

## MEMORIA ESCRITA DEL PROYECTO

CFGS Desarrollo de Aplicaciones Multiplataforma

# TradingJournal

**Autor:** Khirann Moreno Orrego

**Tutor:** Isabel Duarte Tosso

**Fecha de entrega:** 28/04/2024

**Convocatoria:** 1s2324

**Documentos del proyecto:**

[https://drive.google.com/drive/folders/1FgHbLmkJIU0FX2XwSWofyFmcZplVapqf?usp=drive\\_link](https://drive.google.com/drive/folders/1FgHbLmkJIU0FX2XwSWofyFmcZplVapqf?usp=drive_link)



## Índice de contenidos

1. Introducción (3-4 Páginas).....	4
1.1. Motivación.....	4
1.2. Abstract.....	5
1.3. Objetivos propuestos (generales y específicos).....	6
2. Estado del Arte (4-5 páginas) .....	8
3. Metodología usada (2-4 páginas).....	11
4. Tecnologías y herramientas utilizadas en el proyecto(2-3 páginas) .....	13
5. Planificación, Diagnóstico y Contexto Laboral (3-4 páginas) .....	15
1.1 Planificación.....	15
2.2 Diagnóstico .....	18
Contexto Laboral .....	18
6. Análisis del proyecto (8-16 páginas) .....	18
Modelo de Datos Detallado para TradingJournal .....	20
7. Diseño del proyecto (6-14 páginas).....	21
8. Despliegue y pruebas (4-10 páginas) .....	26
9. Conclusiones (1-2 páginas).....	28
Conclusiones .....	26
10. Vías futuras (1-2 páginas).....	30
1. Gestión de Usuarios y Seguridad .....	30
2. Personalización y Comunicación .....	30
3. Seguridad y Privacidad de los Datos .....	30
4. Recordatorios y Alertas .....	30
5. Mejoras en la Experiencia del Usuario .....	31
6. Integración de Análisis de Datos Avanzados .....	31

7. Ampliación de la Compatibilidad y la Accesibilidad.....	31
8. Funcionalidades de Networking y Comunidad .....	32
9. Expansión del Soporte de Datos.....	32
10. Feedback y Soporte al Usuario Mejorado .....	32
11. Bibliografía/Webgrafía (1-2 páginas).....	32
12. Anexos.....	35
1.1. Manual de Usuario .....	35
1.2. Manual de Instalación de TradingJournal .....	45
Glosario de Términos y Siglas .....	47

# 1. Introducción (3-4 Páginas)

## 1.1. Motivación

La creación de esta aplicación de TradingJournal responde a una serie de tendencias en el ámbito del trading, la tecnología y las necesidades del mercado, lo que demuestra su viabilidad y oportunidad como proyecto.

Desde una perspectiva de mercado financiero, existe una creciente demanda de herramientas que permitan a los traders gestionar y analizar sus operaciones de manera efectiva. Esta demanda se debe a la necesidad de mejorar el rendimiento comercial y tomar decisiones más informadas. La aplicación de TradingJournal proporcionará una plataforma integral para registrar, analizar y mejorar el desempeño de las operaciones, lo que ayudará a los traders a identificar patrones, áreas de mejora y estrategias exitosas, lo que les permitirá su rendimiento y tomar decisiones más informadas.

Tecnológicamente, Con la creciente digitalización en todas las áreas de la vida, incluido el trading, existe una tendencia hacia el uso de herramientas digitales para gestionar operaciones y tomar decisiones comerciales. Esta aplicación de TradingJournal se alinea con esta tendencia al ofrecer una solución digital completa para la gestión de operaciones.

Enfoque en la educación y mejora continua, muchos traders están interesados en mejorar sus habilidades y conocimientos en el mercado. Una aplicación que no solo registre las operaciones, sino que también proporcione análisis detallados y herramientas para calcular riesgos y beneficios puede servir como una herramienta educativa invaluable.

En cuanto a las tendencias de mercado financiero, la gestión emocional y la psicología del trading son aspectos cada vez más reconocidos y valorados. La aplicación de TradingJournal permitirá a los usuarios registrar y analizar su comportamiento emocional durante las operaciones, lo que les ayudará a ser más conscientes de sus reacciones emocionales eh identificar patrones ya que esto mejorará su capacidad de tomar decisiones comerciales objetivas y desarrollar estrategias para gestionarlas de manera efectiva. Además, la integración de herramientas de gestión de riesgos, como una calculadora de tamaño de lote, ayudará a los traders a establecer límites claros y gestionar su capital de manera más eficiente.

### **Ejemplos concretos:**

Un ejemplo concreto de cómo la aplicación beneficiaría a los traders es mediante la función de análisis detallado. Supongamos que un trader realiza una serie de operaciones y desea

identificar patrones en su desempeño. Utilizando las herramientas de análisis avanzadas de la aplicación, el trader puede visualizar fácilmente su historial de operaciones en forma de gráficos interactivos y estadísticas de rendimiento, lo que le permite identificar áreas de fortaleza y áreas de mejora en su estrategia de trading.

Otro Ejemplo podría ser el uso de la calculadora de riesgo. Supongamos que un trader quiere realizar una operación, pero no está seguro del tamaño del lote o el nivel de riesgo que debería asumir. Utilizando la calculadora integrada en la aplicación, el trader puede ingresar los parámetros de la operación y obtener rápidamente información sobre el tamaño del lote, el riesgo potencial, lo que le permite tomar decisiones más informadas y gestionar su capital de manera más efectiva.

En resumen, la creación de esta aplicación de TradingJournal se basa para ayudar a los traders a mejorar su rendimiento, gestionar sus emociones y riesgos de manera efectiva. Estos factores, entre otros, respaldan la motivación para crear una aplicación de este tipo.

## 1.2. Abstract

The creation of this TradingJournal application responds to a series of trends in the trading, technology, and market needs domains, demonstrating its feasibility and opportunity as a project.

From a financial market perspective, there is a growing demand for tools that allow traders to manage and analyze their operations effectively. This demand stems from the need to improve commercial performance and make more informed decisions. The TradingJournal application will provide a comprehensive platform for recording, analyzing, and improving the performance of operations, helping traders identify patterns, areas for improvement, and successful strategies, thereby enhancing their performance and enabling more informed decision-making.

Technologically, with the increasing digitization in all areas of life, including trading, there is a trend towards the use of digital tools to manage operations and make commercial decisions. This TradingJournal application aligns with this trend by offering a complete digital solution for operations management.

Focus on education and continuous improvement, many traders are interested in enhancing their skills and knowledge in the market. An application that not only records operations but

also provides detailed analysis and tools for calculating risks and benefits can serve as an invaluable educational tool.

Regarding financial market trends, emotional management and trading psychology are increasingly recognized and valued aspects. The TradingJournal application will allow users to record and analyze their emotional behavior during operations, helping them become more aware of their emotional reactions and identify patterns, thereby improving their ability to make objective trading decisions and develop strategies to manage them effectively. Additionally, the integration of risk management tools, such as a lot size calculator, will help traders set clear limits and manage their capital more efficiently.

Concrete examples:

A specific example of how the application would benefit traders is through the detailed analysis function. Suppose a trader conducts a series of operations and wants to identify patterns in their performance. Using the advanced analysis tools of the application, the trader can easily visualize their operation history in the form of interactive graphs and performance statistics, allowing them to identify strengths and areas for improvement in their trading strategy.

Another example could be the use of the risk calculator. Suppose a trader wants to carry out an operation but is unsure of the lot size or level of risk they should assume. By using the integrated calculator in the application, the trader can enter the parameters of the operation and quickly obtain information about the lot size, risk potential, enabling them to make more informed decisions and manage their capital more effectively.

In summary, the creation of this TradingJournal application is aimed at helping traders improve their performance, manage their emotions and risks effectively. These factors, among others, support the motivation to create an application of this type.

### 1.3. Objetivos propuestos (generales y específicos)

La aplicación de TradingJournal que proponemos tiene como objetivo revolucionar la manera en que los traders gestionan, analizan y mejoran su desempeño en el mercado financiero. Nuestros objetivos generales están orientados a proporcionar una plataforma intuitiva y accesible, fomentar la mejora continua del rendimiento comercial, construir una comunidad colaborativa y garantizar la sostenibilidad del modelo de negocio. Estos objetivos reflejan

nuestro compromiso con la calidad, la innovación y el apoyo a los traders en su jornada hacia el éxito.

Al mismo tiempo, nuestros objetivos específicos se centran en abordar las necesidades técnicas y operativas que permitirán alcanzar nuestros objetivos generales. Desde la implementación de herramientas de análisis avanzadas hasta la integración de sistemas de gestión de riesgos, cada objetivo específico está diseñado para establecer una base sólida y funcional para la experiencia del usuario y la creación de alianzas estratégicas también destaca nuestro compromiso de ofrecer un valor real y sostenible para todos los usuarios. En última instancia, nuestro propósito es proporcionar una aplicación que no solo satisfaga las necesidades actuales de los traders, sino que también establezca nuevos estándares en el mundo del trading.

#### Objetivos Generales:

- Fomentar el registro de tus operaciones de trading y emociones durante la operación mediante una plataforma interactiva.

#### Objetivos Específicos:

Para proporcionar una plataforma donde los usuarios puedan analizar la tendencia de sus operaciones, el comportamiento emocional que experimentan en cada operación y una calculadora para calcular el lote de riesgo y beneficio deseado por el usuario, podríamos seguir estos pasos:

- **Análisis de Tendencias de Operaciones:**

- Implementar gráficos interactivos que muestren el rendimiento histórico de las operaciones.
- Permitir a los usuarios filtrar y analizar sus operaciones por diferentes criterios como instrumento financiero, fecha, tipo de operación, etc.
- Proporcionar estadísticas clave como ganancias totales, pérdidas totales, tasa de éxito, etc.

- **Registro del Comportamiento Emocional:**

- Integrar un sistema de registro donde los usuarios puedan registrar sus emociones durante cada operación.
- Generar gráficos que muestren la correlación entre el comportamiento emocional y el rendimiento de las operaciones.

- **Calculadora de Lote de Riesgo:**

- Desarrollar una calculadora que permita a los usuarios ingresar el riesgo deseado en cada operación.

- Calcular automáticamente el tamaño del lote que el usuario debe utilizar para alcanzar el nivel de riesgo.

