

Distributed Computing: Fall 2018

Programming Assignment #3

1. Objective

Solidity는 이더리움 기반의 Smart Contract를 구현하기 위한 프로그래밍 언어이다. 본 과제에서는 Solidity를 활용하여 경매 시스템을 구현한다. 경매 시스템의 Frontend 인 웹사이트는 HTML, JavaScript로 구현하고 Backend는 Smart Contract로 구현한다.

- Solidity 프로그램 코딩
- JavaScript – Smart Contract 연동 프로그램 코딩

2. Program Coding

이 섹션에서는 Solidity 프로그램 코딩에 대해 설명하고 HTML 예제를 제공한다.

Smart Contract를 활용하여 경매 시스템에 필요한 데이터를 수집, 반환한다.

HTML 파일은 Github Repository에서 확인할 수 있다.

경로는 <https://github.com/ppk212/ppk212.github.io>의 Assignment_2 폴더이다.

2.1. Web Server

해당 시스템을 구현하기 위해 별도의 웹서버를 구성하는 것은 비효율적이므로 GitHub 계정을 활용하여 서버를 구현한다. 시스템을 구현하고 과제 제출 후, 보안을 위해 소스 파일을 Repository에서 삭제하는 것을 권장한다.

2.2. HTML

Smart Contract 코딩 작업을 수행하기 전에, 해당 웹사이트의 기능에 대해 이해해야 한다.


DCC Auction (Ropsten Testnet)

Contract Address : 0x25932587363FBA8E57ab745E39CE58D10047a6a6

Account Address : 0xfc0215bb749966a336ef7a54e89c313649922d84

Holding ETH Count : 61.786254988ETH

Holding Token Count : 225



iPhone 7


iphone 7

Highest : 125

My bid : 125

Number of tokens to buy

Bid



Galaxy S9


Galaxy S9

Highest : 30

My bid : 25

Number of tokens to buy

Bid



iPhone 8


iphone 8

Highest : 52

My bid : 25

Number of tokens to buy

Bid



Galaxy Note 9


Galaxy Note 9

Highest : 30

My bid : 0

Number of tokens to buy

Bid



iPhone X


iphone X

Highest : 30

My bid : 0

Number of tokens to buy

Bid



LG G7

LG G7

Highest : 0

My bid : 0

Number of tokens to buy

Bid

Tokens

Tokens Info	Value
Price Per Token	0.01ETH
Token Count In The Contract	15ETH

Purchase Tokens

Number of tokens to buy

Buy

- 웹사이트 상단에 해당 시스템의 CA, EOA, EOA가 소유한 Ether, 해당 Contract를 사용하는데 필요한 Token 개수를 보여준다.
- 이 때, CA는 JavaScript에 Hard Coding하고 EOA와 소유한 Ether는 Web3 Library를 사용하여 구현하며 Ether로 Token을 환전해서 Auction 시스템에서 사용할 수 있도록 한다.
- 경매를 진행하는 상품은 총 6개이며 각 상품마다 경매 최고가(Highest), 자신의 입찰 금액(My bid) 데이터를 가진다.
- 각 상품의 Bid 버튼과 좌측의 span을 활용하여 자신의 입찰 금액을 입력한다. 이미 입찰한 금액이 있다면 기존 Token에서 추라고 Token을 추가한다.
- 자신이 입찰한 금액이 각 상품의 경매 최고가보다 높으면 해당 상품의 경매 최고가를 자신의 입찰 금액으로 대체한다.
- 웹사이트 하단에 해당 시스템을 이용하기 위한 Token의 가격과 CA가 소유한 Ether를 보여준다.
- 해당 시스템에서 사용하는 Token을 구매하는 버튼을 구현한다.

2.3. Solidity

Remix IDE를 활용하여 위의 기능들을 구현하기 위한 Smart Contract를 작성한다.

- Solidity의 버전은 0.4.24 Version을 사용한다.
- Smart Contract를 작성한 .sol 파일을 과제로 제출한다.

2.4. JavaScript

HTML로 구현된 웹 사이트의 기능을 구현하기 위한 JavaScript를 구현한다.

- JavaScript와 EVM과의 연동은 Google Chrome의 Plugin 확장 프로그램인 Metamask를 활용하며 JavaScript에서 연동 기능을 구현한다.
- HTML의 Component들의 기능을 구현하고 Smart Contract와 Web3을 활용하여 필요한 데이터를 사용한다.

2.5. 다큐먼트 파일

Smart Contract 소스에 대해 설명한다.

- 변수에 대한 설명
- 함수에 대한 설명

3. Grading

채점은 다음 항목을 기준으로 한다.

- 경매 시스템의 기능 구현 정확도
- 구현한 Smart Contract 의 기능 설명
- 과제 제출 기한

4. Submission

구현한 프로그램의 소스파일을 제출한다.

- HTML 파일
- JavaScript 파일
- Solidity 파일
- 구현한 Smart Contract의 기능과 Dapp 전반 동작을 설명하는 다큐먼트 파일

과제는 위의 소스파일을 zip파일로 압축한 후, Canvas 프로그램을 사용하여 제출할 것

Assignment # 3으로 제출하고 제출 기한은 2018-12-20 23:59:59 까지 설정된다

과제에 대한 문의사항은 ppk212@naver.com 이메일 문의할 수 있음.