CINEFORUM
ISSN: 0009-7039
Vol. 66. No. 2, 2026

Empirical Study of Customer Satisfaction Towards Cloud Kitchen Food Delivery System Based on Block chain and Smart Contract Technologies

Deepak Sharma
Research Scholar
Maharishi Markandeshwar (Deemed to be University), Mullana-Ambala
Email: deepaksharmaphd1988@gmail.com
Himali Lakhera
Assistant Professor
Maharishi Markandeshwar (Deemed to be University), Mullana-Ambala
Email: himali.lakhera@mmumullana.org
Ishan Bakshi
Associate Professor
Maharishi Markandeshwar (Deemed to be University), Mullana-Ambala
Email: ibakshi1988@gmail.com

Corresponding Author: Ishan Bakshi (ibakshi1988@gmail.com)

ABSTRACT

Such platforms for food delivery have been gaining prominence with advanced internet access and the rise of smart devices. But this fast expansion has also raised some challenges around (i) protecting sensitive data and user privacy, (ii) the negative impact of shoddy B2B discounting, and (iii) the need for strict operating practices. User financial and personal information protection stays as priority. Moreover, restaurants are under tremendous pressure to make money so they are discounting heavily, which drives so many orders that restaurants cannot keep up. As a result, these practices often sacrifice timely deliveries and diminish the quality of food. This study proposes a new food delivery framework based on Block chain and Smart Contract to face these issues. Their goal is to get rid of the commission models that intermediaries implement and reduce service delays caused by too many orders. In the Ethereum test network, the proposed protocols have been carefully implemented and tested Confirmations. The results confirm the effectiveness of the proposed system, indicating that as far as payment processes are concerned, they are conducted autonomously by the system. In addition, all participants (i.e., restaurants, delivery agents, and consumers) are required to comply with the relevant policies, and violators will be penalized accordingly.

Keywords: Block chain, food delivery system, smart contract, business models, actor restaurant

1. Introduction

The food industry is experiencing constant disruption, driven by innovative technologies and business models. One key shift in this space is the increasing demand for online food delivery services. Because of the fast pace of modern days, people usually do not have enough time to cook at home or go to fancy restaurants, except for maybe special occasions [1], As a result, demanders have adapted to digital tools to simplify their processes, try to save time and money. Nevertheless, it should be noted that the lack of strong trust mechanisms in the third-party delivery applications has led to a series of problems for the customers. There are many instances where orders arrive lukewarm owing to delays, or in some extreme cases, fail to materialize at all after placement [2-3]. These inefficiencies often arise from the violation of temporal obligations. Moreover, while we know that in developing countries many of the operational policies are not enforced, the enforcement remains rather glaringly lacking. However, customer contentment has become the crux of the expanding online food delivery industry. These organizations coordinating such services are intermediaries, connecting patrons to food establishments via their apps and courier staff. They control prices, and extract money from a transaction every time there is interaction in the ecosystem [4]

These platforms offer significant discounts on food to gain new customers, in turn, incurring massive losses for small scale restaurants [5]. It attracts more customers with discounts and attractive advertisements. To win this race and capture the market, the restaurants accept as many orders as they can, and they frequently do not deliver on time. Furthermore, rides based on earnings per delivery tend to do the same as app development to earn more cash. Customers are victims in both instances [6].

In the realm of online food delivery services, particularly within emerging markets, a critical challenge lies in understanding the intertwined dynamics affecting customer allegiance in the virtual domain [7] . Despite its significance, this interplay is often overshadowed, leaving pivotal aspects underexplored. This investigation endeavors to mitigate the prevalence of price wars, address challenges stemming from commission-driven models, and amplify both customer gratification and sustained loyalty within the online food delivery ecosystem [8]. The core intent of this scholarly inquiry is to delineate:

a) The online food quality and to ensure the hygiene.
b) The role of a mediator (the indirect influence of customer happiness and perceived value).

The operational strategies employed by online food delivery services often lack a customer-centric perspective, leading to dissatisfaction and a potential erosion of user loyalty [9]. To secure steadfast patronage from their clientele, it becomes imperative to address these shortcomings comprehensively. Since its inception, block chain technology [10] has demonstrated the capability to mitigate several of these persistent challenges. When coupled with the functionality of smart contracts [11], the prospects for enhanced efficiency and reliability emerge distinctly. However, achieving seamless harmonization within a decentralized ecosystem encompassing eateries, end-users, and delivery agents remains an elusive milestone [12], within this paradigm, the structuring of culinary services and their corresponding pricing would be governed by a suitable regulatory framework embedded in a decentralized application. Compliance with these standards would be rigorously enforced through the application of smart contracts, which would subsequently be recorded immutably onto the block chain ledger [13]. This approach ensures that the stipulated commercial terms and conditions are both tamper-proof and irreversible, thereby solidifying the operational agreements. Below is a synthesis of the primary contributions delineated in this scholarly discourse [14].

a) A penalty-based ordering in a decentralized food delivery ecosystem. Monetary penalization forces compliance to the regulation amongst the actors.

b) We use the penalization in our proposal of a reputation mechanism to achieve more trust between the actors.

c) The decentralized implementation based on smart-contract and block-chain ensure data security across the system.

d) A service that improves customer happiness by replacing commission-based services in the online meal delivery sector.

The structure of this document is articulated as follows. A meticulous exploration of the pertinent literature is delineated in the subsequent segment (Section 2). Section 3 expounds comprehensively on the proposed framework, delving into the intricacies of its components, the pivotal entities involved, and their respective functionalities within the architecture. The fourth section elucidates the intricate implementation nuances of the system and delineates the methodology employed for its evaluation. In Section 5, a simulated examination of the system is chronicled, encompassing the entire operational spectrum from the initiation of an order to its fulfilment, alongside a discourse on the merits and impediments encountered. Concluding observations and reflective insights are encapsulated in the terminal section.

2. *Background of the Study*

This section will provide a brief overview of technical perspective in order to explore the potential for resolving the present issues with online meal delivery services.

2.1 Block chain

Initially garnering prominence through its deployment in cryptocurrencies like Bitcoin, this innovation enabled direct peer-to-peer exchanges without reliance on intermediaries [15]. Block chain is characterized as an append-only data structure, where each block is cryptographically interlinked in a chronological sequence. Each block harbours a cryptographic hash of its predecessor, a timestamp marking its creation, a nonce, and transactional data. A pointer utilizing a Merkel tree structure is embedded within each block [16], through its inherent properties, Block chain ensures that all transactions within its network are discernible, authenticated, recorded, and securely processed, allowing vast volumes of data to traverse and update seamlessly across the ecosystem [17]. An illustrative depiction of a Block chain network is portrayed in Figure 1.

While conventional cloud storage systems exhibit a file loss probability reaching 100%, block chain-based systems maintain a near-zero file loss rate [18]. By leveraging smart contracts, block chain facilitates the creation of a robust, extensible stream processing engine optimized for distributed environments (whether cloud or edge), ensuring standardized communication in a block chain-centric ecosystem. Block chain’s innate resilience to modifications has

positioned it as the cornerstone for multifarious applications, including its deployment in the food industry to ensure food safety [19]. A secure delivery framework, founded on block chain principles, was proposed in [20], which safeguards not only data but also ensures user confidentiality and data integrity. This methodology was assessed for its efficiency in data insertion and read/write operations. Block chains, akin to contemporary cloud storage systems, have emerged as a competitive alternative in data management [21]
block chain is a formidable contender for secure e-transactions, surpassing most traditional mechanisms. Its immutable, consensus-driven ledger mitigates the influence of intermediaries in supply chains while demonstrating long-term cost benefits for such systems [22]

