

Article

# A Blockchain-Based Auction Framework for Location-Aware Services

Khaled Almiani <sup>1,2,\*</sup> , Mutaz Abu Alrub <sup>3</sup>, Young Choon Lee <sup>4</sup> , Taha H. Rashidi <sup>5</sup>   
and Amirmohammad Pasdar <sup>4</sup> 

<sup>1</sup> Faculty of Computer Information Science, Higher Colleges of Technology, Fujairah Women's Campus, Fujairah P.O. Box 1626, United Arab Emirates

<sup>2</sup> Computer Science Department, Faculty of Information Technology, Al-Hussein Bin Talal University, Ma'an 71111, Jordan

<sup>3</sup> Decapolis, Amman 11185, Jordan; abualrub.mutaz@gmail.com

<sup>4</sup> School of Computing, Macquarie University, Sydney, NSW 2109, Australia; young.lee@mq.edu.au (Y.C.L.); amirmohammad.pasdar@hdr.mq.edu.au (A.P.)

<sup>5</sup> School of Civil and Environmental Engineering, University of New South Wales, Sydney, NSW 2052, Australia; rashidi@unsw.edu.au

\* Correspondence: kalmiani@hct.ac.ae

**Abstract:** As a critical factor in ensuring the growth of the electronic auction (e-auction) domain, the privacy and security of the participants (sellers and buyers) must always be guaranteed. Traditionally, auction data, including participant details, are stored in a third party (auctioneer) database. This leads to a high risk of a single point of failure in terms of privacy and security. Toward this end, blockchain technology has emerged as a promising decentralized communication paradigm to address such risks. This paper presents a blockchain-based auction framework as a decentralized e-auctioning framework for location-aware services. In particular, the framework consists of three components: pre-auctioning, main auctioning, and post-auctioning processes with four algorithms. Our primary focus is on location-aware services, such as storage space rental, apartment rental, transport hire, and parking space rental. The trading volumes are expected to be high; hence, simplifying the design is a crucial requirement. In addition to the benefits of eliminating the centralized entity (the auctioneer), fees are redistributed among participants as rewards. Further, we incorporate a service quality monitoring method that ranks the services provided by participants. This ranking allows participants to determine the rank of other participants with whom they wish to trade. We have conducted several experiments to evaluate the proposed framework's cost feasibility and to ensure the ease of the business flow.

**Keywords:** Ethereum; e-auctions; blockchain; decentralized applications; blockchain oracle problem



check for updates

**Citation:** Almiani, K.; Alrub, M.A.; Lee, Y.C.; Rashidi, T.H.; Pasdar, A. A Blockchain-Based Auction Framework for Location-Aware Services. *Algorithms* **2023**, *16*, 340. <https://doi.org/10.3390/a16070340>

Academic Editors: Michele Squizzato and Gopal Pandurangan

Received: 16 June 2023

Revised: 12 July 2023

Accepted: 14 July 2023

Published: 16 July 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Blockchain is a distributed ledger technology (DLT) [1], where several network nodes have identical copies of business transactions. The decentralized nature of blockchain has eliminated the need for a centralized authority, in which smart contracts have highlighted the automation capabilities of processes [2]. A smart contract is a program (code) executed based on a prespecified set of events, and such events belong to the smart contract implementation. This disruptive technology has become a promising platform to expand the horizon of the decentralized applications domain [3], including but not limited to e-commerce domains such as business auctions.

Electronic auctions (e-auctions) have become an essential daily business trading application in which the buyers and the sellers (bidders) are not required to be physically present at the auction site. Furthermore, the entire auction process relies on auction agencies (auctioneers) to facilitate, monitor, and run the auction. These agencies typically charge

administrative fees to perform the auction and play a crucial role in this business; however, they increase the risk of privacy concerns [4]. This is rooted in the need for storing business information in a centralized fashion and on several databases that belong to these agencies. Hence, this information sharing mechanism increases the vulnerability of the stored data in terms of manipulation, damage, and loss.

In this direction, blockchain technology can address e-auction privacy concerns and improve the e-auction trading experience. This is due to the existence of blockchain-based auction systems employed to facilitate auctioning items that cost millions of dollars [5]. Thus, such reliance on the blockchain has eliminated the need for a central authority, ensuring the integrity of the e-auction business process. In addition, blockchain transparency and its promising decentralized features incentivize government entities to be involved as trading parties (sellers or buyers), since no single entity controls the entire auction process, improving the trading experience.

Two bidding strategies in e-auctions are open bid and sealed bid auctions [6]. In the former, when a bidder places a bid, the other bidders are notified about the new bid price, and bidders can typically bid several times when the auction is in progress. In the latter, a bidder is unaware of the price submitted by other bidders, each bidder can place only one bid, and the auctioneer announces the winning bid at the end. Compared to open bid auctions, the sealed bid strategy further supports the bidders' anonymity since any bid information is not accessible to the bidders. Additionally, the sealed bid auction structure motivates the bidders to submit the best price they can offer since each bidder can submit just one bid price. These characteristics emphasize the benefits of employing a sealed bid auction strategy due to supporting the bidder's privacy further and reducing the number of bidding-related transactions.

The roles of participants in e-auctions can be classified into forward and reverse auctions [6]. In forward auctions, sellers offer goods for sale, and buyers compete by outbidding each other, aiming to find the highest possible price. In a reverse auction, the buyer announces the details of the requested service or goods, and the sellers compete by providing the lowest possible price for the requested service or goods, focusing on lowering the cost.

Moreover, the auction strategy can be further categorized into absolute, reservation, and combination strategies [6–8]. Using the absolute strategy, the bidder with the highest bid is the winner of the auction, and each bidder is only allowed to submit one bid. Using the reservation strategy, the seller can cancel the auction if the offers do not reach a pre-determined threshold price. In the combination strategy, a mix of absolute and reservation strategies can be established, whereby the bidder with the highest bid wins the auction if their bid reaches a pre-determined price threshold. In this work, we employ a combination strategy to utilize the benefits of both the absolute and the reservation strategies.

This paper proposes a blockchain-based, sealed bid, reverse auction framework as a decentralized auction framework consisting of pre-auctioning, main auctioning, and post-auctioning processes. This location-aware framework employs these components powered by lightweight algorithmic solutions to ensure the ease of the business flow. The sealed bid auction nature reduces the pressure on the security-related mechanism, which further supports the use of blockchain technology. Additionally, the limited number of offers in such a bidding strategy contributes toward minimizing the cost of running the auction on a public blockchain. With minor modifications, the proposed framework can address open-forward auction scenarios.

The proposed framework primarily deals with four types of rental services: (1) storage space rental, (2) apartment rental, (3) transport hire, and (4) parking space rental. Each auction party (seller or buyer) in this framework is associated with a service index (SI) that represents the quality of the service they provide. The SI is cumulatively calculated, considering the fulfillments of the auction requirements. Moreover, the proposed framework employs a fee redistribution strategy between the participating sellers to encourage the sellers to place the best possible bids. This redistribution relies on the blockchain cryp-

tocurrency that facilitates the processing of financial transactions in a decentralized manner. In the case of raising disputes, the framework provides a decentralized violation dispute component to resolve any dispute between the bidders and sellers during the contract period. This component uses the information collected during the auction to determine whether a quality of service (QoS) violation occurred and penalizes the violating party.

