Universidad de San Andrés

Escuela de Negocios

Licenciatura en Finanzas

Trabajo de graduación

Smart contracts como Clearing House. Un modelo de exchange descentralizado

Autores: Jeronimo Houlin y Juan Rossi

Tutor: Gabriel Basaluzzo

Febrero 2021

Trabajo de graduación - Licenciatura en Finanzas

Smart contracts como *Clearing House*. Un modelo de *exchange* descentralizado

Febrero 2021

Autores

Jeronimo Houlin y Juan Rossi

Tutor

Gabriel Basaluzzo

Índice

Agradecimientos

A nuestras familias y a todos los profesores y compañeros que nos acompañaron durante la licenciatura.

Resumen

Para contestar la pregunta de si es posible construir un modelo de intercambio de valores que permita a los usuarios realizar inversiones en activos tradicionales utilizando activos digitales, indagamos en el actual estado de la materia, y proponemos en detalle un modelo de *exchange* descentralizado donde se operen derivados, haciendo uso de *Smart Contracts*. Para esto modelamos funciones del exchange tales como el algoritmo de *Order matching* del libro de órdenes y los supuestos que tomamos acerca del comportamiento de los usuarios.

Para llevar a cabo este trabajo, nos sumergimos en la literatura preexistente de los inicios de la tecnología blockchain, el surgimiento de los *smart contracts*, sus aplicaciones, y cómo podríamos aplicarlos a nuestro modelo, también indagamos el material que existía acerca de los exchanges tradicionales y los más contemporáneos *Decentralized Exchanges* (DEX) y por último en la materia del contrato *Contract for Difference* (CFD) que buscaremos replicar en un ambiente descentralizado.

Finalmente, realizamos una simulación de Monte Carlo para analizar los posibles diferenciales de precio entre la plataforma y el mercado del activo subyacente utilizando series de precios generadas por un movimiento browniano.

Capítulo 1. Introducción

Frente al resurgimiento en popularidad de activos digitales, ó criptomonedas, la historia financiera podría encontrarse frente a una revolución aún más importante que la creación de los primeros bancos comerciales en Europa durante el siglo XV, o la introducción de derivados estandarizados en el *Chicago Board of Trade* (CBOT) en el siglo XIX.

Las criptomonedas nacen en un contexto en el que gran parte de la población sintió un descontento general con el sistema financiero tradicional, tal como lo fue la post-crisis subprime 2008-2009. En 2009, fue publicado un paper titulado "Bitcoin: A Peer-to-Peer Electronic Cash System" en el cual presenta un sistema de transferencia de valores completamente digital con sólidos fundamentos matemáticos y criptográficos.

Años después, esta nueva moneda llamada Bitcoin (BTC) pasaría de ser un activo desconocido o generalmente asociado a actividades ilícitas, a tener una capitalización de mercado por encima del trillón de dólares, y ser utilizado como *asset class* por varios fondos institucionales. Su importancia también radica en el hecho de que sentaron las bases para cientos de otras redes, *tokens,* protocolos y proyectos descentralizados.

Actualmente, millones de personas están utilizando monedas digitales como almacenamiento de valor, ya que poseen numerosas ventajas, principalmente la seguridad, privacidad y la ausencia de fricciones y costos de transacción para realizar intercambios.

Sin embargo, uno de los principales problemas de las criptomonedas como reserva de valor es la alta volatilidad de sus cotizaciones frente a monedas fiduciarias como el dólar americano. Por lo cual proponemos un modelo de *exchange* en el que es posible utilizar criptomonedas para invertir en activos estables como índices, acciones, commodities o pares de monedas fiduciarias (*forex*), con el fin de brindar diversificación a inversores *retail*.

En el proceso analizaremos la industria de *brokerages* y sociedades de bolsa tradicionales, es decir, aquellas compañías que permiten a los inversores intercambiar sus monedas fiduciarias, emitidas por bancos centrales, por activos tales como acciones de compañías. También

estudiaremos los nuevos exchanges descentralizados en los cuales se pueden adquirir activos digitales.

Nuestra propuesta supone una fusión entre ambos modelos, esto permitiría a inversores acceder de forma rápida y eficiente a la exposición financiera de todo tipo de activos. De esta forma estos podrían gozar de una mayor diversificación, ya que permitiremos el acceso a mercados que son inaccesibles para inversores no institucionales. Al mismo tiempo lograrlo sin incurrir en los costos de tener que primero convertir sus cripto-activos en monedas fiduciarias y operar con diversos agentes de bolsa y bancos.

Este vendría a ser nuestro mercado objetivo, ó clientela para el corto/mediano plazo. Aunque el fin último sería generar una única plataforma en donde tanto los inversores retail como institucionales puedan tener mayor acceso a diferentes mercados y con un modelo sencillo.

El principal desafío radica en la actual incompatibilidad de estas dos clases de activos, y como solución proponemos la creación de derivados que repliquen el precio de un subyacente y su liquidación sea realizada por contratos inteligentes.

El trabajo estará estructurado de la siguiente manera:

En la primera parte del trabajo analizaremos la literatura previa relacionada, que nos proporcionará el marco teórico sobre el cual realizaremos nuestro desarrollo. Estos trabajos abordan los distintos modelos de brokerage, los *exchanges* descentralizados, los contratos por diferencia, los s*mart contracts* y sus aplicaciones y el marco regulatorio.

A continuación en el segundo capítulo brindaremos el contexto y los conceptos básicos con el fin de que el lector pueda interpretar con mayor precisión el desarrollo de los capítulos siguientes. En este capítulo también será analizado el ecosistema de aplicaciones financieras descentralizadas y las soluciones actuales al problema que abordamos.

En el tercer capítulo comenzaremos a plantear nuestro desarrollo desde el nivel del activo que proponemos. En primer lugar realizaremos la descripción teórica del mismo y luego señalaremos las similitudes y diferencias con activos existentes conocidos como contratos por diferencia (CFD).

En el siguiente capítulo continuaremos la descripción teórica de nuestro desarrollo en cuanto a la estructura del exchange. En este detallaremos sus características y funcionamiento así como también su integración con el activo introducido en el anterior capítulo.

Posteriormente, en "simulación del exchange", realizaremos una simulación del modelo propuesto mediante el lenguaje de programación Python. En este capítulo expondremos las variables y metodologías que proporcionan el funcionamiento de dicho modelo. Este motor de simulación nos permitirá analizar tanto cuantitativamente como visualmente las distintas métricas de funcionamiento y eficiencia.

El foco del quinto capítulo consiste en medir cuál sería la diferencia entre los precios operados dentro de nuestra plataforma y los del mercado subyacente. Realizaremos un análisis de los resultados obtenidos. Finalmente incluimos una conclusión donde pondremos en perspectiva los resultados y propondremos posibles mejoras para el modelo.

Capítulo 2. Literatura previa y conceptos

A continuación se presenta una revisión de la literatura disponible relacionada a los distintos temas y conceptos que serán utilizados en el desarrollo. En el proceso también proporcionaremos el marco teórico sobre los conceptos principales mencionados para facilitar la posterior comprensión del trabajo.

Contratos por diferencia o CFD

Un contrato por diferencia, es un derivado financiero cuyo valor está directamente ligado al de un activo subyacente, los cuales comúnmente son acciones, índices, commodities y *Forex*. Estos contratos tienen una estructura similar a la de un contrato a futuro en cuanto a sus flujos de fondos, ya que las diferencias en el precio del activo subyacente son intercambiadas diariamente entre las contrapartes dependiendo de su exposición de *long* o *short.*

El término *Long* es utilizado para describir cuando un inversor se beneficia de la suba del precio de un activo y *Short* cuando este se beneficia ante una disminución del precio. La principal diferencia con los futuros es que los CFD´s no tienen fecha de vencimiento, este intercambio de flujos de fondos continuará diariamente, otra diferencia es que el precio del subyacente es el precio *spot* del activo. El precio *spot* es el costo de adquirir el activo instantáneamente en el mercado.

Una importante característica de estos contratos es que permiten realizar operaciones con apalancamiento, es decir, operaciones en las que el valor nocional de la operación es mayor que el monto necesario para entrar en el contrato. El valor nocional de una posición está dado por el producto entre el precio unitario del activo subyacente y la cantidad especificada en el contrato.

Los flujos que se intercambian diariamente son las diferencias entre valor nocional de un día al otro, menos una comisión *overnight* del proveedor del contrato CFD por haber financiado la posición[1].

Para operar CFD´s las plataformas que ofrecen este tipo de contrato requieren una cuenta de margen que funciona de la misma manera que las cuentas de margen para cualquier derivado.

[1] Esta comisión overnight depende de la estructura de costos de cada exchange.

Esta es una cuenta en la que el inversor, previo a ejecutar una posición, deposita el llamado "Margen de mantenimiento" que está dado por el valor nocional y el nivel de apalancamiento. Periódicamente los flujos son acreditados o debitados en dicha cuenta, en caso de que el saldo baje de un cierto nivel el broker realizará una llamada de margen o *Margin call* en la que le comunica al dueño que debe depositar fondos o de lo contrario la posición puede ser liquidada de forma forzosa.

Este mecanismo está diseñado con el fin de mitigar el riesgo de crédito de la contraparte ya que asegura que ambas partes disponen de los fondos suficientes para afrontar sus obligaciones ante fluctuaciones en los precios. Cabe destacar que sigue existiendo riesgo de crédito ya que el funcionamiento del contrato depende de un proveedor o *broker*, quien también hace de custodia de las cuentas de margen. En el capítulo "Estructura y funcionamiento del *exchange*" hacemos mención a una posible solución a este problema.

El nivel de apalancamiento permitido debe estar basado en una encuesta de KYC *(Know Your Client)* en la que el cliente detalla su nivel educativo, su capital personal y su experiencia en los mercados. Este procedimiento de administración de riesgo fue promovido por las agencias reguladoras de CFDs como lo son la *Financial Services Authority (FSA)* del Reino Unido y la BaFin de Alemania.

Con este tipo de contratos es posible acceder a beneficios de gran variedad de activos sin la necesidad de comprarlos físicamente, y por este motivo, es posible comprar o vender el activo con un alto nivel de apalancamiento.

Los autores *Brown, Christine & Dark, Jonathan & Davis, Kevin* desarrollaron sobre los problemas existentes en el diseño del contrato para que se cumpla una relación de arbitraje entre el precio de los CFDs ofrecidos por la mayoría de los proveedores y el del activo subyacente.

