

CAPITULO I

EL PROYECTO

I.1 Presentación del proyecto

I.1.1 Título del proyecto.

Mejorar las Estrategias de Inversión en el Mercado de Divisas.

I.1.2 Área del proyecto.

Desarrollo de sistemas automáticos de inversión “Trading Algorítmico”.

I.1.3 Director del proyecto.

Aparicio Apellido Paterno	López Apellido Materno	Wilfredo Marcial Nombres	568522 Tja. CI.
Grupo 5	Ingeniería	Facultad de Ciencias y Tecnología	
Grupo de Taller	Informática		
III	Carrera		
----- Teléfono	75120906 Celular	apariciolopez.wilfredo@gmail.com Correo Electrónico	Firma

Tabla 1: Director del proyecto

I.1.4 Institución cooperante

Nombre: “Plataforma de inversión Metatrader 5”
Dirección
https://www.mql5.com/

Tabla 2: Institución cooperante

I.1.5 Duración (meses).

Duración Aproximada de 8 Meses

I.1.6 Actividades previstas para los integrantes del equipo de investigación.

Responsable	Actividades
<p style="text-align: center;">Director</p> <p>Wilfredo Aparicio Marcial López</p>	<p style="text-align: center;">Jefe de Proyecto:</p> <ul style="list-style-type: none"> • Organizar requerimientos del equipo de trabajo. • Planificar las actividades y controlar el cronograma del proyecto. • Asignar y gestionar recursos de los distintos componentes y actividades del proyecto. • Mantener el proyecto enfocado en los objetivos. • Realizar el seguimiento a cada etapa del proyecto. • Supervisar el desarrollo del proyecto. • Presentación final del sistema. <p style="text-align: center;">Analista de Sistemas:</p> <ul style="list-style-type: none"> • Capturar la especificación y validación de requisitos interactuando con los usuarios mediante cuestionarios, entrevistas. • Elaborar el análisis, diseño y desarrollo del sistema. • Elaborar el Modelo de la Base de Datos del Sistema. • Delineación de los Diagramas UML. <p style="text-align: center;">Programador:</p> <ul style="list-style-type: none"> • Realizar la Programación del Sistema Informático. • Desarrollo de prototipos. <p style="text-align: center;">Ingeniero de Software:</p> <ul style="list-style-type: none"> • Preparación de pruebas de validación para el sistema. <p style="text-align: center;">Capacitador:</p> <p>Formar al personal en el manejo del producto final.</p>

Tutor	<ul style="list-style-type: none"> • Asesoramiento de los aspectos tecnológicos para el desarrollo del proyecto. • Asesoramiento en el uso de la Metodología SCRUM de la herramienta UML. • Evaluación del documento del proyecto según agenda.
--------------	--

Tabla 3: Actividades previstas para el equipo de desarrollo

I.2 Descripción del proyecto.

I.2.1 Resumen ejecutivo

Las operaciones realizadas en los mercados bursátiles, siempre han estado caracterizados por ser mercados de gran incertidumbre para los que participan en él debido a su naturaleza, por otra parte se piensa que solo pueden acceder a él personas de alto poder adquisitivo o altamente capacitadas, lo cual no es del todo cierto, es importante mencionar que la mayoría de la clase media o baja desconoce la forma en cómo funciona este tipo de mercado o entiende muy poco o nada de términos financieros básicos, lo cual no ocurre en economías altamente desarrolladas como los países de Norte América o Europa donde las familias de clase media tienen invertido parte de su patrimonio en acciones de una o más compañías.

La falta de orientación básica respecto a operaciones financieras en países latinoamericanos constituye una de las razones fundamentales por las cuales muchas personas de clase media se mantienen al margen de este tipo de mercado, dejando escapar así la oportunidad de generar potenciales ingresos adicionales. Otra cosa curiosa que acontece en este tipo de escenario es que las personas que actualmente participan en este tipo de mercados lo hacen de manera emocional y no de forma racional, es decir que se dejan guiar por sus emociones y/o comentarios de terceros a la hora de comprar o vender una acción, lo cual por lo general produce mínimas ganancias y en el peor de los casos la pérdida del capital inicial resultado de sus operaciones, esto se debe a que la

mayoría de ellos no cuenta con un sistema o instrumento que le ayude a tomar decisiones oportunas a la hora de operar en estos mercados.

Es ahí donde entra en acción el presente proyecto, ya que el mismo está enfocado en la obtención de potenciales beneficios en el corto y mediano plazo resultado de la transacción del intercambio de divisas, si bien no es posible identificar el punto exacto donde realizar una compra o venta, si es posible realizar una aproximación con un pequeño margen de error que es uno de los principales desafíos de esta investigación, para de esta manera estimular a los actuales inversionistas a seguir participando en este tipo de mercados y a los que no a que se animen a hacerlo, ya que se pretende diseñar un modelo que permita reducir las inseguridades que todo inversionista presenta a la hora de tomar decisiones en este tipo de mercado.

1.2.2 Descripción y fundamentación del proyecto

1.2.2.3 Fundamentación. La predicción de la cotización de los activos financieros en bolsa ha sido un tema muy tratado por economistas que han estudiado e intentado comprender los diferentes movimientos en la bolsa de valores, debido a que dichas inversiones están sujetas a riesgos, y que los rendimientos son variables. Debido a este fenómeno es común que los inversionistas recurran a la estrategia convencional de comprar la acción de su preferencia basada principalmente por la emocionalidad y atracción a ciertas compañías o en base a rumores de otros inversionistas, lo cual conduce en la mayoría de los casos al fracaso y pérdida del capital inicial invertido en dichos valores. Por tanto, sería muy provechoso poder predecir la tendencia, y si fuera posible la cotización de dichos activos financieros, ya que con esa información los inversionistas podrían tomar decisiones más acertadas, realizando movimientos apropiados y

de esta manera poder ganar dinero que es el objetivo principal de cualquier inversionista que invierte en bolsa.

La situación es tal que el índice general a nivel mundial que se maneja sobre las inversiones en la bolsa, es que 9 de cada 10 inversionistas no ganan dinero realmente, si bien algunos no lo pierden, pero tampoco lo ganan, y la estadística es más desalentadora para personas que quieren adentrarse en el mundo de las inversiones bursátiles ya que el 98% de los nuevos inversionistas pierde dinero resultado de sus inversiones en bolsa.

Teniendo como referencia dicha información muchas personas piensan que invertir en bolsa es demasiado arriesgado, y es visto más como una apuesta que como una inversión, razón por la cual muchas personas optan por alternativas diferentes de inversión y no se atreven a involucrarse en el mundo bursátil debido a la ignorancia e incertidumbre que genera este tipo de mercado.

I.2.2.2 Justificación académica. La Universidad tiene como misión el formar profesionales integrales, con valores éticos y morales; creativos e innovadores; solidarios y con responsabilidad social; con el fin de contribuir mediante la investigación científico-tecnológica y la extensión universitaria, vinculadas a las demandas y expectativas del entorno social.

Es por esto que se desarrolla el papel importante, de intervenir en los intereses de la población, dando respuesta a sus necesidades. Con los conocimientos adquiridos en la formación universitaria en el entorno tecnológico.

I.2.2.3 Justificación social. El desarrollo de sistemas inversión en los mercados financieros, hoy en día, se constituyen en un apoyo muy importante para grandes y pequeños inversores, en especial los inversores que no cuentan con gran conocimiento en el análisis de estos mercados,

Al realizar el presente proyecto se contribuirá al desarrollo e investigación de este tipo de sistemas.

I.2.2.4 Justificación tecnológica. Nuestro medio cuenta con la tecnología adecuada que se requiere para desarrollar el sistema de inversión propuesto. Ya que los requerimientos son accesibles para un inversor experimentado o un inversor en desarrollo.

En el presente proyecto se hará uso de las tecnologías asociadas a estos mercados, además de las herramientas más actualizadas, contribuyendo al desarrollo de la tecnología en nuestra región.

I.2.2.5 Análisis de involucrados.

Grupo	Intereses	Problemas percibidos	Recursos y mandatos
Inversionista	Herramienta encargada de analizar fenómenos y comportamiento del mercado a través de los números de forma verídica o analítica	<ul style="list-style-type: none"> ➤ Complejidad para identificar la tendencia del mercado bursátil. ➤ Variación porcentual de las cotizaciones. ➤ Volumen de operaciones del mercado bursátil. 	Verificar si las cotizaciones de los activos financieros dentro de los mercados, se comportan de acuerdo a una tendencia identificable.

Tabla 4: Análisis de involucrados

I.2.2.6 Análisis de causas de problemas

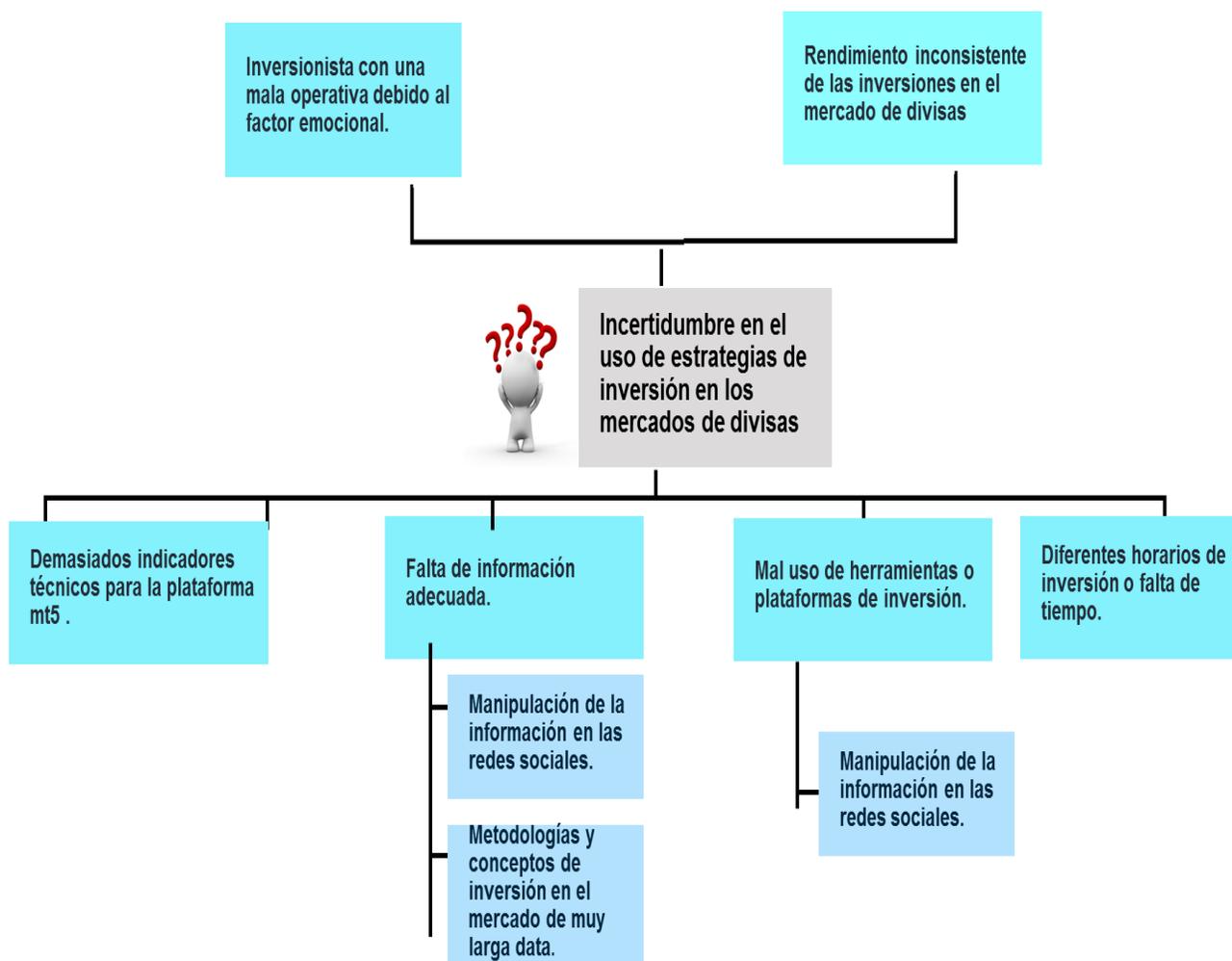


FIGURA 1: Árbol de problemas

I.2.2.7 Análisis de objetivos

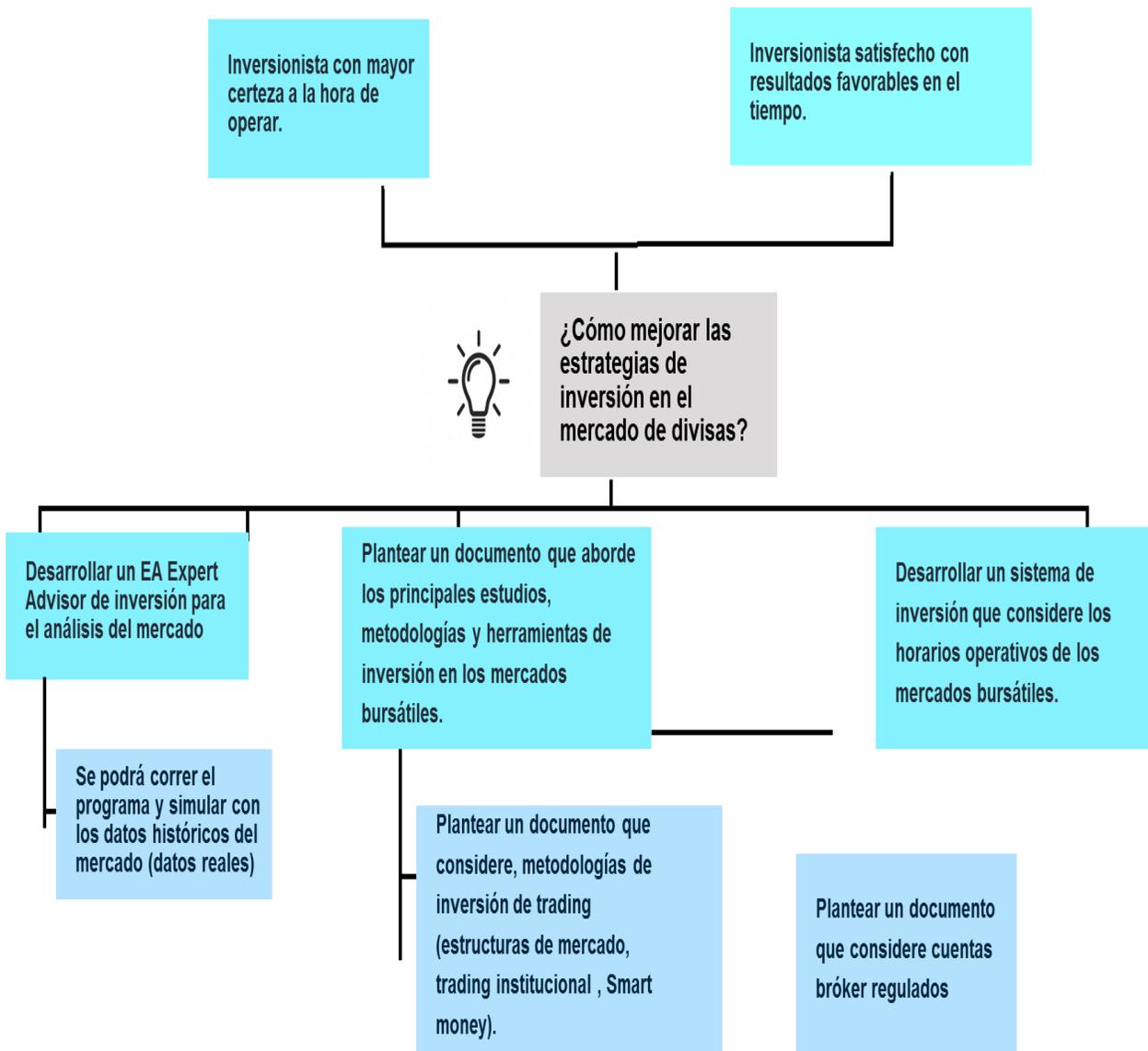


FIGURA 2: Árbol de objetivos

I.2.3 Objetivos.

I.2.3.1 Objetivo general.

Diseñar una estrategia para operar en un sistema automático de inversión en el mercado de divisas.

I.2.3.2 Objetivos específicos.

- Analizar y revisar la literatura correspondiente al funcionamiento de los mercados financieros, así como también las principales teorías del comportamiento de los activos financieros que se negocian en él.
- Indagar sobre los principales elementos, indicadores y parámetros que deben tomarse en cuenta a la hora de participar en los mercados de divisas, así también como la revisión de los principales métodos que existen para predecir activos financieros.
- Desarrollar un sistema automatizado de inversión (robot de trading algorítmico) para la plataforma Metatrader 5 que permita mejorar las decisiones de inversión con activos financieros en el mercado de divisas.
- Realizar una simulación con datos reales del sistema automatizado de inversión creado y plantear un escenario de inversión.

I.2.3.3 Situación planteada con y sin proyecto.

Situación sin proyecto	Situación con proyecto
<p>➤ Con la gestión de las operaciones actuales que se realizan de forma manual, la consistencia en los beneficios de las inversiones no es estable. Y muchas veces negativas.</p>	<p>➤ La gestión de las operaciones se realiza de manera automática siguiendo una estrategia definida en el sistema de inversión que se convierte en una herramienta indispensable siempre disponible y confiable para el inversor.</p>

Tabla 5: Análisis de situación con y sin proyecto de la empresa

I.2.4 Metodología de desarrollo del proyecto.

I.2.4.1 Metodología de desarrollo ágil. Los procesos ágiles de desarrollo de software, conocidos anteriormente como metodologías livianas, intentan evitar los tortuosos y burocráticos caminos de las metodologías tradicionales enfocándose en la gente y los resultados.

Es un marco de trabajo conceptual de la ingeniería de software que promueve iteraciones en el desarrollo a lo largo de todo el ciclo de vida del proyecto. Existen muchos métodos de desarrollo ágil.

El software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requerimientos, diseño, codificación, revisión y documentación. Una iteración no debe agregar demasiada funcionalidad para justificar el lanzamiento del producto al mercado, pero la meta es

tener un demo (sin errores) al final de cada iteración. Al final de cada iteración el equipo vuelve a evaluar las prioridades del

I.2.4.2 Metodología scrum. Scrum es una metodología que se caracteriza por ser adaptable al cambio que permite tener productos entregables y funcionales a medida que avanza el desarrollo de forma iterativa e incremental. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que podamos mejorar.

Cada componente dentro del marco de trabajo sirve a un propósito específico y es esencial para el éxito de Scrum y para su uso. Las reglas de Scrum relacionan los eventos, roles y artefactos, gobernando las relaciones e interacciones entre ellos. (Schwaber & Sutherland, 2013)

I.2.4.2.1 Elementos de scrum: dentro del marco de scrum se les conoce a todos los elementos de la gestión de transparencia como artefactos de scrum, están pensados y diseñados para poder garantizar la transparencia dentro de tu equipo de scrum y que puedas evitar el riesgo de tener una mala comunicación, y mal manejo de información en la entrega de tu proyecto. Por ello estos son los elementos:

I.2.4.2.1.1 Sprint: Es el corazón de Scrum, puesto que son los ciclos o iteraciones que se tendrá dentro de un proyecto, el sprint es un intervalo de tiempo durante el cual se crea un incremento de producto que aporte valor al cliente. La duración habitual de un Sprint debería estar entre dos semanas y un máximo de ocho semanas (Guerrero Amaya et al.,2017).

I.2.4.2.1.2 Incremento: Es una versión terminada del producto durante un Sprint y que además está totalmente operativa, según los resultados que se obtengan, el cliente puede ir haciendo los cambios necesarios en el proyecto. (Trigas Gallego, 2012)

I.2.4.2.1.3 Requisitos del producto (Product Backlog): Contiene todos los requerimientos funcionales y no funcionales priorizados que tendrá el producto. Los mismos serán especificados en: casos de uso, diagramas de flujo de datos, prototipos, etc. El Product Backlog será definido en reuniones de planeamiento con los stakeholders y el Product Owner. A partir de ahí se definirán los Sprints, que servirán para ir aumentando el producto de una forma evolutiva (Calderón & Valverde Rebaza, 2007).

I.2.4.2.1.4 Requisitos de la iteración (Sprint Backlog): Contiene la lista de tareas y/o requerimientos que se realizarán en un Sprint, además aporta visibilidad al dueño del producto, ScrumMaster y al equipo Scrum para tener claro en que deben concentrar sus esfuerzos durante el Sprint. La duración promedio de un Sprint es de entre 2 a 4 semanas como máximo (Guerrero Amaya et al., 2017).

I.2.4.2.2 Roles. Personas involucradas que tienen diferente cargo en el momento de desarrollar el SCRUM.

I.2.4.2.2.1 Product owner (dueño del producto). Según [Henrik Kniberg y Mattias, 2010]. El Dueño de Producto es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto podría variar ampliamente entre distintas organizaciones. El Dueño de Producto es la única persona responsable de gestionar la Lista del Producto (Product Backlog).

I.2.4.2.2 Scrum master (Líder del proyecto). Responsable del proceso SCRUM, de cumplir la meta y resolver los problemas. Así como también, de asegurarse que el proyecto se lleve a cabo de acuerdo con las prácticas, valores y reglas de SCRUM y que progrese según lo previsto. (Palacios, 2008)

I.2.4.2.3 Fases del proceso scrum. Según (Palacios, Juan, 2008). Es una metodología Ágil, está basada en iteración y revisiones. El ciclo de vida de SCRUM está compuesto de tres fases que son el pre – Game, Game y el post – Game.

I.2.4.2.3.1 Pre – game. Las tareas que se realizan en esta primera etapa son:

Planeación: Todos los miembros del equipo incluyendo el cliente se reúnen para determinar el análisis del problema. En este paso se puede dividir las tareas en:

Recopilación: Donde se extrae los requerimientos para conformar el producto backlog, priorizados de acuerdo al cliente y los usuarios que interactúan con el proyecto.

Análisis de riesgos y controles apropiados para los riesgos, la selección del tipo de herramienta a trabajar, cálculo y la estimación del costo.

Arquitectura: El objetivo de esta etapa es diseñar como los elementos del backlog del producto serán puestos en ejecución. Se revisa los ítems del backlog, el análisis y el tiempo aproximado para terminar la tarea.

I.2.4.2.3.2 Game. Una vez realizado el pre – Game se opta por realizar los siguientes puntos:

Planeación del Sprint. Antes de comenzar cada sprint, se lleva a cabo reuniones para refinar y priorizar nuevamente el producto backlog luego pasara a ser un Sprint backlog con las Actividades realizadas, los responsables y la duración de cada actividad.

Desarrollo de Sprint. El trabajo generalmente se organiza en iteraciones de 2 a 3 semanas. El sprint es el desarrollo de la nueva funcionalidad del producto. Esta fase provee la siguiente documentación.

Revisión del Sprint. Al final de cada iteración se lleva a cabo una reunión de revisión en donde se encuentra la nueva funcionalidad del producto, las metas incluyendo la información de las funciones, diseño ventaja, inconvenientes y esfuerzo del equipo.

I.2.4.2.3.3 Post – game. La etapa final, denominada según SCRUM, es el cierre o Post – Game: En esta última etapa se realiza la preparación operacional, incluyendo la documentación final necesaria para la prestación.

Realizando las Pruebas de Rendimiento o Esfuerzo del Proyecto, también a esta etapa se debe realizar dependiendo del tipo de producto las interfaces finales para el usuario y el entrenamiento del Plantel (usuarios) o el marketing para la venta del nuevo producto.

I.2.4.2.4 Reuniones de scrum. Si hablamos del método Scrum, que se basa en la división del trabajo en etapas y la retroalimentación entre el líder del proyecto y el cliente, el éxito de cualquier proyecto pasa por atender un elemento en concreto: la comunicación. Esto se logra básicamente a través de las reuniones de equipo, las cuales deben atender a necesidades concretas

I.2.4.2.4.1 Reunión de planificación del Sprint (Sprint Planning Meeting): Es una reunión en la que participan el Product Owner, el Scrum Master y el equipo Scrum, aquí se definen: que tareas se tienen que realizar y cuáles son sus objetivos, si durante el desarrollo del Sprint no se lo puede realizar por algún inconveniente se puede hacer una nueva planificación, si el equipo no puede comprometerse a cumplir todo el Backlog se consultará al Product Owner para decidir qué ítems se deben eliminar o agregar al Sprint. Considerar que el Backlog tiene que estar definido antes de esta reunión (Manuel Trigas Gallego, 2012).

I.2.4.2.4.2 Reunión diaria (Scrum Daily Meeting): En esta reunión, el equipo comenta que están realizando, las tareas que han terminado y si se tiene alguna dificultad. También se comparte información relativa al desarrollo y colaborarán para hacer las adaptaciones necesarias, aumentando así su productividad. Esta reunión tiene como particularidad el Backlog del Sprint y no suele durar más allá de 15 minutos. Además, se deben contestar tres preguntas básicas (Menuel Trigas Gallego, 2012): - Dependencias:

¿Qué aporte, he dado desde la última reunión, para apoyar a cumplir el objetivo de todo el equipo?

Progreso: ¿Cómo voy a ayudar a cumplir el objetivo?

Obstáculos: ¿Qué obstáculos existen o podrían surgir, que desencadenen el cumplimiento del objetivo?

I.2.4.2.4.3 Reunión de revisión del sprint (Sprint Review Meeting): Esta reunión se lleva a cabo una vez finalizado el Sprint, aquí el equipo presenta el producto entregable y el Product Owner analiza si la entrega cubre lo pactado en el Sprint Backlog, si la funcionalidad no está completa no se puede hacer la presentación. También se hace un análisis sobre los problemas obtenidos durante el proceso de implementación del entregable. Esta reunión sirve para tomar decisiones sobre lo más adecuado para alcanzar el objetivo deseado y tiene una duración de máximo 4 horas (1 hora por semana / iteración) (Deemer, 2009)

I.2.4.2.4.4 Reunión de retrospectiva del sprint (Sprint Retrospective Meeting): En esta reunión, el equipo analiza aspectos relacionados con el último Sprint finalizado y los cambios que se podrían hacer para mejorar el próximo Sprint y que este sea más productivo.

Por lo general quien organiza la reunión es el ScrumMaster y tiene una duración máxima de 3 horas, los asistentes a esta reunión son: El ScrumMaster, el equipo y el Product Owner(Deemer et al., 2009).

1.2.4.2.5 Ingeniería web Según el sitio oficial de UWE [Web Engineering Group,2014]. La Ingeniería Web propone nuevos métodos para el diseño de aplicaciones que se ejecutan en esta nueva plataforma que es la World Wide Web. Uno de estos métodos es UWE (UML Web Engineering), el cual aprovecha la notación estándar del UML e incorpora elementos que son propios del desarrollo Web y representa por un caracterizado logo.

1.2.5 Resultados esperados

1.2.5.1 Sistema Automático de Inversión. - Se tendrá un sistema automático de inversión para el inversor independiente, que funcionará sobre plataforma Metatrader 5, automatizado 90% de sus funciones. Este sistema permitirá operar de forma automática en el mercado de divisas y todo lo necesario para garantizar la eficiencia y calidad de funcionamiento del sistema, también se obtendrá toda la documentación de desarrollo del proyecto y del sistema, y el manual de usuario, cumpliendo así el propósito del proyecto y contribuyendo al inversor de este tipo de mercados y nuevos inversores que se interesen por esta área de la tecnología.

1.2.5.2 Backtesting y Análisis de Datos del sistema. - una vez se considera que el sistema esté terminado, se procede a realizar las pruebas para identificar el comportamiento del algoritmo en las condiciones del mercado. Consiste en hacer correr el algoritmo en las condiciones pasadas del mercado. Es decir, identificar los resultados del algoritmo con los datos históricos del mercado (datos reales). De esta forma se tiene un primer acercamiento a la eficacia y eficiencia de las operaciones realizadas. Se obtiene información que permite proyectar y analizar diferentes

escenarios. Análisis de resultados, con los datos obtenidos a través del BACKTESTING se realizan análisis estadísticos planteando diferentes escenarios futuros del mercado, con el fin de analizar y optimizar e ir llenando los criterios y parámetros que permitan una posible selección del algoritmo para su ejecución real.

1.2.6 Beneficiarios del proyecto

1.2.6.1 Beneficiarios Directos: El grupo de beneficiarios son todos los Inversionistas que operen el mercado de divisas y mercados bursátiles que se interesen por el trading algorítmico serán los beneficiados directos con el proyecto.

1.2.6.2 Beneficiarios indirectos: Los potenciales inversionistas o personas interesadas en las inversiones dentro de los mercados bursátiles, así como también estudiantes que quieran profundizar más sobre el estudio y desarrollo de este tipo de sistemas automáticos de inversión serán los beneficiados con el desarrollo de proyecto.

I.2.7 Cronograma de actividades

	Actividad	N° días	Fecha inicio	Fecha fin	M1	M2	M3	M4	M5	M6	M7
.	Componente 1: Sistema automático de inversión.	158									
.1	Relevamiento de información (análisis de requerimientos según la norma IEEE 830 y modelado con casos de uso)	40	13-05	20-06	x	x					

.2	Análisis y Diseño del sistema (modelado del sistema mediante UML uso de metodología SCRUM)	55	21-06	20-08		x	x	x			
.3	Construcción del sistema (Programación del sistema de inversión)	58	21-08	23-10				x	x	x	
.4	Evaluación del Sistema	1	23-10	23-10						x	

.5	Validación del sistema	2	24-10	24-10							x
.6	Estrategia Aceptada	5	25-10	29-10							
.	Componente 2: Capacitación del sistema.	14									
.1	Elaboración de manuales de instalación y de usuario	6	30-10	05-11							x
.2	Planificación de la capacitación	5	06-11	10-11							x

	(planificación)										
.3	Ejecución de la capacitación	3	10-11	12-11							x
	(Realización de las jornadas de capacitación durante 2 días, 2 horas diarias)										

Tabla 6: Cronograma de actividades

I.2.8 Marco lógico del proyecto

Resumen Narrativo del Proyecto	Indicadores	Medios de Verificación	Supuestos
<p>Fin Contribuir a mejorar los rendimientos obtenidos resultado de la operativa en el mercado de divisas.</p>	<p>A partir de la implementación del robot de trading algorítmico la forma de participar en el mercado de divisas pasa de ser manual a ser 100% automática.</p>	<p>La plataforma Metatrader 5 nos permite generar un informe visual sobre las operaciones realizadas.</p>	<p>El Computador del inversor está encendida, con conexión a internet y ejecuta el robot de trading.</p>
<p>Objetivo General (Propósito) Estrategia mejorada, para operar en un sistema automático de inversión en el mercado de divisas.</p>	<p>Al finalizar el proyecto, se pone a disposición de inversores una estrategia que, en condiciones normales, asegura una rentabilidad al menos del 5% mensual, del capital inicial.</p>	<p>Los Resultados son verificados por los asesores el docente y el tribunal en la defensa del proyecto.</p>	<p>El computador del inversor debe cumplir los requisitos recomendados para la ejecución de la plataforma metatrader 5. El inversor inicia sesión en su cuenta de trading del bróker registrado. El inversor selecciona la divisa que va operar el robot de trading.</p>

Objetivos Específicos (Componentes)			
<p>1. Sistema automatizado de inversión (robot de trading algorítmico) para la plataforma mt5 que permita mejorar las decisiones de inversión con activos financieros en el mercado de divisas, desarrollado.</p> <p>2. Capacitación a los involucrados en el uso y los beneficios del sistema de trading algorítmico, Realizado.</p>	<p>1. Al finalizar el proyecto se ha desarrollado un sistema en un 90% automatizado de inversión. Para la participación en el mercado de divisas.</p> <p>1. Al culminar el proyecto se capacita a los traders en una conferencia virtual. En dicha capacitación se explica el uso y las ventajas del sistema en el Mercado de Divisas. A principios de noviembre del 2020 se ha capacitado a los usuarios que desean operar de forma automática .</p>	<p>1. Documentación del sistema desarrollado completo.</p> <p>2. Simulación con datos reales de sistema automatizado de inversión creado</p> <p>1. Material de capacitación con el registro de firmas de los partícipes en los cursos de capacitación.</p> <p>2. Fotografías tomadas durante la capacitación.</p>	<p>Se tiene todo el acceso y conocimiento necesario de manera oportuna para desarrollar el sistema.</p> <p>Se elabora el documento de manera oportuna y en los plazos establecidos del desarrollo de proyecto.</p> <p>1. Los participantes tienen acceso a internet y cuentan con una computadora o celular al momento de la capacitación, lo que permite una comunicación correcta.</p>
<p>Actividades:</p>	<p>Entre la Fecha: 21/06/2022 hasta 20/08/2022. Se recolecto al</p>		

<p>COMPONENTE 1</p> <p>1. Diseño del sistema y determinación de requerimientos</p> <p>3. Desarrollo del sistema.</p> <p>4. Evaluación.</p> <p>5. Validación.</p> <p>6. Estrategia Aceptada</p>	<p>menos el 80 % de los requerimientos, basados en la norma IEEE 830. para el desarrollo del sistema.</p> <p>Entre la Fecha: 21/08/2022 hasta 23/10/2022. Se desarrollo el sistema de Trading algorítmico con al menos un 90% de su programación en sus respectivos esprints y diagramas UML, según la Metodología Scrum.</p> <p>Entre la Fecha: 23/10/2022 hasta 25/10/2022. Se Realizó al menos en un 85% de la evaluación del Sistema.</p> <p>En la Fecha: 24/10/2022. Se desarrolló al menos un 90% de la validación sistema.</p> <p>Entre la Fecha: 24/10/2022 hasta 29/11/2022. Se llevó acabo la aceptación de al menos en un 85% estrategia a través de una</p>		<ul style="list-style-type: none"> ➤ Se cuenta con todas las herramientas necesarias para la elaboración del sistema de inversión. ➤ Los Utilización de las normas de modelado a los diagramas de análisis y diseño del sistema, mediante metodología Scrum. ➤ Documentación del método de la estrategia.
--	---	--	--

<p>COMPONENTE 2</p> <p>1. Socialización del uso de la Tecnología.</p> <p>2. Definido Medios y estrategias de sociabilización.</p> <p>3. Llevado a cabo de las estrategias de capacitación en el manejo del sistema informático.</p>	<p>prueba de realizando una operación real.</p> <p>Entre la Fecha: 30/10/2022 hasta 10/11/2022. Se Analizó al menos en un 85% los medios de socialización a través de las TIC</p> <p>Entre la Fecha: 10/11/2022 hasta 12/10/2022. Se desarrolló al menos en un 90% la capacitación del sistema de trading Algorítmico.</p> <p><u>Resumen presupuesto</u></p> <p>20000. Servicios No personales: 2.760 Bs.</p> <p>30000 materiales Y suministros: 669Bs.</p> <p>40000 activos Reales: 800 Bs.</p> <p>Total, Bs: 4.229 Bs.</p>	<p>➤ Manuales de usuario y de instalación impresos.</p> <p>➤ Informe de ejecución de la capacitación, emitido por la empresa.</p>	<p>➤ Los participantes cuentan con acceso a la plataforma de zoom por Google o por la aplicación en un celular o computadora para la capacitación.</p>
---	---	---	--

Tabla 7: Matriz de marco lógico

I.2.9 Presupuesto y Justificación del desarrollo del proyecto

IT EM	RUBROS	Aporte Universidad	Otro Aporte	TOTAL (Bs.)
20000	SERVICIOS NO PERSONALES		----	
	21000. Servicios Básicos		----	2760
	22000. Servicios de transporte		----	
	23000. Alquileres		----	
	24000. Mantenimiento y reparación		----	
	25000. Servicios Profesionales y Comerciales		----	
	Sub total rubro			2760
30000	MATERIALES Y SUMINISTROS			
	31000. Alimentos y Productos Forestales	----	----	
	32000. Productos de Papel, Cartón e Impresos	----	----	320

	33000. Textiles y Vestuario.			
	34000. Productos Químicos, Combustibles y Lubricantes	---	---	349
	39000. Productos Varios.			
	Sub total rubro			669
40000	ACTIVOS REALES	----	----	----
	43000. Maquinaria y Equipo.			800
	46000. Descripción de estudios y proyectos para inversión.			
	49000. Otros Activos			
	Sub total rubro			800
	TOTAL			4229

Tabla 8: Presupuesto y justificación del desarrollo del proyecto

- GRUPO 20000. SERVICIOS NO PERSONALES

1.4.1 SUB GRUPO 21000. Descripción de los gastos de servicios básicos

Partida	Tipo de servicio básico *	Costo	Tiempo mes	Costo Total
21100	Comunicación	250	8	2000
21200	Energía Eléctrica	60	8	480
21300	Agua			
21400	Servicios Telefónicos	35	8	280
Total				2760

Tabla 9: Descripción de Gastos de Servicios básicos

*Se refiere principalmente a los gastos por servicios; como: servicio de correo, radiogramas, servicio telefónico, fax, Internet.

- SUB GRUPO 32000. Descripción del gasto de Productos de Papel, Cartón e Impresos

Partida	Tipo de material *	Cantidad	Costo/Unitario	Total
32100	Papel de Escritorio Resmas hojas bond	8	40	320
32200	Productos de Artes Gráficas, Papel y Cartón			
32300	Libros			
32400	Textos de Enseñanza			
Total				320

Tabla 10: Descripción del gasto de Productos de Papel, Cartón e Impresos

*Se refiere a la adquisición de; papel y cartón en sus diversas formas y clases, impresos y publicaciones, periódicos, revistas, libros, fotocopias, etc.

- GRUPO 39000. Descripción del gasto en productos varios

Partida	Productos de cuero y caucho	Cantidad	Costo/Unitario	Total
39100	Material de Limpieza			
39400	Instrumental Menor - Médico Quirúrgico			
39500	Útiles de Escritorio y de Oficina	1	25	25
	Agenda personal	6	3	18
	Bolígrafos, lápices	20	3	60
	CD, DVD	3	2	6
	Tinta negra	4	60	240
	Tinta a colores			
39700	Útiles y Materiales Eléctricos			
39800	Otros Repuestos y Accesorios			
Total				349

Tabla 11: Descripción del gasto en productos varios

*Se refiere principalmente a los gastos por productos de limpieza, todo lo referente a la funcionamiento de la oficina en material de escritorio.

- GRUPO 40000. ACTIVOS REALES

1.4.11 SUB GRUPO 43000. Descripción del gasto de Maquinaria y Equipo

Partida	Tipos de productos	Cantidad	Costo/Unitario	Total
43100	Equipo de Oficina y Muebles Impresoras	1	800	800
43200	Maquinaria y Equipo de Producción			
43300	Equipos de Transporte, Tracción y Elevación			
43400	Equipo Médico y de Laboratorio			
43700	Otra Maquinaria y Equipo			
Total				800

Tabla 12: Descripción del gasto de Maquinaria y Equipo

- Se refiere principalmente a los gastos por muebles y enseres, equipo de oficina, comunicación, equipamiento.

CAPITULO II

COMPONENTES

II.1. COMPONENTE I:

Mejorar las estrategias de inversión en el mercado de divisas.

II.1.1 Marco teórico.

II.1.1.1 Objetivos.

II.1.1.1.1 Objetivo general : Desarrollar un sistema informático Basado en Expert advisor proporcionados por la plataforma metatrader 5.

II.1.1.1.2 Objetivos específicos

- Recolectar la mayor cantidad de información que permita entender el funcionamiento de los mercados financieros.
- Analizar toda la información recolectada para así poder determinar junto con el personal involucrado, los requerimientos que cubrirá el sistema.
- Diseñar y estructurar el sistema de forma que este cumpla todos los requerimientos identificados y además pueda soportar futuros cambios.
- Desarrollar los módulos especificados y contemplados para el sistema
- Elaborar el sistema tomando en cuenta las tecnologías que mejor se adapten a su propósito.
- Aplicar el lenguaje de modelado unificado (UML) para el desarrollo del sistema.
- Aplicar la metodología de desarrollo SCRUM

II.1.1.2 Antecedentes. Inversión en Valor fue concebido por Benjamin Graham poco después del desplome de Wall Street de 1929, impulsado por conseguir una forma de inversión más segura. Poco después en 1934 con ayuda de David Dodd publicaron lo que sería y es, una de las mayores referencias literarias en el mundo de las inversiones “Security Analysis”. En este libro sentaron las bases de lo que se conoce ahora por inversión en valor. Su principal objetivo, es la diferenciación entre valor y precio. Busca oportunidades en el mercado en compañías en las que el precio de mercado es inferior al valor real de la compañía, lo que definió como “margen de seguridad”. La acción en el largo plazo debería tender a reducir este margen de seguridad, ajustando el precio de la cotización a su valor fundamental.

La mayor crítica que se le puede realizar este estilo de inversión es la ausencia de información perfecta en el mercado. Esto nos puede conducir a malas estimaciones de lo que consideramos el valor de una empresa. No porque los modelos de valorización de las empresas sean erróneos si no porque tenemos una percepción parcial de lo que es la empresa.

