

Article

Reduction of Online Fraudulent Activities in Freelancing Sites Using Blockchain and Biometric

Amreen Batool and Yungcheol Byun * 

Department of Computer Engineering, Major of Electronic Engineering, Jeju National University, Institute of Information Science & Technology, Jeju 63243, Korea; amreen.cs@yahoo.com

* Correspondence: ycb@jejunu.ac.kr

Abstract: Digital freelancing refers to hiring potential employees to work on specific projects on a contract basis by organizations, companies, and individuals. Existing freelancing systems are based on client-server models and managed in a centralized way. Centralized freelancing systems are originating several issues: lack of trust, payment delays, central control over all operations, less reliability, online frauds, and lack of fairness. Hence, we propose a decentralized freelancing system based on blockchain and biometrics. Blockchain in the freelancing system makes contracts decentralized, and no one can control the interaction and deals between buyers and sellers. Biometric signatures on smart contracts in the freelancing system enhance the authenticity and validity of agreements between buyers and sellers. Our decentralized freelancing system is an online web-based platform based on blockchain and biometrics and built on Python. It makes the interaction and deals between buyers and sellers via the innovative approach of smart contracts. The proposed work illustrates the development, implementation, and functionalities of the freelancing system, such as registration as buyer or seller, logging in to the system, posting a job, bidding on jobs, hiring process, finalizing deals with smart contracts having a biometric signature of both parties and distributing smart contracts among all stakeholders via blockchain. Our system proves the fairness, clarity, security, and reliability in all processes of posting and hiring without the involvement of any central management authority.

Keywords: blockchain; biometrics; decentralized freelancing; smart contract; biometrics



Citation: Batool, A.; Byun, Y. Reduction of Online Fraudulent Activities in Freelancing Sites Using Blockchain and Biometric. *Electronics* **2022**, *11*, 789. <https://doi.org/10.3390/electronics11050789>

Academic Editors: Ioannis Mavridis, Spyros Voulgaris and Panagiotis Fouliras

Received: 25 January 2022

Accepted: 28 February 2022

Published: 3 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 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

Freelancing is a contract-based job where companies, organizations, or individuals recruit potential professionals [1–3]. Companies hire freelancers according to projects via online platforms specifically designed for said purpose. Based on their expertise, skills, experience, and knowledge, freelancers complete the assigned job and charge from organizations against the provided services [4]. Freelance professionals are always free to sign projects with multiple clients without remaining committed to a single company or organization simultaneously [5,6]. Clients usually post a job in online freelancing platforms, and freelancers bid on several jobs. Clients hire a freelancer for their projects based on several characteristics, such as experience and freelancers' ratings [7]. The global freelance workforce is increasing rapidly. The study found that approximately 51% of the population will become freelance professionals in several developed countries in the coming years [8].

Currently, all major freelancing platforms, such as Upwork, Fiverr, and Guru, are based on traditional client-server models and managed centrally by officials of said organizations [9,10]. In the client and server architecture, organizations use servers to host their web applications and provide access to those servers to online communities worldwide [11]. People connect with that server to use the services provided by organizations. Existing client-server architecture allows the complete control and access to data of all users to organizations [12]. In centralized systems, organizations can alter their policies, such as

terms and conditions, transactions fees, and other statements, at any time to facilitate the users or handling of complaints. However, several issues are associated with these centralized platforms, such as the owners of centralized freelancing platforms charging high fees, lack of fairness, and deleting any user's account from platforms without any intimation or notice [13]. The possibilities of online scams and fraud are always present in centralized freelancing platforms. Several studies suggest the decentralized freelancing platform avoids payment delays and unfair treatment with accounts [14]. In contrast, decentralized system nodes are not dependent on a single master node; control is spread among many nodes. Cryptocurrency service suppliers, such as Bitcoin, Litecoin, Ethereum, and many other providers use this model [15,16].

A blockchain is a decentralized ledger holding records of transactions in blocks, and all blocks are connected via a chain [17]. The concept of blockchain was introduced and developed in 2008 [18]. A wide range of features of blockchain technology [19] make it the most demanding technology in several application areas, such as the government sector, education sector, business sector, healthcare sector, and law and judicial sector [20–24]. The previous study developed a decentralized freelancing system named Hire Chain, based on the Ethereum blockchain using solidity [25]. In the Hire Chain project, developers assign the Ethereum address to each user consisting of public and private keys generated with an elliptical curve cryptography algorithm [26,27]. The decentralized system based on blockchain technology was proposed to protect temporary workers' rights and manage their employment [28,29]. White et al. [30] introduced a decentralized network-based fundraising and a freelancing system where financiers use the IRT protocol token to deliver the funds for the projects submitted by clients. Lim et al. [31] proposed a solution based on blockchain to improve a reputation system in the gig economy to build trust between freelancers and clients. Rodrigo et al. [32] introduce a blockchain system that allows smart contract to access and use real-world information within the blockchain. Oscar et al. [33] discuss the opportunities and challenges in integrating blockchain and biometrics, emphasizing biometric template storage and protection. Recently, a study proposed an advanced method of implementing a collaborative business process based on the Hyperledger fabric permissioned blockchain to improve centrally controlled business model performances and overcome the most severe issues of centralized systems [34–36].

However, no study reports a decentralized freelancing system using blockchain where the biometric signature on smart contracts helps to reduce the percentage of online fraud in any means. In our research work, we use a survey through Google forms to ask freelancers whether they face any online fraud in their freelancing career or not. Most freelancers reported, as given in Appendix A, that they face sudden changes in project requirements after the delivery of projects, and as a result, they lose their money and the deletion of accounts after the bogus complaint to the freelancing platform management from the client's side. Based on the opinions of the freelance community in an online survey, we extract the functional requirements of our system, such as the fact that the system should be reliable, secure, transparent, and unbiased from all aspects primarily related to user account treatments.

In the present study, we propose a decentralized freelancing system using blockchain and smart contracts to solve the above stated issues of centralized freelancing platforms. Biometric signatures on smart contracts and the further distribution of those smart contracts via decentralized blockchain technology mitigate fraud in freelancing platforms, such as once the deal is conducted via smart contracts, both parties, i.e., clients and freelancers, become unable to change the content of the agreement on the smart contract, as, otherwise, all parties are notified about the alteration. The users of our proposed system are buyers and sellers. Buyers can post a job on the platform with specific details about the projects, such as the job's budget, timeline, and requirements. Sellers are freelancing professionals who join our system to sell their services. The proposed system will remove the burden of high fees from clients and freelancers. The system will provide equal opportunities to both clients and freelancers to bid for projects by reducing scams and fraudulent activities.

2. Methodology and Development of System Design

2.1. Methodology

To develop the proposed decentralized freelancing platform system, we followed the agile software development methodology explained somewhere else [37]. Agile software development methodology has five phases of system development. Each phase continues the development process towards completion and remains concurrent with other phases. Table 1 describes the connection of agile methodology with the development of the proposed decentralized freelancing system.

Table 1. Connection of agile methodology with the proposed decentralized freelancing system.

Agile Methodology Phases		Tools and Technology	Timespan (Days)
Brainstorm	✓	Online survey on Google forms and literature review	60
System Design	✓ ✓	Unified modeling language for high-level sketching of system design Drawing and prototyping tools	60
System Development	✓ ✓ ✓	Python programming language Hypertext Markup language (HTML) and Cascaded Style Sheets (CSS) for frontend development of the system. ChainSQL database API with MySQL at the backend.	120
Quality Assurance	✓ ✓	Security and Reliability quality test Speed metric, efficiency, and accuracy tests	15
Launching	✓	Running on the localhost and accessing the interface of the system on the browser	5
Documentation/dissertation writing	✓	Microsoft Office (MS Word)	60

2.2. System Design and Development

The system design included the different perspectives of several system prototypes, including the system's working process for both types of users, i.e., buyers and sellers, layered architecture of the system, database design, profile creation, job posting, bid placing, and project submission.

2.2.1. Transaction Processing Architecture of the Proposed System

The Hyperledger fabric blockchain platform has three steps in the unique transaction processing mechanism as shown in Figure 1:

- ✓ In the first phase of the transaction, systems users initialize the transaction with some necessary parameters for transaction identification, including Chain Code ID, Timestamp, and transaction payload.
- ✓ In the next phase, a verification signature is generated by approving nodes in the network to ensure the authenticity of the user, i.e., whether they are authorized to perform transactions in the system.