> "*...They were initially introduced as OTC products by financial firms (referred to as CFD providers) which provide bid-ask quotes for traders. No settlement date is specified, and traders close out positions at dates of their discretion by offsetting trades at the price quoted by the CFD provider. That price is not contractually tied to the underlying stock price, creating potential basis risk for the trader (in addition to counterparty risk). CFD*

> *providers manage their counterparty risk by requiring traders to post margin accounts to which profits and losses are added… These practical features mean that the pure arbitrage based pricing argument, involving a fixed expiry date when CFD and physical prices converge, does not immediately hold."* (2010, p. 8).

El hecho de que, tal como los autores señalan, el precio no esté ligado contractualmente al activo subyacente es uno de los puntos sobre los cuales nuestro desarrollo de CFDs descentralizados representan potencialmente una importante innovación. Indagaremos más en esta temática en el capítulo 3.

Modelos de *Brokerage*

Los mercados financieros funcionan gracias a la existencia de *exchanges* que otorgan a inversores la posibilidad de comprar activos, en esencia son mercados ya sean físicos o electrónicos donde acciones, bonos o derivados pueden ser comercializados.

A lo largo de la historia los exchanges han evolucionado de la mano de la tecnología. En un principio las firmas que prestaban este servicio a sus clientes tenían empleados en los *trading floors* de los *exchanges* y ejecutaban órdenes utilizando un sistema de señas, alzando su mano si estaban dispuestos a subir su oferta. Si bien esta práctica no desapareció por completo, hoy en día gracias a los avances tecnológicos la mayor parte del comercio de activos financieros es llevada a cabo de forma electrónica.

Resulta importante aclarar que los contratos por diferencia, introducidos en la sección anterior, son derivados de naturaleza *Over The Counter* u OTC. Esto significa que no son operados en exchanges tradicionales tales como el *Chicago Board Of Trade*, y si bien pueden ser operados en plataformas que desde el punto de vista del usuario son indiferenciables, la mecánica detrás de la ejecución de las órdenes difiere.

En la industria de los proveedores de CFDs existen dos modelos ampliamente utilizados, el modelo de *brokerage* STP (*Straight-Through Processing*) y modelo de *Market Maker*. El primer modelo consiste en la conexión del *broker* a una red electrónica de puntas de compra y venta también conocidas como *liquidity pools*, en donde múltiples *market makers* compiten para llenar las órdenes disponibles. Un market maker es una firma o un individuo que provee

tanto un precio de compra como un precio de venta para un activo, estos forman una parte central de los mercados ya que garantizan liquidez, estos benefician por la diferencia entre estos dos precios que también es conocida como *bid-ask spread*. Los *brokers* que operan bajo este modelo no corren ningún riesgo de mercado ya que si bien son la contraparte de las operaciones de sus clientes, estas operaciones están íntegramente cubiertas por compras o ventas en el subyacente.

> "*...In practise, some CFD providers simply "pass through" CFD orders from customers as buy/short sell orders directly to the physical market on their own account to ensure a hedged position, thereby linking CFD prices quoted directly to the underlying.*" (2010, p.9)

Por otro lado existe el modelo de *Market Maker* en el que el *broker* también es la contraparte de las operaciones de los clientes, pero en este caso son ellos mismos quienes determinan los precios de compra y venta. Dado que estos proveedores toman operaciones en ambas direcciones, long y short, estos tienen una exposición direccional a cada activo. La suma de todas las posiciones de un mismo activo determina la exposición neta del broker, que también es conocida como delta. El *broker* puede elegir neutralizar esta exposición resultante a través de la compra o venta del subyacente pero no tiene la obligación de hacerlo.

Smart Contracts y sus aplicaciones

El modelo que propondremos en nuestro desarrollo hará uso de la tecnología de *smart contracts* o contratos inteligentes, ya que estos presentan numerosas ventajas para la implementación de contratos financieros. A continuación serán expuestos los conceptos básicos detrás del funcionamiento de estas tecnologías.

Una *blockchain* es una base de datos que lleva registro de todas las transacciones que se llevaron a cabo en la red. Esta base de datos es replicada y distribuida entre todos los participantes. La principal característica es que permite que participantes que no se conocen y sin necesidad de confiar entre sí, puedan realizar transacciones de forma segura sin la necesidad de una tercera parte. Estas transacciones se procesan en bloques y cada bloque es identificado por un *hash* criptográfico.

Cada bloque hace referencia al anterior, es por esto que recibe el nombre de cadena de bloques y una vez que el bloque es creado y agregado a la cadena no puede ser modificado o revertido, lo cual hace que se mantenga la integridad de las transacciones. Para procesar y verificar los bloques, nodos especiales llamados mineros, resuelven un algoritmo matemático y reciben una recompensa por invertir poder computacional y propagar los nuevos bloques a la red. El resto de los nodos verifican que los nuevos bloques sean verídicos y continúan construyendo sobre el de ser así.

Estas *blockchains*, pilares de la estructura de cualquier criptomoneda, tienen la característica de poder ejecutar líneas de código conocidas como contratos inteligentes. Toda *blockchain* tiene esta característica, pero a diferencia de la red de BTC que solo permite la ejecución de programas básicos, la red de Ethereum permite elevar la complejidad de los contratos y la interacción entre sí, abriendo las puertas a una infinidad de proyectos o protocolos financieros descentralizados.

Los *smart contracts*, por otro lado, son un código ejecutable que funciona sobre una blockchain y, entre otras, tienen la capacidad de recibir y enviar criptomonedas entre los usuarios participantes del contrato automáticamente. También pueden leer información de fuentes externas y utilizarla en el proceso de cálculo, a través de un mecanismo conocido como *oracles*. Esta característica podría resultar útil para crear derivados financieros ya que, por ejemplo, pueden tomar la cotización de mercado del activo para luego calcular y distribuir los cash flows entre los participantes.

Si bien esta tecnología potencialmente reduce la fricción de transacciones y el costo de operar, estos no serían eliminados por completo. La red de Ethereum al igual que el resto de las *blockchains* no es una simple plaza, sino un conjunto de nodos o computadoras, con capacidad limitada de correr bytes de información a la vez. Es por esto que cada nodo que corra el código emitido va a ser remunerado por su servicio, en forma de GAS Fee proporcionado en Ethereum.

Este costo, ó retorno que perciben los mineros depende del mismo código emitido, es decir, uno puede enviar un código a la red para que lo ejecuten, y ofrecer no pagar poco a los nodos, pero en este caso ningún nodo tendrá iniciativa de procesar el contrato y éste expirará sin haberse ejecutado. En definitiva, este costo es manipulable por nosotros, en casos donde

queremos darle prioridad a nuestros contratos ante los demás existentes en la red podríamos aumentar el GAS, y de lo contrario bajarlo.

"Compared to traditional contracts, smart contracts do not rely on a trusted third party to operate, resulting in low transaction costs." - (Maher Alharby, 2017).

En este trabajo, el autor lleva a cabo un análisis en profundidad sobre las ventajas y desventajas de los *smart contracts* o contratos inteligentes. Dado que es un trabajo publicado por un Phd en Ciencias de la Computación es de carácter más bien técnico, pero ilustra perfectamente dónde radica la importancia de este tipo de contratos para la industria financiera. El autor propone una interesante definición:

"*A smart contract is executable code that runs on the blockchain to facilitate, execute and enforce the terms of agreement between untrusted parties. It can be thought of as a system that releases digital assets to all or some of the involved parties once the pre-defined rules have been met*" (Alharby, 2017).

También señala que la red de Ethereum es superior a la de Bitcoin para la implementación de *smart contracts*, siendo esta que el lenguaje de programación de Bitcoin no cumple con la completitud de Turing, mientras que el de Ethereum si lo hace. La completitud de Turing significa que un lenguaje puede realizar las mismas operaciones que la célebre máquina de Turing, en este caso el lenguaje de Ethereum, a diferencia del de Bitcoin, permite la introducción de *loops* o bucles recursivos lo que hace al código altamente óptimo para contratos financieros.

Exchanges descentralizados (DEX)

Se conoce como *exchanges* descentralizados a los mercados digitales donde las contrapartes pueden intercambiar activos digitales haciendo uso de la tecnología de blockchain sin necesidad de una tercera parte que lleve a cabo la liquidación o custodia de los activos. El término es utilizado tanto para *exchanges* que funcionan íntegramente de forma descentralizada así como también para aquellos que hacen uso de la tecnología en cierta proporción, ya que estos pueden optar por realizar procedimientos o funcionalidades fuera de

la *blockchain* haciendo variar así su nivel de descentralización, y realizando un *Trade-off* entre velocidad y descentralización.

Las diferencias entre un exchange descentralizado y uno centralizado son numerosas. Una de las diferencias principales es la posesión de los activos, mientras que en los exchanges centralizados los clientes deben depositar sus activos en cuentas pertenecientes al exchange (conocidas como; *custodial accounts*), en un exchange descentralizado los fondos nunca están bajo custodia de este, y son transferidos de usuario a usuario o de usuario a contrato (*non-custodial*).

En cuanto a los costos de transacción, en un exchange descentralizado este factor está dado por el costo de interactuar con la blockchain sobre la cual opera, lo que depende del nivel de uso de la red, y el esquema de retribución a los validadores.

Finalmente otra diferencia a destacar es la anonimidad, cuando un cliente abre una cuenta en un exchange centralizado este debe proveer una gran cantidad de datos personales y fiscales ya que estos operan bajo la estricta supervisión de organismos gubernamentales, por otro lado para operar en un exchange descentralizado es posible hacerlo con un grado muy bajo de *KYC* y acceder a servicios que de otra forma serían inaccesibles, como por ejemplo derivados dado que en numerosas jurisdicciones los exchanges centralizados no tienen permitido ofrecer este servicio.

Los *exchanges* descentralizados a pesar de ser un campo relativamente nuevo dentro de las finanzas han sido tratados en numerosas publicaciones científicas, quizás por la potencial revolución que estos pueden implicar. (Lindsay, 2019) analiza la arquitectura de los *exchanges* descentralizados desde un punto de vista teórico. En esta publicación, la autora hace una interesante descomposición de estos en los siguientes componentes: la plataforma de *blockchain* y su implementación técnica, el mecanismo de descubrimiento de contraparte, el algoritmo de *Order-Matching* y el protocolo de *settlement* de las operaciones. A continuación detallaremos los conceptos detrás de estos componentes mencionados por la autora.

1. La infraestructura técnica y la *blockchain*.

Como se ha mencionado, en el espacio de las finanzas descentralizadas existen distintas *Blockchains* que poseen ciertas características técnicas. Sobre cada una de ellas existen distintas criptomonedas que comparten estas características. Un problema es el hecho de que las monedas que existen en distintos *blockchains* no son directamente intercambiables. En caso de que el exchange haga uso de contratos inteligentes estos deben funcionar sobre una única blockchain y podrán realizar transacciones en las monedas que compartan sus características técnicas.

