SETU Carlow

HyperLedger Fabric

Auctioning System – Research Document

Student - Oisin Hickey – C00247185

11-25-2022

Abstract

The popularity of blockchain technologies in the modern-day world has led to increased implementation throughout modern-day life. From DRM protection in video games to NFTs being rapidly bought and sold, Blockchain is a prevalent and secure method to do business. In this project, I will aim to implement a Hyperledger-based auction system.

The system allows users to bid on an item, publicly revealing the bidding once the auction completes. All of this can be controlled and managed via HyperLedger chaincode. The application utilises node.js, go, and Docker to host its virtual test server. The system allows for cross-platform secure auctioning in a simplistic and straightforward application for the user.

Table Of Contents & Figures

1. Introduction

To fulfill the brief, the project will include creating software on a blockchain platform. In this instance, Hyperledger Fabric is the proposed solution by the brief.

Users will be able to sell and buy products by bidding using a GUI to communicate with a blockchain network. When an auction is over, the highest bid wins. Other users' bids will be hidden from peers until the auction ends, a functionality of which HyperLedger can provide due to its permissioned nature.

The project will also use Node.js, Fabric SDK, and Docker as other technologies. The business logic for the Fabric network will be provided via smart contracts or chaincode in Hyperledger lingo, which may be developed in JavaScript, Java, or Go. Fabric components are hosted as Docker image services. Good troubleshooting abilities and some familiarity with Linux administration are required for this project.

To assist beginning developers, Hyperledger Fabric offers several step-by-step guides, sample smart contracts, and apps, which I will utilise fully to prepare myself for this project.

2. Blockchain

This section will provide an overview and explanation of blockchain technologies. It will begin with a history, followed by a description of the components and architecture of a blockchain, before going into more detail about how it is used for cryptocurrencies. (Blockchain - Wikipedia, n.d.)

Blockchain technology has rapidly spread in recent years and is widely used and accepted for many use cases. Some of the most popular blockchain's are Bitcoin and Ethereum.

Blockchain technology provides transparency because every transaction is recorded in a public ledger. Industries that use blockchain technology include:

- Financial services
- Supply chain management
- Real estate
- Insurance industry
- Healthcare field
- Entertainment industry

("Blockchain Use Cases in 2021: Real World Industry Applications | ConsenSys," n.d.)

2.1 History

Most blockchain experts credit its conception to Stuart Haber and W. Scott Stornetta, who envisioned the first cryptographically secured and tamperproof file chain. In 1992 they implemented Merkle trees in their system, further expanding functionality and security.(“Blockchain - Wikipedia,” n.d.)

Officially, Blockchain has been around since 2009 but was not fully defined until 2013 when Satoshi Nakamoto released his white paper titled 'Bitcoin: A Peer-to-Peer Electronic Cash System'.(“Bitcoin - Wikipedia,” n.d.)

The original Bitcoin software was released as open-source software that could be freely used, contributing to the spread of the cryptocurrency. (“Bitcoin - Wikipedia,” n.d.) Once sent, these bitcoins would be recorded on a ledger and available for anyone to see. So, when you sign up for the network using your public and private keys, you record that initial transaction. Then, all the blockchain computers verify that transaction so everyone knows it is legitimate, allowing diverse types of information to be stored on a blockchain, including money and personal messages. (Nakamoto, n.d.)

Satoshi Nakamoto has not been identified or verified by any official sources.
While confirming whether Satoshi Nakamoto is a person is difficult, some have hypothesised that they may be individuals. Some sources claim to have identified him, but no one has ever been confirmed. (“Satoshi Nakamoto - Wikipedia,” n.d.) As of this research documents writing he has vanished.

In 2010 the first purchase of Bitcoin occurred, amounting to 10,000BTC (Bitcoin Currency), and by 2013 the Bitcoin marketplace surpassed $1 billion in value. However, something much more significant than Bitcoin was about to appear in the blockchain space. (“Blockchain Technology History: Ultimate Guide,” n.d.)

The first use of blockchain technology was Bitcoin, but an even more successful application was Ethereum. Bitcoin was just a digital currency (“Ethereum - Wikipedia,” n.d.), but Ethereum allowed developers to create decentralised apps (Dapps) with its platform-specific cryptographic token, Ether. Ethereum was developed in 2013 by Vitalik Buterin, who felt Bitcoin could have been more cutting-edge and applied. (“Blockchain Technology History: Ultimate Guide,” n.d.) However, it is worth noting that the Ethereum platform is built on a modified version of the Bitcoin network that allows people to create decentralised applications. Smart contracts for the Ethereum network are written in a JavaScript like language called solidity.

These Dapps allow users to interact in ways that were not possible before with other apps, such as creating their own game or a social network. Ethereum is like Bitcoin in that it has a public blockchain, but instead of being used as a digital currency, Ethereum introduced something called Smart Contracts. (“Ethereum - Wikipedia,” n.d.)

Smart Contracts are a way to write code that executes automatically when certain situations occur. This can be used for more complex applications where the execution of code is essential. For example, if I am sending you money, I could put in my address and the amount I want to

send you and then when I hit 'send', it would send the money directly from my bank account to your account without me having to sign anything manually. The other advantage of smart contracts is that they can also be backed up by law or executed by courts. (“What are smart contracts on blockchain? | IBM,” n.d.)

In 2014 saw, the Ethereum blockchain was crowdfunded in full. This also was the year that R3 technology was formed. R3 technology allows for the secure and regulated use of blockchain technologies, allowing companies like crypto.com, revolut and trading212 to engage in cryptocurrency trading and transactions. (“Blockchain Technology History: Ultimate Guide,” n.d.)

In 2015 Ethereum was officially launched and has become one of, if not the biggest, blockchain technologies to provide smart contracts and various functions. That same year its genesis block became active and resulted in the platform taking off. (“Blockchain Technology History: Ultimate Guide,” n.d.)

This was also the year that the Linux foundation unveiled Hyperledger.

