

Securing Ownership Management and Transfer of Consumer IoT Devices with Blockchain-Based Self Sovereign Identity (SSI)

by

Nazmus Sakib

19101609

Marzia Islam Mumu

19101516

Nuran Mubashshira Momo

23141105

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
June 2023

© 2023. Brac University
All rights reserved.

Declaration

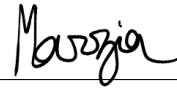
It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Nazmus Sakib
19101609



Marzia Islam Mumu
19101516



Nuran Mubashshira Momo
23141105

Approval

Securing Ownership Management and Transfer of Consumer IoT Devices with Blockchain-Based Self-Sovereign Identity (SSI)” submitted by

1. Nazmus Sakib (19101609)
2. Marzia Islam Mumu (19101516)
3. Nuran Mubashshira Momo (23141105)

Of Spring, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on June 11, 2023.

Examining Committee:

Supervisor:
(Member)

Md. Sadek Ferdous

Dr. Md. Sadek Ferdous
Associate Professor
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Chair)

Md. Golam Rabiul Alam, PhD
Professor

Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

With the power of automation and connectivity, the use of IoT devices is increasing over the last 10-12 years consistently. IoT devices are taking place in our homes beside industries with the capability of collecting datas from the environment, analyzing them, and sharing this datas with the motivation of the automotive world. During this time, IoT devices have shown us the difficulties of user identity management, ownership transfer and security flows beside the light of hope. Here, a better identity management system is the main condition for successful ownership management and transfer systems. Scientists have tried various types of identity management systems and security protections to overcome these issues but mostly failed to make IoT devices user-centric. They have tried to apply blockchain technology to IoT devices to ensure security and gain faith in IoT devices. The distributed ledger technology or blockchain has successfully provided us with a secure data storing decentralized database but failed to make the devices and identity data transaction user-centric. But blockchain still can be the solution here. With the help of blockchain, combined with the decentralized identifiers, varifiable credentials, and distributed ledger, a new identity management system called Self-Sovereign Identity (SSI) can win this race. SSI will help IoT devices to gain more trust of the user by making them user-centric with the help of SSI. Besides, it will be the solution for easier ownership management and ownership transfer of these devices.

Keywords: IoT; Identity Management Systems; Decentralized Identity Model; User-centric Identity Management Model; Self-Sovereign Identity(SSI); Verifiable Credetrial; Issuer; Holder; Verifier; Decentralized Identity Model; Blockchain.

Table of Contents

Declaration	i
Approval	ii
Abstract	iii
Table of Contents	iv
List of Figures	vi
List of Tables	1
1 Introduction	2
1.1 Research Problem	5
1.2 Research Objective	5
1.3 Report Structure	6
2 Background	7
2.1 Internet of Things Devices (IoT Devices)	7
2.2 Consumer IoT Device	7
2.3 Identity Management	8
2.3.1 Centralized Identity Model	9
2.3.2 Federated Identity Model	10
2.3.3 Decentralized Identity Model	11
2.3.4 Self-sovereign Identity	11
3 Literature Review	12
4 System Proposal and Threat Modeling	14
4.1 Threat modeling	14
5 Requirement Analysis	20
6 Architecture design	22
7 Use Case and Protocol Flow	24
8 Discussion	39
8.1 Analysing Requirements	39
8.2 Advantages Limitations	40
8.3 Future Work	41

9 Conclusion	42
Bibliography	43

List of Figures

1.1	Global IoT end-user spending worldwide 2017-2025	3
1.2	Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030, by use case	3
1.3	Global Consumer IoT Market Size	4
2.1	IoT Device A	8
2.2	Centralized Identity Model	9
2.3	Federated Identity Model	10
2.4	SSI Architecture	11
4.1	System proposal and work flow	14
6.1	Architecture design of ownership management system	22
7.1	Protocol flow of SSI connection establishment between two entities	25
7.2	Protocol flow of getting ownership from manufacturer	26
7.3	Protocol flow of transferring ownership to the new owner	27
7.4	Application design - manufacturer login	29
7.5	Application design - Manufacturer Action	29
7.6	Application design - Manufacturer's QR code to establish SSI connection	30
7.7	Application design - Owner's Login	30
7.8	Application design - Owner's Connection Establishment	31
7.9	Application design - Manufacturer's connection establishment notification	31
7.10	Application design - Alice information containing VC sending	32
7.11	Application design - New product registration	32
7.12	Application design - Ownership VC accepting for Alice	33
7.13	Application design - VC generation notification at Manufacturer's end	33
7.14	Application design - SSI Connection establishment between Alice and Bob	34
7.15	Application design - Manufacturer's transfer ownership action	34
7.16	Application design - SSI connection establishment between manufacturer and Bob	35
7.17	Application design - Connection establishment notification to Bob	35
7.18	Application design - Connection establishment notification to Manufacturer	36
7.19	Application design - Two VCs received from Alice sent to Manufacturer	36
7.20	Application design - VC verification at Manufacturer's end	37

7.21	Application design - Bob's identity contained VC sent to Manufacturer	37
7.22	Application design - New ownership credential received by Bob	38

List of Tables

7.1	First and Second Segment P-value (Bat First)	24
7.2	Web-SSI Protocols Part 1	25
7.3	Web-SSI Protocols Part 2	26
7.4	Web-SSI Protocols Part 3	28

Chapter 1

Introduction

For the time being, people have tried to make their life easier. Controlling life to make it easier is one of the common tendencies of humans from the very primitive time. With the advantage of technology, people are always trying to make their life easier by controlling them. IoT is the result of that. When people started to improve their life using technology, they had to interact with the computer or machines. IoT is the solution to remove this direct human-machine interaction. These devices collect data from the environment automatically with sensors, process them, and send them to the destination. This destination can be human or the other machines to set a proper communication or for the desired output. Though people can interact with these devices to set up, give instruction, or access the data, the main activities of these devices are completely automated.

IoT devices were initially being used for industrial purposes. IoT devices are helping businesses to monitor the real-time result of their systems, automate logistic operations and supply chain, automate the production process, reduce the production cost, and maintain customer interaction and faster delivery for decades. But it is a new trend to use IoT devices to make home life easier too. In the consumer sector, IoT devices, such as smartphones or computer-controlled lighting systems, and automated kitchen equipment are making home life easier. Besides these devices are monitoring temperature, and ring automatic danger alarms for human safety[1].

Technology helps people to lead a smart life, control life smoothly, and finally, make life easier. The interest of people in that technology will naturally increase with time. From the current stats of market demand and market capital, we can see a clear picture of that.

According to the Global IoT end-user spending worldwide 2017-2025, the total market size was 110 billion in 2017 and it will reach 800 billion dollars in 2023 and it will be 1567 billion in 2025 (Figure 1.1). So, it is clear that people are being interested in IoT devices day by day and spending money on them. IoT devices are being successful in making their life easier than before. People are also going towards automation [2].

According to the number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030, by use case”, the number of total connected IoT devices will be 25444.5 million in 2030 which was only 7741 million in 2019 and currently 13146.3 million in 2023 (Figure 1.2). It means that IoT devices are increasing and being connected with the network, they are serving people [3].

According to Global Consumer IOT Market Size By Type, By Application, By Geo-

graphic Scope, And Forecast, the consumer IoT market will be 153.80 billion dollars in 2028 which will be 44.46 billion dollars in 2020 (Figure 1.3). It clearly illustrates that the demand for IoT devices at the consumer level is also increasing. People are thinking about smart homes, smart kitchens, and maybe a smart device for entertainment for a smart and automated home lifestyle [4].

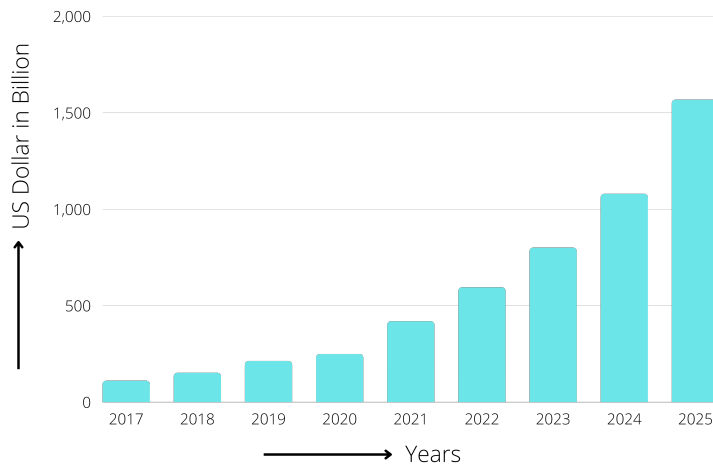


Figure 1.1: Global IoT end-user spending worldwide 2017-2025

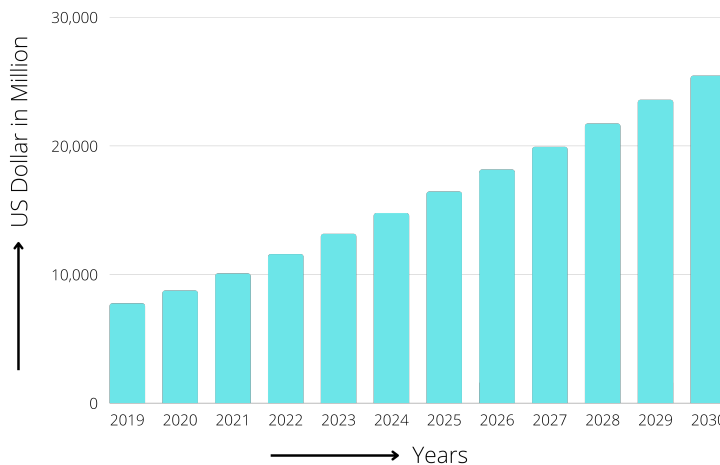


Figure 1.2: Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030, by use case

When the number of devices and users increases, people change their devices frequently. In this scenario, we have to think about the ownership transfer and ownership management of IoT devices. When a man/woman buys a device for him/her, that device has been assigned with the name of the buyer. The manufacturer’s authority provides the power of access and controls of the device to that owner. Here the new owner can access their device with their identity credentials and the family members also can access those devices. This time, the device has to manage the

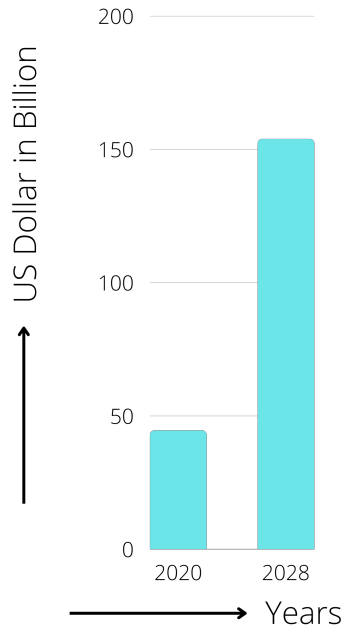


Figure 1.3: Global Consumer IoT Market Size

identity of the users, and also users have to provide their identity credentials. This is called the identity management of the IoT device. It is directly connected with ownership management because when we transfer our ownership to another person, we have to exchange the owner’s power of action for that device. The new user has to have access to that device and has to be authorized to control the device. He/she also has to be able to redeem the warranty and guarantee of that device from the manufacturer.

To perform this action, we have to follow the model of identity management. At the consumer level, in most cases, we follow the first model of identity management which is the Silo model [5]. In this model, each manufacturer company issues a digital identity credential to the users to allow them to access the device. Each time when a user is associated with a new device, he or she has to maintain a full new identity credential to access that device. In this way, if we want to transfer ownership to the new users, we have to transfer old users’ identity credentials which is a direct break of security and privacy. Besides, managing a lot of username and password is also a hassle. Along with this, hacking user IDs and passwords is the easiest way to hack a device. The second model of digital identity is called the Federated one. Because of the poor user experience of the first model, third parties began issuing digital identity credentials that allow users to log in to services and other websites. The best examples of this are “Login with Facebook” and “Login with Google” functionalities. Companies “outsourced” their identity management to major corporations that have an economic interest in amassing such large databases of personal data. This, of course, raises privacy and security concerns.

