California State University, San Bernardino

CSUSB ScholarWorks

Electronic Theses, Projects, and Dissertations

Office of Graduate Studies

5-2021

RESHAPING ORGANIZATIONAL PROCESSES AND WORKFLOWS THROUGH INTEGRATION OF BLOCKCHAIN TECHNOLOGY

Elijah E. Maggini

California State University - San Bernardino

RESHAPING ORGANIZATIONAL PROCESSES AND WORKFLOWS

THROUGH

INTEGRATION OF BLOCKCHAIN TECHNOLOGY

A Project

Presented to the

Faculty of

California State University,

San Bernardino

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

in

Information Systems and Technology

---

by

Elijah E. Maggini

May 2021

RESHAPING ORGANIZATIONAL PROCESSES AND WORKFLOWS THROUGH INTEGRATION OF BLOCKCHAIN TECHNOLOGY

A Project

Presented to the

Faculty of

California State University,

San Bernardino

by

Elijah E. Maggini

May 2021

Approved by:

Conrad Shayo, Ph.D, Committee Chair

Jay Varzandeh, Ph.D, Chair, Department of Information & Decision Sciences

Jack H. Brown College of Business and Public Administration

ABSTRACT

Cybercrime is becoming increasingly sophisticated and devastating as time carries on while many processes and workflows that exist within organizations are stagnant. This project analyzed Blockchain Technology as a use-case for building upon simple processes and workflows that are often overlooked within organizations for the purpose of hardening security and strengthening non-repudiation. This project examined three main questions relating to; how Blockchain can enhance traditional cyber security practices, how Blockchain can be introduced to organizations as a ground-breaking and worthwhile solution to countering cyber-attacks, and the benefits and risks of implementing Blockchain within an organization. An investigation of traditional cyber security practices and existing use-cases of Blockchain Technology was conducted alongside the development of two prototype Blockchain applications in order to illustrate organizational use-cases. The results of this study concluded that Blockchain is capable of enhancing traditional cyber security best practices by serving as an effective method of access control; Blockchain can be introduced to organizations through platforms such as SIMBA that are easy to use and simple to understand; and lastly, there are various benefits and risks of implementing Blockchain within an organization such as the benefit of increased transparency and accountability and the risk of the technology being costly and complicated to implement. Areas of further research include the utilization of

Blockchain-based voting systems to ensure the integrity of elections, as well as tracking those who have or have not received vaccinations.

ACKNOWLEDGEMENTS

It is with pleasure that I dedicate this project to everyone in my life who has encouraged me or even discouraged me at one time. It is because of you that I am who I am today. Cheers to the future, this is only the beginning.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

Cybersecurity is recognized today to be more applicable within the lives of billions of individuals around the world than at any other time in the modern era. There is no longer a day that passes without there being reports of breaches that have taken place within organizations, usually resulting in the loss/theft of data belonging to millions of individuals. These breaches occur within government agencies, multinational conglomerates and even small to medium-sized businesses. Regardless of the size of the organization, they are all required and motivated by laws and regulations, e.g., HIPPA, Sarbanes Oxley; to do all that they possibly can to secure their data and prevent breaches from occurring. They are also motivated by the simple principle of keeping their customers happy.

Some of the technologies that are considered to be competitors of Blockchain Technology are the implementation of domain controllers within an organization to authenticate and grant users permission to navigate through a network, a traditional network-attached storage system for storing files and granting permissions to said files, as well as software providers such as LogRhythm which is used for capturing logs and tracking changes that are made to files for the purpose of ensuring compliance to regulations such as those mentioned above. Blockchain Technology is superior in comparison to the other technologies due to it being able to: provide an architecture for authentication, enforce read/write permissions for users, serve as a secure method to store and transfer data, as well as to track changes that are made on a system, ensuring non-repudiation for the purpose of adhering to compliance. This project will demonstrate how Blockchain Technology can be used to enhance traditional processes and workflows within organizations, specifically in the cybersecurity and supply chain environments.

Blockchain is the technology that can undeniably transform the way that organizations protect data that is precious to both themselves as well as their customers. Blockchain has been applied successfully in providing secure cyber environments in organizations and even in an entire country. For example, the country of Estonia hashes their medical records and then records them to a Blockchain every second to ensure that data cannot be tampered with (Tinianow, 2019). Blockchain has also been applied to authentication processes within three Irish banks. The Institute of Banking, Bank of Ireland, and Alb and Ulster Bank partnered with Deloitte to implement an Ethereum-based platform that will “support the verification, tracking, direct access to, and management of, regulatory and other professional designations, education qualifications and lifelong learning credentials” (Suberg, 2019). Furthermore, British Airways is implementing Blockchain within their organization for the purpose of streamlining their security checkup processes while ensuring that they execute safely (Iredale, 2020).

This project will illustrate the potential that blockchain technology has to preserve the confidentiality, integrity, and availability of data by serving as a foundation of need to know environments within organizations and to furthermore enhance cyber security and supply chain best practices by playing a critical role in implementing risk management strategies to protect intellectual property. At this time, there have been few attempts to implement blockchain within organizations for the specific purpose of enhancing best practices and risk management. For example, there has been a case study done on implementing Blockchain within Denmark's health system for identity and access management to patient data (Jacobsen & Makula, 2018). In addition, Blockchain would be a worthwhile component of implementing a Multi-Level Security (MLS) policy within organizational domains. The two projects done respectably by Jake Hyun and Garo Panossian emphasize Multi-Level Security based on the Bell-La Padula (BLP) model, which ultimately serves as a model of access control based on security levels such as Unclassified, Confidential, Secret, and Top-Secret (Panossian, 2019). This project will illustrate how Blockchain Technology can be an effective component of Multi-Level Security implementations.

A blockchain is essentially a digital ledger that stores data within blocks that are cryptographically chained together (NIST, 2021). Once data is stored within a block, it cannot be modified or deleted. One of the first implementations of blockchain was to serve as a decentralized public ledger to store the entirety of transactions that take place using the Bitcoin cryptocurrency. More recently, financial institutions have experimented with blockchain to securely facilitate cash transactions [Fintech News, 2020]. Blockchain has the potential to provide value to organizations in the sense that it can be tailored to meet the needs of those entities who choose to utilize it. A blockchain can be implemented in a way that allows it to be permissioned, meaning that only users who are given permission can read and/or write data to the blockchain, or it can be permissionless, meaning that anyone can read and/or write to the blockchain. The characteristics of blockchain, specifically its security mechanisms bring about three questions:

1. In what ways can blockchain serve to enhance traditional cyber security practices?
2. How can blockchain be introduced to organizations as a ground-breaking and worthwhile solution to countering cyber-attacks?
3. What are the benefits and risks of implementing blockchain technology within an organization?

This project will analyze and evaluate blockchain as a concept, its potential to be compatible with traditional cyber security and supply chain practices, as well as lay out the benefits and risks of blockchain and whether one outweighs the other. It will additionally cover blockchain as a secure solution for various interorganizational processes. This will be demonstrated by the design and implementation of a prototype Blockchain Solution. The NIST Framework will be referenced to illustrate the importance of blockchain in enhancing traditional cyber security best practices.