## 2. Estado del Arte (4-5 páginas)

La historia del trading se remonta a miles de años atrás, desde los mercados antiguos donde se intercambiaban bienes y productos. Sin embargo, el trading moderno como lo conocemos hoy en día se desarrolló a lo largo de siglos, con hitos importantes como la creación de la Bolsa de valores de Ámsterdam en el siglo XVII y la posterior evolución de los mercados financieros en todo el mundo.

En las últimas décadas, ha habido avances significativos en la tecnología del trading. Esto incluye la adopción de sistemas electrónicos de negociación, algoritmos de trading de alta frecuencia, plataformas de trading en línea y el uso de inteligencia artificial y aprendizaje automático para analizar datos y tomar decisiones comerciales.

La era digital ha transformado por completo el trading, la introducción de la tecnología ha llevado a una mayor velocidad de ejecución, acceso a los mercados globales las 24 horas del día, 7 días a la semana, una mayor transparencia en los precios y la transformación del mercado. Además, la digitalización ha democratizado el trading, permitiendo que una amplia gama de personas participe en los mercados financieros.

La historia del trading ha evolucionado hacia una narrativa de innovación continua. A medida que avanza la tecnología y cambian las condiciones del mercado, los traders y las instituciones financieras están constantemente buscando nuevas formas de mejorar sus estrategias comerciales, optimizar la ejecución de órdenes y adaptarse a un entorno comercial en constante cambio.

En el ecosistema de las aplicaciones de Trading, existen diversas opciones tanto de escritorio como en línea que facilitan desde el registro hasta el análisis y gestión de operaciones financieras. Estas herramientas han sido fundamentales para el crecimiento y la evolución del trading, permitiendo a los usuarios mejorar su rendimiento y tomar decisiones informadas con recursos relativamente accesibles.

Aplicaciones de escritorio y en línea:

**MyFxBook:** Ofrece tanto una plataforma en línea como una aplicación de escritorio. La plataforma en línea permite a los usuarios acceder a sus datos desde cualquier lugar con



conexión a internet, mientras que la aplicación de escritorio ofrece una experiencia más integrada para aquellos que prefieren trabajar desde su computadora.

**TraderSyn:** Ofrece una aplicación en línea y una versión de escritorio. Los usuarios pueden elegir entre acceder a la plataforma a través de un navegador web o descargar la aplicación de escritorio para una experiencia más personalizada.

**Edgewonk:** Ofrece una aplicación en línea y una versión de escritorio. Los usuarios pueden acceder a la plataforma desde cualquier navegador web o descargar la aplicación de escritorio para acceder a todas las funciones y datos sin necesidad de una conexión a internet.

Cada una de estas aplicaciones tiene sus propios puntos fuertes y se adapta a distintos flujos de trabajo y niveles de experiencia. Mientras que las aplicaciones de escritorio suelen ofrecer más opciones de personalización y control sobre el proceso de producción, las aplicaciones en línea destacan por su facilidad de uso y acceso desde cualquier lugar, además de la colaboración en tiempo real. Con el auge del trading, estas herramientas continúan evolucionando, incorporando nuevas funcionalidades que responden a las necesidades de una base de usuarios en crecimiento y cada vez más diversa.

Enumeremos las ventajas y desventajas de cada herramienta/aplicación nombrada para hacernos una idea de cómo funcionan este estilo de aplicaciones relacionadas con el mundo del trading.

NOTA: Una desventaja importante a considerar es que todas estas aplicaciones requieren un pago para acceder a todas sus funcionalidades. Si bien algunas ofrecen versiones de prueba gratuita con características limitadas, para aprovechar al máximo todas las herramientas y beneficios, los usuarios deben suscribirse a un plan de pago. Esto puede representar un obstáculo para aquellos que buscan soluciones económicas o están limitados por restricciones presupuestarias.

### **MyFxBook:**

#### *Ventajas:*

- Amplia gama de herramientas de análisis y estadísticas comerciales.
- Gran comunidad de usuarios y características sociales robustas.
- Integración con muchas plataformas de trading populares.

#### *Desventajas:*

- Interfaz puede resultar abrumadora para algunos usuarios, especialmente para principiantes.
- Algunas características avanzadas pueden requerir una suscripción premium.

### **TraderSync:**

#### *Ventajas:*

- Herramientas avanzadas de análisis y seguimiento de operaciones.
- Funcionalidades de gestión de riesgos y emociones.
- Interfaz intuitiva y fácil de usar.

#### *Desventajas:*

- Puede ser costoso para usuarios que buscan una solución más económica.
- Algunas características pueden ser excesivas para traders principiantes o aquellos que buscan una solución más básica.

### **Edgewonk:**

#### *Ventajas:*

- Enfoque en la psicología del trading y la gestión emocional.
- Herramientas de análisis detallado y seguimiento del rendimiento.
- Personalización y flexibilidad en la adaptación a las necesidades individuales de los traders.

#### *Desventajas:*

- Puede ser costosa en comparación con otras aplicaciones similares.
- Curva de aprendizaje puede ser empinada para algunos usuarios, especialmente para principiantes.
- No es tan popular como otras aplicaciones, lo que puede limitar la disponibilidad de recursos y comunidad de usuarios.

Nuestra aplicación TradingJournal tiene sus puntos clave en la organización y el análisis estratégico de operaciones financieras por categorías específicas, conscientes de la creciente importancia de la gestión financiera personal y el desarrollo profesional en este campo, nuestra plataforma innova al incorporar herramientas dedicadas al análisis emocional y la gestión del riesgo. Esto abre un nuevo horizonte para el crecimiento individual de los traders, permitiendo que la autoevaluación y el aprendizaje se realicen de una manera moderna, flexible y escalable. La posibilidad de analizar el comportamiento emocional durante las operaciones no solo optimiza la toma de decisiones, sino que también promueve una mentalidad más consciente y disciplinada en el trading.

Además, nuestra aplicación ofrece una Calculadora de Riesgo integrada que permite a los usuarios calcular fácilmente el tamaño del lote, el riesgo potencial de cada operación, ayudándoles a establecer límites claros y gestionar su capital de manera más efectiva. La interfaz intuitiva de nuestra aplicación también es un punto destacado, ya que proporciona una experiencia de usuario amigable y fácil de usar. Esto permite a los usuarios navegar sin esfuerzo por la aplicación, acceder a todas las funciones y herramientas de manera eficiente y personalizar su experiencia según sus necesidades individuales.

Para terminar y considerar el estado actual de las aplicaciones existentes y sus capacidades, nuestra aplicación se distingue por su propuesta de valor. No solo abraza las tendencias actuales en la tecnología financiera, sino que también las expande hacia nuevos horizontes de análisis y gestión de operaciones. Es una visión que reconoce y capitaliza la versatilidad y el poder del trading, asegurando que se mantenga a la vanguardia de la gestión financiera personal y la educación en trading.

### 3. Metodología usada (2-4 páginas)

La metodología en cascada es un enfoque secuencial y lineal para el desarrollo de software, es un enfoque secuencial que se estructura en fases bien definidas y ordenadas. Cada fase del ciclo de vida del proyecto debe completarse antes de pasar a la siguiente. La metodología en cascada es lineal y sistemática, y se caracteriza por su enfoque riguroso y su estructura fácil de entender.

Aquí hay un ejemplo de las fases de cómo podríamos aplicar la metodología en cascada para el desarrollo de nuestra aplicación:

1. Definición de requisitos o Análisis: En la fase inicial, trabajaremos en estrecha colaboración con los stakeholders para definir claramente los requisitos de la aplicación. Esto incluirá identificar las características clave que se desean, como el registro de operaciones, el análisis emocional y la calculadora de riesgo.
2. Diseño: Una vez que los requisitos estén establecidos, pasaremos a la fase de diseño. Aquí, crearemos un diseño detallado de la aplicación, incluyendo la arquitectura del sistema, la interfaz de usuario y la estructura de la base de datos.
3. Implementación: Con el diseño en su lugar, procederemos a la fase de implementación, donde se escribirá y se codificará el código de la aplicación. Esta fase implica traducir el diseño en software funcional y asegurarse de que cumpla con los requisitos establecidos.
4. Pruebas: Una vez que se complete la implementación, pasaremos a la fase de pruebas. Aquí, probaremos exhaustivamente la aplicación para identificar y corregir cualquier error o defecto. Esto incluirá pruebas de unidad, pruebas de integración.
5. Despliegue: Una vez que la aplicación haya sido probada y aprobada, estará lista para el despliegue. En esta fase, implementaremos la aplicación en un entorno de producción y la pondremos a disposición de los usuarios finales.
6. Mantenimiento: Finalmente, una vez que la aplicación esté en funcionamiento, pasaremos a la fase de mantenimiento. Aquí, proporcionaremos soporte continuo, realizaremos actualizaciones y correcciones de errores según sea necesario para garantizar el rendimiento y la funcionalidad continuos de la aplicación.

Decidimos emplear la metodología en cascada en este proyecto por varias razones fundamentales. En primer lugar, siendo un desarrollo solitario, gestionar un proyecto con un enfoque más ágil, que requiere mayor colaboración y adaptación continua, podría resultar más complejo y menos eficiente. La metodología en cascada permite una planificación y una ejecución paso a paso, lo cual puede ser más manejable para una sola persona.

En segundo lugar, el proyecto parece tener requisitos claramente definidos y un alcance bien delimitado desde el principio. Esto se ajusta al modelo en cascada, que funciona mejor cuando los requisitos son estables y hay una comprensión clara del producto final desde el inicio.

Además, considerando que se anticipa la creación de un sistema con componentes y funciones específicas centradas en el alojamiento y la categorización de contenido, la capacidad de definir estos elementos de manera clara y detallada desde la fase de requisitos favorece la estructura secuencial de la cascada.

