Criptoanálisis y automatización de transacciones.

Nityananda Lorenzo Hernández,
I.E.S San Vicente, San Vicente del Raspeig.

1 Introducción

Actualmente la idea de comercio mediante monedas completamente virtuales está en alza, el intercambio de este tipo de divisas se está convirtiendo en un mercado en auge que cada vez llama la atención de más gente. Se trata de un mercado volátil, fuertemente digitalizado, en el que existe la posibilidad de obtener beneficios aprovechando los márgenes y variaciones del propio poder adquisitivo de las diferentes divisas que emergen día tras día.

Basándome en los conocimientos aprendidos durante el curso, aunque las tecnologías no son exactamente las aprendidas durante el mismo, ya que por lo general hemos estudiado el desarrollo de apis basado en Spring Java, he considerado que la realización del proyecto en ASP .NET ofrece ventajas frente a Java en muchos aspectos del desarrollo.

Para el proyecto, haré uso de una base de datos en Postgres, cuyo cometido principal es almacenar los resultados obtenidos a través de diferentes configuraciones de la aplicación, los detalles de usuarios que se registren en la aplicación, y múltiples características variables de la aplicación, además de un histórico de valores de transacción para poder usar como base frente a una posible ampliación futura de las funcionalidades del servicio.

La arquitectura de mi aplicación incluye varios servicios individuales, y ciertos puntos de mensajería en tiempo real. El servicio principal se hará cargo de, en base a ciertos valores almacenados en la base de datos, realizar consultas sobre el estado actual de las varias criptomonedas que están siendo trabajadas, y una vez actualizado su estado, enviar un mensaje con los nuevos datos a todo servicio que se haya suscrito a las actualizaciones de dicho estado.

Como sistema de seguridad y autenticación he decidido implementar un sistema basado en Oauth2, el servicio estará habilitado de forma externa a la funcionalidad principal, y será consultado cada vez que un usuario trate de registrarse o iniciar sesión en el servicio principal.

Los usuarios normales podrán indicar la cantidad de criptomonedas que disponen en los respectivos bancos con los que trabaja mi aplicación, los márgenes de compra-venta, y sus credenciales para habilitar las mismas, mientras que los usuarios administradores de la aplicación podrán incluir nuevos bancos insertando detalles de los varios endpoint necesarios para la automatización de transacciones en la base de datos. La aplicación será capaz de consultar cualquier endpoint que devuelva una respuesta en formato JSON, para ello se emplearán herramientas en la lógica de la aplicación como las variables dinámicas de .NET y expresiones de JSONpath.

En resumen, el objetivo principal de este proyecto es crear un servicio que permita consultar el estado actual de cualquier moneda virtual sin importar la estructura del api que ofrezca dicho dato, almacenando los resultados para disponer de un histórico de valores controlado, y manteniendo una estructura normalizada para su consulta por el servicio de automatización de transacciones. En base a los datos históricos acumulados, el objetivo secundario a futuro es ser capaz de generar un modelo predictivo para la toma de decisiones a la hora de determinar los márgenes de transacción viables para varias criptomonedas, y la aplicación de las mismas.

2 Antecedentes

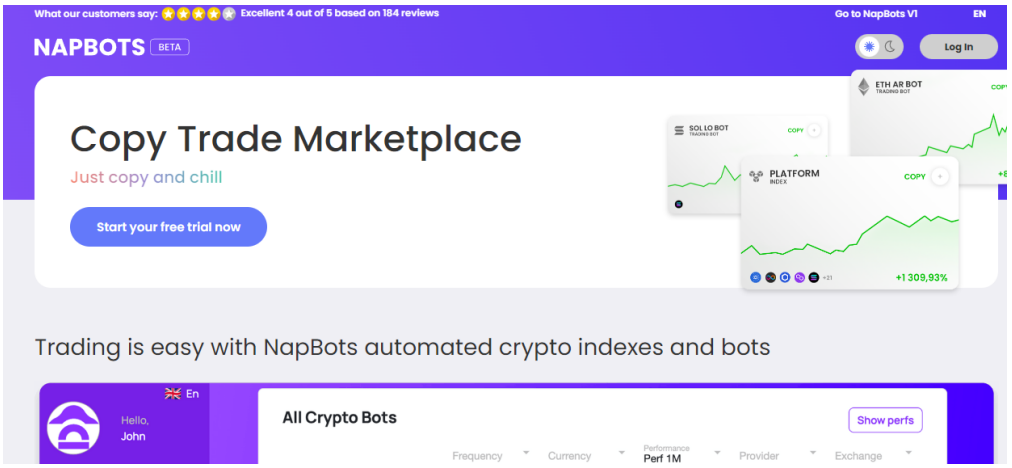
Tras realizar un estudio de mercado he detectado que existe una cantidad considerable de bots de arbitraje. Los bots existentes aseguran confidencialidad, control sobre los movimientos y las configuraciones de los intercambios y estadísticas sobre los resultados, no obstante la mayoría están sujetos a una serie de mercados concretos, y aunque ofrezcan confianza, cuando haces uso de este tipo de servicios estás depositando bienes de gran valor en manos de terceros, y suelen ser servicios de pago ya sea mediante comisiones o suscripción.

