

الجمهورية الشعبية الديمقراطية الجزائرية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

المدرسة العليا للإعلام الآلي - 08 ماي 1945 – بسيدي بلعباس
Ecole Supérieure en Informatique
-08 Mai 1945- Sidi Bel Abbès



Mémoire de Fin d'étude

Pour l'obtention du diplôme d'ingénieur d'état

Filière : **Informatique**

Spécialité : **Ingénierie des Systèmes Informatiques (ISI)**

Thème

A Blockchain-based Transparent Food Vending System.

Présenté par :

- Mr Ait Yahia Messaoud
- Mr Benafghoul Abdelaaziz

Soutenu le : **00/00/2022**

Devant le jury composé de :

- M/Mme/Mlle Alaa Eddine BELFEDHAL
- M/Mme/Mlle ANANI Djihed
- M/Mme/Mlle KHALDI Miloud

Encadreur
Président
Examineur

Année Universitaire : 2021 / 2022

Abstract

The agricultural sector and food markets are facing great challenges in terms of high prices and monopolies, and the only victim is the citizen. Many countries are competing to invent new solutions to advance the agricultural sector and, in particular, food supply chain management.

Through this work, we have proposed a new solution to improve food supply chain management. The goal of this platform is to bring blockchain technology, which ensures transparency of transactions, identifies the original source of food, and increases trust.

The platform is a web and mobile application that allows farmers, wholesalers, retailers and customers to exchange goods by buying and selling reliably and quickly through the blockchain network, electronic signature and QR code.

The major objective of the platform is to determine the true origin of food, as well as its distribution, in order to protect the health of citizens while lowering prices and eliminating fraud, confidentiality, and monopoly of goods.

Keywords: Food Supply Chain Management, Agriculture, Blockchain, Digital signature, Trust, Traceability, Transparency.

Résumé

Le secteur agricole et les marchés alimentaires sont confrontés à de grands défis en termes de prix élevés et de monopoles, et la seule victime est le citoyen. De nombreux pays sont en compétition pour inventer de nouvelles solutions afin de faire progresser le secteur agricole et, en particulier, la gestion de la chaîne d’approvisionnement alimentaire.

A travers ce travail, nous avons proposé une nouvelle solution pour améliorer la gestion de la chaîne d’approvisionnement alimentaire. L’objectif de cette plateforme est d’apporter la technologie blockchain, qui assure la transparence des transactions, identifie la source originale des aliments et augmente la confiance.

La plateforme est une application web et mobile qui permet aux agriculteurs, aux grossistes, aux détaillants et aux clients d’échanger des marchandises en achetant et en vendant de manière fiable et rapide grâce au réseau blockchain, à la signature électronique et au code QR.

L’objectif majeur de la plateforme est de déterminer la véritable origine des aliments, ainsi que leur distribution, afin de protéger la santé des citoyens tout en faisant baisser les prix et en éliminant la fraude, la confidentialité et le monopole des marchandises.

Mots-clés : Gestion de la chaîne d’approvisionnement alimentaire, agriculture, Blockchain, signature numérique, confiance, traçabilité, transparence.

ملخص

واجه القطاع الزراعي وأسواق المواد الغذائية تحديات كبيرة, من بين هذه التحديات ارتفاع الأسعار والاحتكارات ، و الضحية الوحيد هو المواطن .العديد من البلدان تتنافس على ابتكار حلول جديدة للنهوض بالقطاع الزراعي ،على وجه الخصوص ، إدارة سلسلة التوريد الغذائي.

ولهذا، اقترحنا منصة جديدة مبنية على تقنية بلوكشين لضمان شفافية المعاملات و زيادة الثقة بين مختلف الاطراف، وكذلك تتبع مصدر المنتج.

المنصة عبارة عن تطبيق ويب وهاتف محمول يسمح للمزارعين وتجار الجملة وتجار التجزئة والعملاء بتبادل البضائع عن طريق الشراء والبيع بشكل موثوق وسريع من خلال شبكة بلوكشين والتوقيع الإلكتروني ورمز الاستجابة السريعة.

الهدف الرئيسي للمنصة هو تحديد المصدر الحقيقي للغذاء من أجل حماية صحة المواطنين وكذلك خفض الأسعار والقضاء على الغش واحتكار السلع.

الكلمات الرئيسية: إدارة سلسلة التوريد الغذائية ، الزراعة ، سلسلة الكتل ، التوقيع الرقمي ، الثقة ، التتبع ، الشفافية.

Contents

Introduction	1
I Background	4
1 Blockchain Technology	5
1.1 Introduction	6
1.1.1 The History Of Blockchain	6
1.1.2 Benefits Of Blockchain	7
1.1.3 The Meaning Of Decentralization?	7
1.2 Blockchain Categories	8
1.2.1 Public Blockchains	8
1.2.2 Consortium Blockchains (Federated)	8
1.2.3 Private Blockchains	8
1.3 Blockchain Components	9
1.3.1 Distributed Ledgers	9
1.3.2 Public Key Cryptography	10
1.3.3 Cryptographic Hash Functions	13
1.3.4 Merkle Tree	13
1.3.5 Structure Of A Block	15
1.3.6 Defining Mining, Difficulty, Validation	16
1.3.7 The Chain Of Blocks	17
1.3.8 Nodes And Networks	17
1.4 Consensus Models	19
1.4.1 Proof Of Work	19
1.4.2 Proof Of Stake	19

1.4.3	Proof Of Importance	19
1.4.4	Proof Of Authority	20
1.5	Types Of Blockchain	20
1.5.1	Only Cryptocurrency : Bitcoin	20
1.5.2	Currency + Business logic : Ethereum	21
1.5.2.1	Smart Contract	21
1.5.2.2	EVM	22
1.5.2.3	Accounts In Ethereum	22
1.5.2.4	Accounts Vs Wallet	22
1.5.2.5	Transaction Types	23
1.5.2.6	Gas Fees	23
1.5.2.7	Decentralized Applications (DApps)	24
1.5.2.8	Web3 :	24
1.5.2.9	Popular Dapps :	25
1.5.2.10	Bitcoin Vs Ethereum	25
1.5.3	Only Business Logic : Hyperledger	26
1.6	Blockchains Frameworks	27
1.7	IPFS	29
1.7.1	How To Use IPFS	29
1.8	Blockchain Applications	30
1.9	Challenges And Solutions	31
2	Traditional Supply Chain Management	32
2.1	Introduction	33
2.2	The Importance Of Supply Chain Management	33
2.3	Supply Chain Models	34
2.4	Elements Of Supply Chain Management	36
2.5	Existing Companies	37
2.6	Food Supply Chain Management	38
2.7	Stages Of The Food Supply Chain Management	38
2.8	The Importance Of Food Supply Chain Management	39
2.9	Main Problems With Food Supply Chains Management	40
2.10	Conclusion	40
II	Our Solution	42
3	System Design	43
3.1	Introduction	44

3.2	Functionality Considerations	44
3.3	Global Architecture	46
3.3.1	Metamask	47
3.3.2	Infura	47
3.3.3	Polygon	47
3.3.4	Truffle	47
3.3.5	Ganache	47
3.3.6	Firebase	48
3.4	Use Case Diagrams :	48
3.4.1	Farmer's Use Case Diagrams	48
3.4.2	Wholesaler's Use Case Diagrams	50
3.4.3	Retailer's Use Case Diagrams	51
3.4.4	Customer's Use Case Diagrams	52
3.5	Classe Diagram:	53
3.6	Activity Diagrams:	54
3.6.1	Buying Activity Diagram	54
4	Implementation	55
4.1	Software Technologies	56
4.1.1	Blockchain Technologies	56
4.1.1.1	Solidity	56
4.1.1.2	Web3.js	56
4.1.1.3	Truffle	56
4.1.1.4	Ganache	56
4.1.2	Web Technologies	56
4.1.2.1	React.js	56
4.1.2.2	Material UI	56
4.1.2.3	HTML/CSS	57
4.1.2.4	Framer motion	57
4.1.3	Mobile Technologies	57
4.1.3.1	Flutter	57
4.1.3.2	Web3dart	57
4.1.3.3	Provider	57
4.1.3.4	qr_flutter	57
4.1.3.5	qr_code_scanner	58
4.1.3.6	image_picker	58
4.1.3.7	image_cropper	58
4.1.4	Database Technologies	58

4.1.4.1	Firestore Cloud Firestore	58
4.1.4.2	Firestore Cloud Storage	58
4.2	Platform Functionalities	58
4.2.1	Smart Contract Functionalities	59
4.2.1.1	Traceability:	59
4.2.1.2	Our Token:	59
4.2.1.3	Digital Signature:	59
4.2.1.4	Code Optimisation	60
4.2.2	Web App Functionalities	61
4.2.2.1	Client SignUp	61
4.2.2.2	Farmer Add A Product	63
4.2.2.3	Client Buy A Product	64
4.2.2.4	Move Product From Stock To Sale	70
4.2.2.5	Modify a Product	71
4.2.3	Mobile Fonctionalities	72
4.2.3.1	Creation of account for farmer :	72
4.2.3.2	Adding Product For Farmer :	73
4.2.3.3	Create An Account For Wholesaler:	77
4.2.3.4	Buy A Product For Wholesaler:	80
4.2.3.5	Confirme Receiving A Product For Wholesaler: . . .	83
4.2.3.6	Farmer Information After An Order :	84
4.2.3.7	Confirme Sending Of Product For The Farmer : . . .	85
4.2.3.8	farmer Information After Confirm Sending Of Product: .	86
4.2.3.9	Wholesaler Information After Receiving Product : . .	87
4.2.3.10	Add Product To Sell For Wholesaler :	88

List of Figures

1.1	A Ledger Table [3]	9
1.2	Key-based asymmetric algorithm [1]	11
1.3	Digital signatures [1]	11
1.4	the relationship between the private key, the public key and the address.[18]	12
1.5	Private key and Public key [18]	12
1.6	Transactions Hashed in a Merkle Tree[2]	14
1.7	search for an element in the naive approach and the Merkle tree[14]	15
1.8	Decomposition of a block [19]	16
1.9	The Chaining of Blocks [19]	17
1.10	Bitcoin Network Interactions [5]	18
1.11	Bitcoin Logo	20
1.12	Ethereum Logo	21
1.13	Smart contract Exemple	22
1.14	Ether transfert [9]	23
1.15	Gas fees [9]	24
1.16	web3	25
1.17	popular dapps	25
1.18	Bitcoin VS Ethereum [9]	26
1.19	hyperledger Logo	26
1.20	Azure Logo	27
1.21	Fabric Logo	27
1.22	IBM Logo	28
1.23	iCommunity Logo	28
1.24	Skuchain Logo	29
1.25	IPFS etaps	29

LIST OF FIGURES

2.1	Supply Chain Management Schema	33
2.2	SCOR Model's elements	35
3.1	Global Architecture	46
3.2	Farmer use case diagram	49
3.3	Wholesaler use case diagram	50
3.4	Retailer use case diagram	51
3.5	Customer use case diagram	52
3.6	Customer use case diagram	53
3.7	Activity diagram	54
4.1	Traceability Core Code	59
4.2	Coin Core Code	59
4.3	Digital Signature Function Part1	60
4.4	Digital Signature Function Part2	60
4.5	Digital Signature Function Part3	60
4.6	Code Optimisation With Events	60
4.7	Code Optimisation With IDs List	61
4.8	Web Account Creation Interface	62
4.9	Web Account Email Validation Interface	63
4.10	Farmer Add A Product Interface	64
4.11	Buy A Product Interface Part1	65
4.12	Buy A Product Interface Part2	66
4.13	Sign The Product Interface	67
4.14	Confirme Signing The Product Interface	68
4.15	Balance After Buying Interface	69
4.16	Transferring a product from stock to sale Interface Part1	70
4.17	Transferring a product from stock to sale Interface Part2	71
4.18	Modifying A Product Interface	72
4.19	Farmer create an account	73
4.20	Adding product for Farmer	75
4.21	Farmer Add Product	76
4.22	Account Creation for wholesaler.	78
4.23	account creation for wholesaler.	79
4.24	Account Creation for wholesaler	81
4.25	Mobile Account Creation for wholesaler	82
4.26	Confirme receiving of products.	83
4.27	Farmer information after order.	84

LIST OF FIGURES

4.28 farmer confirm sending. 85

4.29 farmer information after confirm sending. 86

4.30 wholesaler information after receiving. 87

4.31 sell product wholesaler. 88

Introduction

Context and Problem statement

In Algeria, there are many problems in the agricultural and commercial sectors because of a lack of homogeneity and cooperation between them. Among these problems is the difficulty of counting the quantity of agricultural production for each season and each region. As well as the difficulty of marketing the products to the farmer, sometimes the farmer cannot find someone to sell his product to, which makes the farmer exposed to bankruptcy and corruption of his product. There is also the issue of price. Sometimes the farmer does not get a profit from his crop because of the low price resulting from the abundance of the product in large quantities, or on the contrary, the price is very high, which affects the purchasing power of the citizens as well as another problem, which is the presence of shortage places that suffer from the scarcity or lack of products due to the lack of market organization between farmers and wholesale and retail sellers.

Objectives

The goal of this project is to create a blockchain platform for food supply chain management and manage all the previous issues. Our solution has many advantages, including:

- Regulate the prices and fight corruption.
- Treating areas that suffer from a lack of nutrients.
- Achieving self-sufficiency.
- Save lives in the case of food contamination by tracking the distribution of all products and knowing their owners, which allows us to eliminate the product before it reaches the customer.
- Get fresh food by knowing the real source.
- The sale is improved by integrating a digital signature and a QR code.
- Enhance the business by gaining the trust of the customers.

Project outline

We have structured our report as follows:

Part I gives the introduction and explains the main problem. Part II, which presents the background part, gives an overview of the general concepts used in the thesis. The background part covers the field of blockchain with its architecture. Then, we give some concepts of food supply chain management. Then, in part II, we present the system design, which includes the global architecture, use case diagrams, class diagram, and activity diagram. In Part III, we describe the implementation of our solution. We provide the software technologies in addition to the platform features with a test.

Finally, we conclude our report in Part IV by presenting our prospects and future works.

Part I

Background

Chapter **1**

Blockchain Technology

1.1 Introduction

Blockchains are distributed digital ledgers that are both transparent and resilient, they open up the possibility of a decentralized world in which users of the technology can be empowered without being dependent on third-party power brokers or a central authority in general. This industry is emerging and expanding, and many believe that blockchain like the internet, is revolutionary.

Our goal is to learn more about the architecture of blockchain, its categories, and where and how it can be used most effectively.

1.1.1 The History Of Blockchain

In the early 1990s, the first paper describing the use of a chain of cryptographically secured blocks to preserve the integrity of past information and keep it resistant to tampering was published by two physicists, Stuart Haber and W Scott Stornetta, they had no idea that this could be basic starting research for a revolutionary technology[4].

The notion of proof-of-work was devised in 1993 in reaction to the rise of spam and other network abuses, with the goal of providing countermeasures.

During the global financial crisis of 2007-2008, the problems with centralization came to a head, it mean we must trust central authorities such as banks, governments, and other institutions. For example: if your credit card information is stolen from a bank's database, the bank's centralization of your information has been used against you. Satoshi Nakamoto (an pseudonym for an unnamed individual or group of individuals) published the Bitcoin Whitepaper titled "Bitcoin: A Peer-to-Peer Electronic Cash System". In the midst of the financial crisis, Bitcoin attracted attention for its ability to allow peer-to-peer transactions without the use of a centralized intermediary, and some people believe that Bitcoin is the blockchain, which is incorrect[4].

Vitalik Buterin, a community contributor, saw some flaws in Bitcoin's architecture and began developing an open source protocol in late 2013, which is now known as Ethereum, the second biggest project after Bitcoin. By using Ethereum Virtual Machine (EVM) that allow smart contracts (programs) to be deployed onto blockchain, opens the door to wide rang of use cases such as ownership tracking, real estate, and so on.

1.1.2 Benefits Of Blockchain

We determine the architectural style to be used based on the purpose and requirements of the project, which can be decentralized, centralized or both. The use of blockchain, where the architecture is decentralized, brings plenty of benefits to the table:

- A blockchain ledger's data cannot be changed.
- There is really no downtime on the blockchain because it is decentralized, so transactions can therefore be sent at any time and from any location.
- Disintermediation : without the need of intermediaries, blockchain allows for peer- to-peer digital asset transfers, this is the most important aspect of blockchain technology.
- It can be public or private, depending on the needs of an individual or company, and is therefore adaptable.
- Everyone can validate and add transactions to the database because it is open to the public.
- It's a highly secure database that executes transactions using public and private keys.
- The ledger can be audited at any moment.

With all these shared advantages, each blockchain network is introduced to the market with a particular objective, for example: Bitcoin's main objective was to create an alternative digital currency in the market and thus build a totally secure and transparent payment and transaction system. Ethereum, on the other hand, was created as a platform for contracts and peer-to-peer applications[4].

1.1.3 The Meaning Of Decentralization?

Decentralization is primarily about transferring power and authority in a community away from a single central authority and placing it in the hands of the community's members, allowing them to exercise self-sovereignty. The centralized authority may result to companies like Facebook, and Google finding methods to monetize user data in ways that people may not want or even be informed of. They offer free online services but use customers as a product. Their data is valuable[4]. BitTorrent is a nice example, it is a peer-to-peer file-sharing technology that does not rely on a single server, corporation, or entity to function. Bitcoin is similar in that it does not

require a bank to act as a central arbiter between two parties wishing to exchange value, the protocol gives them the ability to accomplish it on their own.

1.2 Blockchain Categories

The blockchains we have been talking about are public blockchains, anyone may join Bitcoin and Ethereum since they are open public networks[19]. This isn't the case for all networks, there are several different types of blockchains: public, consortium, and private blockchain.

1.2.1 Public Blockchains

Public blockchain (permissionless) networks are decentralized ledger platforms open to anyone to download the software, create a node, publishing blocks, and read the blockchain, without needing permission from any authority. Because permissionless blockchain networks are available to everyone, malevolent individuals may try to disrupt the system by publishing blocks. To avoid this, public blockchain networks often use a "consensus" model, which compels users to consume or maintain resources in order to publish blocks.

1.2.2 Consortium Blockchains (Federated)

Consortium blockchains, also known as shared permission blockchains[19], that only allow specific members to be nodes. You may have a consortium blockchain with three companies nodes, all of these entities would be able to sign transactions and would have access and audit this ledger, this might be classified as a low-trust condition. Organizations have a degree of trust in one another but not total trust. Although power is not centralized in any one corporation, the transactions are not visible to the general public.

1.2.3 Private Blockchains

Only authorized users maintain the blockchain and issue transactions in a private blockchain, which makes a high level of trust. It is particular use by developers who can set up and control their own blockchain instance to test their software and experiment with prototypes[4]. Private blockchain networks can have the same traceability of digital assets as they pass through the blockchain, as well as the same distributed, resilient, and redundant data storage system.

1.3 Blockchain Components

Now that we know what blockchain is and its different types, let's look at the different components of blockchain individually and try to understand what makes blockchain functional. Through the intelligent use of distributed ledgers, cryptography, and computer science, blockchain can create value, trust, and truth[4].

1.3.1 Distributed Ledgers

A ledger is a collection of transactions, similar to the old paper ledgers that were used to record the exchange of goods and services, except that in the blockchain, ledgers are recorded digitally and they are distributed, figure 1.1 shows an example of a ledger. Ledger has to assume two characters at the same time, the first is that a ledger may be used to prove ownership by reading historical data stored in the ledger, the second is that a ledger must also document any transfer of ownership, which means new data must be created and entered to the ledger.

\$	97.67	From:	Ripley	->	Lambert
\$	48.61	From:	Kane	->	Ash
\$	6.15	From:	Parker	->	Dallas
\$	10.44	From:	Hicks	->	Newt
\$	88.32	From:	Bishop	->	Burke
\$	45.00	From:	Hudson	->	Gorman
\$	92.00	From:	Vasquez	->	Apone

Figure 1.1: A Ledger Table [3]

Here are [19] the differences between centralized and decentralized ledgers:

- Ledgers held by the central government can be lost or destroyed. A blockchain network is built to be distributed, with many backup copies that all update and synchronize on the same ledger data across peers.
- Centrally ledgers are on a homogenous network, a cyberattack on one portion of the network will affect the entire network. But a blockchain network is heterogeneous, an attack on one node does not assure that an assault on other

nodes would succeed.

- Since central ledgers are situated in specific geographic locations, the ledger and services may be unavailable during network disruptions. The blockchain network on the other hand, is comprised of geographically varied nodes located all over the world, it can tolerate the loss of nodes.
- The transactions on a centralized ledger are not transparent and may not be valid, so the user must trust the owner. The blockchain network must validate all transactions, if a malicious node sent out incorrect transactions, others would notice and refuse them.
- It's possible that transaction data on a centrally ledger has been changed. The use of cryptographic mechanisms in the blockchain network provides obvious and resilient records.
- Because of the distributed nature of blockchain, it provides no centralized point of attack to steal information. In general, the information on a blockchain network is publicly available and cannot be stolen.

1.3.2 Public Key Cryptography

In the digital world where all people are connected, Alice wants to send a message to Bob and that no one and no third party can read it, the message can be a bank information or any other personal or sensitive data, here is where cryptography comes in. Cryptography is the study of secure communication mechanisms that enable only the sender and recipient of a message to read its contents. There are two main types of cryptography: symmetric-key and asymmetric key.

The use of the same key for encryption and decryption of data called symmetric-key cryptography, which is incredibly quick to compute, however, we will face the problem of how to send the key over a public network without anyone being able to read it, as a result, we must employ asymmetric-key cryptography, in which the data is encrypted using symmetric-key cryptography and the symmetric-key is encrypted using asymmetric-key cryptography. Asymmetric key cryptography uses a pair of keys: a public key and a private key that are mathematically related to each other. As illustrated in the figure 1.2, the receiver's public key used to encrypt the message and the receiver's private key used to decrypt the message.

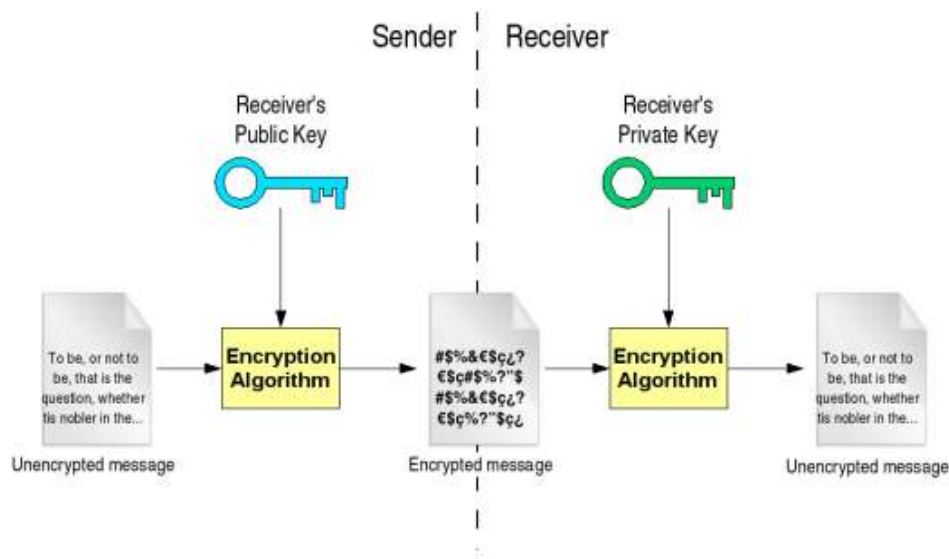


Figure 1.2: Key-based asymmetric algorithm [1]

Blockchain technology uses asymmetric key cryptography to generate verifiable historical records of transactional data and to generate addresses. To digitally sign transactions, private keys are utilized, and the recipient can verify that the transaction was sent by you by verifying the signature with your public key (see figure 1.3). By signing a transaction, you create a trust connection between users who do not know or trust each other.

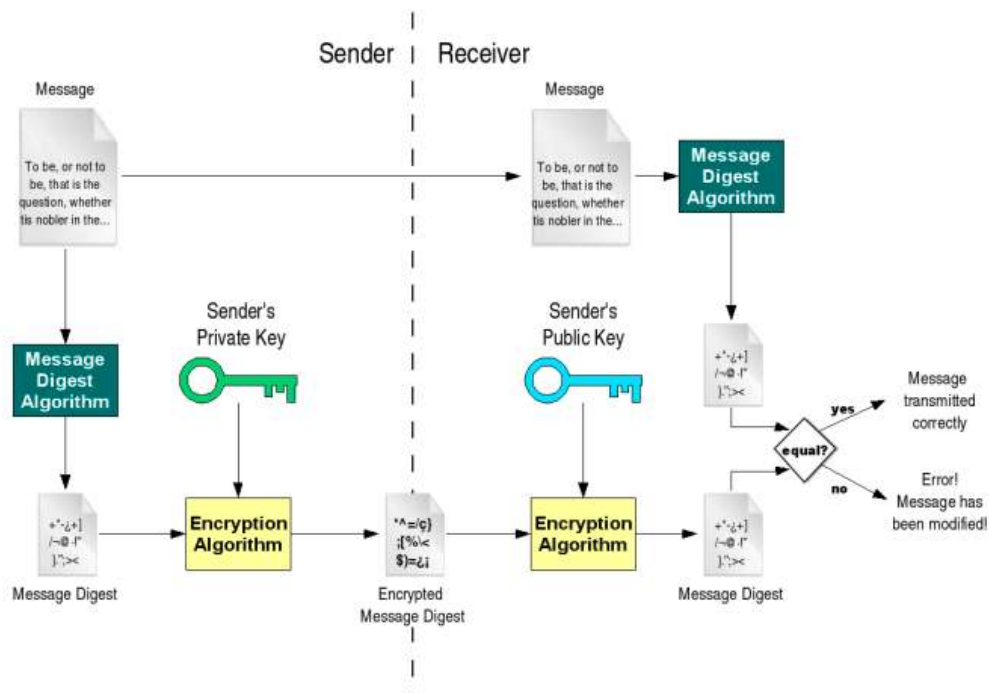


Figure 1.3: Digital signatures [1]

as shown in figure 1.4 ,The Private Key is used to obtain the Public Key mathematically, which is then processed with a hash function to give the address that others can view.



