización
Breve descripción	Este caso de uso describe el proceso de gestión de una organización en la plataforma, permitiendo realizar acciones relacionadas con la supervisión de la entidad.
Precondición	La organización tiene una cuenta activa en la plataforma y ha iniciado sesión.
Postcondición	Los cambios realizados en la gestión de la organización se reflejan.
Escenario básico	<ol style="list-style-type: none"> 1. La organización inicia sesión a la plataforma. 2. La organización accede a la sección de gestión de la organización 3. La organización ve la configuración actual. 4. La organización realiza las acciones de limitación. 5. La organización guarda los cambios.

	6. El sistema procesa los cambios
Escenario alternativo	4a. La organización no introduce toda la información necesaria para actualizar el perfil. 4a1. El sistema muestra un mensaje de error indicando que faltan datos.

CU-05 Gestión de usuarios de la organización	
Actores	Organización
Breve descripción	Este caso de uso describe el proceso de gestión de usuarios asociados a una organización.
Precondición	La organización ha iniciado sesión y tiene acceso a la gestión de usuarios de la organización.
Postcondición	Los cambios realizados en la gestión de usuarios se aplican, afectando el acceso y miembros dentro de la organización.
Escenario básico	<ol style="list-style-type: none"> 1. La organización inicia sesión en la plataforma. 2. La organización accede a la sección de gestión de usuarios. 3. La organización realiza acciones de gestión, como agregar, modificar o eliminar usuarios. 4. El administrador guarda los cambios. 5. El sistema procesa los cambios
Escenario alternativo	3a. La organización intenta eliminar un usuario y este tiene operaciones pendientes o activos asociados. 3a1. El sistema notifica que se tienen que finalizar las operaciones pendientes antes de continuar.

CU-06 Inicio de sesión seguro	
Actores	Bróker, Organización
Breve descripción	Este caso de uso describe el proceso de inicio de sesión seguro para brokers y organizaciones.
Precondición	El bróker u organización tiene credenciales válidas y acceso a la plataforma.
Postcondición	El bróker u organización inicia sesión con éxito y tiene acceso a las funcionalidades correspondientes según su rol.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker u organización accede a la página de inicio de sesión de la plataforma. 2. El sistema presenta los campos para ingresar correo electrónico y contraseña. 3. El bróker u organización ingresa los datos correspondientes. 4. El sistema valida las credenciales y autentica el usuario. 5. El bróker u organización obtiene acceso a sus datos.
Escenario alternativo	3a. El bróker u organización no ha introducido todos los datos. 3a1. El sistema muestra un mensaje de error indicando que faltan datos. 4a. El bróker u organización ha introducido datos erróneos.

	<p>4a1. El sistema muestra un mensaje de error indicando que los datos son inválidos.</p> <p>4b. El bróker u organización ha intentado iniciar sesión sin éxito varias veces.</p> <p>4b1. El sistema bloquea la cuenta y notifica por correo al propietario.</p>
--	--

CU-07		Búsqueda de brokers
Actor	Bróker, Organización, Administrador, Administrado grupo	
Breve descripción	Este caso de uso describe el proceso de búsqueda de brokers por parte de un bróker en la plataforma, permitiéndole encontrarlo y evaluarlo.	
Precondición	El bróker ha iniciado sesión en la plataforma y tiene acceso a las funcionalidades de búsqueda.	
Postcondición	El bróker visualiza la lista de brokers que coinciden con los criterios de búsqueda especificados.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la función de búsqueda brokers. 2. El sistema presenta un campo para ingresar el nombre. 3. El bróker ingresa el nombre de quien desea encontrar. 4. El sistema devuelve una lista de brokers que coinciden con los criterios especificados. 5. El bróker visualiza la información detallada de los brokers, incluyendo la calificación de otros usuarios. 	
Escenario alternativo	<p>3a. El bróker no ha introducido texto.</p> <p>3a1. El sistema muestra un mensaje de error conforme no ha introducido criterios.</p> <p>4a. El sistema no obtiene coincidencias con lo introducido por el bróker.</p> <p>4a1. El sistema retornara una lista vacía con el mensaje de no existe tal usuario.</p>	

CU-08		Enviar mensaje a bróker
Actor	Bróker	
Breve descripción	Este caso de uso describe el proceso de un bróker para enviar mensajes a otros brókers.	
Precondición	El bróker ha iniciado sesión y ha encontrado el usuario mediante la búsqueda.	
Postcondición	El mensaje es enviado al bróker destinatario y se registra en el sistema.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker busca a otro bróker. 2. El sistema retorna los brokers coincidentes. 3. El bróker selecciona al bróker destinatario. 4. El bróker redacta el mensaje y lo envía. 5. El sistema registra la comunicación y notifica al bróker. 6. El bróker destinatario recibe la notificación. 	
Escenario alternativo	4a. El bróker no ha introducido texto.	

	<p>4a1. El sistema muestra un mensaje de error conforme no ha introducido todos los datos.</p> <p>4b. El bróker destinatario tiene la función desactivada o ha bloqueado al remitente.</p> <p>4b1. El sistema mostrará un mensaje al remitente que no ha podido enviar el mensaje.</p>
--	--

CU-09 Creación de portafolio	
Actor	Bróker
Breve descripción	Este caso de uso describe el proceso de un bróker para crear un nuevo portafolio.
Precondición	El bróker ha iniciado sesión y tiene acceso a la funcionalidad de creación de portafolio.
Postcondición	Se crea un nuevo portafolio con los activos seleccionados por el bróker.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la función de creación de portafolios. 2. El sistema presenta un formulario para la creación. 3. El bróker rellena los datos y selecciona los activos. 4. El bróker acepta la creación. 5. El sistema crea el nuevo portafolio.
Escenario alternativo	<p>3a. El bróker no ha introducido texto.</p> <p>3a1. El sistema muestra un mensaje de error conforme no ha rellenado todos los campos.</p> <p>3b. El bróker no ha seleccionado activos.</p> <p>3b1. El sistema muestra un mensaje de error conforme no ha seleccionado activos.</p>

CU-10 Gestión de portafolio	
Actor	Bróker
Breve descripción	Este caso de uso describe el proceso de gestión de un portafolio.
Precondición	El bróker ha iniciado sesión y tiene portafolios creados.
Postcondición	Los cambios realizados se reflejan y afectan al portafolio seleccionado.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la sección de portafolios y selecciona el portafolio a gestionar. 2. El bróker selecciona la opción de editar. 3. El sistema presenta las gestiones posibles mediante formulario. 4. El bróker realiza las acciones de gestión, que pueden incluir la adición o eliminación de activos, ajustes en la distribución o modificación de la estrategia. 5. El bróker guarda los cambios. 6. El sistema guarda los cambios.
Escenario alternativo	4a. El bróker no ha introducido los datos necesarios.

	4a1. El sistema muestra un mensaje de error conforme no ha rellenado todos los campos.
--	--

CU-11 Listar portafolios	
Actor	Bróker
Breve descripción	Este caso de uso describe el proceso de listar portafolios de demás brokers.
Precondición	El bróker ha iniciado sesión en la plataforma y tiene portafolios creados.
Postcondición	Se muestran todos los portafolios.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la sección de portafolios. 2. El bróker selecciona "Ver portafolios". 3. El sistema muestra los portafolios de ese usuario.
Escenario alternativo	<p>2a. El bróker no tiene portafolios.</p> <p>2a1. El sistema muestra un mensaje de error conforme no tiene portafolios.</p>

CU-12 Búsqueda de activos por categoría	
Actor	Bróker, Usuario
Breve descripción	Este caso de uso describe el proceso de búsqueda de activos por categoría por parte de brokers y usuarios.
Precondición	Se encuentra en la página principal y tiene acceso a la funcionalidad de búsqueda.
Postcondición	El actor visualiza una lista de activos que pertenecen a la categoría seleccionada.
Escenario básico	<ol style="list-style-type: none"> 1. El usuario o bróker accede a la función de búsqueda de activos. 2. El sistema presenta opciones para seleccionar diferentes categorías de activos. 3. El usuario o bróker elige la categoría de activos de interés. 4. El sistema devuelve una lista de activos que pertenecen a la categoría seleccionada. 5. El usuario o bróker visualiza la información detallada de los activos.
Escenario alternativo	<p>4a. El sistema no tiene activos disponibles para esa categoría.</p> <p>4a1. El sistema retorna una lista vacía con un mensaje conforme no existen activos.</p>

CU-13		Búsqueda de activos por nombre
Actor	Bróker, Usuario	
Breve descripción	Este caso de uso describe el proceso de búsqueda de activos por nombre por parte de brókers y usuarios.	
Precondición	Se encuentra en la página principal y tiene acceso a la funcionalidad de búsqueda.	
Postcondición	El actor visualiza una lista de activos que pertenecen al nombre introducido.	
Escenario básico	<ol style="list-style-type: none"> 1. El usuario o bróker accede a la función de búsqueda de activos. 2. El sistema presenta un formulario para introducir un texto libre. 3. El usuario o bróker introduce el nombre del activo de interés. 4. El sistema devuelve una lista de activos que coinciden con lo introducido. 5. El usuario o bróker visualiza la información detallada de los activos. 	
Escenario alternativo	<p>4a. El sistema no tiene activos disponibles para ese nombre.</p> <p>4a1. El sistema retorna una lista vacía con un mensaje conforme no existen activos.</p>	

CU-14		Seguimiento de activos por nombre
Actor	Bróker	
Breve descripción	Este caso de uso describe el proceso de seguimiento de activos por nombre por parte de un bróker, permitiéndole recibir notificaciones.	
Precondición	El bróker ha iniciado sesión y tiene acceso al mercado.	
Postcondición	El activo seleccionado queda marcado como favorito y el bróker recibe notificaciones relacionadas con el activo.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede al mercado 2. El bróker busca su activo a consultar. 3. El sistema devuelve el activo. 4. El bróker selecciona el activo deseado. 5. El bróker elige la opción de "Seguir" para el activo seleccionado. 6. El sistema marca el activo como favorito y configura notificaciones relacionadas con cambios en ese activo. 	
Escenario alternativo	<p>5a. El bróker ya sigue a ese activo.</p> <p>5a1. El sistema permitirá desmarcarlo para dejarlo de seguir.</p>	

CU-15 Consultar sentimiento del mercado	
Actor	Bróker
Breve descripción	Este caso de uso describe el proceso de un bróker para consultar el sentimiento del mercado, permitiéndole obtener información sobre el mercado.
Precondición	El bróker ha iniciado sesión y tiene acceso al mercado.
Postcondición	El bróker obtiene información actualizada sobre el sentimiento general del mercado.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede al mercado y elige el activo a valorar. 2. El sistema presenta toda la información del activo y los indicadores del sentimiento general de ese activo, que incluye tendencias. 3. El bróker examina la información proporcionada y utiliza el valor como uno de los factores de toma de decisiones.
Escenario alternativo	<p>2a. El sistema tiene problemas técnicos al cargar la información.</p> <p>2a1. El sistema mostrará un mensaje de que algo no está yendo bien y que lo intente dentro de un tiempo.</p>

CU-16 Compra de activos	
Actor	Bróker
Breve descripción	Este caso de uso describe el proceso de compra de activos por parte de un bróker.
Precondición	El bróker ha iniciado sesión, tiene fondos disponibles y ha seleccionado el activo que desea comprar.
Postcondición	Se registra la transacción de compra y los activos adquiridos se añaden al portafolio general.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la función de compra de activos. 2. El sistema presenta un formulario que permite al bróker ingresar los detalles de la transacción. 3. La bróker rellena y revisa la información 4. El bróker confirma la transacción. 5. El sistema verifica la disponibilidad de fondos. 6. Se ejecuta la transacción. 7. Se actualiza el portafolio general del bróker. 8. El sistema emite una transacción.
Escenario alternativo	<p>5a. El bróker no tiene fondos suficientes.</p> <p>5a1. El sistema muestra un mensaje conforme no se ha podido realizar la transacción ya que no tiene fondos suficientes.</p>

CU-17		Venta de activos
Actor	Bróker	
Breve descripción	Este caso de uso describe el proceso de venta de activos por parte de un bróker.	
Precondición	El bróker ha iniciado sesión y tiene activos en su portafolio que desea vender.	
Postcondición	Se registra la transacción de venta en la plataforma, y los activos vendidos se elimina de la cartera del bróker.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la función de venta de activos. 2. El sistema presenta una lista de los activos del bróker. 3. El bróker selecciona los activos a vender especificando la cantidad. 4. El bróker revisa la operación y confirma la transacción. 5. El sistema verifica la disponibilidad de los activos seleccionados. 6. El sistema ejecuta la transacción de venta y se actualiza el portafolio del bróker eliminando los activos vendidos. 7. El sistema registra la transacción. 	
Escenario alternativo	<p>5a. El bróker no tiene disponibilidad suficiente del activo.</p> <p>5a1. El sistema muestra un mensaje conforme no se puede realizar la operación ya que no tiene esa cantidad de activos.</p>	

CU-18		Ejecución de órdenes avanzadas
Actor	Bróker	
Breve descripción	Este caso de uso describe el proceso de ejecución de órdenes avanzadas por parte de un bróker, permitiéndole poner stop-loss y take-profit.	
Precondición	El bróker ha iniciado sesión y tiene activos en su cartera que desea automatizar.	
Postcondición	Se ejecuta las órdenes especificadas por el bróker cuando se cumplan las reglas.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede al activo a introducir las órdenes. 2. El sistema presenta opciones de stop-loss y take-profit. 3. El bróker selecciona el tipo de orden a ejecutar. 4. El bróker especifica los parámetros de la orden. 5. El bróker revisa la información proporcionada. 6. El sistema registra la orden avanzada y la ejecutará cuando se cumplan las reglas. 	
Escenario alternativo	<p>1a. El bróker no tiene activos.</p> <p>1a1. El sistema no dará la opción de ordenes avanzadas.</p>	

CU-19 Capacidad de recomendación de activos personalizados	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad de un bróker para recibir recomendaciones personalizadas de activos por una AI.
Precondición	El bróker ha iniciado sesión y ha proporcionado información sobre su estilo de trading y metas.
Postcondición	El bróker recibe recomendaciones de activos personalizadas adaptadas a sus metas y tolerancia al riesgo.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la función de recomendación de activos personalizados. 2. El sistema solicita la información adicional al bróker. 3. El bróker proporciona la información. 4. El sistema utiliza algoritmos de IA para analizar y generar recomendaciones de activos. 5. El bróker revisa las recomendaciones proporcionadas. 6. El sistema ofrece detalles sobre cada recomendación. 7. El bróker puede decidir si seguir la recomendación y ejecutar la operación.
Escenario alternativo	<p>3a. El bróker no proporciona información suficiente o precisa.</p> <p>3a1. El sistema muestra un mensaje conforme no puede darle recomendaciones.</p>

CU-20 Capacidad de realizar trading automático	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad de un bróker para realizar trading automático, permitiéndole automatizar la ejecución de operaciones.
Precondición	El bróker ha iniciado sesión y ha configurado previamente estrategias o señales para el trading automático.
Postcondición	Se ejecuta automáticamente las operaciones de trading según las estrategias o señales predefinidas por el bróker.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la función de trading automático. 2. El sistema presenta opciones para configurar estrategias o indicadores. 3. El bróker configura las reglas y parámetros para la ejecución automática de operaciones. 4. El bróker revisa y confirma la configuración del bot. 5. El sistema verifica la disponibilidad de fondos para la ejecución. 6. El sistema inicia el trading automático.
Escenario alternativo	<p>3a. El bróker no proporciona información suficiente o precisa.</p> <p>3a1. El sistema muestra un mensaje conforme no puede empezar el trading automatico</p> <p>5a. El bróker no tiene disponibilidad suficiente del activo.</p> <p>5a1. El sistema muestra un mensaje conforme no se puede realizar la operación ya que no tiene esa cantidad de activos.</p>