2. El mecanismo de descubrimiento de contraparte

El mecanismo de descubrimiento de contraparte y de libro de órdenes es lo que habilita a compradores a descubrir vendedores quienes están dispuestos a ejecutar transacciones bajo términos mutuamente aceptados. Para realizar esta tarea los exchanges generalmente utilizan un libro de órdenes. Este en esencia es una lista de todos las órdenes de compra y de venta para un determinado activo a un determinado precio.

Este mecanismo puede funcionar fuera de la blockchain agrupando y cruzando todas las órdenes de un mismo activo con un mismo precio tal como sucede en exchanges tradicionales. Por otro lado, algunos exchanges descentralizados utilizan un libro de órdenes que funciona en la blockchain, en el que cada orden emitida va siendo validada por los nodos de la red. Esto trae como beneficio una mayor seguridad, en cuanto a la manipulación de las órdenes o su censura, pero también tiene la desventaja de perder velocidad ya que esta dependerá de la latencia de la red.

3. El algoritmo de *Order-Matching*

El algoritmo de cruzado de órdenes utiliza el libro de órdenes u otro sistema alternativo para determinar cuáles de estas órdenes serán ejecutadas. Este proceso puede suceder de forma automática, cuando en el libro de órdenes existen dos posiciones opuestas con un precio en común, tal como funciona en la gran mayoría de los *exchanges*.

Por otra parte, existe otra alternativa ampliamente utilizada en exchanges descentralizados, y es el mecanismo de llenado manual de órdenes. En este los usuarios publican sus ofertas especificando el precio límite y la cantidad que están dispuestos a intercambiar. La diferencia con el método automático es que las órdenes no son agregadas y aparecen individualmente en el libro de órdenes, para que otro usuario tome la contraparte si está de acuerdo con los

términos de intercambio. Estos son llamados *Makers* y *Takers* respectivamente, ya que unos crean liquidez en el mercado y los otros la toman o reducen.

4. Protocolo de *settlement.*

En cuanto al mecanismo de liquidación de las transacciones, es el único requisito fundamental para poder llamar a un exchange descentralizado, y es el trabajo principal que llevan a cabo los smart contracts.

La virtud de estos contratos es que gracias al hecho de que tienen la capacidad de recibir, almacenar y enviar valores, la liquidación de los contratos puede ser llevada a cabo de forma automática sin necesidad de una tercera parte.

Otra virtud que tienen es que al funcionar sobre una *Blockchain*, la información de todas las transacciones es almacenada y distribuida en la red[2], lo que previene que esta información pueda ser alterada en detrimento de alguna de las partes, y también ayuda a verificar que las transacciones se llevaron a cabo acorde a los términos deseados por los participantes. El trade-off de realizar el settlement mediante smart-contracts es una vez más es la latencia involucrada en confirmar de forma segura las transacciones en la red por descentralización y mayor seguridad.

Finalmente, la autora menciona los factores que representan una ventaja con respecto a modelos centralizados.

> "*Decentralized exchanges provide a number of important benefits, including (1) lower counterparty risk (i.e., no need to trust a centralized exchange to secure and manage private keys), (2) the potential for lower transaction fees, and (3) a more diverse array of trading pairs that can unlock access to riskier or less liquid cryptocurrencies. As demand for these features increases, decentralized exchange technology may witness tremendous growth in usage, development, and adoption within the next couple of years.*" (Lindsay, 2019).

[2] Por más detalle ver "funcionamiento del exchange".

Marco regulatorio

Uno de los desafíos y potenciales obstáculos son las regulaciones existentes con respecto a la tenencia y operaciones con activos digitales. La creciente popularidad y uso de activos digitales han atraído la atención de los entes reguladores alrededor del mundo. Un punto importante es que no todas las jurisdicciones han tomado el mismo enfoque, ya que los reguladores tienen por un lado la tarea de proteger a los inversores y evitar crisis en los mercados, y por otro permitir la innovación y el desarrollo de la industria financiera.

A continuación analizaremos una de las principales jurisdicciones, Estados Unidos. Este tiene normas muy estrictas con respecto a la operación de derivados financieros. La comercialización de Contratos por diferencia está prohibida en el país ya que estos violan una norma de la *Securities Exchange Commission (*SEC*)*, que establece que todo activo que replique el precio otro activo, es decir, un derivado, debe ser operado en un exchange nacional regulado, como por ejemplo *Chicago Board Options Exchange*.

De todas formas, al día de hoy EEUU no tiene en claro cómo tratar las criptomonedas en sí, por más que los derivados sean completamente prohibidos para sus residentes, el futuro de este mercado es incierto.

"*The Securities and Exchange Commission (SEC) typically views cryptocurrency as a security, while the Commodity Futures Trading Commission (CFTC) calls Bitcoin (BTCUSD) a commodity, and the Treasury calls it a currency*"

- Timothy Smith (Investopedia 21 septiembre 2021).

Por otro lado, China impone regulaciones incluso más estrictas a sus ciudadanos, ya que además de prohíbir la minería de blockchain desde mayo 2021, no permiten que ninguna institución o individuo comercialice criptomonedas dado que no las considera un activo lícito. De hecho, Binance el exchange más grande de criptomonedas y derivados por volumen diario, fué fundado en China, pero tuvo que cambiar su sede principal a las islas de Caimán en el 2017 debido al clima legal del país.

La mayoría de los países que forman parte de la Unión Europea consideran legales a las criptomonedas, pero sus regulaciones acerca de exchanges, derivados y normas impositivas

varían según el país. En septiembre de 2020 la *European Comission* propuso establecer normas claras para proteger al inversor con las regulaciones del *Markets in Crypto-Assets* (MiCA).

Algunos países imponen regulaciones menos estrictas y poco específicas, como Rusia quien no prohibió las criptomonedas, pero no permite el pago de bienes y servicios con ella.

En definitiva, sigue siendo un tema abierto a discusión por la inmadurez de la materia, y es importante estudiar el marco regulatorio dado que estas cuestiones están cambiando actualmente. Sabemos que las cripto y todos sus derivados no están completamente establecidos, y es por esto que se debe tratar la comercialización del producto con mucho cuidado y una estructura legal que valide el negocio.

Capítulo 3. Ecosistemas y plataformas existentes

- Historia de las finanzas descentralizadas (DeFi)

El campo de las finanzas descentralizadas comenzó con la creación de bitcoin en 2009. Se dice que fue el comienzo ya que este permitió la transferencia de valor de forma descentralizada. Pero su lenguaje tiene limitaciones que no permitían el desarrollo de funciones financieras más complejas como préstamos, créditos y derivados.

Luego en 2015 con el lanzamiento de Ethereum, marcaría el comienzo de las aplicaciones descentralizadas gracias a su lenguaje de programación Turing completo y el estándar ERC-20 para la emisión de monedas. Este hecho es conocido como el nacimiento de las *blockchains* de segunda generación.

Otro evento importante se dio en 2017 con el lanzamiento de Dai, la primera stablecoin cuyo valor está ligado 1 a 1 con el dólar y está colateralizado en Ether[3]. En 2017 también surgió EtherDelta, el primer exchange descentralizado, el cual hacía uso de un libro de órdenes que funcionaba íntegramente sobre la blockchain de Ethereum. Dado que EtherDelta operaba como un exchange no registrado y ofrecía sus servicios a ciudadanos Estadounidenses, la SEC prohibió que este continúe operando y multó al fundador. A pesar del nivel de descentralización, operar en jurisdicciones en las cuales existen regulaciones que no permiten este tipo de operaciones puede conllevar serias consecuencias legales.

El periodo entre 2017 y 2018 es conocido como la *ICO Mania*, o *Initial Coin Offering Manía*, ya que cientos de proyectos comenzaron a financiarse a través de la emisión de monedas. Durante este periodo surgieron algunos de los proyectos más importantes en este campo hasta el dia de hoy, como AAVE, lending and borrowing, Synthetix un exchange de derivatives, y 0X un *exchange* peer to peer de criptomonedas.

Además de ser un momento en el que surgieron numerosas plataformas innovadoras, este período fue clave para la industria ya que la tecnología de blockchain y las criptomonedas ganaron una inmensa popularidad y llegaron a impactar un público masivo, mientras que

[3] Token nativo de la red de Ethereum.

previamente era un mundo reservado para unos pocos entendidos de las ciencias de la computación.

En noviembre de 2018 fue publicado Uniswap, el que hoy en día es el exchange descentralizado más grande por volumen de transacciones, con un volumen promedio diario de 500 millones de dólares. Uniswap hizo uso del modelo User-to-contract, un modelo de intercambio en el cual los usuarios interactúan directamente con los smart contracts para realizar transacciones. De esta forma es posible llevar a cabo intercambios de monedas sin una contraparte directa.

En el campo de los derivados actualmente no existe ningún exchange descentralizado que permita ganar exposición a activos tradicionales. Existe una plataforma llamada Synthethix, la cual ofrece activos sintéticos de forma descentralizada, pero actualmente solo ofrecen criptomonedas y fiat como subyacente.

- Tokenized Stocks

Para una compañía, un banco o individuo que posea criptomonedas en su cartera, es tanto costoso como agotador buscar dónde almacenar sus activos digitales y cómo liquidarlos en caso de querer convertir esa proporción de su cartera a *equity*, *commodities*, *real estate* ó a efectivo. Este proceso no solo lleva tiempo sino conocimiento del mercado de *wallets* y *exchanges* de cripto, lo que requiere tener todos sus valores en una plataforma separada, aislada y muchas veces fuera del alcance de los demás valores, sea debido a regulaciones como costos de transferencias y transacción.

Durante nuestra investigación preliminar para la confección de este trabajo, a principios de 2021, se anunció que uno de las mayores *exchanges* (Centralizados) de criptomonedas, Binance, estaba trabajando en una solución para ofrecer exposición a acciones utilizando criptos.

Finalmente, las *Stock Tokens* fueron lanzadas por la plataforma Binance el 12 de abril del 2021 como forma innovadora de exponerse a los rendimientos de las acciones de empresas como Apple, Tesla y Microsoft. Aun así, no fué este el primer *exchange* en lanzar este tipo de

activo. El 29 de octubre del 2020 FTX, exchange de derivados de cripto, lanzó sus propias acciones *tokenizadas*.

Al indagar en el funcionamiento de estas propuestas descubrimos que ambas plataformas utilizan la misma metodología para ofrecer el servicio. La reciente aparición de la oferta de este servicio por parte de importantes jugadores en la industria demuestra que es un tema de suma importancia en la actualidad y es la vanguardia de la industria.