II.1.1.3 Estudio de Variable independiente

II.1.1.3.1 Mercado de divisas

II.1.1.3.1.1 Definición. Actualmente se puede definir un mercado como el espacio, la situación o el contexto donde se lleva a cabo el intercambio, la venta y la compra de bienes, servicios o mercancías por parte de unos compradores que demandan esas mercancías y tienen la posibilidad de comprarlas, y unos vendedores que ofrecen éstas mismas.

“Se denomina divisa al dinero real y a cualquier medio de pago cifrado en una moneda distinta a la doméstica” (ELERONOMISTA, s.f.) Es decir, hace referencia al grupo de monedas extranjeras que se utilizan en un país para llevar a cabo, actividades o transacciones a nivel internacional.

Teniendo en cuenta la definición de mercado y de divisas, se puede decir que el mercado de divisas es el espacio, situación o el contexto donde se lleva a cabo el intercambio, la venta y la compra de monedas extranjeras llamadas divisas. Al ser un mercado, el precio de las divisas está determinado por la oferta y la demanda que existan sobre ellas.

El mercado de divisas (también conocido como Forex, abreviatura del término inglés (FOREign Exchange Market) es un mercado interbancario y descentralizado, donde se intercambia una divisa por otra.

En general es un conjunto de transacciones con volúmenes y precios determinados entre los agentes del mercado internacional de divisas, que intercambien una moneda por otra. El tipo de cambio es determinado por los factores de mercado – la demanda y oferta.

Forex, se caracteriza por tener un alto grado de liquidez, gracias a que el volumen diario de transacciones en el mercado es equivalente a 3 trillones de dólares (USD), este número de operaciones en el mercado internacional de divisas crece constantemente debido al desarrollo del comercio internacional y eliminación de restricciones monetarias en muchos países.

Adicionalmente Forex no tiene ninguna ubicación física. Todo funciona a través de la red electrónica de bancos, corporaciones y personas que negocian una determinada divisa; gracias a esta característica, el mercado Forex puede funcionar las 24 horas del día, pasando de una zona horaria a otra de esta manera cubriendo los principales mercados financieros.

“El mercado de divisas FOREX es la base para las transacciones internacionales de capitales y su liquidez y volumen son mucho mayores que cualquier otro mercado financiero”

(¿Que es Forex?, s.f.)

II.1.1.3.1.2 Tipos de operaciones en el mercado de divisas

II.1.1.3.1.2.1 Transacciones de contado “Spot” En términos del mercado de divisas, las operaciones al contado o spot son aquellas en las que una divisa se intercambia por otra a un precio determinado, con la obligación para ambos participantes de entregar las respectivas cantidades de moneda, no más tarde de dos días hábiles (los sábados no son considerados como tales), después de haberse efectuado la operación.

Es decir, las operaciones al contado son aquellas que se liquidan a más tardar dos días hábiles posteriores a su fecha de contratación.

II.1.1.3.1.2.2 Transacciones de plazo Cualquier transacción en el mercado de divisas que implique la entrega de las mismas en un plazo superior a dos días hábiles después de haberse efectuado la operación es denominada operación a plazo. Raramente el precio de la divisa al contado coincide con el precio futuro o a plazo (forward price), influyendo en esta disparidad dos factores principalmente:

- a) la diferencia de los tipos de interés de las respectivas divisas
- b) la tendencia al alza o a la baja de las propias divisas

Los precios *forward* o futuros normalmente son diferentes a los precios spot o contado; puede suceder que el precio a plazo sea superior al de contado, lo que se conoce como que la divisa *cotiza con prima a plazo*, o por el contrario, que el precio al contado, es superior al precio a plazo, y se diría que la divisa cotiza con descuento.

Normalmente, las monedas débiles cotizan con descuento frente a las fuertes, cotizando éstas con prima frente a las débiles: (López Domínguez, s.f.)

II.1.1.3.1.3 Funciones y características del mercado de divisas



Ilustración 1/Figura 3 Funciones mercado de divisas

(Recuperado de :Mercado de Divisas, s.f.)

Figura 4. Características del mercado de divisas

Comprar y vender divisas

Tomando como ejemplo el caso del EUR/USD podríamos decir que, si un inversionista o especulador compra el par, sería correcto decir que este está comprando euros y vendiendo dólares.

También se puede decir que el inversionista está recibiendo euros a cambio de dólares. Al momento de vender el EUR/USD se podrá decir, por lo tanto, que el inversionista está comprando dólares y vendiendo euros o que está recibiendo dólares a cambio de euros (Weithers, 2006). Para el ejemplo de la tabla III.1 si un inversionista comprara 1 EUR/USD a su precio de apertura de 1.13537 y lo vendiera a 1.3555, este obtendría una ganancia de 0.00018 dólares por euro. Por supuesto estas cifras parecen insignificantes cuando hablamos en términos de 1 unidad monetaria, pero cuando hablamos de 100.000 unidades monetarias, la ganancia para nuestro ejemplo anterior sería de 18 dólares, un movimiento que puede suceder en minutos si no en segundos. En la actualidad cuando hablamos del mercado forex spot, por lo general estaremos hablando de un

mercado donde existe apalancamiento. En palabras sencillas, el apalancamiento es cualquier técnica utilizada para producir resultados superiores a los normales (Block & Hirt, 2008).

¿Qué es un Pip?

Se define un pip como el mínimo cambio posible en el valor de un par de divisas. En este caso la cantidad de pips o puntos mide la diferencia entre el precio de entrada y el precio de salida en una operación Forex cualquiera.

Por el ejemplo si el GBP/USD se mueve de 1.6225 a 1.6250, eso es una subida de 25 pips. Se puede definir un pip como el último decimal en la cotización. Es por medio de los pips que se calculan las ganancias o pérdidas en una operación. Debido a que cada divisa tiene su propio valor, se hace necesario calcular el valor de un pip para cada divisa en particular. (¿Que es un Pip?, s.f.)

¿Qué es un Lote?

Los lotes en Forex están definidos en relación a la cantidad de dinero en cada transacción. Dependiendo del monto de cada lote se dividen en los siguientes tipos: Tipos de lotes en el Forex

- **Lote estándar:** Son los lotes formados por una cantidad de 100000 unidades de la divisa base en cualquier par. En este caso el monto que debe poner el inversionista depende del apalancamiento que se use. Por ejemplo, si es de 1:100, para efectuar una operación con 1 lote el inversionista tiene que arriesgar 1000 unidades de la divisa base. Entonces si negociamos en el par EUR/USD con un lote y con un apalancamiento de 1:100, tenemos que arriesgar 1000 euros.

Los lotes se usan normalmente en cuentas estándar.

- **Mini lote:** Un mini lote por otra parte, está formado por 10000 unidades de la divisa base, lo que equivale a 0.1 lotes. En este caso si operamos con un mini lote utilizando un apalancamiento de 1:100, esto significa que arriesgamos 100 unidades de la divisa base de nuestra

cuenta (**margen inicial**) mientras que el bróker nos "presta" el resto para que se pueda abrir la posición.

- **Micro lote:** Un micro lote representa en realidad 0.01 lotes, es decir la centésima parte de un lote estándar. Si operamos con micro lotes y con un apalancamiento de 1:100, esto significa que en cada operación arriesgamos como mínimo 10 unidades de la divisa base. Con cada micro lote se negocian 1000 unidades de la divisa base del par. (¿qué es un lote?, s.f.)
- La relación general en el mercado de divisas para estos dos factores es.

TAMAÑO DEL LOTE (millones de divisa base)	VALOR DEL PIP (USD/PIP)
1	100
0,1 (estándar)	10
0,01	1
0,001	0,1

Tabla 13:Relacion General en el mercado de Divisas

II.1.1.3.1.4 Productos negociados : En la compra y venta de capital no se compran activos a cambio de dinero, como se hace en un mercado de acciones, en este caso, se está comprando simultáneamente una divisa y vendiendo otra, es decir, haciendo un cambio.

En el mercado de valores, los traders compran y venden acciones; en el mercado de futuros, los traders compran y venden contratos; en el mercado de divisas, los traders compran y venden "lotes". Cuando se compra un lote, está especulando sobre el valor de una divisa frente a otra, es decir, el tipo de cambio en sí. Para poder diferenciar las divisas existentes se utiliza el código ISO. Este código suele utilizar tres siglas donde las dos primeras suelen hacer referencia al país y la última al nombre de la divisa. Una excepción es el euro, cuyo código es EUR

Las divisas se cotizan en pares. Cada compra de una divisa implica una venta recíproca de la otra moneda, y viceversa. Esto significa que la compra es igual a la venta, Pero el hecho es que estamos comprando y vendiendo el tipo de cambio, no una única moneda.

El valor de una moneda no cambia en sí mismo, lo que cambia es su valor en relación a otras monedas. Esta es una característica de un sistema de tipo de cambio flotante. El primer miembro de cada par se conoce como la divisa "base", y el segundo miembro se denomina divisa "cotizada" o "contra" divisa. El tipo de cambio muestra cuánto vale la divisa base medida en términos de la contra divisa. Por ejemplo, si el tipo EUR/ USD es igual a 1,358, quiere decir que euro vale 1,358 dólares. (FXSTREET. Tipos de cambio y pares, s.f.)

Divisas más negociadas

USD (Dólar estadounidense)

EUR (Euro)

JPY (Yen Japonés)

GBP (Libra Esterlina)

CHF (Franco Suizo)

CAD (Dólar Canadiense)

AUD (Dólar Australiano)

NZD (Dólar de Nueva Zelanda)

II.1.1.3.1.5 Participantes . Los intercambios de divisas generalmente se hacen en grandes centros financieros del mundo: Londres, Nueva York y Singapur, son unos de los más importantes.

Los principales participantes en el mercado de divisas son los bancos, las multinacionales, las instituciones financieras no bancarias y los bancos centrales. Para el caso de los bancos, éstos

efectúan transacciones entre cuentas de clientes a nivel internacional y realizan operaciones con divisas como negocio, los clientes de los bancos pagan un precio mayor al cual los bancos compran (pues los bancos las compran “al por mayor”). La diferencia en los precios es la compensación que reciben los bancos por desarrollar esta actividad.

Las instituciones financieras no bancarias, como los fondos de pensiones, al manejar grandes sumas de dinero, también negocian con monedas diferentes.

Los bancos centrales también participan activamente en el mercado de divisas. Las reservas internacionales de los países están invertidas, en su mayoría, en divisas. Dependiendo de las economías, el dinero en circulación en éstas y las políticas económicas, los bancos centrales deciden entrar o no en el mercado de divisas para comprar o vender. (Banco de la republica, s.f.)

II.1.1.3.2 Estilos de inversión en el mercado de divisas al contado

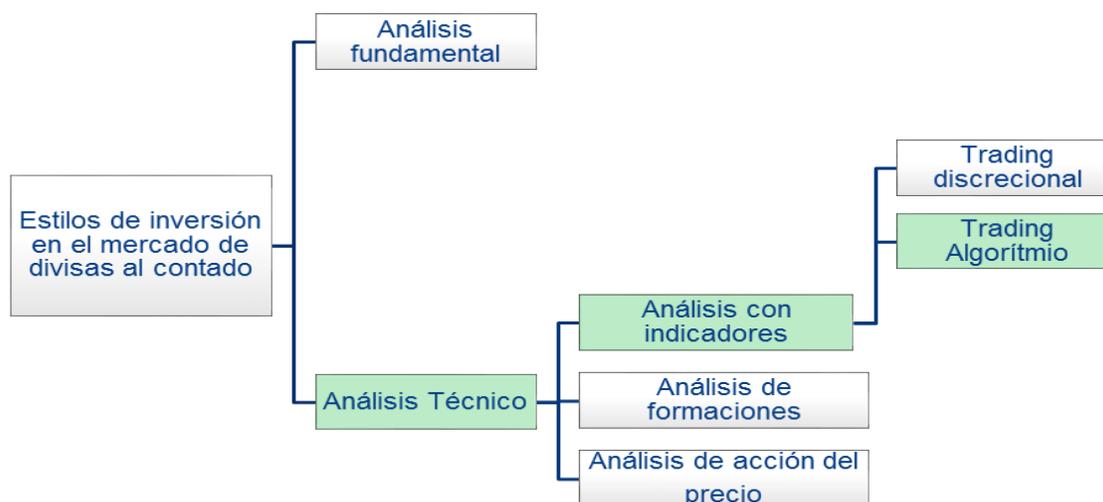


FIGURA 3:Estilo de inversión en el mercado de divisas

II.1.1.3.2.1 Análisis fundamental. Cuando se habla del análisis fundamental en el mercado Forex, el análisis se enfoca en factores macroeconómicos que podrían afectar el valor de una moneda o la fuerza relativa de una moneda soberana versus otra. Estos factores incluyen el estado de la

economía, la política de bancos centrales y las políticas económicas de países. (Brooks, 2013) Existen una gran variedad de indicadores económicos que los analistas observan día a día, pero solo algunos podríamos caracterizarlos como disruptivos. (López, s.f.) Estos son los siguientes:

- Tasas de interés de Bancos Centrales
- Producto Interno Bruto (PIB)
- Tasa de desempleo
- Inflación (IPC)
- Nominas de empleo no agrícola (en EEUU)

II.1.1.3.2.2 Análisis Técnico : El Análisis Técnico consiste en el estudio de movimientos en el mercado, sobre todo mediante la observación del gráfico, con el objetivo de averiguar futuras tendencias en los precios. Cuando nos referimos a “movimientos en el mercado” nos centramos principalmente en el precio y en el volumen de negociación. Sus pilares básicos son:

- ***El precio lo descuenta todo:*** cualquier elemento que afecta al mercado, tanto económicos, políticos, especulativos... Todos ellos los refleja el precio con la lucha constante entre oferta y demanda.
- ***El precio se mueve en tendencias:*** estas pueden ser largas o cortas, y el objetivo del Análisis Técnico es encontrarlas y aprovecharlas, ya que cuando se da una tendencia es más probable que se repita en el tiempo que cambie.
- **El mercado tiene memoria:** la psicología humana afecta a los inversores a la hora de actuar en situaciones similares que han vivido en el pasado, actuando de la misma forma provocando la repetición de pautas en el precio.

a. 1.1.3.2.2.1 Origen: Teoría del Dow Jones

Las bases del análisis técnico tienen su origen en el año 1882, cuando Charles Henry Dow y Edward Jones fundaron la Dow Jones & Company. Dos años más tarde Dow publicó por primera vez en la historia un índice compuesto por nueve empresas de ferrocarriles y dos industriales. Con esto pretendía realizar un diagnóstico de la salud de la economía estadounidense. Con el paso de los años se fueron creando índices para cada industria y añadiendo más empresas a los mismos. En el año 1928 el índice industrial contaba con 30 compañías, al igual que el reconocido índice Dow Jones en la actualidad.

La Teoría de Dow se compone de seis principios básicos:

1. Las medias lo descuentan todo: al igual que el precio lo descuenta todo para un activo, media de un sector nos indica toda la información del mismo.

2. Un mercado puede tener tres tendencias: con las que realizaba un símil con el mar, la primaria que era la marea, la secundaria que decía que eran las olas y la menor que eran las ondas. En la Figura 1 lo podemos ver de una forma más clara.



FIGURA 4:Tendencia teoría del Dow Jones realizado sobre el S&P500 con datos diarios.
Fuente: elaboración propia en XTB.

3. Las tendencias primarias tienen tres partes: como así se representa en la Figura 2, acumulación (compra por los agentes más audaces), participación pública (entran al mercado los “sigue tendencias”) y distribución (llamados por las altas rentabilidades empiezan a invertir el público en general).

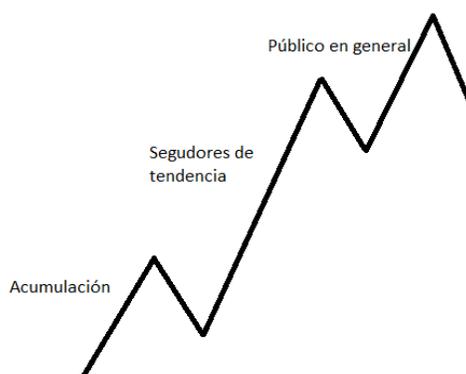


FIGURA 5:Partes tendencias primarias. Fuente: elaboración propia

4. Confirmación de medias: para que una señal de una media tuviera sentido debía ser confirmada por la media de otro sector industrial.

5. El volumen tiene que confirmar la tendencia: está en un segundo plano, pero también es importante. En una tendencia ascendente deberá aumentar a medida que los precios suben y descender a medida que los precios bajan. En una descendente debe aumentar a medida que los precios bajan y crecer a medida que los precios suben.

6. Se presupone que la tendencia sigue activa hasta que el cambio de tendencia es notorio: hay diferentes teorías al respecto, algunos defienden como en la parte izquierda de la Figura 3 que se produce un cambio de tendencia alcista hacia una tendencia bajista cuando se

supera a la baja el primer mínimo, y otros como podemos ver en la parte derecha de la Figura 5 cuando se ha superado el segundo mínimo decreciente. Para una tendencia bajista sería justamente, al contrario.

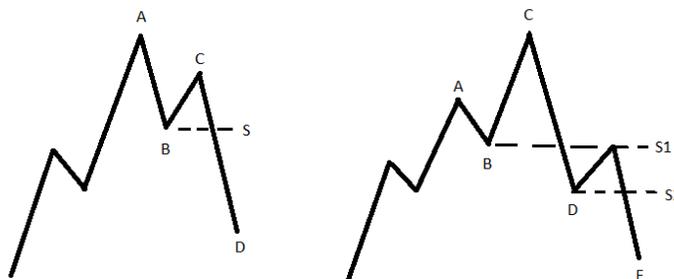


FIGURA 6: Cambios de tendencia. Fuente: elaboración propia.

b. 1.1.3.2.2 Tipos de gráficos

En Análisis Técnico existen diferentes tipos de gráficos, cada uno de ellos nos muestra una información. Entre estos los más utilizados son:

- **Gráfico de líneas:** el más sencillo de todos, nos muestra el precio de cierre del activo en cada momento del tiempo.
- **Gráfico de barra:** este gráfico algo más complejo que el anterior, como podemos ver en la Figura 4, nos informa del precio de apertura, máximo, mínimo y cierre de una cotización. En la Figura 4 a la izquierda podemos ver una barra alcista y a la derecha una barra bajista.

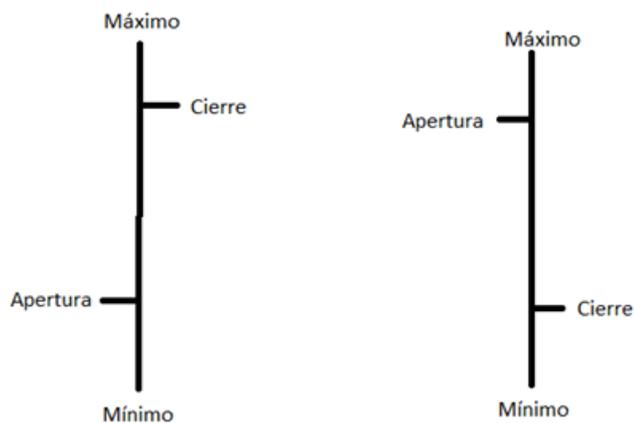


FIGURA 7: Gráfico de barras. Fuente: elaboración propia.

- Gráfico de velas japonesas:** denominado así por su origen, se remontan hasta el siglo XVII en Japón en el mercado de futuros del arroz. Este gráfico es el más utilizado, aunque como podemos observar en la Figura 5 nos proporciona la misma información que el gráfico de barras. Representado en verde tenemos una vela alcista (el precio de cierre supera el precio de apertura), por su parte, la vela bajista representada en rojo (el precio de cierre se encuentra por debajo del precio de apertura).

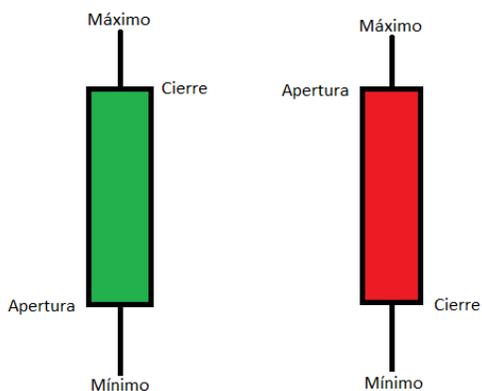


FIGURA 8: Gráfico de velas japonesas. Fuente: elaboración propia.

En estos dos últimos tipos de gráficos es donde entra en juego la temporalidad de los datos, ya que cada barra o vela nos muestra dicha información. Podemos llegar a encontrar gráficos con temporalidades desde 1 minuto hasta de un mes.

c. 1.1.3.2.2.3 Análisis Chartista

El Análisis Chartista consiste en identificar patrones en las velas que se repitan y que nos ayuden a construir una estrategia de mercado eficaz. Vamos a explicar las bases del análisis Chartista mediante varios ejemplos básicos de (Murphy, 1999)

Tendencia: es el elemento más básico, pero a su vez esencial para entender este tipo de análisis. A grandes rasgos se considera tendencia a la dirección en la que se mueve el precio de un activo. Este movimiento, en la gran mayoría de los casos no es lineal, y suelen caracterizarse por realizar zigzags, lo que genera máximos y mínimos. Como podemos observar en la Figura 6, para detectar una tendencia alcista (verde) es necesario unir una serie de mínimos crecientes y para detectar una tendencia bajista (rojo), al contrario, una serie de máximos decrecientes. Pero las tendencias no siempre son alcistas o bajistas, también podemos encontrar tendencias laterales (azul).

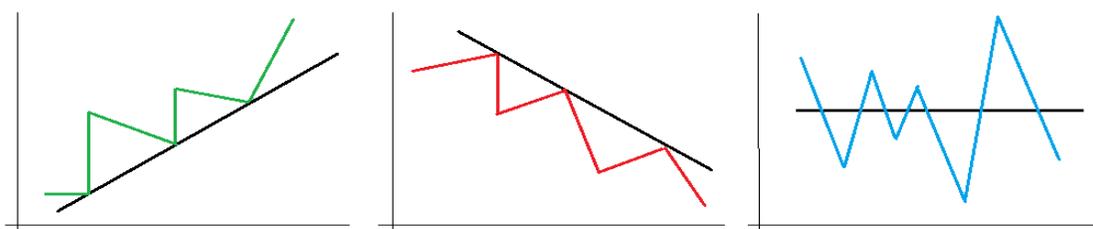


FIGURA 9: Tipos de tendencias. Fuente elaboración propia.

Soportes y resistencias: son denominados soportes a los mínimos de los que hablábamos previamente, y resistencias a los máximos. Cuando nos encontramos en una tendencia ascendente las resistencias dificultan dicha subida, a su vez con los soportes ocurre

igual en las tendencias bajistas. Con este mecanismo se fundamenta en uno de los principios previamente mencionado en los principios del análisis técnico, el que el mercado tiene memoria. Ya que los inversores recuerdan anteriores máximos y mínimos lo que produce que estos precios sean conflictivos en el futuro. Por ello como podemos observar en la Figura 7 los soportes (verde) pueden convertirse en resistencias (rojo) y viceversa.

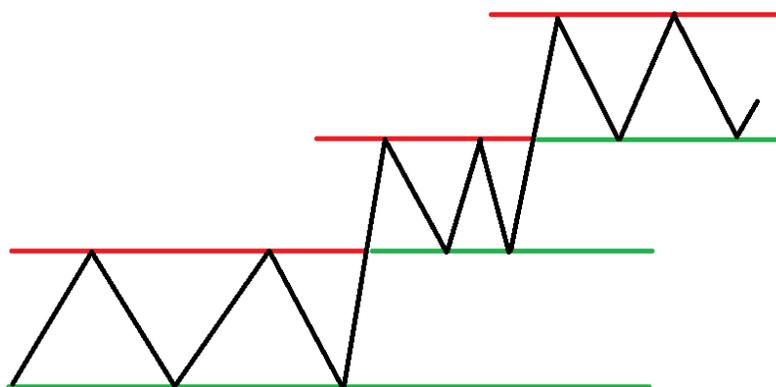


FIGURA 10: Soportes y resistencias. Fuente: elaboración propia.

Figuras de cambio de tendencia: como doble techo, doble suelo, suelos, techos redondeados, hombro cabeza hombro, etc. Esta última es la más conocida, la forma de esta Figura es la que su propio nombre indica. Visto en la Figura 8 encontramos con la “línea del cuello” la cual hace de soporte a los mínimos de la estructura, además tenemos un primer “hombro” seguido de la “cabeza” (máximo superior al del hombro) y después otro hombro inferior a la cabeza. Una vez rota la línea del cuello debemos buscar una caída como mínimo de la altura de la cabeza.



FIGURA 11:Hombro cabeza hombro sobre EURUSD en velas de 15 min.

Figuras de continuación de la tendencia: como triángulos cuñas, bandera, ensanchamiento, diamante... De este grupo de Figuras los triángulos son las más conocidas. Como podemos observar en la Figura 9, se trata de un conjunto de máximos decrecientes (triángulo alcista) o mínimos crecientes (triángulo bajista) que producen una acumulación que termina rompiendo hacia arriba (alcista) o hacia abajo (bajista), y la proyección mínima esperada es la altura máxima del triángulo.

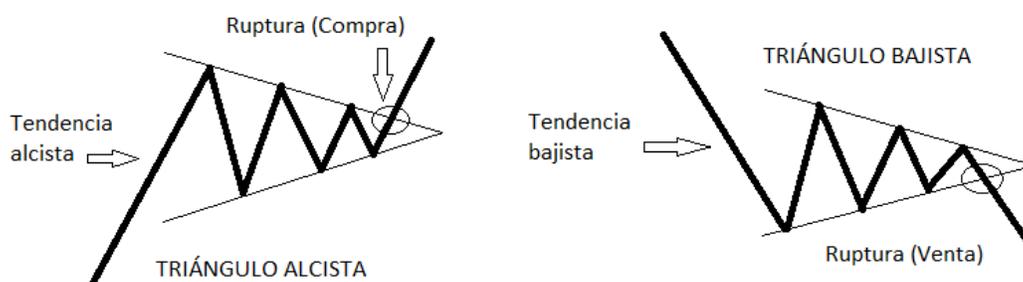


FIGURA 12:Triángulo alcista y triángulo bajista. Fuente: elaboración propia.

La gran crítica que se le puede achacar al Análisis Técnico Chartista es su subjetividad, ya que cada uno de los inversores especializados en esta disciplina puede tener

un diagnóstico distinto a la hora de interpretar el mismo gráfico, ya que no sigue unas reglas estrictas y puede dar lugar a señales de inversión erróneas. Con el avance de la inteligencia artificial se han podido desarrollar algoritmos capaces de detectar algunos de estos patrones cuantificables.

d. 1.1.3.2.4 Análisis Técnico Estadístico

Se trata de un tipo de análisis que utiliza diferentes indicadores u osciladores, los cuales eliminan en parte la subjetividad del análisis de la información que nos proporciona el precio y el volumen. A diferencia del análisis gráfico el cual podía llevar a dos inversores que observan el mismo gráfico a tomar decisiones opuestas, estos indicadores arrojan los mismos datos a todos sus usuarios y son fácilmente automatizables. Existen infinidad de indicadores técnicos, desde los más simples a los más complejos. Cada uno de ellos tiene unos pros y unos contras, que desarrollaremos más a fondo a continuación.

1. Medias móviles

Se trata como su propio nombre indica de un promedio de un determinado conjunto de datos. Lo que nos ayuda a suavizar los datos para una sencilla interpretación. Puedes elegir para ello tanto el precio de apertura, cierre, máximo o mínimo, además es aplicable para cualquier periodo muestral.

Existen diferentes tipos de medias móviles en función de cómo se calcula la misma. Las más utilizadas son:

- **Media móvil simple o aritmética (SMA):** es la más utilizada. Como podemos ver en la ecuación 1 cada valor tiene el mismo peso, esta es su mayor crítica. En la Figura 7 aparece representada en verde.

$$SMA_i(n) = \frac{\sum_{t=i-n}^i P_t^{Cl}}{n}$$

donde

- P_t^{Cl} = precio del activo en el instante i .
- n = número de periodos considerados

• **Media ponderada linealmente (LWMA):** soluciona parte de la crítica a la MA, como podemos ver en la ecuación 2, en la ecuación 3 vemos que da menor importancia a los valores más alejados del periodo actual. En la Figura 10 está representada en naranja

$$LWMA_i(n) = \frac{\sum_{t=i-n}^i P_t^{Cl}(n) \cdot w_t}{n}$$

$$w_t = \frac{n-i}{n}$$

• n

donde:

- $P_t^{Cl}(n)$ = precio del activo en el instante i .
- n = número de periodos considerados.
- $w_p(i)$ = ponderación lineal en el periodo i .

• **Media móvil exponencial (EMA):** sigue el mismo patrón que la media linealmente ponderada, pero más suavizada como podemos ver en la ecuación. En la Figura 10 se representa en color azul.

$$EMA_i(n) = [P_i^{Cl} - EMA_{i-1}(n)] * w_i^e + EMA_{i-1}(n)$$

$$w_i^e = \frac{2}{n+1}$$

donde:

- P_i^{Cl} = precio del activo en el instante i.
- n = número de periodos considerados.
- w_i^e = ponderación exponencial en el periodo i.
- $EMA_{i-1}(n)$ = media móvil exponencial de n periodos en el periodo anterior al instante i.

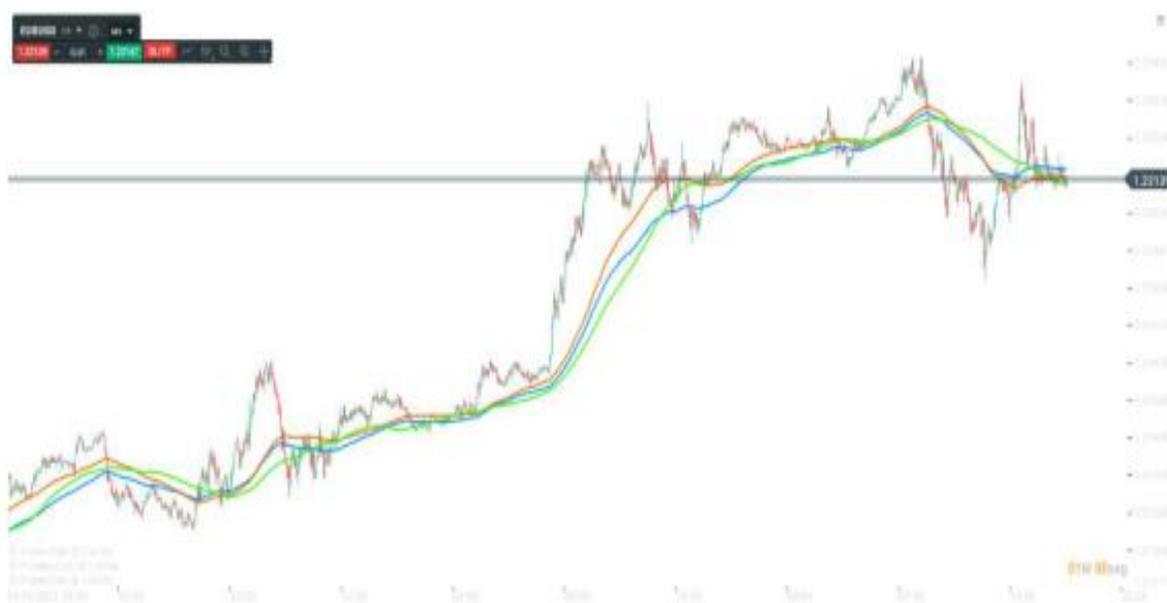


FIGURA 13:Media Movil Simple, Media Movil Ponderada y Media Movil Exponencial

Fuente: elaboración Propia con Plataforma XTB

Podemos observar cómo pese a tener en cuenta los últimos cien periodos estas medias no se comportan igual. La LWMA sigue más de cerca al precio, por lo comentados anteriormente, después la EMA y por último la MA.

Todas las medias móviles se pueden usar de diferentes formas y aplicar en cualquier activo financiero y en cualquier gráfico temporal, Murphy (1999) las más comunes son:

- ***Cruce de medias con el precio:*** cómo podemos ver en la Figura 11, cuando precio cruza por encima de una media móvil se genera una señal de compra (círculo verde) y cuando el precio cruza por debajo de la media se genera una señal de venta (círculo rojo).



FIGURA 14: Señales con una SMA de 200 periodos en EURUSD con datos diarios.

- ***Cruce de medias:*** añadiendo medias móviles de diferentes periodos como en la Figura 12 podríamos considerar una señal de compra (círculo verde), cuando la media más rápida

(azul claro) cruza por encima de la media más lenta (azul oscuro), y de venta, al contrario, cuando la media más rápida cruce por debajo de la más lenta (círculo rojo).



FIGURA 15: Señales con cruce de medias EMA (50) y EMA (100) sobre el DAX alemán durante la crisis de coronavirus con velas de 4 horas. Fuente: elaboración propia con la plataforma XTB.

- **Soportes o resistencias:** observamos como la media es utilizada como soporte por el precio hasta en cinco ocasiones en una tendencia alcista.



FIGURA 16:EMA (100) utilizada como soporte en el DAX alemán con datos de 4 horas.

Las estrategias de las medias pueden ser muy eficaces, sobre todo cuando los movimientos del mercado son tendenciales, ya que cuando el mercado se sitúa en rango nos puede dar lugar a muchas señales falsas que acabarían dinamitando nuestra rentabilidad.

2. Bandas de Bollinger

Introducidas por el analista financiero John Bollinger en los años ochenta, las Bandas de Bollinger como podemos ver en la ecuación 6 constan de una media móvil de n periodos y en la ecuación 7 vemos la banda superior con D desviación típica positiva y y en la ecuación 8 otra inferior con D desviación típica negativa.

$$BC_i(n) = SMA_i(n)$$

$$BS_i(n) = SMA_i(n) + (D * StdDev)$$

$$BI_i(n) = SMA_i(n) - (D * StdDev)$$

donde:

- $BC_i(n)$ = banda central.
- $BS_i(n)$ = banda superior.
- $BI_i(n)$ = banda inferior.
- $StdDev$ = desviación típica.
- El n más usado es 20.
- La D más usada es 2.

Al añadirle las dos desviaciones típicas logramos capturar dentro de las bandas en 95% de los datos que nos proporciona el precio (si le añadiéramos tres el 99%). En la Figura 14 podemos observar las Bandas de Bollinger, vemos como el precio fluctúa prácticamente todo el tiempo dentro de las BB, con la excepción de pequeños impulsos que rápidamente se corrigen y el precio vuelve a entrar dentro de las bandas.

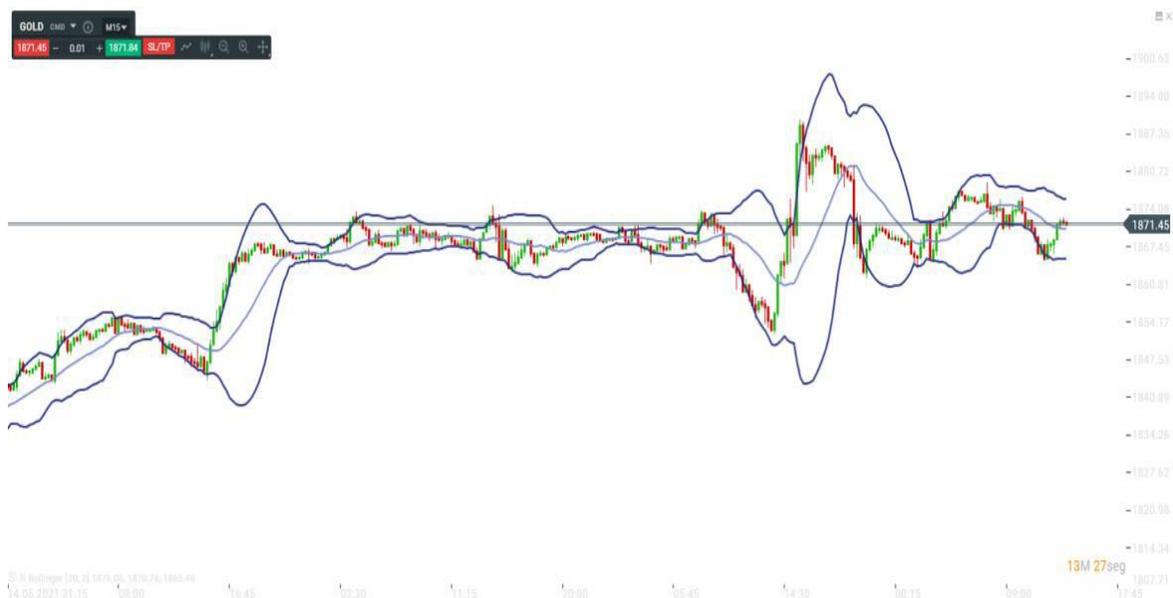


FIGURA 17: Aplicación de la Bandas de Bollinger en el Oro con velas de 15 minutos.

Además, el ancho de las bandas nos ofrece una información extra muy valiosa, ya que miden la volatilidad del activo, cuando las bandas están muy separadas la volatilidad es alta, por el contrario, cuando la volatilidad es baja las bandas se sitúan cerca de la media.

Las Bandas de Bollinger se pueden usar de diferentes formas. Usando como señal de compra (círculo verde) un cruce del precio por encima de la banda central y como objetivo la banda superior (rectángulo verde), y para una venta funcionaría, al contrario, cuando el precio cruce por debajo de la banda central se produce la señal de venta (círculo rojo) con un objetivo en la banda inferior (rectángulo rojo). (Elder, 2017)



FIGURA 18: Estrategia con Bandas de Bollinger en el Oro con velas de 15 minutos.

El problema de esta estrategia es que corta demasiado rápido las ganancias, mientras que las pérdidas no, además puede dar lugar a demasiadas señales falsas.

3. Osciladores

A diferencia de las medias móviles las cuales son muy eficaces a la hora de detectar tendencias los osciladores funcionan muy bien cuando el precio se mueva en un rango horizontal (Murphy, 1999). Otra diferencia entre osciladores y las medias es que los osciladores se representan fuera del gráfico del precio, normalmente en la parte inferior del gráfico. Cada

oscilador posee una escala propia, aunque la más común es de 0 a 100. Los usos más importantes del oscilador según Murphy (1999) son los siguientes:

- **Sobrecompras y sobreventas:** los osciladores nos dan mucha información cuando tocan valores extremos, cuando es un valor extremo muy alto lo consideramos que el activo esta sobrecomprado, y cuando está demasiado bajo consideramos que esta sobrevendido. Nos advierte sobre un posible cambio de tendencia.

- **Divergencias:** consideramos que se está dando una divergencia cuando el precio de un activo y el oscilador no van en concordancia, normalmente se salda con un cambio de tendencia.

- **Cruce línea central:** en algunos indicadores puede que sea un cero, en otros un 50, pero la interpretación es la misma, nos puede dar una señal sobre la dirección de la tendencia.

Osciladores más utilizados:

A. Momentos

El momento es el oscilador más sencillo, ya que como podemos ver en la ecuación 9, consta simplemente de una resta entre el ultimo precio de cierre y el ultimo precio de cierre de hace n días.

$$MOM_n(i) = P_{cl}(i) - P_{cl}(i - n)$$

donde:

- $P_{cl}(i)$ = precio de cierre en el instante i
- n = periodos utilizados.

Como podemos observar en la Figura 16 gracias a la estrategia de momentos podemos apreciar una divergencia entre el precio y el indicador. El precio está creando máximos crecientes mientras que el indicador está creando máximos decrecientes, dicha divergencia se salda con una caída considerable del precio.



FIGURA 19: Estrategia con Momentos (14) en el gráfico del Russell 2000 con velas de 1 hora. Fuente: elaboración propia con la plataforma XTB.

B. Índice de Fuerza Relativa.

Más conocido como RSI (Relative Strength Index), fue desarrollado por Wilder Jr. el cual presentó en 1978 en su libro “New Concepts in Tecchnical Trading System”. Creado con el objetivo de suavizar los cambios bruscos que nos proporciona el indicador de momentos, es sin duda el oscilador más conocido y más utilizado. Además, como podemos ver en la ecuación 10, los resultados obtenidos aparecen acotados entre 0 y 100, lo que nos ayuda a eliminar la parte subjetiva y circunstancial del oscilador de momentos.

$$RSI_t(n) = 100 - \frac{100}{1+RS}$$

$$RS = \frac{MAA_n}{MAB_n}$$

donde:

- RS = fuerza relativa
- MAA_n = media de cierres alcistas en los últimos n periodos.
- MAB_n = media de cierres bajistas en los últimos n periodos.

En la Figura 17 podemos ver un ejemplo de las tres opciones para interpretar un oscilador. Por un lado, representado en amarillo podemos ver como se forma una tendencia alcista la cual detectamos ya que RSI se sitúa por encima del valor 50, durante gran parte de la misma. Después en el canal azul vemos como las señales de sobre compra ($RSI > 70$) y sobreventa ($RSI < 30$) funcionan correctamente como señales de compra y venta. Y, por último, en azul marino podemos detectar un cambio de la tendencia intermedia debido a una divergencia entre indicador y precio.