It has been cleared that none of the centralized models can ensure the 100 percent-age security of our identity. At that time the emergence of blockchain technology started to give new hope. The hyper ledger technology which is blockchain can give us a decentralized database to store user information or identity credentials. So the database becomes unhackable. With the benefit of this technology and combined with the decentralized identifiers and verifiable credentials, a new identity man-

agement system has been developed with Self Sovereign identity. It is completely user-centric. People can hold their identity in their identity wallet and can decide which info they want to share to verify them.

1.1 Research Problem

For a better identity management system, [6] IoT devices are facing challenges like 1) Credential Abuse, 2) Default Password Risks, 3) Virtual Eavesdropping, and 4) Ownership Management. As traditionally consumer IoT electronic devices come with the Silo identity management model, these are taking place. A centralized identity management model can always be the jackpot for hackers to hack users' information from it. Besides, the Silo and Federated identity management model cannot get rid of us from usernames and passwords. We have to rely on the organization directly or on the third-party identity management service provider to log in to our device. So, logically our information is still on another hand.

Besides people exchanging or selling their IoT devices to another person, it is hard to transfer the access control mechanism and the full authorization power. If we want to follow the Silo model here, the new user just has to create a new account again with the username and password and the old user has to give him the initial access and adminship. But still, the old user can access the device as an owner later which will hamper the privacy of the new user. And also, when the new user will get the adminship, he/she may get the old user information from the device.

As it will be a password-based system, the new user may have to change his/her password later. The manufacturer company may restrict him/her from the full control of the device as he/she is not listed as the owner in their system yet. Again, the new user may have to redeem the warranty and guarantee service from the company if he/she faces any issues with the device. The manufacturer company may deny giving him/her service as they do not know the new user yet. So, things are being complicated and the devices which are made to make the world automated are going through a manual system.

1.2 Research Objective

This research aims to solve the complicity of ownership management and ownership transfer of IoT devices by integrating with a new identity management model. This new identity management model which is SSI (Self-sovereign Identity), is a decentralized identity management model that will provide IoT devices with a decentralized safe, and secure database and help the devices to communicate and securely share data with decentralized identifiers and verifiable credentials. So the motivation for this research is:

1. Understanding the security issues of consumer IoT devices, particularly during the transfer of ownership of such devices.
2. Understanding the existing models of identity management of IoT devices.
3. Understanding the Self-sovereign Identity Model

4. Designing architecture with SSI following a rigorous set model and requirement analysis.
5. Development and evaluation of the proposed architecture.

1.3 Report Structure

In Chapter One, we have described our research problem and research objective. We have tried to elaborate that why the existing technology is not enough to ensure user-centric identity management and how we can attain that with the help of SSI. We have tried to elaborate on the background of our research in Chapter Two. We have presented the structure and working methodology of IoT devices, the demand for consumer IoT devices with statistics, and most importantly the previous identity models. Then, in Chapter Three, we have come up with some other research on this domain of ownership transfer in the digital world. We have tried to find out the problems of that articles and come up with a solution. As we are going to build a new system that will help to transfer ownership among people, we have discussed the threat model in Chapter Four to be prepared for future obstacles. Accordingly in Chapter Five, we have illustrated the requirements of our system in two different parts which are functional requirements and non-functional requirements. In Chapter Six, we have given an idea about the entities and their internal interaction through an architectural design. We have drawn a sequence diagram of every action and interaction among the entities with protocols and cryptographic notations in Chapter Seven. Finally, we conclude in Chapter Eight and also tried to give an idea of our future works.

Chapter 2

Background

To implement SSI on the ownership management of IoT devices, we have to learn from books, research papers, and also articles to understand every single detail of the factors which are working in the background. We have to learn about the IoT devices and how it works, the identity management system and its importance, the types of identity management systems, flaws of old identity management systems, the relationship between identity management and ownership management, blockchain, and SSI.

2.1 Internet of Things Devices (IoT Devices)

IoT devices are the devices that are made to automate our work. Devices with this technology can work automatically by themselves. There is no need for human-human interaction or human-machine interaction to run these devices. These devices can collect data automatically from the environment with the sensor and can process and send this data to the destination to achieve the goal. An IoT ecosystem is made up of smart devices that can connect to the internet and use embedded systems like processors, sensors, and communication hardware to collect, send, and act on data they get from their environments. IoT devices share the sensor data they collect by connecting to an IoT gateway or another edge device. From there, the data is either sent to the cloud to be analyzed or is analyzed locally. Sometimes, these devices talk to each other and act based on the information they get from each other. Most of the work is done by the devices without human help, but people can set them up, give them instructions, or access the data. The way these web-enabled devices connect, network, and talk to each other depends a lot on the IoT applications that are being used. IoT can also use artificial intelligence (AI) and machine learning to make it easier and more dynamic to collect data [1](Figure 2.1)

2.2 Consumer IoT Device

The internet of things has numerous real-world applications, ranging from consumer IoT and enterprise IoT to manufacturing and industrial IoT(IIoT). IoT applications span numerous industries, such as automotive, telecommunications, and energy. In the consumer market, smart homes equipped with smart thermostats, smart appliances, and connected heating, lighting, and electronic devices can be remotely

controlled via computers and smartphones. Wearable devices equipped with sensors and software can collect and analyze user data, sending messages to other technologies about the users in an effort to make their lives more convenient and comfortable. Wearable devices are also used for public safety, for instance, to improve the response times of first responders during emergencies by providing optimized routes to a location or by monitoring construction workers' or firefighters' vital signs at life-threatening sites. IoT offers numerous advantages in healthcare, including the ability

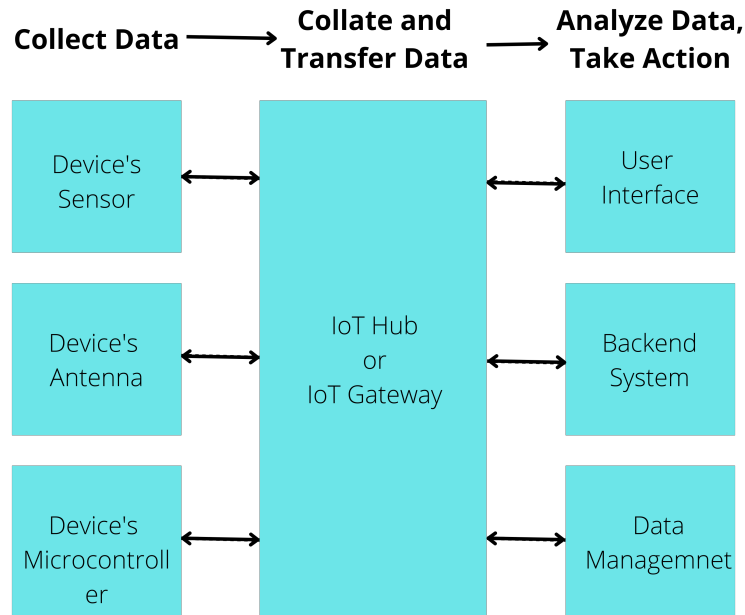


Figure 2.1: IoT Device A

to monitor patients more closely by analyzing generated data. Hospitals frequently employ IoT systems for tasks such as inventory management of pharmaceuticals and medical instruments. Using sensors to detect the number of occupants in a room, intelligent buildings can reduce energy costs, for example. The temperature can be adjusted automatically, for instance by turning on the air conditioner when sensors detect a full conference room or turning down the heat when everyone has left for the day. Using connected sensors, IoT-based smart farming systems can monitor, for example, the light, temperature, humidity, and soil moisture of crop fields. IoT also contributes to the automation of irrigation systems. IoT sensors and deployments, such as smart streetlights and smart meters, can reduce traffic, conserve energy, monitor and address environmental concerns, and enhance the sanitation in a smart city [1].

2.3 Identity Management

Online identity is the information to verify ourselves to the system. It helps us to authenticate by the system and allows us to access the system to use data or take the service. We have to verify our identity by providing a user id and password or the digital token to claim access to the system and to have the authorized power. The system has to manage users' data, assign and manage roles, and verify their access to run the system. This is called identity management. It was planned to

create a security layer in our digital communication. It helps to know the identity of our connected users with which we are sharing or exchanging our data over the internet.[7]

2.3.1 Centralized Identity Model

This one is also known as the password-based identity model. We are using this one in our daily life to verify our identity to access any online account. In this model, the service provider is the identity provider itself. This means, on which platform we are going to open an account, they are the holder of our identity and also providers when we need that. When we open an account on that platform, we give all of our information to verify our identity. That platform receives all the information regarding our identity from us and provides us a user id and password instead of that. Whenever we access our account, we have to provide that user id and password to verify ourselves [8](figure 2.2) . But This model has a lot of drawbacks, which

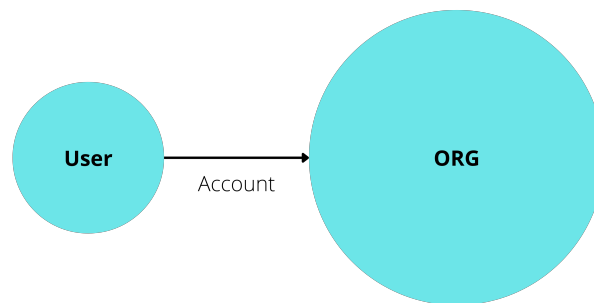


Figure 2.2: Centralized Identity Model

are:

1. Users can still provide wrong information, and fake identities to open an account as there is no way to verify the identity papers from the origin authorities for the company.
2. The users have to give their identity-related papers whenever they go to open an account on various platforms.
3. The user's private information is going to the service providers. It is not being secret.
4. Users have to remember a lot of user IDs and passwords to access multiple accounts.
5. The identity data is not portable as using different types of passwords is recommended to use in different sites.
6. The service providers have to maintain a large database to store the information of the users.
7. The databases are the honeypots for hackers and hacking this database can publicize the private data of a lot of users.

By keeping the problem in hand, people started to find solutions. And then they had found the federated identity model. Though this model is the better version of the centralized identity model but still has failed to solve all the problems. Is also a centralized identity model with a new and common identity service provided.

2.3.2 Federated Identity Model

This model uses an identity service provider in the middle of the user and the various internet-based service providers and helps both. The identity service provider is a platform where we give all our identity-related papers to verify ourselves like in the centralized identity model. But this time the identity service provider is not the actual internet-based various service provider itself. It is a different entity. And this time we can use this entity whenever we open an account across multiple service providers. That means that we don't have to provide our identity-related information and papers every time we open an account. So our data is being portable this time. Based on this idea, later on, we have seen SAML, OpenID, and OAuth get real success in the industry. Till today we are signing up to other service providers' accounts with Facebook, Google, etc [8](Figure 2.3). (Figure 2.3) But still here are

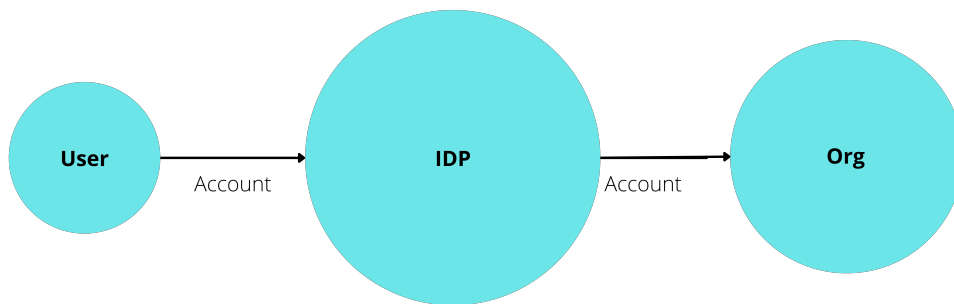


Figure 2.3: Federated Identity Model

some problems:

1. Users can still provide wrong information, and fake identities to open an account as there is no way to verify the identity papers from the origin authorities for the company.
2. Though users don't have to share their information every time to open an account, they have to share their information at least one time with the identity service provider.
3. Though users don't have to remember a lot of passwords, there is still at least one user ID and a password for the identity provider. But all internet service providers do not use the same identity service provider. That is why the ID and password can still increase.
4. Though there are no separate information databases for all the companies, at least a single large database for the identity service provider. And this database can also be hacked and can give more data than the centralized model to bad hands.

