

Desarrollo de una aplicación web para realizar backtesting sobre acciones o ETFs de la bolsa de valores de New York

Yoan Esteban Lopez Garcia



Facultad de Ingenierías
Programa de Ingeniería de Sistemas y
Computación
2022

Yoan Esteban Lopez Garcia

Desarrollo de una aplicación web para realizar backtesting sobre acciones o ETFs de la bolsa de valores de New York.

Trabajo de grado presentado como requisito parcial para optar al título de Ingeniero de Sistemas y Computación.

Universidad Tecnológica de Pereira

Facultad de Ingenierías

Programa de Ingeniería de Sistemas y Computación

Marzo de 2022

Director del trabajo

Ramiro Andrés Barrios Valencia

Tabla de contenido

Título	7
Resumen	8
Introducción	9
Definición del problema	9
Antecedentes	9
Causas	10
Descripción del problema	11
Consecuencias	11
Justificación	11
Objetivos	13
Objetivo general	13
Objetivos específicos	13
Marco Referencial	13
Estado del Arte	13
Marco juridico	16
Marco Conceptual	16
Definición de conceptos	16
Bolsa de valores	16
Acciones	17
ETF	18
Trading	18
Análisis técnico	18
Indicador técnico	19
Cruce de medias móviles	19
Bandas de bollinger bands	20
Donchian channels	21
Nubes de Ichimoku	22

Estrategia	23
Backtesting	24
Forward testing	25
Win Rate	25
Profit/loss	26
Metodología	27
Marco metodológico	27
Fase de investigación:	27
Fase de desarrollo:	27
Fase de finalización:	27
Actividades	28
Desarrollo	29
Características	29
Usuarios del sistema	29
Historias de usuario	29
Diagramas de caso de uso	37
Diagrama de base de datos	45
Diagrama de clases	46
Diagrama de despliegue	47
Diagrama de arquitectura	47
Fase de implementación	48
Lenguaje de programación	48
Motor de base de datos	48
Framework	48
IDE de desarrollo	49
Adquisición de los datos históricos	49
Trabajos futuros	51
Conclusiones	52
Anexos	54
Anexo 1. Cronograma del proyecto	54

Lista de tablas

Tabla 1 Usuarios del sistema	29
Tabla 2 Historia de Usuario 001	30
Tabla 3 Historia de Usuario 002	30
Tabla 4 Historia de Usuario 003	31
Tabla 5 Historia de Usuario 004	32
Tabla 6 Historia de Usuario 005	33
Tabla 7 Historia de Usuario 006	33
Tabla 8 Historia de Usuario 007	34
Tabla 9 Historia de Usuario 008	35
Tabla 10 Historia de Usuario 009	36
Tabla 11 Historia de Usuario 010	37
Tabla 12 Cronograma del proyecto	54

Lista de ilustraciones

Ilustración 1	Gestion de usuarios	37
Ilustración 2	Módulo Backtesting	38
Ilustración 3	Servicio de consumo	38
Ilustración 4	Registro de usuarios	39
Ilustración 5	Iniciar sesión	39
Ilustración 6	Cambiar nombre de usuario	40
Ilustración 7	Eliminar cuenta	40
Ilustración 8	Recuperar contraseña	41
Ilustración 9	Iniciar backtest	42
Ilustración 10	Ver historial	43
Ilustración 11	Borrar backtest	43
Ilustración 12	Servicio de consumo	44
Ilustración 13	Diagrama de base de datos	45
Ilustración 14	Diagrama de clases	46
Ilustración 15	Diagrama de despliegue	47
Ilustración 16	Diagrama de arquitectura	47
Ilustración 17	Adquisición de los datos históricos	51

1. Título

Desarrollo de una aplicación web para realizar backtesting sobre acciones o ETFs de la bolsa de valores de New York.

2. Resumen

El siguiente proyecto trata sobre el desarrollo de una plataforma web para la aplicación de backtesting sobre acciones o ETFs de la bolsa de valores de New York, parametrizado por dos estrategias y cuatro indicadores para personas principiantes que empiezan su trayectoria en el mundo del trading.

Se definen los problemas que poseen los traders novatos para la realización de backtesting en diferentes plataformas y aplicaciones que ofrece el mercado y concluir en una solución en línea para un acercamiento más sencillo y amigable en el empleo de este procedimiento.

Este trabajo de grado abarca el desarrollo de la plataforma desde cómo se hizo además de las herramientas utilizadas y el testeo de sus funcionalidades. Incluyendo los aspectos teóricos, técnicos; adjuntando el diseño de modelado del sistema por medio de ingeniería de software y los respectivos diagramas arquitectónicos.

3. Introducción

En el mundo del trading se encuentra una gran diversidad de estrategias comerciales, cada una cuenta con gratos resultados para vender y comprar acciones o ETFs particulares de la bolsa de valores, la elección de escoger una estrategia ideal para estos mercados es un verdadero reto; el backtesting es uno de los resultados de este reto, este se encarga de evaluar la estrategia escogida y observar los niveles de asertividad por medio de los datos históricos del ticket sobre el cual se va a operar, además da al usuario la oportunidad de optimizar la estrategia.

El proyecto se enfoca en diseñar y construir un prototipo funcional de una aplicación web en la cual los usuarios pueden verificar diferentes estrategias de trading sobre al menos cuatro señales distintas. Comprendiendo su comportamiento en los diversos tickers que tiene ofrece la aplicación y afianzando sus conocimientos.

En la actualidad existen varias herramientas del trading que ofrecen a los usuarios la opción de realizar procesos de backtest, sin embargo estos necesitan tener conocimientos adicionales en el mundo de la programación para poder desarrollarlos, generando una barrera para la masificación de su uso.

4. Definición del problema

4.1. Antecedentes

Existen aplicaciones web y móviles enfocadas en el mundo de los mercados de valores que ofrecen diversas herramientas para el análisis de acciones de empresas o ETFs, estas permiten desarrollar backtesting por medio de plugins el cual el usuario deberá programar, esto hace que sean muy difíciles de utilizar por parte de estos usuarios ya

que deben poseer habilidades de programación para poder evaluar sus estrategias al mercado escogido.

Algunas herramientas son las siguientes:

<https://finviz.com> es una plataforma muy conocida por los traders, posee un conjunto de herramientas muy completas y con una buena atención de usuario. Con el backtest puede tantear aproximadamente 100 indicadores técnicos en datos de stock siempre y cuando sea un programador experimentado y además sea un cliente Finviz Élite, para un usuario principiante le costará bastante poder evaluar su estrategia.

<https://www.amibroker.com> es una plataforma de análisis técnico integral desarrollada pensando en los usuarios avanzados, su backtesting tiene mucha flexibilidad, desde el tamaño de cartera hasta la sub porciones de tickers a las cual le gustaría aplicar el backtesting. Estas configuraciones para los usuarios nuevos pueden ser abrumadoras y requieren superar una curva de aprendizaje alta para aprovechar al máximo la herramienta.

<https://ninjatrader.com> suministra un motor de backtesting de alto rendimiento que permite a sus clientes probar y verificar sus estrategias de negociación de futuros siempre y cuando esté familiarizado con la plataforma y sus diferente gama de materiales educativos que se tornan dificultoso dependiendo del nivel de experiencia del usuario.

4.2. Causas

Los servicios de backtesting ofrecidos por las grandes plataformas de mercados de valores, tienen un alto nivel de complejidad para usuarios principiantes.

La necesidad de programar el backtesting para evaluar las estrategias básicas del trading.

4.3. Descripción del problema

Los traders principiantes no tienen a disposición una aplicación web que les permita experimentar con un backtesting básico, y que no requiera habilidades en el ámbito de la programación.

4.4. Consecuencias

Las personas que quieren incursionar en el vasto mundo del trading se verán abrumadas por las herramientas tan avanzadas y poco didácticas para el testeado de estrategias.

Realizar operaciones sin tener idea de la tasa de éxito de la estrategia tomada, conlleva a una pobre comprensión del mercado y una baja percepción de los niveles de entrada y salida.

5. Justificación

El mercado de valores es uno de los temas más importantes en la actualidad, juega un papel importante en la economía global, hace posible tanto a empresas como gobiernos financiar proyectos rentables y de desarrollo, y así producir empleos. Está centraliza la oferta y la demanda de acciones, bonos, instrumentos de corto plazo etc, fomentando la aportación de interesados en invertir mediante la oferta pública.

Muchos nuevos participantes en el mercado de valores están cegados por los altos rendimientos y las oportunidades de

obtener ganancias, y están demasiado entusiasmados para enfrentar los altos riesgos que conlleva. A primera vista, comprar y vender instrumentos financieros en el mercado de valores es una excelente manera de ganar dinero. Sin embargo, como en cualquier profesión, la experiencia es necesaria para que estas acciones sean rentables.

El desarrollo de esta aplicación ayudará a dar una mayor comprensión del mercado bursátil a aquellas personas que inician en este camino, ofreciendo la posibilidad de aprender y obtener experiencia del comportamiento de las acciones o ETFs de diferentes empresas u organizaciones por medio del backtesting.

Al contrario de las diversas plataformas existentes que requieren adicionalmente fundamentos de programación para su uso y masificación, la aplicación permitirá que los usuarios practiquen cómo se desempeñan diferentes sistemas compuestos con diversas reglas de entrada y salida para un ticker en particular, a través de sus datos históricos sin tener previos conocimientos computacionales.

Además ofrece una vista amigable con la cual interactuar y analizar los movimientos pasados, adaptado la mente a diferentes patrones que forman los gráficos, reduciendo miedos de incursionar en el mercado, además de ser un instrumento con el propósito de dar a identificar, cuál es la mejor estadística de ganancia y pérdida, para generar una rentabilidad a futuro.

6. Objetivos

6.1. Objetivo general

Desarrollar una aplicación web para realizar backtesting sobre acciones o ETFs de la bolsa de valores de New York.

6.2. Objetivos específicos

- Definir al menos 2 estrategias y cuatro indicadores para parametrizar el backtesting.
- Generar un mecanismo que consuma algún servicio para la obtención de los datos históricos de las principales acciones y ETFs de la bolsa de New York.
- Implementar bajo modelo vista controlador la parametrización y ejecución del backtest.
- Diseñar una vista amigable para la generación de reportes con resultados estadísticos de la estrategia evaluada.

7. Marco Referencial

7.1. Estado del Arte

Con relación a los trabajos aplicativos relacionados con el objetivo de este proyecto se obtienen los siguientes:

- 7.1.1.** Gallo, Crescenzo & Fratello, Angelo. (2014). The Forex Market in Practice: A Computing Approach for Automated Trading Strategies. International Journal of Economics and Management Sciences.

En este artículo examinaron los problemas asociados con el área de la informática en aplicaciones de finanzas, particularmente en el mercado financiero

de Forex y como los instrumentos computacionales afrontan el problema de la previsión, resaltando su importante papel no solo en hacer predicciones sino también en resolución de problemas empresariales, justificando su mérito sobre los métodos estadísticos y otros métodos tradicionales, además, de exponer sus limitaciones.

- 7.1.2.** Kari Kostianen, (2016). Development of Trading Algorithm Backtest Environment, Helsinki Metropolia University of Applied Sciences.

