



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Aplicación para la trazabilidad de operaciones en el  
mercado de valores

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Manolescu, Silviu Valentin

Tutor/a: Busquets Mataix, José Vicente

CURSO ACADÉMICO: 2022/2023



## Resumen

---

La bolsa de valores es un mercado donde se negocian multitud de activos financieros, pero implica riesgos debido a la volatilidad a la hora de operar con estos. Los inversores buscan mantener un control periódico sobre las operaciones que realizan mediante el análisis y la implementación de diferentes estrategias para minimizar pérdidas. La monitorización constante de los activos es crucial para tomar decisiones informadas, adaptando las carteras a las condiciones cambiantes del mercado con el fin de optimizar el rendimiento y disminuir el riesgo. Para facilitar este control y supervisión, se ha desarrollado una aplicación en Python que permite la trazabilidad de las operaciones en la bolsa, trasladando la información desde el bróker de Interactive Brokers hasta la plataforma de gestión de carteras Investing.com. Esta labor, que implica transferir los datos de posiciones abiertas y el historial de transacciones de una plataforma a otra, puede ser un proceso prolongado y tedioso que a menudo desincentiva al usuario a realizarla de forma manual. Sin embargo, esta herramienta facilita al inversor mantener su cartera actualizada en la nueva plataforma y, además, gestionar dos carteras adicionales diseñadas para estrategias de inversión a largo y corto plazo de una forma sencilla y automática.

**Palabras clave:** Bigdata; Bolsa de valores; Python

## Resum

---

La borsa de valors és un mercat on es negocien multitud d'actius financers, però implica riscos a causa de la volatilitat a l'hora d'operar amb aquests. Els inversors busquen mantindre un control periòdic sobre les operacions que realitzen mitjançant l'anàlisi i la implementació de diferents estratègies per a minimitzar pèrdues. El monitoratge constant dels actius és crucial per a prendre decisions informades, adaptant les carteres a les condicions canviants del mercat amb la finalitat d'optimitzar el rendiment i disminuir el risc. Per a facilitar aquest control i supervisió, s'ha desenvolupat una aplicació en Python que permet la traçabilitat de les operacions en la borsa, traslladant la informació des del bróker de Interactive Brokers fins a la plataforma de gestió de carteres Investing.com. Aquesta labor, que implica transferir les dades de posicions obertes i l'historial de transaccions d'una plataforma a una altra, pot ser un procés prolongat i tediós que sovint desincentiva a l'usuari a realitzar-la de manera manual. No obstant això, aquesta eina facilita a l'inversor mantindre la seua cartera actualitzada en la nova plataforma i, a més, gestionar dues carteres addicionals dissenyades per a estratègies d'inversió a llarg i curt termini d'una forma senzilla i automàtica.

**Paraules clau:** Bigdata; Borsa de valors; Python

# Abstract

---

The stock market is a market where a multitude of financial assets are traded, but it involves risks due to the volatility when operating with them. Investors seek to maintain periodic control over the operations they carry out by analyzing and implementing different strategies to minimize losses. Constant monitoring of assets is crucial to making informed decisions, adapting portfolios to changing market conditions in order to optimize performance and reduce risk. To facilitate this control and supervision, an application has been developed in Python that allows the traceability of operations on the stock market, transferring the information from the Interactive Brokers to the Investing.com portfolio management platform. This task, which involves transferring open position data and transaction history from one platform to another, can be a lengthy and tedious process that often discourages the user from doing it manually. However, this tool makes it easier for the investor to keep their portfolio updated on the new platform and, in addition, manage two additional portfolios designed for long and short-term investment strategies in a simple and automatic way.

**Keywords:** Bigdata; Stock market; Python



# Índice general

---

1. Introducción .....	8
1.1 Motivación .....	11
1.2 Objetivos .....	12
1.3 Impacto esperado.....	12
1.4 Metodología .....	13
1.5 Estructura de la memoria .....	14
2. Estado de la técnica .....	16
2.1 Bróker .....	16
2.1.1 Interactive Brokers .....	16
2.1.2 Alternativas bróker .....	18
2.2 Análisis y gestor de portfolio.....	18
2.2.1 Investing.com.....	18
2.2.2 Alternativas gestor portfolio.....	19
2.3 Python.....	19
2.3 Crítica al estado del arte.....	20
2.4 Propuesta.....	21
3. Funcionamiento y entorno de la bolsa .....	22
3.1 La bolsa de valores: concepto y funcionamiento .....	22
3.2 Instrumentos financieros: clases y características .....	24
3.3 Operaciones en largo y corto.....	25
3.4 Monitorización para operar en bolsa.....	27
3.5 Lógica de las carteras .....	28
3.5.1 Funcionamiento Cartera 1.....	28
3.5.2 Funcionamiento Cartera 2.....	29
3.5.3 Funcionamiento Cartera 3.....	30
4. Análisis del problema.....	32
4.1 Análisis de seguridad .....	32
4.2 Análisis energético .....	33
4.3 Análisis del marco legal y ético.....	34
4.4 Análisis de riesgos .....	34
4.5 Estructura de las plataformas .....	35
4.5.1 Análisis plataforma Interactive Brokers .....	35

4.5.2 Análisis plataforma Investing.com .....	36
4.6 Solución propuesta.....	43
4.7 Plan de trabajo .....	46
5. Diseño .....	48
5.1 Iniciación carteras .....	48
5.2 Actualizar carteras.....	49
6. Implementación .....	55
6.1 Conexión a la API de IBKR.....	55
6.2 Guardado de los datos en CSV - Pandas.....	59
6.3 Automatización acceso a Investing.....	61
6.3.1 Automatización ventanas .....	61
6.3.2 Selenium.....	62
6.3.3 Tarea diaria de acceso a Investing .....	63
6.5 Estructura scripTask.py .....	64
6.6 Configurar programa por primera vez.....	64
7. Test y pruebas.....	70
7.1 Pruebas unitarias .....	70
7.2 Prueba de integración .....	71
7.3 SonarQube.....	73
8. Conclusiones .....	75
8.1 Aprendizaje personal.....	75
8.2 Relación del trabajo desarrollado con los estudios cursados.....	76
8.3 Errores y dificultades.....	76
8.4 Trabajo futuro .....	77
9. Bibliografía.....	78
ANEXO A: Detalles de la Trader Workstation .....	82
A.1 Instalación y configuración Trader Workstation .....	82
A.2 Operar con Trader Workstation.....	83
A.3 Detalles de la obtención de los datos de la cuenta.....	84
ANEXO B: ODS .....	85

# Índice de tablas

---

<b>Tabla 3.1:</b> Funcionamiento de la Cartera 1 .....	29
<b>Tabla 3.2:</b> Funcionamiento de la Cartera 2.....	30
<b>Tabla 3.3:</b> Funcionamiento de la Cartera 3.....	31
<b>Tabla 4.1:</b> Análisis de riesgos identificados .....	34
<b>Tabla 7.1:</b> Clases de pruebas creadas .....	71
<b>Tabla 7.2:</b> Contenido historial transacciones para la prueba .....	72
<b>Tabla 7.3:</b> Contenido Cartera1 para la prueba.....	72
<b>Tabla 7.4:</b> Resultados esperados en la prueba .....	73

# Índice de figuras

---

<b>Figura 1.1:</b> Visualización cartera en Interactive Brokers.....	9
<b>Figura 1.2:</b> Visualización cartera en Investing.com .....	10
<b>Figura 3.1:</b> Ejemplo operación en largo.....	26
<b>Figura 3.2:</b> Ejemplo operación en corto.....	26
<b>Figura 4.1:</b> Página principal Investing.com.....	36
<b>Figura 4.2:</b> Página principal Investing.com - Otros elementos.....	37
<b>Figura 4.3:</b> Pestaña “My Watchlist” de Investing.com.....	37
<b>Figura 4.4:</b> Creación de una cartera.....	38
<b>Figura 4.5:</b> Importar cartera - Paso 1 .....	39
<b>Figura 4.6:</b> Importar cartera - Paso 2 .....	39
<b>Figura 4.7:</b> Importar cartera - Paso 3 .....	39
<b>Figura 4.8:</b> Añadir una nueva posición.....	40
<b>Figura 4.9:</b> Introducir datos nueva posición.....	40
<b>Figura 4.10:</b> Ejemplo cartera de inversiones .....	41
<b>Figura 4.11:</b> Editar una posición abierta.....	42
<b>Figura 4.12:</b> Cerrar una posición abierta.....	42
<b>Figura 4.13:</b> Gráfico de evolución activo .....	43
<b>Figura 4.14:</b> Diagrama de flujo ejecución diaria .....	44
<b>Figura 5.1:</b> Diagrama de flujo iniciación carteras.....	48
<b>Figura 5.2:</b> Diagrama de flujo procesamiento Cartera 1 .....	50
<b>Figura 5.3:</b> Diagrama de flujo procesamiento Cartera 2 .....	52
<b>Figura 5.4:</b> Diagrama de flujo procesamiento Cartera 1 - Modificada.....	53
<b>Figura 5.5:</b> Diagrama de flujo procesamiento Cartera 3 .....	54
<b>Figura 6.1:</b> Aspecto portfolio.csv .....	60
<b>Figura 6.2:</b> Aspecto transactionsXXX.csv.....	60
<b>Figura 6.3:</b> Aspecto registry.csv .....	61
<b>Figura 6.4:</b> Panel de acciones Programador de tareas.....	65
<b>Figura 6.5:</b> Crear una tarea básica.....	66
<b>Figura 6.6:</b> Definir frecuencia tarea .....	66
<b>Figura 6.7:</b> Tipo de acción a elegir .....	67
<b>Figura 6.8:</b> Localización archivo a ejecutar .....	67
<b>Figura 6.9:</b> Parámetros adicionales - Parte 1 .....	68
<b>Figura 6.10:</b> Parámetros adicionales - Parte 2 .....	69
<b>Figura 7.1:</b> Resultados obtenidos en SonarQube.....	74



---

# 1. Introducción

---

En el ámbito financiero, las inversiones y la bolsa de valores juegan un papel esencial en la generación de riqueza y el impulso económico. A lo largo de la historia, muchas personas han aprovechado la oportunidad de invertir en acciones y otros activos financieros para incrementar su patrimonio y alcanzar sus objetivos económicos.

Las inversiones permiten asignar recursos financieros disponibles con el propósito de obtener un rendimiento. Una de las modalidades más populares de inversión es la bolsa de valores, donde se negocian acciones de empresas y otros instrumentos financieros como bonos y opciones [1]. Las acciones representan una parte de la propiedad de una empresa y ofrecen a los inversores la posibilidad de beneficiarse tanto del crecimiento de la empresa como de sus ganancias. Por ello, el valor de las acciones está relacionado con el desempeño de la empresa. Un negocio que reporta buenos beneficios muestra perspectivas favorables o mejora su imagen tenderá a incrementar el valor de sus acciones [2] [3]. Sin embargo, el efecto contrario puede suceder si la empresa muestra una tendencia negativa sostenida.

Estas fluctuaciones en el valor no solo provocan posibilidades de ganar dinero, también de que los inversores pierdan, generando así un posible riesgo a la hora de operar [4]. Por este motivo, los inversores toman decisiones con precaución al comprar y vender acciones y buscan gestionar adecuadamente sus carteras de inversión. Esto conlleva monitorizar y evaluar periódicamente el estado de los activos, realizar ajustes cuando sea necesario y mantener una estrategia de inversión coherente.

Este trabajo final de grado trata de aportar una herramienta software que permita al inversor una mejor supervisión de su cartera de valores. No se trata pues de ningún robot de inversión automática ni de un programa de análisis de estrategias de inversión con datos históricos.

Para la monitorización de la cartera, los brókers proporcionan unas tablas donde se refleja para cada valor en cartera el número de títulos, el precio de compra, el precio actual, el porcentaje variación, el valor de la inversión y el valor actual entre otros. Pero todo ello en tiempo real y para el instante actual.

Sin embargo, el inversor-ahorrador muchas veces vende (infraponderar) o compra (sobreponderar) una parte del total de acciones que ostenta de un determinado valor. Por ejemplo, cuando hay un máximo con indicadores de sobrecompra, se puede vender una parte para cuando haya un ajuste a la baja, recomprar de nuevo las acciones vendidas a un mejor precio. Para esta operativa, es indispensable saber a qué precio se vendieron las acciones. Y aquí radica la novedad de la herramienta aportada en este proyecto. Se han diseñado dos carteras virtuales derivadas de la real que nos ayudan a monitorizar esos lotes de acciones “vendidas” para saber si se ha reducido su precio y en qué porcentaje para dar argumentos de decisión de cuando se puede recomprar esas acciones vendidas con anterioridad. Son dos carteras virtuales porque una es para las ventas y otra para las compras como se explicará más adelante.

Las herramientas de visualización que proporcionan los brókers no permiten introducir estas carteras (o portafolios) virtuales, ya que solo nos ofrecen anotar listas de seguimiento (*watchlist*) que no es lo mismo que una cartera donde se reflejan las operaciones de compra y venta.

Por este motivo es necesario trasladar la cartera real y las dos virtuales a otra plataforma que nos permita introducir carteras para su monitorización. Todo este traslado y su algorítmica es el trabajo a realizar por el proyecto.

Para las operaciones reales se emplea Interactive Brokers, también conocido como IBKR, un bróker que actúa como intermediario financiero para facilitar la compra y venta de activos en nombre de los inversores [5]. Con una reputación sólida en la industria, IBKR destaca como líder en servicios de corretaje en línea, ofreciendo a los inversores acceso a una amplia gama de mercados y productos financieros a través de una plataforma robusta como en la **Figura 1.1** [6].

PYG ...	INST. FIN.	PSCÓN	VLR D	MRC D	PRC MD	ÚLTIMO	VARIAC...
-342	AAPL	50	9,143	190.12 <sup>3</sup>	182.91	-6.79	
-75	AMZN	55	7,475	130.25	135.88	-1.39	
-71	GOOG	55	7,448	119.10 <sup>2</sup>	135.48	-1.23	
-72	TSLA	14	3,519	238.03	251.40	-5.09	
-62	LYFT	71	779	10.44	10.98	-0.86	
-28	META	43	12,879	290.04 <sup>1</sup>	299.56	-0.59	
-6	MSFT	12	3,996	335.14 <sup>6</sup>	333.17	-0.38	
-6	BBVA	40	301	7.70 <sup>8</sup>	7.51	-0.16	
0	FORD	15	12	0.9307 <sup>6</sup>	0.8100	-0.0320	
EUR EFECTIVO			1,002,647				
USD EFECTIVO			-43,525				

*Figura 1.1: Visualización cartera en Interactive Brokers*

Por el otro lado, para la monitorización de las carteras optaremos por la plataforma líder en el panorama actual: Investing.com. Es una plataforma que ofrece cotizaciones en tiempo real, gráficos interactivos y noticias del mercado, además de proporcionar una gran cantidad de información adicional para tomar decisiones informadas a la hora de operar [7]. Nos brinda la posibilidad de confeccionar un número ilimitado de listas de seguimiento y lo que nos interesa, carteras virtuales, como en la **Figura 1.2**.

Añadir posición

Posiciones Transacc. Gráficos Lo más popular Estadísticas

Resumen	Precio	Fundamental	Técnico	Ejecución	Patrones de velas	Personalizar			
Nombre	Símbolo	Tipo	Cantidad	Precio medio	Precio actual	Val. mercado	B/P diario	% B/P neto	B/P neto
Tesla	TSLA	Compra	36	238,34	251,92	9.069,12 \$	-164,51 \$	5,69%	489,02 \$
LYFT	LYFT	Compra	73	10,49	10,97	800,81 \$	-63,51 \$	4,56%	34,94 \$
Amazon.com	AMZN	Compra	56	130,60	135,36	7.580,16 \$	-106,96 \$	3,64%	266,57 \$
Apple	AAPL	Compra	52	189,64	182,91	9.511,32 \$	-363,07 \$	-3,54%	-349,72 \$
Microsoft	MSFT	Compra	12	335,15	332,88	3.994,56 \$	-8,03 \$	-0,67%	-27,19 \$
Meta Platforms	META	Compra	42	289,95	299,17	12.565,14 \$	-41,15 \$	3,18%	387,38 \$
Alphabet C	GOOG	Compra	55	119,10	135,37	7.445,35 \$	-73,69 \$	13,65%	894,77 \$
Forward Industries	FORD	Compra	15	0,930760	0,821800	12,33 \$	-0,30 \$	-11,70%	-1,63 \$
BBVA ADR	BBVA	Compra	40	7,707	7,520	300,80 \$	-6 \$	-2,42%	-7,49 \$

Figura 1.2: Visualización cartera en Investing.com

Con esto en mente, a través de la interfaz de Interactive Brokers se procederá descargar los datos de interés del inversor para poder trasladarlos hasta la segunda plataforma, Investing.com. En esta última se usará la librería de Selenium para automatizar el uso de la página web y así poder llevar a cabo el seguimiento de la diferentes estrategias de inversión mediante la creación de 3 carteras [8]:

- **Cartera 1:** se trata de un reflejo real de la cartera que tiene en posesión actualmente el inversor. Esto quiere decir que podrá consultar las posiciones abiertas que tiene en ese momento y poder visualizar datos como la valoración de la cartera, distribución de los activos, rendimiento...
- **Cartera 2:** cartera enfocada en hacer un seguimiento de las operaciones en largo, donde se compran activos de los cuales se espera que aumenten su valor en el tiempo, generando ganancias por el incremento del precio. En esta cartera se registra la cantidad de cada activo vendido de la Cartera 1, con el propósito de que el inversor pueda al cabo de un tiempo recomprar la acción en el futuro. Esta cartera no es un reflejo de activos en posesión, se anotan posiciones ficticias simplemente para recordar a que precios de venta se cerraron las anteriores.
- **Cartera 3:** en esta última en cambio se hace un seguimiento de las operaciones en corto, es decir, vender un activo que se ha tomado prestado con la expectativa de recomprarlo a un precio más bajo en el futuro, generando así ganancias de la caída del precio del activo. Al igual que la anterior, esta cartera no es un reflejo de activos en posesión, se anotan posiciones ficticias simplemente para recordar a que precios de venta se cerraron las anteriores.

De este modo, las inversiones y la bolsa de valores desempeñan un papel clave en la creación de riqueza. La gestión cuidadosa de los portafolios de inversión y el acceso a plataformas como Investing.com brindan a los inversores herramientas para tomar decisiones informadas y realizar un seguimiento efectivo de sus inversiones. Mediante la automatización es posible optimizar las estrategias de inversión, aumentando las posibilidades de obtener resultados positivos en las carteras y facilitar la vida al inversor.

## 1.1 Motivación

La tarea de gestionar de forma efectiva los portafolios de inversiones no es algo fácil. El hecho de tener que hacer un seguimiento para cada uno de los activos que se tienen en posesión es un proceso tedioso y que dependiendo de la cantidad de diferentes acciones puede llevar bastante tiempo. Esto se relaciona con la diversificación que juega un papel necesario en el proceso de gestión, ya que ayuda a mitigar parte del riesgo que se asume al comprar una acción, distribuyéndolo entre varias diferentes [9]. Por tanto, se puede llegar a perder mucho tiempo si se tienen diferentes activos, realizando tareas repetitivas y que realmente si se pudiese aprovechar en otras actividades se acabaría generando un mayor valor. Adicionalmente, el sector de la bolsa cada vez más goza de una mayor cantidad de aplicaciones y servicios informáticos que anteriormente no existían, por ello intentar aportar una nueva solución que pueda servir a determinados inversores que se encuentren en una misma situación descrita en este Trabajo Final de Grado es una gran motivación.

Por otro lado, desde un enfoque más personal y habiendo estudiado simultáneamente la carrera de Administración y Dirección de Empresas, he desarrollado un creciente interés por el mundo de las inversiones. Este proyecto representa una oportunidad adicional para sumergirme en este ámbito y aprender cómo utilizar la programación para abordar las necesidades identificadas. Dado que los inversores dedican mucho tiempo a tareas que, en esencia, podrían automatizarse, me impulsa a ofrecer una solución útil.

## 1.2 Objetivos

El objetivo general de este trabajo es desarrollar una aplicación que permita una trazabilidad de las operaciones en el mercado de valores de una forma sencilla y automática que facilite las tareas de control y supervisión a los inversores. Los objetivos específicos se desglosan de la siguiente forma:

1. Desarrollar una aplicación empleando Python que permita supervisar y monitorizar las operaciones en el mercado de valores llevando los datos de un bróker a una plataforma de gestión del portafolio.
2. Automatizar la ejecución de forma periódica de la aplicación para que el usuario tenga la menor implicación y toda la información en el gestor de portafolio esté al día.
3. Posibilitar en el gestor de portafolios el uso de las estrategias de operaciones en corto y largo.
4. Facilitar el código para que se pueda utilizar por otros posibles usuarios a través de un repositorio público.

## 1.3 Impacto esperado

A través del desarrollo de la aplicación de trazabilidad de este proyecto se pretende brindar una solución lo más automatizada posible para que el usuario tenga la menor preocupación. De esta forma los inversores, usuario principal para el que se dirige la solución, obtendrán las siguientes ventajas:

- Menor tiempo empleado en tareas repetitivas para la monitorización del portafolio.
- Mejor seguimiento de las operaciones de compra y venta de acciones.
- Control más sencillo del gestor de portafolio.
- Poder seguir operaciones de venta en largo y corto de forma automática.
- Automatización del control de operaciones con una mínima intrusión en el sistema operativo.
- Asentar bases para dar la posibilidad de introducir nuevas estrategias de inversión.
- Facilitar acceso a la información del rendimiento del portafolio actualizado.

## 1.4 Metodología

La metodología empleada en este Trabajo final de grado se basa en iteraciones sobre las distintas funcionalidades a implementar [10]. Para cada una se siguen una serie de pasos que repiten hasta la finalización del proyecto, asegurando un flujo de trabajo estructurado y planificado en todo momento. Los pasos son los siguientes:

1. **Toma de contacto:** en esta primera fase se busca comprender de forma básica los conceptos a desarrollar más adelante para entender el grado de complejidad y la cantidad de pasos a desarrollar.
2. **Planificación:** se establecen los objetivos a seguir para lograr alcanzar la tarea propuesta. Esta fase es crucial puesto que se determinará una estimación aproximada del tiempo a dedicar.
3. **Aprendizaje:** se adquiere conocimiento y comprensión en profundidad sobre la tarea a desarrollar, desde aspectos teóricos como prácticos. Se realiza una investigación sobre los aspectos más relevantes, se identifican las mejores prácticas y se estudian proyectos con objetivos similares para obtener ideas y referencias.
4. **Alternativas:** a menudo para llegar al éxito de los objetivos se pueden alcanzar de diferentes formas y con distintas soluciones. Se exploran todas las opciones y enfoques para abordar la tarea, considerando los pros y contras de cada opción. El objetivo de esta fase es seleccionar la alternativa más adecuada para llegar a la meta cumpliendo con los objetivos.
5. **Desarrollo e implementación:** se lleva a cabo la ejecución, traduciendo los planes en acciones. En esta etapa práctica y activa se construye la solución y se integran los componentes necesarios.
6. **Pruebas:** gracias a este paso se asegura la calidad y la correcta ejecución del proyecto. Se busca detectar errores y casos no implementados, así como que el proyecto cumple con los requisitos establecidos para posibilitar implementar las correcciones necesarias antes de la entrega.
7. **Mejoras:** a través de múltiples ejecuciones del programa se detectan las deficiencias y áreas de mejora para garantizar la mayor calidad y eficiencia posible. Realmente se trata de optimizar y perfeccionar las distintas dimensiones que constituyen el componente.
8. **Documentación:** empleando la herramienta de planificación Trello paralelamente se irá redactando las soluciones aportadas, enfoques adoptados y diagramas de flujo de cada proceso en el orden de ejecución.



## 1.5 Estructura de la memoria

La estructura del presente Trabajo Final de Grado cuenta con unos apartados similares a otros trabajos de la ETSINF, ya que se desarrolla un programa informático.

En primer lugar, en este mismo [capítulo 1](#) de Introducción se explican las bases junto al enfoque que se planea desarrollar a lo largo de todo el proyecto. De esta forma va acompañado de la motivación, el impacto esperado y con la metodología que se ha empleado para tener un flujo de trabajo estructurado.

En el [capítulo 2](#) se tratan las diferentes alternativas para elegir el bróker y gestor de portafolios adecuados, ya que son los elementos centrales alrededor de los cuales se va a crear el programa. Asimismo, se toma como ejemplo otros proyectos elaborados en la ETSINF para ver si pueden servir como fuente de información y realizar aportaciones más elaboradas.

Continuando con el [capítulo 3](#), se explican las bases de la bolsa de valores y como operan los inversores para comprender el funcionamiento de todo este complejo mecanismo. Es indispensable entender este apartado al emplear en futuros capítulos términos de forma constante relativos a operar con instrumentos financieros.