CHAPTER TWO

BLOCKCHAIN

An Overview of Blockchain

Blockchain is a relatively new technology having emerged in 1991 through work done by Stuart Haber and W. Scott Stornetta (Iredaleon, 2020). Their initial purpose behind Blockchain was to incorporate a cryptographic chain of blocks that would ultimately prevent timestamps of documents from being tampered with.

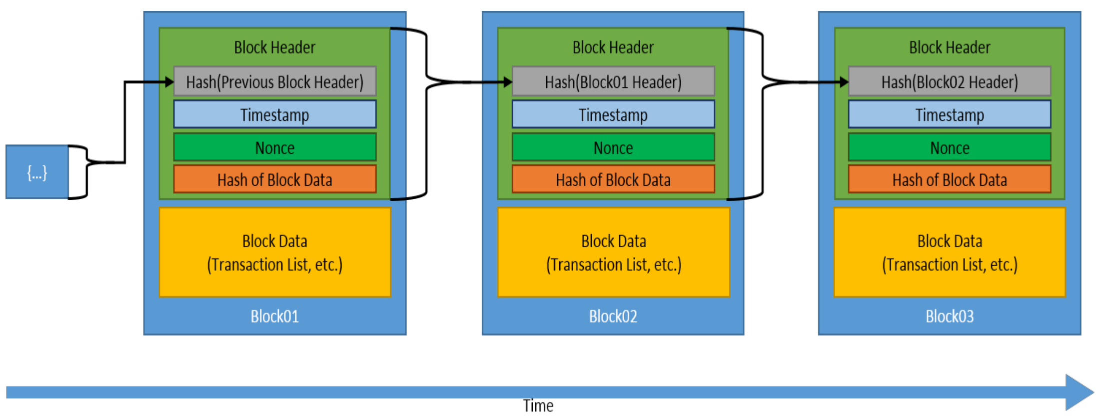

Figure 2.1: Blockchain Diagram (NIST, 2021)

Satoshi Nakamoto was responsible for the first whitepaper that illustrated Blockchain as being equipped to enhance digital trust through decentralization, meaning that there would not be any single entity who could control the Blockchain. This is the same technology that powers over 8,000 cryptocurrencies with a current total market capitalization of over $1.5 trillion and daily transaction volume of over $120 billion (CoinMarketCap, 2021). The two types of Blockchain technology that will be discussed throughout this paper are permissionless and permissioned.

Permissionless Blockchain

Within a permissionless blockchain anyone can read and write to the blockchain without authorization. Permissionless blockchains are usually open-source and maintained by the public (NIST, 2021). The public is able to maintain permissionless blockchain networks by contributing resources such as computing power to the network (mining) or through staking tokens within a network. These two processes can be referred to as 'reaching consensus'.

According to the Blockchain Council, there are three primary characteristics that are associated with permissionless blockchains: Digital Assets. Transparency, and Decentralization. The cryptocurrencies known as Bitcoin and Ethereum exist on top of two of the most notable permissionless blockchain networks.

Benefits. Permissionless Blockchains can be beneficial to implement if the entity is seeking to implement Blockchain to provide transparency. The data transmitted across a permissionless blockchain should never be highly-sensitive or classified but rather unclassified. Another major benefit of utilizing a permissionless blockchain is that “network changes of any type can only be achieved if 51% of the users agree to it” (Sharma, 2019).

Drawbacks. Permissionless Blockchains contain a few drawbacks depending on the entity seeking to implement. The first drawback is that security of the Blockchain is in the hands of the members of the Blockchain. The main security risk is the likelihood that the network can fall victim to a 51% attack. Although, users on permissionless blockchains can utilize the blockchain through public and private keys. The transparency that exists allows for the transaction history to be tracked and therefore less anonymous in nature.

Permissioned Blockchain

Within a permissioned blockchain, authorization is required to be able to read and write to the blockchain. Permissioned blockchains are ideal for use within organizations since they allow for data to be exchanged privately within the corporate network. The three primary characteristics of permissioned blockchains are: Transparency and Anonymity, Varying Decentralization, and Governance (Sharma, 2019).

For example, assigned users are able to grant read and write permissions to individuals or groups for data that is located on the blockchain through utilizing Microsoft Azure Active Directory combined with Azure Blockchain. This project will illustrate a secure implementation of Microsoft Azure Blockchain alongside Microsoft Azure Active Directory and other Azure components.

Benefits. Permissioned Blockchains can be beneficial to implement if the entity is seeking to customize their blockchain or perhaps enforce access controls on all or certain portions of the blockchain. Another benefit that is provided by a permissioned blockchain is that it is able to scale efficiently. Permissioned blockchains are ideal for organizations to implement for internal use cases such as sharing information with auditors or vendors, or perhaps amongst trusted partners.

Drawbacks. Permissioned Blockchains have drawbacks to them such as that authority figures that exist in the blockchain could cause great damage if they choose to go rogue or if their account is compromised. It is possible that a single individual could shutdown or even delete a permissioned blockchain that exists within organizational boundaries. This is considered to be a primary drawback that exists in permissioned blockchains since consensus is not enforced.

CHAPTER THREE

BLOCKCHAIN APPLIED INCYBERSECURITY

The CIA Triad

Blockchain has the potential to improve upon the CIA triad by ensuring the confidentiality, integrity, and availability of any piece of information that is on it. It is important to be aware of the fact that a public blockchain would not be suitable for ensuring the confidentiality of information, but rather the transparency. However, a permissioned Blockchain could be tailored to ensure the confidentiality of data by allowing individuals to access specific sections of data on the Blockchain based on their permissions. The integrity of the data stored in a Blockchain can be upheld since data cannot be edited/removed once it is in the Blockchain. New transactions could occur on the Blockchain but they would not be able to be deemed valid by all of the existing nodes on the Blockchain. The availability of data in a Blockchain could be guaranteed since it would be located on every node that exists within the chain. There would not be a single-point of failure.

Mandatory Access Control

Mandatory access control (MAC) is best defined as a method on limiting the access of an object from a subject, based on the sensitivity of the data along with a need-to-know requirement (Hyun, 2020). Blockchain technology is capable of enforcing MAC either directly or indirectly. Blockchain technology can enforce MAC directly through containing embedded user access levels. Each user who will have access to the Blockchain will contain an attribute that defines their usertype (Umberhocker et al., 2020). Access control can be guaranteed by the user types that will have to correlate with the transactions on the Blockchain. All users may have access to the blockchain, however, a smart contract that defines each usertype field and their varying access levels is capable of enforcing MAC on objects when the user attribute of an object matches the usertype field in a user profile, resulting in the condition of the smart contract being met(NIST, 2021). The final result is that the user is permitted to access only the transactions that they need to perform their duties.