El objetivo de este proyecto era desarrollar una aplicación de escritorio en Windows de backtesting para simular el comercio algorítmico, promete hacer cálculos y tomar iniciativa sobre sí realizar una compra, venta, transacción o ninguno de los anteriores, simulando detalles de la transacción y su correspondiente almacenado. No obstante solo muestra gráficos relacionados con intercambios, perdiendo todos los posibles cierres, a pesar de que ofrece una plataforma de operación simple, el usuario necesitará agregar sus algoritmos para el proceso.

- 7.1.3.** Verma, Ritik & Gohrani, Kunal & Mitra, Anirban & Das, Anirban & Chakrabarti, Tulika & Chakrabarti, Prasun. (2021). A computing-based system developed on approaches and strategies to analyze share values for trading intentions in the Stock Market Domain.

El trabajo se encaminó a discutir e implementar el enfoque algorítmico para encontrar tendencias en el valor de una acción en la bolsa de valores, su objetivo es facilitar la vida de un comerciante al trabajar con tickers en el mercado, básicamente el usuario puede configurar una alerta en el sistema propuesto y está se activará tan pronto como una

estrategia entre en juego sobre las acciones previamente seleccionadas, dando al usuario las oportunidades comerciales de la notificación de la llamada de alerta. El equipo ha procurado crear una aplicación de sencillo uso para principiantes, con estrategias codificadas y un conocimiento básico de la bolsa de valores de un usuario pueda con facilidad utilizar la aplicación con fines comerciales y así obtener ganancias.

7.1.4. Ni, Jiarui & Zhang, Chengqi. (2005). An Efficient Implementation of the Backtesting of Trading Strategies.

Este artículo se demostró el paso a paso de la implementación eficiente de backtesting de una estrategia comercial compleja, por medio de un algoritmo genético paralelo (PGA) que aceleran este proceso dando como resultado un rango de consumo de tiempo comprensible y apropiado para que la estrategia comercial ensayada se logre poner en marcha correctamente a tiempo.

7.1.5. Spörer, Jan, (2020). Backtesting of Algorithmic Cryptocurrency Trading Strategies.

Esta tesis expone un instrumento para hacer backtesting de estrategias de trading algorítmicas para criptomonedas. Dicha herramienta es llamada quantbacktest suministra una estructura conveniente de ejecutar backtesting y además proporciona cálculos de los rendimientos de la cartera y métricas notables para el riesgo de manera realista, lamentablemente toda su configuración está compuesta en lenguaje python, destinada a individuos con conocimientos en dicho lenguaje.

8. Marco juridico

La plataforma contará con registros e información personal de cada cliente, se protegerá la información proporcionada evitando así su divulgación.

- Ley 1273 de 5 de enero de 2009.
- Ley de Protección de datos de 1988.
- Ley 1273 de 5 de enero de 2009.

9. Marco Conceptual

9.1. Definición de conceptos

9.1.1. Bolsa de valores

La bolsa de valores es una organización, privada o pública, que permite a sus componentes introducir ordenes y realizar transacciones de compra o venta de valores, siendo estos los derechos que tienen sus propietarios.

Los valores de una sociedad o de una compasiva anónima son las acciones. Los precios de la negociación de los valores se producen en un entorno real y seguro para el inversor, ya que las transacciones de compra y venta entre las partes se realizan obedeciendo a la legalidad vigente. Las bolsas ayudan al desarrollo económico y financiero en el mundo.

En la bolsa participan:

- Las empresas que requieren financiarse, pero no recurren a los bancos, por lo tanto prefieren al inversor privado.
- Inversores agotados de no hallar rentabilidad a su dinero en una cuenta bancaria.
- Intermediarios que aspiran ganar dinero actuando entre compradores y vendedores.

9.1.2. Acciones

Las acciones son unidades de propiedad de capital en una corporación. Para algunas empresas, las acciones existen como un activo financiero que proporciona una distribución equitativa de los beneficios residuales, si se declaran, en forma de dividendos . Al establecer una corporación, los propietarios pueden optar por emitir acciones ordinarias o acciones preferentes a los inversores. Las empresas emiten acciones a los inversores a cambio de capital, que se utiliza para hacer crecer y operar la empresa.

A diferencia del capital de deuda, obtenido a través de un préstamo o emisión de bonos, el capital social no tiene un mandato legal para ser reembolsado a los inversionistas y las acciones, aunque tienen la posibilidad pagar dividendos como una distribución de utilidades, no pagan intereses. Casi todas las empresas, desde pequeñas sociedades o hasta corporaciones multinacionales, emiten acciones de algún tipo.

9.1.3. ETF

Un exchange traded fund (ETF) es un tipo de valor que rastrea un índice, sector, producto básico u otro activo, pero que se puede comprar o vender en una bolsa de valores de la misma forma que lo hace una acción normal. Un ETF se puede organizar para rastrear cualquier cosa, desde el precio de un producto individual hasta una colección grande y diversa de valores. Los ETF incluso se pueden formular para realizar un rastreo de estrategias de inversión específicas.

9.1.4. Trading

En general la mayoría de las personas ha hecho trading alguna vez en su vida, pues este término se describe simplemente como el intercambio de dinero en una tienda por un producto. Cuando se habla de operar en los mercados bursátiles, es el mismo fundamento, cuando un individuo negocia con acciones, lo que realmente está haciendo es comprar acciones de una empresa. Si la utilidad de esas acciones incrementa, entonces el individuo gana dinero vendiéndolas nuevamente a una cotización más alta. Esto es el trading, comprar algo por un precio y luego venderlo por otro, tal vez más alto obteniendo ganancias y viceversa.

9.1.5. Análisis técnico

Es el estudio de la fluctuación de precios y patrones de un valor. Al analizar el historial de precios de un valor, primordialmente a través de indicadores y gráficos, los comerciantes pueden prever la dirección futura del precio.

9.1.6. Indicador técnico

Los indicadores técnicos son señales heurísticas, se fundamentan en patrones producidos por el precio, volumen, entre otros, que aprovechan los comerciantes que persiguen el análisis técnico. Estos indicadores técnicos pueden asistir a un comerciante a embarcarse de forma productiva a través de un periodo de negociación, o un periodo en el que una acción o ETF fluctúa entre un nivel de resistencia y un nivel de soporte.

En el presente proyecto se trabajará con 4 señales o indicadores técnicos básicos:

Cruce de medias móviles

Es el indicador más básico, provee un seguimiento a la tendencia del precio y expresa la relación entre dos o más promedios móviles del precio de un valor. En el proyecto se trabajó con dos tipos de promedios móviles los cuales son:

- Media móvil simple (SMA): es el tipo más simple y conocido, se calcula tomando la media aritmética de un conjunto dado de valores durante un periodo específico.

$$SMA = \frac{A_1 + A_2 + \dots + A_n}{n}$$

donde:

A = Promedio en el periodo n

n = Número de periodos de tiempo

- Media móvil exponencial: este tipo de media móvil brinda mayor peso a los precios recientes en un intento de evaluar mejor la información. Su cálculo es de la siguiente forma:

$$EMA_t = a * \text{precio de cierre} + (1 - a) * EMA_{t-1}$$

donde:

$$a = \frac{2}{n+1}$$

n = Número de periodos de tiempo

Cuando el cruce del promedio móvil de menor periodo esté por encima de los demás promedios se dará una señal de entrada para una operación de compra, y cuando el promedio móvil decae su valor por debajo de los demás promedios significaba una señal de salida de la operación.

Bandas de bollinger bands

Es un indicador un poco más complejo por poseer señales como medias móviles simples y conjuntos de reglas basadas en desviaciones estándar dando como resultado la generación de señales y tendencia del mercado. Hay tres líneas que componen las Bandas de Bollinger, una media móvil simple (banda media) y una banda superior e inferior. Su cálculo se proce de la siguiente manera:

$$BOLU = MA(TP, n) + m * \sigma[TP, n]$$

$$BOLD = MA(TP, n) - m * \sigma[TP, n]$$

donde:

BOLU = Banda de Bollinger superior

BOLD = Banda de Bollinger inferior

MA = Promedio Móvil

$$TP = \frac{High + Low + Close}{3}$$

n = Número de periodos de tiempo

m = Desviación estándar

$\sigma[TP, n]$ = Desviación estándar en los últimos *n* periodos de TP

Esta señal es muy popular entre comerciantes dado que cuanto más se acercan los precios a la banda superior se cree que el mercado está sobrecomprado y cuando está en la banda inferior se cree que el mercado está sobrevendido.

Donchian channels

Los canales de Donchian son tres líneas generadas por cálculos de medias móviles que comprenden un indicador formado por bandas superior e inferior alrededor de una banda de rango medio o mediano. La banda superior marca el precio más alto de un valor durante *N* períodos, mientras que la banda

inferior marca el precio más bajo de un valor durante N períodos. El área entre las bandas superior e inferior representa el Canal Donchian. La fórmula para los canales de Donchian es:

$UC = \text{Máximo más alto de los últimos } n \text{ períodos.}$

$LC = \text{Mínimo más bajo de los últimos } n \text{ períodos.}$

$$\text{Canal Medio} = \frac{UC + LC}{2}$$

donde:

$UC = \text{Canal superior}$

$LC = \text{Canal inferior}$

$n = \text{Número de períodos de tiempo.}$

Nubes de Ichimoku

Es un conjunto de indicadores técnicos que presentan los niveles de soporte y resistencia, así como la fuerza y dirección de la tendencia, por medio de múltiples promedios y trazos en un gráfico en forma de nube con el propósito de pronosticar donde el precio puede detectar soporte o resistencia en el futuro.

La nube de Ichimoku se constituye de cinco líneas o cálculos:

$$\text{Línea de conversión (tenkan sen)} = \frac{9PH + 9PL}{2}$$

$$\text{Línea base (kijun sen)} = \frac{26PH + 26PL}{2}$$

$$\text{Span A} = \frac{CL + \text{Línea Base}}{2}$$

$$\text{Span B} = \frac{52PH + 52PL}{2}$$

Línea de retraso = close trazado 26 periodos en el pasado.

donde:

PH = Periodo Alto

PL = Periodo Bajo

CL = Línea de conversión

9.1.7. Estrategia

Una estrategia de negociación es una metodología sistemática que se utiliza para comprar y vender en los mercados de valores. Una estrategia comercial se basa en reglas y criterios predefinidos que se utilizan al tomar decisiones comerciales.

Una estrategia de negociación puede ser simple o compleja e implicar consideraciones tales como estilo de inversión (por ejemplo, valor frente a crecimiento), capitalización de mercado, indicadores técnicos, análisis fundamental, sector industrial,

nivel de diversificación de la cartera, horizonte temporal o período de tenencia, tolerancia al riesgo, apalancamiento, consideraciones fiscales, etc. La clave es que una estrategia comercial se establezca utilizando datos y análisis objetivos y se cumpla con diligencia. Al mismo tiempo, una estrategia comercial debe reevaluarse y ajustarse periódicamente a medida que cambian las condiciones del mercado o los objetivos individuales.