¿Cómo funcionan las acciones *tokenizadas* de FTX y Binance?
Estos exchanges permiten que los usuarios adquieran una criptomoneda que replica el valor de acciones o índices. De la misma manera que las *stablecoins* que replican el valor del dólar estadounidense, estas están colaterizadas por el subyacente del cual replican el precio. La colateralización es llevada a cabo gracias a un acuerdo comercial entre estos *exchanges* y CM Equity, una institución financiera alemana regulada. Cuando un usuario de FTX desea comprar una *stock token*, este emite la orden en la plataforma de FTX, y la orden es pasada a CM Equity, quien deberá vender en el mercado la posición respectiva a las del libro de órdenes de la plataforma. Esto garantiza que el exchange no corra riesgo ante movimientos súbitos del mercado, ya que cuando un usuario desee vender una stock token luego de una suba para realizar una ganancia, la plataforma dispondrá de la ganancia generada por el subyacente.

En julio de 2021, Binance anunció que a partir de octubre ya no podrán ser operados stock tokens en su plataforma. Esto sucedió a raíz de una disputa con distintos entes reguladores. El Reino Unido (FCA) , Alemania (BaFin), Hong Kong (FSC) consideraban que este tipo de activo no podría ser comercializado en sus respectivas jurisdicciones por no estar regulado por una plataforma licenciada y por falta de un prospecto de inversión.

Un prospecto de inversión es un documento formal que debe ser presentado ante las autoridades regulatorias que provee los detalles de una propuesta de inversión cuando ésta es ofrecida al público con el objetivo de que los potenciales clientes puedan tomar decisiones más informadas. Este documento debe ser presentado por compañías cuando desean emitir acciones o bonos al mercado, y dado que las que las stock tokens representan acciones que ya se encuentran operando en otros mercados cae en una zona gris en cuanto a su regulación.

Currrency.com también utiliza este método para comercializar acciones "tokenizadas" y se encargan por su propia cuenta de contar con la estrategia de hedging al ser la contraparte de sus usuarios:

"Because you may deal with Currency.com as the counterparty in every transaction, you will have direct exposure to us in relation to each of your transactions and are reliant on our ability to meet our obligations to you under the terms of each transaction. This risk is described as a 'counterparty risk'."[4]

[4] Currency.com WhitePaper; https://currency.com/static/Currencycom_WP_EN_14102020.pdf

Capítulo 4. Decentralized CFD

En este capítulo detallaremos las características y el funcionamiento del activo que es el componente principal de nuestro desarrollo, de ahora en adelante nos referiremos al mismo como Decentralized CFD o DCFD.

Como el nombre lo sugiere, este producto busca replicar los flujos de fondos de un contrato CFD tradicional, los cuales dependen del desempeño de un activo subyacente, con la diferencia de que la liquidación de los contratos suceda de forma descentralizada, haciendo uso de los beneficios de los contratos inteligentes.

Para poder iniciar la operación, los agentes deberán depositar un margen nominado en alguna criptomoneda estable como el USDT[5].

- Cálculo del margen

El margen necesario para abrir una posición es calculado de la siguiente manera.

$$Margen\ Inicial\ =\ N\ *\ P\ *\ L$$

Donde:

N : Cantidad unitaria de subyacente

P : Precio unitario del activo subyacente en USDT

L : Fracción de apalancamiento

Por ejemplo, el margen requerido para una posición tanto *long* como *short* en un Decentralized CFD sobre 50 acciones de Apple Inc. con un apalancamiento de 10 a 1 sería el siguiente.

$$50\ *\ \$160\ *\ 1/10\ =\ \$800$$

Esta es la cantidad de fondos disponibles, en moneda estable USDT, que el usuario deberá tener disponible en la cuenta de margen para poder enviar la orden al libro de órdenes del *exchange.*

[5] A partir de este momento, usaremos al Tether (USDT) como criptomoneda estable del modelo.

Hacer uso de esta moneda supone una gran ventaja ya que permite que el margen sea mucho más estable en relación al resto de las criptomonedas y de esta forma evitar que las posiciones sean liquidadas por fluctuaciones del precio de la moneda del margen contra el dólar. Dado que la mayoría de los activos subyacentes que ofreceremos tienen una cotización de referencia contra el dólar americano, la capacidad de que el margen de las posiciones esté en dicha moneda es fundamental.

Esta implementación permite dos características, una es que los contratos sean no colateralizados, eliminando así la necesidad de una tercera parte que haga de custodia de los instrumentos subyacentes de los contratos, como es el caso de las stock tokens de Binance y FTX. Y la segunda característica que permite es que los agentes del mercado pueden tomar posiciones apalancadas, es decir, por un valor nocional superior al capital del que disponen. De todas formas, es importante remarcar que el apalancamiento también aumenta considerablemente el riesgo de contraparte y este estará limitado a un 10:1. Para controlar que los usuarios no pierdan más fondos de los que poseen y garantizar que existan fondos disponibles en la plataforma para transferir a los usuarios con posiciones ganadoras, implementamos un mecanismo de gestión de riesgo.

- Mecanismo de gestión de riesgo

Periódicamente junto con el cálculo de las ganancias y pérdidas de las posiciones del usuario será calculado un ratio de riesgo. Este ratio de riesgo en el momento t está dado por la siguiente fórmula.

$$Ratio\ de\ riesgo_t = \frac{Margen_t}{Margen\ inicial} = \frac{N * P_i * L}{N * P_0 * L}$$

En el momento 0, es decir al momento que el usuario abre la posición, este ratio será 1, y este tiene una relación entre este y el precio depende de la dirección de la operación, en caso de una operación Long, el ratio decrece a medida que el precio baja y viceversa para una posición *Short*.

Las condiciones que se fijaron en el modelo (ver código de Python en el anexo) para hacer margin calls y liquidaciones automáticas también dependen la respectiva volatilidad del subyacente. Debajo las dos condiciones de riesgo:

$$Margin\ call\ =\ Ratio\ de\ riesgo_{t}\ <\ 50\%\ +\ Volatilidad\ anualizada\ /10$$

$$Liquidación\ =\ Ratio\ de\ riesgo_{t}\ <\ 20\%\ +\ Volatilidad\ anualizada\ /10$$

En donde se toma en cuenta la volatilidad anual del subyacente para exigirles más margen a las posiciones sobre subyacentes más riesgosos. Aunque, estas barreras podrían bien ser arbitrarias o depender de la condición que desea ajustar el administrador del *exchange*. Por este motivo introducimos una constante que divide la volatilidad; como una medida extra de seguridad que depende del activo en cuestión. En caso de querer subir la exigencia de margen aún más a cierto activo, ya sea por una característica sectorial, fundamental, ó por un evento, el denominador podrá ser ajustado a una constante más chica, y viceversa.

- Flujo de fondos del contrato

Como fue mencionado, los participantes del mercado tendrán la posibilidad de iniciar operaciones tanto *Long* como *Short*, y se especificarán en el contrato junto a las características de cada participante de la operación, cada uno de estos deberá disponer de su respectivo margen para poder iniciar la operación. Este margen será sujeto a un *mark to market*[6] y servirá para eventualmente cubrir las ganancias de la contraparte.

Si agente "A" abre una posición long por un valor nominal equivalente a $100 dólares y agente "B" es su contraparte por esta misma cantidad de valor nominal, cuando el activo incrementa x% en valor, se le acreditará x% de $100 en cuenta de A que provendrá de la cuenta de margen de B.

En la tabla a continuación podemos observar los flujos de fondos de una hipotética posición sobre 10 acciones de Apple Inc con un apalancamiento de 5:1.

[6] *Mark-to-market* es un método para medir el valor de las tenencias de un portafolio de inversiones en la que el valor de este es ajustado periódicamente basado en el valor de mercado de los activos.

	Precio del subyacente	Cantidad	Valor nocional	Margen usuario *Long*	Margen usuario *Short*
t = 0	$160	10	$1600	$320	$320
t = 1	$165	10	$1650	$370	$270
t = 2	$150	10	$1500	$220	$420

En el ejemplo es una simplificación, ya que ambos usuarios entraron en un contrato por la misma cantidad de subyacentes y con el mismo nivel de apalancamiento. En la práctica los contratos pueden tener varias contrapartes para cubrir los $1600 de posición *Long* del agente A, ya que no necesariamente existirá un solo agente que esté dispuesto a iniciar una posición *Short* en este activo por exactamente dicho monto. En caso de que uno de los agentes desee cerrar su posición, esta será enviada automáticamente al libro de órdenes del exchange para que otra contraparte asuma su posición.

Capítulo 5. Funcionamiento del exchange

En este capítulo detallaremos los componentes del exchange donde serán operados los Decentralized CFDs. Esto será llevado a cabo analizando los cuatro componentes de un exchange descentralizado propuestos en Lin, L. X., 2019.

1. Infraestructura técnica y *blockchain*.

Dado que el exchange descentralizado operará con *smart contracts* estos deben funcionar sobre una blockchain específica. Esta decisión tendrá un gran impacto en la facilidad de uso de la plataforma pero principalmente en su costo y eficiencia.

La plataforma no será íntegramente descentralizada, en el sentido que, todos los registros, de cada transacción e interacción de los usuarios con el exchange, incluyendo el proceso de calcular el P&L y los márgenes de sus posiciones, será almacenado internamente ó mejor dicho, centralizadamente en los servidores del *exchange*. Pero la custodia de los márgenes de las posiciones, y el repago ó cobro de cada parte involucrada en un DCFD será ejecutada vía un protocolo descentralizado.

Una de las opciones estudiadas es construir los contratos sobre la red de Ethereum y hacer uso de la *token* USDT. Esto tendría como ventaja la facilidad de uso para los usuarios ya que es el estándar más adoptado, así como también la confiabilidad de la red ya está en funcionamiento desde 2015. Por otro lado, la desventaja es el costo por transacción en esta red. El hecho de que sea la red más usada y que el mecanismo de verificación de los bloques sea *proof of work*, que requiere mucho poder computacional y por ende eléctrico, haciendo que el costo sea considerable.

Otra de las opciones es hacer uso de alguna blockchain de las llamadas tercera generación como Solana o Cardano, que además de tener la capacidad de procesar *smart contracts,* brindan una mayor capacidad de procesamiento de transacciones por segundo, así como también costos de transacción mucho menores.

Estructura de costos de distintas *black chains*:

	Ethereum	Solana	Cardano	Binance Chain
Transacciones por segundo	15	> 50.000	200-250	100
Costo promedio por transacción[7]	$15 Usd	$0,0015 Usd	$0,25 Usd	$0,5 Usd
Número de validadores	> 200.000	1.150	-	21
Método de validación	*Proof of work*	*Proof of history*	*Proof of stake*	*Proof of stake*

Por más que estas, ó futuras tecnologías blockchain compatibles con contratos inteligentes, tengan un bajo costo, surge un problema de escalabilidad si cada posición que toma un inversor es validada y liquidada a través de un *smart contracts* individuales.

