



MEMBRANA

블록체인 금융 플랫폼

WHITEPAPER

목차

개요	6
암호화폐 시장	7
믿을 수 있는 자산 관리	7
Membrana 블록체인 플랫폼	8
컨셉	9
Membrana의 사용자	9
트레이더	9
투자자	9
현재 상황	10
트레이더를 위한 기회	10
투자자를 위한 기회	11
Membrana의 기술	12
신탁관리자금의 안전한 전송	12
투자자와 트레이더 간의 계약	12
블록체인을 활용한 트레이더의 과거 수익성 확인	13
여러 독립적인 투자자들의 자산 분배	13
Membrana의 사용료	13
Membrana의 장점	13
트레이더용	14
투자자용	14
암호화폐 거래 생태계	14
활용 기술	16
이더리움 스마트 컨트랙트	16
거래소 API	16
투자자를 위한 펀드 보안	16
투자자의 API 키 보호	17
오라클	17
메타마스크	17

이더리움 인증을 지원하는 모바일 브라우저	18
웹 기술의 스택	18
멤브레인 시스템의 운영 알고리즘	19
권한 부여	19
API 키 추가	19
트레이더의 신뢰할 수 있는 관리 제안	20
투자자의 트레이더 선택	20
계약 체결	21
트레이더의 등급	21
목표 수익 달성	21
계약 종료	22
손실 방지	22
금지된 투자자의 행동	23
계약 조기 종료	23
투자자와 트레이더 간 지불	24
투자자와 트레이더 사이의 계약 예시	24

고급 수준의 플랫폼 아키텍처 (베타 버전)	27
프론트엔드	27
백엔드	27
데이터베이스	27
이더리움 블록체인	28
Etherscan.io	28
오라클	28
마이크로 서비스	29
 스마트 컨트랙트 코드	 30
 MEMBRANA 플랫폼 인터페이스 (베타 버전)	 33
트레이더를 위한 단일 거래 터미널	33
트레이더 등급	34
투자자의 개인 계정	35
트레이더 프로필	36
 관심사항 기능	 37
Membrana 플랫폼의 API	37
전문화 된 뉴스 피드	37
신뢰할 수 있는 이더리움 월렛 펀드 관리	37
트레이더의 손실 보상이 가능한 계약	38
조건부 거래소 주문	38
매니저 계약: 투자자와 포트폴리오 매니저의 계약	38
Membrana 토큰의 경제	39
다중 계약 요청	39
 프로젝트 로드맵	 40
Q1 2017. 개발 컨셉	40
Q2 2017. 컨셉 타당성 검증	40
Q3 2017. 인터페이스 제작	40
Q1 2018. 알파 버전	40
Q3 2018. 베타 버전	41

Q4 2018. 주요 업데이트	41
Q1 2019. 조건부 주문 및 UI 업데이트	42
Q2 2019. 고급 기능	42
Q4 2019. 최종 배포	42
Q2 2020. 추가 기능	42
Q1 2021. 주식 시장 확대	43
팁	44
어드바이저	48
링크와 연락처	50



개요

Membrana는 암호화폐 자산의 신뢰성 있는 관리를 위하여 투자자와 거래처 간의 상호 이익을 실현하고 안전한 계약을 체결하기 위해 개발된 블록체인 플랫폼입니다.

링크: beta.membrana.io, membrana.io

관련 문서: [Membrana Pitch_deck](#), [Membrana Business Overview](#)

암호화폐 시장

블록체인 기술 덕분에 비트코인, 라이트코인, 이더리움 등 새로운 형태의 암호화폐들이 대거 시장에 진입하였습니다. 이 자산들은 모두 탈중앙화 디지털 화폐로서, 각각 소유자들이 월렛 간의 거래가 가능합니다.

다양한 암호화폐만큼 시장경제 원리에 근거하여 거래소들도 많습니다. Bittrex, Kraken, Fitfinex, Hitbtc 등 수많은 거래소들은 투자자들이 서로 다른 거래 전략을 활용하고, 금리 변화로부터 이윤을 창출할 수 있도록 많은 암호화폐를 동시에 거래할 수 있는 기능을 제공하고 있습니다. 점점 더 많은 트레이더들이 성공을 거두면서, 매년 수백 혹은 수천 퍼센트에 이르는 안정적이고 높은 수익률을 보여주고 있습니다. 굴곡은 많았지만, 주요 암호화폐인 비트코인처럼 돈을 벌 기회가 많았습니다. 하지만 이 것이 보통 일반 투자자들에게 쉬운 일이 아니기 때문에 다른 방법은 없는 것일까요?

사업 기회

우리는 모두 은행 시스템과 은행 예금을 알고 있습니다. 우리들은 이것이 가장 안전한 투자로 간주되고 있지만, 평소 은행 예금 금리는 어떨까요? 일년에 몇 퍼센트? 이것은 때때로 인플레이션보다 낮기 때문에, 투자 수익을 기대하기는 어렵습니다. 브로커 등을 활용한 다른 투자 방안은 높은 진입 장벽을 가지고 있습니다. 그래서 기본적으로 아무도 손실을 원하지 않기 때문에, 사람들이 은행에 예금을 예치하는 것 이외에는 선택의 여지가 없는 편입니다.

그럼 암호화폐 시장은 어떨까요? 우선 일반적인 수준의 안전성을 갖추지 못한 신뢰성 낮은 생태계입니다. 일반 트레이더에게서는 이윤을 추구하기가 힘듭니다. 두번째로는 우리 모두가 현재의 암호화폐 시장이 매우 불안정 하다는 것을 알고 있습니다. 한편으로, 이로 인해 성공적인 트레이더들이 매우 높은 수익률을 달성할 수 있게 합니다. 하지만, 이는 모든 시장 참여자들, 특히 초보 투자자들의 리스크가 됩니다. Forex와 같은 전통적인 거래소와는 달리, 암호화폐 거래소에서는 레버리지가 없어도 대부분의 자산을 하루만에 잃을 수도 있습니다.



차이점

하지만 여러분들이 암호화폐 시장에서 이윤을 취할 수 있는 능력, 특별한 특징을 지니고 혼자서 거래하거나 은행 시스템에서만 사용할 수 있었던 안전 수준의 전문 트레이더들을 위한 믿을수 있는 시스템을 상상해보세요. Membrana 플랫폼이 그 해답입니다. 금융의 자유로운 시대에 오신 것을 환영합니다.

믿을 수 있는 자산 관리

신뢰할 수 있는 자산 관리는 늘 수요가 있던 분야입니다. 투자자들은 수익을 창출하기 위하여 유명한 트레이더들에게 자신의 자산을 위탁하고자 합니다. 암호화폐 시장 역시 예외가 아니며, 투자자와 트레이더들의 관심 또한 높아지고 있습니다. 투자자들은 자신의 자산을 숙련된 트레이더들에게 양도하고, 트레이더들은 그들의 자산을 보유하게 됩니다. 암호화폐 시장에서 높은 수익을 얻을 수 있는 확률이 높고, 동시에 경험이 부족한 투자자들의 공통점인 리스크로 발생하는 손실에 대한 위험성을 최소화 시킬 수 있습니다.

하지만, 현재로서는 투자자와 트레이더 간 암호화폐 계약 체결에 대한 편리하고 안전한 방법이 없는 실정입니다. 그럼에도 불구하고, 이런 계약들은 채팅이나 포럼 등에서 활발하게 올라오고 있습니다. 이는 일부 트레이더들이 투자자의 자금을 횡령하거나 은폐, 도주하는 등의 사기행위가 기승하게 되는 요인이 됩니다.

Membrana 블록체인 플랫폼

Membrana 블록체인 플랫폼은 금융시장과의 협력을 위한 다양한 도구를 제공하는 멀티 레벨 플랫폼입니다. 우선, 투자자와 트레이더들이 상호 이익이 되는 블록체인 보호 계약을 체결하여 암호화폐 자산 위탁 관리를 위하여 거래를 체결 수 있도록 하기 위하여 만들어졌습니다. Membrana는 투자자와 트레이더에게 투명하고 탈중앙화 되어있으며 안전한 시스템을 제공하여, 이 시스템이 서로간에 벌어들인 수익 시점까지 계약을 체결하고 이행하는 과정을 진행하게 됩니다. 두번째로는 Membrana는 한 터미널에서 여러 거래소를 효율적으로 거래할 수 있는 도구를 제공합니다. 고급 UI/UX, 조건부 주문, 분석, algo-trading 및 algo builder, 클라우드 예측, 메신저 알림 등 이런 다양한 도구를 Membrana 플랫폼에서 활용할 수 있습니다.



컨셉

Membrana의 사용자

트레이더

트레이더는 거래소의 암호화폐 시장에 참여하는 사람으로서, 투자자의 자산 증대와 관련된 수익성 있는 거래를 진행하고 지급되는 보수를 위해 디지털 자산을 관리하고 보수를 받기를 원하는 사람들입니다. 트레이더는 주목할만한 가치가 있는 곳에 투자하는 것에 관심이 있으며, 자신의 비용 및 관리를 받고자 이전받은 자금으로 서로 다른 암호화폐를 거래하기 위한 관리에 관심이 많습니다.