ZCash was launched in October 2016 and was the first cryptocurrency that allows users to send money anonymously. The issue with Bitcoin was that it did not allow for anonymous transactions. Instead, each account was linked to a public key, making it easy for anyone to see who sent money where. ZCash is another coin released based on Bitcoin, but some improvements allow for more private transactions. This would benefit the banking industry, making it harder for people to steal or fraudulently take money from others. It would also allow for more secure processing of legal contracts.(“Zcash - Wikipedia,” n.d.)

Zcash uses zero-knowledge proofs to allow users to make anonymous transactions. The transmitted information is obscured when users send money with Zcash, so it is not apparent who sent or received it. However, whenever someone sends money with Zcash, there is an option where they can choose to hide the amount of money that was sent. (“Zcash’s Zero Knowledge Proofs, ZK Snarks, and More | Gemini,” n.d.)

One of the first developments on Ethereum was creating the Cryptokitties game by Axiom Zen in 2017. This involved creating a virtual world where unique digital cats could be created and traded on a blockchain. The contract for this game was coded on Ethereum's platform and executed automatically when certain things happened, such as when someone bought one of these digital cats. In addition, it uses zero-knowledge proofs to ensure that your identity is hidden from everyone but the person you are sending money. (“CryptoKitties - Wikipedia,” n.d.)

In 2017, the number of projects related to Blockchain, or cryptocurrency, had increased by 2,845 per cent since 2012, having a market cap increase of 4.7 billion USD. In addition, in November 2017, the number of GitHub projects related to Blockchain had increased by more than three hundred per cent since January 2016, with most programming languages represented.("Blockchain Technology History: Ultimate Guide," n.d.)

2.2 Blockchain Architecture

Blockchain is a growing list of records (called blocks) linked using cryptography and peer-to-peer networking. Each block contains a cryptographic hash of the previous block, a timestamp and transaction data. All transactions are recorded in one ledger and sent to each node in the decentralised network. (“Blockchain Facts: What Is It, How It Works, and How It Can Be Used,” n.d.)

Removing old or incorrect information from the Blockchain is possible, although it is extremely expensive to add fake information it is not impossible. To understand how Blockchain works and is used, we will first look at some common traits of blockchains, their architecture, their components and how those work together in blockchain technologies.

The main components of blockchain technologies are:
- Blockchain network nodes
- Consensus algorithms
- Smart contracts
- Digital Assets
- Data management platforms
- P2P Networks
- Hashing Algorithims
- Types of Blockchain

2.2.1 Blockchain network nodes

The builders of a decentralised network's architecture are, in many ways, blockchain nodes. Their foremost duty is to uphold the public ledger's consensus, which differs according to the type of ledger or data. (“What is Blockchain? | Oracle Middle East Regional,” n.d.)

A node's primary operational functions are maintenance, validation, and accessibility.

They maintain consistency across all copies of the ledger and store encrypted information about previous transactions while accepting new blocks for expansion.

Any device with an IP address, including routers, modems, switches, hubs, servers, and printers, can act as a node. Surfing the internet is like connecting to a blockchain network. Nodes perform the same task as a browser: they can communicate with other network nodes because they are familiar with the network protocol, security protocols and standards of that network. A node has nothing to lose because it is always free to join or exit the network, with other nodes ensuring its function is still fulfilled and its data is persisted. (“What Are Blockchain Nodes and How Do They Work? | Built In,” n.d.)

A fundamental nexus of nodes power a blockchain in place of a central authority. An elected team of nodes work together to fulfil utility and governance processes, such as verifying transactions and carrying out decision-making procedures, taking democratic control over a network.

2.2.2 Consensus algorithms

Despite the absence of a central authority to confirm and validate the transactions, every Blockchain transaction is considered one hundred per cent safe and validated. (“5 Basic Components of A Blockchain Network,” n.d.)

Only the consensus algorithm, a fundamental component of every Blockchain network, makes this feasible.

A consensus algorithm is a process that allows every peer in the Blockchain network to agree on the distributed ledger's current state. (“5 Basic Components of A Blockchain Network,” n.d.) Consensus algorithms accomplish reliability in the Blockchain network and build confidence between unidentified peers in a distributed computing setting in this way. (“Consensus Algorithms in Blockchain - GeeksforGeeks,” n.d.)

Blockchain consensus algorithms, as an entity, have specific goals, like reaching a consensus, cooperating, giving every node equal rights, and requiring each node to participate in the consensus process. Consensus algorithms allow nodes to be an effective governing body for a blockchain network, as they allow the nodes to have a set of rules to follow in the management of the network. (“What Is a Blockchain Consensus Algorithm? | Binance Academy,” n.d.) An example of these rules would be verifying is the incoming updated chain object valid? Is the chain old or outdated or contradicted by several other nodes copies of that chain?

2.2.3 Smart contracts

To automatically execute, control, or document legally noteworthy events and activities following the provisions of a contract or an agreement, a smart contract is a computer programme or transaction protocol often used in blockchain technologies. (“What are smart contracts on blockchain? | IBM,” n.d.)

The goals of smart contracts are to decrease the need for trustworthy intermediaries, arbitration fees, fraud losses, and malicious and unintentional exceptions by automating a large amount of the processes and tasks involved.

The smart contracts provided by Ethereum are widely regarded as a vital building block for decentralised finance and applications. Smart contracts are frequently linked to cryptocurrencies. (“Smart contract - Wikipedia,” n.d.)

A vending machine is the earliest example of a smart contract implementation technology, as its control flow is similar.

The Bitcoin protocol is described as a weak implementation of the smart contract concept as initially established by Nick Szabo in Vitalik Buterin's original Ethereum white paper from 2014. A more robust implementation based on the Turing complete Solidity language is suggested as an alternative an implemented in Ethereum. (Buterin, n.d.)

Since Bitcoin, several cryptocurrencies have supported scripting languages, enabling the creation of more sophisticated smart contracts between unreliable parties. The best and most general way to view smart contracts is as applications run on blockchain networks.

2.2.4 Genesis block & blocks

Like regular blockchains, Hyperledger uses a Genesis block. The rest of the network and its child blocks are hashed upon this block. Therefore, the need for a Genesis Block is theoretically unnecessary. Nevertheless, creating a foundation everyone can rely on is essential in any blockchain. (“Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation,” n.d.)