Para el [capítulo 4](#) se analiza el problema desde todas sus perspectivas, así como los contratiempos que puede llegar a presentar. Adicionalmente, se hace un análisis minucioso de la plataforma Investing.com para plantear una solución general al problema abordado y poder concretar los diferentes componentes en el siguiente [capítulo 5](#) de Diseño. En este a través de múltiples diagramas de flujo se detalla el funcionamiento de los elementos centrales del programa que se plantea elaborar.

En la implementación del siguiente [apartado 6](#) se pone en marcha todos los conceptos previamente detallados para obtener los primeros resultados. De la misma forma, se muestran aspectos interesantes que han condicionado esta solución y cómo funcionan determinados componentes. Por otra parte, antes de terminar se incluirán las pruebas necesarias para asegurar la calidad del programa en el [capítulo 7](#) y su correcto funcionamiento, acabando mediante unas [conclusiones](#) recogidas de todo el proyecto.

En los anexos se recogerá unos capítulos adicionales para explicar ciertos aspectos interesantes del código elaborado, así como los Objetivos de Desarrollo Sostenible que han sido impulsados gracias a esta aportación.





---

## 2. Estado de la técnica

---

En este apartado se expondrán los motivos por el cual se han escogido las plataformas de Interactive Brokers e Investing.com, dado que hay otras alternativas que también podrían haber sido factibles. Adicionalmente, se va a indagar en otros proyectos con funcionalidades iguales o parecidas ya realizados para ver las aportaciones y como este Trabajo Final de Grado podría encaminarse en traer nuevas soluciones.

### 2.1 Bróker

Como ya se ha comentado previamente en la introducción de este documento, es imprescindible tener al alcance un bróker. Este término hace referencia a la entidad que facilita la compra y venta de instrumentos financieros en nombre de sus clientes, es decir, un intermediario. Su función principal es actuar de puente entre inversores y los mercados financieros, llevando a cabo las órdenes de compra y venta, además de poner al alcance servicios y asesoramiento relacionados. Por último, la regulación y la seguridad son aspectos clave en el funcionamiento de un bróker, ya que deben cumplir ante las autoridades financieras de cada país para asegurar los intereses de los clientes [11]. Asimismo, la elección del bróker no solo debe estar determinada por la cantidad de servicios y ventajas al inversor, también es necesario poder acceder a una interfaz de programación de aplicaciones, o API por sus siglas en inglés, de forma gratuita para la realización de este trabajo.

#### 2.1.1 Interactive Brokers

Interactive Brokers es una reconocida empresa de corretaje y servicios financieros estadounidense y hoy en día es una de las más reconocidas en el ámbito de comercio de valores. Lo que diferencia a Interactive Brokers de las demás es el enfoque electrónico y la ejecución de órdenes rápidas, además de ofrecer otros servicios y productos para sus clientes. Entre los consumidores de IBKR se encuentran principalmente: inversores minoristas, *traders* activos, *hedge funds*, instituciones financieras y corredores. Comprendiendo ya la base de esta plataforma y su alcance ahora es momento de profundizar en las características y ventajas que ofrece, ya que muchas de estas son el motivo por el cual se ha escogido.

Una de las ventajas más destacables es su plataforma de negociación, llamada Trader Workstation (TWS) [12]. Se trata de una potente herramienta que ofrece la posibilidad de poder acceder a variados mercados financieros, con la opción de acciones, opciones, bonos, futuros, divisas y muchas más. Esta plataforma proporciona cotizaciones en tiempo real, numerosos gráficos interactivos, herramientas de análisis técnico, órdenes avanzadas, noticias financieras y la posibilidad de implementar

estrategias complejas de trading. Por otra parte, Interactive Brokers no solo ofrece TWS, la IBKR Web Trader se consolida como una muy buena alternativa con una interfaz basada en web para acceder a la misma cuenta sin ninguna necesidad de descargar e instalar ningún software. Asimismo, se han expandido los servicios hasta las aplicaciones móviles para poder operar con dispositivos de todo tipo y en cualquier lugar. En cuanto al aspecto más relacionado con la programación IBKR cuenta con una API, dividida en 3 modalidades [\[13\]](#):

- **API de Client Portal:** se trata de una REST API que tiene la capacidad de negociar, controlar y gestionar las operaciones de la cuenta.
- **API de Trader Workstation (TWS):** como se menciona en la página web se trata de una interfaz con muchas funcionalidades que “automatiza sus estrategias de negociación, solicitud de datos de mercado, y controla los saldos de cuenta y cartera en tiempo real”. Se incide de muchas veces que a pesar de ser bastante simple tiene una gran potencia que la convierte en ideal para el desarrollo de aplicaciones en diferentes lenguajes.
- **FIX:** enfocada a instituciones para crear sistemas de negociación avanzados.

Otro aspecto destacable, que no afecta en gran parte a este trabajo, pero justifica la popularidad de la plataforma es la ejecución de órdenes de bajo costo. Interactive Brokers ofrece comisiones competitivas e incluso descuentos para operadores frecuentes, convirtiéndolo así en una muy buena alternativa de mercado al minimizar los costos de transacción. De igual importancia, a pesar de los bajos costos permite a los consumidores acceder a una amplia gama de mercados globales, lo que posibilita a su vez una mejor diversificación de las carteras al operar en diferentes regiones del mundo.

En cuanto a la regulación y la seguridad, otro punto importante en los aspectos de un bróker, IBKR está sujeto a la supervisión de diversas autoridades financieras y entre estas se encuentran la Comisión de Valores y Bolsa de Estados Unidos (SEC) y la Comisión Nacional del Mercado de Valores (CNMV). Asimismo, los clientes gozan de una seguridad en sus fondos ya que están protegidos bajo los programas de seguros de la Securities Investor Protection Corporation (SIPC) y, adicionalmente por una póliza proporcionada por Interactive Brokers [\[14\]](#).

Por último, la mayor ventaja de Interactive Brokers sin duda alguna es el servicio de *Paper Trading*. Este término hace referencia al trading simulado que se realiza a través de la misma plataforma permitiendo a los usuarios practicar sus estrategias de inversión sin tener que arriesgarse en ningún momento, es decir, totalmente gratis. Esta opción es ideal especialmente para los principiantes que se están iniciando en el mundo de las inversiones y quieren experimentar.

En resumen, Interactive Brokers es una plataforma líder en servicios de corretaje que la convierten en una ideal elección como bróker no solo por su diversidad de servicios, también por sus bajos costes y posibilidad de desarrollo de aplicaciones que funcionen en paralelo con la plataforma.

### 2.1.2 Alternativas bróker

A parte de Interactive Brokers se ha investigado también otras plataformas reconocidas. Sin duda alguna, se trata de un sector muy competitivo en el que ofrecer la mayor cantidad de servicios, el mayor número de mercados accesibles y manteniendo los menores costes son el objetivo de todas estas. Como factor excluyente se ha vuelto a considerar la falta de API, aquellas que no ofrezcan suficientes funcionalidades.

De esta forma, según diversas páginas de rankings se han encontrado brókers como: Alpaca, Tradier, TD Amerite, Freedom 24, AvaTrade... [\[15\]](#) [\[16\]](#) Sin embargo, como era de esperar Interactive Brokers es una plataforma bastante más potente y con mayor profundidad. Si se toma Alpaca como ejemplo, esta dispone de menor número de mercados y productos. Todas las demás opciones se han descartado por motivos similares o incluso por falta de una comunidad de desarrolladores que ayude a ver el potencial de las API. Por lo tanto, se ha elegido entre otros motivos a Interactive Brokers al ser una opción robusta, con una comunidad de desarrolladores activa, gran número de servicios y la posibilidad de una cuenta gratuita.

## 2.2 Análisis y gestor de portfolio

Con el bróker elegido ya solo queda una plataforma que posibilite hacer una adecuada gestión del portfolio y que proporcione una gran cantidad de información. Como ya se ha comentado, es crucial tomar decisiones informadas y por ello es necesario tener al alcance noticias, rendimientos de las carteras, indicadores, gráficos y cualquier otro elemento que minimice el riesgo al operar.

### 2.2.1 Investing.com

Es un sitio web de información financiera y de inversión ampliamente reconocido y utilizado por inversores de todo el mundo. En esta plataforma se ofrecen una gran cantidad de servicios que son de gran utilidad a la hora de realizar un seguimiento de los activos. Entre aquellos que son más relevantes se encuentran [\[17\]](#):

- **Acceso a datos y cotizaciones en tiempo real:** los usuarios pueden acceder a los datos de interés en todo momento y siempre estarán actualizados para seguir los movimientos del mercado en tiempo real.
- **Calendario económico:** la plataforma muestra en un calendario eventos y anuncios de gran interés que ocurrirán en las próximas fechas. Esta funcionalidad es de gran utilidad a los inversores puesto que determinados acontecimientos pueden alterar los mercados.

- **Herramientas seguimiento:** Investing.com pone al alcance herramientas de análisis técnicos con indicadores, gráficos interactivos y también comparadores de instrumentos.
- **Noticias:** gran cantidad de noticias, análisis y artículos de muchas fuentes para estar actualizado en las últimas tendencias y sucesos.

### 2.2.2 Alternativas gestor porfolio

Se han buscado más alternativas aparte de Investing.com. En primer lugar, Sharesight es también otra plataforma de gestión de porfolio muy potente y con una interfaz muy práctica para el análisis del rendimiento, sin embargo, se trata de un servicio de pago que se ha descartado de entrada [\[18\]](#).

Google Finance y Yahoo Finance son otras alternativas que también se han contemplado, pero después de navegar un poco en las plataformas se ha visto que no llegan al mismo nivel de profundidad que Investing en cuanto a análisis del rendimiento de los portafolios [\[19\]](#).

Cabe destacar que en un primer momento se ha querido buscar un gestor de porfolio que tuviese una API para migrar los datos desde un bróker con API a otra API. Lamentablemente de las opciones comentadas anteriormente solo Sharesight dispone de dicho servicio, pero se ha descartado puesto que es de pago. Al intentar contactar con ellos para una cuenta gratuita no han podido proporcionar ninguna.

Ante esta situación y viendo todas las alternativas presentes se elige Investing.com como mejor opción al tener tanto alcance y herramientas de análisis.

## 2.3 Python

Se ha seleccionado Python como lenguaje de programación principal en el desarrollo de este proyecto. Se trata de uno de los lenguajes más destacados de estos años, siendo uno de los más utilizados a nivel mundial [\[20\]](#). Es un lenguaje de programación de alto nivel e interpretado que se emplea en muchas áreas como el desarrollo software, inteligencia artificial, análisis de datos, automatización...

Una de las ventajas de Python es su sencilla sintaxis que la hace fácil de leer, ideal para aprender sin tener conocimientos previos. En parte esto se debe al uso de espacios en blanco para delimitar bloques de código, que impulsa el aspecto anteriormente mencionado de una sintaxis legible. Por otro lado, se trata de un lenguaje multiparadigma que combina tanto la programación orientada a objetos como la programación funcional, haciéndolo ideal para la creación de scripts de poco tamaño y también grandes proyectos que tengan mayor complejidad.

Python al mismo tiempo es compatible con gran cantidad de sistemas operativos, dándole la posibilidad de ser ejecutado en todos ellos sin necesidad de modificaciones adicionales. Con esto se permite poder cambiar de entorno fácilmente y poder adaptarlo con la menor cantidad de cambios. Esto va mucho más allá dado que se puede integrar fácilmente con otros lenguajes de programación abriendo la posibilidad de escoger Python como lenguaje principal del programa o para complementos adicionales.

En último lugar, otro de los motivos principales por el cual se ha escogido Python es la amplia comunidad y bibliotecas disponibles. Al ser uno de los lenguajes más populares implica que hay una gran cantidad de personas compartiendo sus proyectos y dispuestos a resolver problemas de código, ideal para poder ver las experiencias de los demás usuarios. Asimismo, con las bibliotecas creadas por los demás se puede llegar a ahorrar un gran cantidad de tiempo al poder importarlas directamente y disminuir el tiempo invertido en programar elementos y funcionalidades desde cero.

### 2.3 Crítica al estado del arte

Es importante antes de iniciar un proyecto ver trabajos ya realizados para determinar así las contribuciones que se pueden aportar. En esta sección se tiene como objetivo analizar Trabajos Final de Grado presentados en la Escuela Técnica Superior de Ingeniería Informática para identificar así posibles incoherencias, fallos y ausencias de ciertos aspectos en los proyectos ya presentados.

En primer lugar, se encuentra el trabajo “Aplicación para Android: agenda comercial y detalles de Clientes”, que tiene algunos pasos similares como el hecho de recuperar cartera de usuarios [21]. Sin embargo, este proyecto se trata de una aplicación Android y que las mismas peticiones son a través de servicios web REST, por lo tanto, no es muy similar al desarrollo que se quiere hacer. En este proyecto no se explica la creación del porfolio dando a entender que este ya se encuentra en la base de datos de la empresa.

Otro ejemplo, que en este caso es más parecido es “Diseño de un portal web de gestión de carteras de acciones” [22]. Se desarrolla una aplicación web para administrar carteras de valores, es decir, un gestor de porfolios que además a partir de este se pueden emitir órdenes ficticias para operar. Se echa en falta un poco más de interacción con servicios reales relacionados con los mercados.

A pesar de todo, a partir de los casos analizados más parecidos pueden resultar interesantes a la hora de extraer ideas generales para la realización de este mismo.

## 2.4 Propuesta

Con Trabajos Final de Grados ya realizados por otros alumnos y analizados ahora se puede encaminar un poco mejor la planificación de este proyecto para abordar áreas de conocimiento no tratadas.

Con los ejemplos vistos anteriormente destaca como desarrollan las ideas desde cero, creando aplicaciones o portales web siendo un trabajo muy completo, pero también extenso y costoso. Una de las ventajas en la informática es que muchas veces se pueden integrar lo mejor de varios servicios en una misma funcionalidad o trabajo, como si se tratase de importar librerías. Por ello, aprovechando programas y páginas webs ya creadas, como las de Interactive Brokers e Investing.com, se extraerán al máximo las funcionalidades que estas ofrecen.

Por una parte, gracias a trabajar con un bróker se simplifica para el usuario realizar sus operaciones del día a día sin modificar su *modus operandi* de compraventa de acciones. Todo esto migrándolo a un potente gestor de porfolio para realizar adecuadamente el seguimiento de sus activos sin la mínima preocupación se convierte en un entorno ideal para el inversor.

---

## 3. Funcionamiento y entorno de la bolsa

---

### 3.1 La bolsa de valores: concepto y funcionamiento

La bolsa de valores es un lugar que conecta a demandantes y oferentes de capital; es decir, une a entidades con déficit económico, como empresas o instituciones, con aquellas que tienen un superávit económico, como ahorradores [23]. Los primeros necesitan capital adicional para desarrollar sus operaciones o emprender nuevos proyectos, mientras que los segundos disponen de capital para invertir a cambio de un rendimiento. La transacción entre estas partes se realiza mediante intermediarios autorizados para actuar en la bolsa de valores. Esta bolsa puede ser física o virtual y es gestionada por una empresa que, a su vez, está bajo el control de una o más entidades regulatorias.

En ella se negocian instrumentos financieros como acciones y bonos, entre otros, con el objetivo de facilitar su compraventa, dando origen a un mercado organizado, regulado y transparente que asegura la equidad de las operaciones bursátiles. La organización y estructura del mercado de valores están diseñadas para funcionar adecuadamente, respaldadas por organismos regulatorios que vigilan constantemente, garantizando el cumplimiento de las normas y requisitos para las empresas que cotizan en él. De esta manera, gracias a estos mecanismos de control, se previene que los agentes involucrados participen en prácticas fraudulentas, protegiendo así la integridad del mercado.

Hay que comprender que la bolsa de valores son mercados secundarios de valores, es decir, se negocian instrumentos financieros que ya han sido emitidos anteriormente en un mercado primario. Para clarificar este concepto, aquel inversor que acude a la bolsa de valores a comprar un activo en realidad no se lo está comprando a la compañía que lo ha emitido en un principio, en realidad lo está adquiriendo de un tercero que las ha puesto en venta. De esta transacción la compañía emisora no está ganando un beneficio directo, sin embargo, las acciones que acaban teniendo una mayor demanda reflejan una empresa de mayor valor puesto que el precio del activo irá subiendo. Con todo esto, se entiende así que el mercado secundario de activos es en realidad dónde más transacciones se realizan, ya que en el primario una vez se han comprado para los futuros intercambios siempre será en el mercado secundario.

De esta forma, las funciones de la bolsa de valores se pueden definir en los siguientes aspectos [24]:

- **Liquidez:** aportar liquidez a las unidades con déficit puesto que los inversores están dispuestos a ofrecer su capital a cambio de un activo. Este es el motivo principal por el que las empresas “salen a Bolsa”, en general para crecer más rápido y asegurarse una posición estable en el mercado.

- **Impulsar productividad:** al aportar liquidez a las unidades con déficit que disponen de nuevas ideas se impulsa la productividad del tejido económico empresarial.
- **Veracidad:** actuar como un espacio en el que poder reunir información sin ninguna posible manipulación. En la bolsa de valores se podrá analizar información sobre las empresas que cotizan, transacciones realizadas, así como diferente activos con los que se operan.
- **Seguridad:** al estar bajo vigilancia de las entidades regulatorias la bolsa goza de una mayor prevención de actividades ilegales.

En cuanto al funcionamiento de la bolsa de valores se recuerda que los usuarios implicados no pueden realizar transacciones entre ellos por sí mismos. Para poder realizar el intercambio debe haber una empresa de servicios de inversión que gestione este proceso, también sería válido una entidad de crédito. En esta línea, los principales usuarios que se pueden detectar en el funcionamiento de la bolsa son:

- **Inversor:** persona o empresa que opera en la bolsa comprando y vendiendo activos. El objetivo de este usuario es obtener unos beneficios de la operaciones que realiza en tiempo real dentro del mercado. Como se ha comentado el inversor por sí no tiene acceso directamente para realizar transacciones, por ello actúa como cliente de un bróker.
- **Bróker:** término ya introducido anteriormente en el capítulo de introducción. Se trata de la entidad autorizada para operar dentro del mercado ejecutando las órdenes de los inversores. No actúan por su propia cuenta en ningún momento, simplemente acatan las órdenes dictadas por el inversor facilitándole sus servicios. Desde luego que al ofrecer todas estas facilidades, especialmente cuando se tratan de brókers online que además desarrollan toda una plataforma, piden una contraprestación. En este caso la metodología habitual es exigir una comisión por las operaciones realizadas.

Con la posibilidad ya de operar se pueden comprar y vender los activos al precio al que coticen. Este precio no está fijado a un valor determinado, sino que en realidad depende en todo momento de la ley de oferta y demanda, variando constantemente el valor de los activos [\[25\]](#). Por ejemplo, en caso de que se quiera operar con acciones:

1. Si en este momento en el mercado hay muchos inversores que están comprando constantemente las acciones de una determinada empresa lo que ocurrirá es que el precio subirá.
2. Por el lado contrario, en caso de haya muchos inversores tratando de vender las acciones, pero no haya nadie comprándolas el precio irá cayendo.





### 3.2 Instrumentos financieros: clases y características

A estas alturas puede haber surgido la duda cuando se habla de instrumentos financieros ya que se introduce como uno de los principales términos de este capítulo. Según la CNMV se trata de “contrato que genera, al mismo tiempo, un activo financiero para una de las partes y un pasivo financiero para la otra.” [26] Por tanto, lo que se entiende de esta definición es que una de las partes tiene el derecho de obtener ciertos beneficios económicos, ya que ha gastado dinero para adquirirlo, y la otra parte se ve sometida a la obligación de liquidar dicho derecho.

Sin embargo, cuando se habla de instrumentos financieros estos pueden ser de diferentes tipos [27] [28]. En primer lugar, se pueden separar en instrumentos financieros no complejos, que son aquellos que presentan la ventaja de no necesitar mucha experiencia para operar con ellos:

- **Valor de renta variable:** se trata de un valor que proporciona tanto parte de la participación en la empresa como derechos frente a esta para percibir dividendos o votar en las juntas de accionistas. Es la opción más popular y la que más se va a ver a lo largo de este Trabajo Final de Grado, aunque, los otros tipos de activos también podrán ser una opción válida a la hora de operar con ellos.
- **Valor de deuda:** dentro de esta categoría se pueden incluir: Bonos del Estado, Bonos societarios, acciones preferentes y obligaciones de deuda garantizada. La ventaja de esta categoría, especialmente los bonos, es que se ofrecen unas condiciones y parámetros preestablecidos, confiriendo al comprador un activo que implica un menor riesgo.
- **Fondo de inversión:** varios inversores juntan el dinero en una misma sociedad o bajo un especialista para ser administrado. A la persona nombrada de tomar las decisiones sobre dicho capital se le conoce como gestor del fondo. Asimismo, se separan en dos tipos: fondos de cobertura y fondos mutuos.

Por otra parte, existen instrumentos financieros complejos cuyo valor no está determinado solamente por la oferta y demanda del mercado. Este hecho implica que son bastante más difíciles de entender y aprender a operar con ellos, requiriendo de un nivel más alto de conocimientos. Los instrumentos financieros complejos, también llamados derivados, y su precio depende de otros activos como por ejemplo índices bursátiles o acciones. Los principales tipos que se pueden encontrar son los siguientes:

- **Futuros:** acuerdo para intercambiar un activo subyacente a un precio preestablecido para una fecha en el futuro. Sin embargo, a pesar de definir el contrato desde un principio si ambas partes están de acuerdo pueden llegar a cambiar las condiciones.
- **Opciones:** se le otorga al comprador el derecho de comprar o vender un activo subyacente a un precio preestablecido para una fecha en el futuro, pero no comporta ninguna obligación.

- **Swaps:** también llamados permutas, se constituye como un contrato en que las partes se intercambian flujos de caja [29]. Es decir, se pagan cada cierto tiempo una serie de flujos monetarios para más tarde recibir otra serie de flujos.
- **CFD:** producto derivado en el que un inversor solo necesita una fracción del valor total de la inversión para abrir una posición.

Ante esta gran variedad de tipos de instrumentos financieros, puede llegar a surgir una duda muy importante. ¿Cómo se distinguen los diferentes instrumentos sin llegar a confundirlos? La respuesta es el código ISIN. Se trata de código para identificar de forma unívoca los diferentes valores mobiliarios de todo el mundo, ya que un mismo tipo de activo puede estar intercambiándose en diferentes mercados financieros [30]. Por ejemplo, el ISIN US0378331005 se corresponde a las acciones de Apple del mercado de NASDAQ. Entonces, como se puede observar se trata de un código alfanumérico de 12 dígitos donde los dos primeros caracteres indican el país, en este caso US. (para España sería ES).

### 3.3 Operaciones en largo y corto

Es común pensar que para operar simplemente se debe comprar un activo para venderlo más tarde y así obtener las ganancias, sin embargo, esto puede ir más allá.

En primer lugar, se pueden encontrar las operaciones en largo como en la **Figura 3.1**. Para estas, se empieza adquiriendo un activo que actualmente tiene un determinado precio y se espera que en el futuro suba de valor. Cuando supuestamente alcance ese determinado precio esperado se procederá con la venta. Esto también se conoce como comprar en mínimos y vender en máximos.

Un ejemplo podría ser de un inversor que decide comprar 10 acciones a 50€ cada una, con un coste total de 500€. Un mes más tarde el inversor decide venderlas ya que han alcanzado el valor de 100€, recibiendo un total de 1000€. El beneficio final obtenido en esta operación asciende a 500€.

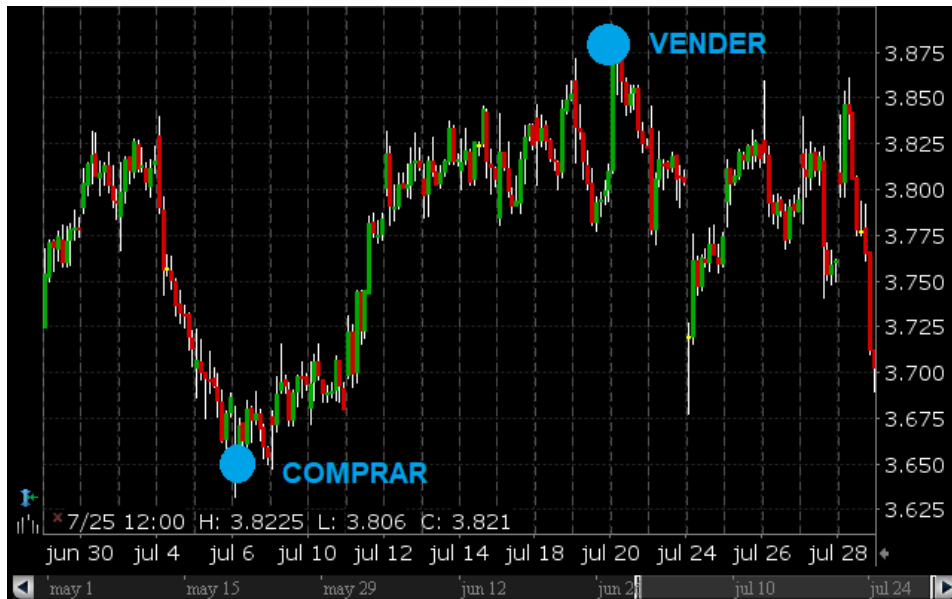


Figura 3.1: Ejemplo operación en largo

Al contrario, se puede operar en corto. Se trata de hacer exactamente lo opuesto como en la **Figura 3.2**, con la idea de que pronto el activo bajará de precio se vende en este instante que el activo tiene un alto valor para comprarlo más tarde.