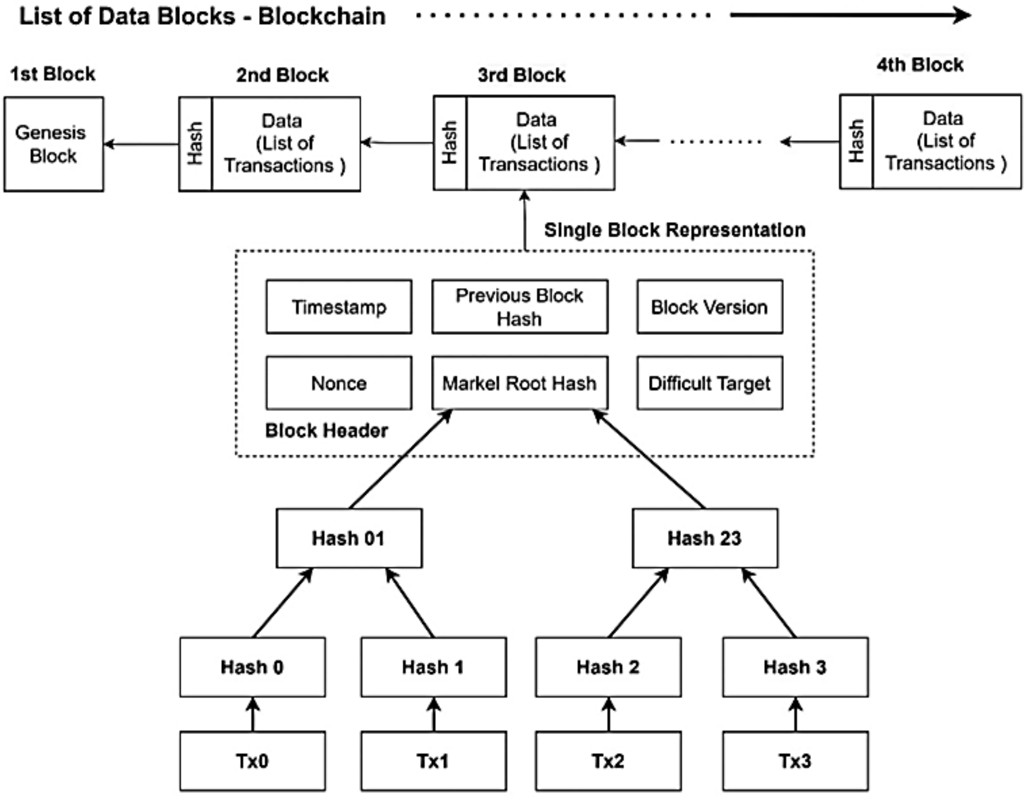

Figure 1 Block chain Structure

2.2 Smart Contract Using Block chain Platform

This technology is replicated in each node of the network and works in a self-sufficient and independent manner according to transaction data that triggers it [23]. For this reason, a smart contract can be displayed in a step-by-step formulation as in figure 2. Smart contracts aim to attained and secure data across the network. There are different use cases for smart contracts; for instance, a shipping cost related to an item can be described in terms of smart contracts that change depending on the arrival time restrictions. As per an agreement between both parties, which is recorded on the ledger and in the smart contract money changes hands automatically once an item is received [24].

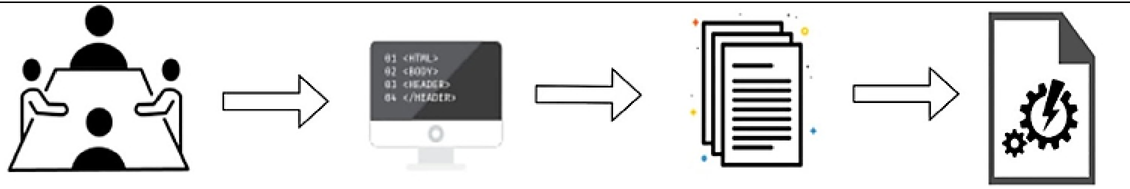

Figure 2 steps of smart contract

Hawk compiler transforms software into a cryptographic protocol, facilitating communication between the block chain infrastructure and its users BMC Protector serves as an innovative solution for managing music copyrights, utilizing block chain technology and smart contracts as its foundation. Within this system, smart contracts orchestrate every pivotal operation, ranging from musical composition to royalty allocation. This demonstrates the robust security provided by deploying smart contracts within a block chain ecosystem as the consensus mechanism renders alterations or modifications to a deployed smart contract virtually unattainable. Through B-Ride, drivers can autonomously engage in ride-sharing services, bypassing reliance on intermediary applications. Both drivers and passengers can execute transparent transactions while safeguarding their personal data. However, this level of anonymity within block chain systems occasionally enables fraudulent individuals to exploit the platform covertly, filing multiple requests to secure better deals [25]. Consensus remains indispensable for generating a new block within a block chain. Smart contracts seamlessly embed business logic into real-world applications, offering a dependable framework for all participating entities. Once a node crafts a block, its creation details are disseminated across the network. Subsequently, the smart contract scrutinizes the data in the newly created block against pre-established agreements. If deemed valid and aligned with specified criteria, the block is appended to the block chain. This ensures that the block chain of each network participant is promptly synchronized and updated.

2.3 Present centralized system and its shortcomings

1) The brunt of heavy discounting is borne by restaurant owners and food businesses. Without definitive criteria for how to regulate these situations, the result is lower earnings for each land.
2) Delivery of food late: Customers become less satisfied with food delivered late [26]. Consequently, it must take into account external factors like traffic, order volume and weather conditions. At times, inadequate training, irregular staffing, a wrong schedule, or unproductive routing can lead to a mismanaged delivery schedule. Some customers orders food from various locations, that creates a routing and timing issue (Kohar & Jakhar, 2021). Moreover, it needs quite an effort on the portion of the delivery stage’s clients care representatives to smooth over clients in these sorts of cases.

3) Chaotic order: For restaurant owners, reconciling online demand with available resources can become a colossal headache. This leads to them receiving a higher volume of orders which they are often unable to fill. Delivery workers hurry to the restaurant or deliver more than one due to increased orders and waste [27-28]. Food quality declines and delivery are delayed as a result.

3. *Block chain integrated food delivery system*

Never before has there been a impact as potentially revolutionary as that of the block chain-based food delivery sector. Well, the days of people being happy simply because they got a complimentary pizza dropped off in under half an hour are gone. Today, customers demand convenience and speed. The digitization has resulted in a rise in food delivery services over the last five years. Because it combines block chain technology and machine learning, "Wooberly Eats" is one of the newest products on the market. Wooberly uses "Flutter," an open-source UI framework, to facilitate data interchange between the overall system.

A platform for platform-to-consumer (P2C) delivery services is the second; examples are Food Panda, Hungrynaki, and Uber Eats. Food delivery services are systems that deliver meals and take out from restaurants to end-users. Food delivery online systems have played a significant role in making our modern life better. To state their appetite, patrons can select from a variety of eateries and cuisines [29]. Other difficulties were discovered as different business models for app-based meal delivery were created [30].

There is no time maintenance in the food delivery apps, hence the orders are chaotic, and the customers are not happy; as a result, the restaurant’s rating goes down. On the other hand, opportunity with this time manipulation, deliverymen also take as many foods as they can in hand with multiple deliveries at once and this results in late delivering foods. So once again unsatisfied customers and a decline in restaurant prestige. This research developing a solution to this time barrier and the disorderliness of the acceptance of the order by the restaurants using Block chain [31]

4. *Proposed Model*

4.1 Overview of the system

Centralized food distribution is the main source of the issue, and decentralized food distribution is the main remedy, we would like to introduce a block chain-based food delivery system. The current system's problems have been fixed, and our goal is to make things better for everyone—from the consumer to the restaurant to the delivery person. The system's general architecture is shown in Figure 3. Management of Online. Any entrepreneur can launch their own meal delivery business with us thanks to our food ordering and delivery management platform. This solution makes use of block chain technology to address some of the practical issues that current online meal delivery platforms encounter. It allows for transacting in crypto currency, which decentralizes and secures money, as I outlined in the research discussion [32]

Decentralization: Since commissions are high and customers frequently end up paying more when they use an application, as a result, the organizations' ability to grow is limited.
The following is the block chain solution tailored to the situation.