Figure 1.4: the relationship between the private key, the public key and the address.[18]

Addresses are used as the "to" and "from" endpoints in most blockchain systems, and they are shorter than public keys. The public key can be made public without compromising the process's security, but the private key must be kept secret since it can decode communications encrypted with public key. Because the Public Keys are generated using a complicated mathematical method, it is extremely difficult to "reverse" the procedure to determine the private key based on knowledge of the public key(Figure 1.5).

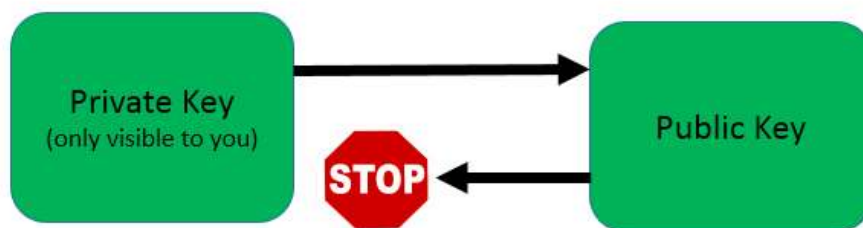


Figure 1.5: Private key and Public key [18]

The use of public key cryptography is critical to the security of blockchain. It serves as the foundation for how digital wallets operate, tokens are traded, and identities are validated. It is also used to produce reliable historical recordings of transactional data.

1.3.3 Cryptographic Hash Functions

Hashing is an algorithm (digital mechanism) that accepts any digital data of any size as input and returns another string of random characters with a fixed length as output known as a digest, and it is hard to determine the input string from the output string. There are several types of hashing algorithms including SHA-1, SHA-2, and SHA-256. The great feature of hash functions is that they are deterministic (the same input will always yield the same output), irreversible (it is difficult to predict input based on output), and collisions are rare, this means that inputs that are close to each other end up completely different.

One use of hash functions is password storage in databases, a hash of your password can be generated and saved, next time when you provide the password, it will be hashed using the same function you used to create it, then the two are compared. If an attack occurs, the attacker is unable to read or guess the password.

A blockchain uses hash functions to generate a record of the data added to the blockchain, making it easy to identify any changes made to a single bit of data. The Bitcoin blockchain uses the SHA-256 (Secure Hashing Algorithm) hash function that generates a 256-bit (64-character hexadecimal) output, while the Ethereum blockchain uses the Keccak-256 hash algorithm [19].

1.3.4 Merkle Tree

The Merkle tree is used to easily check if a data element is present in the block with the fastest way [4], for example: If a user wants to verify that a specific transaction is present and completed, in the tradition, he has to download the whole block (all data) to check an information, but using merkle tree, he needs to download some piece of data. Now we will see the construction of the Merkle tree (see figure 1.6):

1. The actual values called leaf nodes, are at the bottom of the tree.
2. Each of these transactions is first hashed.
3. Each hash value is paired with another hash value adjacent to it, to generate a new hash value.
4. This procedure is repeated until you obtain a single hash of all the transactions in the current block, which is known as the Merkle Root.
5. This Merkle root now goes to the block header as well as the following block, where it is recorded as the preceding block's hash.

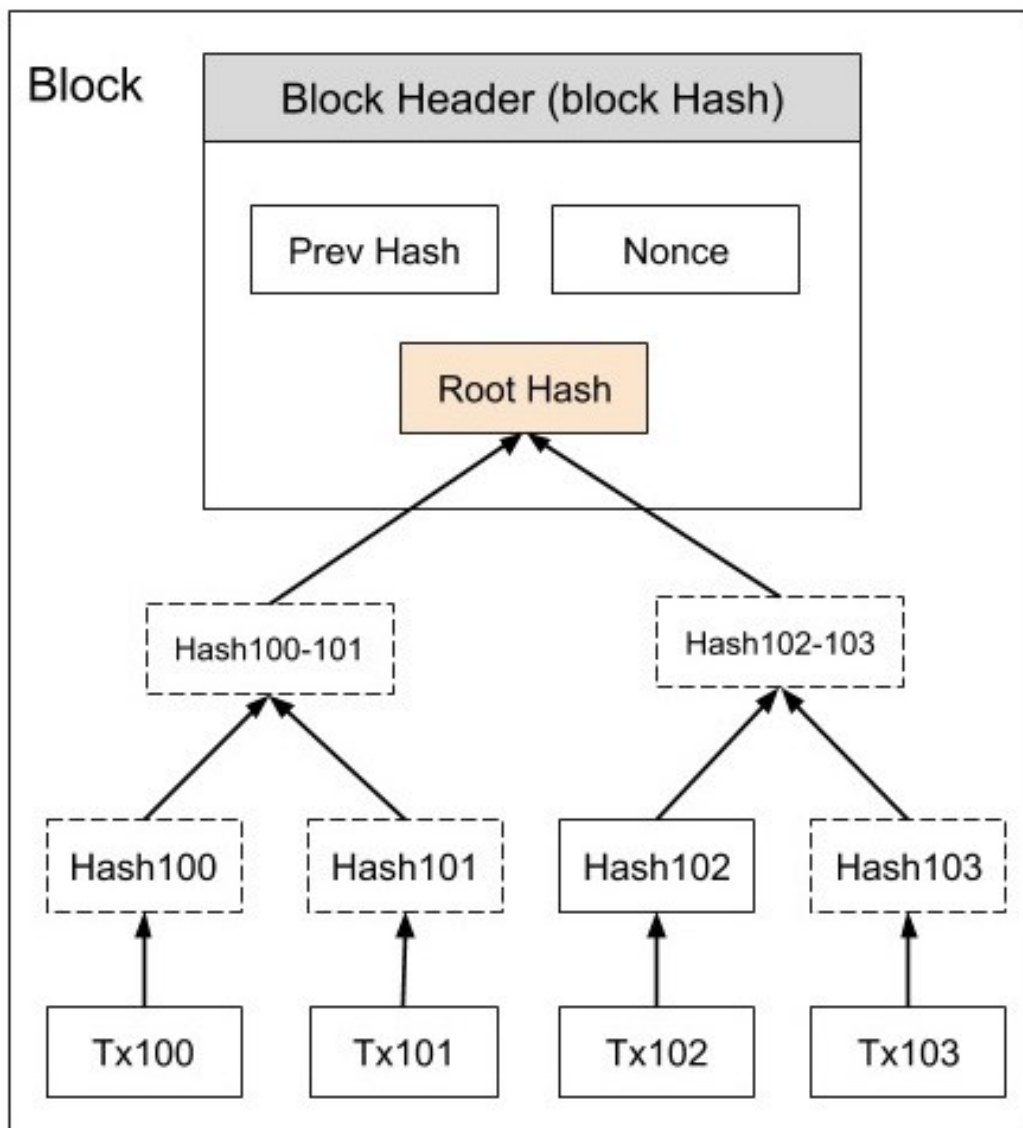


Figure 1.6: Transactions Hashed in a Merkle Tree[2]

We take an example (figure 1.7) of storing information about bananas, and see the difference between the Naive approach and the Merkle tree :

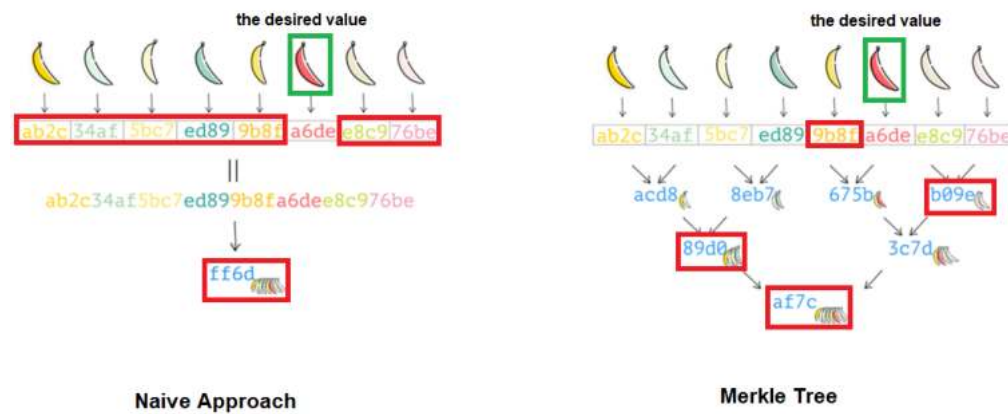


Figure 1.7: search for an element in the naive approach and the Merkle tree[14]

To verify that the "Red banana" is part of the bunch represented by the final hash "ff6d", by using Naive Approach there is no way without the entire bunch, that mean it requires rerunning the entire process and verify that it's consistent with the final hash, but using Merkle tree we only need handful of hashes, that mean it requires only a few computations and verify that it's consistent with Merkle root.

1.3.5 Structure Of A Block

We have talked a lot about hashes and blocks, now we will see what the block contraction is. The Blockchain is built with chained blocks, as we have already seen, they are chained with the hash of the previous block. The main purpose of a block is to record transactions, and the number of transactions per block varies between blockchains and is limited by block size constraints in order to avoid network congestion.

Every block has a unique number called height, a timestamp, a strange number called a nonce, some other information(transactions) and a hash of the previous block in the chain, as illustrated by the fugue 1.8:

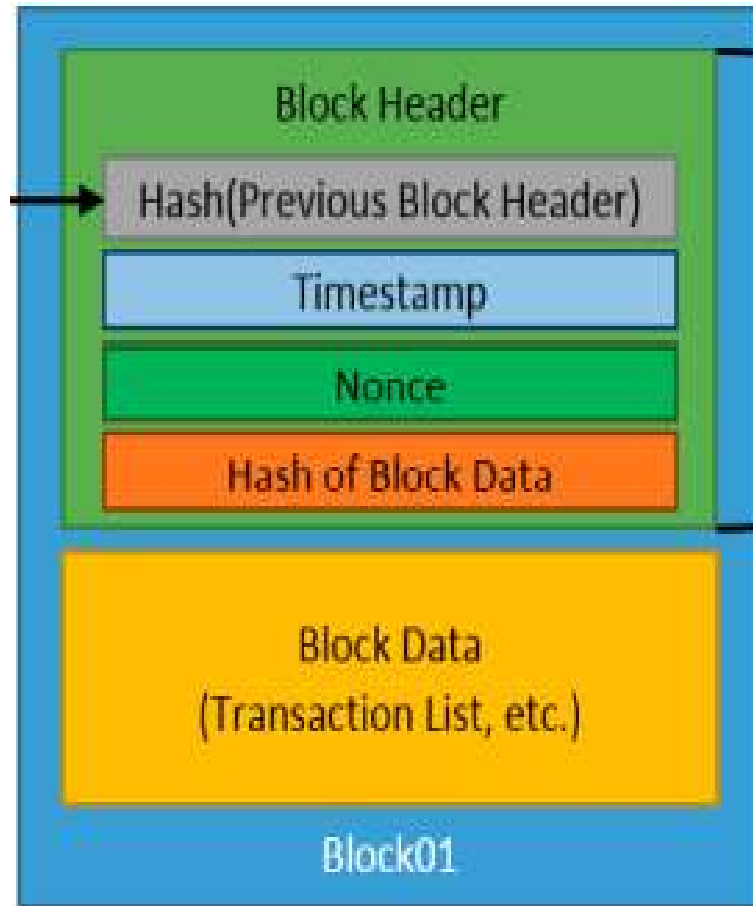


Figure 1.8: Decomposition of a block [19]

1.3.6 Defining Mining, Difficulty, Validation

There would be no value if anyone could simply build blocks and throw them on the chain, and the networks would never agree on which blocks should go on the chain. The vast majority of blocks are not valid, and identifying a valid block is the hard task that miners perform when they mine, this is called difficulty, and its level can be varied to ensure that blocks are created at regular intervals. A block is only considered valid if the hash value of the entire block is less than a certain threshold number, and the difficulty determines that threshold. We receive a unique signature for all of the data when we hash it all together, and any modification of the data will cause a modification of the hash. The miner changes a piece of data at random every time until they get a valid hash, that piece of data is known as the nonce, once they have found a nonce that is less than the difficulty threshold, the block is finally considered valid and can be broadcast to the network, and the miner receiving a reward for their efforts[19]. On the Bitcoin blockchain, Miners must now find hashes beginning with 19 zeroes [4]. When other nodes in the network receive

a valid block, they validate it by hashing it and ensuring that it is less difficult than the target difficulty, it is then added to their blockchain and work begins on the next block, which includes the hash of the previous block in its own content.

1.3.7 The Chain Of Blocks

The chaining of blocks (see figure 1.9) made by putting the hash of each previous block into the next block in the chain, that's why the data uploaded to the blockchain is considered permanent. If an attacker changes a transaction, it makes all the following blocks invalid, this is what makes the blockchain immutable. The first block of this chain is called block Genesis.

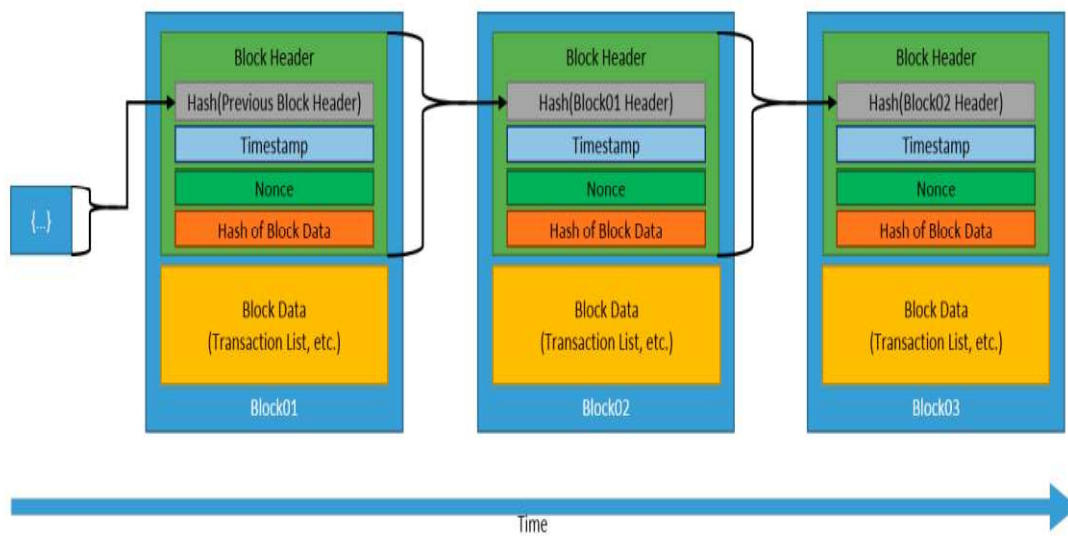


Figure 1.9: The Chaining of Blocks [19]

1.3.8 Nodes And Networks

Nodes are the computers that make up a blockchain network. In a server based system we can observe many problems, it requires high bandwidth to support all the incoming and outgoing traffic, if the server goes down or is hacked, the whole system will be unusable. Researchers found another solution which is the peer to peer model. The peers all connect to each other and are all equal. This network system is decentralized because it is composed of many nodes connected to each other, it is resistant to breakdowns, attacks, and optimizes the use of bandwidth, and the most attractive feature is that anyone can join by downloading the software and making their computer a node.

Nodes guarantee the blockchain's validity and keep local copies of it, there are several types of nodes: Full nodes hold the whole blockchain and validate each and every transaction, and other nodes known as light nodes, it just store a piece of the blockchain. Miners are distinct from nodes, they do not store the blockchain, instead they are network members that generate blocks and send them to nodes for verification and inclusion or rejection (see figure 1.10).

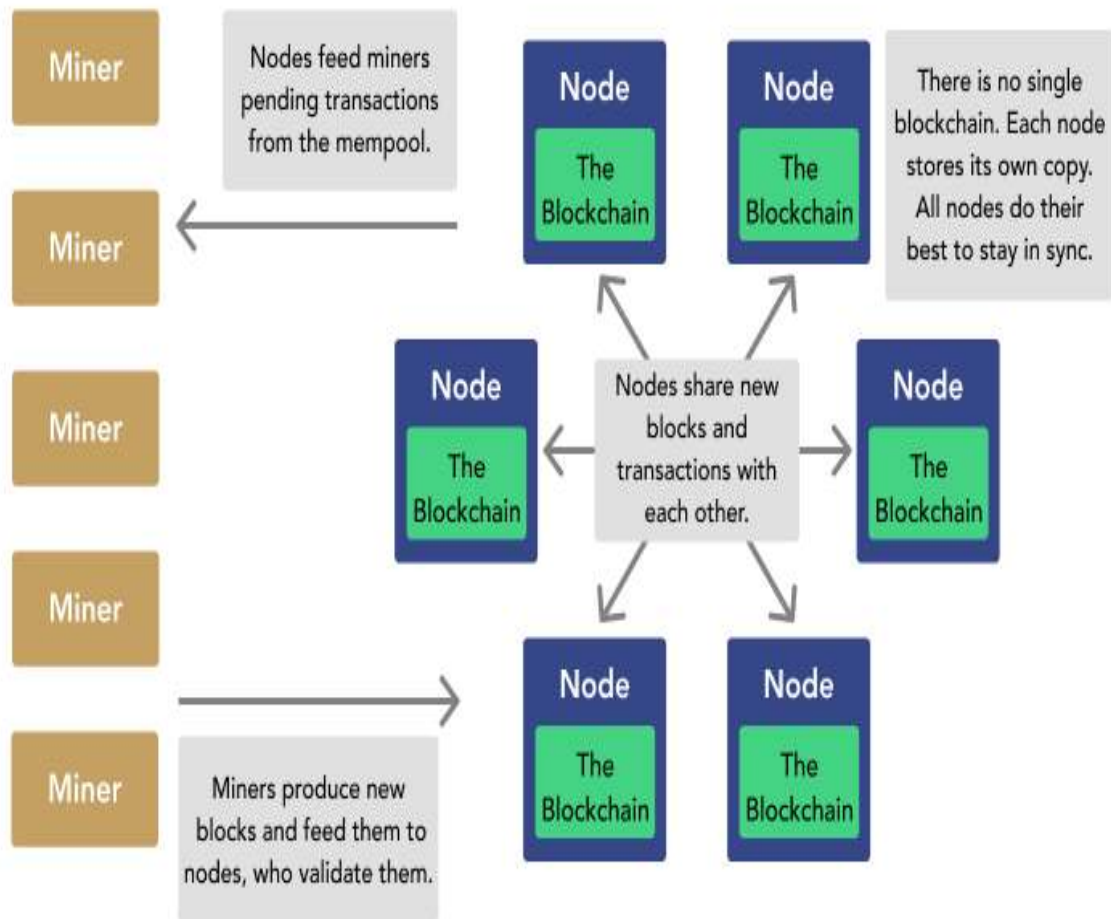


Figure 1.10: Bitcoin Network Interactions [5]

When a full node receives a valid block from a miner, it adds it to its own local copy of the blockchain and broadcasts it to a few other connected nodes that validate the block and broadcast it out as well, and so on [19]. As a result, the block travels around the network, and the process begins again on the following block.

1.4 Consensus Models

We have learned a little bit about how nodes agree on things, now we'll go a little further and see how blocks are validated by the miner and then distributed to the nodes, a valid block is proof of a lot of job. A consensus mechanism in a blockchain is a specific set of rules that allows a number of diverse nodes to trust each other and agree on the current state of the blockchain in a safe way, there are several consensus models [4, 19]:

1.4.1 Proof Of Work

The proof of work consensus method was initially introduced with Bitcoin, and it rewarded miners by sharing compute power to validate a block, the more calculation power you have, the more rewards you earn, so the one who solves it the fastest wins. This model is also used by Ethereum, it is safe but consumes a lot of energy and is slow.

Because proof of work consumes a lot of energy in a big network, transaction fees can be expensive, and they will continue to rise as the network develops, those fees represent the miners' profit. If forks occur as a result of various miners agreeing to distinct side chains, the longest chain that advances the fastest is the most trustworthy, others will follow immediately that chain, and other side chains will be eliminated.

1.4.2 Proof Of Stake

It rewards forgers based on the proportion of currency they own rather than computing power, so the more currency you own, the more rewards you earn. When creating a block, if the valid block includes a fraud, the fraudster loses his stake, this rule protects the network against attacks because the attacker has a large capital in the network. The currency represents a certain number of votes, and the forgers serve as witnesses to the transaction's processing and validation.

1.4.3 Proof Of Importance

It is similar to proof of stake, it is based on a higher probability of forging blocks rather than a larger share of the currency, some other variables are thrown into the mix such as preferring the one with more recent transaction activity.

1.4.4 Proof Of Authority

This model is used by Ethereum's Parity[19], where authorized accounts (validators) validate transactions in this consensus model, and individuals can only become a validator after presenting acceptable proof of identification. As a result, mining is unnecessary, scalability and speed are excellent, and security is enhanced.

1.5 Types Of Blockchain

- Type 1 : only cryptocurrency, example Bitcoin.
- Type 2 : currency + business logic, example ethereum.
- Type 3 : only business logic, example the linux foundation's hyperledger.

1.5.1 Only Cryptocurrency : Bitcoin

Bitcoin (logo in figure 1.11) is a cryptocurrency created by Satoshi Nakamoto in 2009 with no central authority to support peer-to-peer payment system that is based on the blockchain. The Bitcoin blockchain is the mother of all blockchains [9], it's free and open-source, The complete source code is accessible on GitHub, the minor's fees are now 6.5 BTC. This is how a new bitcoin currency is created.



Figure 1.11: Bitcoin Logo

1.5.2 Currency + Business logic : Ethereum

Ethereum (logo in figure 1.12) is a open-source blockchain support code execution, created by Vitalik Buterin in 2013, the platform's native cryptocurrency is Ether (ETH), a smart contract is the heart and soul of the ethereum blockchain.



Figure 1.12: Ethereum Logo

1.5.2.1 Smart Contract

A smart contract that is a program deployed and runs on a blockchain node. A smart contract is similar to a class in an object-oriented programming, it contains data, variables, functions, methods, getters and setters functions (see Figure 1.13). For coding smart contracts specific programming languages have been used, one such language is solidity.

```
pragma solidity ^0.4.22 <0.7.0;
contract Example{
uint data;

function getData() public view returns (uint){
    return data;
}

function setData(uint dataToSet) public {
    data = dataToSet;
}
}
```

Figure 1.13: Smart contract Exemple

1.5.2.2 EVM

Ethereum Virtual Machine is a smart contract code executor enables smart contracts to run on the ethereum blockchain, embedded within each ethereum node, every node will host the same smart contract codes on the EVM.

1.5.2.3 Accounts In Ethereum

Every account has a coin balance.

1. Externally Owned Accounts (EOA) : are accounts of nodes or for users, this accounts are keypairs of public and private keys, the balance is controlled by anyone with the private key.
2. Contract Accounts (CA) : are the accounts of smart contract, the balance is controlled by the code .

1.5.2.4 Accounts Vs Wallet

An account is not a wallet, an account is the keypair for a user-owned Ethereum account. A wallet is an interface or application that lets you interact with your ethereum account. like MetaMask [15].

1.5.2.5 Transaction Types

There are two types of transactions.

1. Transaction for Ether transfer (see Figure 1.14) .

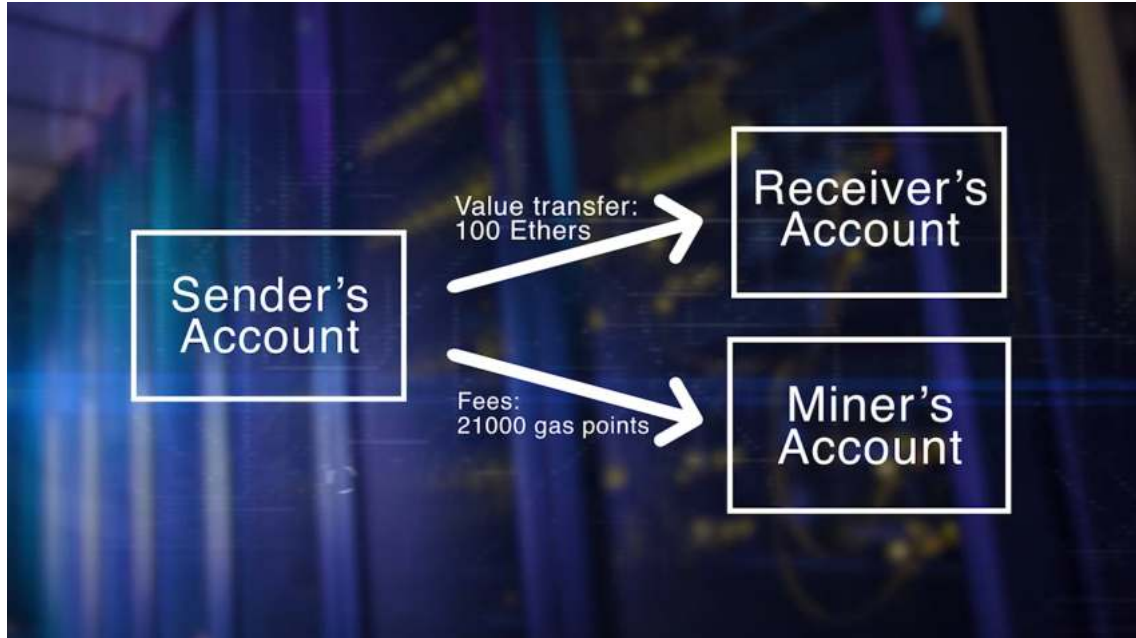



Figure 1.14: Ether transfert [9]

2. Transaction to invoke a smart contract code or execute smart contract's function.

1.5.2.6 Gas Fees

Is a unit of measurement for the amount of computing effort necessary to perform various activities on the ethereum network [15].



