Procurelink Smart Contract

Overview

The smart contract is designed to facilitate auctions for Procurelink. It allows users to create auctions, place bids and settle the auctions. The contract supports multiple bids and includes functionality for the service provider/ the seller to receive payment when the auction is completed.

Goals

1. Enable users to create auctions for their “tender” for the items/ services they are providing.
2. Allow users to place bids on the auctions, considering the minimum bid requirements.
3. Facilitate the settlement of auctions by transferring the item/services to the winner of the bid and sending the payment to the seller.
4. Keep a record of auctions created, the live auctions and auctions won by the users.

Specifications

The specifications are in the comments written on the smart contract. The smart contracts aim to create a secure, fair and reliable platform. It ensures that the user will get the best price for their items and facilitates the exchange of items/services along with payments between users.

Integration with front end

1. Our front end was developed using Javascript, React, HTML, CSS
2. We want to connect to the blockchain network by using a library or api such as Algosigner to connect the front-end application.
3. Integration with the smart contract will require methods and functions that we could use to call the data. We will use the Algorand SDK's or libraries to invoke the transactions from the front end.
4. Wallet integration, we have a wallet tab on the front end, that allows us to integrate the wallet with functionality that allows users to sign transactions and interact with the smart contract securely.
5. Display auction information using the fetch via the Algorand SDK to display it onto the front-end ui. We will be retrieving auction details, bidder information, settlement status and providing real-time updates to the users.
6. Transaction monitoring and notifications, implement transaction monitoring functionality to track the status of the transactions. When a successful bid has been placed on the post or auction settlement, a notification should alert the users accordingly.
7. Using the local algorand testnet, we will test the integration between the front end and the application and ensure that the application is accessible to users.
8. We will iterate and monitor the application based on user feedback.

Future features of the smart contract

I. Escrow functionality

Implement escrow capability to secure funds during the auction process. This ensures that payments are delivered to the seller and bidder only when certain conditions are met, such as auction completion or settlement.

II. Job completion verification

Implement a system for validating the completion of a project or service before paying the service provider. This could entail integrating external systems to evaluate task completion criteria and automatically initiate the payment procedure.

III. Analytics and reports

To track auction performance, bid trends, and user behavior, leverage data analytics and reporting services. This can give users and platform managers useful information for optimizing auction techniques and user engagement.

IV. Secure and enhanced user profiles

Create user profiles that show auction history, winning auctions, and other pertinent information. This increases openness and allows users to highlight their auction participation and success.