1) Commission rates are moderated by smart contracts: A smart contract can foster confidence between restaurants and service providers at affiliate prices for food markdowns. It can shield eateries from having to pay exorbitant commissions and keep them from going hungry by accepting too many orders that can't be fulfilled in the allotted time [33]
2) Trust on delivery partner and service provider capabilities
3) Cryptocurrency payment: the system offers to use cryptocurrency for completing payment; customers can be rewarded with discounts and nice offers to pay for foods by cryptocurrency.

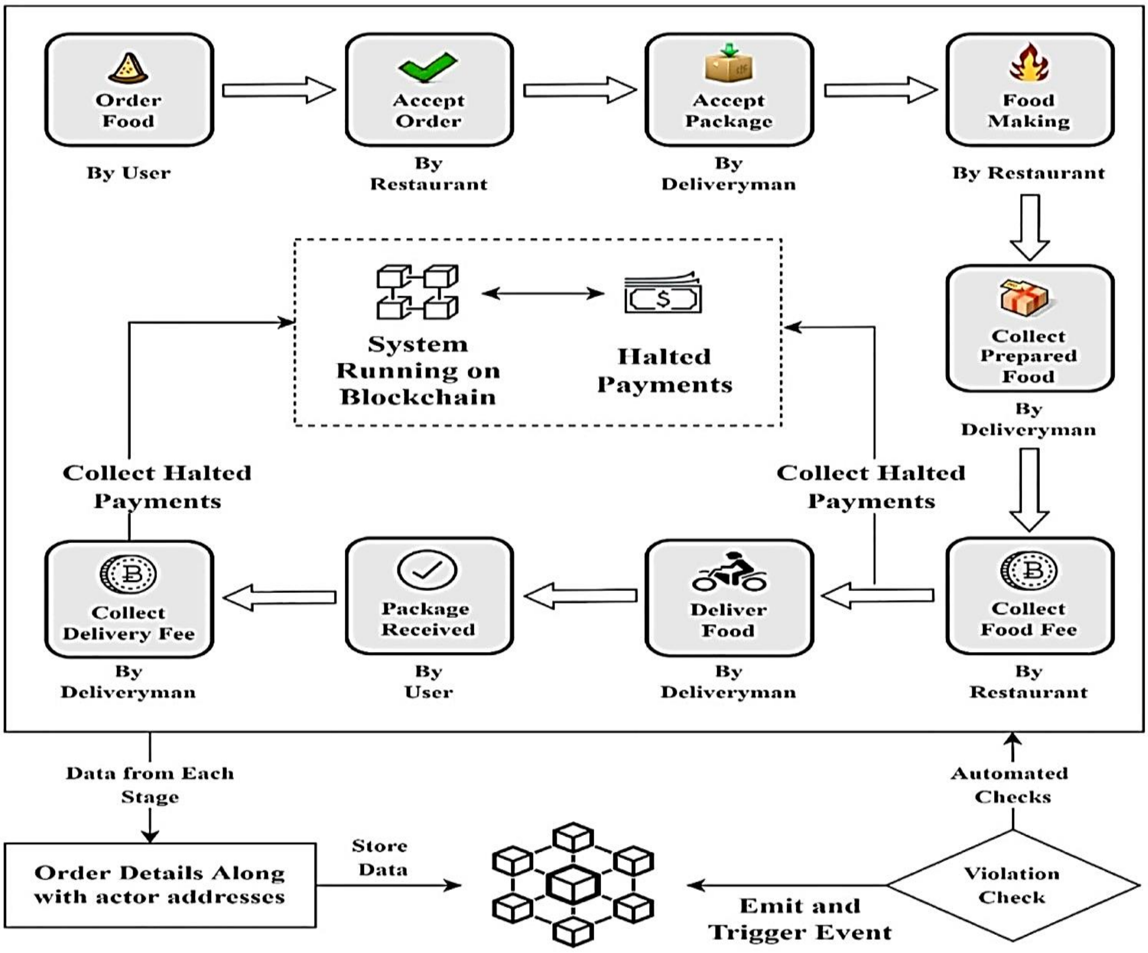

Figure 3. Proposed system overview

Security: Every transaction is protected by smart contracts. A smart contract is a piece of computer code that controls data transfer using a predetermined parameter.

The deliveryman, the consumers, and the restaurants are the system's primary actors, as seen in Figure 3. First, the competent authority will issue the contract in a block chain system. Food orders are on one hand and collecting food delivery fee on the other, containing some critical data on Block chain. Fig 3 shows these steps in chronological order. Smart contracts perform automated verifications on the various participants' information maintained within the trusted framework for security, traceability, and quality assurance [34]

4.2 System Design

Customers, delivery drivers, and food establishments are among the actors involved. In this instance, the contract deployer is also involved. You will be working on a number of components to specify how the players and the system interact. The roles of each actor and component are demonstrated in the sections that follow.

Performers: Despite the wide range of parties involved, they are all in risk from the same thing: an unworthy system that is unchangeable. Our proposed method encourages four actors together using technical means. The next section describes the characteristics of the performers [35]

1) Contract owner: The owner of the contract controls an excessive amount of the system.

2) Restaurants: This category includes coffee shops, lodging facilities, and the majority of food preparation jobs. Their main objective is to prepare meals quickly and serve it to clients.

3) Deliverymen pick up order packages from eateries and deliver them to patrons. When food needs to arrive on time, they use their modes of transportation.

4) Customer: Customer ordered food to the restaurant using technical platform and play an important role for improving Indian economy.

The goal, in turn, is to build a system where these parties collaborate to increase the visibility of such products.

CONSTRAINTS: These limitations limit the components that can be used to implement our system's characteristics, which include automation, security, availability, immutability, and the avoidance of third-party intermediaries. A. Smart Contracts and Block chain The two main components of the system we envision are block chain technology and smart contracts. Below is further information about these parts and how they interact with one another in our system.

BLOCK CHAIN: The legitimacy and stability of the system are improved by block chain technology. This initiative seeks to eliminate commission business in the food industry, provide tamper-proofing for historical data, and make restaurant food trackable data more transparent. As the delivery process progresses, a number of things occur at crucial moments, and the transaction record on the block chain records this information. This data is immutable, meaning it can never be edited or tampered without forcing the Block chain to cease to exist. We have implemented our block chain using Ethereum block chain. Regarding with the collaborative aspect of the block chain network, figure 4 shows the interaction of our actors and the system. By combining attributes, functions, events, and modifiers, smart contracts simultaneously automate our operations and eliminate the need for any third-party middleman (market-controlling software). Whereas events are the execution of a group of statements selected to be executable, functions are the execution of a specified task, moderators are the delegation of an actor's authority over the system, and attributes are storage variables. The system is made up of three separate smart contracts that were based on the Ethereum block chain and written in the Solidity programming language. We eventually combined them into a single smart contract. The first contract is with the customers. All of a customer's characteristics, features, and modifications are included in this contract. The contract's figure 5 provides an illustration of this.