C. Estocástico.

El proceso estocástico fue desarrollado a finales de la década de los cincuenta por George Lane y un grupo de inversores en Chicago. George Lane es un conocido inversor del siglo XX, especializado en el Análisis Técnico, fue el presidente de Investment Educators Inc., donde enseñaba a inversores a entender los mercados financieros. Era recurrente escuchar a George Lane esta frase para explicar el funcionamiento de su indicador: “El estocástico mide el impulso del precio. Si visualizamos un cohete que sube en el aire, antes de que pueda bajar, debe reducir la velocidad. El impulso siempre cambia de dirección antes de precio.” Este oscilador compuesto por %K y por %D. En la ecuación 12 vemos %K que mide porcentualmente donde se encuentra el precio de cierre en relación con la franja de precios de un periodo n. En la Figura 13 vemos %D que simplemente es una media móvil de m periodos. (Murphy, 1999).

$$\%Ki(n) = \frac{Pi\ Cl - PL(n)}{PH(n) - PL(n)}$$

$\%Di(m) = SMA_m(\%Ki)$ (13) donde:

- $Pi\ Cl$ = Precio de cierre en el instante anterior.
- $PL(n)$ = Precio mínimo en los últimos n periodos.
- $PH(n)$ = Precio máximo en los últimos n periodos.
- SMA_m = Media móvil simple de m periodos.
- $n > m$ = ya que consideramos una media móvil rápido.

En la Figura 18 podemos observar el oscilador estocástico con un n de 14 periodos y un m de 3 periodos. Vemos las señales de compra (circunferencia verde), cuando se produce un cruce de %D (naranja) por encima de %K (rojo) al cruzar la zona del 20%. Y al revés sucede para las

señales de venta (circunferencia roja), cruce de %D por debajo de %K al cruzar la línea del 80%. (Sainz Bravo, 2021)

Análisis Técnico Aplicado con un Sistema de Trading Algorítmico



FIGURA 20: Estrategia Oscilador Estocástico en el Footsie 100 con vela de 1 día

Fuente: Elaboración Propia

II.1.1.3.2.2.1 Trading algorítmico

1. El trading frente a las inversiones tradicionales.

Realizar operaciones de trading e invertir son dos formas distintas de exponerse a los mercados financieros. Cuando se invierte, se hace una operación de compra de un activo por un largo periodo de tiempo asumiendo posibles reducciones del precio del activo, pero con el objetivo de obtener beneficios en un futuro si los mercados continúan comportándose como lo han hecho históricamente.

Frente a las inversiones tradicionales se encuentra el *trading*, cuyo objetivo es obtener rentabilidades beneficiándose de los movimientos de los precios a corto o medio plazo de los activos financieros en los que se opera, abriendo y cerrando posiciones de compra y venta frecuentemente.

A diferencia del tipo de inversión tradicional, el *trader* se preocupa por las pequeñas tendencias y correcciones del mercado del corto plazo y además las posiciones de compra (también llamadas largas) o de venta (también llamadas cortas) son igual de comunes mientras que las inversiones tradicionales a largo plazo se suelen centrar en operaciones de compra únicamente

En cambio, cuando se especula sobre movimientos de los precios de activos financieros a corto y medio plazo, abriendo y cerrando posiciones con más frecuencia, se estarán realizando operaciones de *trading*.

2. La automatización en el *trading*

Según cómo se ejecuten las operaciones de compra y venta, el *trading* se puede clasificar en manual o automático.

Si es un humano el único responsable de ejecutar estas operaciones, se hablará de **trading manual**. Al ser una persona la encargada de realizar estas operaciones, puede hacerlo de dos formas:

- empleando su propio criterio e intuición para tomar las decisiones de compra y venta, pudiéndose ayudar de herramientas de análisis, pero no siguiendo unas reglas sistemáticamente, en este caso se habla de **trading discrecional**;
- siguiendo las indicaciones de un sistema previamente diseñado en el que quedan definidos todos los puntos de entrada y salida al mercado, limitando al máximo la subjetividad

del trader, en este caso se trata de **trading sistemático o algorítmico** (del que se hablará más adelante) con ejecución manual. (Guerola Perez, 2021)

En cambio, si son los programas informáticos los que realizan estas operaciones sin intervención humana, **se tratará de trading automático**.

Como se ha mencionado, el **trading algorítmico o sistemático**, se basa en programas informáticos que contienen un conjunto de reglas que indican el instante de entrada y salida mercado, eliminando la intervención humana a la hora de tomar cualquier decisión. Estas operaciones de compra y venta pueden ser ejecutadas de forma manual o automática.

3. Definición de Trading Algorítmico

Básicamente el trading algorítmico se puede definir como la aplicación de cálculos matemáticos y trading automatizado para negociar de manera eficiente diferentes activos, mediante cantidades determinadas. Este enfoque es también conocido como de modelos cuantitativos de negociación.

Para una cantidad definida, los modelos de trading algorítmico determinan el precio, monto, tiempo y demás parámetros definidos usando cálculos matemáticos. Un modelo se puede considerar como efectivo cuando permite minimizar la ejecución por parte del operador y los resultados de las operaciones son como mínimo los obtenidos mediante el análisis y ejecución manual por parte del mismo.

Los algoritmos eliminan los aspectos emocionales en el proceso de negociación, dando como resultado un rendimiento más consistente

Estos sistemas operan constantemente en el mercado, pero se deben distinguir dos modalidades distintas:

Los que utilizan **sistemas de baja frecuencia** para operar, se incluyen las personas que realizan trading a nivel particular.

Los **sistemas de alta frecuencia** o “*High Frequency Trading*” (HFT), que utilizan tecnologías y algoritmos muy avanzados para poder abrir y cerrar numerosas órdenes al mercado en cortos periodos de tiempo (operaciones que duran fracciones de segundo). Durante los últimos años este tipo de sistemas de alta frecuencia han supuesto un gran impacto en los mercados. (Guerola Perez, 2021)

II.1.1.3.3 Metodo para una estrategia

El diseño de una estrategia de *trading* debe ser un proceso ordenado y seguir una metodología, en este apartado se explicarán los pasos a seguir para construir una estrategia partiendo de una base de conocimientos muy pequeña, no siendo necesario seguir todos ellos, pero sí recomendable.

A continuación, se explicarán en detalle cada una de las fases. No es mandatorio realizar todas para dar por validada una estrategia, pero si recomendable ya que dará mayor confianza.

II.1.1.3.3.1 Diseño

1. Infraestructura

i. *Hardware*

Es importante disponer de un PC de una cierta potencia y una conexión a internet de alta velocidad. Posteriormente se necesitará un servidor virtual privado (VPS) para tener la estrategia corriendo las 24 horas y sin interferencia de otras aplicaciones. Algunas plataformas ya incluyen la opción de alojar las estrategias en su nube, en cuyo caso no sería necesario VPS.

ii. *Software*

a) Plataforma de trading

Se debe elegir una plataforma de trading, en esta plataforma se realizará el diseño y evaluación de la estrategia. Esta plataforma debe permitir el *trading* algorítmico. Como ejemplos de plataforma de trading de este tipo, se tienen: NinjaTrader, Tradestation, *Visualchart*, *ZorroTrader*, *TraderWorkStation*, *Darwinex*...

b) Bróker

El bróker tiene la función de enviar las órdenes a mercado, según sea comandado por la estrategia. Es decir, es la interfaz entre el mercado y el sistema. Algunos ejemplos de bróker son: *Interactive Brokers*, *Tradestation* o *Visualchart*, entre otras.

La elección del bróker es importante ya que este cobra comisiones (comisión inicial, por mantenimiento, por la apertura y cierre de un trade...) por lo que se deberá elegir uno con las menores comisiones posibles atendiendo a los activos que se van a operar.

c) Datos históricos

Los datos históricos o “*data feed*” son series de precios de las cotizaciones de un activo en el pasado. Los datos pueden tener diferentes calidades según su precisión y los ajustes que sobre ellos se realizan. Disponer de una completa base de datos de cotizaciones es esencial para evaluar el rendimiento de los sistemas.

Como ejemplo de plataformas que proveen de estos datos se encuentran *Kinetick*, *Tradestation*, *Interactive Brokers*... A menudo una plataforma puede actuar como bróker y proveedor de datos, aunque en teoría se podría trabajar con cualquier combinación de proveedor y bróker. Como se verá en el apartado práctico al operar en modo simulación con datos del mercado en tiempo real, usar una plataforma y un bróker proveedor de datos diferentes genera muchos problemas de configuración y sincronización.

2. Lógica inicial

i. Condicionantes/ Preferencias Personales

En primer lugar, se debe identificar los condicionantes o preferencias del trader ya que estos serán determinantes a la hora de diseñar la estrategia. **Los condicionantes a tener en cuenta son: el estilo de trading, el capital inicial que se empleará, el riesgo y objetivos de la estrategia y el grupo de activos sobre los que se quiere operar.** Una vez se tiene esto claro se establece una idea inicial que concuerde.

a) Estilos de *trading*

Las estrategias se pueden dividir en los siguientes grupos atendiendo al estilo del que la crea. La principal diferencia entre estos estilos es el periodo de tiempo que se mantienen abiertas las posiciones en el mercado.

Estilo de Trading	Periodo de tiempo	Tipo de operaciones	Observaciones
“Long Trading” o Trading de posiciones	Largo Plazo	<ul style="list-style-type: none"> - La duración de las operaciones en el mercado puede ser desde meses hasta años - Se intentan captar las grandes tendencias a largo plazo e ignorar pequeñas fluctuaciones. 	Es el estilo que más se parece a la inversión tradicional, siendo la principal diferencia que la inversión implica operaciones de compra únicamente mientras que en el “ <i>long trading</i> ” se intentan aprovechar tendencias tanto de compra como de venta
“Swing Trading”	Medio plazo	<ul style="list-style-type: none"> - Operaciones abiertas durante más de un día e incluso semanas. - El objetivo es capturar las tendencias del precio a corto plazo. 	Este estilo de trading no requiere una constante supervisión, algo que puede resultar atractivo para determinadas personalidades.
Trading Intradía	Corto plazo	<ul style="list-style-type: none"> - Las posiciones se abren y cierran en el mismo día, no dejando operaciones abiertas a fin de sesión. 	Este tipo de estrategias no se exponen al riesgo del mercado por la noche y además evitan las comisiones “overnight” por mantener posiciones durante la noche.

<p>“Scalping” o HFT</p>	<p>Muy corto plazo</p>	<ul style="list-style-type: none"> - Se pueden colocar decenas o centenas de operaciones al día. - Realizan muchas operaciones con ganancias muy pequeñas 	<p>Debido a la alta frecuencia operativa es necesario que las comisiones sean lo más bajas posible. Además, implica una alta dedicación por parte de inversor que debe supervisar y monitorear el sistema.</p>
--------------------------------	------------------------	--	--

Tabla 14:Estilos de Trading

b) Capital inicial

El capital inicial constituye el tamaño de la cuenta, lo cual afecta a la capacidad para diversificar en mercados y estrategias, al tipo de productos que se pueden operar y al tamaño de la posición.

c) Riesgo asumible/ Objetivos

El creador de una estrategia debe conocer cuál es su objetivo, este debe ser cuantificable y realista. Por ejemplo, un objetivo puede ser tener una pérdida máxima del 12%, y en base a eso decidir el tipo de sistema; otro objetivo puede ser obtener una rentabilidad anual determinada.

Al diseñar una estrategia es muy importante tener en cuenta el perfil de aversión al riesgo del inversor ya que el riesgo y el retorno que se puede obtener están altamente relacionados. A continuación, un ejemplo de cómo podría cuantificarse cada perfil inversor.

- **Perfil conservador** arriesgará en torno a un 2% en cada operación.
- **Perfil agresivo** arriesgará en torno a un 6% en cada operación.
- **Perfil temerario** arriesgará en torno a un 12% en cada operación.

d) Pares de interés

Al diseñar una estrategia para futuros se debe fijar el pares de futuros en los que se pretende que funcione ya que una estrategia que pueda resultar adecuada para ellos.

ii. Tipos de estrategias

Una vez se establecen los condicionantes se selecciona el tipo de estrategia que se va a diseñar. Existe diferentes tipos de estrategias, como ejemplos se darán las más comunes.

Interbank Price Delivery Algorithm(IPDA), Esta estrategia se basa en operar en base a las “huellas” que el algoritmo interbancario de “SMART MONEY” nos va dejando y a fuertes movimientos en el precio que son producidos por grandes cantidades de capital que entran al mercado.

¿Qué es Liquidez?

La liquidez en el mercado se encuentra encima/debajo de cada high/low que hay en el mercado, así como también por encima de niveles de soportes o resistencia.

Podemos interpretar la liquidez como las zonas en donde existen ordenes pendientes como Buy Stops (Buy Side Liquidity) o Sell Stops (Sell Side Liquidity).

Piensa en la Liquidez como el combustible que el precio necesita para moverse. Una vez que el mercado captura o barre la liquidez de algún High, es casi seguro que hará una reversión e irá hacia la siguiente zona con mayor cantidad de ordenes pendientes como un Daily Low.

Tipos de Liquidez

El mercado está siendo manipulado constantemente por lo que debes aprender interpretar los distintos tipos de liquidez y evitar operar en esas zonas y solo operar únicamente cuando la captura de liquidez haya sucedido para evitar caer en la “manipulación”.

EQH's: Equal Highs - *Altos iguales en donde por encima hay liquidez*

EQL's: Equal Lows - *Bajos iguales en donde debajo hay liquidez*

BSL: Buy Side Liquidity – *Liquidez de compradores*

SSL: Sell Side Liquidity – *Liquidez de vendedores*

IRL: Internal Range Liquidity – *Liquidez de Estructura Interna*

ERL: External Range Liquidity – *Liquidez de Estructura Externa* LG: Liquidity Grab –

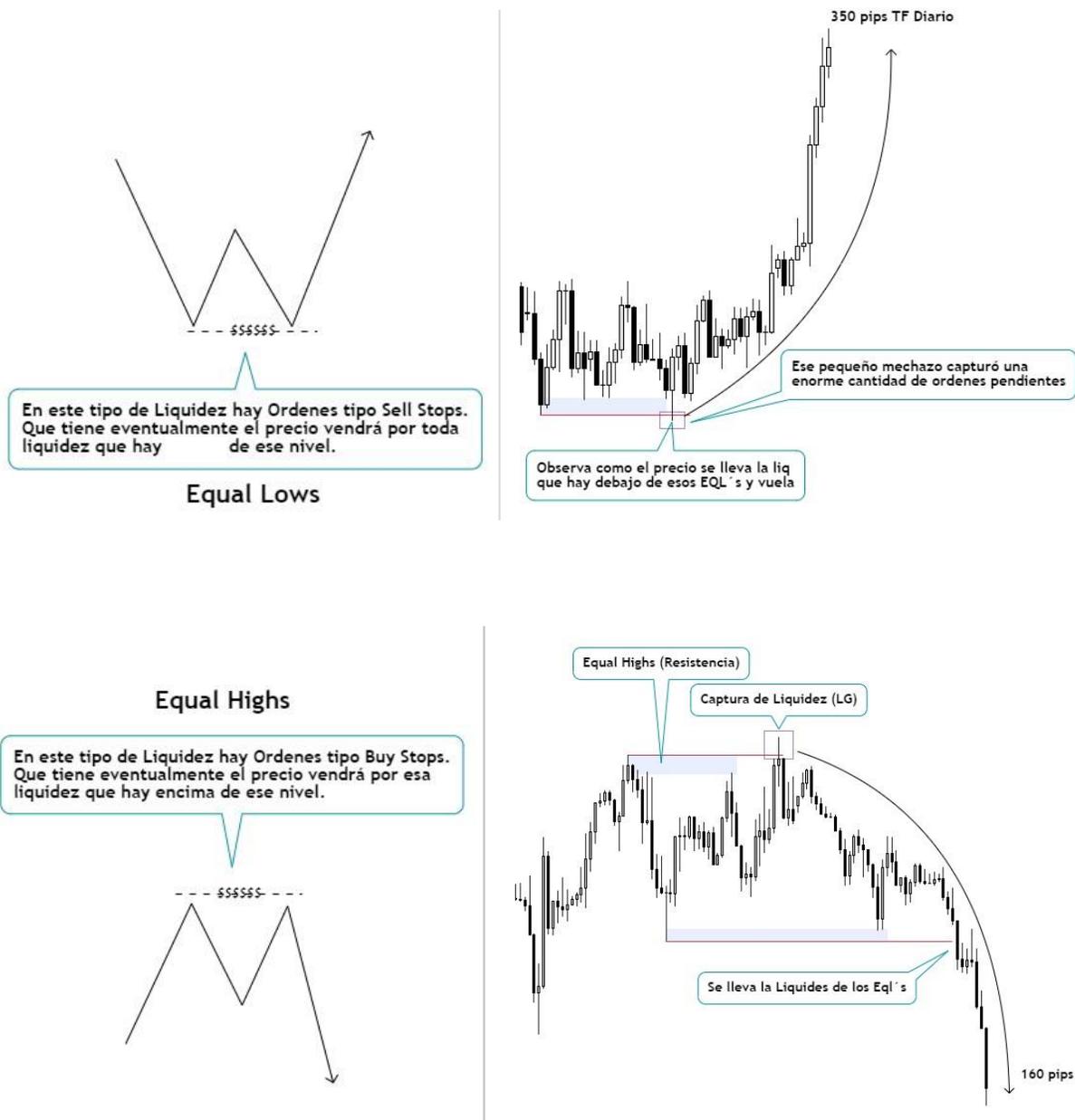


FIGURA 21: Captura/barrido de Liquidez

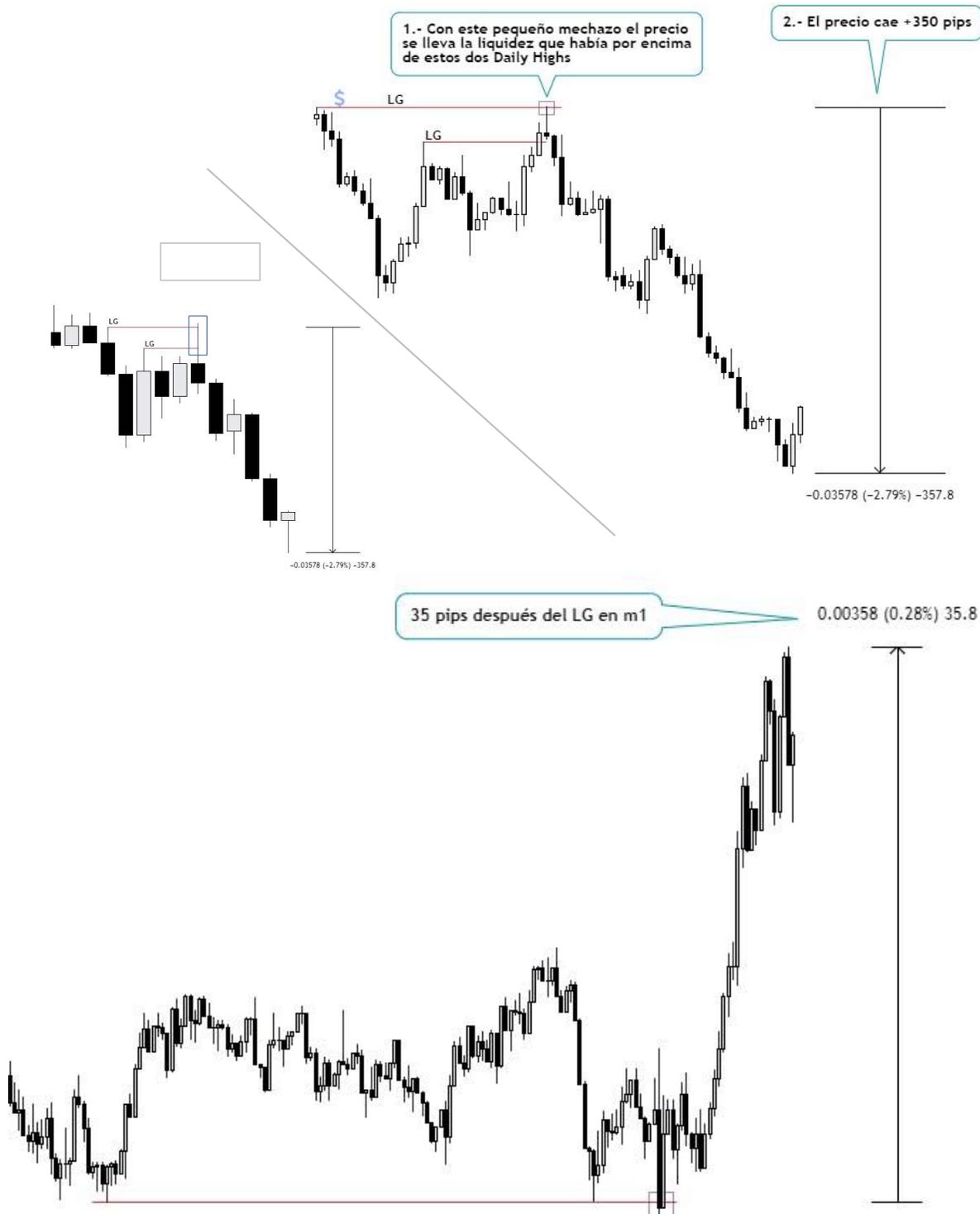


FIGURA 22: Captura/barrido de Liquidez

Tendenciales. Se basan en seguir la dirección de las tendencias del mercado tanto alcistas como bajistas. Algunos ejemplos de sistemas de este tipo son los **sistemas de medias móviles o los de ruptura de canales.**

Antitendenciales o “mean reversion”. Estos sistemas se basan en la idea de que los precios tienen a oscilar por encima y por debajo de un cierto nivel durante un determinado periodo, de tal forma que si el precio se encuentra muy por encima de este nivel se presupone que disminuirá por lo que se toma una posición que se beneficie de ello.

Volatility Break Out (VBO). Son sistemas caracterizados por operar cuando la volatilidad es alta, es decir se diseñan para beneficiarse de los cambios bruscos del mercado. Destaca el uso de indicadores para medir la volatilidad.

Open range breakout (ORB). Estos sistemas se benefician de una alta direccionalidad en la apertura del mercado que está condicionada por las fluctuaciones “*after hours*”.

iii. Seleccionar el time frame de la estrategia

El time frame o temporalidad del gráfico hace referencia al tiempo necesario para la creación de cada una de las velas o barras. El time frame de la estrategia debe concordar con el estilo y el tipo de la estrategia que se quiere diseñar. A priori sólo será necesario seleccionar un valor aproximado para poder graficar los datos históricos. Una vez diseñada la estrategia se puede escoger el que obtenga unos resultados más adecuados.

iv. Lógica inicial

Una vez identificados los condicionantes previos, lo primero para desarrollar el sistema de trading es tener una lógica o idea inicial. Esta idea, como ya se ha dicho, puede ser basada en estrategias ya existentes, en algún tipo de patrón que se haya detectado sobre el precio, en algún comportamiento de un indicador, etc.

La lógica del sistema está constituida por el conjunto de variables y condiciones que determinan los puntos de entrada y de salida. Constituye el núcleo de la estrategia, susceptible de ser mejorada.

La idea inicial debe concordar con los condicionantes descritos, el *time frame*, el activo y el tipo de estrategia. Además, debe ser simple y definir las entradas y salidas del mercado de forma inequívoca para después transformar dicha idea en un script que pueda ser interpretado por la plataforma de *trading* y poder evaluar su potencial sobre datos históricos.

En la lógica inicial se deberán definir las siguientes partes:



FIGURA 23: Lógica Inicial

Para establecer las reglas de la estrategia a menudo se utilizan **indicadores**, los indicadores son herramientas ya creadas que aportan más información o análisis sobre los precios. En la lógica inicial de la estrategia también se definirán **parámetros**, que se emplean para definir los

indicadores (Ej: Periodo de una media móvil) o como parte de la estrategia (Ej: número de barras con cierre positivo).

Una vez se tiene la estrategia descrita, puede resultar interesante dividirla en bloques más pequeños y escribir el algoritmo mediante un **diagrama de flujo**.

3. Programación de la estrategia

Una vez se tiene descrita la lógica base de la estrategia se pasa a su codificación en el lenguaje de la plataforma de *trading* para poder ser ejecutada y probada. Por lo general, las estrategias sencillas siguen una estructura general:

No todas las plataformas utilizan el mismo lenguaje de programación, por lo que se necesitará conocer cuál es el de la plataforma que se va a utilizar.

II.1.1.3.3.2 Evaluación. Se recuerda la tabla con los pasos a seguir hasta esta etapa. Una vez finalizado el diseño del sistema, el siguiente paso será hacer un análisis de los resultados que obtendría utilizando los datos históricos. El objetivo será: por una parte, probar la idea inicial e ir la mejorando, por otra, determinar si la estrategia es robusta y viable para el mercado real o, si por el contrario debe ser descartada.

a) Definición de los periodos *In Sample* y *Out Of Sample*

Lo primero que hay que hacer antes de comenzar la evaluación del sistema es seleccionar la parte del histórico que se va a emplear. Al contrario de lo que pueda parecer, no es conveniente emplear todo el histórico en la evaluación de la estrategia.

Se debe separar el conjunto de los datos históricos en dos períodos:

- **Periodo *In Sample* (IS)**, servirá para evaluar el comportamiento de la estrategia, optimizar los parámetros e ir añadiendo filtros sobre el núcleo de la estrategia.

- **Periodo *Out of Sample* (OS)**, se utiliza para comprobar que el funcionamiento de la estrategia se mantiene estable con datos que no han sido probados previamente, sería lo más parecido a operar en real.

b) **Backtest**

El backtest es la principal diferencia entre el trading discrecional y el trading cuantitativo. Es una herramienta muy útil que permite conocer cómo se hubiera comportado una estrategia si hubiera estado operando en el pasado, para evaluar este comportamiento se utilizan diferentes ratios y métricas de rendimiento.

Tanto el *backtest* como la optimización son procesos sencillos de realizar, ya que vienen implementados en las plataformas de *trading*.

Estas métricas permiten conocer el rendimiento del sistema y proporcionan un buen punto de partida para probar una estrategia de trading.

Algunas de las estadísticas más significativas en el backtesting son:

Ratio	Definición	Observaciones
Beneficio Total Neto o <i>Total Net Profit</i>	Muestra el beneficio total de la estrategia durante el periodo analizado. $\text{Beneficio Bruto} - \text{Pérdida Bruta}$	- No debe ser analizado de manera individual ya que no ofrece ninguna información acerca del riesgo de la estrategia ni de como los beneficios han sido distribuidos.
Factor de Beneficio o <i>Profit Factor</i>	$\frac{\text{valor de operaciones ganadoras}}{\text{valor de operaciones perdedoras}}$	- Si Profit Factor > 1: Estrategia será rentable - El <i>Profit Factor</i> puede servir para comparar dos estrategias.

<p>Máximo Drawdown</p>	<p>Mide la mayor pérdida que se puede dar para una estrategia durante un periodo determinado.</p>	<ul style="list-style-type: none"> - No puede ser evaluado de manera individual ya que la determinación de su tamaño está directamente relacionada con los beneficios obtenidos. - Si la mayor cantidad que un operador está dispuesto a arriesgar < drawdown máximo, la estrategia no es adecuada para esa persona.
<p>Número Total de Trades o Total Number of Trades</p>	<p>Indica el número total de operaciones realizadas durante un periodo.</p>	<ul style="list-style-type: none"> - Es un dato que se debe tener en cuenta para saber si la actividad de la estrategia se ajusta al tipo de operativa. - Es recomendable que la muestra de resultados sea significativa, esto es, que el número de <i>trades</i> del periodo seleccionado sea mayor de 30, como mínimo, para tener validez estadística.
<p>Porcentaje de Operaciones Ganadoras o Percent Profitable</p>	<p>Indica el porcentaje de operaciones positivas realizadas por la estrategia.</p> $\frac{\text{Número de operaciones ganadoras}}{\text{Número de operaciones total}}$	<ul style="list-style-type: none"> - El valor de esta métrica variará según el estilo de cada estrategia. Por ejemplo, una estrategia que trata de capturar largas tendencias del mercado no necesitará un gran porcentaje de operaciones ganadoras para que el sistema resulte rentable ya que las operaciones ganadoras generarán grandes beneficios y las perdedoras, pérdidas más pequeñas.
<p>Beneficio Medio por Operación o Average Trade</p>	<p>Representa la media de las ganancias o pérdidas de todos los trades.</p>	<ul style="list-style-type: none"> - Una buena regla es exigir que el BMO bruto sea al menos 3 veces el gasto originado por comisiones y deslizamientos.

FIGURA 24: Ratios Backtest

c) Esperanza matemática

Una métrica tremendamente importante es la **esperanza matemática** (*Mathematical Expectation*), supone otra forma de evaluar el rendimiento del sistema y se define como “la suma de la probabilidad de cada posible suceso multiplicado por la frecuencia de dicho proceso”, lo que

en términos de trading equivale a la cantidad media que se puede esperar ganar o perder con cada trade.

$$E[X] = (\% \text{ de operaciones ganadoras} * \text{Ganancia media}) - (\% \text{ de operaciones perdedoras} * \text{Pérdida media})$$

Para que un sistema tenga el potencial de dar ganancias, la esperanza matemática debe ser positiva, esto indicaría que en promedio gana más de lo que pierde. Por tanto, lo recomendable será siempre operar con sistemas de trading que tengan una esperanza matemática positiva, ahora bien, está no aporta información sobre la distribución de las ganancias ni la volatilidad de los resultados.

d) Costes de la operativa

Los costes de la operativa o costes de transacción suponen todos los costes que van implícitos al realizar una operación en el mercado, por tanto, al evaluar el sistema deberán tenerse en cuenta ya que van a existir durante la operativa real. Estos costes son:

Comisión del bróker, ya se ha hablado de ella, depende del bróker que se utilice y va en función del volumen operado. Se cobra por entrar y salir del mercado y se llama *roundtrip*.

Además, existen otros costes indirectos que afectan a la operativa:

Spread, también llamado horquilla, se define como la diferencia entre el precio de compra (*ask*) y el de venta (*bid*).

Slippage, por lo general ocurre en momentos de alta volatilidad, cuando el precio se mueve tan rápidamente que al ejecutar una orden de entrada a un precio se acaba ejecutando a un precio distinto del predefinido, puede producirse tanto en la entrada como en la salida y jugar a favor o en contra.

1. Simulación en tiempo real

Antes de empezar a operar con el sistema de trading con dinero real conviene dedicar un periodo para simular la estrategia en tiempo real, donde no se arriesga. En esta fase se podrá detectar si hay algún error de código, y, sobre todo como se verá en la siguiente sección, tomar conciencia de las diferentes incidencias que pueden aparecer cuando se opera a mercado, en lugar de en Backtest. Sobre todo, de sincronización, y especialmente si bróker y plataforma están separados.

II.1.1.3.3 Validación

Una vez completas las fases de diseño y evaluación se pasa a la validación de la estrategia.

a) Backtest en el periodo Out of Sample

Hasta ahora no se ha utilizado el histórico reservado para el periodo Out of Sample, al llegar a esta fase se realiza un *backtest* sobre este periodo. Los resultados obtenidos deberían mantenerse respecto al periodo IS

II.1.1.3.4 Estrategia aceptada

Una vez se han realizado las etapas previas, se puede dar por aceptada la estrategia y comenzar a operar con capital real, esto conlleva una serie de disciplinas.

a) Operativa Real: Monitorización

Una vez realizados los pasos anteriores, se estará en disposición de comenzar la operativa real.

La operativa real consiste en dejar que el sistema de trading automático opere en el mercado con dinero de verdad, sin embargo, esto no es sinónimo de dejar al sistema ejecutándose y esperar

a que lleguen los beneficios sin preocuparse por nada más. La monitorización del sistema de trading es un proceso fundamental que se debe realizar en la operativa real. Se deben vigilar:

II.1.1.4 Estudio de Variable dependiente: Sistema de Trading Algoritmico

La economía digital ha crecido en los últimos años y los sistemas de trading son la muestra de cómo el proceso de digitalización de las finanzas ha transformado el mundo de los negocios. En la actualidad, la tecnología ha llegado a todos los ámbitos de nuestra vida y eso incluye el de las operaciones de inversión online.

De hecho, una de las características más importantes del trading online es que son operaciones de inversión cuyo objetivo es obtener una rentabilidad económica a corto plazo. Por lo tanto, es necesario hacer un seguimiento atento y permanente a los mercados para poder aprovechar las oportunidades de inversión. Es por ello que el presente robot podrá colaborar eficazmente al trabajar con una estrategia automatizada al inversor a la hora de colocar sus operaciones en el mercado.

II.1.1.5 Metodología de Desarrollo

II.1.1.5.1 Metodología SCRUM

Según Scrum Alliance: es "un framework basado en equipos para desarrollar sistemas y productos complejos". Scrum utiliza un proceso iterativo en donde cada iteración (llamadas sprint) es lo más corta posible, manteniendo un ritmo parejo a medida que se avanza con planificación, ejecución y reflexión las cuales permiten descubrir disfunciones en la organización. Scrum es un marco de trabajo iterativo e incremental para el desarrollo de proyectos, productos y aplicaciones. Estructura el desarrollo en ciclos de trabajo llamados Sprint. Son iteraciones de 1 a 4 semanas, y se van sucediendo una detrás de otra. Los Sprint son de duración fija terminan en una fecha específica, aunque no se haya terminado el trabajo, y nunca se alargan. Se limitan en tiempo. Al

comienzo de cada Sprint, un equipo multi- funcional selecciona los elementos (requisitos del cliente) de una lista priorizada. Se comprometen a terminar los elementos al final del Sprint. Durante el Sprint no se pueden cambiar los elementos elegidos. Todos los días el equipo se reúne brevemente para informar del progreso, y actualizan unas gráficas sencillas que les orientan sobre el trabajo restante. Al final del Sprint, el equipo revisa el Sprint con los interesados en el proyecto, y les enseña lo que han construido. (Sutherland, s.f.).

Definición. - Scrum es una metodología para el desarrollo ágil de productos que busca mejorar la eficiencia del producto final, basándose en principios, prácticas y valores ágiles, para de esa manera obtener un ritmo constante de desarrollo con entregas iterativas e incrementales, aportando nuevas funcionalidades en cada iteración o sprint.

Existen 3 roles principales en Scrum: El Scrum Master es un líder servicial para el equipo, y un agente de cambio dentro de la organización. El Dueño del Producto es la principal voz del cliente, establece una visión y usa un proceso de priorización continua para lograrla. El Equipo es auto-gestionado, multi-funcional, con poder para realizar el trabajo.

(Sutherland) Todo en SCRUM está dividido en periodos de tiempo, esto nos ayuda a mantener el rumbo del proyecto y a entregar en las fechas previstas ya que cada reunión tiene unas pautas y unos límites de tiempo preestablecidos. Están prohibidas esas reuniones interminables,

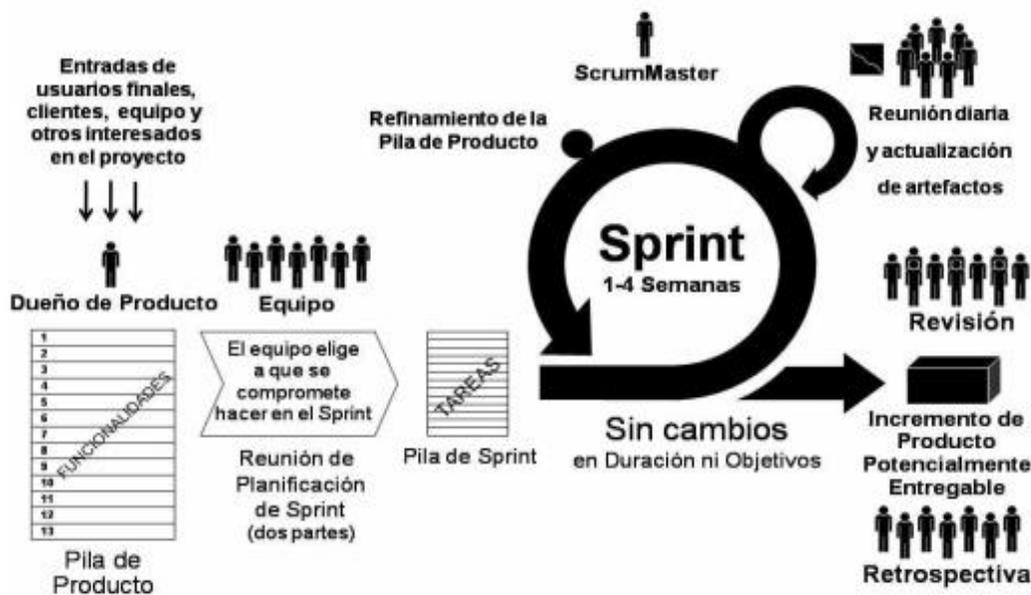


Figura 25: Proceso SCRUM

improductivas y tremendamente costosas.

II.1.1.5.2 UML (Lenguaje Unificado de Modelado).

UML es ante todo un lenguaje. Un lenguaje proporciona un vocabulario y unas reglas para permitir una comunicación. En este caso, este lenguaje se centra en la representación gráfica de un sistema.

a) Los objetivos de UML Según Johnny Quintero son muchos, pero se pueden sintetizar sus funciones:

- **Visualizar:** UML permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- **Especificar:** UML permite especificar cuáles son las características de un sistema antes de su construcción.

- **Construir:** A partir de los modelos especificados se pueden construir los sistemas diseñados.
- **Documentar:** Los propios elementos gráficos sirven como documentación del sistema desarrollado que pueden servir para su futura revisión. (Quintero, s.f.)

b) Un modelo UML está compuesto por tres clases de bloques de construcción:

- **Elementos:** Los elementos son abstracciones de cosas reales o ficticias (objetos, acciones, etc.).
- **Relaciones:** relacionan los elementos entre sí.
- **Diagramas:** Son colecciones de elementos con sus relaciones.

c) UML es además un método formal de modelado. Esto aporta las siguientes ventajas:

Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y a la inversa (a partir del código fuente generar los modelos). Esto permite que el modelo y el código estén actualizados, con lo que siempre se puede mantener la visión en el diseño, de más alto nivel, de la estructura de un proyecto. (Ventajas y Desventajas UML, s.f.)

- Mayor rigor en la especificación.
- Permite realizar una verificación y validación del modelo realizado.

II.1.1.5.2.1 Tipos de Diagramas Utilizados

II.1.1.5.2.1.1 Diagramas de Clases

Es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargarán del funcionamiento y la

relación entre uno y otro. (Class Diagram, 2017)

II.1.1.5.2.1.2 Diagramas de Casos de Uso

Es un diagrama que captura las interacciones de los casos de uso y los actores. Describe los requisitos funcionales del sistema, la forma en la que las cosas externas (actores) interactúan a través del límite del sistema y la respuesta del sistema. (Use case, 2015)

II.1.1.5.2.1.3 Diagramas de Secuencias

Es un diagrama que muestra una representación estructurada del comportamiento como una serie de pasos secuenciales a lo largo del tiempo. Se usa para representar el flujo de trabajo, el paso de mensajes y cómo los elementos en general cooperan a lo largo del tiempo para lograr un resultado. (Visual Studio, 2015)

II.1.1.5.2.1.4 Diagramas de Actividades

Que se usa para modelar el comportamiento de un sistema, y la manera en que este comportamiento está relacionado con un flujo global del sistema. Se usan los caminos lógicos que sigue un proceso basado en varias condiciones, concurrencia en el proceso, los datos de acceso, interrupciones y otras alternativas del camino lógico para construir un proceso, sistema o procedimiento.

II.1.1.5.2.1.5 Diagramas de Paquetes

Muestra como un sistema está dividido en agrupaciones lógicas mostrando las dependencias entre esas agrupaciones. Dado que normalmente un paquete está pensado como un directorio, los diagramas de paquetes suministran una descomposición de la jerarquía lógica de un sistema. Los Paquetes están normalmente organizados para maximizar la coherencia interna dentro de cada paquete y minimizar el acoplamiento externo entre los paquetes. Con estas líneas 70 maestras sobre la mesa, los paquetes son buenos elementos de gestión. Cada paquete puede

asignarse a un individuo o a un equipo, y las dependencias entre ellos pueden indicar el orden de desarrollo requerido.

II.1.1.5.2.1.6 Diagrama de Componentes

Ilustra los fragmentos de software, controladores embebidos, etc. que conformarán un sistema. Un diagrama de componentes tiene un nivel de abstracción más elevado que un diagrama de clase - usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Estos son bloques de construcción, como así eventualmente un componente puede comprender una gran porción de un sistema.

II.1.1.5.2.1.7 Diagrama de Navegación

Muestra cómo y dónde se desplegará el sistema. Las máquinas físicas y los procesadores se representan como nodos, y la construcción interna puede ser representada por nodos o artefactos embebidos. Como los artefactos se ubican en los nodos para modelar el despliegue del sistema, la ubicación es guiada por el uso de las especificaciones de despliegue. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria. (Rosenberg D. & Stephen M., 2007)

II.1.1.6 Herramientas de Construcción de Software

II.1.1.6.1 Software Ideas Modeler

Software Ideas Modeler es una herramienta esencial en la ingeniería de software. Es una herramienta CASE fácil de usar que ayuda a diseñar y documentar software utilizando varias notaciones de diagramas y herramientas de modelado. Incluye un modelador UML, un diseñador de diagramas entidad-relación, un creador de diagramas de flujo, herramientas de creación de esquemas para el desarrollo de aplicaciones web, móviles y de escritorio, un editor BPMN y otras características útiles de creación de diagramas.