Operation name	Gas Cost
Step	1
Load from memory	20
Store into memory	100
Transaction base fee	21000
Contract creation	53000
...	...

Figure 1.15: Gas fees [9]

1.5.2.7 Decentralized Applications (DApps)

DApp is an end-to-end application using blockchain technology. People, apps, and systems that are not necessarily known to one other can transact peer to peer using DApp, a DApp has a front-end for a client interface and a back-end with a blockchain and a smart contract, and Web3 library to communicate with the back-end, DApp can be web app, mobile application, desktop application, or even IoT. [11]

1.5.2.8 Web3 :

It's a library for interacting with the ethereum blockchain in transaction sending, smart contract interaction, block data reading, and other use cases (see Figure 1.16), it can be written in several programming languages like JavaScript (web3.js), Python (web3.py), Dart (web3.dart)

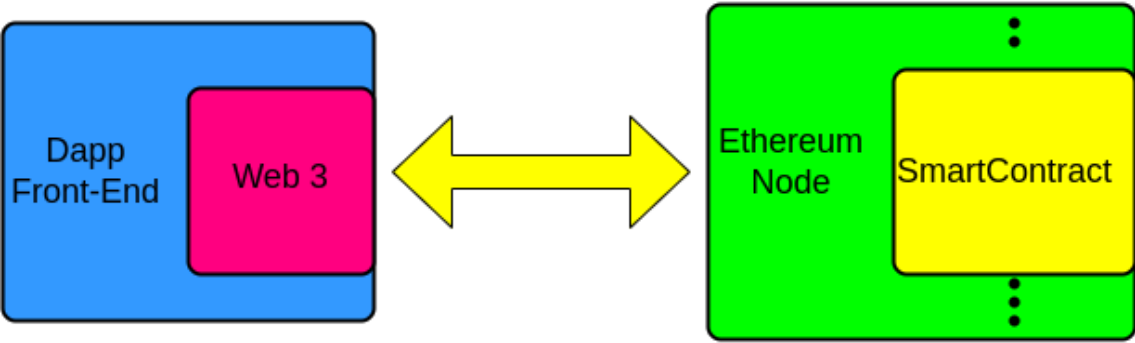


Figure 1.16: web3

1.5.2.9 Popular Dapps :

# ?		les plateformes	la catégorie	Utilisateurs (24h) ?	Le volume (7d) ?	Activité de développement (30d) ?	Activité de l'utilisateur (30d) ?
1	OpenSea Peer-to-peer marketplace for scarce digital goods	Ethereum	NFT	46,156 -12.88%	465,048 ETH 1,400,520,273 USD +6.35%	1,139 +17.79%	
2	Upland Property trading game with real-world addresses	EOS	NFT	63,454 -3.58%	0 EOS 0 USD -	-	
3	Solarbeam Shine a light on Defi, powered by Moonriver	Moonriver	DeFi	1,094 +3.66%	48,050 MOVR 4,256,776 USD -53.20%	72 +30.91%	
4	Oasis Where you can borrow Dai against your collateral	Ethereum	La finance	19,602 +4.83%	183,041 ETH 551,230,204 USD -50.26%	2,256 +77.50%	
5	Tether Digital money for a digital age	Ethereum	DeFi	28,487 +22.84%	2 ETH 7,427 USD -57.58%	-	
6	Uniswap Protocol for automated token exchange	Ethereum	DeFi	130 -6.12%	1 ETH 1,937 USD -99.40%	27,177 +451.59%	

Figure 1.17: popular dapps

1.5.2.10 Bitcoin Vs Ethereum

There is a difference between ethereum and bitcoin stack, ethereum used to develop a decentralized application based on a smart contract unlike bitcoin used only for currency. See Figure 1.18.

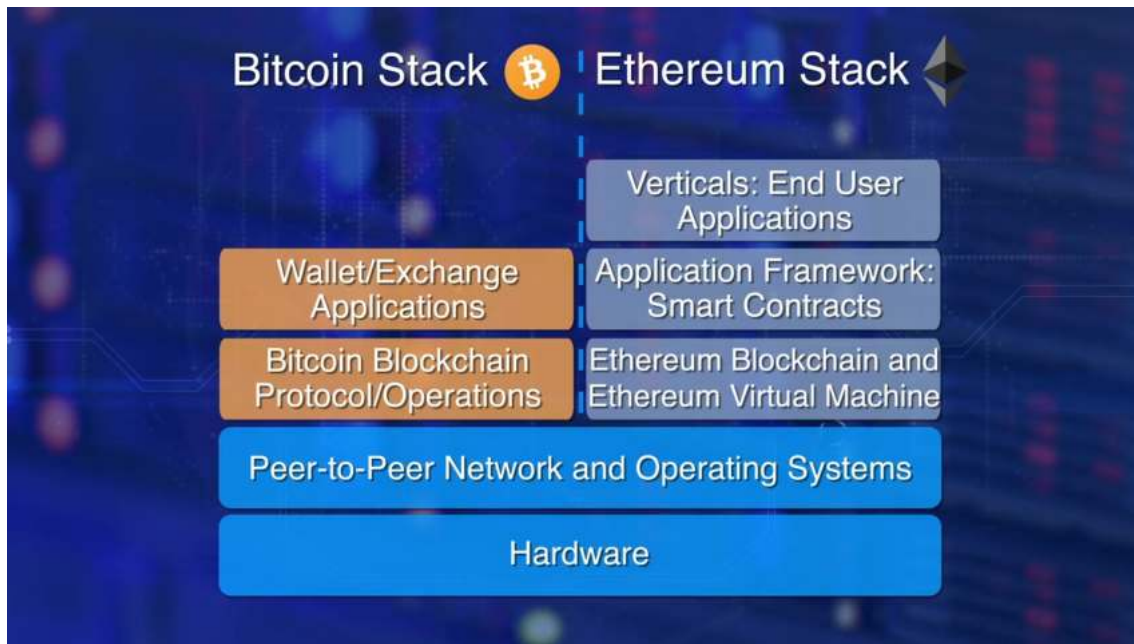


Figure 1.18: Bitcoin VS Ethereum [9]

1.5.3 Only Business Logic : Hyperledger

Hyperledger (logo in figure 1.19) is a project initiated by the linux foundation in 2015. The purpose was and still is to bring stakeholders, technology suppliers, and developers together to accelerate the development and acceptance of blockchain solutions, the hyperledger ecosystem supports not only the blockchain protocol, distributed ledger, and smart contract (smart contract is called chaincode in hyperledger), but also the structure and tools for active involvement and cooperation between developers, corporations, and other stakeholders [?].



Figure 1.19: hyperledger Logo

1.6 Blockchains Frameworks

1. Microsoft azure's blockchain as a service (logo in figure 1.20) : is plateforme service to easily create and test a variety of blockchains For example: ethereum or hyperledger. [10]



Figure 1.20: Azure Logo

2. Fabric Hyperledger (logo in figure 1.21) : from linux fondation is permis-sionned blockchain, unknown peers are unable to join and leave the network as they choose [?].

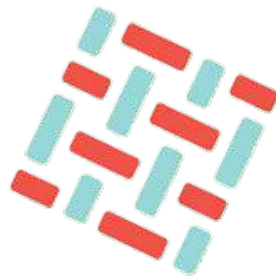


Figure 1.21: Fabric Logo

3. IBM blockchain as service (logo in figure 1.22) :is a paid service that allows companies to digitize transactions via a secure, shared and distributed ledger, improving efficiency [?].



Figure 1.22: IBM Logo

4. iCommunity (logo in figure 1.23): is a Blockchain-as-a-Service (BaaS) provider, with tools and use cases applications that allow companies to implement this technology in a simple cheap way [?].

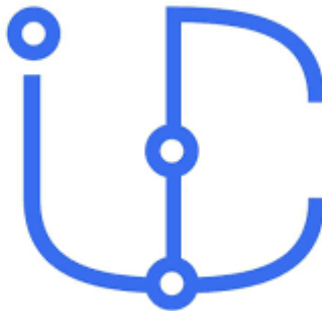


Figure 1.23: iCommunity Logo

5. Skuchain (logo in figure 1.24) offers a secure, digitized solution that can increase speed, reduce costs and make financing available to small and medium-sized businesses in previously unreachable locations [?].



Figure 1.24: Skuchain Logo

1.7 IPFS

IPFS (Interplanetary File System): is a decentralized system for peer to peer file transfer created by Juan Benet, for blockchain applications with a lot of data, it can play an important role as decentralized storage [10].

1.7.1 How To Use IPFS

First you upload the file to IPFS, then you get the hash of the file, then you can access the file by hash (see Figure 1.25)

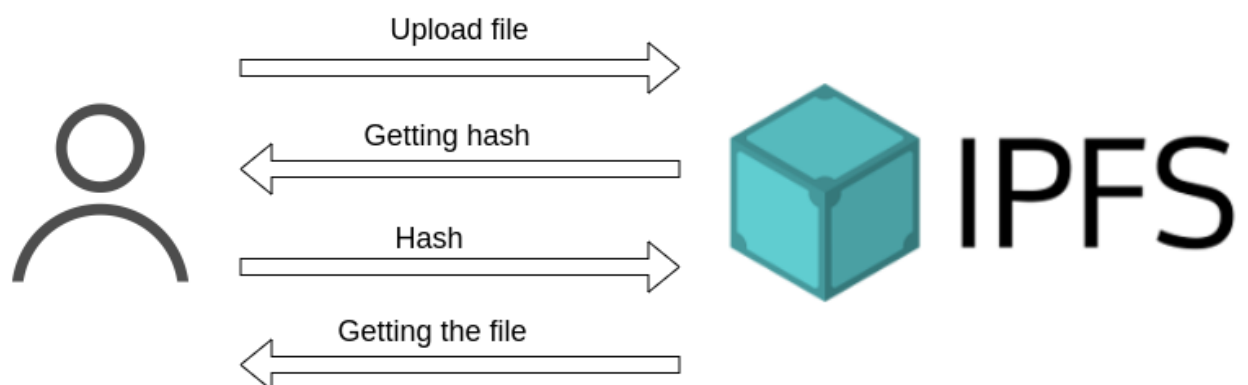


Figure 1.25: IPFS etaps

1.8 Blockchain Applications

Blockchain technology can be used in a variety of industries, including finance, healthcare, government, travel and tourism, Goods transfert and Retail, games, cryptocurrency, NFT...etc.

1. Finance (decentralized finance DeFi): blockchain technology is already being used in a variety of innovative ways in the financial services industry, This eliminates the need for brokers or intermediaries, while also ensuring transactional data transparency and efficiency, So that all parties can access the exact same data about the transaction [6]. Blockchain technology simplifies the entire process associated with asset management and payments.
2. Healthcare: blockchain has the potential to transform the healthcare industry by improving the privacy, security, and interoperability of patient data. It has the potential to alleviate many of the industry's interoperability issues and enable safe data exchange among the many companies and persons engaged in the process. It removes third-party influence while also avoiding overhead expenditures. Healthcare records may be kept in distributed data bases using Blockchains by encrypting them and using digital signatures to assure privacy and validity [6].
3. Government: government operations and services might be transformed by blockchain technology. It has the potential to make a significant difference in the Government sector's data transactional difficulties. With blockchain, data can be linked and shared properly, allowing for improved data management across multiple departments . It increases openness and makes monitoring and auditing transactions easier.
4. Travel and Tourism: blockchain may be used for money transactions, keeping crucial documents such as passports and other identity cards [6], making bookings, maintaining travel insurance, and rewarding loyalty.
5. Goods Transfert and Retail: the application of Blockchain technology in this sector has enormous potential. This involves confirming the authenticity of high-value commodities, avoiding fraudulent transactions, identifying stolen items, providing virtual warranties, maintaining loyalty points, and optimizing supply chain processes.
6. Cryptocurrencies: are a digital currencies without the need for a central bank. There are two types, Coins used its own blockchain like Ether on the contrary

the other type of token used the blockchain of a coin like USDC.

7. NFT(Non-fungible token) : NFT refers to a digital file such as photos, videos, and audio, which a digital certificate of authenticity or proof of ownership has been attached, NFT can be sold and traded.

1.9 Challenges And Solutions

1. Consensus Problem : consensus in a blockchain context means the general agreement of entire nodes in adding the next block to the chain, Bitcoin use proof of work which is computationally intensive, which means it uses a lot of energy.

Solution : change to proof of stake , proof of importance or proof of authority [10].

2. Scalability Problem : A system's scalability refers to its capacity to execute well at all levels of transactions, measured by transactions per second, the problem is that all nodes execute the functions of validation, verification, and recording of the transactions sequentially, not in parallel, Every full node stores the entire chain of blocks, All these functions take time This affects the transaction speed .

Solution : increase the size of the block for more transactions per block, or other solutions like state channel, sharding, parallel processing [10].

3. Privacy and Confidentiality : The term "privacy" refers to the protection of information from The untrusted parties, Data confidentiality ensures that only authorized parties have access to the information.

Solution : use permissioned blockchain, encrypt the data transacted and digitally sign[10] it.

Chapter 2

Traditional Supply Chain Management

2.1 Introduction

Supply Chain Management (SCM) is the process of managing the flow of goods and services from the point of raw material manufacture to consumer consumption (as shown in Figure 2.1). To transfer a product through each stage of this process, an organization needs a network of suppliers (links in the chain). It is the active streamlining of a company's supply-side activities to increase customer value and gain competitive advantage in the marketplace. It is a deliberate effort by supply chain companies to develop and manage supply chains in the most effective and efficient way possible [16].

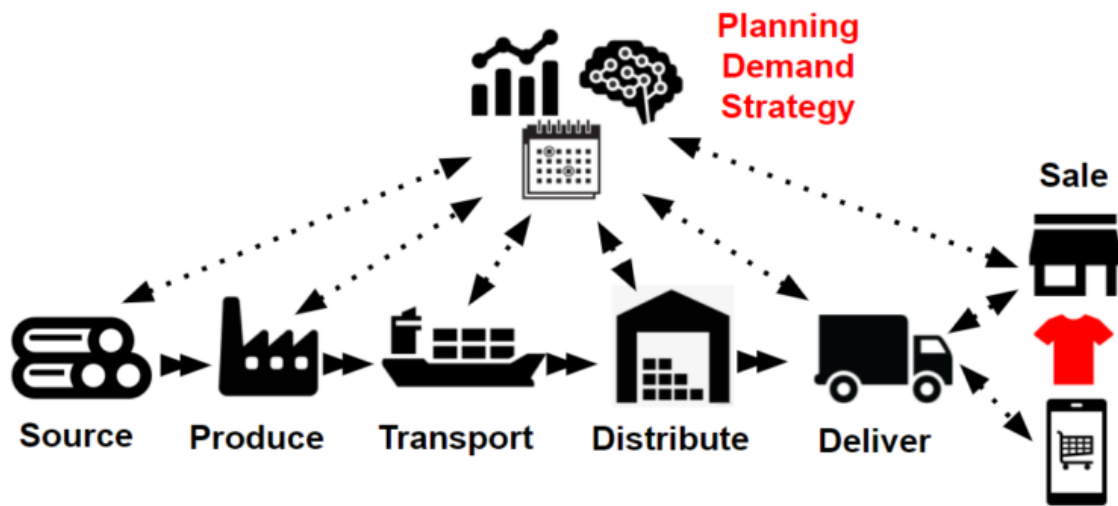


Figure 2.1: Supply Chain Management Schema

2.2 The Importance Of Supply Chain Management

- The flow and availability of essential human needs such as food, electricity, medicine, and modern infrastructure is a societal function of supply chain management[13].
- Improving Quality: Quality Assurance is defined as the observance of numerous customer-specified quality parameters, such as performance and specialized features. Following food safety requirements, exhibiting ethical and sustainable practices, and other comparable efforts are examples of this [13].

- Effective supply chain management enhances a company's financial position by producing value that is aligned with the company's business strategy [13].
- Supply Chain Management benefits businesses by improving the distribution process. It is vital to create adequate coordination between various transportation channels and warehouses in order to promote faster goods flow. Supply chain management allows companies to cut expenses while also delivering goods more rapidly. As a result, the entire distribution system has been upgraded, allowing products to be delivered at the right time and place. As a result, it's a good idea to invest in technology that allows you to efficiently manage inventory, provide detailed data, automate delivery, give real-time tracking, and conduct other distribution duties [13].
- The goal of supply chain management is to increase coordination among the company's many stakeholders. Employees, customers, and suppliers will be able to communicate with the company more efficiently thanks to the establishment of a channel. In the event of an emergency, managers may immediately lead their workforce, and employees can contact their supervisors via the established channel. Customers can also access relevant information through self-portals set up as part of the customer support system. It promotes information sharing among all stakeholders and aids in the creation of a well-coordinated organization [13].
- Through the distribution of products and services, supply chain management plays an important role in customer satisfaction. The scale of profitability for large organizations is relative to the management of an organization's supply chain. Good supply chain management is crucial for decreasing operating expenses from procurement operations [13].

2.3 Supply Chain Models

The supply chain models are tailored to the needs of the business. These models, along with their applications, are listed in the following [8]:

- **The Agile Model:** is suitable for organizations that deal with custom orders.
- **Continuous Flow Model:** provides consistency in a high-demand market with little variance.
- **Custom Configured Model:** Custom configurations are available at the production and assembly levels.

- **Efficient Chain Model:** is for enterprises in competitive markets where end-to-end efficiency is a priority.
- **Fast Chain Model:** is for companies that sell fashion items with a short life cycle.
- **Flexible Model:** allows you to meet high demand peaks while also managing long times of low volume movement
- **Supply Chain Operations Reference (SCOR) Model:** is to establish standards and continuous improvement in the SCM system.

- SCOR Model has three levels (Figure2.2) :

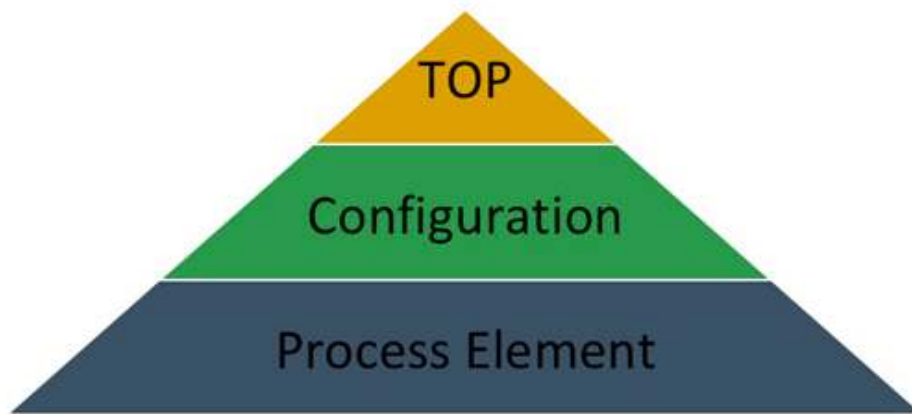


Figure 2.2: SCOR Model's elements

- There are five different processes at the top-level as follows, which are also known as supply chain management components [8]:
- **Plan:** Companies always try to match supply with overall demand by developing a plan of action through analysis. In addition, to avoid the bullwhip effect, it is prudent to closely monitor changes in demand along the value chain.
 - **Source:** is the process of locating vendors who would efficiently acquire products and services to satisfy anticipated demand. Suppliers must guarantee that the organization will supply high-quality items to the client. In the case of perishable goods, a minimum lead time from the supplier is required to enable a low-inventory strategy. In the case of non-perishable

items, the supplier's claimed lead time must be shorter than the number of days before inventory hits zero, resulting in no revenue loss.

- **Make:** According to the consumer's preferences, the company will handle all actions associated with the transformation of raw materials into finished products. Assembling, testing, and packaging are all done in this section of Supply Chain Management. Consumer feedback produces a win-win situation for both the producer and the end-user, since it allows the company to constantly improve their production procedures.
- **Deliver:** another crucial aspect of supply chain management is assisting with direct and indirect consumer integration. It has made a substantial contribution to the firm's brand image. Consumers' expectations for finished goods and services must be met through the company's delivery channels and logistical services. The company uses a variety of freight modes, including road, air, and rail, to ensure a smooth delivery.
- **Return:** It's a type of post-delivery customer service method that's linked to all sorts of returned items. It's also referred to as "reverse logistics." To avoid potential customer connection deterioration, it is one of the most critical components of supply chain management. On the other hand, this approach ensures that the firm's suppliers receive the same treatment.

2.4 Elements Of Supply Chain Management

- **Integration:** When the Communicating and Collaborating with All Parties teams are present and incorporated into the sequence of activities, it is easier to evaluate overall operations and uncover potential for future development [17].
- **Operations:** The backbone of the supply chain process is operations, which ensures that your employees have constant work. Managers keep an eye on day-to-day operations to ensure that all parts of the supply chain are running well. Many businesses have embraced lean manufacturing practices, in which all processes are assessed on a regular basis to see which elements of the business can be made more efficient. The operations team can make significant improvements to the supply chain system by monitoring equipment to ensure efficiency or understanding when to reduce staff [17].

- **Purchasing:** It's critical to understand exactly what things to purchase for your business, whether they're materials, supplies, tools, or equipment. A good supply chain starts with hiring qualified purchasing professionals and ensuring that your employees understand inventory management. This ensures that your organization does not have a material shortage, which can lead to significant production delays [17].
- **Distribution:** Customers receive your product, either from a retail shelf or via direct shipping, as the final step in the supply chain. Supply chain distribution must be well-planned in order for things to reach their final destination. Implementing logistics software for staff to learn or outsourcing to a third-party logistics (3PL) provider can ensure that products are handled properly and reach clients fast, which is a distributor's purpose [17].

2.5 Existing Companies

- **Lalamove:** is a Hong Kong-based logistics company that specializes in on-demand and same-day delivery, with a market capitalization of \$10.00 billion (February 2019). They provide a handy service by delivering right to your door or selected pickup spot [7].
- **RELEX Solutions:** It is Based in Finland and valued at \$5.70 billion, it offers an integrated retail and supply chain planning system that delivers results for customers all over the world (February 2022) [7].
- **Flexport:** Flexport is a low-cost, full-service global freight forwarder and logistics platform based in the United States that lowers friction in international trade. 3.20 billion dollars (as of April 2018) [7].
- **Huitongda :** is a Chinese integrated logistics service provider with a broad range of solutions that allow Chinese consumers to buy at home the largest selection of trusted international brands, with a market capitalization of \$3.18 billion (December 2017) [7].
- **Delhivery:** is a modern logistics service company based in India that creates bespoke supply chain solutions for businesses with a market capitalization of \$3.00 billion (February 2019) [7].
- **Convoy:** It is a real-time, demand-based digital freight network based in the United States that aims to improve efficiency and transparency between shippers and carriers. It is worth \$2.75 billion as of September

2018 [7].

- **Project44:** it is a logistics technology firm based in the United States that provides a visibility solution that spans the whole shipment workflow, with a market capitalization of \$2.62 billion (as of June 2021) [7].
- **Deliverr:** It is a startup firm based in the United States that provides shipping services for e-commerce businesses. It is valued at \$2.00 billion (as of November 2021) [7].
- **Exotec:** It provides a cutting-edge order preparation system based on a fleet of collaborative mobile robots valued at \$2.00 billion (as of January 2022) in France [7].
- **Loggi:** It is Based in Brazil and valued at \$2.00 billion (as of June 2019), is a technological platform utilized by delivery firms and their couriers. It enables couriers to pick up deliveries, ship items, track packages individually or in groups, communicate with one another and with clients, update delivery statuses in real time, and much more [7].

2.6 Food Supply Chain Management

A food supply chain refers to all of the steps involved in getting food from a farm to the dinner table. This encompasses the production, administration, use, and disposal of food to function. Every aspect of this activity necessitates the use of man-made resources or raw materials. Because each stage of the supply chain has an impact on the others, it's critical to simplify the entire process to avoid high costs or inefficiencies [12].

2.7 Stages Of The Food Supply Chain Management

- **Farm:** This is where the ingredients, such as meat, fruits and vegetables, meals, and beverages, come from and where they are purchased.
- **Processing:** Plants and animals are turned into edible forms during this stage.
- **Distributing:** Food is transported and distributed once it has become edible.

- **Wholesaler:** A wholesaler is a business or individual that buys large amounts of goods from farmers, they are stored in warehouses and then sold to retailers.
- **Retailer:** This is the method by which products are delivered to customers. Everything from obtaining the distributed items to selling them is covered.
- **Customer:** The food item is purchased by the customer from the retailer [13].

2.8 The Importance Of Food Supply Chain Management

The goal of the grocery store and restaurant industries is to obtain high-quality food at a low cost from the supplier so that they can still make a profit and offer consumers competitive prices. This improves customer satisfaction, brand loyalty, supply chain efficiency, and everyone's happiness. To achieve these goals, the supermarket and restaurant industries must monitor each stage of the supply chain [13]. Problems arise when food is lost or wasted during any part of the supply chain process. Unfortunately, experts estimate that roughly 30% of all food produced is thrown away. This has a negative impact on food security, the economy, and the environment. Food loss and waste can reduce the amount of food available on the market, resulting in higher prices and less access to food for low-income people.

Furthermore, if food quality deteriorates to the point that food must be sold at a lower price or thrown away, it can have an influence on farmers' and producers' well-being and livelihood.

Food loss and waste can be reduced by using food management to coordinate and control all aspects of the supply chain. Food management entails overseeing the supply chain to guarantee that all foods offered are of good quality, taste, and safety.

The goal is to ensure that any goods sold to shops comply with the requirements set by health regulators and government officials.

Food inspectors are in charge of preventing and detecting contamination, which can lead to food loss, higher prices, and food insecurity. All stages of the supply chain may suffer if effective food management measures are not implemented.

2.9 Main Problems With Food Supply Chains Management

There is often a lack of communication between partners in a food supply chain who are located around the world, resulting in food insecurity and public health issues. In addition, the entire food supply system has been pushed to adapt and change due to a global epidemic [13].