- ✓ After the signature verification of the user, the user is allowed to propose several parameters specified in the system, such as the identity of the user, signature of the user, and channel identity. Channel identity is referred to as an address of private blockchain between two or more users involved in the project, and they can share the ledger privately with other users of the system.
- ✓ After broadcasting the required information in the system, the validity is confirmed by time status, i.e., assigned or not assigned, before engaging the professionals with that broadcasted job.
- ✓ After the confirmation of validity, the user is allowed to participate in it. After the necessary documentation, such as deal, time, and agreements, the transaction becomes part of the blockchain and smart contracts are executed when conditions are met successfully as stated in the agreement form.

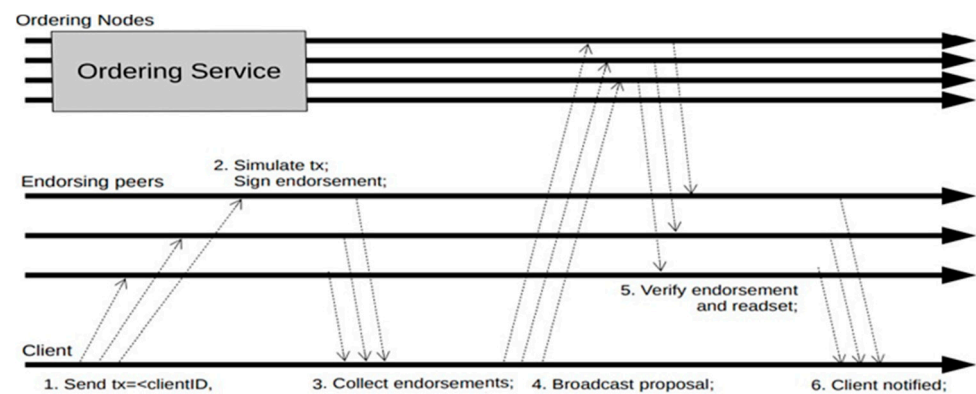


Figure 1. General overview of the Hyperledger fabric transaction process.

According to the above-described details, the three steps involved in transaction processing in our platform are given in Figure 2:

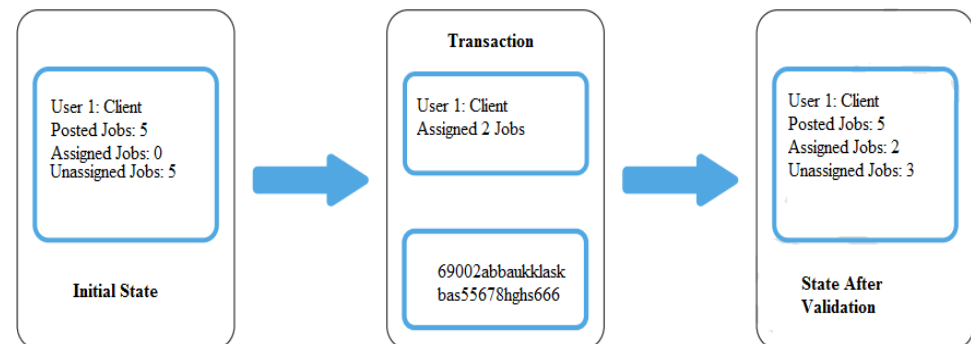


Figure 2. Validation phase of the transaction.

Order: Order the information in the required form, i.e., entering job specifications, identity, cost matrix, and timespan required to complete the job.

Validate: Checking the job validation, i.e., either available or assigned and active or canceled. When ordering the transaction, the confirmation of posting authority with identity and other details are specified. Validation steps help to avoid the double-spending problem, i.e., making the same deal with more than one professional in our system case. Figure 2 illustrates the validation of the transaction in the case of our system.

Execute: The execution of an agreement process after the confirmation of the transaction's validity by sending a proposal and accepting the proposal by the other side.

The unique three-step transaction processing mechanism in the Hyperledger fabric platform offers several key benefits to our system.

2.2.2. Work Process of System

As described earlier, there were two main system actors: seller and buyer. Sellers are freelancing professionals in our proposed system and clients are buyers. Both types of system actors have different work processes in the system. First, we define the buyer's work process in the system. Figure 3 illustrates the buyer work process.

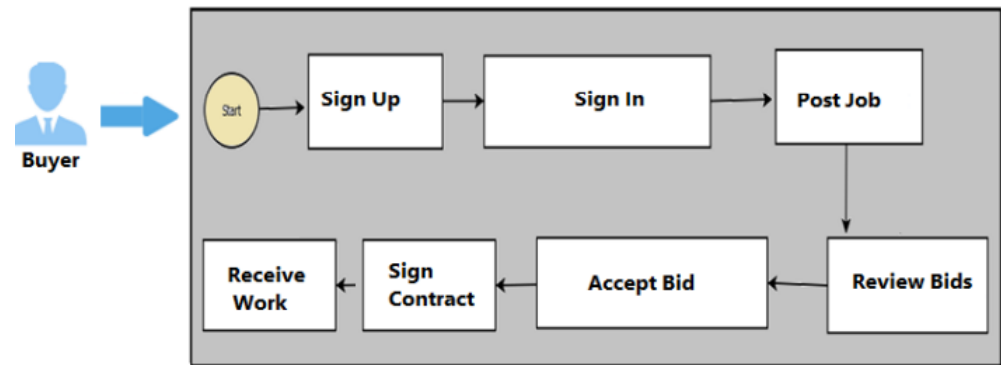


Figure 3. Buyer work process in the system.

Figure 3 shows that, at the time of starting with the system, buyers sign up with the system by providing the required details. After the successful registration with the system, the buyer logs into the system by providing valid credentials. After logging in to the system, buyers post the job. After posting the job, the buyer must wait for some operation from the seller's side. The seller operations on the system is given in Figure 4.

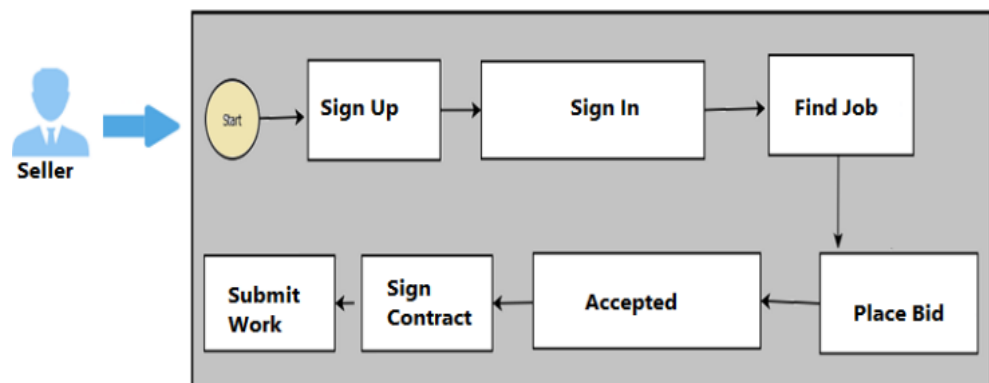


Figure 4. Seller work process in system.

As in the case of the buyer, the seller also signs up and signs in to successfully start the system as shown in Figure 4. After signing into the system, the seller can find jobs posted by buyers, as discussed earlier. After searching for a job according to interest and professional field, they can place a bid on the job. More than one seller can place a bid on a single job. Buyers can review all bids placed by several sellers on the posted job. Based on some personal criteria, the buyer accepts a single proposal out of many. The seller will be notified accordingly upon acceptance by the buyer. The buyer initialized the agreement by accepting the bid of the seller. After that, the seller responds to the understanding and places the biometric signature on the smart contract. After this step, the transaction will become part of the blockchain, and the information will be dispatched to each stakeholder via blockchain. One copy of the data will also be saved into the database in hashed format for recovery if required. After completing the job, the seller will submit the job to the buyer based on the job specifications described earlier in the agreement. If the requirements are fulfilled in the submitted work, the smart contract will be executed, and the state of the transaction will be changed accordingly.

2.2.3. Layered Architecture of the System

The presentation, application, and data access layers in the proposed decentralized freelancing system are shown in Figure 5. From the user's perspective, the presentation layer is a high-level layer that interacts with the system. Users perform required operations on a system via the presentation layer. The application layer holds the system's core libraries, binaries, and coding between the system database and the interface. The innermost layer of the system is the data access layer. From the user's perspective, the data access layer is a low-level layer and holds the actual database schema. As the name suggests, the data access layer is responsible for accessing the data from the database according to the queries generated by users from the presentation layer. After accessing the data from the database, the data access layer transmits data to the layer above it, i.e., the application layer. The application layer shows the data to the presentation layer according to rules defined in the application layer. The layered model of the system, discussed specifically from blockchain architecture, has four layers and holds some differences from the traditional layered architecture. Blockchain layered architecture consists of a presentation, decentralized application, and hardware [38].