9.1.8. Backtesting

El backtesting es el método general para ver qué tan bien se comporta una estrategia. El backtesting evalúa la viabilidad de una estrategia comercial al descubrir cómo se desarrollaría utilizando datos históricos. Si el backtesting funciona, los comerciantes y analistas pueden tener la confianza de utilizarlo en el futuro. El backtesting permite a un operador simular una estrategia comercial utilizando datos históricos para generar resultados y analizar el riesgo y la rentabilidad antes de arriesgar capital real. Un backtest bien realizado que arroja resultados positivos asegura a los operadores que la estrategia es fundamentalmente sólida y es probable que genere ganancias cuando se implemente en la realidad. Por otra parte, una prueba retrospectiva bien realizada que genere resultados subóptimos impulsa a los operadores a modificar o rechazar la estrategia.

Algunos inversores y comerciantes pueden explorar la destreza de un programador calificado para desarrollar el concepto de la estrategia en una forma comprobable, implicando que el programador codifique este concepto en el lenguaje patentado por la plataforma comercial donde opere.

El programador integra variables establecidas por el usuario que permiten modificar el sistema como por ejemplo el sistema de cruce de medias móviles, alterando las constantes definidas en el sistema, luego se realizan pruebas de backtesting para precisar el valor de promedios móviles más apto para los datos históricos.

9.1.9. Forward testing

Después de haber optimizado la estrategia en el backtesting, llega la hora de verificar su comportamiento con operaciones fuera de muestra o como es conocido mayormente, forward testing. Se trata de una simulación de una situación comercial que representa con mayor claridad cómo se puede utilizar un sistema comercial optimizado en sesiones reales. En pocas palabras, el forward testing es una de esas fases de backtesting. No obstante se tiene que prever que aunque el backtesting y el forward testing aportan una idea de su estrategia, no garantiza con exactitud cómo se comportaría un sistema de trading durante el trading en vivo. Los operadores suponen que más pruebas no mejoran la estrategia, pero puede determinar un nivel de fiabilidad que sufraga el riesgo efectivo real.

9.1.10. Win Rate

Es uno de los indicadores de observación de rendimiento prácticos, es sencillamente el número de operaciones ganadoras dividido el número total de operaciones.

$$\text{win rate} = \frac{\text{Número de operaciones ganadoras}}{\text{Número total de operaciones}}$$

9.1.11. Profit/loss

Es la relación de ganancia promedio de las operaciones ganadoras sobre la pérdida promedio de las operaciones perdedoras en el transcurso de un periodo de tiempo específico. Esta relación actúa como un marco de referencia para el comerciante cuyo motivo principal es maximizar las ganancias.

$$\text{profit/loss} = \frac{\text{Ganancia total}}{NWT} \div \frac{\text{pérdida total}}{NLT}$$

donde:

NWT = Número de operaciones ganadoras

NLT = Número de operaciones perdedoras

10. Metodología

10.1. Marco metodológico

El desarrollo de la aplicación se llevará a cabo en tres ciclos: investigación, desarrollo y finalización. Estos tres ciclos se desglosan en varias etapas para garantizar que todo se lleve a cabo de forma correcta en el tiempo estimado de tres (sumó cuatro) meses.

10.1.1. Fase de investigación:

Para el desarrollo del proyecto es necesario dar un tiempo a la investigación para enfocar la manera más adecuada de utilizar los recursos. Se da un plazo de acuerdo a las necesidades del proyecto para realizar un análisis de las tecnologías adecuadas para las diferentes partes del software: frontend y backend.

10.1.2. Fase de desarrollo:

La parte que llevará más tiempo del proyecto es el desarrollo de un prototipo funcional, que debe cumplir los objetivos especificados.

10.1.3. Fase de finalización:

La fase final será de aproximadamente de dos semanas, dando tiempo para la revisión completa del proyecto y la resolución de cualquier conflicto encontrado. Esta fase está dedicada a pruebas y documentaciones finales.

10.2. Actividades

- 10.2.1. Ciclo de investigación e inicio del proyecto:
 - Análisis de recursos
 - Definición de Estrategias.
 - Determinar indicadores.

- 10.2.2. Construcción de mecanismo consumidor de datos:
 - Diseño del consumidor de la api.
 - Manejo de solicitudes .
 - Modelaje de datos.

- 10.2.3. Implementación Modelo Vista Controlador:
 - Pseudocódigo de las estrategias
 - Creación de la base de datos.
 - Modelo de usuarios.
 - Guardar consultas
 - Implementación
 - Pruebas

- 10.2.4. Diseño de la vista:
 - Modelo de reporte.
 - Diseño gráfico del reporte.
 - Guardado de reporte.

- 10.2.5. Finalización:
 - Realización de Testeos
 - Corrección de errores
 - Deploy

11. Desarrollo

11.1. Características

11.1.1. Usuarios del sistema

Actor	Cliente				
Descripción	Este usuario se registrará con el objetivo de iniciar sesión en el sistema G-Backtest para usar las facultades específicas.				
Características	El usuario podrá realizar, consultar y eliminar los diferentes test en la plataforma.				
Relaciones	Relacionado con el método de compra, reserva y búsqueda de libros.				
Referencia	El sistema.				
Autor	Yoan Esteban Lopez Garcia	Fecha	21.03.22	Versión	1.0

Tabla 1 Usuarios del sistema

11.1.2. Historias de usuario

Historia de Usuario	
Número:001	Nombre: Registro de clientes
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: El usuario se podrá registrar llenando los datos correspondientes.	
Observaciones: Al finalizar, el usuario estará registrado y con la capacidad de usar el sistema.	
Criterio de Aceptación: El product owner confirma la historia	

Mockup:

Tabla 2 Historia de Usuario 001

Historia de Usuario	
Número:002	Nombre: Inicio de sesión
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: Un usuario llega al sistema e ingresa con su respectivo usuario y contraseña.	
Observaciones: Al terminar el proceso el usuario estará en condición de usar el sistema por completo.	
Criterio de Aceptación: El product owner confirma la historia	
Mockup:	

Tabla 3 Historia de Usuario 002

Historia de Usuario	
Número:003	Nombre: Restaurar contraseña.
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: Un usuario llega al sistema y tras olvidar su contraseña puede restablecerla por medio del correo.	
Observaciones: El usuario habrá restablecido correctamente su contraseña.	
Criterio de Aceptación: El product owner confirma la historia	
Mockup:	

Tabla 4 Historia de Usuario 003

Historia de Usuario	
Número:004	Nombre: Editar el perfil
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: Los usuarios podrán editar su nombre de usuario.	

Observaciones: El usuario habrá cambiado su nombre de usuario correctamente.
Criterio de Aceptación: El product owner confirma la historia
Mockup:


Tabla 5 Historia de Usuario 004

Historia de Usuario	
Número: 005	Nombre: Eliminar cuenta
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: Los usuarios podrán dar de baja sus respectivas cuentas.	
Observaciones: Los usuarios habrán eliminado su cuenta del sistema.	
Criterio de Aceptación: El product owner confirma la historia	
Mockup:	



Tabla 6 Historia de Usuario 005

Historia de Usuario	
Número:006	Nombre: Realizar Backtesting
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: El usuario puede entrar al método de Backtest e interactuar con todo lo que identifica un backtesting.	
Observaciones: EL usuario realiza pruebas de backtesting.	
Criterio de Aceptación: El product owner confirma la historia	
Mockup:	

Tabla 7 Historia de Usuario 006

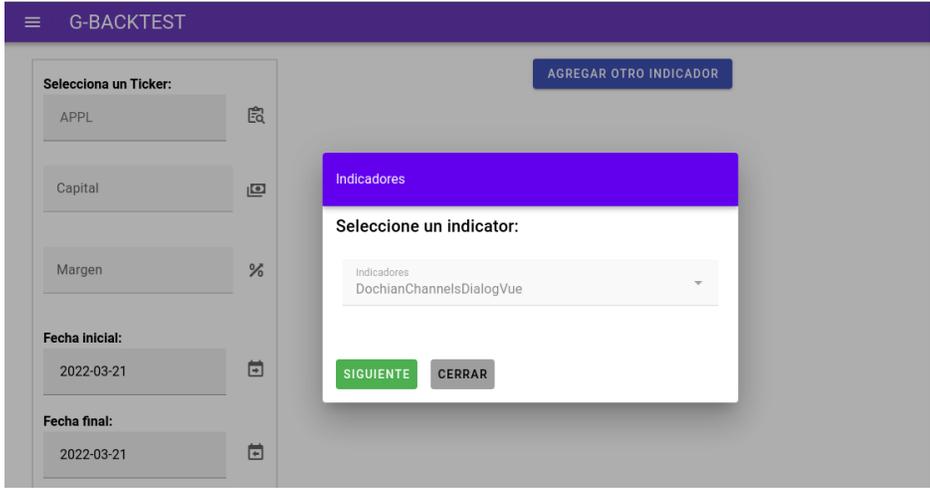
Historia de Usuario	
Número:007	Nombre: Agregar Indicador
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: El usuario puede entrar al método de Agregar indicador y elegir el indicador deseado.	
Observaciones: EL usuario agrega indicadores al backtesting.	
Criterio de Aceptación: El product owner confirma la historia	
Mockup:	
	

Tabla 8 Historia de Usuario 007

Historia de Usuario	
Número:008	Nombre: Configurar ticker
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: El usuario ingresara un ticker y su respectiva configuración para el backtesting	
Observaciones: El usuario habrá ingresado un ticker con su configuración básica.	
Criterio de Aceptación: El product owner confirma la historia	

Mockup:

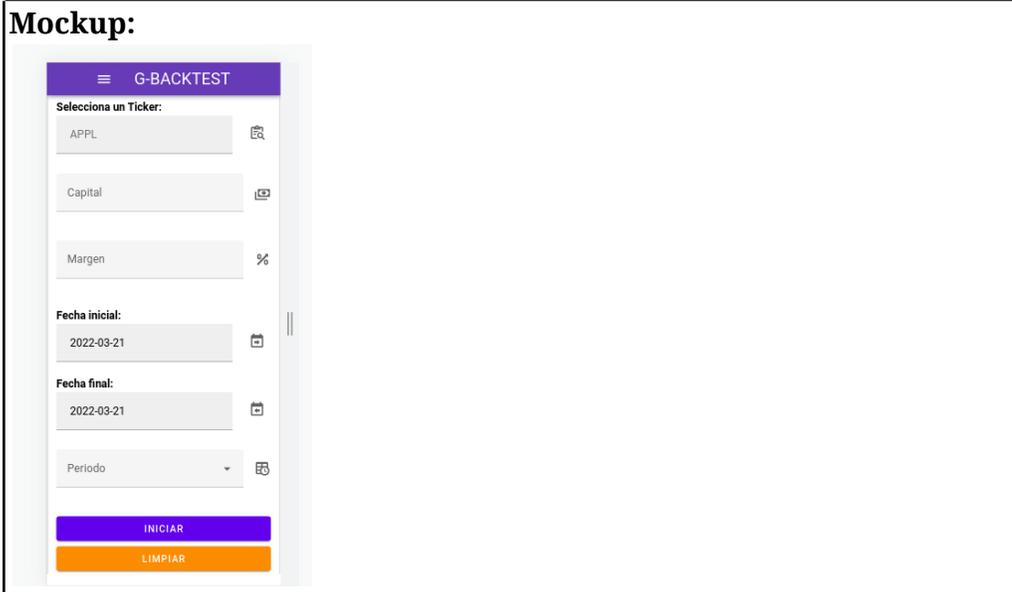


Tabla 9 Historia de Usuario 008

Historia de Usuario	
Número:009	Nombre: Ver Resultados
Prioridad en Negocio: Alta	Responsable: Desarrollador
Rol: Cliente	Iteración Asignada: 1
Descripción: El usuario podra observar y analizar los resultados de la consulta.	
Observaciones: El usuario habrá visualizado resultados del backtesting.	
Criterio de Aceptación: El product owner confirma la historia	
Mockup:	