The main issues facing the food supply chain management are:

- **Farming-Labor and Shortages:** Grain prices have dropped simply because of the decline in global oil consumption since the COVID epidemic. As a result, it is more difficult to sell grain and make a profit, given all the costs associated with harvesting and harvesting. In addition, farmers are not recruiting as many migrant workers because of limits on immigration.
- **Poor Communication Between Supply Chain Participants:** Both the food system and the supply chain are fragmented. Each company has its own logistics system, data sharing protocols, and government regulations to comply with. As a result, it can be difficult for supply chain players to communicate with each other.
- **Growing Regulations:** Unfortunately, regulations that serve to protect people from harm have also caused immense harm. For example, the ELD mandate requires carriers to record the hours worked by their drivers. This requirement has led to layoffs, higher prices, and greater inefficiency.
- **Restaurants Inventory Management:** Restaurants encountered a number of problems due to a lack of visibility, poor communication between participants, and COVID-19. Many restaurants and their suppliers lack the technology to see the supply and demand of their inventory in real time. The relationship between the restaurant owner and their supplier is strained due to poor inventory management. The result is over-or under-ordering, loss, waste, and uncertainty of supply, which reduces customer satisfaction and damages the restaurant's reputation.

2.10 Conclusion

Supply chain management is an important part of every organization, although there are difficulties as it improves effectiveness, efficiency, resource management, etc. It

also establishes good and memorable relationships with stakeholders like suppliers, customers,...etc.

Part II

Our Solution

Chapter 3

System Design

3.1 Introduction

In this chapter, we will define the system design of our platform, which includes the application functionality, global architecture, use case diagrams, class diagram and activity diagram. At the end, you will cover the entire solution concept.

3.2 Functionality Considerations

In this section, we will give an overview of the different functionalities of each actor in our system, which will help to clarify things.

Farmer:

- Add a product
- consult the marketplace
- confirme sending the product by scaning QR code
- show the history
- consult personal information
- show in sale products
- get balance
- modify the product
- get suspended balance to receive

Wholesaler:

- consult the marketplace
- buy a product
- confirme receiving the product by generating a digital signature
- move the product from stock to sell
- confirme sending the product by scaning QR code
- show the history
- show in sale products
- show in stock products

- consult personal information
- get balance
- modify the product
- get suspended balance to receive
- get suspended balance to send

Retailer:

- consult the marketplace
- buy a product
- confirme receiving the product by generating a digital signature
- move the product from stock to sell
- confirme sending the product by scaning QR code
- show the history
- show in sale products
- show in stock products
- consult personal information
- get balance
- modify the product
- get suspended balance to receive
- get suspended balance to send

Customer:

- consult the marketplace
- buy a product
- confirme receiving the product by generating a digital signature
- show the history
- consult personal information
- get balance
- get suspended balance to send

3.3 Global Architecture

In section, we will cover the global architecte of our decentralized application (Figure 3.1)

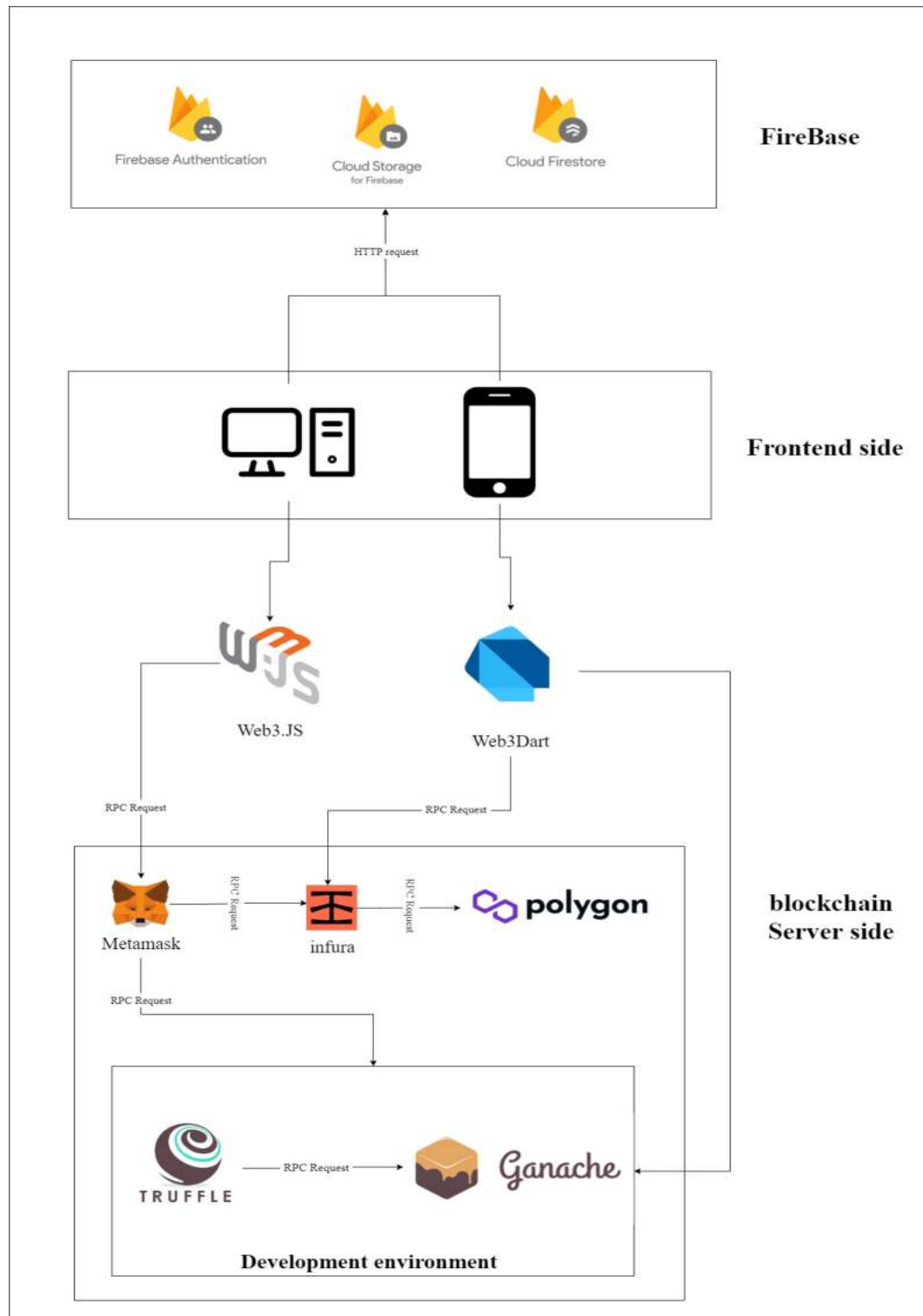


Figure 3.1: Global Architecture

3.3.1 Metamask

It is a crypto-currency wallet that also serves as a gateway to blockchain applications, with more than 30 million users worldwide relying on it. As a browser extension and mobile app, MetaMask provides a key vault, secure login, token wallet, and token exchange as a browser extension and mobile app, giving you everything you need to manage your digital assets. MetaMask is the easiest and most secure way to connect to blockchain applications.

3.3.2 Infura

It is a set of tools that allow anyone to create an Ethereum-connected application. It interacts with the Ethereum blockchain and manages users' nodes on their behalf. The Infura API suite provides instant access to the Ethereum network via HTTPS and WebSockets. It's never been easier to set up the infrastructure for your decentralized application. Infura is built on a cutting-edge microservice-driven architecture that dynamically scales to meet the demands of our APIs. Infura aims to make life easier for developers by dealing with issues such as the cost of storing data and connecting to the Ethereum blockchain.

3.3.3 Polygon

Polygon is a secondary scaling solution for the Ethereum blockchain that is compatible with and complements it. Polygon aspires to be a better blockchain development network than Ethereum. Polygon is more about affordable pricing, better bandwidth, and scalability than the Ethereum platform, which is more about functionality and security.

3.3.4 Truffle

Truffle is a world-class development environment, testing framework, and asset pipeline for Ethereum Virtual Machine (EVM)-based blockchains that aims to make life easier for developers. With over 1.5 million lifetime downloads, Truffle is largely regarded as the most popular tool for blockchain application development.

3.3.5 Ganache

Ganache is a personal Blockchain of Ethereum to build and test developments such as Dapps and Smart Contracts. It is available as both a desktop application and a

command-line tool (formerly known as TestRPC). Ganache is available for Windows, Mac and Linux.

3.3.6 Firebase

Google Firebase is a Google-backed app development platform that allows developers to create apps for iOS, Android, and the web. Firebase delivers analytics tracking, reporting, and app issue fixes, as well as marketing and product experimentation capabilities. When your apps only need a basic level of interaction with legacy systems or third-party services, Firebase is the way to go. When your application does not require substantial data processing or any type of complex user verification, Firebase becomes an excellent solution.

3.4 Use Case Diagrams :

To demonstrate the functionality of our platform, the use case diagram is used. The diagram represents the functionalities of the actors who interact with our platform.

3.4.1 Farmer's Use Case Diagrams

The (Figure 3.2) presents the use case diagram that describes the functionalities of the farmer.

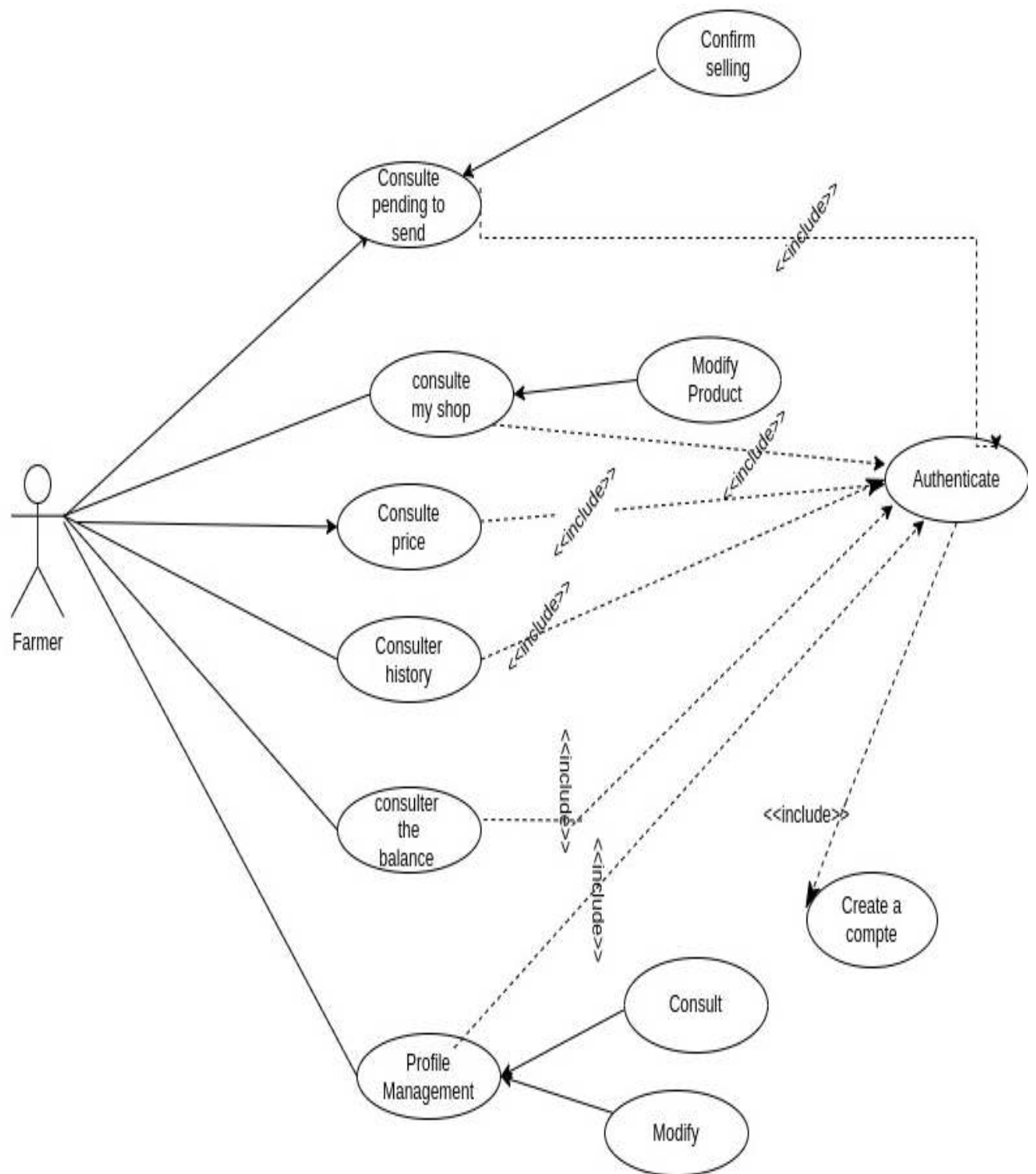


Figure 3.2: Farmer use case diagram

3.4.2 Wholesaler's Use Case Diagrams

The (Figure 3.3) presents the use case diagram that describes the functionalities of the wholesaler.

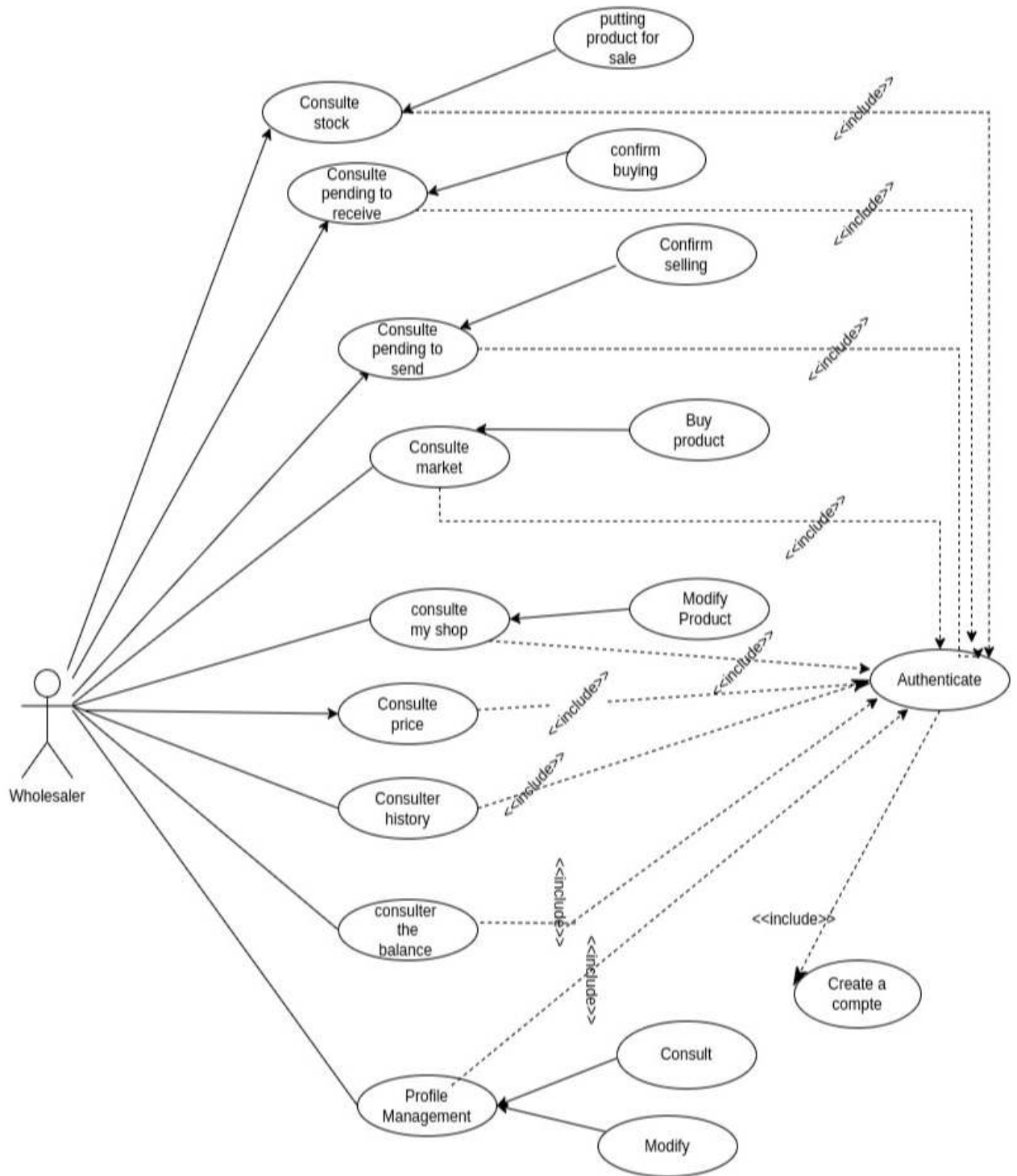


Figure 3.3: Wholesaler use case diagram

3.4.3 Retailer's Use Case Diagrams

The (Figure 3.4) presents the use case diagram that describes the functionalities of the retailer.

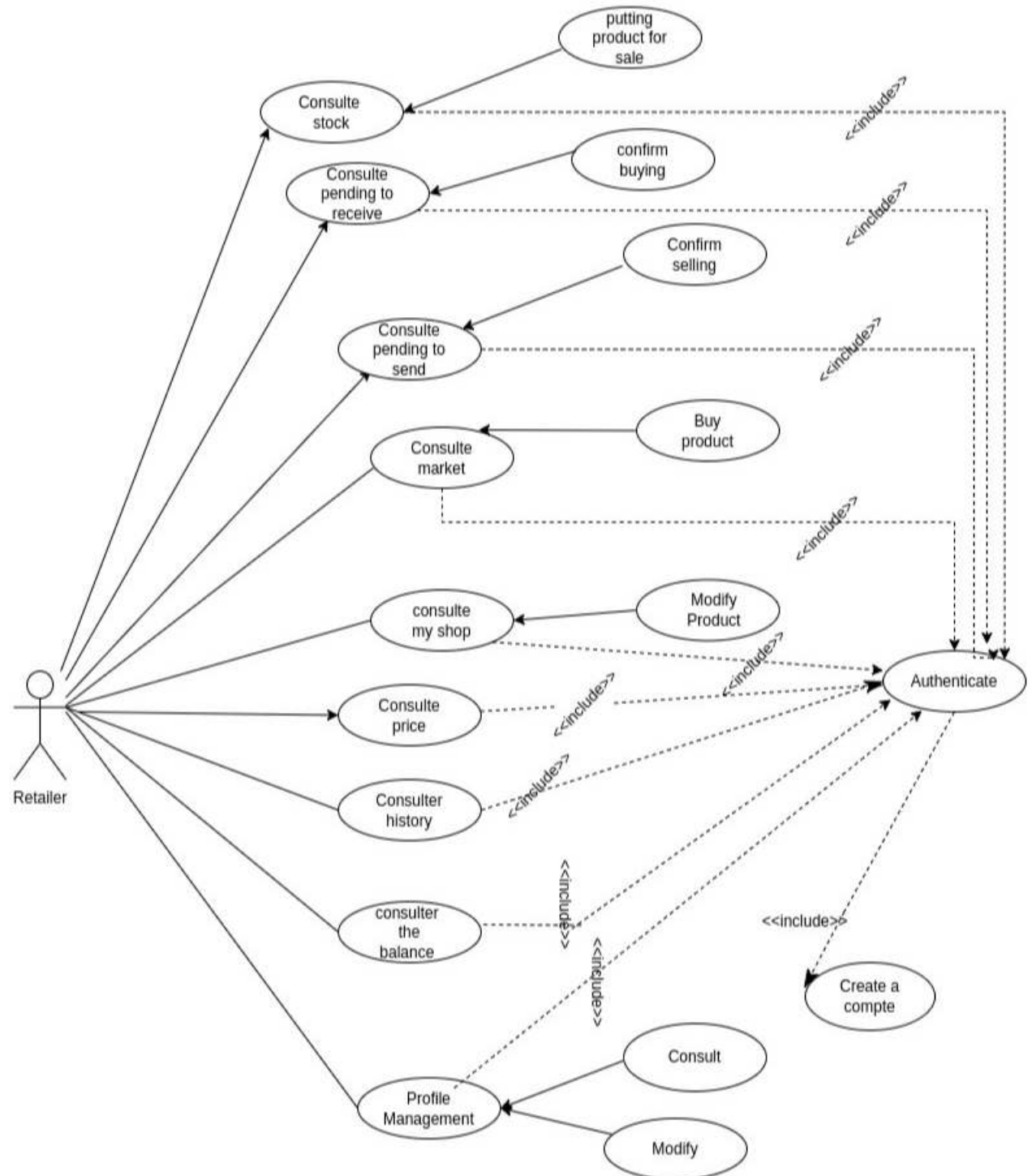


Figure 3.4: Retailer use case diagram

3.4.4 Customer's Use Case Diagrams

The (Figure 3.5) presents the use case diagram that describes the functionalities of the customer.

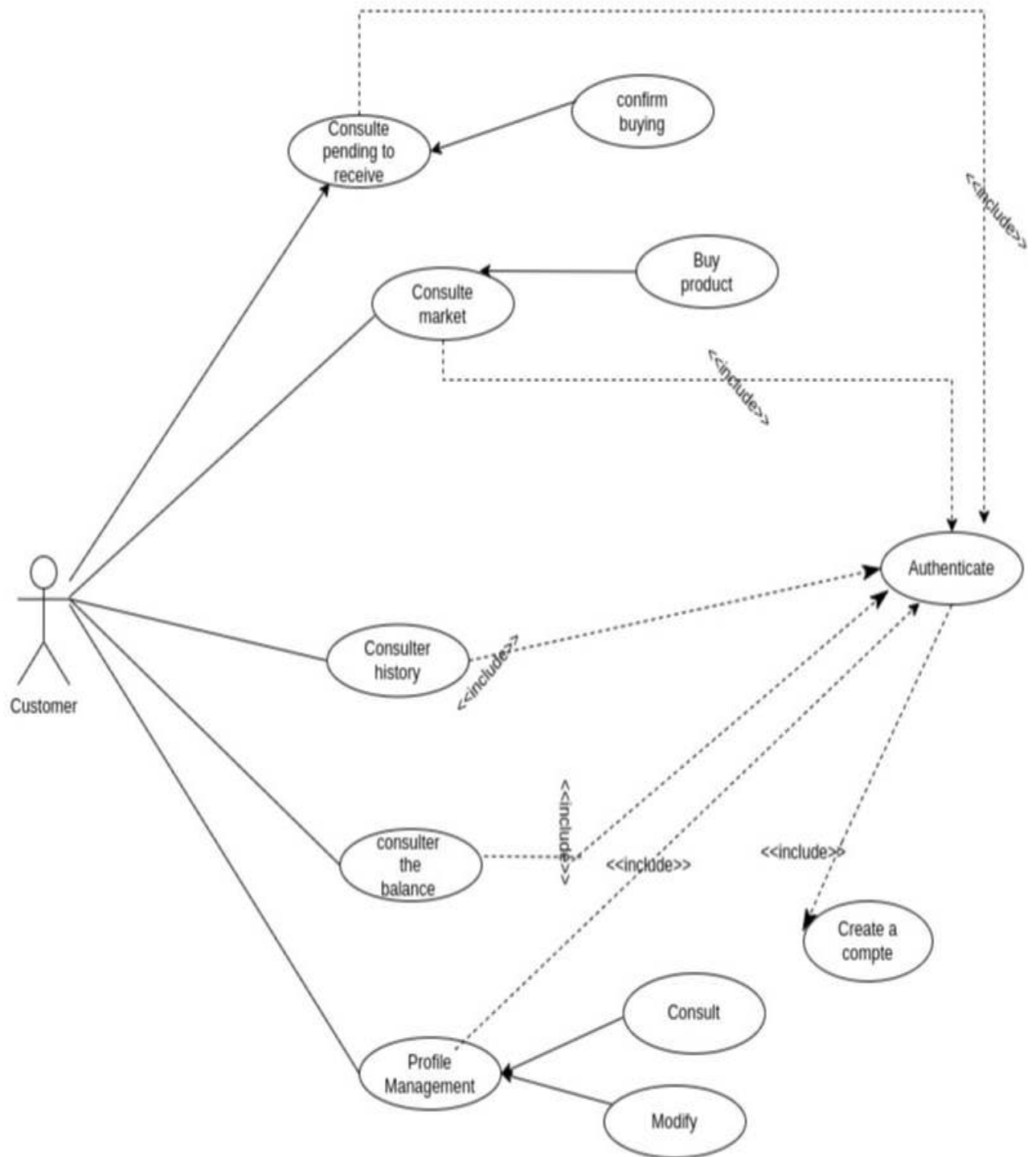


Figure 3.5: Customer use case diagram

3.5 Classe Diagram:

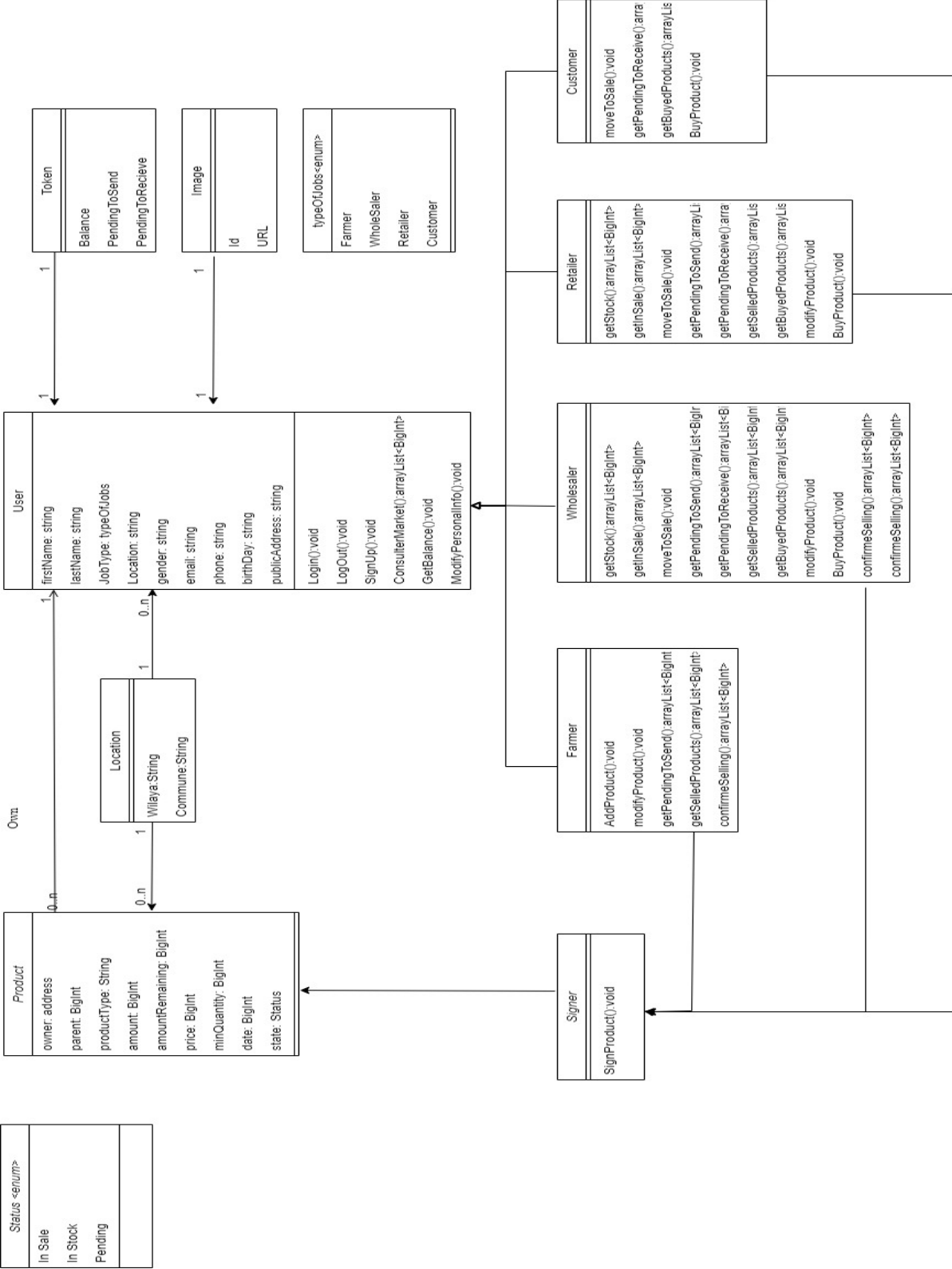


Figure 3.6: Customer use case diagram

3.6 Activity Diagrams:

3.6.1 Buying Activity Diagram

The (Figure 3.7) presents the activity diagram that describes the buy functionalities between farmer and wholesaler.