Figura 3.2: Ejemplo operación en corto

En la misma dirección, es comprensible que hasta el momento se piense que todas estas operaciones se lleven a cabo mediante activos que se encuentren en la cartera del inversor. No obstante, cuando se trabaja en corto puede darse la situación de que el inversor pueda vender activos que no tiene en posesión. Realmente lo que

ocurre es que se toman prestados a otro agente para así poder llevar a cabo la operación, todo esto desde luego a cambio de pagar una comisión.

Un ejemplo de este tipo de operación es que un inversor que actualmente tiene bajo supervisión unas acciones con un valor de 100€. El inversor tiene la esperanza de que dentro de 3 días las acciones bajarán a 70€, por lo tanto, le pide a su bróker que le deje prestado 10 acciones de este tipo y acto seguido el inversor vende todas las acciones por un valor total de 1000€. Al pasar 3 días el valor de dichas acciones disminuye a 70€ y el inversor las vuelve a comprar, por un valor total de 700€. De esta forma si se realizan la diferencia entre los valores de venta y compra asciende a 300€ de beneficio, de los cuales una parte se pagarán como una comisión al bróker.

Sin embargo, puede pasar el escenario para el cual no ocurra lo que espera el inversor. En un escenario alternativo, al pasar 3 días el valor de dichas acciones podría haber aumentado a 130€ y el inversor tiene 2 posibles salidas. En la primera opción esperar más días con la esperanza de que el precio baje de nuevo, sin embargo, incurrirá en grandes comisiones por cada día que tenga acciones prestadas. En la segunda opción, vender a pesar de que tendrá que afrontar pérdidas, siendo así un coste total de 1300€. Las pérdidas serían en este caso de 300€, puesto que en un principio le costó 1000€ la inversión total, cantidad a la que habría que sumar el coste de las comisiones.

### 3.4 Monitorización para operar en bolsa

La monitorización de la bolsa de valores es una práctica esencial para todos los inversores. Implica una serie de tareas como seguir diferentes fuentes de información y analizar constantemente la evolución que experimentan los precios de los activos en los que se quiere invertir. La importancia de esta actividad se ve reflejada en los siguientes puntos [\[31\]](#):

1. **Decisiones informadas:** monitorizar de forma constante permite a los inversores informarse antes de tomar acciones. De esta forma, llegado el momento de realizar un compra o venta siempre se realiza conociendo de antemano las posibles implicaciones.
2. **Prevenir riesgos:** una de las ventajas al ir siguiendo los precios y las últimas noticias es construir una cartera con el menor riesgo posible.
3. **Responder a eventos:** la volatilidad del mercado como ya se ha visto es un factor muy importante a tener en cuenta. Una mala noticia puede provocar la caída en picado del valor de un activo, por ello, enterarse cuanto antes puede ayudar a disminuir las pérdidas o tener grandes beneficios.
4. **Oportunidades:** analizar las tendencias del mercado y encontrar sectores que puedan crecer en el futuro próximo son ejemplos de cómo un inversor puede generar beneficios adicionales.



- 5. Aprender y mejorar la capacidad de análisis:** el hecho de estar analizando continuamente el mercado y leyendo noticias incrementará la experiencia del inversor. Se adquieren así nuevos conocimientos y una mejor comprensión de cómo funciona el mercado en el que se opera.

En pocas palabras, la monitorización de la bolsa de valores es crucial para los inversores, ya que posibilita tomar decisiones fundamentadas en hechos y asegurar mayores ganancias, aprender a adaptarse a los entornos cambiantes y estar actualizado en las últimas tendencias. Al mismo tiempo esto puede ir un paso más allá, los inversores a menudo no emplean las mismas estrategias de inversión, por ello que usan varias cateras de inversión. Con este aspecto se complica aún más la monitorización al haber activos que se compren con propósitos diferentes a los demás, por este motivo tener que hacer un seguimiento y control para cada uno de estos se puede convertir en una larga y compleja tarea para el inversor.

### 3.5 Lógica de las carteras

En la Introducción este Trabajo Final de Grado se ha presentado la necesidad de un inversor que tiene diversas carteras con las que realiza la labor de control y seguimiento en Investing.com. Antes de adentrarse en cómo se programará cada una de estas hay que comprender la lógica operar con ellas.

#### 3.5.1 Funcionamiento Cartera 1

Primera cartera de la que dispone el inversor tratándose de la principal, alrededor de la cual funcionarán todas las demás. Simplemente se constituye como la cartera en la que los componentes son las posiciones que tiene actualmente abiertas en la plataforma de inversión Interactive Brokers. Dicho de otra manera, es la situación actual del inversor con los activos que tiene actualmente en posesión.

Gracias a esta cartera el inversor puede comprobar el estado actual de sus pérdidas y ganancias, es decir, el rendimiento global que tiene actualmente. Además, como se explicará más adelante con la plataforma de Investing.com será posible para cada activo que compone la cartera ver distintos indicadores, tendencias, gráficos y otros elementos que facilitarán el seguimiento y control de las inversiones.

Las operaciones realizadas dentro de esta cartera se reflejarán de forma directa. En la siguiente **Tabla 3.1** se puede observar el comportamiento de la Cartera 1, donde para cada compra o venta realizada en IBKR en la primera columna se muestra con un + o un -, respectivamente. Para la Cartera 1 se traducirá exactamente de la misma forma, variando la cantidad en el mismo valor y dejando la cartera en la situación correspondiente. Cabe destacar que la cantidad de un activo en la posición abierta

puede ser inferior a 0, puesto que se refleja la casuística de tomar prestados activos en el caso de querer vender más de lo que se dispone.

<b>IBKR</b>	<b>Cartera 1 - Investing.com</b>	
<b>Transacción</b>	<b>Transacción</b>	<b>Situación</b>
0	0	0
+10	+10	10
+15	+15	25
-5	-5	20
+20	+20	40
-50	-50	-10
+30	+30	20
-10	-10	10

*Tabla 3.1: Funcionamiento de la Cartera 1*

### 3.5.2 Funcionamiento Cartera 2

La Cartera 2 por su parte no es un reflejo real de los activos en posesión del inversor, es decir, no muestra activos que tiene actualmente. Sirve más como herramienta para hacer un seguimiento de las operaciones en largo cuando se tiene la esperanza de que un activo vaya a aumentar de precio en el futuro.

Antes de explicar el funcionamiento de esta cartera se debe comprender que los inversores suelen mantener un cierto número de activos en la cartera principal, la Cartera 1. Sin embargo, con cierta parte será con el que irán haciendo distintas transacciones y es importante acordarse a que precios se compran y venden los activos para más tarde volver a la cantidad anteriormente fijada como objetivo. ¿Y por qué se hace esto? Todo en realidad tiene que ver con la gestión del riesgo y buscar un rendimiento óptimo. Al mismo tiempo que se diversifica y mantiene múltiples activos, los inversores logran minimizar riesgos asociados a cada uno de ellos, aumentado de esta forma los retornos obtenidos. Entre otras ventajas se puede destacar: disponer en todo momento de liquidez, ya que dada una situación de necesidad de dinero inmediato siempre se podrán vender los activos de la cartera; ir cumpliendo objetivos de inversión de forma escalonada sin incurrir en demasiado riesgo y adaptarse a las inesperadas situaciones del mercado de forma más fácil al disponer de activos en todo momento.

Con esto en mente, entender el funcionamiento de la Cartera 2 resulta mucho más simple. En esta se anotarán activos que se han vendido de la Cartera 1 para poder realizar un seguimiento a lo largo del tiempo. La visión de esta cartera va enfocada en mantener el número de activos fijado por el inversor, por ejemplo, un inversor puede decidir siempre tener 50 acciones de cierta empresa. En caso de que venda, el inversor sabe que en el futuro quiere volver a comprar dichas acciones cuando el precio esté



más bajo. Dicho de otra manera, se anotan posiciones ficticias simplemente para recordar a que precios de venta se cerraron las anteriores.

IBKR	Cartera 1 - Investing.com		Cartera 2 - Investing.com	
Transacción	Transacción	Situación	Transacción	Situación
0	0	0	0	0
+10	+10	10	-10	0
+15	+15	25	-15	0
-5	-5	20	+5	5
+20	+20	40	-5	0
-50	-50	-10	+40	40
+30	+30	20	-30	10
-10	-10	10	+10	20

*Tabla 3.2: Funcionamiento de la Cartera 2*

Como se puede observar de la **Tabla 3.2**, al trasladar las operaciones a la Cartera 2 siempre serán en sentido inverso a las que ocurrían en la Cartera 1. Conviene subrayar el caso de la 5ª fila de +20, donde al no disponer de suficientes acciones en la cartera 2 la cantidad final no puede ser inferior a 0, ya que no tendría sentido. Gracias a este seguimiento que se realiza el inversor será capaz de anotar a qué precio vendió las últimas acciones para así en el futuro cuando las recompre sea a un menor precio. También en la 6ª fila al cerrar 50 posiciones (-50) solo se abren 40 en la Cartera 2, dado que 10 de ellas se han tomado prestadas para operar en corto.

### 3.5.3 Funcionamiento Cartera 3

Para la última cartera se trata de hacer un seguimiento de las operaciones en corto, siendo así otra cartera ficticia más. La expectativa es que en el futuro próximo los activos sigan una tendencia a la baja, disminuyan su valor para así recomprarlos más tarde. En esta situación el inversor no dispone de dichos activos, sino que los pide prestados a otra persona o a su mismo bróker a cambio de una comisión. Por lo tanto, pide prestados activos con un precio alto, se venden y dentro de cierto tiempo los recompra a un menor precio devolviéndolos.

Al tener que recomprar dichas acciones hay que acordarse al precio que se vendieron y aquí es donde entra en juego esta cartera. Gracias a la Cartera 3 se podrá saber cuándo recomprar las acciones para maximizar los beneficios en el mejor momento.

Siguiendo la **Tabla 3.3**

**Tabla 3.3**, solo se verá afectada en la operación en la que se venden 50 acciones al tomar prestadas 10. Al tomarlas prestadas se deben anotar en esta cartera para recordar el número y precio al que se vendieron. En la siguiente transacción se devuelven.

<b>IBKR</b>	<b>Cartera 1 - Investing.com</b>		<b>Cartera 3 - Investing.com</b>	
<b>Transacción</b>	<b>Transacción</b>	<b>Situación</b>	<b>Transacción</b>	<b>Situación</b>
0	0	0	0	0
+10	+10	10	0	0
+15	+15	25	0	0
-5	-5	20	0	0
+20	+20	40	0	0
-50	-50	-10	+10	10
+30	+30	20	-10	0
-10	-10	10	0	0

**Tabla 3.3:** Funcionamiento de la Cartera 3





---

## 4. Análisis del problema

---

El principal problema que tienen los inversores que operan es la larga y compleja tarea de realizar un seguimiento de sus los activos. Tener un adecuado control sobre los instrumentos financieros en posesión es una difícil tarea que dependiendo de la cantidad puede incluso volverse imposible de realizar de forma manual.

Hay que considerar que la cantidad de información que se debe analizar en relación con las inversiones de forma periódica puede ser gran tamaño. Aquí se pueden incluir informes financieros, datos actualizados del mercado, noticias, indicadores y un sinnúmero de otros aspectos a considerar que puedan afectar a las decisiones de un inversor para decidir si comprar o vender. A esto hay que sumarle que la mayoría de los inversores tienden a diversificar sus carteras para incluir diferentes tipos de activos o, aunque sean del mismo tipo estén emitidos por diferentes entidades. El seguimiento y control de todos estos diferentes componentes que puedan formar una cartera requiere un esfuerzo considerable, ya que para cada uno hay que considerar unas características y aproximaciones diferentes.

Al mismo tiempo, muchos mercados financieros tienden a ser altamente volátiles implicando fluctuaciones importantes en las cotizaciones de los activos en cortos periodos de tiempo. Por este motivo la volatilidad supone una dificultad añadida en la monitorización de los activos, los cambios rápidos en los precios, así como condiciones exigen a menudo rápidas decisiones para evitar pérdidas importantes o aprovechar grandes oportunidades. Sin embargo, muchas personas que invierten no tienen el tiempo necesario para realizar sus propios análisis dedicándose a tiempo completo al seguimiento y control de sus inversiones. Aspectos exógenos como responsabilidades laborales y personales pueden limitar la cantidad de atención que se puede dedicar a esta tarea. Incluso se podría añadir que no todo el mundo que invierte tiene el conocimiento necesario para evaluar correctamente el rendimiento de los activos a partir de los datos originales, ralentizando aún más este proceso.

En definitiva, el control y monitorización de las inversiones puede constituir un desafío complejo. Debido a los motivos anteriores surge la necesidad de herramientas tecnológicas que aporten una simplificación de esta tarea, siendo necesarias plataformas que aporten información ya procesada, necesaria, oportuna y precisa.

### 4.1 Análisis de seguridad

El análisis de seguridad de un programa es una parte fundamental para garantizar la integridad del sistema y la protección de datos. Mediante este apartado se busca garantizar la protección y establecer medidas de seguridad adecuadas para fortalecer la seguridad.

Sin embargo, la aplicación desarrollada se ejecutará en un entorno local y no habrá interacciones exteriores con otros agentes que puedan comprometer datos sensibles. Al ser ejecutado en el mismo sistema operativo e interactuando solo con páginas web de confianza implicará que en todo momento se asegure la seguridad del sistema. Asimismo, es también responsabilidad del usuario garantizar la seguridad del dispositivo para que no haya un robo o pérdida de datos de gran importancia por conexiones externas.

## 4.2 Análisis energético

En el ámbito de la programación, el análisis energético constituye un papel fundamental para medir el rendimiento y escalabilidad de los algoritmos desarrollados. Con este aspecto se pretende evaluar los tiempos de ejecución y empleo de recursos de los diferentes componentes que forman la aplicación en función de los datos de entrada. Al entender y poder cuantificar la eficiencia se permite tomar decisiones a la hora de decidir sobre su implementación y optimización, para lograr un rendimiento adecuado y en el menor tiempo.

En la implementación realizada de este proyecto se ha tenido en cuenta que al tener que ejecutar el código diariamente pueda suponer un coste energético a lo largo del tiempo. Por ello se ha intentado reducir al máximo y optimizarlo teniendo en cuenta los siguientes aspectos:

1. **Minimización de repeticiones:** se detectan trozos de código se repiten en la ejecución para en vez de volver a llamar a dichas funciones aprovechar ya los elementos previamente creados y así ahorrar en tiempo y recursos empleados.
2. **Reducir complejidad temporal:** se han evitado crear estructuras que supongan un gran coste temporal.
3. **Reducir complejidad espacial:** la aplicación no emplea gran cantidad de datos simultáneamente, por lo tanto, este apartado no ha sido un gran problema.

Aun así, el código debe ejecutar de fondo varias aplicaciones como la Trader Workstation durante unos segundos hasta importar el porfolio y el navegador Chrome para la ejecución de las operaciones que dependerá de la cantidad de transacciones, programas que son indispensables para la ejecución. Es decir, dependiendo de las especificaciones del dispositivo se puede llegar a afectar el rendimiento de este, pero no será durante mucho tiempo. Además, se ha establecido que es preferible que la aplicación se ejecute durante periodos de inactividad del sistema, para así no influir en las actividades del usuario y ralentizar su flujo de trabajo.



### 4.3 Análisis del marco legal y ético

En cuanto al marco legal de esta aplicación en relación con la protección de datos es necesario que el usuario introduzca de forma voluntaria sus credenciales tanto de Interactive Brokers como Investing.com. Es indispensable adicionalmente que al entrar en estas plataformas se extraigan y se ejecute código que procese información relacionada con las inversiones del mismo usuario.

Por tanto, al introducir las credenciales en este programa el usuario entiende que se van a leer las transacciones de sus cuentas personales en las plataformas de inversión. Sin embargo, en ningún momento los datos que vayan a ser procesados por el programa se entregarán a otras fuentes que no sean Interactive Brokers e Investing.com. Adicionalmente, cualquier archivo generado durante la ejecución se mantendrá localmente dentro del mismo directorio y de la cuenta del mismo usuario del sistema operativo.

### 4.4 Análisis de riesgos

El análisis de riesgos es esencial para identificar y mitigar riesgos que puedan afectar tanto a la aceptación del usuario como a la satisfacción de este. Mediante la identificación de cada uno de ellos se podrán tomar medidas para reducir dicho impacto, así se presenta la siguiente tabla para enumerarlos, el impacto que pueden tener sobre el proyecto y las medidas tomadas:

<b>Tipo de riesgo</b>	<b>Impacto</b>	<b>Medidas</b>
Apagado sistema	Pérdida orden de ejecución transacciones	Prevenir apagado sistema antes de la finalización del programa
Falta conocimientos usuario para la ejecución de la tarea	Imposibilidad de ejecutar programa	Detallar los pasos a seguir
Estado servidores de las plataformas fuera de servicio	Imposibilidad de ejecutar satisfactoriamente el programa	Reintentar ejecución más tarde
Conexión a Internet	Imposibilidad de ejecutar satisfactoriamente el programa	Reintentar ejecución más tarde
No ejecutarse diariamente	Últimas transacciones no se han procesado	Guardar última fecha de ejecución
No ejecutarse en los últimos 7 días	Pérdida registro transacciones > 7 días	Guardado última fecha de ejecución y reiniciar portfolio

*Tabla 4.1: Análisis de riesgos identificados*

En la **Tabla 4.1** se pueden observar riesgos tanto de interrupciones del sistema, por el servidor y especialmente riesgos asociados al propio usuario. Se prevé que el usuario le sea imposible ejecutar todos los días el programa por ello que se iniciará desde donde se ha ejecutado la última vez sin que suponga ningún problema para este.

## 4.5 Estructura de las plataformas

Antes de pasar a analizar las posibles soluciones es necesario entender la estructura de las plataformas escogidas. Esto implicará navegar por sus servicios y entender el flujo de acciones que conllevan para interactuar con ellas para lograr las acciones deseadas. Se indagará por una parte en la estructura de Interactive Brokers, plataforma que proporcionará la cartera del inversor y también por Investing.com, plataforma que gestionará las carteras.

### 4.5.1 Análisis plataforma Interactive Brokers

Como ya se ha comentado en el capítulo 1 de este Trabajo Final de Grado, Interactive Brokers tiene dividida en 3 modalidades su API:

- **API de Client Portal:** destaca por sus simplicidad a la hora de utilizarlo, sin embargo, tiene menos funcionalidades que las otras modalidades que puede llegar a limitar el alcance de este proyecto. Habría sido una interesante elección puesto que el acceso al servicio se hace mediante navegador web, en la misma línea que se planea el acceso a Investing.com.
- **API de Trader Workstation (TWS):** la modalidad más potente, a través de la aplicación de TWS se pueden realizar numerosas acciones para interactuar con tu cuenta de Interactive Brókers, desde extraer información de mercado, descargar información de las carteras como realizar órdenes.
- **FIX:** al estar enfocada para instituciones no tiene sentido indagar más en esta herramienta. Se trata de una modalidad extremadamente potentes para construir sistemas de *trading* complejos que no se contemplan en este proyecto.

Por lo tanto, ante esta situación queda claro que la herramienta más ideal según el análisis de este mismo apartado es la Trader Workstation o TWS. Gracias a ella se podrán extraer las posiciones abiertas de la cartera del inversor, así como el historial de transacciones realizadas en los últimos días.

#### 4.5.2 Análisis plataforma Investing.com

A continuación, se presentará la estructura del página de Investing.com. Es crucial entender cuáles son los principales servicios dentro de este página y como estos se estructuran para poder más adelante poder realizar un correcto diseño de la solución.

- **Portada:** la página principal de Investing.com se trata de una interfaz muy cargada con gran cantidad de datos, información e imágenes. Como se observa en la **Figura 4.1**, en esta se pueden encontrar noticias, gráficos para mostrar la evolución de los activos y índices. Asimismo, si se continúa bajando hasta la **Figura 4.2** la cantidad de herramientas de análisis ofrecidas continúan apareciendo como el análisis técnico, buscador de acciones y datos del mercado financiero en tiempo real.

ID	1W	1M	6M	1Y	5Y	Max
US 30	35,100.0	+38.8	+0.11%			
US 500	4,559.2	-6.5	-0.14%			
Dow Jones	35,061.21	+109.28	+0.31%			
S&P 500	4,565.72	+10.74	+0.24%			
Nasdaq	14,358.02	+4.38	+0.03%			
S&P 500 VIX	13.94	+0.18	+1.31%			
Dollar Index	99.895	-0.093	-0.09%			

Figura 4.1: Página principal Investing.com

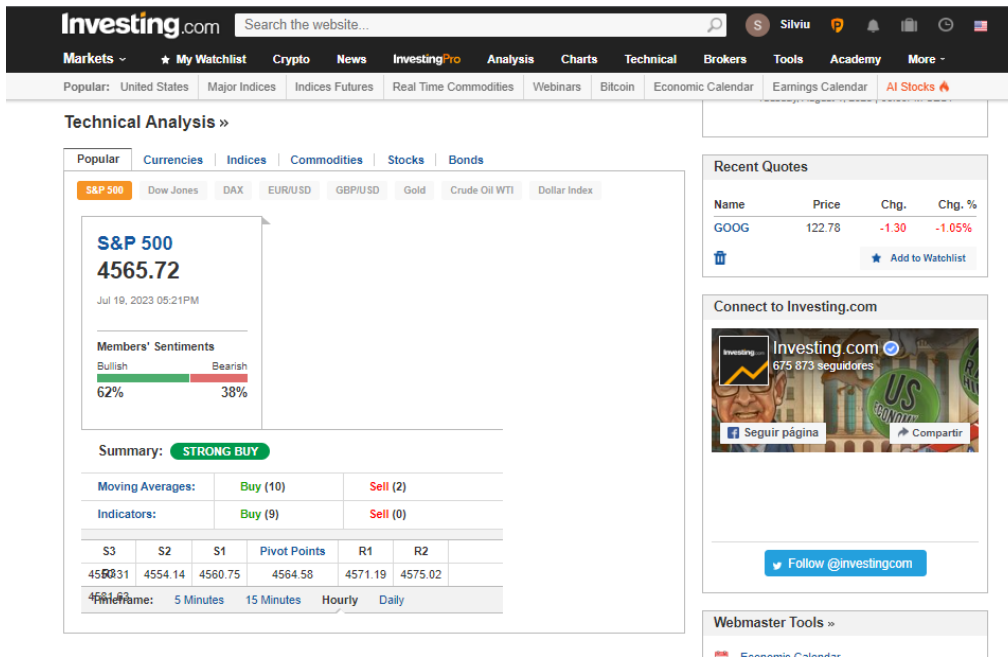


Figura 4.2: Página principal Investing.com - Otros elementos

- Mi cartera:** en la **Figura 4.3** se puede observar el principal componente “My Watchlist” que ayuda al inversor a gestionar su portafolio. Se trata de una herramienta muy potente en la que se pueden ir añadiendo los activos a placer del usuario ya que no supone ninguna transacción real. A partir de la se puede ver que se permite crear diferentes carteras para hacer un seguimiento individual, muy útil cuando se emplean diferentes estrategias de inversión.