CHAPTER FOUR

PROTOTYPE BUILT ON AZURE

Microsoft Azure was used to deploy the Blockchain within a virtual environment. The virtual environment consists of: a virtual network, a single transaction node along with an assigned public IP address that the Blockchain can be reached by invited users. The main component of the Blockchain prototype in Microsoft Azure is called the Azure Blockchain Service. The Blockchain Service is what contains the applications that run on Blockchain within the organization. The initial configuration of the Blockchain Service is below:


```
{
    "location": "eastus",
    "name": "myblockchainejibk6bl",
    "kind": "Quorum",
    "properties": {
        "validatorNodesSku": {
            "capacity": 1
        },
        "userName": "myblockchainejibk6bl",
        "password": null,
        "consortium": "myblockchainejibk6co",
        "consortiumRole": "ADMIN",
        "consortiumMemberDisplayName": "myblockchainejibk6bl",
        "consortiumManagementAccountAddress": "0x2e752e750ca575d004fd4800df6e93f2eb14e155",
        "consortiumManagementAccountPassword": null,
        "firewallRules": [
            {
                "ruleName": "OpenAll",
                "startIpAddress": "0.0.0.0",
                "endIpAddress": "255.255.255.255"
            }
        ],
        "rootContractAddress": "0xb255f55e8d600f09ebc1035dd2118acec1018912",
        "publicKey": "W0+1ieDLSHHnm6rWUEuyIGNUlxgGv7VWGH2ixPKUVTE=",
        "nodeProvisioningState": "Succeeded",
        "provisioningState": "Succeeded",
```

```
        "dns": "myblockchainejibk6bl.blockchain.azure.com",
        "protocol": "Quorum"
    },
    "type": "Microsoft.Blockchain/blockchainMembers",
    "id": "/subscriptions/cdbf8abb-5960-4538-9674-
66b1889bb81c/resourceGroups/MyBlockchain/providers/Microsoft.Blockchain/blockchainMembers/myblockchain
ejibk6bl",
    "tags": null,
    "sku": {
        "name": "B0",
        "tier": "Basic"
    }
}
```


Azure Active Directory Configuration

After deploying the Blockchain Service, a Cloud PowerShell session was used to invoke the integration of Azure Active Directory with the Blockchain Service through an application known as the Azure Blockchain Workbench. This allows for authentication to be enforced through Active Directory, as well as any policies and permissions that are in place.

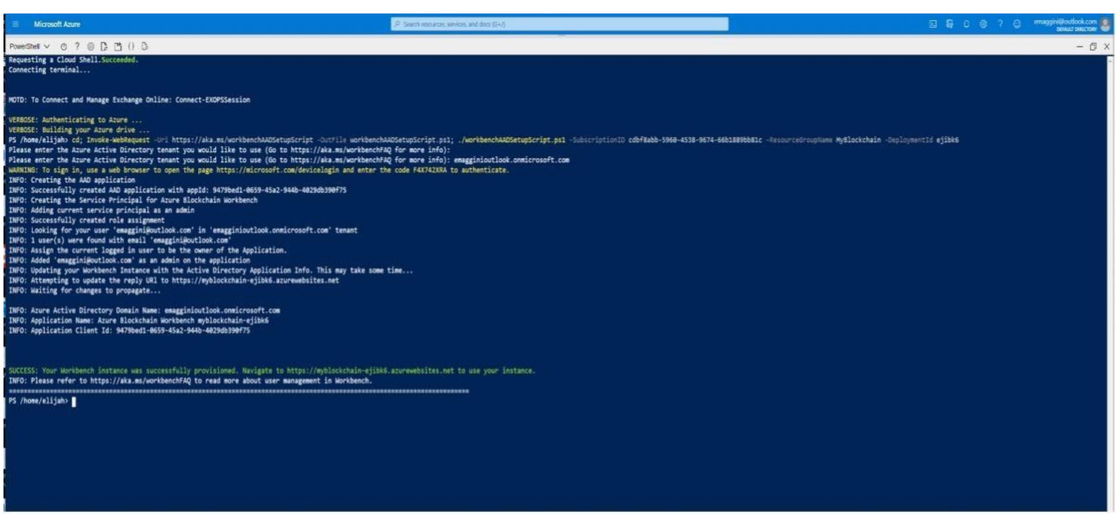

Figure 4.1: Azure Active Directory Powershell Configuration

<u>Users and Groups</u>

Azure Active Directory was integrated into the Blockchain for the purpose of authentication. There are three users and four groups that were used in the configuration of the Prototype Blockchain Application:

Name	User principal name	User type	Directory synced	Identity issuer
EM Elijah Maggini	emaggini_outlook.com#EXT#@emagginioutlook.onmic...	Member	No	emagginioutlook.onmicrosoft.com
U1 User 1	User1@emagginioutlook.onmicrosoft.com	Member	No	emagginioutlook.onmicrosoft.com
U2 User 2	User2@emagginioutlook.onmicrosoft.com	Member	No	emagginioutlook.onmicrosoft.com

Name	Object Id	Group Type	Membership Type
BU Blockchain Users	789f051f-fbb5-4a3d-a668-d684a64d1622	Security	Assigned
EU External Users	f53dcdef-e080-46e2-8e1c-d260dc3db698	Microsoft 365	Assigned
HR Human Resources	3861613b-9f83-4eb0-9222-0c81f937dbb4	Microsoft 365	Assigned
S Sales	625dcf7c-99e3-4f13-bd2c-1fca6207020d	Microsoft 365	Assigned

Figure 4.2: Users and Groups

Configuring the Blockchain Application

The two configuration files that need to be uploaded to Azure to create the Blockchain Application are the JSON and SOL files shown below.

<u>UserAccessRequest.json</u>

```
{
  "ApplicationName": "UserAccessRequest",
  "DisplayName": "User Access Request",
  "Description": "A simple application to request and respond to user access requests",
  "ApplicationRoles": [
    {
      "Name": "Requestor",
      "Description": "A person sending a request."
    },
    {
      "Name": "Responder",
      "Description": "A person responding to a request"
    }
  ],
  "Workflows": [
    {
      "Name": "UserAccessRequest",
      "DisplayName": "Request Response",
      "Description": "A simple workflow to request and respond to user access requests.",
      "Initiators": [ "Requestor" ],
      "StartState": "Request",
      "Properties": [
        {
          "Name": "State",
          "DisplayName": "State",
          "Description": "Holds the state of the contract.",
          "Type": {
            "Name": "state"
          }
        },
        {
          "Name": "Requestor",
          "DisplayName": "Requestor",
          "Description": "A person sending a request.",
          "Type": {
            "Name": "Requestor"
          }
        },
        {
          "Name": "Responder",
          "DisplayName": "Responder",
          "Description": "A person sending a response.",
          "Type": {
            "Name": "Responder"
          }
        },
        {
```

```
            "Name": "RequestMessage",
            "DisplayName": "Request Message",
            "Description": "A request message.",
            "Type": {
              "Name": "string"
            }
          },
          {
            "Name": "ResponseMessage",
            "DisplayName": "Response Message",
            "Description": "A response message.",
            "Type": {
              "Name": "string"
            }
          }
        ],
        "Constructor": {
          "Parameters": [
            {
              "Name": "message",
              "Description": "...",
              "DisplayName": "Request Message",
              "Type": {
                "Name": "string"
              }
            }
          ]
        },
        "Functions": [
          {
            "Name": "SendRequest",
            "DisplayName": "Request",
            "Description": "...",
            "Parameters": [
              {
                "Name": "requestMessage",
                "Description": "...",
                "DisplayName": "Request Message",
                "Type": {
                  "Name": "string"
                }
              }
            ]
          },
          {
            "Name": "SendResponse",
            "DisplayName": "Response",
            "Description": "...",
            "Parameters": [
```

```
      {
        "Name": "responseMessage",
        "Description": "...",
        "DisplayName": "Response Message",
        "Type": {
          "Name": "string"
        }
      }
    ]
  }
],
"States": [
  {
    "Name": "Request",
    "DisplayName": "Request",
    "Description": "...",
    "PercentComplete": 50,
    "Value": 0,
    "Style": "Success",
    "Transitions": [
      {
        "AllowedRoles": ["Responder"],
        "AllowedInstanceRoles": [],
        "Description": "...",
        "Function": "SendResponse",
        "NextStates": [ "Respond" ],
        "DisplayName": "Send Response"
      }
    ]
  },
  {
    "Name": "Respond",
    "DisplayName": "Respond",
    "Description": "...",
    "PercentComplete": 90,
    "Value": 1,
    "Style": "Success",
    "Transitions": [
      {
        "AllowedRoles": [],
        "AllowedInstanceRoles": ["Requestor"],
        "Description": "...",
        "Function": "SendRequest",
        "NextStates": [ "Request" ],
        "DisplayName": "Send Request"
      }
    ]
  }
]
```

