Distributing Trust & Blockchains

Assignment 2

> The following text describes the required packages, the steps to run the code, and the logic behind the desired functionalities.

Required Packages

You need to have NodeJS and Truffle installed. Instructions are present in the links attached.

On the terminal, type

```
cd client
npm install
cd ..
```

Steps to Run

On the terminal, type

```
cd backend
truffle migrate
cd ../client
npm start
```

Flow For Auction

Add New Item For Auction - Seller The seller logs in using their account via metamask. The seller portal is opened. This page contains a form where the seller can add details about the item to be added for auction. Item name, description and Auction Type are chosen.

```
addItemForAuction(itemName, itemDescription, auctionType)
```

The details added by the seller are used to call the above function on the deployed contract and a new item is added for auction.

View Items Available For Auction - Buyer The users can see all the items available for auction on the AuctionPlace page. Only items added by other users are visible on the AuctionPLace as one can not bid for their own item.

Place Bid - Buyer The moment a seller adds an item it is available for bidding. Users place their bid for the item of their choice. This bid is kept secret untill the auction ends. Following function of the contract is used to place a bid.

```
bidAtAuction(uint256 itemID, bytes32 hashedBid)
```

The amount bid by the user along with a secret password/signature of the bidder are hashed using SHA-3 to get the hashed bid. This way the bids placed by the users are kept a secret. These hashed bids are mapped to the buyer address and stored in the contract.

Stop Bidding - Seller Items added by the seller for auction are visible to them on the Portals page. Whenever the seller decides to stop the bidding no more bids can be placed.

```
stopBidding(uint256 itemID)
```

Pay Bid - Buyer After bidding stops bidders start paying the amount that they bid and disclose the password that they had used to hash the bid. The contract verifies that the hash of bid and password match the hashed bid submitted by the bidder. If the hash does not match the bid is disqualified. Along with paying the bid this function is also used for sending the public key of the bidder to the contract. After the auction ends, the pulic key of the winner is used to encrypt the secret string and send to the winner.

```
payAndVerifyBid( itemID, publicKey, password)
```

Stop Auction - Seller The seller calls this function from the Portals page to stop auction for the given itemID and declare the winner.

```
stopAuction(itemID)
```

The winner is decided based on the auction type provided by the seller

- **First Price Auction Winner** The highest bidder wins the auction they have to pay exactly what they had bid for. Rest of the participants in the auction are refunded their submitted bids instantly.
- **Second Price Auction Winner** The highest bidder wins the auction they have to pay the price equal to the second highest submitted bid .All the participants other than the winner are refunded their submitted bids instantly. While the winner is refunded the difference between the highest and second highest bid.

- **Average Price Auction Winner.** The winner is the person who bids closest to the average of all received bids. The winner in average price auction pays what they bid. All the other participants get a refund for their bids.

Deliver Secret String to the winner - Seller After the auction is over the seller enters the secret string on the portals page.

This string is encrypted using the RSA algorithm and the public key of the winner. This way only the winner can decrypt the secret string using their private key and no other user knows about the secret string.

```
deliverItem( itemID, encryptedString)
```

Users can view items that they own on the My Cart page. Untill the seller sends the encrypted secret string it shows wait for delivery. The seller must send the encrypted secret string to recieve the amount from the contract.

Recieve Secret Stirng - Buyer After the item is delivered the winner uses their private key to decrypt and get the secret string string. This is displayed to the user on the Carts page.

Flow For Fixed Price Sale

Add New Item For Sale - Seller The seller logs in using their account via metamask. The seller portal is opened. This page contains a form where the seller can add details about the item to be added for sale. Item name, description and Price are entered.

```
addItemForSale(itemName, itemDescription, itemPrice)
```

The details added by the seller are used to call the above function on the deployed contract and a new item is added for Fixed Price Sale.

View Items Available For Sale - Buyer The users can see all the items available for Sale on the AuctionPlace page. Only items added by other users are visible on the AuctionPLace as one can not bid for their own item.

Buy Item - Buyer The buyer pays the price of buying the item and this ether/wei goes to the contract and is stored their untill the seller delivers the secret string. Along with paying the price for item this function is also used for sending the public key of the buyer to the contract. After the seller sends the secret string the pulic key of the buyer is used to encrypt the secret string and send to the buyer.

```
buyItem(itemID, publicKey)
```

Deliver Secret String to the Buyer - Seller After the buyer buys the item, the seller enters the secret string on the portals page. This string is encrypted using the RSA algorithm using the public key of the winner. This way only the winner can decrypt the secret string using their private key and no other user knows about the secret string.

```
deliverItem(itemID, encryptedString)
```

Users can view items that they own on the My Cart page. Untill the seller sends the encrypted secret string it shows wait for delivery. The seller must send the encrypted secret string to recieve the amount from the contract.

Receive Secret String - Buyer After the item is delivered the buyer can use their private key to decrypt and get the secret string. This is displayed to the user on the Carts page.

Additional Information

- All relevant code has been properly explained with Doxygen Comments.
- A few sample test cases have been provided too. They are present as a markdown file `tests.md` in the folder.
- Please refresh after you change account in MetaMask, or add a listing, or place a bid, or deliver an item, etc. for the changes to be reflected on the page.

Group 1

- Manish (2018101073)
- Akshat Goyal (2018101075)
- Tanish Lad (2018114005)