CU-21 Herramientas de análisis técnico	
Actor	Bróker
Breve descripción	Este caso de uso describe la utilización de herramientas de análisis técnico por parte de un bróker, permitiéndole realizar un análisis detallado del activo.
Precondición	El bróker ha iniciado sesión, tiene acceso al mercado y ha seleccionado un activo.
Postcondición	El bróker ha realizado un análisis técnico utilizando las herramientas disponibles.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker ha seleccionado un activo del mercado. 2. El sistema muestra las herramientas de análisis. 3. El bróker elige las herramientas específicas que desea utilizar. 4. El sistema presenta gráficos y datos históricos del activo seleccionado. 5. El bróker interpreta los resultados del análisis técnico y utiliza la información para respaldar la decisión.
Escenario alternativo	<p>2a. El sistema tiene problemas internos en dar las herramientas.</p> <p>2a1. El sistema únicamente mostrará la gráfica del valor sin herramientas de análisis técnico acompañado de un mensaje de error.</p>

CU-22 Gestión de impuestos automatizada	
Actor	Bróker, Organización
Breve descripción	Este caso de uso describe la capacidad de la plataforma para calcular los impuestos asociados a las transacciones realizadas por los brokers.
Precondición	El bróker u organización ha realizado transacciones que generan obligaciones fiscales.
Postcondición	Se han gestionado de forma automatizada los cálculos fiscales de las transacciones realizadas.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker u organización accede a la sección de impuestos. 2. El sistema calcula automáticamente los impuestos generados por las transacciones dependiendo del país donde opera. 3. El sistema presenta una tabla con las transacciones realizadas. 4. El bróker u organización revisa los movimientos y el impuesto generado.
Escenario alternativo	<p>2a. El sistema tiene discrepancias con las transacciones y la localidad.</p> <p>2a1. El sistema genera un mensaje de alerta al bróker para que arregle el problema de descuadre.</p>

CU-23		Monitoreo y seguimiento de portafolio
Actor	Bróker, Organización	
Breve descripción	En este caso de uso describe la capacidad de los brokers y las organizaciones para monitorear y realizar un seguimiento detallado de su portafolio.	
Precondición	El bróker u organización tiene activos en su portafolio y accede a la sección de portafolio.	
Postcondición	Se ha realizado el monitoreo y seguimiento del portafolio, y la información actualizada está disponible.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker u organización accede a la sección de portafolio. 2. El sistema presenta una visión general de los activos en el portafolio, incluyendo su valor actual. 3. El bróker u organización puede explorar detalles adicionales de cada activo, incluyendo gráficos. 	
Escenario alternativo	<p>2a. El sistema tiene problemas técnicos a generar la presentación del portafolio.</p> <p>2a1. El sistema muestra un mensaje de error conforme no ha sido posible y que lo intente durante el transcurso de 10 minutos.</p>	

CU-24		Notificaciones y alertas personalizadas
Actor	Bróker	
Breve descripción	Este caso de uso describe la capacidad del bróker para recibir notificaciones y alertas personalizadas.	
Precondición	El bróker ha iniciado sesión y ha configurado sus preferencias de notificación.	
Postcondición	Se ha enviado notificaciones y alertas personalizadas al bróker según sus preferencias.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker marca un activo para recibir notificaciones. 2. El sistema presenta opciones y reglas para notificar al bróker. 3. El bróker configura sus preferencias de notificaciones seleccionando precios del activo. 4. El sistema confirma la configuración de notificaciones del bróker. 5. El sistema envía la notificación o alerta al bróker cuando se produce el evento configurado. 	
Escenario alternativo	<p>3a. El bróker cambia sus preferencias de notificación.</p> <p>3a1. El sistema actualiza las configuraciones y ajusta el envío.</p>	

CU-25 Listar portafolios de social trading	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad de listar portafolios para realizar social trading.
Precondición	El bróker ha iniciado sesión y se encuentra en la sección de Social Trading.
Postcondición	Se muestra todos los portafolios de brokers que tienen habilitado el social trading.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker va a la sección de social trading. 2. El sistema presenta una tabla con los portafolios que se puede entrar a ver y hacer seguimiento. 3. El bróker selecciona a que portafolio ir y ver sus operaciones.
Escenario alternativo	<p>2a. El sistema tiene problemas técnicos a generar la presentación del portafolio de social trading.</p> <p>2a1. El sistema muestra un mensaje de error conforme no ha sido posible y que lo intente durante el transcurso de 10 minutos.</p>

CU-26 Seguimiento de portafolio de social trading	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para realizar un seguimiento del portafolio de social trading.
Precondición	El bróker ha iniciado sesión en la plataforma y ha decidido ver perfiles de brokers exitosos para recopilar sus operaciones.
Postcondición	El bróker ha realizado un seguimiento efectivo y detallado.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la sección de comunidad. 2. El sistema presenta una visión general del portafolio de social trading del bróker. 3. El bróker explora detalles adicionales de las operaciones realizadas.
Escenario alternativo	<p>1a. El sistema tiene problemas técnicos a generar la presentación del portafolio de social trading.</p> <p>1a1. El sistema muestra un mensaje de error conforme no ha sido posible y que lo intente durante el transcurso de 10 minutos.</p>

CU-27 Copiar portafolio de social trading	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para copiar el portafolio de un bróker exitoso en el contexto de social trading, permitiendo al bróker replicar automáticamente las operaciones realizadas por el bróker seguido.
Precondición	El bróker ha iniciado sesión y ha decidido copiar el portafolio de otro broker.

Postcondición	El bróker ha copiado exitosamente el portafolio del bróker seleccionado, y las operaciones se replican automáticamente.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a social trading y selecciona el bróker cuyo portafolio desea copiar. 2. El sistema presenta opciones para copiar el portafolio. 3. El bróker elige la cantidad de fondos a asignar al portafolio copiado. 4. El sistema confirma la configuración y solicita la confirmación. 5. El bróker confirma la acción, y el sistema inicia la replicación.
Escenario alternativo	<p>1a. El sistema tiene problemas técnicos a generar la presentación del portafolio de social trading.</p> <p>1a1. El sistema muestra un mensaje de error conforme no ha sido posible y que lo intente durante el transcurso de 10 minutos.</p>

CU-28 Desvincular portafolio de social trading	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para desvincular su portafolio de social trading, permitiéndole dejar de replicar automáticamente las operaciones de otro bróker.
Precondición	El bróker ha iniciado sesión y tiene un portafolio vinculado a otro bróker con social trading.
Postcondición	El bróker ha desvinculado exitosamente su portafolio del bróker seleccionado y las operaciones ya no se replican.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la sección de configuración de social trading. 2. El sistema presenta la lista de brokers cuyos portafolios están vinculados. 3. El bróker selecciona el bróker del cual desea desvincular su portafolio. 4. El sistema muestra un mensaje para confirmar la acción. 5. El bróker confirma la acción y el sistema desvincula el portafolio.
Escenario alternativo	<p>1a. El sistema tiene problemas técnicos a generar la presentación del portafolio de social trading.</p> <p>1a1. El sistema muestra un mensaje de error conforme no ha sido posible y que lo intente durante el transcurso de 10 minutos.</p>

CU-29 Pruebas de estrategias trading en entorno sandbox	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para realizar pruebas de estrategias de trading en un entorno sandbox.
Precondición	El bróker ha iniciado sesión y ha seleccionado un activo.
Postcondición	El bróker ha realizado pruebas exitosas de estrategias de trading en el entorno de sandbox.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede al activo y hace la misma operación de compra de activos con el entorno sandbox marcado. 2. El sistema realiza la operación lógica pero no ejecuta la transacción. 3. El bróker monitorea el rendimiento de la estrategia en tiempo real.
Escenario alternativo	<p>3a. El bróker ha detectado que la estrategia no esta funcionando.</p> <p>3a1. El bróker puede modificar en tiempo real el valor sin alterar nada de su portafolio.</p>

CU-30 Capacidad de exportación sobre los datos	
Actor	Bróker, Organización
Breve descripción	Este caso de uso describe la capacidad del bróker u organización para exportar todos los datos recopilados, permitiéndolos usar para análisis adicionales.
Precondición	El bróker u organización ha iniciado sesión y está en la configuración de la cuenta.
Postcondición	Los datos seleccionados han sido exportados y están disponibles para su uso fuera de la plataforma.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker u organización accede a la sección de exportación de datos. 2. El sistema presenta la opción de exportar todos los datos. 3. El bróker u organización configura los parámetros de la exportación, como el rango de fechas. 4. El sistema inicia el proceso de exportación. 5. El bróker u organización descarga el archivo CSV.
Escenario alternativo	<p>4a. El sistema tiene problemas técnicos a generar el CSV.</p> <p>2a1. El sistema muestra un mensaje de error conforme no ha sido posible y que lo intente durante el transcurso de 10 minutos.</p>

CU-31 Creación de grupos	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para crear grupos permitiendo organizar y gestionar las comunidades.
Precondición	El bróker ha iniciado sesión y está en la sección grupos.
Postcondición	Se ha creado exitosamente un nuevo grupo en la plataforma.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker accede a la sección grupos. 2. El sistema presenta la opción de crear un nuevo grupo. 3. El sistema muestra un formulario solicitando información para la creación del grupo. 4. El sistema crea el nuevo grupo y presenta opciones para asignar elementos.
Escenario alternativo	<p>3a. El bróker introduce un nombre del grupo que ya está en uso.</p> <p>3a1. El sistema muestra un mensaje de error y solicita que proporcione un nombre válido.</p>

CU-32 Gestión de grupos	
Actor	Administrador grupo
Breve descripción	Este caso de uso describe la capacidad del administrador del grupo para gestionar y organizar los elementos dentro de un grupo.
Precondición	El administrador del grupo ha iniciado sesión y tiene permisos para gestionar el grupo.
Postcondición	Los elementos dentro del grupo han sido gestionados según las acciones realizadas por el administrador del grupo.
Escenario básico	<ol style="list-style-type: none"> 1. El administrador del grupo accede a la sección de grupos. 2. El administrador edita el grupo. 3. El sistema proporciona opciones para realizar acciones de gestión como invitar a nuevos miembros, agregar nuevos canales, etc. 4. El administrador realiza los cambios y los guarda. 5. El sistema procesa los cambios.
Escenario alternativo	<p>4a. Si se añaden nuevos canales, el sistema necesitará información.</p> <p>4a1. El sistema imprimirá un formulario pidiendo toda la información necesaria para llevar a cabo la acción.</p>

CU-33 Listar brokers	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad de listar los brókeres dados de alta en el sistema.
Precondición	El bróker ha iniciado sesión y se encuentra en una sección donde el sistema permite listar brokers.

Postcondición	Se muestra todos los brokers que se han dado de alta en el sistema mediante unos filtros.
Escenario básico	<ol style="list-style-type: none"> 1. El sistema muestra un área de texto libre para escribir el parámetro a buscar. 2. El bróker introduce el texto que desea buscar. 3. El sistema muestra un listado con todos los brokers coincidentes con el texto indicado por el bróker.
Escenario alternativo	<p>3a. El sistema tiene problemas técnicos al buscar brokers.</p> <p>3a1. El sistema muestra un mensaje de error conforme no ha sido establecer conexión con el servicio de usuarios.</p>

CU-34 Listar grupos	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad de listar los grupos dados de alta en el sistema.
Precondición	El bróker ha iniciado sesión y se encuentra en la sección de grupos.
Postcondición	Se muestra todos los grupos que se han dado de alta en el sistema mediante unos filtros.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker se encuentra en la sección grupos. 2. El sistema muestra un área de texto libre para escribir el parámetro a buscar. 3. El bróker introduce el texto que desea buscar. 4. El sistema muestra un listado con todos los grupos coincidentes con el texto indicado por el bróker.
Escenario alternativo	<p>4a. El sistema tiene problemas técnicos al buscar grupos.</p> <p>4a1. El sistema muestra un mensaje de error conforme no ha sido establecer conexión con el servicio de grupos.</p>

CU-35 Creación de canales	
Actor	Administrador grupo
Breve descripción	Este caso de uso describe la capacidad del administrador del grupo para crear canales dentro de un grupo.
Precondición	El administrador del grupo ha iniciado sesión y tiene permisos para crear canales dentro del grupo.
Postcondición	Se ha creado exitosamente un nuevo canal dentro del grupo, y los miembros pueden participar en el grupo del canal.
Escenario básico	<ol style="list-style-type: none"> 1. El administrador del grupo accede a la sección canales. 2. El sistema presenta opciones para crear un nuevo canal. 3. El administrador del grupo proporciona la información requerida y confirma la creación del nuevo canal. 4. El sistema crea el nuevo canal.
Escenario alternativo	<p>3a. El administrador del grupo introduce un nombre del canal que ya está en uso.</p> <p>3a1. El sistema muestra un mensaje de error y solicita que proporcione un nombre válido.</p>

CU-36		Gestión de canales
Actor	Administrador grupo	
Breve descripción	En este caso de uso describe la capacidad del administrador del grupo para gestionar el canal dentro de un grupo, permitiendo configurarlo.	
Precondición	El administrador del grupo ha iniciado sesión y tiene permisos para gestionar el canal.	
Postcondición	El canal dentro del grupo ha sido gestionado según las acciones realizadas por el administrador del grupo.	
Escenario básico	<ol style="list-style-type: none"> 1. El administrador del grupo accede a la sección de canal. 2. El sistema presenta una lista de canales disponibles. 3. El administrador del grupo selecciona el canal y selecciona el botón de editar. 4. El sistema proporciona opciones para realizar acciones de gestión, como modificar la configuración del canal. 5. El administrador del grupo realiza las acciones deseadas y confirma las mismas. 	
Escenario alternativo	<p>4a. El usuario administrador eliminar un canal.</p> <p>4a1. El sistema muestra una advertencia conforme se va a eliminar un canal del grupo.</p> <p>4b. El usuario administrador cambia los permisos del canal.</p> <p>4b1. El sistema muestra una advertencia conforme los cambios afectarán a la privacidad o accesibilidad del canal.</p>	

CU-37		Listar canales
Actor	Bróker, Administrador grupo	
Breve descripción	Este caso de uso describe la capacidad de listar los canales dados de alta en un grupo.	
Precondición	El bróker ha iniciado sesión y se encuentra en la sección de grupos.	
Postcondición	Se muestra todos los canales que se han dado de alta en el grupo.	
Escenario básico	<ol style="list-style-type: none"> 1. El bróker se encuentra dentro de un grupo. 2. El sistema muestra todos los canales creados para ese grupo. 	
Escenario alternativo	<p>2a. El sistema tiene problemas técnicos al buscar los canales.</p> <p>2a1. El sistema muestra un mensaje de error conforme no ha sido establecer conexión con el servicio de grupos.</p>	

CU-38 Enviar invitación de grupo	
Actor	Administrador grupo
Breve descripción	Este caso de uso describe la capacidad del administrador del grupo para enviar invitaciones a nuevos miembros para unirse al grupo.
Precondición	El administrador del grupo ha iniciado sesión en la plataforma y tiene permisos para enviar invitaciones.
Postcondición	Se ha enviado una invitación al nuevo miembro, y este puede aceptar o rechazar la invitación.
Escenario básico	<ol style="list-style-type: none"> 1. El administrador del grupo accede a la sección de gestión del grupo en concreto. 2. El sistema presenta la opción de invitar, solicitando información como la dirección de correo electrónico. 3. El administrador del grupo proporcionará la información requerida. 4. El sistema envía una invitación al correo electrónico o notificación al miembro.
Escenario alternativo	<p>3a. El administrador del grupo ha proporcionado un usuario inexistente.</p> <p>3a1. El sistema mostrará un mensaje conforme ese usuario no ha sido encontrado.</p>