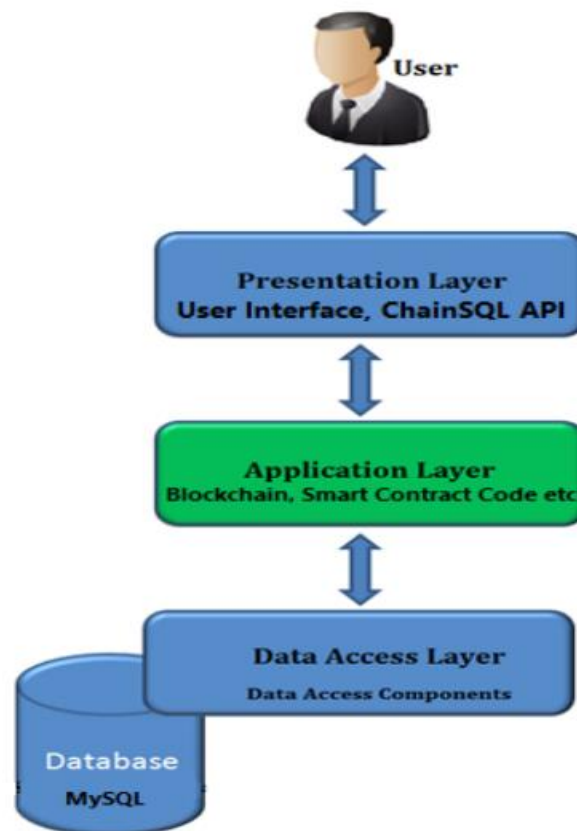


Figure 5. Three-layered architecture of the system.

2.2.4. Database Design

ChainSQL has blockchain integrity and can process queries of the traditional database systems quickly [39]. The building blocks of ChainSQL include application interface, network consensus, database operations, and access grants. The application interface provides access to the blockchain. Our proposed decentralized freelancing system sends commands in SQL format. The given commands are wrapped up in transaction format by the application interface and dispatched to the Blockchain network for agreement. Network consensus is paramount to data integrity. If agreement on the transaction is not achieved, the transaction is restricted to become a part of the blockchain. Precisely to our system context, if an agreement is not achieved between buyer and seller after the

buyer accepts the seller's bid, all existing operations are rolled back, and the transaction does not change its state from available to assigned. The transaction will become part of the blockchain and database once an agreement between both parties is achieved. After achieving a consensus, the blockchain transmits the data of transactions to the concerning database. The database maintains the record of all transaction data and can be retrieved when required via a traditional database's easy query processing mechanism. Figure 6 illustrates the database design of the system. The owner can authorize or deauthorize the rights of several operations via simple SQL commands.

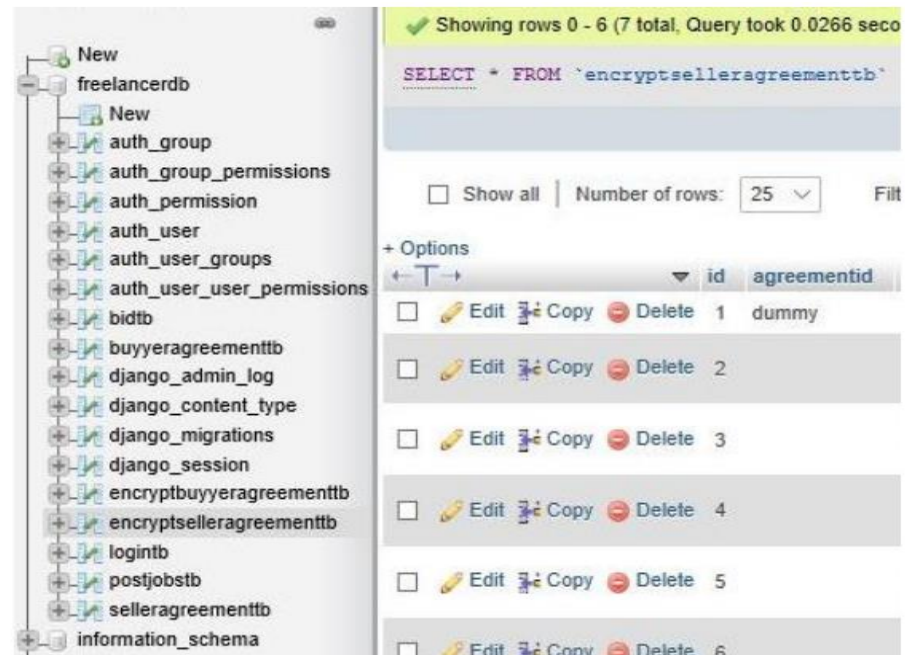


Figure 6. Backend connected database design of the system.

2.2.5. Prototype Design

Prototype Design

Figure 7a describes the steps required for creating the user profiles. During the sign-up phase, all the details about users are stored in the data store. The unique user id must be guaranteed with hashing. Detailed information is required to register with the system for both users, i.e., buyers and sellers. The information that is collected from the users at the time of profile creation is name, username, user type, email, phone, and password. Figure 7b illustrates the registration interface of the system. Upon successful registration with the system, the user can log into the system by providing the valid credentials set at registration. The login interface of the system is shown in Figure 7c.

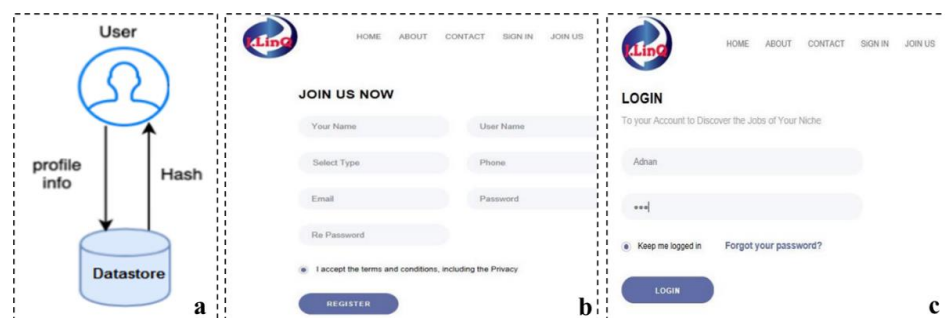


Figure 7. Prototype design includes (a) users profile creation in the system, (b) registration interface of the system, and (c) login interface of the system.

Job Posting and Bid Placing on Job

Figure 8 displays the graphical representation of the job posting and bid placing mechanism in the system. For posting a job in the system, the buyer provides the required details and information about the job, including the job's title, the job's budget, the time required to complete the job, skills needed to handle the job, and a detailed description about the job. Once the required details are provided, the job is appended into the database and becomes visible in the uploaded, but unassigned jobs list. All details entered with the job become public and visible to every seller. Sellers can select a job according to their skills and expertise from the list and place a bid. The freelancer submits a detailed proposal and bid amount on the posted job to establish a bid on the job.

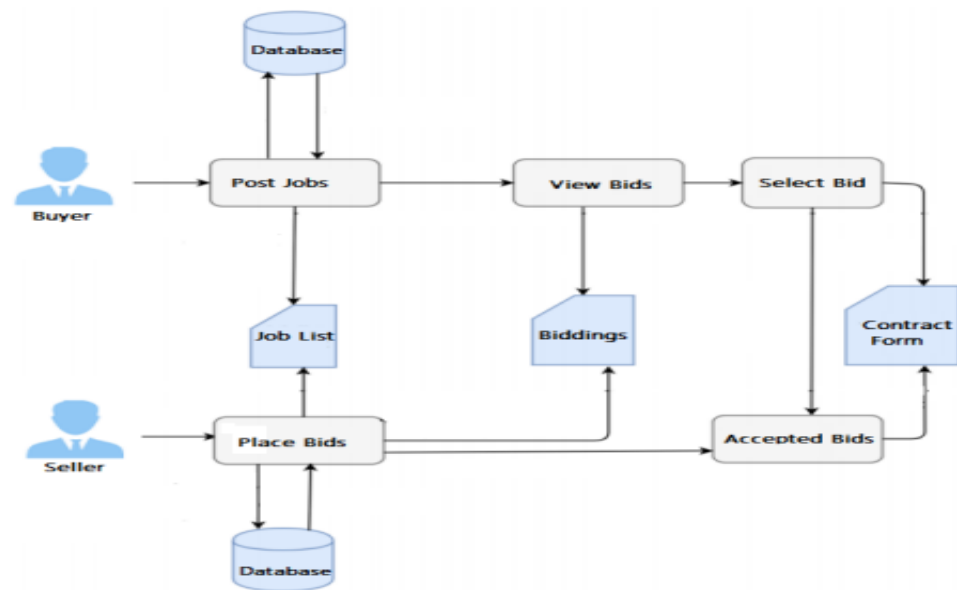


Figure 8. Job posting and bid placing on jobs in the system.