The "prior hash" value of the Genesis Block is set to 0. This indicates that prior to the Genesis Block, no data was processed. All subsequent blocks will have numbers beginning with one and a "prior hash" value set to the preceding block's hash. ("What is Genesis Block and why Genesis Block is needed? | by Tecra Space | Medium,” n.d.)

A more common implementation is that each block has a last hash value and the subsequent blocks fill this value with the hash of the previous block. This allows each block to be verified independently of the next but still in an ordered and cryptographically safe manner.

The "Bitcoin Genesis Block" block, Satoshi Nakamoto produced on January 3rd, is the most well-known Genesis block. The 50-bitcoin reward for this block expires after one day. The Genesis Block reward's mining status is unknown, and Satoshi Nakamoto has yet to comment. Below is an image of his block: (“what is Genesis Block and why Genesis Block is needed? | by Tecra Space | Medium,” n.d.)

Figure 1: Bitcoin Origin Block

(“File:Bitcoin-Genesis-block.jpg - Wikimedia Commons,” n.d.)

2.2.5 Digital Assets

Anything uniquely recognised, kept digitally, and used by companies to realise its value is considered a digital asset. For example, documents, music, video, logos, slide shows, spreadsheets, and websites are a few digital assets. Even social media accounts can be digital assets depending on the size of their following. (“Definition of Digital Assets - Gartner Finance Glossary,” n.d.)

The popularity and value of digital assets have increased as technology has grown more pervasive in our personal and professional lives.

A digital asset must first have the capacity to create value by being used in a way that adds value to the owner to qualify as an asset. The digital asset should then be able to transfer ownership by way of a purchase, a gift, or some other method of granting the rights to someone else, as well as the value the thing can add. Additionally, information needs to be findable or stored in a location where it can be located, e.g. a blockchain. (“Digital Asset Definition,” n.d.)

A blockchain specific example of a digital asset could be a particular smart contract, a cryptocurrency token or a digital NFT.

2.2.7 P2P Networks

A distributed application architecture called peer-to-peer (P2P) computing or networking divides jobs or workloads across peers. Peers are equally qualified and capable members of the network. They are referred to as the nodes in a peer-to-peer network. (“Peer-to-peer - Wikipedia,” n.d.)

Without central coordination by servers or reliable hosts, peers make a portion of their resources directly available to other network users, such as processing power, disk storage, or network bandwidth. In contrast to the conventional client-server architecture, where the consumption and supply of resources are segregated and controlled, peers are both resource suppliers and consumers.

Peer-to-peer architecture, divided into structured, unstructured, and hybrid peer-to-peer networks, is appropriate for various use cases. Unstructured peer-to-peer networks are created by nodes connecting at random, but they are less effective than those structured in connection. (“Blockchain & Role of P2P Network | Blockchain Council,” n.d.). They do however have the added advantage of making data transmission harder to trace as peers may leave and join the network regularly.

The nodes are grouped in structured peer-to-peer systems, enabling each node to search the network for the needed data efficiently. Compared to structured and unstructured P2P systems, hybrid models—which combine P2P and client-server models—tend to deliver networks with improved overall performance.

The key distinction between the P2P Network system and Client-Server is that it utilises a specific client base and a dedicated server. (“Blockchain & Role of P2P Network | Blockchain Council,” n.d.)

2.2.8 Hashing Algorithms

Cryptographic hash functions are created via hashing algorithms. These are mathematical formula that convert data of any size into a fixed-size hash.

A hash function algorithm aims to create a one-way function that is impossible to invert, ensuring the encrypted data is secure even during a data leak. Unfortunately, however, several hashing techniques have been hacked recently. For instance, the widely used MD5 hash function, which was intended to be a cryptographic hash function but is now reasonably simple to reverse. As a result, it could only be used to check data for accidental corruption or duplicate files. ("Hashing Algorithms | Jscrambler Blog," n.d.)

In IT, cryptographic hash functions are frequently employed. They can be used for message authentication codes (MACs), digital signatures, and other types of authentications. They can also be used as checksums, fingerprints, file identification, duplicate detection, and indexing data in hash tables.

A notable example of hashing is its use in the "Awesome Duplicate Photo Finder" software. ("Awesome Duplicate Photo Finder - Find and Remove Duplicate or Similar Images," n.d.) When using this program, a user can select search for 100% matches which only works by hashing the files block by block and comparing the resulting hashes to find matches. ("performance - Fastest algorithm to detect duplicate files - Stack Overflow," n.d.)

There are several hashing algorithms which are prudent to blockchain systems:

MD5

MD5 (Message-Digest Algorithm 5) is a widely used cryptographic hash function that produces a 128-bit hash value. It is a one-way function that takes an input message and produces a fixed-size output, which is usually represented as a hexadecimal string. MD5 has been used for a variety of purposes, such as digital signatures, data integrity checks, and password storage. However, due to its vulnerabilities to collision attacks, it has been deprecated and is no longer recommended for use in security-sensitive applications. ("What is MD5 (MD5 Message-Digest Algorithm)?," n.d.)

BIP39

BIP39 (Bitcoin Improvement Proposal 39) is a standard for creating mnemonic phrases for deterministic wallets in Bitcoin and other cryptocurrencies. It provides a way to generate a series of random words from which private keys can be derived. BIP39 enables users to back up and recover their wallets easily, without needing to remember long strings of random characters. Mnemonic phrases are generated using a combination of a predefined wordlist and a random seed, and the resulting phrase is hashed using the SHA256 algorithm to produce a seed that can be used to generate private keys. ("bips/bip-0039.mediawiki at master · bitcoin/bips," n.d.)