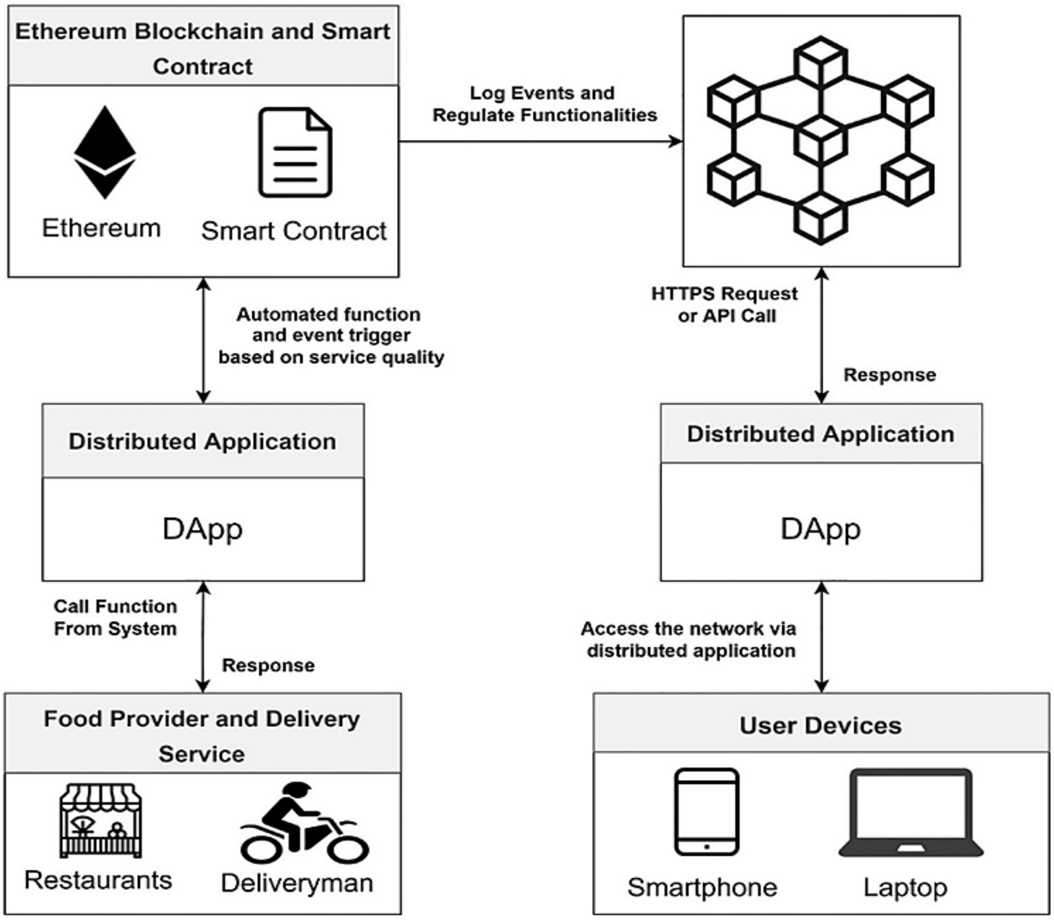

Figure 4. System-block chain interaction

5. Implementation of the Proposed Model

An Ethereum-class block chain environment was used to construct and test the block chain technology. We wrote smart contracts using Solidity. A basic prototype implementation of the proposed system was built up and tested using the Remix environment. We do not focus on the complete system development. But our work shows an architecture-based approach. In this section, we will describe the implementation and show the simulations and testing results.

Our study shows cross-platform compatibility of Block chain technology to some extent and that how much it is suited for food delivery systems where all stakeholders do their work properly and to consumer satisfaction. This study's primary goal is to show how block chain technology and smart contracts may be used to automate the entire process, from placing an order at a restaurant to having the deliveryman bring the food to the customer's home. Additionally, these systems track how long it takes to deliver meals, and Figures 3, 4, and 8 show the interactions between system actors and components. We divided the entire procedure into the four sections shown below: Customer & restaurant registration, Customer–restaurant, Restaurant–deliveryman, Deliveryman–customer. Descriptions of how each segment operate is mentioned henceforth.