Figure 4.3: User Access Request JSON File

UserAccessRequest.sol

```
pragma solidity >=0.4.25 <0.6.0;
contract UserAccessRequest {

//Set of States
    enum StateType { Request, Respond}

    //List of properties
    StateType public  State;
    address public  Requestor;
    address public  Responder;

    string public RequestMessage;
    string public ResponseMessage;

    // constructor function
    constructor(string memory message) public
    {
        Requestor = msg.sender;
        RequestMessage = message;
        State = StateType.Request;
    }

    // call this function to send a request
    function SendRequest(string memory requestMessage) public
    {
        if (Requestor != msg.sender)
        {
            revert();
        }

        RequestMessage = requestMessage;
        State = StateType.Request;
    }

    // call this function to send a response
    function SendResponse(string memory responseMessage) public
    {
        Responder = msg.sender;

        ResponseMessage = responseMessage;
        State = StateType.Respond;
    }
}
```

Figure 4.4: User Access Request Solidity File

Demonstration of the UserAccessRequest Blockchain Application

Overview

There are two users, user 1 and user 2. User 1 is a manager submitting a request for their new employee to be granted access to the HR folder located on a Network Attached Storage server. User 2 is an IT administrator with privileges to grant access to files. The UserAccessRequest Blockchain Application timestamps and hashes the request and the response ensuring non-repudiation. Confidentiality is ensured through the permissioned architecture of Azure Blockchain since User 1 and User 2 are required to authenticate through Azure Active Directory before being able to access the UserAccessRequest application. The two users additionally must possess permissions to read and write to the Blockchain.

Process

Upon navigating to https://myblockchain-ejibk6.azurewebsites.net/, User 1 and User 2 must first enter their username and password before accessing the application. Following authentication, they are greeted with the home screen below that contains their Blockchain Application(s).

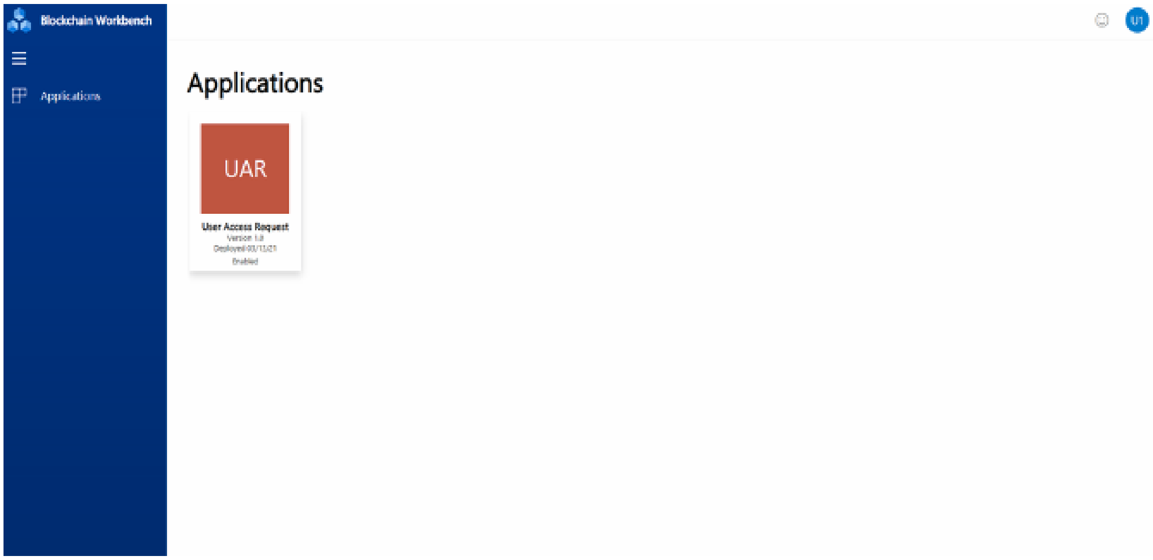

Figure 4.5: Blockchain Workbench Homepage

After clicking on their application, they arrive at a page that allows for them to view details of the smart contract transaction that they invoked by submitting a request or a response or to create a new request.

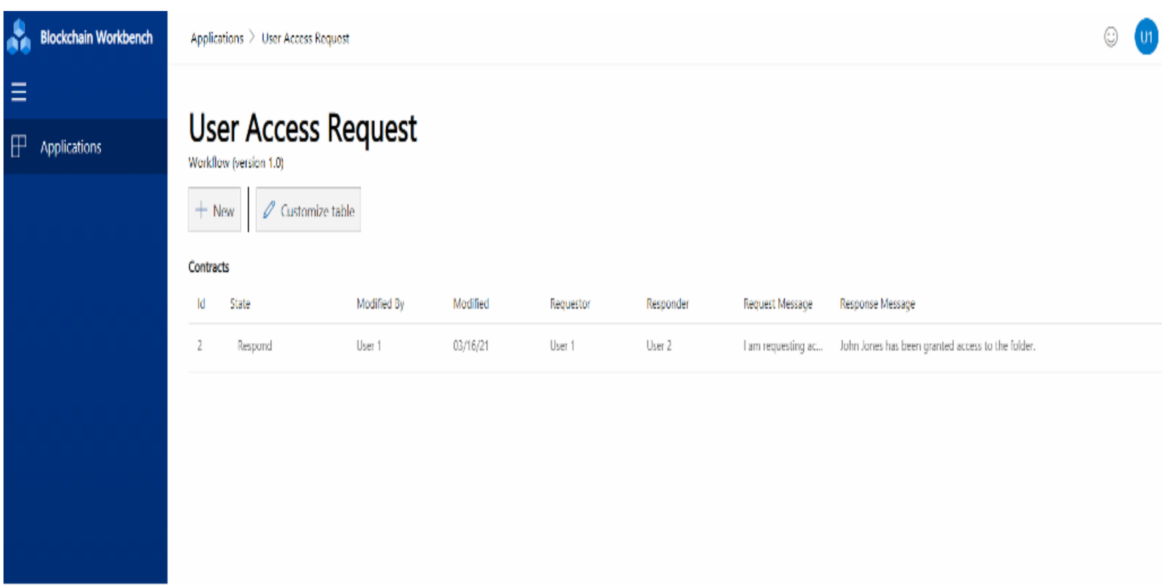

Figure 4.6: User Access Request Application Homepage

After clicking on the ‘New’ button, a window titled “New Contract” appears, allowing for the user to input their request.