The screenshot shows a web interface for 'G-BACKTEST'. At the top, there are three tabs: 'Resumen', 'Operaciones', and 'Grafico'. Below the tabs is a summary card with the following data:

Operations	20
Win Rate	25%
Total Profit/Loss	-9.506600000000127
P/L %	%
Max Profit	139.36
Max Loss	-36.639999999999986

Tabla 10 Historia de Usuario 009

Historia de Usuario																																									
Número:010	Nombre: Ver Historial de Backtest																																								
Prioridad en Negocio: Alta	Responsable: Desarrollador																																								
Rol: Cliente	Iteración Asignada: 1																																								
Descripción: El usuario podrá ver su historial de backtesting realizados																																									
Observaciones: El usuario habrá visualizado todas sus test hechos previamente.																																									
Criterio de Aceptación: El product owner confirma la historia																																									
Mockup:																																									
<p>The mockup shows a table with the following columns: #, Ticker, Capital, Fecha de inicio, Fecha final, Intervalo, Indicadores, and Opciones. There are four rows of data:</p> <table border="1"> <thead> <tr> <th>#</th> <th>Ticker</th> <th>Capital</th> <th>Fecha de inicio</th> <th>Fecha final</th> <th>Intervalo</th> <th>Indicadores</th> <th>Opciones</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>AAAU</td> <td>10000</td> <td>2021-12-05</td> <td>2022-03-16</td> <td>30min</td> <td>Indicadores</td> <td>Play, Delete</td> </tr> <tr> <td>2</td> <td>AAAU</td> <td>10000</td> <td>2022-01-01</td> <td>2022-03-16</td> <td>30min</td> <td>Indicadores</td> <td>Play, Delete</td> </tr> <tr> <td>3</td> <td>AAPL</td> <td>10000</td> <td>2021-11-01</td> <td>2022-03-17</td> <td>30min</td> <td>Indicadores</td> <td>Play, Delete</td> </tr> <tr> <td>4</td> <td>ACHC</td> <td>10000</td> <td>2022-01-01</td> <td>2022-03-21</td> <td>30min</td> <td>Indicadores</td> <td>Play, Delete</td> </tr> </tbody> </table>		#	Ticker	Capital	Fecha de inicio	Fecha final	Intervalo	Indicadores	Opciones	1	AAAU	10000	2021-12-05	2022-03-16	30min	Indicadores	Play, Delete	2	AAAU	10000	2022-01-01	2022-03-16	30min	Indicadores	Play, Delete	3	AAPL	10000	2021-11-01	2022-03-17	30min	Indicadores	Play, Delete	4	ACHC	10000	2022-01-01	2022-03-21	30min	Indicadores	Play, Delete
#	Ticker	Capital	Fecha de inicio	Fecha final	Intervalo	Indicadores	Opciones																																		
1	AAAU	10000	2021-12-05	2022-03-16	30min	Indicadores	Play, Delete																																		
2	AAAU	10000	2022-01-01	2022-03-16	30min	Indicadores	Play, Delete																																		
3	AAPL	10000	2021-11-01	2022-03-17	30min	Indicadores	Play, Delete																																		
4	ACHC	10000	2022-01-01	2022-03-21	30min	Indicadores	Play, Delete																																		