SHA256

SHA256 (Secure Hash Algorithm 256) is a widely used cryptographic hash function that produces a 256-bit hash value. Like MD5, it is a one-way function that takes an input message and produces a fixed-size output. SHA256 is considered more secure than MD5 due to its resistance to collision attacks. It is used in a variety of applications, including digital signatures, data integrity checks, and password storage. Bitcoin, for example, uses SHA256 in its proof-of-work algorithm to mine new blocks on its blockchain. (May, 2012)

2.2.9 Types of Blockchain

The four primary varieties of blockchain networks are public blockchains, private blockchains, consortium blockchains, and hybrid blockchains. Each of these platforms has advantages, disadvantages, and ideal applications. ("What are the 4 different types of blockchain technology?," n.d.)

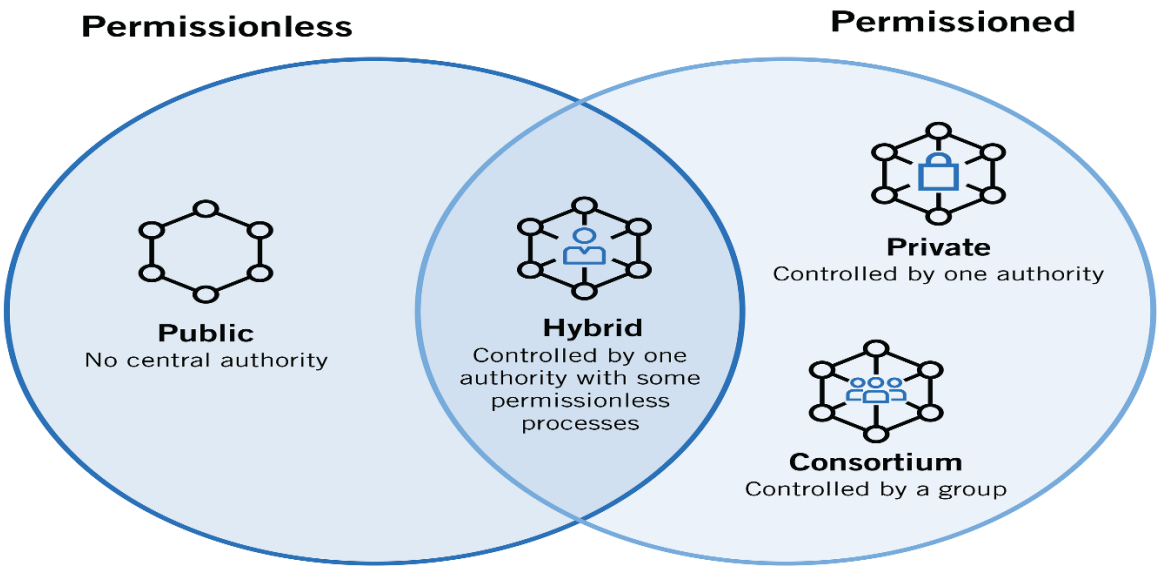

Figure 2: Blockchain Types Venn Diagram

("Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP," n.d.)

Both private and public blockchains have limitations; private blockchains are more susceptible to fraud and bad actors, whereas public blockchains typically offer shorter validation times for new data and are completely open to bad actors and crime. As a result, consortium, and hybrid blockchains were created to overcome these issues.

Public Blockchain Systems

Public blockchains are entirely decentralized, permissionless, and open to everybody. Public blockchains allow all nodes equal access to the Blockchain, to adding new blocks of data, and to validate existing blocks of data. ("Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP," n.d.)

Public blockchains are utilised nowadays for Bitcoin mining and trading. Popular public blockchains are systems like Bitcoin, Ethereum, and Litecoin. The nodes "mine" for coins on these open blockchains by constructing blocks for the network-requested transactions by resolving cryptographic puzzles. The miner nodes receive a small sum of cryptocurrency as payment for their laborious effort. The miners effectively take on the role of modern-day bank tellers who create a transaction and collect (or "mine") payment in exchange. ("Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP," n.d.). Permissionless systems are the most common and most utilized due to their public accessibility.

Private Blockchain Systems

Private blockchains, also known as managed blockchains, are permissioned blockchains under the management of a single company or entity. The central authority decides who is permitted to be a node on a private blockchain. The central authority may only sometimes accord each node an equal right to execute specific responsibilities. ("Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP," n.d.)

Due to restrictions on public access, private blockchains are only partially decentralised. The business-to-business virtual currency exchange network Ripple and the open-source blockchain application framework Hyperledger are an instance of private Blockchain.

Hybrid Blockchain Systems

A hybrid blockchain is frequently described as a fusion of public and private blockchains. It combines key elements of both public and private blockchains. Combining the most significant features of both creates private transactions and data that allow for individual verification of these transactions as needed. Confidential information is preserved within the network but is still verifiable. ("Guide to Hybrid Blockchain, Benefits and Use Cases," n.d.)

Once a user has been granted access to the hybrid blockchain platform, they can participate in its standard operations. They have the same equal privileges to perform, examine, and add or change transactions. However, one item that is kept secret from others is the identity of other users or transactions which an entity has marked specifically to be kept private. The user's privacy is safeguarded in this way. Only the user with whom a user interacts learns the other users' identity throughout a transaction. In industry applications companies can perform KYC (Know Your Customer) procedures to properly make the identification procedure work. Financial institutions must manage these correctly since they cannot enable a user whose identity is not fully known to the Blockchain to complete a transaction. ("Hybrid Blockchain- The Best Of Both Worlds," n.d.)

Consortium Blockchain Systems

Several private blockchains from various organisations are combined to form a consortium blockchain. These blockchains act as a node on the chain and a stakeholder in the alliance, and it is only possible for them to quit or join the network with the consent of current stakeholders. ("Consortium Blockchain | Bybit Learn," n.d.)

The data can be viewed, exchanged, and distributed by organisations within the consortium, even though each entity runs its node or Blockchain. By doing this, solutions that cut across organisations and technologies can be created to enhance their current processes, accountability, and transparency, addressing individual blockchains' problems and difficulties. (“Consortium Blockchain | Bybit Learn,” n.d.). Although Hyperledger is technically one of these each companies’ individual system could be viewed as a permissionless system.

2.2.10 Basics of a functioning blockchain

This process will be explained in plain language, but it is worth noting that highly complex computation takes place to allow this process to work.