## 4. Tecnologías y herramientas utilizadas en el proyecto

### (2-3 páginas)

En este proyecto, hemos seleccionado un conjunto de tecnologías específicas que juntas ofrecen una solución robusta y eficiente para el desarrollo de la aplicación TradingJournal. A continuación, se detalla cada tecnología y la razón de su elección:

- **Lenguaje de Programación (Kotlin):** Kotlin es el lenguaje oficial de desarrollo de Android respaldado por Google. Ofrece una sintaxis concisa y segura que mejora la productividad del desarrollador y reduce la probabilidad de errores.
- **Android Studio:** Es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones Android. Proporciona herramientas para diseñar interfaces de usuario, depurar aplicaciones, y gestionar proyectos completos específicos para aplicaciones Android.
- **Framework de Desarrollo de Aplicaciones Móviles (Android SDK):** Utilizar el framework nativo de Android permite aprovechar al máximo las características y funcionalidades específicas de la plataforma Android. Esto garantiza un rendimiento óptimo y una integración perfecta con el ecosistema Android.
- **Base De Datos SQLite:** Hemos seleccionado SQLite como nuestra base de datos por su integración nativa con Android, proporcionando una gestión de datos local eficiente y segura sin necesidad de un servidor externo. SQLite destaca por su bajo consumo de recursos, soporte robusto para SQL y facilidad de mantenimiento, lo que garantiza respuestas rápidas y reduce la latencia, aspectos cruciales para una aplicación de trading que demanda un procesamiento rápido de información. Además, su capacidad para operar en dispositivos con capacidades limitadas y su potencial de escalabilidad hacen de SQLite una opción ideal para manejar los datos complejos de nuestra aplicación móvil, manteniendo una experiencia de usuario óptima y eficiente.
- **Biblioteca de Gráficos (MPAndroidChart):** MPAndroidChart es una biblioteca ampliamente utilizada para crear gráficos interactivos en aplicaciones Android. Ofrece muchos tipos de gráficos y opciones de personalización, lo que permite visualizar tendencias y estadísticas de operaciones efectivas.

- **Material Design Components:** Componentes de interfaz de usuario que siguen las pautas de Material Design de Google para Android. Estos componentes ayudan a crear una experiencia de usuario coherente y moderna en la aplicación.
- **RecyclerView:** Un componente de Android utilizado para mostrar listas o cuadrículas de elementos de manera eficiente. Es utilizado en este proyecto para listar transacciones o datos relacionados, proporcionando manejo eficiente de vistas recicladas al desplazarse.
- **Intents y Activity Navigations:** Mecanismos para manejar la navegación entre diferentes pantallas (Activities) y para pasar datos entre ellas.
- **Coroutines de Kotlin:** Utilizadas para manejar operaciones asíncronas como accesos a la base de datos y otras operaciones de I/O sin bloquear el hilo principal de la interfaz de usuario.
- **Data Binding y View Binding:** Técnicas para conectar los componentes de la interfaz de usuario directamente a las fuentes de datos en el código, reduciendo el boilerplate y posibles errores en la manipulación de la UI.
- **LiveData:** Utilizado junto con el patrón de diseño de ViewModel para observar cambios en los datos y actualizar la UI automáticamente de manera eficiente y segura respecto al ciclo de vida de las actividades y fragmentos.
- **JUnit y Espresso:** JUnit es un marco de pruebas unitarias ampliamente utilizado para Android, mientras que Espresso es un marco de pruebas de interfaz de usuario que nos permite escribir pruebas de integración para asegurar la calidad y fiabilidad de nuestra aplicación.
- **Git y GitHub:** Utilizar Git como sistema de control de versiones nos permite realizar un seguimiento de los cambios en el código, colaborar con otros desarrolladores y mantener un historial de versiones completo. GitHub proporciona una plataforma de alojamiento de repositorios que facilitan la colaboración y la revisión de código entre el equipo de desarrollo.
- **Generación de recursos gráficos (Canva):** Hemos seleccionado Canva como nuestra herramienta principal. Canva ofrece una solución intuitiva y accesible para crear gráficos visualmente atractivos en línea. Su amplia variedad de plantillas y elementos de diseño nos permite desarrollar los elementos visuales de nuestra aplicación de manera eficiente, incluso sin tener habilidades avanzadas en diseño gráfico.

La integración de estas tecnologías garantiza la solidez, capacidad de escalabilidad y accesibilidad de la plataforma, al mismo tiempo que ofrece una experiencia de usuario placentera y de alta calidad.

## 5. Planificación, Diagnóstico y Contexto Laboral (3-4 páginas)

Para la planificación detallada de nuestro proyecto de la aplicación TradingJournal, hemos adoptado un enfoque meticuloso que organiza rigurosamente el proceso en etapas secuenciales. Comenzaremos el proyecto el 07 de marzo y nos hemos comprometido con una metodología en cascada para guiar nuestro desarrollo.

### 1.1 Planificación

**Objetivo General:** Desarrollar una aplicación móvil llamada TradingJournal que permita a los traders registrar y analizar sus operaciones financieras para mejorar su rendimiento en el mercado.

#### Etapas del Proyecto:

##### 1. Análisis de Requisitos (07 de marzo - 20 de marzo):

- Realizar sesiones de brainstorming con stakeholders para definir las necesidades y expectativas.
- Documentar los requisitos funcionales y no funcionales de la aplicación.
- Validar y priorizar los requisitos con los stakeholders.

##### 2. Diseño del Sistema (21 de marzo - 03 de abril):

- Definir la arquitectura de la aplicación y las tecnologías a utilizar.
- Crear wireframes y prototipos de las interfaces de usuario para visualizar el flujo de la aplicación.
- Elaborar el modelo de datos y planificar la integración con bases de datos y servicios externos.

##### 3. Implementación (04 de abril - 18 de abril):

- Desarrollar los componentes de back-end y front-end de la aplicación según el diseño.
- Implementar funciones para el registro de operaciones, análisis de rendimiento, y visualización de datos.
- Integrar sistemas de gestión de riesgos y calculadoras de rendimiento.

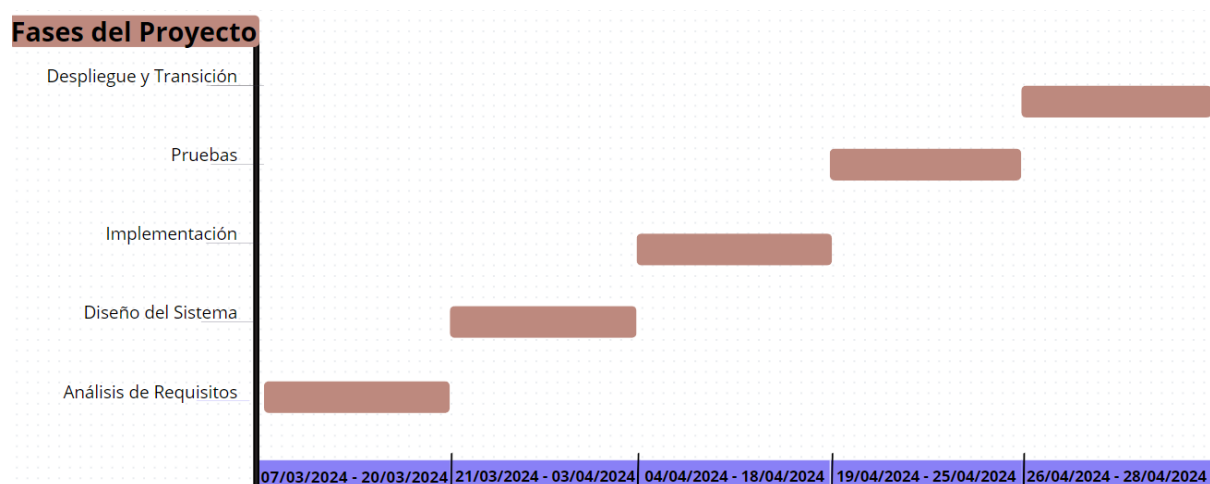
#### 4. Pruebas (19 de abril - 25 de abril):

- Realizar pruebas unitarias y de integración para asegurar la funcionalidad y estabilidad de la aplicación.
- Llevar a cabo pruebas de usabilidad con usuarios potenciales para recoger feedback y realizar ajustes.
- Preparar la aplicación para despliegue asegurando que cumpla con todos los requisitos de seguridad y privacidad.

#### 5. Despliegue y Transición (26 de abril - 28 de abril):

- Lanzar la aplicación en plataformas seleccionadas (Android).
- Monitorizar el rendimiento de la aplicación en el entorno de producción y resolver cualquier incidencia técnica.
- Proporcionar formación y soporte técnico inicial a los usuarios.

En nuestro proyecto, cada fase se inicia inmediatamente después de concluir la anterior, lo que asegura un flujo de trabajo constante desde la definición de requisitos hasta el establecimiento del plan de mantenimiento. La fase de implementación, siendo la más compleja, dispone del mayor tiempo asignado. Utilizamos un diagrama de Gantt para visualizar y monitorizar el progreso de cada etapa, lo que facilita mantener un seguimiento claro de nuestro itinerario y asegura que cada componente recibe la atención necesaria. Este diagrama es crucial para garantizar que el proyecto se desarrolle conforme a lo planificado y para construir una base sólida para nuestra solución innovadora en el ámbito del podcasting.



Para hacer el diagnóstico de nuestro proyecto vamos a utilizar un análisis DAFO es una herramienta estratégica que permite identificar las Debilidades, Amenazas, Fortalezas y



Oportunidades de un proyecto o negocio. Aquí tienes un esquema básico de cómo podría estructurarse un análisis DAFO para nuestra plataforma:

#### **Fortalezas:**

**Interfaz Intuitiva y Amigable:** Facilita la adopción por parte de usuarios nuevos y experimentados, reduciendo la curva de aprendizaje.

**Integración de SQLite para Almacenamiento Local:** Garantiza un rápido acceso y manipulación de datos sin depender de conexiones a internet.

**Funcionalidad Completa en Dispositivo Móvil:** Permite a los usuarios gestionar sus operaciones en cualquier momento y lugar, ofreciendo una mayor flexibilidad.

**Análisis y Visualización de Datos Avanzados:** Proporciona herramientas potentes para el análisis de tendencias y rendimiento de operaciones, lo que ayuda a mejorar la toma de decisiones.

#### **Debilidades:**

**Limitación a Plataformas Android:** Restringe el acceso a usuarios de otros sistemas operativos móviles como iOS.

**Ausencia de Sincronización en la Nube:** Limita la capacidad del usuario para acceder a sus datos desde diferentes dispositivos.

**Falta de Integración con Plataformas de Trading Externas:** Puede limitar la utilidad para usuarios que operan a través de múltiples plataformas.

**Recursos Limitados para Marketing y Promoción:** Puede restringir la capacidad de alcanzar una base de usuarios más amplia rápidamente.

#### **Oportunidades:**

**Expansión a Otras Plataformas Móviles:** Desarrollar versiones para iOS y otros sistemas operativos para capturar una mayor cuota de mercado.

**Integración con Servicios en la Nube:** Ofrecer opciones de sincronización y backup en la nube para mejorar la accesibilidad y seguridad de los datos.