2.3.3 Decentralized Identity Model

To solve all the problems, based on cryptographic keys, a federated identity model has been developed. Though it comes with a hybrid approach where connections are peer to peer, the FIDO alliance performs key management centrally. But it was fundamentally decentralized. The most significant difference of this model is it is not account-based yet. Instead, it works as an identity in the real world: i.e., it is based on a direct relationship between you and another party as peers.

It is like a string that you are both holding—if either one of you lets go, the string will drop. But as long as you both want it, the connection will persist.

2.3.4 Self-sovereign Identity

SSI is the new identity model based on Blockchain technology, Decentralized Identifiers, and Verifiable Credentials. It is a secure and digital peer-to-peer channel established between ID Issuer, ID Owner, and ID Verifier. When credentials are exchanged not even the Self-Sovereign Identity system provider knows what is being exchanged. Credential issuing becomes simpler and faster. SSI Credentials are tamper-proof through the use of cryptography. They are private and under your control. SSI uses Selective Identity disclosure technology. Self-Sovereign Identity credentials can be verified anywhere, at any time. Even if the issuer does not exist anymore. Personal Data is not stored on centralized servers. Meaning that for hackers to steal 50 million digital identity records they would have to hack those 50 million people individually. Considerably more difficult. Self-Sovereign Identity tries to abolish multiple passwords. You just need to know your wallet password [5](Figure 2.4).

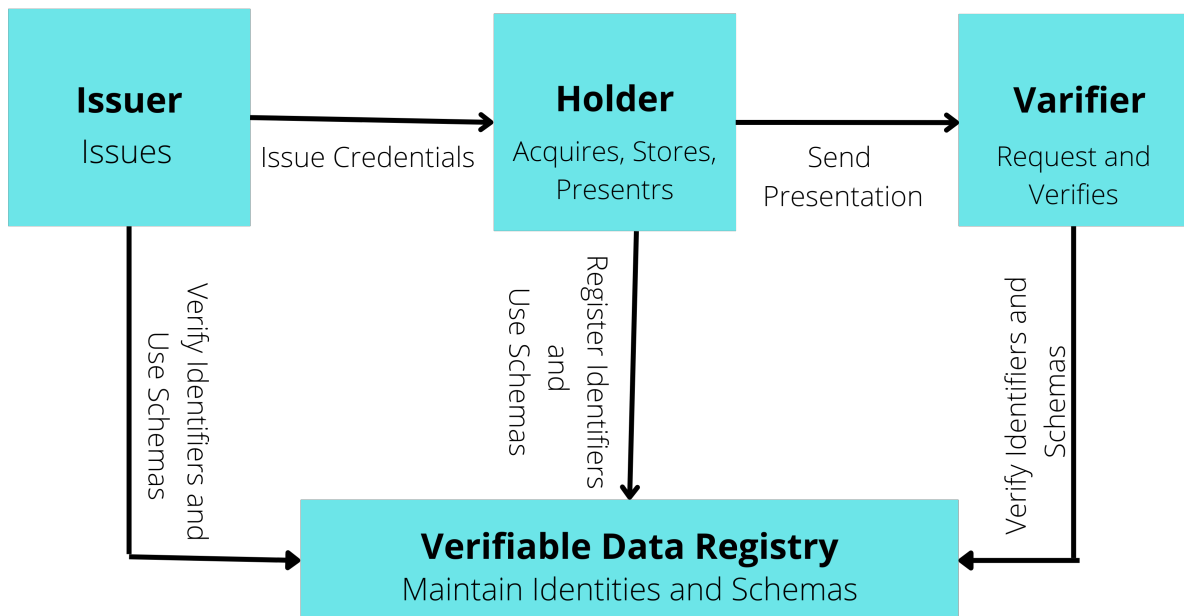


Figure 2.4: SSI Architecture

Chapter 3

Literature Review

The idea of self-sovereign identity is pretty new and people have started to implement it in various sectors. In the last few years, we have found some research work in which people have tried to improve the ownership management system in various ways. But the implementation of SSI in consumer-based IoT electronic devices and ownership transfer is the very first attempt and we have been motivated by some papers to go with this idea of research.

There are a few researchers who have worked on the ownership transfer system with the help of blockchain. Later we found only two research articles where researchers have found the problem of existing blockchain-based solutions and tried to solve the problem. But none of them have used SSI technology to solve the problem and improve the system. We have elaborated on those two articles and the problems of their solutions.

According to [9], researchers have proposed an idea of ownership transfer based on a trusted third party. In this proposal owner registration and the establishment of the security credentials with the particular IoT device are required. Moreover, TTP is responsible for generating ownership challenges. Furthermore, the IoT device has a pre-installed secret shared with TTP and has a one-way function $h()$ to generate the secret key for ownership. An ownership creation model is used when the IoT device ownership first is registered. The transfer procedure requires both the current owner and the new owner to be registered with the TTP. To transfer ownership a trust should be established between both owners: the current and new owner. This includes permissions and access control. A framework based on item-level access control through a mutual trust has been presented for IoT devices. In short, the newly manufactured device is given a key by a trusted third party But it still fails to ensure the real meaning of security. The involvement of a third party is always best to avoid.

According to [10], researchers have tried to secure ownership transfer by using the smart contract. As only blockchain-based ownership transfer is dependent on the trusted third party, it still can be hacked. By using smart contracts they come up with a solution to that problem. But a smart contract can not be the best solution in this case as it is hard to change its program.

Although smart contracts seek to eliminate third-party involvement, it is not possible to eliminate them. Third parties assume different roles from the ones they take in traditional contracts. For example, lawyers will not be needed to prepare individual contracts; however, they will be needed by developers to understand the terms to

create codes for smart contracts. Besides, it is not as user-centric as SSI [11].

Chapter 4

System Proposal and Threat Modeling

As we are working to develop a system based on SSI which will deal with sensitive information of the user and ensure the user-centric identity management independence we have to through sequential activities by following the systems development life cycle.

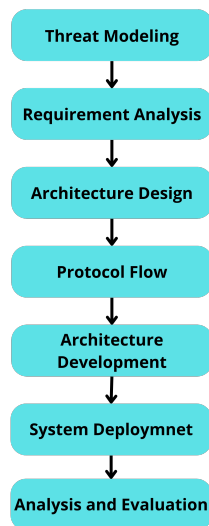


Figure 4.1: System proposal and work flow

4.1 Threat modeling

Threat modeling is a structured and step-by-step process to identify the threats to an application before building it. The threat modeling process helps to keep track of known security threats to an application and decide how to deal with them even before starting to write a single line of code. As it helps to detect the threat early in the software development life-cycle, it helps to identify the security requirements of an application earlier. It assists developers in identifying vulnerabilities and understanding the security implications of their design, code, and configuration choices. To perform a threat modeling we can follow the steps which are:

1. Diagram: A scenario of what application we are going to develop.
2. Identify threats: The list of threats that can be harmful to our system. To identify threats there are various threat-identifying models. We are going to use STRIDE. As we are developing an application that is mainly based on identity management, STRIDE is one of the best models as it covers the maximum vulnerable ways of breaking the security of an application.
3. Mitigate: We have to find out ways to defend against the threats we have found.
4. Validate: We have to verify that all of our previous analysis is working well or not.

As we have already said, we are going to use STRIDE to identify the threats to our future application based on the scenarios we have imagined. STRIDE is an acronym for each of the threat categories it deals with: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of privilege.

We are building a system that will secure ownership management and transfer consumer IoT Devices with a Blockchain-based self-sovereign identity. Here ownership will be managed in IoT devices and secure the system with block-chain and SSI. We will model the threats of the system. STRIDE is a threat modeling framework for ensuring secure application design in the context of securing (IT) assets and financial transactions. The process of creating a secure system that is used to identify, convey, and understand hazards and mitigation techniques is called threat modeling. To simulate assaults, we utilized the well-known Microsoft threat model STRIDE, which includes a number of security risks as seen below.

- **Spoofing Identity:** Spoofing is pretending to be someone other than his/her identity. A spoofing attack happens when a person (called a "Spoofing") pretends to be someone else to trick their target into giving them their personal information or doing something on their behalf. Most of the time, the spoofer will take time and make an effort to build trust with their target. This makes it more likely that the target will share sensitive information with them.

In our system, we are working on the management of ownership and securing the identity. So, let us say Alice is pretending to be Alice who has ownership of the system and owner of an IoT device. But this Alice is not the actual owner of the device but she is trying to transfer the ownership to her system. So the scenarios are:

1. In our system, the owner can be the fake one and can transfer the ownership of a product that is actually a fake one.
2. In our system, the buyer can be fake and can pretend to be a real buyer and ask for sensitive information about the product and also about the buyer.
3. In our application, the producer company of the IoT device can be fake. It can also hamper the warranty policy of that product. The first owner may receive a false ownership token from the hacker.

Now there are other types of spoofing:

1. Spoofing a Machine: Since we are dealing with IoT devices, the attacker can make a fake identity such as the MAC/IP address of the legitimate user, and an identity spoofing attacker can claim to be another legitimate IoT device.
 2. Spoofing a Person: Major categories of spoofing people include access to the person's account and pretending to be them through an alternate account. Phishing is a common way to get access to someone else's account. However, there's often little to prevent anyone from setting up an account and pretending to be you.
- **Tempering Threats:** When data or information is changed without permission, this is called "data tampering." A bad guy could do tampering by changing a configuration file to take control of the system, adding a malicious file, or deleting or changing a log file. Attackers will try to break into applications by changing target parameters or code to change application data like user credentials and permissions or other important things. Attacks like Cross Site Scripting (XSS) and SQL injection, which try to change the application, hurt its integrity.
 1. Tempering with File: Attackers can modify files wherever they can get written permission. In our system, this can be where the information will be stored. The transaction history or the assets the user owns in our system. All the information can be stored in a file. The file will be uploaded to our server. Or sometimes we attach our files with some other links.
Attackers/hackers can modify the documents of consumers of IoT devices or buyers. Also in our system, an attacker can temper the buyer's receipt if he can get written permission.
Tempering the receipt means the amount of personal information can be tempered by the attacker.
 2. Tempering with a Network: In order to get the data to the attacker's system, network tampering frequently includes a number of techniques. Tempering with the network means the flow of data in the machine can be tempered through the network. Suppose, the users are using wifi, while transaction or giving ownership to another user the attacker will be the third party in the network. The attacker will be able to eavesdrop on the conversation between the owner and the buyer. The attacker can get into the network and temper or change the data of the buyer or can get the id of the owner used for the transaction. Thus the hacker/attacker will get the most important data of the user. The attacker then transfers some intact and some modified data. Another file is the cache file. Which is given by a server. The cache file has important information about a user. Like which site the user follows, and what he/she likes. Sometimes passwords are stored in the cache file so that the attacker can break through the network. With radio interfaces like WiFi and Bluetooth, more and more data may now be transmitted across the air without the need for trickery.