Both buyers and sellers can view their activities on the system via a separate dashboard developed for both buyers and sellers. From the respective dashboard, the buyer can see the bids placed by all sellers on a particular job with a detailed proposal and bidding amount from the seller's side.

Job proposals are stored in a database, and buyers can retrieve the proposal information by using its hash value. After reviewing all the bids placed on the job posted by the buyer, the buyer will select the best seller based on several parameters, i.e., experience, qualification, proposal statement, age, gender, and interest. The system does not specify these parameters for choosing the best seller because every buyer can choose the seller according to their criteria. When the buyer decides the appropriate seller, the seller is notified accordingly, and the agreement process starts with the contract form. A contract form is shown to both parties. The contract form holds a detailed description of the job, including time, budget, requirements, and other details. At first, the buyer places the fingerprint on an electronic biometric reader device to place the biometric signature on the contract form. After carefully reading the contract form, the seller must establish a biometric signature, i.e., fingerprint on the contract form, to make the deal. Once the deal is achieved, the posted job's transaction state is changed from unassigned to assigned and becomes invisible from the available job list. If the seller did not sign the contract for any reason, then, they can discard the contract form. In this case, the state of the posted job transaction will remain the same as it was previously, i.e., unassigned, and stay visible in the available jobs list so other sellers can place a bid on that job to complete the job against money.

Job Submission after Completion

Figure 9 illustrates the job submission after completion of the prototype design. After successfully matching the job requirements, the job is declared completed from the seller's side. The seller submits the completed job via the interface provided at the respective dashboard. The buyer will be notified accordingly after submitting the job files marked as completed by the seller.

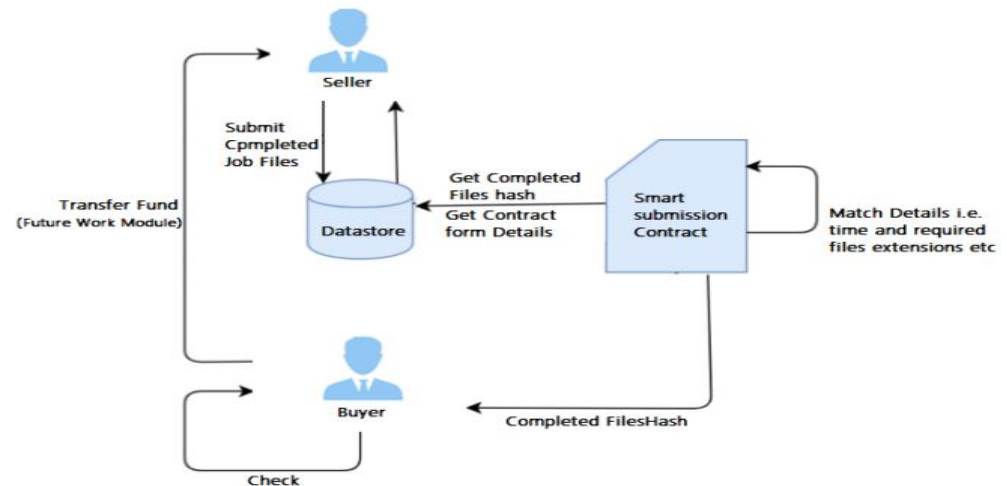


Figure 9. Prototype design of submission of a job after completion.

The smart submission contract in this diagram is a code that validates specific details from the smart contract form signed by the buyer and seller with the biometric signatures as described in the previous section. Smart submission contract marks the job completion time and requirements on the contract form and displays the report to buyer, i.e., valid or invalid. If the job is marked valid and completed by smart submission contract, it retrieves the submitted files and sends them to the buyer. The buyer checks the files submitted by the seller and confirms the job completion.

3. Results and Discussion

3.1. User Registration and Login

Users can be registered via accessing the user registration interface of the system. The system allows two types of users to register, either buyer or seller. Therefore, the registration page provides a list box for selecting appropriate user types during registration with the system. Figure 10a shows the system's user registration web interface, which allows the user to open the account in our blockchain and biometric-based decentralized freelancing system. After registering and login, they can check the opportunities in the system according to their role.

JOIN US NOW

Username: Ambreen

Password: Ambreen512

Confirm Password: +923465765012

Select Type:
 Select Type
 Buyer
 Seller

☒ I accept the terms and conditions, including the Privacy

REGISTER

a

LOGIN

To your Account to Discover the Jobs of Your Niche

Username: Ambreen512

Password: *****

☒ Keep me logged in [Forgot your password?](#)

LOGIN

b

Figure 10. (a) User registration and (b) login interface of the system.

The required fields for successfully registered users are name, user id, type, mobile number, and password. Submitted user details are stored in the database. Additionally, with each agreement in the system, a hash of user information is associated with the transaction to identify the user based on unique constraints, such as email, user id, and mobile number. The user registration form requires only basic information from users. Users can modify their information and add extra information in their profile, i.e., profile pictures and addresses, from their private dashboards generated for each user upon successful registration with the system. Users can land on their dashboards after the successful login to the system. Figure 10b illustrates the login interface of the system. The user has to enter valid credentials to access the private dashboard and profile at the system's login interface.

3.2. Users Dashboard and Profile

3.2.1. Seller Dashboard

After logging into the system, the user can perform the operations according to their role from the dashboard. Each user, i.e., seller and buyer, has different performance options in the system dashboard. Figure 11 shows the seller dashboard and profile after logging into the system.

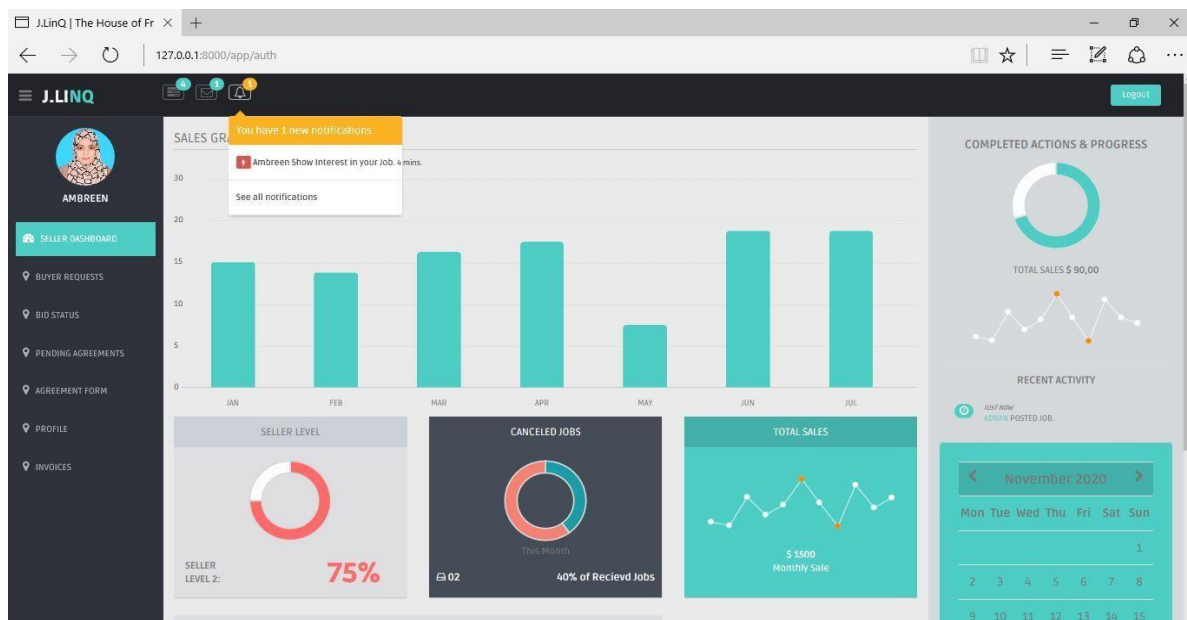


Figure 11. Seller dashboard and profile.