**Alianzas Estratégicas con Plataformas de Trading:** Establecer colaboraciones para integrar directamente con plataformas de trading existentes, mejorando la funcionalidad.

**Aumento en la Demanda de Herramientas de Trading Móviles:** Tendencia creciente hacia el trading móvil que puede ser aprovechada para impulsar el crecimiento y la adopción.

#### **Amenazas:**

**Competencia Intensa:** Numerosas aplicaciones de trading ya establecidas podrían eclipsar a TradingJournal.

**Cambios Tecnológicos Rápidos:** La necesidad constante de actualizaciones y mejoras para mantenerse al día con las nuevas tecnologías y expectativas de los usuarios.

**Regulaciones del Mercado Financiero:** Cambios en las regulaciones que podrían afectar las operaciones de los usuarios de la aplicación.

**Vulnerabilidades de Seguridad:** Riesgos asociados con la protección de datos financieros sensibles almacenados localmente.

## 2.2 Diagnóstico

El análisis de mercado realizado indica una creciente demanda de herramientas digitales de trading que ofrecen seguimiento detallado y personalizado. La aplicación se diferenciará por su enfoque en el análisis emocional y gestión de riesgos personalizada, aspectos menos desarrollados en aplicaciones existentes.

### Contexto Laboral

#### Equipo de Desarrollo:

- **Gerente de Proyecto:** Coordina el proyecto, asegura el cumplimiento de los plazos y la calidad del producto.
- **Desarrolladores de Software:** Implementan la aplicación según los diseños y especificaciones.
- **Diseñadores UX/UI:** Refinan la interfaz para una experiencia de usuario óptima.
- **Especialistas QA:** Realizan pruebas para garantizar la funcionalidad y seguridad.
- **Analistas de Datos:** Integran análisis avanzados dentro de la aplicación.

**Ambiente de Trabajo:** El equipo trabajará en un entorno ágil, con énfasis en la adaptabilidad y comunicación efectiva. Las reuniones diarias y revisiones semanales permitirán una rápida respuesta a desafíos y cambios en los requisitos del proyecto.

## 6. Análisis del proyecto (8-16 páginas)

### Requisitos Funcionales

- 1. Registro de Operaciones de Trading:** Los usuarios deben poder ingresar detalles de cada operación, incluyendo el instrumento financiero, cantidad, precio de entrada, precio de salida, fecha, etc...
- 2. Análisis de Tendencias y Rendimiento:** La aplicación debe proporcionar herramientas analíticas que muestren el rendimiento histórico de las operaciones a través de gráficos interactivos.
- 3. Registro del Comportamiento Emocional:** Permitir a los usuarios registrar sus emociones y pensamientos para cada operación y visualizar la correlación entre las emociones y el rendimiento de trading.
- 4. Calculadora de Riesgo:** Implementar una calculadora que permita a los usuarios calcular el tamaño adecuado del lote basado en el capital de riesgo por operación deseado.
- 5. Búsqueda y Filtro Avanzado:** Los usuarios deben poder buscar y filtrar operaciones por varios criterios como fecha, instrumento financiero, emoción, sesión, estrategia, etc...
- 6. Journal:** Un panel de control que muestre resúmenes y estadísticas claves como total de ganancias/pérdidas por porcentaje, y distribución de instrumentos.

#### **Requisitos No Funcionales:**

##### **1. Disponibilidad:**

Asegurar que la aplicación sea completamente funcional mientras el dispositivo esté operativo, independientemente de la conectividad a internet.

##### **2. Rendimiento:**

Garantizar tiempos de respuesta rápidos y tiempos de carga cortos para todas las operaciones, especialmente durante el acceso y la manipulación de la base de datos local.

##### **3. Escalabilidad:**

La aplicación debe manejar un incremento en la cantidad de datos sin degradar el rendimiento, permitiendo a los usuarios almacenar años de datos de trading.

##### **4. Seguridad:**

Implementar medidas robustas para la protección de datos almacenados localmente, asegurando la confidencialidad y la integridad de la información del usuario.

##### **5. Compatibilidad:**

Soportar las versiones principales de Android e iOS y ser compatible con diferentes tamaños de pantalla y especificaciones de hardware.

##### **6. Usabilidad:**

Ofrecer una interfaz intuitiva y fácil de navegar que minimice la curva de aprendizaje y mejore la experiencia del usuario.

## 7. Accesibilidad:

Asegurar que la aplicación cumpla con las normativas de accesibilidad para ser utilizable por el mayor rango posible de usuarios, incluyendo aquellos con discapacidades.

## 8. Mantenibilidad:

Mantener el código bien organizado y documentado para facilitar futuras actualizaciones y mantenimientos.

## 9. Backup y Recuperación:

Proporcionar opciones para que los usuarios realicen copias de seguridad de sus datos y los recuperen fácilmente para evitar pérdidas de información en caso de fallos del dispositivo.

## Modelo de Datos Detallado para TradingJournal

### Entidad: allTransactions

Esta entidad almacenará los detalles de cada operación de trading realizada por el usuario. La estructura de la tabla es la siguiente:

- **id (PK):** Un identificador único para cada operación.
- **instrument:** El instrumento financiero utilizado en la operación (por ejemplo, acciones, forex, criptomonedas).
- **day:** El día en que se realizó la operación.
- **month:** El mes en que se realizó la operación.
- **year:** El año en que se realizó la operación.
- **strategy:** La estrategia de trading aplicada.
- **entryPrice:** El precio de entrada de la operación.
- **stopLoss:** El precio de stop loss establecido para limitar las pérdidas.
- **takeProfit:** El precio de toma de beneficios establecido.
- **risk:** El riesgo asumido en la operación, generalmente expresado como un porcentaje del capital total.
- **win\_Loss:** Indica si la operación resultó en ganancia o pérdida.
- **percentage:** El porcentaje de ganancia o pérdida resultante de la operación.
- **emotion:** Las emociones registradas por el usuario relacionadas con la operación.
- **session:** La sesión de trading durante la cual se realizó la operación (por ejemplo, sesión asiática, europea, americana).

### Relaciones entre Datos

- **allTransactions y Instrument:**

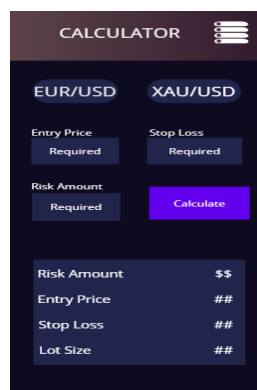
- Cada transacción está vinculada a un instrumento financiero específico que fue negociado. Permite análisis detallados del rendimiento por tipo de instrumento, lo que es crucial para los traders que operan en múltiples mercados y necesitan optimizar estrategias en diferentes entornos de trading.
- **allTransactions y Year, Month, Strategy:**
  - Las transacciones están fechadas por día, mes y año, y cada una está asociada con una estrategia de trading específica. Facilita análisis temporales y estratégicos.
- **allTransactions y Year, Session, Strategy, Instrument, Emotion:**
  - Esta es una relación compleja que vincula múltiples dimensiones para un análisis más profundo. Una transacción puede ser examinada en el contexto del año, la sesión de trading en la que se ejecutó, la estrategia utilizada, el instrumento financiero involucrado, y las emociones registradas. Proporciona una vista holística del trading.

## 7. Diseño del proyecto (6-14 páginas)

- 1. La clase **HomeActivity** actúa como la pantalla principal de la aplicación de diario de trading, donde configuramos la interfaz de usuario y manejamos la navegación hacia diferentes actividades especializadas como **GraphicsActivity**, **JournalActivity**, **AnalysisActivity**, y **CalculatorActivity**. Al iniciar, inflamos el layout utilizando **ActivityHomeBinding** y establecemos **db** para gestionar la base de datos con **MyDatabaseHelper**. Cada botón en la interfaz activa un método **navigateTo** correspondiente que inicia la actividad deseada mediante un **Intent**. Este patrón de diseño mantiene el código organizado y facilita la adición de nuevas funciones o actividades en el futuro.

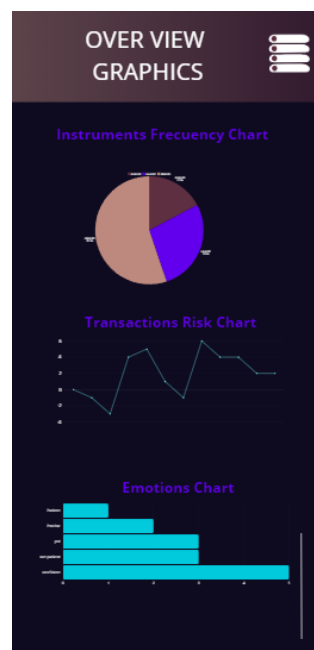


- 2. La clase **CalculatorActivity** en nuestra aplicación Android sirve como una actividad donde los usuarios pueden calcular el tamaño de lote adecuado para operaciones de trading con EUR/USD y GOLD (XAU). La actividad permite a los usuarios seleccionar el instrumento de trading deseado mediante la interacción con **CardView** para cada instrumento, y proporciona campos de texto para introducir el precio de entrada, el stop loss y el riesgo por operación. Dependiendo del instrumento seleccionado, se realiza un cálculo específico cuando el usuario pulsa el botón calcular, y el tamaño de lote calculado junto con los valores introducidos se muestran en la pantalla. La actividad también maneja la navegación de regreso a **HomeActivity** y ajusta visualmente los componentes seleccionados cambiando su color de fondo para reflejar la selección actual del usuario. El manejo de errores está incluido para evitar problemas como la división por cero durante los cálculos.



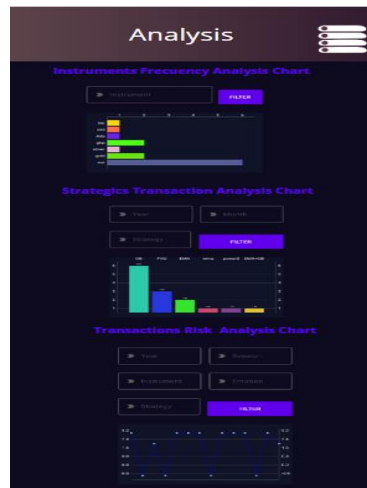
- 3. La clase **GraphicsActivity** en nuestra aplicación Android es responsable de mostrar gráficos visuales para análisis de datos, utilizando la biblioteca

MPAndroidChart para generar gráficos de pastel, línea y barra horizontal. Estos gráficos presentan diferentes aspectos del trading, como la distribución de tipos de instrumentos, análisis de riesgo y frecuencia de emociones asociadas a las transacciones, con datos obtenidos de una base de datos mediante **MyDatabaseHelper**. La actividad configura cada gráfico con colores personalizados, animaciones y configuraciones específicas como deshabilitar descripciones y ajustar ejes. Además, se ofrece un botón para regresar al **HomeActivity**, mejorando la navegabilidad en la aplicación. La interfaz de usuario es dinámica y se adapta a los datos en tiempo real, ofreciendo también interactividad mediante marcadores personalizados que muestran detalles adicionales al seleccionar puntos específicos en el gráfico de línea.