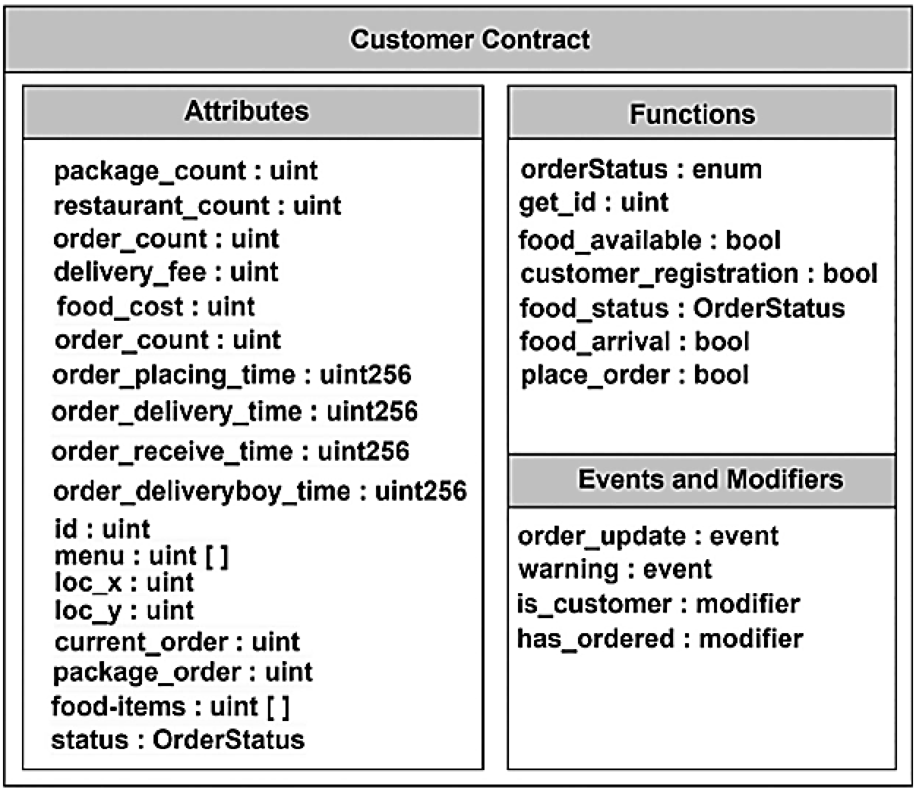

Figure 5 Customer smart contract system

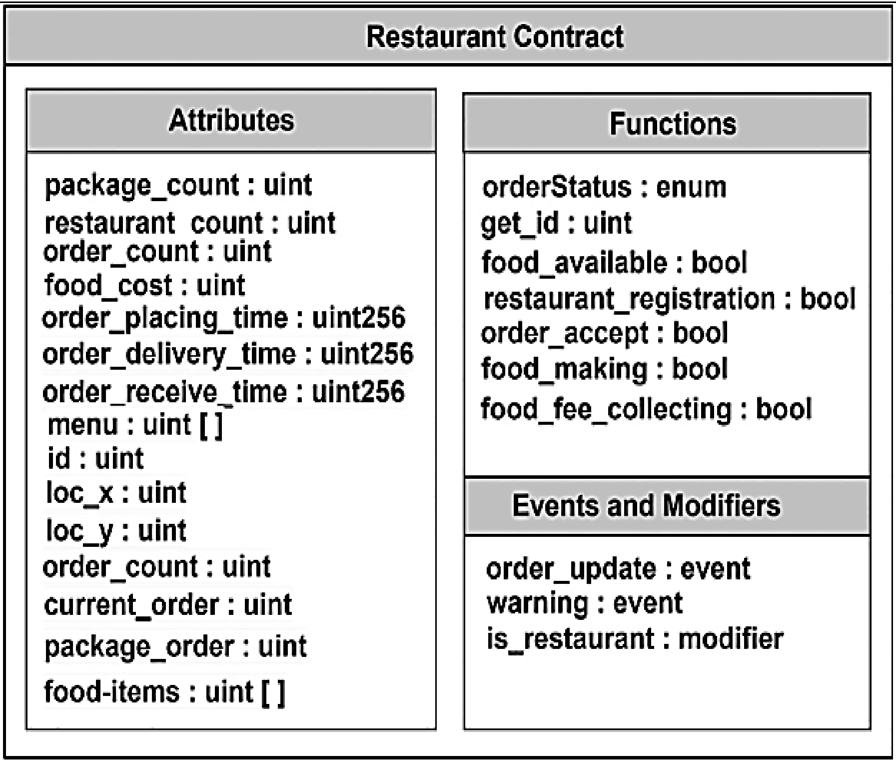

Figure 6 Smart Contract for Customer

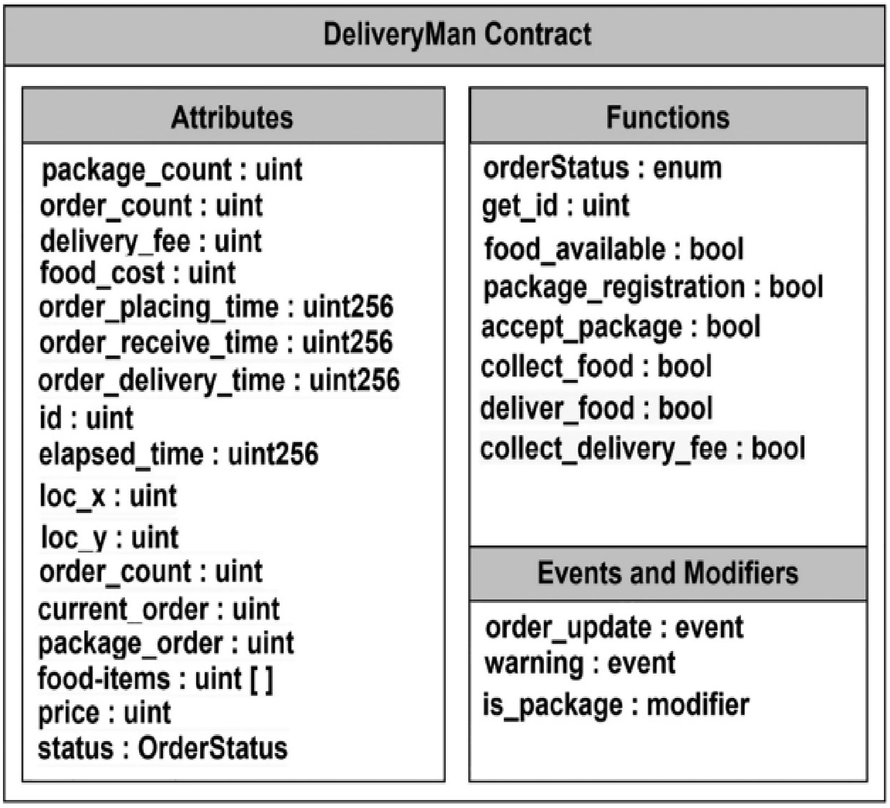

Figure 7 Deliveryman smart contract system

5.1 Registration

First, all three of our system's agents—the client, the restaurant, and the package deliveryman—had to log in. Every actor follows the same procedure. The system mainly saves clients, restaurant, and package addresses during the register step. Here, we make it clear that the restaurant registration algorithm is slightly different, even if all three actors employ the same registration procedure. The distinction relates to restaurant registration; restaurants must submit their menus, which are arrays containing the IDs of the foods, to the system.

Customer to Restaurant Mapping

When a food item is ordered, the smart contract will use the food_available() method to determine whether the item is in supply. The customer's order will then be entered into the system. The customer calls the place_order() method on the network to place an order for food, if it is available. a hash for every order that is in conformity. The accept_order() method is then used by the restaurant to accept the order.

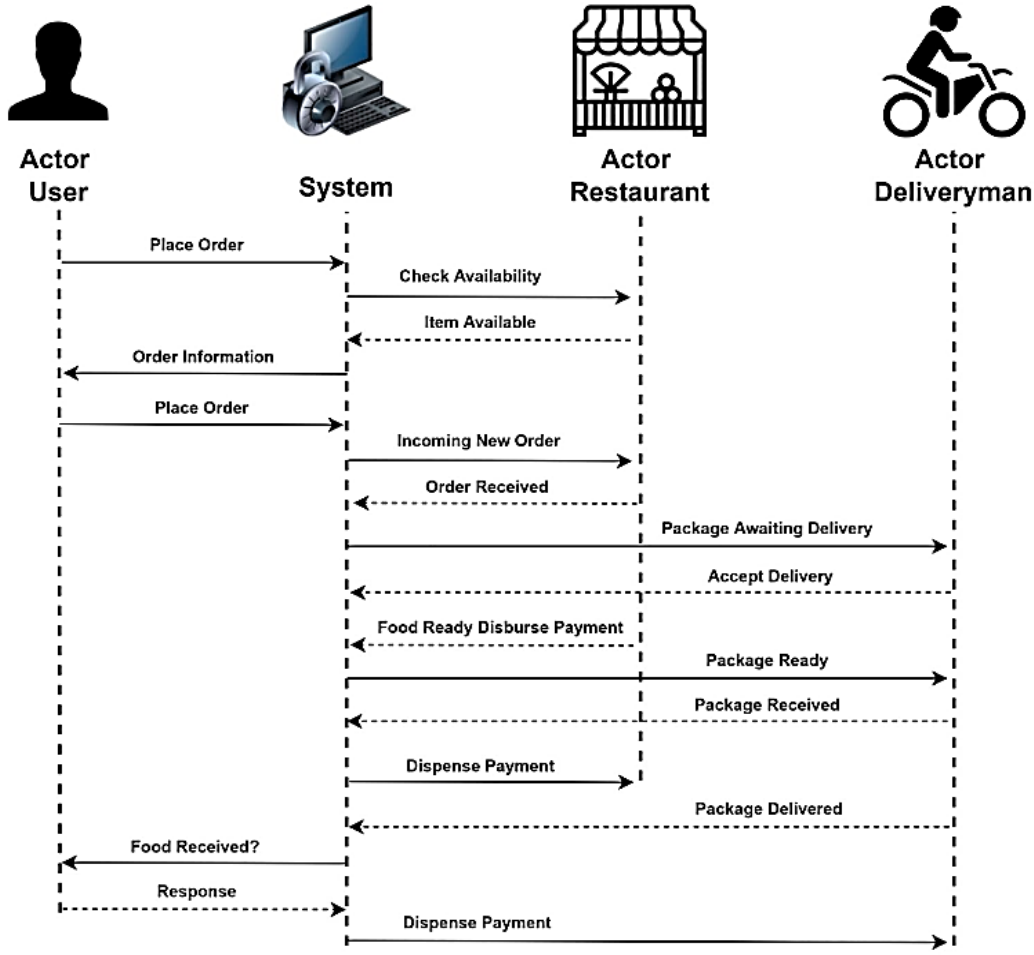

Figure 8 order flow diagram sequence

5.2 Restaurant to deliveryman Mapping

The subsequent phase bridges the gap between the eatery and the courier. In the preceding step, the customer places the request, and the eatery acknowledges it. At this juncture, the interaction between the customer and the eatery yields an encoded identifier (order hash). This hash is employed to notify the courier that restaurant ‘X’ has a parcel awaiting dispatch. Upon acknowledging this notification, the courier accepts the consignment, activating the systems accept_package() function, assigning the order hash to the courier’s identifier. Simultaneously, the eatery initiates the culinary process by invoking the food_making() function, embedding the order hash as its parameter. Following the completion of meal preparation, the system enables the courier to acquire the parcel by executing the collect_food () function in the background. Finally, the restaurant procures the overall payment for the prepared meal through the execution of the food_fee_collecting () function within the system. The operational framework ensures adherence to a predefined preparation window and a cap on simultaneous orders while connecting the eatery to the courier. In scenarios where breaches occur, such as delays beyond the stipulated timeframe or excessive order acceptance causing handover bottlenecks, the eatery’s preparation compensation undergoes a proportional deduction. This deduction, adjustable based on the restaurant’s quality benchmarks, aims to maintain accountability. The method for handling time violations is delineated in Algorithm 1.