User dashboard provides the facility for users to perform almost all system functionalities for which they join the system. As shown in Figure 11, the user dashboard allows the users to access different operations, including the current month's statistics, messages, notifications, total sales, an average of canceled sales, recent activities in the system, bid status, pending agreements, invoices, buyers request, and calendar, on the first page of the dashboard.

3.2.2. Buyer Dashboard

The dashboard of buyers is different from the seller from several perspectives according to the changing nature of the role. Figure 12 illustrates the system's buyer dashboard.

The proposed system allows the buyers to perform different dashboard tasks, including posting jobs, accessing seller's requests, reviewing the agreements and invoices.



Figure 12. Buyer dashboard and profile.

3.3. Job Posting

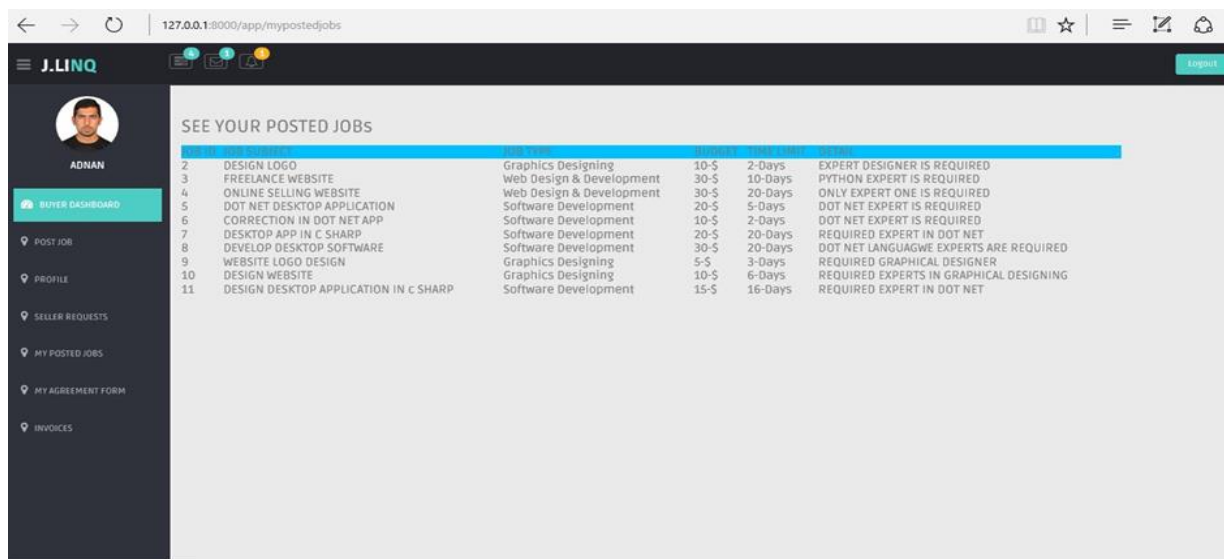
The system provides the interface to the buyer for posting the job from the dashboard. Figure 13 shows the job posting interface of the system for buyers.



Figure 13. Job posting interface of the system.

Posting a job on the platform system requires some mandatory input fields to be completed to add job details, such as the subject of the job, nature of the job, budget, time duration for completing the job, job description, and files of job to upload if required. After entering the data in all the necessary input fields, the user can post the job. All posted jobs by the same buyer are visible in the form of a list at the next tab of the dashboard. Figure 14 shows the list of all posted jobs by the same buyer.

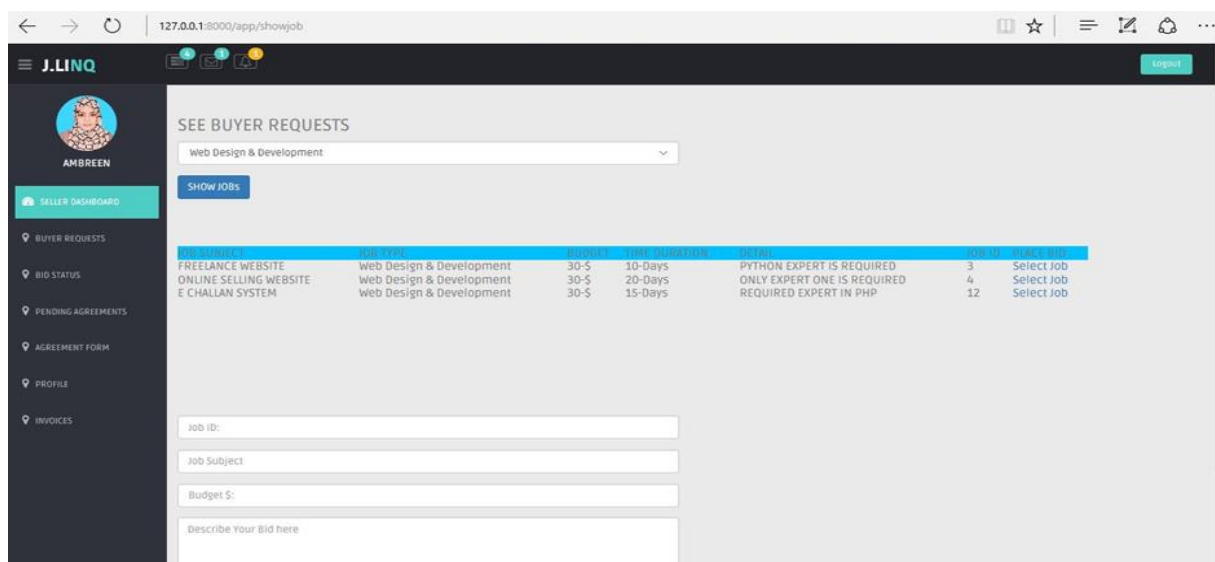
This list of jobs is public and visible to all sellers in the system. Sellers can filter the job searching according to their expertise, and only filtered jobs can become visible to those according to their search criteria. Figure 15 illustrates the interface of job searching from the seller dashboard.



The screenshot shows the 'SEE YOUR POSTED JOBS' section of a buyer's dashboard. It features a table with 11 job listings. Each listing includes a job ID, a description, the job type, budget, time duration, and specific requirements.

Job ID	Job Description	Job Type	Budget	Time Duration	Requirements
2	DESIGN LOGO	Graphics Designing	10-\$	2-Days	EXPERT DESIGNER IS REQUIRED
3	FREELANCE WEBSITE	Web Design & Development	30-\$	10-Days	PYTHON EXPERT IS REQUIRED
4	ONLINE SELLING WEBSITE	Web Design & Development	30-\$	20-Days	ONLY EXPERT ONE IS REQUIRED
5	DOT NET DESKTOP APPLICATION	Software Development	20-\$	5-Days	DOT NET EXPERT IS REQUIRED
6	CORRECTION IN DOT NET APP	Software Development	10-\$	2-Days	DOT NET EXPERT IS REQUIRED
7	DESKTOP APP IN C SHARP	Software Development	20-\$	20-Days	REQUIRED EXPERT IN DOT NET
8	DEVELOP DESKTOP SOFTWARE	Software Development	30-\$	20-Days	DOT NET LANGUAGE EXPERTS ARE REQUIRED
9	WEBSITE LOGO DESIGN	Graphics Designing	5-\$	3-Days	REQUIRED GRAPHICAL DESIGNER
10	DESIGN WEBSITE	Graphics Designing	10-\$	6-Days	REQUIRED EXPERTS IN GRAPHICAL DESIGNING
11	DESIGN DESKTOP APPLICATION IN C SHARP	Software Development	15-\$	16-Days	REQUIRED EXPERT IN DOT NET

Figure 14. List of the posted job by the same buyer in the platform.



The screenshot shows the 'SEE BUYER REQUESTS' section of a seller's dashboard. It includes a dropdown menu for selecting a job area (currently set to 'Web Design & Development') and a 'SHOW JOBS' button. Below this is a table of job requests with columns for job ID, description, job type, budget, time duration, requirements, and a 'Select Job' button. At the bottom, there is a form for placing a bid, including fields for Job ID, Job Subject, Budget \$, and a description box.

Job ID	Job Description	Job Type	Budget	Time Duration	Requirements	Job ID	Select Job
3	FREELANCE WEBSITE	Web Design & Development	30-\$	10-Days	PYTHON EXPERT IS REQUIRED	3	Select Job
4	ONLINE SELLING WEBSITE	Web Design & Development	30-\$	20-Days	ONLY EXPERT ONE IS REQUIRED	4	Select Job
12	E CHALLAN SYSTEM	Web Design & Development	30-\$	15-Days	REQUIRED EXPERT IN PHP	12	Select Job

Figure 15. Job searching interface in the seller dashboard.