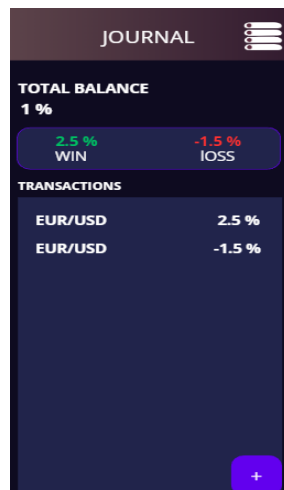
- 4. La clase **AnalysisActivity** en nuestra aplicación Android se centra en proporcionar un análisis visual detallado del rendimiento de trading mediante gráficos de barras y líneas, utilizando la biblioteca MPAndroidChart. Esta actividad es capaz de filtrar y visualizar datos en base a múltiples parámetros como instrumento, estrategia, año, mes, sesión y emoción, ofreciendo así una herramienta flexible y poderosa para el análisis de datos de trading. Cada función en la actividad extrae datos específicos de la base de datos a través de **MyDatabaseHelper** y representa visualmente esta información en diversos gráficos que se pueden personalizar con colores dinámicos y marcadores interactivos para mejorar la interpretación de los datos. Además, ofrece

botones para filtrar los datos según criterios específicos y para regresar a **HomeActivity**, mejorando la navegación y la usabilidad de la aplicación.

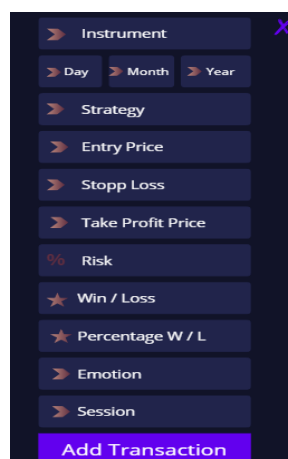


- 5. La clase **JournalActivity** de nuestra aplicación Android es una actividad crucial diseñada para gestionar y visualizar transacciones de trading almacenadas en una base de datos local. Utiliza un **RecyclerView** para listar las transacciones, permitiendo a los usuarios ver, agregar y eliminar registros. La interacción para añadir nuevas transacciones se facilita mediante un botón flotante que lanza **AddTransactionActivity**, mientras que las transacciones existentes se pueden eliminar deslizando hacia la derecha en el ítem correspondiente en el **RecyclerView**. Además, incluye una funcionalidad de deshacer mediante un **Snackbar** que permite a los usuarios restaurar transacciones recientemente eliminadas.
- Para proporcionar más detalles a los usuarios, **JournalActivity** incluye una función llamada **navigateToDetailedActivity**, que se encarga de abrir **DetailedActivity** con detalles específicos de la transacción seleccionada, facilitando la navegación entre vistas a través de **Intents** que pasan el ID de la transacción seleccionada.
- Además, la actividad calcula y muestra un resumen de las transacciones como ganancias y pérdidas expresadas en porcentajes. Esto se actualiza dinámicamente reflejando cualquier cambio en la base de datos, asegurando que los usuarios tengan acceso a información actualizada en todo momento. Con todas estas características, **JournalActivity** no solo mejora la gestión de datos de trading sino que también enriquece la experiencia del usuario al proporcionar un sistema interactivo y actualizado para el seguimiento de su rendimiento en trading.

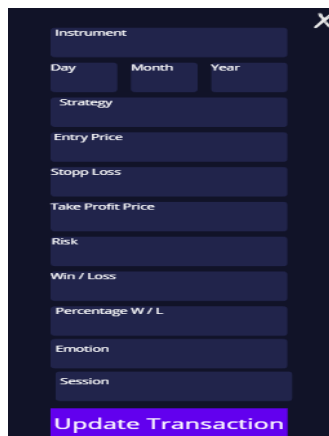




- 6. La clase **AddTransactionActivity** en nuestra aplicación Android sirve como una actividad dedicada para añadir nuevas transacciones de trading a la base de datos. Utiliza **ActivityAddTransactionBinding** para enlazar elementos de la interfaz de usuario y manejar entradas del usuario con validaciones extensivas. Los usuarios pueden ingresar detalles como instrumento, fecha, estrategia, precios de entrada y de salida, riesgo, porcentaje de ganancia, emoción y sesión de trading. La actividad verifica la validez de cada entrada, mostrando errores específicos si los campos están vacíos o los datos no son válidos (por ejemplo, win\_loss y sesiones correctas). Una vez que todas las entradas son validadas y aceptadas, la transacción se guarda en la base de datos mediante **MyDatabaseHelper**. Además, la actividad proporciona retroalimentación visual mediante **Toast** sobre el éxito o fallo al guardar los datos y permite cerrar la actividad y volver a la vista anterior con un botón de cierre. También cuenta con listeners de cambio de texto que resetean los mensajes de error a medida que el usuario corrige sus entradas, mejorando la experiencia de interacción.



- 7. La clase **DetailedActivity** en nuestra aplicación Android sirve para visualizar y editar los detalles de una transacción específica de trading. Esta actividad recupera la información de una transacción usando su ID, proporcionado a través de un **Intent**, y muestra los datos en campos editables gestionados con **ActivityDetailedBinding**. Los usuarios pueden actualizar cualquier parte de la transacción, incluyendo instrumento, fecha, estrategia, precios de entrada y salida, riesgo, resultado, porcentaje de ganancia o pérdida, emoción y sesión de trading. Cada campo tiene validaciones para asegurar la integridad de los datos ingresados. Si el usuario modifica algún campo, el botón de actualización se muestra, permitiendo guardar los cambios que luego se verifican y actualizan en la base de datos mediante **MyDatabaseHelper**. También incluye una funcionalidad para cerrar la actividad y esconder el teclado virtual al tocar fuera de los campos de entrada, mejorando la experiencia de usuario.



## 8. Despliegue y pruebas (4-10 páginas)

Para desarrollar un plan de despliegue y diseñar un plan de pruebas efectivos, es crucial organizar las actividades y tener en cuenta los entornos, herramientas y estrategias requeridas para ejecutar estas tareas con eficacia. A continuación, describo la aproximación que propongo para nuestro proyecto.

### Plan de Despliegue

### **Preparación del Entorno de Desarrollo y Producción:**

- **Configuración del Entorno de Desarrollo:** Establecer y configurar un entorno de desarrollo con todas las herramientas necesarias como Android Studio, el SDK de Android, y emuladores o dispositivos físicos para pruebas.
- **Integración de SQLite:** Integrar y configurar la base de datos SQLite dentro de la aplicación para manejar operaciones de datos localmente sin depender de conexiones a internet o servidores externos.

### **Pruebas en el Entorno de Staging:**

- **Simulación de Entorno de Producción:** Antes del lanzamiento, realizar pruebas en un entorno que simule el de producción lo más cercanamente posible, incluyendo pruebas en diferentes versiones de Android y dispositivos para asegurar la compatibilidad y rendimiento.

### **Despliegue de la Aplicación:**

- **Compilación y Preparación:** Compilar la aplicación en modo de producción, asegurando que todas las configuraciones de desarrollo estén deshabilitadas o ajustadas para el lanzamiento.
- **Distribución:** Utilizar Google Play Store o distribuciones directas APK para desplegar la aplicación a los usuarios finales.

### **Verificación Post-Despliegue:**

- **Monitorización Inicial:** Monitorear la aplicación justo después del despliegue para capturar cualquier error o comportamiento inesperado en los dispositivos de los usuarios.

### **Respuesta a Incidentes:**

- **Soporte Técnico:** Establecer un sistema eficiente para gestionar las respuestas a los problemas técnicos que los usuarios puedan informar después de la instalación.

### **Plan de Pruebas**

#### **Pruebas de Interfaz de Usuario (UI):**

- **Verificación Visual:** Asegurar que todas las interfaces se renderizan correctamente en diferentes tamaños de pantalla y orientaciones.
- **Interacción con el Usuario:** Verificar que las interacciones como toques, deslizamientos, y otras gestiones táctiles funcionan como se espera.

#### **Pruebas Funcionales:**

- **Operaciones CRUD:** Probar todas las operaciones de creación, lectura, actualización y eliminación sobre la base de datos SQLite para asegurar que los datos se manejan correctamente.
- **Persistencia de Datos:** Verificar que los datos se guardan y recuperan de manera correcta entre sesiones de la aplicación.

#### **Pruebas de Seguridad:**

- **Seguridad de la Base de Datos:** Asegurar que la base de datos SQLite está adecuadamente asegurada contra accesos no autorizados, especialmente si la aplicación maneja información sensible.
- **Pruebas de Inyección de SQL:** Aunque SQLite es menos vulnerable en un entorno controlado como una aplicación local en Android, es prudente verificar la manipulación de entradas para prevenir inyecciones SQL.

#### **Pruebas de Rendimiento:**

- **Tiempo de Respuesta y Carga:** Evaluar cómo la aplicación maneja grandes volúmenes de datos y transacciones para asegurar que no hay retrasos ni cuelgues.
- **Uso de Recursos:** Monitorizar el uso de CPU, memoria, y almacenamiento para asegurar que la aplicación no consume recursos excesivamente.

Este plan ajustado garantiza que la aplicación no solo sea funcional y segura, sino que también ofrezca una experiencia de usuario fluida y agradable en diferentes dispositivos y versiones de Android.