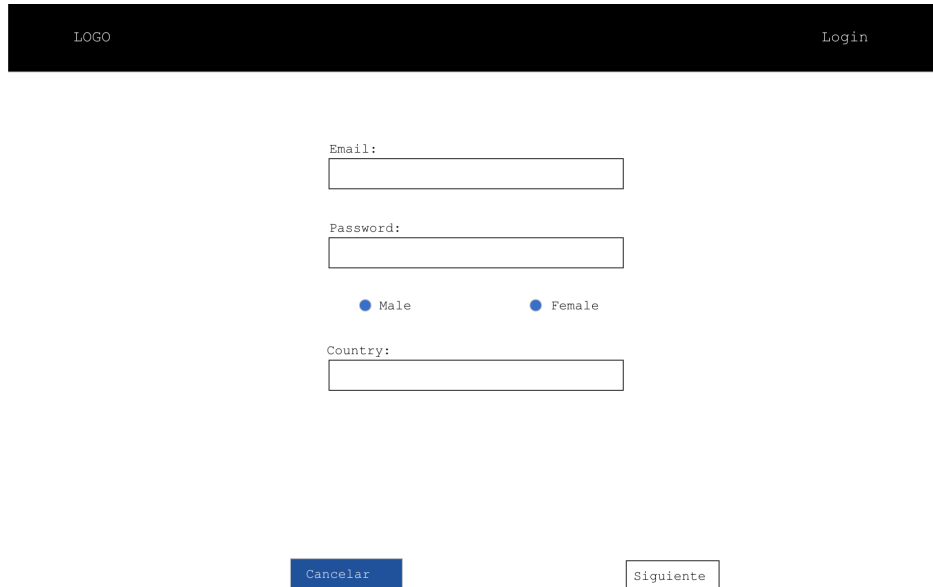
CU-39 Aceptar invitación de grupo	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para aceptar una invitación para unirse a un grupo.
Precondición	El bróker ha recibido una invitación para unirse a un grupo en la plataforma.
Postcondición	El bróker ha aceptado la invitación y ahora es un miembro activo del grupo.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker recibe una notificación o correo electrónico con la invitación. 2. El bróker accede a la plataforma y revisa la invitación pendiente. 3. El sistema presenta la opción de aceptar la invitación. 4. El bróker confirma su deseo de unirse al grupo. 5. El sistema actualiza el estado del bróker a miembro activo.
Escenario alternativo	<p>4a. El bróker decide no realizar ninguna acción con la invitación.</p> <p>4a1. El sistema marcará como caducada la invitación después de 72 h.</p>

CU-40 Rechazar invitación de grupo	
Actor	Bróker
Breve descripción	Este caso de uso describe la capacidad del bróker para rechazar una invitación para unirse a un grupo en la plataforma.
Precondición	El bróker ha recibido una invitación para unirse a un grupo en la plataforma.
Postcondición	El bróker ha rechazado la invitación y no es un miembro del grupo.
Escenario básico	<ol style="list-style-type: none"> 1. El bróker recibe una notificación o correo electrónico con la invitación. 2. El bróker accede a la plataforma y revisa la invitación pendiente. 3. El sistema presenta la opción de rechazar la invitación. 4. El bróker confirma su deseo de rechazar la invitación. 5. El sistema marca la petición como rechazada.
Escenario alternativo	<p>4a. El bróker decide no realizar ninguna acción con la invitación.</p> <p>4a1. El sistema marcará como caducada la invitación después de 72 h.</p>

CU-41 Bloquear usuarios	
Actor	Administrador
Breve descripción	Este caso de uso describe la capacidad del administrador para bloquear usuarios en la plataforma.
Precondición	El administrador ha iniciado sesión y está en el panel de control.
Postcondición	El usuario selección ha sido bloqueado y tiene restricciones de acceso según la acción realizada por el administrador.
Escenario básico	<ol style="list-style-type: none"> 1. El administrador accede al panel de control de administración. 2. El sistema presenta la opción de bloquear usuario. 3. El sistema muestra un listado de usuarios registrados y el administrador selecciona los usuarios a bloquear. 4. El administrador filtra por dirección de correo electrónico los usuarios y los selecciona proporcionando una razón. 5. El sistema bloquea al usuario seleccionado. 6. Se notifica al usuario bloqueado sobre la acción realizada por el administrador.
Escenario alternativo	<p>4a. El sistema no obtiene coincidencias con lo introducido por el bróker.</p> <p>4a1. El sistema retornara una lista vacía con el mensaje de no existe tal usuario.</p>

2.3. Prototipo de pantallas

CU-01 - Registro de brokers



Logo Login

Email:

Password:

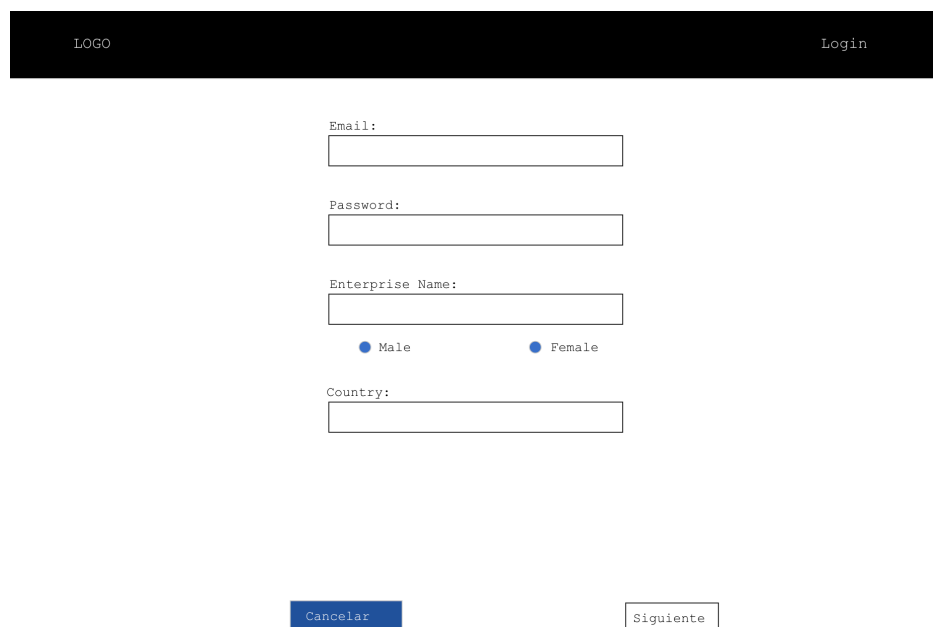
Male Female

Country:

Cancelar Siguiete

Ilustración 12 - Prototipado CU-01 Registro de brokers

CU-02 - Registro de organizaciones



Logo Login

Email:

Password:

Enterprise Name:

Male Female

Country:

Cancelar Siguiete

Ilustración 13 - Prototipado CU-02 Registro de organizaciones

CU-03 - Gestión de perfil bróker

LOGO Home Portafolio Mercado Comunidad IA @john.doe

john@doe.com | @john.doe

Email:

Password:

Fecha de nacimiento:

Recibir alertas

Guardar

Ilustración 14 - Prototipado CU-03 Gestión de perfil bróker

CU-04 - Gestión de organización

LOGO Home Portafolio Mercado Comunidad IA #BlackRock

admin@johndoe.enterprise

Enterprise Owner:

Password:

CIF:

Enterprise Name:

Guardar

Ilustración 15 - Prototipado CU-04 Gestión de organización

CU-05 - Gestión de usuarios de la organización

LOGO Home Portafolio Mercado Comunidad IA #BlackRock

Brokers

John Doe	1.000.000 €	<input type="button" value="X"/>
Prueba	1.000.000 €	<input type="button" value="X"/>
Prueba 2	2.000.000 €	<input type="button" value="X"/>

Ilustración 16 - Prototipado CU-05 Gestión de usuarios de la organización

CU-06 - Inicio de sesión seguro

LOGO

Login

Email:

Password:

Ilustración 17 - Prototipado CU-06 Inicio de sesión seguro

CU-07 - Búsqueda de brokers

LOGO Home Portafolio Mercado Comunidad IA @john.doe

Buscar bróker

Búsqueda por nombre:

John Doe	Ver perfil
Prueba	Ver perfil
Prueba 2	Ver perfil

Ilustración 18 - Prototipado CU-07 Búsqueda de brokers

CU-08 - Enviar mensaje a bróker

LOGO Home Portafolio Mercado Comunidad IA @john.doe

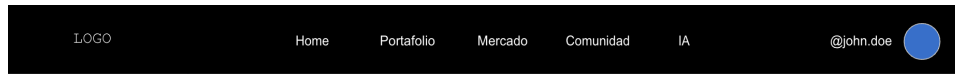


john@doe.com | @john.doe

Contactar

Ilustración 19 - Prototipado CU-08 Enviar mensaje a bróker

CU-09 - Creación de portafolio



Portafolio

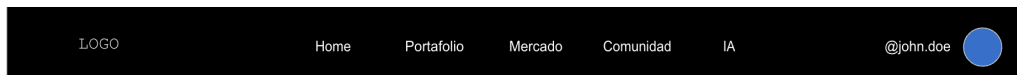
Creación de portafolio

Portafolio name:

Guardar

Ilustración 20 - Prototipado CU-09 Creación de portafolio

CU-10 - Gestión de portafolio



Portafolio

Gestión de portafolio

Name:

Public

Private

Guardar

Ilustración 21 - Prototipado CU-10 Gestión de portafolio

CU-11 – Listar portafolios



Portafolio

Portafolio 1
Portafolio 2
Portafolio 3

Ilustración 22 - Prototipado CU-11 Listar portafolios

CU-12 - Búsqueda de activos por categoría



Mercado

Forex
Criptomonedas
Materias primas
Empresas

Ilustración 23 - Prototipado CU-12 Búsqueda de activos por categoría

CU-13 - Búsqueda de activos por nombre

LOGO Home Portafolio Mercado Comunidad IA @john.doe

Mercado

Búscar activo

- Apple
- Bitcoin
- EUR/USD

Ilustración 24 - Prototipado CU-13 Búsqueda de activos por nombre

CU-14 - Seguimiento de activos por nombre

LOGO Home Portafolio Mercado Comunidad IA @john.doe

Mercado

Bitcoin 

Gráfica de cotización

Ilustración 25 - Prototipado CU-14 Seguimiento de activos por nombre

CU-15 - Consultar sentimiento del mercado



Ilustración 26 - Prototipado CU-15 Consultar sentimiento del mercado

CU-16 - Compra de activos



Ilustración 27 - Prototipado CU-16 Compra de activos

CU-17 - Venta de activos



Ilustración 28 - Prototipado CU-17 Venta de activos

CU-18 - Ejecución de órdenes avanzadas



Ilustración 29 - Prototipado CU-18 Ejecución de órdenes avanzadas

CU-19 - Capacidad de recomendación de activos personalizados (IA)

Este prototipo muestra una interfaz de usuario para la recomendación de activos personalizados. La barra superior es negra y contiene el texto 'LOGO' a la izquierda y 'Home', 'Portafolio', 'Mercado', 'Comunidad' e 'IA' en el centro. A la derecha de la barra superior se encuentra el texto '@john.doe' y un círculo azul que representa un perfil de usuario. El contenido principal de la interfaz está encerrado en un recuadro con una línea punteada y tiene el título 'Recomendación por IA' centrado. Debajo del título, hay dos campos de entrada de texto: 'Nivel de riesgo a asumir:' y 'Tipo de activo:'. Debajo de estos campos, hay un botón rectangular con el texto 'Generar'.

Ilustración 30 - Prototipado CU-19 Capacidad de recomendación de activos personalizados (IA)

CU-20 - Capacidad de realizar trading automático

Este prototipo muestra una interfaz de usuario para el trading automático. La barra superior es negra y contiene el texto 'LOGO' a la izquierda y 'Home', 'Portafolio', 'Mercado', 'Comunidad' e 'IA' en el centro. A la derecha de la barra superior se encuentra el texto '@john.doe' y un círculo azul que representa un perfil de usuario. El contenido principal de la interfaz está encerrado en un recuadro con una línea punteada y tiene el título 'Trading Bot' centrado. Debajo del título, hay tres campos de entrada de texto: 'Nivel de riesgo a asumir:', 'Tipo de activo:' y 'Deposito:'. Debajo de estos campos, hay un botón rectangular con el texto 'Generar'.