In our system, we can predict that:

1. The hacker can be involved in private and sensitive conversations between the owner and the seller.
 2. Browser cache may contain contents of private messages.
 3. Hackers or attackers may be able to read, change or alter other users' information.
 4. Hackers can delete information if the authorization fails.
- **Repudiation Threat:** Repudiation threat is mainly denying the false or harmful action that has been carried out by the actor. Here the actor can be a normal user and also an attacker(malicious user). The action can happen with a bad intention and also can be by mistake. In ownership management and transfer application, there can be various ways to perform a criminal action or wrong action and the actor can deny that easily. We can predict that:
 1. In our application, an IoT device can be transferred to a new owner and the old owner (attacker) can deny that he did not transfer it to the new one.
 2. In our application, an IoT device can be transferred to the new owner with the wrong quantity. For example, the previous owner is saying that he has transferred the device to the new owner but the quantity of that item is not the same. It can happen by mistake or by an attacker. In our application, it can happen that the new owner denies that he has received the ownership of the product. It can happen when the old owner has transferred it but it has been intercepted by an attacker or a malicious owner is denying that they have received the ownership of the product.
 3. In our application, the producer company of the IoT device can deny that they did not transfer the ownership to the first owner after buying it. It can also hamper the warranty policy of that product.
 - **Information Disclosure Threats:** Information Disclosure Threats are all about leaking information to the attackers or to those who are not supposed to see the information. In our system, we are going to deal with lots of information from users like buyers and sellers who are legitimate owners.
 1. Information Disclosure from Database: There are a lot of ways how the data can be leaked. The first set of causes is the failed secure mechanism. Not setting strong security hoping that no one will find an obscure file. If cryptographic keys are found by the attacker, it will be much easier for them to decrypt the file and that will disclose more information about the user or owner.
 2. Information Disclosure from a Data Flow: When information is moving over a network, data flows are more vulnerable to information disclosure attacks. In addition to just reading data from the network, attackers may also reroute traffic to themselves (typically by forging some network control protocol), which would allow them to view it while they are not on the usual path.

- **Denial-of-Service Threats:** The goal of a Denial-of-Service (DoS) attack is to shut down a machine or network so that its users can't use it. DoS attacks do this by sending too much traffic to the target or information that makes it crash.

In our system:

1. If we want to buy or sell any product in an emergency. DoS attract will hamper that.
 2. If we want to transfer ownership of fast movable assets like bitcoins and stocks of share markets, DoS attract will restrict our system to execute that.
- **Elevation of Privilege Threats:** Elevation of privilege is allowing someone to do something they're not authorized to do. This means a security threat that can allow someone to do something to the system they are not authorized to do. If a user can get admin access to a system it hampers the security of the system. In our system:
 1. A user can sign in without a proper authentication and authorization process.
 2. Transaction data can be revoked by the attacker.
 3. Data from the owner's wallet can be transferred.
 4. Ownership can be taken back to the old owner from the new owner after selling a product without consent.
 5. Ownership certificates from the producer company of the device can be sent to the false owners.

Without these, we have found some more threats which can be crucial for our ownership management and transfer system. Which are:

1. **Transfer Without Consent:** In our application, a user or the new buyer can receive ownership without his/her desire. For example, the customer is not interested in buying a product after a bargain but he has received it by force.
2. **Control Transfer:** When the old user is transferring his ownership of the IoT device, the control of that previous owner on this device must have been terminated. Otherwise, the security of the new owner will be hampered.
3. **Delivery of new updates:** The producer company can offer a new update to the owner of their product. This update, feature, or extra benefit can still be received by the previous owner.
4. **Private Information of The Old Owner:** The private data of the old owner can be left in the IoT device and the new owner can receive that data without consent. It can break the privacy of the old owner.
5. **Second Owner Warranty Policy:** In our application, the producer company can deny giving the warranty service to the second owner as they

did not receive the ownership transfer notification from the first owner. That is why we have to ensure functionality that will notify the mother company of the product about the change of ownership. It will help them to serve warranty services to the new owner.

Chapter 5

Requirement Analysis

From the thread modeling we have to know some of the most vulnerable areas which can be dangerous for our system. Besides, we have understood that some functionality can make our system more secure and user-friendly. We have to design our site by considering the requirements necessary to solve all the problems and make our system more user-friendly. As we are more concerned about security, there will be no administrator access or any kind of extra beneficial authorization to our site.

- **Functional Requirements :**

1. Authentication:
 - Our system should integrate a password-less login and sign-in system.
 - Our system should be able to contact the smart wallets.
 - Our system should be able to verify a user by accepting crypto-tokens from the wallets.
 - Our system will take permission from the user before accepting any kind of data.
2. Validation:
 - Our system must ensure that the ownership certificate of the owner is legal and issued from the mother company.
3. Ownership Update:
 - Our system must update the name of the new owner of the device and prove a new ownership certificate to the buyer.
4. Authorization Management:
 - Our system must transfer the control of the device to the new owner
 - Our system will be able to remove the access of the old user from the device.
5. After sell service:
 - Our system will notify the mother company about the new owner of the product as he/she can avail of the after-sell services of that product.
6. Self-sovereignty:

- Our system will ensure that the information which is going to be transferred will be after taking the permission of the user. Users will be able to choose which information he/she is interested to share.

Non - functional Requirements :

1. Security:

- Our system must have a technology to ensure the real identity of the user of it. The system must ensure that the buyer and the seller are not fake and are verified by smart technology.
- Our system must have to verify the user every time before any kind of transaction of sensitive information.
- Our system must ensure that the user data of our system must be secured in a way that the hackers can not edit, delete or even view without authorization.
- Our system must ensure a passwordless authorization system that will prevent password cracking.
- Our system must be developed with the help of safe functions of the specific programming language which will help to prevent attacks like SQL injection.
- Our system must record the activity and actions of the user to show who has changes and files or functions. It will help to prevent repudiation attacks.
- Our system must be developed in a way that can protect our site from DoS attacks.

2. Usability:

- The application should have a good UI and UX for a smooth experience.
- The application should run on any device like a tablet, pad, mobile, laptop, etc.
- The application should run on all browsers.
- The application should generate a transaction certificate.
- The application should be built with all the new technologies.
- The application should be able to give notifications. The application should be SEO-friendly.
- The application should have analytics integrated with that for better data science implementation.

3. Performance:

- The application should be lite to load for a browser.
- The system should be first to load.
- The system should not take more memory.
- The system should run during the update period also.
- The system will not be stopped for a single second.
- The system should not be crashed.

Chapter 6

Architecture design

To ensure a safe and secure ownership transferring system where the control of data will be in the hand of the user, we have designed an ownership transferring system based on SSI where getting ownership from the manufacturer company and transferring that ownership to the next buyer will be safe and secure. Besides, the ownership data of the second buyer will also be validated by the manufacturer company. It will help the 2nd, 3rd, and so on buyers to avail warranty facilities and proper updates and facilities.

In our system, the manufacturer company is the main and the first issuer of ownership of an IoT product. When a person will buy a product as the first user of that IoT product, these manufacturer companies will establish an SSI connection with the buyer and send the ownership credential or VC to the user. All the manufacturing companies will be associated with SSI agents and will be connected with the distributed ledger. After getting the VC, the buyer will be the official owner of the product and he/she will store that VC in his/her wallet as proof of the ownership. As the buyer is storing the VC in his/her wallet, he/she also will be the holder of that VC. As the buyer will check the validation and originality of the VC, he/she is the verifier at the same time.

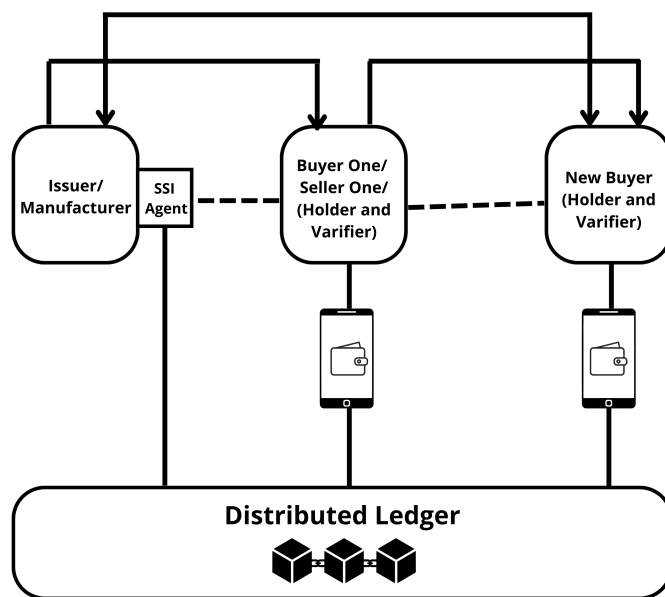


Figure 6.1: Architecture design of ownership management system

When this first buyer goes to sell the product, he/she will transfer the ownership VC that he/she has received from the manufacturer. Along with that, he/she will also generate a new VC as proof that he/she wants to sell the product. At this point, the 1st buyer will also be an issuer as he/she is creating a new VC and sending it to the new buyer.

The new buyer will receive two VCs and send them to the manufacturer for the new and updated VC. The manufacturer will be an issuer again and the new seller will verify that VC as verifier, store that VC in the wallet as a holder, and will issue a new VC as issuer when he/she will sell the product again.

Chapter 7

Use Case and Protocol Flow

Now, in this section, we will represent the use cases and protocol flow which will illustrate the interaction between different components of our system. We have divided our flow model into three parts. In (Table 7.1), we introduce all the mathematical notations.

Notations	Descriptions
A	Alice
B	Bob
M	Manufacturer
$[\dots]_{HTTPS}$	Communication through secure HTTP Channel
QR_M	QR code from manufacturer
DID_A^M	DID of A for M
URL_M	URL of M
DLT	Distributed ledger
$DIDDoc_A^M$	DIDDoc of A for M
DID_M^A	DID of M for A
$DIDDoc_M^A$	DIDDoc for A from M
$[\dots]_{K_{M-1}}$	Signed and Encrypted with the public key of M
VC_M^A	VC of M for A
$[\dots]_{K_A}$	Encrypted with the public key of Alice
$[\dots]_{K_{A-1}}$	Signed and Encrypted with the public key of Alice
VC_A^B	VC of A for B
VC_M^A	VC of M for A
$[\dots]_{K_M}$	Encrypted with the public key of M
$[\dots]_{K_B}$	Encrypted with the public key of B
VC_M^B	VC of M for B

Table 7.1: First and Second Segment P-value (Bat First)

The first one will show the interactions between various components to establish SSI communication between two different entities. (Figure 7.1) We have considered

Manufacturer and Alice as the two entities who want to establish SSI communication between them.

