Gwangju Institute of
Science and Technology

School of Electrical Engineering and Computer Science

DeFi 강의 및 실습

<한국금융ICT융합학회 추계워크샵: 빅뱅하는 디지털금융—디파이 NFT 메타버스>
일시: 11.27토-28일

광주과학기술원 (GIST)
INFONET 연구실
이흥노 교수
최해웅 박사과정
박하영 석사과정

강의 목차

1. 스마트 컨트랙트, 토큰 정의 (1)
2. DEX 개념 (10)
   - 기존 중앙거래소 와 DEX 차이점
   - Automatic Market Maker (AMM)
   - 스왑, 가격영향, 슬리피지 예제
   - 수수료
   - 유동성 제공 및 제거
   - 비영구적 손실 설명 및 예제
3. DEX 서비스 예시 (2)
4. Flash loan 개념 (2)
5. Flash loan attack (1)
6. Lending 프로토콜 소개 및 예시 (4)
7. Stable coin 소개 및 예시 (DAI) (1)
8. SWAP 실습

스마트 컨트랙트, 토큰 정의

이더리움: 비트코인과 비교하면, 네트워크 상에 구현할 수 있는 계약의 종류가 무궁무진함.

- 스마트 컨트랙트의 작성과 배포, 사용이 편리함. 대부분의 DeFi 서비스가 이더리움 프로토콜 기반으로 동작함.

스마트 컨트랙트: 계약을 프로그램으로 작성하여 블록체인에 등록함으로써, 정확히 프로그래밍 한 그대로 실행되는 것을 보장함.

ERC20 토큰: 이더리움 상에서 스마트 컨트랙트를 사용하여, 발행, 교환, 소각 과정이 ERC20 표준에 명시된 대로 작동하는 가상 자산. 암호화폐 프로토콜 속에 있는 또다른 암호화폐.

DEX

기존 중앙거래소 와 DEX 차이점

- 블록체인은 중앙 서버나 기관이 존재하지 않기 때문에 사용자들이 네트워크를 유지할 필요가 있다.
- 네트워크를 유지하는데 기여하는 사용자들은 보상으로 암호화폐를 받을 수 있다.
- 암호화폐 간 교환을 위해 거래소가 필요
- 중앙화된 거래소를 사용할 경우 거래소의 신뢰성 문제, 단일 장애점 공격의 위험 같은 문제가 존재함.
- 같은 블록체인 프로토콜을 사용하는 토큰간의 교환의 경우, 해당 블록체인 상의 스마트 컨트랙트를 이용하여 탈중앙화된 거래소 (DEX) 를 구현하여 이와 같은 문제를 해결할 수 있음.

DEX

Automatic Market Maker (AMM) 개념 설명

- 전통적인 시장에서는 오더북(order book) 방식 사용.
- 오더북 방식은 판매자 및 구매자가 직접 거래 희망 가격을 제시하고, 적절한 거래 희망가를 제시한 판매자와 구매자 간에 거래가 체결되는 방식이다.
- 하지만 희망 판매가와 희망 구매가가 맞지 않을 경우 거래가 체결이 무한정 지연될 수 있고, 거래가 체결되기 전 까지 제시된 거래 희망가와 물량 정보를 계속 저장 및 추적해야 하므로, 용량이 한정된 블록체인 환경에 적절치 못함.
- 대신에, 현재 거래소에 예치된 유동성 양에 따라 가격이 자동 결정되는 AMM 방식이 많이 사용됨.
- 유동성 제공자는 유동성 (토큰 A 및 토큰 B) 를 거래소에 위탁함. 사용자가 거래소에 거래를 요청하면, AMM 알고리즘에 의해 결정된 가격으로 즉시 거래가 체결된다.
- 장점: 가격이 알고리즘에 의해 자동 결정되며, 거래가 즉시 체결됨.
- 단점: 거래량에 따라 거래 가격이 달라지는 price impact 현상, 거래 요청 시점과 거래 체결 시점의 가격이 달라지는 slippage 현상이 일어날 수 있음