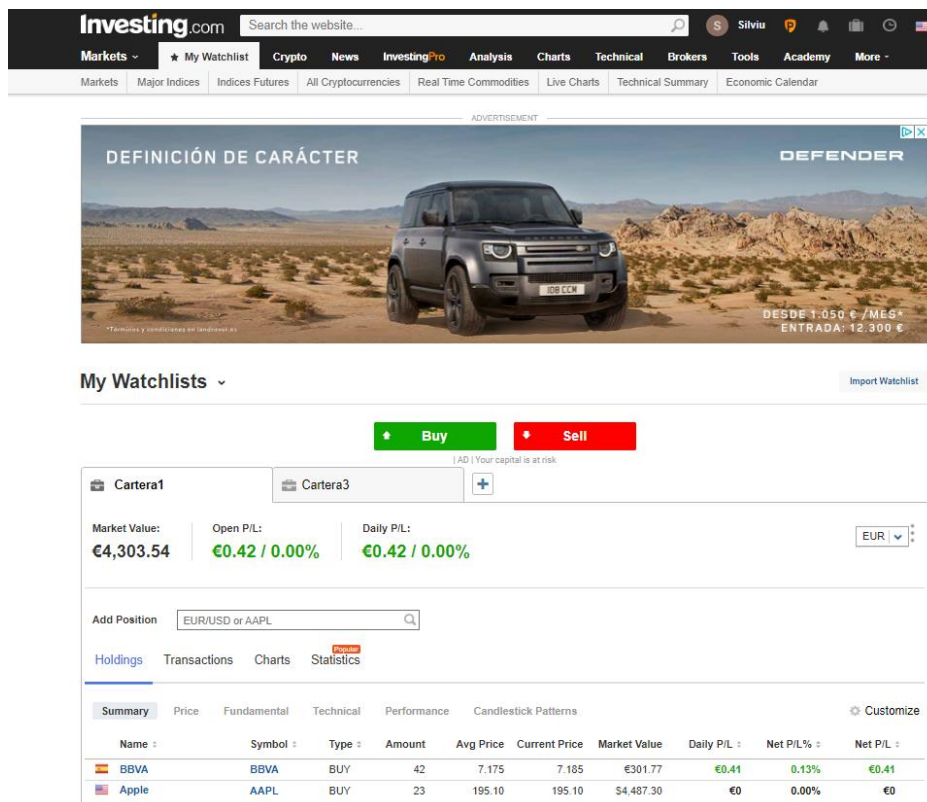
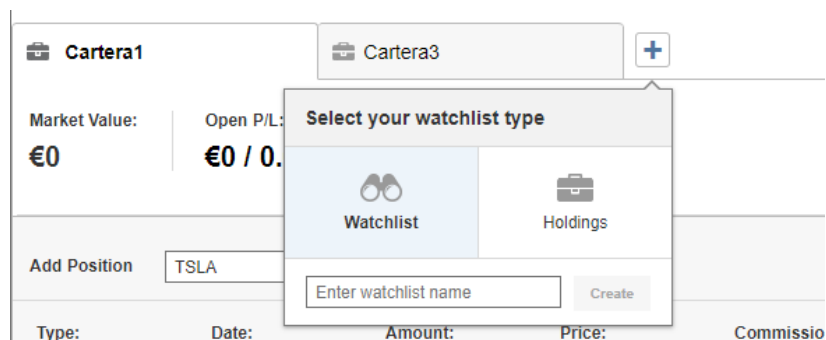


Figura 4.3: Pestaña “My Watchlist” de Investing.com

- **Crear nueva cartera:** en Investing.com se te permite crear múltiples carteras para hacer un seguimiento y control efectivo de los activos al hacer click en el símbolo +. En este momento se abrirá como en la **Figura 4.4** para elegir 2 tipos de opciones junto a designarle un nombre:
  - **Watchlist:** se trata simplemente de una lista de activos para tenerlos en el punto de mira y seguir su evolución. No es una cartera en realidad, simplemente se elegirán activos para ver como va cambiando su precio y otros indicadores.
  - **Holdings:** hace referencia a la cartera, en esta se pueden añadir los activos que el inversor tenga introduciendo todos los datos necesarios pertenecientes a las transacciones.



*Figura 4.4: Creación de una cartera*

- **Importar cartera existente:** en la misma página de “My Watchlist” hay un botón para importar una cartera. Esta situación es ideal cuando se opera en un bróker externo y se quiera importar el portafolio de activos a Investing.com. Este proceso consta de 3 sencillos pasos y se irá reflejando en las siguientes 3 :
  1. Escoger el portafolio de los archivos locales, asignándole un nombre y el tipo de moneda con la que se va a operar y reflejar las variables numéricas. **Figura 4.5**
  2. Se leerá el CSV anterior y el detector de Investing.com reconocerá automáticamente las columnas del archivo asignándoles la variable adecuada. **Figura 4.6**
  3. En el último paso se identifican los activos que componen la cartera. Es crucial que en esta etapa tanto el nombre de la empresa o ISIN sea correcto para que se seleccione automáticamente sin necesitar pasos adicionales. **Figura 4.7**

## Import Watchlist

Cancel Import

### Step 1/3: Choose a portfolio to import

**Upload your existing portfolio file**

 [Browse](#)

**Import to**

**Portfolio type**

Watchlist

Positions

**Portfolio currency:**

Supported file format: CSV

[Next](#)

Figura 4.5: Importar cartera - Paso 1

### Step 2/3: Choose the correct columns

Portfolio type: Positions

**Info** Please verify that the columns have been identified correctly. Choose the appropriate field for each column if necessary.

**Mandatory fields:**  Symbol/ISIN  Open Price  Amount

<input type="checkbox"/>	Symbol/ISIN	Type	Amount	Open Price
<input type="checkbox"/>	US0378331005	STK	50.0	189.698466
<input type="checkbox"/>	US0231351067	STK	54.0	129.68962965
<input type="checkbox"/>	US05946K1016	STK	40.0	7.707275
<input type="checkbox"/>	US0846561076	STK	20.0	1.4241
<input type="checkbox"/>	US3498623004	STK	15.0	0.93076
<input type="checkbox"/>	US02079K1079	STK	55.0	119.10127275
<input type="checkbox"/>	US55087P1049	STK	59.0	10.04335765
<input type="checkbox"/>	US30303M1027	STK	54.0	289.19435185
<input type="checkbox"/>	US5949181045	STK	12.0	335.14583335
<input type="checkbox"/>	US88160R1014	STK	-8.0	275.01640095

[Delete](#)

[Back](#) [Next](#)

Figura 4.6: Importar cartera - Paso 2

### Step 3/3: Confirm and edit tickers

Portfolio type: Positions

**Info** Please review the data below. You may fix or change anything that doesn't match by clicking on the relevant box. Use the toggle button at the bottom of the box to filter for only incorrect or unidentified items.

<input type="checkbox"/>	Symbol	Name	Exchange	Open Price	Open Date	Amount	Type
<input type="checkbox"/>	<input type="text" value="AAPL"/>	Apple Inc <sup>1</sup>	Stock NASDAQ	189.698466	07/20/2023	50	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="AMZN"/>	Amazon.com Inc <sup>1</sup>	Stock NASDAQ	129.68962965	07/20/2023	54	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="BBVA"/>	Banco Bilbao Viscaya ... <sup>1</sup>	Stock NYSE	7.707275	07/20/2023	40	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="BGRY"/>	Berkshire Grey Inc <sup>1</sup>	Stock NASDAQ	1.4241	07/20/2023	20	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="FORD"/>	Forward Industries Inc <sup>1</sup>	Stock NASDAQ	0.93076	07/20/2023	15	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="GOOG"/>	Alphabet Inc Class C <sup>1</sup>	Stock NASDAQ	119.10127275	07/20/2023	55	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="LYFT"/>	LYFT Inc <sup>1</sup>	Stock NASDAQ	10.04335765	07/20/2023	59	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="META"/>	Meta Platforms Inc <sup>1</sup>	Stock NASDAQ	289.19435185	07/20/2023	54	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="MSFT"/>	Microsoft Corporation <sup>1</sup>	Stock NASDAQ	335.14583335	07/20/2023	12	<input type="text" value="BUY"/>
<input type="checkbox"/>	<input type="text" value="TSLA"/>	Tesla Inc <sup>1</sup>	Stock NASDAQ	275.01640095	07/20/2023	8	<input type="text" value="SELL"/>

[Delete](#)

Show only unvalidated rows

[Back](#) [Import Watchlist](#)

Figura 4.7: Importar cartera - Paso 3





- **Añadir nueva posición:** dentro de la pestaña de “My Watchlist” para añadir una nueva posición a una nueva cartera bastará con acceder al buscado e introducir el nombre de la empresa. En este caso en la **Figura 4.8** se han buscado acciones de Tesla Inc. Una vez elegido el activo que se quiere añadir a la cartera se deben introducir los datos de este como en la **Figura 4.9**. El significado de cada una de las variables es la siguiente:
  - **Type:** se refiere al tipo del activo que queremos introducir. Puede ser de tipo “Compra”, si estamos comprando el activo o “Venta en función de si lo estamos vendiendo.
  - **Date:** fecha de realización de la transacción del activo.
  - **Amount:** cantidad de activos involucrados en la transacción
  - **Price:** precio por activo al que se ha realizado la transacción. En este caso el tipo de moneda de las operaciones está definido durante el proceso de creación de la cartera.
  - **Commission:** coste que cobran los intermediarios por realizar la transacción de compraventa en nombre de un inversor. Cuanto más alto es este valor menores serán los beneficios.

The screenshot shows the 'Add Position' search interface. At the top, there are three portfolio tabs: 'Cartera1', 'Cartera3', and a plus sign. Below the tabs, there are three summary boxes: 'Market Value: €0', 'Open P/L: €0 / 0.00%', and 'Daily P/L: €0 / 0.00%'. A currency dropdown is set to 'EUR'. The search bar contains 'tesla'. A dropdown menu is open, showing search results for Tesla Inc and various Tesla-related ETFs and ETPs. The 'Type' is set to 'BUY'. The search results are categorized by 'All', 'Indices', 'Equities', 'ETFs', 'Funds', 'Commodities', 'Currencies', 'Bonds', and 'All Countries'. The results include TSLA (Tesla Inc), TSETNQJ (FNB ETN on TESLAQ SEP25), TSETNCJ (FNB ETN on TESLAC SEP25), 2STS (Leverage Shares -2x Short Tesla ETP...), TS3S (Leverage Shares -3x Short Tesla ETP...), 1TSL (LS 1x Tesla Tracker ETP Securities), 3TSL (Leverage Shares 3x Tesla ETP Securi...), 3STS (GraniteShares 3x Short Tesla Daily ETP), and 3LTS (GraniteShares 3x Long Tesla Daily ETP).

Figura 4.8: Añadir una nueva posición

The screenshot shows the 'Add Position' form with 'TSLA' entered in the search bar. The form fields are: Type: BUY, Date: 07/20/2023, Amount: (empty), Price: 291.26, Commission: \$. There is an 'Add Position' button at the bottom right.

Figura 4.9: Introducir datos nueva posición

- **Tabla de la cartera:** contiene todas las posiciones abiertas de la cartera que actualmente se visualiza. Contiene gran cantidad de datos que ayudan al inversor a hacer posible el seguimiento al igual que en **Figura 4.10:**
  - **Name:** nombre completo del activo.
  - **Symbol:** se trata de un identificador o abreviatura empleada para representar el activo. Suele tratarse del mismo nombre de la empresa, pero acortándolo.
  - **Type:** se refiere al tipo del activo que queremos introducir. Puede ser compra o venta.
  - **Amount:** cantidad o número de ese activo que posee el inversor.
  - **Avg Price:** precio promedio al que el inversor ha adquirido el activo. Se puede calcular dividiendo el coste total de las transacciones entre Amount para obtener así esta variable.
  - **Current Price:** precio actual en el mercado.
  - **Market Value:** valor total de la inversión que se consigue multiplicando Amount por el Current Price.
  - **Daily P/L:** se refiere a la ganancia o pérdida diaria en la que se ha incurrido. Se puede calcular como la diferencia entre el valor actual y el valor del día anterior.
  - **Net P/L %:** se refiere a la ganancia o pérdida en término porcentual respecto con la inversión inicial de toda la inversión de ese activo.
  - **Net P/L:** se refiere a la ganancia o pérdida en término absoluto de toda la inversión de ese activo.

Summary	Price	Fundamental	Technical	Performance	Candlestick Patterns	⚙️ Customize			
Name	Symbol	Type	Amount	Avg Price	Current Price	Market Value	Daily P/L	Net P/L %	Net P/L
🇺🇸 Microsoft	MSFT	BUY	32	343.67	349.13	\$11.17K	\$174.60	1.58%	\$174.60
🇺🇸 Tesla	TSLA	SELL	8	275.02	270.56	-\$2,164.48	\$35.65	1.62%	\$35.65
🇺🇸 Meta Platforms	META	BUY	54	289.19	307.29	\$16.59K	\$977.16	6.25%	\$977.16
🇺🇸 LYFT	LYFT	BUY	59	10.04	11.62	\$685.58	\$93.02	15.69%	\$93.02
🇺🇸 Alphabet C	GOOG	BUY	55	119.10	122.09	\$6,714.95	\$164.37	2.50%	\$164.37
🇺🇸 Forward Industries	FORD	BUY	15	0.930760	0.895200	\$13.42	-\$0.53	-3.82%	-\$0.53
🇺🇸 Berkshire Grey	BGRY	BUY	20	1.424	1.400	\$28	-\$0.48	-1.69%	-\$0.48
🇺🇸 BBVA ADR	BBVA	BUY	40	7.707	8.015	\$320.60	\$12.30	3.99%	\$12.30
🇺🇸 Amazon.com	AMZN	BUY	54	129.69	132.19	\$7,138.26	\$135.01	1.92%	\$135.01
🇺🇸 Apple	AAPL	BUY	50	189.70	194.41	\$9,720.50	\$235.57	2.48%	\$235.57

*Figura 4.10: Ejemplo cartera de inversiones*

- **Editar Posición:** si se desea cambiar algún dato por haberlo introducido mal durante la etapa de añadido, se puede hacer click en la posición abierta para modificar esa transacción. Se puede observar en la **Figura 4.11** que se podrán modificar todos los parámetros iniciales: Date, Amount, Price y Commission.

## Aplicación para la trazabilidad de operaciones en el mercado de valores

Add Position

Holdings Transactions Charts **Statistics**

Summary									
Name	Symbol	Type	Amount	Avg Price	Current Price	Market Value	Daily P/L	Net P/L%	Net P/L
🇺🇸 Tesla	TSLA	SELL	8	275.02	274.28	-\$2,194.24	\$5.89	0.26%	\$5.89
🇺🇸 Microsoft	MSFT	BUY	12	335.15	353.18	\$4,238.16	\$216.40	5.38%	\$216.40
Open Date:		Amount:	Open Price:	Total Commission:		Actions			
07/20/2023		12	335.14583335	\$0.00		Close Position Edit position			
🇺🇸 Meta Platforms	META	BUY	54	289.19	312.14	\$16.85K	\$1,239.06	7.93%	\$1,239.06
🇺🇸 LYFT	LYFT	BUY	59	10.04	11.77	\$694.43	\$101.87	17.19%	\$101.87
🇺🇸 Alphabet C	GOOG	BUY	55	119.10	123.10	\$6,770.50	\$219.92	3.35%	\$219.92
🇺🇸 Forward Industries	FORD	BUY	15	0.930760	0.885000	\$13.28	-\$0.68	-4.91%	-\$0.68
🇺🇸 Berkshire Grey	BGRY	BUY	20	1.424	1.400	\$28	-\$0.48	-1.69%	-\$0.48
🇺🇸 BBVA ADR	BBVA	BUY	40	7.707	8.035	\$321.40	\$13.10	4.25%	\$13.10
🇺🇸 Amazon.com	AMZN	BUY	54	129.69	133.26	\$7,196.04	\$192.79	2.75%	\$192.79
🇺🇸 Apple	AAPL	BUY	50	189.70	195.42	\$9,771	\$286.07	3.01%	\$286.07

Figura 4.11: Editar una posición abierta

- Close position:** se trata de cerrar una posición abierta de la actual cartera. Al presionar el botón de “Close position” la primera vez se tendrán que introducir los datos de la transacción. Cabe destacar que en Amount no se puede poner un número mayor que la cantidad total de esa posición. Si se quiere cerrar 15 posiciones de Microsoft como en la **Figura 4.12**, primero se cierran las 12 de abajo y después 3 de arriba.

Name	Symbol	Type	Amount	Avg Price	Current Price	Market Value	Daily P/L	Net P/L%	Net P/L
🇺🇸 Microsoft	MSFT	BUY	32	343.67	349.82	\$11.19K	\$196.68	1.78%	\$196.68
Close Date:		Amount:	Close Price:	Total Commission:		Actions			
07/20/2023		20	348.79	\$0.00		Close Position Edit position			
07/20/2023		12	349.22	\$0.00		Close Position Cancel			

Figura 4.12: Cerrar una posición abierta

- Otros elementos:** otras funcionalidades al igual que la **Figura 4.13** pueden resultar de interés dentro de la cartera son las otras pestañas:
  - Transactions:** se trata de un historial de las transacciones abiertas y cerradas.
  - Charts:** diferentes gráficos interactivos que resultan de gran utilidad a la hora de realizar un seguimiento de los activos que componen una cartera.
  - Statistics:** estadísticas de varios indicadores a través de elementos gráficos.

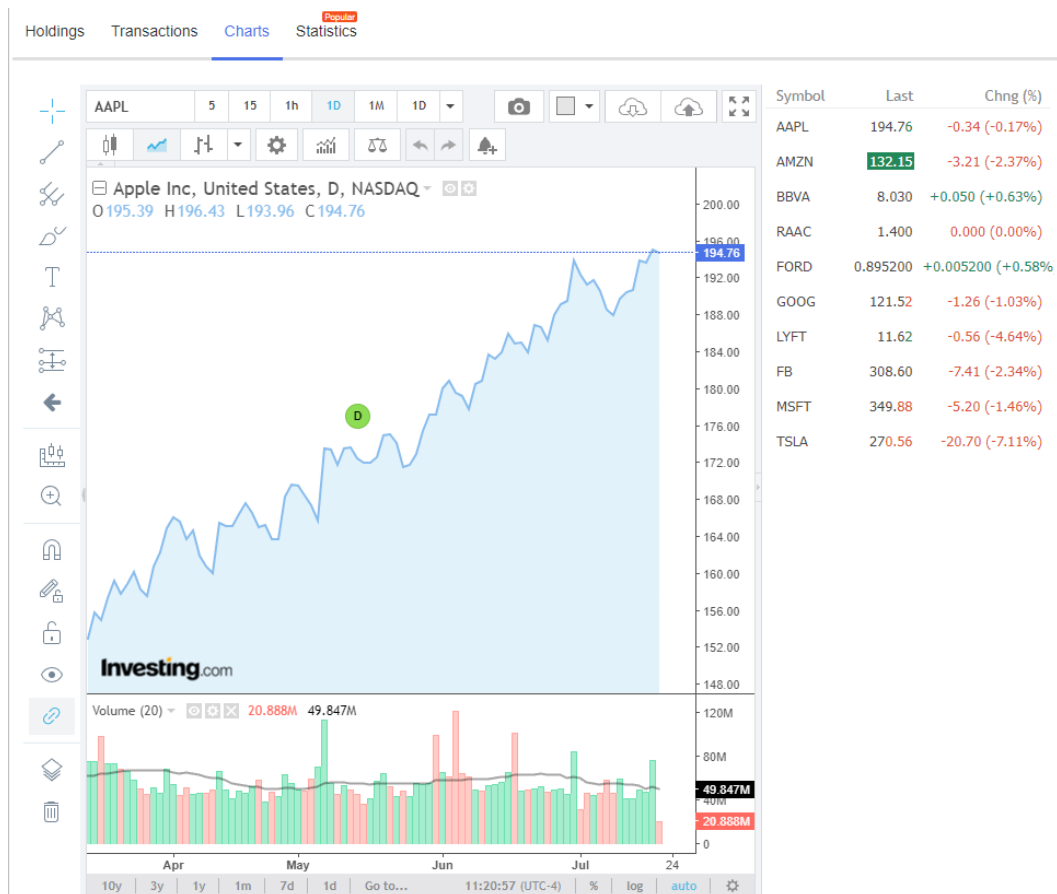


Figura 4.13: Gráfico de evolución activo

## 4.6 Solución propuesta

La solución que se va a proponer a continuación se hará en base a todos los análisis anteriormente realizados y los conocimientos que se han ido adquiriendo. La solución gira entorno a la creación de un programa sin interfaz capaz de llevar los datos tanto del portfolio como el historial de transacciones de Interactive Brokers hasta Investing.com.

En este camino se deberá acceder a una API, así como automatizar la interacción con la página web de Investing.com para poder lograr todos los objetivos.

# Aplicación para la trazabilidad de operaciones en el mercado de valores

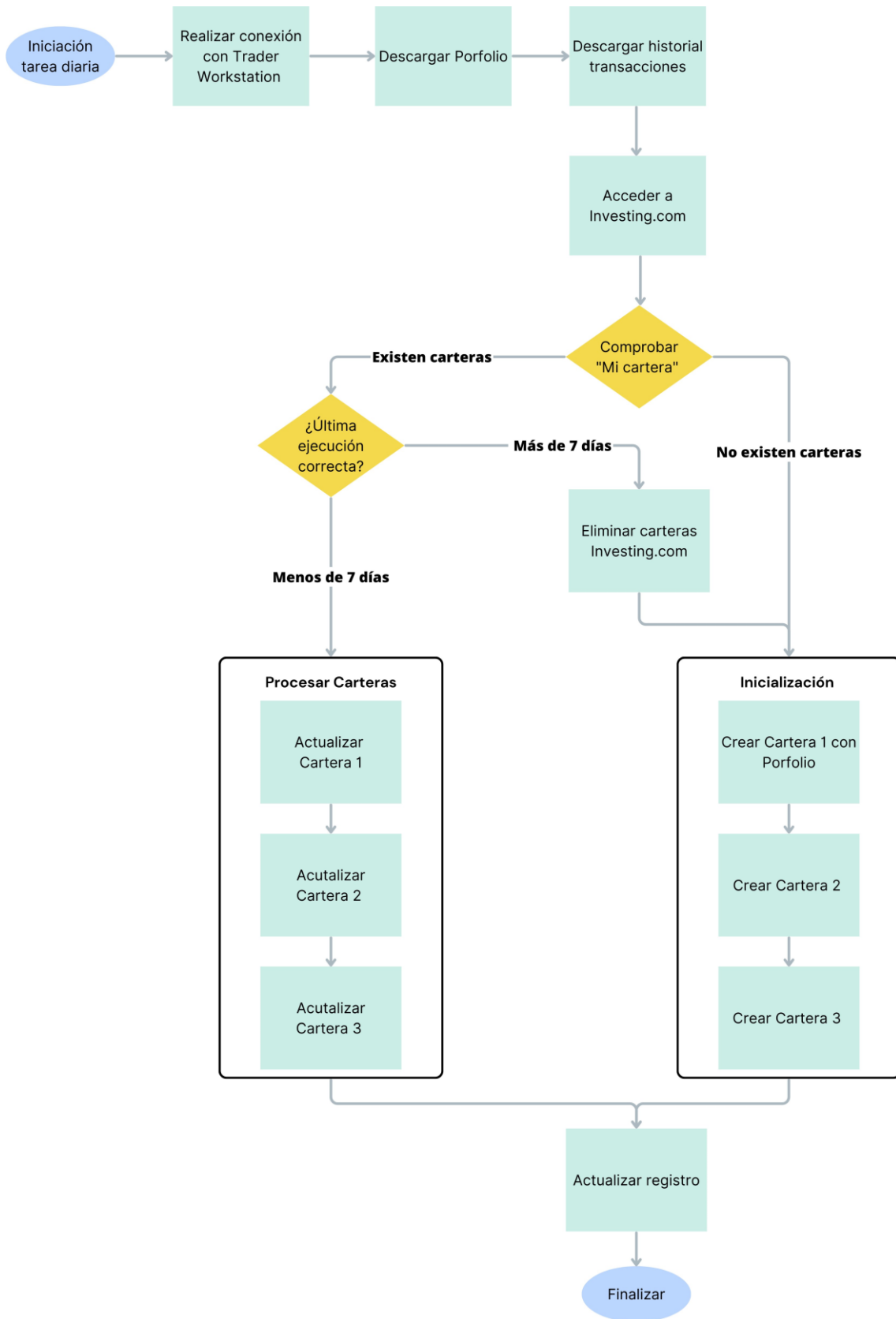


Figura 4.14: Diagrama de flujo ejecución diaria

En la **Figura 4.14** se puede observar el diagrama de flujo general que seguirá la solución propuesta. Se trata de un programa que se pretende ejecutar de forma lo más discreta posible para no interrumpir las actividades del usuario y que no suponga ningún inconveniente. Adicionalmente, se pretende en la misma dirección que se preocupe lo menos posible por la continua ejecución.

En primer lugar, se pretende planificar la ejecución del programa de forma automática para que este todo los días se ejecute siguiendo un horario. Por ello, será el sistema operativo el encargado de asegurarse del inicio del programa.

Con el programa iniciado la primera tarea será abrir el software Trader Workstation o TWS de Interactive Brokers mediante las credenciales previamente guardadas en el código y establecer la conexión con la API. Con este paso anterior ya es posible acceder a todos los datos del inversor y se procederá a guardar en primer el lugar porfolio, es decir todas las posiciones abiertas que tiene actualmente en IBKR. Además, es necesario descargar las transacciones realizadas en los últimos días para así poder procesar las operaciones que se van realizando.