Tabla 11 Historia de Usuario 010

11.1.3. Diagramas de caso de uso

Caso de uso: Gestion de usuarios

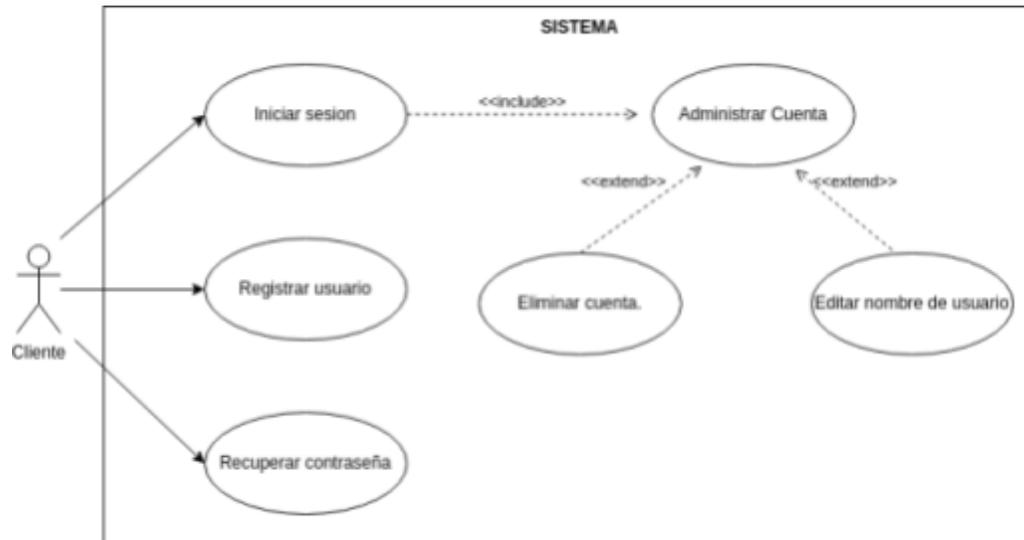


Ilustración 1 Gestion de usuarios

Caso de uso: Módulo Backtesting

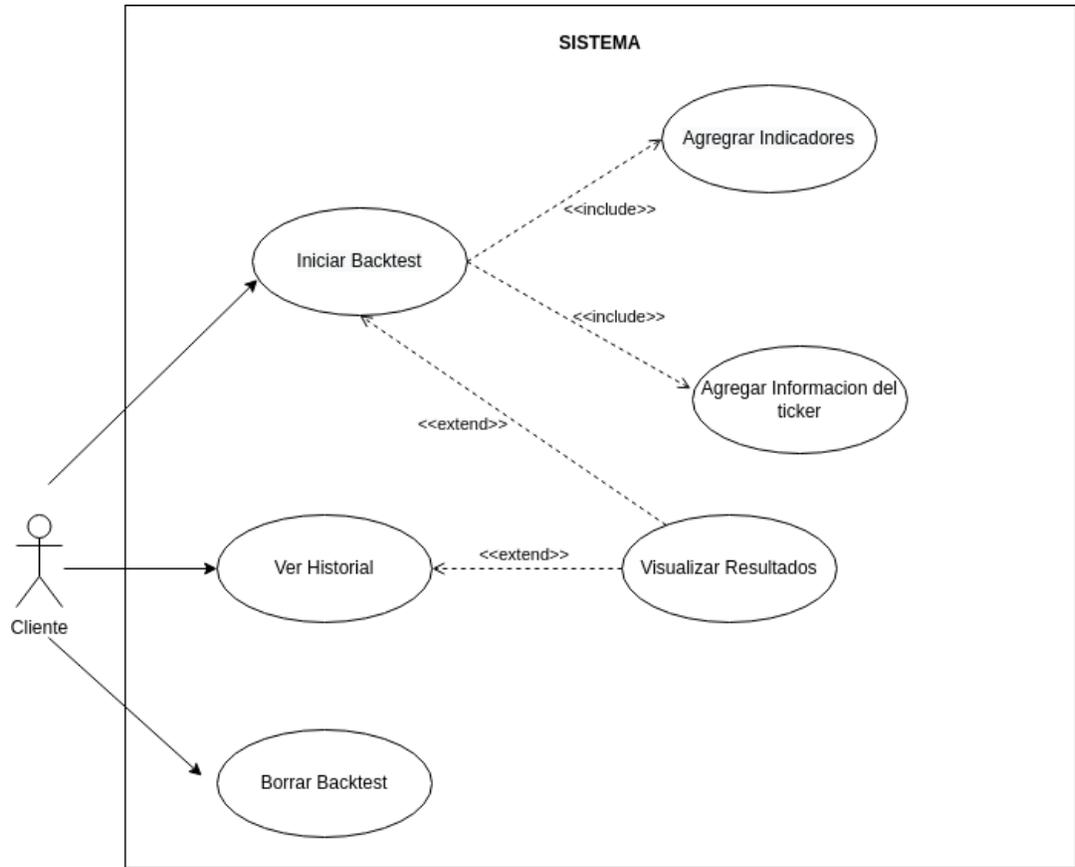


Ilustración 2 Módulo Backtesting

Caso de uso: Servicio de consumo

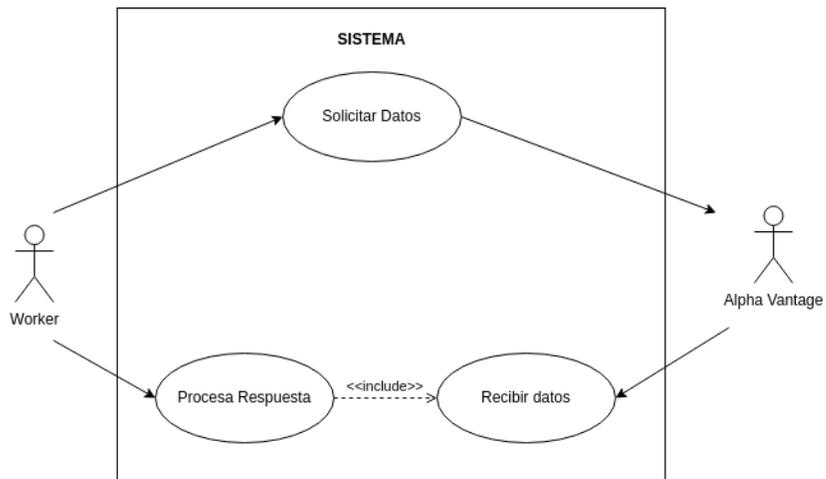


Ilustración 3 Servicio de consumo

11.1.4. Diagramas de secuencia o actividades

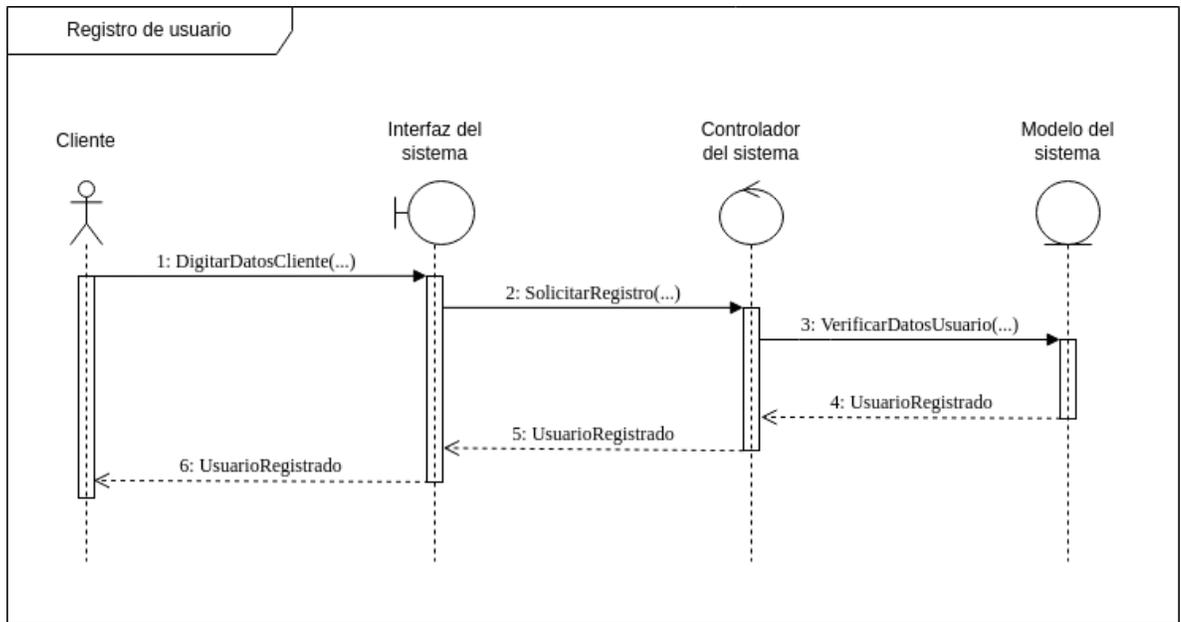


Ilustración 4 Registro de usuarios

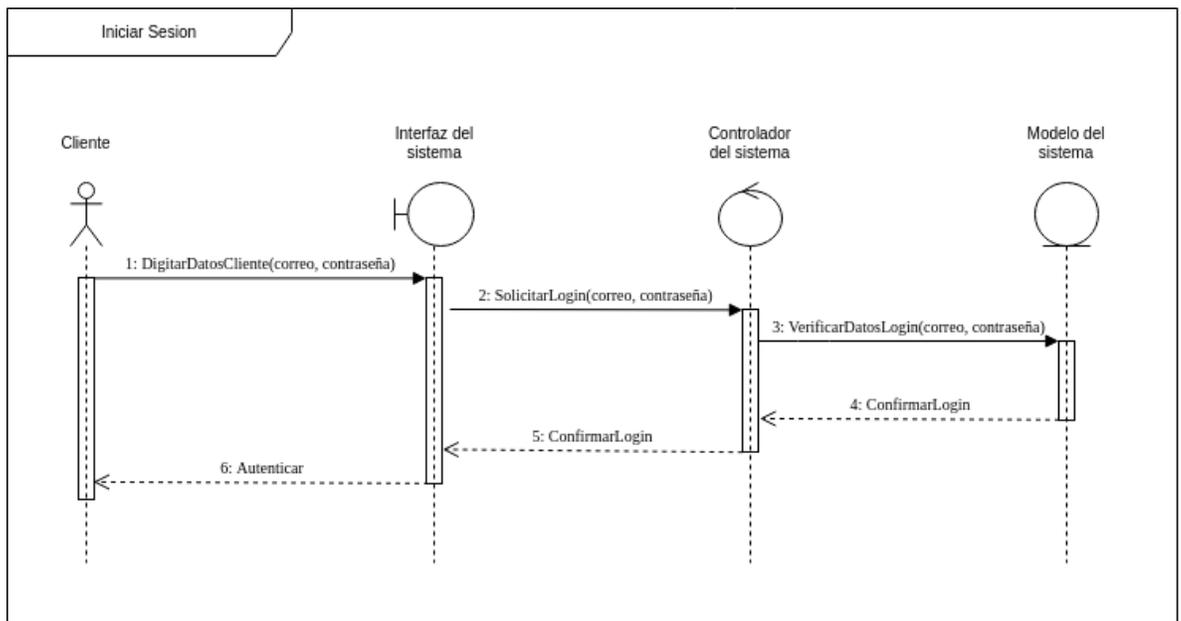


Ilustración 5 Iniciar sesión

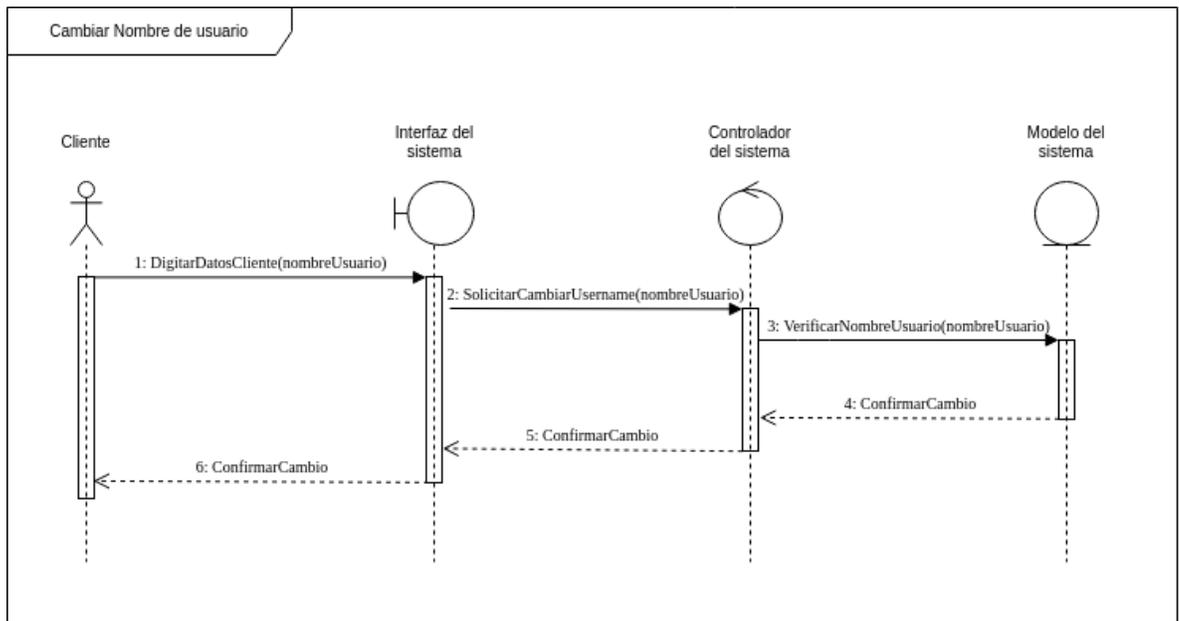