DEX

Automatic Market Maker (AMM) 개념 설명

- 가장 유명한 AMM 알고리즘은 Prediction market에서 주로 사용된 Logarithmic market scoring rules
- 그러나 이는 리소스가 많이 필요해 블록체인 상에서 적절하지 않음.
- 블록체인상에서는 주로 Constant Function이 사용됨
- 유니스왑과 같은 주요 DEX들에서는 Constant Product function 을 사용한 AMM (CPMM) 을 사용함

CPMM 예제

- 토큰 A 와 토큰 B가 예치되어 있는 유동성 풀(pool) 을 가정해보자.
- x 가 토큰 A의 예치량이고, y 가 토큰 B의 예치량이라고 하면, 거래 전후로 두 토큰의 예치량의 곱 k 가 유지되도록 가격을 결정한다. 즉,

$$x \cdot y = k$$

DEX

DEX 스왑 예제: 토큰 A 가 10개, 토큰 B 가 20 개 예치된 풀에 토큰 A를 1개를 판매하면 토큰 B를 몇 개 받을 수 있을까?

sol)

$$x = 10,\ y = 20$$

- 받을 수 있는 토큰 B의 양을 y_{out} 이라고 하면, 스왑 거래가 체결된 이후 풀의 예치량은 다음과 같음

$$x' = (10+1),\ y' = (20 - y_{out})$$

- 스왑 전후로 예치량의 곱이 일정해야 하므로,

$$10 \times 20 = 11 \times (20 - y_{out})$$

$$y_{out} = 20 - \frac{200}{11}$$

$$= 1.82$$

- 즉, 1.82 개의 토큰 B를 받을 수 있다.

DEX

Price Impact 예제

- 앞선 예제에서, 예치된 토큰 A 및 토큰 B의 비율은 1:2 이다. 직관적으로 생각했을 때, 토큰 A를 1개 지불하면 토큰 B를 2개 얻을 수 있어야 하는데, 실제 결과는 1.82 개만 얻을 수 있었다. 이는 예치량 대비 거래량이 너무 많아서 일어나는 price impact 현상이다.
- 같은 예제에서, 토큰 A 의 초기 예치량이 1000개, 토큰 B의 초기 예치량이 2000개라고 가정해보자.

스왑 전: $x = 1000,\ y = 2000$

스왑 후: $x' = 1001,\ y' = 2000 - y_{out}$

$y_{out} = 1.998$ 개의 B 토큰을 받을 수 있다.

- 즉, 풀의 예치량이 많을 수록 거래 가격은 예치량 비율과 비슷해지는 것을 확인할 수 있다.

DEX

슬리피지 예제

- 앞선 스왑 예제에서, $x = 10, y = 20$ 인 토큰 A/B 풀에서, 토큰 A를 1개 지불하여 토큰 B를 1.82 개 얻을 수 있음을 계산할 수 있다.
- 이 계산에 기반하여, 사용자가 토큰 B를 1.82개 얻을 것을 예상하고 스왑 거래를 등록할 것이다.
- 하지만, 네트워크 딜레이, 혹은 먼저 제출된 스왑거래로 인해 사용자가 거래 할 때의 실제 교환비가 달라질 수 있다.
- 실제 교환 결과로 사용자가 예상보다 많은 토큰을 얻는다면, 아무런 문제가 없으므로 그대로 실행해도 된다.
- 하지만 실제 교환 결과값이 예상 결과값 보다 적다면, 손실이 일어나게 된다.
- 사용자는 이러한 손실 (슬리피지) 의 허용 한도를 설정하여, 일정 이상의 슬리피지가 일어날 경우 스왑 거래 자체를 취소하도록 할 수 있다.
- 단 슬리피지 허용 오차를 너무 작게 설정할 경우, 스왑 거래가 지연될 수 있다.

DEX

수수료

- 위와 같은 DEX 가 원활히 운용되려면, DEX에 예치된 유동성이 많아야 한다.
- 유동성을 확보하기 위해서 유동성 제공자에게 보상을 주어야 한다.
- 이에 더해, DEX 운영 비용도 충당해야 한다.