Ilustración 31 - Prototipado CU-20 Capacidad de realizar trading automático

CU-21 - Herramientas de análisis técnico



Ilustración 32 - Prototipado CU-21 Herramientas de análisis técnico

CU-22 - Gestión de impuestos automatizada

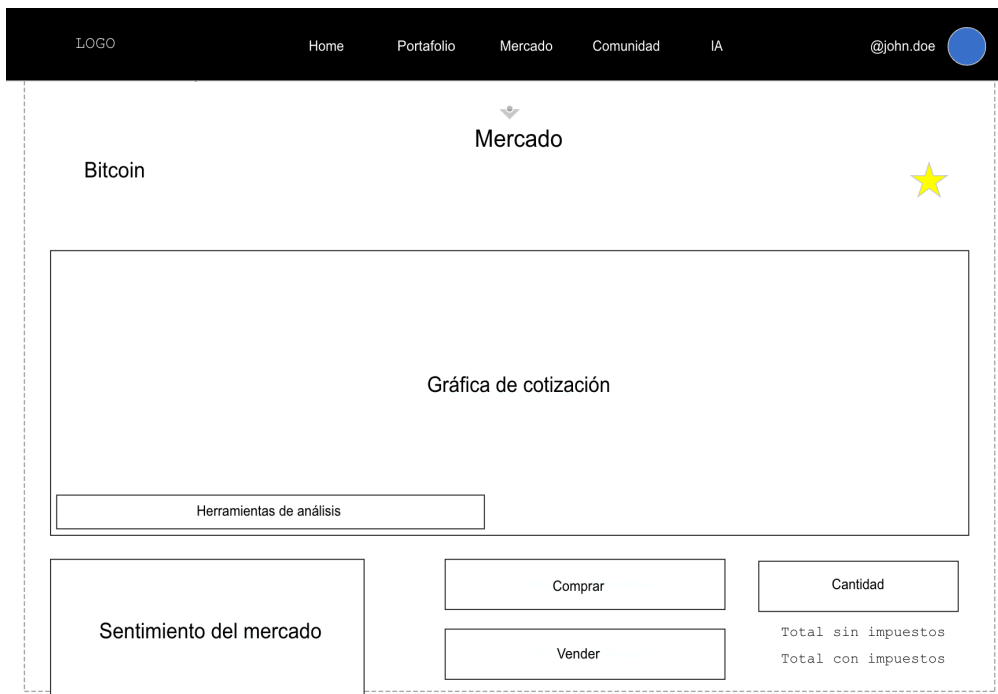


Ilustración 33 - Prototipado CU-22 Gestión de impuestos automatizados

CU-23 - Monitoreo y seguimiento de portafolio

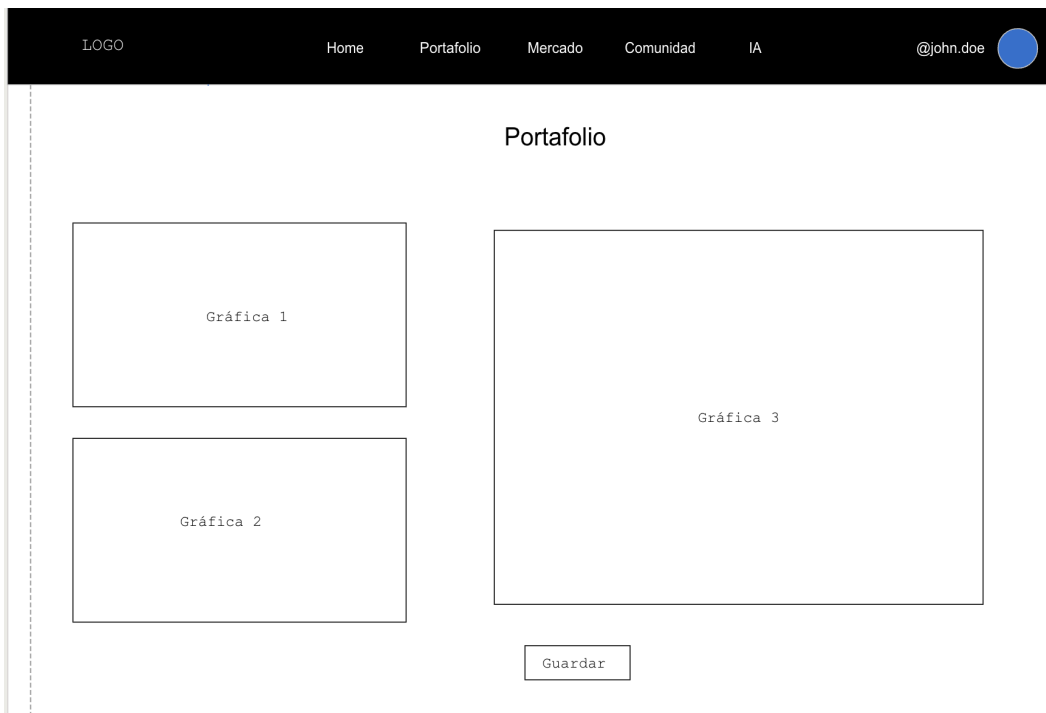


Ilustración 34 - Prototipado CU-23 Monitoreo y seguimiento de portafolio

CU-24 - Notificaciones y alertas personalizadas

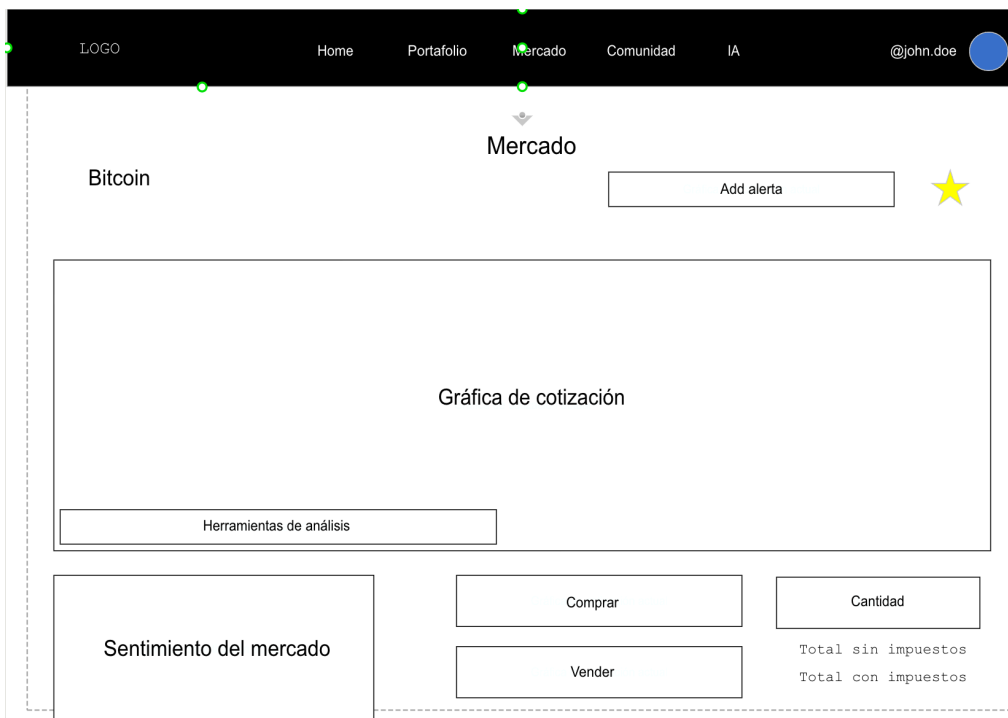


Ilustración 35 - Prototipado CU-24 Notificaciones y alertas personalizadas

CU-25 – Listar portafolios de social trading

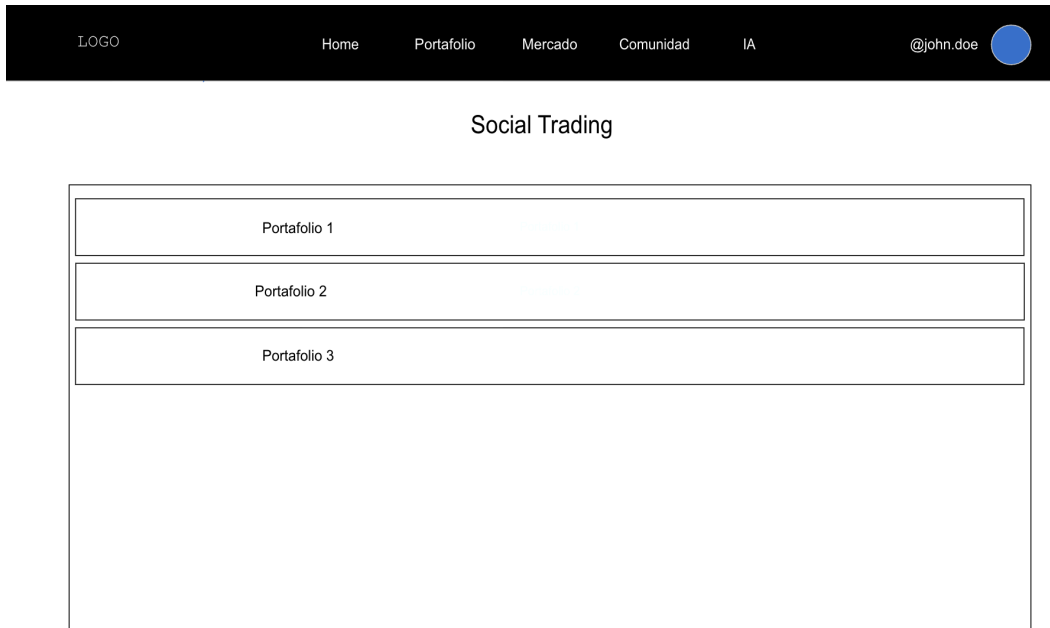


Ilustración 36 - Prototipado CU-25 Listar portafolios de social trading

CU-26 - Seguimiento de portafolio de social trading

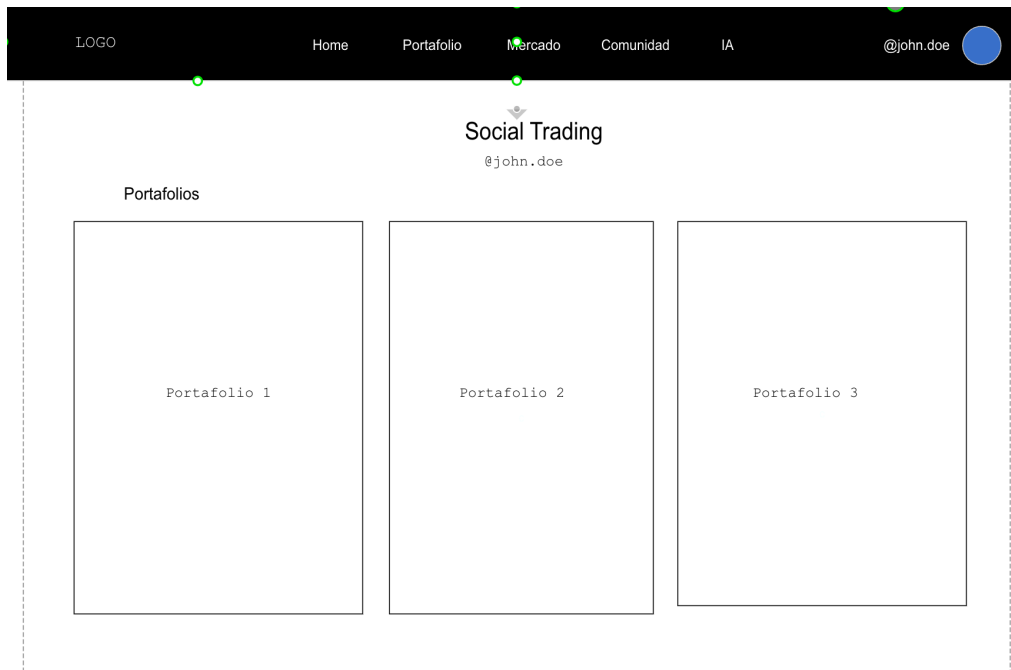


Ilustración 37 - Prototipado CU-26 Seguimiento de portafolio de social trading

CU-27 - Copiar portafolio de social trading

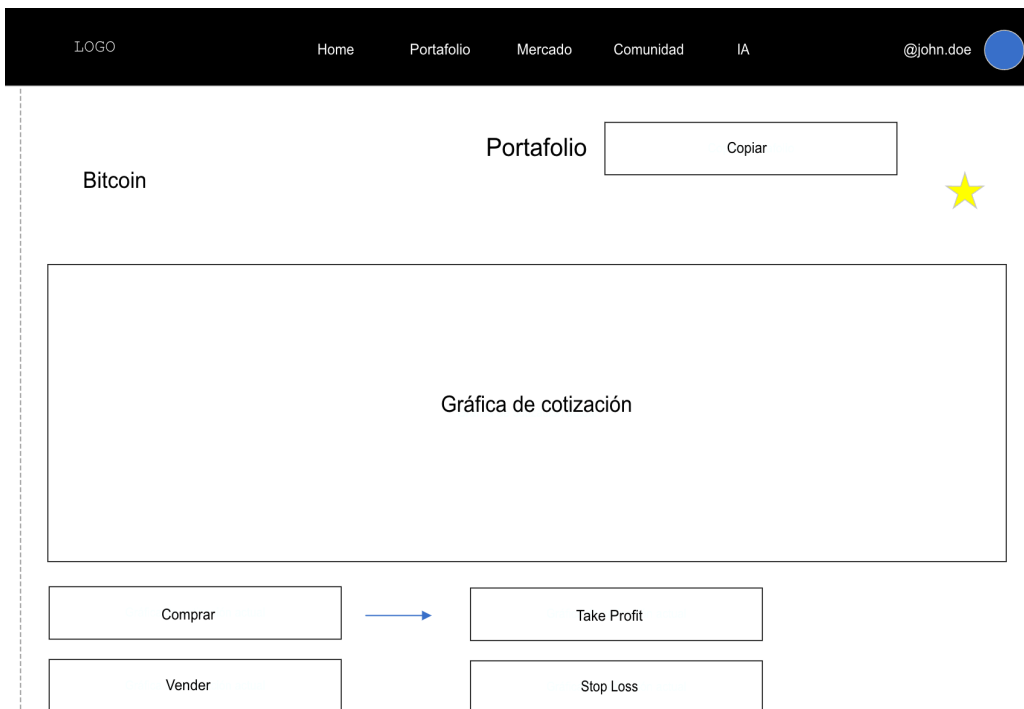


Ilustración 38 - Prototipado CU-27 Copiar portafolio de social trading

CU-28 - Desvincular portafolio de social trading

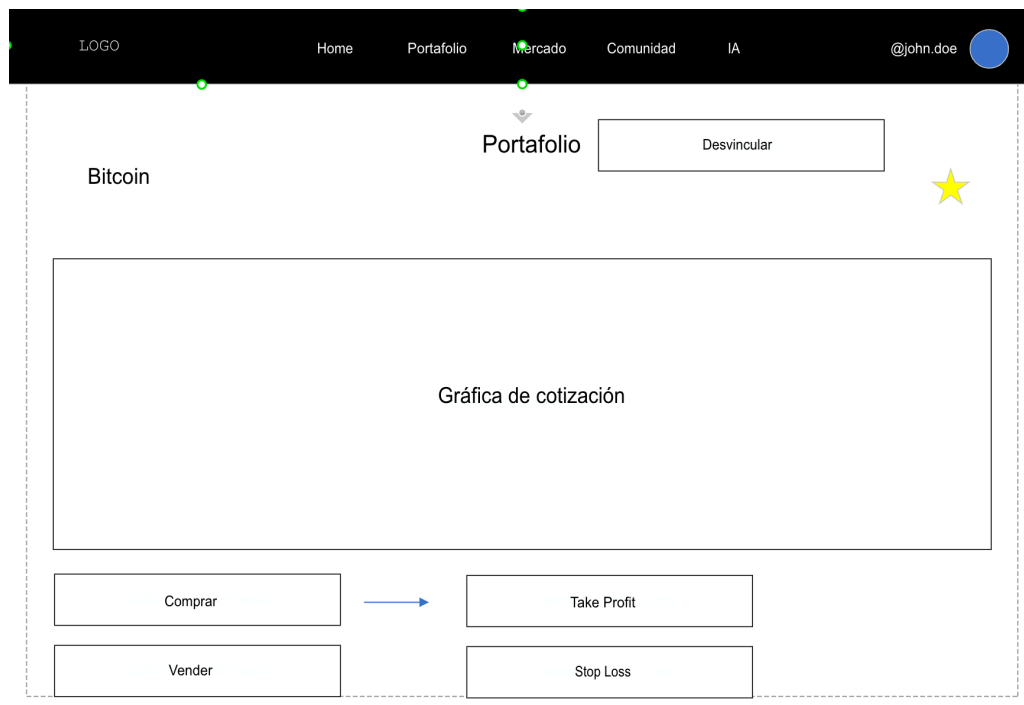


Ilustración 39 - Prototipado CU-28 Desvincular portafolio de social trading

CU-29 - Pruebas de estrategias trading en entorno sandbox



Ilustración 40 - Prototipado CU-29 Pruebas de estrategias trading en entorno sandbox

CU-30 - Capacidad de exportación sobre los datos

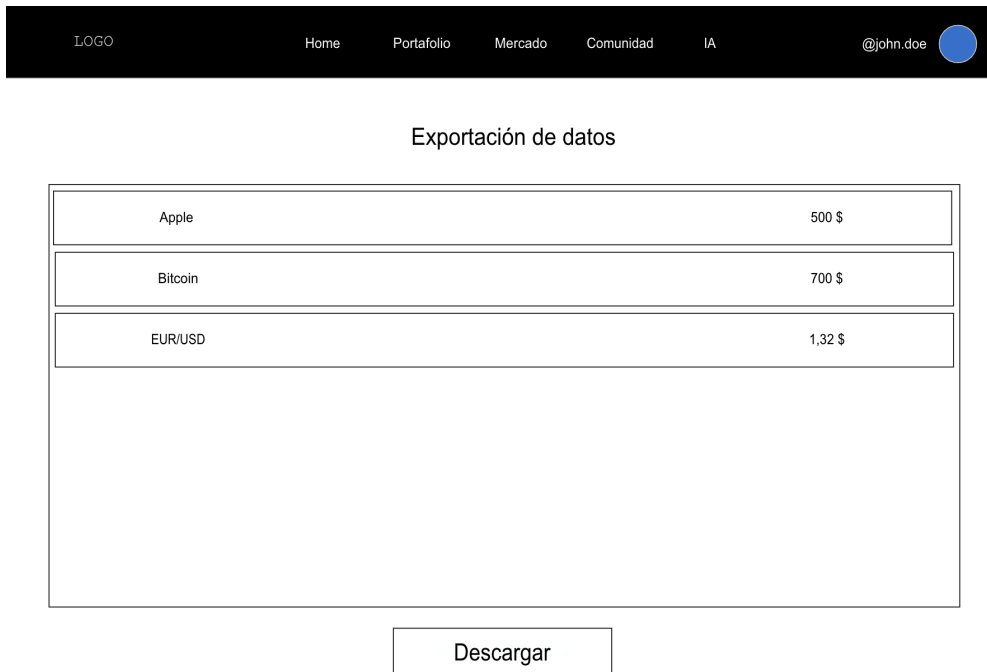


Ilustración 41 - Prototipado CU-30 Capacidad de exportación sobre los datos

CU-31 - Creación de grupos

LOGO Home Portafolio Mercado Comunidad IA @john.doe

Grupos

Bitcoin Community	<input type="button" value="Ver"/>
Ibex Community	<input type="button" value="Ver"/>
Forex Community	<input type="button" value="Ver"/>

Ilustración 42 - Prototipado CU-31 Creación de grupos

CU-32 - Gestión de grupos

LOGO Home Portafolio Mercado Comunidad IA @john.doe

Crear grupo

Nombre grupo

Descripción grupo

Ilustración 43 - Prototipado CU-32 Gestión de grupos

CU-33 – Listar brokers

LOGO Home Portafolio Mercado Comunidad IA @john.doe

Brokers

Buscar broker ...