The contributions of this paper can be summarized as follows:

- We devise a rewarding algorithm that encourages the sellers (bidders) to submit the best possible offers.
- A service index calculation component is designed, which employs a ranking algorithm to distinguish between the sellers in terms of the provided QoS.
- A decentralized service violation dispute component is proposed to facilitate the disputation of any raised service violation claims during the auction contract period.
- The proposed framework is implemented by Solidity (0.8.17). Hardhat is used as the development environment and Ethereum blockchain as the targeted blockchain.
- Several experiments are conducted to analyze the cost and the overall efficiency.

The analysis results show that each auction participant is expected to pay, on average USD 5.5 to participate in the auction. The experiments indicate that the rewarding mechanism motivates the bidders to submit their best offers. This behavior is established since the rewards of the bidders, who submitted offers close to the winning offer, are expected to be substantial. Furthermore, the efficiency of the proposed SI is investigated to highlight its potential to improve overall performance.

The rest of the paper is organized as follows. Section 2 provides a background, followed by our review and discussion of the related work. Section 3 details the proposed blockchain-based auction framework. Section 4 provides a detailed discussion about the proposed framework. Section 5 presents the experiment results and we conclude the paper in Section 6.

## 2. Background and Related Work

This section discusses the most relevant studies that address the benefits of adopting blockchain technology in the e-bidding business domain.

Chen et al. [9] proposed a three-layer framework to simplify writing e-auctions' smart contracts. The proposed framework achieves its objectives by defining the process of mapping typical contract details into smart contracts. It scans the syntax of the user-provided contract and applies the required processes to transform the user input contract into smart contracts. In ref. [4], the authors proposed a collusion-resistant blockchain e-auction to avoid collusion among buyers through introducing parameters with a stochastic nature that are part of the auction processes. To reduce the probability of overbidding (winner's curse), Shih et al. [10] proposed a blockchain-based auction system with price recommendation capabilities. The price recommender component gives the bidder the advantage by predicting the price (value) of the auctioned items. Lee et al. [11] introduced a blockchain-based system for double auctions, in which buyers and sellers submit their prices to the auction and a matching strategy is executed to match the sellers and buyers. In ref. [11], a genetic-algorithm-based solution is investigated to address the pairing of the sellers and the buyers. The use of a blockchain-based solution to address double auction problems is also investigated by Liu et al. [12].

Blass et al. [13] proposed a blockchain-based auction protocol that guarantees bid confidentiality, and in case of a dispute, the settlement of the dispute will be relayed to the parties with the majority behavior. In ref. [14], the authors have investigated the impact of blockchain technology on eliminating traditional e-auction problems, such as the presence of the third-party regulator entity and the possibility of leaking bid-related information. Braghin et al. [15] discussed and analyzed the benefits of employing a blockchain-based e-auction strategy. The authors have shown the importance of such a strategy for transparency and integrity. The former refers to the ability to verify the bidders' entire history, while the latter refers to the ability to ensure the bidders' privacy. The use of blockchain-based

e-auction solutions is analyzed by Omar et al. [16], in which the authors proposed a general framework that utilizes blockchain technology to address traditional e-auction problems such as the presence of a third-party authority entity and privacy-related issues. Moreover, the authors showed that the cost of executing auction-related processes on the Ethereum blockchain is typically low; hence, it motivates using blockchain technology in the e-auction application domain.

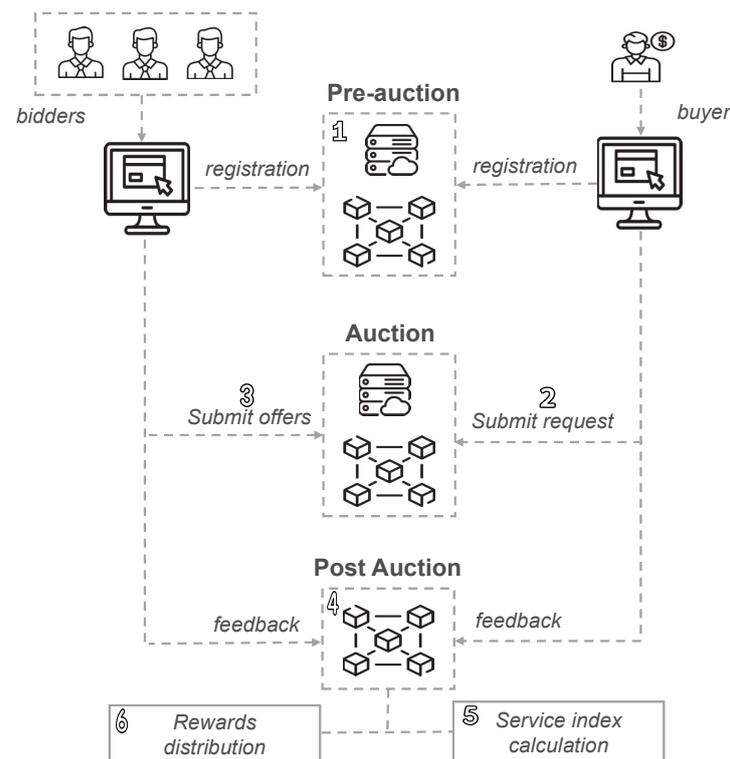
Pop et al. [17] presented an Ethereum-based implementation for English, Dutch, and first-price sealed bids. English and Dutch auctions are open auctions, but in English auctions, the bidders compete and the bidder with the highest bid wins the auction. In a Dutch auction, the auctioneer starts with a relatively high price and keeps decreasing it sequentially until a bidder offers a bid and wins the auction. The authors performed several experiments to capture the performance of the three proposed auction models to determine their impact on the execution cost. Compared to English and Dutch auctions, the sealed bid has an extra step, revealing the winning bid that requires an additional cost. The results showed that the sealed bid indicates constant and predictable behavior due to the simplicity of its architecture. Wang et al. [18] proposed an e-auction system built on top of a permissioned blockchain (Hyperledger Fabric [19]). The use of a permissioned blockchain supported the system by reducing the probability of running into the risk of having distributed denial of service (DDoS).

Our work differs from these previous works in terms of auctioned item types (location-aware services), the automatic evaluation of the quality of the provided service, and the volume of trades. In our work, the quality of service is automatically evaluated using smart contracts. Unlike auctioned items in many previous works, such as jewelry and antiques, our work targets location-aware services, often with a higher volume of trade. Hence, they require strategies to encourage user participation and ensure the ease of the business flow, addressed by proposing the reward distribution and the SI components.