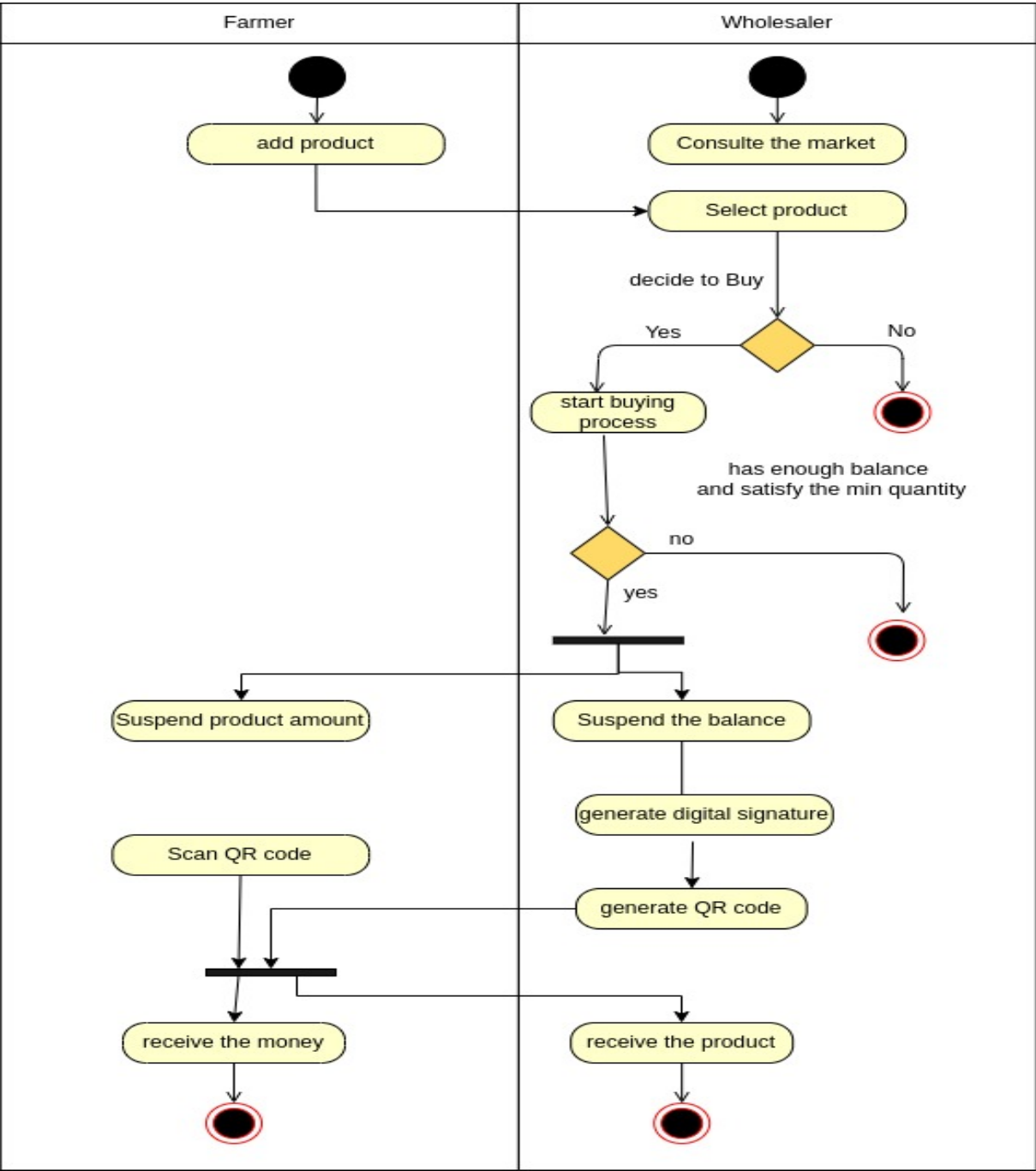


Figure 3.7: Activity diagram

Chapter 4

Implementation

4.1 Software Technologies

4.1.1 Blockchain Technologies

4.1.1.1 Solidity

Solidity is a high-level, object-oriented, statically typed programming language for building smart contracts on the Ethereum Virtual Machine (EVM). C++, Python, and JavaScript all have an influence. Other features supported by Solidity include inheritance, libraries, and sophisticated user-defined types.

4.1.1.2 Web3.js

web3.js is a collection of libraries that allow you to interact with a local or remote ethereum node using HTTP, IPC or WebSocket.

4.1.1.3 Truffle

Truffle is a world-class development environment, providing a test framework and asset wallet for Blockchain using the Máquina Virtual Ethereum (Ethereum Virtual Machine - EVM)

4.1.1.4 Ganache

Ganache is a personal Blockchain of Ethereum to build and test developments such as Dapps and Smart Contracts. It is available as both a desktop application and a command-line tool (formerly known as TestRPC). Ganache is available for Windows, Mac and Linux.

4.1.2 Web Technologies

4.1.2.1 React.js

React is a User Interface (UI) library for developing UI components that was built by Facebook. It allows developers to create massive web apps that may modify data without reloading the page. React's major goal is to be scalable, quick, and easy to use.

4.1.2.2 Material UI

Material-UI is a React UI toolkit that allows us import and use many components to construct a user interface in our React apps. It saves developers a lot of time

because they don't have to build everything from scratch.

4.1.2.3 HTML/CSS

HTML (Hypertext Markup Language) and CSS (Cascading Style Sheets) are two of the most common Web page building technologies. For a variety of devices, HTML provides the page structure and CSS provides the (visual and aural) layout. HTML and CSS, along with graphics and scripting, are the foundations for creating Web pages and Web applications.

4.1.2.4 Framer motion

An open source and production-ready React on the web motion library for all creative developers.

4.1.3 Mobile Technologies

4.1.3.1 Flutter

Flutter is a Google open source framework that allows you to create beautiful, natively built, multi-platform apps from a single codebase.

4.1.3.2 Web3dart

A dart library that connects to the Ethereum blockchain to communicate with it. It connects to an Ethereum node and allows you to send transactions, interact with smart contracts, and more .

4.1.3.3 Provider

Provider is an easy-to-use and powerful Flutter package. State management with a high level of performance, By providing a simple and appealing syntax that does not degrade app performance, Provider enables developers to achieve a high level of productivity. The user interface is separated from the presentation logic, business logic, dependency injection, and navigation. This aids the creation of clean code by default.

4.1.3.4 qr_flutter

is a Flutter library that allows you to render QR codes using a Widget or custom painter.

4.1.3.5 qr_code_scanner

By natively embedding the platform view into Flutter, a QR code scanner that works on both iOS and Android is created. The connection with Flutter is seamless, far superior to performing the scan in a native Activity .

4.1.3.6 image_picker

A Flutter plugin for picking images from the image library and capturing new pictures using the camera for iOS and Android.

4.1.3.7 image_cropper

Cropping images is possible with a Flutter plugin for Android, iOS, and Web. Because this plugin is based on three different native libraries, each platform's UI will be different.

4.1.4 Database Technologies

We use Firebase to handle the rest of our platform data, such as images, in addition to the Blockchain where we deploy our smart contract that takes care of saving rides information to protect users' data privacy.

4.1.4.1 Firebase Cloud Firestore

Firestore is a NoSQL document database that focuses on ease of use, automatic scaling, and high performance. While the Firestore interface offers many of the same features as traditional databases, it differs from a NoSQL database in how it shows relationships between data objects.

4.1.4.2 Firebase Cloud Storage

Cloud Storage for Firebase is a powerful, simple, and cost-effective Google-scaled object storage solution. The Firebase SDKs for Cloud Storage provide Google security for file uploads and downloads for your Firebase apps, regardless of network conditions.

4.2 Platform Functionalities

In this chapter, we will highlight the benefits of the platform and then present our solution, which is a web and mobile application. We will go through all the features

and describe each of them.

4.2.1 Smart Contract Functionalities

4.2.1.1 Traceability:

Figure 4.1 shows the main part of the code, which is a mappings that stores and manages the product IDs.

```
mapping (uint => productListData) farmersProductByType;
mapping (uint => productListData) wholesalersProductByType;
mapping (uint => productListData) retailersProductByType;
mapping(uint => ProductData) public productsData ;
mapping (address => farmersData) farmers;
mapping (address => wholeSalerRetailerData ) wholesalers;
mapping (address => wholeSalerRetailerData) retailers;
mapping (address => customerData) customers;
mapping (address => string) public usersJobType;
```

Figure 4.1: Traceability Core Code

4.2.1.2 Our Token:

Figure 4.2 shows the balances list that stores users' coins.

```
// SPDX-License-Identifier: MIT
pragma solidity >=0.4.22 <0.9.0;
pragma experimental ABIEncoderV2;
contract MyCoin {
mapping(address => uint) public balanceOf;
mapping(address=>toSend) public balanceToSend;
mapping(address=>toReceive) public balanceToReceive;
```

Figure 4.2: Coin Core Code

4.2.1.3 Digital Signature:

Figure 4.3 shows the function that handle the digital signature process. It uses a hash functions 4.4 and a special verification function 4.5.

```

function transfertWithSignature
(address receiver,address sender,bytes memory _signature,uint productId)
public returns(bool){
    bytes32 hash = getHash(toString(productId));
    bytes32 ethSignHash = getEthSignedHash(hash);
    address signerAddress = verify(ethSignHash, _signature);
    //ProductData memory product= productsData[productId];
    //address wholeSalerOwner = product.owner;
    // require(signerAddress==wholeSalerOwner, "invalid signature");
    require(signerAddress== sender,"signer are not the sender in input");
    balanceOf[receiver]+=balanceToReceive[receiver].keyAmount[toString(productId)];
    balanceToReceive[receiver].total-=balanceToReceive[receiver].keyAmount[toString(productId)];
    balanceToReceive[receiver].keyAmount[toString(productId)]=0;
    balanceToSend[sender].total-=balanceToSend[sender].keyAmount[toString(productId)];
    balanceToSend[sender].keyAmount[toString(productId)]=0;
    return true;
}

```

Figure 4.3: Digital Signature Function Part1

```

function getHash(string memory str) public pure returns (bytes32) {
    return keccak256(abi.encodePacked(str));
}
function getEthSignedHash(bytes32 _messageHash) public pure returns (bytes32) {
    return keccak256(abi.encodePacked("\x19Ethereum Signed Message:\n32", _messageHash));
}

```

Figure 4.4: Digital Signature Function Part2

```

function verify(bytes32 _ethSignedMessageHash, bytes memory _signature) public pure
returns (address) {
    (bytes32 r, bytes32 s, uint8 v) = splitSignature(_signature);
    return ecrecover(_ethSignedMessageHash, v, r, s);
}

```

Figure 4.5: Digital Signature Function Part3

4.2.1.4 Code Optimisation

To improve the code, we have optimized some features. To reduce gas fees, we used events (Figure 4.6) that store data in the blockchain network instead of smart contracts. And rather than sending several transactions separately, we send a list of products in one go (Figure 4.7).

```

event ProductLocation(uint indexed productId, string location);
event ProductAdded(address indexed owner,uint productId);
event ProductBought(address indexed owner,address indexed receiver,uint productId);

```

Figure 4.6: Code Optimisation With Events


```
function productDatafromList( uint[] memory _productsId) public view
returns ( ProductData[] memory){
    ProductData[] memory productssData = new ProductData[](_productsId.length);
    for (uint i = 0; i < _productsId.length; i++) {

        ProductData storage productProgress = productsData[_productsId[i]];
        productssData[i] = productProgress;
    }
    return productssData;
}
```

Figure 4.7: Code Optimisation With IDs List

4.2.2 Web App Functionalities

4.2.2.1 Client SignUp

This page (Figure 4.8) allows you to create a new account for the user using Firebase authentication with email verification (Figure 4.9). The user can only have one of the 4 types of jobs (Farmer, Wholesaler, Retailer, or Customer). The metamask address is needed to identify the user on the blockchain.

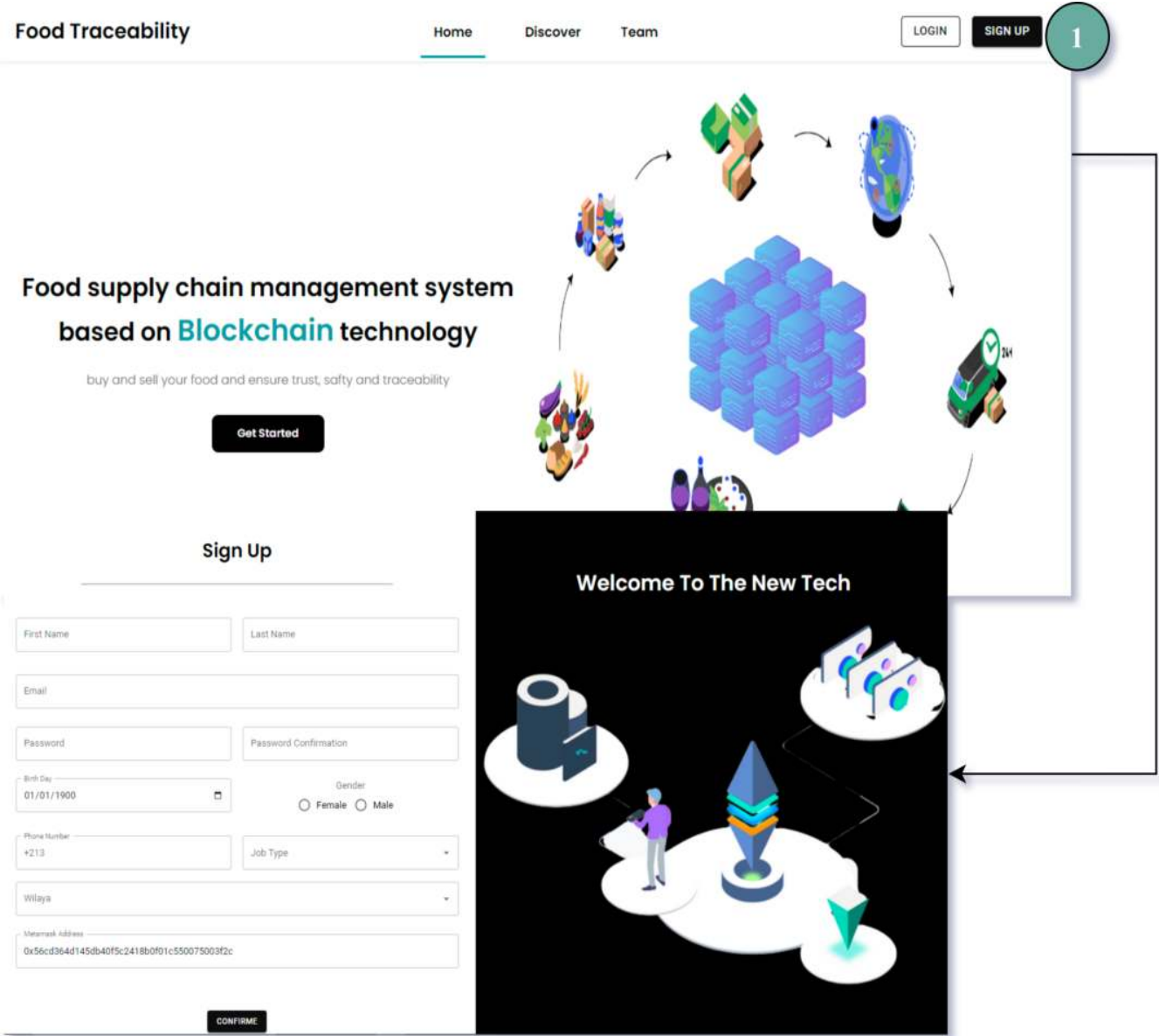


Figure 4.8: Web Account Creation Interface

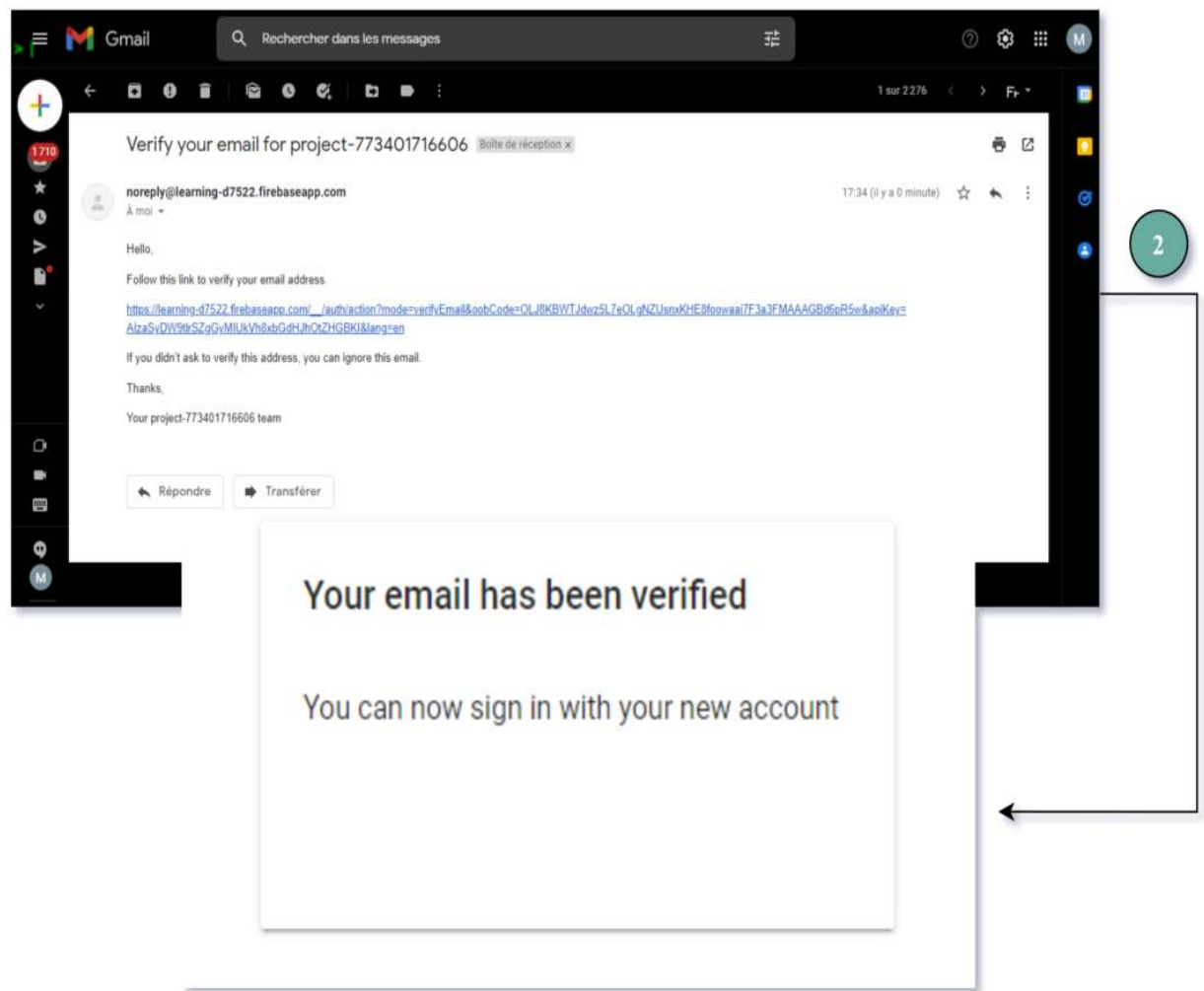


Figure 4.9: Web Account Email Validation Interface

4.2.2.2 Farmer Add A Product

As shown in (Figure 4.10), the farmer is the only user that can add a product to the market.

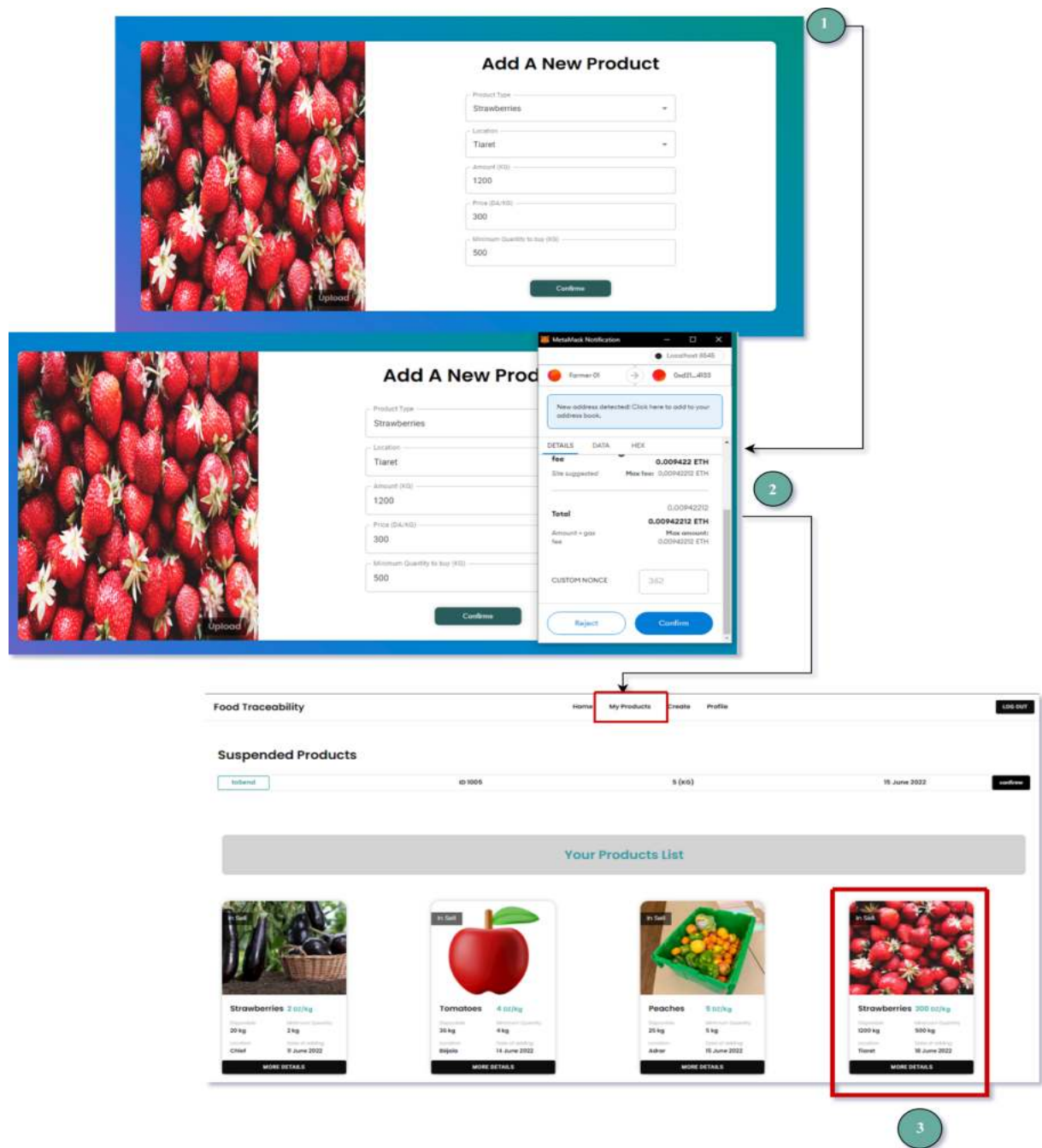


Figure 4.10: Farmer Add A Product Interface

4.2.2.3 Client Buy A Product

the wholesaler, retailer or customer can buy a product on the market by simply having a sufficient balance of our token (Figure 4.11 and 4.12). then the product and the balance are suspended in the smart contract. The buyer and the seller have to confirm the process by using the digital signature (Figure 4.13 and 4.14). After that, the seller receives his money (Figure 4.15), and the buyer receives his product

in his stock.