Broker 3
Broker 2
Broker 1

Ilustración 44 - Prototipado CU-33 Listar brokers

CU-34 – Listar grupos

LOGO Home Portafolio Mercado Comunidad IA @john.doe


Grupos

Buscar grupo ...

Grupo 1
Grupo 2
Grupo 3

Ilustración 45 - Prototipado CU-34 Listar grupos

CU-35 - Creación de canales

LOGO Home Portafolio Mercado Comunidad IA @john.doe 

Bitcoin Community

Crear canal

Nombre canal


Descripcion canal

Añadir bróker

Crear canal

Ilustración 46 - Prototipado CU-35 Creación de canales

CU-36 - Gestión de canales

LOGO Home Portafolio Mercado Comunidad IA @john.doe 

Bitcoin Community
Nuevos

Gestionar canal

Nombre canal

Descripcion canal

Añadir bróker

Guardar

Ilustración 47 - Prototipado CU-36 Gestión de canales

CU-37 - Listar canales

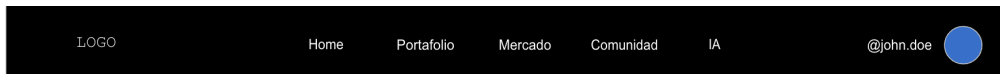


Grupo 1



Ilustración 48 - Prototipado CU-37 Listar canales

CU-38 - Enviar invitación de grupo



Has recibido una invitación a unirte al grupo

Aceptar

Rechazar

Ilustración 49 - Prototipado CU-38 Enviar invitación de grupo

CU-39 - Aceptar invitación de grupo



Has recibido una invitación a unirse al grupo

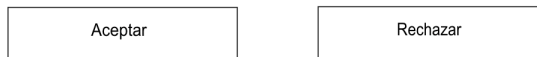
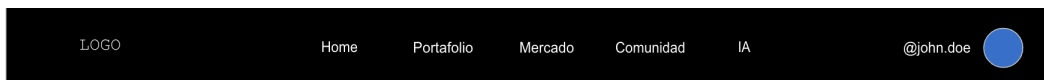


Ilustración 50 - Prototipado CU-39 Aceptar invitación de grupo

CU-40 - Rechazar invitación de grupo

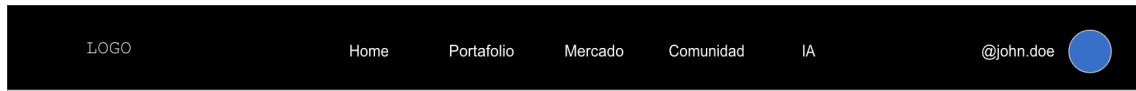


Has recibido una invitación a unirse al grupo



Ilustración 51 - Prototipado CU-40 Rechazar invitación de grupo

CU-41 - Bloquear usuarios



Bloquear usuario

Buscar bróker

Bloquear bróker

Ilustración 52 - Prototipado CU-41 Bloquear usuarios

2.4. Modelo relacional de la base de datos

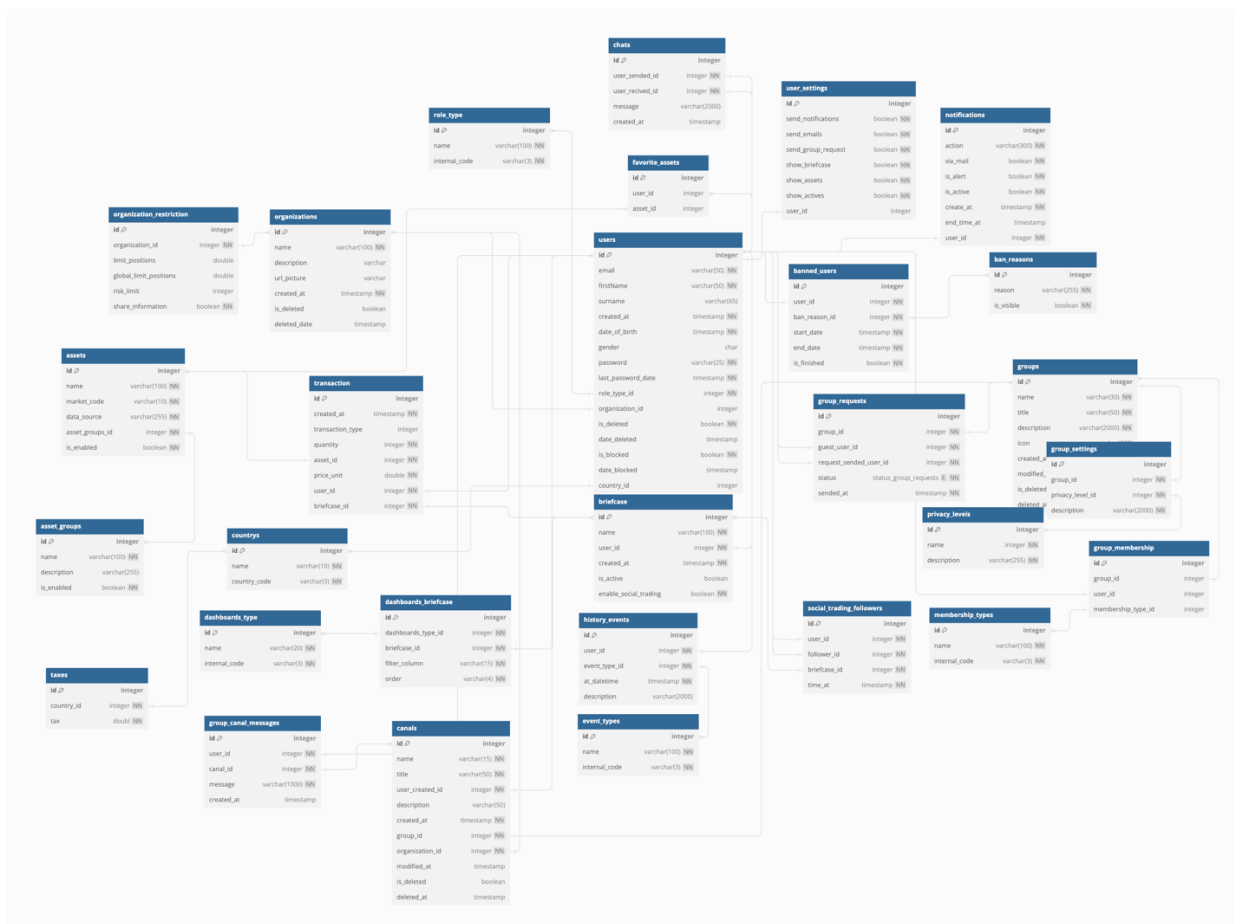


Ilustración 53 - Modelo relacional de la base de datos

A continuación, se adjunta el enlace para obtener una visión general y personalizada del modelo relacional sobre la base de datos: https://dbdocs.io/edouton/TFGEDGAR_DOUTON

La base de datos está compuesta por las siguientes relaciones:

- **users y countrys:** Relación de muchos a uno. Un usuario pertenece a un país.
- **users y chats:** Relación de uno a muchos. Un usuario puede enviar y recibir varios mensajes.
- **users y user_settings:** Relación de uno a uno. Un usuario tiene una configuración de usuario.
- **ban_reasons y banned_users:** Relación de uno a muchos. Un motivo de bloqueo puede estar asociado con varios usuarios bloqueados.
- **organizations y organization_restriction:** Relación de uno a uno. Una organización tiene restricciones específicas.

- **assets y transaction:** Relación de uno a muchos. Un activo puede estar asociado con varias transacciones.
- **assets y transaction:** Relación de uno a muchos. Un activo puede estar asociado con varias transacciones.
- **groups y group_requests:** Relación de uno a muchos. Un grupo puede tener varias solicitudes de pertenencia.
- **groups y group_membership:** Relación de uno a muchos. Un grupo puede tener varios miembros.
- **privacy_levels y group_settings:** Relación de uno a uno. Un nivel de privacidad está asociado con una configuración de grupo.
- **users y canals:** Relación de uno a muchos. Un usuario puede crear varios canales.
- **groups y group_canal_messages:** Relación de uno a muchos. Un grupo puede tener varios mensajes en el canal.
- **event_types y history_events:** Relación de uno a muchos. Un tipo de evento puede estar asociado con varios eventos históricos.
- **users y history_events:** Relación de uno a muchos. Un usuario puede tener varios eventos históricos.
- **asset_groups y assets:** Relación de uno a muchos. Un grupo de activos puede tener varios activos.
- **assets y favorite_assets:** Relación de muchos a muchos. Varios usuarios pueden tener varios activos como favoritos, y viceversa.
- **users y social_trading_followers:** Relación de uno a muchos. Un usuario puede tener varios seguidores en el social trading.
- **briefcase y dashboards_briefcase:** Relación de uno a muchos. Un portafolio puede tener varios *dashboards* asociados.
- **dashboards_type y dashboards_briefcase:** Relación de uno a muchos. Un tipo de *dashboard* puede estar asociado con varios dashboards en portafolios.
- **countries y taxes:** Relación de uno a muchos. Un país puede tener varios impuestos asociados.
- **users y notifications:** Relación de uno a muchos. Un usuario puede tener varias notificaciones asociadas.

2.5. Diagrama de clases

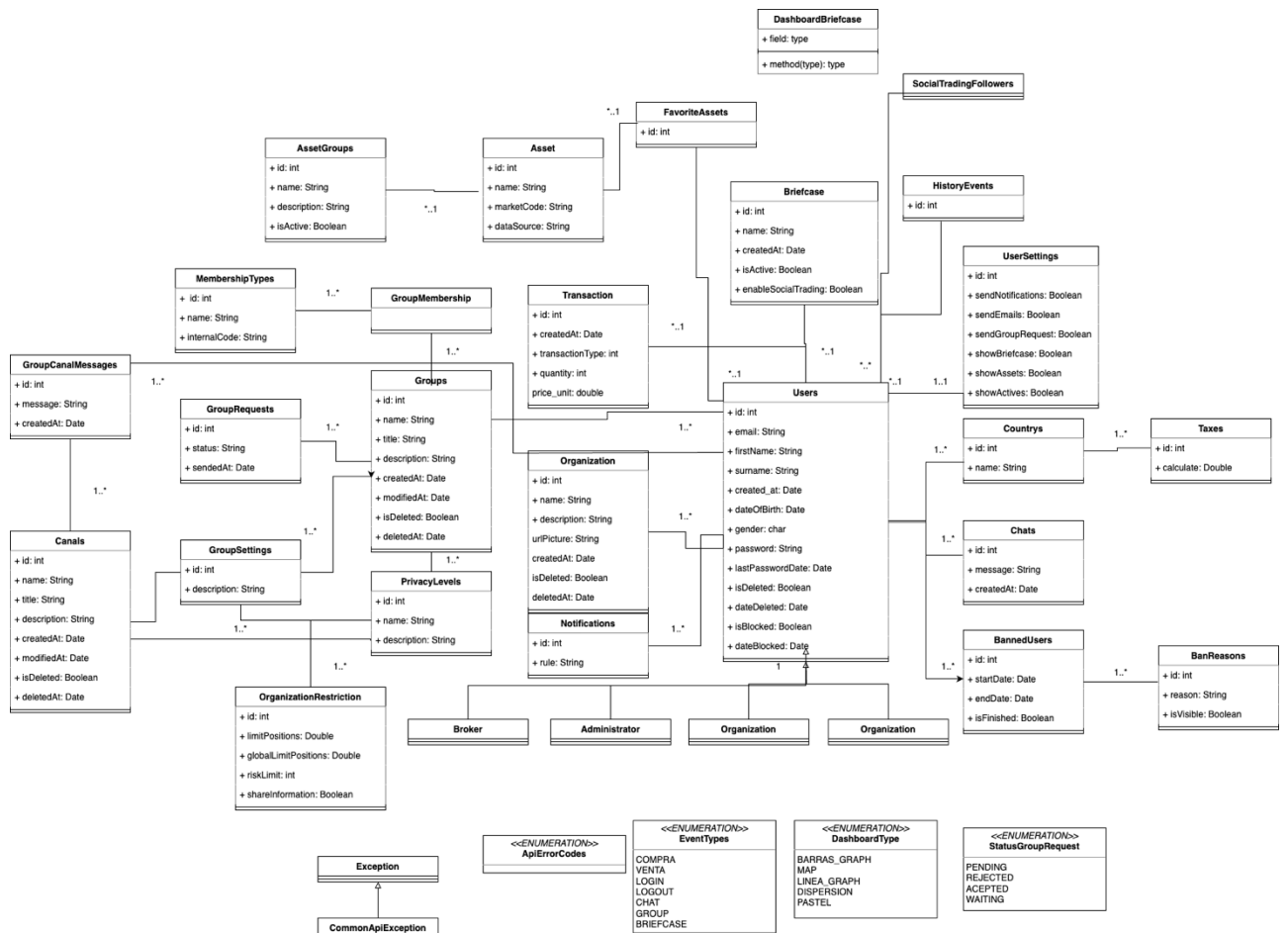


Ilustración 54 - Diagrama de clases

Restricciones

Users:

- El atributo email debe ser único para cada usuario.
- La relación con roleType debe apuntar a un tipo de rol existente.
- La relación con organizations indica que un usuario puede pertenecer a una organización, pero puede ser nulo, lo que significa que un usuario puede no pertenecer a ninguna organización.
- La relación con countrys indica la ubicación del usuario.

Chats:

- Las relaciones con users indican que los usuarios involucrados en el chat deben existir.

UserSettings:

- La relación con users indica que las configuraciones del usuario deben estar asociadas a un usuario existente.

BannedUsers:

- Las relaciones con users y banReasons indican que un usuario debe estar presente en la tabla users y el motivo de la prohibición debe ser válido.