Es decir, una plataforma que ejecute transacciones entre billeteras de sus usuarios vía *smart contracts* cada vez que un usuario decida tomar una posición en el mercado sería muy costoso y dejaría de lado el inversor *retail*, y este no es nuestro objetivo.

Por esta razón hemos analizado alternativas para estructurar el proyecto, estas estructuras surgieron de a partir de las necesidades del producto que buscamos lanzar y las soluciones existentes, en el mundo crypto, DeFi, y de *smart contracts* es uno en desarrollo.

Decentralized Autonomous Organization (DAO).

Una DAO es una organización, o sociedad, estructurada como conjunto de *smart contracts*, con el fin de establecer reglas de la sociedad. La idea detrás de estas organizaciones es poder generar un ambiente descentralizado, sin que sus usuarios tengan que, en conjunto, interactuar con una blockchain, lo que implica una reducción de costos sin necesariamente sacrificar el grado de seguridad.

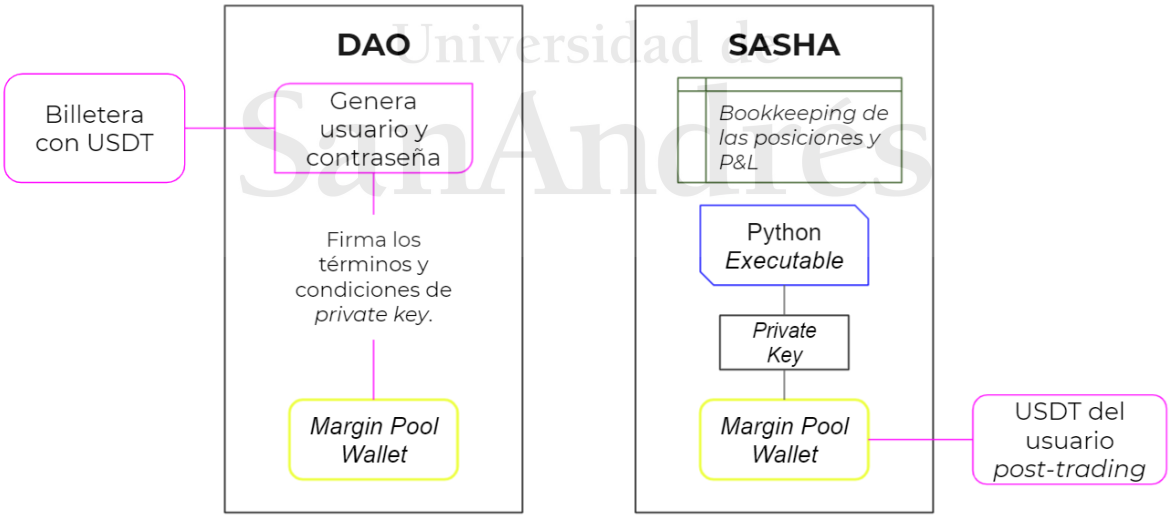
Generamos una organización de tipo DAO, con el único propósito de gestionar el acceso a una billetera virtual de la plataforma de manera descentralizada, con un conjunto de términos y condiciones que permitan únicamente a un código previamente escrito acceder a ella.

[7] El costo de transacción puede variar significativamente, este número representa el promedio del costo de transacción de todas las transacciones del periodo 2021/10/01 - 2021/11/01.

La DAO tiene capacidad de parametrizar reglas de transferencias tal como las de un banco, conocidas como *Automated Teller Machine* (ó ATM). Y es por esto que, la DAO podría habilitar un entorno donde ninguna única entidad, ni los creadores de la sociedad, tengan control sobre los fondos de la organización. Cuando un usuario decida querer retirar sus fondos de la plataforma, será un sistema predeterminado y abierto al público el que verificará la transacción y retirará los fondos acordes.

En la actualidad esta es la función que cumple un cajero automático ó ATM, con el fin de validar el código PIN de una tarjeta de débito, y acceder a los fondos del banco. En nuestro caso, sin que la administración / dueños del banco tengan la capacidad de acceder a los fondos de sus clientes.

De esta manera, no tendremos que lanzar costosos *smart contracts* por cada operación del usuario. En vez, la plataforma guardará la información de cada operación y posiciones de los usuarios, y los *smart contracts* de la DAO descentralizan el proceso de *clearing* cuando estos se quieran retirar.[8]



Una vez que el usuario haya fondeado USDT en la billetera de margen de la DAO, podrá acceder a la plataforma de Sasha y operar tranquilamente sabiendo que su historial de órdenes y posiciones serán almacenadas en una base de datos interna.

[8] Por más información acerca de las DAO's y como funcionan, es recomendable leer acerca del caso de "The DAO" un fondo de inversión que gracias a este protocolo logró demostrar una resiliencia a nivel seguridad muy importante: https://www.gemini.com/cryptopedia/the-dao-hack-makerdao

Nadie podrá acceder a los fondos de la billetera de la DAO de forma arbitraria, las condiciones que rigen el funcionamiento de la DAO habilitan únicamente a un codigo de Python que tenga la private key de esta billetera

Una estructura como la de la DAO gestionará sus usuarios y la billetera virtual, pero un punto de desarrollo futuro será el del manejo de las private keys que habilitan retirar fondos de su billetera. Como creadores de la organización queremos que este acceso sea únicamente otorgado a un código de Python y de alguna manera securitizado con un sistema que garantice el resguardo de estas llaves.

Un ejemplo que podría resolver esta cuestión, dado una reunión que tuvimos con los representantes de Fireblocks[9], es usar una plataforma como esta que brinde servicios de MPC (*multi-party computation*) en donde las llaves que acceden a la billetera virtual de la plataforma sea dividida en trés partes (o *shards*). Por ejemplo, una parte la tendrá el Python Executable que se active dado la condición de que un usuario se quiera retirar, y las otras dos partes distribuidas en servicios de informática en la nube[10].

Y, la última característica de esta estructura que podría ser optimizada, ó descentralizada de mejor manera para la seguridad de los usuarios, es la descentralización del *bookkeeping* de sus posiciones y órdenes abiertas. Esto, por el momento, lo podríamos solucionar incorporando la información de cada usuario en una plataforma como la de *filecoin*.

Los costos de almacenamiento de *filecoin* son muy competitivos, dado el sistema de resguardo de información que proponen. Básicamente un mercado abierto donde cualquier agente con capacidad de memoria en su computadora o servidor puede vender el espacio a un costo muy competitivo.

"*There's a ton of storage in the world that's not getting used. Think of it like Airbnb. You had people with rooms that weren't being used; Airbnb built a marketplace for them*."

-Juan Benet, CEO of *Filecoin*.

[9] Una compañía de SaaS dedicada al manejo de private keys de las billeteras virtuales de sus clientes.
[10] Como bien podrían ser Microsoft Azure ó IBM Cloud.

Además de que, en nuestro modelo actual (ver archivos csv que crea la plataforma) son archivos de texto muy livianos y por ende poco caros de almacenar, solamente contienen información acerca de las transacciones del usuario como parámetro que toma el *smart contract* para validar el patrimonio de un usuario antes de llevar a cabo un *settlement*.

Todas estas soluciones se puedan ajustar en un futuro con soluciones y tecnología más eficiente, y que se pueda cambiar la combinación de estas soluciones para proponer un esquema más robusto, al día de hoy, este esquema es robusto, sencillo y *cost-efficient*.

Además, logramos nuestro objetivo de descentralizar los contratos CFD's sin la costosa necesidad de *smart contracts* para cada transacción.

2. Mecanismo de descubrimiento de contraparte

En este segmento se estudia el tipo de libro de órdenes que utilizará nuestra plataforma, y su nivel de descentralización. Hemos visto el modelo centralizado de los exchanges tradicionales, también el modelo completamente descentralizado como el EtherDelta, y soluciones alternativas como la de *User-to-contract* del protocolo Uniswap.

En lo que respecta a este mecanismo para nuestra plataforma decidimos tomar un enfoque tradicional; un libro de órdenes centralizado, que funcione fuera de la blockchain. Este libro de órdenes, agrupa por un lado las órdenes de compra y las órdenes de venta ordenadas principalmente por precio y secundariamente por el orden cronológico en el que llegan al libro.

Basándonos en el supuesto de que el exchange dispondría de la suficiente liquidez en ambos lados, este sistema funciona perfectamente para cumplir el objetivo de la plataforma. Eventualmente se podrían registrar las transacciones de manera descentralizada, resguardando esta información en la blockchain.

3. Algoritmo de *Order-Matching*

El algoritmo de cruzado de órdenes que decidimos utilizar es *First in first out (FIFO)* el cual prioriza el orden en el cual las órdenes llegan al exchange. En este caso las órdenes a precio más competitivo tienen mayor prioridad y luego se ejecutan por órden de llegada.

En la simulación de nuestra plataforma la cual será expuesta en detalle en el capítulo siguiente llevamos a cabo una implementación de este algoritmo. El funcionamiento técnico del mismo también se encuentra en el código del anexo, aunque a continuación se encuentra un resumen de su funcionamiento.

Una vez que las órdenes fueron emitidas por los usuarios y ordenadas por precio en sus respectivos libros, se comprueba que el mayor precio al que alguien está dispuesto a comprar sea mayor o igual que el máximo precio al que alguien está dispuesto a vender[11].

En caso de que dos órdenes cumplan la condición previamente mencionada, el precio de la operación es el promedio entre el precio de compra y el de venta. En el siguiente ejemplo demostraremos como nuestro algoritmo cruza las órdenes iterativamente.

Estado inicial :
Libro de órdenes de compra

Precio	Cantidad
148	10
150	20

Estado intermedio:
Después de ejecutar 10 a $150

Precio	Cantidad
148	10
150	10

Estado final:
Después de ejecutar 10 a $149,5

Precio	Cantidad
148	10

Libro de órdenes de venta

Precio	Cantidad
150	10
149	20

Precio	Cantidad
149	20

Precio	Cantidad
149	10

[11] Esto es equivalente a invertir la condición, es decir, que el precio al que alguien esté dispuesto a vender sea menor o igual al que alguien esté dispuesto a comprar.

En una primera instancia, el libro ejecuta 10 contratos a $150, luego se ejecutan 10 unidades a $149,5 ya que el comprador estaba dispuesto a pagar 150 y el vendedor a recibir 149, por ende el precio realizado es (150+149)/2. Finalmente, quedan dos órdenes en el libro sin ejecutar dado que los precios de compra y venta no coinciden, así, el último precio operado del activo en este libro de órdenes queda en $149,5.