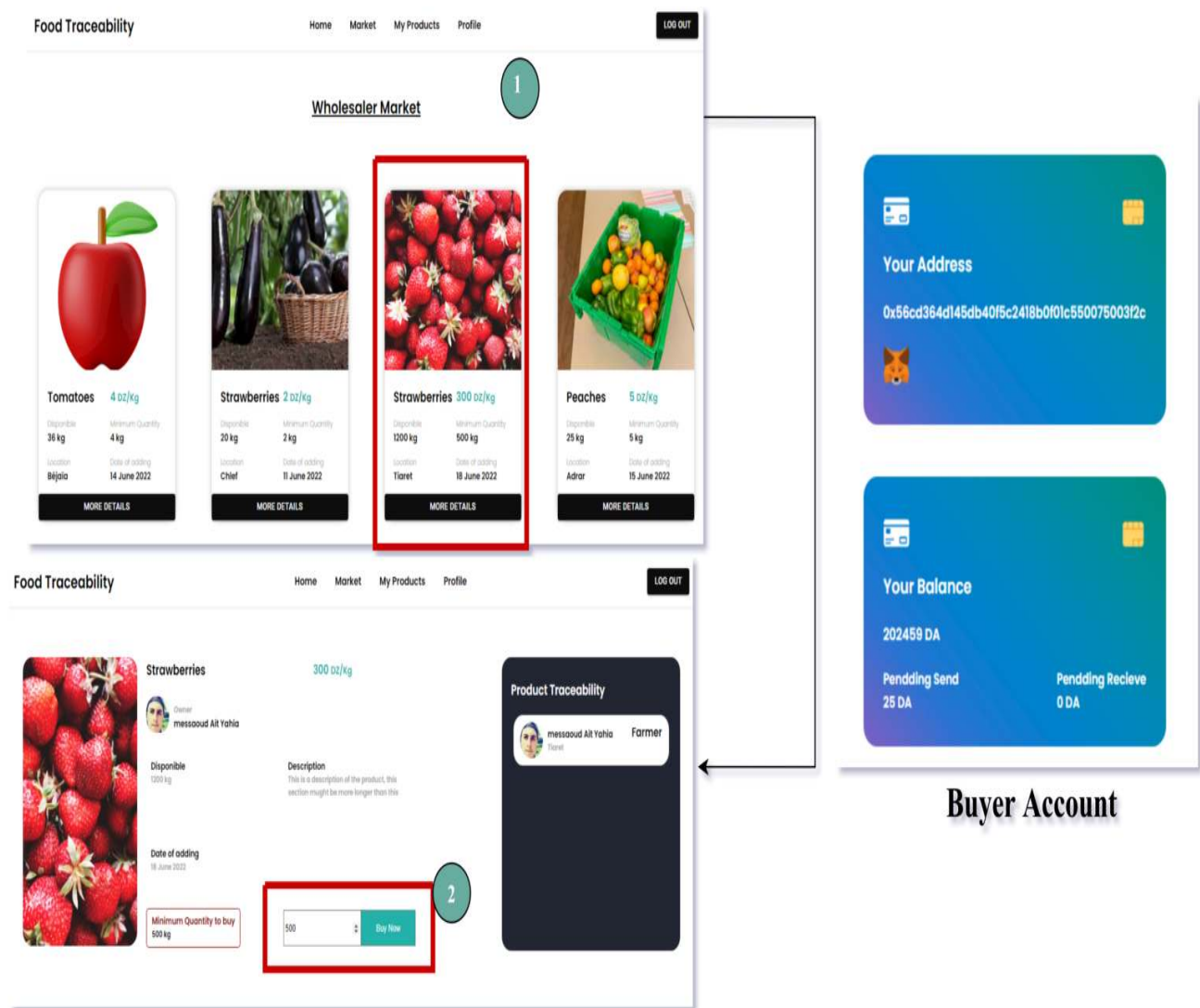


Figure 4.11: Buy A Product Interface Part1

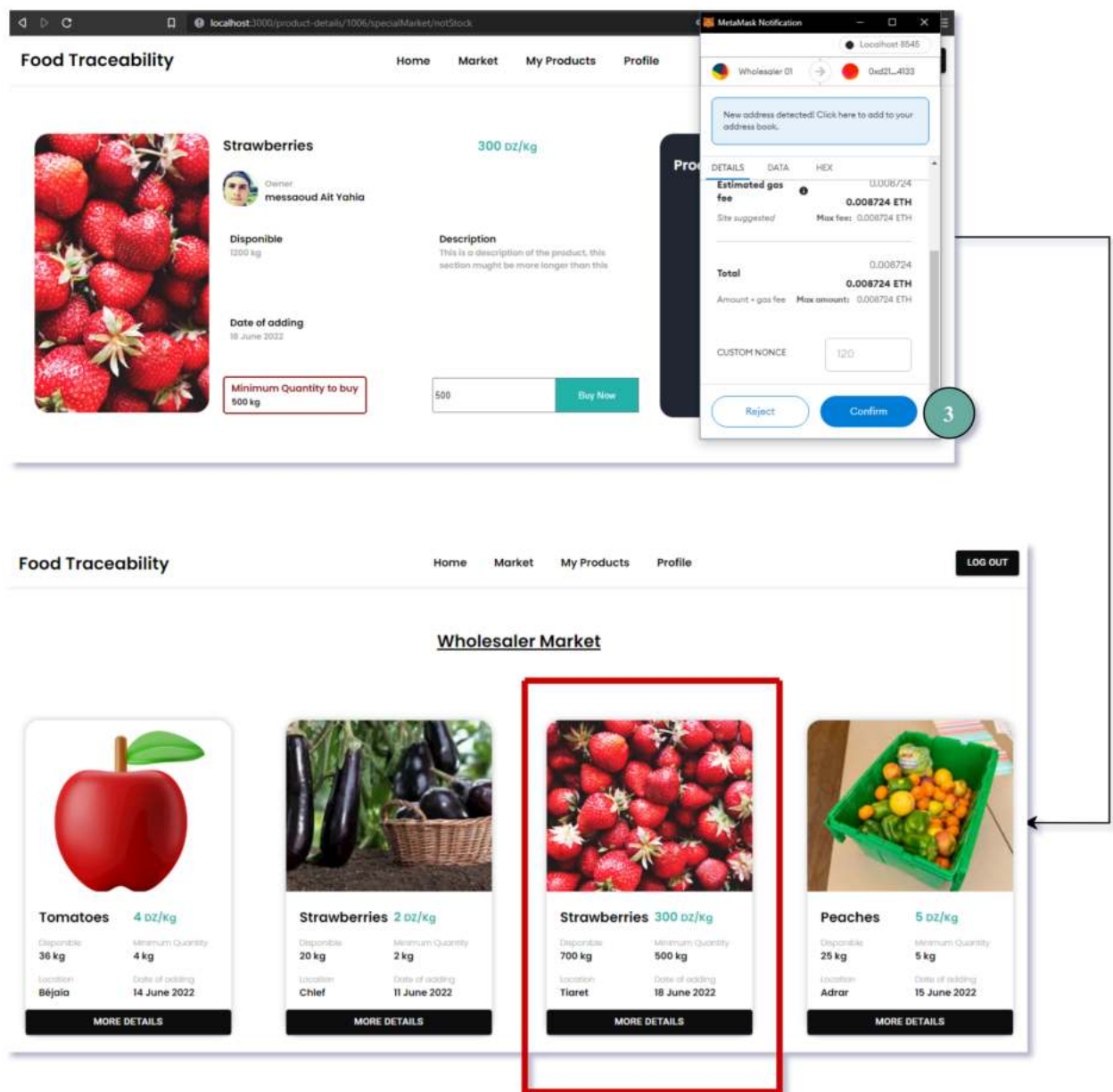


Figure 4.12: Buy A Product Interface Part2

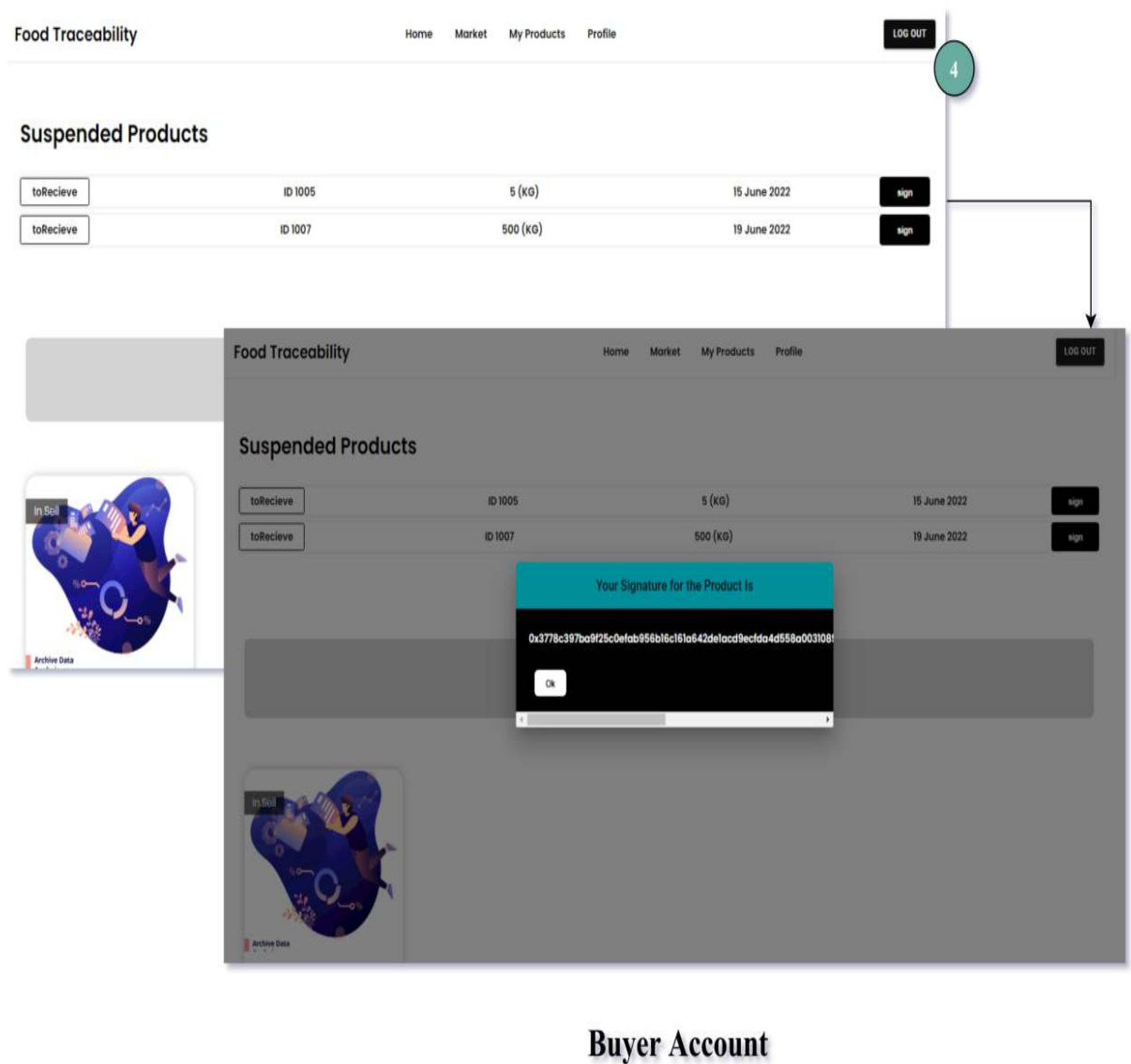
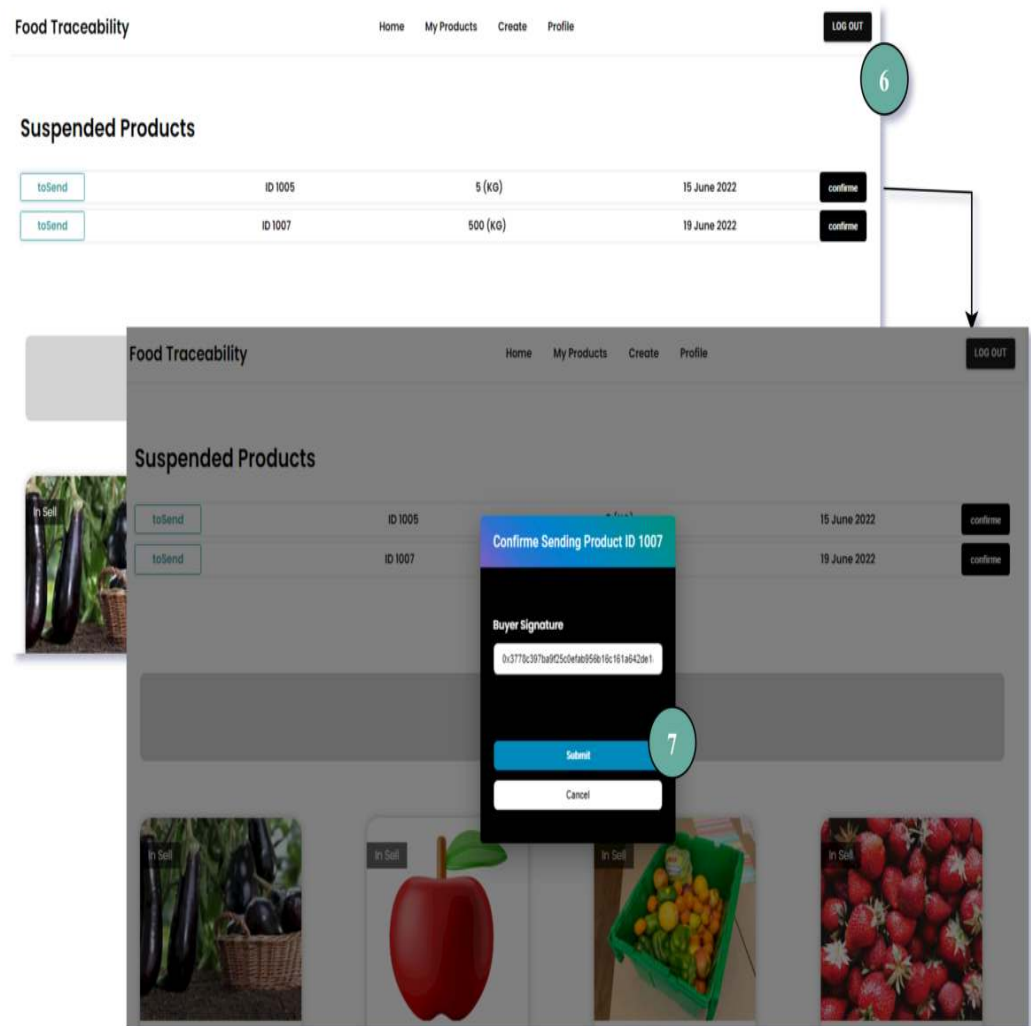


Figure 4.13: Sign The Product Interface



Saler Account

Figure 4.14: Confirme Signing The Product Interface



Figure 4.15: Balance After Buying Interface

4.2.2.4 Move Product From Stock To Sale

Figure 4.16 and 4.17 show the process of transferring a product from stock to sale. New information about the product is required in order to have the latest food quality status.

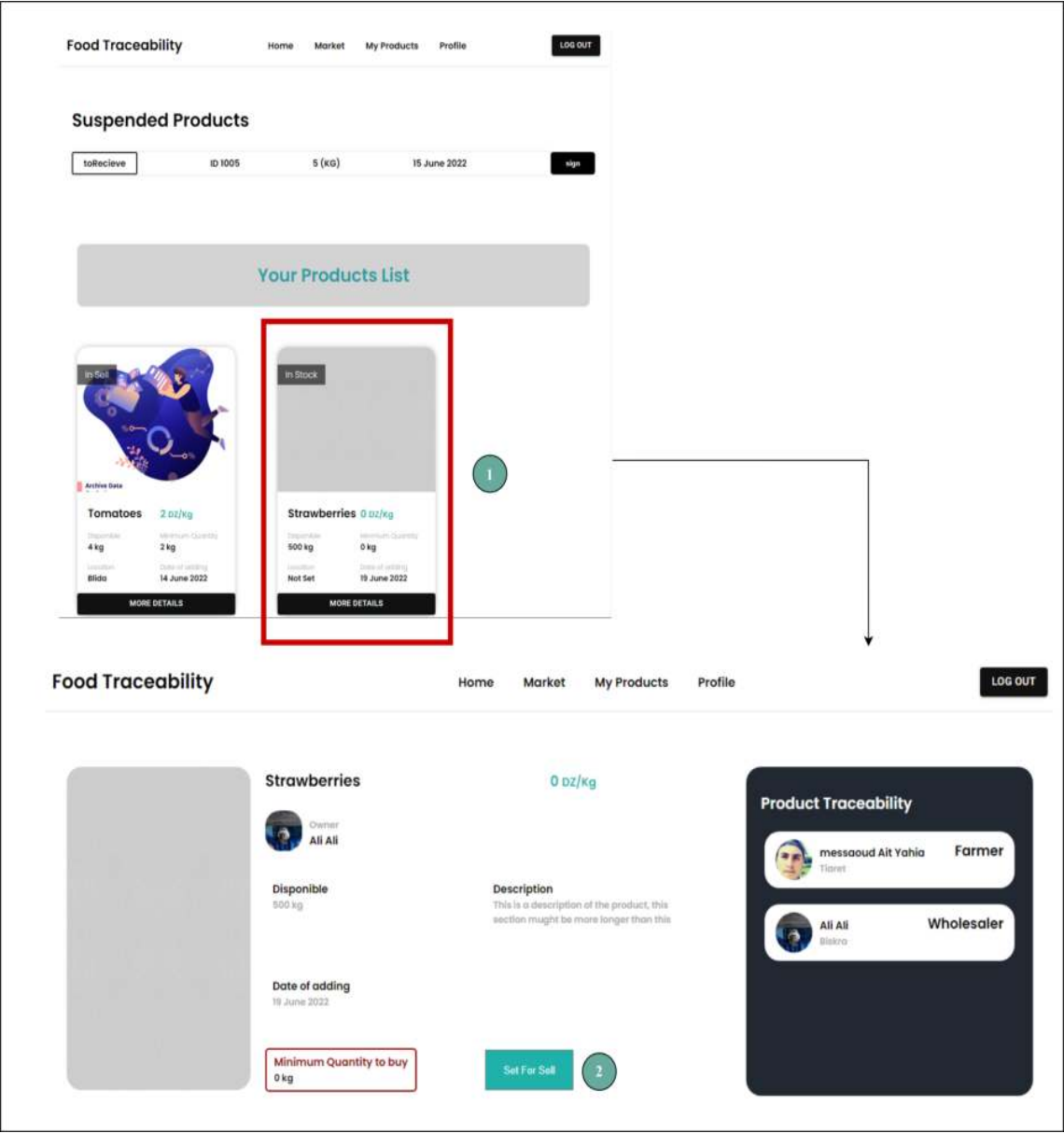


Figure 4.16: Transferring a product from stock to sale Interface Part1

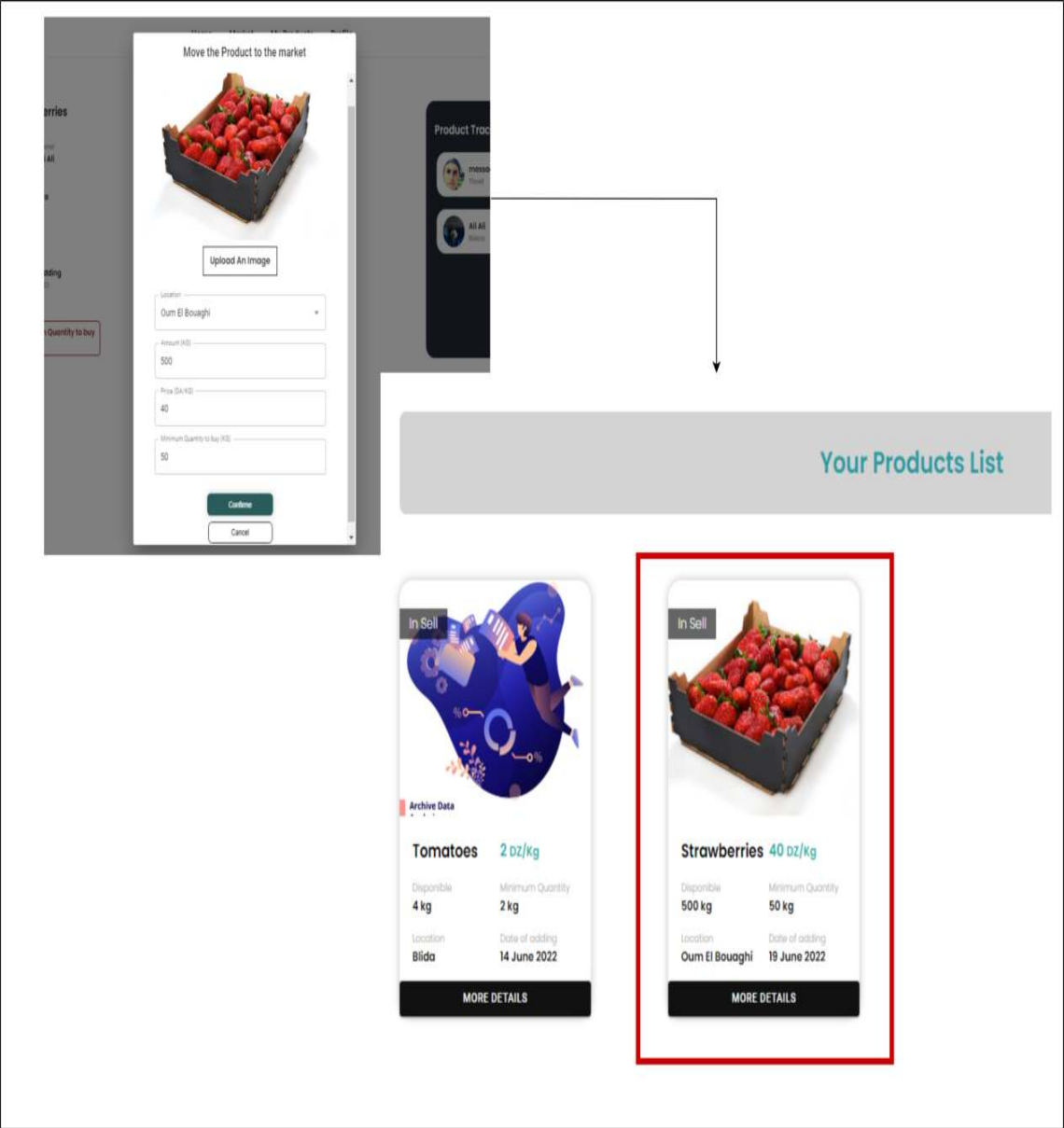


Figure 4.17: Transferring a product from stock to sale Interface Part2

4.2.2.5 Modify a Product

Figure 4.18 shows the process of modifying a product.

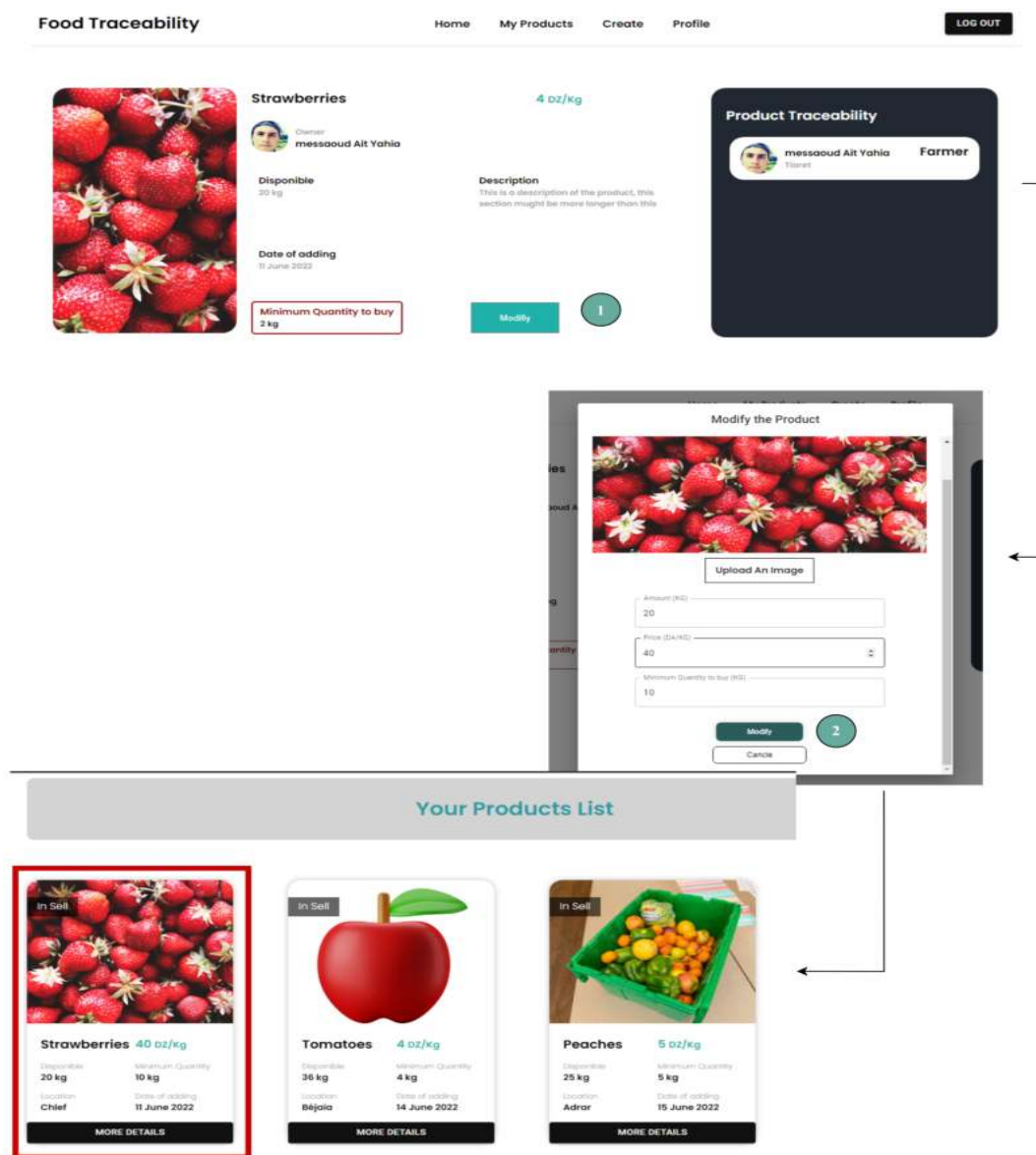


Figure 4.18: Modifying A Product Interface

4.2.3 Mobile Fonctionalities

4.2.3.1 Creation of account for farmer :

The image below 4.19 shows the following steps of creating an account for the farmer.