해결법: 거래마다 수수료를 징수하여 유동성 제공자에게 보상(유동성 제공 수수료) 및 DEX 운영진에게 지불 (프로토콜 수수료) 한다.

- Uniswap 의 경우 사용자에게 입력 토큰의 0.3%를 수수료로 징수하여 모두 유동성 제공자에게 보상으로 지불한다. (현재 프로토콜 수수료는 없음.)
- Pancakeswap 의 경우 사용자에게 입력 토큰의 0.25%를 수수료로 징수한다. 그 중 0.17% 는 유동성 제공자에게 보상하고, 나머지 0.03%는 프로토콜 수수료, 0.05% 는 CAKE (거버넌스 토큰) 의 소각에 사용.

DEX

유동성 제공 및 제거

- 유동성을 제공하면, 현재 풀의 예치량 대비 지분 비율량에 해당하는 LP 토큰을 얻게 된다. 자신의 지분 비율은 (자신의 LP토큰량)/(LP토큰의 총 발행량).
- 제공중이던 유동성을 제거하면, 현재 풀의 총 토큰중 자신의 지분 비율 만큼 출금한다. LP토큰은 소각한다. 유동성 중 일부만 제거하는 것도 가능.
- 유동성을 제공하고 있는 도중, 스왑 거래가 일어날 경우 유동성 제공 수수료가 풀에 적립되어 풀의 전체 가치가 상승한다.
- 제공중이던 유동성을 제거할 경우 풀 가치의 상승비율 만큼의 차익을 실현할 수 있다.

DEX

비영구적 손실

- 스왑 수수료가 풀에 계속 적립되므로, 토큰의 가치가 일정하게 유지된다면 유동성 제공으로 차익실현이 가능함
- 하지만 토큰의 가치가 급격히 변할 경우, 유동성 제공으로 인해 손실을 입을수 있음
- 이 손실은 토큰 가치가 이전 수준으로 다시 복구될 경우 회복될 수 있음
- 이를 비영구적 손실이라 지칭함

DEX

비영구적 손실 예제

- 토큰 A가 10개, 토큰 B가 20개 예치된 풀을 가정하자. ($x = 10, y = 20, xy = 200$)
- 그 중 토큰 A 1개, 토큰 B 2개는 자신이 제공한 유동성이다. 즉, 풀의 지분 10% 를 가지고 있다.
- 토큰 A의 가격이 $10, 토큰 B의 가격이 $5 라 가정하자. 그렇다면 풀에 예치된 내 자산의 총 가치는 ($10 * 1) + ($5 * 2) = $20 이다.

이 때, 토큰 A의 가격이 $1로 폭락한다고 생각해보자.

- 만약 내가 유동성 제공을 하지 않고, 토큰 A 1개와 토큰 B 2개를 그대로 가지고 있었다고 가정하면, 내 자산의 총 가치는 ($1 * 1) + ($5 * 2) = $11 가 되었을 것이다.
- 폭락 직후, 풀의 유동성을 즉시 제거한다면 내 자산의 총 가치는 여전히 $11일것이다.

하지만 토큰 A가 폭락한다면, 많은 사람들이 스왑 거래를 통해 토큰 A를 팔고 토큰 B를 사갈 것이다.

- 풀의 예치량 비율이 가격 비율 (1:5) 에 가까워 질 때 까지 교환이 진행될 것이므로, 평형 상태에 도달한 풀의 최종 예치량은 다음과 같을 것이다.