## **9. Conclusiones (1-2 páginas)**

El desarrollo y despliegue de nuestra aplicación de trading journal ha sido un proceso exhaustivo que ha implicado múltiples etapas, desde la planificación inicial hasta la implementación y las pruebas finales. A lo largo de este proceso, hemos logrado alcanzar varios hitos importantes y hemos identificado áreas de mejora para futuras iteraciones. A continuación, se presentan algunas conclusiones clave derivadas de nuestro trabajo en la aplicación:

### **1. Experiencia del Usuario y Funcionalidad:**

- Hemos priorizado la experiencia del usuario al diseñar una interfaz intuitiva y fácil de usar que permita a los traders registrar y analizar sus operaciones de manera eficiente.

- La funcionalidad de la aplicación ha sido robusta, permitiendo a los usuarios registrar detalles específicos de cada operación, como instrumento, estrategia, precios de entrada y salida, así como también emociones asociadas y sesiones de trading.

## **2. Integración de la Base de Datos SQLite:**

- La elección de SQLite como base de datos local ha demostrado ser acertada, ya que ofrece un rendimiento eficiente y una fácil integración con la aplicación Android.
- La implementación de la base de datos SQLite ha permitido un almacenamiento seguro y estructurado de la información de las transacciones, facilitando su acceso y consulta en la aplicación.

## **3. Planificación del Despliegue y Pruebas:**

- Se ha elaborado un plan detallado para el despliegue de la aplicación, incluyendo la configuración del entorno de producción, la migración de datos y la verificación post-despliegue.
- Las pruebas se han llevado a cabo de manera exhaustiva, cubriendo aspectos como la interfaz de usuario, funcionalidad, seguridad y rendimiento, garantizando así la calidad y estabilidad de la aplicación.

## **4. Áreas de Mejora Futura:**

- A pesar de los logros alcanzados, identificamos áreas de mejora para futuras iteraciones, como la implementación de funciones avanzadas de análisis de datos y la expansión de la aplicación para soportar múltiples dispositivos y resoluciones de pantalla.
- También reconocemos la importancia de continuar mejorando la seguridad de la aplicación, implementando prácticas de almacenamiento seguro de datos y protección contra posibles vulnerabilidades.

En resumen, el desarrollo y despliegue de nuestra aplicación de trading journal ha sido un proceso gratificante que ha implicado un trabajo colaborativo y una atención meticulosa a los detalles. Con un enfoque continuo en la mejora y la innovación, estamos seguros de que nuestra aplicación seguirá evolucionando para satisfacer las necesidades cambiantes de nuestros usuarios y mantenerse como una herramienta invaluable en el arsenal de todo trader.

## 10. Vías futuras (1-2 páginas)

### 1. Gestión de Usuarios y Seguridad

- **Registro y Autenticación por Correo Electrónico:** Permitir a los usuarios registrarse utilizando su dirección de correo electrónico y establecer una contraseña. Esto proporcionaría una forma segura y conveniente de crear cuentas de usuario en la aplicación.
- **Recuperación de Contraseña:** Implementar un proceso de recuperación de contraseña mediante correo electrónico, permitiendo a los usuarios restablecer su contraseña en caso de olvido.
- **Verificación por Correo Electrónico:** Enviar correos electrónicos de verificación para garantizar la autenticidad de las cuentas de usuario y proteger contra el fraude.

### 2. Personalización y Comunicación

- **Notificaciones por Correo Electrónico:** Enviar notificaciones y actualizaciones importantes por correo electrónico, como confirmaciones de transacciones, alertas de seguridad o resúmenes periódicos de actividad de trading.
- **Comunicación Directa:** Facilitar la comunicación directa con los usuarios a través del correo electrónico para proporcionar soporte, compartir noticias y actualizaciones sobre la aplicación, y recopilar comentarios y sugerencias de mejora.

### 3. Seguridad y Privacidad de los Datos

- **Protección de la Privacidad:** Garantizar que la información del usuario enviada por correo electrónico esté protegida y encriptada para preservar la privacidad y la seguridad de los datos personales.
- **Cumplimiento de Normativas de Privacidad:** Asegurarse de que el manejo de los datos de los usuarios a través del correo electrónico cumpla con las regulaciones de privacidad, como el Reglamento General de Protección de Datos (GDPR).

### 4. Recordatorios y Alertas

- **Recordatorios Personalizados:** Enviar recordatorios y alertas por correo electrónico sobre eventos importantes relacionados con el trading, como fechas de vencimiento

de contratos, próximos informes económicos relevantes o cambios significativos en el mercado.

- **Seguimiento de Actividad:** Proporcionar resúmenes periódicos de la actividad de trading del usuario a través del correo electrónico, incluyendo estadísticas, métricas clave y recomendaciones basadas en su historial de operaciones.

## 5. Mejoras en la Experiencia del Usuario

- **Acceso Universal:** Permitir a los usuarios acceder a ciertas funciones de la aplicación, como la gestión de la cuenta y la configuración de preferencias, a través de correos electrónicos en lugar de depender exclusivamente de la aplicación móvil.
- **Flexibilidad y Comodidad:** Brindar a los usuarios la flexibilidad de interactuar con la aplicación y recibir actualizaciones importantes desde cualquier dispositivo con acceso al correo electrónico.

## 6. Integración de Análisis de Datos Avanzados

- **Machine Learning e Inteligencia Artificial:** Incorporar algoritmos de machine learning para analizar tendencias de trading, predecir resultados de transacciones basados en datos históricos y proporcionar recomendaciones personalizadas a los usuarios.
- **Dashboard Analíticos Mejorados:** Desarrollar dashboards más interactivos y completos que permitan a los usuarios visualizar sus datos de trading de manera más dinámica, con gráficos, heatmap de operaciones exitosas/fallidas, y análisis de correlación.

## 7. Ampliación de la Compatibilidad y la Accesibilidad

- **Soporte Multiplataforma:** Expandir la aplicación para que sea accesible no solo en dispositivos Android sino también en iOS y plataformas web, permitiendo a los usuarios acceder a su diario de trading desde cualquier dispositivo.
- **Mejoras en la Interfaz de Usuario:** Optimizar la interfaz de usuario para adaptarla a tablets y otros dispositivos con pantallas grandes, mejorando la usabilidad y la experiencia del usuario.

## 8. Funcionalidades de Networking y Comunidad

- **Integración Social:** Permitir a los usuarios compartir sus análisis y resultados de trading dentro de una comunidad segura, fomentando la interacción y el aprendizaje entre pares.
- **Funcionalidad de Coaching o Tutoría:** Integrar una plataforma donde los usuarios puedan recibir tutorías de traders más experimentados o inscribirse en cursos y webinars directamente desde la aplicación.

## 9. Expansión del Soporte de Datos

- **Integración de Diversas Fuentes de Datos:** Incorporar datos de múltiples fuentes y mercados, incluyendo criptomonedas, opciones, futuros y forex, para proporcionar una visión más amplia y detallada del entorno de trading.

## 10. Feedback y Soporte al Usuario Mejorado

- **Sistema de Feedback en la Aplicación:** Implementar un sistema donde los usuarios puedan fácilmente proporcionar feedback y reportar problemas directamente a través de la aplicación, facilitando mejoras continuas basadas en las necesidades del usuario.

Agregar la gestión de usuarios mediante correo electrónico no solo fortalecería la seguridad y la privacidad de la aplicación, sino que también mejoraría la comunicación y la personalización de la experiencia del usuario, lo que en última instancia conduciría a una mayor satisfacción y compromiso por parte de los usuarios.

## 11. Bibliografía/Webgrafía (1-2 páginas)

### Libros y Publicaciones

1. Doe, J. (2020). Effective Trading Strategies. New York, NY: Financial Press.

Este libro proporciona una visión exhaustiva de las estrategias de trading contemporáneas que hemos adaptado para desarrollar módulos de análisis en nuestra aplicación.

2. Smith, A., & White, B. (2018). User Interface Design for Modern Applications. London, UK: Tech Publisher.



Ofrece fundamentos clave en el diseño de interfaces de usuario que fueron cruciales para el diseño de nuestra aplicación, asegurando que fuera tanto funcional como estéticamente agradable.

#### Artículos de Revistas

1. Johnson, K., & Lee, T. (2019). "Data Security in Mobile Applications," Journal of Cybersecurity, 15(3), 234-248.

Este artículo discute técnicas avanzadas de seguridad que fueron consideradas para proteger la información del usuario en nuestra aplicación.

#### Recursos Web

1. Android Developers Documentation. (n.d.). Retrieved March 5, 2024, from <https://developer.android.com/guide>

La documentación oficial de Android proporcionó guías esenciales para el desarrollo de la aplicación, especialmente para asegurar la compatibilidad y el rendimiento óptimo en todos los dispositivos Android.

2. SQLite Documentation. (n.d.). Retrieved March 10, 2024, from <https://www.sqlite.org/docs.html>

Proporciona información técnica sobre SQLite, la base de datos elegida para nuestro proyecto, facilitando la implementación de la base de datos local en la aplicación.

#### Documentos y Reportes Técnicos:

1. IEEE Software Requirements Specification (SRS)

Este documento de norma de IEEE explica cómo redactar especificaciones de requisitos de software detalladas que aseguren la claridad y la precisión requeridas para el desarrollo de software eficaz. IEEE SRS Document

#### Software y Herramientas:

## 1. Android Studio

El IDE oficial para el desarrollo de aplicaciones Android, proporciona todas las herramientas necesarias para diseñar, desarrollar, depurar y desplegar aplicaciones móviles.

<https://developer.android.com/studio>

## 2. SQLite

Un motor de base de datos relacional ligero que se integra directamente en la aplicación, ideal para dispositivos móviles y aplicaciones independientes. <https://www.sqlite.org/index.html>

## Normativas y Estándares:

### 1. General Data Protection Regulation (GDPR)

Regulación de la UE que establece directrices para la recolección y procesamiento de información personal de individuos dentro de la Unión Europea. <https://gdpr-info.eu/>

### 2. ISO/IEC 27001 - Information Security Management

Un estándar internacional que proporciona un marco para las mejores prácticas de gestión de seguridad de la información (SGSI). <https://www.iso.org/standard/54534.html>

## Foros y Comunidades:

### 1. Stack Overflow

Una comunidad en línea donde los desarrolladores pueden hacer preguntas y compartir conocimientos sobre programación y desarrollo de software. <https://stackoverflow.com/>

### 2. GitHub Community

Un foro para colaborar en proyectos, rastrear errores y solicitar nuevas características para el desarrollo colaborativo de software. <https://github.community/>

## Tutoriales y Guías:

### 1. Android Development Fundamentals by Google

Cursos y tutoriales ofrecidos por Google para enseñar los fundamentos del desarrollo de aplicaciones Android, incluyendo UI, funcionalidad y despliegue. Android Fundamentals

## 2. Codecademy: Learn SQL

Un curso interactivo que enseña SQL desde lo básico hasta temas avanzados, útil para manejar bases de datos en aplicaciones como SQLite.