Ilustración 6 Cambiar nombre de usuario

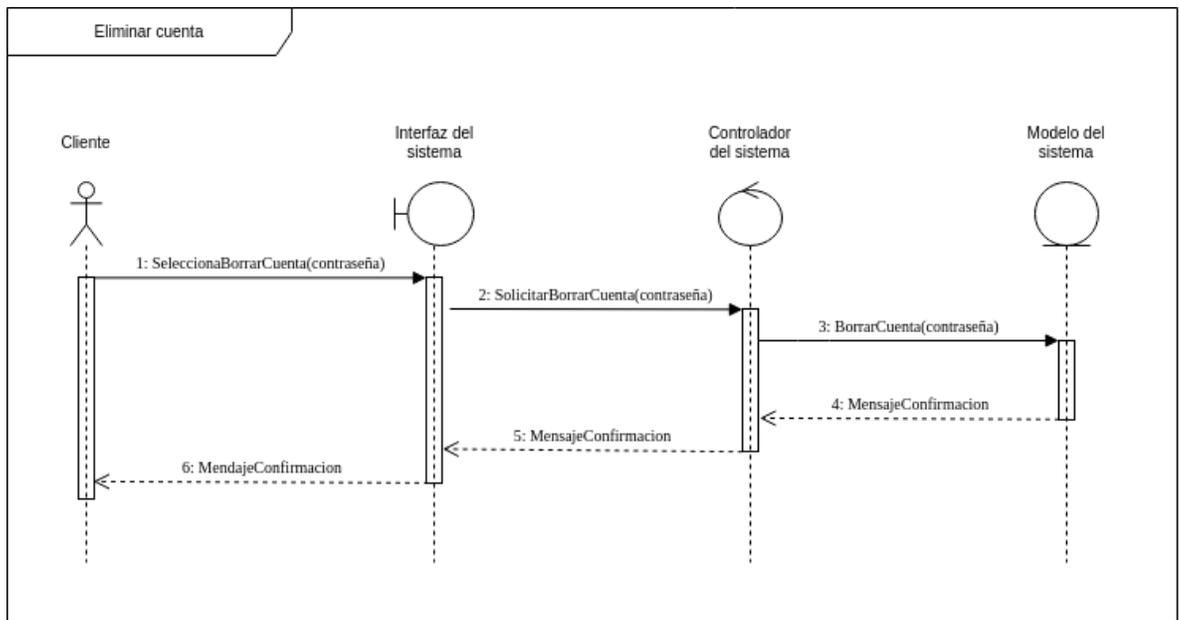


Ilustración 7 Eliminar cuenta

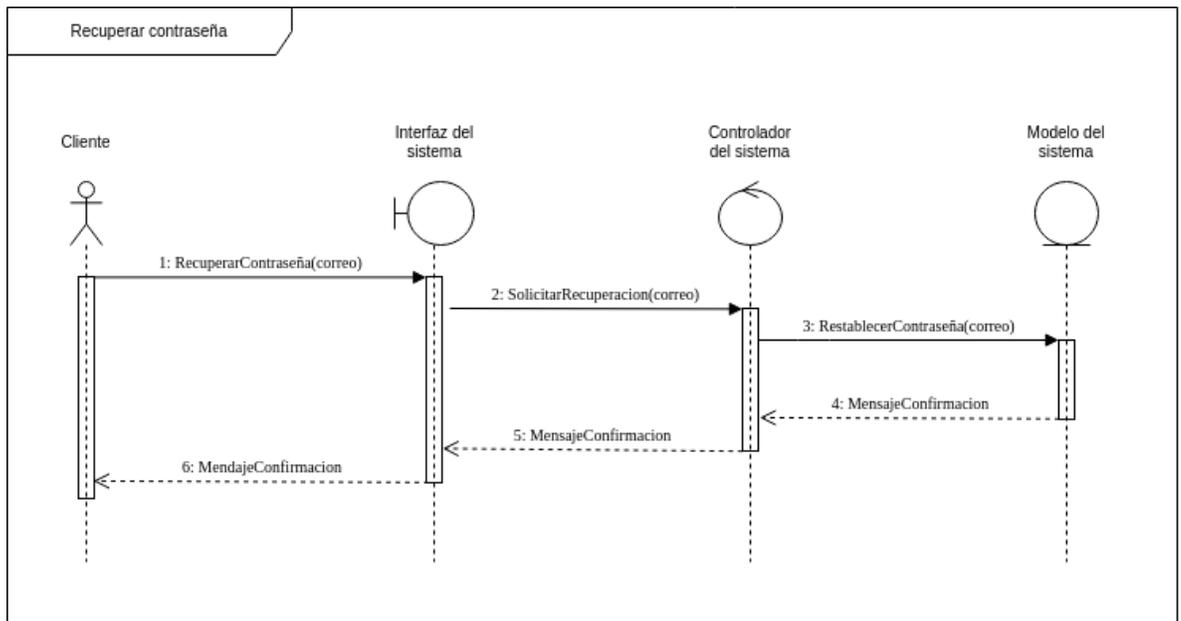


Ilustración 8 Recuperar contraseña

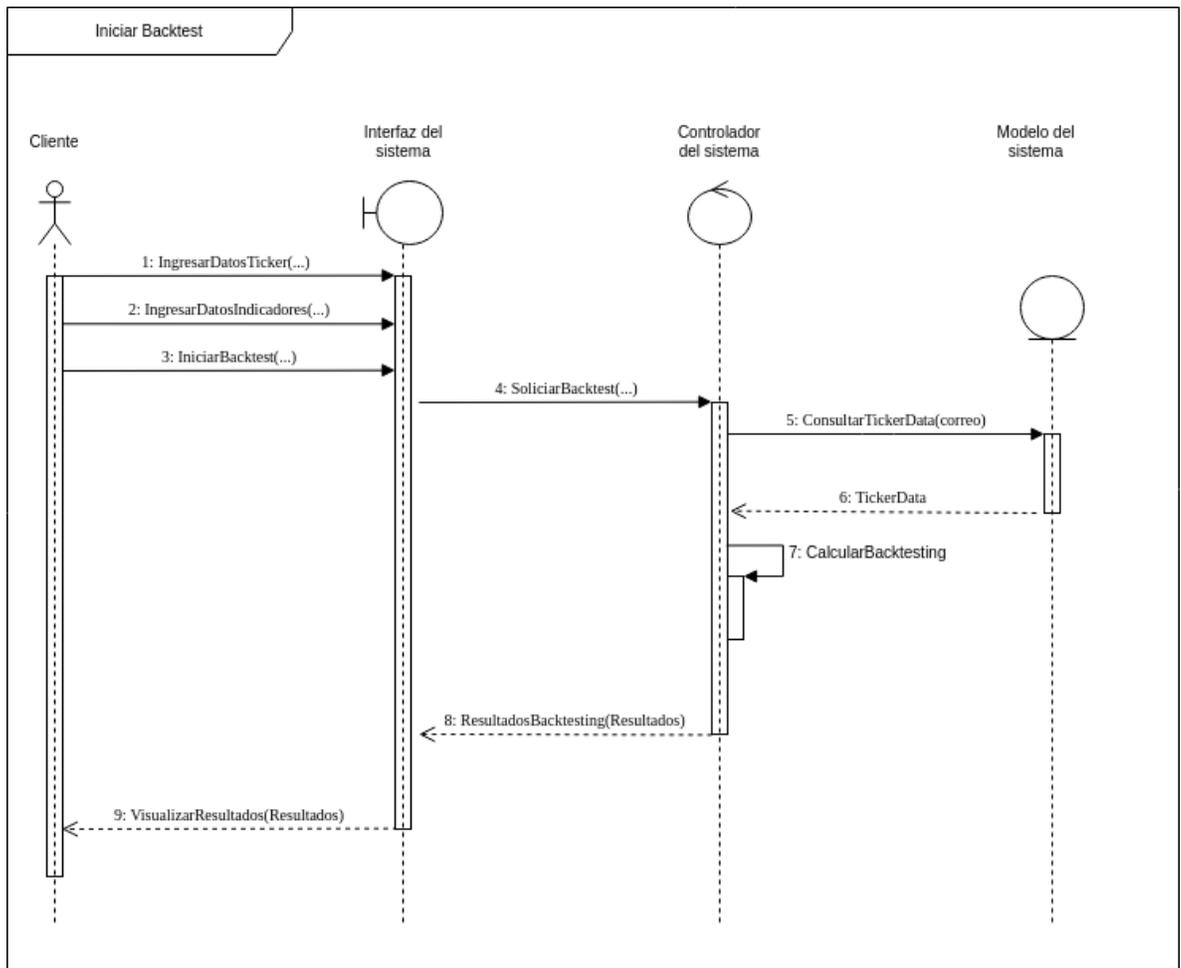


Ilustración 9 Iniciar backtest

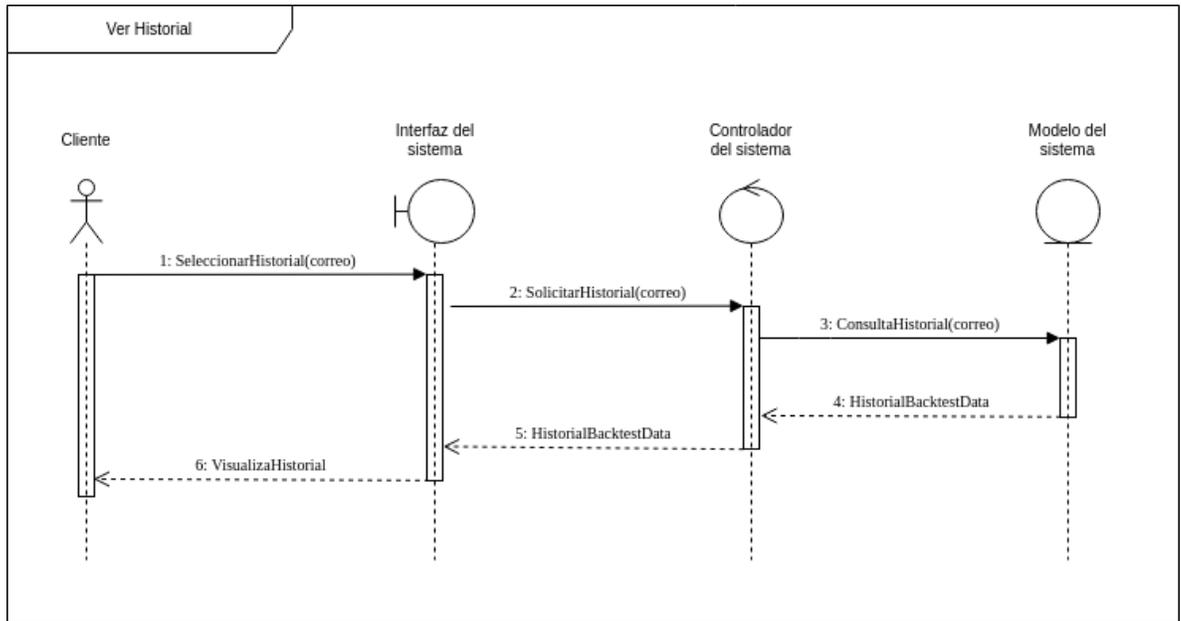


Ilustración 10 Ver historial

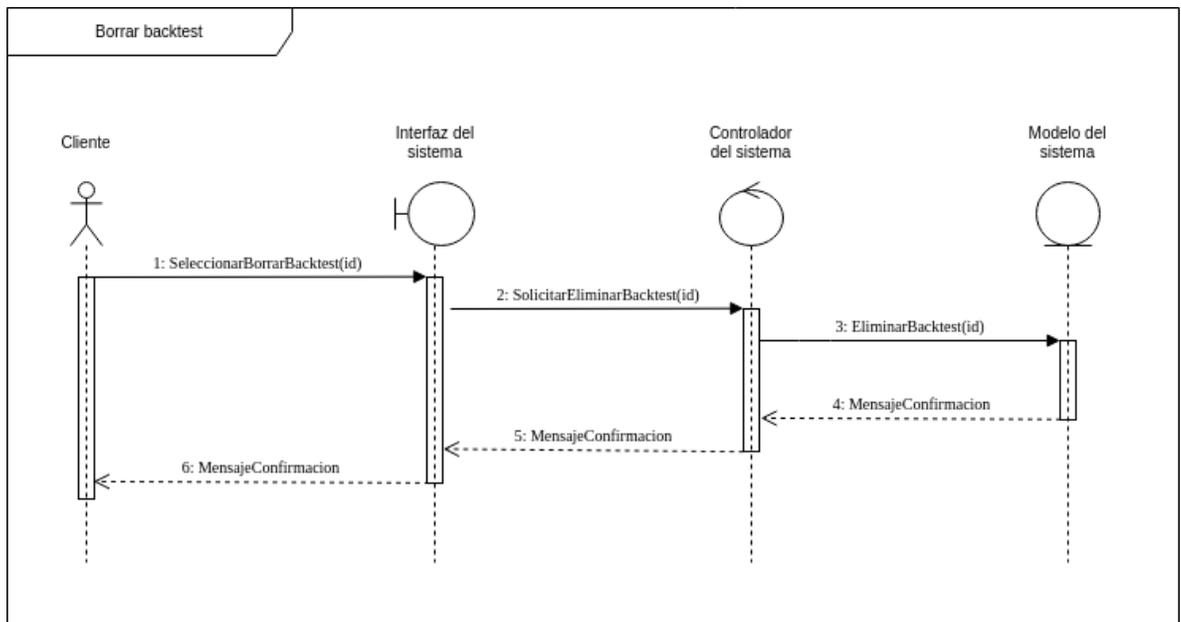


Ilustración 11 Borrar backtest

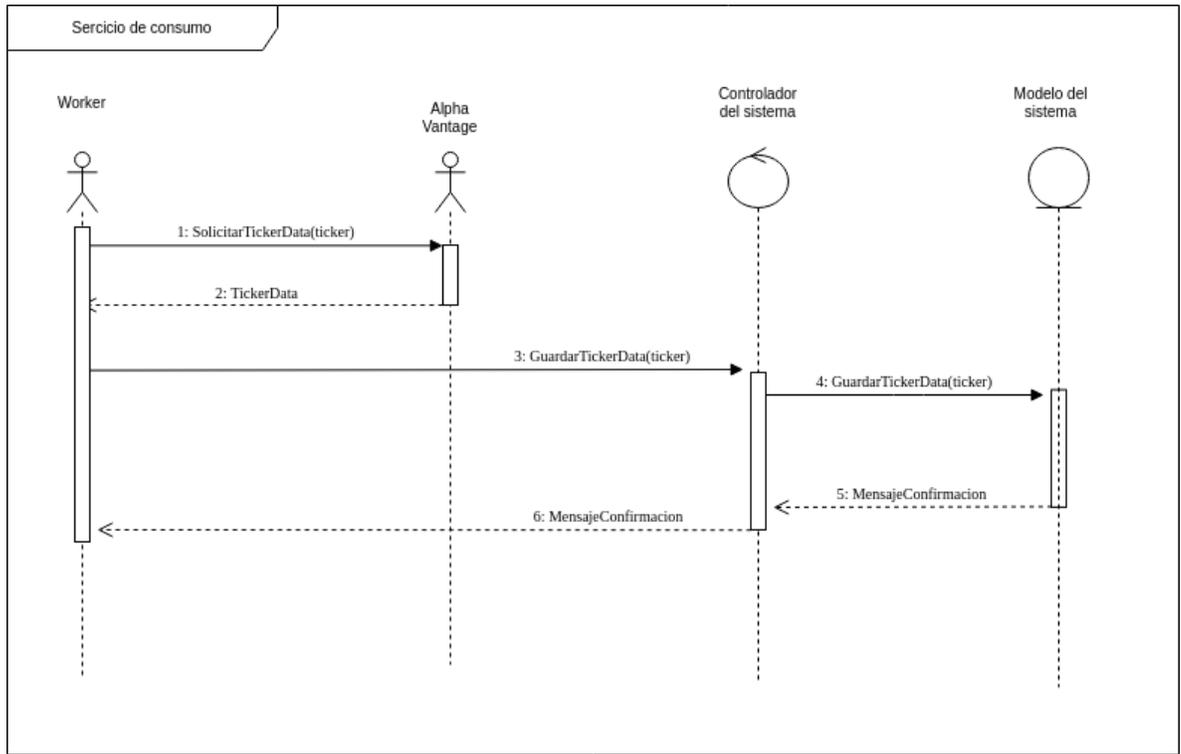


Ilustración 12 Servicio de consumo

11.1.5. Diagrama de base de datos

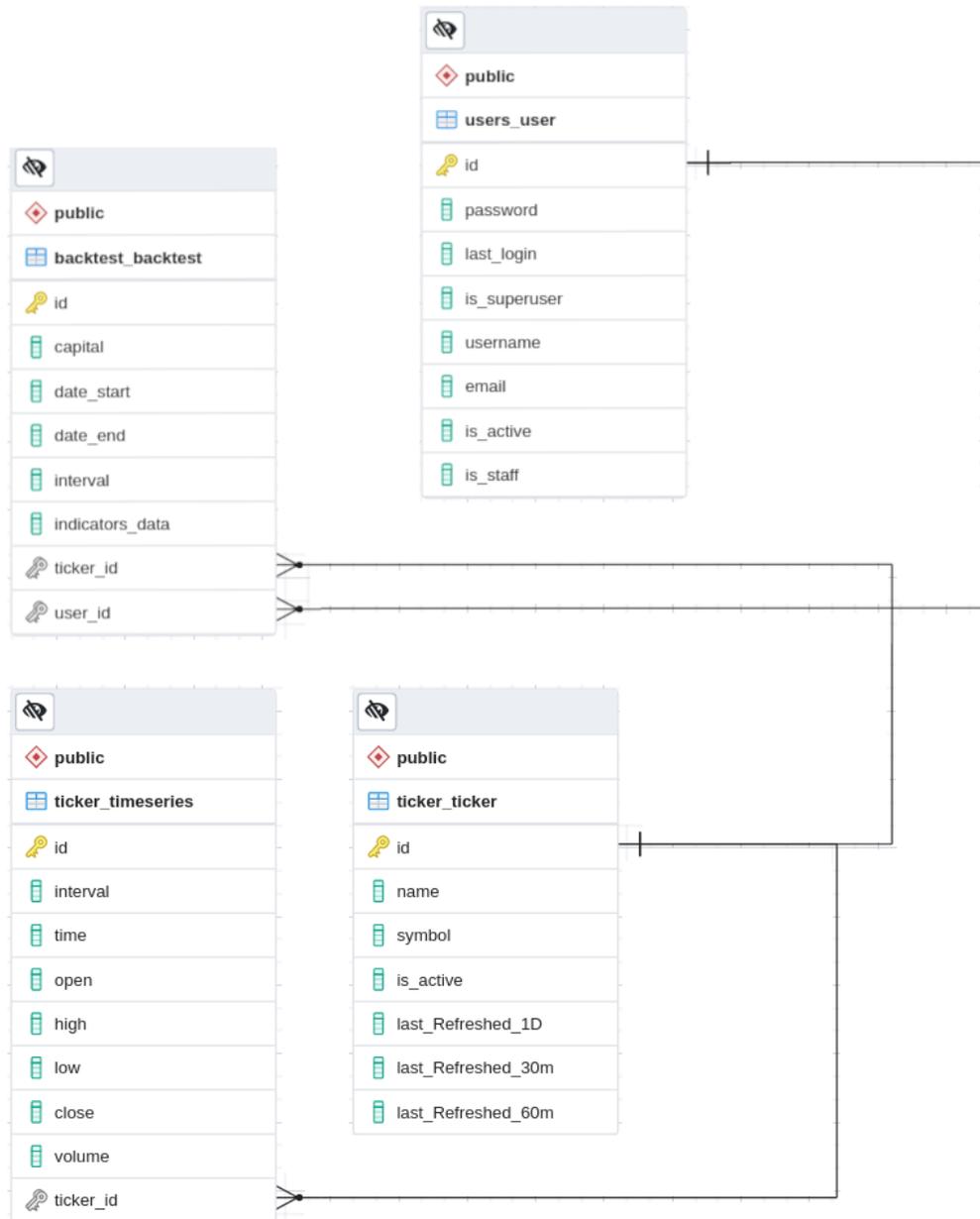


Ilustración 13 Diagrama de base de datos