4. Protocolo de *settlement*

La liquidación o *settlement* de las posiciones será llevada a cabo mediante *smart contracts* previamente descritos, dado la estructura del exchange propuesto, todas las posiciones de un usuario, que en caso de hacer uso de *Filecoin* estarían registradas en la blockchain, serán utilizadas para verificar que los montos que quiera retirar coincidan con sus P&L históricos.

A partir de este momento, el contrato inteligente será activado y, dado que este tiene acceso a las private keys de la billetera que almacena los fondos del margen de los usuarios, se encargará de hacer la transferencia de USDT desde esa billetera a la personal del usuario. Con este modelo, el usuario termina incurriendo en las comisiones de *Gas Fee* únicamente al entrar y salir del exchange, y a las comisiones de trading internas.

Una representación gráfica del proceso de *exit* del usuario se puede apreciar en el anexo.

Capítulo 6. Simulación de la plataforma

Con el fin de exponer gráficamente el funcionamiento de la plataforma desarrollamos una implementación del modelo utilizando el lenguaje de programación Python y una librería llamada Streamlit que nos permite crear un Front-end[12]. Esta simulación es la representación de la plataforma desde el punto de vista de los usuarios.

Ver interfaz de la plataforma en anexo 1.1

- Funcionalidad

En cuanto a funcionalidad incorporamos la posibilidad de ejecutar órdenes en tiempo real, a precio de mercado[13], haciendo uso del panel de trading. Una vez seleccionados los parámetros de la operación, los cuales son el activo subyacente, el nivel de apalancamiento y la cantidad unitaria, al hacer click en el botón "Long" o "Short" la simulación desplegará una serie de gráficos y tablas con información sobre la evolución del contrato.
Ver anexo 1.2.

Esta simulación incorpora también nuestro mecanismo de gestión de riesgo, en caso de que el nivel del margen necesario baje del nivel preestablecido, está cerrará automáticamente la posición de la contraparte afectada y enviará la orden al libro de órdenes para que alguien más cubra la posición.

- Modelado del precio del subyacente

Para modelar el precio del subyacente utilizamos un proceso estocástico conocido como *Geometric Brownian Motion*. Este proceso también es utilizado para generar posibles senderos del precio de un activo en simulaciones de Montecarlo.

$$S_t = S_{t-1}(\mu\,\Delta t \;+\; \sigma\sqrt{\Delta t}\,e)$$

[12] Front-end es la parte de un software que transforma las computaciones realizadas por el Back-end en una interfaz con la que el usuario puede interactuar.
[13] En la visualización que creamos, no existe la posibilidad de emitir *limit-orders,* aunque en el libro de órdenes que modelamos si podemos encontrar órdenes de tipo límite.

Donde:

$$S = Precio\ del\ activo.$$
$$\Delta t = Incremento\ temporal.$$
$$\mu = Retorno\ esperado\ del\ activo.$$
$$\sigma = Desvío\ estándar\ de\ los\ retornos\ del\ activo.$$
$$e = Variable\ aleatoria.$$

Esta tiene un componente determinístico que incrementa el precio según el parámetro μ, proporcionalmente al incremento de tiempo Δt. Por otro lado posee un componente de difusión que está dado por la variable aleatoria e que sigue una distribución normal con media 0 y desvío estándar 1, multiplicado por la volatilidad escalada por el parámetro temporal. A continuación podemos ver nuestra implementación de dicha función en Python.
Ver código en anexo; función "brownian_motion".

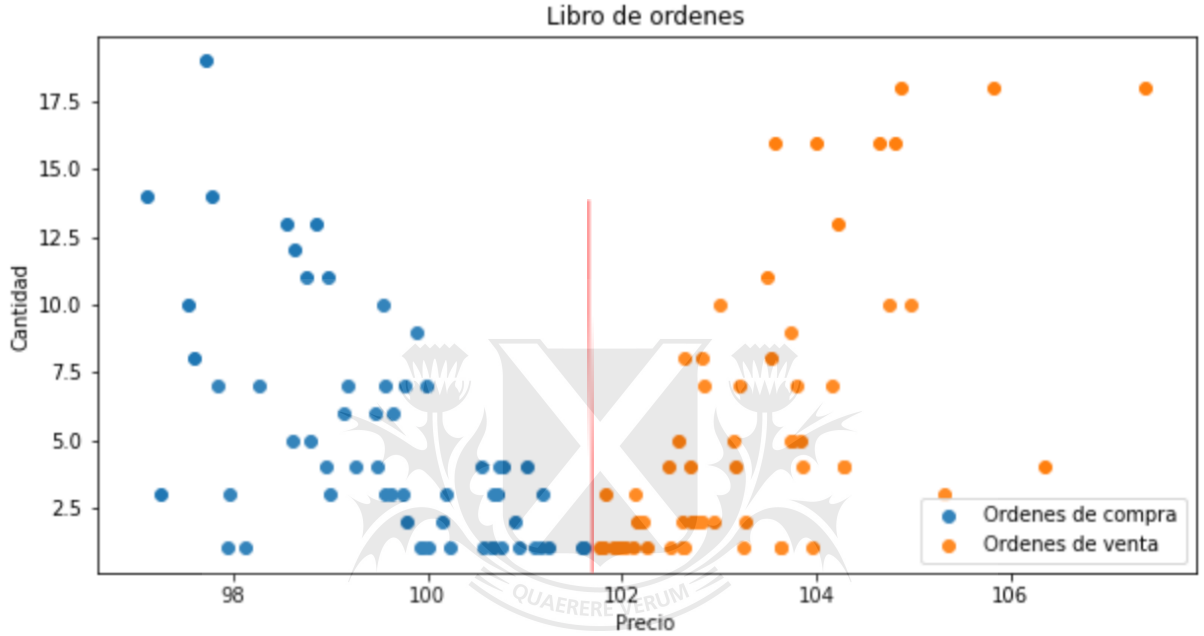
Esta función produce series de precios que se ajustan a los objetivos de la demostración. Ejecutar esta función nos permite generar precios en tiempo real y simular el mercado del subyacente en funcionamiento, sobre el cual podemos simular operaciones y calcular la evolución de estas con el fin último de demostrar el funcionamiento de la plataforma.

- Modelado del comportamiento de los agentes

Con el objetivo de simular liquidez en la plataforma diseñamos la siguiente función que modela el comportamiento de los agentes en lo que respecta al flujo de órdenes. Esta implementación busca replicar el flujo de órdenes de una forma natural y similar al comportamiento observado en los mercados. Para entender este comportamiento es importante diferenciar entre cantidad de órdenes en un nivel de precios y cantidad unitaria de activos a ser intercambiados a cierto precio.

"*We find that the flow of order placement is concentrated at the inside of the bid-ask spread. A large fraction of the order placements improves upon the best bid or ask quote. [...] Despite the concentration of order activity at the quote, the quantities in the order book are not concentrated near the quote, this is a consequence of trading activity consuming liquidity at the quotes.*" Biais, B., Hillion, P., & Spatt, C. (1995).

La cantidad de órdenes incrementa a medida que nos acercamos al precio de mercado, es decir, el precio al que se realizó la última operación. Y si bien la cantidad de órdenes aumenta, la profundidad[14] del mercado decrece cuando nos acercamos a precio de mercado, ya que cerca de un equilibrio, habrá una mayor cantidad de agentes dispuestos a comprar o vender a precios más convenientes. A continuación podemos ver un ejemplo del estado del libro de órdenes producido por nuestra función.



Esto demuestra que nuestra función modela el comportamiento de una forma similar a la que describen Biais, B., Hillion, P., & Spatt, C. (1995) en su investigación.
Ver anexo 1.2 (Ver código; "create_orders").

El producto, o la salida, de esta función esta es una serie de órdenes compuestas por un precio, una cantidad y la dirección de la billetera virtual del usuario que emitió la orden.

Dado que asumir una cantidad de órdenes arbitrarias hubiera producido resultados sesgados, la cantidad de órdenes notada como n, que son emitidas también está dada por un variable aleatoria uniformemente distribuida en el intervalo [1,b]. La elección del fin del intervalo de esta distribución es sumamente importante dado que va a definir el nivel de profundidad y liquidez del libro de órdenes del mercado. Esto implica que cada vez que, periódicamente al ser actualizado el precio de mercado externo, serán emitidas entre 1 y b órdenes, con

[14] La profundidad de un libro de órdenes hace referencia a la cantidad de activos disponibles a operar en cada nivel de precio. Un libro profundo tendrá una alta cantidad de unidades para cada nivel de precio, tanto en el lado de compra como de venta.

probabilidad 1/b. En el capítulo siguiente están expuestos los resultados para distintos valores del fin del intervalo, b.

$$n = U[1, b]$$

Cada una de estas órdenes que genera la función, dado nuestros supuestos, tendrán un precio y una cantidad. La forma en la que estas variables fueron modeladas se encuentran a continuación.

Los precios comienzan desde el precio de mercado, y se alejan de este en incrementos de 0,5%.

$$Precio\ de\ compra_i = Precio\ de\ mercado * (1 + \frac{i}{200}),\ donde\ i\colon\ 0, 1,\ldots, n$$

$$Precio\ de\ venta_i = Precio\ de\ mercado * (1 - \frac{i}{200}),\ donde\ i\colon\ 0, 1,\ldots, n$$

Con el fin de simular el tamaño de estas órdenes, lo modelamos como variables aleatorias que siguen una distribución uniforme donde el fin del intervalo es una función cuadrática de n, la cantidad de órdenes. La función cuadrática genera una curvatura que reproduce un libro de órdenes mejor simulado.

$$T \sim U[1, i^2],\ donde\ i\colon 1,\ldots, n{+}1$$

Donde T representa el tamaño de las órdenes

En nuestra simulación de la plataforma las órdenes no solo son publicadas en el libro de órdenes, sino que también implementamos el algoritmo que cruza órdenes, *First in First out.* Implementar este algoritmo nos permite obtener información acerca del comportamiento de las ejecuciones y los precios de nuestro libro de órdenes. Esto nos permitirá obtener métricas útiles como el diferencial entre los posibles precios de nuestro mercado con los del activo subyacente de su *exchange* original.

Capítulo 7. Resultados

Como resultado principal del presente trabajo buscamos comparar cuáles serían los precios de operar en nuestra plataforma dados los supuestos teóricos que planteamos, contra los precios del activo subyacente generados por el movimiento browniano.