### 3. The Proposed Blockchain-Based Auction Framework

The proposed framework consists of three components: the pre-auction, the auction, and the post-auction. The pre-auction component is for participants' registration and identity verification. To ensure the legitimacy of the auction process, participants must submit the requested information to identify themselves during the registration process. Once a participant registers with the system, they can act as a seller or buyer of services.

Figure 1 illustrates the interaction between the auction functionalities. To initiate an auction, the buyer submits the requested service details to the blockchain. These details include the maximum price a participant is willing to pay. In addition, the buyer must submit the requested submission fee to start the auction. All submitted fees during an auction will be deposited into an administrative account for redistribution at a later stage. The buyer must also submit the required SI for the participating sellers (bidders) in the submitted auction request. In the proposed framework, each bidder and buyer has an SI, where the value of this index can be either (1) undecided, (2) positive, or (3) negative. A participant's SI is determined based on the feedback received about this participant in previous interactions (auctions). Participants who have not previously interacted with other participants are assigned an undecided SI. Based on the feedback received during past interactions, positive and negative service indices are assigned to participants. A participant's SI captures the overall expected quality of service provided by the participant. For instance, participants with a positive SI value are expected to provide a higher quality of service compared to participants with a negative SI value.



**Figure 1.** The interaction between the proposed system stages.

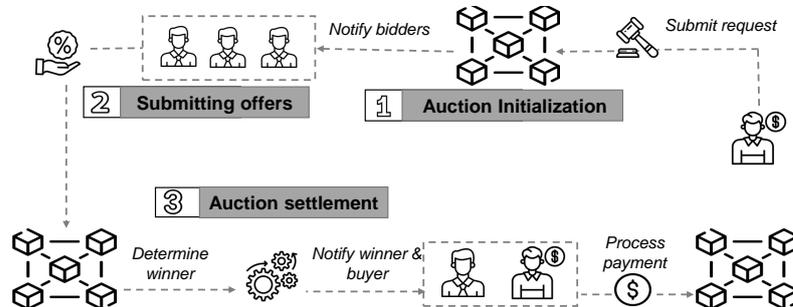
Once the auction is initiated, the submitted request is propagated to the bidders that satisfy the requested SI criteria. Interested sellers submit their offers along with a participation fee to be deposited into the administrative account. Once the auction ends, the details of the cheapest (winning) bid will be submitted to the buyer, and the buyer's wallet will be charged based on the payment mechanism submitted as part of the auction request. This mechanism could be either full payment or installments. The money is deposited to the service provider (winning bidders) using the same charging mechanism. During the post-auction period, the participating bidders are rewarded by the distribution of the collected fees in the administrative account among them. Once a particular auction agreement period is over, the seller and the buyer's SI will be updated based on their interaction experience. Moreover, during the contract agreement period, any of the auction parties (seller or buyer) can raise a service violation claim. This claim is processed by the violation dispute component, which works in a decentralized manner to resolve any dispute between the auction parties.

### 3.1. Pre-Auction

Participants (bidders or buyers) must be registered users to interact with any active auction. To register with the system, a new participant must first determine their role in the system (bidder or buyer). Each participant must submit official documents for identification (e.g., government ID or passport). Once the identity of a participant is verified, a public key will be generated and linked to the verified participant account. Accordingly, a private key will be given to the participant to manage their decentralized identity (DID). As part of the registration process, the seller has to indicate the type of service they can provide. In this paper, we will assume that the bidders (sellers) provide rental and transportation services. Thus, we limit the types of services to (1) a storage area, (2) an apartment, (3) a parking area, and (4) goods transport. A seller can modify the provided services list during the system operating lifetime. Once registered, the participant can link their Ethereum address to their account on the system. If the participant does not have an Ethereum address, a new address will be created and linked to their account on the auction system.

### 3.2. Auction

This stage deals with auction initiation, the submission of offers, and auction settlement. Figure 2 illustrates the interaction between the auction processes and the actors.



**Figure 2.** The interactions between the auction stage processes.

#### 3.2.1. Initialization

Algorithm 1 illustrates the auction initiation process. To initiate an auction, the buyer must submit a request along with the required fields summarized in Table 1, including paying the auction registration fee. This fee deposited into the administrative account is necessary to ensure the buyer's commitment. This fee is refundable to the user only if no offer is received from the bidders within the bidding timeline. The service location represents the physical address of the requested service (e.g., city and street name). In the service type field, the buyer determines the service category that he/she would like to rent (storage area, apartment, etc.). In the service description field, the buyer determines the duration of the required service. For instance, the user may specify that he/she wants to rent the described service for one month. In the maximum price field, the buyer determines the maximum price he/she is willing to pay for the service. The bidders' SI field represents the seller's SI requested by the buyer. Accordingly, a bidder with an SI that does not match the requested SI will not be notified about the auction. They can choose whether all bidders (undecided, positive, and negative) may participate in the auction or only sellers with a positive SI can participate. Using the additional requirements input, the buyer can describe requirements, such as the temperature of the storage area.

---

#### Algorithm 1: initializeAuction

---

**Data:**  $L, T, sd, max_p, SI$ , and  $ad$

- 1  $P \leftarrow determineEligibleBidders(L, T, SI)$
- 2 **if**  $|P| \neq zero$  **then**
- 3      $r \leftarrow createRequest(L, T, sd, max_p, SI, ad)$
- 4      $notifyBidders(P, r)$
- 5 **else**
- 6      $cancelRequest()$
- 7 **end**

---

Based on the service location, a submitted request can be accepted or rejected by the auction initialization process. This process is done by determining if there is any eligible registered bidder in the specified service location. If no bidder is found, the buyer will be notified and the request will be canceled. However, in other cases, the auction request is accepted, and the corresponding bidders who satisfy the requirements are notified about the auction.

**Table 1.** Inputs to the auction initialization process.

Input	Notation
auction registration fee	$AF$
service location	$L$
service type	$T$
service description	$sd$
auction deadline	$td$
maximum price	$max_p$
bidders' SIs	$SI$
additional requirements	$ad$

### 3.2.2. Offers Submission

Interested bidders can submit a single offer to any auction accessible to them, and this strategy motivates the bidder to submit the best possible offer. In this context, bidders have no information about the prices submitted by their counterparts. Each submitted offer is hashed off-chain by the bidder and the hashed value is submitted to the blockchain along with the bidder's address. A bidder address is used during the hashing process to establish a link between the bidders and their offers. Hashing the offer off-chain is required to ensure the privacy of the submitted information. The bidder must deposit the auction participation fee into the administrative account along with the submitted offer. Along with the offer, the bidder has the option to specify the required insurance amount that the buyer must deposit to use the provided service. Once the auction deadline is reached, bidders can no longer submit offers. At this stage, each bidder must reveal their offer by resubmitting the offer value un-hashed. To ensure that the bidder does not manipulate the revealed offer value, the revealed offer will be hashed by the smart contract using the bidder's address. In situations where the hashed value by the smart contract does not match the submitted hashed offer, the bidder's revealed offer would be rejected. If the two hashed values are equal, the revealed offer will be accepted, and in case the number of accepted offers is zero, the buyer (auction initiator) will be notified and the auction will be canceled.