The first and most base process to occur in a blockchain is the establishment of block structure, usually decided by a smart contract or source code.

The block usually includes a timestamp, a digital signature, a last hash, and other crucial details that particular chain relies on. However, it should be emphasised that the block does not always contain the same information on every blockchain. The block is then sent around the entire network, and when the correct user uses his private key to match with the block, the transaction is completed.("How Does Blockchain Work? Everything You Need to Know [Updated]," n.d.)

After authorised nodes have verified the transaction, a block is added to the current transaction pool. Miners are nodes in public blockchain networks; they are frequently compensated for this labour using a procedure known as Proof of Work, or PoW and the newly emerging Proof of Stake, or PoS; typically, they receive payment in the form of the chain currency, Bitcoin in the case of the Bitcoin blockchain.(“What is blockchain and how does it work?,” n.d.)

Mining is a complex mathematical process. It starts by solving complex mathematical puzzles containing information from the block header to solve a hash. However, a miner must go through a process of trial and error to choose which string to use as a nonce before starting the procedure. Miners will concentrate on the nonce (a string number) associated with the hashed data of the preceding block when one is found. A hash must be less than or equal to the target hash to be successful. Furthermore, in return, the miner will receive payment for including the block in the Blockchain. This process will vary depending on the chain in question. (“Explained: What Is Hashing in Blockchain? | Bybit Learn,” n.d.)

A great demonstration of this is: https://andersbrownworth.com/blockchain/

3. Existing Crypto Application Case Study

Currently, there is a vast selection of running blockchain solutions, with some of the most important and explorable being Cryptocurrency solutions. This area of Blockchain is where the modern-day craze and population of Blockchain stem from.

3.1 Monero

Monero is a privacy-focused cryptocurrency that was launched in April 2014. (“About Monero | Monero - secure, private, untraceable,” n.d.) It was created as a fork of the cryptocurrency Bytecoin, with the aim of improving its privacy and security features. The development team

behind Monero introduced several unique features, including ring signatures and stealth addresses, to make it harder to trace transactions and identify users.

3.1.1 History

Monero has gained a reputation as one of the most private cryptocurrencies and has been adopted by several darknet marketplaces. Its market capitalization as of April 2022 was just under $3 billion. (“Monero price: XMR to USD chart | Ledger,” n.d.)

Since its launch, Monero has undergone several updates to improve its privacy and scalability. In 2016, the development team introduced the Ring Confidential Transactions (RingCT) protocol, which further improved the privacy of transactions by hiding the amount of Monero being transferred. In 2021, Monero implemented the Triptych protocol, which further improved its scalability by reducing the size of transaction data. (“Triptych Prioritization Meeting: 21 April 2021, 17:00 UTC · Issue #568 · monero-project/meta,” n.d.)Monero's privacy features have made it a popular choice for those who value anonymity and security, but its association with darknet markets has also made it a target for regulatory scrutiny.

3.1.2 Components

RandomX Proof of Work

Monero originally used a proof-of-work (PoW) consensus algorithm based on the CryptoNight hash function however it was later updated to use the RandomX algorithim. This algorithm is designed to be ASIC-resistant, meaning it is difficult to create specialised hardware to mine Monero more efficiently allowing for more equal mining between users. Part of what allows for this is that its creative approach uses virtual machines and memory control so excessive resources on a host machine don’t offer a miner an advantage. PoW also helps prevent Sybil attacks, which could otherwise allow an attacker to control the network by creating numerous fake identities. (“tevador/RandomX: Proof of work algorithm based on random code execution,” n.d.)

Transactions over Tor

Monero allows users to route their transactions over the Tor network, which helps to hide their IP addresses and location from prying eyes. This is done through an onion service, which encrypts the traffic and routes it through a series of nodes before it reaches its destination. By using Tor, users can also access Monero nodes and wallets without revealing their real IP addresses. Connecting via tor requires minimal configuration and can even be done via a GUI as seen on their website. (“Connecting your local wallet to your own daemon over Tor | Monero - secure, private, untraceable,” n.d.)

Stealth Addresses

Monero uses a technology called stealth addresses to enhance the privacy of its users. When a user creates a new address to receive funds, the system generates a one-time public key to create a unique stealth address. This means the sender can only see the stealth address, not the user's public address. As a result, it is much more difficult for an outsider to track a user's transaction history or link them to a specific address. In their whitepaper they refer to this as their unlinkable payment system. (“research-lab/whitepaper.pdf at master · monero-project/research-lab,” n.d.)

RingCT

Ring Confidential Transactions (RingCT) is a technology Monero uses to hide the transaction amount. Instead of revealing the exact amount being sent, RingCT allows a user to send a range of values that includes the actual amount. This makes it difficult for anyone to determine the exact value of a transaction, which enhances the privacy and fungibility of Monero. (Möser et al., 2018)

Dandelion++

Dandelion++ is a protocol that Monero uses to obfuscate the IP addresses of nodes on the network. This protocol works by routing a transaction through a randomised path of nodes before broadcasting it to the network. This makes it difficult for an attacker to trace the origin of a transaction or link it to a specific node. (Fanti et al., 2018)

3.2 Cardano

"Cardano" is a smart contract platform, and just like other blockchains, it can be used to send money from one place to another; however, its functionality may be expanded upon. Cardano employs design concepts to address problems with scalability, interoperability, and regulatory compliance in earlier coins. In addition, it is open source and more sustainable.

Cardano asserts that it solves issues with the cryptocurrency market, primarily that Ethereum is neither secure nor scalable and Bitcoin is too slow and rigid. Cardano was initially led by Charles Hoskinson, who co-founded Ethereum.

This means that Cardano is much more available at a lower price than other cryptocurrencies in the same category. However, doing so leads to less value due to its supply being over two times the supply of Monero.

Blockchains that use proof-of-stake instead of proof-of-work also consume far less energy. In February 2021, Hoskinson projected that the Cardano network utilised 6 GWh yearly or less than 0.01% of the 110.53 TWh that the University of Cambridge believed the Bitcoin network used. (“Cardano (blockchain platform) - Wikipedia,” n.d.)