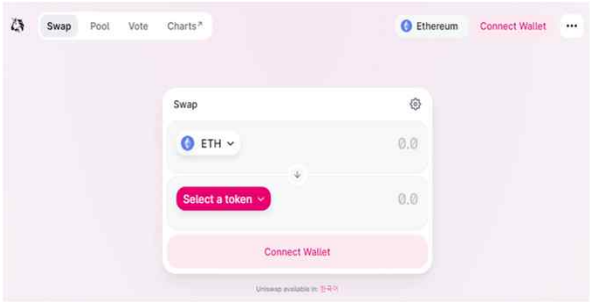
$x = 31.56, y = 6.32$ (xy=200, 수수료가 없다고 가정)

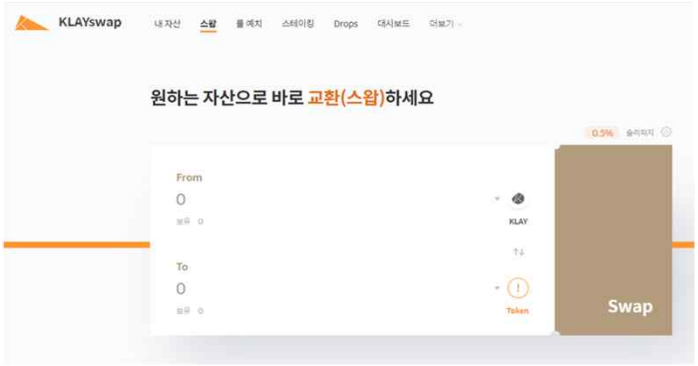
- 이 풀에 대한 내 지분은 10% 이므로, 나의 자산은 토큰 A 3.156 개 및 토큰 B 0.632개이다. 총 자산 가치는 ($1 * 3.156) + ($5 * 0.632) = $6.32 가 되었다.
- 즉, 유동성 제공을 하지 않았을 때의 자산가치에 비해 $4.68 의 손실이 발생하였다.

DEX 서비스 예시

유니스왑: https://app.uniswap.org/#/swap

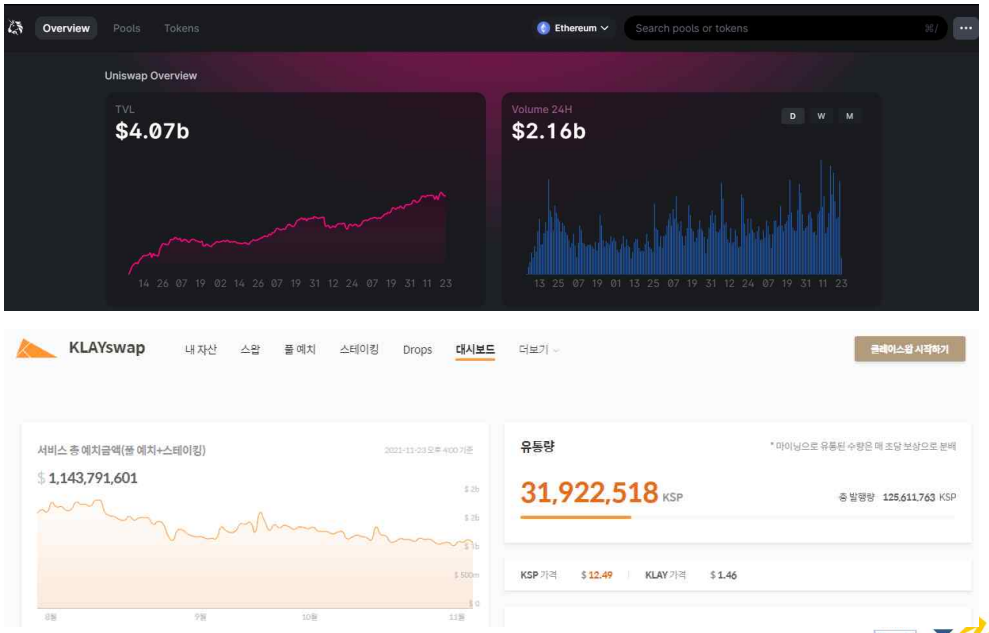
클레이스왑: https://klayswap.com/exchange/swap





DEX 서비스 예시

총 예치량 (Total Value Locked; TVL) : 40억 USD (Uniswap), 11억 USD (Klayswap)



Flash Loan

사용자가 유동성 풀에 토큰 A를 입금하고, 토큰 B를 출금하는 시나리오에서:

- 사용자가 풀에 토큰 A를 입금한 후, 그에 상응하는 가치의 토큰 B를 출금해가는 과정이 하나의 트랜잭션에서 처리됨.
- 하지만, 사실 하나의 트랜잭션에서 처리되기만 한다면, 사용자가 풀에서 토큰 B를 출금한 이후, 그에 상응하는 가치의 토큰 A를 풀에 입금 해도 문제 없음.
- 여기에 더해, 사용자가 토큰 B를 출금하여, 다른 탈중앙 서비스를 통해 토큰 A를 얻은 이후, 그 토큰 A를 풀에 상환할 수 도 있음.
- 즉, 사용자가 토큰 A를 전혀 가지고 있지 않은 상황에서도, 토큰 B를 빌려 차익거래로 토큰 A를 얻어 상환하는 대출이 가능함.

단, 대출과 상환을 하나의 거래로 묶어서, 상환에 오류가 발생하면 대출또한 무효가 되도록 해야함.

하나의 거래 안에 대출과 상환이 동시에 이루어지므로 플래시론 이라는 이름이 붙음.

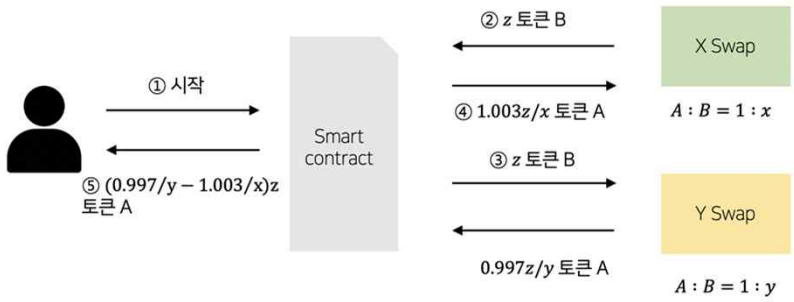
플래시론을 사용하면 아무런 자산이 없는 상태에서도 차익거래가 가능함. 차익거래가 쉬워지므로 여러 DEX 간의 가격안정화에 도움이 될 수 있음.

Flash Loan

플래시론 예시

X Swap과 Y Swap을 통해 토큰 A/토큰 B 풀에 대해 플래시론으로 차익거래를 시도하는 시나리오에서:

- A와 B의 교환 비율은 X Swap에서 1:x, Y Swap 에서 1:y 로 가정
- 가격 영향(Price Impact)을 고려하지 않고 플래시론과 스왑 수수료는 0.3%로 가정
- 먼저 X Swap에서 토큰 B를 z만큼 빌려, 이를 Y Swap에서 토큰 A로 교환한다. 이 때 얻을 수 있는 토큰 A의 개수는 $0.997z/y$ 이다.
- 이를 X Swap에 상환한다. 이때 상환은 토큰 A를 이용해 이뤄지며, 수수료를 포함하여 $1.003z/x$개 만큼의 토큰 A를 상환해야 한다.
- 사용자는 $0.997z/y$ - $1.003z/x$ 개의 토큰 A 만큼의 차익을 얻는다.



Flash Loan Attack

플래시론 기능 자체는 순기능이 있으나, 플래시론을 악용한 공격이 가능함.

- 해커가 플래시론을 사용하여 대량의 암호화폐를 확보함.
- 이를 사용해 암호화폐 시세를 순간적으로 조작하여 이득을 얻음.

최근 플래시론 공격 예시: Pancake BUNNY 공격 (2021.05.19, 바이낸스 체인)

Pancake BUNNY: 암호화폐 예치량에 따라 보상으로 BUNNY 토큰을 지급하는 DeFi 서비스.