### 3.2.3. Settlement

Algorithm 2 shows the various steps of the auction settlement process. Once the auction is finalized, the settlement process will be initiated to determine the auction winner and to process the payment. The bidder with the cheapest offer will be declared the winner, and their offer details will be forwarded to the buyer (lines 1–3). If the buyer rejects the proposed winning offer, the collected fees from the buyer will be transferred to the bidder's account. All other fees collected from the participating bidders will also be refunded to their accounts. If the buyer accepts the winning offer, the location-related information will be exchanged between the buyer and the seller. The nature of the exchanged information is determined based on the service type. For instance, when the service type is the provision of a storage area, an apartment, or goods transport, the buyer and bidder will exchange contact information, including addresses and phone numbers. When the requested service is a parking space, the car specifications and parking location are the only shared information. Furthermore, the payment for the service will be deducted from the buyer's wallet based on specific criteria. For instance, a user can specify a monthly payment strategy for the service.

### 3.3. Post-Auction

At this stage, as an encouragement strategy, the collected fees are distributed among the participating bidders. Furthermore, during this stage, based on the type of the provided service, the evidence collection process will be triggered (if applicable) to gather supporting information about the provided services. The collected information can be used by the violation dispute component as evidence. Additionally, the service indices for the winning bidder and the buyer are recalculated in this stage.

**Algorithm 2:** settleAuction

---

**Data:**  $ID$ : auction ID

```

1  $o \leftarrow \text{determineBestOffer}(ID)$ 
2  $b \leftarrow \text{getBuyerID}(ID)$ 
3  $\text{notifyBuyer}(b, o)$ 
4 if  $b$  accepted  $o$  then
5   |  $\text{exchangeInformation}(b, \text{getBidderID}(o))$ 
6   |  $\text{processPayment}(b, o)$ 
7 else
8   |  $\text{processFee}(ID, b)$ 
9 end

```

---

## 3.3.1. Distribution of Rewards

Once an auction is settled, the fees collected during the auction process will be distributed among the eligible nonwinning bidders. To qualify for a reward, the accepted revealed offer price submitted by the bidder must be less than the average price of all accepted offers. A fair-weighted strategy is employed to distribute the rewards, in which each eligible bidder's share depends on the difference between the bidder's offer price and the winning price. Algorithm 3 illustrates the reward distribution process. Once the identity of the eligible bidder is determined (line 7), the weight for a bidder ( $p_i$ ) is calculated as follows:

$$w(p_i) = \frac{o_w}{o_i - o_w} \quad (1)$$

where  $o_w$  represents the price offered by the winning bidder and  $o_i$  represents the price offered by the bidder ( $p_i$ ). By employing this strategy, the weight for each bidder must be less than or equal to one, and one is assigned as the weight for the winning bidder. Based on the obtained weight, the bidder ( $p_i$ ) share of the fees is calculated as follows:

$$s(p_i) = \frac{w(p_i)}{\sum w(p_j)} \times TF \quad (2)$$

where  $TF$  represents the total collected fees and  $\sum w(p_j)$  represents the total weight for all eligible bidders. If only one offer is made and the winner is declared by default, the fees submitted by the winner will be returned to their account.

## 3.3.2. Feedback and Service Index Calculation

Once the service duration is completed, the participants in the auction (both seller and buyer) are expected to provide feedback about each other. The feedback value can be either positive, negative, or undecided. The feedback is used to recalculate the SI score for the participants. The calculation of the SI takes into account the past feedback about the participants. In this process, each participant is associated with a cumulative weight ( $CW$ ) that is used as a credibility indicator for its feedback. Initially, the SI for all participants is determined as undecided.

Algorithm 4 illustrates the SI calculation, which is triggered once feedback about a participant is received. Accordingly, if feedback is received from the participant ( $p_A$ ) about their interaction with the participant ( $p_B$ ), the cumulative weight for the second participant ( $p_B$ ) will be recalculated as follows:

$$CW(p_B) = F \times \frac{MAX_{CW} + CW(p_A)}{2 \times MAX_{CW}} + CW(p_B) \quad (3)$$

where  $F$  represents the feedback provided by participant (1: positive, -1: negative, 0: undecided) and  $MAX_{CW}$  represents the maximum weight assigned for a participant. The weight of participant  $p_B$  will be updated by adding a value between -1 and 1. Furthermore,

the value of the update depends on the weight of the feedback provider, and the feedback provided by a participant with a high weight has more impact than feedback provided by the lower-weight participant. The SI for participant  $p_B$  will be changed based on its new cumulative weight ( $CW(p_B)$ ). If the value of the participant's weight is greater than one, the SI will become positive. However, if the value of the participant's weight is less than zero, the SI will become negative. A weight value between 0 and 1 results in an "undecided" SI to be assigned to the participant. The presence of the gap window [0–1] aims to determine the SI based on the overall cumulative feedback provided by the participants. Accordingly, switching a participant's SI from negative to positive requires consistent efforts to provide a reliable service. In contrast, more than one negative feedback is required to switch a participant's SI from positive to negative. The size of the gap window is adjustable, and reducing this size might result in the frequent shifting of a participant's SI from one status to the other. Furthermore, increasing the size of this gap window will increase the effort required from a participant to change the status of their SI. Furthermore, the initial cumulative weight for each participant is assigned to be the middle of the gap window (0.5). Such a placement virtually locates the participant at the same distance from the positive and the negative groups.

---

**Algorithm 3:** distributeRewards
 

---

**Data:**  $TF$ : total calculated fees,  $o_w$ : winning offer,  $O$ : submitted offers

**Result:**  $S$ : eligible bidders shares

```

1  $O' \leftarrow \{O\}/o_w$ 
2  $avg \leftarrow calculateAVG(O')$ 
3  $MAX_o \leftarrow IdentifyMAX(O')$ 
4  $E \leftarrow \emptyset$ 
5 if  $|O'| \neq 0$  then
6   for  $o_i \in O'$  do
7     if  $o_i > avg$  then
8        $w(p_i) = \frac{o_w}{o_i - o_w}$ 
9       add  $o_i$  to  $E$ 
10    end
11  end
12  calculateShare(  $E$  )
13 else
14    $s(p_w) = TF$ 
15 end
16 return  $S$ 

```

---

### 3.3.3. Evidence Collection

During the auction agreement period, the seller and the buyer must document their trading experience by uploading images to the system. The seller and/or buyer can upload such images at the beginning and the end of the agreement period. The images uploaded at the beginning of the agreement period must be digitally signed by both parties to start the service period officially. If both parties sign the uploaded images at the end of the service contract, the service contract will be closed. However, the seller or the buyer can refuse to sign the images uploaded by the other party. In this case, a service violation claim will be submitted to the violation dispute component. This claim can also be raised during the service contract period, and the claim owner can submit any additional information to support their service violation claim.