Aproveche las herramientas proporcionadas para un mejor diseño del sistema de software. Describir la arquitectura del software utilizando la documentación estructurada, los glosarios y las funciones de control de versiones del modelo.

Las funciones de ingeniería inversa y generación de código en Software Ideas Modeler permiten a los usuarios convertir sus bases de datos y código fuente en diagramas, así como generar código fuente y script de base de datos a partir de sus diagramas y modelos. (software ideas, 2022)



FIGURA 26: Software Ideas Modeler

II.1.1.6.2 Meta editor 5

Es un editor de robots comerciales que permite escribir aplicaciones de cualquier complejidad

Para desarrollar robots comerciales e indicadores técnicos en MetaTrader 5 se usa un editor especializado propio: MetaEditor. Forma parte integrante del entorno de desarrollo Integrado EDI MQL5 y guarda estrecha relación con la plataforma comercial MetaTrader 5.

MetaEditor es un editor para crear robots comerciales, indicadores técnicos, scripts y bibliotecas. El editor sirve para trabajar con el código fuente de los programas escritos en MQL5 y su posterior compilación en archivos de ejecución. Debido a la estrecha integración del editor en el terminal, los programas preparados aparecen en MetaTrader 5 automáticamente y pueden ser ejecutados de forma inmediata.

MetaEditor es capaz de reconocer diferentes construcciones de lenguaje. Esto permite dar pistas sobre el uso de funciones e iluminar diferentes elementos del código. De esta forma, el

propio editor hace más fácil orientarse en el código de los programas comerciales y acelera su desarrollo. (metatrader5, 2022)



FIGURA 27:Meta Editor5

II.1.1.6.3 Quandata Manager

Descarga rápido los datos históricos en alta calidad para forex, acciones, criptomonedas, CFD, metales de múltiples fuentes diferentes, en calidad de tick o minuto.

Se utiliza servidores CDN propios para una descarga de datos rápida

Importa datos de varias fuentes, el importador que permite importar datos históricos desde un archivo de texto o CSV en prácticamente cualquier formato.

Facilita la manipulación, el uso, la exportación y el análisis. (QuantDataManager, 2022)



FIGURA 28:QuantDataManager

II.1.1.6.4 Metatrader5

MetaTrader 5, también conocida como MT5, es una plataforma de trading desarrollada por MetaQuotes Software Corp en el año 2010. Es una plataforma multimercado que le asiste a la hora de comerciar en el mercado Fórex, la bolsa de valores, así como con futuros. Proporciona oportunidades para trabajar en todos los mercados financieros: análisis técnico y fundamental,

trading algorítmico, permite copiar operaciones de traders exitosos, y también da la posibilidad de escribir sus propios indicadores técnicos y robots comerciales.

Gracias al desarrollo de la tecnología moderna y a internet ahora existe la oportunidad de trabajar en la bolsa de valores con la ayuda de programas de computadora multi-función: las plataformas de negociación. El trader establece en ella órdenes para comprar o vender acciones, y el bróker hace en su nombre la operación en la bolsa. MetaTrader 5 es precisamente ese tipo de plataforma, pues proporciona al usuario los instrumentos más potentes para comerciar en la bolsa, así como análisis y trading algorítmico y social.

MetaTrader 5 da soporte al trading algorítmico, es decir, el comercio en el mercado de valores según un algoritmo implementado con la ayuda de robots comerciales. (metatrader5., 2021)



FIGURA 29:Metatrader5

II.1.1.6.5 MQL5

MetaQuotes Language 5 (MQL5) es un lenguaje especializado de alto nivel de programación orientada objetos, que permite crear robots comerciales e indicadores técnicos. Está basado en el concepto del lenguaje de programación ampliamente difundido C++. Pero gracias a

su especialización en los mercados financieros, MQL5 permite resolver de la mejor forma posible las tareas relacionada con el trading.

MQL5 contiene una gran cantidad de funciones necesarias para el análisis de las cotizaciones, igualmente incluye indicadores técnicos, así como medios de gestión y control de posiciones comerciales. Gracias a estas amplias posibilidades, se puede delegar en las aplicaciones MQL5 todo el comercio y el análisis en los mercados financieros.

Los programas escritos en MQL5 tienen distintas propiedades y finalidades:

Un indicador personalizado es un indicador técnico pensado exclusivamente para analizar las cotizaciones de las divisas y las acciones. A diferencia de los indicadores incorporados, usted mismo puede crear esta herramienta a su medida y colocar dentro cualquier algoritmo que considere necesario. Los indicadores no tienen acceso a las funciones de trading y no pueden realizar operaciones comerciales.

Un script es un programa que ejecuta alguna acción programada una sola vez. Estos programas también tienen acceso a todas las funciones analíticas y comerciales. Sin embargo, a diferencia de los Asesores Expertos, los scripts se ejecutan solamente una vez y no con cada tick.

Una biblioteca es un conjunto de funciones personalizadas que sirve para almacenar y distribuir los bloques de programas usados con frecuencia. (Manual de referencia del lenguaje MQL 5, 2000-2022)

The logo for MQL5, with 'MQL' in a dark grey font and '5' in a bright orange font.

FIGURA 30:MQL5

II.1.1.7 Sistema de información Automatizado

II.1.1.7.1 El internet

Algunos definen Internet como "La Red de Redes", y otros como "La Autopista de la Información". Efectivamente, Internet es una Red de Redes porque está hecha a base de unir muchas redes locales de ordenadores, o sea de unos pocos ordenadores en un mismo edificio o empresa. Por la Red Internet circulan constantemente cantidades increíbles de información. Por este motivo se le llama también La Autopista de la Información. Hay 50 millones de "Internautas", 86 es decir, de personas que "navegan" por Internet en todo el Mundo. Se dice "navegar" porque es normal el ver información que proviene de muchas partes distintas del Mundo en una sola sesión. Una de las ventajas de Internet es que posibilita la conexión con todo tipo de ordenadores, desde los personales, hasta los más grandes que ocupan habitaciones enteras. Incluso podemos ver conectados a la Red cámaras de vídeo, robots, y máquinas de refrescos, etc.

II.1.1.7.2 La Word Wide Web o la Web

La World Wide Web consiste en ofrecer una interface simple y consistente para acceder a la inmensidad de los recursos de Internet. Es la forma más moderna de ofrecer información. El medio más potente. La información se ofrece en forma de páginas electrónicas. El World Wide Web o WWW o W3 o simplemente Web, permite saltar de un lugar a otro en pos de lo que no interesa. Lo más interesante es que con unas pocas ordenes se puede mover por toda la Internet. La World Wide Web permite una manera más organizada de acceder a la información disponible en Internet, presentando una interfaz amigable con el usuario mediante navegadores como Netscape, Mosaic y Microsoft Internet Explorer, Mozilla FireFox. El surgimiento de la World Wide Web ha ayudado a un crecimiento considerable de Internet en la actualidad. Compañías pequeñas, empresas grandes, ayuntamientos, estados, gobiernos de distintos países, universidades, bibliotecas, están presentes en Internet. Es básicamente un medio de comunicación de texto,

gráficos y otros objetos multimedia a través de Internet, es decir, la web es un sistema de hipertexto que utiliza Internet como su mecanismo de transporte o desde otro punto de vista, una forma gráfica de explorar Internet.

II.1.1.8 Sistemas de Información vía Web

La evolución de Internet como red de comunicación global y el surgimiento y desarrollo del Web como servicio imprescindible para compartir información, creó un excelente espacio para la interacción del hombre con la información hipertextual, a la vez que sentó las bases para el desarrollo de una herramienta integradora de los servicios existentes en Internet. Los sitios Web, como expresión de sistemas de información, deben poseer los siguientes componentes:

- Usuarios. 87
- Mecanismos de entrada y salida de la información.
- Almacenes de datos, información y conocimiento.
- Mecanismos de recuperación de información.

Pudiésemos definir entonces como sistema de información al conjunto de elementos relacionados y ordenados, según ciertas reglas que aporta al sistema objeto, es decir, a la organización a la que sirve y que marca sus directrices de funcionamiento, la información necesaria para el cumplimiento de sus fines; para ello, debe recoger, procesar y almacenar datos, procedentes tanto de la organización como de fuentes externas, con el propósito de facilitar su recuperación, elaboración y presentación. Actualmente, los sistemas de información se encuentran al alcance de las grandes masas de usuarios por medio de Internet; así se crean las bases de un nuevo modelo, en el que los usuarios interactúan directamente con los sistemas de información para satisfacer sus necesidades de información.

II.1.2 Plan de desarrollo de software.

II.1.2.1 Introducción.

Este Plan de Desarrollo del Software es una versión preliminar preparada para ser incluida en la propuesta elaborada como respuesta al proyecto de prácticas de la asignatura de Taller III de la Carrera de Ingeniería Informática de la Facultad de Ciencias y Tecnología de la Universidad Autónoma Juan Misael Saracho. Este documento provee una visión global del enfoque de desarrollo propuesto.

El proyecto ha sido ofertado por el universitario Wilfredo Marcial Aparicio López, basado en la metodología de Rational Unified Process (RUP) en la que únicamente se procederá a cumplir con las tres primeras fases, las cuales marcan la metodología. Es importante destacar esto puesto que utilizaremos la terminología SCRUM en este documento. Se incluirá el detalle para las fases de Inicio, Elaboración y Construcción.

El enfoque de desarrollo propuesto constituye una configuración del proceso SCRUM de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los entregables que serán generados. Este documento es a su vez uno de los artefactos de la metodología SCRUM.

II.1.2.2 Propósito.

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él que se describe el enfoque de desarrollo del software.

Los miembros del equipo de desarrollo lo usan para entender lo qué deben hacer, cuándo deben hacerlo y qué otras actividades dependen de ello.

Los usuarios del Plan de Desarrollo del Software son:

- El director responsable del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.

II.1.2.3 Alcance del documento.

El plan de desarrollo del software describe el plan global usado para el desarrollo del proyecto titulado **mejorar las estrategias de inversión en el mercado de divisas**

el detalle de las iteraciones individuales se describe en los planes de cada iteración. lo que se pretende alcanzar con este plan es documentar las mejoras planteadas en la gestión de las operaciones realizadas en los mercados financieros.

posteriormente el avance del proyecto y seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas, siendo este documento la última versión.

II.1.2.4 Resumen.

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

Vista General del Proyecto.- proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

Organización del Proyecto.- describe la estructura organizacional del equipo de desarrollo.

Gestión del Proceso.- explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

II.1.3 Vista general del proyecto

II.1.2.5.1 Propósito.

Contribuir a mejorar las estrategias de inversión en general en el mercado de divisas automatizando las operaciones.

II.1.2.5.2 Alcance

Gestión de las operaciones.

II.1.2.5.3 Limitaciones

El sistema no gestionara múltiples cuentas al mismo tiempo.

II.1.2.5.5 Suposiciones y restricciones

II.1.2.5.5.1 Suposiciones

- El personal involucrado apoya de manera constante y voluntaria, proporciona toda la información necesaria para un correcto desarrollo y evolución del sistema.
- El equipo de trabajo cuenta con todas las herramientas necesarias para la elaboración del sistema.

II.1.2.5.5.2 Restricciones

- Se realizará el Proyecto haciendo énfasis en la gestión y automatización de la operación que se realizan en los mercados financieros.

II.1.2.7 Organización del proyecto

II.1.2.7.1 Participantes

Empresa objetivo del proyecto

El ámbito donde se realiza el proyecto es las principales bolsas de valores.

Director y Desarrollador del Proyecto

Aparicio López Wilfredo, alumno del 9no semestre de la Carrera de Ingeniería Informática en la Facultad de Ciencias y Tecnología de la Universidad Autónoma Juan Misael Saracho. Con experiencia en metodologías de desarrollo, notaciones UML, herramientas CASE, desarrollo de sistemas en entorno Escritorio y Web, diseño de Datos, conocimiento del lenguaje de programación Java y metodologías de Prueba.

Cabe recalcar que el universitario Aparicio López Wilfredo tendrá todos los roles que se necesita para el desarrollo de un proyecto que son: Director de Proyecto, Analista de Sistemas, Programador e Ingeniero de Software.

II.1.2.7.2 Roles y responsabilidades

A continuación, se describen las principales responsabilidades de cada uno de los puestos en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en Scrum.

Puesto	Responsabilidad
Director de Proyecto	El jefe de proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos. El jefe de proyecto también establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, el jefe de proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos. Planificación y control del proyecto.

Analista de Sistemas	Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas. Elaboración del Modelo de Análisis y Diseño. Colaboración en la elaboración de las pruebas funcionales y el modelo de datos.
Programador	Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario.
Ingeniero de Software	Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.

Tabla 15: Roles y Responsabilidades

II.1.2.7.3. Estimaciones del Proyecto

II.1.2.7.3.1. Plan del Proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto durante el desarrollo del proyecto.

II.1.2.7.3.2. Plan de fases

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra una la distribución de tiempos y el número de iteraciones de cada fase (para las fases de Construcción y Transición es sólo una aproximación muy preliminar).

FASE	NRO.DE INTERACCIONES	DURACION (días)
Fase de inicio	1	25
Fase de Elaboración	1	20
Primer Sprint	1	25
Segundo Sprint	1	25
Tercer Sprint	1	25
Cuarto Sprint	1	25
Pruebas adicionales al sistema	1	25
Capacitación	1	1

Tabla 16:Plan de Fases

Los Hitos Que Marcan El Final De Cada Fase Se Describen En La Siguiete Tabla.

ROLES	HITO
Etapa de Requerimientos	Investigación sobre la automatización de operaciones de trading en el mercado de divisas a través en la página oficial de la plataforma Metatrader 5.
Elaboración	Selección de Herramientas de desarrollo y software basado en la información recabada para el desarrollo de sistemas de inversión.

Primer Sprint	Se descarga los históricos de gráficos de las divisas y realización de test de precios y tiempos
Segundo Sprint	Automatización de la información recabada con funcionamiento de identificación de zonas horarias operativas y tendencias del mercado.
Tercer Sprint	Implementación de la Estrategia para automatizar las operaciones en el Mercado de divisas.
Evaluación	Ejecución del algoritmo en condiciones pasadas del mercado en un periodo IS.
Validación	Ejecución del algoritmo en condiciones pasadas del mercado en un periodo OS.
Capacitación	Los inversores que interactúan con el sistema recibirán una capacitación sobre el uso del sistema

Tabla 17: Los Hitos Que Marcan El Final De Cada Fase

II.1.2.8 Oportunidad del Negocio

Con este sistema se mejorará el proceso de introducir operaciones en la plataforma de metatrader5.

II.1.2.8.1 Sentencias que define el Proyecto

El problema de	No se tiene un proceso más fácil y seguro para realizar operaciones en el mercado de divisas.
Afecta a	Los inversores
El impacto asociado es	Mejorar el proceso y tiempo de inversión para los usuarios.

Una solución adecuada sería	Automatizar todos los procesos mediante una estrategia de trading.
------------------------------------	--

Tabla 18: Sentencias que definen el Proyecto

II.1.2.8.2 Sentencias que define la posición del proyecto

Para	Inversores
Quienes	Interactúan de manera directa con el sistema.
El nombre del producto	Mejorar las estrategias de inversión en el mercado de divisas.
Que	Mejorará el Proceso de Inversion Mejorará el tiempo del ingreso de la operación. Ofrecerá una estrategia con grandes probabilidades de éxito.
No como	El procedimiento que se realiza actualmente, es un proceso que puede tener percances, porque se lo realiza manualmente.
Nuestro Producto	Coloca operaciones de Forma automática. No depende de ningún otro sistema.

Tabla 19: Sentencias que define la Posición del Proyecto

II.1.3 Desarrollo de la estrategia o trading algorítmico

II.1.3.1 Diseño

II.1.3.1.1 Infraestructura

II.1.3.1.1.1 Hardware

Es importante disponer de un PC corei3 con 8 Gb. de ram y una conexión a internet de velocidad media.

II.1.3.1.1.2 Software

Se empleará la plataforma de trading **Metatrader 5**, el bróker en el que se hace las pruebas es The5rs y para codificar y evaluar la estrategia se usara MetaQuotex. Se dispone de una serie de datos históricos del 1 de enero de 2017 Al 23 de noviembre del 2022

II.1.3.1.2 Lógica inicial

II.1.3.1.2.1 Estilo de Trading: Las posiciones se abren y cierran en el mismo día, no dejando operaciones abiertas a fin de sesión por lo que el estilo de trading de la estrategia es Trading intradia .

II.1.3.1.2.2 El capital inicial: debe ser suficiente para cubrir el DDmax;

II.1.3.1.2.3 El Riesgo asumible: de esta estrategia considera un perfil conservador arriesgará en torno a un 1% en cada operación.

II.1.3.1.2.4 Par de Divisas a Operar: Se ha elegido el par de GBP/JPY: Libra / Yen y GBP/USD: Libra / Dólar para la ejecución del trading algorítmico, porque son pares de alta volatilidad en la bolsa de Londres y son parte de la estrategia se basa en la sesión anterior que sería la sesión asiática.

II.1.3.1.2.5 Requisitos del Algoritmo, Product Backlog:

Product Backlog					
Story ID	Organization	Taller III		Priority	Sprint #
	Project	Algoritmo De Trading			
	Scrum Master	Wilfredo Aparicio			
	Product Owner	Wilfredo Aparicio			
Title	Descripción del requerimiento				
1	REQUISITOS ALGORITMO.	DEL	El algoritmo se diseña a partir de una combinación de ideas de análisis técnico y nuevos conceptos de trading institucional para inversiones a corto plazo, que busca aprovechar las zonas de mayor liquidez guiados, la operativa de Volatility Break Out (VBO) y una serie de datos como zonas de soporte o resistencia del precio y por lo tanto como áreas de impulso dentro de una tendencia definida.	5	1
2	CONSIDERACIONES GENERALES		<ul style="list-style-type: none"> Los sistemas de trading basados en la utilización Promedios Móviles tienen buenos resultados especialmente en fuertes tendencias del mercado. Su mayor rentabilidad se genera en las “rupturas de precio” que generan extensos movimientos alcistas o bajistas. La estrategia se plantea indistintamente para movimientos alcistas y bajistas. La liquidez de compradores y vendedores se puede identificar con mayor facilidad en máximos y mínimos iguales dentro de un rango operativo El algoritmo se debe crear en el lenguaje MQL5, utilizando para ello la herramienta Metaquotex como IDE de programación enfocado en el desarrollo de sistemas expertos. <p>Para que se dé una operación de compra se deben cumplir tres parámetros técnicos. Todos deben quedar definidos con precios BID (oferta):</p> <ul style="list-style-type: none"> ZONA HORARIA: La zona horaria operativa del mercado debe haber terminado la sesión asiática. El precio toma liquidez por debajo del rango asiático es decir debe haber precio menor al rango de Asia 	5	1
4	PARAMETROS DE OPERACIÓN, REGLAS DE ENTRADA, DE COMPRA	DE	<ul style="list-style-type: none"> CONFIGURACION DE LA VELA: La Última Vela debe tener una configuración Alcista, dada por las diferencias de los precios de apertura, mínimo y máximo. Si $(P \text{ Apertura} - P \text{ Mínimo}) > (P \text{ Máximo} - P \text{ Apertura})$ Entonces VELA ALCISTA POSICION DE LA VELA: La última Vela debe tener su precio Mínimo por debajo del Rectángulo dibujado del rango asiático. ORDEN PENDIENTE DE COMPRA: Se coloca una orden pendiente en el 50% del Rectángulo dibujado del rango asiático con un stop por debajo del rango y un beneficio 1:3. <p>Si se Precio > 50% rango Asia se abre posición de compra.</p>	5	1

5	PARAMETROS DE OPERACIÓN, REGLAS DE ENTRADA, DE VENTA	<p>Para que se dé una operación de compra se deben cumplir tres parámetros técnicos. Todos deben quedar definidos con precios ASK (demanda):</p> <ul style="list-style-type: none"> ZONA HORARIA: La zona horaria operativa del mercado debe haber terminado la sesión asiática. CONFIGURACION DE LA VELA: La Ultima Vela debe tener una configuración bajista, dada por las diferencias de los precios de apertura, mínimo y máximo. Si $(P \text{ Apertura} - P \text{ Mínimo}) < (P \text{ Máximo} - P \text{ Apertura})$ Entonces VELA BAJISTA POSICION DE LA VELA: La última Vela debe tener su precio máximo por encima del Rectángulo dibujado del rango asiático. ORDEN PENDIENTE DE VENTA: Se coloca una orden pendiente en el 50% del Rectángulo dibujado del rango asiático con un stop por arriba del rango y un beneficio 1:3. <p>Si se Precio < 50% rango Asia se abre posición de venta.</p> <ul style="list-style-type: none"> Todas las Operaciones de Compra y de Venta: Debe cerrar todas las operaciones abiertas de compra y venta, cuando hayan llegado a su take proffit (objetivo de beneficio). O inicia la sesión asiática de un nuevo día. 	5	1
6	PARAMETROS DE OPERACIÓN, REGLAS DE CIERRE.	<p>Se podrá operar durante el horario de mercado de todas la sesiones Diarias es decir (NEW YORK, LONDRES ,TOKYO, SIDNEY).</p>	5	1
7	HORARIO EN EL QUE SE VA A LLEVAR A CABO LAS OPERACIONES			
8	OTRAS CONSIDERACIONES OPERATIVAS.	<ul style="list-style-type: none"> Lote por Operación: El algoritmo debe trabajar con tamaño de lote de acuerdo al riesgo no mayor al 1% de la cuenta para todas las operaciones, y modificable en el momento que se inicie a operar. Pares a Operar: Debe operar en cualquier producto par de divisas de USD, GBP Y JPY, modificable al momento que se ponga en operación el algoritmo. Temporalidad de la Operativa: El “time frame”, la temporalidad de mejor rendimiento es de M15 es decir gráficos de velas con 15 minutos. 	5	1

II.1.3.1.2.6 Diagrama de flujo Del Algoritmo.

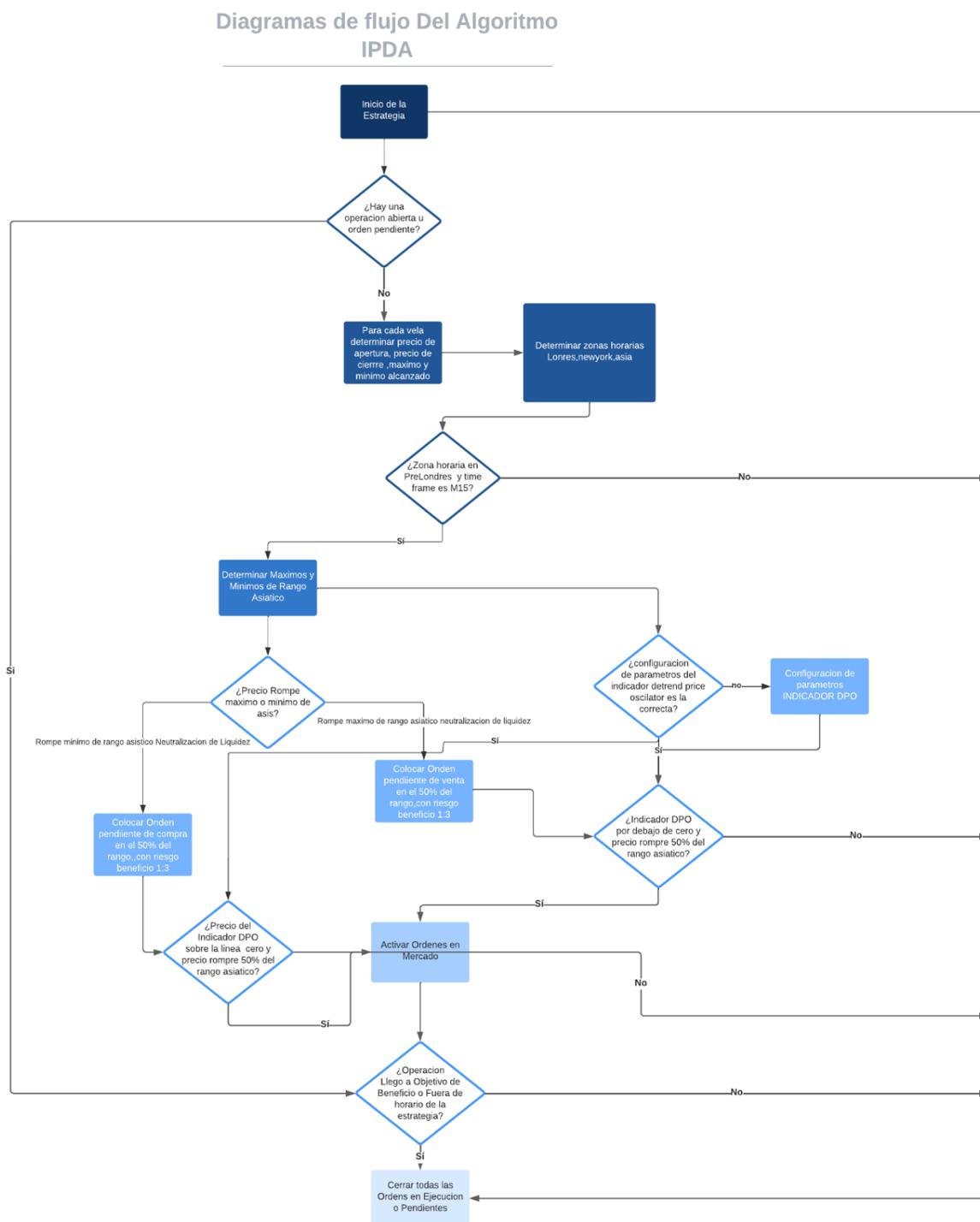


FIGURA 31:Diagrama de Flujo del Algoritmo de Trading

II.1.4.1.3 Etapa de Sprints

II.1.4.1.3.1 Sprint 1

II.1.3.1.3.1.1 Descripción. El Primer Sprint Comprende el test de precios y tiempos en los gráficos de las divisas., que son la base para el funcionamiento de todo el sistema

II.1.3.1.3.1.2 Propósito. El propósito del primer Sprint de administración es obtener las bases para el sistema a través de la prueba de la calidad de los gráficos históricos

II.1.3.1.3.1.3 Diagramas de Comportamiento

II.1.3.1.3.1.3.1 Diagramas de Casos de Uso

II.1.3.1.3.1.3.1.1 Diagrama de Caso de Uso de Actores Del Sistema

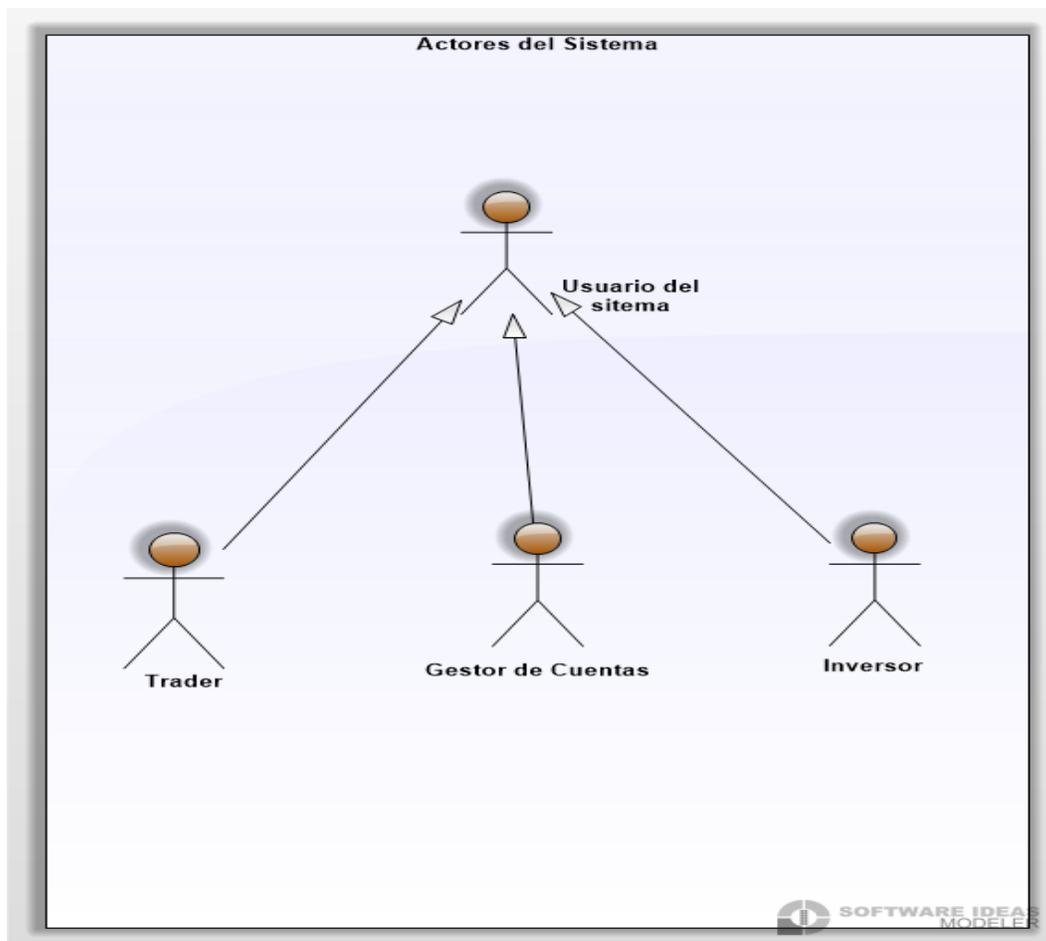


FIGURA 32:Diagrama de actores del sistema

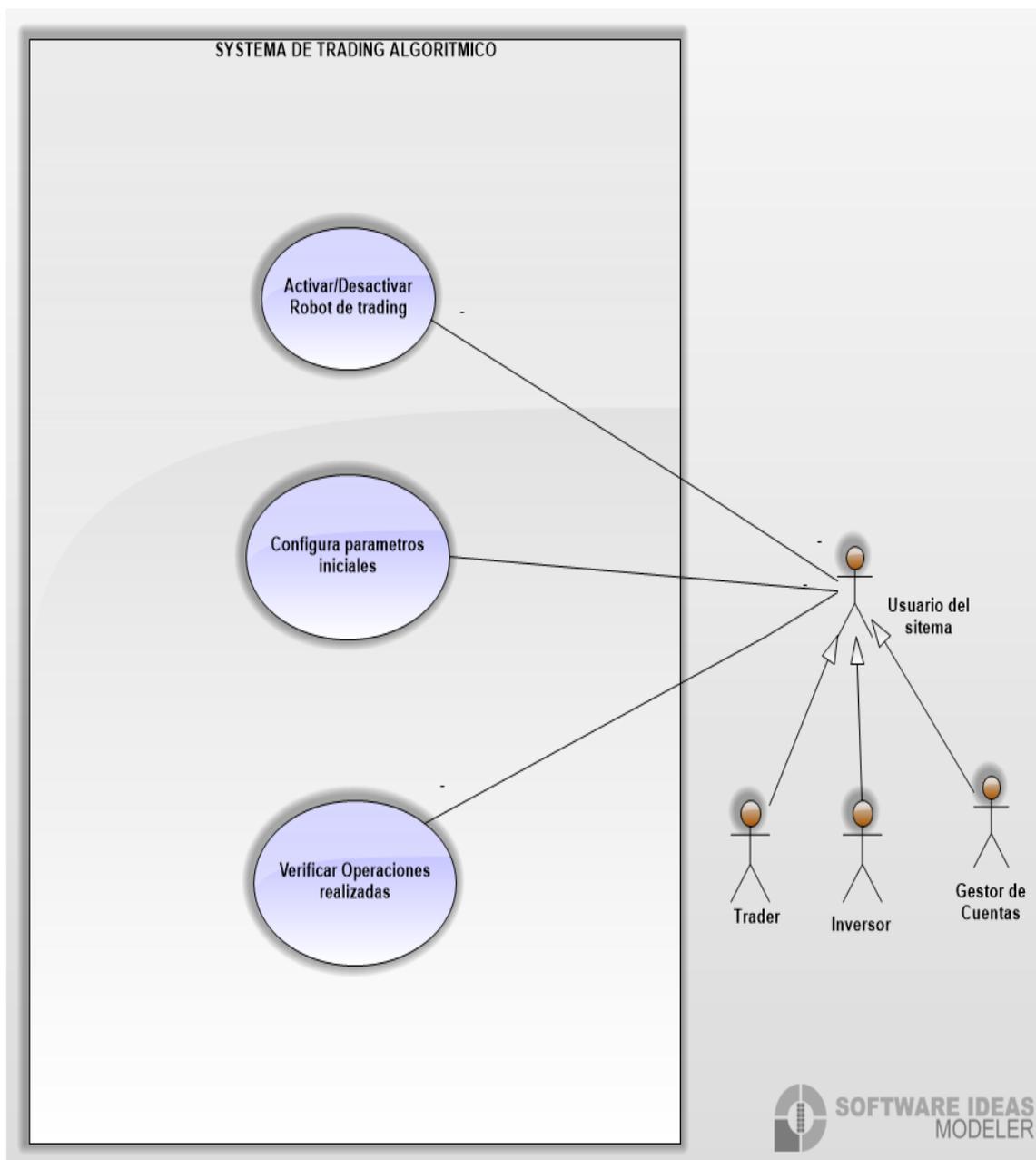
II.1.3.1.3.1.3.1.2 Diagrama De Casos De Uso De Iniciar Sistema De Algoritmico

FIGURA 33: Diagrama De Casos De Uso De Iniciar Sistema De Algoritmico

II.1.3.1.3.1.3.1.2.1 Especificación Casos de Uso De Iniciar Sistema De Trading

Nombre del caso de uso:	Iniciar Sistema Trading
Actor:	Usuario del Sistema
Descripción:	Especifica el flujo para que el usuario del sistema pueda Iniciar el sistema de Trading en la plataforma de inversión.
Precondiciones:	La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5.
Flujo normal:	<ol style="list-style-type: none"> 1. El Usuario del sistema con un doble click inicia el Robot de Trading, el sistema muestra una interfaz donde se configura parámetros iniciales y parámetros de entrada. 2. El Usuario del Sistema Activa el Robot de Trading en la plataforma de inversión Metatrader 5, en la opción activar trading algorítmico. 3. El Usuario Del Sistema puede verificar la operaciones realizadas u operaciones en ejecución en la plataforma de Inversion.
Flujo alternativo:	El Usuario del Sistema puede desactivar el Robot de Trading para detenerlo.
Pos condiciones:	El sistema actualizado el balance de capital.

Tabla 20: Especificación Casos de Uso De Iniciar Sistema De Trading

II.1.3.1.3.1.3.1.3 Diagrama De Casos De Uso Del Sistema De Trading

Algorítmico

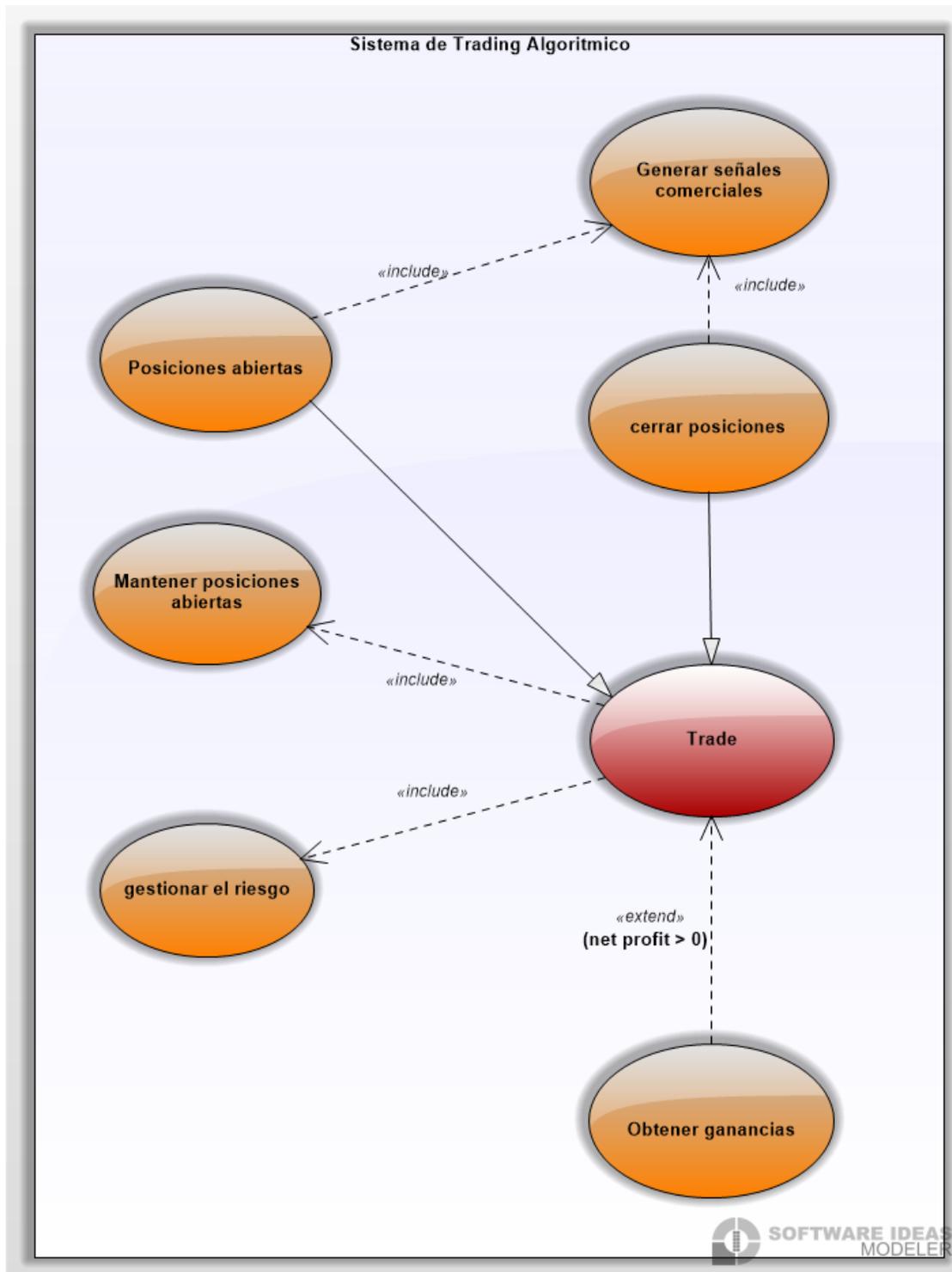


FIGURA 34:Diagrama De Casos De Uso Del Sistema Automático De Trading

II.1.3.1.3.1.3.1.3.1 Especificación Casos de Uso Sistema de Trading Algorítmico

Nombre del caso de uso:	Sistema Automático de Trading Algorítmico
Actor:	Robot de Trading
Descripción:	Especifica el flujo para de La operativa que desarrolla al interactuar en el mercado de divisa.
Precondiciones:	<ul style="list-style-type: none"> ✓ El Robot de trading debe estar Habilitado desde la plataforma de trading Metatrader 5 en el sistema con su correspondiente configuración. ✓ Los datos del bróker deben estar sincronizador en horario del código del Robot de Trading. ✓ El Inversor debe asignar el par de divisa a operar por el Robot de trading.
Flujo normal:	<ol style="list-style-type: none"> 1. El Robot de Trading comprobara si no existen operaciones abiertas cada vez que inicie su ejecución. 2. El Robot de Trading comprobara los precios de apertura y cierre de cada vela en tiempo real de la divisa cotizada cada vez que inicie su ejecución. 3. El Robot de trading analiza las condiciones que debe cumplir el mercado antes de realizar una operación. 4. Una vez el mercado cumple las condiciones de la estrategia el robot de trading realiza una operación llamada (Trade). 5. El Robot de Trading gestionara el trade colocando un límite de perdida y un objetivo de ganancia. 6. El Robot de Trading gestionara el tiempo que el trade debe estar abierto en el mercado comprobando zonas operativas por día.

	<p>7. El Robot de Trading gestionara el capital antes de realizar cada trade.</p> <p>8. El Robot de Trading gestionara cuando cerrar cada trade.</p>
Flujo alternativo:	9. Al ser un sistema en tiempo real si la configuración de los datos del Robot de trading no es correcta o no existe una conexión a internet el Robot de Trading no se ejecuta.
Pos condiciones:	El sistema actualizado el balance de capital.

Tabla 21 Especificación de Casos de Uso Sistema de Trading Algorítmico

II.1.3.1.3.1.3.2 Especificación Casos de Uso Trade.

Nombre del caso de uso:	Trade
Actor:	Robot de Trading
Descripción:	Especifica el flujo y estado de una orden ya sea de compra o de venta que ejecuto el robot de trading.
Precondiciones:	<ul style="list-style-type: none"> ✓ La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. ✓ El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5. ✓ El Robot de Trading debe estar configurado y activado en la plataforma de inversión.
Flujo normal:	<ol style="list-style-type: none"> 1. El Robot de Trading verifica si hay Posiciones abiertas (Trade) 2. El Robot de Trading verifica y cierra posiciones abiertas si se dan las condiciones antes de generar una nueva señal.