As shown in Figure 15, the seller selects the job area of web design and development from all categories. The system filters the jobs related to the specified area and gives the option to bid on several jobs to the seller. Required input fields for placing the bids on the job are the budget specification and description of the bid. After completing the fields needed for placing the bid, the seller confirms the bid placement. These bids become visible to the buyer and notify the buyer in the notification bar of the system.

3.4. Approving the Placed Bids on Jobs

Once the seller places the bid on a job, the next step towards finalizing the deal is the approval of the bid from the buyer's side. The system notifies the buyers when sellers bid on their posted job. More than one seller can place a bid on a single job until the buyer accepts a bid and the state of the posted job transaction changes from unassigned to assigned. Figure 16 illustrates the interface where buyers can review the bids placed on their job.



ID	Seller Name	Bid Description	Budget	Status
2	ambreen	I WILL DO IT WITHIN TIME WITH 100 % ACURRACY	10-\$	2-Days Approved Proceed To Agreement
4	ambreen	I WILL DO IT	30-\$	20-Days Approved Proceed To Agreement
5	ambreen	I CAN DO IT	20-\$	5-Days Approved Proceed To Agreement
6	ambreen	I AM EXPERT IN DOT NET	10-\$	2-Days Approved Proceed To Agreement
7	ambreen	I HAVE AN EXPERIENCE OF 10 YEAR IN DOT NET LANGUAGE	20-\$	20-Days Approved Proceed To Agreement
8	ambreen	I HAVE 10 YEAR EXPERIENCE IN DOT NET LANGUAGE	30-\$	20-Days Approved Proceed To Agreement
9	ambreen	I HAVE AN EXPERIENCE IN THIS FIELD	5-\$	3-Days Approved Proceed To Agreement
11	ambreen	I AM EXPERT IN DOT NET LANGUAGE	15-\$	16-Days Approved Proceed To Agreement

Figure 16. A bids review interface where bids are placed on jobs posted by the buyer.

The interface shown in Figure 16 shows the bids placed by the seller on jobs. The seller's name, bid description, and budget quotes are also visible. The buyer can review the bids by keeping several aspects in mind, i.e., the qualification and efficiency level of the seller. The buyer can view the details about the seller by simply clicking on the seller's name at this stage. Once the seller's proposal is approved, the next step is to agree, as shown in Figure 16.

3.5. Agreement on the Contract Form

As shown in Figure 16, once the buyer clicks on the "Proceed to agreement" link after approving the seller's proposal, the system displays the agreement form or smart contract form, which automatically extracts the job requirements and descriptions and only requires verification from the buyer via a biometric signature to confirm the agreement. When the buyer places their finger on the biometric signature reader device, the API reads the fingerprint scan and puts it in the buyer area of the contact form. Figure 17a shows the concept graphically, described above.

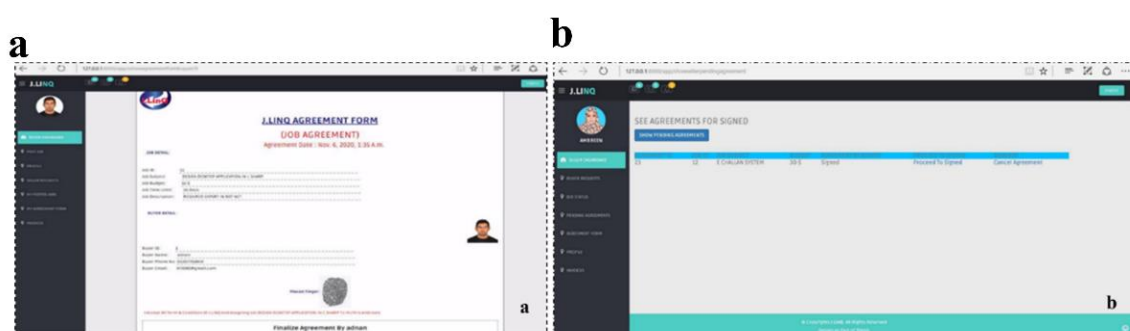


Figure 17. (a) Finalize agreement with a biometric signature from the buyer side. (b) Options for the seller to sign the contract of canceling the agreement.

The seller will notify the system accordingly when the buyer finalizes the signed biometric contract. At this stage, the seller has two options: (1) to sign the contract form to confirm the agreement or (2) to discard the deal if they do not want the work. Both options are available on the seller's dashboard, as shown in Figure 17b.

As shown in Figure 17b, the seller has two options: either proceed towards signing the contract form or cancel the transaction. In our case, to test the system functionality, we moved towards signing the contract form. Once the seller clicks on the "Proceed to Sign" link shown in Figures 6 and 10, the system displays the contract form to the seller, which

was signed by the buyer earlier with a biometric fingerprint and which holds the job details in it. To confirm the agreement, the seller places the biometric signature on the contract form, and the agreement between buyer and seller will be achieved, as shown in Figure 18.

Figure 18. Finalization of the agreement with the biometric signature from the seller's side.

Once the seller completes the biometric signature on the smart contract form, the job posting transaction will be validated and executed to change its state from unassigned to assigned. Further, the transaction will become part of the blockchain network after execution. Once the agreement between buyer and seller is achieved on smart contract form with a biometric signature, the hash code of smart contract form is generated and transferred to blocks of the blockchain. One copy of hashed data is stored in the database to recover the needed information. Figure 19 illustrates the storage of the hash code of all job details mentioned on the smart contract form in the data store.

0.266 seconds.)

menttb

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Filter rows: Search this table Sort by key: None

mentid	jobid	jobsubject	jobbudget	jobtimelimit	jobdescription
ny		dummy	DUMMY	dummy	dummy
	Spbkdf2-sha256S200 S2DvHeA9BCCCEeFSqdU4pRQsJq4TUs4J...	Spbkdf2-sha256S200 Saa1VirGW8p7zntOak5lyRgSilaHkZiZ...	Spbkdf2-sha256S200 SZ	Spbkdf2-sha256S200 SdO79H.Pc.5/zfg.hFIJwDgS5ddjCcCG	Spbkdf2-sha256S200 SynkPQSiF0DoHA
	Spbkdf2-sha256S200 S1BYCQGhtbS2FUKr1vte6lwsNmQsqz3L...	Spbkdf2-sha256S200 SZGxtjdH6H6M0RmiNUcoZww\$Zy.xft3h...	Spbkdf2-sha256S200 Sf	Spbkdf2-sha256S200 SrRXi3BvjXCiFSnCnFOCdK7A50b1ps8Lg	Spbkdf2-sha256S200 SQ8g5x/lfk9L6v7c
	Spbkdf2-sha256S200 SVUopRUGJodR6LwWAECKk1A5EdJLC70...	Spbkdf2-sha256S200 Sd875X6v1r83RmgN4ZwzpgSm./vzux...	Spbkdf2-sha256S200 Sw	Spbkdf2-sha256S200 SCqFUqnUuRei9VwoBAMDYOw\$swagweDp9	Spbkdf2-sha256S200 SkzKmVOodgzCG
	Spbkdf2-sha256S200 SK0XoHYMQluRcS6IV6j0nZA\$11ybL.EN...	Spbkdf2-sha256S200 S995by7mXkiKKkRJCCGEM4Q\$OenJ/Zo...	Spbkdf2-sha256S200 SY	Spbkdf2-sha256S200 STOn9Pwda6B0DgBDCOod8Tw\$Olv5aDel	Spbkdf2-sha256S200 S5dxbC.EcQ8hZC
	Spbkdf2-sha256S200 SlwRASKmVcg4BwFhrTXGGASlw4sAniB...	Spbkdf2-sha256S200 SxZgTAmAsBQCgVGotRYiRMgS66nkp.ML...	Spbkdf2-sha256S200 SQ	Spbkdf2-sha256S200 SyBmD8H7vHaMUGlBKkKwXMGQ\$RQUKrxui	Spbkdf2-sha256S200 SMWbs/b/XWuv9/
	Spbkdf2-sha256S200 SKdK6d6517p1T6I37PyckJAST3hJicw...	Spbkdf2-sha256S200 SY4zxHqP0HmNs7V3L2RujIAS/12YbwQm...	Spbkdf2-sha256S200 S9	Spbkdf2-sha256S200 Seq8VImSMcW6I9T4nBABgbASK81y9yKj	Spbkdf2-sha256S200 SL8U4R0iJEUlopt

Edit Copy Delete Export

Figure 19. Storage of job details mentioned on the smart contract form in a hashed format.