OrganizationRestriction:

- La relación con organizations indica que una restricción debe asociarse a una organización existente.

Briefcase:

- La relación con users indica que un portafolio debe pertenecer a un usuario existente.

Transaction:

- Las relaciones con assets y users indican que una transacción debe involucrar a un activo y un usuario existentes.

Groups:

- La relación con canals indica que un grupo puede tener varios canales asociados.

Canals:

- Las relaciones con users, groups, y organizations indican que un canal debe ser creado por un usuario, pertenecer a un grupo y a una organización.

GroupCanalMessages:

- Las relaciones con users y canals indican que un mensaje de canal debe ser enviado por un usuario y pertenecer a un canal existente.

FavoriteAssets:

- Las relaciones con users y assets indican que un activo favorito debe pertenecer a un usuario y ser un activo existente.

SocialTradingFollowers:

- Las relaciones con users y briefcase indican que un seguidor debe ser un usuario, al igual que el usuario seguido, y la cartera debe ser válida.

DashboardsBriefcase:

- Las relaciones con dashboardsType y briefcase indican que un panel de control de cartera debe asociarse a un tipo de panel de control y una cartera existente.

Taxes:

- La relación con countrys indica que un impuesto debe asociarse a un país existente.

2.6. Diagrama de arquitectura

Esta plataforma contendrá el modelo arquitectónico por capas. Los principales motivos para utilizar este patrón es separar las lógicas de la parte de la interfaz gráfica de la plataforma de la lógica de la parte de negocio y de persistencia de esta. Por lo tanto, tendremos diferentes capas las cuales se encargarán de diferentes funciones, además se encontrarán totalmente desacopladas, lo cual facilita el mantenimiento de la aplicación en un posible futuro, ya que se va a utilizar Vue 3 para realizar el frontend, pero en un futuro se querrá crear otros tipos de frontend para aplicaciones móvil como puede ser NativeScript o cualquier otro que puede y esto no afectaría al resto de capas de la aplicación.

La arquitectura por capas como su nombre indica se estructura por capas y cada una de ellas se encarga de una función concreta dentro de la aplicación. Habitualmente se utilizan 3 capas: Aplicación, Dominio, Infraestructura y Base de datos, aunque en la mayoría de casos estas dos últimas capas se juntan en una sola, debido a la utilización de lógica de persistencia como HSQL o JSQL está incorporada en los componentes de la capa de dominio. Por otro lado, hay que mencionar que la comunicación entre ellas es inmediata, ya que la capa de aplicación solo se comunica con la de dominio y viceversa, la capa de dominio solo se comunica con la de infraestructura y viceversa.

Aplicación: Se encarga de la vista a mostrar al usuario, así como las interacción y solicitudes del usuario las cuales se trasladarán a la capa de dominio.

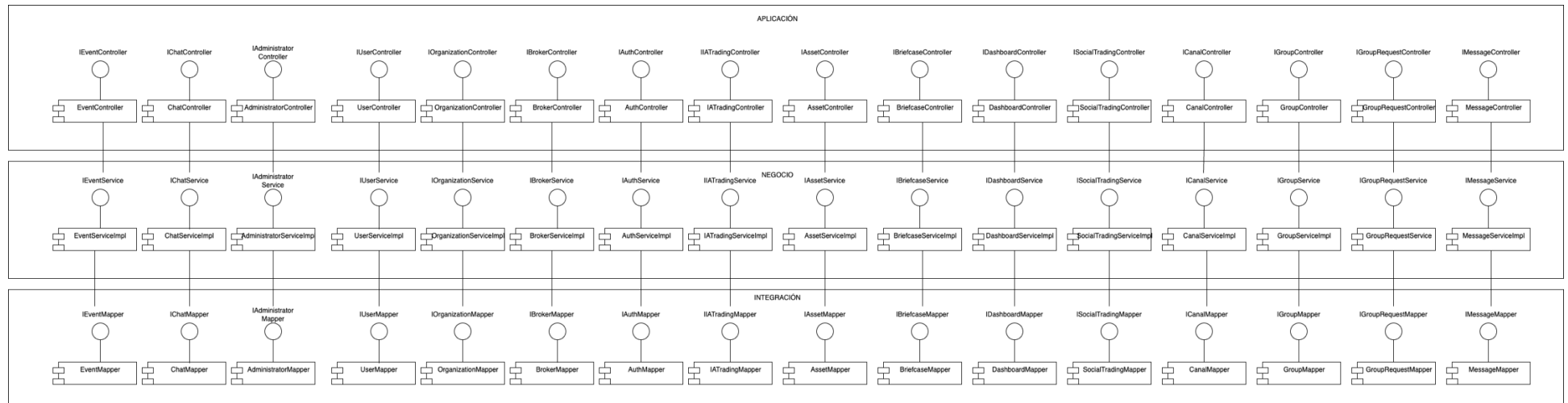
Dominio: Se encarga de toda la lógica de negocio y de trasladar sus peticiones para la obtención o envío de datos con la capa de infraestructura.

Infraestructura: Se encarga se ocupa de gestionar los datos y de guardarlos en la base de datos.

Por otra parte, la elección de este patrón arquitectónico es que es bastante sencillo de implementar y es una gran elección para plataformas pequeñas como aplicaciones web, ya que nuestro caso es una plataforma web con poca lógica de frontend.

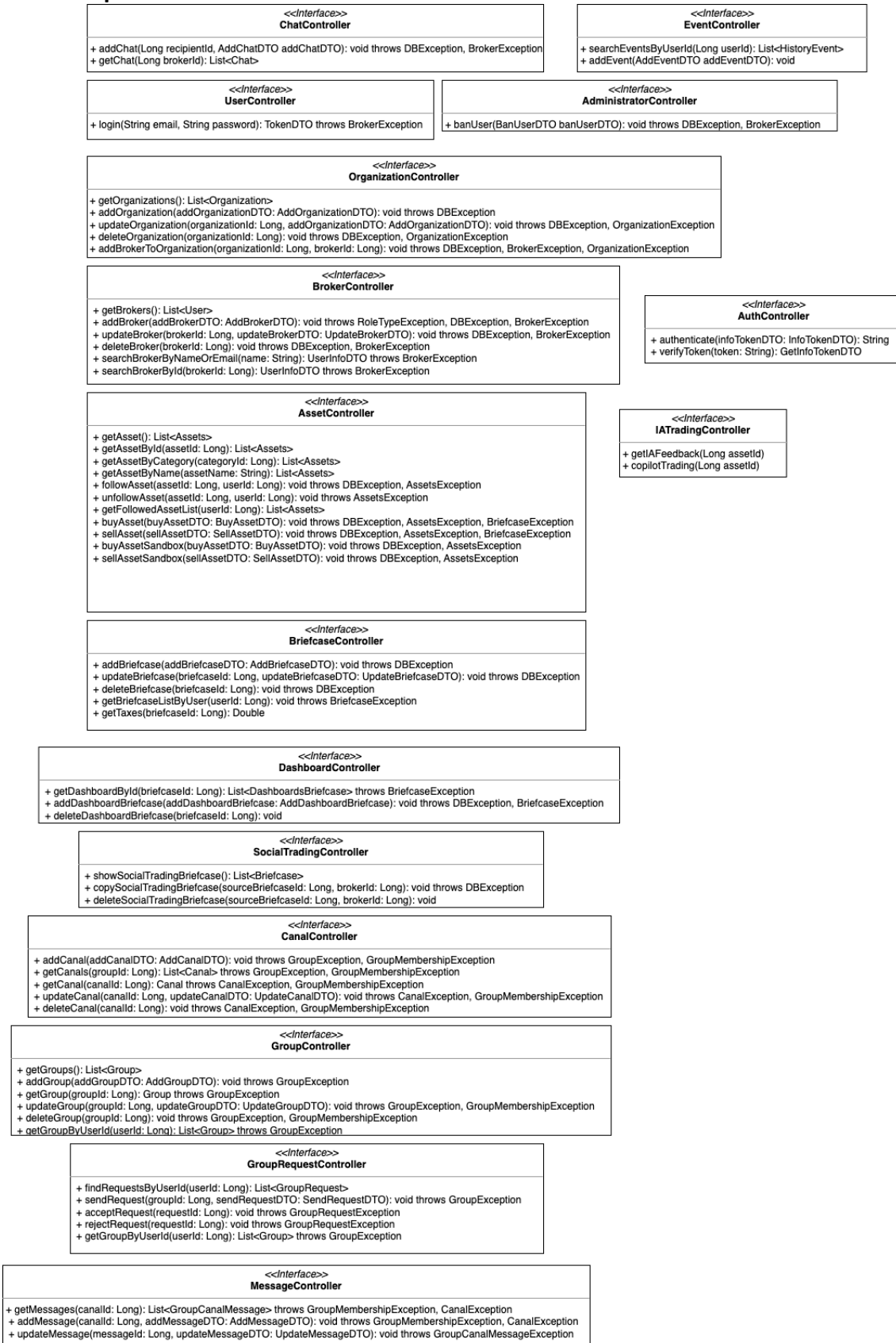
Finalmente, al tener un plazo de tiempo ajustado para el diseño y la implementación de aplicación hace que esta arquitectura sea ideal para aplicarla a nuestro proyecto.

Por lo tanto, el diagrama de componentes nos queda de la siguiente manera:

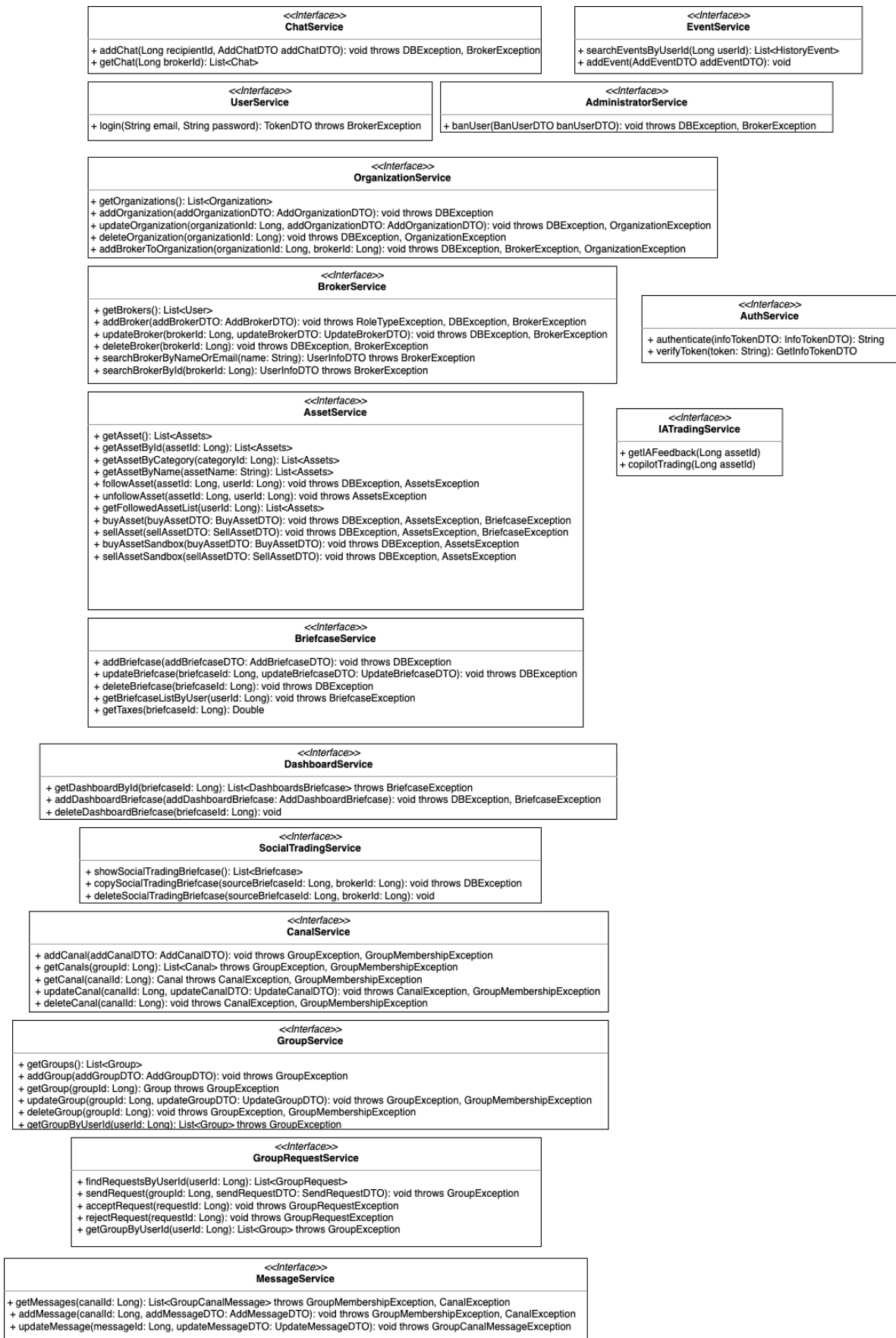


A continuación, vamos a ver el contenido de las diferentes interfaces:

Capa de Presentación



Capa de Negocio



Capa de Infraestructura

<p style="text-align: center;"><<Interface>> CommonMapper</p> <p>+ delete(ID id): int + save(E Entity): int + findById(ID id): E + findAll(): List<E> + findByFilter(E Entity): List<E> + update(E Entity): int</p>
<p style="text-align: center;"><<Interface>> ChatMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> UserMapper extends BaseModel</p> <p>findByEmail(String email): User</p>
<p style="text-align: center;"><<Interface>> OrganizationMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> AssetMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> BriefcaseMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> DashboardMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> SocialTradingMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> CanalMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> GroupMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> GroupRequestMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> MessageMapper extends BaseModel</p>
<p style="text-align: center;"><<Interface>> EventMapper extends BaseModel</p>