3.2.1 History

Cardano employed the Ouroboros proof-of-stake system instead of the proof-of-work protocols used by Bitcoin and Ethereum (though the latter switched over in 2022). It is the first Blockchain to use this new kind of protocol. Ouroboros was co-developed by academics at the University of Edinburgh and engineers at IOHK in 2015. As a result, Bitcoin has a much faster emission rate with a block time of 120 seconds, compared to 2.5 minutes for Bitcoins and 1 minute for Monero. (“Cardano | Home,” n.d.)

3.2.2 Components

Proof of Work

Proof of Work (PoW) is a consensus algorithm used by blockchain networks, such as Bitcoin and the initial phase of Cardano, to validate transactions and create new blocks. The PoW algorithm involves solving complex mathematical problems that require a significant amount of computational power. The first node to solve the problem is rewarded with a block, which is

added to the blockchain. However, PoW algorithms are energy-intensive and slow, making them less scalable and environmentally friendly. Therefore, Cardano migrated to the Proof of Stake (PoS) consensus algorithm, which is more energy-efficient and scalable ("What Does Proof-of-Stake (PoS) Mean in Crypto?," n.d.). PoS allows nodes to validate transactions and create new blocks based on the amount of cryptocurrency they hold, rather than computational power. As of 2023 the sustainability of this consensus algorithm is even displayed on their website home page. ("Cardano | Home," n.d.)

The Settlement Layer

The Settlement Layer is the foundation of the Cardano network, where the multi-era ledger implementation and formal specifications are defined. It provides the core functionality for all other components of the Cardano network. The Settlement Layer is responsible for processing transactions and storing them in the blockchain. It also ensures the security and integrity of the blockchain. ("Cardano Components | Cardano Developer Portal," n.d.; "input-output-hk/cardano-ledger: The ledger implementation and specifications of the Cardano blockchain.," n.d.)

The Consensus Layer

The Consensus Layer is responsible for validating transactions and creating new blocks in the Cardano blockchain. It is based on the Ouroboros family of protocols, which provides a secure and efficient PoS consensus algorithm. The Ouroboros protocol is based on a verifiable random function (VRF), which allows nodes to generate unpredictable values for selecting the next block creator. The Hard-Fork Combinator is also implemented in the Consensus Layer, allowing seamless upgrades to the network without disrupting the Blockchain's integrity. (Kiayias et al., 2019)

The Networking Layer

The Networking Layer provides a peer-to-peer networking stack specifically designed for PoS systems. It includes a framework for writing typed protocols with support for pipelining, multiplexing, and various protections against adversarial peers. The Networking Layer ensures that nodes can communicate securely and efficiently, enabling the Cardano network to scale. The Cardano flavour of network implementation uses its own Ouroboros network API. ("input-output-hk/ouroboros-network: An implementation of the Ouroboros family of consensus algorithms, with its networking support," n.d.)

The Scripting Layer

The Scripting Layer, also known as Plutus ("Formal Specification of the Plutus Core Language," n.d.), is a Turing-complete scripting language embedded in the Cardano ledger. It provides a platform for developers to build and deploy smart contracts and dApps on the Cardano network. Plutus is based on a typed Lambda-Calculus, which acts as a low-level interpreted assembly code. The use of a typed language ensures that smart contracts are more secure and less prone to errors. ("input-output-hk/plutus: The Plutus language implementation and tools," n.d.)

Cardano Node

The Cardano Node is the software responsible for connecting to the Cardano network and maintaining a full copy of the Blockchain. It provides APIs for developers to interact with the Blockchain and deploy smart contracts. The Cardano Node is available as open-source software,

enabling developers to build on top of the Cardano network. (“input-output-hk/cardano-node: The core component that is used to participate in a Cardano decentralised blockchain.,” n.d.)

3.3 Hyperledger Fabric

HyperLedger Fabric is a project that aims to 'build better blockchains' by enabling high-performance and industry-specific applications. It was forked from the earlier blockchain fabric project, which was built as an open-sourced blockchain solution for enterprise IT.

It is an open-source blockchain framework with many components that make it easy for organisations to build, test, and deploy their Blockchain or cryptocurrency on top of a robust system. The project was announced in July 2015 at the first Hyperledger meeting in San Francisco as a collaboration between American Express, IBM, and Digital Asset; however, the history of this project spans back to 2005, when domain ledgers were first discussed.

Its objective is to provide a foundation for blockchain-based distributed ledgers that can support multiple applications with varying requirements. Therefore, it can be considered one of the best frameworks for enterprise companies to use when building custom or heterogeneous private or permissioned blockchains, which fall into the category of solutions that do not use a public blockchain infrastructure like Ethereum's public blockchain platform.

3.3.1 History

The History of Hyperledger is complicated and less clear cut than other chains. There is no definitive history, even though most sources agree that this project was announced as early as July 2015 at the second HyperLedger meeting in Berlin, Germany.

Hyperledger Fabric is a blockchain framework implementation and one of the Hyperledger projects hosted by The Linux Foundation. It was forked from the earlier blockchain fabric project, which was built as an open-sourced blockchain solution for enterprise IT.

The fabric project became one of the Hyperledger projects in early 2016 after it became apparent that developing a single cross-industry open standard would be more valuable than building only a single commercial solution.

Today, HyperLedger Fabric can be considered a general-purpose platform designed to support multiple industries with diverse needs and requirements. In 2015 the Hyperledger project was announced as a collaboration between IBM, Digital Asset, and American Express. In 2017 the project grew immensely with seven new member organisations.

3.3.2 Components

Like regular blockchain networks, Hyperledger Fabric consists of numerous parts and components, allowing it to function as it does. Although several parts of its architecture are nearly identical to regular blockchain structures, some of these components have subcategories or extensively modified functionality which will be further explored in this section.

Cryptogen

Cryptogen is a tool Fabric uses to generate the necessary key signing materials to maintain Fabric security. It is offered to set up a network in advance for testing. In regular usage, a production network would not use it. (“cryptogen — hyperledger-fabricdocs main documentation,” n.d.)