Algorithm 1: Instruction volition detection:

Outputs:

1. Time infractions are issued by Intelligent Contract.
2. Restaurant payment
3. Order preparation ():
   Record all instruction for deliveryman, customer and restaurant ():
4. process Food payment ():
   if (order Delivery Time-order Placement Time) exceeds the promised delivery time.
   Trigger a warning event.
   Log the warning event on the block chain.
   Reduce the payment amount by x%.
   Else
   Process full payment to the restaurant.
   End if

End Algorithm

5.3 Deliveryman to Customer

At this juncture, the ultimate phase of the proposed framework encompasses the interaction between the delivery personnel and the end consumer. Once the culinary preparation is finalized, the restaurant relinquishes the parcel to the delivery agent, who is tasked with ensuring its arrival at the customer’s doorstep within a predefined temporal boundary. The delivery personnel's application activates the deliver_food() function embedded within the system, which meticulously records parcel metadata onto the Block chain. This data is secured with a hash identical to the one generated during the initiation of the customer's food request. Subsequently, the food_arrival() function is triggered when the customer acknowledges receipt of the package. Crucially, the delivery agent's remuneration is contingent upon the customer's confirmation.

Furthermore, akin to the third phase concerning the connection between restaurants and delivery personnel, this conclusive phase imposes strict constraints, such as a rigid delivery timeframe and a cap on the maximum permissible orders assigned to a single delivery agent within this interval. In instances where the prescribed delivery timeline is breached, or if the agent delays excessively in transferring the order to the customer, financial penalties are levied by reducing the agent's compensation. Such infractions starkly contrast with the expectation of receiving meals that retain their freshly-cooked essence. Consumers often anticipate the delivered cuisine to rival the quality of food served directly from a kitchen. However, the tendency of delivery agents to bundle multiple orders for simultaneous distribution compromises both the service standard and the quality of the delivered meals. This framework integrates the constraints and fee-collection mechanism into Algorithm 2, which is intricately detailed in the system. Culminating with phase four, the system achieves its overarching goal: facilitating the seamless transition of food from the restaurant to the consumer. This concept is further elucidated through a sequence diagram (Figure 8), offering a visual narrative that delineates the roles of all stakeholders within the food ordering and delivery ecosystem.

Algorithm 2: Step 4. Timing Infringement Identification and Deliveryman Reimbursement

Output:

1. A time infraction is flagged by an intelligent contract.
2. Deliveryman processing of payments with restrictions.
3. collect Food ():
   Record delivery Start Time using block.
4. food collection and arrival ():
   Record order Arrival Time.
5. process Delivery Payment ():
   If (order Arrival Time – delivery Start Time) exceeds the intimate delivery time:
   Trigger a warning event.
   Log the warning event on the block chain.
   Reduce the payment amount by x%.

Else:

   Process full payment to the deliveryman.

End if

End Algorithm

6. *Result Analysis*

Testing

The structural design and interactive framework bridging smart contracts and the Ethereum block chain were meticulously conceptualized, engineered, and operationalized by our team. Remix, a sophisticated web-based Integrated Development Environment (IDE), equips users with an Ethereum wallet create a system for easiness of customer, deliveryman and customer. Moreover, Remix offers a robust framework that facilitates the execution and deployment of contracts onto the Ethereum block chain, aligning seamlessly with our defined goals and operational prerequisites. This section elucidates the interplay of our smart contracts with the Block chain, illustrating their functional dynamics and how these elements harmonize. In our system, three principal actors engage in a synergistic dialogue, executing varied tasks that encompass requests, responses, and validations amongst themselves. This discourse underscores the entire lifecycle of a food order initiated by the customer, delineating how it materializes as a block within the Block chain network and progresses through subsequent stages until the customer receives the order. During the food preparation and delivery phases, specific infractions are detected, triggering penalties against the responsible parties—a phenomenon also systematically outlined. Once all the participants are appropriately registered within our ecosystem, the second operational phase commences, focusing on customer and restaurant interactions. At this juncture, the customer initiates a food order by invoking the place_order() function. Comprehensive information regarding the food order process is recorded within the system logs, as exemplified in figure 9. The Block chain serves as a repository, encapsulating details such as food items or their unique identifiers, restaurant identifiers, order identifiers, pricing data, and the cryptographic hashes corresponding to both the customer and the restaurant.

Figure 9 order placed inside the block chain

The ultimate phase of the outlined system encompasses the seamless transition of food from the eatery to the customer’s possession. Depending on the cuisine's complexity, the establishment may stipulate a specific preparation time frame. As depicted in Figure 10, critical data such as the customer’s and restaurant’s locations, order specifics, and estimated cooking duration are securely documented within the Block chain ledger. This process ensures that an order placed at the eatery simultaneously alerts the delivery personnel, where one of them accepts the task. Upon completing the preparation, the restaurant hands over the food to the assigned delivery agent for onward dispatch to the end consumer. The system logs are updated to include details such as the delivery person’s and customer’s hash identifiers, the unique hash of the food package, and the timestamp of dispatch. Figure 11 illustrates the formal acknowledgment of the handover between the eatery and the courier. The trajectory and timing of the food parcel are meticulously monitored until it reaches the recipient. Figure 12 showcases Block chain records, documenting the successful receipt of an order alongside the respective addresses of the customer, the package, and the courier. Previously, we delved into the concept of time infractions within this ecosystem, identified as a warning () event. This mechanism enforces punctuality by penalizing restaurants that fail to deliver within the agreed timeframe, deducting a portion of their earnings using the food_fee_collecting() method. Conversely, tardiness on the part of the delivery agent incurs a similar penalty, executed through the collect_delivery_fee () method. Violation logs for both instances are depicted in

Figure 13, with Figure 13(a) detailing restaurant-related breaches and Figure 13(b) outlining courier-related delays.

Figure 10. order place by the customer

```
hash            0x6c2623193ec28d0bc47fdd3bacf838475a21aeef94ad4d88ed08c38394930cff
input           0x36d...00001
decoded input   {
                        "uint256 order_id": "1"
                }
decoded output  {
                        "0": "bool: true"
                }
logs            [
                        {
                                "from": "0xEf9f1ACE83dfbB8f559Da621f4aEA72C6EB10eBf",
                                "topic": "0x0c2f387cf39fbb79382e223d10c940a8f2eaada67a10a1d377ff2e6f54b450d0",
                                "event": "order_update",
                                "args": {
                                        "0": "1",
                                        "1": 4,
                                        "order_id": "1",
                                        "status": 4
                                }
                        },
                        {
                                "from": "0xEf9f1ACE83dfbB8f559Da621f4aEA72C6EB10eBf",
                                "topic": "0xd61481a11e5449d64f97e1edf49aec06a9ff79cef718c30115a331a73ae49cb4",
                                "event": "event2",
                                "args": {
                                        "0": "Your package has been received by the deliveryman",
                                        "msg": "Your package has been received by the deliveryman"
                                }
                        }
                ]
```