Diagrama de arquitectura de microservicios

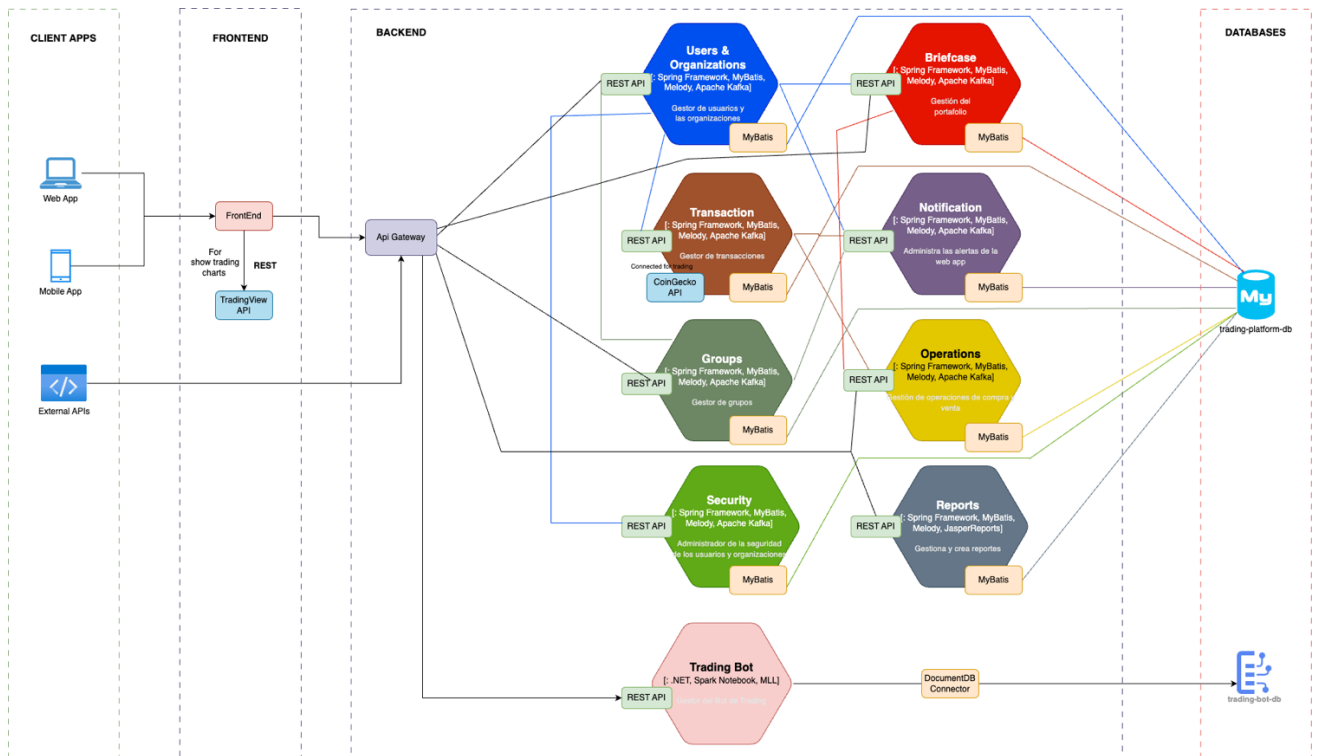


Ilustración 55 - Diagrama de arquitectura de microservicios

Diagrama de arquitectura on-premise

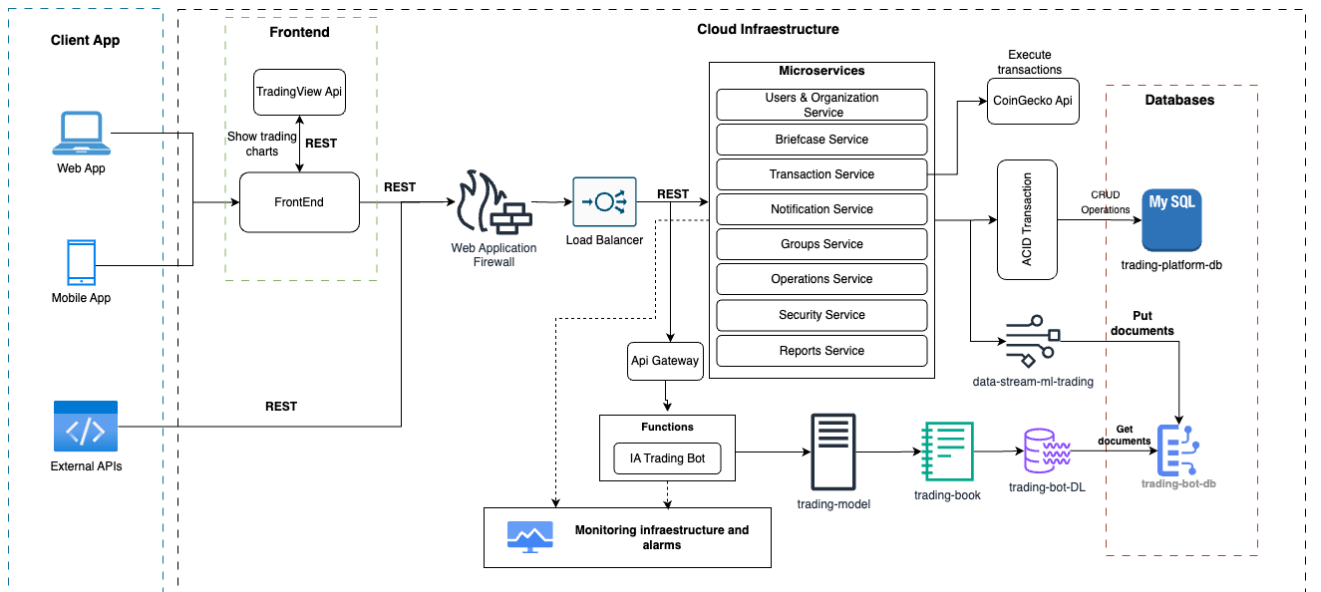


Ilustración 56 - Diagrama de arquitectura on-premise

Diagrama de arquitectura en Azure Cloud

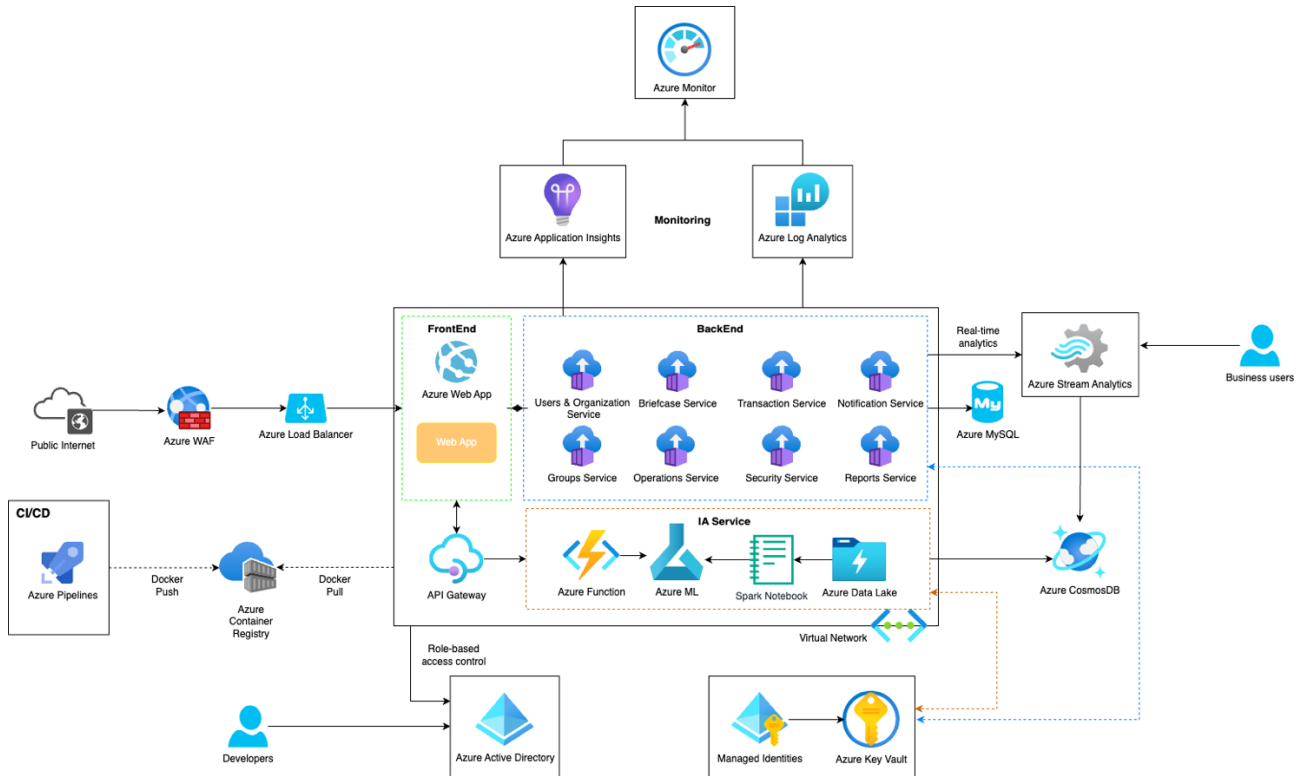


Ilustración 57 - Diagrama de arquitectura en Azure Cloud

3. Implementación

3.1. Tecnologías utilizadas

BACKEND

Spring

Spring es un framework de Java, que nos ayuda bastante ofreciéndonos una serie de componentes y utilidades. Algunas de sus características principales son la inyección de dependencias, el uso del patrón Modelo – Vista – Controlador o bien el uso de ORM para el acceso de datos y persistencias JPA. Por otro lado, permite el uso ágil de gestores de dependencias como puede ser Maven o Gradle.



Ilustración 58 - Logotipo de spring

Spring Boot

Spring Boot es una extensión del proyecto Spring que nos ayuda a simplificar radicalmente el desarrollo de aplicaciones Java. Por defecto, proporciona configuraciones sensatas y un conjunto de herramientas integradas que permiten a los desarrolladores crear aplicaciones empresariales de manera más rápida y sencilla.

Spring Boot se centra más en las convenciones que en la configuración. Esto significa que muchas tareas comunes se pueden completar con una configuración mínima, lo que permite a los desarrolladores centrarse en la lógica empresarial en lugar de en la configuración.



Ilustración 59 - Logotipo de spring boot

MyBatis

Mybatis es un marco de persistencia que proporciona una solución simplificada y flexible para el acceso a bases de datos relacionales en entorno de Java. A través de su enfoque declarativo, MyBatis nos permite mapear objetos Java a operaciones SQL sin la necesidad de escribir código JDBC repetitivo.

Una de las principales características de MyBatis es su archivo de configuración XML, que define la relación entre las clases Java y las sentencias SQL correspondientes. Esto nos facilita la creación y mantenimiento de consultas SQL correspondientes. Esto facilita la creación y mantenimiento de consultas SQL de manera centralizada, mejorando la legibilidad del código y reduciendo la complejidad asociada con la interacción directa con la base de datos.



Ilustración 60 - Logotipo de MyBatis

Java Melody

JavaMelody es una herramienta de supervisión y análisis de rendimiento para aplicaciones Java que en nuestro caso nos ayuda a controlar el rendimiento de los microservicios. Permite realizar un seguimiento detallado del rendimiento de las aplicaciones, identificar cuellos de botella y evaluar métricas clave como el tiempo de respuesta, el uso de recursos y la frecuencia de errores, proporcionando una visión valiosa para la optimización y la toma de decisiones informadas en el desarrollo y despliegue de aplicaciones Java.

Se puede acceder en cada microservicio mediante el navegador con la siguiente url:
<http://localhost:{puertoMicroservicio}/monitoring>



Ilustración 61 - Logotipo de java melody

Swagger 3 (OpenApi 3)

Swagger es una herramienta que utilizamos para simplificar y agilizar el proceso de desarrollo de las APIs. Con Swagger, podemos diseñar, documentar y probar las APIs de manera eficiente, facilitando la colaboración entre equipos de desarrollo y asegurando que los microservicios sean comprensibles y fáciles de usar para otros desarrolladores.

Se puede acceder en cada microservicio mediante el navegador con la siguiente url: <http://localhost:{puertoMicroservicio}/swagger-ui/index.html>



Ilustración 62 - Logotipo de swagger

Lombok

Lombok es una biblioteca para Java el cual, mediante el uso de anotaciones, automatiza la generación de código repetitivo, como getters, setters y constructores, reduciendo la cantidad de código y mejorando la legibilidad, lo que nos contribuye a una mayor productividad en el desarrollo de software.



Ilustración 63 - Logotipo de project lombok

MySQL

MySQL es un sistema de gestión de bases de datos relacional que hemos utilizado en este proyecto. Personalmente, lo encuentro confiable y eficiente para almacenar y recuperar datos en aplicaciones web y sistemas empresariales. Con MySQL, podemos diseñar esquemas de bases de datos, ejecutar consultas complejas y garantizar un rendimiento sólido en el manejo de datos, lo que lo convierte en la mejor elección para cumplir con las necesidades de persistencia de datos en entornos Java.



Ilustración 64 - Logotipo de MySql

JasperReports

JasperReports es una herramienta que hemos utilizado en el microservicio de reports para la generación de informes. Permite diseñar y generar informes complejos y detallados de manera eficiente, facilitando la presentación de datos de manera estructurada y visualmente atractiva. Con JasperReports, podemos integrar fácilmente informes en las aplicaciones Java, proporcionando a los usuarios una forma efectiva de analizar y entender grandes conjuntos de datos.

Spring Boot Starter Mail

Spring Boot Starter Mail es una funcionalidad que se ha integrado en el microservicio de notificaciones. Esta característica facilita la configuración y el uso de servicios de correo electrónico, permitiéndonos enviar correos electrónicos de manera eficiente y simplificada. Con Spring Boot Starter Mail, podemos gestionar fácilmente la configuración del servidor de correo, el envío de correos y la personalización de plantillas, lo que agiliza el desarrollo y mejora la capacidad de enviar notificaciones y comunicaciones por correo electrónico en mis aplicaciones.

Spring Boot Security

Spring Boot Security es una herramienta que hemos integrado en todos los microservicios basados en Spring Boot. Se utiliza para gestionar la seguridad y la autenticación, proporcionando funciones de inicio de sesión, autorización y protección contra amenazas comunes. Con Spring Boot Security, podemos implementar fácilmente características de seguridad robustas en mis aplicaciones, asegurando que los usuarios accedan de manera segura y que los recursos estén protegidos de manera efectiva. Esta funcionalidad simplifica la implementación de prácticas de seguridad en los proyectos, mejorando la confiabilidad y la protección de los datos sensibles.

FRONTEND

Vue 3

Vue 3 es un marco de JavaScript que hemos explorado y aplicado en el desarrollo de interfaces de usuario interactivas. Lo hemos elegido por su enfoque reactivo y su capacidad de construir aplicaciones web eficientes y modernas. Vue 3 nos permite estructurar de manera clara y modular el código del cliente, facilitando la creación de componentes reutilizables y la gestión del estado de la aplicación de manera más intuitiva. Con su sistema de reactividad avanzado y las mejoras en el rendimiento, Vue 3 se ha convertido en una elección valiosa para crear experiencias de usuario dinámicas y altamente responsivas.

3.2. Repositorios del proyecto

A continuación, se adjuntan todos los repositorios del proyecto BlackTradingBot.

FRONTEND

<https://github.com/IEdgaar/btb-frontend>

BACKEND

Reports: <https://github.com/IEdgaar/btb-reports-service>

Operations: <https://github.com/IEdgaar/btb-operations-service>

Security: <https://github.com/IEdgaar/btb-security-service>

Transaction: <https://github.com/IEdgaar/btb-transaction-service>

Users/Organization: <https://github.com/IEdgaar/btb-users-organization-service>

Notification: <https://github.com/IEdgaar/btb-notification-service>

Groups: <https://github.com/IEdgaar/btb-groups-service>

Briefcase: <https://github.com/IEdgaar/btb-briefcase-service>

INFRAESTRUCTURA

IA: <https://github.com/IEdgaar/btb-azure-infra>

3.3. Características de la aplicación

En este apartado, se procede a mencionar a grandes rasgos las características implementadas en la aplicación.



Rol Bróker

The screenshot shows the registration page of the BlackTradingBot application. At the top, there is a dark header with the text "BlackTradingBot" on the left and "Sign Up" and "Log In" buttons on the right. The main content area is a white box titled "Registrate". Inside this box, there is a blue button labeled "Soy una organización". Below this, there are several input fields: "Nombre" (with the placeholder "Escribe tu nombre completo"), "Apellido" (with the placeholder "Escribe tu apellido"), "Correo electrónico" (with the placeholder "Escribe tu correo electrónico"), "Fecha de nacimiento" (with a calendar icon), and "Género" (with a dropdown arrow). There are also two password fields: "Contraseña" and "Confirma contraseña". Below the password fields, there is a small note: "Debe tener al menos un caracter especial, una minúscula, una mayúscula y un número". At the bottom of the registration form, there is a blue "Registrate" button.