투자자

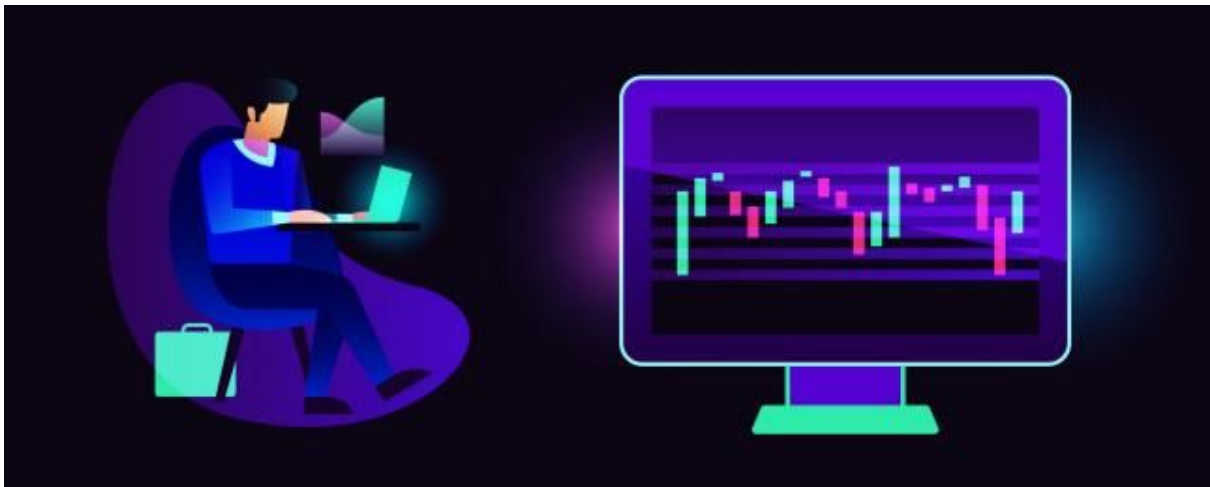
투자자는 디지털 자산을 가진 암호화폐 시장의 참여자로서, 이러한 자산을 성공적인 트레이더를 통해 신탁 관리를 하기를 원합니다. 투자 수익률을 높이기 위해 투자자들은 안전한 펀드와 트레이더들로부터 완전하고 신뢰할 수 있는 정보를 얻는 것에 관심이 많습니다.

현재 상황

Membrana 플랫폼의 주목적은 투자자와 트레이더를 직접 연결하여 상호간의 이득을 취하고, 안전하고 신뢰할 수 있는 계약을 체결하는 것입니다.

Membrana는 투자자와 트레이더 모두의 리스크를 피하고, 신뢰할 수 있는 거래 관리를 보다 이윤을 취하고 안전하게 만들기 위하여 다음의 문제를 해결하고자 특별한 알고리즘을 개발하였습니다:

- 자금의 안전과 계약 조건 이행을 위한 보증인 참여 필요성
- 투자금에 대한 권리 상실 (제 3자에게 자금 이전)
- 양 당사자 간의 신뢰성 부족
- 거래자의 과거 수익률을 정확하게 확인할 수 있는 기회 부족
- 계약 조건 위배 위험성
- 여러 다른 거래소에서 한번에 거래할 수 있는 편리한 톨의 부재



트레이더를 위한 기회

Membrana 플랫폼은 트레이더들에게 다음과 같은 기회를 제공합니다:

- 단일 인터페이스를 통한 다양한 암호 화폐 거래
- 거래소 계좌와 투자자들에 의한 신탁 관리를 위해 트레이더 계좌에서의 거래
- 잠재 투자자에게 신뢰 관리에 관한 서비스 제공
- 투자자와 트레이더간의 신뢰관리 계약 체결
- 투자자와 트레이더가 함께 암호화폐 자산의 신탁 관리 계약 체결
- 이러한 조건을 완성하기 위하여 블록체인 이더리움 스마트 컨트랙트 기술로 계약 조건을 수정



투자자를 위한 기회

Membrana 플랫폼은 투자자들에게 다음과 같은 기회를 제공합니다:

- 신뢰 관리에 대한 서비스를 제공하는 트레이더의 완전하고 명확한 정보를 기반으로 적합한 트레이더 선택
- 블록체인에 저장된 트랜잭션의 해시 합계를 통하여 확인된 트레이더의 과거 활동의 수익성에 대한 신뢰할 수 있는 데이터를 가져올 수 있습니다.
- 투자자의 계좌에서 다양한 거래소에서 암호화폐 자산의 신탁 관리를 위한 계약 체결
- 직접 자금을 이체하지 않고 트레이더가 자신의 자금을 신탁 관리팀에 양도합니다. Membrana 플랫폼을 활용하면 투자자는 펀드를 관리할 수 있지만, 자금을 소유 하진 않습니다. 트레이더들은 투자자들의 거래소 계좌에 거래할 기회를 얻게 되지만, 그들 또한 자금을 소유하지 않습니다. 투자자들 또한 Membrana 플랫폼으로 자금을 이전하지 않습니다.
- 하나의 거래소 계좌에서 여러명의 트레이더에게 관리 자금을 이전하고, 각 통화로 관리하기 위하여 이전된 금액을 표시
- 허용된 거래 통화 집합과 최대 거래 손실 (손실 방지) 등 신뢰 관리로 이전된 거래소 계좌와의 거래 제한 설정
- 체결된 계약의 이행 진행 상황에 대한 정보 획득



Membrana의 기술

신탁관리자금의 안전한 전송

투자자의 거래소 계좌를 신탁 관리 트레이더에게 양도할 목적으로, 거래소가 제공하는 API키를 활용합니다. API키는 트레이더에게 전송되지 않고, Membrana의 플랫폼에 안전한 데이터 베이스로 저장됩니다. 이 트레이더는 Membrana 단일 거래소를 통해 거래하게 됩니다. 따라서 트레이더는 투자자의 자산에 접근할 수 없게 됩니다. 모든 투자자들의 자금은 그들의 거래소 계좌에 남아있으며, 트레이더에게 양도되지 않습니다. Membrana는 계약 단계에서 중지 손실, API 키에 의한 거래 기간 등 설정된 제한사항도 유지 됩니다.

투자자와 트레이더 간의 계약

투자자와 트레이더 사이의 거래를 체결하기 위하여 이더리움 스마트 컨트랙트를 활용합니다. 트레이더의 대가는 미리 스마트 컨트랙트로 적립되며, 신탁관리에서 이전한 금액의 백분율로 설정된 목표 이익에 도달하거나 스마트 컨트랙트 기간이 만료되면 자동으로 지급됩니다.

블록체인을 활용한 트레이더의 과거 수익성 확인

Membrana 플랫폼을 통하여 트레이더들이 체결한 모든 계약은 데이터베이스에 저장됩니다. 이 정보는 ROI, 등급 등과 같은 지표를 활용하여 트레이더의 성공 확률을 계산하는데 활용됩니다. 이들 지표가 투자자들에게 제공되어 적합한 거래자를 선전할 수 있는데 도움이 될 것입니다.

Membrana 플랫폼은 블록체인을 사용하여 트레이더들의 세부사항을 검증합니다. 이를 위해 당일 트레이더의 거래의 해시 합계를 블록체인 이더리움 스마트 컨트랙트에 저장합니다. 해시 합계가 저장되는 블록은 날짜가 위조되어 지정될 수 없습니다. 따라서 트레이더의 거래소 거래에 대한 모든 데이터의 유효성은 각각 거래일로부터 나온 블록체인의 해시 합계로 확인됩니다. 따라서 어떤 거래가 수익성이 있는지 알 수 없기 때문에 데이터베이스에서 수익성 있는 거래와 그 해시합만 외부적으로 저장하는 것은 불가능합니다.

여러 독립적인 투자자들의 자산 분배

Membrana 플랫폼은 트레이더가 계약 조건에 명시된 대로 자신의 자금의 일부만을 환전 계좌로 이전하고 트레이더에게 양도할 수 있습니다. 나머지 자금은 투자자의 관리하에 있으며, 독립적인 거래, 다른 트레이더에게 이체, 또는 거래소에서 인출하는 데 사용할 수 있습니다.

암호화폐는 실제로 다른 트레이더들이나 투자자가 직접 한 거래소 계좌에서 발생합니다. 그럼에도 불구하고 Membrana 플랫폼은 각 거래자와 투자자 자신이 신탁 관리자에게 이전한 금액 내에서 독립적인 거래를 제공합니다.

투자자는 계약이 종료되는 시점에 신탁관리로 이전한 금액을 사용할 회계 통화로 이전된 금액을 신탁 관리인으로부터 결정합니다.

Membrana의 사용료

Membrana의 수수료는 금액의 0.5%가 신탁관리에 이전됩니다. 자세한 내용은 Membrana [비즈니스 개요](#)를 참조하세요.