Teniendo ya todos los datos en el sistema queda subirlos a la plataforma de Investing.com. Sin embargo, cabe recordar que no disponen de API para poder facilitar esta tarea, por ello que se tendrá que emplear librerías que permitan la automatización de las tareas en la página web. Para empezar, se debe acceder así a [www.investing.com](http://www.investing.com), iniciar sesión y proceder a navegar hasta al apartado de “My Watchlist” o “Mi cartera” para poder visualizar los datos que actualmente hay guardados. A continuación, pueden ocurrir 2 situaciones:

- 1. No existen carteras:** quiere decir que no existen datos guardados para las cartera de IBKR y habrá que empezar desde cero. Ante esta situación se realizará un proceso de inicialización en el cual se crearán las diferentes carteras que se han explicado en anteriores capítulos. Para la Cartera 1 habrá que importar todos los datos del porfolio actual que hay en Interactive Brokers y de esta forma quedará actualizada con los nuevos datos. Para la Cartera 2 y Cartera 3 al tratarse de la primera ejecución se crearán completamente vacías, ya que no hay nuevas transacciones que procesar. Con todo esto finalizaría el proceso de inicialización.
- 2. Existen carteras:** en cambio en esta situación se complica un poco más, puesto que dependerá de cuando fue la última ejecución en la que se procesaron correctamente todas las carteras. Por este motivo, como en el programa de Trader Workstation solo se guardan transacciones ocurridas en los últimos 7 días hay que mantener un registro de las ejecuciones. Si el programa se ha ejecutado hace más de 7 días queda claro que ya se han podido perder múltiples transacciones y aunque se procesasen las que actualmente están en el sistema las carteras de Investing quedarían desactualizadas. Por consiguiente, solo quedará eliminar toda la información que hay actualmente en Investing.com, borrando las tres carteras y empezando de nuevo. Para esto habrá que realizar de nuevo todo el proceso de inicialización. Sin embargo, para el otro caso en que existan las carteras y se haya ejecutado hace menos de 7 días habrá que actualizarlas. Para lograrlo habrá que procesar cada una de las transacciones

siguiendo la lógica del capítulo 3.5 3.5 Lógica de las carteras empleando el archivo de las operaciones que se ha descargado en el principio de la ejecución.

Con todo esto se cerrarán todos los programas abiertos y se procederá a actualizar el registro de ejecución. Este archivo simplemente guarda para cada fecha si se ha ejecutado correctamente o ha incurrido en un problema inesperado. Gracias a guardar esta información el usuario puede acceder los datos y ver si simplemente la ejecución automática del programa de trazabilidad de operaciones se está efectuando conforme a la planificación configurada.

Para finalizar, se cerrarán todos los programas abiertos y no consumir más recursos del sistema.

Como se ha podido observar a lo largo de este diagrama de flujo de ejecución se trata una simplificación del problema. Cada paso en realidad tiene muchas más problemáticas y tareas que en realidad se corresponden a un diseño más minucioso de las acciones a tomar. Otro aspecto por notar ha sido como en todo el proceso no ha habido ni una sola vez en la que se ha necesitado la presencia del usuario para tomar una decisión y es lo que se pretende conseguir. Una vez con el programa ya configurado por el usuario según sus preferencias se planea que no tenga la obligación de volver a tocar o cambiar un parámetro para así convertir el proceso de monitorización y control de las operaciones en una tarea más llevadera.

## 4.7 Plan de trabajo

Para este proyecto se ha querido tener organizado el flujo de trabajo de forma que fuera lo más lineal posible y no perderse realizando múltiples tareas al mismo tiempo. Una herramienta que ha servido de mucha ayuda en este aspecto es Trello. [XXX].

Esta herramienta es idónea en la gestión de tareas para diferentes proyectos, desde educativos hasta profesionales para poder así estructurar las tareas a realizar. En esta se pueden anotar ideas, dudas, fechas límite, subtareas y links, todo para simplificar el proceso de llevar a cabo un proyecto que pueda resultar complejo de organizar al tener tantos pasos para completarlo. Esta plataforma puede ir un paso más allá invitando a otros compañeros a participar en tu planificación, aunque en este caso no se ha empleado.

Hay que comprender que en Trello se trabaja por tableros, que podrían verse como una especie de pizarra en la que vas anotando todos los detalles de un proyecto. Cada tablero por su parte se puede separar en diferentes columnas, que en este caso han sido: nuevas ideas, por empezar, desarrollando, por mejorar y acabadas. Las dos alternativas contempladas para trabajar con Trello han sido:

- Crear diferentes tableros para cada fase del proyecto. Se ha descartado al llevar bastante tiempo gestionar varios de ellos pudiendo crear un deseo de abandono al tener que hacer tanta planificación paralelamente.

- Trabajar con un único tablero de forma lineal a medida que se va avanzando en este. Al emplear metodología ágil en cada nueva temporada se establecían una serie de metas a lograr junto a unas fechas límite. En esta misma línea, cuando se cumplían todos los objetivos se terminaba el sprint y se volvían a fijar unas nuevas metas.

Desde luego que los objetivos iniciales nunca eran definitivos y siempre se acababan modificando al aparecer nuevos problemas y funcionalidades que resolver. Gracias a llevar esta planificación junto a Trello se ha simplificado todo este proceso y se ha hecho más llevadero anotar todos los pasos a seguir.



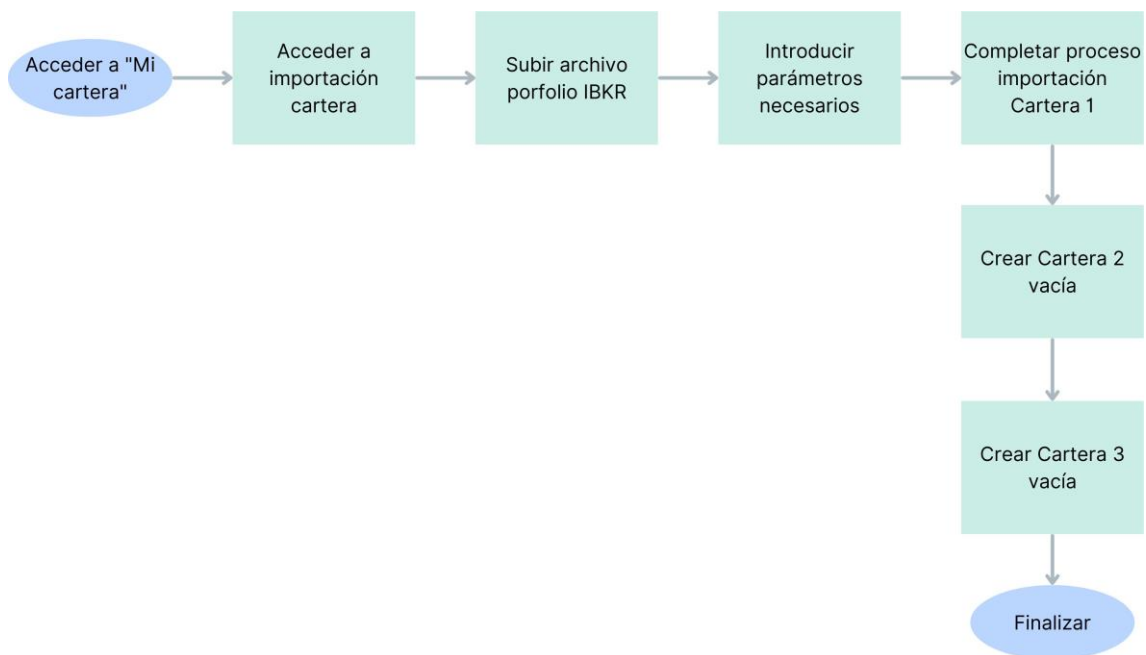


## 5. Diseño

En el capítulo anterior se presenta la solución adoptada de forma general sin adentrarse en todos los procesos a seguir y es aquí donde se detallarán para comprender mejor todas las implicaciones que tendrá.

En este nuevo capítulo se tratará el diseño de la solución adoptada a través de las diferentes partes que la componen. Esta etapa del proyecto es sumamente importante ya que se estructura los pasos a seguir para llevar a cabo las tareas explorando el enfoque adoptado, decisiones clave que orientaron el diseño y se destacaran las visión para alcanzar los objetivos del Trabajo Final de Grado.

### 5.1 Iniciación carteras



*Figura 5.1: Diagrama de flujo iniciación carteras*

En la anterior **Figura 5.1** se presenta el diagrama de flujo seguido para la iniciación de las carteras dentro de Investing.com. Todo este proceso comienza posteriormente de haber accedido a Interactive Brokers y haber descargado el portfolio del inversor con las posiciones actualmente abiertas. De esta forma, navegando a la página web de Inveesting.com y luego a las carteras del usuario se empieza la tarea de crear la Cartera 1. Para esta se seguirá una metodología diferente a las demás puesto que se trata de la cartera principal.

Los pasos por seguir para lograrlo se corresponden al apartado 4.5.2 Análisis plataforma Investing.com en el que se muestran 3 pantallas de configuración de parámetros empezando en primer lugar por subir el archivo .csv que contiene todas las posiciones abiertas.

Con el archivo subido y los parámetros introducidos el proceso de crear la Cartera 1 desde cero quedará completado y ya se podrá empezar a hacer un seguimiento a los activos. Mientras tanto, para las otras dos carteras, Cartera 2 y Cartera 3, se tendrán que crear totalmente vacías sin importar ningún otro archivo. Esto se debe a que al tratarse de la primera ejecución no hay transacciones nuevas que procesar y por tanto no se pueden realizar cambios en dichas carteras, será en próximas ejecuciones cuando se empiecen a abrir nuevas posiciones y cerrarlas.

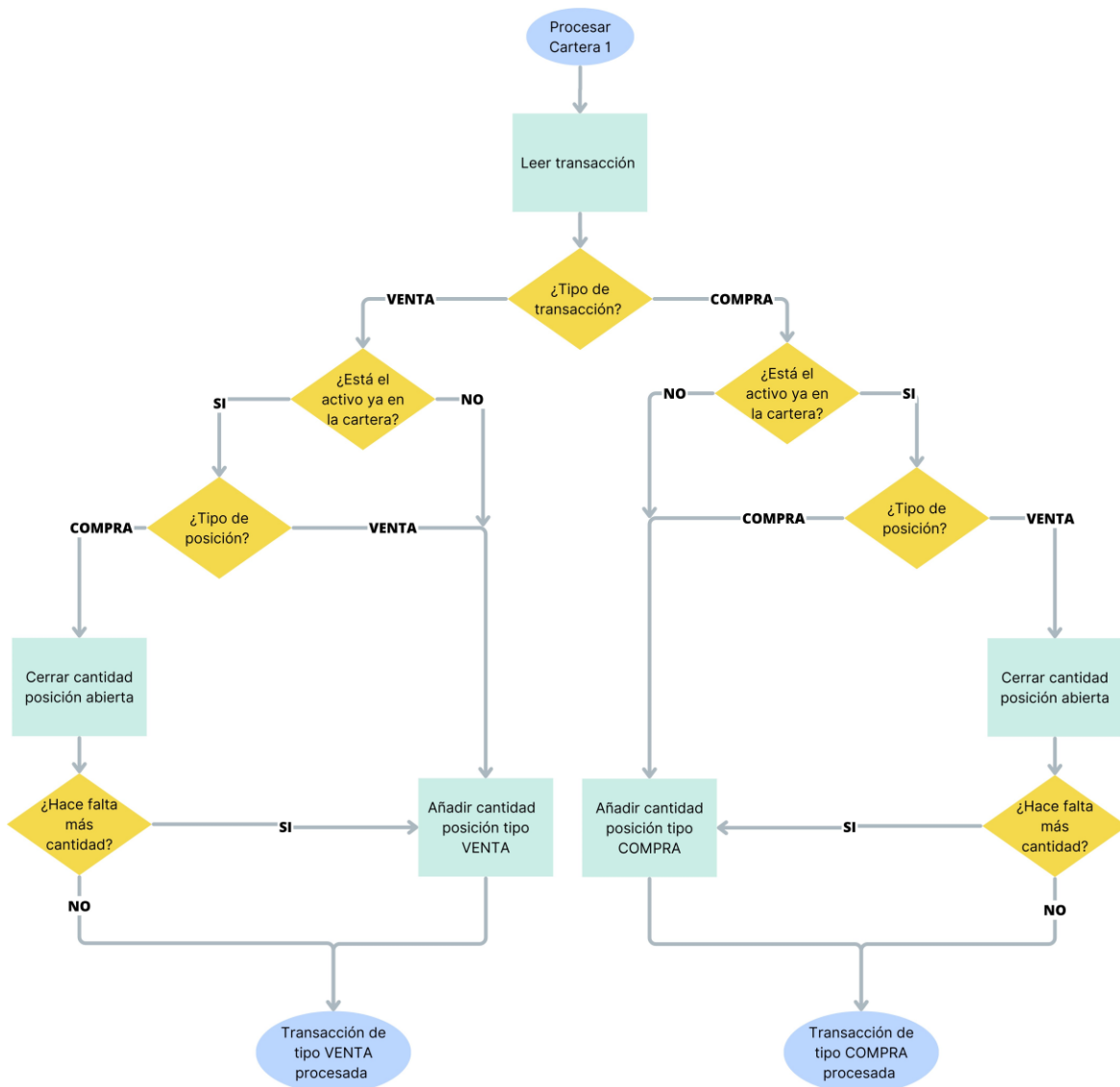
## 5.2 Actualizar carteras

Se trata del proceso que se va a repetir diariamente en el ordenador del inversor. Es importante que el presente código sea eficiente y no gaste muchos recursos para así no afectar el rendimiento del sistema, por este motivo se ha intentado reducir al máximo el número de pasos.

Un aspecto importante que destacar es que no todas las carteras siguen la misma metodología de actualización. Como ya se ha visto en el capítulo 3.5 Lógica de las carteras siguen su propio procedimiento: Cartera 1 es el porfolio actual, Cartera 2 refleja operaciones en largo y Cartera 3 para las operaciones en corto.



**Cartera 1**



*Figura 5.2: Diagrama de flujo procesamiento Cartera 1*

En la **Figura 5.2** se representa el diagrama de flujo que hay que seguir para poder procesar la Cartera 1. Se trata de un flujo bastante largo, pero que se explicará paso a paso para entenderlo ya que es uno de los elementos centrales de este trabajo.

Todo comienza por leer el archivo de transacciones que han ocurrido en los últimos días y viendo en primer lugar de que tipo de transacción se trata:

- **Transacción VENTA:** representa la operación de venta de un activo.
- **Transacción COMPRA:** representa la operación de compra de un activo.

No hay que confundir los anteriores términos con el tipo de las posiciones según se contaba en la sección 4.5.2 Análisis plataforma Investing.com que puede ser a su vez de tipo VENTA o COMPRA:

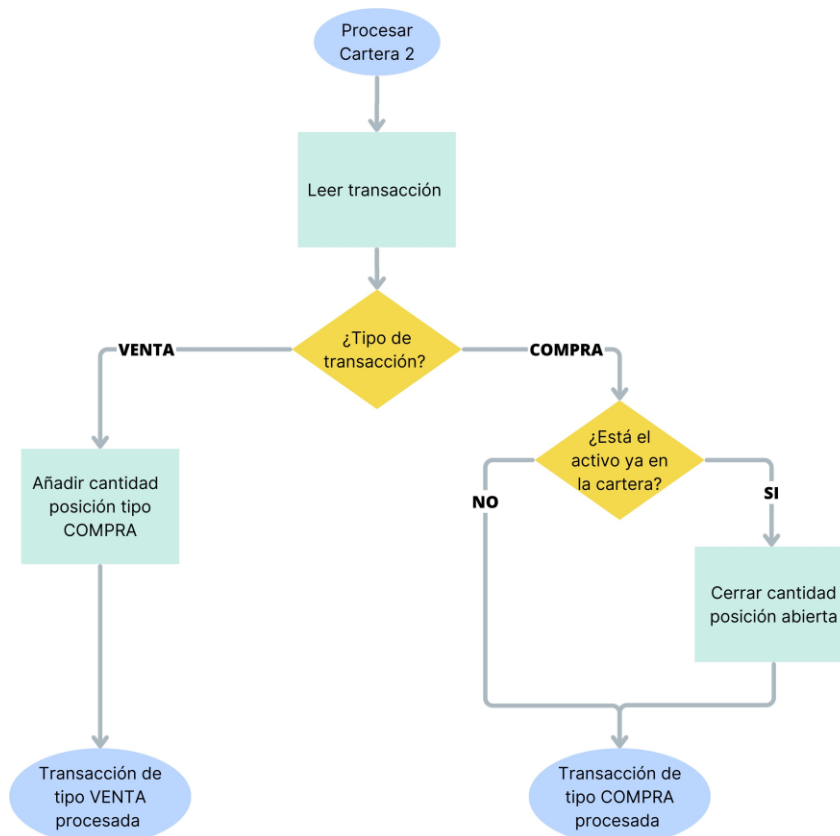
- **Posición VENTA:** representa un activo en posesión que se ha tomado prestado.
- **Posición COMPRA:** representa un activo en posesión que se ha adquirido.

Por un lado, si se trata de una venta lo que hay que comprobar primero es si el activo ya se encuentra dentro de la Cartera 1. En caso de que no sea así y no se encuentre el activo dentro de la cartera se resuelve fácilmente añadiendo una nueva posición de tipo venta con el apartado de “Add position” al mismo tiempo que se fijan todos los parámetros, finalizando de esta forma la transacción actual.

Sin embargo, si se da la situación de que el activo ya se encuentra dentro de la cartera el proceso se complica un poco más especialmente cuando estos son de tipo COMPRA. Lo que hay que hacer es en primer lugar iterar por todas las posiciones actualmente abiertas hasta llegar a la deseada y ahí abrir el desplegable del activo. En este desplegable aparecen múltiples posiciones abiertas para un mismo activo ya que pueden ser en fechas, precio, cantidad diferentes. Empezando por aquellas de la parte inferior, ya que son las más viejas por fecha, se empezarán a cerrar posiciones mediante el botón de “Close position”. En caso de cerrar una posición y no haber alcanzado la cantidad se continuarán cerrando posiciones superiores. No olvidarse en este paso fijar los parámetros correctos de cantidad y precio de cierre adecuado. Para la situación en la que se hayan cerrado todas las posiciones de ese activo y aún no se haya alcanzado toda la cantidad de la transacción habrá que añadir nuevos activos, en este caso del tipo VENTA por la cantidad restante. Este último paso también habría que hacer para la situación cuando haya activos, pero de tipo VENTA. Cuando se alcance la cantidad necesaria con esto habrá finalizado de procesarse la transacción de tipo VENTA.

El proceso por seguir para la situación en que la transacción sea de tipo COMPRA es exactamente la opuesta. Si el activo ya se encuentra dentro de la cartera habrá que ver de qué tipo es: si coincide con el tipo de la transacción simplemente se añadirán nuevas, en caso contrario se empiezan a cerrar posiciones de tipo VENTA hasta llegar a la cantidad. Si se da que no hay suficiente cantidad se añadirán posiciones de COMPRA. Para el otro caso de que el activo no se encuentre en la cartera directamente se añaden posiciones de tipo COMPRA. Cuando se alcance la cantidad necesaria con esto habrá finalizado de procesarse la transacción de tipo COMPRA.

**Cartera 2**



*Figura 5.3: Diagrama de flujo procesamiento Cartera 2*

En segundo lugar, se encuentra el procesamiento de la Cartera 2 a partir de la **Figura 5.3**, mucho más simple que la anterior. Para esta cartera se trata de hacer las operaciones inversas con la condición de que nunca puede haber posiciones de tipo VENTA. Lo que implica la anterior afirmación es que nunca se podrán reflejar activos que se han tomado prestados y vendidos, puesto que eso sería para operaciones en corto y la Cartera 2 refleja operaciones en largo. Para ello se extraen todas esas operaciones en corto y se deja un historial limpio de transacciones y se explicará con más detalle en la Cartera 3.

Se comienza leyendo la transacción y averiguando su tipo. En caso de que sea una transacción de tipo venta se añadirán nuevas posiciones de tipo compra completando así el procesamiento de la transacción de tipo VENTA. ¿Por qué se hace así? Al realizar una transacción de tipo VENTA de una posición de la Cartera 1 de tipo COMPRA significa que se opera en largo y para más adelante se quiere anotar el precio de venta de estos activos. Por este motivo, para tener anotados los precios y cantidad se crea una nueva posición en la Cartera 2.

En caso de que se recompren dichos activos dentro de cierto tiempo al tener una posición abierta ya en la Cartera 2 habrá que cerrarlas por la misma cantidad. En caso de que se compren y no exista una posición abierta en esta cartera no habrá que realizar ninguna operación.

### Cartera 3

Para procesar la Cartera 3 es imposible lograrlo solo a partir de un archivo de transacciones en el que se reflejan las operaciones, ya que depende en todo momento del estado de la Cartera 1. Con esta información queda claro que el funcionamiento de la Cartera 3 está ligado de forma directa al funcionamiento de la Cartera 1. Por este motivo durante la ejecución de la Cartera 1 hay que anotar las operaciones que tengan que ver con el tratamiento de posiciones de tipo VENTA, tal cual se observa en la **Figura 5.4** con los recuadros rojos. En primer lugar, cuando se quieran añadir una cantidad posiciones de este mismo tipo, guardando la operación como si de una venta se tratase. En segundo lugar, cuando se compren acciones de tipo VENTA habrá que anotar esa operación por la cantidad modificada. Cabe destacar que cuando se anotan esas cantidades se quitan de las transacciones que procesará la cartera 2.

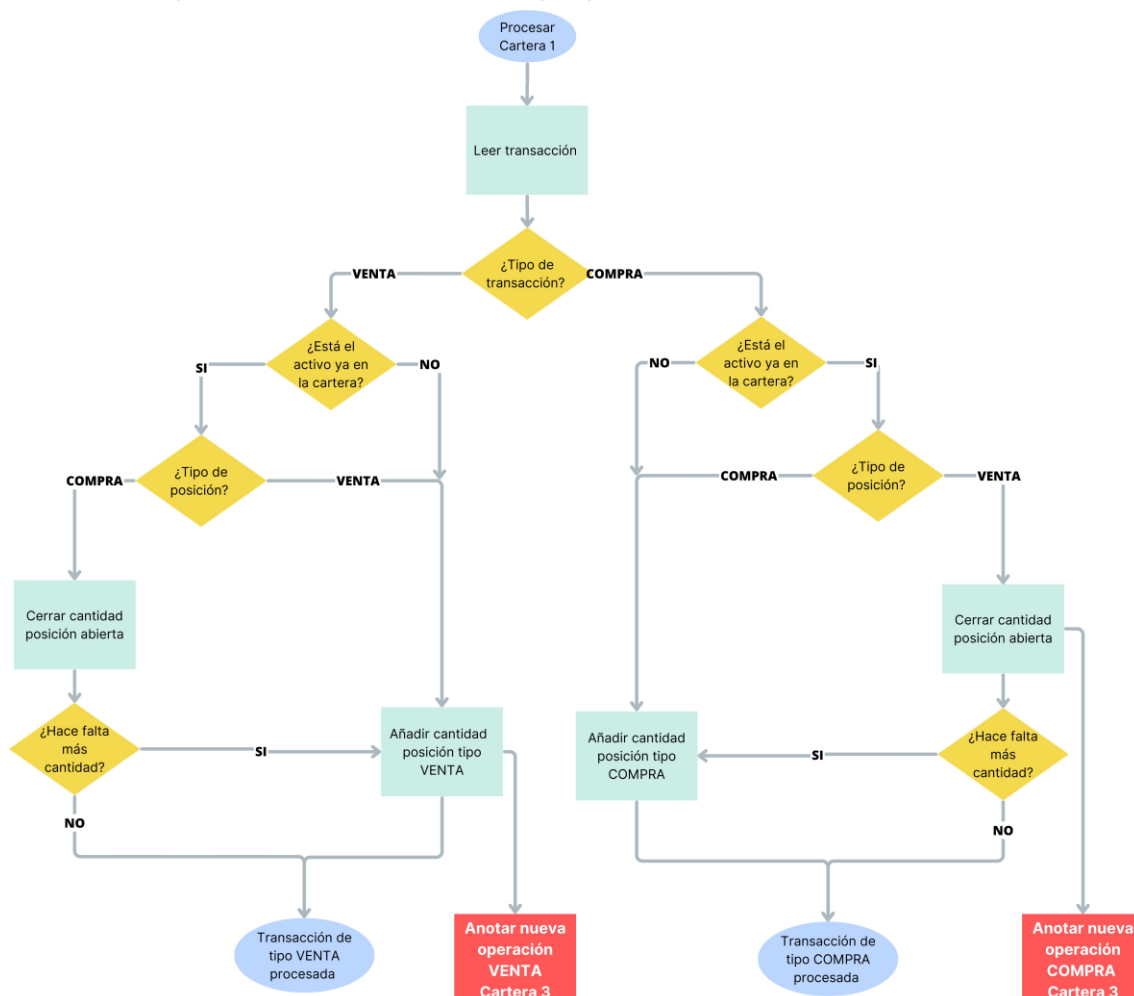


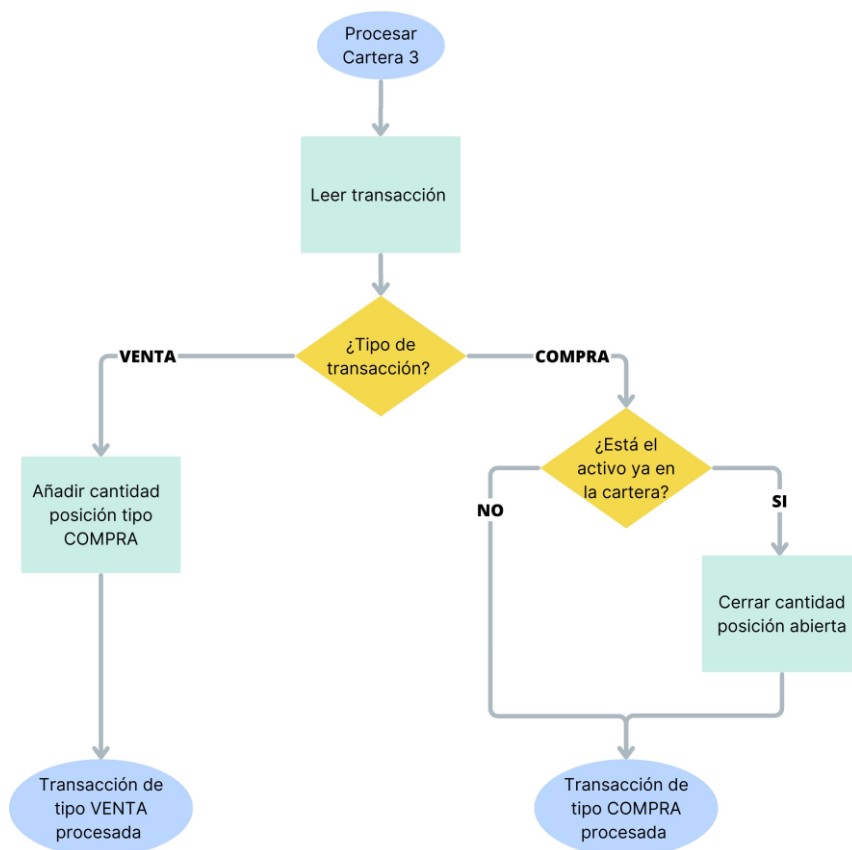
Figura 5.4: Diagrama de flujo procesamiento Cartera 1 - Modificada



Con esto se planea crear un nuevo registro de transacciones que empleará la Cartera 3. En este solo aparecerán transacciones que afecten a esta cartera y el flujo de acciones a seguir es parecido al de la Cartera 2, **Figura 5.5**.