Para cuantificar el diferencial entre el precio que genera nuestro libro de órdenes contra el del activo subyacente decidimos llevar a cabo un análisis de este diferencial a través de una simulación de Monte Carlo, este método es empleado dada la cantidad de variables aleatorias que el modelo posee. Hemos estresado las variables de nuestro modelo para corroborar el funcionamiento y la resiliencia del modelo propuesto.

Para obtener un tamaño de muestra que nos permita obtener conclusiones robustas computamos el promedio de 1000 posibles escenarios. En cada uno de estos 1000 escenarios computamos un serie de 30 precios diarios, creamos órdenes de compra y de venta, ejecutamos el algoritmo de *Order matching* y finalmente computamos el diferencial promedio entre el precio interno operado y el precio del mercado para cada operación ejecutada dentro del exchange.

En términos matemáticos, el diferencial en el momento t, de la iteración i está dado por la siguiente fórmula:

$$Diferencial_{i,t} = \frac{Precio\ operado\ en\ el\ exchange_{i,t}}{Precio\ de\ mercado\ externo_{i,t}} - 1$$

El cálculo que efectivamente realiza la simulación de Monte Carlo es el siguiente

$$\overline{Diferencial} = \sum_{i=0}^{1000} \left(\frac{\sum_{t=0}^{30} Diferencial_{i,t}}{30} \right) * \frac{1}{1000}$$

En la tabla a continuación podemos ver los resultados obtenidos a partir de distintos parámetros en la distribución de la cantidad de órdenes emitidas cada día y la volatilidad del subyacente. La cantidad de órdenes es modificada a través del fin del intervalo de la

distribución uniforme $U[1, b]$ y la volatilidad anual se ajusta con el parámetro *sigma* del movimiento browniano.

Diferencial	Volatilidad: 10%	Volatilidad: 20%	Volatilidad: 50%
Cantidad de órdenes menor. [1,2]	Percentil 90%: 0,445% Media: 0,0002% Percentil 10%: -0,933% Operaciones totales: 39.911	Percentil 90%: 0,788% Media: 0,017% Percentil 10%: -1,708% Operaciones totales: 43.801	Percentil 90%: 1,765% Media: 0,1039% Percentil 10%: -3,962% Operaciones totales: 46.417
Cantidad de órdenes intermedia. [1,5]	Percentil 90%: 0,885% Media: 0,021% Percentil 10%: -1,665% Operaciones totales: 60.738	Percentil 90%: 1,451% Media: 0,083% Percentil 10%: -3,004% Operaciones totales: 81.820	Percentil 90%: 2,709% Media: 0,1756% Percentil 10%: -5,669% Operaciones totales: 103.234
Cantidad de órdenes mayor. [1,10]	Percentil 90%: 1,031% Media: 0,061% Percentil 10%: -2.02% Operaciones totales: 71.646	Percentil 90%: 2,053% Media: 0,107% Percentil 10%: -3,202% Operaciones Totales: 118.807	Percentil 90%: 3,786% Media: 0,154% Percentil 10%: -6,398% Operaciones totales: 175.296

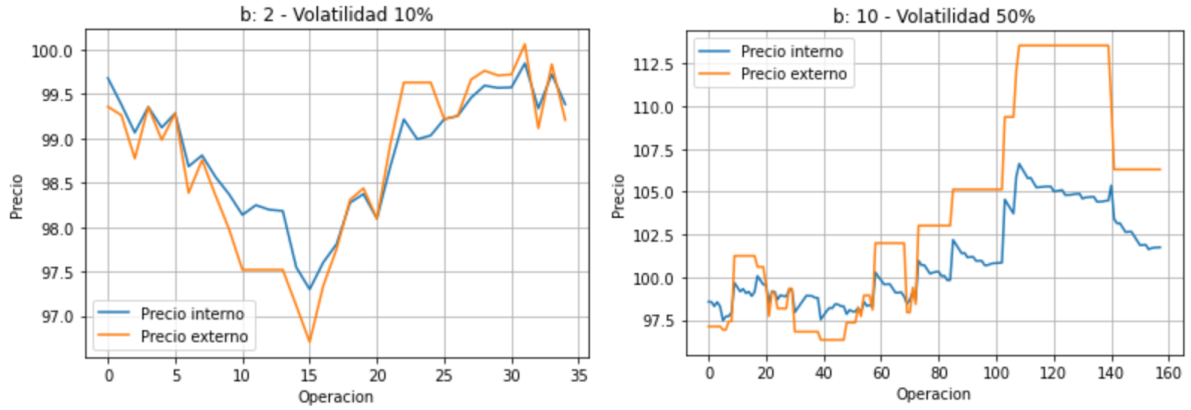
[15]

El hecho de que una menor cantidad de órdenes haya resultado en un menor diferencial puede resultar contra-intuitivo, pero después de analizar los resultados vemos que este se debe a la proporción de órdenes a precio de mercado con respecto al total de las órdenes emitidas. En los casos en los que creamos un mayor número de órdenes se acumula mucha profundidad a un nivel de precio alejado al precio de mercado, y luego un movimiento repentino del precio de referencia hace que estas órdenes emitidas anteriormente a precios alejados de los nuevos, sean ejecutadas contra nuevas órdenes a precio de mercado, y esto impacta en el promedio.

Este efecto es amplificado por la volatilidad ya que estas fluctuaciones abruptas en el precio suceden con mayor frecuencia. Por esta razón, el escenario donde más *diferencia* existe entre este libro de órdenes y el de precio de referencia, es aquel en donde existe mucha profundidad en un activo muy volátil.

En los siguientes gráficos podemos ver la evolución de los precios internos comparados con los precios de mercado para una iteración de Monte Carlo. Considerando en el eje X las operaciones realizadas internamente basadas en los precios simulados.

[15] Donde el percentil es la cantidad de casos debajo de ese nivel porcentual y dónde operaciones totales es sobre todas las 1000 iteraciones de la simulación de Montecarlo.



En estos ejemplos, podemos ver que cuando el precio externo tiene movimientos sustanciales, se produce que las órdenes de tipo límite de nuestro libro de órdenes sean ejecutadas contra las nuevas órdenes de mercado. Esto aumenta el diferencial promedio entre ambos mercados.

Este efecto podría ser mitigado poniendo límite a la cantidad de tiempo que una orden de tipo límite puede permanecer en el libro de órdenes antes de expirar. Otra posible solución que podría mejorar considerablemente el diferencial, sobre todo en situaciones de estrés, podría ser un mecanismo que limite la diferencia porcentual máxima entre las órdenes emitidas en nuestro exchange y el precio de referencia en un momento dado.

Capítulo 8. Conclusión

Los resultados demuestran que bajo la suficiente cantidad de liquidez, ó de agentes en el mercado que operan a precios de mercado, y bajo un comportamiento racional de los agentes, el precio interno del derivado replica el precio del mercado subyacente de una forma suficientemente cercana para ser una alternativa viable para ganar exposición a una multitud de mercados, de forma descentralizada y utilizando activos digitales.

La estructura que concluimos que sería la óptima, terminó difiriendo de nuestra hipótesis inicial, ya que más bien se propuso una solución en donde el libro de órdenes no es descentralizado, para evitar las fricciones y costos relacionados con un libro de órdenes que corre en una blockchain. Esta estructura, si bien posee componentes centralizados, interactúa con un sistema de *clearing* que depende de contratos inteligentes para liquidar las posiciones de los usuarios.

El potencial de crecimiento y mejoras de, tanto el modelo cuantitativo que da soporte al proyecto como el de la estructura del exchange en sí, es actualmente un trabajo en proceso con vías y soluciones múltiples que aún se están desarrollando. Por ejemplo, la incorporación de más *stablecoins* podría ser el puente que una mercados que cotizan en diferentes monedas con estos derivados DCFD.

Por último, a pesar de que en un futuro las criptomonedas no terminen prosperando, debido a por ejemplo, la incorporación de una legislación global que las prohíba completamente, no quita la importancia de indagar en este sector cuya tecnología subyacente puede permitir avances radicales en la industria financiera.

IV. Bibliografía

Brown, Christine & Dark, Jonathan & Davis, Kevin. (2010). Exchange traded contracts for difference: Design, pricing, and effects. Journal of Futures Markets. https://www.researchgate.net/publication/227517001_Exchange_traded_contracts_for_difference_Design_pricing_and_effects

Lin, L. X., 2019. *Deconstructing decentralized exchanges. Stanford Journal of Blockchain Law & Policy.* Stanford Journal of Blockchain Law & Policy. https://stanford-jblp.pubpub.org/pub/deconstructing-dex/release/1.

Maher Alharby, Maher Alharby. 2017. *Blockchain-based smart contracts : a systematic mapping study*. Newcastle UK. Newcastle University. http://airccj.org/CSCP/vol7/csit77211.pdf.

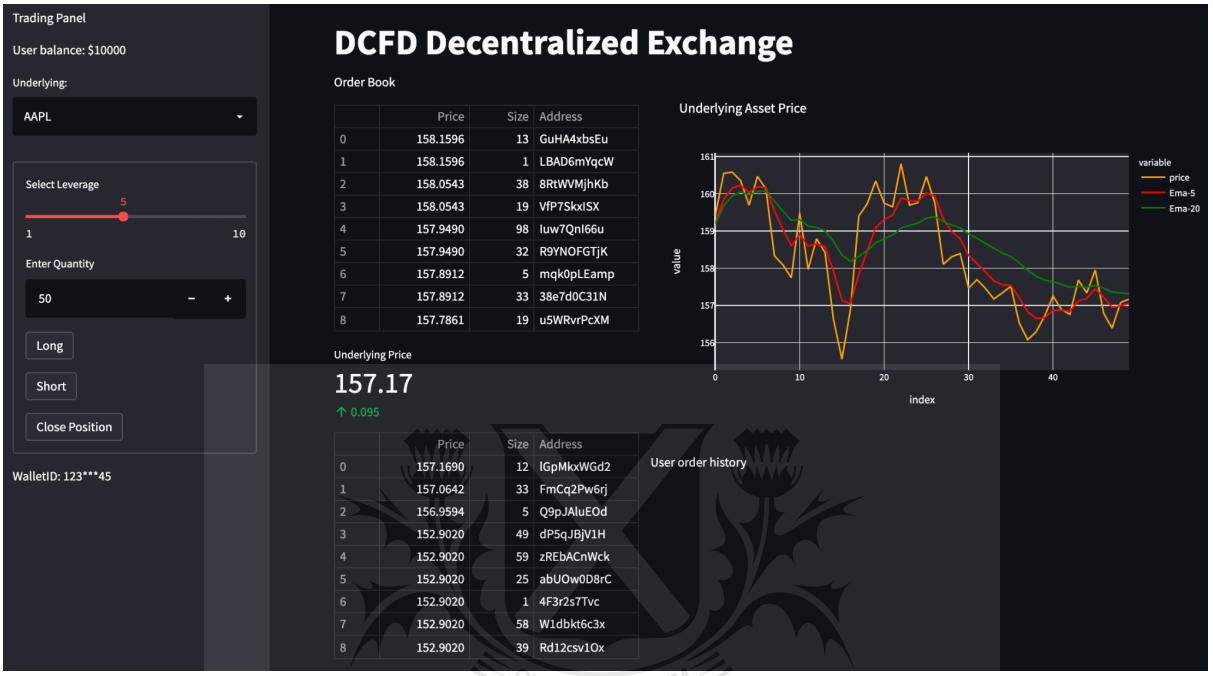
OpenNode Team. 2020. *Bitcoin Regulation: Which Countries are Bitcoin-Friendly*. https://www.opennode.com/blog/bitcoin-regulation-which-countries-are-bitcoin-friendly/.