Figure 11. order received by the deliveryman

Figure 12. order received by the customer

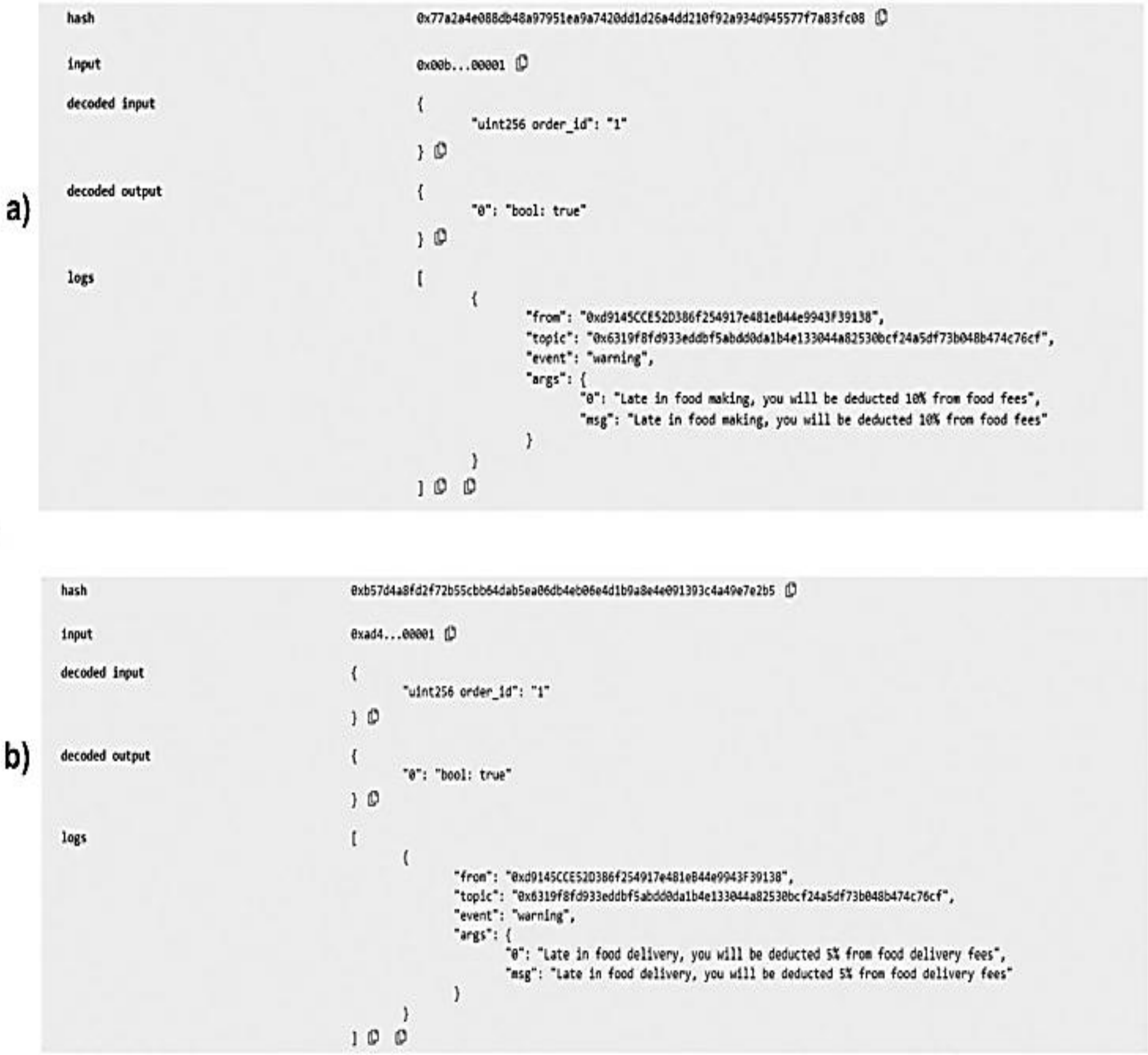

Figure 13. Instruction code for the deliveryman

7. Comparative Analysis

This section seeks to evaluate the suggested system and pinpoint the justifications for integrating block-chain-based food delivery. Table 1 compares the challenges and benefits of our method to earlier studies that used block chain technology and smart contracts. Below are the findings of the system's comparison with the current approaches listed in Table 1 above.

Table 1. Comparative Table

Result Parameters	**[23]**	**[21]**	**[17]**	**[2]**	**Proposed Model**

Monitorability	Not Available	Available	Not Available	Not Available	Available
Transparency	Not Available	Not Available	Available	Available	Available
Intelligent Contracts	Available	Available	Available	Available	Available
Violation of time	Not Available	Not Available	Not Available	Not Available	Available
Strictures	Not Available	Not Available	Not Available	Not Available	Available

● In our proposed system, we implemented time violation penalties to increase Block chain acceptance. Each of the aforementioned ways has a different time violation and penalty.

● Not all of these approaches provide the archetypal properties of unchangeable data storage and the elimination of third-party middlemen; however, they are part of our proposed system and, therefore, raise the quotation for receiving greater acceptance.

7.1 Challenges

While the application of immutable ledger technology and elevated levels of control within smart contracts mitigate deficiencies in the regulatory landscape and bolster online food delivery frameworks, significant systemic obstacles persist. As Block chain technology remains in an evolutionary stage, its practical, real-time deployment poses substantial challenges. Mining operations, though critical to the infrastructure, demand exorbitant computational resources, driving up both execution time and operational costs. The distributed nature of Block chain networks, while ensuring reliability, often imposes a delicate compromise between robustness and user accessibility. The unalterable nature of Block chain, though a cornerstone of its security architecture, presents a paradox; it renders the rectification of errors virtually unattainable. Unlike traditional backend database systems, Block chain's rigorous security mechanisms prohibit any alteration of confirmed transactional data. This inflexibility underscores its divergence from conventional systems. Smart contracts, intricately designed to govern the network, seamlessly orchestrate processes but add layers of complexity to an already intricate ecosystem. It is immutable once being placed on the Block chain, and features are not addable or modifiable within it. A small amount of money is needed to configure the block chain ecosystem initially. There are legal issues too - depending on what country you trade through; cryptocurrency may not be legal. Table 2 shows the gas costs for running and executing functions.

Table 2. Total Expenditure for execution and transactions

Activity/Performer	Expenses	Implementation Expense
Registration ()/ Restaurant Customer package	385,355 93,566	385,355 93,566

	140,108	140,108
Order from customer	222,974	222,974
Order accepted by Restaurant	50,971	50,971
Order package by deliveryman	74,581	74,581
Food making preparation	54,167	54,167
Food collection by deliveryman	56,016	56,016
Collecting charge by restaurant	56,123	56,123
Food delivered by deliveryman	27,656	27,656
Food collect by customer	76,345	76,345
Delivery fees collection	323,43	323,43

8. Conclusion

The focus of this research was about block chain's importance in the food delivery system, how block chain is used to make data easily available for the customers with privacy, integrity as well as to ensure the trust between a restaurant, its customer and a deliveryman. The main concluding points are given below:

1. This research is totally focus on the upgrading the process of food making, its delivery and also to maintain its quality using block chain and its related technical aspects.
2. Data privacy is also a major concern and it is resolved in this research by using Ethereum test network.
3. The integration of new technology improving the overall process but also make some policies and regulation which have to follow by restaurant, customer and deliveryman.
4. Customers' rights and benefits are ultimately jeopardized in such a commission-based industry, while other stakeholders profit greatly from systemic flaws.
5. Block chain technology serves as the stronghold for the whole system, making it completely secure.
6. The execution of our study has been made flexible and wide-reaching. Our long-term objectives also include allowing it to get a penalty, just like everyone else, installing the system on a permissioned block chain, developing new features, assessing the quality of the meal before it is prepared, and focusing on future work in the near future.

 # Acknowledgement
Both authors acknowledge the submission of manuscript
 # Funding
There is no funding associated with this manuscript
 # Conflict of Interest
There is no conflict of interest related to this manuscript
 # Author Contributions
The first author contributes to design methodology and result analysis and second author contributed worked as a structured format

 # Ethics approval

This manuscript ethically approved by the both authors

 # Data availability

Data available on request

References

Bader, L., Burger, J. C., Matzutt, R., & Wehrle, K. (2019). Smart contract-based car insurance policies. 2018 IEEE Globecom Workshops, GC Wkshps 2018 – Proceedings. https://doi.org/10.1109/ GLOCOMW.2018.8644136.

Bartolini, N., Coletta, A., Maselli, G., & Piva, M. (2020). DRUBER: A trustable decentralized drone-based delivery system. Proceedings of the 6th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications, DroNet. https://doi.org/10.1145/3396864.3399706.

Baza, M., Lasla, N., Mahmoud, M., & Abdallah, M. M. (2019). (1) B-Ride: Ride Sharing with privacy-pres- ervation, trust and fair payment atop public block chain | request PDF.https://www.researchgate.net/publication/333993190_BRide_Ride_Sharing_with_Privacy-preservation_Trust_and_Fair_ Payment_atop_Public_Block chain.

Bodkhe, U., Tanwar, S., Parekh, K., Khanpara, P., Tyagi, S., Kumar, N., & Alazab, M. (2020). Block chain for Industry 4.0: A comprehensive review. IEEE Access, 8, 79764–800. https://doi.org/10.1109/ ACCESS.2020.2988579

Chen, J.-f., Wang, L., Wang, S., Wang, X., & Ren, H. (2021). An effective matching algorithm with adap- tive tie-breaking strategy for online food delivery problem. Complex & Intelligent Systems, 8(1), 107–28. https://doi.org/10.1007/S40747-021-00340-X

DeCusatis, C., Zimmermann, M., & Sager, A. (2018). Identity-based network security for commercial block chain services. 2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC (pp. 474–477). https://doi.org/10.1109/CCWC.2018.8301713.