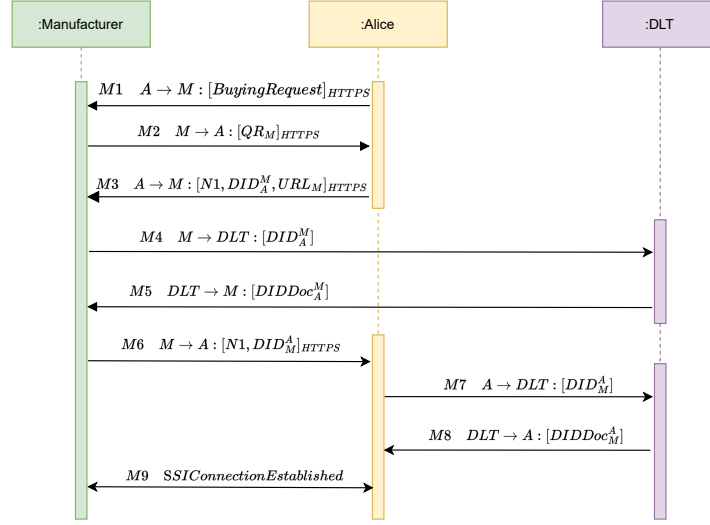


Figure 7.1: Protocol flow of SSI connection establishment between two entities

(Table 7.2) The protocols of part one are:

Protocols		
M1	$A \rightarrow M$: $[BuyingRequest]_{HTTPS}$
M2	$M \rightarrow A$: $[QR_M]_{HTTPS}$
M3	$A \rightarrow M$: $[N1, DID_A^M, URL_M]_{HTTPS}$
M4	$M \rightarrow DLT$: $[DID_A^M]$
M5	$DLT \rightarrow M$: $[DIDDoc_A^M]$
M6	$M \rightarrow A$: $[N1, DID_M^A]$
M7	$A \rightarrow DLT$: $[DID_M^A]$
M8	$DLT \rightarrow A$: $[DIDDoc_M^A]$
M9	$SSIConnectionEstablished$	

Table 7.2: Web-SSI Protocols Part 1

1. In M1, Alice wants to buy a product from manufacturer M. Alice sends a buying request to M. This request is going through a secured encrypted HTTPS channel.
2. In step M2 the manufacturer shows/sends a QR code to the buyer through an HTTPS channel again.
3. By scanning the QR, Alice will get a URL, and Alice sends this URL along with the DID of Alice to the manufacturer through an HTTPS channel in step M3.
4. In M4, the manufacturer will send this DID of Alice to the DLT for confirmation.
5. In M5, the manufacturer will receive DIDoc as a result of confirmation from DLT.
6. After getting the confirmation, the manufacturer will send its DID to Alice in step M6.
7. In step M7, Alice will send the DID of the manufacturer to DLT to check the validity.
8. Alice receives the DIDoc in step M8 as a result of confirmation.
9. After all of these processes, the manufacturer and Alice will establish SSI communication between them in step m9.

When two entities want to transfer ownership of a product between them, they establish an SSI connection between them in this same way every time.

(Figure 7.2) In the second one, Alice can start the process of getting ownership of his product from the manufacturer. This is the process of getting ownership as the first buyer from the manufacturer. (Figure 7.2) The protocols of this part are:

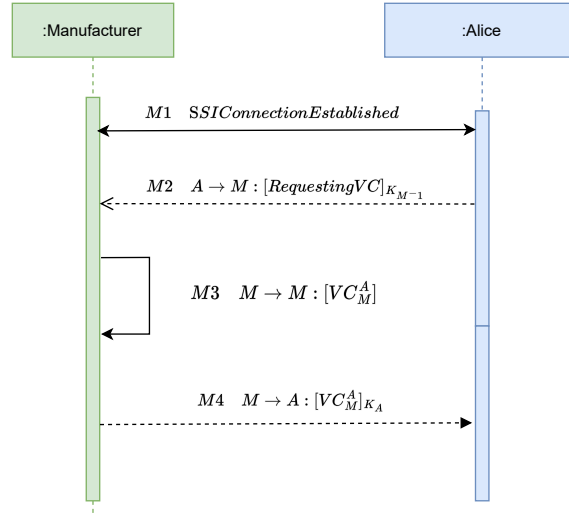


Figure 7.2: Protocol flow of getting ownership from manufacturer

(Figure 7.2) The protocols of this part are:

Protocols		
$M1$	$SSIConectionEstablished$	
$M2$	$A \rightarrow M$	$: [RequestingVC]_{K_{M-1}}$
$M3$	$M \rightarrow M$	$: [VC_M^A]$
$M4$	$M \rightarrow A$	$: [VC_M^A]_{K_A}$

Table 7.3: Web-SSI Protocols Part 2

1. In step M1, the SSI connection will be established as it is described .in part 1.
2. In step M2, Alice will request the manufacturer to get his ownership VC. Alice will send this request by giving the signature and encrypting it with the public key of the manufacturer.
3. After getting the request from the buyer, the manufacturer will generate a VC in step M3 for Alice.
4. In step, M4 Manufacturer will send this VC to Alice by encrypting it with the public key of Alice.

Now in (figure 7.3), we will illustrate the flow between different components if Alice wants to sell his product to Bob. This is the process of transferring ownership from the first buyer to the second buyer.

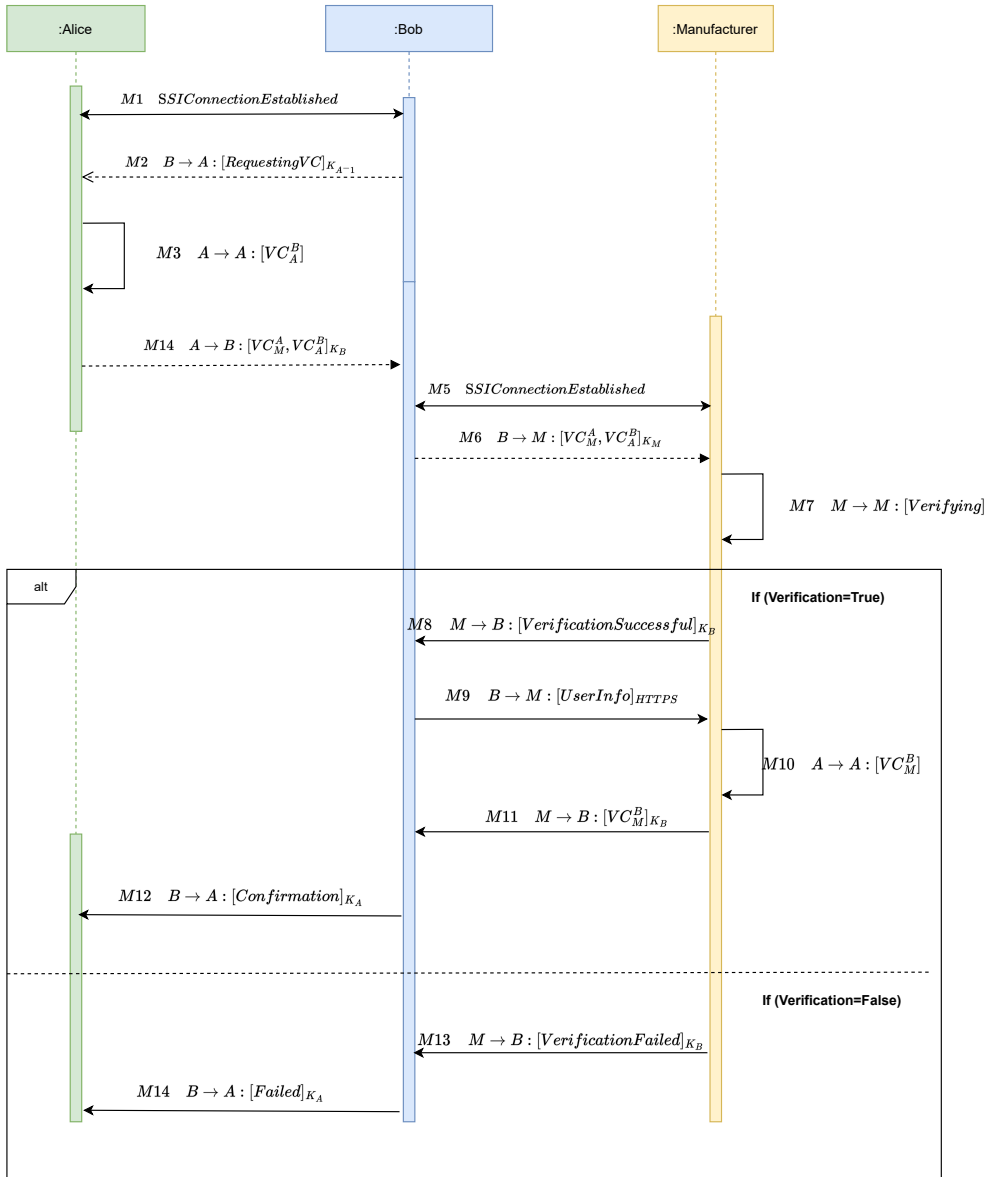


Figure 7.3: Protocol flow of transferring ownership to the new owner

Protocols		
M1	<i>SSIConectionEstablished</i>	
M2	$B \rightarrow A$: $[RequestingVC]_{K_{A^{-1}}}$
M3	$A \rightarrow A$: $[VC_A^B]$
M4	$A \rightarrow B$: $[VC_M^A, VC_A^B]_{K_B}$
M5	<i>SSIConectionEstablished</i>	
M6	$B \rightarrow M$: $[VC_M^A, VC_A^B]_{K_M}$
M7	$M \rightarrow M$: $[Verifying]$
M8	$M \rightarrow B$: $[VerificationSuccessful]_{K_B}$
M9	$B \rightarrow M$: $[UserInfo]_{HTTPS}$
M10	$A \rightarrow A$: $[VC_M^B]$
M11	$M \rightarrow B$: $[VC_M^B]_{K_B}$
M12	$B \rightarrow A$: $[Confirmation]_{K_A}$
M13	$M \rightarrow B$: $[VerificationFailed]_{K_B}$
M14	$B \rightarrow A$: $[Failed]_{K_A}$

Table 7.4: Web-SSI Protocols Part 3

1. At this time they have to establish an SSI connection between Alice and Bob again as previously according to the same procedure of (figure 7.1) in step M1.
2. In step M2, Bob will send a request to Alice for VC through a signed message which is encrypted with a public key of Alice.
3. Alice will create a VC for Bob in M3.
4. In step M4, Alice will send this buying request accepting VC along with the old ownership VC by encrypting it with a public key of Bob.
5. After getting the ownership VC of Alice and buying request acceptance VC from Alice, in M5, Bob will establish an SSI connection with the manufacturer like before according to the same procedure of (Figure 7.1).
6. Bob will send these two VCs to the manufacturer by encrypting them with the public key of the manufacturer in step M6.
7. The manufacturer will verify these two VCs in step M7.
8. If the verification is successful, in step M8, the manufacturer sends Bob a confirmation message by encrypting it with the public key of Bob.
9. Bob sends his identity information to the manufacturer in step M9 through an HTTPS channel.
10. With this information, in step M10, the manufacturer will create a new ownership VC for Bob.
11. In step M11, the Manufacturer sends it to Bob by encrypting it with the public key of Bob.
12. After getting the new ownership VC, in M12, Bob confirms to Alice about the transaction, and the ownership has been transferred successfully.
13. But, if the verification is not successful in M7, the manufacturer will send a message to Bob about the failure of the verification in M13 by encrypting the public key of Bob.
14. In Step M14, Bob will notify Alice of the failure message by encrypting the public key of Alice.