En caso de que la transacción sea de tipo VENTA significa que se están tomando prestados activos para operar en corto y venderlos. Con esto lo que hay que hacer es añadir esa nueva posición a la nueva cartera finalizando así esta rama de ejecución.

Por el otro lado, si es de tipo compra implica que se están recomprando acciones para devolverlas al prestamista. En la Cartera 3 significa que se dejará de operar con ese activo en corto y hay que cerrar la posición. La principal diferencia con la Cartera 2 es para una operación de tipo compra, donde el activo no se encuentra en la cartera, esto implica que el programa ha hecho una inicialización previamente con activos ya prestados y no tiene sentido en futuras ejecuciones tener en cuenta esta operación.



**Figura 5.5:** Diagrama de flujo procesamiento Cartera 3

---

# 6. Implementación

---

En este capítulo se analizarán aspectos relacionados con la puesta en marcha de la aplicación, así como la estructura de los diferentes archivos que componen el programa.

Se revisará en primer lugar como se realiza la conexión a la API de Interactive Brokers a la plataforma de Trader Workstation. A continuación, se revisará el siguiente paso de la automatización tanto de la página web de Investing.com como la interacción con ventanas de Windows. A lo largo de estos apartados se irán creando diferentes archivos para almacenar datos indispensables para la correcta ejecución del programa y se analizará su contenido.

Todos estas funcionalidades están recogidas en 3 archivos de Python que se irán viendo a lo largo de las siguientes páginas: `projectIBKR.py`, `investingAuto.py` y `scripTask.py`. Al final de este apartado se configurará la ejecución diaria del programa a través del Windows Scheduler.

## 6.1 Conexión a la API de IBKR

En esta sección se tratará la conexión a la API de Interactive Brokers, así como posteriormente la descargar de los datos necesarios para conseguir los archivos CSV y poder ejecutar la aplicación. Para lograrlo se importará la librería `ibapi` y así disponer de todas las herramientas necesarias

Toda la funcionalidad anteriormente mencionada se condensa dentro del archivo `projectIBKR.py` en donde se diferencia 2 tipos de conexiones. Una para descargar los datos relacionados con el porfolio y la otra para descargar el histórico de transacciones. En ambas situaciones para lograrlo es necesario definir una nueva clase con la que más tarde se trabajará.

En primer lugar, `porfolioApp` es la clase creada con el propósito de recuperar todas las posiciones abiertas de la cuenta del usuario en Interactive Brokers. En esta se emplean las siguientes estructuras:

- **`__init__(self)`:**
  - Constructor de la clase para configurar los parámetros del objeto a través del cual se interactuará con la API.
- **`nextValidId(self, orderId)`:**
  - Función para indicar que la conexión se ha completado y nuevos mensajes pueden ser enviados a la API, se invoca al recibir el siguiente ID válido. En este caso se emplea para llamar a la función `start`.





- ***updatePortfolio(self, contract: Contract, position: float, marketPrice: float, marketValue: float, averageCost: float, unrealizedPNL: float, realizedPNL: float, accountName: str):***
  - Función para recibir la información referente al porfolio del usuario actualizado. Se trata del método principal de esta clase y que posibilita la obtención de los datos deseados referentes a las posiciones abiertas.
  - Las diferentes variables involucradas en este método son las siguientes:
    - **Contract:** objeto para representar los contratos de los instrumentos financieros. Como se ha explicado en anteriores capítulos puede ser referente a una acción, opción futuro... Dentro de este objeto se almacena toda la información referente como el nombre del activo.
    - **ISIN:** código ISIN para identificar de forma segura el instrumento financiero.
    - **Position:** cantidad de la posición abierta de un instrumento financiero.
    - **MarketPrice:** precio de mercado actual del instrumento financiero, es decir, al que se negocia en este momento.
    - **MarketValue:** valor total de mercado de un instrumento financiero que se tiene en posesión. Se puede obtener multiplicando la cantidad por el marketPrice.
    - **AverageCost:** valor promedio de los instrumentos financieros en posesión.
    - **UnrealizedPNL:** diferencia entre valor total de mercado actual de una posición y el coste promedio de la adquisición.
    - **RealizedPNL:** ganancias o pérdidas obtenidas al cerrar una posición.
    - **AccountName:** nombre de la cuenta de la que se está obteniendo la información. Puede resultar sumamente útil cuando se operan con múltiples cuentas.
- ***accountDownloadEnd(self, accountName: str):***
  - Función para indicar que la recepción de información se ha terminado para la cuenta bajo el nombre de accountName. Puede resultar interesante a la hora de terminar para llamar a otras funciones o indicar la finalización por consola

- **start (self):**
  - Función que comienza la recepción de información a través de la llamada de las funciones indicadas. A su vez realiza una llamada a reqAccountUpdates(True, "") para suscribirse a la información de la cuenta y porfolio, obteniendo todos los datos más recientes. Para esta situación "True" es para indicar que se desea la suscripción a la información y las comillas "" para elegir la cuenta con el ID actual.
- **Stop (self):**
  - Función para detener y desconectar la conexión con la plataforma Trader Workstation. Es muy importante no olvidarse de usar esta función ya que sirve para liberar los recursos, como los Id de conexiones, además de evitarse problemas de seguridad.

En segundo lugar, transactionApp es una clase empleada para recuperar el historial de transacciones de los últimos 7 días. Esta clase cuenta con una estructura muy similar a la anterior, ya que las únicas diferencias que presenta son en el constructor y updatePortfolio se sustituye por execDetails:

- **\_\_init\_\_(self):**
  - Esta vez en el constructor se definen las siguientes variables para posteriormente guardar los datos de cada transacción:
    - **Id:** identificador de la transacción. Puede resultar útil a la hora de localizar ciertas transacciones.
    - **Sectype:** tipo de contrato financiero al que se refiere el contrato de la transacción. Por ejemplo, para representar las acciones sería STK.
    - **ISIN:** código ISIN para identificar de forma segura el instrumento financiero.
    - **Exchange:** mercado en el que se ha negociado el instrumento financiero. Por ejemplo, puede ser la bolsa NYSE de Nueva York, NASDAQ...
    - **Currency:** tipo de moneda empleada en la transacción para el intercambio. Para dólares americanos es USD, para euros es EUR.
    - **Action:** acción realizada en el contrato, puede ser tanto como de compra "BOT" o venta "SLD".
    - **Quantity:** cantidad intercambiada en la transacción del instrumento financiero.

- **ExecutionPrice:** precio de ejecución al que se ha ejecutado la transacción.
  - **ExecutionTime:** hora de ejecución a la que se ha ejecutado la transacción.
- **execDetails(self, reqId, contract, execution):**
    - Método principal de la clase encargado de que para cada transacción en el sistema ir guardando los datos en las variables del constructor anterior. En este proceso se ven implicadas estas nuevas variables:
      - **reqId:** identificador de la ejecución de esta solicitud.
      - **execution:** objeto que representa una transacción en el que se almacenan datos relacionados a este como podrían ser la hora de ejecución, identificador de la cuenta, precio....

Adicionalmente, en el momento del desarrollo de la aplicación se detectó un problema bastante importante al no poder disponer de la variable ISIN, ya que las clases de por sí en realidad no incluían este dato. Por este motivo se ha creado un método adicional `get_isin(id)` en el que al pasarle un identificador de contrato se devuelve todos los datos relativos a este. En esta tarea se ha empleado la librería `ib_insync`

Como se ha podido ver la estructura de las anteriores dos clases es relativamente simple y sin muchas complicaciones. Con ellas ya definidas solo queda usarlas para poder crear una conexión la API y de esta forma obtener todos los datos necesarios. Para ello, hay que seguir los siguientes pasos:

1. Crear y guardar una instancia de la nueva clase. Puede ser tanto `transactionApp()` como `portfolioApp()`.
2. Establecer la conexión de la variable anterior con la plataforma Trader Workstation. Los parámetros de esta conexión son:
  - **IP:** en el presente caso se usará siempre `127.0.0.1` porque se quiere establecer la conexión para esta máquina se está usando en este momento.
  - **Puerto:** en el presente caso se usará siempre `7497` ya que es el puerto predeterminado fijado por la aplicación. Si se desea siempre se puede cambiar.
  - **Número de cliente:** la aplicación permite la conexión de varios clientes simultáneos, pero estos no se pueden solapar. Se puede poner `0` para empezar por el primer orden.
3. Ejecutar la aplicación mediante el método `start()`.
4. Ejecutar el método `stop()`
5. Ejecutar el método `run()` para gestionar todos los mensajes que se van a recibir a continuación.

Con todos estos pasos se han recogido toda la información de las transacciones, sin embargo, queda ver como se almacenan.

## 6.2 Guardado de los datos en CSV - Pandas

En la anterior sección se ha presentado el problema de tener que ir guardando los datos que se obtienen. La alternativa elegida para solucionarlo es Pandas. Se trata de una librería empleada a menudo en el análisis de datos para manipular, limpiar, realizar análisis y poder obtener diferentes visualizaciones de los datos [32]. Una de las principales ventajas es que se establecen estructuras de datos, como el *Dataframe*, para así trabajar de forma eficiente con una gran cantidad de información.

Un *Dataframe* no es más que una estructura bidimensional de datos empleada para organizar los datos en un formato tabla, es decir, en filas y columnas. Sin embargo, el manejo constante de cada una de las celdas que la componen puede llegar a complicarse si no fuera por los diferentes métodos que ofrece Pandas para modificar, eliminar o simplemente visualizar la información. Con todo esto, se ha elegido Pandas como librería para manejar los datos por su facilidad de uso, eficiencia y flexibilidad a la hora de trabajar.

En esta sección se va a explicar que archivos se van generando a lo largo de la ejecución del código y permanecen en el sistema. El motivo por el que se ha optado por el guardado de ciertos datos es el siguiente:

1. **Comprobar el estado:** guardando los archivos en el sistema el usuario puede comprobar que no haya un problema de sincronización entre el programa y las diferentes plataformas con la que se interactúa.
2. **Ejecuciones futuras programa:** antes de suprimir los datos viejos se comprueba que datos había anteriormente.
3. **Ejecución actual:** durante la misma ejecución se leen repetidamente esos datos para efectuar operaciones en las plataformas.

### Portfolio.csv

Este archivo CSV es generado por `projectIBKR.py`. Se trata de la información que refleja el estado del porfolio actual. En este archivo cada fila corresponde a una posición abierta del inversor. En la siguiente **Figura 6.1** se puede observar como en la primera fila se corresponde a una posición de Apple, su código ISIN, STK para indicar que es una acción, USD para los dólares, cantidad y precio promedio. En realidad habría muchos más datos que se podrían haber extraído, sin embargo, no serían de ninguna utilidad al subirlos en la plataforma de Investing.com.

Symbol ▲ ▼	ISIN ▼	Sec Type ▼	Currency ▼	Quantity ▼	Average Cost ▼
AAPL	US0378331005	STK	USD	48	190.27
AMZN	US0231351067	STK	USD	56	129.84
BBVA	US05946K1016	STK	USD	40	7.71
FORD	US3498623004	STK	USD	15	0.93
GOOG	US02079K1079	STK	USD	55	119.1
LYFT	US55087P1049	STK	USD	64	10.25
META	US30303M1027	STK	USD	45	289.19
MSFT	US5949181045	STK	USD	12	335.15
TSLA	US88160R1014	STK	USD	-8	271.92

Figura 6.1: Aspecto portfolio.csv

### TransactionsXXX.csv

El siguiente archivo CSV también es indispensable para la ejecución del programa, donde cada fila representa una transacción u operación efectuada. Este se compone de gran cantidad de datos que se han explicado anteriormente: un identificador para la transacción, el símbolo al que hace referencia, el identificador ISIN, si se ha vendido “SLD” o comprado “BOT”, la cantidad implicada en la operación, el precio y la hora exacta en la que se ha completado. Todos los anteriores detalles se pueden ver en **Figura 6.2** Este archivo también es generado dentro de projectIBKR.py.

Este archivo tiene un detalle muy importante que radica en su nombre, en este se refleja la ultima hora de ejecución. Un ejemplo podría ser “Transactions20230731 191315.csv” para indicar que se ha generado por última vez a las el 31 de Julio de 2023 a las 19:13:15. Esto ayuda posteriormente al generar un nuevo archivo para borrar todas las transacciones más viejas que esa fecha y de esa forma no volver a procesarlas.

Id ▼	Symbol ▼	ISIN ▼	Action ▼	Quantity ▼	Price ▼	Time ▼
00012ec5.64c732a6.01.01	AAPL	US0378331005	SLD	5	195.95	2023-07-31 10:00:23
00012ec5.64c7330b.01.01	TSLA	US88160R1014	SLD	4	266.75	2023-07-31 10:01:18
00012ec5.64c75aab.01.01	META	US30303M1027	SLD	6	325.55	2023-07-31 15:39:31
00012ec5.64c75ab5.01.01	META	US30303M1027	SLD	3	325.55	2023-07-31 15:39:31
00012ec5.64c7abee.01.01	AAPL	US0378331005	SLD	2	195.94	2023-07-31 18:13:25
00012ec5.64c7abf0.01.01	AMZN	US0231351067	SLD	1	133.1	2023-07-31 18:13:29
00012ec5.64c7ac8c.01.01	TSLA	US88160R1014	BOT	3	267.73	2023-07-31 18:16:56
00012ec5.64c7b459.01.01	LYFT	US55087P1049	BOT	1	12.61	2023-07-31 19:02:14

Figura 6.2: Aspecto transactionsXXX.csv

## Registry.csv

El último archivo generado es más bien un registro que lleva las ejecuciones del programa de trazabilidad. En este cada vez que se ejecuta un programa se guarda una nueva entrada: si se ha completado o ha resultado en un error, la fecha exacta y el mensaje de error en caso de que hubiese alguno. En la **Figura 6.3** hay 2 ejecuciones que se han completado satisfactoriamente, mientras que en la última ha resultado en un error por acceder a un índice fuera del rango de una lista. Gracias a esto es más fácil saber si se está ejecutando bien el programa y en caso de que haya un error se tenga una noción del posible problema.

Execution	Date	Error Message
COMPLETED	19:10:40 31/07/2023	
COMPLETED	19:12:24 31/07/2023	
ERROR	20:08:55 06/08/2023	list index out of range

Figura 6.3: Aspecto registry.csv

## 6.3 Automatización acceso a Investing

Ya se dispone de todos los datos necesarios para iniciar la automatización en Investing.com. Como ya se ha comentado esta plataforma no dispone de ninguna API para poder simplificar las tareas a realizar, complicando así todo el proceso. Por una parte, se tendrá que automatizar el acceso a la página web, navegando entre los diferentes apartados y automatizar acciones en las ventanas de Windows, abriendo y cerrando programas.

### 6.3.1 Automatización ventanas

El primer paso para que el programa funcione es iniciar el programa de Trader Workstation. Cabe recordar que sin esta plataforma de Interactive Brokers no se podrían obtener los diferentes datos de la cuenta del inversor, es decir, sirve como enlace para poder acceder a la API. Por ello, mediante pyautogui, una librería que tiene el propósito de automatizar interacciones con aplicaciones del sistema operativo Windows, macOS y Linux [33].

Para lograr este paso en primer lugar se inicia el programa Trader Workstation a través de la librería os con "startfile". A continuación con el método typewrite("texto") se introducen las credenciales del usuario: nombre de usuario y contraseña de la cuenta. A continuación, para interrumpir lo menos posible las actividades del usuario se minimiza la ventana con el siguiente código:

```
pyautogui.getWindowsWithTitle("Interactive")[0].minimize()
```

Lo que hace es identificar la ventana a la que se quieren dirigir las acciones, en este caso aquella que empiece con la palabra “Interactive”. Una vez ubicada se le indica que se quiere minimizar. Lógicamente después de completar el proceso de generar los CSV de portfolio y transacciones se procederá a cerrar el programa con close().

### 6.3.2 Selenium

Selenium es una herramienta de código abierto de gran utilidad a la hora de automatizar navegadores. Gracias a ella se podrán simular acciones de los usuarios para interactuar con las páginas web en tareas que sean repetitivas [34]. Las operaciones que se pueden realizar mediante esta Librería pueden ser muy variadas: rellenar información, presionar diferentes botones, localizar la existencia de elementos... Aparte de la automatización del navegador y realizar pruebas Selenium se usa a menudo para el Web Scrapping. Este término no es más que extraer información de sitios web por los que se navega y así tener los datos necesarios para otros procesos [35].

Por lo tanto, se puede navegar como si se fuese un usuario real con la única condición que se conozca de antemano la estructura HTML de la página y adaptar el código para interactuar con esta. Un inconveniente de esta solución es que si la estructura del HTML cambia drásticamente puede suponer que el código de automatización necesite una actualización a los nuevos parámetros para poder volver a hacerlo funcionar.

Los recursos necesarios para poner en marcha un programa de automatización empleando Selenium no es más que instalar e importar la librería, seguido de tener disponible un navegador web con su controlador. Este último puede ser prácticamente cualquiera de los más conocidos ya que tienen soporte: Google Chrome, Edge, Mozilla Firefox o Safari. En el caso de este proyecto al emplear Chrome como navegador se tiene que descargar su propio controlador de la página oficial y ubicarlo en el mismo directorio que el programa [36].

Con todos los elementos ya listos se puede empezar a navegar por las páginas web. Es recomendable, al igual que en el código creado en este trabajo establecer una serie de parámetros del controlador para que funcione mejor el programa:

- **Headless:** permite controlar el navegador sin tener que ver la ventana del programa. En otras palabras, permite hacerlo como si se tratase de un programa en segundo plano.
- **Start-maximized:** agrandar la ventana del navegador.
- **Windows\_size:** establecer el tamaño de la ventana.
- **Disable notifications:** impedir que aparezcan notificaciones de las páginas en las que se accede.

- **Eager:** parámetro que permite ejecutar el código del programa tan pronto como la estructura HTML de la página esté disponible. La ventaja es que agiliza las ejecuciones, pero puede llegar a consumir más recursos al tener la página cargando al mismo tiempo que ya se procesan operaciones de la automatización.
- **Exclude switches:** parámetro para deshabilitar ciertos interruptores de líneas de comandos que ocurren al iniciar Chrome.

Con el controlador ya importado y los parámetros establecidos para interactuar con un elemento de una página solo hay que buscarlo. En la solución adoptada en este caso los tipos de búsqueda que se han empleado son las siguientes:

1. CSS Selector
2. XPath
3. Id
4. Link text
5. Class name
6. Name

### 6.3.3 Tarea diaria de acceso a Investing

Todos los aspectos de la anterior sección se ven reflejados en el archivo `investingAuto.py`. En este se automatiza el acceso a `Investing.com` para subir y actualizar los datos de forma diaria manteniendo las carteras del inversor al día. Este archivo es el más grande al contener multitud de métodos que posibilitan y facilitan interactuar con la página web.

Desde una perspectiva más amplia el código se divide en 2 procesos generales que en realidad se corresponden con las diferentes etapas de los diagramas de flujo de la sección 5:

- **Process portfolio:** método encargado de procesar el portfolio que se pasa como entrada. Dependiendo de si es Cartera 1, Cartera 2 o Cartera 3 se realizarán diferentes acciones.
- **Portfolio initiation:** método encargado de que en caso de que sea la primera ejecución o la información presente en `Investing.com` esté anticuada suba el portfolio a la Cartera 1 y cree vacías la Cartera 2 y 3.

A lo largo de la invocación de estos métodos principales hay una multitud de otros métodos que se invocan en operaciones que se repiten a menudo: añadir posición,





cerrar posición, cambiar pestaña de cartera, encontrar posición, diferentes *get* para conseguir datos específicos, expandir detalles de una posición...

## 6.5 Estructura scripTask.py

ScriptTask.py se constituye como el archivo Python principal de este programa. Es el encargado de coordinar todos los elementos para que haya una ejecución planificada y asegure que no se interrumpa hasta la finalización.

En primer lugar, se abre un archivo llamado “userCredentials.txt” para obtener las credenciales. Este es un archivo de texto que contiene los nombres de usuario y contraseñas en las plataformas de Interactive Brokers e Investing.com. Una vez obtenidas las cuentas lo primero que se hace es establecer que el sistema operativo no se apague si así lo pide el usuario, puesto que sería imposible volver a sincronizar las plataformas. Para ello mediante la librería “kernel132” de Windows se llama a función “SetThreadExecutionState()” con los siguientes parámetros:

- **ES\_CONTINUOUS:** valor para indicar que en caso de que el usuario esté inactivo el sistema no debe entrar en modo suspensión.
- **ES\_SYSTEM\_REQUIRED:** valor para indicar que en caso de que el usuario solicite la suspensión se impida la acción.

A continuación, se crea la carpeta “Datos” en caso de que no exista en la que se irán guardando los diferentes archivos de datos CSV y se realizan las acciones analizadas en la sección 6.3.1 Automatización ventanas. En esta sección en resumidas cuentas se abre Trader Workstation, se importa el portfolo y las transacciones, finalizando con la ejecución de investingAuto.py. Al final se introduce una nueva entrada en el archivo registry.txt y se devuelve el estado del sistema a la configuración anterior para poder apagar el sistema operativo.

## 6.6 Configurar programa por primera vez

El actual programa está publicado en GitHub para que puedan acceder a él otros usuarios interesados en el proyecto. La configuración inicial del programa es bastante fácil y no requiere avanzados conocimientos en informática. Adicionalmente en GitHub se han detallado los pasos a seguir de una forma más resumida y fácil de entender sin entrar en detalles que puedan llevar a confusión.