The Law Library of Congress, Global Legal Research Center, 2018. *Regulation of Cryptocurrency Around the World*. https://www.loc.gov/law/help/cryptocurrency/cryptocurrency-world-survey.pdf.

Biais, B., Hillion, P., & Spatt, C. (1995). An Empirical Analysis of the Limit Order Book and the Order Flow in the Paris Bourse. *The Journal of Finance*, *50*(5), 1655–1689. https://www.jstor.org/stable/2329330

V. Anexo

- Descripción de la interfaz de la plataforma

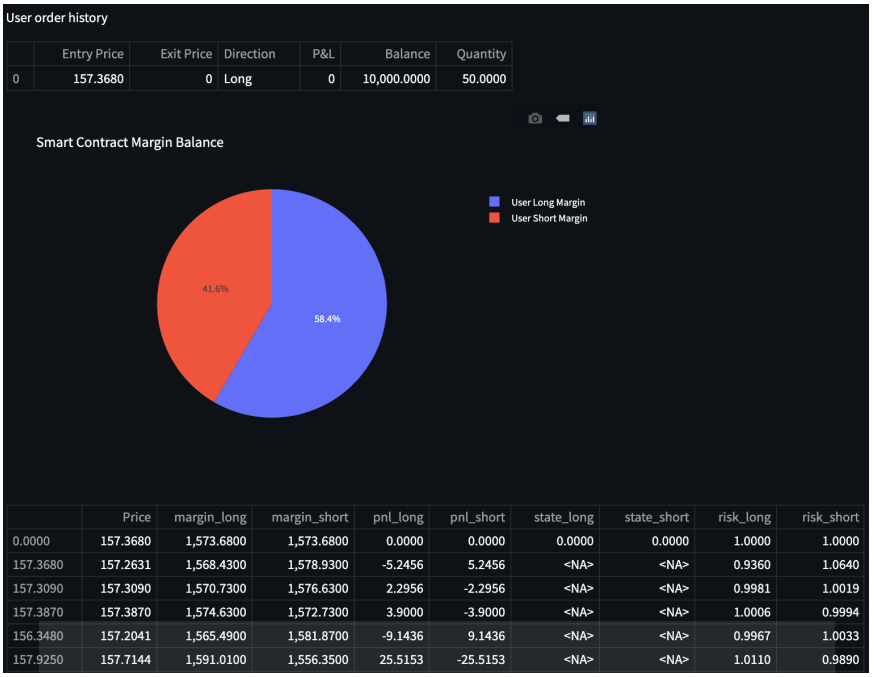


En la parte izquierda podemos ver el panel de trading, donde se muestra el balance del que dispone el usuario en USDT. Luego este puede seleccionar las especificaciones del contrato que desee operar, siendo estas el activo subyacente, el nivel de apalancamiento, la cantidad de unidades del subyacente y la dirección de la operación; *long* o *short*.

En la parte central podemos ver el libro de órdenes, dividido en órdenes de compra y órdenes de venta. Entre medio de ambos se encuentra el último precio operado y su cambio con respecto al precio anterior.

En la parte derecha de la pantalla se encuentra el gráfico del precio del subyacente, en el que también graficamos dos medias móviles.

Cuando el usuario emite una orden y esta es ejecutada, la plataforma despliega una serie de tablas y gráficos con información acerca de la evolución de la posición

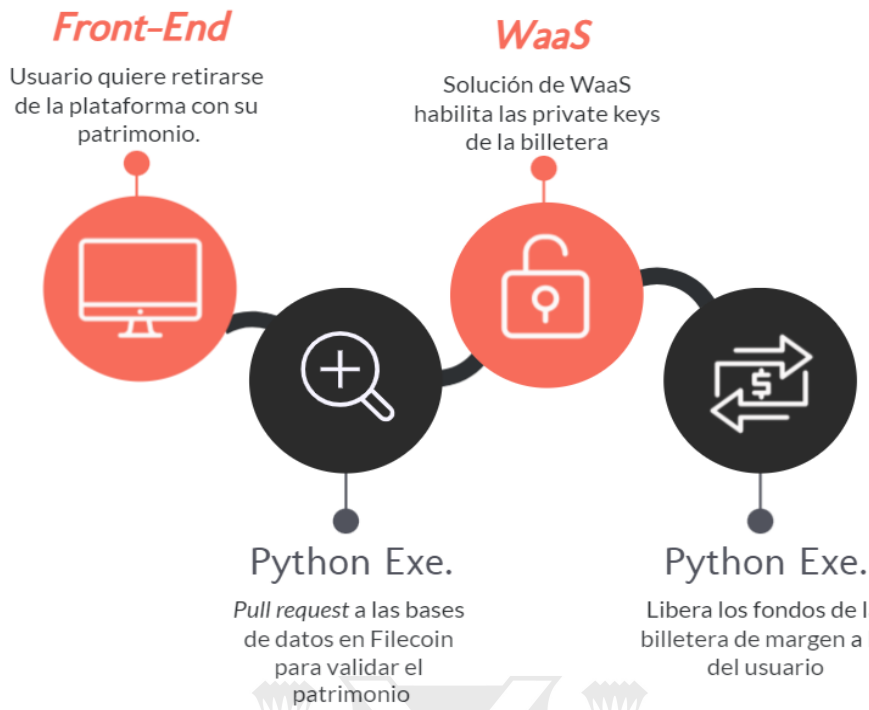


En la parte superior tenemos una tabla con el historial de las órdenes que, en este caso, indica que se abrió una posición *long* a un precio de $157,36 por 50 unidades.

Debajo vemos un gráfico de torta que nos representa el balance del smart contract y qué proporción de los fondos corresponde a cada usuario. Podemos ver que el desde el momento de apertura del contrato el precio subió, por ende 58,4% del margen corresponde al usuario *long*.

Al final podemos ver una tabla que contiene el *mark to market* de la posición a medida que el precio del subyacente evoluciona. Encontramos el margen y el P&L, o pérdidas y ganancias, de cada usuario así como también sus ratios de riesgo.

Exit del modelo actual



- Código de Python

A continuación se encuentra el código que utilizamos para obtener los resultados obtenidos y puede ser utilizado para replicarlos.

- Librerías utilizadas

```
import pandas as pd
import random as rd
import numpy as np
import matplotlib.pyplot as plt
```

- Funciones

```
def brownian_motion(df, u, sigma):
   dt = 1/360
   new_price = df.loc[df.index[-1]][0]*(1+u*dt + sigma * np.random.normal(0, 1) *
np.sqrt(dt))
   df.loc[df.index[-1]+1] = new_price

def create_orders(market_price,b, side):
   if side == 'buy':
       n = rd.randint(1,b)
       buy_orders_price = [market_price*(1-i/200) for i in range(n)]
       buy_orders_size = [rd.randint(1, i**2) for i in range(1,n+1)]
       return pd.DataFrame({'Price': buy_orders_price, 'Size':buy_orders_size})
```

```
    if side == 'sell':
        n = rd.randint(1,b)
        sell_orders_price = [market_price*(1+i/200) for i in range(n)]
        sell_orders_size = [rd.randint(1, i**2) for i in range(1,n+1)]
        return pd.DataFrame({'Price': sell_orders_price, 'Size':sell_orders_size})



def match_orders(i):
    global order_book_b, order_book_s,trades, trade_n
    try:
        while min(order_book_s['Price']) <= max(order_book_b['Price']):
            trade_n += 1
            if order_book_b['Price'][0] >= order_book_s['Price'][0]:
                price = (order_book_s['Price'][0]+order_book_b['Price'][0])/2
                size = min(order_book_b['Size'][0], order_book_s['Size'][0])
                trades = trades.append(pd.DataFrame({'Price':price, 'Size':size, 'MP':i},
index=[trade_n]))

                order_book_b.loc[0,'Size'] -= size
                order_book_s.loc[0,'Size'] -= size

                if order_book_b.loc[0,'Size'] == 0:
                    order_book_b = order_book_b.drop(0).reset_index(drop=True)

                if order_book_s.loc[0,'Size'] == 0:
                    order_book_s = order_book_s.drop(0).reset_index(drop=True)
    except:
        print('No orders to match')
```

- Simulador de montecarlo para los diferenciales

```
trade_n = 0
b = 10
spreads = []
for i in range(1):
    price_series = pd.DataFrame(100,columns=['price'], index=[0])

    order_book_b = pd.DataFrame(create_orders(price_series.price[0], b, 'buy'),
columns=['Price', 'Size']).sort_values(by=['Price'],
ascending=False).reset_index(drop=True)
    order_book_s = pd.DataFrame(create_orders(price_series.price[0], b, 'sell'),
columns=['Price', 'Size']).sort_values(by=['Price']).reset_index(drop=True)

    trades = pd.DataFrame(columns=['Price','Size', 'MP'])

    for i in range(30):
        brownian_motion(price_series, 0, 0.5)
        price = price_series.price.iloc[-1]
```

```
        new_buy_orders = create_orders(price, b, 'buy')
        order_book_b = order_book_b.append(new_buy_orders).sort_values(by=['Price'],
ascending=False).reset_index(drop=True)

        new_sell_orders = create_orders(price, b, 'sell')
        order_book_s =
order_book_s.append(new_sell_orders).sort_values(by=['Price']).reset_index(drop=True)

        match_orders(price)
    spreads.append(((trades['Price']/trades['MP'])-1).mean())
```

- Gráfico del precio interno vs precio externo

```
plt.plot(trades[['Price', 'MP']].reset_index(drop=True))
plt.legend(['Precio interno', 'Precio externo'])
plt.ylabel('Precio')
plt.xlabel('Operacion')
plt.grid()
```