Membrana의 장점

트레이더용

- 신탁 관리 투자 유치 가능성
- 플랫폼 내 단일 거래 터미널을 통한 모든 주요 거래소 거래 지원
- 실적에 따른 트레이더 등급 제도
- 트레이더와 투자자 간의 거래 조건은 스마트 컨트랙트에 명시된 대로 이행하기 때문에 변경 불가
- 계약 시점의 거래에 대한 거래 수수료 설정
- 모바일을 통한 거래소 및 계약 작업 가능

투자자용

- 모든 거래에 대한 안전성 및 투명성
- 펀드는 항상 투자자의 계좌에 존재함
- 등급에 따라 가장 성공적인 투자 트레이더를 선정할 가능성
- 트레이더의 성공에 대한 신뢰할 수 있는 정보 보장
- 모든 거래 조건은 스마트 컨트랙트로 설정되기 때문에 변경이 불가능
- 원하는 화폐 선택: BTC, ETH 및 USD
- 신탁 관리를 위한 다양한 거래자들의 단일 거래 계좌에 대한 자금 배분
- 신탁 관리를 위해 양도되지 않은 펀드의 자기 관리를 승인한 거래소 계좌에서 신탁 관리

자금의 일부를 이전 가능

- 모바일을 통한 Membrana 플랫폼 작업 가능
- 여러 계약 및 트레이더가 함께 작업하기 위한 사용자 친화적인 인터페이스

암호화폐 거래 생태계

- 추가적인 투자 가능성
- 신규 투자자로 인한 시장 자산 유입
- 암호화폐 거래소에 대한 트랜잭션 수 및 볼륨 증가
- 트레이더를 선정하여 그들에게 자산을 이전하는 것으로 충분하기 때문에 암호화폐 거래소에 대한 요구사항 완화
- 트레이더의 수입 및 포트폴리오 내역 증가
- 암호화폐 시장 신탁 평판 개선

활용 기술

이더리움 스마트 컨트랙트

투자자와 트레이더 사이의 신뢰 관리를 위한 거래를 체결하기 위해 이더리움 스마트 컨트랙트를 활용합니다. 스마트 컨트랙트는 계약의 모든 세부 사항이 포함되어있으며 관련 계좌에 투자자는 목표 이익이 도달했을 때 지급할 금액으로 트레이더의 의뢰 보상을 미리 적립해 둡니다. 보상은 계약 기만이 만료되고 투자자가 소득을 수령하면 자동으로 트레이더에게 지급됩니다. 목표 이익 설정치가 있는 경우, 트레이더가 수령하거나, 투자자가 금지 조치를 취하는 경우에 계약은 조기 종료될 수 있습니다. 금지 조치의 예로는 투자자가 거래소 계좌에서 돈을 인출하거나, 명시적 허가 없이 이 계좌를 통한 거래를 수행하는 것이 있습니다. 스마트 컨트랙트는 거래소 계정의 현재 상태에 대한 정보를 얻기 위해 아래에 설명된 바와 같이 신탁을 사용합니다.



거래소 API

대부분의 거래소는 자동 거래를 위한 응용 프로그래밍 인터페이스 (API)를 제공합니다. API는 거래소 웹사이트에서 사용하기 위한 계정 소유자가 획득한 키 값을 활용하여 특정 거래소 계정으로 액세스합니다.

Membrana 플랫폼은 트레이더들에게 지원되는 모든 거래소에서 거래를 수행할 수 있는 단일 거래 터미널을 제공합니다. Membrana는 트레이더가 소유하고 있는 API 키를 활용하여 거래소 API를 통해 각 거래를 진행하거나, 신탁관리 트레이더에게 양도하게 됩니다. API는 Membrana 플랫폼 내의 보호된 데이터 베이스에 저장됩니다.

투자자를 위한 펀드 보안

Membrana 플랫폼에서 구현된 기술은 투자자들에게 자금 보장을 제공합니다. - 신탁 관리 트레이더들에게 자금이 이전됩니다. 그 펀드는 어떤 상황에서도 트레이더에게 직접 전달되지 않으며, 트레이더들이 투자자의 API에 접근할 수 없습니다. API키는 Membrana 플랫폼의 안전한 데이터 베이스로 저장됩니다. 트레이더들은 Membrana 플랫폼의 단일 거래 단말기를 통해 거래를 수행합니다. 각 신탁 관리 계약은 거래할 수 있는 특정한 통화 제한을 가지고 있기 때문에, 트레이더가 더미 통화에 투자자의 자금을 이용하는 것이 불가능합니다.

투자자의 API 키 보호

백엔드 시스템에 의하여 원래의 API키의 보안이 보장됩니다. 백엔드 시스템은 개인 계정과 거래 터미널의 기능을 관리하며, 거래소와 직접 상호작용하는 서버에 대한 접근을 허용하지 않습니다. 거래 시스템은 격리된 환경에 있으며, 이 시스템을 통한 거래에 대한 접근은 ECDSA 알고리즘을 활용한 서명 승인 세션에서만 가능합니다.

우리가 진행중인 주된 관심사는 SSH가 아닌 보안 등급에 전혀 접근할 수 없는 RAM 전용 방어

시스템입니다. 완전히 잠겨있으며 거래에 서명하기 위해 암호화된 연결 포트는 하나뿐입니다. 이는 공격 가능성과 잠재적인 취약성을 낮추기 위해 제한된 소프트웨어 세트만 허용합니다. 연결을 암호화하기 위해 구글이 추천하는 ChaCha20-Poly1305 aead 암호화 기술을 활용합니다.

오라클

신탁은 외부로부터 이더리움에 데이터를 제공하는 소프트웨어 구성요소입니다. 이러한 데이터는 스마트 컨트랙트에서 더 많이 활용될 수 있습니다.

Membrana 플랫폼에서는 백엔드 시스템이 정보를 블록체인, 환전 계좌의 통화 잔액 자료, 특정 거래소 계좌의 거래, 암호화폐 시세, 특정 거래자의 거래 등을 입수하여 기록하는데 활용됩니다.

메타마스크

Membrana 플랫폼에 대한 사용자 권한 부여는 메타마스크(MetaMask)를 활용하여 실행됩니다. 메타마스크는 웹브라우저용 플러그인으로 (현재 크롬과 파이어폭스에서 지원 중) 사용자는 개인키 없이 JavaScript web3 오브젝트를 통해 이더리움에서 블록체인 트랜잭션을 수행할 수 있습니다. Membrana 플랫폼은 스마트 컨트랙트와 함께 작동하기 위하여 web3을 사용합니다.



이더리움 인증을 지원하는 모바일 브라우저

현대의 기술을 활용하여 사용자 인터페이스를 제작하는 것은 Membrana 플랫폼 인터페이스가 CIPHER와 Toshi와 같은 이더리움 인증을 지원하는 브라우저에서 데스크톱 브라우저와 모바일 장치에서 모두 작동할 수 있도록 합니다. Membrana 웹사이트 또한 모바일 기기용으로 제작되었습니다.

웹 기술의 종류

Membrana 시스템과 사용자의 상호 작용은 웹사이트를 통하여 이루어집니다. 이 사이트는 SPA(Single page Application) 기술을 기반으로 합니다. 이것은 다음과 같은 기술 종류가 활용됩니다:

- MongoDB
- Node.JS
- Web3
- RabbitMQ
- Redux
- React.JS

멤브레인 시스템의 운영 알고리즘

권한 부여

데스크탑 브라우저 사용자는, 인터넷 브라우저용 메타마스크 플러그인을 설치하고 이를 활용하여 이더리움에 로그인 합니다.

모바일 기기에서 사용자는 블록체인 이더리움 네트워크에서 인증을 지원하는 브라우저를 설치합니다. Cipher나 Toshi 브라우저는 지금 활용 가능합니다. 권한 부여는 브라우저의 인터페이스를 통해 이루어집니다.

API 키 추가

유저가 암호화폐 거래소 웹사이트에 로그인하여 자동 거래용 API 키를 요청합니다. 또한, membrane.io 웹사이트에서 사용할 API 키를 추가하는 옵션이 있는데, 여기서 키 자체와 그 이름은 키 할당에 대한 더 나은 이해를 위해 입력됩니다. 예를 들어 “투자의 키”로 볼 수 있습니다.

이 키가 신뢰 관리를 위해 트레이더에게 전달되면, 추가된 API키를 사용하면 동시에 거래할 수 있는 통화 수를 제한할 수 있어 USD, BTC, ETH (지정된 통화)에 쉽게 접근할 수 있도록 보장합니다. 이러한 제한은 투자자의 자금으로부터 불법, 사기 또는 기타 바람직하지 않은 통화를 구매하고 활용하는 트레이더로부터 투자자를 보호합니다.

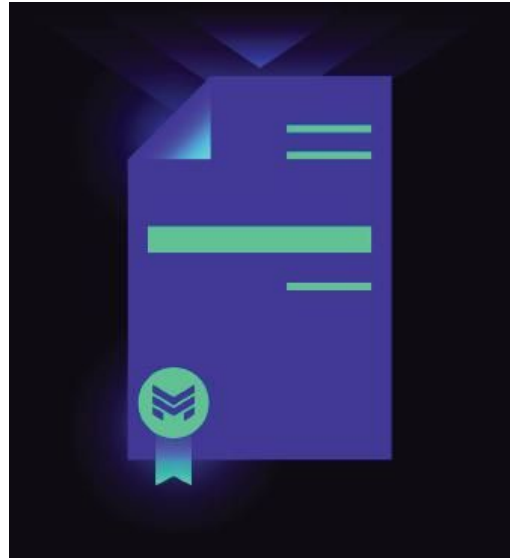
Membrana에 API키를 추가한 후, 이 키를 통해 트레이더의 자체 거래가 단일 거래 단말기를 통해 나타납니다. 또한 이 API 키를 활용하여 이 키가 속하는 거래소 계좌의 펀드 신탁 관리 계약을 체결할 수 있게 됩니다.

하나의 거래소 계정의 경우, 하나의 API 키만 Membrana 시스템에 추가할 수 있습니다.