All parties involved in this transaction and a part of the blockchain network can retrieve the data from the store using a hash code of information. However, no one can

alter the information already stored without being notified because a minor change in information can change the entire stream of the hash code, and the defaulter can easily be identified. Thus, our system ensures information security and the validity of the agreement or contract.

4. Conclusions

We implemented a decentralized freelancing platform based on blockchain and biometric technology in this research work. The system allows users to create accounts, post jobs, place bids, update profiles, and come to an agreement between sellers and buyers via a smart contract form. In this study, we implemented and demonstrated the complete workflow of the system, from posting a job by buyers to finalizing an agreement with biometric signatures on the smart contract form. This research demonstrates how blockchain, smart contracts, and biometric technology create decentralized freelancing platforms. It is a better choice than traditional centralized freelancing platforms. Due to the decentralization and autonomous operations of the system, the platform fees are eliminated, ensuring a fair treatment with accounts as accounts are never deleted or frozen based on bogus or vague complaints, and no one can alter the requirements of the tasks at any stage of the work. Smart contracts handle the independent verification of the completed tasks. Hence, fraudulent activities are mitigated in the proposed system, which is present in centralized freelancing platforms.

The proposed system lessens the commission earned by a centralized freelancing system from freelancing professionals. The free, open-source blockchain platform Hyperledger Fabric by Linux foundation was used in this research work for several reasons, such as its unique three-phase transaction processing mechanism, permissioned, private blockchain, and the fact that it is supported by several programming languages, including Python and Java. The proposed system demonstrates the transparent means of agreeing between buyers and sellers and mitigates the chances of fraudulent activities, unfair and biased treatment of accounts, and high fees with enhanced security, reliability, validity; it also offers authenticity to clients and freelancers who do not face the burden of high fees or additional charges. In brief, the proposed system's main feature is providing equal importance for both clients and freelancers and equal opportunities to bid to both new and old freelancers in the system. The proposed system mitigates accounts' unfair treatment by avoiding scams and fraudulent activities. Moreover, the system allows the posting of jobs and bids to clients and freelancing professionals.

Author Contributions: Conceptualization, A.B. and Y.B.; methodology, A.B.; software, A.B. and Y.B.; validation, A.B. and Y.B.; formal analysis, A.B.; investigation, A.B.; resources, Y.B.; data curation, A.B.; writing—original draft preparation, A.B.; writing—review and editing, A.B. and Y.B.; visualization, A.B.; supervision, Y.B.; project administration, Y.B.; funding acquisition, Y.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financially supported by the Ministry of Small and Medium-sized Enterprises (SMEs) and Startups (MSS), Korea, under the “Regional Specialized Industry Development Program (R&D, S3091627)” supervised by Korea Institute for Advancement of Technology (KIAT). This research was also financially supported by the Ministry of SMEs and Startups (MSS), Korea, under the “Startup growth technology development program (R&D, S3125114)”.

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

Appendix A

The survey results of the freelancing community about fraudulent activities in centralized freelancing platforms.

Are You a freelancer?

15 responses

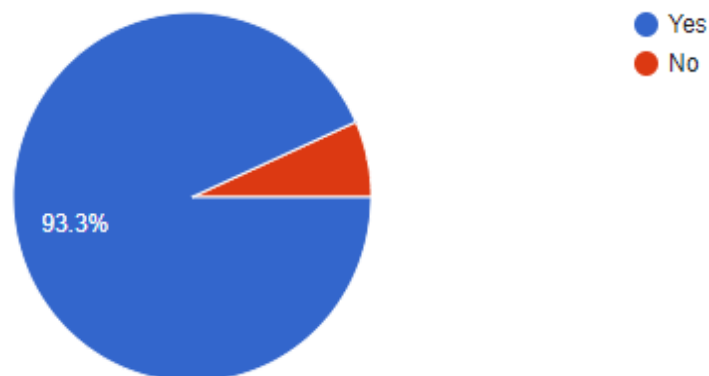


Figure A1. A survey report showing responders profession.

Please Select your age group

15 responses

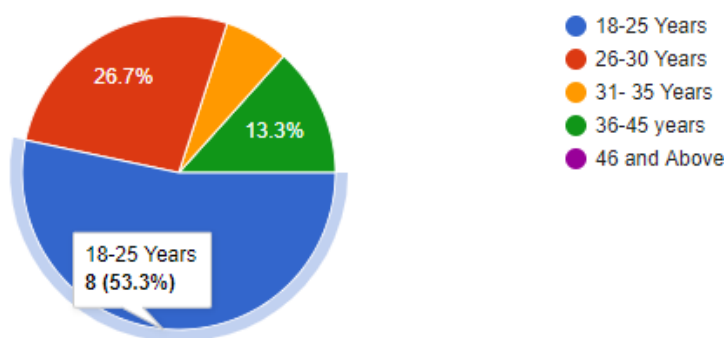


Figure A2. A survey report showing responders ages.

What are Yours Skills

15 responses

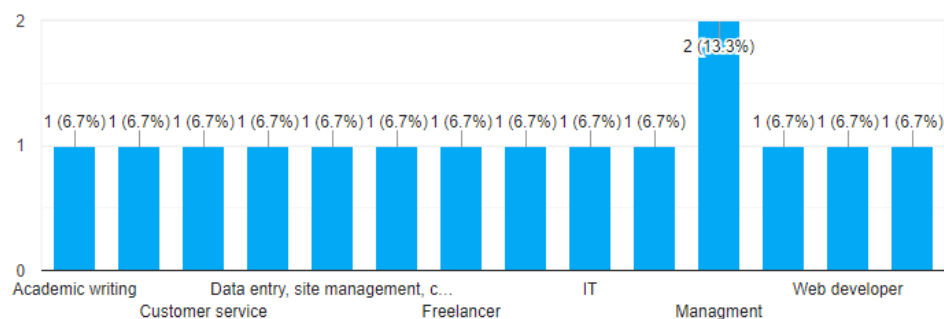


Figure A3. A survey report showing responders skills.

Have You ever face any type of fraudulent activity in your freelancing career
15 responses

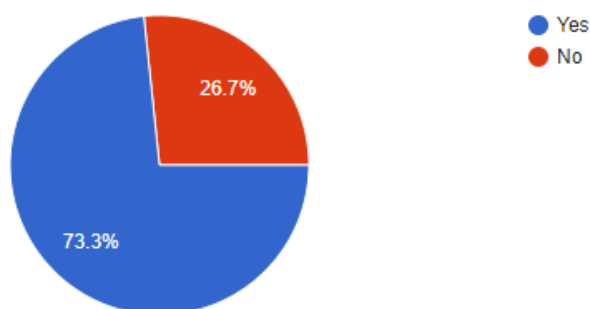


Figure A4. A survey report showing percentages of responders faced fraudulent activities.

Appendix B

<https://github.com/Amreen-source/Paper-Blockchain> (accessed on 27 February 2022).