Now, we can present a visual representation of our application design. We can divide

this presentation according to the use cases and describe it accordingly. The use cases are:

1. The manufacturer will sell the product:

- To sell the product, the manufacturer company has to open its own system and have to log in with the credentials like branch number and password. In this stem, to ensure more security, the manufacturer can log in with the VC it has gotten from the government to produce IoT products. (figure 7.4)

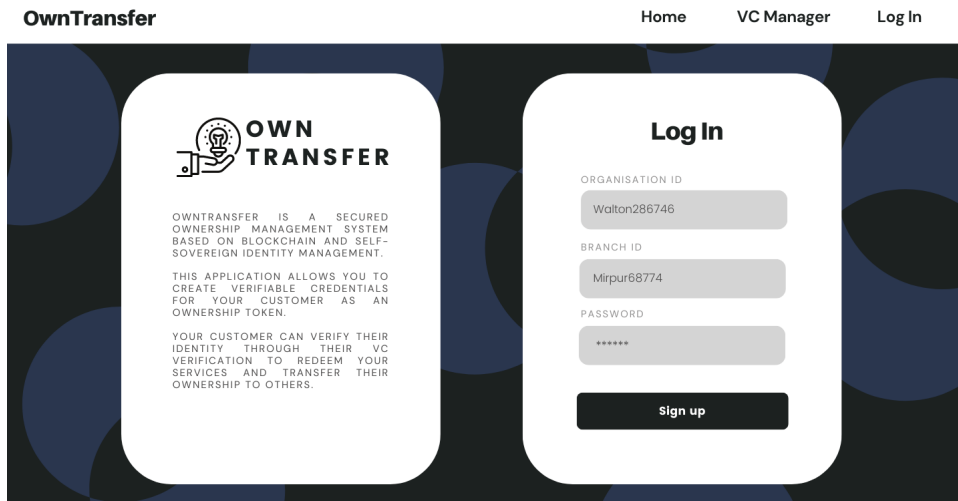


Figure 7.4: Application design - manufacturer login

-After signing up/login, the manufacturer will choose the option “New Ownership” according to the request of Alice. (figure 7.5)

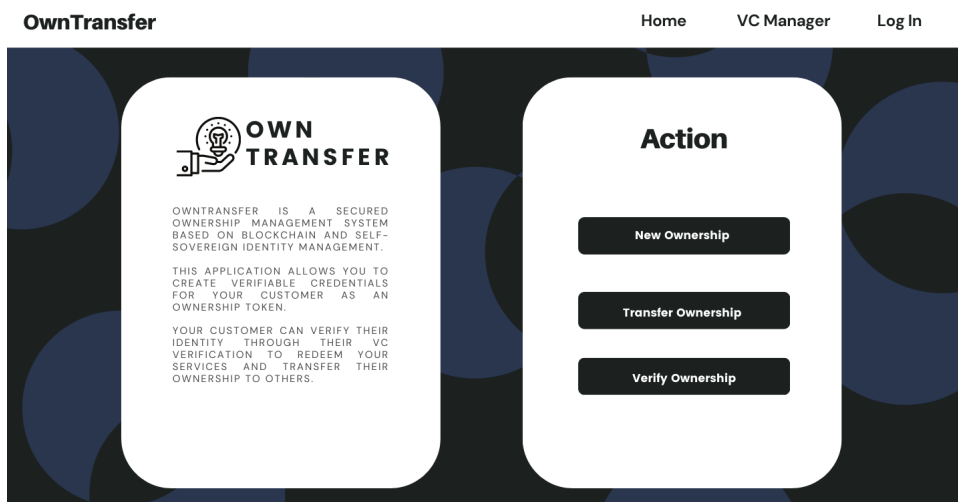


Figure 7.5: Application design - Manufacturer Action

-After that, the manufacturer will show a QR code to Alice to establish the SSI connection between them. (figure 7.6)



Figure 7.6: Application design - Manufacturer's QR code to establish SSI connection

2. Alice will receive his ownership certificate:

- To get VC, Alice has to scan the QR code from the manufacturer to establish the SSI connection.
- To scan the VC, Alice will open his wallet with a credential or biometric, and choose the option "Add New Credential" (figure 7.7)

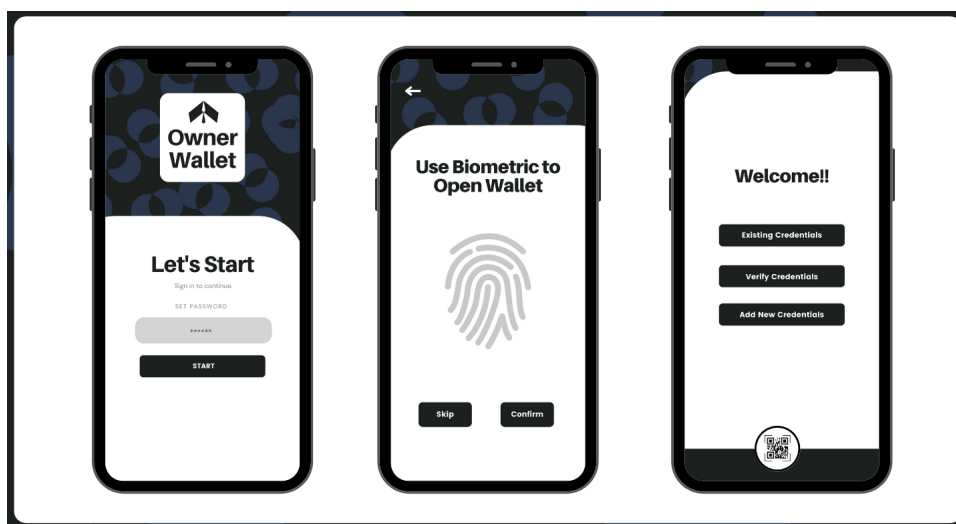


Figure 7.7: Application design - Owner's Login

- After that, Alice has to Scan the QR code with the help of a mobile camera and the SSI connection will be established. (figure 7.8)

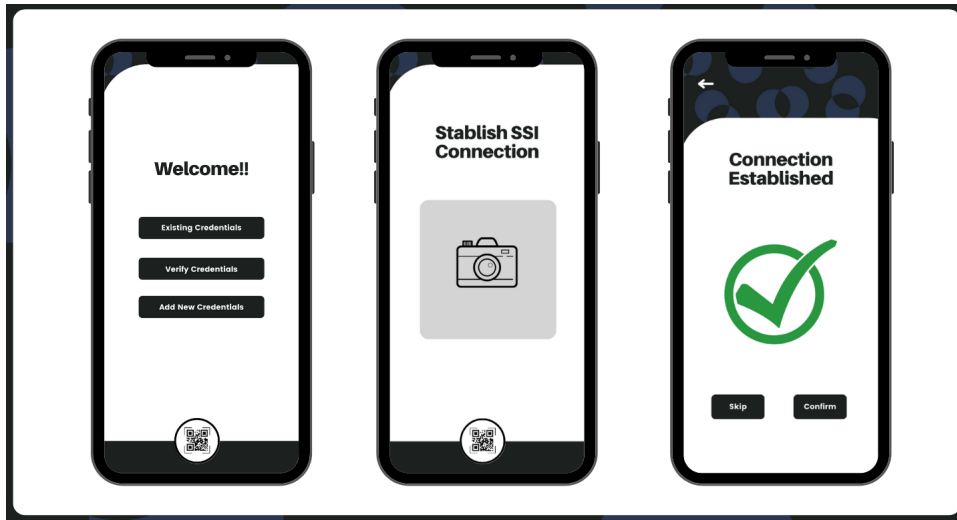


Figure 7.8: Application design - Owner's Connection Establishment

- The manufacturer will also get a confirmation of establishing an SSI connection (figure 7.9)

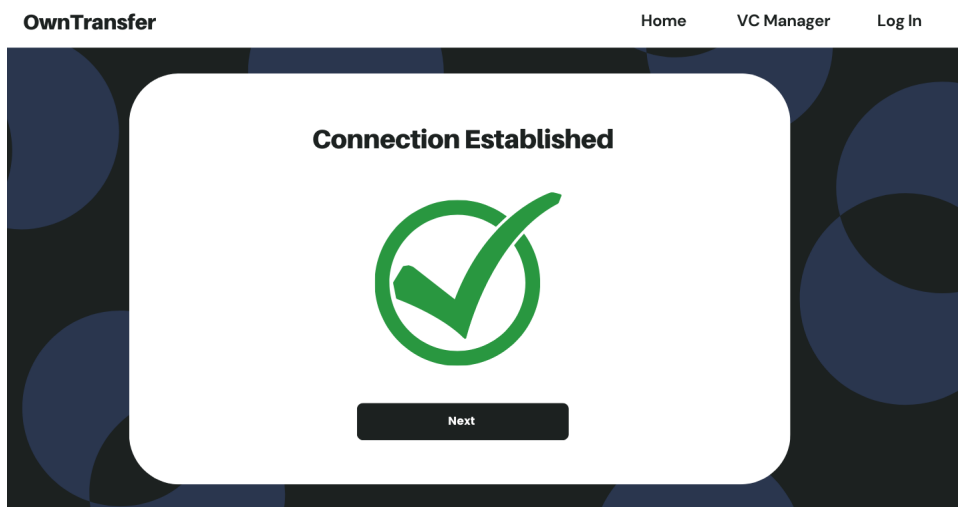


Figure 7.9: Application design - Manufacturer's connection establishment notification

- To get VC, Alice has to give his identity-related information to the manufacturer. She will transfer VCs according to necessity. (figure 7.10)

-After that, the manufacturer will input the product serial number and the date of purchase and generate VC (figure 7.11) - Now , Alice has to accept the VC and store it in her wallet.(figure 7.12)

3. Alice will transfer his ownership to Bob:

- To transfer VC to Bob, Alice has to establish the SSI connection with Bob and Bob has to scan the QR with his wallet. (figure 7.13).