	<ol style="list-style-type: none"> 3. El Robot de Trading verifica que se cumplan las condiciones, genera una señal de entrada y coloca una orden en el mercado ya sea de venta o de compra (Trade), según el análisis de la estrategia de robot de trading. 4. El Robot de Trading Gestiona el riesgo definiendo un límite de pérdida y un objetivo de beneficio del Trade. 5. El Robot de Trading mantiene las posiciones abiertas (Trade), si el mercado se encuentra en zona y horario operativo según estrategia. 6. El Robot de Trading verifica si hay Posiciones abiertas (Trade), cierra ordenes que no se hayan activado, que llegaron al objetivo de beneficio o tocado el límite de pérdida y actualiza el balance del capital para un nuevo día.
<p><i>Flujo alternativo:</i></p>	<ol style="list-style-type: none"> 7. El Robot de Trading detecte Posiciones abiertas (Trade) que no llegan al objetivo de beneficio 8. El Robot de Trading no genere ninguna señal debido a las condiciones del mercado. 9. El Robot de Trading sigue Monitoreando el evento Ontick().
<p><i>Pos condiciones:</i></p>	<p>El sistema actualizado el balance de capital.</p>

Tabla 22 Especificación de Casos de Uso Trade.

II.1.3.1.3.1.3.3 Especificación Casos de Uso Posiciones abiertas.

Nombre del caso de uso:	Posiciones Abiertas
Actor:	Robot de Trading
Descripción:	Especifica el flujo y estado de una orden ya sea de compra o de venta que ejecuto el robot de trading.
Precondiciones:	<ul style="list-style-type: none"> ✓ La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. ✓ El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5. ✓ El Robot de Trading debe estar configurado y activado en la plataforma de inversión. ✓ El Robot de Trading ejecuto un orden en el mercado(Trade).
Flujo normal:	<ol style="list-style-type: none"> 1. El Robot de Trading verifica para cada vela si hay posiciones abiertas en su evento onTick y su método PositionsTotal() si el resultado es ≥ 1 entonces existe una posición abierta y continua ejecutándose el trade hasta que cumpla condiciones para cerrarlo. 2. El Robot de Trading verifica para cada vela si hay posiciones abiertas en su evento onTick y su método PositionsTotal() si el resultado es $= 0$ entonces no existe un TRADE, continua analizando la siguiente condición.
Flujo alternativo:	<ol style="list-style-type: none"> 1. El Robot de Trading sigue Monitoreando el evento Ontick().
Pos condiciones:	El sistema actualizado el balance de capital.

Tabla 23 Especificación de Casos de Uso Posiciones abiertas.

II.1.3.1.3.1.3.1.3.4 Especificación Casos de Uso Cerrar Posiciones.

Nombre del caso de uso:	Cerrar Posiciones
Actor:	Robot de Trading
Descripción:	Especifica el flujo y estado de una orden ya sea de compra o de venta que ejecuto el robot de trading.
Precondiciones:	<ul style="list-style-type: none"> ✓ La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. ✓ El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5. ✓ El Robot de Trading debe estar configurado y activado en la plataforma de inversión. ✓ El Robot de Trading ejecuto un orden en el mercado(Trade).
Flujo normal:	<ol style="list-style-type: none"> 1. El Robot de Trading cierra automáticamente el trade cuando llega a el objetivo de beneficio de la posición colocada. 2. El Robot de Trading verifica para cada vela si la posición abierta está dentro de la zona operativa en su evento onTick y su método TimeCurrent()<ExpiracionOrdenPendiente esto nos permite saber si el tiempo actual esta dentro el rango máximo de tiempo definido para mantener abierto el trade.
Flujo alternativo:	<ol style="list-style-type: none"> 1. El Robot de Trading sigue Monitoreando el evento Ontick().
Pos condiciones:	El sistema actualizado el balance de capital.

Tabla 24: Especificación Casos de Uso Cerrar Posiciones.

II.1.3.1.3.1.3.1.3.5 Especificación Casos de Uso Generar Señales.

Nombre del caso de uso:	Generar Señales Comerciales
Actor:	Robot de Trading
Descripción:	Especifica el flujo del Robot de trading para generar las señales de orden ya sea de compra o de venta a ejecutar por el robot.
Precondiciones:	<ul style="list-style-type: none"> ✓ La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. ✓ El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5. ✓ El Robot de Trading debe estar configurado y activado en la plataforma de inversión. ✓ El Robot de Trading verifica posiciones abiertas ✓ El Robot de Trading verifica y cierra posiciones abiertas
Flujo normal:	<ol style="list-style-type: none"> 1. El Robot de Trading en su evento onTick atreves del método NuevaVela() comprueba que se cumplen todas las demás condiciones y generan una señal de compra y llama a ComprabuyStop() o venta y llama a VentasellStop(). 2. El Robot de Trading envía todos los atributos y la solicitud de orden (Trade) al Broker.
Flujo alternativo:	<ol style="list-style-type: none"> 3. El Robot de Trading sigue Monitoreando el evento Ontick().
Pos condiciones:	El sistema actualizado el balance de capital.

Tabla 25 Especificación de Casos de Uso Generar Señales.

II.1.3.1.3.1.3.1.3.6 Especificación Casos de Uso Mantener Posiciones Abiertas.

Nombre del caso de uso:	Mantener Posiciones Abiertas
Actor:	Robot de Trading
Descripción:	Especifica el flujo y estado de una orden ya sea de compra o de venta que ejecuto el robot de trading.
Precondiciones:	<ul style="list-style-type: none"> ✓ La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. ✓ El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5. ✓ El Robot de Trading debe estar configurado y activado en la plataforma de inversión. ✓ El Robot de Trading ejecuto un orden en el mercado(trade).
Flujo normal:	1. El Robot de Trading en su evento Ontick(), comprueba para cada nueva vela mediante el método PositionTotal(), si las hay salta el evento y la posición sigue abierta hasta que llega a objetivo o salga de horario operativo y no se active el trade.
Flujo alternativo:	2. El Robot de Trading sigue Monitoreando el evento Ontick().
Pos condiciones:	El sistema actualizado el balance de capital.

Tabla 26 Especificación de Casos de Uso Mantener Posiciones Abiertas

II.1.3.1.3.1.3.1.3.7 Especificación Casos de Uso Gestionar el Riesgo.

Nombre del caso de uso:	Gestionar el Riesgo
Actor:	Robot de Trading
Descripción:	Especifica el flujo y estado de una orden ya sea de compra o de venta que ejecuto el robot de trading.
Precondiciones:	<ul style="list-style-type: none"> ✓ La plataforma de inversión Metatrader 5 debe estar correctamente instalada e iniciada sesión con las credenciales de inversor en el bróker Correspondiente. ✓ El Robot de trading debe estar previamente instalado en la plataforma Metatrader 5. ✓ El Robot de Trading debe estar configurado y activado en la plataforma de inversión. ✓ El Robot de Trading ejecuto un orden en el mercado(Trade).
Flujo normal:	<ol style="list-style-type: none"> 1. El Robot de Trading en su evento Ontick(), antes en enviar la solicitud de una Orden a los métodos de compra o venta ComprabuyStop(), VentasellStop() define atributos de una orden. El límite de pérdida llamado (StopLoss), objetivo de beneficio llama (TakeProfit). 2. El Robot de Trading realiza el cálculo de riesgo Tamaño de la orden Llamado (Volumen) que se lo calcula en el método CalculoVolumen() de acuerdo al porcentaje de Riesgo definido por el Inversor .
Flujo alternativo:	<ol style="list-style-type: none"> 3. El Robot de Trading sigue Monitoreando el evento Ontick().
Pos condiciones:	El sistema actualizado el balance de capital.

II.1.3.1.3.1.3.2 Diagrama De Estados.

II.1.3.1.3.1.3.2.1 Diagrama De Estado Base del Robot de Trading

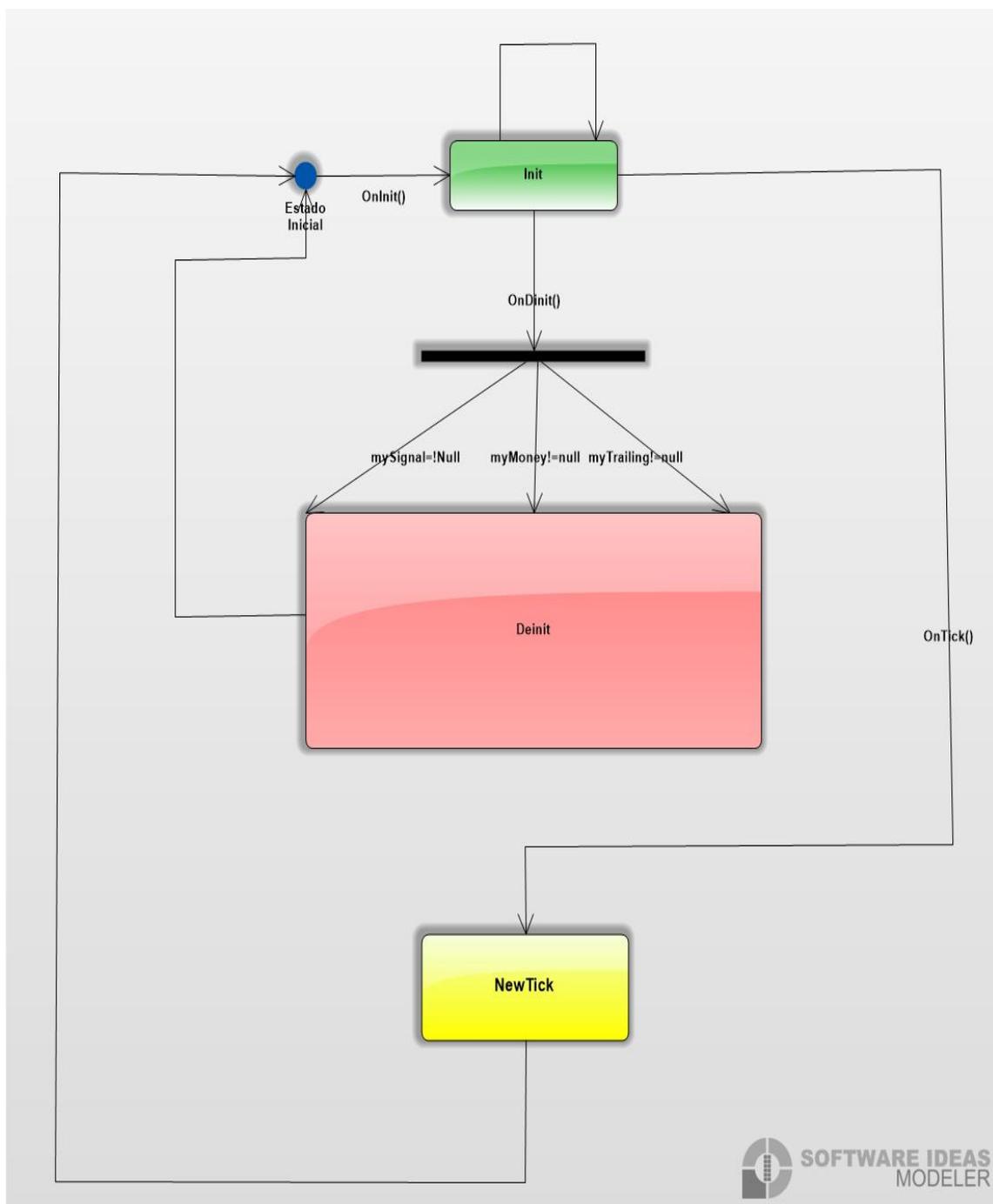


Tabla 26 Diagrama de Estado Base

II.1.3.1.3.1.3.2.2 Diagrama De Estado Base del Robot de Trading Iniciado

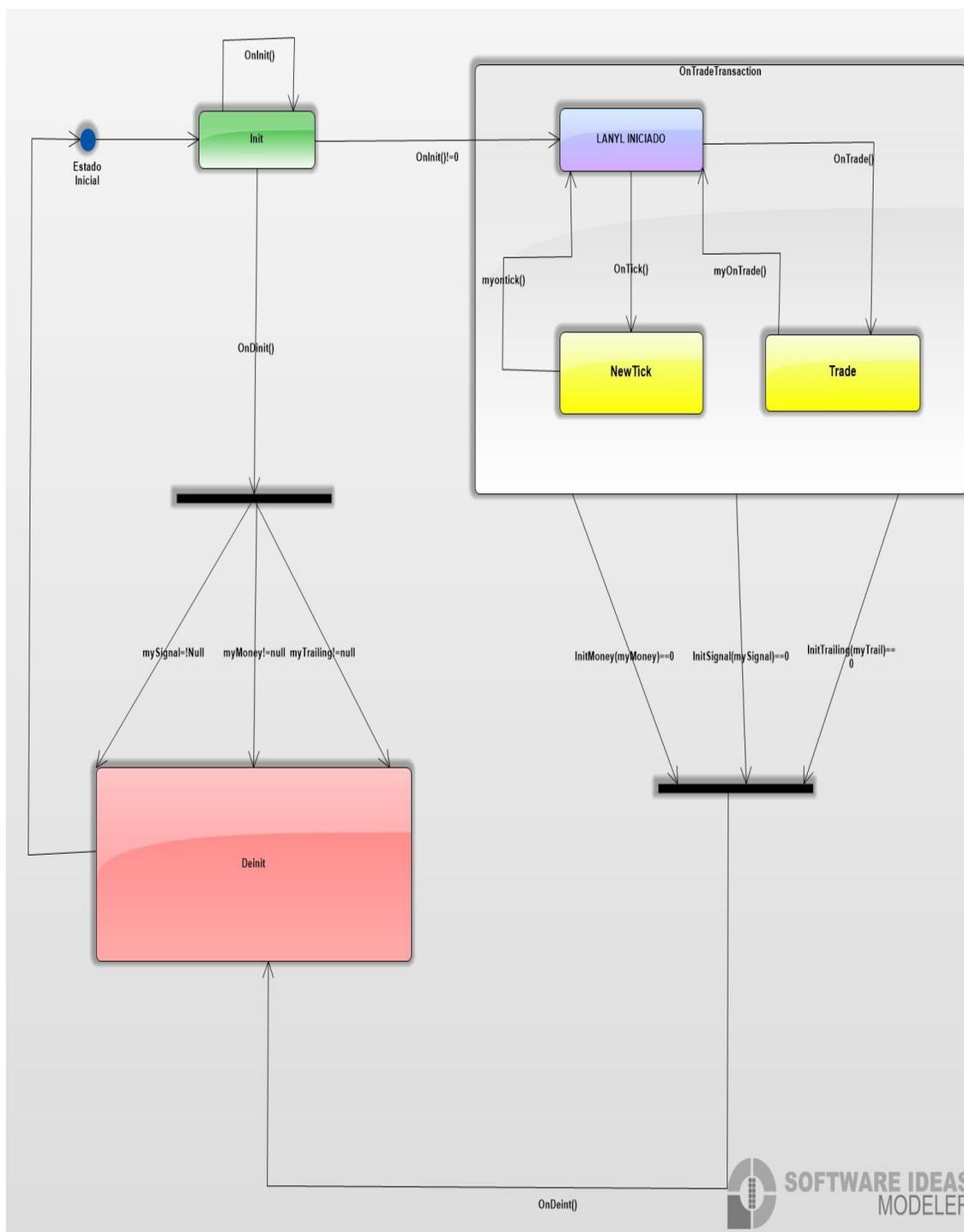


Tabla 27 Diagrama de Estado Robot de Trading Iniciado

II.1.3.1.3.1.3.3 Diagrama De Actividades

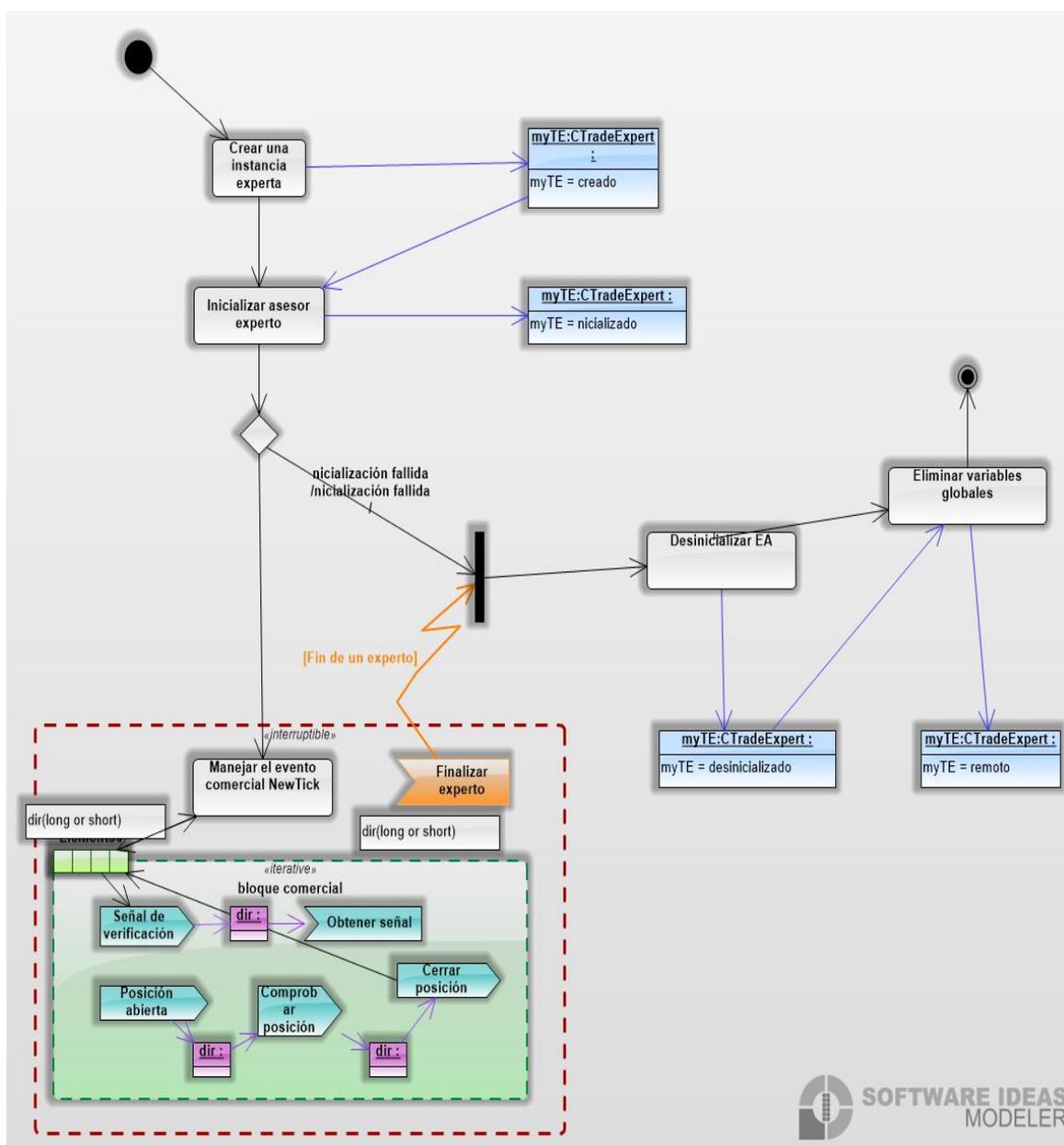


FIGURA 35:Diagrama De Actividades

II.1.3.1.3.1.3.4 Diagrama De Secuencia

II.1.3.1.3.1.3.4.1 Diagrama De Secuencia General del Robot de Trading

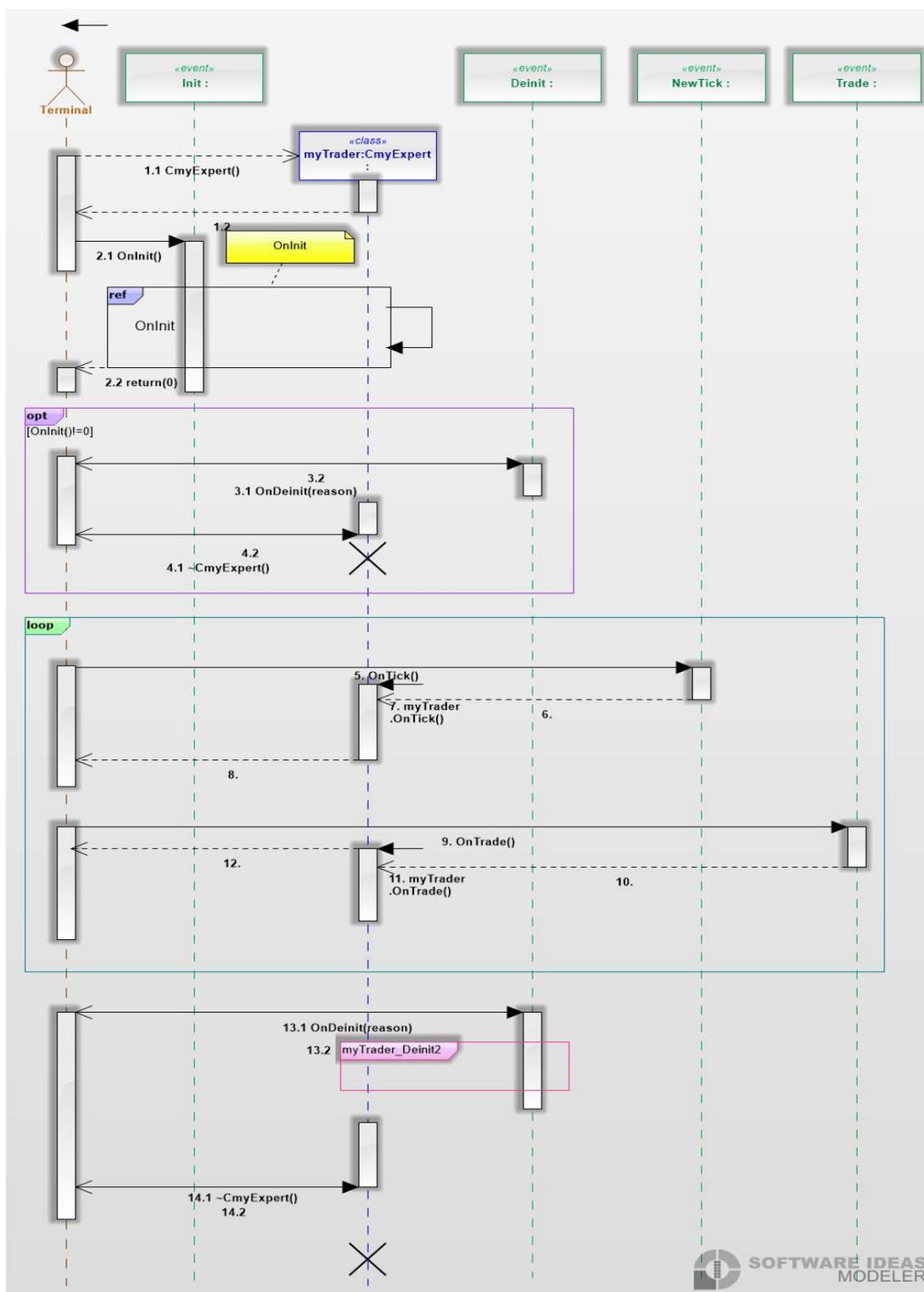


FIGURA 36:Diagrama De Secuencia

II.1.3.1.3.1.3.4.2 Diagrama De Secuencia del Robot de Trading Denegado

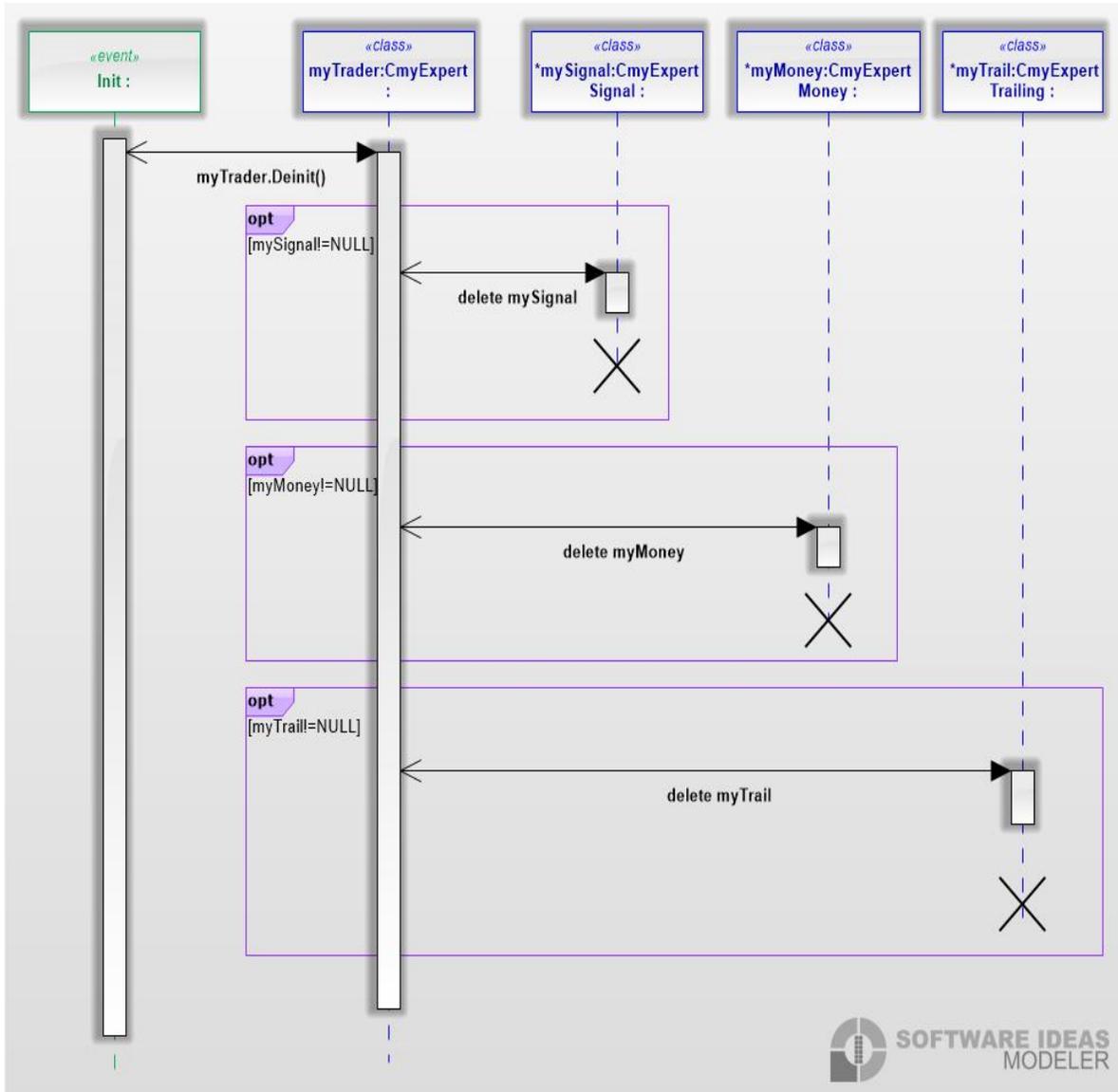


FIGURA 37:Diagrama De Secuencia Denegado

II.1.3.1.3.1.3.5 Diagramas de Estructura

II.1.3.1.3.1.3.5.1 Flujograma De Datos de las Divisas

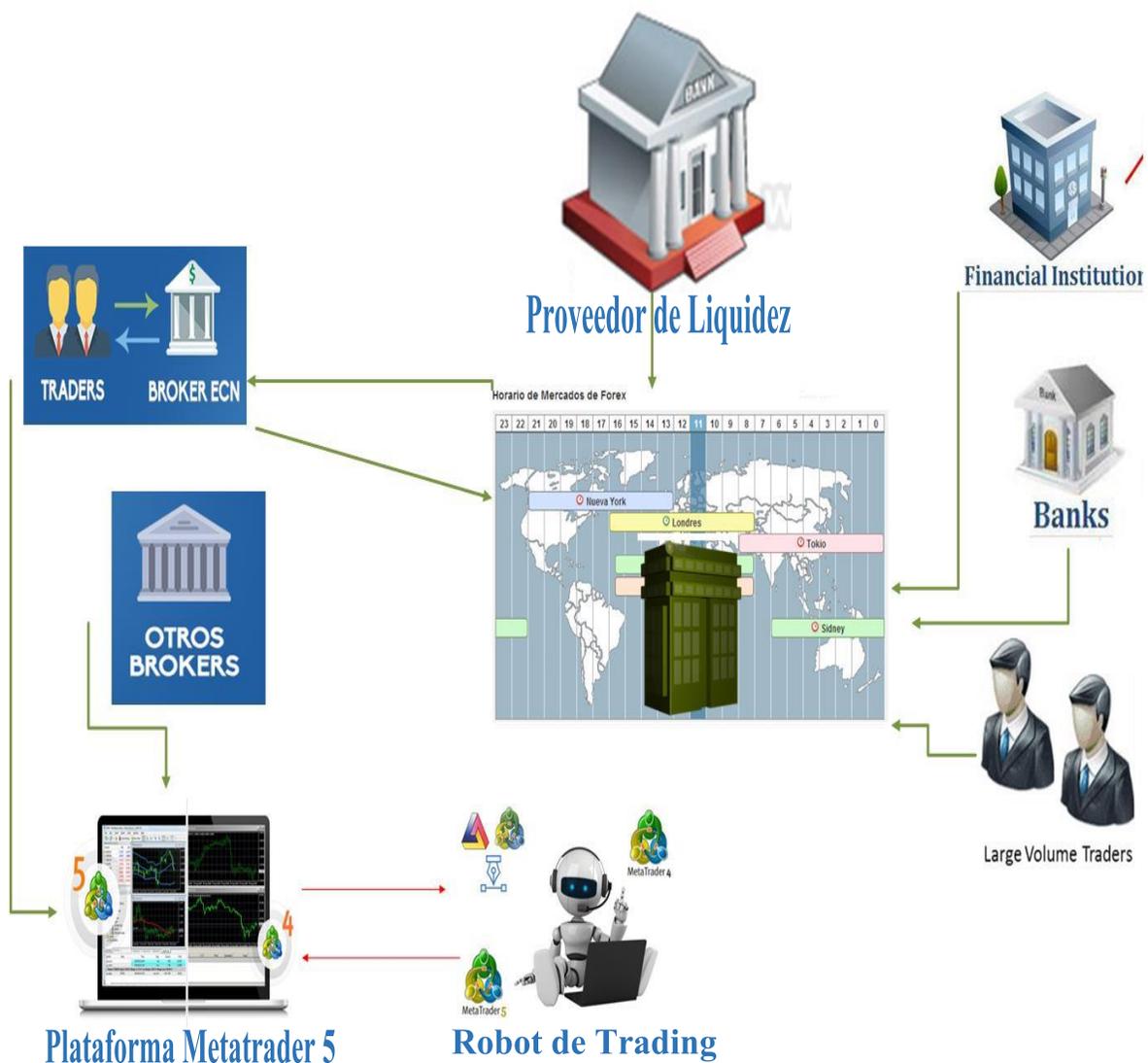


FIGURA 38:Flujograma De Datos de las Divisas

II.1.3.1.3.1.3.5.2 Flujograma De Datos Históricos de las Divisas

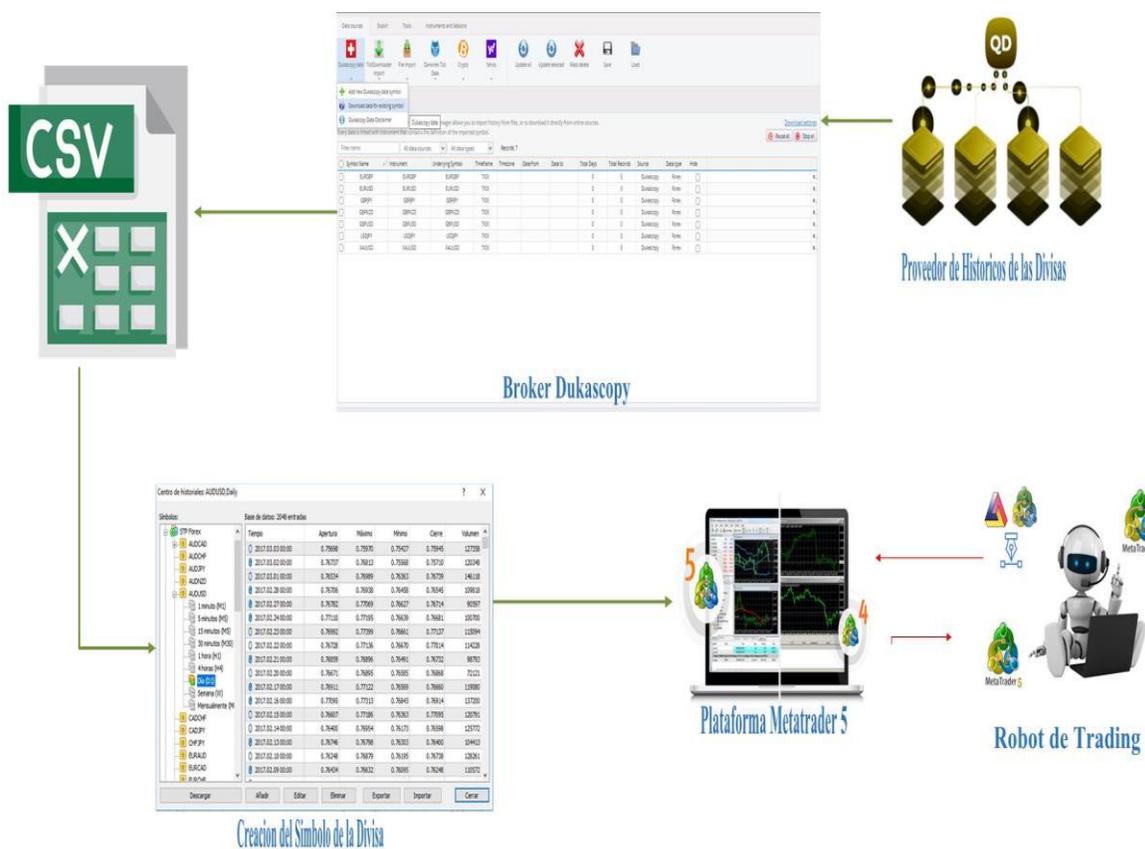


FIGURA 39:Flujograma De Datos Históricos de las Divisas

II.1.3.1.3.1.3.6 Diagrama de Clases

II.1.3.1.3.1.3.6.1 Diagrama de Clases General de Sistemas de Inversión

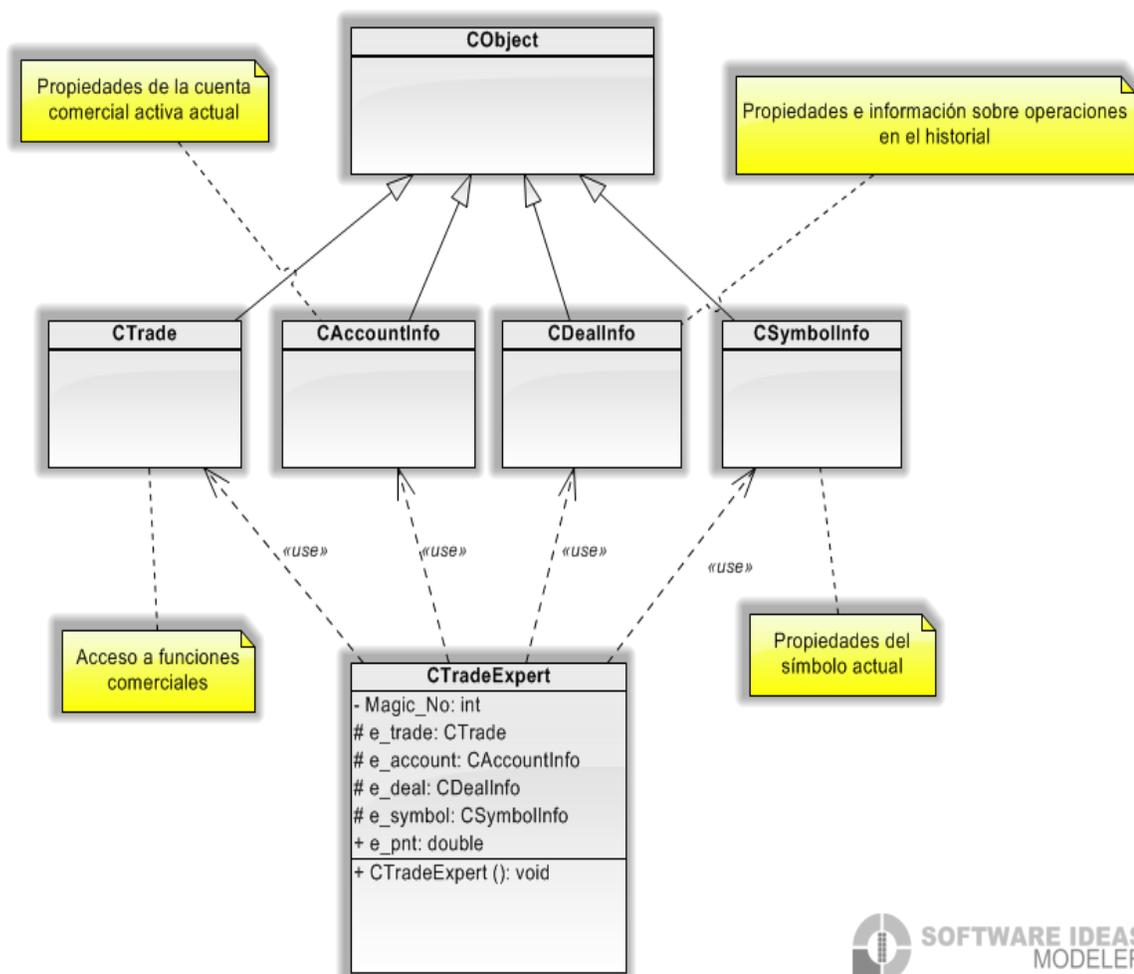


FIGURA 40:Diagrama de Clases General de Sistemas de Inversión

II.1.3.1.3.1.3.6.2 Diagrama Estructura de Clases para Sistemas expertos de

Inversión

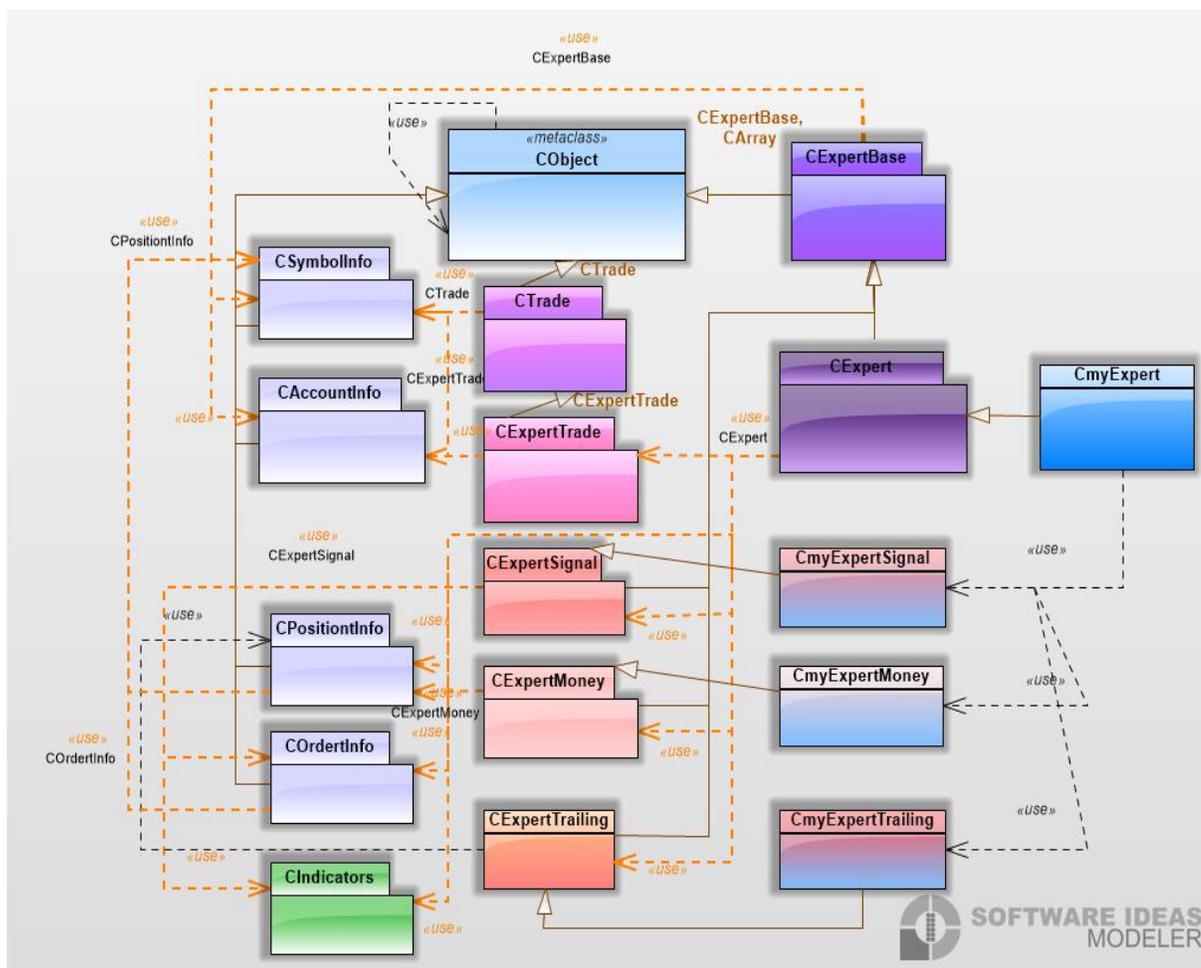


FIGURA 41:Diagrama Estructura de Clases para Sistemas expertos de Inversión

II.1.3.1.3.2 Sprint 2

II.1.3.1.3.2.1 Descripción. -

El Segundo Sprint Comprende el desarrollo del sistema experto que será la estructura del robot.