The cryptographic components of the Test Network are created using a configuration file. Then Docker compose files can be used to start the consortium network with them. (“Two Ways to Generate Crypto Materials in Hyperledger Fabric: Cryptogen and CA Server | by KC Tam | Medium,” n.d.)

Peers

Peers can be organised according to the requirements of the ledger and contracts being managed and are connected via channels. It is a node that houses the Hyperledger Fabric System's ledger and chain code.("Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation," n.d.)

Hosting a ledger instance by a peer without hosting any chaincodes that access that ledger is possible. Each peer can host multiple installed ledgers and chaincodes. Since the chaincode does not need to be loaded on every peer, in this situation, other peers communicate using the Gossip protocol to update the ledger with the most recent state. Numerous organisations cannot control a peer at the same time. (“Role of Peers in Hyperledger Fabric Blockchain | by Chetan Gadgilwar | Medium,” n.d.) These peers are reasonably similar to peers in a standard blockchain like Bitcoin.

State Database

LevelDB and CouchDB are the peer-state databases that Fabric officially supports. The peer process and its default key-value state database, LevelDB, are integrated within the peer node. According to Fabric's official documentation, LevelDB is the default state database. (“Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation,” n.d.)

Google created the key/value store called LevelDB. It can handle a string key to string values ordered mapping. A log-structured merge tree is LevelDB's primary storage architecture (LSM). Instead of modest random writes, it is optimised for massive sequential writes. (“LevelDB - Database of Databases,” n.d.)

When the chaincode data is built in JSON format, CouchDB is another optional state database that can enable extensive data query functionality. One of the so-called NoSQL solutions is CouchDB. CouchDB is a document-oriented database, and fields are kept as key-value maps within each document. Simple key/value pairs, lists, and maps are all fields. (“What is CouchDB and Why Should I Care?,” n.d.)

An object's current state values are all contained in a set of data called world state that is persistently stored by a chaincode. Key-value pairs are used to arrange all world state data. Chain codes can use put, get, and delete operations to communicate with world states. In a state, the database kept the most recent values for each key in chaincodes.("Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation," n.d.)

Chaincode

One of the essential components is chaincode, which comes in two varieties: a general chaincode provided by a developer or user and a system chaincode hosted by the Fabric framework to facilitate communication. (“Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation,” n.d.)

Chaincode is quite like smart contracts in other blockchains, with the main difference being that it is packaged for enterprise use.

A chaincode's creation state is limited to that chaincode alone and cannot be directly accessed by another chaincode. However, a chaincode inside the same network may invoke another chaincode to access its state if it has the necessary authorisation. (“Chaincode Tutorials — hyperledger-fabricdocs master documentation,” n.d.)

HyperLedger Fabric also offers Chaincode in two views, one for developers and one for operators. The main difference is that the operator chaincode view is more aimed towards currently running HyperLedger setups that need upgrades or maintenance. This view is not used to develop a solution from scratch. (“Chaincode Tutorials — hyperledger-fabricdocs master documentation,” n.d.)

Channels

To conduct private and confidential transactions, Hyperledger Fabric channels serve as a private "subnet" of communication between two or more network users. Members (organisations), the shared ledger, the chaincode application, and the ordering node all contribute to the definition of a channel. (“Channels — hyperledger-fabricdocs main documentation,” n.d.)

Each party must be authenticated and authorised to transact on a certain channel, where every transaction on the network is conducted. A membership services provider (MSP), which authenticates each peer to its channel peers and services, assigns each peer who joins a channel its identity. (“Channels — hyperledger-fabricdocs main documentation,” n.d.)

Even though a single anchor peer may be a member of several channels and maintain multiple ledgers, no ledger data may cross channels. The identity membership service, the gossip data distribution protocol, and the configuration chaincode describe and implement this separation of ledgers by channel. (“Channels — hyperledger-fabricdocs main documentation,” n.d.)

Channels run on the gossip protocol. Which peer interacts with the ordering service on behalf of each member on a channel is decided by the election of a leading peer for each member. A leader can be found using an algorithm if no leader is found. The consensus service organises transactions and delivers them in blocks to each leading peer. The leading peer then uses the gossip protocol to distribute the block to its member peers over the channel.

MSPs (Membership Service Providers)

Any component in Fabric has an identity, which determines the component's precise rights and role. The MSP is the organisation that controls the identities of all players in this blockchain network and establishes the guidelines, rights, and roles for various components in the Fabric network.

Public Key Infrastructure (PKI) and Certificate Authority (CA) are used to implement participant IDs . All cryptographic activities, including certificate issuance and validation, are abstracted by MSPs.

MSPs, as a concept, are a tool that developers can use to define the identities they need to interact with organisations, peers, ordering services, users, and applications. ("Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation," n.d.)

Gossip protocol

Peers use gossip to broadcast ledger and channel data efficiently. Each peer on a channel continuously receives up-to-date and consistent ledger data from various peers due to gossip messaging's continuous nature. In addition, each gossiped message is signed, allowing Byzantine participants to quickly identify those who are sending fraudulent messages and preventing the spread of the message(s) to undesirable targets. ("Gossip data dissemination protocol — hyperledger-fabricdocs main documentation," n.d.)

Gossip-based broadcasting works by peers receiving messages from other peers on the channel and forwarding these messages to a configurable constant-sized number of randomly chosen peers on the channel. In addition to using the pull mechanism, peers can send messages without waiting for them to be delivered. This cycle keeps repeating, maintaining the accuracy and synchronisation of channel membership, ledger, and state information. The channel's leader peer gathers information from the ordering service to spread rumours about new blocks to peers within its own company. ("Gossip data dissemination protocol — hyperledger-fabricdocs main documentation," n.d.)

The Hashgraph, developed by Leemon Baird in 2016, is an intriguing illustration of a distributed system that uses a gossip protocol. It is a distributed ledger technology that uses the aBFT consensus mechanism or asynchronous Byzantine Fault Tolerance. A hash graph network's nodes acquire and summarise data regarding transactions and other events, then disseminate this information to randomly selected neighbour nodes. As a result, the Hashgraph network creates a tree of events where all information is captured rather than a chain of blocks (no data is ever discarded). ("Gossip Protocol | Binance Academy," n.d.)