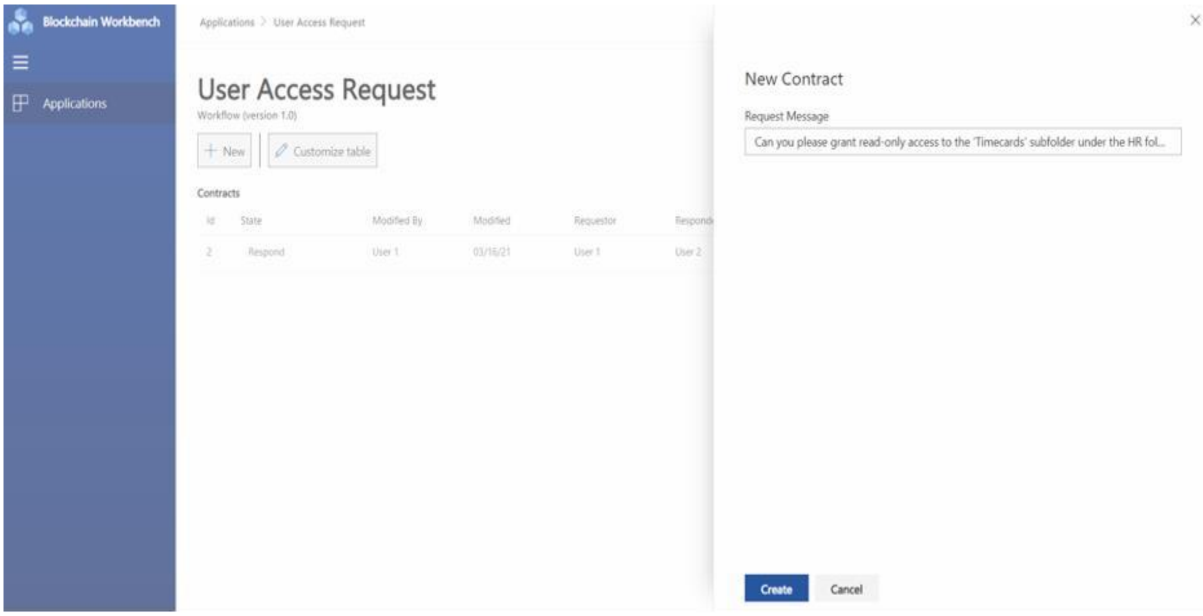

Figure 4.7: New Contract with Request Message

Upon the creation of a request, the user is able to view the status, timestamp, state of the contract, the contract address, and the initial message. Since User 1 is the user who initiated the request, User 2 will login to the application and see that they have a pending action to respond to.

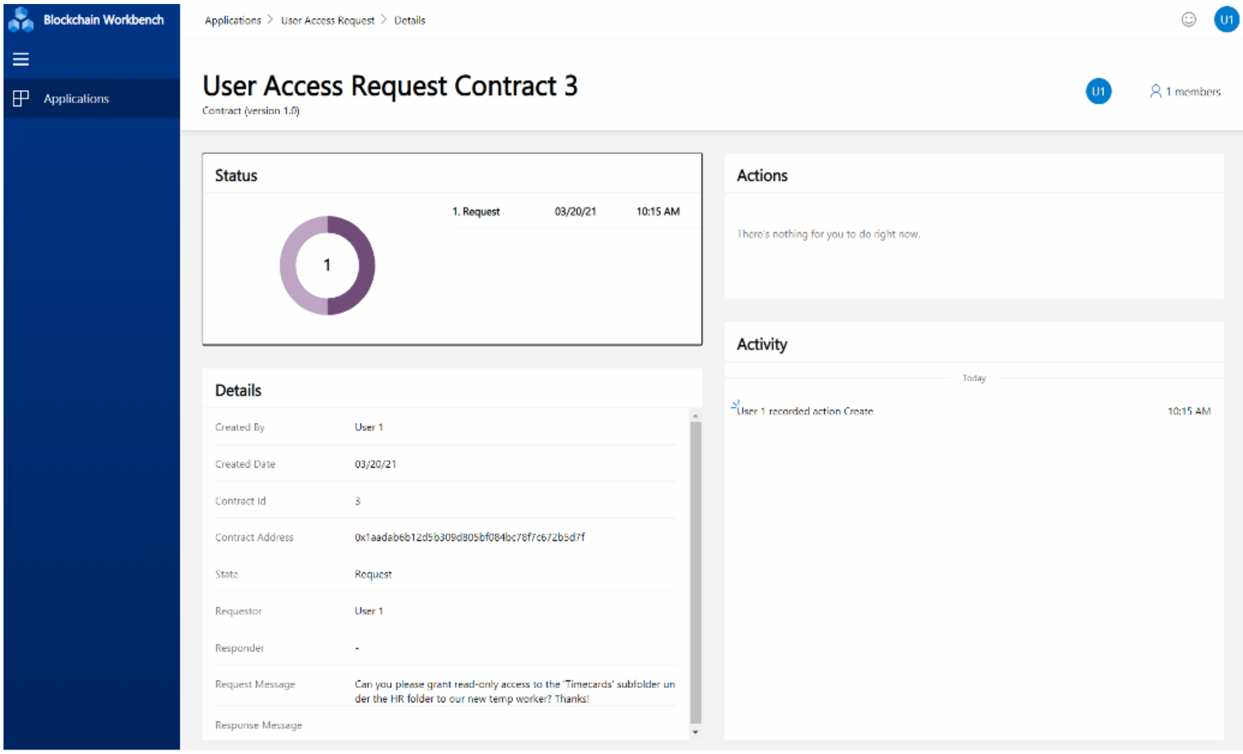

Figure 4.8: Details of User Access Request Transaction

CHAPTER FIVE

BLOCKCHAIN APPLIED IN THE SUPPLY CHAIN

Supply Chain Transparency

Supply Chain Transparency is defined as organizations being aware of what is happening upstream in the supply chain and communicating that knowledge internally and externally (Bateman & Bonanni, 2019). Supply Chain Transparency has become an increasingly important topic within the past decade. According to an article published in the Harvard Business Review, organizations are being pressured to reveal information regarding their supply chains for reasons such as: treatment of workers and proper sourcing of ingredients or materials (Bateman & Bonanni, 2019). Blockchain technology has all of the characteristics that would allow for organizations to accurately track and disclose assets that travel through their supply chain(s). IBM describes its Blockchain technology as being able to “Authenticate product origins” and “Trace inventory throughout the supply chain in near real time” (IBM, 2021).