트레이더의 신뢰할 수 있는 관리 제안

신탁 관리 자금을 받을 의향이 있는 트레이더는 투자자와의 계약의 매개변수를 다음과 같이 정합니다.

- 회계 통화: BTC, ETH 및 USD. 트레이더에게 이전하고, 계약에 따라 이익을 계산하며, 당사자간의 지불을 위해 활용됨.
- 신탁으로 이전한 금액의 백분율로 목표 이익 달성 시 계약 종료.
- 최소 투자금액과 회계 통화가 있는 경우
- 계약 기간
- 손실 방지: 최대 허용 손실 (자산 초기 금액의 백분율)
- 거래자 수수료 금액 (목표 이익의 백분율)



투자자의 트레이더 선택

신탁 관리로 자금을 이전할 의향이 있는 투자자는, 필터를 통해 부적절한 옵션을 배제하고, 트레이더의 명성, 수익성, 계약 금액, 계약 기간 등과 같은 다양한 조건을 제공받아, 편리한 방법으로 적극적인 트레이더들의 제안들을 검토할 수 있습니다.

트레이더의 프로필을 확인할 때, Membrana 플랫폼을 통해 만들어진 트레이더의 기록을 확인할 수 있습니다. 이 거래 이력은 투자자가 트레이더의 성공 여부를 분석할 수 있습니다. 이 데이터의 신뢰성은 이더리움 블록에 저장된 해시 합계 (sha256)에 의해 검증됩니다. 사용자는 누구나 트레이더의 거래 데이터를 확인할 수 있는 기회를 제공합니다. <http://stherscan.io>와 같은 웹사이트에 게시된 멤브라나 플랫폼의 특별한 스마트 컨트랙트 주소에 기록된 것과 비교하면서 과거 어느 종목을 거래 했는지에 대하여 그들의 SHA256을 확인 할 수 있습니다.



계약 체결

적절한 제안을 선택한 투자자는 트레이더에게 계약 체결을 위한 신청서를 보냅니다. 이 계정을 관리하려면 투자자는 암호화폐 거래소에 대한 계정과 Membrana 플랫폼에 API 키를 추가해야 합니다.

투자자가 거래소 계좌에 다른 통화를 가지고 있더라도, 투자자는 관리를 위해 이전한 금액을 회계 통화로 설정합니다. 그러나 이 회계 통화는 투자자의 계정에 있어야 하며 관리자를 위해 이전한 금액보다 크거나 같아야 합니다. Membrana 플랫폼은 트레이더가 특정 회계 통화로 이전된 금액만 거래할 수 있도록 분배 유형을 제공합니다. 이외의 다른 모든 펀드는 투자자가 완전한 권한을 가지고 있습니다.

이후 트레이더는 신청을 받아들이게 됩니다. 체결된 계약은 이더리움 스마트 컨트랙트에 기록됩니다. 최대 수수료는 트레이더의 이더리움 월렛에서 스마트 컨트랙트로 이전되며 향후 트레이더에게 지불하기 위하여 사용됩니다.

트레이더의 등급

Membrana 플랫폼을 사용하는 각 트레이더는 트레이더 등급 제도에 참가합니다. 등급제는 트레이더의 과거 실적에 대한 종합 분석에 기초합니다. 이러한 매개변수에는 트레이더의 ROI, 진행한 계약 수, 수익을 달성한 계약, 즉 현재 관리 중이거나 과거에 관리 되어 온 총 자금 금액 등이 포함됩니다.

트레이더 등급이란:

- 트레이더를 찾는데 활용되는 편리한 도구
- 특정 트레이더의 성과를 평가하는데 활용하는 데 사용되는 편리한 도구
- 트레이더의 동기 부여 제공
- Membrana 플랫폼의 효율적인 성과에 대한 지표

계약 종료

트레이더와 투자자의 계약은 다양한 이유로 종료될 수 있습니다.

목표 수익 달성

거래 체결시점에서 기록된 목표 이익을 달성 시, 자동으로 계약이 완료됩니다. 그 후 유보되어 있던 트레이더의 지불금이 전액 지급됩니다.

계약 만료

계약이 만료되면 다음 두 가지 옵션이 생성됩니다.

옵션 1: 트레이더의 거래 성과는 부정적입니다. 이 경우에는, 스마트 컨트랙트에서 유보된 트레이더의 지급액은 트레이더에게 지급되지 않고, 전액 투자자에게 반환됩니다.

옵션 2: 트레이더의 거래 성과가 긍정적입니다. 이 경우 트레이더는 계약 조건에 따라 보상을 받게 됩니다. 트레이더의 지급액은 목표 이익에 대한 실제 이익의 비율로 계산됩니다. 지급 준비금의 미지급된 부분은 투자자에게 반환됩니다.



손실 방지

투자자를 큰 손실로부터 보호하기 위하여, 회계 통화의 최초 금액의 백분율로 손실 방지 값을 스마트 컨트랙트에 고정합니다. 트레이더가 거래한 결과 펀드 금액이 일정액으로 줄어들면 자동으로 계약이 종료됩니다. 따라서 보상은 트레이더에게 지급되지 않고 투자자의 월렛으로 반환됩니다. 트레이더가 매입한 모든 통화는 자동으로 회계 통화로 시세에 맞게 교환되어 처리됩니다.

손실 방지를 계산할 때, 이러한 통화의 구매를 위한 거래소의 주문량을 고려합니다. 가장 수익성이 높은 통화 매입 입찰은 물량이 부족할 수 있으므로, 다음 입찰 등을 실시합니다.

금지된 투자자의 행동

투자자는 트레이더가 관리하기 위해 송금한 금액을 초과하는 금액으로 환전 계좌에서 돈을 인출할 수 없습니다. 이와 비슷하게, 투자자는 사용가능한 자금을 초과하는 금액에 대하여 거래소 계좌에서 거래할 수 없습니다. 트랜잭션에 필요한 자금이 충분하지 않을 경우 오라클은 스마트 컨트랙트를 통지하고 자동으로 트랜잭션을 종료합니다. 그러나 투자자는 트레이더에게 전액 상환된 예약 수수료를 물수합니다. 일부 금지 투자자의 조치의 예시는 다음과 같습니다.

예시 1: 어떤 투자자는 거래소 계좌에 10개의 BTC를 가지고 있습니다. 이 10개 중 투자자는 관리를 위해 3개의 BTC를 트레이더에게 이전하였습니다. 그 후, 투자자는 그의 거래소 계좌에서 8BTC를 인출하였습니다. 이는 3BTC가 현재 계약 조건에 따라 거래할 수 없기 때문에 진행이 불가능합니다. 따라서 위반된 계약은 자동으로 취소되며, 트레이더는 그의 지급액을 전액 받게 됩니다.

예시 2: 어떤 투자자는 거래소 계좌에 10개의 BTC를 가지고 있습니다. 이 10개 중 투자자는 관리를 위해 3개의 BTC를 트레이더에게 이전하였습니다. 그 후, 투자자는 교환 인터페이스를 통해 BTC 9개를 90 ETH와 교환하였습니다. 이는 지정된 회계 통화의 금액이 부족하기 때문에 투자자는 계약을 위반하게 되며, 계약이 취소되어 트레이더는 그의 지급액을 전액 받게 됩니다.

다음은 승인 가능한 거래의 예시입니다.

예시 3: 어떤 투자자는 거래소 계좌에 10개의 BTC를 가지고 있습니다. 이 10개 중 투자자는 관리를 위해 3개의 BTC를 트레이더에게 이전하였습니다. 그 후, 투자자는 자신의 계좌에서 BTC 7개를 인출하여 차를 구매하였습니다. 이는 3개의 BTC가 계약에 위배되지 않고 거래가 가능하기 때문에 허용됩니다.

계약 조기 종료

만약 이 계약에 따른 현재의 재무 결과가 부정적인 경우 투자자는 즉시 계약을 종료할 권리를 얻게 됩니다. 이와 동시에 모든 계약상의 신청 사항 또한 취소됩니다. 원하는 통화 목록에 포함되지 않은 모든 통화는 거래소 가격으로 회계 통화로 전환됩니다.



투자자와 트레이더 간 지불

신탁 관리를 위한 거래를 체결할 때, 결제 통화는 고정됩니다. BTC, ETH 및 USD 중에서 선택할 수 있는 옵션이 있습니다. 예를 들어 계약상 결제를 하기 위하여, 취득한 손익의 계산과 투자자의 거래소 계좌에 있는 모든 통화는 결제 시점의 거래소 가격의 회계 통화로 전환됩니다. 동시에, 이러한 통화의 구매를 위한 거래소의 신청 물량을 고려합니다.

계약이 종료되는 때, 이유와 관계없이, 거래자가 매입한 모든 통화는 거래소 가격으로 지금으로 교환됩니다.

종종 거래자가 목표치를 초과하는 이익을 달성하는 상황이 발생하기도 합니다. 마지막 거래 시기와 목표 이익의 가치에 대한 정확한 시세가 다를 가능성이 있기 때문입니다. 이 경우, 목표 이익이 달성되는 시기만이 아니라, 계약이 종료되고, 투자자는 추가 이익을 받는 동시에, 트레이더의 용역에 대한 대가를 지불하도록 송장을 제공받습니다.