<https://www.codecademy.com/learn/learn-sql>

## 12. Anexos

### 1.1. Manual de Usuario

El manual de usuario está diseñado para ayudar a los usuarios a navegar y utilizar eficazmente la aplicación TradingJournal. Proporciona instrucciones paso a paso sobre cómo realizar las funciones principales, consejos útiles y soluciones a problemas comunes.

#### Índice

1. Introducción
2. Instalación de la Aplicación
3. Iniciar Sesión
4. Interfaz Principal
5. Registro de Operaciones de Trading
6. Modificación de Operaciones de Trading
7. Over View Graphics de Tendencias y Rendimiento
8. Análisis de Tendencia y Rendimiento
9. Uso de la Calculadora de Riesgo y Beneficio
10. Visualización del Dashboard de Resumen
11. Preguntas Frecuentes
12. Solución de Problemas
13. Contacto de Soporte

#### 1. Introducción

Bienvenido al manual de usuario de TradingJournal, su herramienta confiable para registrar y analizar sus actividades de trading. Este manual le proporcionará todas las instrucciones necesarias para aprovechar al máximo todas las funcionalidades que nuestra aplicación tiene para ofrecer. Prepárese para transformar su enfoque de trading con análisis detallados y gestión eficiente de sus operaciones.

## 2. Instalación de la Aplicación

Para instalar TradingJournal en su dispositivo Android, siga estos pasos:

- Visite Google Play Store.
- Busque "TradingJournal" en la barra de búsqueda.
- Seleccione "Instalar" y permita que la aplicación se descargue e instale en su dispositivo.
- Una vez instalada, abra la aplicación tocando el ícono en su pantalla de inicio.

## 3. Iniciar Sesión

- Como nuestra aplicación no requiere inicio de sesión, simplemente toque "Comenzar" para acceder a las funcionalidades principales.

## 4. Interfaz Principal

La interfaz principal de la aplicación le permite acceder rápidamente a todas las características. Se compone de los siguientes elementos:

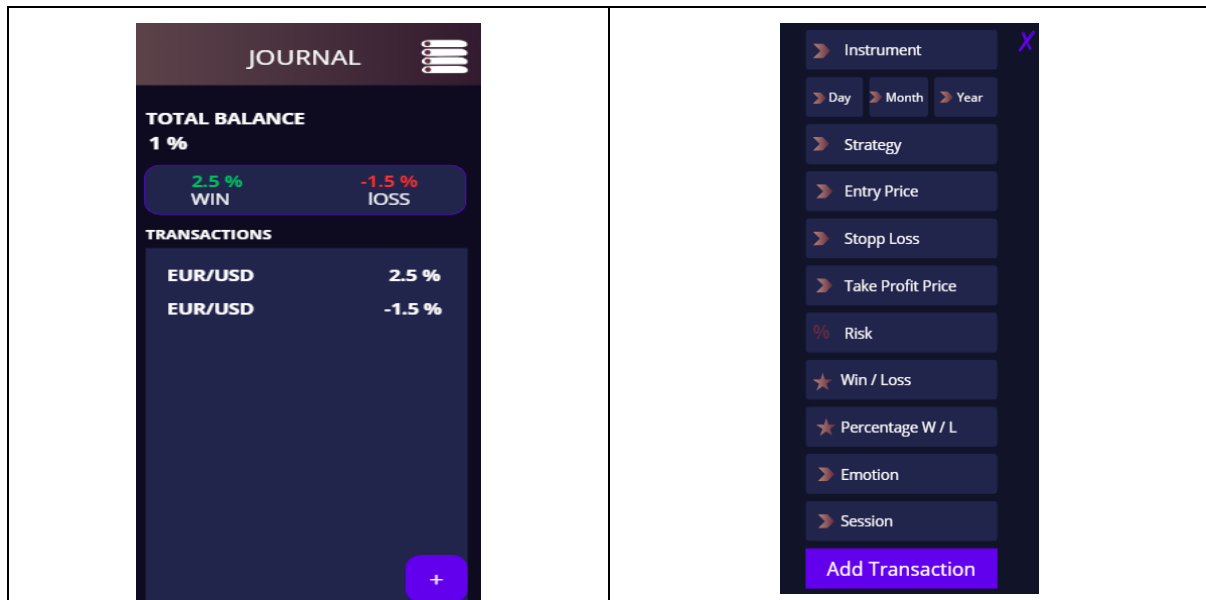
- **Overview Graphics:** Vista general de gráficos estadísticos.
- **Journal:** Registro y gestión de operaciones individuales.
- **Analysis:** Herramientas analíticas y gráficos de rendimiento.
- **Calculator:** Calculadora de riesgo y beneficio.



## 5. Registro de Operaciones de Trading

Para registrar una nueva operación:

- Navegue a la sección "Journal".
- Toque el botón "Add" situado en la esquina inferior derecha.
- Complete el formulario con detalles como el instrumento, tipo de operación, precios de entrada y salida, entre otros.
- Guarde la operación usando el botón "Add Transaction".



## 6. Modificación de Operaciones de Trading

Para Modificar una operación:

- Toque la operación que desea modificar
- Modifique el campo que desea actualizar
- Guarde la operación modificada usando el botón "Update Transaction".

X

Instrument

Day

Month

Year

Strategy

Entry Price

Stopp Loss

Take Profit Price

Risk

Win / Loss

Percentage W / L

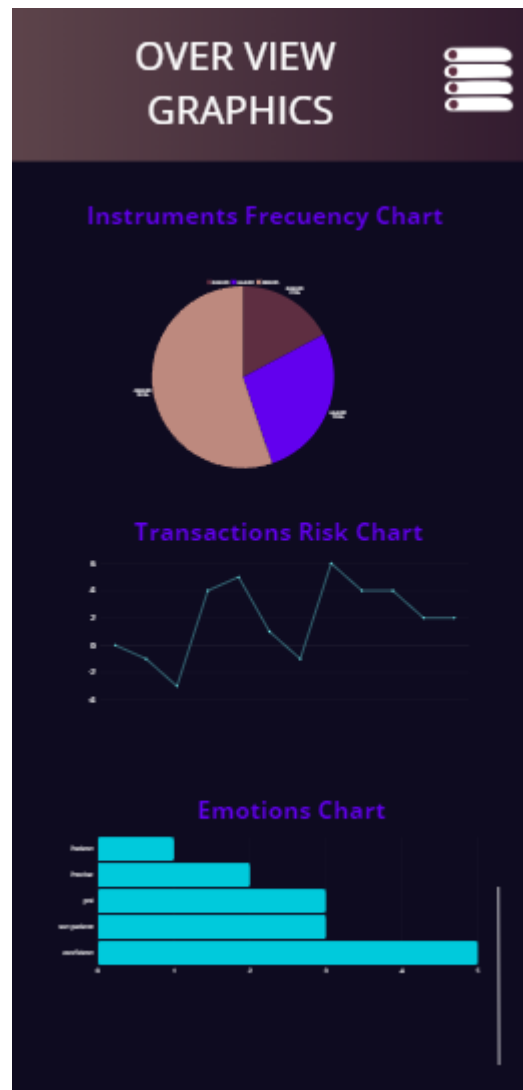
Emotion

Session

Update Transaction

## 7. Over View Graphics de Tendencias y Rendimiento

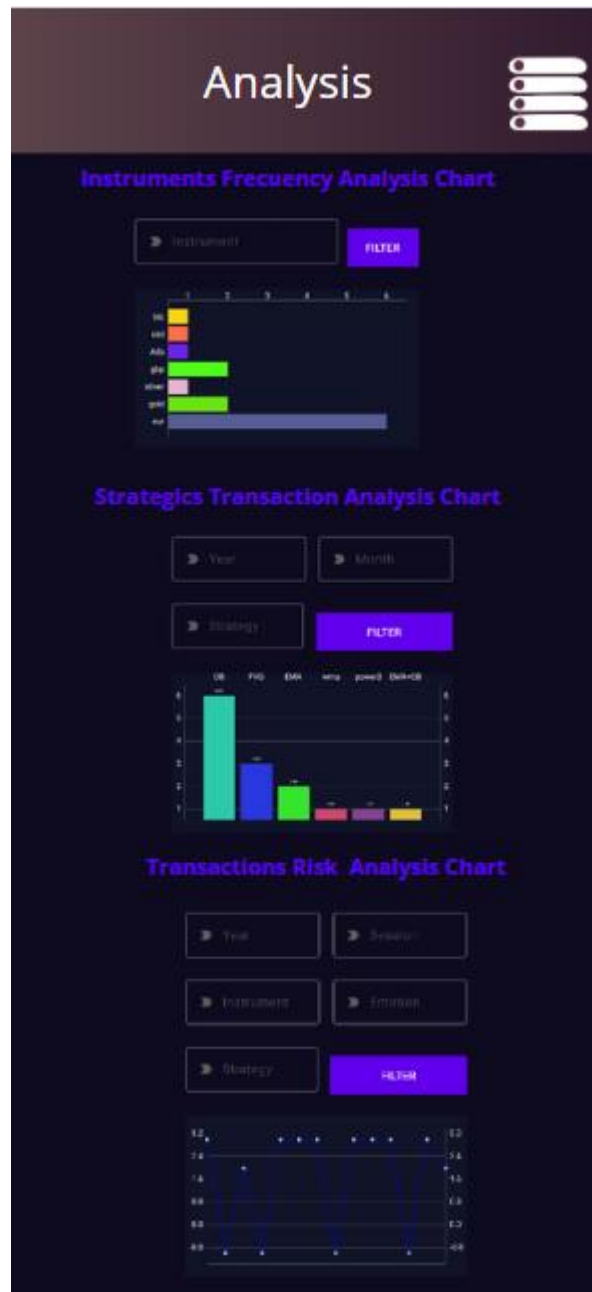
- Acceda a la sección "Over View Graphics" para ver gráficos y estadísticas sobre su rendimiento de trading.



## 8. Análisis de Tendencia y Rendimiento

- Acceda a la sección "Analysis" para proporcionar un análisis visual detallado del rendimiento de trading mediante gráfico. Esta actividad es capaz de filtrar y visualizar datos en base a múltiples parámetros como instrumento, estrategia, entre otros.