Prototype Built on SIMBA

SIMBA was used to develop a prototype Blockchain application that is capable of verifying the integrity of COVID19 vaccines as they journey through the supply chain. This is possible through the composition of a smart contract that is constantly requesting for sensors to respond with the temperature at which the vaccine is being stored at. The requests and responses will be stored on a Blockchain as transactions.

Designing the Smart Contract

The smart contract that powers the prototype Blockchain application was designed as follows:

1. Input “Vaccine” as an asset on the SIMBA graph. Input “retrieveTemperature” as a transaction on the SIMBA graph. Then proceed to illustrate their asset to transaction relationship by connecting them together.

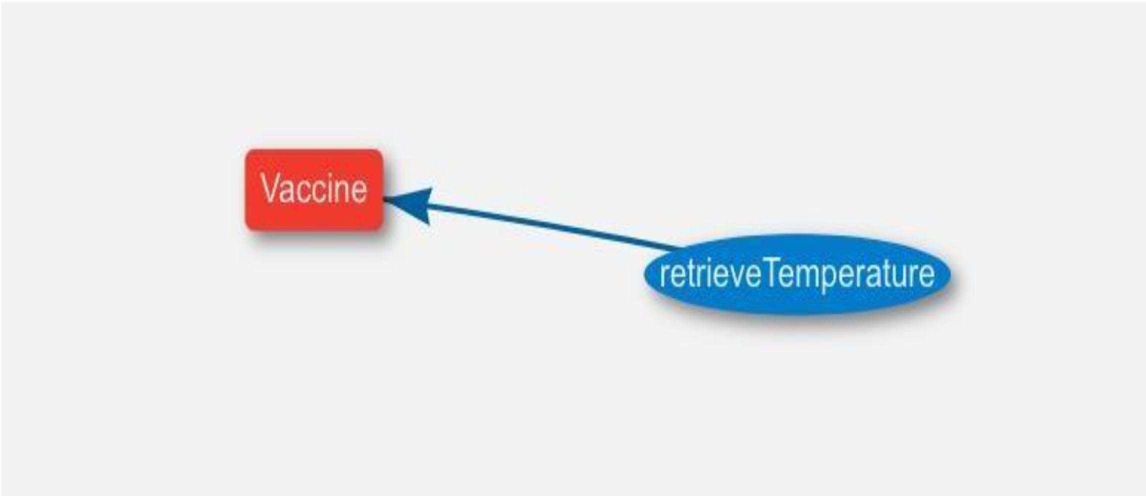

Figure 5.1: Graph Illustrating Vaccine Temp Smart Contract

2. After setting up the graph, click on the asset and transaction to input their attributes.

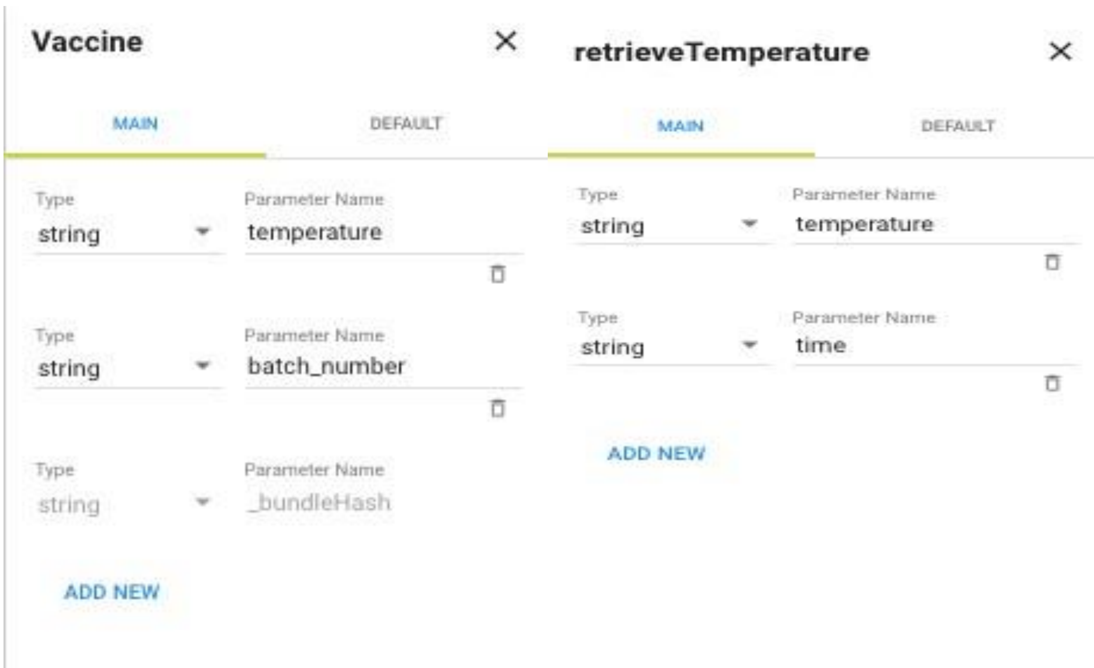

Figure 5.2: Inputting Attributes of Assets in SIMBA

3. After saving the attributes, click on “<>” at the bottom right of the screen to view the code of the smart contract.

```
pragma solidity ^0.5.12;

contract Application {

    constructor() public {}


    function Vaccine (
        string memory temperature,
        string memory batch_number,
        string memory _bundleHash,
        string memory __Vaccine
    )
    public {
    }

    function retrieveTemperature (
        string memory temperature,
        string memory time,
        string memory __Vaccine
    )
    public {
    }
}

```

Figure 5.3: Toggle from Graph to Code View in SIMBA

4. Click on the cloud icon on the bottom left of the screen to save the newly created smart contract

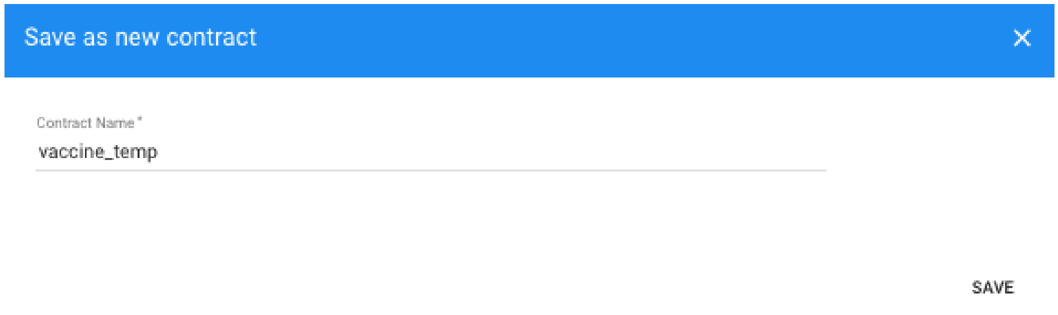

Figure 5.4: Saving the Smart Contract

Deploying the Smart Contract

The smart contract "vaccine_temp" was deployed on the Quorum Testnet as follows:

1. Click on the "application" tab on the SIMBA dashboard and choose your newly created smart contract.
2. Choose the intended Blockchain Type and Network Type.