투자자와 트레이더 사이의 계약 예시

트레이더 Bob은 오랫동안 암호화폐 거래처로 거래해 왔고, 많은 돈을 벌었으며, 자신의 거래에서 수익을 얻기 위해 다른 사람의 돈을 관리할 수 있다고 자신하고 있습니다. Bob은 Membrana 웹사이트를 방문하여 자신에게 적합한 계약 매개 변수를 설정합니다.

- 회계 통화 - BTC
- 최소 수량은 12 BTC
- 목표 수익 - 20%
- 최대 손실 - 10%
- 계약 조건 - 30일
- 커미션 - 30%

투자자인 Alice는 Bittrex 거래소에 계좌를 가지고 있습니다. 이 계정에는 10 BTC와 50 ETH를 가지고 있습니다. Alice는 이 자금들 중 일부를 경험이 풍부한 트레이더의 신탁 관리에게 이전할 계획입니다.

Membrana 웹사이트의 트레이더들의 평점에서 Alice는 Bob을 발견하고 그의 프로필에서 관리를 위한 조건을 읽습니다.

Alice가 가장 먼저 주목한 것은 계약을 체결하기 위해서는 최소 12개의 BTC가 필요하다는 것입니다. 하지만 그녀는 10개의 BTC 만 가지고 있습니다. Alice는 메타마스크를 통해 Membrana 웹사이트에서 승인을 받기로 결심하고, Bittrex 웹사이트의 계정으로부터 API 키를 받고, 이 키를 Membrana 웹사이트에 추가합니다. 이제 Alice는 Membrana 웹사이트에 있는 단일 거래 터미널을 통해 자신의 자금을 거래할 수 있는 기회와 함께 신탁 관리로 자금을 이전할 수 있는 기회를 갖게 되었습니다.

Alice는 Membrana 웹사이트를 통해 Bob에게 계약을 진행하자는 요청을 보냅니다. Bob은 24시간 이내에 요청을 받습니다. Alice가 Membrana 스마트 컨트랙트의 주소로 Bob과의 계약 예치금 최대액을 송금합니다. 금액 * 목표 이익 * 거래자의 보생 = 12 BTC * 20% * 30% = 0.72 BTC = 7.2 ETH (시세에 맞게 BTC -> ETH로 전환합니다). Bob은 그의 커미션을 받지 않지만, Membrana 스마트 컨트랙트의 주소에 예약되어 있는 내역을 확인합니다.

한달동안 Bob은 관리 목적으로 이전 받은 금액으로 Membrana 거래 터미널을 통해 거래를 진행합니다. 그와 동시에 Alice의 환전 계좌 (30ETH)에 남아 있는 자금도 그에게 제공되지 않습니다. 그러나, Alice는 다음에도 활용할 수 있습니다. Alice는 거래소에서 15ETH를 인출하여 이것을 필요한 곳에 사용합니다.

Membrana 플랫폼은 20일 후 Bob이 매입한 모든 통화를 BTC 회계 통화에 대해 거래소의 현재 주문량을 고려하여 거래소 가격으로 판매할 경우 14.4 BTC를 받을수 있을 것으로 예상합니다. 이는 밥에 의해 계약의 목표 이익 (12 BTC * 120% = 14.4 BTC)이 달성되었고, Membrana는 자동으로 계약이 종료됨을 의미합니다. Bob이 구매한 모든 통화는 자동적으로 시세를 적용하여 BTC로 교환됩니다.

주의:

- Alice는 다른 트레이더나 심지어 Bob과도 마음대로 다른 계약을 체결할 수 있는 기회가 있었습니다.
- Bob은 Alice의 현재 계약과 함께 다른 투자자들과 계약할 기회를 가졌었습니다.
- Bob은 직접 투자자로 활동하여 다른 트레이더들에게 자금을 이전할 기회가 있었지만, 그러한 자금은 이미 Bob이 소유한 거래소 계좌로 입금된 돈이어야 합니다.



고급 수준의 플랫폼 아키텍처 (베타 버전)

프론트엔드

사용자와 상호 작용되는 Membrana.io 플랫폼의 프론트엔드 서버 시스템은 React 기술을 활용해 만들어진 동적 웹페이지입니다.

인터넷 브라우저는 사용자들이 그들의 이더리움 월렛에서 거래를 할 수 있도록 해주는 메타마스크 플러그인을 사용합니다. 메타마스크는 또한 이더리움과 자동으로 연동되는 web3 오브젝트를 제공합니다.

모바일 기기에서 Membrana 플랫폼은 Cipher나 Toshi 등 이더리움 네트워크의 승인을 지원하는 특수 브라우저를 통해 실행됩니다. 프론트엔드 시스템은 다음의 GitHub에 게시됩니다: <https://github.com/membrane-io/frontend>

백엔드

Membrana.io 플랫폼의 서버 파트는 웹사이트를 통하여 발생하는 사용자의 요청을 처리하는 역할을 진행합니다. 백엔드는 Node.js를 사용하여 생성됩니다.

데이터베이스

시스템에 필요한 모든 정보를 저장하기 위하여 MongoDB 데이터 베이스를 사용합니다. 데이터베이스는 백엔드와 상호 작용합니다.

이더리움 블록체인

Membrana 플랫폼은 이더리움 블록체인을 활용합니다. 스마트 컨트랙트 기술은 신탁 자산 관리를 위하여 투자자와 트레이더 사이의 거래 조건을 모두 통합하는 데 활용됩니다. 또한 트레이더의 거래를 게시하는 과정에서 이러한 거래에 대한 데이터 (해시 합계)를 블록체인에 저장해 향후 데이터의

신뢰성을 확인합니다.

이더리움은 플랫폼의 프론트엔드와 상호 작용합니다. 모든 거래는 특정 사용자의 키로 서명됩니다. Membrana 플랫폼에 접속하기 위하여, 사용자들은 메타마스크나 특별한 모바일 브라우저 (Cipher, Toshi)로 로그인 합니다.

Etherscan.io

웹사이트 <http://etherscan.io>는 사용자들이 스마트 컨트랙트와 블록체인 월렛에 대한 정보를 이더리움 네트워크에서 볼 수 있게 해줍니다. Membrana 플랫폼 내에서 투자자와 트레이더 사이의 거래가 이루어지는 스마트 컨트랙트의 주소는 공개되어 있습니다. 누구나 웹사이트 www.etherscan.io에서 다음 정보를 확인할 수 있습니다.

- 투자자와 트레이더 간 거래
- 트레이더의 향후 커미션 합계를 스마트 컨트랙트로 수령
- 계약 종료 시 트레이더에게 수수료 지급
- 트레이더의 지급에 사용되지 않은 투자자에게의 자금 환급
- 특정 트레이더의 주당 거래 합계, Membrana 시스템의 거래에 대한 정보의 신뢰성 확인

오라클

Membrana 플랫폼에서는, 백엔드에서 블록체인에 기록하기 위하여 특정 거래소 계좌의 거래 데이터, 가상화폐류, 그리고 특정 거래자의 거래 등을 활용하고 있습니다.

Membrane 플랫폼에서 오라클은 특정 거래소 계정의 거래와 잔액에 대한 블록체인 데이터를 입수하여 기록하기 위하여 활용됩니다.

마이크로 서비스

마이크로 서비스는 Membrana 서버에서 작동되며 플랫폼에 시세, 거래소에서 거래 이력, 특정 주문 완료 결과 등의 정보를 제공합니다. 이 접근방식은 사용자가 시스템을 쉽게 확장할 수 있을 뿐만 아니라, 안정적인 작동을 보장할 수 있게 합니다. 예를 들어, 이러한 목적의 각 마이크로 서비스는 독립적인 서버에서 복제될 수 있습니다.

스마트 컨트랙트 코드

Membrana 플랫폼에 대한 스마트 컨트랙트는 공식 GitHub에서 확인할 수 있습니다.

링크 : <https://github.com/membrana-io>.

MembranaInstance.sol(베타버전) – 주요 스마트 컨트랙트 코드의 예시는 다음과 같습니다. 이 코드는 투자자와 트레이더 사이의 신뢰 관리 거래를 체결하고 실행하도록 구현되어 있습니다.

```
pragma solidity ^0.4.15;
contract MembranaDeals {address public be = 0x873A2832898b17b5C12355769A7E2DAe6c2f92f7};
enum state { paid, verified, halted, finished}
enum currencyType { USDT, BTC, ETH}
struct Deal {
    state currentState;
    uint start;
    uint deadline;
    uint maxLoss;
    uint startBalance;
    uint targetBalance;
    uint amount;
    currencyType currency;
    string investor;
    address investorAddress;
    string trader;
    address traderAddress;
}
Deal[] public deals;
function MercatusDeals() public payable{
    revert();
}
modifier onlyBe() {
    require(msg.sender == be);
    _;
}
modifier inState(uint dealId, state s) {
    require(deals[dealId].currentState == s);
    _;
}
function getState(uint dealId) public constant returns (uint) {
```



```

        return uint(deals[dealId].currentState);
    }
    function getStart(uint dealId) public constant returns (uint) {
        return deals[dealId].start;
    }
    function setVerified(uint dealId) public onlyBe inState(dealId, state.paid) {
        deals[dealId].currentState = state.verified;
    }
    function setHalted(uint dealId) public onlyBe {
        require(deals[dealId].currentState == state.paid || deals[dealId].currentState == state.verified);
        require(deals[dealId].amount != 0);
        deals[dealId].amount = 0;
        deals[dealId].traderAddress.transfer(deals[dealId].amount);
        deals[dealId].currentState = state.halted;
    }
    function getSplit(uint finishAmount, uint startBalance, uint targetBalance, uint amount) public pure
    returns (uint) {
        return ((finishAmount - startBalance) * amount) / ((targetBalance - startBalance) );
    }
    function setFinished(uint dealId, uint finishAmount) public onlyBe inState(dealId, state.verified) {
        require(deals[dealId].amount != 0);
        deals[dealId].amount = 0;
        if(finishAmount <= deals[dealId].startBalance){
            deals[dealId].investorAddress.transfer(deals[dealId].amount);
        }else if(finishAmount>deals[dealId].targetBalance){
            deals[dealId].traderAddress.transfer(deals[dealId].amount);
        }
        else{
            uint split = getSplit(finishAmount, deals[dealId].startBalance, deals[dealId].targetBalance,
            deals[dealId].amount);
            deals[dealId].traderAddress.transfer(split);
            deals[dealId].investorAddress.transfer(deals[dealId].amount - split);
        }
        deals[dealId].currentState = state.finished;
    }
    function getDealsCount() public constant returns (uint){
        return deals.length;
    }
    function () external payable {
        revert();
    }
    function makeDeal(uint _duration, uint _maxLoss, uint _startBalance, uint _targetBalance, uint _
    amount, string _investor, address _investorAddress, string _trader, address _traderAddress, uint offer,
    uint _currency)
    payable public {
        require( _currency >= 0 && _currency < 3 );
        require(msg.value == _amount);
        deals.push(Deal({
            currentState: state.paid,