Encontrando esta situación, considero que un bot que permita a los usuarios adaptar cualquier mercado a su lista de intercambios conocidos tras incluir algunos detalles del mismo tendría una fuerte ventaja competitiva frente a los que existen actualmente.

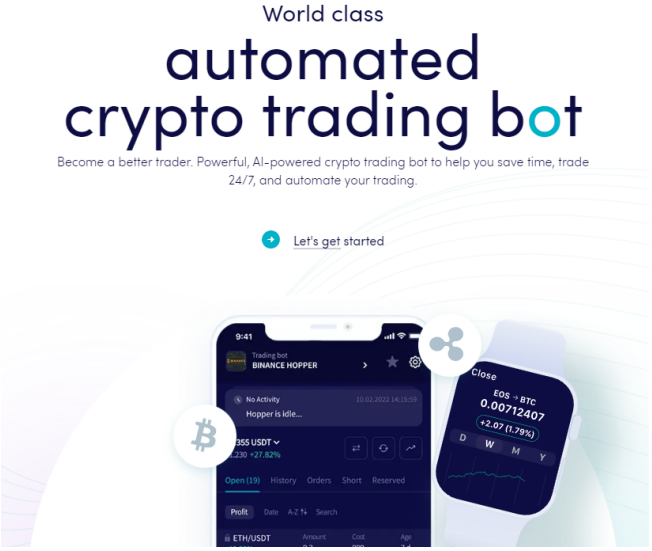
2.1 Diversos bots de arbitraje existentes en el mercado actual

A continuación se incluyen múltiples opciones existentes en el mercado actual que ofrecen características similares a los objetivos de este proyecto.

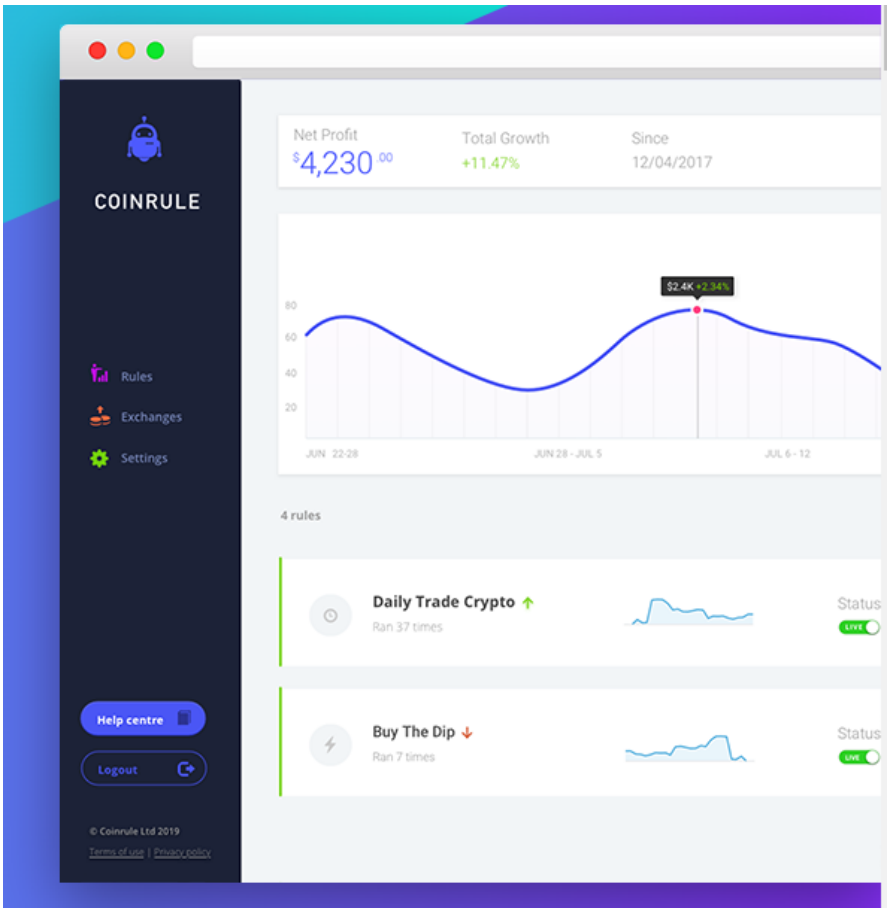
- **NapBots:** Ofrece estadísticas y algoritmos de búsqueda preconfigurados, adjunto a un coste de 83 € / mes en caso de querer hacer uso de todo lo que ofrece.