II.1.3.1.3.2.2 Propósito. -

El propósito del Segundo Sprint es la automatización de la información recabada con funcionamiento de identificación de zonas horarias operativas y tendencias del mercado.

II.1.3.1.3.2.3 Diagramas de Comportamiento. -

II.1.3.1.3.2.3.2 Diagramas De Estructura

II.1.3.1.3.2.3.2.1 Diagramas Clases de Clase Trade

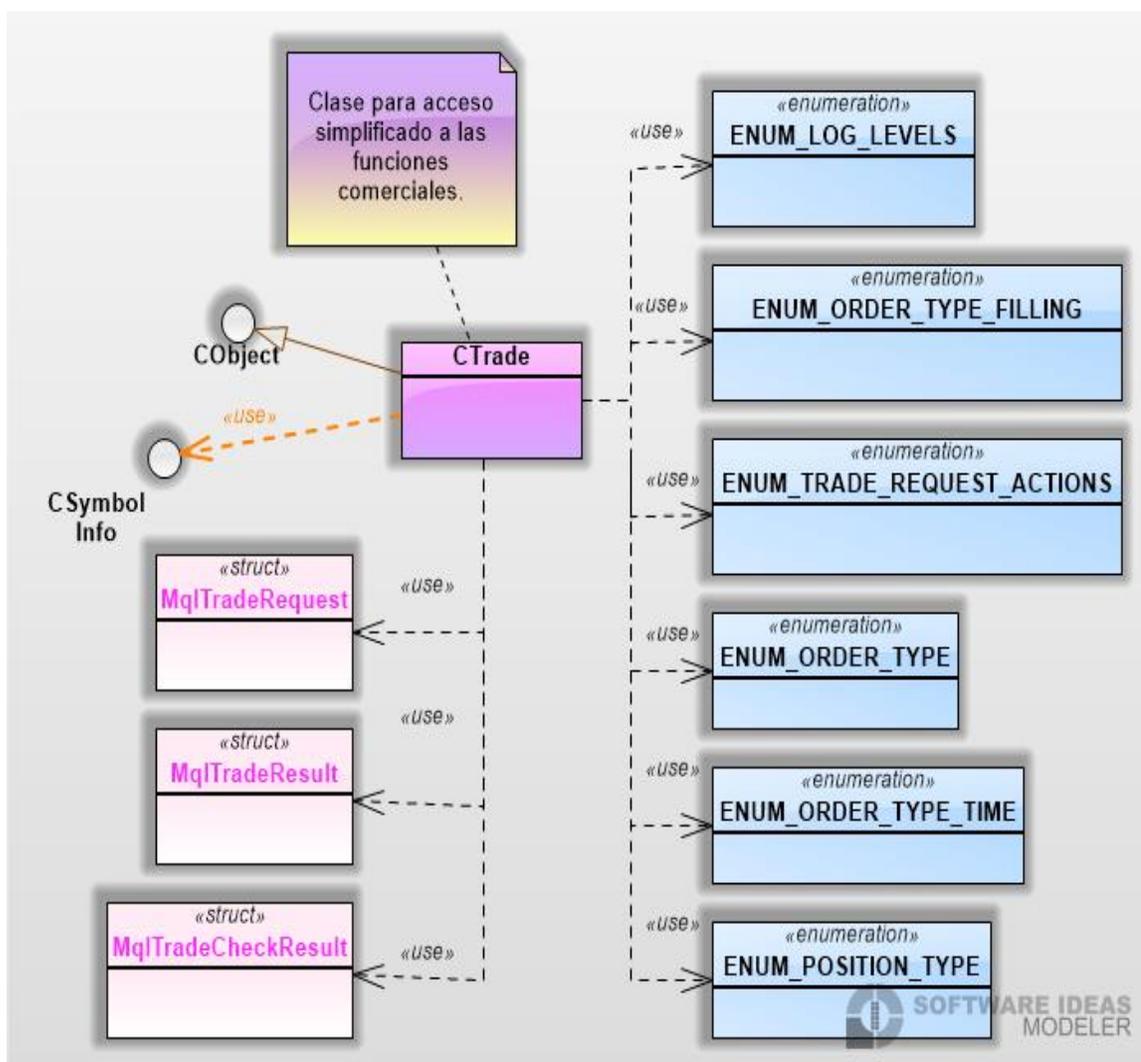


FIGURA 43:Diagramas Clases de Clase Trade

II.1.3.1.3.2.3.2.2 Diagramas Clases de Clase ExperTrade

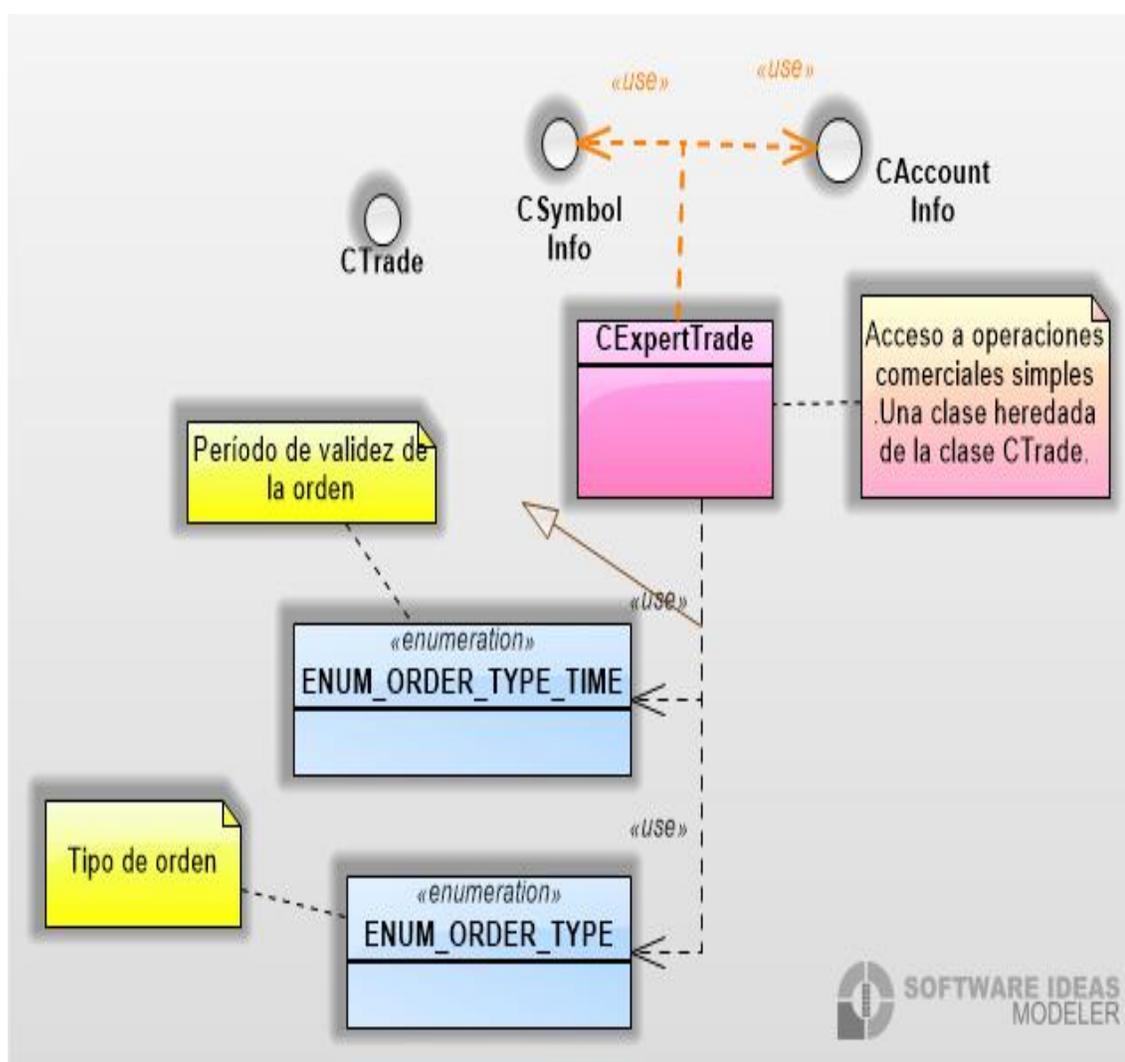


FIGURA 44:Diagramas Clases de Clase ExperTrade

II.1.3.1.3.2.3.2.3 Diagramas Clases de Clase COrderinfo

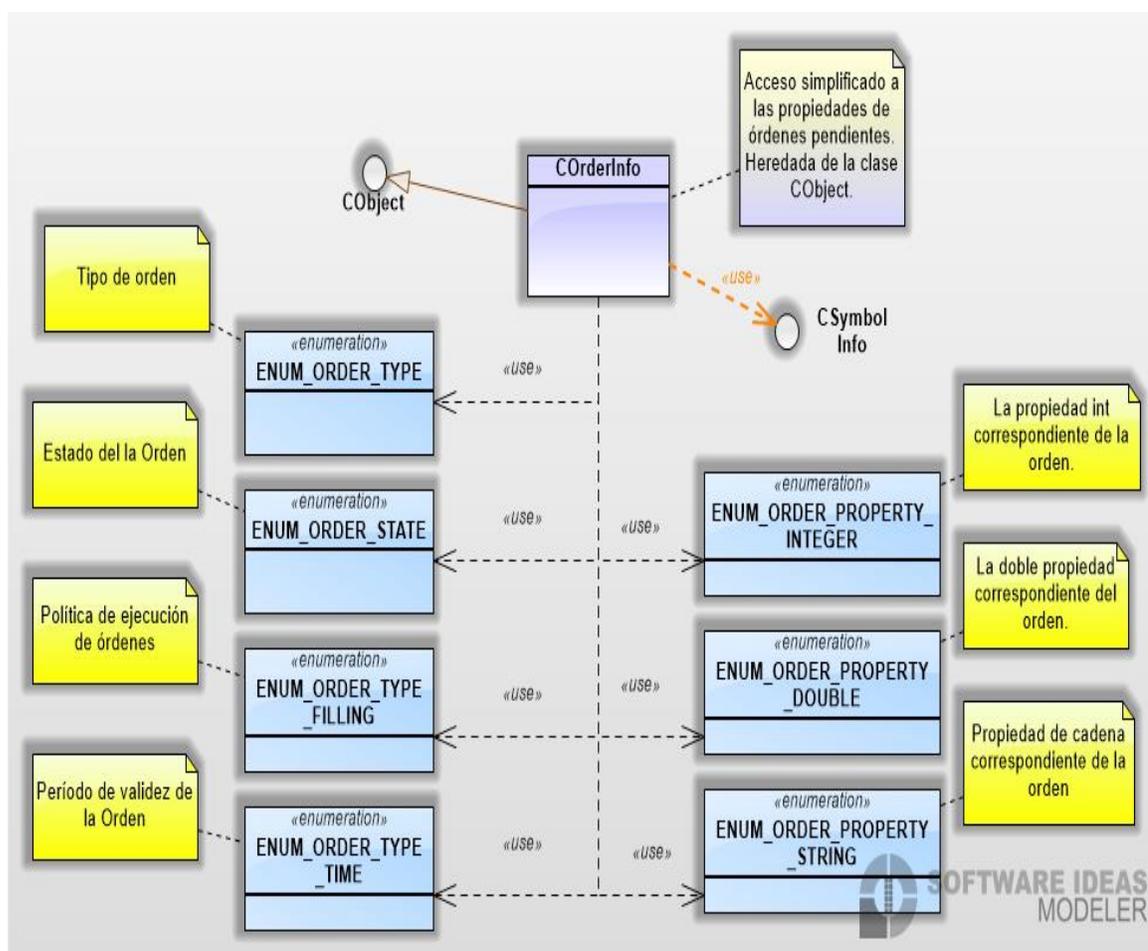


FIGURA 45:Diagramas Clases de Clase COrderinfo

II.1.3.1.3.2.3.2.4 Diagramas Clases de Clase CPositionInfo

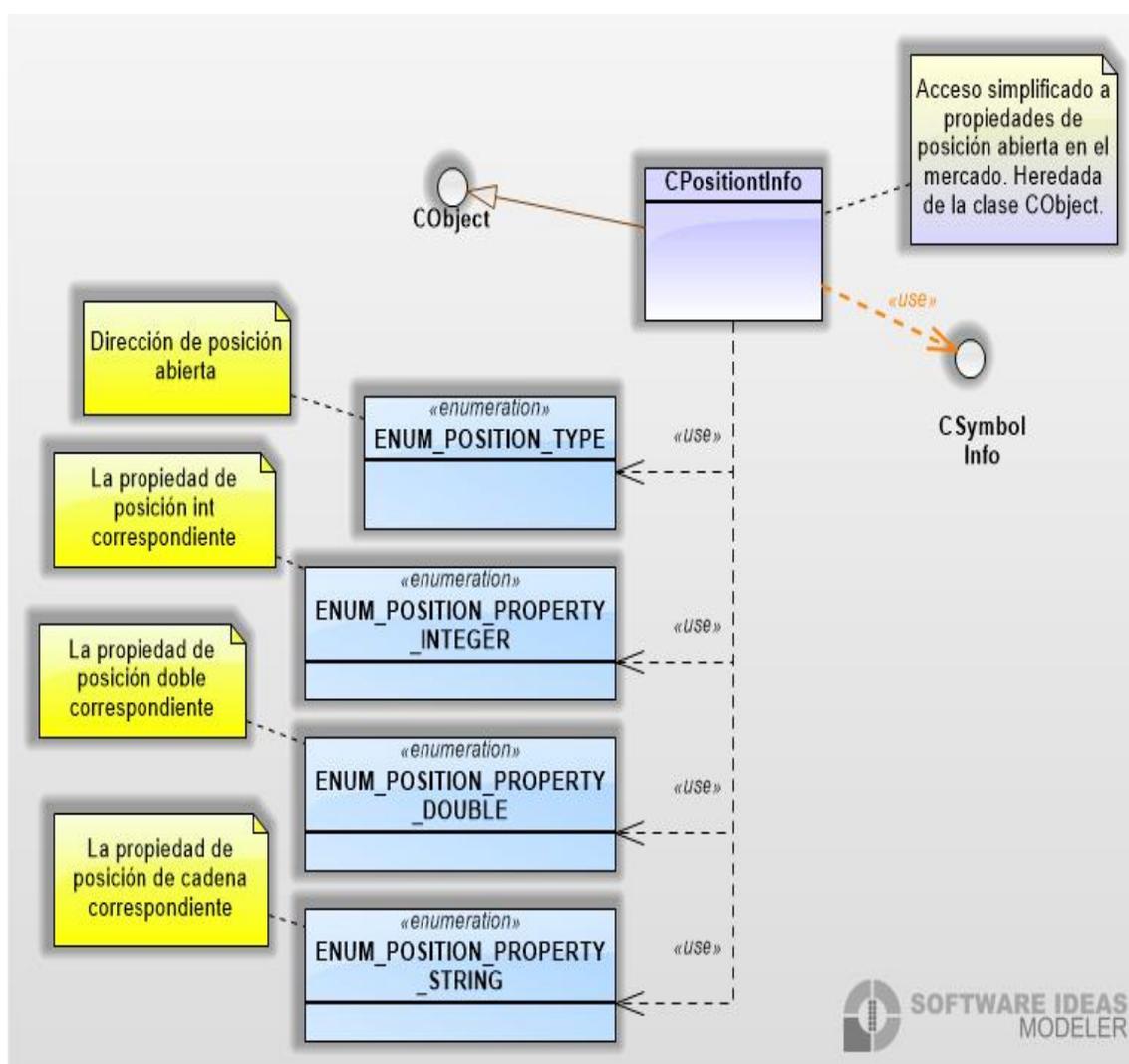


FIGURA 46:Diagramas Clases de Clase CPositionInfo

II.1.3.1.3.3 Sprint 3

II.1.3.1.3.3.1 Descripción. -

El Tercer Sprint Comprende Implementación de la Estrategia de trading al sistema experto.

II.1.3.1.3.3.2 Propósito. -

El propósito del Tercer Sprint es automatizar las operaciones en el Mercado de divisas.

II.1.3.1.3.3.3 Diagrama de Componentes.

II.1.3.1.3.3.3.1 Estrategia de Algoritmo KillzoneLondrez.

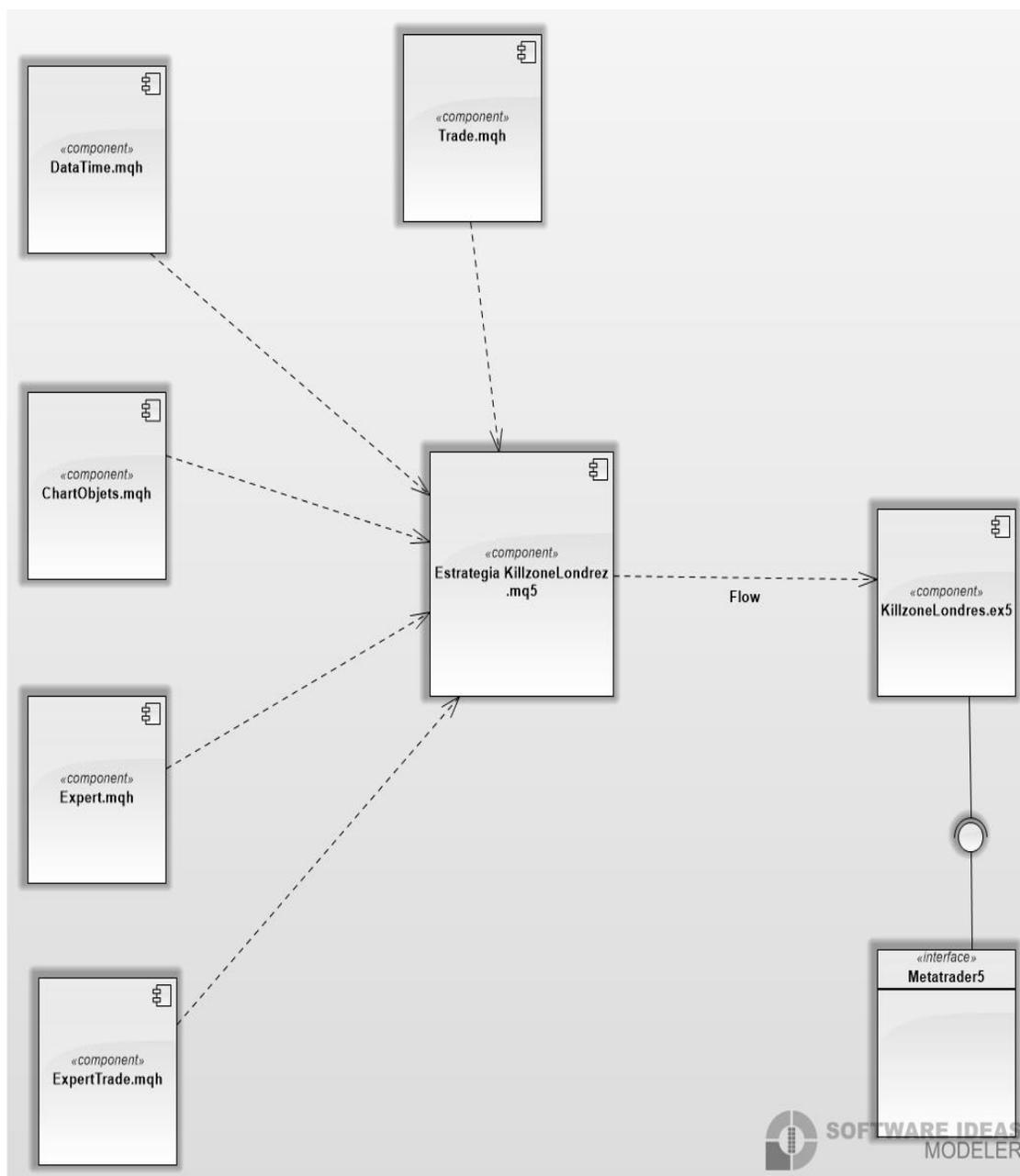


FIGURA 47:Estrategia de Algoritmo KillzoneLondrez.

II.1.3.1.3.3.4 Programación de la estrategia

Una vez se tiene descrita la lógica base de la estrategia se pasa a su codificación en el lenguaje de la plataforma de *trading* para poder ser ejecutada y probada.

The screenshot shows the MetaEditor interface with the following code in the main editor:

```

28 //DECLARACIÓN DE LAS VARIABLES GLOBALES
29 //+-----+
30
31 //Variables globales para la gestión de los datos históricos
32 datetime TiemposVelas[]; //Array que almacena los tiempos de comienzo de las velas
33 double CierreVelas[]; //Array que almacena los precios de cierre de las velas
34 double MaximoVelas[]; //Array que almacena los precios máximos de las velas
35 double MinimoVelas[]; //Array que almacena los precios mínimos de las velas
36 double MaximoVelasActual[]; //Array que almacena el precio máximo de la vela actual
37 double MinimoVelasActual[]; //Array que almacena el precio mínimo de la vela actual
38
39 //Variables globales para la gestión del tiempo
40 MqlDateTime HorarioComienzoReferencia; //Almacena los tiempos de comienzo de cada sesión
41 MqlDateTime HorarioFinReferencia; //Almacena los tiempos de fin de cada sesión así
42 datetime HorarioComienzoReferenciaDT; //Almacena los tiempos de comienzo de cada sesión
43 datetime HorarioFinReferenciaDT; //Almacena los tiempos de fin de cada sesión así
44 MqlDateTime HorarioActual; //Almacena el tiempo actual en formato MqlDateTim
45 bool Control = false; //Variable utilizada para controlar la ejecución
46 datetime TiempoUltimaActualizacion = 0; //Almacena el tiempo de comienzo de la vela actua
47
48 //Variables para dibujar la sesión asiática
49 const string Rectangulo = "Rectangulo"; //Constante que se utilizada como nombre base de
50 int Contador = 0; //Variable contadora que se añade al nombre base
51 string ObjetoRectangulo = NULL; //Almacena el nombre completo de cada objeto rect.
52
53 //Estructura con información de la sesión asiática y europea
54 struct BloqueReferencia //Creación del tipo BloqueReferencia
55 {
56 double PrecioMasAlto; //Almacena el precio más alto de la sesión asiáti

```

A red arrow points from the text "Definición de Variables Globales" to the variable declarations in the code. The text "Definición de Variables Globales" is written in a large, bold font over the code.

The interface also shows a file explorer on the left with a tree view of the project structure, including folders like Experts, Advisors, Examples, and Files. The bottom status bar shows "0 errors, 0 warnings, 1466 msec elapsed".

FIGURA 48:Definición de Variables del Algoritmo

```

203 //+-----+
204 bool NuevaVela(string pSimbolo,ENUM_TIMEFRAMES pPeriodo)
205 {
206     bool nuevaVela = false; //Almacena si ha comenzado una nueva
207     datetime velaDateTime[]; //Array que almacena el tiempo de co
208     AlmacenaTiemposComienzo(_Symbol,pPeriodo,0,1,velaDateTime); //Obtiene el tiempo de comienzo de l
209     if (TiempoUltimaActualizacion != velaDateTime[0]) //Si el tiempo de comienzo de la vel
210     {
211         TiempoUltimaActualizacion = velaDateTime[0]; //Actualiza el tiempo de la última ac
212         nuevaVela = true; //Ha comenzado una nueva vela
213     }
214     return(nuevaVela);
215 }
216
217 // FUNCIÓN QUE CALCULA EL VOLUMEN DE LA FUTURA OPERACIÓN BASADO EN UN PORCENTAJE DE PÉRDIDA Y EN UN STOP
218 //+-----+
219 double CalculoVolumen(double pPorcentaje,double pStopLoss)
220 {
221     double cantidadArriesgar; //Almacena la cant
222     double precioPuntoArriesgar; //Almacena el prec
223     double precioPunto; //Almacena el prec
224     double volumen; //Almacena el volu
225     if(pPorcentaje > 0 && pStopLoss > 0) //El porcentaje de
226     {
227         cantidadArriesgar = AccountInfoDouble(ACCOUNT_BALANCE) * (pPorcentaje / 100); //Calcula el diner
228         precioPuntoArriesgar = (cantidadArriesgar / pStopLoss); //Precio del punto
229         precioPunto = SymbolInfoDouble(_Symbol,SYMBOL_TRADE_TICK_VALUE); //Es el precio del
230         volumen = (precioPuntoArriesgar / precioPunto); //Volumen a aplica
231         volumen = VerificarVolumen(volumen); //Verifica si el v

```

Metodos Para calculo de porcentaje de riesgo y nueva vela

FIGURA 49:Definición de Control de Parámetros Tiempos y Precio en el Algoritmo

MetaEditor - [SinPausaLondrez.mq5]

Archivo Editar Buscar Ver Compilación Depurar Herramientas Ventana Ayuda

Navegador

266 // FUNCIÓN QUE DEVUELVE EL BAJO MÁS BAJO DE UN CONJUNTO DE VELAS
 267 //+-----+
 268 double MasBajo(string pSímbolo, ENUM_TIMEFRAMES pPeriodo,int pPosicionInicio,int pCantidadVelas)
 269 {
 270 double bajos[]; //Array dinámico que almacena los valores más bajo
 271 int indiceMinimo; //Almacena el índice del array donde se encuentra
 272 double masBajo; //Almacena el bajo más bajo
 273 ArraySetAsSeries(bajos,true); //Cambia el array a una serie temporal
 274 CopyLow(pSímbolo,pPeriodo,pPosicionInicio,pCantidadVelas,bajos); //Copia los bajos de un conjunto de velas
 275 indiceMinimo = ArrayMinimum(bajos); //Índice de la posición con el bajo más bajo
 276 masBajo = bajos[indiceMinimo]; //Almacena el bajo más bajo y lo devuelve a la ll
 277 return(masBajo);
 278 }
 279
 280 // FUNCIÓN QUE DEVUELVE EL ALTO MÁS ALTO DE UN CONJUNTO DE VELAS
 281 //+-----+
 282 double MasAlto(string pSímbolo, ENUM_TIMEFRAMES pPeriodo,int pPosicionInicio,int pCantidadVelas)
 283 {
 284 double altos[]; //Array dinámico que almacena los valores más alto
 285 int indiceMaximo; //Almacena el índice del array donde se encuentra
 286 double masAlto; //Almacena el alto más alto
 287 ArraySetAsSeries(alto,true); //Cambia el array a una serie temporal
 288 CopyHigh(pSímbolo,pPeriodo,pPosicionInicio,pCantidadVelas,altos); //Copia los máximos de un conjunto de velas
 289 indiceMaximo = ArrayMaximum(alto); //Índice de la posición con el alto más alto
 290 masAlto = altos[indiceMaximo]; //Almacena el alto más alto y lo devuelve a la ll
 291 return(masAlto);
 292 }
 293
 294

Metodos Para cálculo de maximos y minimos de un rango

Descripción Archivo

Errros Buscar Artículos Code Base Proyectos públicos Diario

FIGURA 50: Precio más Alto y más Bajo en un Periodo

MetaEditor - [SinPausaLondrez.mq5]

Archivo Editar Buscar Ver Compilación Depurar Herramientas Ventana Ayuda

Navegador

365
 366 // FUNCIÓN QUE REALIZA EL EMPLAZAMIENTO DE UNA ORDEN DE COMPRA BUY STOP
 367 //+-----+
 368 bool CompraBuyStop(string pSímbolo,ulong &pTicket,double pVolumen,uint &pCodigo,double pPrecio,double pStop = 0, double pP
 369 ulong pNumeroMagico=1,ENUM_ORDER_TYPE_FILLING pRelleno = ORDER_FILLING_FOK,ENUM_ORDER_TYPE_TIME pExpi
 370 datetime pTiempoExpiracion = 0,string pComentario = NULL)
 371 {
 372 ZeroMemory(Solicitud); //Pone los valores del objeto Solicitud a 0
 373 ZeroMemory(Respuesta); //Pone los valores del objeto Respuesta a 0
 374
 375 Solicitud.action = TRADE_ACTION_PENDING; //La solicitud es para emplazar una orden pendiente
 376 Solicitud.type = ORDER_TYPE_BUY_STOP; //La solicitud es para el orden de compra Buy Stop
 377 Solicitud.symbol = pSímbolo; //La solicitud es para el símbolo pasado por parámetro
 378 Solicitud.sl = pStop; //Stop loss en precio
 379 Solicitud.tp = pProfit; //Take profit en precio
 380 Solicitud.comment = pComentario; //Comentario
 381 Solicitud.volume = pVolumen; //Volumen de la orden
 382 Solicitud.deviation = pDesviacion; //Desviación de la orden
 383 Solicitud.type_filling = pRelleno; //Tipo de política de relleno del volumen de la orden
 384 Solicitud.magic = pNumeroMagico; //Número mágico del Expert Advisor
 385 Solicitud.price = pPrecio; //El precio Stop de la orden pendiente
 386 Solicitud.type_time = pExpiracion; //El tipo de expiración de la orden pendiente
 387 Solicitud.expiration = pTiempoExpiracion; //Es la fecha de expiración de la orden pendiente si el tipo de
 388
 389 EnvioSolicitud = OrderSend(Solicitud,Respuesta); //Envío de la orden al servidor y almacena el resultado del emp
 390 pTicket = Respuesta.order; //Se almacena en el parámetro pasado por referencia pTicket, el
 391 pCodigo = Respuesta.retcode; //Se almacena en el parámetro pasado por referencia pCodigo, el
 392 return(EnvioSolicitud); //Devuelve si la la orden ha sido emplazada o no
 393 }
 394

Metodos Para colocar ordenes de compra y venta

Descripción Archivo

Errros Buscar Artículos Code Base Proyectos públicos Diario

Para abrir la Ayuda, pulse F1

FIGURA 51:Función para colocar una orden pendiente de compra

a) Estructura del código de la estrategia

No todas las plataformas utilizan el mismo lenguaje de programación, por lo que se necesitará conocer cuál es el de la plataforma que se va a utilizar. Además, algunas plataformas de trading como **Metatrader5** ofrecen sus propias herramientas y lenguaje de programación para la construcción de estrategias y programar todo el código.

En **Metatrader5** esta herramienta se llama “*Metaeditor MetaQuotes*” y permite construir sencillas estrategias con una base muy pequeña de programación hasta grandes sistemas de Inversión.

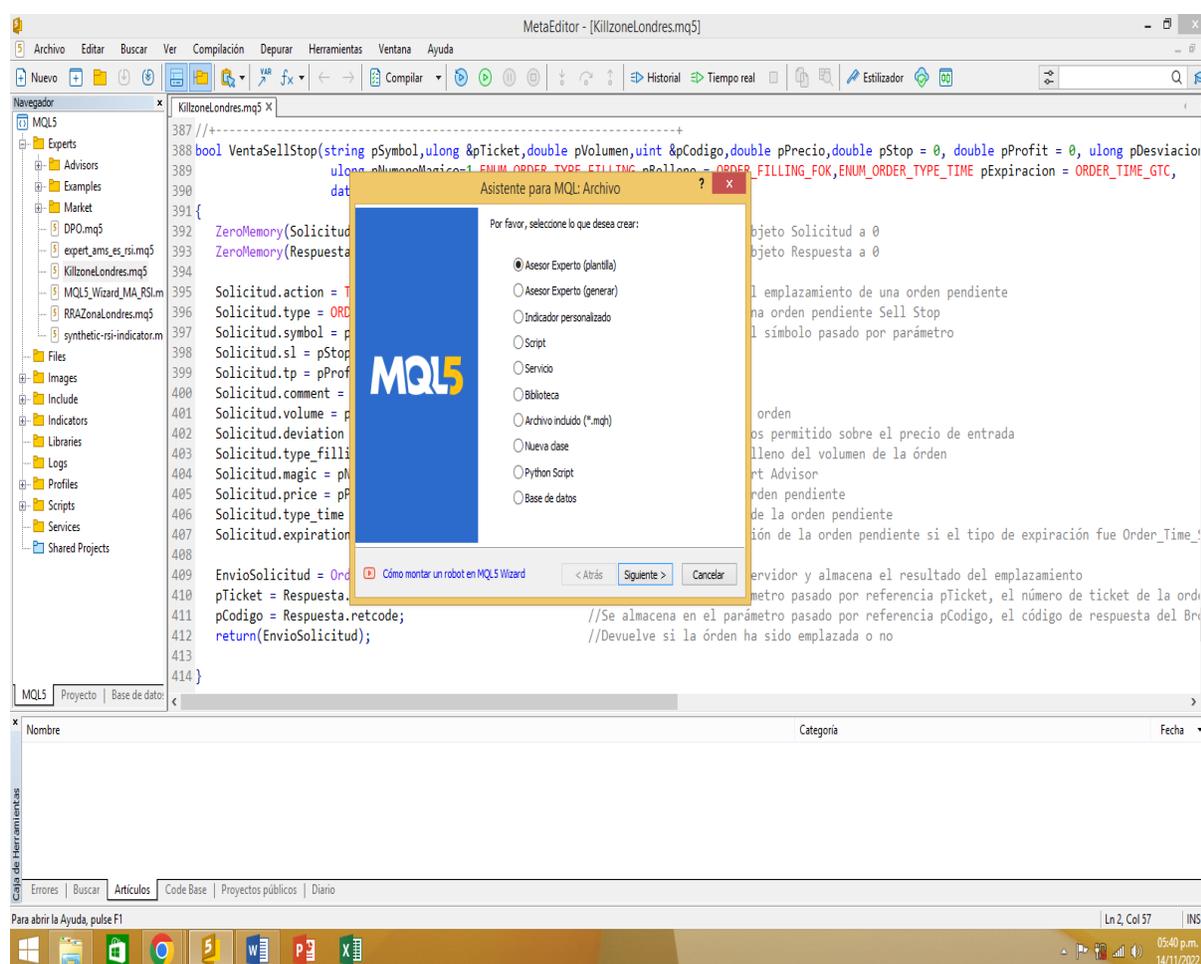


FIGURA 52:Herramienta “MetaQuotex” de la plataforma Metatrader5

II.1.3.1.3.3.5 Pantallas de la Plataforma de Inversión.

II.1.3.1.3.3.5.1 Pantalla 1 de Estructura de Archivos de la

Plataforma

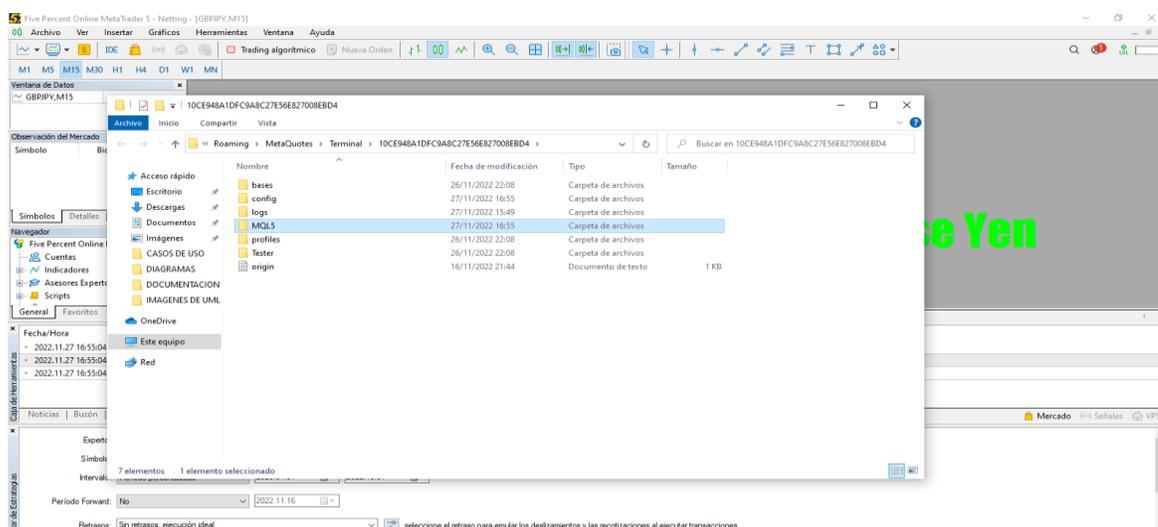


FIGURA 53:Pantalla 2 de Estructura de Archivos de la Plataforma.

II.1.3.1.3.3.5.2 Pantalla 3 Estructura de Archivos para Agregar los Sistemas

Expertos a la Plataforma.

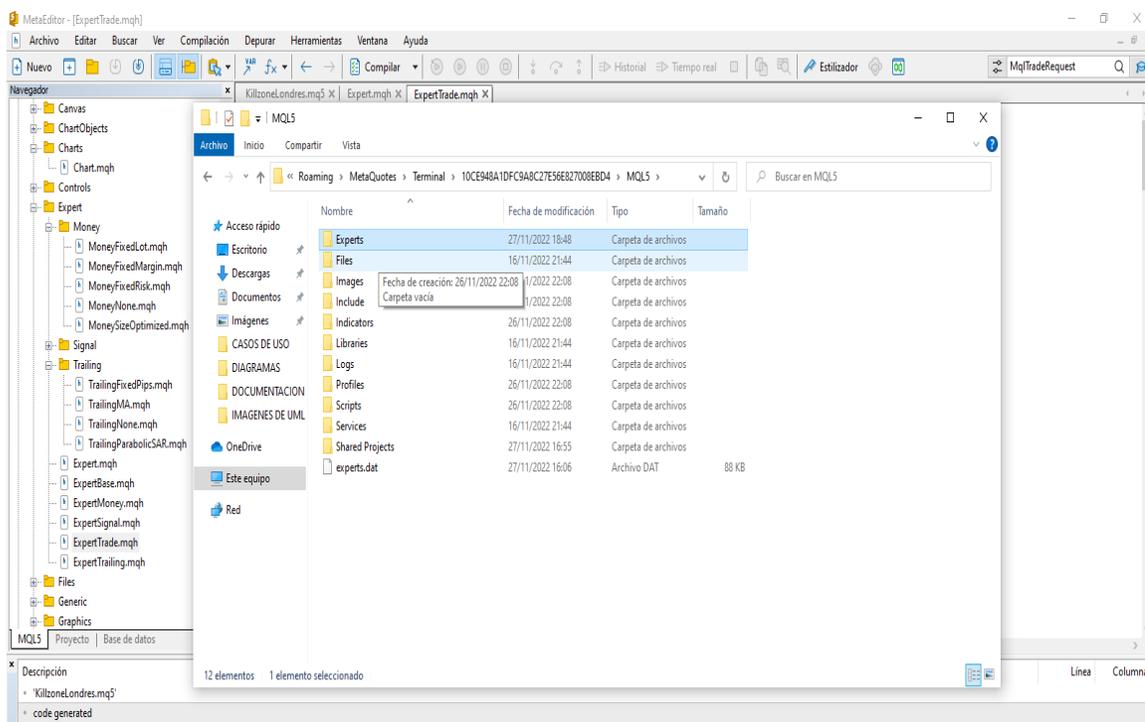


FIGURA 54: Pantalla 3 Estructura de Archivos para Agregar los Sistemas Expertos a la Plataforma.