```

```
        start: now,  
        deadline: 0,  
        maxLoss: _maxLoss,  
        startBalance: _startBalance,  
        targetBalance: _targetBalance,  
        amount: _amount,  
        currency: currencyType(_currency),  
        investor: _investor,  
        investorAddress: _investorAddress,  
        trader: _trader,  
        traderAddress: _traderAddress  
    ));  
    deals[deals.length-1].deadline = now + _duration * 86400;  
    spawnInstance(msg.sender,deals.length-1, now, offer);  
}  
event spawnInstance(address indexed from, uint indexed dealId, uint start, uint offer);  
}
```

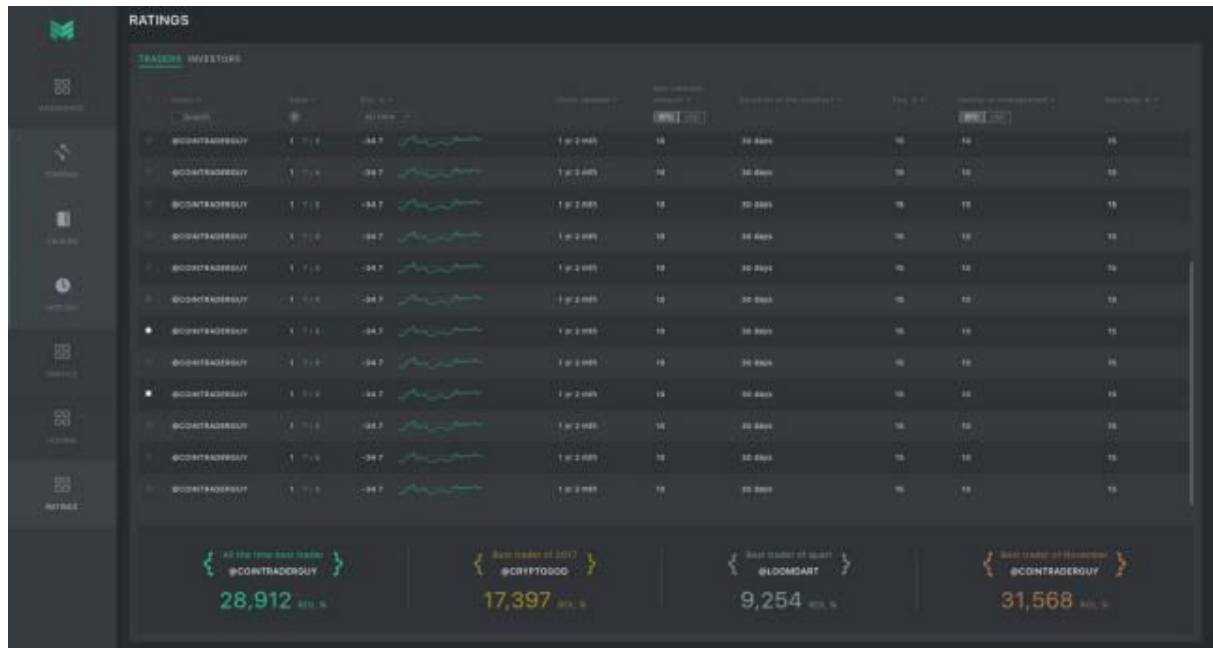
MEMBRANA 플랫폼 인터페이스 (베타 버전)

현재 Membrana 플랫폼의 베타 버전은 개방되어 있으며 이용이 가능합니다.

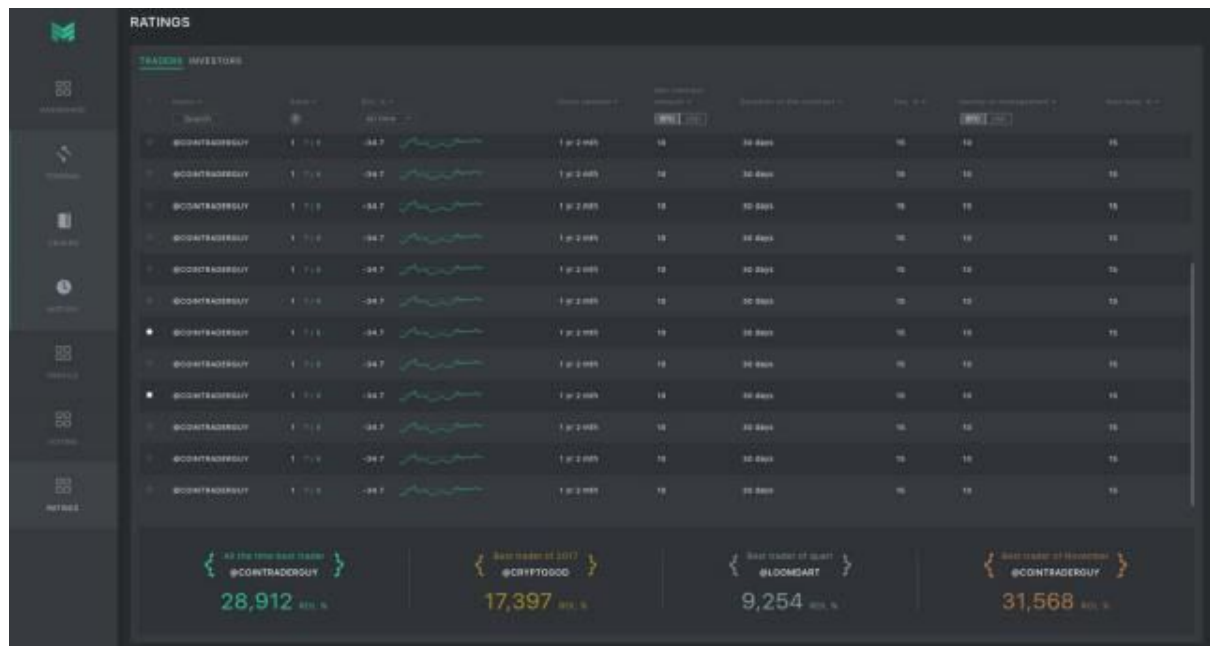
beta.membrana.io

아래는 인터페이스의 스크린샷:

트레이더를 위한 단일 거래 터미널



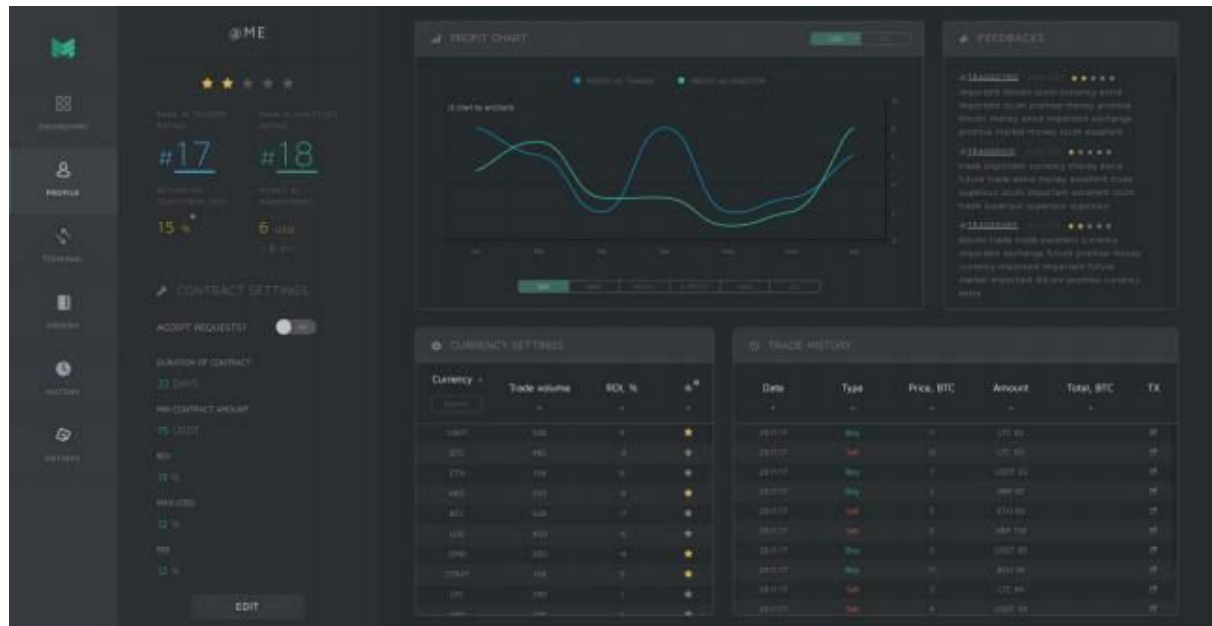
트레이더 등급



투자자의 개인 계정



트레이더 프로필



관심사항 기능

Membrana 플랫폼의 API

Membrana 플랫폼의 전체 기능은 웹사이트의 사용자 인터페이스뿐만 아니라 API를 통해서도 제공됩니다. API는 거래자들이 신탁 관리를 위해 받은 자신의 자금 또는 자산을 자동으로 거래할 수 있게 해줍니다.