- 해커는 Pancake BUNNY가 보상을 산정할 때 Pancakeswap 의 현재 가격을 참조하는 취약점을 공격함.
- Pancakeswap 및 다른 플래시론 프로토콜에서 대량의 코인을 빌려 Pancakeswap 의 BNB-USDT 가격을 조작함.
- Pancake BUNNY 서비스에 토큰을 예치하고, 조작된 가격에 기반하여 막대한 양의 BUNNY 토큰을 발행받음.
- 얻은 BUNNY 토큰을 즉시 처분하여 플래시론을 상환하고 차익을 탈취함.
- 그 결과 BUNNY 토큰의 가격이 폭락함.

DeFi Lending 프로토콜 개념

중간자 없이 담보 대출 서비스 제공

기존 암호화폐 대출 서비스의 문제점:

1) 중앙화된 서비스: 신뢰하기 어려움
2) P2P 대출 프로토콜: 대출과정이 번거롭고 유동성이 극히 제한됨

해결책

1) 이용자들이 자산을 예치하고 이자를 보상받을수 있는 머니 마켓 구축함으로써 유동성 확보
2) 머니 마켓의 예치량에 따라 프로토콜에 의해 자동 계산된 이자율로 즉시 대출
3) 모든 과정이 스마트 컨트랙트 및 거래내역상에 투명하게 공개

DeFi Lending 프로토콜 예제 (Compound)

이자율 자동 계산

- incentive 모델: 대출량이 많을수록 이자율을 높여서 대출억제 및 예치 유도
- 모든 사용자에게 동일한 이자율 적용
- 대출, 예치량이 변화할 때 마다 재계산

$$U_a = Borrows_a \,/\, (Cash_a + Borrows_a)$$

$$Borrowing\ Interest\ Rate_a = 2.5\% + U_a * 20\%$$

DeFi Lending 프로토콜 예제 (Compound)

대출 담보 설정 및 청산

- 대출자는 대출받을 암호화폐 이상의 가치를 가지는 다른 암호화폐를 담보로 설정하여야 함. 최대 담보율은 프로토콜의 거버넌스에 의해 결정됨.
- 가치 책정은 상위 10개 거래소를 기준으로 한 가격 오라클에 의해 수행됨
- 암호화폐의 가격 변동으로 인해 담보가치가 일정 이상 줄어들면, 담보물의 일부를 청산하여 파산을 방지할 수 있음.
- 이 때 담보물을 구입하려는 청산자는 할인된 가격으로 구입가능함.

DeFi Lending 프로토콜 예제 (Compound)

대출 담보 설정 및 청산 예제

- ETH 가격이 $5000, DAI 가격이 $1 라고 가정.
- ETH의 최대 담보율이 50%라고 가정. 즉, 1 ETH를 담보로 최대 2500 DAI를 대출가능.

1) 사용자가 1 ETH를 담보로 2500 DAI를 대출함
2) 그런데 ETH 가격이 $4500 으로 하락하면, 대출자의 담보가치는 $2250 인 데 반해 대출가치는 $2500 인 부실대출로 변함.
3) 이 때, 청산자가 담보물인 ETH 중 일부를 머니 마켓으로부터 구입 가능함
4) 청산자가 0.2 ETH를 약 10% 할인된 가격 810 DAI로 구매함
5) 결과적으로 대출자는 0.8 ETH 를 담보로 1690 DAI를 대출한 것이 되며, 환산 담보가치 (0.8 ETH * $4500 * 0.5 = $1800) 대비 대출가치 $1690 으로 건강한 대출로 전환됨.

Stable coin (DAI) 개념 소개 및 예시

다른 화폐 또는 실물 자산 가격과 연동하여 코인의 가격이 거의 변동하지 않고 안정된 암호화폐

법정화폐 또는 다른 암호화폐를 담보로 잡거나 알고리즘에 의해 공급량 조정을 통해 가격이 안정됨.