The uploaded images must be stored on the InterPlanetary File System (IPFS, <https://ipfs.tech/>, accessed on 6 December 2022) network, whereby the hash of the uploaded images is encrypted and stored on-chain. The mechanism of capturing these images

is outside the scope of this paper. However, a registered phone application can be used to obtain these images.

---

**Algorithm 4:** calculateServiceIndex
 

---

**Data:**  $F$ : feedback,  $p_A$ : feedback provider,  $p_B$ : evaluated participant,  $CW$ : all participants weights  
**Result:**  $SI(p_B)$  participant  $p_B$  SI

```

1 if  $F \neq zero$  then
2    $MAX_{CW} \leftarrow getMaxWeight(CW)$ 
3    $CW(p_B) = F \times \frac{MAX_{CW} + CW(p_A)}{2 \times MAX_{CW}} + CW(p_B)$ 
4   if  $CW(B) > 1$  then
5      $SI(B) \leftarrow P$ 
6   else
7     if  $CW(B) < 0$  then
8        $SI(B) \leftarrow N$ 
9     else
10       $SI(B) \leftarrow U$ 
11    end
12  end
13 end
14 return  $SI(B)$ 

```

---

### 3.3.4. Violation Dispute and the Witness Group

Once a violation claim is received, the submitted claim will then be transferred to the Witness Group (WG). This group consists of several participants who are responsible for disputing any violation claim. The concept of employing witnesses has been used in several application domains [20]. Witnesses are third-party participants, neither seller nor buyer. A witness is assumed to be trustworthy, and the selection of such participants must ensure the satisfaction of any regulation imposed by the government. The process of selecting the members of the witness group is outside the scope of this work, and we adopt the assumption that they are fair and always act honestly.

The members of the witness group have access to the auction agreement details, including the uploaded images. Such images are accessible using their hash values that are stored on-chain. The members of this group use their private keys to decrypt the images' hash values. They must process each submitted claim within a pre-determined time interval ( $t$ ). After reviewing the claim details, each member of this group can confirm or reject the received claim. Once the time deadline ( $t$ ) is reached, the outcome with the highest votes will be selected as the response for the claim. When the violation claim is confirmed and if the claim owner is the buyer, any scheduled payment starting from the claim date will be canceled. In comparison, if the claim owner is the seller, besides canceling the auction agreement, any insurance amount paid by the buyer will be transferred to the seller.

Furthermore, rejected claims will not result in terminating the period of the auction contract. For any confirmed claim, the violator will be given a chance to challenge the outcome during a pre-determined time interval ( $C_t$ ). In this situation, the witness group can either confirm the previous judgment or announce a new final outcome.

The members of the witness group are rewarded for their service, and part of the collected fees during the auction process can be used for rewarding purposes. The decentralization nature of this process depends on the number of members in the witnesses group. Thus, increasing this group's size supports the proposed system's decentralization aspects.

## 4. Discussion

In this section, we discuss how the proposed system is expected to address typical online auction fraudulent activities and the decentralized aspects of the proposed system.

### 4.1. Auction Fraud and Blockchain

Auction fraud is a challenging problem; hence, we discuss how the proposed system can address such a challenging task. Chua et al. [21] classified outline auction fraudulent activities into 10 types. Table 2 summarizes these types and maps them into the context of the proposed system in the paper. The proposed system uses a sealed bid reverse auction strategy. In the presented system, bidders are not aware of each other's bids, and each bidder can place a single bid. Accordingly, concerns regarding shilling and bid shielding fraudulent activities are eliminated, as is the possibility of manipulating the auction price. Additionally, any bid that exceeds the buyer-specified upper limit will be disregarded.

The auctioned items in the proposed system are logistics services. Fraudulent activities such as misrepresenting the traded items or/and the possibility of trading stolen (or damaged) items are not applicable in our case. However, in the proposed system, the seller might violate the QoS constraints to gain financial benefits. In such cases, the violation dispute component works to penalize the violator. This also will be the outcome if the seller did not provide the service (failure to ship) or if the buyer (or the seller) submitted an untruthful violation claim (buy and switch). In addition, the use of the service index aims to reduce the probability of service violations by highlighting the historical QoS provided by the auction party. In the proposed system, the auction processes are enforced by using smart contracts. Such processes limit the impact of having participants with malicious behavior who try to act honestly to gain trust and increase their service index (shell auction). The violation dispute component depends on facts, and the service index for the sellers or the buyer has no impact on the outcome of this process.

Once the auction is closed, the buyer may reject the winning offer. Furthermore, the buyer may attempt to avoid the payment (failure to pay) or even may attempt to avoid closing the auction. Rejecting the winning offer is acceptable since, in this case, the seller will receive the buyer's participating fee. Hence, the auction participation fee must be determined while considering the expected price for the service. Increasing the participating fee value may reduce the number of participating buyers and sellers, whereas reducing this fee may impact the quality of provided service and the behavior of the participants. In the experiments section, a sensitivity analysis is performed to highlight the factors that must be considered while specifying the participation fee amount. After the auction period, if the buyer does not close the auction, any participating seller might initiate the closing request. The reason behind adopting such a strategy is to support the decentralization of the proposed system. Transactions are time stamped and once the pre-specified auction time is over, any participant has the right to close the auction. At any stage, if the violation dispute functionality is called, the agreement number, along with the claim details, will be passed to the witness group members. Accordingly, once the members of this group cast their votes, the violators will be penalized as described in Section 3.3.4.

The process of distributing the rewards among the participants is designed to ensure that sellers do not join the auction just for the sake of the reward. To ensure winning the reward, the seller has to know the winning offer and statistical information about the entire list of submitted offers. The inherited characteristics of the sealed bid reverse auction model help to avoid such a situation, since bidders are not aware of each other bids. In addition, if multiple bidders collude to increase the chances of a specific bidder winning the reward, the set of colluding bidders will lose a proportion of their rewards. Thus, sharing bidding information between sellers to manipulate the rewarding mechanism is not expected to occur.

**Table 2.** Online auction fraud types.

Type	Description	Mapping	Solution
Shilling	Seller bids to manipulate the price	Buyer joins the auction to manipulate the price	Auction strategy
Bid shielding	Bidders collude to manipulate the price	Bidders collude to manipulate the price	(1) Rewards distribution, (2) auction strategy
Fee stacking	Hidden cost	Seller attempts to increase the cost after the auction	Automation
Failure to pay	Buyer refuses to pay	Buyer refuses to pay	(1) Participation fee, (2) automation
Failure to ship	Seller does not ship the items	Sellers do not provide the services	(1) Participation fee
Reproductions and counterfeits	Seller attempts to sell counterfeit goods	Sellers violate the QoS agreement	(2) Weight calculation
Triangulation	Selling stolen goods		(3) Violation dispute processes
Misrepresentation	Seller does not describe the items correctly		
Loss or damage claims	Buyer claims that the items are broken	Untruthful service violation claim	(1) Participation fee, (2) violation dispute processes
Shell auction	Seller joins the auction to build reputation	Sellers with malicious intent act honestly to increase their weight	(1) Weight calculation, (2) violation dispute processes

#### 4.2. Service Index and the Violation Dispute Component

The service calculation process can be linked to the violation dispute component to ensure any negative feedback is justifiable. For the witness group rewards, the sources of these rewards can be changed based on the type of the provided service. For instance, if the provided service is a furnished storage area, the buyer is expected to pay refundable insurance money. Accordingly, if the buyer damages the furnisher (violates the agreement), the insurance money can be used to reimburse the seller and reward the witness group members.