- Filling of the fields.
- Keep the private key safe.
- Email validation.

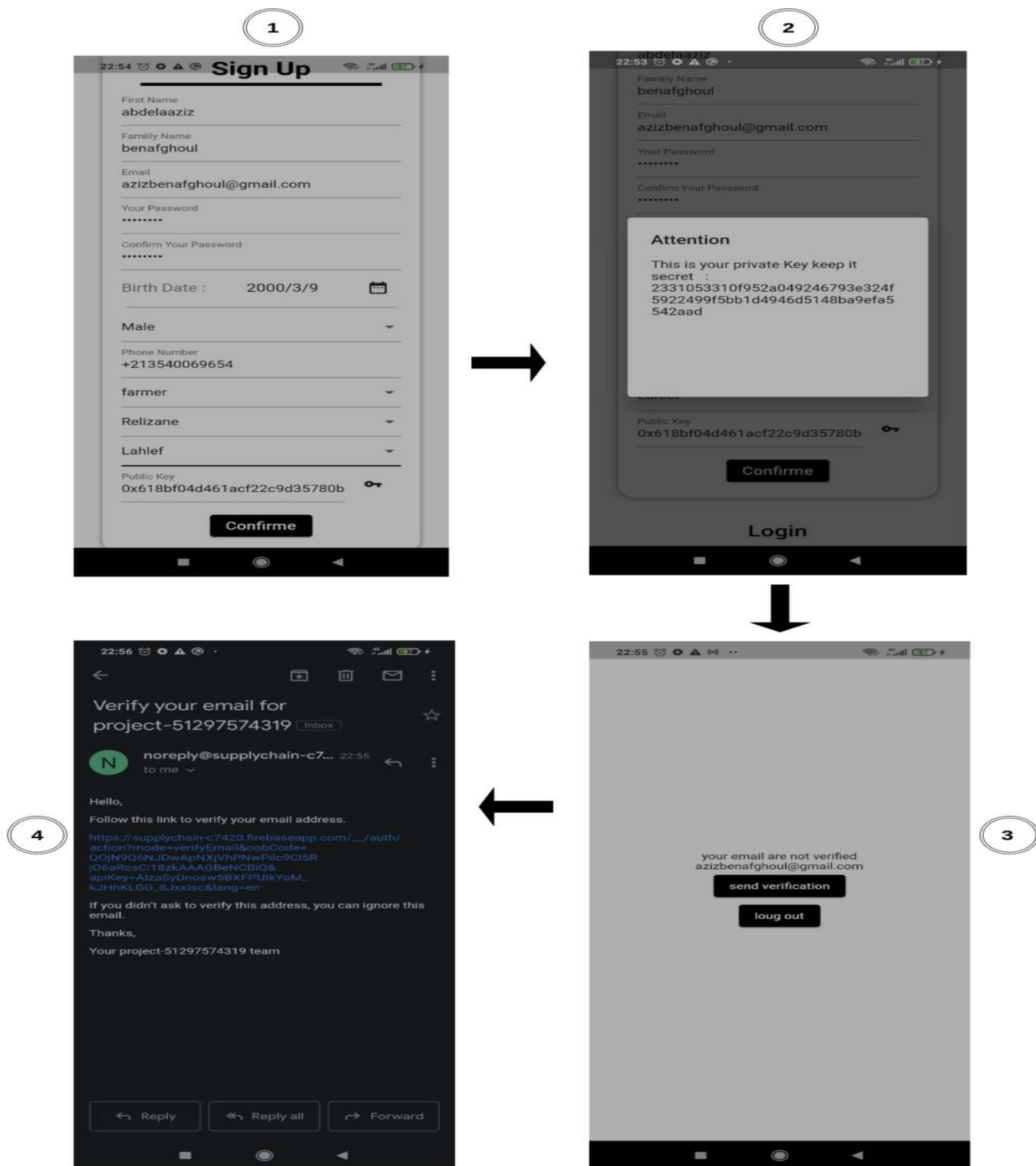


Figure 4.19: Farmer create an account

4.2.3.2 Adding Product For Farmer :

The image below 4.20 shows the following steps of adding product from farmer:

- The sign-in.
- Click to the button.
- Filling of the fields.
- Entering of private key.

Then the shop screen .

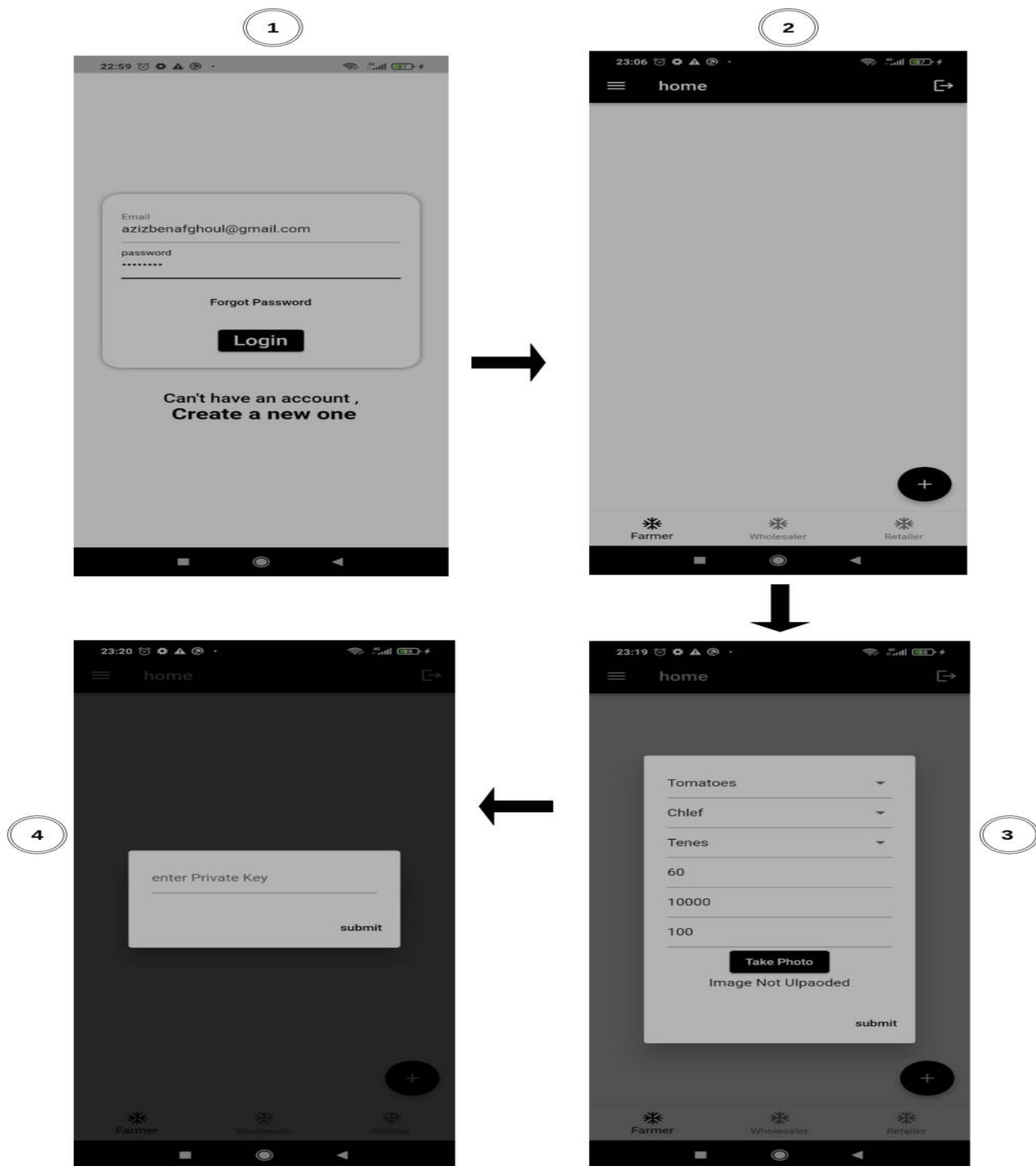


Figure 4.20: Adding product for Farmer

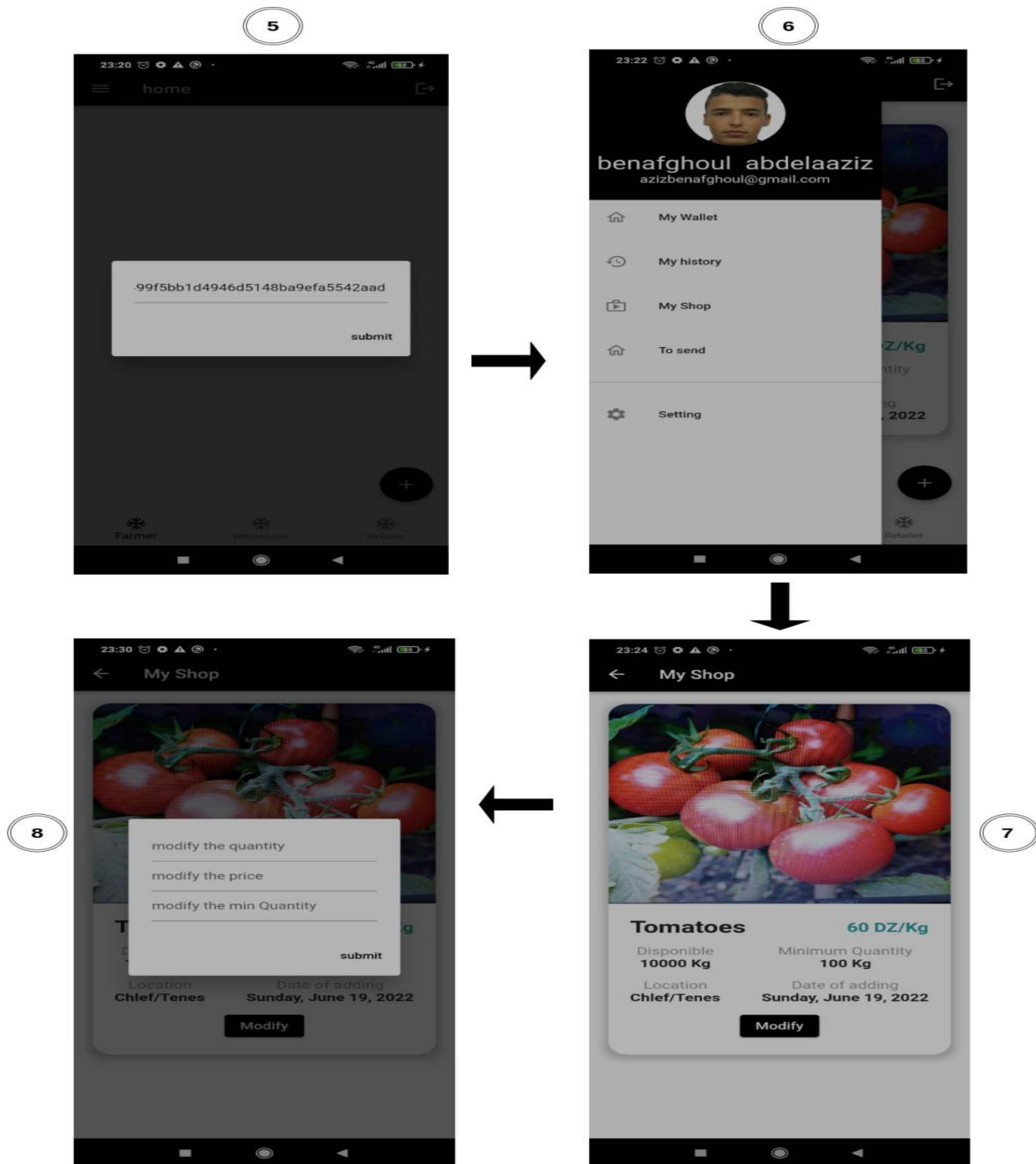


Figure 4.21: Farmer Add Product

4.2.3.3 Create An Account For Wholesaler:

The image below 4.22 shows the following steps of creating compte for the wholesaler

.

- Filling of the fields .
- Keeping the private key safe .
- Email validation .

and the balance after purchase 15000000 token.

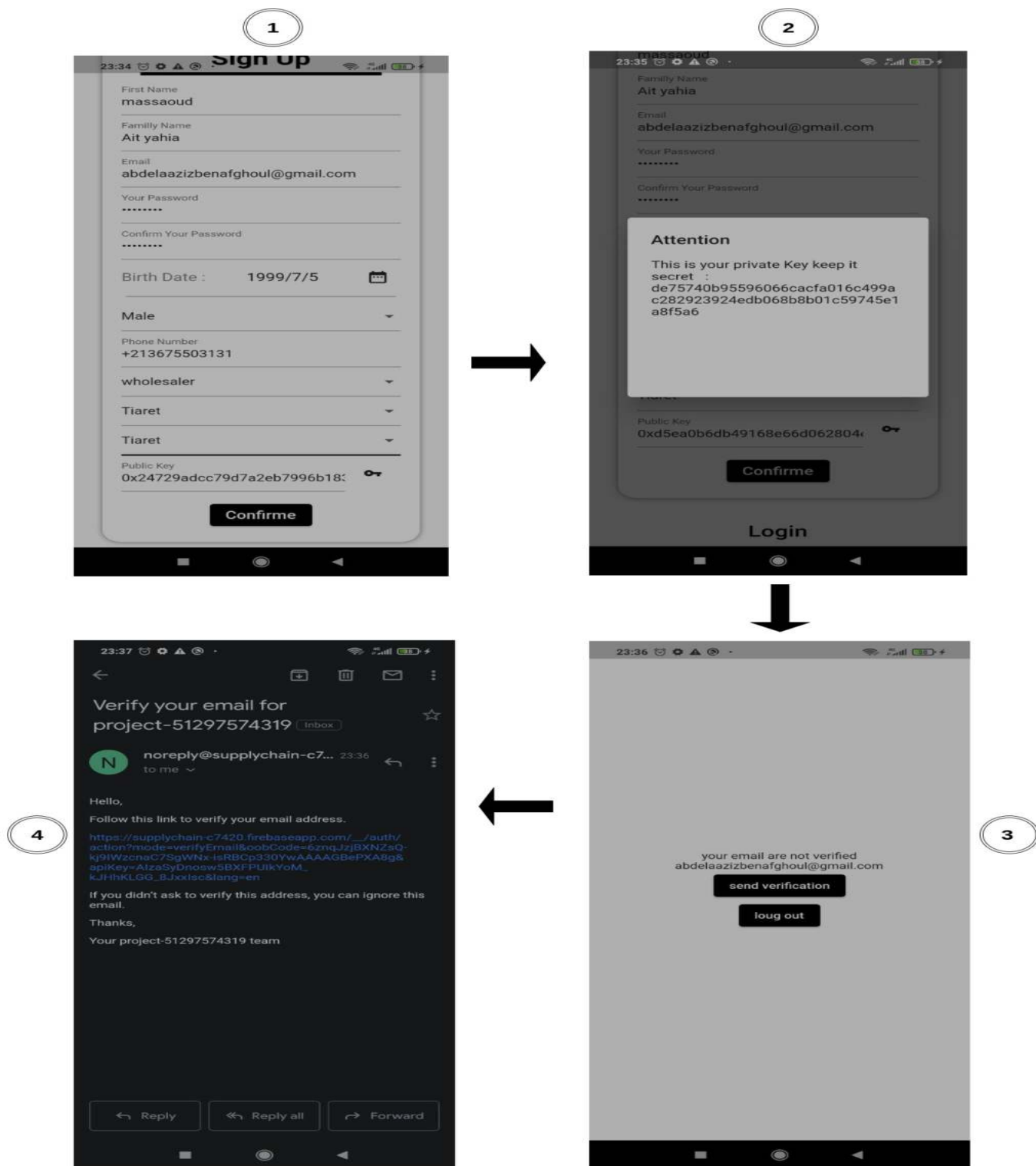


Figure 4.22: Account Creation for wholesaler.

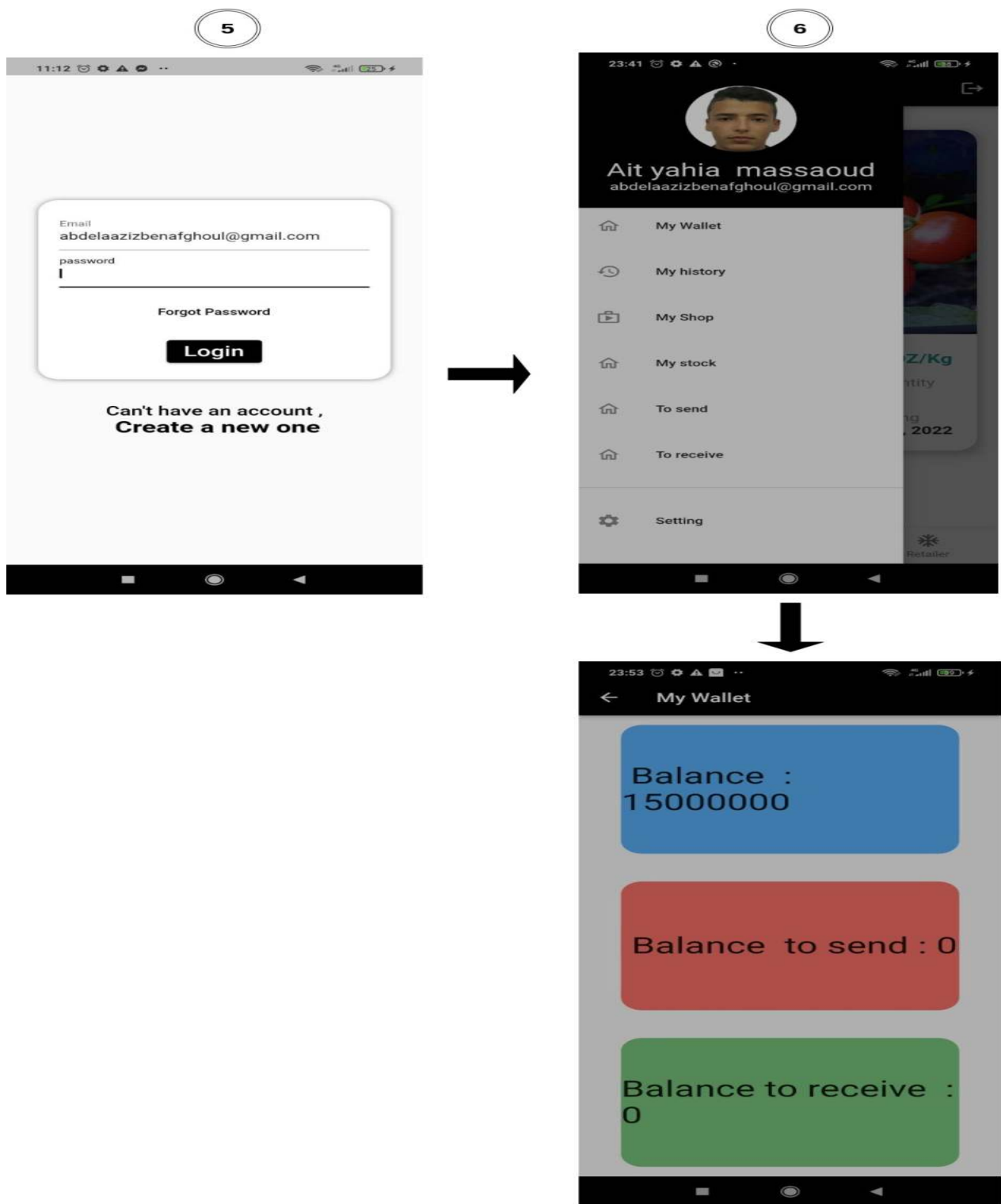


Figure 4.23: account creation for wholesaler.

4.2.3.4 Buy A Product For Wholesaler:

The image below 4.24 shows the following steps of buying product from the farmer

.

- select the product .
- click buy button .
- filling of the fields .

and the update of balance after purchase and the screen of product not received yet.

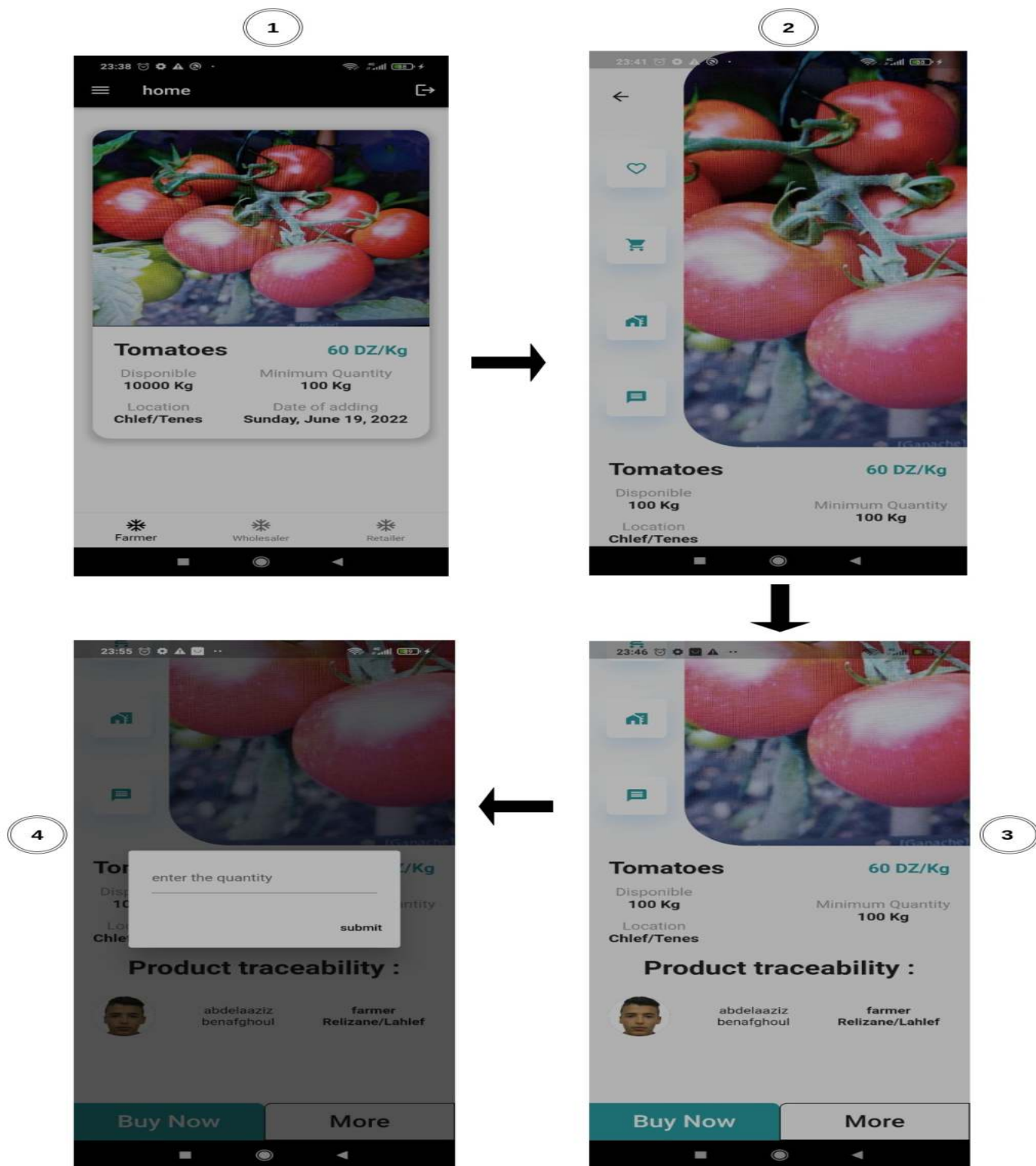


Figure 4.24: Account Creation for wholesaler

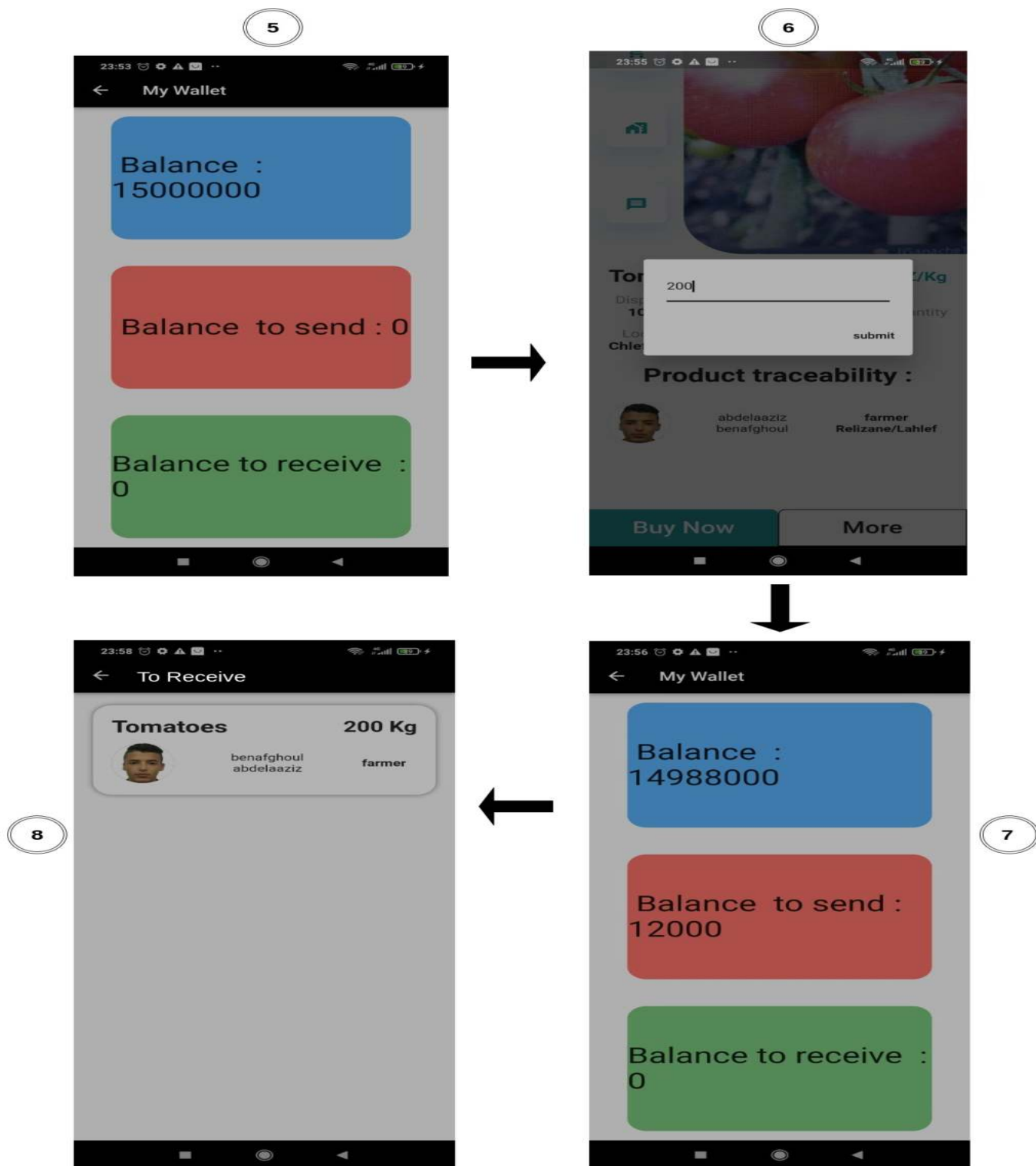


Figure 4.25: Mobile Account Creation for wholesaler

4.2.3.5 Confirme Receiving A Product For Wholesaler:

The image below 4.26 shows how the wholsaler confirm rceiving of the product by generating a signature in form of QR code.