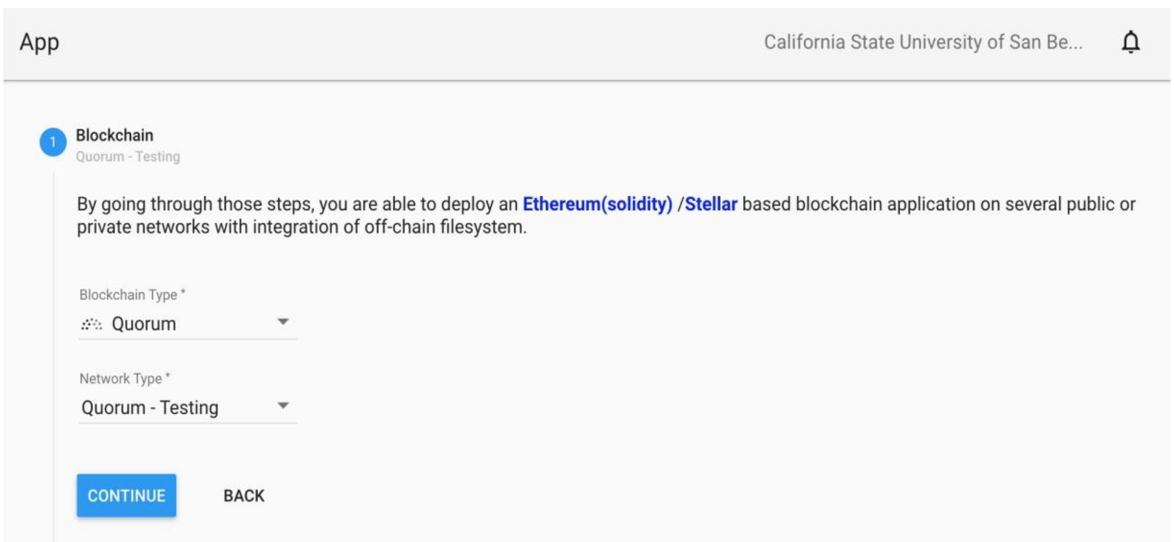

Figure 5.5: Choosing the Intended Blockchain

3. Choose the off-chain filesystem to utilize.

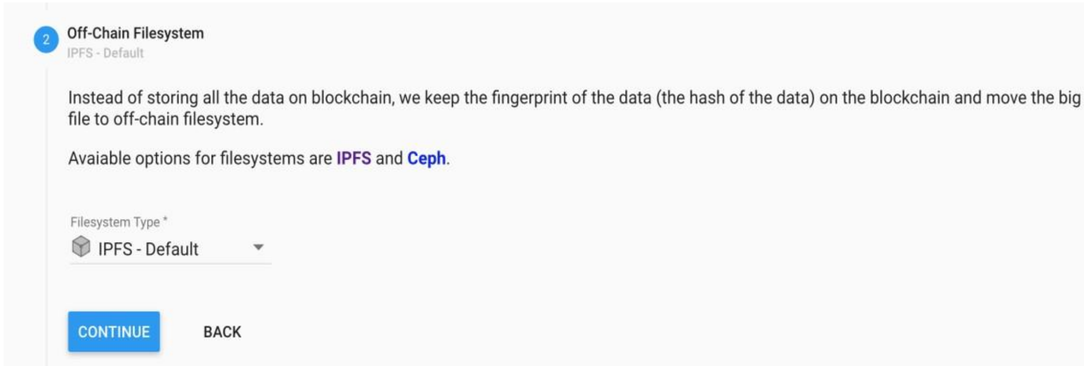

Figure 5.6: Choosing an Off-Chain File System

4. Upload your newly created smart contract to use as the basis of your application.

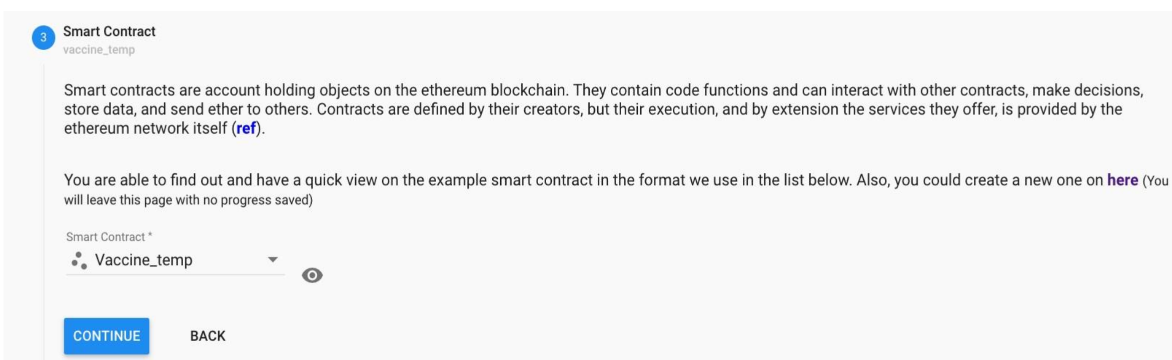

Figure 5.7: Uploading the Smart Contract

5. Next, name the application along with the API appropriately.

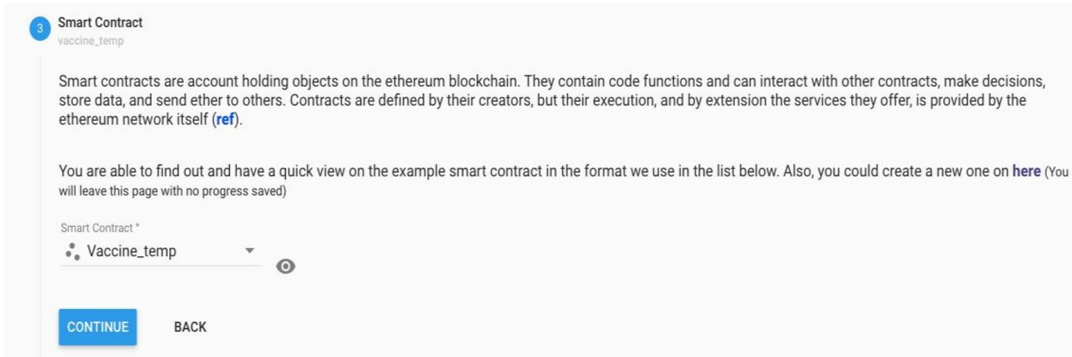

Figure 5.8: Naming the Application and API

6. Lastly, choose a wallet or create one if you do not already have one. Input your password and click “Unlock”. Proceed to click “Deploy” and your application will begin deployment on the Quorum Testnet.