법정화폐 (USD)를 담보로 발행함으로써 가치 유지: USDT, USDC

이더리움을 담보로 발행 및 가격 오라클에 기반하여 가치 유지: DAI

수요와 공급을 조절해 가격안정성 유지: Terra

Gwangju Institute of
Science and Technology

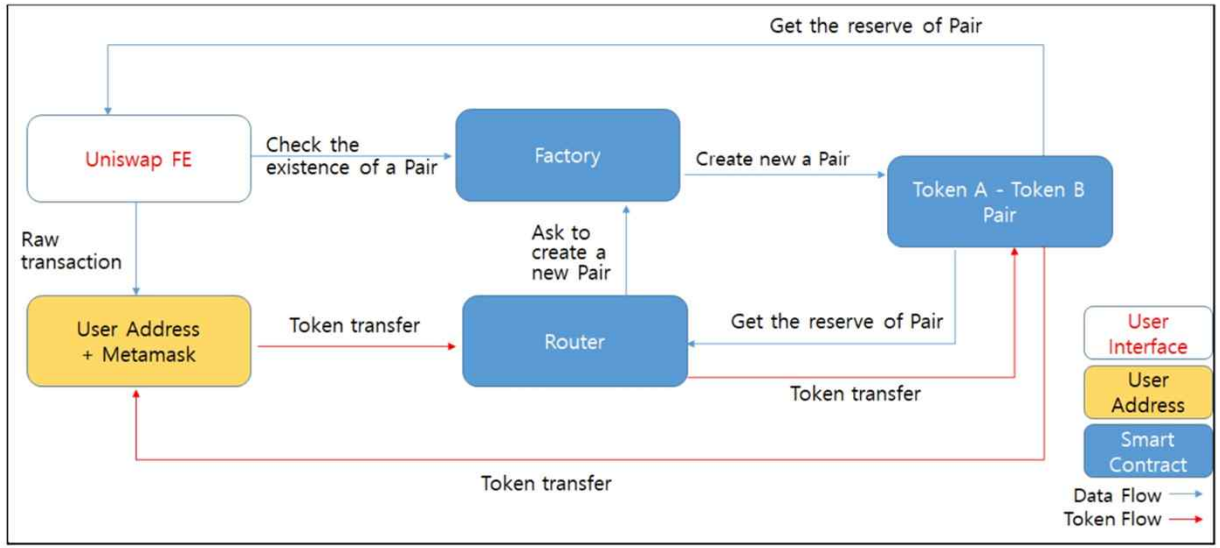
School of Electrical Engineering and Computer Science

SWAP 실습 (GIST swap)

실습 목차

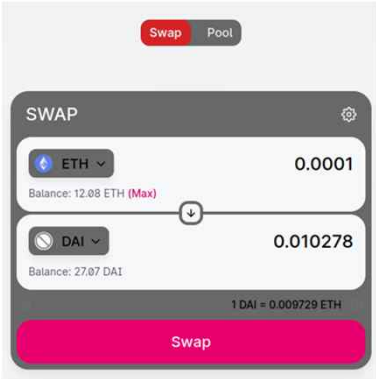
- 실습 전 준비사항: 메타마스크 설치
- 실습 전 준비사항: 테스트 이더 분배
- GIST swap 기능 및 구조 상세
- GIST swap 인터페이스 살펴보기
- 주요 기능Swap/Add Liquidity/Remove Liquidity 실행해보기
- 이더스캔 상에서 확인해보기