Los requisitos para poder iniciar el programa de trazabilidad son los siguientes:

- Instalar Python [\[37\]](#)
- Instalar Trader Workstation [\[38\]](#).<sup>1</sup>
- Clonar el repositorio GitHub para obtener el [código del programa](#).
- Instalar dependencias del archivo “requirements.txt” con “`pip install -r requirements.txt`”
- Tener una cuenta en Interactive Brokers e Investing.com e introducirlas en “userCredentials.txt”

Con todo esto ya se tienen todos los elementos listos para la ejecución. Ahora hay que configurar para que se ejecute de forma periódica sin tener que intervenir cada vez. Para ello, se empleará la herramienta de Windows Scheduler o Programador de Tareas con los siguientes pasos a seguir:

1. Acceder al Programador de tareas.
2. Presionar “Crear tarea básica”

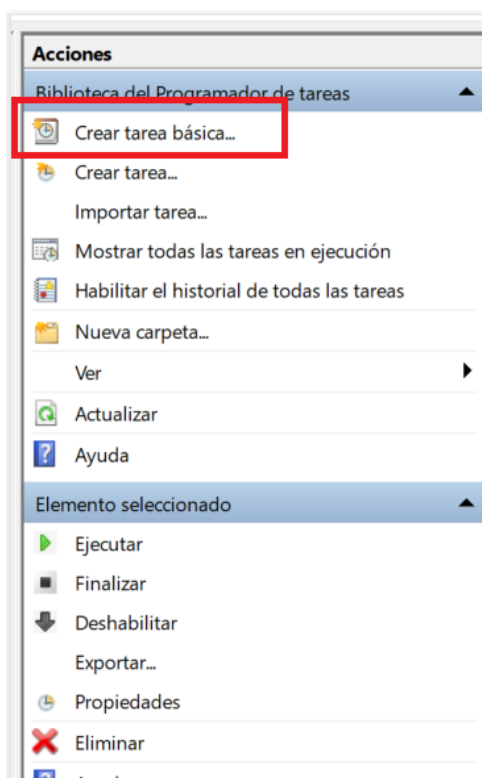


Figura 6.4: Panel de acciones Programador de tareas

<sup>1</sup> En el anexo [A.1 Instalación y configuración Trader Workstation](#) se encuentra de forma detallada como realizar la instalación de forma correcta y configurando todos los parámetros necesarios.

3. Ponerle un nombre a la tarea.

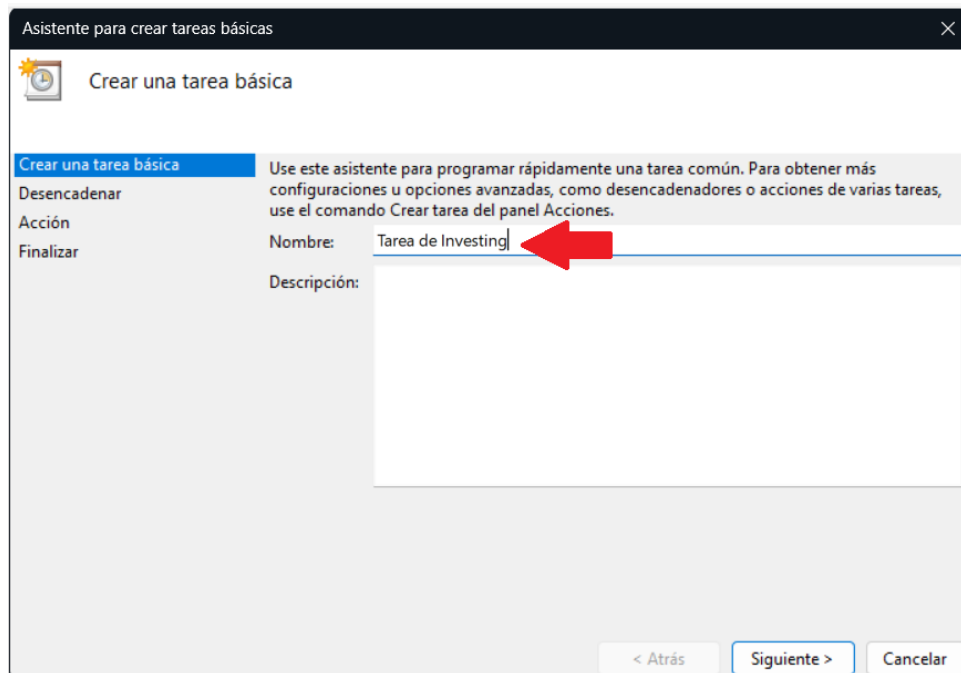


Figura 6.5: Crear una tarea básica

4. Elegir la frecuencia de ejecución. Se elegirá diariamente con ejecución cada día a las 12:00:00

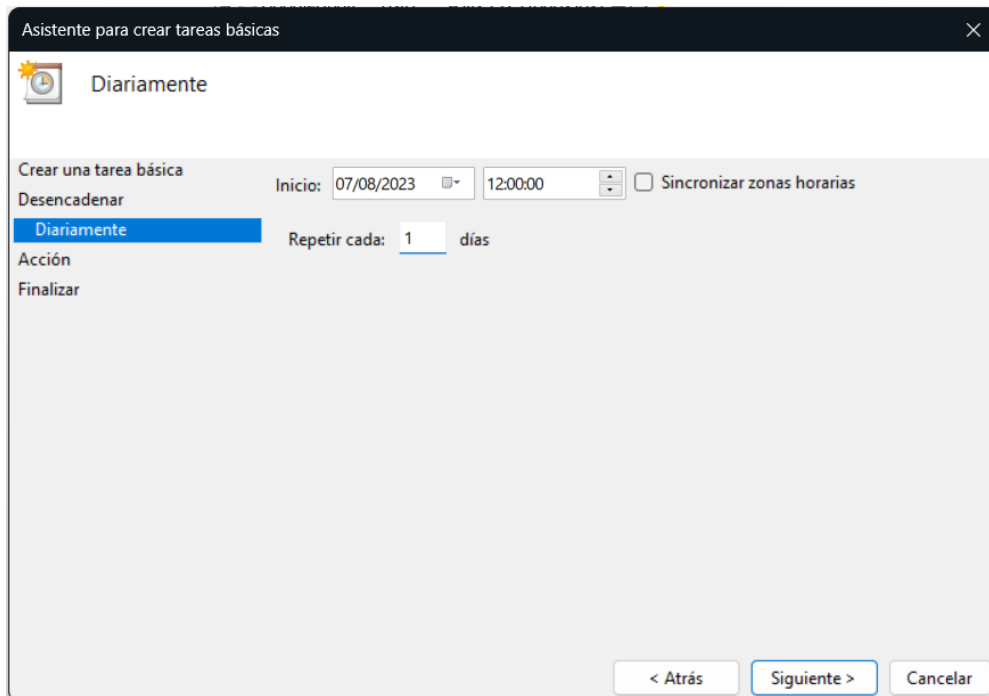


Figura 6.6: Definir frecuencia tarea

5. Elegir acción por realizar “Iniciar un programa”

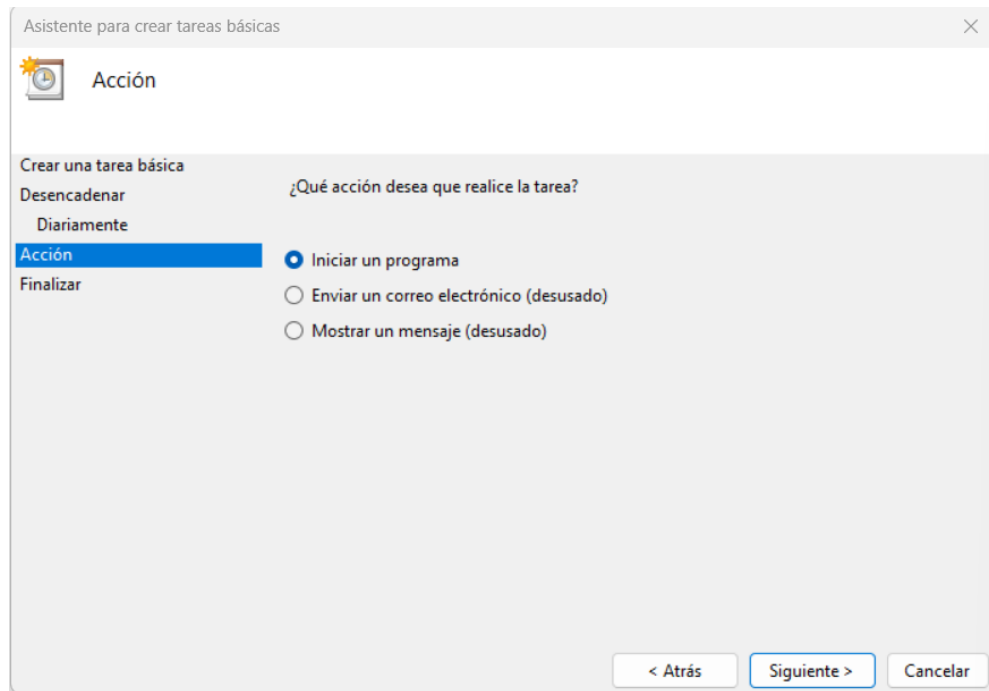


Figura 6.7: Tipo de acción a elegir

6. Indicar la ruta para la ejecución del programa:
- a. Programa o script: directorio del ejecutable .exe de Python.
  - b. Agregar argumentos (opcional): nombre del archivo Python a ejecutar.
  - c. Iniciar en (opcional): directorio de la carpeta que contiene el archivo anterior.

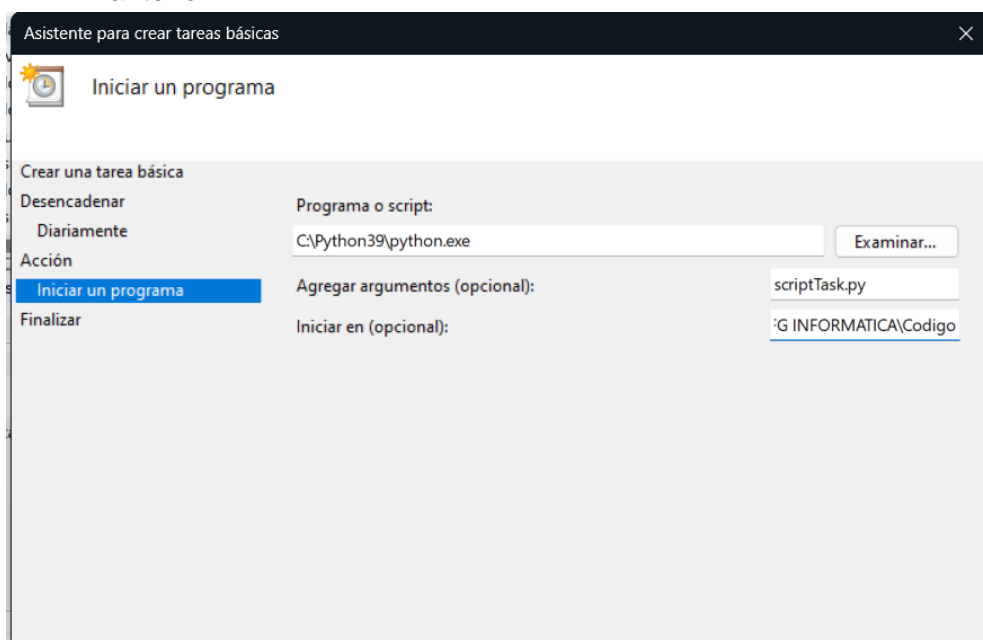


Figura 6.8: Localización archivo a ejecutar

## 7. Finalizar la creación de la tarea

Además de la configuración anterior posteriormente hay que entrar en las propiedades de la tarea creada y activar los siguientes parámetros:

- Activar “Iniciar la tarea solo si el equipo está inactivo durante 1 minuto y Esperar a que esté inactivo durante 2 horas
- Activar “Iniciar solo si la siguiente conexión de red está disponible: Cualquier Conexión”
- Activar “Ejecutar la tarea lo antes posible si no hubo inicio programado”
- Activar “Si la tarea no se ejecuta, reiniciarla cada” 1 hora, con 2 reinicios.

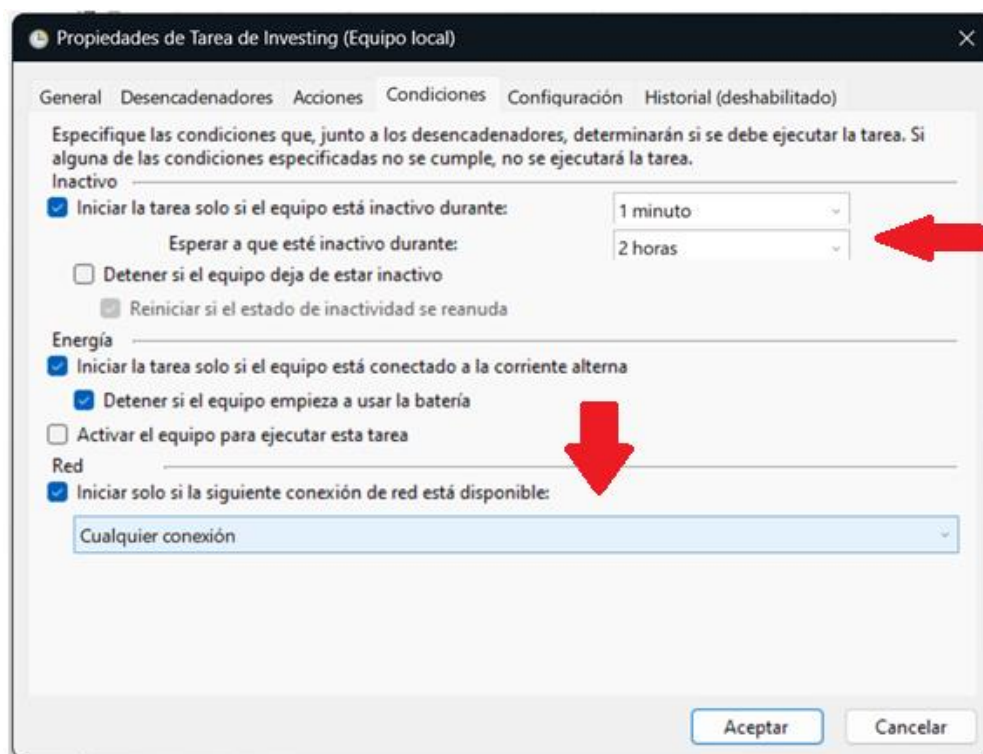
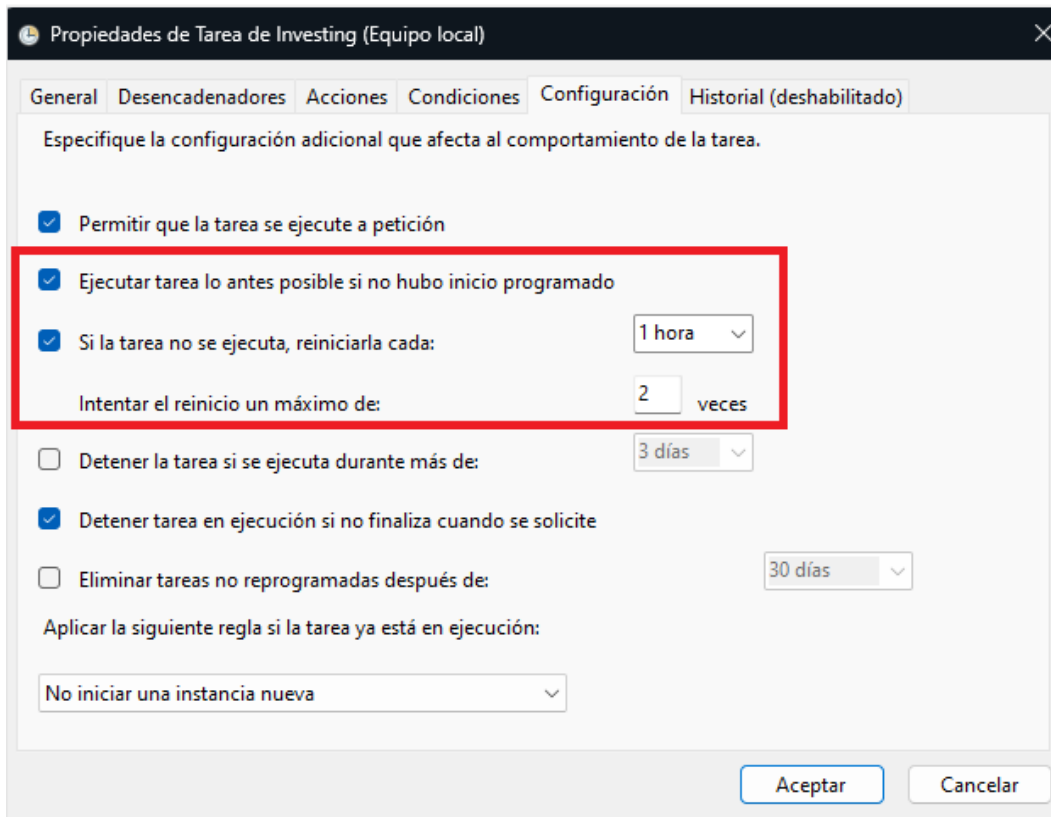


Figura 6.9: Parámetros adicionales - Parte 1



*Figura 6.10: Parámetros adicionales - Parte 2*

Con todos estos pasos el programa se ejecutará de forma diaria sin necesidad de que el usuario se preocupe por el programa de trazabilidad. Al mismo tiempo se ha priorizado la ejecución cuando este no esté realizando tareas delante del ordenador y en caso de que no ocurra se acabará ejecutando más adelante.

Si se quiere hacer para MacOS o Linux basta con hacer estos pasos y cambiar el 12 por la hora deseada y "path\_to\_file" con la ubicación de scriptTask.py:

1. Acceder a la terminal .
2. Escribir "crontab -e" para acceder al planificador de tareas.
3. Escribir "0 12 \* \* \* \* python3 path\_to\_file".

---

## 7. Test y pruebas

---

En este último capítulo antes de cerrar el proyecto se ha comprobado la funcionalidad del programa para ver si los resultados proporcionados son adecuados. Este aspecto implica poner a prueba el código para comprobar que no ocurran errores inesperados que puedan interrumpir el proceso llevado a cabo. Por tanto, se va a comprobar por una parte que el código esté libre de errores y que además efectúe las operaciones correctas en cada momento para tener las carteras correctamente actualizadas.

### 7.1 Pruebas unitarias

Las pruebas unitarias no son más que evaluaciones específicas que se realizan a lo largo de un desarrollo software de cada una de las partes que componen un programa. Cada prueba se enfoca a supervisar el funcionamiento de un método o función y así asegurar que trabajan debidamente. Todo el proceso se debe realizar en un entorno controlado para obtener unos resultados esperados, por este motivo los datos de entrada de las pruebas están predefinidas para que cada unidad funcione según lo esperado y de forma aislada del sistema [39].

Las pruebas unitarias son imprescindibles en la detección temprana de fallos que más tarde puedan resultar en mayores costes y tiempo de corrección. De esta forma, además del motivo anterior también ayuda a mejorar la claridad del código al motivar al programador a crear métodos modulares fomentando cambios seguros y mejoras en el código para el futuro, aportando sistemas mucho más estables y fáciles de mantener desde el punto de vista de software.

Para poder lograr este objetivo se ha importado la librería “unittest” enfocada específicamente en esta tarea. El funcionamiento es muy simple, los aspectos a destacar del archivo test.py que se ha creado son los siguientes:

- **Creación de clases de pruebas:** sirven para separar las pruebas que se realizan en diferentes fragmentos para no superponer la lógica empleada en cada una de ellas. Cabe destacar que en la ejecución del archivo test.py se ejecutan paralelamente cada una de ellas para obtener los resultados mucho más rápido. Un ejemplo de definición de una clase de unittest es el siguiente:

```
class TestPortfolioApp(unittest.TestCase):
```

- **Creación del entorno controlado:** como se ha comentado previamente es imprescindible para las pruebas unitarias crear un entorno en que las entradas sean datos predefinidos. Por este motivo se deben definir un punto de partida en el que se establece la información a partir de la cual todas las pruebas unitarias deben partir. En primer lugar, se define un método de setUp para establecer

dichos datos y finalmente si es necesario un `tearDown` para cerrar de forma adecuada la prueba. La forma de hacer es la siguiente:

```
def setUp(self)
def tearDownClass(self)
```

- **Creación de cada prueba:** dentro de cada clase se definen las pruebas a realizar. Un aspecto importante es que en este trabajo era necesario una ejecución secuencial de las pruebas en una de las clases y la librería `unittest` ejecuta los métodos por orden alfabético. Se ha puesto un nombre a esos métodos con el siguiente formato `test_X`, siendo la X las letras por orden alfabético. Dentro de cada una de ellas al final se deben comprobar que los resultados son adecuados mediante las llamadas a las funciones `assert`, por ejemplo, aquí se comprueba que la cartera PRUEBA se ha creado y existe en la página:

```
self.assertTrue("PRUEBA" in driver.page_source)
```

De esta forma se han creado las pruebas unitarias separadas en las siguientes clases de prueba de la **Tabla 7.1**:

Clase	Objetivo
TestPortfolioApp	Comprobar la clase PortfolioApp tanto en sus métodos como atributos del constructor.
TestTransactionApp	Comprobar la clase TransactionApp tanto en sus métodos como atributos del constructor.
TestGeneralIBKR	Comprobar los métodos de generación de los datos CSV.
TestInvesting	Comprobar los métodos de la automatización de Investing

*Tabla 7.1: Clases de pruebas creadas*

## 7.2 Prueba de integración

Además de las pruebas unitarias se ha creado una prueba de integración. Esta tiene el objetivo de asegurar que los métodos funcionan bien en cooperación. A menudo suele ocurrir que en pruebas aisladas las funciones se ejecutan correctamente pero cuando interactúan con otros elementos pueden originar múltiples errores. Por este



motivo para el archivo de autoInvesting.py se ha creado unas transacciones ficticias con el siguiente aspecto:

<b>Id</b>	<b>Symbol</b>	<b>ISIN</b>	<b>Action</b>	<b>Quantity</b>	<b>Price</b>	<b>Time</b>
1	AAPL	US0378331005	BOT	15	10	09/08/2023 19:23
2	TSLA	US88160R1014	SLD	40	10	09/08/2023 19:23
3	AAPL	US0378331005	BOT	15	10	09/08/2023 19:23
4	AAPL	US0378331005	SLD	40	10	09/08/2023 19:23
5	GOOGL	US02079K3059	BOT	50	10	09/08/2023 19:23
6	TSLA	US88160R1014	SLD	40	10	09/08/2023 19:23
7	AAPL	US0378331005	SLD	30	10	09/08/2023 19:23
8	GOOGL	US02079K3059	BOT	50	10	09/08/2023 19:23
9	AAPL	US0378331005	BOT	30	10	09/08/2023 19:23
10	AAPL	US0378331005	SLD	10	30	09/08/2023 19:23
11	GOOGL	US02079K3059	SLD	20	10	09/08/2023 19:23
12	AAPL	US0378331005	BOT	5	30	09/08/2023 19:23
13	AAPL	US0378331005	BOT	20	30	09/08/2023 19:23
14	AAPL	US0378331005	SLD	50	30	09/08/2023 19:23
15	AAPL	US0378331005	BOT	80	30	09/08/2023 19:23
16	TSLA	US88160R1014	BOT	10	10	09/08/2023 19:23
17	AAPL	US0378331005	SLD	20	30	09/08/2023 19:23

**Tabla 7.2:** Contenido historial transacciones para la prueba

Además, la cartera 1 previamente contenía la siguiente información de la **Tabla 7.3:**

<b>Symbol</b>	<b>ISIN</b>	<b>SecType</b>	<b>Currency</b>	<b>Quantity</b>	<b>AverageCost</b>
AAPL	US0378331005	STK	USD	10	190.1228
LYFT	US55087P1049	STK	USD	40	187.22

**Tabla 7.3:** Contenido Cartera1 para la prueba

Con todos los datos anteriores como input para la prueba se ejecuta la creación y actualización de la carteras de Investing mediante la invocación del método principal. Al final se han comprobado que los componentes de las carteras coinciden con los resultados esperados de la **Tabla 7.4** mediante llamadas a la función assert:

<b>Cartera</b>	<b>Symbol</b>	<b>Type</b>	<b>Quantity</b>
<b>Cartera 1</b>	AAPL	BUY	25
	GOOGL	BUY	80
	TSLA	SELL	70
	LYFT	BUY	40
<b>Cartera 2</b>	AAPL	BUY	20
	GOOGL	BUY	20
	TSLA	BUY	0
<b>Cartera 3</b>	AAPL	BUY	0
	GOOGL	BUY	0
	TSLA	BUY	70

*Tabla 7.4: Resultados esperados en la prueba*

### 7.3 SonarQube

Para este apartado de pruebas y testing se ha querido llegar un paso más allá. Se ha instalado un software de código abierto llamado SonarQube que tiene por objetivo evaluar y mejorar la calidad del código que se crea en el desarrollo software de un proyecto. Entre las funcionalidades que ofrece se puede encontrar [\[40\]](#):

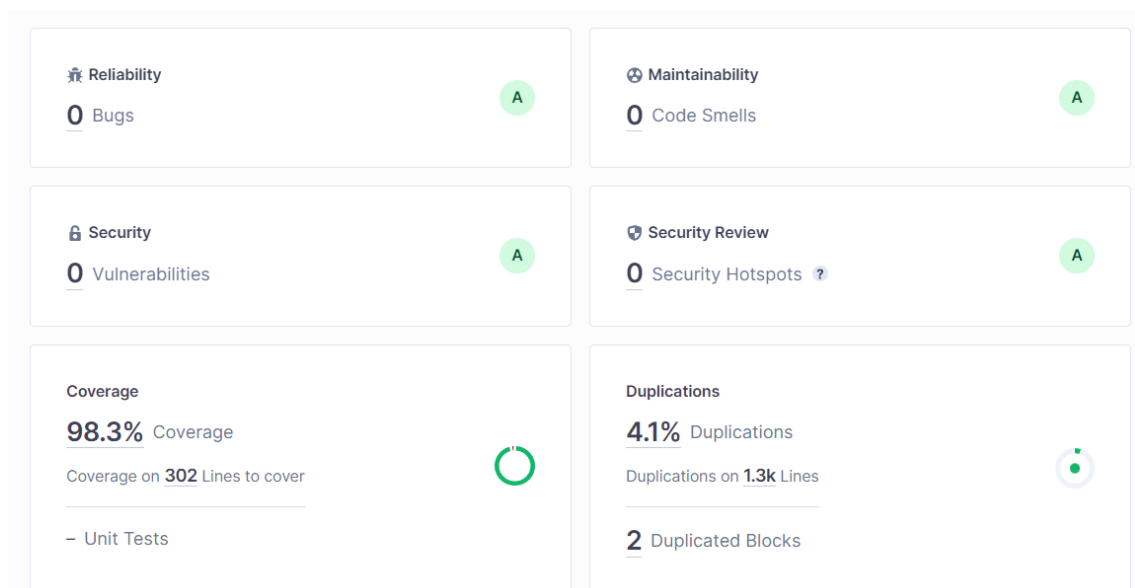
- 1. Calidad del código:** proporciona mediciones de ciertas variables que miden la calidad del código, así como la complejidad, duplicaciones, coverage...
- 2. Analizar código:** la plataforma analiza todo el código fuente para detectar posibles errores y vulnerabilidades que puedan estar presentes.
- 3. Compartir código:** permite que varias personas puedan ver los avances del código, así como realizar aportaciones o comentarios. Esto es ideal en un ambiente empresarial al emprender grandes proyectos, los supervisores pueden ver en todo momento los avances y aprobar los cambios del código, siendo similar a GitHub en este aspecto.
- 4. Múltiples lenguajes:** SonarQube soporta gran variedad de lenguajes de programación, entre ellos Python.