- **CryptoHopper:** Ofrece estadísticas, algoritmos junto a múltiples intercambios y monedas, a cambio de alrededor de 99$ / mes.



- **Coinrule:** Tal y como los antecedentes, ofrece estadísticas y múltiples opciones y servicios, con un coste adjunto de 449 $ / mes.



En general, los equipos de desarrollo de estas aplicaciones varían de entre dos a veinte personas, los cuales obtienen beneficio ya sea mediante un gran porcentaje de comisión por los intercambios realizados con su herramienta, o por las suscripciones mensuales que debes aceptar si deseas hacer uso de las opciones más competitivas de su servicio.

3 Análisis

La aplicación deberá permitir a los usuarios introducir información relevante sobre cualquier intercambio de criptomoneda para que comience a realizar un seguimiento de las estadísticas de la misma. Concretamente, una serie de campos relevantes, como son el valor del último intercambio en la plataforma, la url de consulta, o el volumen de transacciones, entre otros. El usuario también deberá ser capaz de modificar las condiciones de compra / venta, los márgenes, y las monedas activas para los intercambios por cada mercado en el que trabaje.

En base a todos estos detalles, la aplicación gestionará de forma automática y periódica la recopilación de estos datos, lanzando las comprobaciones indicadas por el usuario para realizar los intercambios de criptomoneda tras el primer instante en el que se cumplan.

4 Diseño

Las principales funciones de la aplicación serán:

- Un microservicio de información con:
    - Un punto de acceso público para realizar la configuración de consulta.
    - Un punto de acceso público para activar o desactivar las diferentes monedas intercambiadas.
    - Un punto de acceso público para realizar la configuración de la lógica de intercambio.
    - Un sistema interno de normalización de datos que ajustará la respuesta de cualquier api a las necesidades de la base de datos de la aplicación.
    - Un trabajo de fondo que realizará peticiones a todas las apis conocidas de forma periódica y almacenará los resultados.
- Un microservicio de autenticación con:
    - Un punto de acceso para el registro de nuevos usuarios.
    - Un punto de acceso de validación de usuarios existentes, que devolverá los credenciales de autenticación para poder hacer uso del resto de microservicios.
- La propia aplicación de intercambio con:
    - La capacidad de activar o desactivar los procesos de intercambio.
    - Un trabajo de fondo que solicitará la información que necesita para aplicar la lógica de intercambio, y posteriormente la aplicará.

El objetivo de esta arquitectura de aplicación es tener una torre centralizada con el proceso más pesado de obtención de datos de apis, mientras que el resto de procesos puedan distribuirse y replicarse con facilidad sin tener la necesidad de ser ejecutados en el mismo hardware. De esta manera, la parte de intercambio se podrá ejecutar de forma distribuida, y totalmente focalizada en el proceso de intercambio.

4.1 Listado de endpoints del servicio de información

4.2 Esquema entidad-relación con la configuración del servicio de información