#### 4.3. Decentralization vs. Centralization

Decentralization is a mandatory requirement that the proposed system is designed to achieve. Regarding the administrator address, it is used to deploy the contracts and to enable the identification verification process. This address can be controlled by a Decentralized Autonomous Organization (DAO) or by a trusted entity affiliated with the government. The verifiers' responsibility is to link the users' Ethereum addresses to actual identification documents. In the proposed system, a user who does not have an Ethereum address will receive a new address, and therefore, a verification process is required. The verification process is typically influenced by government blockchain legislation. Accordingly, it is expected to be performed by third party entities or Decentralized Identifiers (DID) solutions such as Elastos (<https://www.elastos.org/en/>, accessed on 6 December 2022) and NEO (<https://neo.org/>, accessed on 6 December 2022). Regarding the members of the witness group, the main input to their process is the images, which are stored using IPFS. The used images are retrievable by using their hash codes that are stored on-chain. By adopting such a mechanism for storing the required documents like this, group members aim to ensure the fairness of the adopted process since the authenticity of the images is guaranteed. Furthermore, the feedback of multiple members is required to support the decentralization of the used system.

### 5. Experiments

In this section, we first give experimental settings, including the implementation of our framework. Then, we present our experimental results to analyze the cost of using the proposed system and to discuss its overall performance.

#### 5.1. Experimental Settings

The proposed framework is implemented in Solidity (0.8.17) to be executed on the Ethereum network. The implementation is performed using the hardhat development framework [22], in which the solidity optimizer is enabled to reduce the "gas" consumption. To interact with a deployed smart contract, a user must own an Ethereum address, which is associated with a balance (Ether). Users are required to have enough balance to pay for the execution of a smart contract. A user is charged based on the type and number of operations executed in a smart contract. During the presented experiments, we limit the number of users (Ethereum address) to 20. The purpose of the presented experiments is to investigate the cost of using the framework, and 20 addresses are a suitable number for such a purpose. Furthermore, the number of participants (Ethereum address) has no impact on the scalability of the proposed framework. Increasing the number of participants is only expected to increase the processing duration for the auction settlement process, which has  $\mathcal{O}(n)$  time complexity.

The proposed framework functionalities are implemented as two smart contracts, namely (1) the distributed identity's (DIDs) contract and (2) the auction contract. We grouped the discussed functionalities into two contracts to reduce the cost associated with calling functions located in different contracts. The DIDs contract maintains the participants' profiles, and this includes each participant's SI. Additionally, it is used to check whether an auction function caller has the right privileges. The auction contract has

the core functionalities of the proposed framework. Table 3 illustrates the main functions of the two contracts.

**Table 3.** Smart contract functions summary.

Contract	Function	Description
DIDs	isVerified	check if the input address belongs to a verified participant
	verifyParticipant	called by the verifier to add the input address as a verified participant
	addVerifier	called by admin to add the input address as a verifier
	removeVerifier	called by admin to remove the input address from the verifiers list
	removeParticipant	called by the verifier to remove the input address from the participants list
	calculateServiceIndex	called to recalculate the SI for the input participant (seller or buyer)
Auction	initializeAuction	initialize an auction with all required fields mentioned in Table 1
	submitOffer	store the input hashed offer
	revealOffers	check if the buyer's revealed offer matches the hashed offer
	settleAuction	settle the auction and call the distributedReward function
	distributeRewards	distribute the collected fees
	setFees	set the values for all required fees
	cancelAuction	cancel the auction
	withdrawReward	withdraw bidder share of reward

### 5.2. Cost Analysis and Discussion

This section explores the cost charges incurred by the framework participants. During this analysis, we assume that a single auction request is submitted by a buyer, where the auction duration is set to 5 min. In the presented experiments, 17 sellers interact with the submitted auction by submitting offers that satisfy the auction requirements. Additionally, one verifier is assigned to handle all verification tasks, and a single account is used by the administrator.

During the operational lifetime of an auction, each participant will be charged based on the computational and/or storage requirements for the called functions. This cost (charges) is represented as a "gas" fee. For each function call, the consumed gas is eventually converted to Ether based on the current gas price. In the presented results, the used price for the gas is 14.3 Gwei, where one Gwei is equal to ETH  $10^{-9}$ . Table 4 shows the charges associated with the performed transactions, in which the cost (USD) is obtained by using the exchange rate provided by Coinbase [23].

**Table 4.** Cost of transactions.

Transaction Type	Gas Used	Cost (ETH)	Cost (USD)	Transaction Source
submit offer	123,760	0.00177	2.241	bidders
reveal offer	100,327	0.00143	1.816	
withdraw reward	28,429	0.00041	0.514	
initialize auction	368,922	0.00527	6.681	buyer
contracts deploy	2,514,376	0.03595	45.528	admin or verifiers
add verifier	48,336	0.00069	0.875	
verify user	91,669	0.00131	1.659	
settle auction	623,308	0.00891	11.286	

From the table, we can see the settleAuction function has the highest cost. Besides the auction settlement functionalities, the settlement function calls and executes the distributedRewards function, which works iteratively to distribute the rewards. A loop statement is implemented to iterate over the entire set of participating bidders (17). Thus, the number of operations and assignments performed by this function is expected to be higher than

the rest. Furthermore, the cost of running the function is expected to grow linearly while increasing the number of bidders. Regarding the `submitOffer` and `revealOffer` functions costs, both of these functions implement a hashing algorithm. The `submitOffer` function incurs the cost of submitting the fees and all verification requirements. The `initializeAuction` function has the highest storage requirement, contributing to its cost. The cost of the contract deployment is the highest. However, this is an initialization cost, and it will be charged only once if no redeployment is needed.

Overall, the charging mechanism used by Ethereum emphasizes the importance of reducing the time and space complexity of all proposed algorithmic solutions. In this work, the presented costs can be categorized into (1) bidders' costs, (2) buyer costs, and (3) shared costs. The shared costs represent the administrative charges that must be paid using the collected fees (without contract deployment fees). Accordingly, each bidder is expected to pay USD 4.57, whereas the buyer must pay USD 6.681. The shared cost is equal to USD 13.82, which represents the cost of executing the `addVerifier`, `verifyParticipant`, and `settleAuction` functions. If each participant contributed USD 3 participating fees, this would result in USD 54 collected fees, which is enough to pay the shared cost and rewarding the eligible bidders. The value of the fees must be proportional to the maximum price determined by the buyer.