## 9. Uso de la Calculadora de Riesgo

- Acceda a la calculadora desde la barra de herramientas.
- Seleccione el instrumento, introduzca el precio de entrada, el stop loss, y el capital que desea arriesgar.
- Toque el botón "Calculate" para obtener el tamaño del lote necesario basado en los parámetros proporcionados.

CALCULATOR

EUR/USD

XAU/USD

Entry Price

Required

Stop Loss

Required

Risk Amount

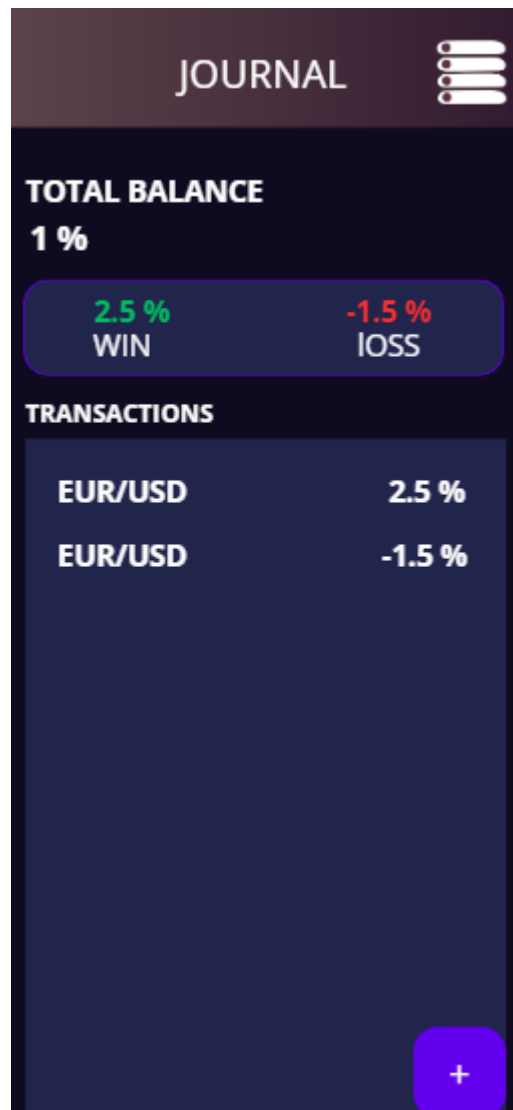
Required

Calculate

Risk Amount	\$\$
Entry Price	##
Stop Loss	##
Lot Size	##

## 10. Visualización del Journal

- El Journal proporciona un resumen visual de sus operaciones, incluyendo ganancias, pérdidas y el balance total en porcentaje.



## 11. Preguntas Frecuentes

### 1. ¿Cómo puedo empezar a usar TradingJournal?

- Respuesta: Simplemente descargue la aplicación desde Google Play Store, instálela en su dispositivo Android y abra la aplicación. No es necesario registrarse, por lo que puede comenzar a agregar transacciones de inmediato.

### 2. ¿Es posible exportar los datos de mis transacciones?

- Respuesta: En la versión actual de TradingJournal, los datos se almacenan localmente y no ofrecemos una función de exportación directa. Estamos considerando agregar esta funcionalidad en futuras actualizaciones.

### 3. ¿Cómo puedo analizar el rendimiento de mis operaciones de trading?

- Respuesta: Puede utilizar la sección 'Analysis' para ver gráficos y análisis de rendimiento basados en sus operaciones ingresadas. Esta sección proporciona visualizaciones detalladas que ayudan a identificar tendencias y patrones.

#### **4. ¿Puedo editar una operación después de haberla guardado?**

- Respuesta: Sí, simplemente vaya a la sección 'Journal', seleccione la operación que desea modificar y haga los cambios necesarios. Asegúrese de guardar los cambios antes de salir de la pantalla de edición.

#### **5. ¿Qué sucede si olvido ingresar una operación?**

- Respuesta: Puede agregar operaciones en cualquier momento. Si se olvidó de agregar una operación en la fecha correcta, simplemente seleccione la fecha adecuada al ingresar los detalles de la nueva operación.

## **12. Solución de Problemas**

### **1. Problema: La aplicación se cierra inesperadamente.**

- Solución: Asegúrese de que su dispositivo tenga suficiente memoria y espacio de almacenamiento disponible. Intente cerrar otras aplicaciones en segundo plano y reiniciar TradingJournal. Si el problema persiste, reinstale la aplicación.

### **2. Problema: No puedo ver los gráficos en la sección de análisis.**

- Solución: Verifique que ha ingresado suficientes datos para generar análisis estadísticos. Los gráficos requieren múltiples entradas de datos para mostrar tendencias y patrones. Si tiene suficientes datos y aún así no aparecen los gráficos, intente reiniciar la aplicación y asegúrese de que su dispositivo está actualizado con la última versión del sistema operativo.

### **3. Problema: Los datos ingresados no se guardan correctamente.**

- Solución: Asegúrese de tocar el botón 'Add Transaction' o 'Update Transaction' después de ingresar o modificar cualquier dato en la aplicación. Si sigue teniendo problemas, verifique el espacio de almacenamiento en su dispositivo, ya que la falta de espacio puede impedir que la aplicación guarde nuevos datos.

### **4. Problema: Errores al intentar agregar una nueva transacción.**

- **Solución:** Verifique que todos los campos obligatorios estén llenos y que todos los datos numéricos o texto ingresados sean válidos. Si el problema continúa, cierre la aplicación completamente y vuelva a abrirla.

### 13. Contacto de Soporte

Si necesita asistencia adicional, no dude en contactar nuestro equipo de soporte técnico a través de:

- **Email:** [support@tradingjournalapp.com](mailto:support@tradingjournalapp.com)
- **Teléfono:** +123 456 7890

Este manual está destinado a ser un recurso valioso para ayudarle a familiarizarse y maximizar el uso de TradingJournal.

## 1.2. Manual de Instalación de TradingJournal

### 1. Introducción

- **Propósito del Documento:** Este manual proporciona instrucciones detalladas para la instalación de la aplicación TradingJournal en dispositivos Android, diseñada para facilitar a los traders el registro y análisis de sus operaciones financieras de forma local utilizando SQLite.
- **Alcance del Documento:** Incluye los requisitos del sistema, procedimientos de instalación y verificación para asegurar una instalación adecuada.

### 2. Requisitos del Sistema

- **Requisitos de Hardware:** Asegúrate de que tu dispositivo Android tenga al menos 2 GB de RAM y 10 GB de espacio libre en disco.
- **Requisitos de Software:** Android 8.0 (Oreo) o superior.
- **Dependencias:** No se requieren dependencias externas, ya que la aplicación utiliza SQLite para la gestión de datos local.

### 3. Preparación de la Instalación

- **Verificación de Requisitos:** Verificar que el dispositivo cumpla con los requisitos mínimos especificados.
- **Configuración del Entorno:** Asegurarse de que el dispositivo esté actualizado y tenga suficiente espacio libre disponible.

### 4. Procedimiento de Instalación

- **Descarga del Software:** Accede a Google Play Store desde tu dispositivo Android.
- **Instrucciones Paso a Paso:**
  - Busca "TradingJournal" en Google Play Store.
  - Selecciona "Instalar" para descargar e instalar automáticamente la aplicación en tu dispositivo.

### 5. Verificación de la Instalación

- **Comprobaciones Post-Instalación:** Abre la aplicación para asegurarte de que se instala correctamente y se ejecuta sin problemas.
- **Resolución de Problemas Comunes:** Si encuentras problemas durante la instalación, verifica la conexión a internet y asegúrate de que el dispositivo tenga suficiente espacio.

### 6. Configuración Inicial

- **Inicio de la Aplicación:** Al iniciar TradingJournal por primera vez, la aplicación configurará automáticamente la base de datos SQLite local.
- **Uso Inmediato:** Puedes comenzar a registrar y analizar tus operaciones de trading sin necesidad de configuraciones adicionales o registro de usuario.

### 7. Apoyo

- **Contacto de Soporte:** Proporciona información de contacto para soporte técnico, que se puede encontrar en la sección "Manual de Usuario".

## Glosario de Términos y Siglas

**TradingJournal:** Nombre de la aplicación diseñada para ayudar a los traders a registrar y analizar sus operaciones financieras de manera sistemática y eficiente.

**SQLite:** Sistema de gestión de bases de datos relacional que se integra en la aplicación para almacenar y gestionar datos localmente sin necesidad de un servidor separado.

**Android:** Sistema operativo para dispositivos móviles desarrollado por Google, sobre el cual funciona la aplicación TradingJournal.

**Operación de Trading:** Cualquier tipo de transacción financiera realizada en los mercados financieros, como la compra o venta de acciones, bonos, divisas, etc.

**Interfaz de Usuario (UI):** Parte del software que interactúa con el usuario, permitiendo que este controle y opere la aplicación.

**Experiencia de Usuario (UX):** Refiere a la experiencia que el usuario tiene al interactuar con la aplicación, que incluye cómo percibe su utilidad, facilidad de uso y eficiencia.

**Funcionalidades:** Capacidades específicas de la aplicación, como registro de operaciones, análisis de tendencias y cálculo de riesgo y beneficio.

**Google Play Store:** Plataforma de distribución digital de aplicaciones móviles para dispositivos con sistema operativo Android, donde los usuarios pueden descargar TradingJournal.

**Tendencias de Trading:** Movimientos generales del mercado que pueden ser identificados y analizados para tomar decisiones informadas sobre operaciones futuras.

**Registro Emocional:** Función de la aplicación que permite a los usuarios anotar sus emociones y sentimientos asociados con operaciones específicas para un mejor análisis del impacto emocional en sus decisiones de trading.

**Calculadora de Riesgo:** Herramienta dentro de la aplicación que ayuda a los usuarios a calcular el riesgo potencial esperado de sus operaciones basándose en parámetros específicos introducidos por el usuario.

**Datos Locales:** Información almacenada directamente en el dispositivo del usuario en lugar de en un servidor remoto o en la nube.

**Desarrollo de Software:** Proceso de concebir, especificar, diseñar, programar, documentar, probar y corregir bugs involucrados en la creación y mantenimiento de aplicaciones y frameworks.

**API (Interfaz de Programación de Aplicaciones):** Conjunto de rutinas y estándares establecidos por software que permiten la comunicación entre distintos componentes de software.