Membrana API는 트레이딩 봇들에게 플랫폼의 잠재적 사용자들의 영역을 크게 확장시킬 트레이더의 역할을 할 수 있도록 능력을 제공할 것입니다. 이는 또한 트레이딩 봇들이 추가 자금을 조달하고 더 많은 이익을 창출할 수 있는 좋은 기회를 제공합니다.

전문화 된 뉴스 피드

트레이더들은 트레이딩 터미널의 표준 지표와 동시에 각 통화에 대한 뉴스 피드를 볼 수 있는 기회를 갖게 되어 트레이더의 시장 상황을 완전히 분석할 수 있도록 능력을 강화합니다. 이 정보는 기술 지표뿐만 아니라 외부로부터 받은 정보로부터도 획득됩니다.

뉴스 피드는 편리한 설정을 제공합니다. 트레이더는 관심있는 뉴스를 선택할 수 있으며, 자신의 암호화폐 거래 결정에 영향을 미치는 관련 정보만 볼 수 있습니다.

신뢰할 수 있는 이더리움 월렛 펀드 관리

투자자는 신탁 관리를 위한 자산을 자신의 거래소 계좌에 있는 자금뿐만 아니라 ETH도 자신의 이더리움 지갑에서 트레이더에게 양도할 수 있게됩니다. 이렇게 되면 거래소 등록은 물론 신원 확인, 환전 계좌 자금 송출 등의 프로세스가 간소화 되기 때문에 투자자의 신탁관리 계약을 체결하는 절차가 간소화 됩니다. 이더리움 월렛 안에 있는 투자자의 자금은 Membrana 플랫폼의 특별한 스마트 컨트랙트로 전달되고 계약 기간이 끝날 때까지 예치됩니다. Membrana 플랫폼은 거래소 계좌에 있는 자신의 자금을 트레이더에게 양도합니다.

계약이 종료되면 투자자의 자금은 이익과 함께 ETH의 이더리움 월렛으로 반환됩니다. 트레이더는 또한 그 보상으로 ETH를 받습니다.

이것은 Membrana 플랫폼이 모든 거래소에 자금을 가지고 있기 때문에, 트레이더가 어떤 지원 거래소에서든 투자자들의 돈을 거래할 수 있는 기회를 제공합니다.

트레이더의 손실 보상이 가능한 계약

투자자와 트레이더 사이의 신뢰도를 높이기 위하여, Membrana 플랫폼은 트레이더가 투자자의 손실을 가능한한 부분적으로 보상하기 위하여 미리 자금을 예치할 수 있도록 지원합니다.

만약 계약이 완료되었을 때, 트레이더의 거래 결과가 부정적이면, 이 자금은 투자자에게 이전됩니다.

비록 투자자의 손실에 대한 보상이 전체 손실에 비하여 비율이 적다 하더라도, 이 정책은 성공적인 거래에 대한 트레이더의 관심을 크게 증가시키고 각각 트레이더에 대한 투자자들의 신뢰를 형성하게 됩니다.

예시: 트레이더인 Alice가 목표 이익을 20%, 최대 손실을 10%로 30일동안 10 BTC를 진행할 준비가 되어있습니다. 투자자의 신뢰와 제안에 대한 관심을 높이기 위하여 앨리스는 가능한 손실의 30%를 보상할 준비가 되어 있습니다. 투자자인 Bob은 Alice와 거래를 하기로 하고, Alice는 10 BTC의 관리를 받게 됩니다. 20일 후에, 계약을 이행하지 못하여 조기 종료됩니다. Bob은 계약 당시 예약되어 있던 Alice의 자금에서 손실에 대한 보상으로 $1\text{BTC} \times 30\% = 0.3\text{ BTC}$ 를 받습니다. 따라서 Alice는 성공적으로 종료되지 않은 계약으로 인한 손실의 일부를 만회하기 위하여 활용하였습니다.

조건부 거래소 주문

Membrana 플랫폼은 트레이더가 거래소에서 조건부 주문을 가능할 수 있게 해줍니다. 조건부 주문은 현재 가격, 거래량 또는 기타 시장 매개 변수와 같이 미리 결정된 특정 조건을 충족할 때 거래소에 배치됩니다.

매니저 계약: 투자자와 포트폴리오 매니저의 계약

투자자는 성공적인 트레이더를 선택하는 일을 거래자를 선택하는 일을 다른 경험이 풍부한 투자자에게 위임할 기회를 갖습니다. 그렇기 때문에 특별한 유형의 숙련된 투자자와 매니저 계약이 만들어 질 수 있습니다. 경험이 풍부한 투자자는 고객의 자금을 관리하기 위하여 트레이더에게 이전함으로써 고객을 대신하여 트레이더와 거래를 할 수 있는 기회를 얻습니다. 이 경우 경험이 풍부한 투자자는 수수료를 받게 됩니다.

Membrana 토큰의 경제

Membrana 플랫폼에는 자체 이더리움 토큰인 MBR가 활용됩니다. MBR은 플랫폼 기능에 관련된 유틸리티 토큰의 유형을 말합니다. Membrana 사용자는 메인 시스템 기능에 무료로 접속할 수 있습니다. 주요 기능은 지원되는 거래소에서 단일 거래소를 통한 거래, 적절한 투자자 또는 트레이더를 찾는 것, 투자자와 트레이더 사이의 계약을 체결하는 것, 그리고 종료시까지의 계약 지원 등입니다.

다중 계약 요청

트레이더는 서로 다른 투자자들과 다양한 신탁 관리 계약을 진행할 수 있습니다. 어느 시점에 와서는 모든 계약에 대한 동일한 주문이 가능합니다. 예를 들어, 거래자들의 관점에서 봤을 때, 모든 계약에 대해 비트코인을 구매해야 합니다. 여기서 Membrana 플랫폼에는 다수의 계약에 자동 주문할 수 있는 기능이 제공됩니다. 또한 트레이더는 거래 목록에서 특정 계약을 선택하여 주문 매개 변수를 지정할 수 있습니다.

프로젝트 로드맵

자세한 내역은 [Membrana.io 프로젝트 로드맵](#)에서 확인할 수 있습니다.

Q1 2017. 개발 컨셉

이번 분기는 암호화폐 자산 신탁 관리 및 개념 개발로 시작되었습니다. 우리는 신탁 시장의 주요 다음과 같은 문제에 대한 해결책을 찾았습니다:

- 투자 자본에 대한 통제력 상실
- 거래 중재자 참여와 관련된 위험
- 투자자와 트레이더간의 신뢰 부족
- 트레이더의 과거 거래 결과의 유효성에 대한 확신 부족
- 거래 결과 조작 가능성

Q2 2017. 컨셉 타당성 검증

2분기에는 참가자의 행동 시나리오가 다른 상황에서 개념 실현 가능성에 대한 완전한 검증을 기준으로 진행되었습니다. 현재 알려져 있는 모든 잠재적 위험에 대한 연구와 분석과 함께 투자 자산의 안전을 보장하기 위하여 내부 프로세스의 최적화 수행에 대하여 추가 검증이 진행되었습니다.

Q3 2017. 인터페이스 제작

이번 분기에는 플랫폼 인터페이스 개발이 진행되었습니다.

Q1 2018. 알파 버전

2017년 마지막 분기는 플랫폼 아키텍처의 개발 및 다음과 같은 기능을 가진 알파 버전 구현에 중점을 두었습니다.

- 메타마스크를 사용한 승인
- Bittrex 암호화폐 거래소의 API 키 추가
- 신탁 요청 전송, 수신
- 내부 거래 단말기를 통한 암호화폐 거래소에 발주
- Etherscan 시스템에서 계약 조건 확인
- 계약서 인도물에 기반한 트레이더 등급 업데이트
- 이더리움에서 권한 부여가 지원되는 브라우저를 사용하여 사이트의 모바일 버전 지원 (Cipher, Toshi)
- Q4는 알파 버전에 대한 비공개 테스트로 마무리

Q2 2018. 베타 버전

이번 분기는 업데이트 된 기능을 갖춘 베타 버전의 공개 테스트로 시작됩니다.