Farouk, A., Alahmadi, A., Ghose, S., & Mashatan, A. (2020). Block chain platform for industrial health- care: Vision and future opportunities. Computer Communications, 154(March), 223–35. https://doi. org/10.1016/J.COMCOM.2020.02.058

Granheim, S. I., Opheim, E., Terragni, L., Torheim, L. E., & Thurston, M. (2020). Mapping the digital food environment: A scoping review protocol. BMJ Open, 10(4), e036241. https://doi.org/10. 1136/BMJOPEN-2019-036241

Hasan, H. R., Salah, K., Jayaraman, R., Ahmad, R. W., Yaqoob, I., & Omar, M. (2020). Block chain-based solution for the traceability of spare parts in manufacturing. IEEE Access, 8, 100308–22. https://doi. org/10.1109/ACCESS.2020.2998159

Hassija, V., Saxena, V., & Chamola, V. (2021). A mobile data offloading framework based on a com- bination of block chain and virtual voting. Software – Practice and Experience, 51(12), 2428–45. https://doi.org/10.1002/SPE.2786

Kohar, A., & Jakhar, S. K. (2021). A capacitated multi pickup online food delivery problem with time windows: A branch-and-cut algorithm. Annals of Operations Research, 1–22. https://doi.org/10. 1007/S10479-021-04145-6

Kosba, A., Miller, A., Shi, E., Wen, Z., & Papamanthou, C. (2016). Hawk: The block chain model of cryp- tography and privacy-preserving smart contracts. Proceedings – 2016 IEEE Symposium on Security and Privacy, SP (pp. 839–858). https://doi.org/10.1109/SP.2016.55.

Lau, T.-C., & Ng, D. (2019). (1) (PDF) online food delivery services: Making Food delivery the new normal.

https://www.researchgate.net/publication/334050513_Online_Food_Delivery_Services_ Making_ Food_Delivery_the_New_Normal.

Liao, W., Zhang, L., & Wei, Z. (2020). Multi-objective green meal delivery routing problem based on a two-stage solution strategy. Journal of Cleaner Production, 258(June), 120627. https://doi.org/10. 1016/J.JCLEPRO.2020.120627

Liu, Y. (2019). An optimization-driven dynamic vehicle routing algorithm for on-demand meal deliv- ery using drones. Computers & Operations Research, 111(November), 1–20. https://doi.org/10. 1016/J.COR.2019.05.024

Nakamoto, S. (2018). Bitcoin: A peer-to-peer electronic cash system. www.bitcoin.org.

Ngamsuriyaroj, S., Likittheerameth, T., Kahutson, A., & Pathummasut, T. (2018). Package delivery system based on block chain infrastructure. Proceeding of 2018 7th ICT International Student Project Conference, ICT-ISPC. https://doi.org/10.1109/ICT-ISPC.2018.8523944.

Niharika, N. P. (2020). A study on consumer behaviour and the impact of food delivery apps on the college students in Bangalore.

https://www.ijresm.com/Vol.3_2020/Vol3_Iss3_March20/IJRESM_ V3_I3_120.pdf.

Novo, O. (2018). Block chain meets IoT: An architecture for scalable access management in IoT. IEEE Internet of Things Journal, 5(2), 1184–95. https://doi.org/10.1109/JIOT.2018.2812239

Pranto, T. H., Noman, A. A., Mahmud, A., & Haque, A. B. (2021). Block chain and smart contract for IoT enabled smart agriculture. PeerJ Computer Science, 7(March), 1–29. https://doi.org/10.7717/PEERJ- CS.407/SUPP-1

Rani, K. S., & Vishali, S. (2021). Block chain driven IoT based Delish2Go decentralized food delivery application. Proceedings – International Conference on Artificial Intelligence and Smart Systems, ICAIS (pp. 1727–1732). https://doi.org/10.1109/ICAIS50930.2021.9395873.

Roeck, D., Sternberg, H., & Hofmann, E. (2019). Distributed ledger technology in supply chains: A transaction cost perspective. Block chain in Transport and Logistics, 58(7), 2124–41. https://doi. org/10.1080/00207543.2019.1657247

Shyamala Devi, M., Aparna Shashikant Joshi, R. S., & Bagate, R. A. (2019). Design of IoT block chain based smart agriculture for enlightening safety and security. Communications in Computer and Information Science, 985(February), 7–19. https://doi.org/10.1007/978-981-13-8300-7_2

Stanciu, A. (2017). Block chain based distributed control system for edge computing. Proceedings – 2017 21st International Conference on Control Systems and Computer, CSCS (pp. 667–671). https:// doi.org/10.1109/CSCS.2017.102.

Suhartanto, D., Ali, M. H., Tan, K. H., Sjahroeddin, F., & Kusdibyo, L. (2018). Loyalty toward online food delivery service: The role of e-service quality and food quality. Journal of Foodservice Business Research, 22(1), 81–97.
https://doi.org/10.1080/15378020.2018.1546076

Szabo, N. (1997). Formalizing and Securing relationships on public networks. First Monday, 2(9), 9. https://doi.org/10.5210/FM.V2I9.548

Tse, D., Zhang, B., Yang, Y., Cheng, C., & Mu, H. (2017). Block chain application in food supply infor- mation security. In 2017 IEEE international conference on industrial engineering and engineering management (IEEM) (pp. 1357–1361). IEEE.

Valenta, M., & Sandner, P. G. (2017). [PDF] Comparison of ethereum, hyperledger fabric and corda | semantic scholar. https://www.semanticscholar.org/paper/Comparison-of-Ethereum%2C-Hyperledger-Fabric-and-Valenta
Sandner/9f4f80c8e596b70ec8e2324f44ede15c48c147b5.

Wesley, D. (2017). Is block chain the invisible answer to your business needs? https://www.forbes. com/sites/forbestechcouncil/2017/12/11/is-block chain-the-invisible-answer-to-your-business- needs/?sh=6385d1db13d0.

Wood Daniel Davis. (2014). [PDF] ethereum: A secure decentralised generalised transaction ledger | semantic scholar. https://www.semanticscholar.org/paper/ETHEREUM%3A-A-SECURE-DECENTRALISED-GENERALISED-LEDGER-Wood/
da082d8dcb56ade3c632428bfccb88ded0493214.

Worku, Z., & Legoabe, R. (2017). (1) (PDF) factors that undermine service delivery by civil engineers in South African local municipalities. https://www.researchgate.net/publication/340892769_
Factors_that_Undermine_Service_Delivery_by_Civil_Engineers_in_South_African_Loca
l_ Municipalities.

Yang, Z., Yang, K., Lei, L., Zheng, K., & Leung, V. C. M. (2019). Block chain-based decentralized trust management in vehicular networks. IEEE Internet of Things Journal, 6(2), 1495–1505. https://doi. org/10.1109/JIOT.2018.2836144

Yeo, V. C. S., Goh, S. K., & Rezaei, S. (2017). Consumer experiences, attitude and behavioral intention toward online food delivery (OFD) services. Journal of Retailing and Consumer Services, 35(March), 150–62.
https://doi.org/10.1016/J.JRETCONSER.2016.12.013

Yewale, A. J. (2018). Study of block chain-as-a-service systems with a case study of hyperledger fabric implementation on kubernetes. UNLV Theses, Dissertations, Professional Papers, and Capstones, August, 8–9. https://doi.org/10.34917/14139923.

Zhao, S., & O'Mahony, D. (2018). BMCprotector: A block chain and smart contract-based application for music copyright protection. ACM International Conference Proceeding Series, December, 1–5. https://doi.org/10.1145/3301403.3301404