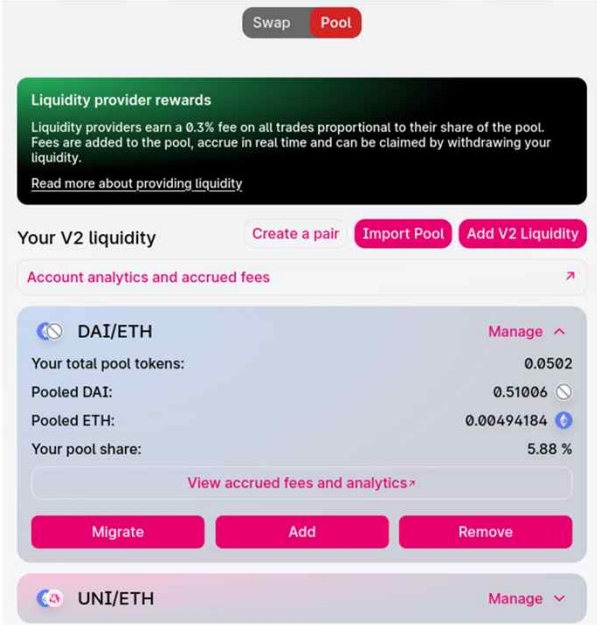
GIST swap 기능 및 구조 상세



GIST swap 인터페이스 살펴보기

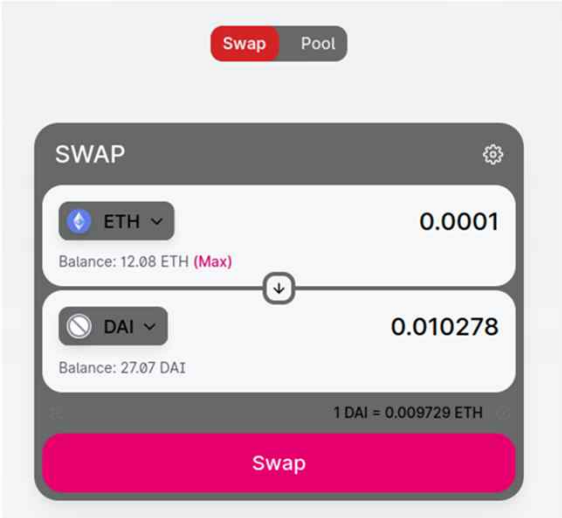
http://203.237.39.177:443





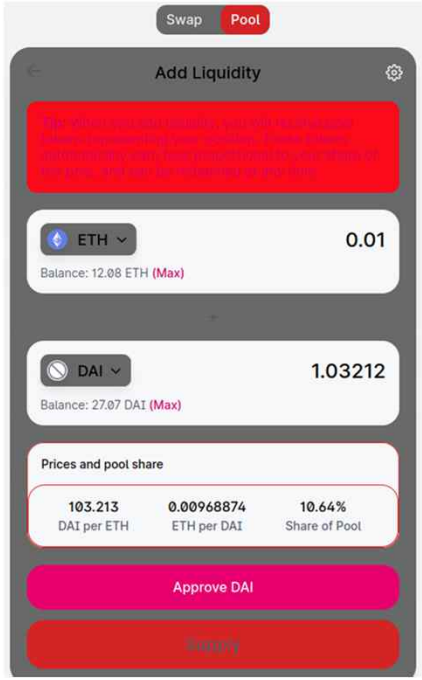
Swap/Add Liquidity/Remove Liquidity 실행해보기

Swap



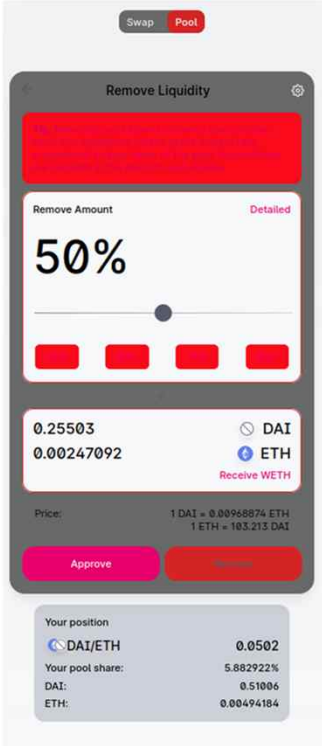
Swap/Add Liquidity/Remove Liquidity 실행해보기

Add Liquidity



Swap/Add Liquidity/Remove Liquidity 실행해보기

Remove Liquidity



이더스캔 상에서 확인해보기

Router:

https://ropsten.etherscan.io/address/0x073b662f25d22a1cda1e7a7100bffd6aef37d394

Test token address (INFONET):

Code: https://github.com/Haeung9/myERC20.git

Deployed: 0xD23E2320cA33178Fd6651A99e61cBBC1612909FF

https://ropsten.etherscan.io/address/0xD23E2320cA33178Fd6651A99e61cBBC1612909FF