II.1.3.1.3.3.5.3 Pantalla 3.1 Estructura de Archivos para Agregar los Sistemas

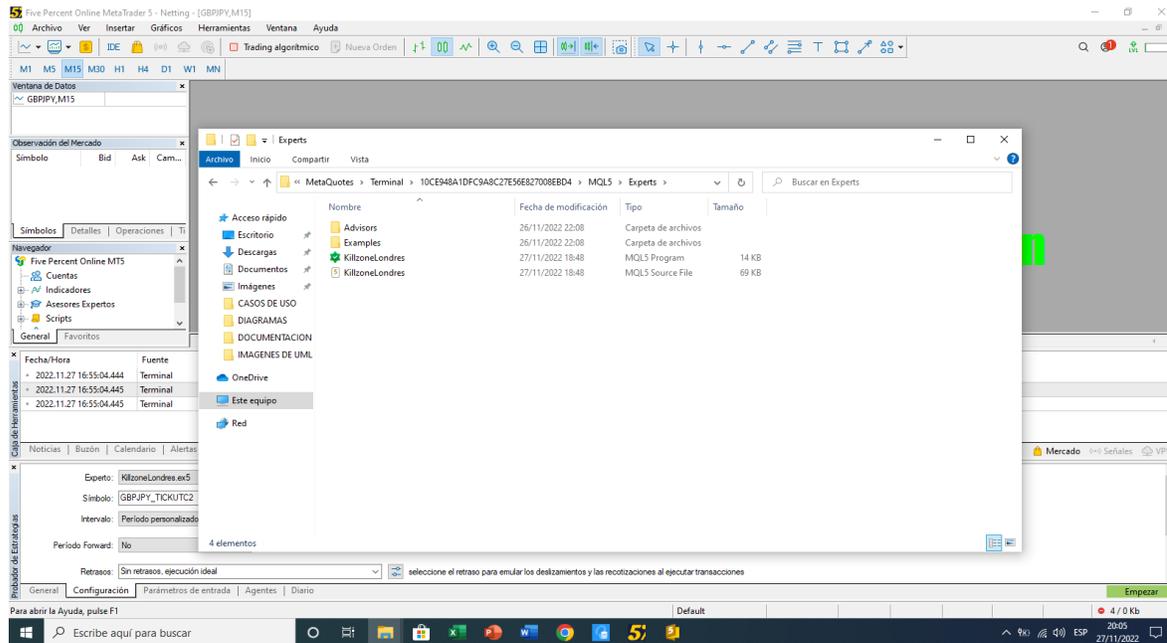


FIGURA 55:Pantalla 3.1 Estructura de Archivos para Agregar los Sistemas

Expertos a la Plataforma

II.1.3.1.3.3.5.4 Pantalla 4 Activación de Trading algorítmico



FIGURA 56:Pantalla 4 Activación de Trading algorítmico

II.1.3.1.3.3.5.5 Pantalla 5 Activación Probador de Estrategias



FIGURA 57:Pantalla 5 Activación Probador de Estrategias

II.1.3.1.3.3.5.6 Pantalla 6 Configuración de Parámetros Iniciales



FIGURA 58:Pantalla de configuracion de Parametros iniciales

II.1.3.2 Evaluación

II.1.3.2.1 Backtest Se realiza un primer backtest sobre el periodo IS para comprobar que las operaciones abren y cierran según el código programado y que efectivamente la lógica ofrezca un buen comportamiento. En este caso para el par GBP/USD, arriba, indicando los parámetros de prueba y el resumen de la simulación

Periodo IS: 01/01/2020 – 29/10/2020

II.1.3.2.1.1 Pantalla Ejecución del Backtest con el Robot



FIGURA 59: Pantalla Ejecución del Backtest con el Robot

II.1.3.2.1.2 grafica de curva de Equidad del Backtest

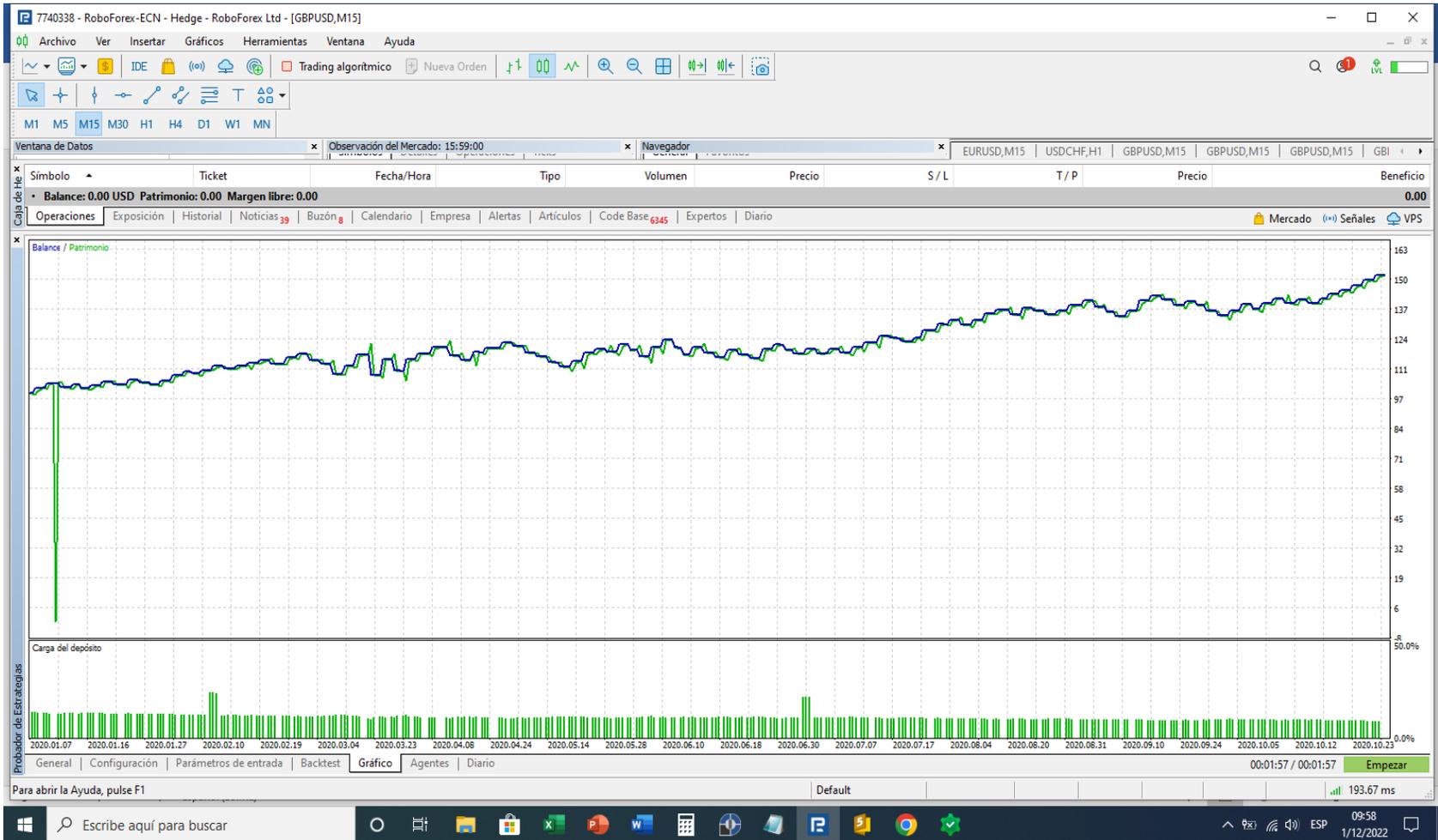


FIGURA 60: Grafica de Equidad del Backtest

II.1.3.2.1.3 Pantalla de resultados del backtesting

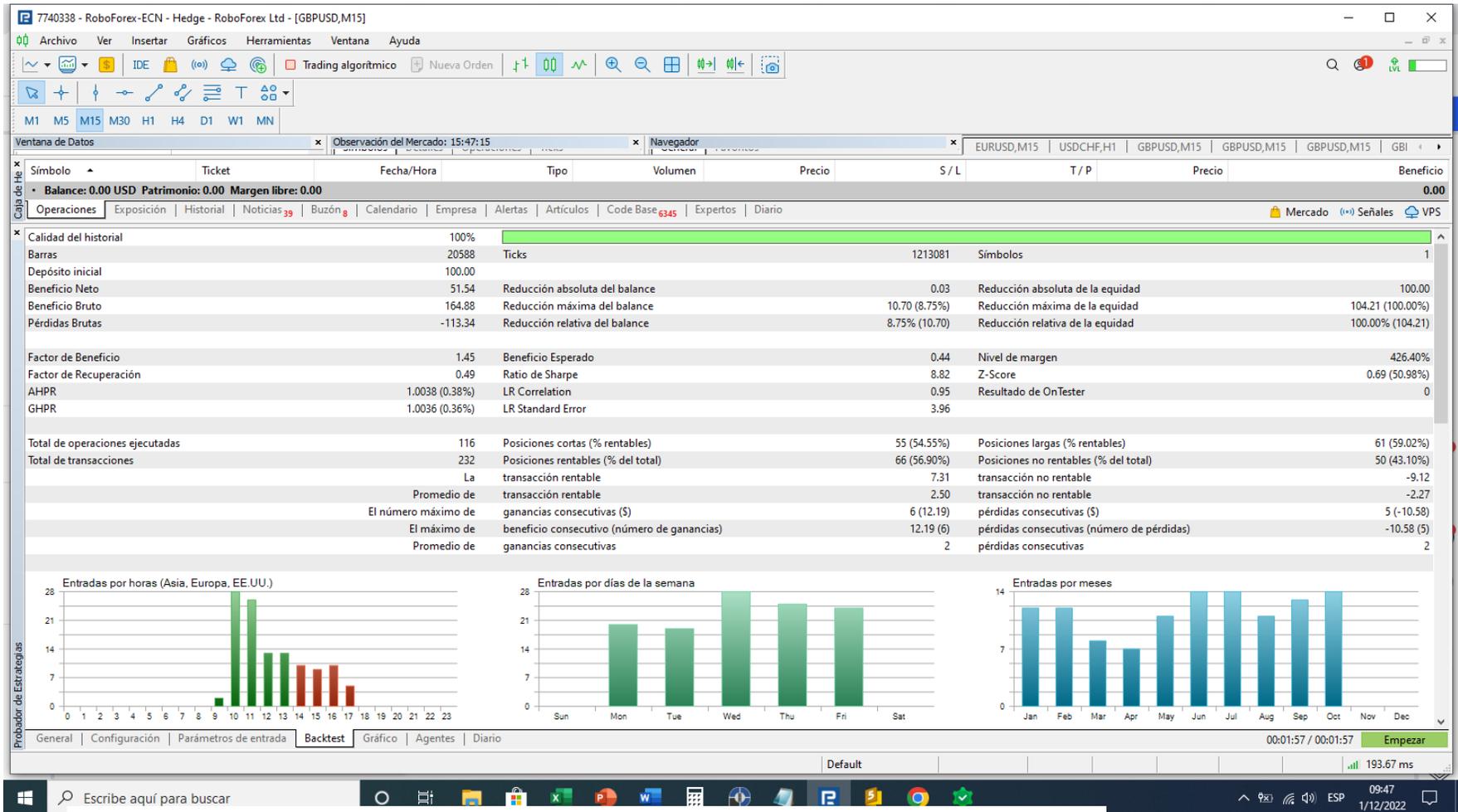


FIGURA 62: Pantalla de resultados del backtesting

II.1.3.2.1.4 Interpretación de los resultados del Backtest

Métricas	Valor	Observación
Beneficio Neto Total	51.54 \$	Aceptable
Profit factor	1,45	>1 se considera rentable
Número de trades	232	Número propio de trading Intradía
Beneficio Medio por Operación	0,22	Aceptable ya que, aunque se le reduzcan las comisiones sigue siendo positivo
Máximo Drawdown	13,34\$ \$	25,88% del beneficio neto

Tabla 28: Interpretación de los resultados del Backtest

Esperanza matemática

Una métrica tremendamente importante es la **esperanza matemática** (*Mathematical Expectation*), supone otra forma de evaluar el rendimiento mismos datos que son obtenidos con los valores de las variables que te ofrece el backtesting.

$E[X] = (\% \text{ de operaciones ganadoras} * \text{Ganancia media}) - (\% \text{ de operaciones perdedoras} * \text{Pérdida media})$

$E[X] = (53,18\% * 2,50) - (46,82\% * 2,27)$

$E[X] = 26,67\%$

Para que un sistema tenga el potencial de dar ganancias, la esperanza matemática debe ser positiva, esto indicaría que en promedio gana más de lo que pierde. Por tanto, se puede observar que se cumple la estrategia en el periodo IS.

II.1.3.3 Validación

II.1.3.3.1 Backtest en el periodo Out of Sample

En la validación se comprueba que los resultados obtenidos de la estrategia en el periodo *In Sample* también funcionan en el periodo *Out of Sample*.

Para ello se realiza de nuevo el *backtest* modificando el periodo. Los resultados obtenidos son los siguientes.

Periodo OS: 01/09/2019 - 01/01/2020

Par de divisas: GBP/USD

II.1.3.3.1 Pantalla Ejecución del Backtest con el Robot en el periodo OS



FIGURA 63:Pantalla Ejecución del Backtest con el Robot en el periodo OS

II.1.3.3.1.2 Grafica de curva de Equidad del Backtest Periodo OS



FIGURA 64: Grafica de curva de Equidad del Backtest Periodo OS

II.1.3.3.1.3 Pantalla de resultados del backtesting del Periodos

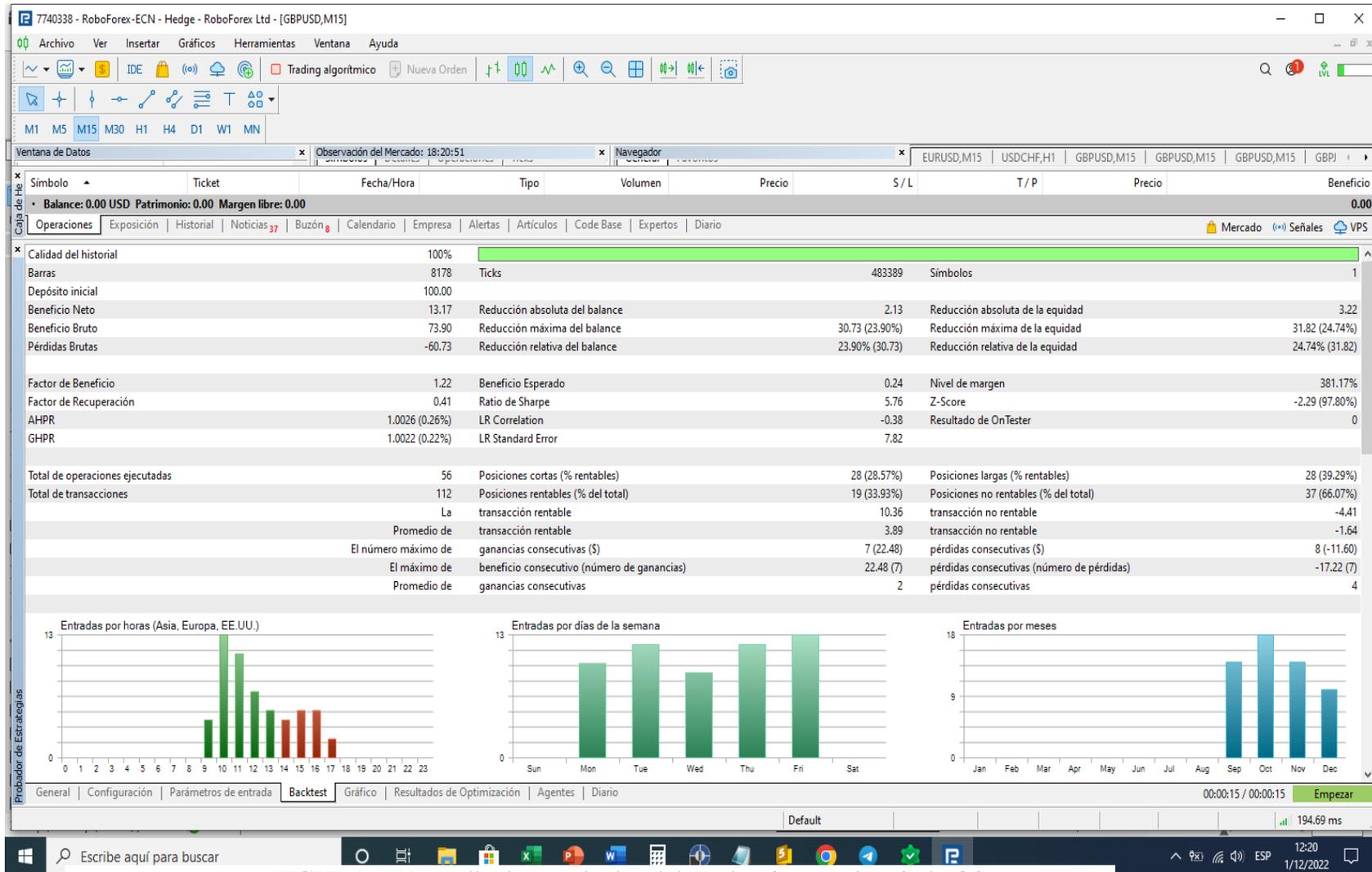


FIGURA 65: Pantalla de Resultados del Backtesting en el periodo OS



FIGURA 66:Pantalla de resultados del backtesting del Periodo OS

II.1.3.3.1.4 Interpretación de los resultados del Backtest del Periodo OS

Métricas	Valor	Observación
capital	100 \$	Total de capital invertido
Beneficio Neto Total	13.17 \$	Aceptable
Profit factor	1,22	>1 se considera rentable
Número de trades	112	Número propio de trading Intradialario
Beneficio Medio por Operación	0,12	Aceptable ya que, aunque se le reduzcan las comisiones sigue siendo positivo
Máximo Drawdown	39,27\$	Aceptable, no supera el capital

Tabla 29: Interpretación de los resultados del Backtest del Periodo OS

Esperanza matemática

Una métrica tremendamente importante es la **esperanza matemática** (*Mathematical Expectation*), supone otra forma de evaluar el rendimiento mismos datos que son obtenidos con los valores de las variables que te ofrece el backtesting.

$E[X] = (\% \text{ de operaciones ganadoras} * \text{Ganancia media}) - (\% \text{ de operaciones perdedoras} * \text{Pérdida media})$

$$E[X] = (40,19\% * 3,89) - (59,81\% * 1,64)$$

$$E[X] = 58,67\%$$

Para que un sistema tenga el potencial de dar ganancias, la esperanza matemática debe ser positiva, esto indicaría que en promedio gana más de lo que pierde. Por tanto, se puede observar se cumple la validación ya que se evidencia ganancia y cumplimiento de la estrategia en el periodo OS tambien.

II.1.3.3.2 Costes de la operativa

Los costes de la operativa o costes de transacción suponen todos los costes que van implícitos al realizar una operación en el mercado, por tanto, al evaluar el sistema deberán tenerse en cuenta ya que van a existir durante la operativa real. Estos costes son:

- **Comisión del bróker**, Se cobra por entrar y salir del mercado un monto de 4 dólares.
- **Spread**, también llamado horquilla, que se define como la diferencia entre el precio de compra (*ask*) y el de venta (*bid*) es :
 - a) Libradolar: 0.1 a 0.5 pips
 - b) Librayen: 3 a 5 pips

II.1.3.4 Estrategia aceptada

II.1.3.4.1 Operativa Real: Monitorización

Una vez realizados los pasos anteriores, se estará en disposición de comenzar la operativa real.

La operativa real consiste en dejar que el sistema de trading automático opere en el mercado con dinero de verdad, sin embargo, esto no es sinónimo de dejar al sistema ejecutándose y esperar a que lleguen los beneficios sin preocuparse por nada más. La monitorización del sistema de trading es un proceso fundamental que se debe realizar en la operativa real.

Periodo: (14/11/2022 - 30/11/2022)

Par de divisas: GBP/USD

II.1.3.4.1 Pantalla Ejecución de la operación real con el Robot



Tabla 30: Pantalla Ejecución de la operación real con el Robot

II.1.3.4.1.2 grafica de curva de Equidad

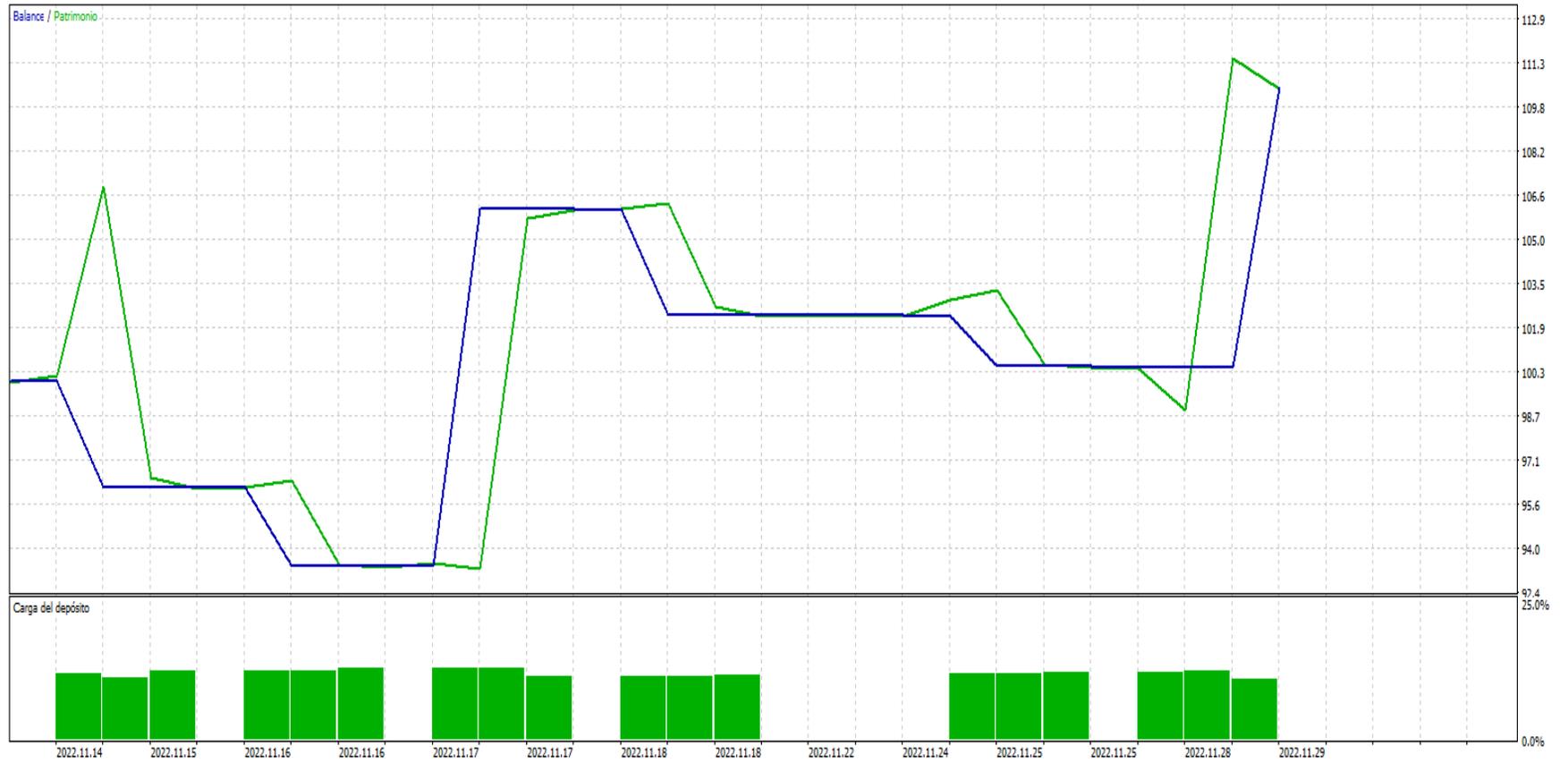


Tabla 31 Grafica de curva de Equidad

II.1.3.4.1.3 Pantalla de resultados de la operativa real

OrdeA61:N85										
Hora de apertura	Orden	Símbolo	Tipo	Volumen	Precio	S / L	T / P	Fecha/Hora	Estado	Comentario
2022.11.14 11:00:00	2	GBPUSD	sell stop	0.01 / 0.01	1,17800	1,18172	1,16014	2022.11.14 12:08:40	filled	
2022.11.15 09:24:40	3	GBPUSD	buy	0.01 / 0.01	0,00000			2022.11.15 09:24:40	filled	sl 1.18172
2022.11.15 09:30:00	4	GBPUSD	sell stop	0.01 / 0.01	1,17680	1,17957	1,16348	2022.11.15 18:00:00	expired	2.11.15 18:00]
2022.11.16 10:00:00	5	GBPUSD	sell stop	0.01 / 0.01	1,18588	1,18866	1,17251	2022.11.16 10:33:40	filled	
2022.11.16 11:00:40	6	GBPUSD	buy	0.01 / 0.01	0,00000			2022.11.16 11:00:40	filled	sl 1.18866
2022.11.17 09:45:00	7	GBPUSD	sell stop	0.01 / 0.01	1,18927	1,19193	1,17650	2022.11.17 11:32:40	filled	
2022.11.17 15:59:40	8	GBPUSD	buy	0.01 / 0.01	0,00000			2022.11.17 15:59:40	filled	tp 1.17650
2022.11.18 11:30:00	9	GBPUSD	sell stop	0.01 / 0.01	1,18935	1,19310	1,17132	2022.11.18 13:45:40	filled	
2022.11.18 14:16:40	10	GBPUSD	buy	0.01 / 0.01	0,00000			2022.11.18 14:16:40	filled	sl 1.19310
2022.11.21 11:15:00	11	GBPUSD	buy stop	0.01 / 0.01	1,18507	1,18068	1,20614	2022.11.21 18:00:00	expired	2.11.21 18:00]
2022.11.22 11:45:00	12	GBPUSD	sell stop	0.01 / 0.01	1,18369	1,18637	1,17080	2022.11.22 18:00:00	expired	2.11.22 18:00]
2022.11.23 11:45:00	13	GBPUSD	sell stop	0.01 / 0.01	1,18907	1,19090	1,18029	2022.11.23 18:00:00	expired	2.11.23 18:00]
2022.11.24 14:00:00	14	GBPUSD	sell stop	0.01 / 0.01	1,20763	1,21128	1,19011	2022.11.24 18:00:00	expired	2.11.24 18:00]
2022.11.25 10:00:00	15	GBPUSD	buy stop	0.01 / 0.01	1,21094	1,20917	1,21941	2022.11.25 10:22:40	filled	
2022.11.25 11:43:40	16	GBPUSD	sell	0.01 / 0.01	0,00000			2022.11.25 11:43:40	filled	sl 1.20917
2022.11.28 10:30:00	17	GBPUSD	sell stop	0.01 / 0.01	1,20510	1,20767	1,19276	2022.11.28 16:03:40	filled	
2022.11.29 23:54:59	18	GBPUSD	buy	0.01 / 0.01	0,00000			2022.11.29 23:54:59	filled	end of test

FIGURA 67:Listado de Compras y Ventas

Transacciones												
Fecha/Hora	ansacción	Símbolo	Tipo	Dirección	Volumen	Precio	Orden	Comisión	Swap	Beneficio	Balance	Comentario
2022.11.14	00:00:00		balance					0,00	0,00	100,00	100,00	
2022.11.14	12:08:40	GBPUSD	sell	in	0.01	1,17800	2	- 0,02	0,00	0,00	99,98	
2022.11.15	09:24:40	GBPUSD	buy	out	0.01	1,18172	3	- 0,02	- 0,03	- 3,72	96,21	sl 1.18172
2022.11.16	10:33:40	GBPUSD	sell	in	0.01	1,18588	5	- 0,02	0,00	0,00	96,19	
2022.11.16	11:00:40	GBPUSD	buy	out	0.01	1,18866	6	- 0,02	0,00	- 2,78	93,39	sl 1.18866
2022.11.17	11:32:40	GBPUSD	sell	in	0.01	1,18927	7	- 0,02	0,00	0,00	93,37	
2022.11.17	15:59:40	GBPUSD	buy	out	0.01	1,17650	8	- 0,02	0,00	12,77	106,12	tp 1.17650
2022.11.18	13:45:40	GBPUSD	sell	in	0.01	1,18935	9	- 0,02	0,00	0,00	106,10	
2022.11.18	14:16:40	GBPUSD	buy	out	0.01	1,19310	10	- 0,02	0,00	- 3,75	102,33	sl 1.19310
2022.11.25	10:22:40	GBPUSD	buy	in	0.01	1,21094	15	- 0,02	0,00	0,00	102,31	
2022.11.25	11:43:40	GBPUSD	sell	out	0.01	1,20917	16	- 0,02	0,00	- 1,77	100,52	sl 1.20917
2022.11.28	16:03:40	GBPUSD	sell	in	0.01	1,20510	17	- 0,02	0,00	0,00	100,50	
2022.11.29	23:54:59	GBPUSD	buy	out	0.01	1,19506	18	- 0,02	- 0,03	10,04	110,49	end of test
								- 0,24	- 0,06	10,79	110,49	

FIGURA 68: Transacciones

II.1.3.4.1.4 Interpretación de los resultados de la operativa real

Métricas	Valor	Observación
capital	100 \$	Total de capital invertido
Beneficio Neto Total	10.49 \$	Aceptable
Profit factor	1,86	>1 se considera rentable
Número de trades	12	Número propio de trading Intradario
Beneficio Medio por Operación	0,87	Aceptable ya que, aunque se le reduzcan las comisiones sigue siendo positivo
Máximo Drawdown	12,25\$	Aceptable, no supera el capital

Tabla 32: Interpretación de los resultados de la operativa real

Esperanza matemática

Una métrica tremendamente importante es la **esperanza matemática** (*Mathematical Expectation*), supone otra forma de evaluar el rendimiento mismos datos que son obtenidos con los valores de las variables que te ofrece el backtesting.

$E[X] = (\% \text{ de operaciones ganadoras} * \text{Ganancia media}) - (\% \text{ de operaciones perdedoras} * \text{Pérdida media})$

$E[X] = (28,57\% * 11,37) - (71,42\% * 3,06)$

$E[X]=107,67\%$

Para que un sistema tenga el potencial de dar ganancias, la esperanza matemática debe ser positiva, esto indicaría que en promedio gana más de lo que pierde. Por tanto, se puede observar que en operativa real también se tiene resultados positivos.

II.1.4 Código de programación

```
//+-----+
//|          Estrategia killzoneLondres.mq5 |
//|          |
//+-----+
```

```
#property copyright "Programando en MQL5"
#property link      "robot de Trading Wilfredo Aparicio"
#property version   "1.00"
```

```
/*
```

Descripción de la estrategia:

1. TimeFrame de trabajo: M15
2. Se toma como referencia la amplitud de la sesión asiática.
3. Se espera a que el precio de cotización rompa por encima o por debajo el rango asiático.
4. Una vez roto se coloca una orden pendiente en sentido contrario a la rotura, justo a la mitad del rango, con un Stop Loss en un extremo del rango y un Take Profit en el otro extremo y cuya distancia es parametrizada.
5. Si cuando termina la sesión europea la orden pendiente no ha sido alcanzada, entonces se elimina la misma.
6. Se repite el punto 1 al día siguiente.

```
*/
```

```
//+-----+
```

```
//[DECLARACIÓN DE LAS VARIABLES GLOBALES]
```

```
//+-----+
```

```
//Variables globales para la gestión de los datos históricos
```

```
datetime TiemposVelas[]; //Array que almacena los tiempos de comienzo de las velas
```

```
double CierreVelas[]; //Array que almacena los precios de cierre de las velas
```

```

double MaximoVelas[];      //Array que almacena los precios máximos de las velas
double MinimoVelas[];     //Array que almacena los precios mínimos de las velas
double MaximoVelasActual[]; //Array que almacena el precio máximo de la vela actual
double MinimoVelasActual[]; //Array que almacena el precio mínimo de la vela actual

//Variables globales para la gestión del tiempo
MqlDateTime HorarioComienzoReferencia; //Almacena los tiempos de comienzo de cada
sesión asiática en formato MqlDateTime
MqlDateTime HorarioFinReferencia; //Almacena los tiempos de fin de cada sesión
asiática en formato MqlDateTime
datetime HorarioComienzoReferenciaDT; //Almacena los tiempos de comienzo de cada
sesión asiática en formato datetime
datetime HorarioFinReferenciaDT; //Almacena los tiempos de fin de cada sesión asiática
en formato datetime
MqlDateTime HorarioActual; //Almacena el tiempo actual en formato MqlDateTime
bool Control = false; //Variable utilizada para controlar la ejecución de un bloque
de código solo una vez por cada sesión asiática
datetime TiempoUltimaActualizacion = 0; //Almacena el tiempo de comienzo de la vela
actual utilizada por la función NuevaVela

//Variables para dibujar la sesión asiática
const string Rectangulo = "Rectangulo"; //Constante que se utiliza como nombre base de
todos los rectángulos que se van a dibujar
int Contador = 0; //Variable contadora que se añade al nombre base de los
objetos rectángulos para que cada uno de ellos tenga un nombre diferente
string ObjetoRectangulo = NULL; //Almacena el nombre completo de cada objeto
rectángulo que se va a dibujar

//Estructura con información de la sesión asiática y europea
struct BloqueReferencia //Creación del tipo BloqueReferencia
{

```

```

double PrecioMasAlto;           //Almacena el precio más alto de la sesión asiática
double PrecioMasBajo;          //Almacena el precio más bajo de la sesión asiática
double AmplitudRango;          //Almacena la amplitud en precio de la sesión asiática
datetime FechaComienzo;        //Almacena la fecha de comienzo de la sesión asiática
};

BloqueReferencia BloquesDeReferencia; //Variable de tipo BloqueReferencia
bool SesionLondres = false; //Indica si estamos en la sesión de Londres

//Variables para las órdenes comerciales

MqlTradeRequest Solicitud; //Declaración de un objeto MqlTradeRequest utilizada
por las funciones que posicionan las órdenes pendientes

MqlTradeResult Respuesta; //Declaración de un objeto MqlTradeResult utilizada
por las funciones que posicionan las órdenes pendientes

bool EnvioSolicitud = false; //Declaración de la variable que indica si la orden ha
sido emplazada o ejecutada correctamente

double TakeProfit; //Take Profit en precio
double StopLoss; //Stop Loss en precio
double PrecioEntrada; //Precio de entrada de la orden pendiente
datetime ExpiracionOrdenPendiente; //Tiempo de expiración de la orden pendiente
string Comentario = NULL; //Comentario de la orden
ulong NumeroMagico =100; //Número mágico del asesor experto
ulong Ticket; //Número de ticket de la orden pendiente
double Volumen; //Almacena el volumen en lotes de la operación
uint Codigo; //Código devuelto por el broker cuando se ejecuta la orden
ulong Desviacion = 0; //Desviación permitida en el precio al colocar la orden
ENUM_ORDER_TYPE_FILLING Relleno = ORDER_FILLING_IOC;//Política de relleno de
volumen de la orden

bool AceptacionSolicitud; //Almacena si la orden ha sido emplazada correctamente

//+-----+

```

```

//|DECLARACIÓN DE VARIABLES INPUT|
//+-----+
input group "Parámetros configurables del Robot" //Separa visualmente este conjunto de
variables input con ese texto
input int HoraComienzo = 0; //Hora de comienzo de la sesión asiática
input int HoraFin = 9; //Hora de fin de la sesión asiática
input bool TiempoLocal=true; //Fecha de referencia para los cálculos del tiempo
(PC local o servidor broker)
input double MulTakeProfit = 1.3; //Factor Riesgo-Beneficio
input double PorcentajeRiesgo = 1; //Porcentaje a arriesgar de la cuenta en cada
operación

//+-----+
//| MANEJADOR DE EVENTOS OnInit() |
//+-----+
void OnInit()
{
    //Convierte el formato de hora del comienzo y fin del bloque de tiempo de la sesión asiática a
    formato datetime y MqlDateTime

    //Para ello utiliza funciones que se encuentran implementadas al final de este código en la sección
    de FUNCIONES

    HorarioComienzoReferencia = CrearMqlDateTime(HoraComienzo,0,0,TiempoLocal);
    HorarioFinReferencia = CrearMqlDateTime(HoraFin,0,0,TiempoLocal);
    HorarioComienzoReferenciaDT = CrearDateTime(HoraComienzo,0,0,TiempoLocal);
    HorarioFinReferenciaDT = CrearDateTime(HoraFin,0,0,TiempoLocal);

    //Proporciona un tamaño de 1 elemento a los arrays que vamos a utilizar y los convierte en series
    temporales

    ArrayResize(TiemposVelas,1);
    ArraySetAsSeries(TiemposVelas,true);

```

```

ArrayResize(CierreVelas,1);
ArraySetAsSeries(CierreVelas,true);
ArrayResize(MaximoVelas,1);
ArraySetAsSeries(MaximoVelas,true);
ArrayResize(MinimoVelas,1);
ArraySetAsSeries(MinimoVelas,true);
ArrayResize(MaximoVelasActual,1);
ArraySetAsSeries(MaximoVelasActual,true);
ArrayResize(MinimoVelasActual,1);
ArraySetAsSeries(MinimoVelasActual,true);
}

//+-----+
//| MANEJADOR DE EVENTOS OnTick() |
//+-----+

void OnTick()
{
    //Comprueba si ha comenzado una nueva vela, utilizando una función que se encuentra en la
    //sección de FUNCIONES

    if(NuevaVela(_Symbol,PERIOD_CURRENT))
    {
        //Convierte el formato de hora del comienzo y fin del bloque de tiempo de la sesión asiática a
        //formato datetime y MqlDateTime

        //Para ello utiliza funciones que se encuentran implementadas al final de este código en la
        //sección de FUNCIONES

        HorarioComienzoReferencia = CrearMqlDateTime(HoraComienzo,0,0,TiempoLocal);
        HorarioFinReferencia = CrearMqlDateTime(HoraFin,0,0,TiempoLocal);
        HorarioComienzoReferenciaDT = CrearDateTime(HoraComienzo,0,0,TiempoLocal);
        HorarioFinReferenciaDT = CrearDateTime(HoraFin,0,0,TiempoLocal);
    }
}

```

```

//Se obtienen los datos históricos utilizando funciones de la sección FUNCIONES

AlmacenaTiemposComienzo(_Symbol,PERIOD_CURRENT,0,1,TiemposVelas);
//Almacena el tiempo de comienzo de la nueva vela

AlmacenaPreciosCierre(_Symbol,PERIOD_CURRENT,1,1,CierreVelas); //Almacena el
precio de cierre de la vela anterior

AlmacenaPreciosMaximo(_Symbol,PERIOD_CURRENT,1,1,MaximoVelas); //Almacena
el precio máximo de la vela anterior

AlmacenaPreciosMinimo(_Symbol,PERIOD_CURRENT,1,1,MinimoVelas); //Almacena el
precio mínimo de la vela anterior

TimeToStruct(TiemposVelas[0],HorarioActual); //Convierte el tiempo actual que
es de tipo datetime al tipo MqlDateTime

//Comprueba si se ha completado el tiempo de referencia de la sesión asiática
if((HorarioFinReferenciaDT == TimeCurrent())&&(!Control))
{
    Control = true;//Variable de control para que este if solo se ejecute una vez con cada nueva
sesión asiática.

    //Datos importantes del rango Asiatico. Inicializa los nuevos valores:

    BloquesDeReferencia.PrecioMasAlto =
MasAlto(_Symbol,PERIOD_CURRENT,1,(HoraFin-HoraComienzo)*4); //Almacena el precio
más alto de la sesión asiática. La multiplicación por 4 es porque la estrategia es para M15 y una
hora contiene 4 intervalos de 15 minutos. Mirar sección FUNCIONES

    BloquesDeReferencia.PrecioMasBajo =
MasBajo(_Symbol,PERIOD_CURRENT,1,(HoraFin-HoraComienzo)*4); //Almacena el precio
más bajo de la sesión asiática. Mirar sección FUNCIONES

    BloquesDeReferencia.AmplitudRango = (BloquesDeReferencia.PrecioMasAlto-
BloquesDeReferencia.PrecioMasBajo)/2;//Almacena la mitad de la amplitud del rango asiático.
Divide el rango entre 2 para obtener la mitad del rango que es donde se tiene que situar la orden
pendiente.

    BloquesDeReferencia.FechaComienzo = TimeCurrent(); //Almacena la fecha y hora actual

    ExpiracionOrdenPendiente = TimeCurrent()+32400; //La orden pendiente si no ha sido
alcanzada expira al finalizar la sesión de Londres. A la hora de comienzo de la sesión de Londres
se le suman 9 horas (en segundos) que durará esta sesión

```

//Dibujo del rectángulo de la sesión asiática:

ObjetoRectangulo = Rectangulo + (string)Contador; //Cada objeto gráfico tiene que tener un nombre. Se toma el nombre base mas la variable contadora

//Al dibujar el rectángulo le resto 15 minutos (900segundos) a la hora fin de bloque para que la sesión de Londres comience a la hora exacta:

ObjectCreate(0,ObjetoRectangulo,OBJ_RECTANGLE,0,HorarioComienzoReferenciaDT,BloquesDeReferencia.PrecioMasAlto,(HorarioFinReferenciaDT-900),BloquesDeReferencia.PrecioMasBajo); //Función el rectángulo de la sesión asiática

ChartRedraw(); //Actualiza el gráfico, sin esperar un nuevo tick

Contador++; //Actualiza el valor del contador de los nombres de los rectángulos.

}

//Comprueba si ha comenzado la sesión asiática

if((HorarioComienzoReferencia.hour == HorarioActual.hour) && Control)

{

Control = false; //El control de la ejecución del if anterior se coloca a false para que se pueda crear otro bloque de referencia de tiempo de la sesión asiática al día siguiente

SesionLondres = false; //Al comenzar la sesión asiática se actualiza el valor de la sesión de Londres a falso.