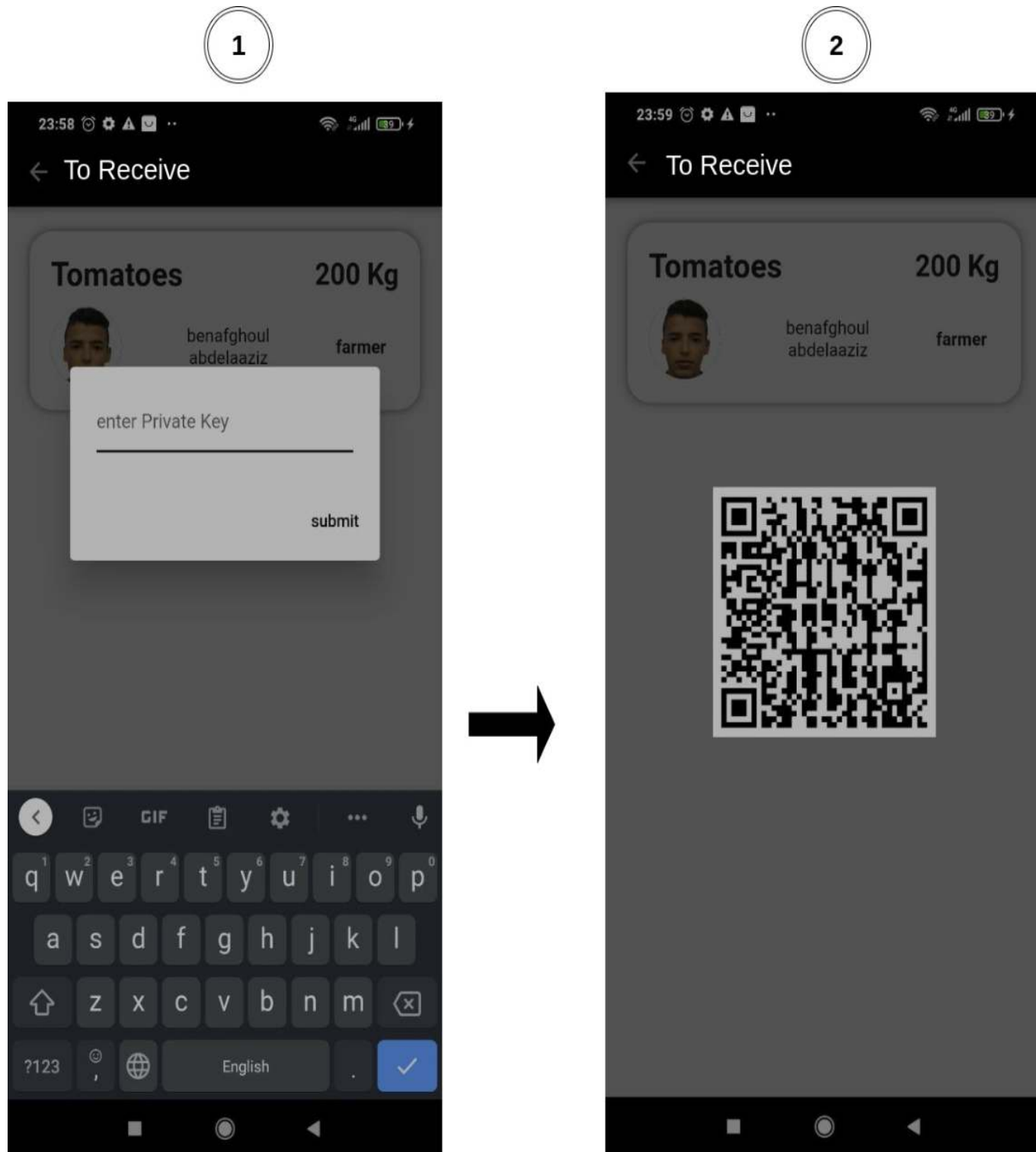


Figure 4.26: Confirme receiving of products.

4.2.3.6 Farmer Information After An Order :

The image below 4.27 shows the update of balance and product quantity after he receiving An order from wholesaler.



Figure 4.27: Farmer information after order.

4.2.3.7 Confirme Sending Of Product For The Farmer :

The image below 4.28 shows how the farmer confirm sending of the product by scanning a signature in form of QR code from the wholesaler.

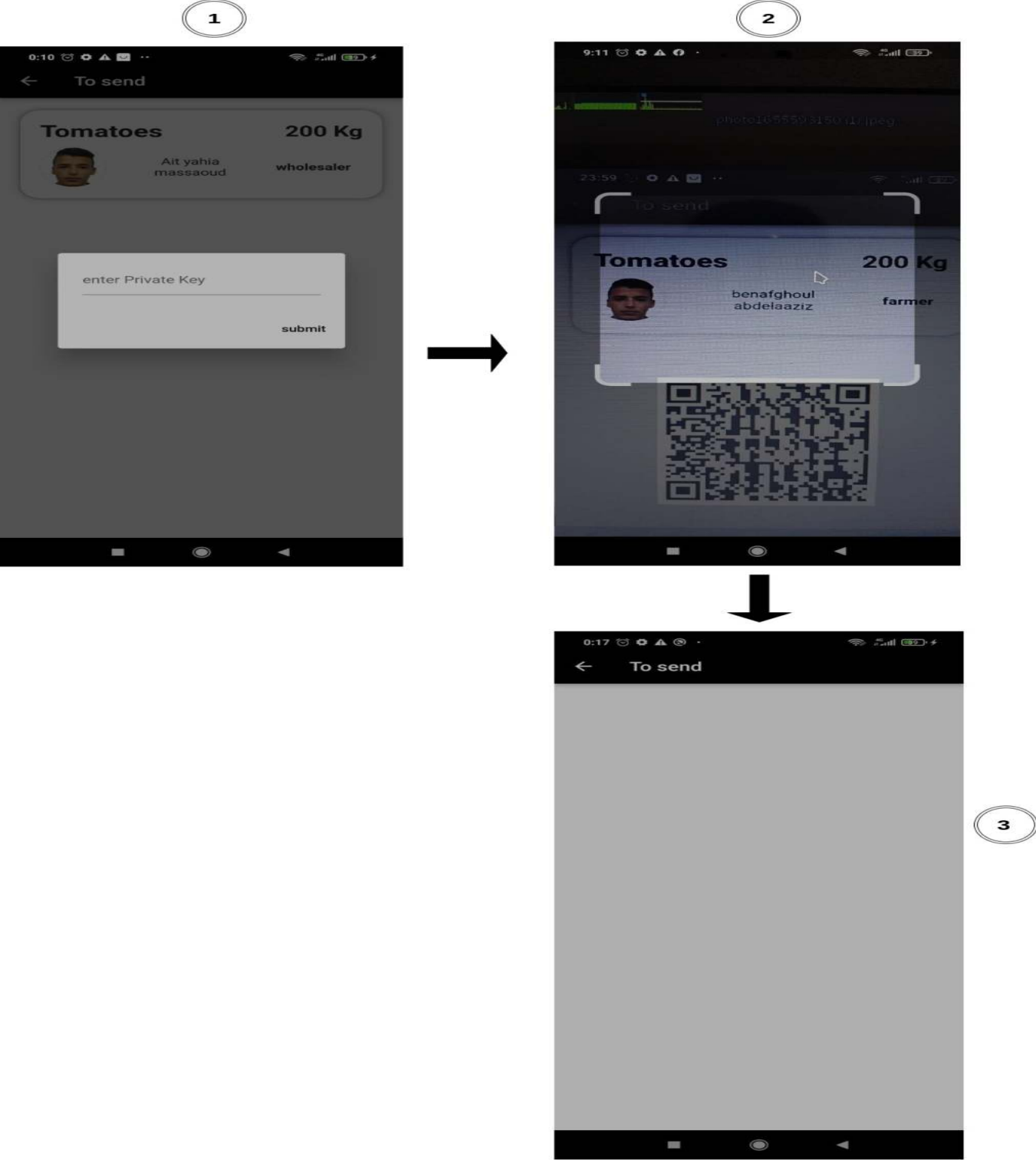


Figure 4.28: farmer confirm sending.

4.2.3.8 farmer Information After Confirm Sending Of Product:

the image below 4.29 shows the balance and history screens of farmer after confirm sending the product.

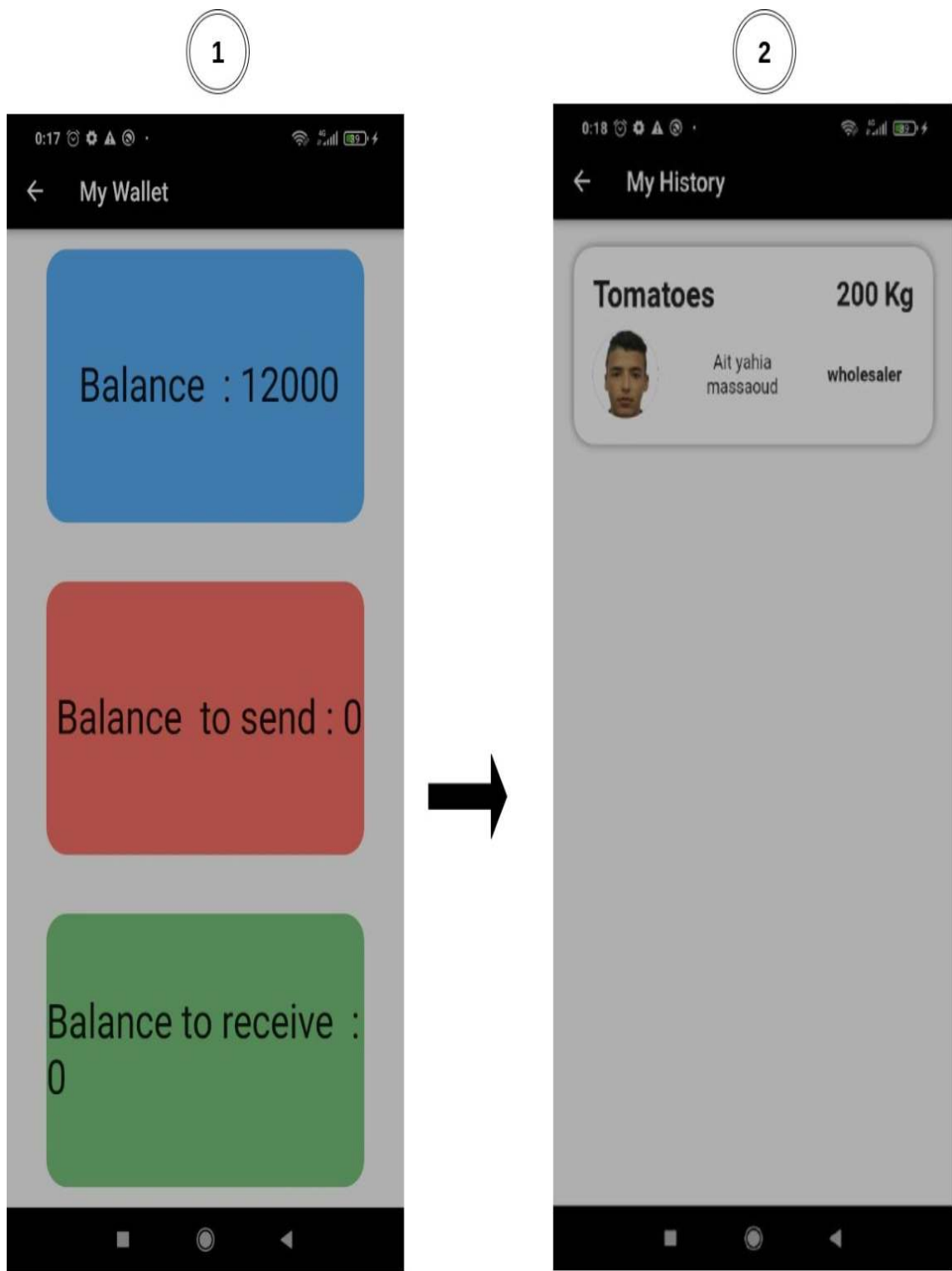


Figure 4.29: farmer information after confirm sending.

4.2.3.9 Wholesaler Information After Receiving Product :

the image below 4.30 shows the balance, history,stock ,and product want to receive screens after confirmation of receiving .

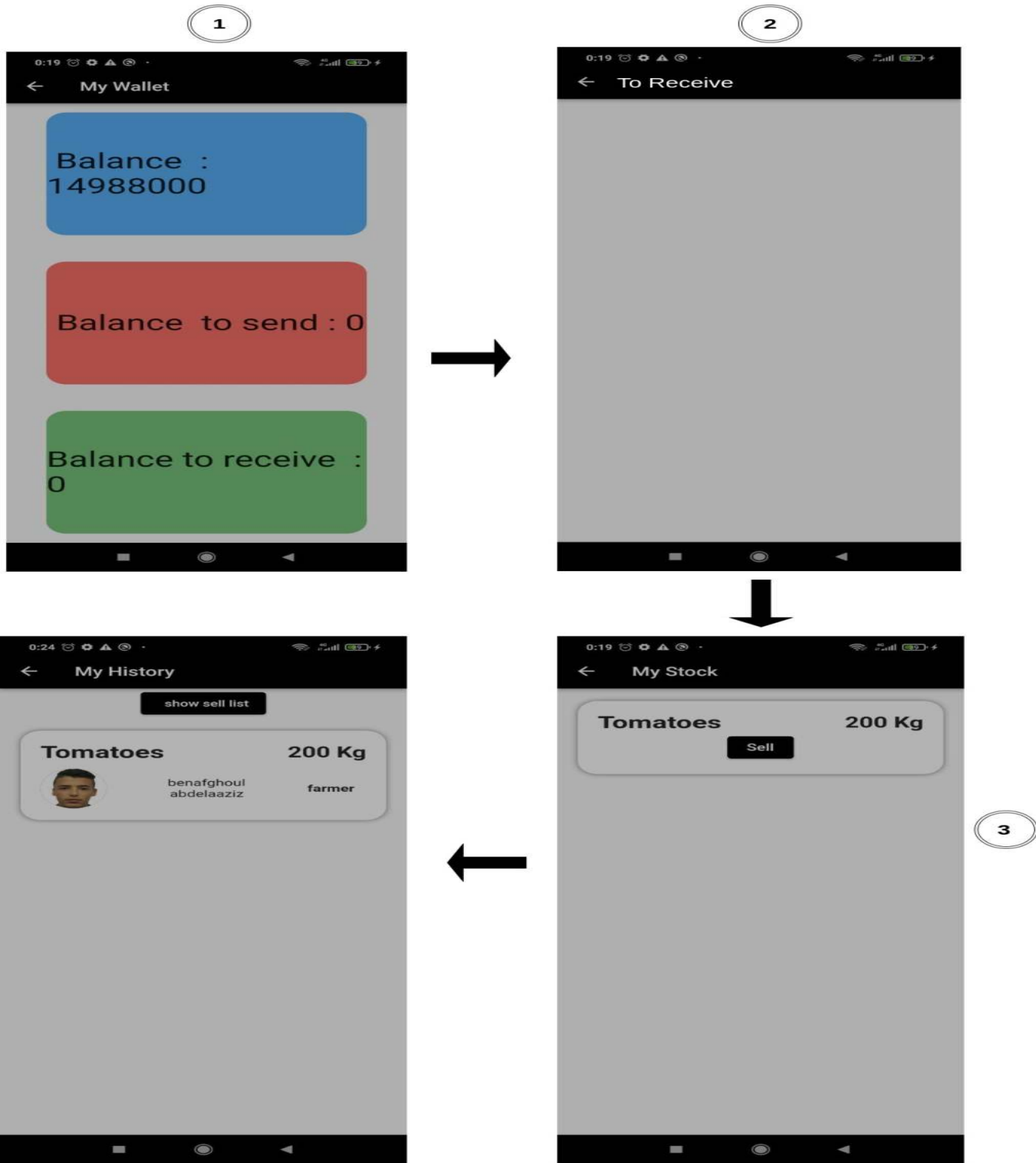


Figure 4.30: wholesaler information after receiving.

4.2.3.10 Add Product To Sell For Wholesaler :

The image below 4.31 shows how the wholesaler sell product of her stock.

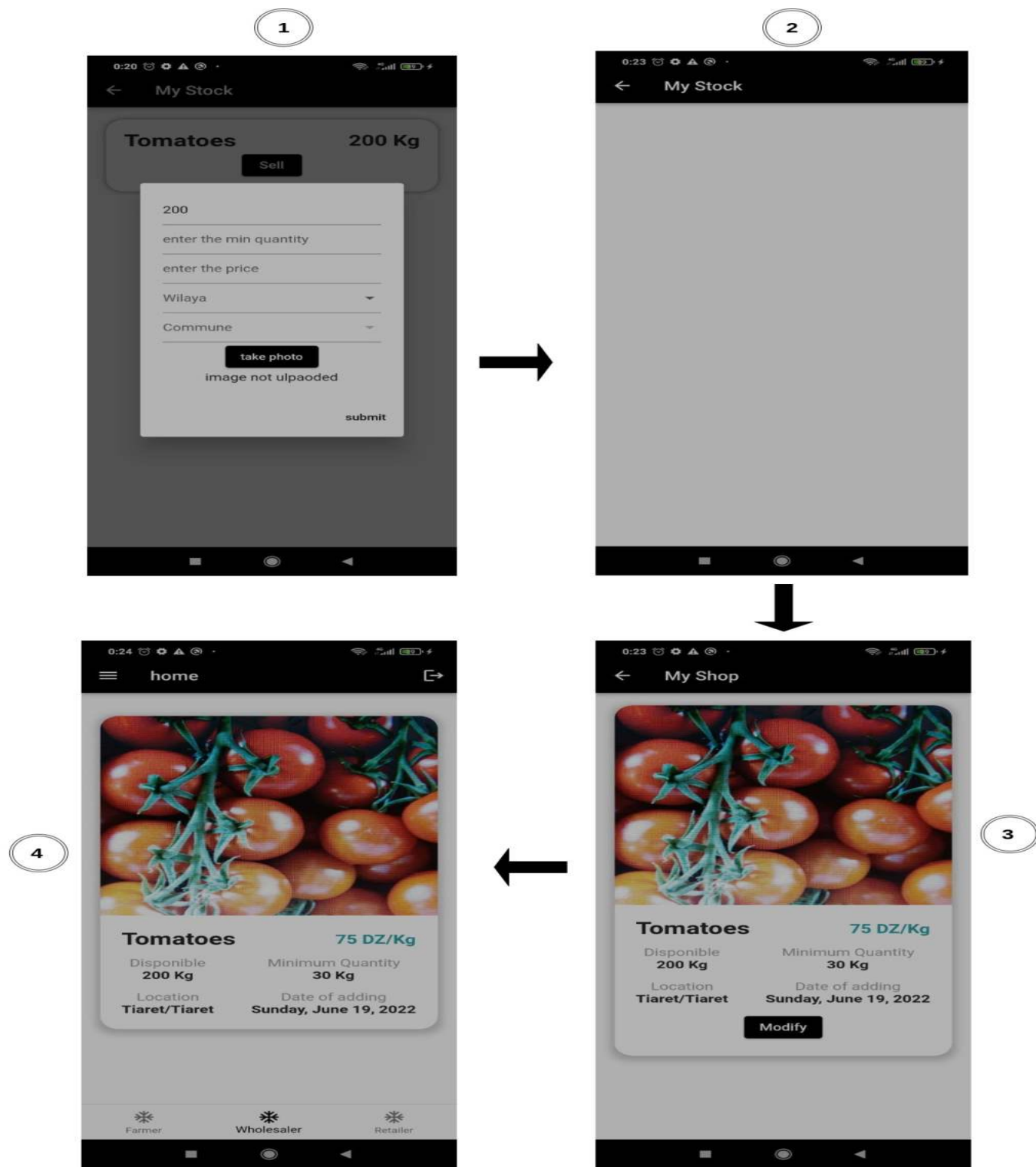


Figure 4.31: sell product wholesaler.

Conclusion

In this project, we proposed the design and implementation of a blockchain platform for food supply chain management. The objective of this platform is to integrate modern technologies into the food supply chain management sector to solve its problems. First, we discussed the basic concepts related to our topic, which are blockchain and traditional food supply chain management. Then, in the system design part, we presented different design diagrams in detail. Finally, in the implementation part, we presented the different tools and platform functionalities.

This project was an opportunity for us to discover blockchain technology, which is revolutionary, and to be among the first to engage in this path.

In the future work, we plan to improve the existing features and add more modules such as product delivery, chat application, google map location, trucking, and make the platform international with the ability to import and export products.

Bibliography

- [1] 9.3. Public key cryptography.
- [2] Blockchain - Merkle Tree.
- [3] Blockchain Demo.
- [4] Blockchain: Foundations and Use Cases.
- [5] What Is a Bitcoin Node? | River Learn - Bitcoin Technology. Section: River Learn - Bitcoin Technology.
- [6] What is Blockchain Technology | Blockchain Applications.
- [7] Top 31 Supply Chain Management Unicorn Companies in 2022t. <https://www.failory.com/startups/supply-chain-management-unicorns>, 2022. [Online; accessed 19-June-2022].
- [8] MRIDUL BHARDWAJ. What Are the Five Basic Components of a Supply Chain Management System? <https://www.iimu.ac.in/blog/what-are-the-five-basic-components-of-a-supply-chain-management-system/>, 2020. [Online; accessed 19-June-2022].
- [9] Ramamurthy Bina, Buffalo University, and The State University of New York. Blockchain basics <https://www.coursera.org/learn/blockchain-basics>.
- [10] Ramamurthy Bina, Buffalo University, and The State University of New York. Blockchain platforms <https://www.coursera.org/learn/blockchain-platforms>.
- [11] Ramamurthy Bina, Buffalo University, and The State University of New York. Decentralized applications <https://www.coursera.org/learn/decentralized-apps-on-blockchain>.

- [12] Lauren Christiansen. The Ultimate Benefits of Food Management. <https://altametrics.com/food-supply-chain/food-management.html>, 2020. [Online; accessed 19-June-2022].
- [13] Lauren Christiansen. A Guide to the Food Supply Chain. <https://altametrics.com/food-supply-chain.html>, 2021. [Online; accessed 19-June-2022].
- [14] ConsenSys. Ever Wonder How Merkle Trees Work?, May 2019.
- [15] Smith Corwin. Ethereum foundation <https://ethereum.org/en/developers/docs>.
- [16] JASON FERNANDO. Supply Chain Management (SCM). <https://www.investopedia.com/terms/s/scm.asp>, 2022. [Online; accessed 19-June-2022].
- [17] Rhiannon Garber. 4 Elements of Supply Chain Management. <https://altametrics.com/food-supply-chain.html>, 2021. [Online; accessed 19-June-2022].
- [18] WeTrustLeonD. Why Do I Need a Public and Private Key on the Blockchain?, February 2017.
- [19] Dylan Yaga, Peter Mell, Nik Roby, and Karen Scarfone. Blockchain technology overview. Technical Report NIST IR 8202, National Institute of Standards and Technology, Gaithersburg, MD, October 2018.