Al ejecutar tanto las pruebas unitarias como la prueba de integración se comprueba que todos los métodos del código funcionan correctamente. Cada línea ejecutada se marca como que ha sido probado durante el testing. Empleando la librería



coverage.py te permite saber qué porcentaje del código ha sido revisado y probado, esto se conoce como *coverage*. En el panel de control de SonarQube de la **Figura 7.1** se puede observar que 98,3% del código presente en projectIBKR.py e investingAuto.py. Esta cifra tiene un valor muy alto y es un indicativo de que el código se ha probado en profundidad. Cabe destacar que elementos generales como métodos principales no se han comprobado de forma unitaria puesto que se contemplan dentro de las pruebas de integración.

Adicionalmente, en la fases más tempranas del proyecto había gran cantidad de “Code Smells”, es decir, código que podría necesitar una revisión o refactorización para mejorar su mantenimiento y calidad. De casi 40 Code Smells se ha pasado a 0, pudiendo resolverlos todos. Asimismo, también había 2 bugs que se han podido solucionar para prevenir problemas de funcionamiento.



**Figura 7.1:** Resultados obtenidos en SonarQube

---

# 8. Conclusiones

---

Una vez desarrollada y puesta en funcionamiento la aplicación, se extraerán conclusiones de este Trabajo Final de Grado con el objetivo de evaluar los diversos aspectos del proceso y determinar si se han logrado los objetivos propuestos.

En primer lugar, mediante la creación de la aplicación de trazabilidad, se ha cumplido el primer objetivo de este trabajo a través de un desarrollo en el que se han aportado diferentes recursos útiles para inversores con un problema real. Durante esta etapa, se emplearon herramientas como Python, Pandas, SonarQube y Selenium, entre otras, lo que permitió presentar una solución con tecnologías y librerías actuales en el ámbito laboral. Estas decisiones tecnológicas facilitan el mantenimiento de la aplicación ante futuras actualizaciones o modificaciones.

Finalmente, la solución propuesta sienta las bases para la implementación de diversas estrategias de inversión, incluyendo operaciones en largo y corto. Así, el inversor tiene a su disposición un software adaptado a sus requerimientos de inversión, que además le permite la integración de nuevas estrategias si tiene habilidades de programación para hacer ajustes al proyecto, el cual está disponible en GitHub. Se ha diseñado la aplicación de tal manera que cause la mínima interrupción en las actividades diarias del usuario al ejecutarse, garantizando que no interfiera con su rutina de trabajo.

## 8.1 Aprendizaje personal

La carrera de Informática ha proporcionado gran cantidad de recursos para el desarrollo de este trabajo. Sin embargo, la dificultad presente en la solución aportada supera a los conocimientos aprendidos durante el grado, siendo necesario profundizar en el aprendizaje de gran variedad de herramientas. Se han explorado diferentes tecnologías, librerías así como conceptos teóricos que han posibilitado aprender multitud de técnicas de las que se desconocía o se tenía un conocimiento limitado. Entre estos aspectos se puede destacar: consolidar como se realizan peticiones API; aprender a programar en Python, donde se tenían conocimientos muy limitados; realizar un testing en profundidad, empleando la herramienta profesional SolarQube; estructurar el código de forma que facilite el mantenimiento de la aplicación; automatizar páginas web, mediante la librería de Selenium y también ser capaz de desarrollar una solución compleja de tamaño considerable.

## 8.2 Relación del trabajo desarrollado con los estudios cursados

La realización de este Trabajo Final de Grado es una aplicación directa de variedad de conocimientos adquiridos durante el grado de Ingeniería Informática. A lo largo de este se han empleado:

- Manejo de datos: poder manejar archivos de datos mediante la lectura o modificación de estos.
- Definir un proceso: poder definir un diagrama de flujo para establecer cuál es el proceso por seguir.
- Programación: a pesar de usar un lenguaje de programación como Python que no se emplea durante la carrera, mediante la base de conocimientos en otros lenguajes como Java ha facilitado aplicar técnicas similares.
- Conexiones con API: establecer conexiones entre diferentes puntos para obtener datos y posibilitar tener información actualizada en todo momento.
- Metodología de trabajo: establecer objetivos menores y estructurados para tener una carga de trabajo constante y planificada en diferentes fases de desarrollo.

En cuanto a las competencias transversales con mayor relevancia en el desarrollo de este proyecto son:

1. **Análisis y resolución de problemas:** afrontar problemas complejos requiriendo realizar previamente un análisis en profundidad para comprender la situación y las necesidades.
2. **Aplicación y pensamiento práctico:** elaborar un plan para resolver una situación mediante la propuesta de diferentes diagramas de flujos.

## 8.3 Errores y dificultades

La elaboración de este proyecto también ha tenido diversos momentos difíciles y se han presentado diferentes errores que ralentizado el proceso:

- **Conexión a plataformas:** uno de los problemas que más han dificultado el proceso ha sido la conexión a las plataformas de Trader Workstation e Investing.com. Se requiere de una conexión adecuada que en caso de emplear un WIFI inestable puede originar errores en la ejecución.
- **ISIN:** en las fases iniciales del desarrollo de la aplicación se desconocía la existencia de esta variable. Esto ocasionó que no se pudieran diferenciar adecuadamente los activos de la bolsa de valores. No fue hasta más adelante que se averiguó la existencia del ISIN solucionando así esta problemática.

- **Selenium:** hacer uso de esta potente librería también puede ser bastante complejo. En muchas ocasiones ha resultado difícil depurar algunos errores puesto que los errores en la consola solo eran fruto de un problema en cascada.
- **Investing.com bugs:** la página de Investing.com tiene diferentes errores que surgen de forma aleatoria y no es posible de prever cuando van a ocurrir. Alguno de ellos implicaba que un desplegable no se actuara y como solución se ha recargado la página, una solución simple y efectiva.

## 8.4 Trabajo futuro

Las funcionalidades que se pueden implementar en este trabajo pueden ir un paso más allá. En un futuro se planea mejorar aún más el proceso de configuración inicial de la aplicación para que sea un simple ejecutable que cree la tarea periódica de ejecución. Esto facilitaría aún más la puesta en marcha del programa para el usuario que tenga conocimientos muy limitados en la informática. Esto implicaría instalar automáticamente todas las dependencias, crear la tarea en el Programador del sistema operativo y que se ejecute directamente.

Además, se plantea incluir un ejecutable para que cada vez que el usuario lo desee pueda ejecutar la aplicación en caso de que en un día haya hecho gran cantidad de transacciones y no quiera esperar hasta la próxima hora planificada de ejecución. Otra solución posible que se está planteando es convertirlo en un proceso de continua ejecución y que detecte los cambios en las transacciones de IBKR para introducir inmediatamente los nuevos datos en la plataforma de Investing.com.



---

## 9. Bibliografía

---

- [1] Vallejo, C., Torres, Ó., & Pereira, Ó. T. (2012). Manual de la inversión en Bolsa.
- [2] Killian, A. (2020). ¿Qué actores influyen en el precio de las acciones? *IG*. <https://www.ig.com/es/estrategias-de-trading/-que-actores-influyen-en-el-precio-de-las-acciones--201201> [Consulta: 16 de marzo de 2023]
- [3] BBVA. (s.f.) Factores que influyen en el precio de las acciones. [https://www.bbva.es/estaticos/mult/Ayudas\\_factores\\_acciones.pdf\\_tcm924-528182.pdf](https://www.bbva.es/estaticos/mult/Ayudas_factores_acciones.pdf_tcm924-528182.pdf) [Consulta: 3 de abril de 2023]
- [4] Montoya, L. A., Arias, S. N. R., & Benjumea, J. C. C. (2006). Metodologías para la medición del riesgo financiero en inversiones. *Scientia et Technica*, 12(32), 275-278.
- [5] Unir, V. (2022). ¿Qué es un bróker? Funciones y formación necesaria. *UNIR*. <https://www.unir.net/empresa/revista/que-es-broker/> [Consulta: 8 de abril de 2023]
- [6] Garcia, J. (2023). Interactive Brokers opiniones: ¿Es un buen bróker para el 2023? *Mejorbanco*. <https://www.mejor-banco.com/opiniones-interactive-brokers/> [Consulta: 8 de abril de 2023]
- [7] Investing.com. (s. f.). Sobre nosotros. <https://es.investing.com/about-us/> [Consulta: 8 de abril de 2023]
- [8] Nyamathulla, S., Ratnababu, P., & Shaik, N. S. (2021). A Review on Selenium Web Driver with Python. *Annals of the Romanian Society for Cell Biology*, 16760-16768.
- [9] Koumou, G. B. (2020). Diversification and portfolio theory: a review. *Financial Markets and Portfolio Management*, 34(3), 267-312.
- [10] Jiménez, R. L. M. (2015). Metodologías ágiles de desarrollo de software aplicadas a la gestión de proyectos empresariales. *Revista tecnológica*, 8.
- [11] Muratori, R. (2022). 6 aspectos que debes considerar al elegir un broker. *Investing.com Español*. <https://es.investing.com/analysis/6-aspectos-que-debes-considerar-al-elegir-un-broker-200450613> [Consulta: 19 de abril de 2023]

- [12] Ludeña, J. A. (2023). InteractiveBrokers. *Economipedia*. <https://economipedia.com/definiciones/interactivebrokers.html> [Consulta: 28 de abril de 2023]
- [13] Interactive Brokers LLC. (s. f.). Soluciones de negociación API de IBKR. <https://www.interactivebrokers.com/es/trading/ib-api.php> [Consulta: 28 de abril de 2023]
- [14] Interactive Brokers LLC. (s. f.). Client Protection - Strength and Security. <https://www.interactivebrokers.com/en/general/security-investor-protection.php> [Consulta: 28 de abril de 2023]
- [15] Braz, P., & Braz, P. (2023). Best Brokers with API Access: API trading platforms reviewed. *Investing in the Web*. <https://investingintheweb.com/brokers/best-api-brokers/> [Consulta: 3 de mayo de 2023]
- [16] Galán, V. (2023). Los 10 mejores brokers para trading de 2023. Rankia. <https://www.rankia.com/blog/mejores-brokers/5678956-mejores-brokers-para-trading> [Consulta: 3 de mayo de 2023]
- [17] Investing.com. (s.f.) Financial Tools | Investment and trading tools. <https://www.investing.com/tools/> [Consulta: 8 de mayo de 2023]
- [18] Sharesight. (s. f.). About us | Sharesight. <https://www.sharesight.com/about-us/> [Consulta: 8 de mayo de 2023]
- [19] Collins, G. (2023). Google Finance vs Yahoo Finance. *Housing Forecasts & Stock Market Forecast*. <https://gordcollins.com/stock-market/google-finance-vs-yahoo-finance/> [Consulta: 8 de mayo de 2023]
- [20] Chazallet, S. (2016). Python 3: los fundamentos del lenguaje. Ediciones *Eni*.
- [21] Domingo Garzarán, I. (2010). Aplicación para Android: agenda comercial y detalles de clientes. <http://hdl.handle.net/10251/9124>. [Consulta: 12 de mayo de 2023]
- [22] Poveda González, A. (2011). Diseño de un portal web de gestión de carteras de acciones. <http://hdl.handle.net/10251/11985>. [Consulta: 12 de mayo de 2023]
- [23] Stoker, R. (2014). Cómo invertir en el Mercado de Valores para principiantes. Babelcube Inc.



- [24] Unir. (2021). ¿Qué es la bolsa de Valores? *UNIR revista*. <https://www.unir.net/empresa/revista/que-es-bolsa-de-valores/> [Consulta: 28 de mayo de 2023]
- [25] De La Cruz, I. (2023). Qué es la Ley de la oferta y la demanda - Investing.com. Investing.com ES. <https://es.investing.com/academy/economy/que-es-ley-oferta-demanda/> [Consulta: 28 de mayo de 2023]
- [26] CNMV (s. f.). Glosario Financiero. <https://www.cnmv.es/Portal/Inversor/Glosario.aspx?id=0&letra=I&idIng=1> [Consulta: 28 de mayo de 2023]
- [27] Zúñiga, E. D. (2021). ¿Qué son los instrumentos financieros y cuántos tipos hay? *IG*. <https://www.ig.com/es/estrategias-de-trading/-que-son-los-instrumentos-financieros-y-cuantos-tipos-hay--210525> [Consulta: 28 de mayo de 2023]
- [28] Pictet Asset Management. (2022). Instrumentos financieros: Qué son y cuáles son los más comunes actualmente. *Pictet Asset Management*. <https://am.pictet/es/blog/articulos/guia-de-finanzas/guia-instrumentos-financieros> [Consulta: 7 de junio de 2023]
- [29] Furió, E. (2023). ¿Qué son los «Swaps» y cómo funcionan? ¿Cuál es su finalidad? *BBVA NOTICIAS*. <https://www.bbva.com/es/swaps-que-son-y-como-funcionan/> [Consulta: 7 de junio de 2023]
- [30] CNMV. (s. f.). ISIN Code. <https://www.cnmv.es/portal/ANCV/CodigoISIN.aspx> [Consulta: 9 de junio de 2023]
- [31] CNMV. (s. f.). El mercado de valores y los productos de inversión – Manual para Universitarios. <https://www.cnmv.es/DocPortal/Publicaciones/Guias/ManualUniversitarios.pdf> [Consulta: 19 de junio de 2023]
- [32] McKinney, W. (2011). pandas: a foundational Python library for data analysis and statistics. *Python for high performance and scientific computing*, 14(9), 1-9.
- [33] Sweigart, A. (2020). Pyautogui documentation. *Read the Docs*, 25.

- [34] Raghavendra, S. (2021). Python testing with selenium. En Apress eBooks. <https://doi.org/10.1007/978-1-4842-6249-8>
- [35] Zhao, B. (2017). Web scraping. Encyclopedia of big data, 1.
- [36] Google. (s.f.). Downloads ChromeDriver WebDriver for Chrome. <https://sites.google.com/a/chromium.org/chromedriver/downloads> [Consulta: 22 de julio de 2023]
- [37] Python. (s.f.) Download Python. <https://www.python.org/downloads/> [Consulta: 22 de julio de 2023]
- [38] Interactive Brokers LLC. (s. f.). TWS SOFTWARE <https://www.interactivebrokers.com/en/trading/tws.php> [Consulta: 22 de julio de 2023]
- [39] Paz, J. A. M. (2016). Análisis del proceso de pruebas de calidad de software. Ingeniería solidaria, 12(20), 163-176.
- [40] SonarQube. (s. f.). Code quality tool & secure analysis with SonarQube. <https://www.sonarsource.com/products/sonarqube/> [Consulta: 12 de agosto de 2023]

---

# ANEXO A: Detalles de la Trader Workstation

---

En este anexo se van a tratar diferentes aspectos que no se han detallado durante la memoria a causa de que es información para los lectores que quieran profundizar con la Trader Workstation de IBKR.

En primer lugar, se tratará la instalación de la plataforma desde su página web. A continuación, se indagará un poco en las funciones que ofrece la Trader Workstation en sus paneles de control de operaciones. Por último, una breve explicación como se han obtenido todos los datos de la cuenta del inversor.

## A.1 Instalación y configuración Trader Workstation

Para poner en marcha la plataforma de Trader Workstation de IBKR se tienen que seguir los siguientes pasos:

1. Descargar el software desde la [página oficial](#). Se recomienda usar la última versión disponible para el sistema operativo.
2. Instalar el programa en el directorio predeterminado.
3. Con una cuenta previamente creada ingresar al programa.
4. Cambiar la configuración desde el panel de control.
  - a. Acceder a “Archivo” → “Configuración Global” → “API” → “Configuración”.
  - b. Marcar la casilla de “Activar clientes ActiveX y Socket”.
  - c. Asegurarse que el “Socket Port” predeterminado es 7497.
  - d. Desmarcar casilla “API solo lectura”.
  - e. Aplicar los cambios y Aceptar.
5. Comenzar a operar para tener datos disponibles.

## A.2 Operar con Trader Workstation

Operar con el software de Interactive Brokers es realmente muy simple y a la vez ofrece gran cantidad de información. En esta sección servirá como una simple toma de contacto para entender cómo se opera con activos empleando esta plataforma.

En primer lugar, para realizar operaciones de compra o venta de activos simplemente hay que dirigirse al panel de la esquina superior izquierda llamado “Entrada de órdenes”. En la siguiente **Figura A.1** se presenta la compra de 100 acciones de Amazon. Para comprarlas solamente habrá que presionar el botón de “Enviar”.



Figura A.1: Panel de órdenes TWS

Se pueden alterar otros parámetros adicionales en la compra y venta de activos como al precio que se quiere realizar pudiendo escoger diferentes estrategias como límites, precio medio, precio actual, mejor PEG, flexible... Asimismo, el software también ofrece algunas herramientas de gestión de portafolio como en el de la siguientes figura a través de gráficos interactivos y diferentes parámetros para ver la evolución del activo.



Figura A.2: Gráfico y parámetros de acciones de Amazon en la TWS

Por último, se ofrece una vista de la cartera con diferentes indicadores para ver el estado de esta de una forma directa y breve, aunque no se compara con la forma que Investing.com ofrece la información. En esta como se ha ido viendo en las anteriores figuras se trata de una interfaz muy cargada con gran cantidad de datos que pueden abrumar al usuario y una estructura que necesita una actualización para no saturar la portada. No se va a indagar más en esta sección puesto que no es el objetivo de este

Trabajo Final de Grado, solo se ha querido mostrar de forma breve como es el día a día de un inversor.

### A.3 Detalles de la obtención de los datos de la cuenta

Para poder conectarse a la API de Interactive Brokers se deben en primer lugar Importar la librería “ibapi”, habiendo previamente hecho un “*pip install ibapi*”. Gracias a esta librería se podrán definir una nueva clase hereda con la siguiente cabecera:

```
class TransactionsApp(EWrapper, EClient):  
  
    def __init__(self):  
  
        EClient.__init__(self, self)
```

Esto permitirá acceder a los diferentes atributos y métodos de esa clase padre, reutilizando el código existente, lo que ahorra tiempo y esfuerzo en la programación. Tanto EWrapper como EClient incorporan una interfaz para simplificar las operaciones a través de la API, el primero para recibir y manejar eventos enviados por el servidor TWS, mientras que el segundo para enviar solicitudes y órdenes al servidor. Estos componentes juntos permiten al programa de trazabilidad creado interactuar de forma adecuada con la plataforma de Interactive Brokers a través de la API.

Otro detalle interesante en la implementación de este archivo es el método *get\_isin(id)*. El problema que existía es que a través de la clase anterior al solicitar los datos de los activos del portfolio y transacciones la variable del ISIN siempre se encuentra vacía. Por este motivo se importa una nueva librería *ib\_insync* que permite recuperar este dato de una forma sencilla y sin generar ningún error. La desventaja de tener que aplicar esta solución es que se debe formalizar una nueva conexión con la API al tener que crear una nueva solicitud separada solamente para tener el ISIN.

---

# ANEXO B: ODS

---

## OBJETIVOS DE DESARROLLO SOSTENIBLE

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

<b>Objetivos de Desarrollo Sostenibles</b>	<b>Alto</b>	<b>Medio</b>	<b>Bajo</b>	<b>No Procede</b>
ODS 1. <b>Fin de la pobreza.</b>			<b>X</b>	
ODS 2. <b>Hambre cero.</b>				<b>X</b>
ODS 3. <b>Salud y bienestar.</b>				<b>X</b>
ODS 4. <b>Educación de calidad.</b>				<b>X</b>
ODS 5. <b>Igualdad de género.</b>				<b>X</b>
ODS 6. <b>Agua limpia y saneamiento.</b>				<b>X</b>
ODS 7. <b>Energía asequible y no contaminante.</b>				<b>X</b>
ODS 8. <b>Trabajo decente y crecimiento económico.</b>	<b>X</b>			
ODS 9. <b>Industria, innovación e infraestructuras.</b>				<b>X</b>
ODS 10. <b>Reducción de las desigualdades.</b>		<b>X</b>		
ODS 11. <b>Ciudades y comunidades sostenibles.</b>				<b>X</b>
ODS 12. <b>Producción y consumo responsables.</b>	<b>X</b>			
ODS 13. <b>Acción por el clima.</b>				<b>X</b>
ODS 14. <b>Vida submarina.</b>				<b>X</b>
ODS 15. <b>Vida de ecosistemas terrestres.</b>				<b>X</b>
ODS 16. <b>Paz, justicia e instituciones sólidas.</b>				<b>X</b>
ODS 17. <b>Alianzas para lograr objetivos.</b>		<b>X</b>		



### **Reflexión sobre la relación del TFG/TFM con los ODS y con el/los ODS más relacionados.**

La relación que existe entre los Objetivos de Desarrollo Sostenible con este trabajo final de grado se puede notar en diversos puntos. Dado que todo el trabajo gira entorno a las finanzas y la informática se van a ver los objetivos que se ha podido contribuir de forma directa.

En primer lugar, el Trabajo Final de Grado contribuye al ODS 8 Trabajo decente y crecimiento económico. La aportación del programa desarrollado gira entorno a una toma de decisiones más informada y tener un mejor control sobre la información a disposición, aumentando así los beneficios que se puedan alcanzar en comparación a no hacer un seguimiento adecuado de los activos de una cartera. Se trata de facilitar estar al tanto de toda la información relativa a los activos de una forma más sencilla y así asegurar que el inversor toma las decisiones adecuadas en el futuro.

En segundo lugar, el ODS 12 Producción y consumo responsables también está implicado en este proyecto. Al emplear una solución que acorta drásticamente el proceso de seguimiento y control de las operaciones del mercado de valores al no tener que hacer una gran cantidad de tareas manualmente, se disminuye el consumo energético.

Asimismo, al crear un repositorio público accesible a cualquier persona interesada a contribuir en este trabajo se impulsa el ODS 17 Alianzas para lograr objetivos. Con más personas implicadas en este proyecto se podrán llegar a soluciones más eficientes y elaboradas que mejoren los procesos realizados en el seguimiento de las operaciones. Poder compartir con otros programadores e inversores este trabajo solo puede generar ventajas para toda la comunidad y así facilitar la vida a todas las personas que empleen este programa

Por último, de forma breve existe una notable implicación con el ODS 1 Fin de la pobreza y el ODS 10 Reducción de las desigualdades. Esto se debe que, al tener una mejor gestión de los activos mediante el seguimiento y control de estos, se llega a una mejor estabilidad económica al mismo tiempo que posibilita el enriquecimiento. Tomar decisiones sin fundamento y sin información puede resultar en grandes pérdidas que acaben con la situación económica de los inversores y que a su vez puedan llevar a crisis un determinado sector. Por ello, una toma de decisiones informada posibilita llegar a un mayor rendimiento de las carteras del inversor y poder salir de una posible situación de mala estabilidad económica. Operar en la bolsa es una tarea muy compleja que hecha de forma rápida sin documentarse lo suficiente puede resultar en graves errores y con este programa se tiene como objetivo simplificar el control y supervisión de los activos para que a cualquier inversor le sea más fácil.