}

//Para que se coloque una orden pendiente de venta tiene que ocurrir:

//1. Que no haya ninguna operación previa abierta, 2. Haber terminado la sesión asiática y estar dentro del tiempo de la sesión de Londres,

//3. Que el precio de cierre de la vela rompa el máximo de la sesión asiática y que el valor mínimo de la vela esté dentro del rango asiático,

//es decir que no sea una vela cuya amplitud rompa a la vez tanto por arriba como por abajo el rango asiático

if((PositionsTotal()== 0) && (TimeCurrent()<ExpiracionOrdenPendiente) && Control && (SesionLondres == false) && (CierreVelas[0]>BloquesDeReferencia.PrecioMasAlto) && (MinimoVelas[0]>BloquesDeReferencia.PrecioMasBajo))

{

SesionLondres = true; //Actualiza el valor de esta variable para que no se ejecute más de una operación

PrecioEntrada = BloquesDeReferencia.PrecioMasAlto - BloquesDeReferencia.AmplitudRango; //El precio de entrada de la orden pendiente se sitúa en la mitad del rango asiático

TakeProfit = PrecioEntrada - BloquesDeReferencia.AmplitudRango * MulTakeProfit; //Take Profit se sitúa con respecto al precio de entrada a una distancia igual a la mitad del rango asiático multiplicado por el factor riesgo-beneficio

StopLoss = PrecioEntrada + BloquesDeReferencia.AmplitudRango; //Stop Loss se sitúa en la parte superior del rango asiático.

Volumen = CalculoVolumen(PorcentajeRiesgo, BloquesDeReferencia.AmplitudRango / _Point); //Función que calcula el volumen en función del porcentaje de riesgo y la distancia en puntos del stop loss. Esta función se encuentra en la sección FUNCIONES

//Función que coloca una orden pendiente de venta sell stop. Esta función se encuentra en la sección de FUNCIONES

AceptacionSolicitud = VentaSellStop(_Symbol, Ticket, Volumen, Codigo, PrecioEntrada, StopLoss, TakeProfit, Desviacion, NumeroMagico, Relleno, ORDER_TIME_SPECIFIED, ExpiracionOrdenPendiente, Comentario);

Print ("Se ha producido el primer cierre de vela por encima del bloque de referencia y se coloca una orden de venta pendiente"); //Muestra por consola este mensaje

}

//Para que se coloque una orden pendiente de compra tiene que ocurrir:

//1. Que no haya ninguna operación previa abierta, 2. Haber terminado la sesión asiática y estar dentro del tiempo de la sesión de Londres,

//3. Que el precio de cierre de la vela rompa el mínimo de la sesión asiática y que el valor máximo de la vela esté dentro del rango asiático,

//es decir que no sea una vela cuya amplitud rompa a la vez tanto por arriba como por abajo el rango asiático

if((PositionsTotal() == 0) && (TimeCurrent() < ExpiracionOrdenPendiente) && Control && (SesionLondres == false) && (CierreVelas[0] < BloquesDeReferencia.PrecioMasBajo) && (MaximoVelas[0] < BloquesDeReferencia.PrecioMasAlto))

{

SesionLondres = true; //Actualiza el valor de esta variable para que no se ejecute más de una operación

```

    PrecioEntrada = BloquesDeReferencia.PrecioMasBajo+BloquesDeReferencia.AmplitudRango;//El precio de
    entrada de la orden pendiente se sitúa en la mitad del rango asiático

```

```

    TakeProfit = PrecioEntrada+BloquesDeReferencia.AmplitudRango*MulTakeProfit;
    //Take Profit se sitúa con respecto al precio de entrada a una distancia igual a la mitad del rango
    asiático multiplicado por el factor riesgo-beneficio

```

```

    StopLoss = PrecioEntrada-BloquesDeReferencia.AmplitudRango; //Stop Loss
    se sitúa en la parte inferior del rango asiático.

```

```

    Volumen = CalculoVolumen(PorcentajeRiesgo,BloquesDeReferencia.AmplitudRango/_Point);//Función que
    calcula el volumen en función del porcentaje de riesgo y la distancia en puntos del stop loss. Esta
    función se encuentra en la sección FUNCIONES

```

```

    //Función que coloca una orden pendiente de compra buy stop. Esta función se encuentra en
    la sección de FUNCIONES

```

```

    AceptacionSolicitud = CompraBuyStop(_Symbol,Ticket,Volumen,Codigo,PrecioEntrada,StopLoss,TakeProfit,Desviaci
    on,NumeroMagico,Relleno,ORDER_TIME_SPECIFIED,ExpiracionOrdenPendiente,Comentari
    o);

```

```

    Print ("Se ha procucido el primer cierre de vela por debajo del bloque de referencia y se
    coloca una orden de compra pendiente"); //Muestra por consola este mensaje

```

```

    }

```

```

    }//Fin nueva vela

```

```

}//Fin Manejador de eventos OnTick

```

```

//+-----+

```

```

//| SECCIÓN DE FUNCIONES. ESTAS FUNCIONES SON EXPLICADAS EN EL CURSO
"PROGRAMANDO EN MQL5" |

```

```

//+-----+

```

```

//| FUNCIÓN QUE VERIFICA, CORRIGE Y NORMALIZA EL VOLUMEN DE LA FUTURA
OPERACIÓN|

```

```

//+-----+
double VerificarVolumen(double pVolumen)
{
    double minimoVolumen = SymbolInfoDouble(_Symbol,SYMBOL_VOLUME_MIN);
//Número de lotes mínimo para una transacción que permite el broker

    double maximoVolumen = SymbolInfoDouble(_Symbol,SYMBOL_VOLUME_MAX);
//Número de lotes máximo para una transacción que permite el broker

    double                precisionVolumen                =
SymbolInfoDouble(_Symbol,SYMBOL_VOLUME_STEP);//Precisión del volumen del broker

    double volumen;                //Volumen normalizado

    if(pVolumen < minimoVolumen)                //Si el volumen del parámetro
es menor que el mínimo
    {
        volumen = minimoVolumen;                //Devuelve un volumen igual al
mínimo volumen permitido por el broker
    }

    else if (pVolumen > maximoVolumen)                //Si el volumen del parámetro
es mayor que el máximo
    {
        volumen = maximoVolumen;                //Devuelve un volumen igual al
máximo volumen permitido por el broker
    }

    else                //Si el volumen es valido se normaliza
    {
        volumen = MathRound(pVolumen / precisionVolumen) *
precisionVolumen;//Normaliza el volumen para que esté permitido por el broker
    }

    return(volumen);
}

```

```
//| FUNCIÓN QUE CALCULA EL VOLUMEN DE LA FUTURA OPERACIÓN BASADO EN
UN PORCENTAJE DE PÉRDIDA Y EN UN STOP LOSS|
```

```
//+-----+
double CalculoVolumen(double pPorcentaje,double pStopLoss)
{
    double cantidadArriesgar; //Almacena la cantidad de
dinero a arriesgar
    double precioPuntoArriesgar; //Almacena el precio del
punto a arriesgar
    double precioPunto; //Almacena el precio del punto
de la cotización actual
    double volumen; //Almacena el volumen de la
operación
    if(pPorcentaje > 0 && pStopLoss > 0) //El porcentaje de
pérdida y el stop loss tienen que ser mayor que 0
    {
        cantidadArriesgar = AccountInfoDouble(ACCOUNT_BALANCE) * (pPorcentaje
/ 100); //Calcula el dinero a arriesgar
        precioPuntoArriesgar = (cantidadArriesgar / pStopLoss); //Precio
del punto en la futura operación
        precioPunto = SymbolInfoDouble(_Symbol,SYMBOL_TRADE_TICK_VALUE);
//Es el precio del movimiento de un punto en un símbolo
        volumen = (precioPuntoArriesgar / precioPunto); //Volumen a
aplicar en la futura operación
        volumen = VerificarVolumen(volumen); //Verifica si el
volumen es correcto y lo actualiza
        return(volumen); //Devuelve el volumen en
lotes de la futura operación
    }
    else
    {
        volumen = 0; //Si no se han pasado los
parámetros correctos para el cálculo del volumen, se asigna el volumen 0
    }
}
```

```

        return(volumen);
    }
}

//| FUNCIÓN QUE DEVUELVE EL BAJO MÁS BAJO DE UN CONJUNTO DE VELAS|
//+-----+
double MasBajo(string pSimbolo, ENUM_TIMEFRAMES pPeriodo,int pPosicionInicio,int
pCantidadVelas)
{
    double bajos[]; //Array dinámico que almacena los valores
    más bajos de la vela
    int indiceMinimo; //Almacena el índice del array donde se
    encuentra el bajo más bajo
    double masBajo; //Almacena el bajo más bajo
    ArraySetAsSeries(bajos,true); //Cambia el array a una serie temporal
    CopyLow(pSimbolo,pPeriodo,pPosicionInicio,pCantidadVelas,bajos); //Copia los bajos
    de un conjunto de velas
    indiceMinimo = ArrayMinimum(bajos); //Índice de la posición con el
    bajo más bajo
    masBajo = bajos[indiceMinimo]; //Almacena el bajo más bajo y lo
    devuelve a la llamada de la función
    return(masBajo);
}

//| FUNCIÓN QUE DEVUELVE EL ALTO MÁS ALTO DE UN CONJUNTO DE VELAS|
//+-----+
double MasAlto(string pSimbolo, ENUM_TIMEFRAMES pPeriodo,int pPosicionInicio,int
pCantidadVelas)
{
    double altos[]; //Array dinámico que almacena los valores
    más altos de la vela

```

```

        int indiceMaximo;                //Almacena el índice del array donde se
encuentra el alto más alto

        double masAlto;                  //Almacena el alto más alto

        ArraySetAsSeries(alto,true);     //Cambia el array a una serie temporal

        CopyHigh(pSimbolo,pPeriodo,pPosicionInicio,pCantidadVelas,alto); //Copia los
máximos de un conjunto de velas

        indiceMaximo = ArrayMaximum(alto); //Índice de la posición con el
alto más alto

        masAlto = alto[indiceMaximo];    //Almacena el alto más alto y lo
devuelve a la llamada de la función

        return(masAlto);
}

//| FUNCIÓN QUE INDICA SI HA COMENZADO UNA NUEVA VELA|
//+-----+

bool NuevaVela(string pSimbolo,ENUM_TIMEFRAMES pPeriodo)
{
    bool nuevaVela = false;             //Almacena si ha comenzado una nueva vela

    datetime velaDateTime[];           //Array que almacena el tiempo de comienzo
de la vela actual

    AlmacenaTiemposComienzo(_Symbol,pPeriodo,0,1,velaDateTime); //Obtiene el tiempo de
comienzo de la vela actual

    if (TiempoUltimaActualizacion != velaDateTime[0]) //Si el tiempo de comienzo de la
vela con la última actualización no coincide con el tiempo de comienzo de la vela actual
    {
        TiempoUltimaActualizacion = velaDateTime[0]; //Actualiza el tiempo de la última
actualización al tiempo de comienzo de la nueva vela

        nuevaVela = true;               //Ha comenzado una nueva vela
    }

    return(nuevaVela);
}

```

```
//| FUNCIÓN QUE A PARTIR DE LOS PARÁMETROS HORA Y MINUTOS, LOS PASA AL TIPO MQLDATETIME|
```

```
//+-----+
MqlDateTime CrearMqlDateTime(int pHora = 0,int pMinuto = 0,int pSegundo = 0,bool
pTiempoLocal=false)
{
    MqlDateTime tiempoMqlDateTime;           //Crea un objeto de la estructura
MqlDateTime
    datetime tiempoActual;                   //Almacena el tiempo actual del servidor o del PC
con el terminal Meta Trader 5
    if(pTiempoLocal == true)                 //Comprueba si el tiempo actual se cogerá
del servidor o del PC local
    {
        tiempoActual = TimeLocal();          //El tiempo actual se toma del PC local
con MetaTrader 5
    }
    else
    {
        tiempoActual = TimeCurrent();        //El tiempo actual se toma del servidor
    }
    TimeToStruct(tiempoActual,tiempoMqlDateTime); //Convierte el tiempo actual
que es de tipo datetime al tipo MqlDateTime
    tiempoMqlDateTime.hour = pHora;          //Asigna el parámetro hora a la
estructura tipo MqlDateTime
    tiempoMqlDateTime.min = pMinuto;         //Asigna el parámetro minuto a la
estructura tipo MqlDateTime
    tiempoMqlDateTime.sec = pSegundo;       //Asigna el parámetro segundo a la
estructura tipo MqlDateTime
    return(tiempoMqlDateTime);              //Devuelve la fecha con la hora y minutos
pasado por parámetros
}
```

|| FUNCIÓN QUE A PARTIR DE LOS PARÁMETROS HORA Y MINUTOS, LOS PASA AL TIPO MQLDATETIME PARA LUEGO PASARLOS AL TIPO DATETIME|

```

//+-----+
-----+

datetime CrearDateTime(int pHora = 0,int pMinuto = 0,int pSegundo = 0,bool
pTiempoLocal=false)
{
    MqlDateTime tiempoMqlDateTime;           //Crea un objeto de la estructura
MqlDateTime

    datetime tiempoActual;                   //Almacena el tiempo actual del servidor o del PC
con el terminal de Meta Trader 5

    if(pTiempoLocal == true)                //Comprueba si el tiempo actual se cogerá
del servidor o del PC local
    {
        tiempoActual = TimeLocal();         //El tiempo actual se toma del PC local
con el terminal de MetaTrader 5
    }
    else
    {
        tiempoActual = TimeCurrent();       //El tiempo actual se toma del servidor
    }

    TimeToStruct(tiempoActual,tiempoMqlDateTime); //Convierte el tiempo actual
que es de tipo datetime al tipo MqlDateTime

    tiempoMqlDateTime.hour = pHora;         //Asigna el parámetro hora a la
estructura tipo MqlDateTime

    tiempoMqlDateTime.min = pMinuto;       //Asigna el parámetro minuto a la
estructura tipo MqlDateTime

    tiempoMqlDateTime.sec = pSegundo;     //Asigna el parámetro segundo a la
estructura tipo MqlDateTime

    tiempoActual = StructToTime(tiempoMqlDateTime); //Vuelve a convertir el tipo
estructura MqlDateTime a un tipo datetime

```

```

        return(tiempoActual); //Devuelve la fecha con la hora y minutos
pasado por parámetros
    }

```

```

//| FUNCIÓN QUE DEVUELVE EL VALOR DE CIERRE DE LA VELA QUE OCUPA LA
POSICIÓN INDICADA POR EL PARÁMETRO|

```

```

//+-----+
void AlmacenaPreciosCierre(string pSimbolo,ENUM_TIMEFRAMES pPeriodo,int
pPosicionInicio,int pCantidadVelas,double &pVelas[])
{
    CopyClose(pSimbolo,pPeriodo,pPosicionInicio,pCantidadVelas,pVelas); //Función
propia de Mql5
}

```

```

//| FUNCIÓN QUE DEVUELVE EL VALOR MÁXIMO DE LA VELA QUE OCUPA LA
POSICIÓN INDICADA POR EL PARÁMETRO|

```

```

//+-----+
void AlmacenaPreciosMaximo(string pSimbolo,ENUM_TIMEFRAMES pPeriodo,int
pPosicionInicio,int pCantidadVelas,double &pVelas[])
{
    CopyHigh(pSimbolo,pPeriodo,pPosicionInicio,pCantidadVelas,pVelas); //Función propia
de Mql5
}

```

```

//| FUNCIÓN QUE DEVUELVE EL VALOR MÍNIMO DE LA VELA QUE OCUPA LA
POSICIÓN INDICADA POR EL PARÁMETRO|

```

```

//+-----+
void AlmacenaPreciosMinimo(string pSimbolo,ENUM_TIMEFRAMES pPeriodo,int
pPosicionInicio,int pCantidadVelas,double &pVelas[])
{
    CopyLow(pSimbolo,pPeriodo,pPosicionInicio,pCantidadVelas,pVelas); //Función propia
de Mql5
}

```

```
}
```

```
///| FUNCIÓN QUE DEVUELVE EL TIEMPO DE INICIO DE LA VELA QUE OCUPA LA  
POSICIÓN INDICADA POR EL PARÁMETRO|
```

```
//+-----+
```

```
void AlmacenaTiemposComienzo(string pSimbolo,ENUM_TIMEFRAMES pPeriodo,int  
pPosicionInicio,int pCantidadVelas,datetime &pVelas[])
```

```
{
```

```
    CopyTime(pSimbolo,pPeriodo,pPosicionInicio,pCantidadVelas,pVelas); //Función propia  
de Mql5
```

```
}
```

```
///|FUNCIÓN QUE REALIZA EL EMPLAZAMIENTO DE UNA ORDEN DE COMPRA BUY  
STOP|
```

```
//+-----+
```

```
bool CompraBuyStop(string pSymbol,ulong &pTicket,double pVolumen,uint &pCodigo,double  
pPrecio,double pStop = 0, double pProfit = 0, ulong pDesviacion = 0,
```

```
    ulong pNumeroMagico=1,ENUM_ORDER_TYPE_FILLING pRelleno =  
ORDER_FILLING_FOK,ENUM_ORDER_TYPE_TIME pExpiracion = ORDER_TIME_GTC,
```

```
    datetime pTiempoExpiracion = 0,string pComentario = NULL)
```

```
{
```

```
    ZeroMemory(Solicitud); //Pone los valores del objeto Solicitud a 0
```

```
    ZeroMemory(Respuesta); //Pone los valores del objeto Respuesta a 0
```

```
    Solicitud.action = TRADE_ACTION_PENDING; //La solicitud es para el  
emplazamiento de una orden pendiente
```

```
    Solicitud.type = ORDER_TYPE_BUY_STOP; //La solicitud es para una orden  
pendiente Buy Stop
```

```
    Solicitud.symbol = pSymbol; //La solicitud es para el símbolo pasado por  
parámetro
```

```
    Solicitud.sl = pStop; //Stop loss en precio
```

```

    Solicitud.tp = pProfit;                //Take profit en precio
    Solicitud.comment = pComentario;      //Comentario
    Solicitud.volume = pVolumen;         //Volumen en lotes de la orden
    Solicitud.deviation = pDesviacion;    //Desplazamiento en puntos permitido sobre
    el precio de entrada
    Solicitud.type_filling = pRelleno ;    //Tipo de política de relleno del volumen de
    la orden
    Solicitud.magic = pNumeroMagico;      //Número mágico del Expert Advisor
    Solicitud.price = pPrecio;            //El precio Stop de la orden pendiente
    Solicitud.type_time = pExpiracion;    //El tipo de expiración de la orden pendiente
    Solicitud.expiration = pTiempoExpiracion; //Es la fecha de expiración de la orden
    pendiente si el tipo de expiración fue Order_Time_Specified o Order_Time_Specified_Day

    EnvioSolicitud = OrderSend(Solicitud,Respuesta); //Envío de la orden al servidor y
    almacena el resultado del emplazamiento

    pTicket = Respuesta.order;            //Se almacena en el parámetro pasado por referencia
    pTicket, el número de ticket de la orden ejecutada.

    pCodigo = Respuesta.retcode;          //Se almacena en el parámetro pasado por referencia
    pCodigo, el código de respuesta del Broker

    return(EnvioSolicitud);               //Devuelve si la la orden ha sido emplazada o no

}

```

//|FUNCIÓN QUE REALIZA EL EMPLAZAMIENTO DE UNA ORDEN DE VENTA SELL STOP|

//+-----+

```

bool VentaSellStop(string pSymbol,ulong &pTicket,double pVolumen,uint &pCodigo,double
pPrecio,double pStop = 0, double pProfit = 0, ulong pDesviacion = 0,
    ulong pNumeroMagico=1,ENUM_ORDER_TYPE_FILLING pRelleno =
ORDER_FILLING_FOK,ENUM_ORDER_TYPE_TIME pExpiracion = ORDER_TIME_GTC,
    datetime pTiempoExpiracion = 0,string pComentario = NULL)

```

```

{
    ZeroMemory(Solicitud);           //Pone los valores del objeto Solicitud a 0
    ZeroMemory(Respuesta);          //Pone los valores del objeto Respuesta a 0

    Solicitud.action = TRADE_ACTION_PENDING;           //La solicitud es para el
emplazamiento de una orden pendiente

    Solicitud.type = ORDER_TYPE_SELL_STOP;           //La solicitud es para una orden
pendiente Sell Stop

    Solicitud.symbol = pSymbol;           //La solicitud es para el símbolo pasado por
parámetro

    Solicitud.sl = pStop;                 //Stop loss en precio
    Solicitud.tp = pProfit;               //Take profit en precio
    Solicitud.comment = pComentario;      //Comentario
    Solicitud.volume = pVolumen;         //Volumen en lotes de la orden
    Solicitud.deviation = pDesviacion;    //Desplazamiento en puntos permitido sobre
el precio de entrada

    Solicitud.type_filling = pRelleno ;    //Tipo de política de relleno del volumen de
la orden

    Solicitud.magic = pNumeroMagico;      //Número mágico del Expert Advisor
    Solicitud.price = pPrecio;            //El precio Stop de la orden pendiente
    Solicitud.type_time = pExpiracion;    //El tipo de expiración de la orden pendiente

    Solicitud.expiration = pTiempoExpiracion; //Es la fecha de expiración de la orden
pendiente si el tipo de expiración fue Order_Time_Specified o Order_Time_Specified_Day

    EnvioSolicitud = OrderSend(Solicitud,Respuesta); //Envío de la orden al servidor y
almacena el resultado del emplazamiento

    pTicket = Respuesta.order;           //Se almacena en el parámetro pasado por referencia
pTicket, el número de ticket de la orden ejecutada.

    pCodigo = Respuesta.retcode;         //Se almacena en el parámetro pasado por referencia
pCodigo, el código de respuesta del Broker

    return(EnvioSolicitud);             //Devuelve si la orden ha sido emplazada o no

```

COMPONENTE II

CAPACITACION

I.2 COMPONENTE II

MODELO DE SOCIALIZACIÓN Y CAPACITACIÓN

II.2.1 Introducción

De acuerdo al modo de vida actual del mundo, el término Capacitación y Sistemas de Información están modificando la forma de trabajo de las empresas, los sistemas de Información ayudan a acelerar los procesos, por tanto, las organizaciones que los implantan logran ventajas competitivas al adoptarlos en sus funciones. Pero si el personal o el usuario no está debidamente capacitado en el uso del sistema la utilidad será deficiente y en mucho caso incompleto.

La capacitación se refiere a las metodologías que se usan para proporcionar a las personas dentro de la empresa, las habilidades necesarias que necesitan para realizar su trabajo de una manera más eficiente, esto contempla desde pequeños cursos que le permitan al usuario entender el funcionamiento básico del sistema nuevo, hasta capacitación más profunda y avanzadas a bases de prácticas y material didáctico como libros (Tutoriales) y otros.

La capacitación es un proceso que lleva a la mejora continua de la producción y con esto implantar nuevas formas de trabajo, como en este caso el manejo adecuado del Sistema automático de inversión desarrollado y Conocimientos en el área de las TIC.

II.2.2 Aspectos Pedagógicos de la Capacitación

II.2.2.1 Definición de la capacitación

La capacitación es un proceso educacional de carácter estratégico aplicado de manera organizada y sistémica, mediante el cual los colaboradores adquieren o desarrollan conocimientos y habilidades específicas relativas al trabajo, y modifica sus actitudes frente a los quehaceres de la organización, el puesto o el ambiente laboral.

II.2.2.2 Objetivos de la capacitación

- Preparar a los colaboradores y usuarios para la ejecución y el correcto uso del sistema de inversión en las diversas oportunidades que presenta de Operar y su comportamiento en el Mercado de Divisas.
- Proporcionar habilidades de uso correcta de la plataforma de inversión actuales sino también para otras funciones para las cuales operador o dueño de la cuenta puede ser considerado.

II.2.2.3 Importancia de la capacitación

Permite que:

- Consolidación en la Familiarización con el uso de plataforma de inversión.
- Mayor identificación con la cultura de la inversión.
- Mayor retorno de la inversión.
- Alta productividad.
- Promueve la creatividad, innovación y disposición a nuevas tecnologías.
- Mejora el desempeño de la herramienta de inversión.
- Desarrollo de una mejor habilidad para identificar patrones de comportamiento del mercado.

II.2.2.4 Beneficios de la Capacitación

Como beneficia la capacitación a las organizaciones o individuos:

- Conduce a rentabilidad más alta y a actitudes más positivas.
- Mejora el conocimiento en el uso de herramientas de inversión en línea.
- Cambia tu perspectiva en las formas de obtener ingresos.
- Ayuda al personal a identificarse y cambiar los objetivos personales o de la empresa.
- Promueve al desarrollo con vistas a la proyección.

- Elimina los costos de recurrir a consultores externos.

II.2.3 Capacitación

La metodología utilizada para esta capacitación i/o taller se basa en el modelo de experiencia —Aprender Haciendol, logrando de esta manera la motivación de ser constructores de su propio conocimiento. Involucra trabajos de completa actividad en Talleres de construcción de contenidos, Laboratorios, Guías, donde se incentiva el trabajo creativo y práctico, experimentándose con procesos o fenómenos a partir de ideas o propuestas teóricas previamente formuladas.

Es una metodología busca desarrollar su sensibilidad frente a problemas reales, estudiar alternativas de solución y evaluar sus implicancias en conjunto con la utilización de tecnologías.

II.2.3.1 Estrategias didácticas

II.2.3.1.1 Sesiones Individuales y Grupales

II.2.3.1.1.1 Descripción

Se desarrolla Actividades Guiadas donde los capacitadores hacen uso del material guía y van descubriendo por sí solos las potencialidades de los programas y herramientas y que luego ellos mismos documentan haciendo énfasis en cuáles de las experiencias descubiertas son de principal apoyo para su práctica pedagógica.

II.2.3.2 Aspectos prácticos

II.2.3.2.1 Actividad Teórico-práctica

Propicia la modalidad del curso teórico con una actividad de la práctica en relación a la temática de estudio. Lo teórico y lo práctico se dan simultáneamente en forma conjunta e interrelacionada y es dirigida directamente por el capacitador participante.

II.2.3.2.2 Entrenamiento Virtual

Formación en la que los elementos teóricos y lo prácticos se da simultáneamente sino en entonos virtuales*. Para este sistema se utiliza el seguimiento de Tutores Facilitadores para cada área temática.

II.2.3.3 Mecanismos de trabajo durante la capacitación

II.2.3.3.1 Fase Inicial:

Análisis de la Realidad Social de la Región: donde los Facilitadores intercambian diferentes criterios con los docentes logrando tener una visión clara de las necesidades pedagógicas de los participantes enfocando la capacitación a la Presentación de Temas Individuales.

Diagnóstico de Iniciación: donde se recolecta información mediante evaluaciones de conocimientos que nos permitirá medir el nivel que poseen los participantes en la utilización de las aplicaciones.

Motivación y Sensibilidad: donde los Facilitadores realizan dinámicas para Motivar y sensibilizar sobre la importancia de desarrollar capacidades en el manejo de las TIC, y de esta manera los capacitadores puedan aplicar la herramienta y aportar en su conjunto al desarrollo de su comunidad.

II.2.3.3.2 Fase de Capacitación

Investigación y documentación: se promueve en todas las estrategias didácticas aplicadas a la investigación continua y descubrimiento de la tecnología reconociendo su valor e identificando su potencial para ser utilizados en su preparación de material, todo esto desarrollado por los mismos capacitadores.

II.2.3.3.3 Fase de finalización

Medición de aprovechamiento: donde los Facilitadores realizan las evaluaciones en forma de preguntas una vez de término de grupo capacitado.

II.2.3.4 Modalidades Prácticas de capacitación

El Plan de capacitación podrá usar otras modalidades que se incorporen según las necesidades.

II.2.3.4.1 Introducción

Su objeto es la ambientación inicial al medio social y físico donde trabaja y se realiza el análisis y la inversión en el mercado.

II.2.3.4.4 Seminarios –Talleres

Son eventos de corta duración, alrededor de 14 horas en tres fechas y sobre temas puntuales que sirvan para reforzar o difundir aspectos técnicos de uso adecuado del sistema de inversión y las plataformas de manera general.

II.2.3.5 Medios de Capacitación

II.2.3.5.1 Conferencia

Permite llegar a una gran cantidad de personas y transmitir un amplio contenido de información o enseñanza. Se puede emplear como explicación preliminar antes de demostraciones prácticas. Por ejemplo, es útil mostrar el funcionamiento en real del sistema.

II.2.3.5.2 Manuales de Capacitación

Manuales de capacitación u otros impresos, diagramas que permiten la exposición repetida, es útil aplicación de secuencias largas o procedimientos complicados que no pueden retenerse en una sola presentación. Pueden combinarse con conferencias y prácticas de tareas reales.

II.2.3.5.3 Videos

Puede sustituir a las conferencias o demostraciones formales, permite la máxima utilización de instructores más capaces. Los cortes, empalmes o en la cámara lenta son útiles para incidir en demostraciones de realidad. Ayudan a la comprensión de ideas abstractas y en la modificación de actitudes.

La grabación y proyección en videos de los colaboradores sujetos a capacitación, son un medio muy eficaz, sobre todo cuando se trata de mejorar la calidad del servicio.

II.2.3.5.4 Simuladores

Dan al aprendiz la posibilidad de participación y práctica repetida mediante la adquisición de habilidades necesarias en el trabajo real, se usan también sustitutos del equipo real. Pueden aislar y combinar las diferentes partes críticas en la inversión dentro los mercados financiero.

II.2.2.4 Socialización

Para realizar la Socialización acerca del uso del Sistema Automático de Inversión, y las herramientas que ofrece las Redes sociales y las plataformas virtuales se hizo uso de técnicas de presentaciones eficaces con el método

II.2.4.1 Contenido

El contenido de la socialización se subdivide en 3 módulos de los cuales se desglosa información a continuación se realiza una exposición, con los siguientes temas:

1. Introducción al Trading e Inversiones en Línea
2. Trading Tradicional Y Trading Algorítmico
3. Conceptos de Análisis Técnico y análisis Fundamental
4. Indicadores y estrategias de inversión
5. Desarrollo de Software Tradicional
6. Beneficiario
7. Instalación, equipos y costos
8. Programador y código Reutilizable

9. Metodologías para desarrollo de software
10. RUP vs UML
11. MQL5 Y lenguajes de programación.
12. Quantdatamanager
13. Modelo de Datos
14. Internet y sociedad
15. Probador de estrategias y backtesting.

II.2.4.2 Medios a utilizar

II.2.4.2.1 Aspectos técnicos

- 1 DataDisplay o monitor LCD.
- 1 computador con el Sistema Desarrollado.
- Diapositivas de Exposición hechas en PowerPoint.

II.2.4.2.2 Aspectos logísticos

- Material digital informativo (cds,dvds)
- Plataforma Virtual de zoom (grabación y clases)

II.4.2.3 Conclusiones

Una vez realizado la capacitación y según la ejecución del contenido de la capacitación, se pudo diferenciar tipos de cambio en el personal, como por ejemplo el desarrollo de destreza y conocimiento de manera directa respecto al tema de Internet; además se desarrolló un cambio de actitud negativa por actitudes más favorables y positivas con respecto a las inversiones en Línea.

CAPITULO III

CONCLUSIONES Y

RECOMENDACIONES

III.1 Conclusiones

En base al trabajo realizado y a la experiencia que se fue recopilando a medida que se desarrolló este proyecto, permite llegar a las siguientes conclusiones:

- Para el diseño y la ejecución del proyecto se utilizó la Metodología SML (Sistema de Marco lógico), el cual consta de tres herramientas (Herramienta de diagnóstico, herramienta de identificación y matriz de marco lógico).

- La Metodología SCRUM nos permite realizar mejoras al software, puesto que el proceso Sprint permite iterar y realizar las modificaciones necesarias para alcanzar la calidad del software.

- La Metodología SCRUM funciona para desarrollar software de manera Ágil y sin excesiva documentación.

- La utilización del Lenguaje UML permitió diseñar la arquitectura del software de manera confiable y mantenible.

- El sistema desarrollado agiliza y disminuye el trabajo de los usuarios puesto que coloca operaciones con solo habilitar la opción de trading automático en la plataforma de Metatrader.

- El lenguaje de Programación MQL5 es un lenguaje potenciado permite desarrollar software con velocidad y calidad en la plataforma de Metatrader.

- Por último, se puede afirmar que el desarrollo del trading algorítmico a sido definitivamente útil para mejorar la estrategia del inversor de modo que podrá operar de manera fácil y oportuna llegando a tener un mejor rendimiento en el mercado de divisas.

III.2 Recomendaciones

- Para el desarrollo de futuros proyectos con características similares a éste se recomienda hacer uso de las metodologías utilizadas como (SCRUM, XP,), ya que éstas son estándares para el desarrollo de este tipo de aplicaciones, ya que permitieron llegar a la conclusión del Proyecto.

- Se recomienda utilizar el lenguaje de programación MQL5 por su compatibilidad con la plataforma Metatrader5.

- Se recomienda utilizar el lenguaje de Modelado UML y seleccionar un tipo de diagrama que permita representar de manera el comportamiento del sistema.

- Se recomienda utilizar QuantDataManager para descargar datos históricos de la divisa por la calidad de los mismos.

GLOSARIO DE TERMINOS

Activo.- Es aquello que intercambiamos en el mercado. Dentro del Forex, EUR/USD, GBP/USD... son activos.

Análisis fundamental.- El análisis fundamental es un método que se emplea para evaluar un activo y para analizar los factores que podrían influir en su precio... Un análisis basado solo en noticias, indicadores económicos y acontecimientos globales.

Análisis técnico.- El análisis técnico es un sistema que permite examinar y predecir los movimientos de precios en los mercados financieros a partir de datos históricos y estadísticas de mercado. Se basa en la idea de que, si un inversor puede identificar patrones previos, entonces podrá predecir los movimientos futuros de los precios de manera bastante exacta.

Apalancamiento.- El apalancamiento es un concepto que puede ayudar a multiplicar su exposición a

Beneficio (Ganancia).- Una cantidad de dinero ganado al cierre de la posición.

Bid (Demanda).- Es el precio de demanda, el precio que se usa cuando se vende.

Bróker.- Un bróker es un intermediario sobre el que recae la responsabilidad de cruzar la oferta con la demanda en un mercado determinado a cambio de una comisión. Existen bróker físicos y bróker online, en los que todo el proceso de gestión de órdenes se lleva a cabo electrónicamente donde los inversores no necesitan desplazarse para realizar las operaciones.

Chartismo.- Se basa en el análisis de gráficos para entender las variaciones de los precios de un activo, con el objetivo de obtener predicciones más fiables y especular sobre su comportamiento en el futuro.

Cuidado con el chartismo, su principal inconveniente es la subjetividad.

Comisión.- Es el coste que nos cobra el bróker por la ejecución de las órdenes.

Corto.- En el trading, con corto nos referimos a una operación en la que se obtendrán beneficios si el activo sobre el que se opera baja de precio. Lo llamaremos ir corto, tener una posición corta.

Divisa base.- La divisa base es la primera divisa nombrada en un par. Por ejemplo, en EUR/USD la divisa base es el EUR (euro).

Divisa cotizada.- La segunda divisa nombra en un par. También se denomina contra-divisa. En el ejemplo anterior, el USD (dólar).

EA (Expert Advisor).- Los Expert Advisors (o EA por sus siglas en inglés) son programas que funcionan en la plataforma de MetaTrader y que se emplean para monitorizar y operar en los mercados financieros a través de algoritmos: se dedican a encontrar oportunidades según parámetros establecidos y, o se las notifican, o abren una posición de manera automática. Una vez que su posición se ha abierto, puede añadir las condiciones de cierre a un EA, así como stops, stops dinámicos y límites.

Flotante.- El flotante es la cantidad (beneficio o pérdida) de las posiciones que tenemos abiertas.

Largo.- En la terminología del trading, largo hace referencia a una posición que genera beneficios cuando aumenta el precio de mercado de un activo. Por regla general, se habla de abrir una posición larga o de ir largo.

Liquidez.- Una medida de mercados que describe la relación entre el volumen de trading y el cambio del precio. Cuando no hay volumen en el mercado y no se está negociando, decimos que existe poca liquidez.

Lote.- La cantidad de unidades o la cantidad de dinero aceptado para manejar las operaciones (normalmente es un múltiplo de 100). Un lote estándar son 100.000 unidades de la divisa base del par

Media móvil.- Uno de los indicadores técnicos más básicos. Muestra un precio medio calculado durante un periodo de tiempo. Medias móviles exponenciales (MME), medias móviles ponderadas (MMP), etc. son sólo formas de ponderar los precios y los períodos.

Metatrader.- Es una plataforma muy utilizada para hacer trading especializada en el mercado de divisas.

Orden.- Una orden es una disposición del trader al bróker a realizar una operación de trading. Como su propio nombre indica, una orden de compra o venta de un activo. Cuando se lleva a cabo es se convierte en una operación.

Orden a mercado.- Es una orden que enviamos al bróker para comprar o vender una cantidad a el precio de mercado.

Orden limitada.- Este tipo de orden se lleva a cabo para comprar la cantidad por un precio fijo o un precio más bajo o vender la cantidad por un precio fijo o un precio mejor. En otras palabras, es una orden que enviamos para comprar como máximo o vender como mínimo a un determinado precio.

Orden stop.- Las utilizamos para comprar a un determinado precio o superior o vender a un precio o inferior. Las órdenes stop son un tipo de orden con las que puede indicar a su bróker que ejecute una operación cuando se alcance un nivel determinado menos favorable que el precio de mercado actual.

Paridad.- El término paridad en forex se refiere a una relación de igualdad. Por ejemplo, decimos que EUR/USD cotiza cerca de su paridad si la cotización de EUR/USD está cerca de 1.

Pérdida.- Es el resultado de cerrar una posición de compra a un precio inferior al que compramos o una de venta a un precio superior. En definitiva, un resultado negativo. Cuidado, porque se puede originar aún cuando hayamos hecho una buena operación pero el coste sea muy elevado, por ejemplo cuando mantenemos una posición durante mucho tiempo con swap negativo.

Pip.- Con un pip nos referimos al movimiento más pequeño de un cruce de divisas (por ejemplo, en EUR/USD, 1 pip (punto) = 0,0001).

Posición abierta.- Las posiciones abiertas son operaciones que se han realizado y que aún pueden generar ganancias o pérdidas. Cuando se cierra una posición, se recogen todas las ganancias o pérdidas y la operación deja de estar activa.

Posición cerrada.- Una posición está cerrada cuando todas las transacciones necesarias se han cerrado.

Precio de mercado.- El precio al que la divisa se está negociando en el mercado.

Rango.- El rango es la diferencia entre el precio máximo y el precio mínimo que existe en un activo dentro de un periodo de tiempo determinado. Principalmente, se utiliza como indicador de volatilidad, si un activo tiene un rango amplio durante un periodo, significa que su volatilidad también lo fue durante ese periodo.

Resistencia.- Una resistencia es un nivel del precio en que la venta intensiva puede llevar un aumento de precio.

Scalping.- Es un estilo de trading que se basa en abrir y cerrar una posición en muy poco tiempo, con el objetivo de obtener beneficios en movimientos de precios muy reducidos.

Soporte.- Un nivel del precio en que la compra intensiva puede llevar a un decrecimiento del precio.

Spread.- El spread es la diferencia entre precios de oferta y demanda de un activo.

Stop loss.- Se utiliza para evitar pérdidas adicionales si el mercado se mueve en la dirección opuesta. Se trata de una orden para vender o comprar una cantidad cuando el mercado se alcanza un cierto precio.

Swap.- Un pago o cobro por mantener una posición abierta durante la noche. Puede ser a favor o en contra en función a la diferencia de las tasas de interés entre las dos monedas del par.

Take profit.- Se utiliza para cerrar los beneficios de las operaciones abiertas cuando el mercado se mueve en la dirección de su posición. Es una orden para vender o comprar una cantidad cuando el mercado se alcanza un cierto precio.

Tendencia.- Una dirección en la que se mueve un activo o mercado (alcista, bajista o en rango).

Tipos de interés.- Un porcentaje que los bancos centrales usan para prestar a los bancos comerciales de país.

Trader.- Los traders son quienes compran y venden en el mercado, es decir, quienes hacen trading.

Trailing stop loss.- Es un stop loss móvil. Un nivel de stop loss que se va acercando al precio de mercado actual a medida que disminuye la pérdida de la posición o aumenta su ganancia.

Volatilidad.- Una medida estadística del número de los cambios del precio en el par de monedas dada por un período de tiempo dado.

Volumen.- En el trading, el volumen es la cantidad de un activo determinado que se realiza durante un período de tiempo determinado.

Scrum.- Un marco de trabajo para la gestión y desarrollo de software basada en un proceso iterativo e incremental utilizado comúnmente en entornos basados en el desarrollo ágil de software.

Sprint.- Período en el cual se lleva a cabo el trabajo en sí, dentro de la metodología Scrum.

Template.- Plantilla o vista HTML dentro de Django.

UML.- Lenguaje Unificado de Modelado.

URL.- Localizador de recursos uniforme, más comúnmente denominado URL, es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet para su localización o identificación.

XP.- Programación Extrema.