- 관리를 위해 여러명의 트레이더에게 이전할 수 있는 거래소 계좌의 통화 잔액 분포
- 계약서 인도물에 기반한 트레이더 및 투자자의 등급
- 트레이더의 거래 결과를 블록체인에 기록
- Binance 및 Kukoin 거래소 지원
- 손실 방지를 조건으로 특정 통화를 계약 종료 시 시장 가격으로 매도
- 실제 유저의 유입, 트레이더 공모전 개최

Q4 2018. 주요 업데이트

- Membrana 플랫폼의 기능에 대한 액세스를 제공하는 API(Application Programming Interface)
- 지원되는 거래소를 자동 모드로 거래할 수 있는 기능
- 최대 손실 – 최대 손실에 도달하면 신탁 관리 계약이 중지되고 자금 접근이 거부됨
- 신뢰 로그 – 투명성과 ROI 증명을 위하여 모든 트랜잭션을 블록체인에 저장

Q1 2019. 조건부 주문 및 UI 업데이트

- 더 나은 UI/UX를 위해 터미널에서 업데이트.
- 조건부 주문 (한계, 시장, 목표, 정지)
- 잔고에 락을 지정하지 않고 주문 설정 허용

Q2 2019. 고급 기능

- 보안 관련 계약 트레이더는 투자자의 손실 가능성을 부분적으로 회복하기 위하여 컨트랙트에 자신의 자금을 넣어둡니다.
- .. 통화 뉴스 피드. 거래 단말기에서 가격 변동 수치와 함께 특정 통화에 관한 뉴스를 동시에 검토할 수 있는 능력
- 분석 – 특정 화폐의 볼륨을 추적하고 정보를 통한 안내 시스템 제공
- 포트폴리오 관리 – 여러 거래소에서 작업하는 동안 포트폴리오와 함께 작동되는 편리한 도구
- Membrana 관리 펀드 – MBN 토큰 사용으로 마진 계약 액세스

Q4 2019. 최종 배포

Membrana.io 플랫폼은 완전한 기능을 갖추어 출시되며, 시장에 진입할 준비가 되어 있습니다:

- API 키를 통한 대부분의 암호화폐 거래소 지원
- 투자자와 트레이더 간의 거래소 문자 메시지
- DEX 거래소 연결
- Membrana API를 발행하여 자동으로 지원되는 거래소를 지원함 (트레이딩 봇)
알고리즘 주문 : Iceberg 주문, Peg 주문, Trailing 주문 등

Q2 2020. 추가 기능

- 투자자가 이더리움 월렛에서 트레이더에게 자금을 이전할 수 있는 기회 (암호화폐 거래소에 대한 계좌를 보유할 필요가 없음)
- 매니저 계약: 최고의 트레이더를 찾는 일을 하는 투자자와 포트폴리오 매니저 사이의 계약
- API 프로토콜을 통한 상위 50개 거래소와 연결
- 플랫폼에 구현된 Algo Builder
- 마진 거래를 지원하는 거래소
- 탈중앙화 거래소 거래 지원

Q1 2021. 주식 시장 확대

- 브로커 및 카운터 패밀리에 관한 정보
- 등급 시스템, ROI, 위험 요소
- 브로커 시스템에 대한 API 연결
- 주식, 선물, 채권, 보증 지원
- 디지털 공유 시장과 보안 토큰

팀



Yuriy Gerasimov, CEO

Peter the Great St.Petersburg Polytechnic University 석사 학위, SxS 법인 마케터, 러시아에서 몇몇 성공적인 기업을 설립하였으며, 여러 블록체인 이벤트의 크립토 분석가이자 연설가

www.linkedin.com/in/jerry-gerasimov



Vyacheslav Mychkin, CTO

KNU 개발 실무에서 “컴퓨터 과학” 학위 취득. 독일 회사에서 근무한 경험을 가진 풀스택 개발자, JS, React 모바일 개발 (Android, iOS) 분야의 전문가, 고부하 시스템을 위한 자체 알고리즘 개발

www.linkedin.com/in/vyacheslavmychkin



Evgeniy Buyev, COO

석사학위, 엔지니어, Peter the Great St.Petersburg Polytechnic University 졸업, 트레이더, 포커 AI 개발자, 블록체인 열정가

www.linkedin.com/in/eugenebuev

www.facebook.com/vtbeuro



Michael Creadon, CBDO

Colombia University의 석사학위, 4Rev의 NY CEO. 2500만 달러 규모의 프로펠러 회사의 트레이더 겸 CEO로 20년간 CME의 이자율 리스크를 관리

www.linkedin.com/in/michael-creadon-31404a30



Kirill Romanov, CMO

통신학 학사, 디지털 기관의 설립자, 웹 개발 및 모바일 광고상 수상, ibeacon 및 로열티 프로그램에서 스타트업을 시작, 2009년부터 마케팅 경력

www.linkedin.com/in/kirill-romanov-4u



Maxim Khukhro, 아키텍처 설계자

Novosibirsk 주립대학 석사 학위, Intel corp에서 팀 리더로 8년, IT 분야에서 18년 경력, AI, 신경망 네트워크 개발, 머신 러닝 전문가

www.linkedin.com/in/maximkhukhro



Shahboz Urinbaev, BDOF

마케팅 책임자, 팀장, 10년 비즈니스 개발자 경력, 당시 15개 이상의 전략과 2개의 클라우드 세일즈 크립토 프로젝트 진행 (Cryptics, Adab)

www.linkedin.com/in/shahboz-urinbaev-844a6179



Igor Knyazev, 백엔드와 Solidity 개발자

KNU '컴퓨터 과학' 학위. 10년 이상 IT 업계 경력으로 여러 명의 정보 보안 전문가 CTF 경연대회 우승자. 2015년 블록체인 관련 프로젝트에 크게 기여함

www.linkedin.com/in/igor-knyazev-237914148



Ilya Miroshnichenko, 프론트엔드 개발자

NNGY Lobachevskyi 대학 학사, 러시아 중앙은행, FSD등 VPS 기업용 내부 보안 시스템 관리를 위한 웹 콘솔 개발자, 프론트엔드 개발자로 “Security Code” 회사와의 실무 경험

www.linkedin.com/in/ilya-miroshnichenko-a10523a5



Alex Badmashkaev, 풀 스택 개발자

Adyghe State University의 소프트웨어 공학 학사 학위, 소프트웨어 엔지니어, 경험 많은 백엔드 개발자, IT 7년 경력

www.linkedin.com/in/alexbadm

www.github.com/alexbadm



Victoriya Matveyeva, QA 엔지니어

Webcademy school 졸업, 프론트엔드 개발자, HTML5, CSS3, jQuery

www.linkedin.com/in/viktoriya-matveyeva-91183a122



Sonder S. Lim, 아시아 지역 브랜드 관리자

암호화폐 자산 지원가, 마케팅 에이전시 설립자, 전략 콘텐츠 관리자

www.linkedin.com/in/sonderslim



Adam Joseph, 인도 지역 브랜드 관리자

전략 마케팅, 비즈니스 개발 관리자

www.linkedin.com/in/adam-joseph-266356153/



Williams Kyei, 한국 지역 브랜드 관리자

한국 스타트업 및 블록체인 에코 시스템, 디지털 미디어 및 마케팅 전문가, 웹 디자이너 및 콘텐츠 제작자 경험이 있는 전문가, 스포츠 매니지먼트에서 8년 이상 경험을 쌓음

www.linkedin.com/in/nomadwilliams

어드바이저



Paresh Masani, 기술 어드바이저

Paresh는 골드만 삭스의 글로벌 투자 연구소의 전직 이사. 블록체인 기술, 보안, 암호화폐, 복잡한 중단간 시스템 개발 등에 정통한 전문가

www.linkedin.com/in/masaniparesh



Marie-Noel Nsana, 재무 어드바이저

금융 및 은행업종 15년 경력, 금융 세탁 방지 위원, 비트코인과 크립토 전문가 (c4 등급)

www.linkedin.com/in/marie-noel-nsana-120a433



Denis Druzhkov, 기술 어드바이저

러시아 금융분석회사 Umnick에서 12년간 근무, 1990년부터 2002년까지 국제 펀드 VPS의 전무이사. 2011년 초 Casys Capital Management의 투자 이사로 취임

www.linkedin.com/in/denis-druzhkov-88a13082

www.facebook.com/denis.leavy



Sam Kushed, 기술 어드바이저

블록체인 개발자. Salus Coin의 창립자, Coinbar Crypto-fund의 회장

www.bitcointalk.org/index.php?topic=1317448.0



Alexander Noxon, 기술 어드바이저

2015년부터 전문 크립토 개발, 10년 이상의 IT 경력자

www.github.com/noxonsu

www.twitter.com/noxonsu



Anton Mogilevskiy, 법률 어드바이저

Mikhail Speransky 법학부 박사 학위이며 국제 로펌 Norton Rose와 Allen & Overy에서 14년 이상 근무 경력

www.linkedin.com/in/anton-mogilevskiy-1167471



Hans Choi, 기술 어드바이저

OpenChain CEO/개발자, 블록체인 개발자 & 아키텍처, 핀테크 전문가

www.linkedin.com/in/hans-namkyu-choi-407618102



Yan Stolyar, 재무 어드바이저

Telecom CEO, IT 및 프로젝트 관리 전문가, MBA 경영 전문가

www.linkedin.com/in/yanstolyar

www.facebook.com/yan.stolyar



웹사이트



플랫폼 베타 버전



GitHub



텔레그램



미디어



info@membrana.io