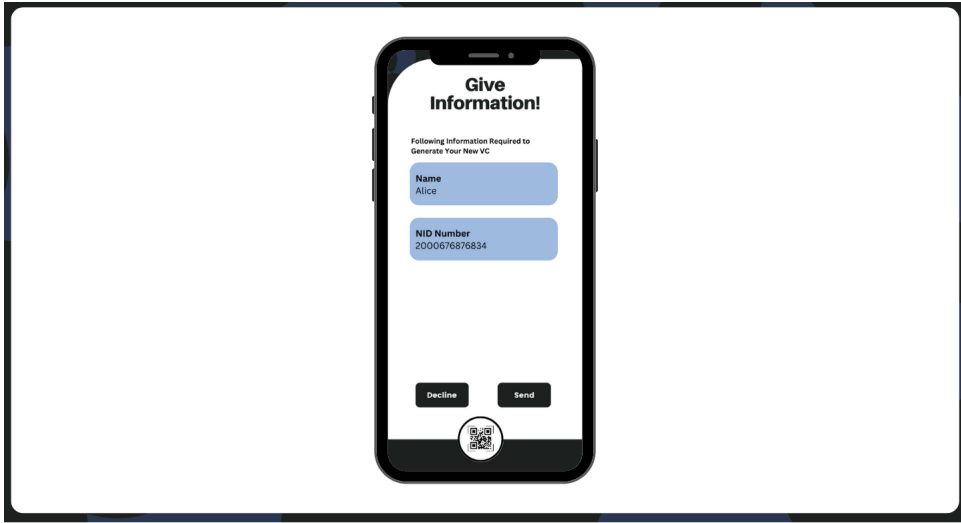


Figure 7.10: Application design - Alice information containing VC sending

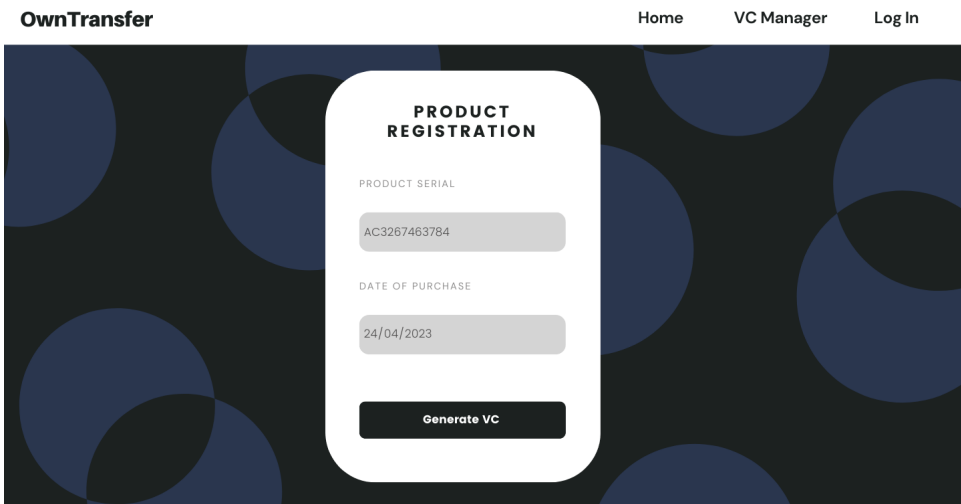


Figure 7.11: Application design - New product registration

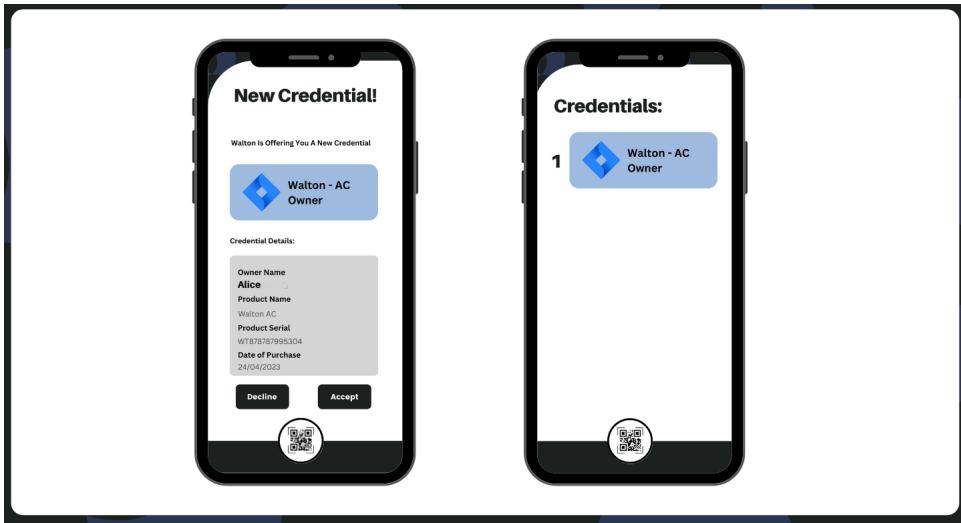


Figure 7.12: Application design - Ownership VC accepting for Alice

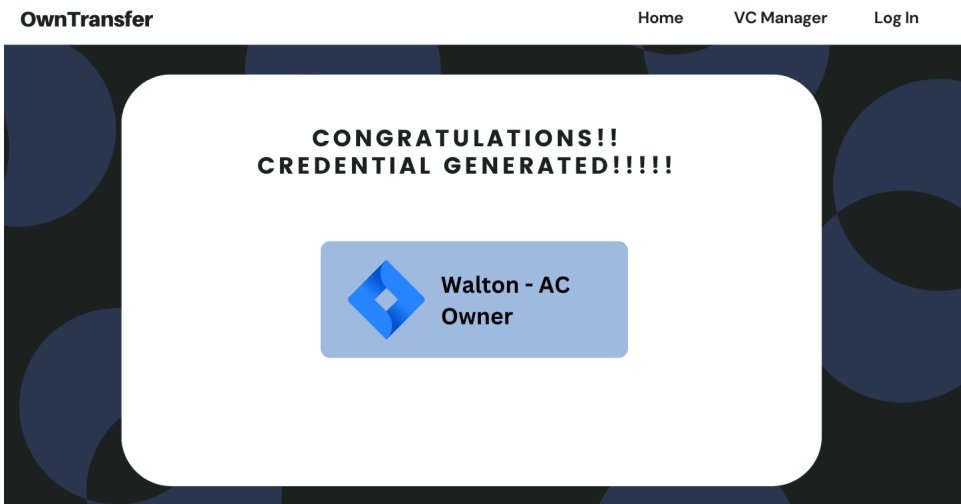


Figure 7.13: Application design - VC generation notification at Manufacturer's end

4. Bob will receive his ownership certificate:

- Now, Bob has to establish an SSI connection with the manufacturer. The manufacturer will select the option “Ownership Transfer” and show the QR code to Bob. (Figure 7.14 and Figure 7.15)

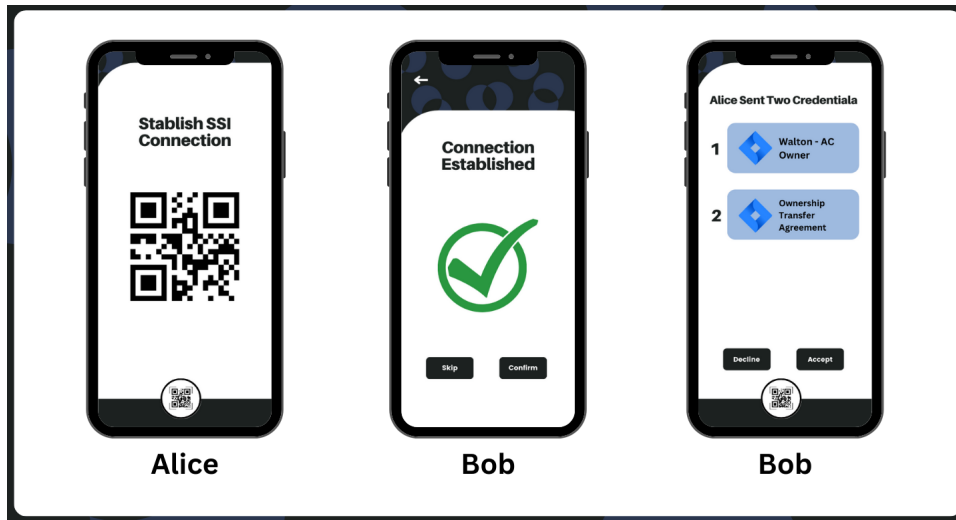


Figure 7.14: Application design - SSI Connection establishment between Alice and Bob

- At this point , Bob and Alice both will get a connection establishment notfiac-tion.(Figure 7.16 and Figure 7.17).
- Now, Bob has to transfer the two VCs which he has hotten from Alice.(Figure

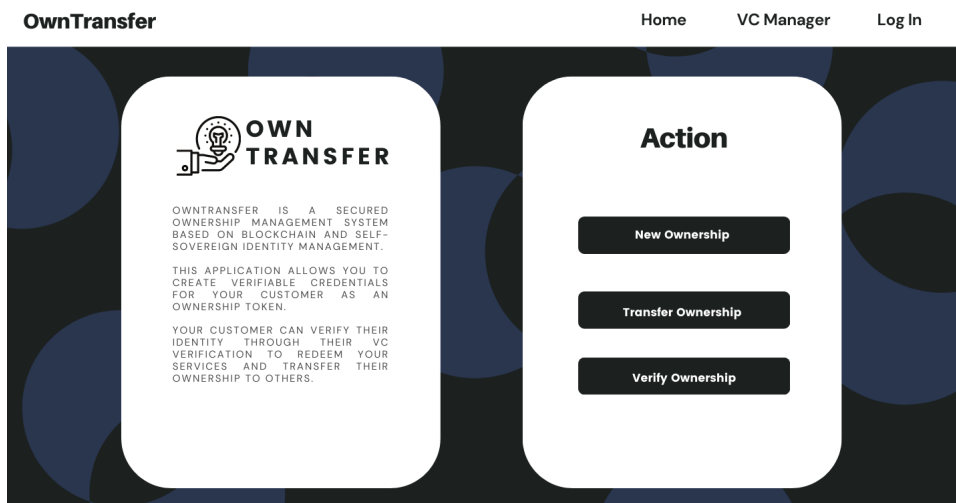


Figure 7.15: Application design - Manufacturer's transfer ownership action

7.16).

- Manufacturer will verify the VCs (Figure 7.19)

- To get VC, now Bob has to transfer his identity information VCs to the manufac-turer (Figure 7.20)



Figure 7.16: Application design - SSI connection establishment between manufacturer and Bob

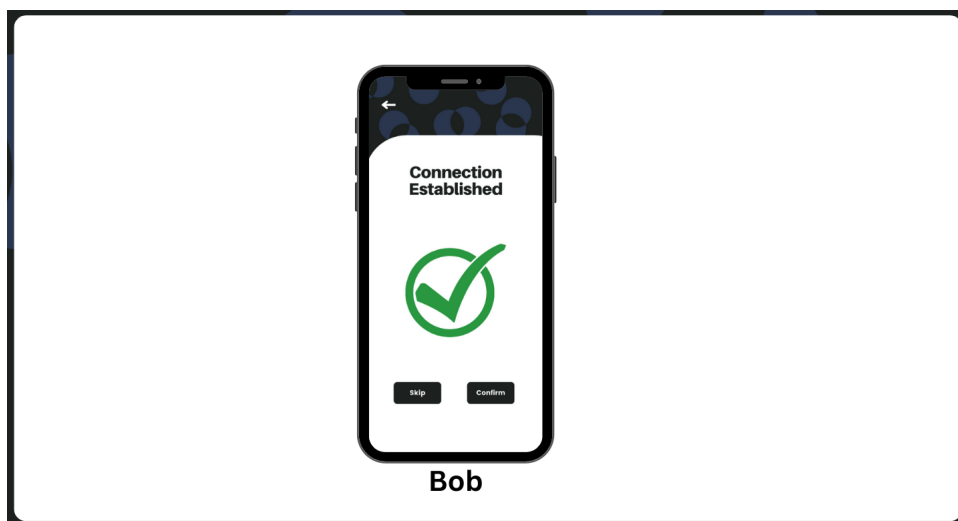


Figure 7.17: Application design - Connection establishment notification to Bob

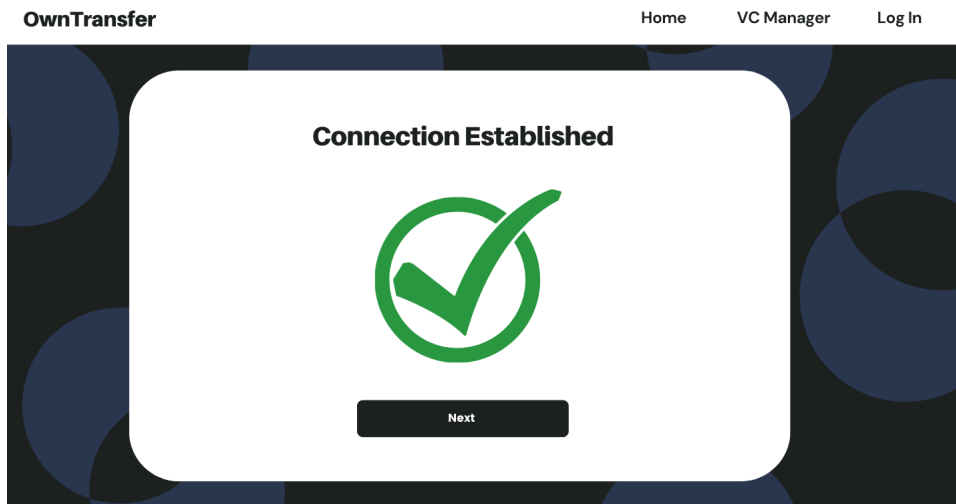


Figure 7.18: Application design - Connection establishment notification to Manufacturer

- Now, the manufacturer will generate a new VC and Bob has to accept and store it (Figure 7.21)

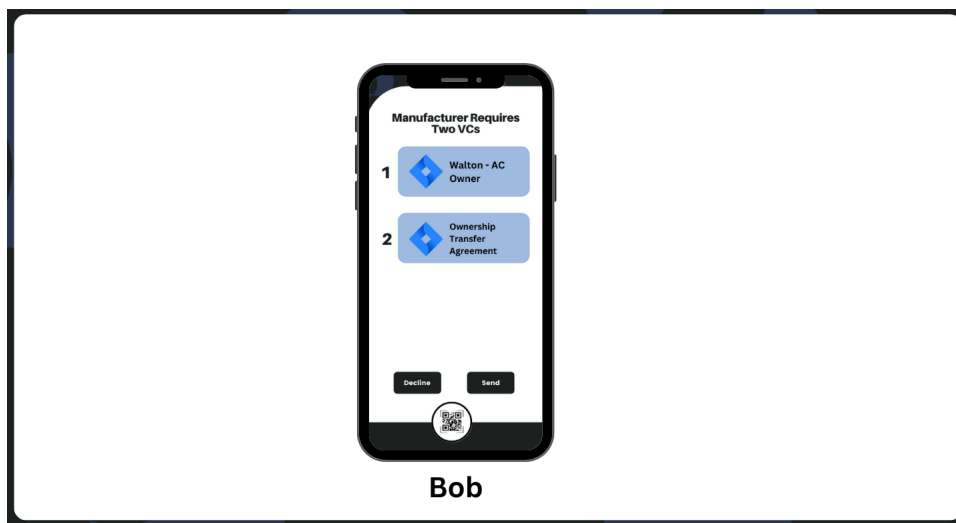


Figure 7.19: Application design - Two VCs received from Alice sent to Manufacturer

The ownership has been transferred from Alice to Bob successfully.

