Executive Summary

Online auctions have been a part of the digital world for a long time. However, most online auction apps use Web2-based client-server systems to host their applications. The centralization of data processing and storage by the auction house can increase the level of distrust between parties involved in the auction process. Decentralized Applications can help alleviate these problems.

A decentralized Application (dApp) is a computer application that runs on a distributed computing system. DApps have been popularized by Distributed Ledger Technologies (DLTs), including, but not limited to, Ethereum, Tezos, and Cardano blockchains. Since DApps exists on a network of users, they do not have a single point of failure, are less prone to attacks as compared to apps with single servers, and are resistant to modification and/or censorship by centralized authorities. Furthermore, given that a vast majority of DLTs support native cryptocurrencies, smart contracts, and wallets, DApps offer developers simple tools to perform user identity checks and facilitate financial transactions in a transparent manner.

Given the problems with the current online auction system and the advantages that DApps offer, we developed an auction house on the Ethereum test network where users can place bids on items. We chose to develop on the Ethereum network due to its popularity and its vast suite of development tools (Solidity, Truffle, Ganache, etc.). The DApp enabled us to deliver a greater level of transparency and trust by publishing auction, and bid-related information to the immutable and permanent Ethereum blockchain. Furthermore, we were able to maintain user privacy by using Ethereum-based wallets for identity checking, which do not contain any user's personal information. Lastly, the Ether cryptocurrency, along with the gas fees and time associated with transaction confirmation helped us in discouraging unfair practices like price gouging.

The "Fine Art Gallery" Auction DApp leverages smart contracts to allow users that have an Ethereum wallet to bid on items offered by the auction house. The web interface contains a list of items, their descriptions, and associated auction information. Users can place the highest bid on any item provided that the proposed bid adheres to the rules of contract. The transactions and information related to the highest bids and bidders are stored on the blockchain.

The smart contract (in the back-end) facilitates the storage and processing of actions by the user. When the web-page is loaded, information is fetched using smart contract and populated on the front end. When users place bids, the contract ensures that the bid value is an integer and that it is higher than the base price plus a minimum increment specified. It also ensures that the highest bidder of an item cannot place consecutive bids on the same item. When the aforementioned criteria are met, the contract is executed and a promise is fulfilled (i.e. the bid is accepted and transaction is completed). This transaction is confirmed by nodes on the blockchain where it is permanently stored.

It should be noted that time constraints limited our project's scope and so our DApp presents a simplified version of a complex auction process. As it stands, the app is predominantly a data management system that tracks auctions and only charges users the gas fee associated with execution of bid contract. When users place bids, the app does not withdraw the actual bid amount as a deposit from the bidder's wallet. In addition, the app only tracks the highest bidder and the highest bid amount for an item. Furthermore, the app also does not allow users to create their own auctions.Also, there is no time limitation for auctions so bids can be placed indefinitely. Lastly, all of the data is written to the blockchain, which can be burdensome, especially if the app is scaled.

In order to address the aforementioned limitations while improving the security of the app, we propose that future developers create contracts that require a user to deposit the bid amount up front. The smart contract can be programmed to return the deposit to the user if the contract expires and that user is no longer the highest bidder. In order to address the scaling limitations associated with the current write-heavy scheme, a native token can be issued by the auctioneer to facilitate payments. This can allow the app to handle small transactions or auctions on the platform off-chain on private servers, and migrate only important auctions with large value to the Ethereum blockchain.

Akshata Puranik, Dylan Mendonca