References

1. Rehman, Z.; Usman, M.; Abrar, M.F.; Ullah, N. Freelancing as an Alternative Market for Software Professionals in Pakistan. *VFAST Trans. Softw. Eng.* **2021**, *9*, 78–92.
2. McKeown, T.; Pichault, F. Independent Professionals as Talent: Evidence from Individual Views of Working as a Contractor. *Hum. Resour. Manag.* **2021**, *60*, 313–328. [\[CrossRef\]](#)
3. Kaur, B.; Manohar, R.M.S.N.; Vamsi, R.R.; Teja, G.E.S. Online Freelancing Website. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2021**, *7*, 509–513. [\[CrossRef\]](#)
4. Chatterjee, A.; Varshney, L.R.; Vishwanath, S. Work Capacity of Regulated Freelance Platforms: Fundamental Limits and Decentralized Schemes. *IEEE/ACM Trans. Netw.* **2017**, *25*, 3641–3654. [\[CrossRef\]](#)
5. Gandhi, M.; Shah, P.; Solanki, D.; Shah, M. IRJET—Decentralized Freelancing System—Trust and Transparency. *Int. Res. J. Eng. Technol.* **2019**, *6*, 2092–2097.
6. Mondon-Navazo, M.; Murgia, A.; Borghi, P.; Mezihorak, P. In Search of Alternatives for Individualised Workers: A Comparative Study of Freelance Organisations. *Organization* **2021**, *28*, 13505084211041709. [\[CrossRef\]](#)
7. Popiel, P. “Boundaryless” in the Creative Economy: Assessing Freelancing on Upwork. *Crit. Stud. Media Commun.* **2017**, *34*, 220–233. [\[CrossRef\]](#)
8. Baitenizov, D.T.; Dubina, I.N.; Campbell, D.F.J.; Carayannis, E.G.; Azatbek, T.A. Freelance as a Creative Mode of Self-Employment in a New Economy (a Literature Review). *J. Knowl. Econ.* **2019**, *10*, 1–17. [\[CrossRef\]](#)
9. Haq, N.U.; Raja, A.A.; Nosheen, S.; Sajjad, M.F. Determinants of Client Satisfaction in Web Development Projects from Freelance Marketplaces. *Int. J. Manag. Proj. Bus.* **2018**, *11*, 583–607. [\[CrossRef\]](#)
10. Shah, R.R. *Blockchain Based Decentralized Freelancing Application*; California State University: Sacramento, CA, USA, 2020; pp. 1–49.
11. Sutherland, W.; Jarrahi, M.H.; Dunn, M.; Nelson, S.B. Work Precarity and Gig Literacies in Online Freelancing. *Work Employ. Soc.* **2020**, *34*, 457–475. [\[CrossRef\]](#)
12. Pallam, B.; Gore, M.M. Boomerang: Blockchain-Based Freelance Paradigm on Hyperledger. In Proceedings of the 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 6–8 July 2019. [\[CrossRef\]](#)
13. Cachon, G.P.; Dizdarer, T.; Tsoukalas, G. Decentralized or Centralized Control of Online Service Platforms: Who Should Set Prices? *SSRN Electron. J.* **2021**, *1*, 1–55. [\[CrossRef\]](#)
14. Wright, C.S. *Bitcoin: A Peer-to-Peer Electronic Cash System*; Whitepaper: Sydney, Australia, 2008; pp. 1–9.
15. Dannen, C. *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners*; University of California: Berkeley, CA, USA, 2017; pp. 1–185. [\[CrossRef\]](#)
16. Ryznar, M. *How to Regulate Bitcoin Futures*; Columbia Law School. Blue Sky Blog; Jerome Greene Hall, NY, USA, 2019; pp. 1–31.
17. Nofer, M.; Gomber, P.; Hinz, O.; Schiereck, D. Blockchain. *Bus. Inf. Syst. Eng.* **2017**, *59*, 183–187. [\[CrossRef\]](#)
18. Wirtschaft, Z.B.W.L. *Digitales Archiv. Mater. Test.* **1999**, *41*, 307. [\[CrossRef\]](#)
19. Monrat, A.A.; Schelén, O.; Andersson, K. A Survey of Blockchain from the Perspectives of Applications, Challenges, and Opportunities. *IEEE Access* **2019**, *7*, 117134–117151. [\[CrossRef\]](#)
20. Sankar, L.S.; Vidyapeetham, A.V.; Sindhu, M.; Vidyapeetham, A.V.; Sethumadhavan, M.; Vidyapeetham, A.V.; Protocol, C.; Slice, Q. A Global Road Map for Ceramic Materials and Technologies: Forecasting the Future of Ceramics. In Proceedings of the International Ceramic Federation—2nd International Congress on Ceramics, ICC 2008, Final Programme, Verona, Italy, 29 June–4 July 2008.

21. Yaqoob, I.; Salah, K.; Jayaraman, R.; Al-Hammadi, Y. Blockchain for Healthcare Data Management: Opportunities, Challenges, and Future Recommendations. *Neural Comput. Appl.* **2021**, *1*, 123456789. [\[CrossRef\]](#)
22. Liu, M.; Wu, K.; Xu, J.J. How Will Blockchain Technology Impact Auditing and Accounting: Permissionless versus Permissioned Blockchain. *Curr. Issues Audit.* **2019**, *13*, A19–A29. [\[CrossRef\]](#)
23. Guustaaf, E.; Rahardja, U.; Aini, Q.; Maharani, H.W. Blockchain-Based Education Project. *Aptisi Trans. Manag.* **2021**, *5*, 46–61. [\[CrossRef\]](#)
24. Aristidou, C.; Marcou, E. Blockchain Standards and Government Applications. *J. ICT Stand.* **2019**, *7*, 287–312. [\[CrossRef\]](#)
25. Marchesi, L.; Marchesi, M.; Destefanis, G.; Barabino, G.; Tigano, D. Design Patterns for Gas Optimization in Ethereum. In Proceedings of the 2020 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE), London, ON, Canada, 18 February 2020. [\[CrossRef\]](#)
26. Gupta, A.; Walia, N.K. Cryptography Algorithms: A Review. *Int. J. Eng. Dev. Res.* **2014**, *2*, 1667–1672.
27. Chen, J.; Xia, X.; Lo, D.; Grundy, J.; Luo, X.; Chen, T. Defining Smart Contract Defects on Ethereum. *IEEE Trans. Softw. Eng.* **2020**, *48*, 327–345. [\[CrossRef\]](#)
28. Michailidis, M.P. Blockchain Technology: The Emerging Human Resources Challenge. *Preprints* **2021**, *1*, 1–16. [\[CrossRef\]](#)
29. Pinna, A.; Ibba, S. *A Blockchain-Based Decentralized System for Proper Handling of Temporary Employment Contracts*; Springer International Publishing: New York, NY, USA, 2019; Volume 857. [\[CrossRef\]](#)
30. Levy, A. A Decentralized Fundraising and Freelancing Network. Whitepaper Revision 1.00-Draft for Community Review 2017; pp. 1–31. Available online: <https://raw.githubusercontent.com/IsraTo/fundraiser/master/whitepaper/IsraTo.whitepaper.en.pdf> (accessed on 27 February 2022).
31. Lim, J.; Na, K.; Kim, S.; Lim, J.; Na, K.; Kim, S. A Study on Information Technology Freelancer Matching with Exploiting Blockchain in Gig Economy. *WSEAS Trans. Bus. Econ.* **2021**, *18*, 455–461. [\[CrossRef\]](#)
32. António, R.; Quelhas, M.; António, R.; Quelhas, M.; António, R.; Quelhas, M. Secure and Connect Smart-Contracts to the Outside World. 2021, p. 1. Available online: <https://estudogeral.uc.pt/handle/10316/98240> (accessed on 27 February 2022).
33. Delgado-Mohatar, O.; Fierrez, J.; Tolosana, R.; Vera-Rodriguez, R. Blockchain Meets Biometrics: Concepts, Application to Template Protection, and Trends. *arXiv* **2020**, arXiv:2003.09262.
34. Valenta, M.; Sandner, P. *Comparison of Ethereum, Hyperledger Fabric and Corda*; The Frankfurt School Blockchain Center: Frankfurt, Germany, 2017; Volume 2, pp. 1–8.
35. Brandenburger, M.; Cachin, C.; Kapitza, R.; Sorniotti, A. Blockchain and Trusted Computing: Problems, Pitfalls, and a Solution for Hyperledger Fabric. *arXiv* **2018**, arXiv:1805.08541.
36. Guerreiro, S.; Silva, D.; Rosado, T.; Vasconcelos, A.; Correia, M.; Sousa, P. Decentralized Business Process Control Using Blockchain An Experience Report from Two Applications: Food Supply Chain and Car Registration. *Enterp. Model. Inf. Syst. Archit.* **2020**, *15*, 1–41. [\[CrossRef\]](#)
37. Beck, K.; Grenning, J.; Martin, R.C.; Beedle, M.; Highsmith, J.; Mellor, S.; van Bennekum, A.; Hunt, A.; Schwaber, K.; Cockburn, A.; et al. *Manifesto for Agile Software Development*; Agile Alliance: Corryton, TN, USA, 2001; pp. 12–14.
38. de Rossi, L.M.; Salviotti, G.; Abbatemarco, N. Towards a Comprehensive Blockchain Architecture Continuum. In Proceedings of the 52nd Hawaii International Conference on System Sciences, Grand Wailea, HI, USA, 8–11 January 2019; pp. 4605–4614. [\[CrossRef\]](#)
39. Muzammal, M.; Qu, Q.; Nasrulin, B. Renovating Blockchain with Distributed Databases: An Open Source System. *Futur. Gener. Comput. Syst.* **2019**, *90*, 105–117. [\[CrossRef\]](#)