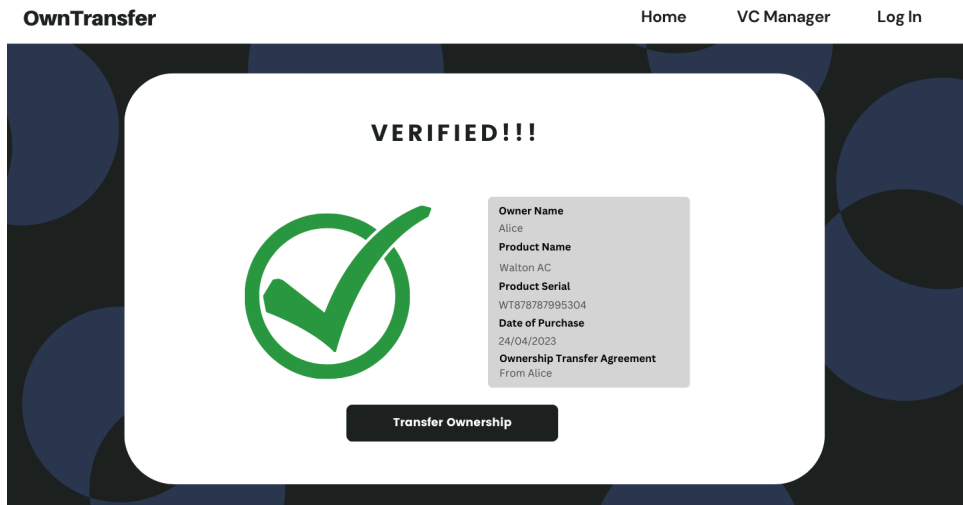


Figure 7.20: Application design - VC verification at Manufacturer's end

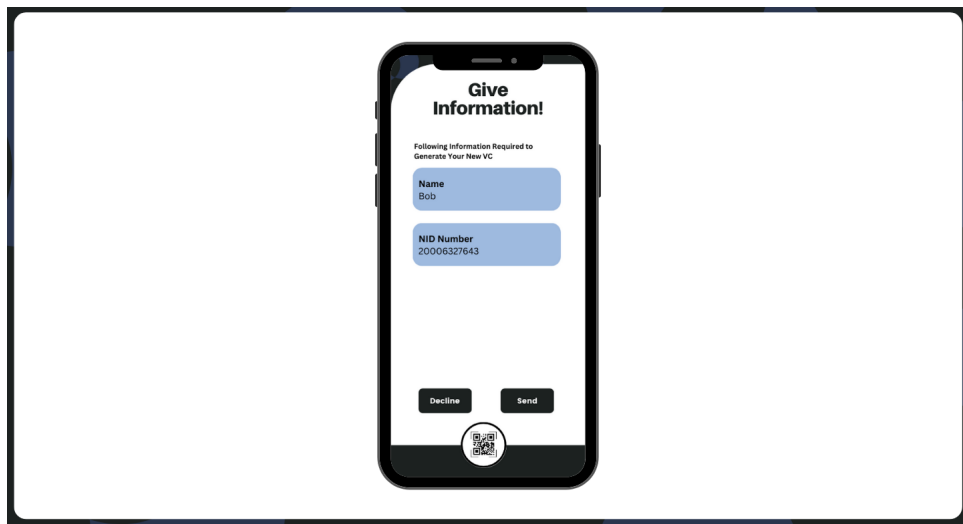


Figure 7.21: Application design - Bob's identity contained VC sent to Manufacturer

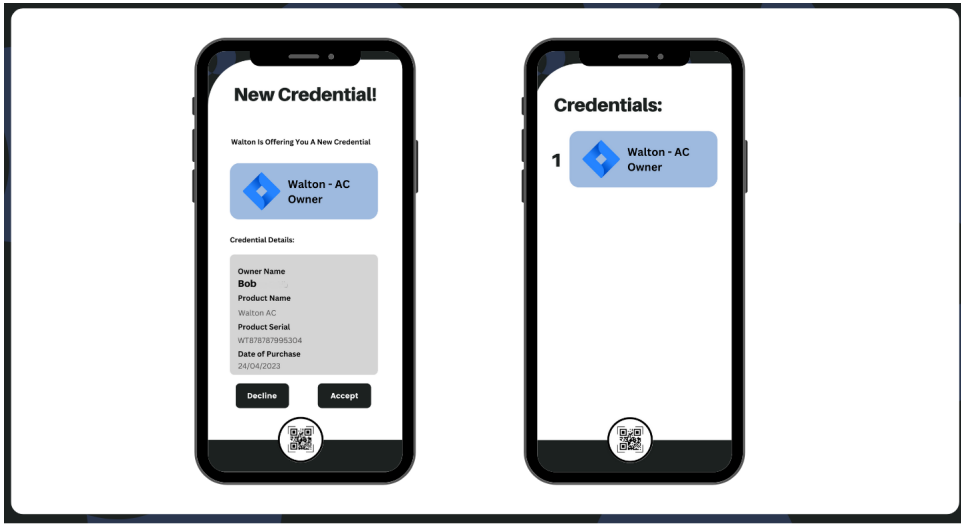


Figure 7.22: Application design - New ownership credential received by Bob

Chapter 8

Discussion

8.1 Analysing Requirements

Functional Requirements:

- By implementing the SSI model in this ownership transfer system, the system has been completely passwordless system. We have seen that Alice and Bob can get control their VCs without entering passwords for individuals and can control their system. This model can verify the user using the crypto-tokens after getting the VCs. Besides, sending VCs, accepting VCs is not possible without the secretive permission of the user. So, it makes the whole process user-centric and handles the FR1 successfully.
- When our system transfers ownership, it accepts the user's identity information through VCs and verifies it before generating a new VC. Besides, as it is an SSI model, the manufacturer will not be able to get VC-creating permission without the permission of the mother company. Besides it follows the trust framework which meets the FR2.
- The system can update the information of the owner and can generate a new VC for the new owner to ensure FR3
- When the manufacturer issue a new credential of ownership, it revokes the previous ownership certificate. So, it is able to remove the access of the first owner from the device. The new owner can register himself with the new VC to the device to get access. In this case, the devices have to be under the DLT network. Then it will secure the FR4
- As the new owner has to connect with the mother company to get new VC, the mother company will be informed about the transfer of the ownership and the owner will get the guarantee and other services from the company. It will mitigate the FR5
- To process every transaction of data, the user is in the middle of our system. Without the permission of the user, no data will be transferred successfully. So, this system is self-sovereign and handles the FR6 successfully.

Security Requirements:

- To provide VC, our system gets owner information from their identity-related VCS. So, the buyer is not fake and their identity is government certified.
- The data are stored in the distributed ledger which is developed based on blockchain technology. So, it is secure and unhackable.
- The system is completely passwordless so password cracking is not possible.
- As the system is not developed completely, the secure functions of the programming language are not implemented. But it is possible and proven.

Usability and Performance Requirements: As the system is not completely developed, the complete UI is not tested yet. But it is designed by following good design principles. Besides, the performance is also not measured as it is not completely developed.

8.2 Advantages Limitations

The advantages of SSI-based ownership management and transfer system are:

- The system is completely passwordless.
- The user data will not go to the manufacturer's database.
- The user data will not be used without the permission of the user.
- The user data will be stored in a distributed ledger, so it will be secured.
- The user information management will be easier for the company.
- The after-sell service will be easier for the owner and also for the manufacturer to track it.
- The control over the IoT devices will be more secure and easier.

The limitation of the SSI-based ownership management and transfer system are:

- As all the identity-providing authority is not connected with the DLT technology, it will be hard to get identity VCs from the owners.
- If the IoT devices are not connected with the DLT, it will not be possible to remove the previous owner's information from it and register it with the new owner.
- As the design of this system is a bit complex, it will be hard to develop its own system for every company. So, small companies have to rely on the 3rd party SSI-based ownership management service providers. Though it is safe but it can be expensive.

8.3 Future Work

- We have to make our system prepare for managing ownership of all the electric products besides IoT devices.
- We have to design a system to connect with the IoT devices directly from the owner's wallet to control it.
- We have to design a system to connect with the IoT devices directly from the manufacturer's end to reset them.
- We have to develop the complete system to make it more realistic in the future

Chapter 9

Conclusion

In this work, we have presented an ownership management and ownership transfer system based on SSI. SSI technology makes our system completely passwordless and self-sovereign where the user is the center of the system. Without the permission of the user, the transaction is not possible which makes a new security layer to our system. The motivation was the data of the user will not be in the hand of anyone without the user, not even in the hand of the manufacturer. Our system design has made it completely possible. The system will also easier the process of authorization and access control of the IoT devices easier. It will reduce the identity breach to the minimum level possible. We have designed its architecture which is based on a threat model and requirement analysis, discussed its application design in detail, and highlighted use cases to show its applicability. With these contributions, we believe that this will open a new domain for the researcher to upgrade with their ideas and developers to develop new applications.

Bibliography

- [1] Alexander S. Gillis (2022) “What is the internet of things (IoT)?” Accessed: 24-04-2022 [Online]. Available: <https://www.techtarget.com/iotagenda/definition/Internet-of-Things-IoT>
- [2] Statistica. (January 22, 2021) “Forecast end-user spending on IoT solutions worldwide from 2017 to 2025”. Accessed: 24-04-2022 [Online]. Available: <https://www.statista.com/statistics/976313/global-iot-market-size/>
- [3] Statistica (Mar 17, 2022) “Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030, by use case”. Accessed: 24-04-2022 [Online]. Available: <https://www.statista.com/statistics/1194701/iot-connected-devices-use-case/>
- [4] Verified Market Research. (July 2021) “Global Consumer IOT Market Size By Type, By Application, By Geographic Scope And Forecast”. Accessed: 24-04-2022 [Online]. Available: <https://www.verifiedmarketresearch.com/product/consumer-iot-market/>
- [5] Tykn. tech. “Self-Sovereign Identity: The Ultimate Beginners Guide!” Accessed:(2022). <https://cutt.ly/xC3vdZz>
- [6] Rakesh Soni (2020) “The Role of IoT Identity Management in 2020.” Accessed: 24-04-2022 [Online]. Available: <https://www.iotforall.com/identity-management-iot>
- [7] Vmware. “ What is Identity Management?” Accessed: 24-04-2022 [Online]. Available: <https://www.vmware.com/topics/glossary/content/identity-management.html>
- [8] Alex Preukschat, Fabian Vogelsteller. (2021). Self-Sovereign Identity
- [9] X. Leng, K. Mayes, and Y. Lien, “Ownership Management in the Context of the Internet of Things,” in Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2014 International Conference on. IEEE, 2014, pp. 150–153.
- [10] M. Alblooshi, K. Salah, Y. Alhammadi, “Blockchain-based Ownership Management for Medical IoT (MIoT) Devices” in 2014 International Conference on Cyber-Enabled Distributed
- [11] CFI, “What are Smart Contracts?” Accessed: 24-04-2022 [Online]. Available: <https://corporatefinanceinstitute.com/resources/knowledge/deals/smart-contracts/>