11.1.6. Diagrama de clases

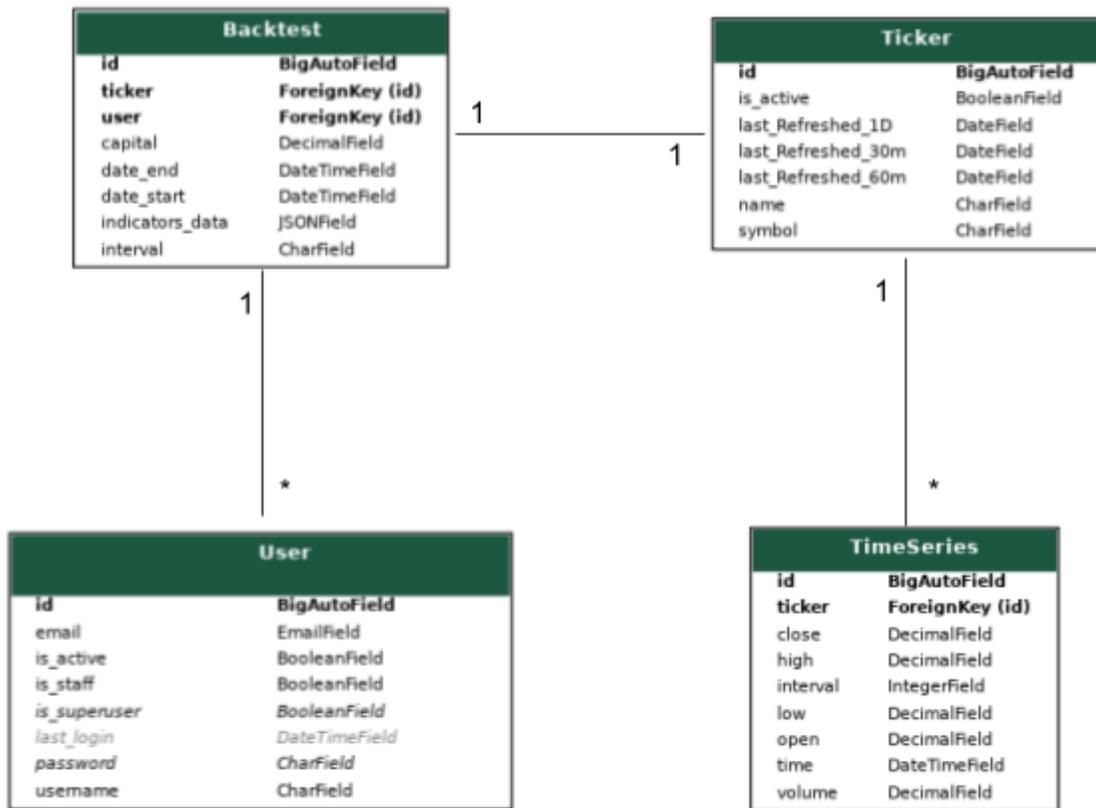


Ilustración 14 Diagrama de clases

11.1.7. Diagrama de despliegue

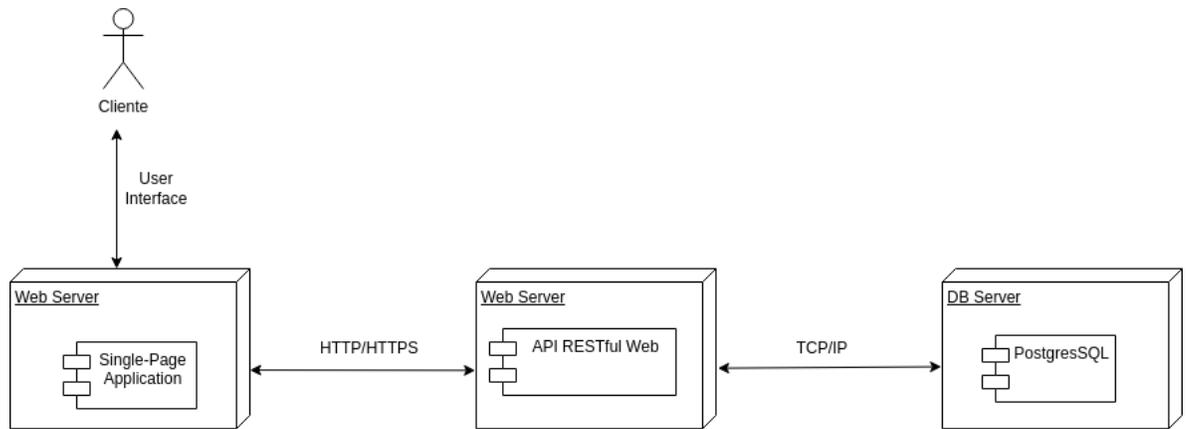


Ilustración 15 Diagrama de despliegue

11.1.8. Diagrama de arquitectura

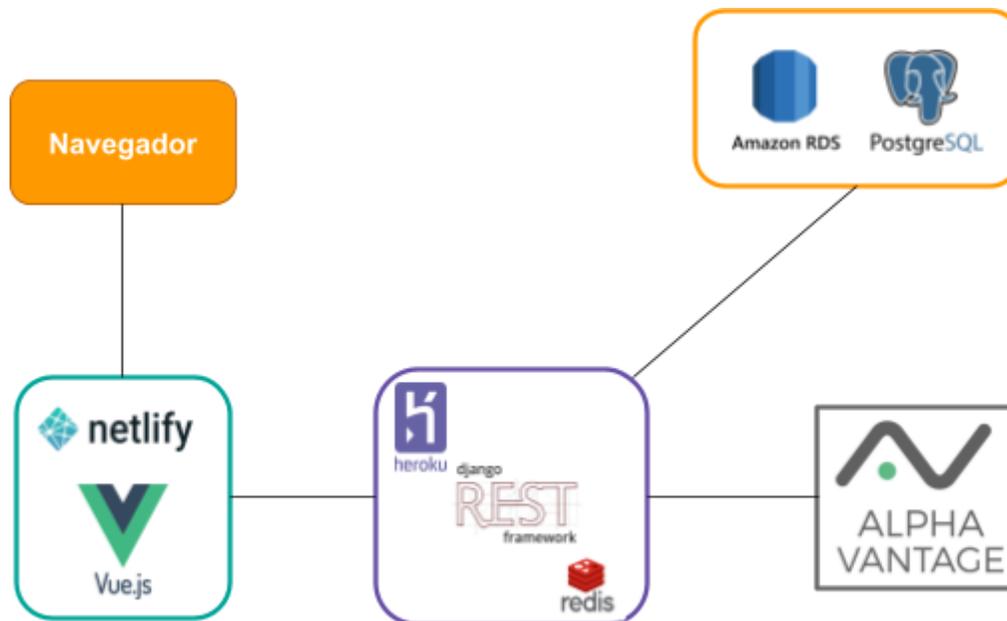


Ilustración 16 Diagrama de arquitectura

11.2. Fase de implementación

11.2.1. Lenguaje de programación

El proyecto está orientado totalmente en un entorno web, para el lado del cliente se hace necesario el lenguaje de programación Javascript, el cual posee un claro poder sobre el código html proporcionando la construcción de contenidos dinámicos.