### 5.3. Participating Fee vs. Participant Reward

In the proposed system, the collected participation fees are used to reward the participants. To clarify the relationship between the collected fees and the reward mechanism, we ran experiments and assumed that 50 bidders participated in an auction. To reduce the number of stochastic variables, we assume that the winning bid value is USD 100 and the highest submitted bid is USD 110. Regarding the rest of the bids, they are randomly placed using an inverse distribution to capture the expected situation in real life. Using such distributions, the values of the bids are expected to be close to the winning bid's value. Figure 3 shows the total charges incurred by each bidder and the reward collected by the eligible bidders. In this experiment, each bidder is expected to pay USD 5 as a participation fee, where the presented charges also include the cost of interacting with the contract (Table 4). From the figure, it is obvious that the reward mechanism encourages the bidders to submit the best offers to be rewarded.

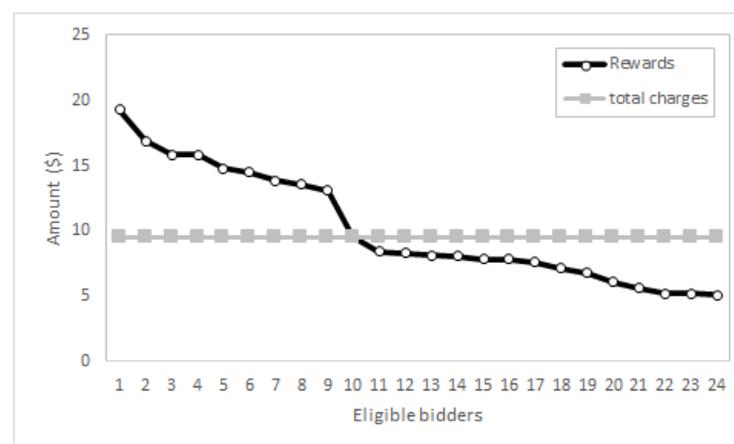


Figure 3. Participating fee vs. participant reward.

The rewarding mechanism can be employed as an optional component, triggered based on the service request details. For instance, if the requested service cost is expected to be relatively high (e.g., above USD 100), activating the rewarding mechanism is expected to attract more bidders, increasing the probability of finding a suitable service provider (seller). However, if such a cost is expected to be relatively small (a few dollars), the reward mechanism could be disabled to reduce the overall operational cost.

### 5.4. Service Index Calculation

To investigate the impact of the gap window on the SI calculation, we ran experiments for different window sizes (Figure 4). In these experiments, we assume that buyer “A” is continuously giving bidder “B” negative feedback. The initial cumulative weight for both participants is assigned to be 10, whereas the maximum cumulative weight is assigned to be 15. The presented experiments aim to capture the amount of required negative feedback to convert the “B” status from positive to undecided. As expected, increasing the gap window size reduces the amount of negative feedback to convert the “B” status to undecided. The gap window can be seen as a safety threshold. Accordingly, the results suggested that the window size should be re-adjusted based on the total number of transactions performed by participants. Waiting for 11 consecutive negative feedback reviews to change a participant’s status may reduce the mechanism’s reliability.

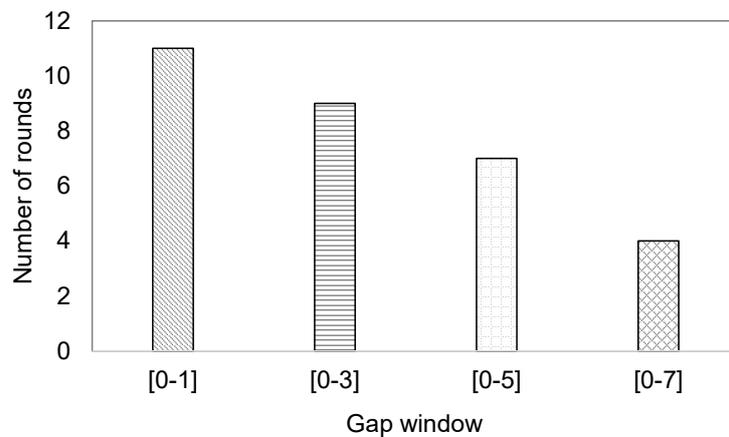


Figure 4. The impact of the gap window on the SI calculation.

To investigate the impact of the feedback provider weight on the recalculation of a participant’s weight (bidder or buyer), we ran the experiments while changing the value of  $MAX_{CW}$  (Figure 5). In these experiments, we also assume that buyer “A” provides negative feedback about bidder “B”. In addition, the cumulative weight for both participants is assigned to be 10. As expected, the highest reduction in the cumulative weight of bidder “B” occurs when the cumulative weight for buyer “A” is equal to  $MAX_{CW}$  (10). The weight of the feedback provider plays an essential role in the SI calculation. However, to ensure fairness, the violation dispute component can be used to ensure that any negative feedback is justifiable.

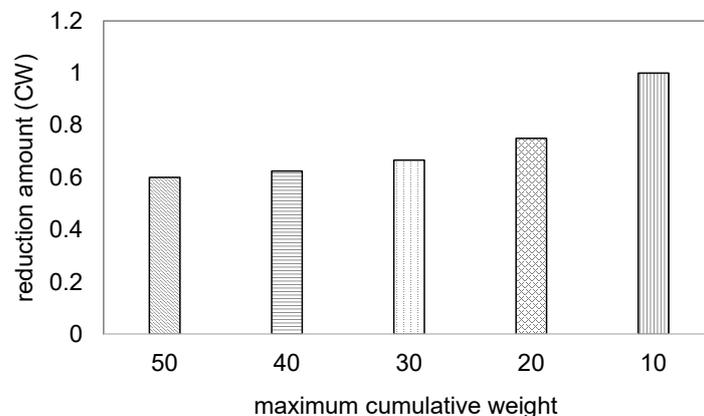


Figure 5. The impact of the feedback provider weight on the recalculation of the participant cumulative weight.

## 6. Conclusions

In this paper, we have investigated the use of a blockchain-based, sealed bid, reverse auction framework. This proposed framework is designed for situations where the auctioned items represent logistics services provided by sellers. Examples of these services include renting a physical space or transportation, as the trade volume for such services is anticipated to be relatively high. Thus, scalability and a low time complexity are crucial factors in the design of the proposed framework. As discussed in the results section, these factors are addressed in the presented framework by reducing the overall number of transactions. Additionally, the architecture of such a framework is expected to increase participants' trust in the system, potentially encouraging more participants to join the auction. To further motivate participants, a user-reward distribution strategy encourages participants to make the most competitive offers in order to receive the reward. Moreover, we have demonstrated that a bidder is expected to pay around USD 5.5 to participate in an auction, a cost that is considered affordable compared to typical administrative fees in traditional e-auctions.