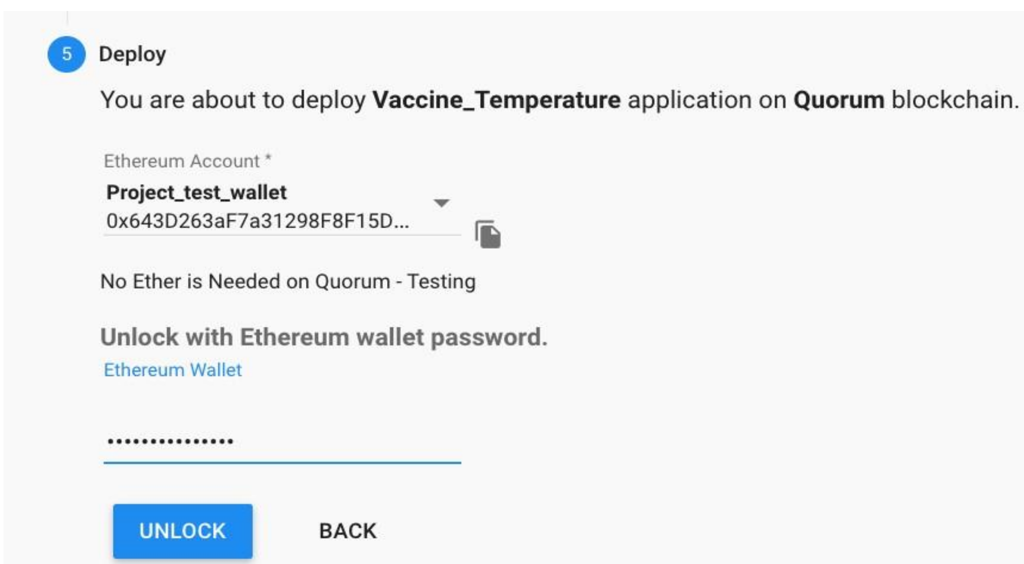

Figure 5.9: Setting the Ethereum Wallet and Deploying the Application

CHAPTER SIX

CONCLUSION

Blockchain technology is capable of providing transparency and security to those who seek it. After conducting research and building the UserAccessRequest blockchain application, I found that Blockchain enhanced the process of submitting requests for access to files. The process is traditionally done via email and ideally stored in a ticketing system. However, Blockchain provided a secure interface that could only be accessible to those who were given permissions in Active Directory. The users who were granted permissions were only able to initiate communication with the Blockchain through inputting a link in their web browser that contained their username and private key for accessing the Blockchain.

After conducting research, I came to the conclusion that there are pros and cons to implementing Blockchain within an organization. Blockchain can provide unrivaled transparency in processes, as well as non-repudiation. Since transactions that are written to a Blockchain cannot simply be deleted and forgotten about, those who utilize Blockchain will be forced to be accountable in terms of the processes that they are a part of. The User Access Request Blockchain application illustrates that there will always be a record of who requested what file with what level of access for whom.

Future Use Cases

This project has demonstrated that there are many use cases for Blockchain on an enterprise and personal level. Blockchain technology's scalability and security are top-tier, allowing for applications to be built on top of a Blockchain and integrated with services such as Active Directory and SQL for authentication and authorization. There is enormous potential for governments to utilize Blockchain in the voting process to ensure integrity of elections. The digital ledger would be able to keep track of votes and ensure that votes could not be thrown out or tampered with once on the Blockchain. Fake votes would not be able to be added to the Blockchain unless they are paired with an existing ID number.

Another use case would be for Blockchain to serve as a platform for recording those who have been administered a vaccine for COVID19 or other viruses that may emerge in the future. For example, a Blockchain mobile application could be programmed and then downloaded by those who have been given the vaccination. The application will contain a QR code that serves as the individual's unique public key and contain PII as well as their vaccination date and type. Airports, concert venues and other locations that host large gatherings would be able to scan the QR code on an individual's device and determine their vaccination status and then proceed to grant or deny access to the location.

These are just a couple of use cases out of many for Blockchain applications. The ability to customize applications on the Blockchain and integrate those applications alongside other applications and within various environments allows for endless opportunity to scale Blockchain and tailor it to solve problems and improve upon traditional processes within organizations and so much more.

APPENDIX A

GLOSSARY

Term	Definition
Decentralization	the transfer of control and decision-making from a centralized entity (individual, organization, or group thereof) to a distributed network.
Consensus	a fault-tolerant mechanism that is used in computer and blockchain systems to achieve the necessary agreement on a single data value or a single state of the network among distributed processes or multi-agent systems
Smart contract	a self-executing contract with the terms of the agreement between buyer and seller being directly written into lines of code
Active directory	a database and set of services that connect users with the network resources they need to get their work done
Powershell	a cross-platform task automation solution made up of a command-line shell, a scripting language, and a configuration management framework
Off-Chain File system	A database or dedicated file system that stores data outside of the blockchain
Transaction	An event that occurs on a blockchain.

REFERENCES

Altimore, P. (2019, November 22). Configure Azure Active Directory access - Azure Blockchain Service - Azure Blockchain. Retrieved March 24, 2021, from https://docs.microsoft.com/en-us/azure/blockchain/service/configure-aad

Bateman, A., & Bonanni, L. (2021, January 20). What Supply Chain Transparency Really Means. Retrieved April 2, 2021, from https://hbr.org/2019/08/what-supply-chain-transparency-really-means

Hyun, Jake Kim, "Multi-Level Security (MLS) Policy Implementation Using Graph Database" (2020). Electronic Theses, Projects, and Dissertations. 1126. https://scholarworks.lib.csusb.edu/etd/1126

Materese, R. (2021, January 15). Blockchain. Retrieved February 6, 2021, from https://www.nist.gov/blockchain

M., L. (2021, February 5). What is a Smart Contract and How do Smart Contracts Work. Retrieved March 20, 2021, from https://www.bitdegree.org/crypto/tutorials/what-is-a-smart-contract

Panossian, G. (2019). Multi-level secure data dissemination (Master’s thesis). California State University, San Bernardino. https://scholarworks.lib.csusb.edu/etd/946/

Sharma, T. K. (2019, November 13). Permissioned and Permissionless Blockchains: A Comprehensive Guide. Retrieved April 3, 2021, from https://www.blockchain-council.org/blockchain/permissioned-and-permissionless-blockchains-a-comprehensive-guide/

SIMBA: Badges and Certificates User Guide. (2021, January 05). Retrieved March 26, 2021, from https://simbachain.com/wp-content/uploads/guides/BadgesAndCertificates.pdf

Umberhocker, A., & Porutiu, H. (2020, March 26). Use access control in your blockchain smart contracts to streamline supply chain operations. Retrieved March 18, 2021, from https://developer.ibm.com/technologies/blockchain/patterns/fabric-contract-attribute-based-access-control

How Blockchain technology is revolutionising Fintech in 2020. (2020, September 18). Retrieved February 6, 2021, from https://www.fintechnews.org/how-blockchain-technology-is-revolutionising-fintech-in-2020-2/