Ilustración 65 - Registro de usuarios Bróker/Organización

The screenshot shows the login page of the BlackTradingBot application. At the top, there is a dark header with the text "BlackTradingBot" on the left and "Sign Up" and "Log In" buttons on the right. The main content area is a white box titled "Iniciar Sesión". Inside this box, there are two input fields: "Correo electrónico" (with the placeholder "Escribe tu correo electrónico") and "Contraseña". Below the input fields, there are two buttons: a blue "Iniciar Sesión" button and a red "Cancelar" button.

Ilustración 66 - Inicio de sesión

The screenshot shows a "Crear chat" dialog box in the BlackTradingBot application. The dialog is a white box with a close button (X) in the top right corner. It has a title "Crear chat" and two input fields: "Introduce email del broker" and "Message". Below the input fields, there are two buttons: a grey "Cancel" button and a blue "Ok" button. The background of the application is dark grey and shows a "Chats" section with a "Crear chat" button.

Ilustración 67 - Creación de chat con otro usuario

Chats

Crear chat

Conversaciones

edouton21@uoc.edu

Mensaje: Prueba mensaje

Ilustración 68 - Visualizar chats

Ajustes de la cuenta

Nombre

Apellido

Fecha de Nacimiento

1999-11-15T00:00:00.000+00:00

Género

Masculino ▾

Modificar

Eliminar cuenta

Ilustración 69 - Ajustes de la cuenta

Comunidad

Grupos

Crear grupo






	wefweewfe wfewfewef	Descripción: qwfewfef Entrar
	fwefwe fewffe	Descripción: fdwefwef Entrar
	fwefwe fewffe	Descripción: fdwefwef Entrar
	fwefwe fewffe	Descripción: fdwefwef Entrar
	weffw wfeff	Descripción: ewffwfe Entrar

Ilustración 70 - Visualizar grupos

Crear grupo

Nombre

Title

Description

Cancel Ok

Ilustración 71 - Crear grupo

Rol Organización

BlackTradingBot Home Portafolio Mercado Comunidad IA Cuenta ▾

Panel de Organización

Añadir broker a organización

Búsqueda:

l

Encontrar

Información

Ilustración 72 - Añadir usuarios a organización

Ajustes de la cuenta

Nombre

Descripción Organización

Escribe aquí tu descripción

Modificar Eliminar organización

Ilustración 73 - Ajustes de la cuenta

¿Quieres eliminar la organización?

¿Quieres eliminar la organización?

Cancelar Eliminar

Ilustración 74 - Eliminar la cuenta

Rol Administrador

BlackTradingBot Home Portafolio Mercado Comunidad IA Cuenta ▾

Panel de administración

Búsqueda:
edouton21231@uoc.edu

Buscar usuario

Bloqueo de usuario

Información del Usuario

Nombre: Edgar
Apellido: Doutón
Fecha de nacimiento: 1999-11-15T00:00:00+00:00

Fecha finalización bloqueo

Motivo

Bloquear

Ilustración 75 - Bloquear usuario

Home Portafolio Mercado Comunidad IA

Confirmar bloqueo

¿Estás seguro de que deseas bloquear el usuario?

Cancel Ok

Ilustración 76 - Confirmación de bloqueo

4. Conclusiones

Este proyecto ha sido un gran viaje de aprendizaje que me ha hecho experimentar una amplia gama de emociones y reflexiones. Empezando por la satisfacción inicial hasta la frustración, he comprendido las complejidades y desafíos asociados a proyectos de esta magnitud.

Una de las conclusiones fundamentales es la autoevaluación honesta sobre mi nivel actual de preparación. Reconozco que aún estoy en una fase temprana de mi desarrollo profesional y que hay mucho por aprender. La necesidad de una visión más holística desde la planificación inicial destaca la importancia de considerar todos los puntos de diseño y requisitos desde el principio para evitar cambios constantes.

La incapacidad percibida para concebir propuestas novedosas resalta mi conciencia sobre la importancia de fomentar la creatividad en futuros proyectos. En la implementación técnica, las dificultades experimentadas, especialmente con el manejo de datos y tecnologías web, indican la necesidad de práctica continua y el reconocimiento de que algunos contratiempos solo se superan con la experiencia.

La utilidad de los conocimientos adquiridos durante mi grado, especialmente en diseño y planificación, se destaca como un activo valioso. Aunque el proyecto ha logrado cumplir con la mayoría de los objetivos, se reconoce humildemente que la aplicación está en una fase beta y hay margen para mejoras.

La falta de pericia técnica y las limitaciones de tiempo han influido en la implementación completa de las funcionalidades previstas. Aunque estoy satisfecho con el progreso alcanzado, la conclusión es que hay oportunidades para mejorar, tanto en términos de habilidades técnicas como en la gestión del tiempo.

En resumen, este proyecto ha sido una experiencia valiosa que ha contribuido significativamente a mi desarrollo. A pesar de los desafíos y las áreas de mejora identificadas, estoy contento con lo logrado y veo este proyecto como un trampolín para futuros proyectos más ambiciosos, donde podré aplicar y fortalecer aún más mis habilidades adquiridas.

5. Glosario

Caso de uso: La técnica de casos de uso, empleada en ingeniería de requisitos, representa la interacción entre el sistema y sus actores en distintos escenarios, ofreciendo una visión detallada de cómo se aplican en diversas situaciones.

Diagrama de clases: Un diagrama de clases, dentro del contexto de la Ingeniería del Software y utilizando el lenguaje de modelado UML, es una representación visual que exhibe la estructura estática de una aplicación, detallando las clases con sus atributos y métodos, así como las relaciones existentes entre ellas dentro del sistema.

Prototipado: El prototipado es una técnica que implica la representación de diversas pantallas en forma de bocetos o esbozos que corresponden a la interfaz gráfica de una aplicación. Su objetivo es proporcionar una visualización anticipada de la estructura potencial de las distintas opciones y elementos de la aplicación.

Java: Java es un lenguaje de programación introducido por Sun Microsystems en 1995, caracterizado por su fuerte tipado, paradigma orientado a objetos y capacidad de ser ejecutado de manera multiplataforma.

SPA: SPA, o Single Page Application, es un tipo de aplicación web que se presenta en una única página para emular la apariencia y comportamiento de una aplicación de escritorio. El propósito es proporcionar una experiencia de usuario ágil, fluida y satisfactoria al consolidar toda la interacción en una sola página.

Diagrama de Gantt: Un Diagrama de Gantt es una técnica utilizada para la planificación de proyectos que visualiza de manera gráfica las distintas tareas, sus respectivos plazos, los hitos y los recursos empleados en un proyecto.

Frontend: Frontend se refiere al desarrollo de la parte del cliente, concentrándose en la creación de la interfaz gráfica de la aplicación web que interactuará directamente con el usuario.

Backend: Backend hace referencia al desarrollo en la parte del servidor, enfocándose en la implementación de la lógica de negocio de la aplicación web, así como en la gestión de bases de datos y el manejo de las peticiones provenientes del cliente.

CRUD: CRUD son las siglas de Create, Read, Update y Delete, representando las operaciones fundamentales que pueden realizarse sobre los datos almacenados en una base de datos.

API: API, o Interface de Programación de Aplicaciones (por sus siglas en inglés, Application Programming Interface), se refiere a un conjunto de normas y procedimientos que facilitan la relación y comunicación entre distintos componentes de software. Proporciona una interfaz detallada y documentada de los métodos y datos disponibles para su utilización por otras aplicaciones.

Framework: Un Framework es un conjunto de librerías predeterminadas que, al ser personalizadas, sirven como base para el desarrollo de aplicaciones, ahorrando considerablemente tiempo. Proporciona características genéricas implementadas y validadas, permitiendo al programador enfocarse en los aspectos específicos de la aplicación que necesita implementar.

Social trading: Social Trading se refiere a una forma de inversión en la que los inversores tienen la oportunidad de observar, seguir y replicar automáticamente las operaciones financieras de traders exitosos. En este enfoque, se crea una comunidad en la que los participantes pueden compartir estrategias, experiencias y conocimientos financieros.

Bróker: Un bróker, también conocido como corredor, es un intermediario financiero que facilita la compra y venta de activos financieros en nombre de sus clientes. Los brokers actúan como intermediarios entre los compradores y vendedores en los mercados financieros, ejecutando órdenes y proporcionando servicios relacionados con la inversión, como el asesoramiento financiero y la investigación de mercado.

Dashboard: Un dashboard, o panel de control, es una interfaz visual que proporciona información en tiempo real y un resumen visual de datos clave, métricas o indicadores relevantes para un usuario o un proceso específico. Por lo general, se presenta de manera gráfica y accesible, permitiendo a los usuarios monitorear y comprender rápidamente el rendimiento de un sistema, proceso o conjunto de datos.

Sandbox: Un sandbox es un entorno de pruebas aislado y controlado que se utiliza para probar y experimentar con software, aplicaciones o configuraciones sin afectar el entorno de producción. Es una especie de espacio seguro donde los usuarios pueden realizar pruebas sin riesgo de dañar el sistema principal.

Bot: Un bot, abreviatura de "robot", en el contexto de software y tecnología, se refiere a un programa informático diseñado para realizar tareas automatizadas de manera autónoma. Los bots pueden ejecutar funciones específicas de forma repetitiva o responder a comandos predefinidos.

6. Bibliografía

Baeldung – Spring Kafka

<https://www.baeldung.com/spring-kafka>

Spring Initializr

<https://start.spring.io/>

Medium – Spring Security JWT authentication and authorization

<https://medium.com/code-with-farhan/spring-security-jwt-authentication-authorization-a2c6860be3cf>

Baeldung – Spring Boot Api Key and secret

<https://www.baeldung.com/spring-boot-api-key-secret>

Baeldung – Spring Jasper

<https://www.baeldung.com/spring-jasper>

Baeldung – Spring Email

<https://www.baeldung.com/spring-email>

Baeldung – Spring Security – security none, filters none, Access permitAll

<https://www.baeldung.com/security-none-filters-none-access-permitAll>

Spring Boot - Documentación

<https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>

Lombok - Documentación

<https://projectlombok.org/features/>

Swagger 3 - Documentación

<https://swagger.io/docs/>

7. Anexos

7.1. Manual de procedimiento para instalación del proyecto

En esta parte de la memoria del proyecto, debido al coste que tiene mantener toda esta infraestructura en un proveedor de la nube para tener alojada la base de datos, los ejecutables de Spring con el servidor de Apache embebido, el frontend Vue y todos los elementos de IA.

Por el momento, tendremos que realizar la instalación y ejecución del proyecto en local. En primer lugar, deberemos ir a los repositorios que se han adjuntado en esta memoria.

Empezamos con el Frontend

Abrimos el terminal y realizamos un git clone del único repositorio. Una vez hecho esto se habrá creado la carpeta btb-frontend.

Como se trata de un proyecto de frontend creado en Vue 3, esto implica tener que instalar Node.js

<https://nodejs.org/es>

Tras instalar Node.js, deberemos instalar Vue CLI.

<https://es.vuejs.org/v2/guide/installation>

En el terminal debemos escribir:

```
npm install vue
```

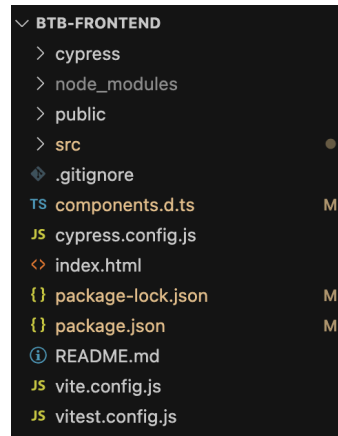
Después de haber instalado Node.js y Vue, ya estamos en disposición de utilizar Vue en nuestro equipo.

El siguiente paso será abrir la carpeta del proyecto con un editor, en nuestro caso recomendamos Visual Studio Code y abrimos la carpeta del frontend.

Una vez abierto, deberemos de abrir la terminal y situarnos en la carpeta del proyecto para realizar la instalación de las dependencias.

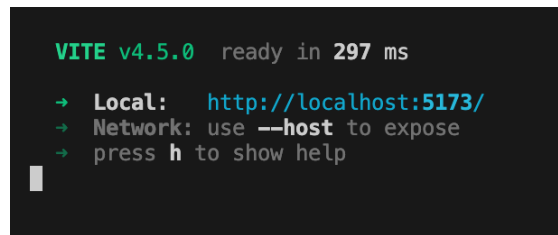
```
npm install
```

Una vez hecho esto, tendremos la siguiente estructura con la carpeta `node_modules` creada.



Ahora, ya podemos iniciar Vue.

`npm run dev`



En el caso de que queramos hacer una puesta en producción, deberemos de ejecutar el comando `build`.

`npm run build`

Tras esto se nos abrirá ya la pantalla principal de la aplicación en el navegador con la dirección localhost:5173



Procedemos con la parte de backend

Abrimos el terminal y realizamos un git clone de todos los repositorios donde se encuentran los microservicios de backend, en total son 8.

A continuación, debemos de instalar el JDK de Java

<https://www.oracle.com/es/java/technologies/downloads/>

Por otro lado, se recomienda instalar un IDE, nosotros recomendamos IntelliJ IDEA Ultimate, pero aun así existe la opción gratuita Spring Tool Suite.

Una vez tenemos el IDE, habrá que instalar el gesto de dependencias Maven, así como configurar las variables de entorno.

<https://maven.apache.org/download.cgi>

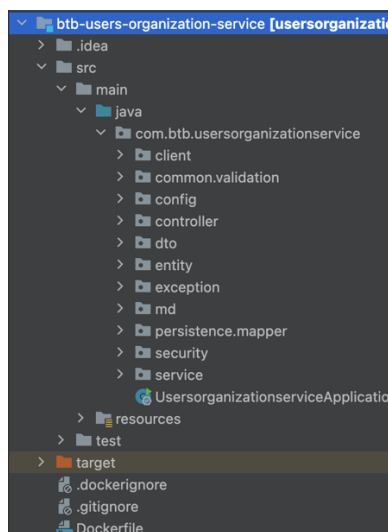
A continuación, deberemos instalar la base de datos, en este caso será MySQL, en concreto MySQL Community Server.

<https://dev.mysql.com/downloads/mysql/>

Finalmente, recomendamos una interface gráfica para trabajar con el servidor de base de datos, desde nuestro punto de vista recomendamos MySQL Workbench o DBeaver.

<https://dev.mysql.com/downloads/workbench/>

Una vez tenemos todo el entorno preparado, abrimos todas las carpetas de los microservicios con el IDE y obtendríamos lo siguiente:



Posteriormente, tendremos que configurar en cada properties de cada microservicio usuario, contraseña y puerto de la base de datos.

Una vez hecho esto, deberemos de lanzar los scripts que se encuentran en la carpeta `init_db` dentro de infraestructura para crear todo el modelo de base de datos y finiquitar esta parte.

Con esto todo hecho, nos iremos al terminal, a la carpeta de cada microservicio para crear el ejecutable del proyecto. Esta parte es opcional ya que lo podemos ejecutar desde el IDE.

```
mvn clean package
```

Una vez obtenemos el jar, deberíamos de ir a la ruta donde se ha generado el archivo jar y ejecutarlo con el siguiente comando.

```
java -jar microservicio
```

Con esto, tenemos toda la aplicación funcionando, ahora tan solo quedaría ir al frontend que está en ejecución.

En el caso que se quiera utilizar las funciones de bot con IA, se deberá de crear un Azure Function y configurar el código .NET que se encuentra dentro de infraestructura.

Si se desea realizar operaciones financieras (ya sea TEST o entorno REAL) hará falta especificar las API KEY's de **Binance**, **CoinGecko** y **TradingView**.