As part of our future work, we plan to explore the use of blockchain to support coalitions between bidders in situations where the requested service exceeds the capability of a single bidder. In such a scenario, multiple bidders could form a coalition and submit a single joint offer. Blockchain technology could also facilitate communication among coalition members. In addition, we aim to automate the process of violation disputes by eliminating the need for a witness group. This can be achieved by integrating an IoT component to collect all necessary information related to the requested services. Consequently, a lightweight smart contract could use this collected data to automate the violation dispute process. Regarding open bid forward auction scenarios, employing our proposed framework will incur additional costs, as traders are charged for each submitted bid on a public blockchain. Therefore, we plan to explore the necessary modifications to apply our proposed framework in such an auction domain.

**Author Contributions:** Formal analysis, K.A., M.A.A., Y.C.L., T.H.R. and A.P.; methodology, K.A., M.A.A., Y.C.L. and T.H.R.; validation, K.A., M.A.A., Y.C.L. and A.P.; supervision, K.A., Y.C.L. and T.H.R.; writing—original draft, K.A. and T.H.R.; writing—review and editing, K.A., M.A.A., Y.C.L., T.H.R. and A.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Higher Colleges of Technology (HCT) under grant 113611.

**Data Availability Statement:** This study does not involve the creation of new data and therefore data sharing is not applicable.

**Acknowledgments:** This publication was supported by grant number 113611 from Higher Colleges of Technology (HCT). Its contents are solely the responsibility of the authors and do not necessarily represent the official views of HCT.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Sunyaev, A. Distributed Ledger Technology. In *Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies*; Springer International Publishing: Cham, Switzerland, 2020; pp. 265–299. [[CrossRef](#)]
2. Mohanta, B.K.; Panda, S.S.; Jena, D. An Overview of Smart Contract and Use Cases in Blockchain Technology. In Proceedings of the 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bengaluru, India, 10–12 July 2018; pp. 1–4. [[CrossRef](#)]
3. Cai, W.; Wang, Z.; Ernst, J.B.; Hong, Z.; Feng, C.; Leung, V.C.M. Decentralized Applications: The Blockchain-Empowered Software System. *IEEE Access* **2018**, *6*, 53019–53033. [[CrossRef](#)]
4. Wu, S.; Chen, Y.; Wang, Q.; Li, M.; Wang, C.; Luo, X. CReam: A Smart Contract Enabled Collusion-Resistant e-Auction. *IEEE Trans. Inf. Forensics Secur.* **2019**, *14*, 1687–1701. [[CrossRef](#)]
5. Shi, Z.; de Laat, C.; Grosso, P.; Zhao, Z. Integration of Blockchain and Auction Models: A Survey, Some Applications, and Challenges. *IEEE Commun. Surv. Tutor.* **2022**, *25*, 497–537. [[CrossRef](#)]
6. Krishna, V. *Auction Theory*; Elsevier: Amsterdam, The Netherlands, 2010.

7. Cui, X.; Lai, V.S.; Lowry, P.B.; Lei, Y. The effects of bidder factors on online bidding strategies: A motivation-opportunity-ability (MOA) model. *Decis. Support Syst.* **2020**, *138*, 113397. [[CrossRef](#)]
8. Sarfaraz, A.; Chakraborty, R.K.; Essam, D.L. A tree structure-based improved blockchain framework for a secure online bidding system. *Comput. Secur.* **2021**, *102*, 102147. [[CrossRef](#)]
9. Chen, E.; Qin, B.; Zhu, Y.; Song, W.; Wang, S.; Chu, C.C.W.; Yau, S.S. SPESC-Translator: Towards Automatically Smart Legal Contract Conversion for Blockchain-Based Auction Services. *IEEE Trans. Serv. Comput.* **2022**, *15*, 3061–3076. [[CrossRef](#)]
10. Shih, D.H.; Wu, T.W.; Shih, M.H.; Tsai, W.C.; Yen, D.C. A Novel Auction Blockchain System with Price Recommendation and Trusted Execution Environment. *Mathematics* **2021**, *9*, 3214. [[CrossRef](#)]
11. Lee, J.S.; Chew, C.J.; Chen, Y.C.; Wei, K.J. Preserving Liberty and Fairness in Combinatorial Double Auction Games Based on Blockchain. *IEEE Syst. J.* **2021**, *15*, 3517–3527. [[CrossRef](#)]
12. Liu, L.; Du, M.; Ma, X. Blockchain-Based Fair and Secure Electronic Double Auction Protocol. *IEEE Intell. Syst.* **2020**, *35*, 31–40. [[CrossRef](#)]
13. Blass, E.O.; Kerschbaum, F. Strain: A Secure Auction for Blockchains. In *Computer Security, Proceedings of the 23rd European Symposium on Research in Computer Security, ESORICS 2018, Barcelona, Spain, 3–7 September 2018*; Lopez, J., Zhou, J., Soriano, M., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 87–110.
14. Chen, Y.H.; Chen, S.H.; Lin, I.C. Blockchain based smart contract for bidding system. In *Proceedings of the 2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, Japan, 13–17 April 2018*; pp. 208–211. [[CrossRef](#)]
15. Braghin, C.; Cimato, S.; Damiani, E.; Baronchelli, M. Designing Smart-Contract Based Auctions. In *Security with Intelligent Computing and Big-Data Services, Proceedings of the Second International Conference on Security with Intelligent Computing and Big Data Services (SICBS-2018), Guilin, China, 14–16 December 2018*; Yang, C.N., Peng, S.L., Jain, L.C., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 54–64.
16. Omar, I.A.; Hasan, H.R.; Jayaraman, R.; Salah, K.; Omar, M. Implementing decentralized auctions using blockchain smart contracts. *Technol. Forecast. Soc. Chang.* **2021**, *168*, 120786. [[CrossRef](#)]
17. Pop, C.; Prata, M.; Antal, M.; Cioara, T.; Anghel, I.; Salomie, I. An Ethereum-based implementation of English, Dutch and First-price sealed-bid auctions. In *Proceedings of the 2020 IEEE 16th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romani, 3–5 September 2020*; pp. 491–497. [[CrossRef](#)]
18. Wang, D.; Zhao, J.; Mu, C. Research on Blockchain-Based E-Bidding System. *Appl. Sci.* **2021**, *11*, 4011. [[CrossRef](#)]
19. Chowdhury, M.J.M.; Ferdous, M.S.; Biswas, K.; Chowdhury, N.; Kayes, A.S.M.; Alazab, M.; Watters, P. A Comparative Analysis of Distributed Ledger Technology Platforms. *IEEE Access* **2019**, *7*, 167930–167943. [[CrossRef](#)]
20. Benini, A.; Chataigner, P.; Noumri, N.; Parham, N.; Sweeney, J.; Tax, L. *The Use of Expert Judgment in Humanitarian Analysis—Theory, Methods, Applications*; Assessment Capacities Project—ACAPS: Geneva, Switzerland, 2017.
21. Chua, C.; Wareham, J. Fighting Internet auction fraud: An assessment and proposal. *Computer* **2004**, *37*, 31–37. [[CrossRef](#)]
22. Foundation, N. Available online: <https://hardhat.org/> (accessed on 6 December 2022).
23. Coinbase. Available online: <https://www.coinbase.com> (accessed on 6 December 2022).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.