Por el lado del servidor se escoge el lenguaje de programación python debido a que posee una amplia trayectoria en operaciones con grandes cantidades de datos, además de su reconocida línea de aprendizaje, el cual facilita la compresión de extensiones o librerías requeridas para el proceso de desarrollo del backtesting.

11.2.2. Motor de base de datos

“También conocido como Postgres , es un sistema de gestión de bases de datos relacionales gratuito y de código abierto (RDBMS) enfatizando la extensibilidad y el cumplimiento de SQL.”

Se hace uso de este motor de base de datos por proveer funcionalidad, rendimiento y seguridad óptimos para el proyecto y además de contar con previa experiencia de uso.

11.2.3. Framework

Se utilizó por el lado del cliente el framework Vue.js en su versión 3, el cual utiliza Javascript para construir interfaces de usuario por medio de procedimientos eficaces independientes de su complejidad, es progresivo en el tiempo y su nueva funcionalidad llamada Composition API convierte la

forma de codificar en menos rígida y más análoga a la programación funcional de JavaScript sin abandonar su principal característica de ser reactivo, además ayuda a la reutilización de código optimizando el desarrollo del proyecto. Paralelamente se emplea Vuetify.js extendido sobre Vue.js como herramienta de experiencias de usuario ricas y atractivas por su construcción en Material Design.

Por el lado del servidor se ocupa el framework Django REST framework como herramienta potente y adaptable para producir API Web, utilizado por muchas empresas y con gran compatibilidad con las extensiones del lenguaje de programación python. Conjuntamente se dispone Pandas como una herramienta poderosa para el análisis de los datos.

11.2.4. IDE de desarrollo

“Visual Studio Code es un editor de código fuente ligero pero potente que se ejecuta en su escritorio y está disponible para Windows, macOS y Linux. Viene con soporte incorporado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes (como C++, C#, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity).”

11.2.5. Adquisición de los datos históricos

En el proyecto se trabajó con la API Alpha Vantage para la obtención de los datos históricos de las diferentes acciones y ETFs de la bolsa de valores de New York. Esta brinda servicios gratuitos y premium; en el proyecto se trabajó con el servicio gratuito, limitando así la cantidad de 500 solicitudes que se pueden realizar por día.

La API proporciona una gran variedad de resoluciones temporales, para la plataforma se decidió trabajar con 3 resoluciones temporales (30 minutos, 60 minutos y diaria), esto con el fin de aliviar las limitaciones mencionadas anteriormente.

En el proyecto se desarrolló un servicio consumidor de está API considerando las anteriores características, se utilizó Celery el cual es una cola de tareas de trabajos asíncronos basado en el paso de mensajes distribuidos. Además de agregar la extensión Django Celery Beat, que permite la ejecución de tareas periódicas en el backend. Celery usa brokers para comunicar los mensajes entre los workers de Celery y Celery Beat con el proyecto Django.

Para el consumo de las 500 solicitudes gratuitas se decide dividir este número en 3 el cual es el número de resoluciones temporales escogidas, redondeando el resultado a 165 y dividir este valor por 86400 el cual es el número de segundos en un día, con la finalidad de obtener el periodo en segundos para la consulta a la API y así equilibrar el trabajo y consultas a la base de datos y servidor y paralelamente dando espacio para no romper la limitación de 5 solicitudes a la API por cada 5 minutos.

$$SW = \frac{500 \text{ solicitudes}}{\text{número de resoluciones temporales}}$$

$$P = \frac{86400s}{SW}$$

donde:

SW = Número de operaciones por worker

P = Periodo de ejecución de cada worker

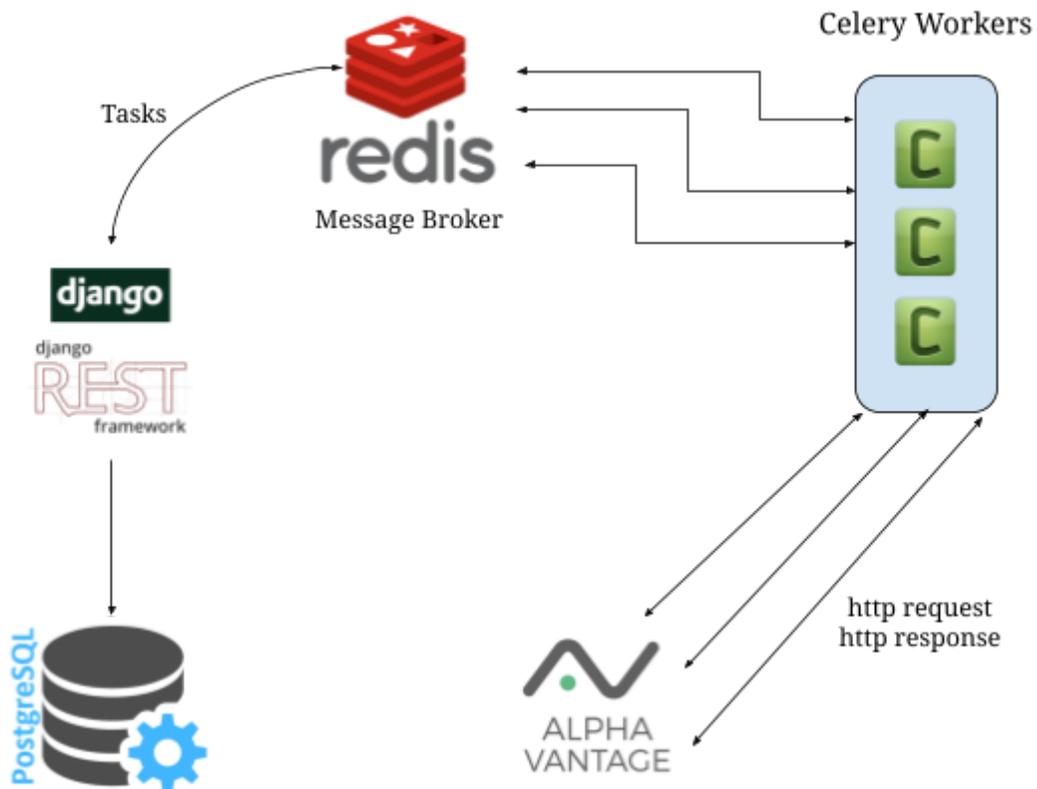


Ilustración 17 Adquisición de los datos históricos

12. Trabajos futuros

Dentro del desarrollo y expansión de la plataforma se prevé aumentar el número de indicadores disponibles para la generación de estrategias comerciales más

complejas, asimismo adicionar diversas configuraciones para cada señal y lograr un backtesting no solo orientado al usuario principiante sino también ofrecerlo como una gran herramienta a aquellos comerciantes experimentados.

Adicionalmente como iniciativa de perfeccionamiento se idea la integración de una librería mas novedosa e intuitiva para la exposición de los datos con los respectivos indicadores escogidos por el usuario.

13. Conclusiones

El desarrollo de la plataforma significó un reto de gran escala debido al dominio del tema de la bolsa de valores y sus distintos enfoques técnicos que conlleva, desde comprender temas sencillos como acciones y empresas hasta temas más complejos como el análisis técnico.

La solución planteada fue completamente satisfactoria, el software no requiere conocimiento mínimos de programación para la realización de pruebas, además de contar con una interfaz de usuario rica en componentes visuales y agradable a primera vista, sus resultados pueden observarse con gran calidad de detalle a través de sus diferentes opciones además de poseer la integración de gráficos ligeros para la observación del historial de mercado del rango solicitado.

Además el cambio de contexto de trabajar con un equipo donde se reparte el trabajo equitativamente, a un modo de trabajo solitario donde se tiene gran responsabilidad de todos los enfoques del proyecto fue difícil de manejar, pero con una correcta división y planificación del trabajo asimismo operando

bajo la metodología Kanban, fue factible la elaboración de la plataforma.

La implementación del framework Vue 3 permitió un trabajo sencillo pero muy completó para todas las funcionalidades requeridas, pero la capa de diseño visual Vuetify 3 género complicaciones debido a que en el momento del desarrollo de este proyecto solo contaba con su versión 3 la cual suministraba recursos en versión alfa, dando numerosos fallos de compatibilidad y disponibilidad con Vue 3.

El suministro de datos desde la API Alpha Vantage se manejó eficazmente gracias al modelado previo del servicio consumidor, logrando agotar las 500 solicitudes diarias gratuitas ofrecidas por la aplicación.

Anexos

Anexo 1. Cronograma del proyecto

FASE	DETALLES	T1				T2													
		DIC				ENE				FEB				MAR					
ACTIVIDADES		SEMANA DEL PROYECTO:		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Ciclo de investigación e inicio del proyecto Análisis de recursos Definición de Estrategias Determinar indicadores			█															
2	Construcción de mecanismo consumidor de datos Diseño del consumidor de la api. Manejo de solicitudes Modelaje de datos			█				█											
3	Implementacion Modelo Vista Controlador Pseudocódigo de las estrategias Creación de la base de datos. Modelo de usuarios. Guardar consultas Implementación Pruebas			█				█				█							
5	Diseño de la vista Modelo de reporte Diseño gráfico del reporte. Guardado de reporte							█				█							
6	Finalización Realización de Testeos Correccion de errores Deploy											█							

FIN DEL PROYECTO

Tabla 12 Cronograma del proyecto

Bibliografía

[1] Murphy, John J. (1999). Análisis técnico de los mercados financieros. Ediciones Gestion 2000

[2] FBS MARKETS INC. (2019, January 30). Backtesting manual de una estrategia de trading. FBS. <https://esfbs.com/analytics/guidebooks/how-to-backtest-a-trading-strategy-266>

[3] Kostiainen, K. (2016). Development of Trading Algorithm Backtest Environment. <https://www.semanticscholar.org/paper/Development-of-Trading-Algorithm-Backtest-Kostiainen/0bede456f4c0956dac6b62b05d18ac15ed933208>

[4] Verma, Ritik & Gohrani, Kunal & Mitra, Anirban & Das, Anirban & Chakrabarti, Tulika & Chakrabarti, Prasun. (2021). A computing-based system developed on approaches and strategies to analyze share values for trading intentions in the Stock Market Domain. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3892735

[5] Gallo, Crescenio & Fratello, Angelo. (2014). The Forex Market in Practice: A Computing Approach for Automated Trading Strategies. International Journal of Economics and Management Sciences. 3. 10.4172/2162-6359.1000169.

[6] Ni, Jiarui & Zhang, Chengqi. (2005). An Efficient Implementation of the Backtesting of Trading Strategies. 3758. 126-131. 10.1007/11576235_17.

[7] Spörer, Jan, Backtesting of Algorithmic Cryptocurrency Trading Strategies (April 28, 2020). Available at SSRN: <https://ssrn.com/abstract=3620154> or <https://dx.doi.org/10.2139/ssrn.3620154>

[8] Wikipedia Contributors, "PostgreSQL," *Wikipedia*, Mar. 09, 2022. <https://en.wikipedia.org/wiki/PostgreSQL> (accessed Mar. 23, 2022).

[9] Microsoft, "Visual Studio Code," *Visualstudio.com*, Nov. 03, 2021. <https://code.visualstudio.com/docs> (accessed Mar. 23, 2022).

[10] Investopedia. (2021). Investopedia. <https://www.investopedia.com/>