Fabric Gateway

A Fabric Gateway service, first made available to Hyperledger Fabric v2.4 peers, offers a streamlined, basic API for submitting transactions to a Fabric network. To facilitate easier application development and transaction submission, requirements formerly imposed on the client SDKs, such as obtaining transaction endorsements from peers of various organisations, are transferred to the Fabric Gateway service running within a peer in version 2.4. ("Fabric Gateway — hyperledger-fabricdocs main documentation," n.d.)

The currently supported Go versions and the current long-term support (LTS) releases of Node and Java are one of the main improvement goals of each minor release of the Fabric Gateway client API. The Gateway peer needs a minimum version of Hyperledger Fabric to function properly. ("fabric-gateway | Go, Node and Java client API for Hyperledger Fabric v2.4+," n.d.)

3.4 Ethereum

Ethereum is a decentralized, open-source blockchain platform that enables the development of smart contracts and decentralized applications (dApps). It was created in 2015 by Vitalik Buterin and has since become one of the most popular blockchain platforms in the world.

3.4.1 History

Ethereum was conceived by Vitalik Buterin in 2013 as a blockchain platform that would allow for the creation of smart contracts and decentralized applications. Buterin was inspired by Bitcoin's limited scripting language and saw the potential for a more flexible platform that could support a wider range of applications.

In 2014, Buterin launched a crowdfunding campaign to fund the development of Ethereum. The campaign was a huge success, raising over $18 million in Bitcoin, making it one of the largest crowdfunding campaigns in history. The Ethereum blockchain went live in July 2015, and since then, it has grown to become the second-largest cryptocurrency by market capitalization, after Bitcoin. (“Blockchain: A Very Short History Of Ethereum Everyone Should Read | Bernard Marr,” n.d.)

3.4.2 Components

Ethereum Virtual Machine (EVM)

The Ethereum Virtual Machine (EVM) is a software environment that runs on the Ethereum network and executes smart contracts. It is a critical component of the Ethereum ecosystem, as it allows developers to create decentralized applications (dApps) and execute smart contracts in a secure and efficient manner. The EVM is designed to be a Turing-complete environment, which means it can run any program that can be expressed in code. This provides developers with the flexibility to create complex applications and execute arbitrary computations on the network.(“Ethereum Virtual Machine (EVM) | ethereum.org,” 2023)

Ether (ETH)

Ether (ETH) is the native cryptocurrency of the Ethereum network. It is used to pay for transaction fees and to incentivize network participants to validate transactions and maintain the blockchain. Ether is also used as a medium of exchange and a store of value, similar to other cryptocurrencies like Bitcoin. It is a critical component of the Ethereum ecosystem, as it ensures that network participants have a financial incentive to act in the best interests of the network. (“What is ether (ETH)? | ethereum.org,” n.d.)

Solidity

Solidity is the primary programming language used to write smart contracts on the Ethereum network. It is a high-level language that is designed to be easy to read and write for developers, even those who are not familiar with blockchain development. Solidity is a statically typed language, which means that the data types of variables must be declared at compile time. It also supports inheritance, libraries, and complex user-defined types, making it a flexible and powerful language for smart contract development. (“Solidity — Solidity 0.8.19 documentation,” n.d.)

Decentralized Applications (dApps)

Decentralized applications, or dApps, are applications that run on the Ethereum network and are built using smart contracts. dApps are designed to be decentralized, meaning they are not controlled by any single entity and are transparent and secure. They are built to be open-source, meaning anyone can access the code and contribute to the development of the application. dApps are also designed to be resistant to censorship and tampering, making them ideal for applications that require high levels of security and trust. ("Decentralized applications (dapps) | ethereum.org," n.d.)

3.5 Polygon

Polygon is a layer 2 scaling solution for Ethereum blockchain, which aims to improve the scalability and interoperability of the Ethereum network. Polygon, previously known as Matic Network, ("maticnetwork/whitepaper: Matic whitepaper," n.d.) was rebranded in February 2021 to reflect its broader vision of providing a multi-chain ecosystem.

3.5.1 History

Polygon was founded by Jaynti Kanani, Sandeep Nailwal, and Anurag Arjun in 2017 as Matic Network, with the aim of improving the scalability and usability of Ethereum. In 2020, Matic Network rebranded as Polygon and shifted its focus to building a multi-chain ecosystem.

Polygon has quickly gained popularity within the blockchain community, as its solutions have helped to overcome some of the scalability issues faced by Ethereum. Polygon has partnered with various blockchain projects, including Aave, Curve, and Decentraland, to improve the speed and cost of transactions on their platforms. ("About Us | Polygon," n.d.)

3.5.2 Components

Polygon SDK

Polygon SDK is a modular framework that allows developers to build their own customized blockchain networks. The SDK offers a set of tools and resources to developers for creating a blockchain network that is optimized for specific use cases, such as gaming or finance. With Polygon SDK, developers can build a blockchain network that is tailored to their requirements and can be easily integrated with other networks. The SDK is open-source and offers a high level of flexibility and scalability. Developers can use the SDK to build decentralized applications (dApps) on the Polygon network. ("Get started | Polygon Wiki," n.d.) ("What is Matic Network? (MATIC) Blockchain Scaling Platform," n.d.)

Polygon PoS Chain

Polygon PoS Chain is a proof-of-stake (PoS) blockchain that offers fast and low-cost transactions. The PoS chain is a layer 2 solution that is secured by the Ethereum mainnet, providing a high level of security. With Polygon PoS Chain, developers can build dApps that offer fast and efficient transactions. The PoS chain offers a high level of scalability and can handle a large number of transactions per second. ("Polygon PoS | The most efficient blockchain protocol," n.d.)

Polygon Bridge

Polygon Bridge is a bridge technology that can potentially connect different blockchain networks, enabling seamless interoperability between them. The bridge enables the transfer of assets between different blockchain networks, with its current focus on Ethereum and Polygon and in turn all other Ethereum based currencies. With Polygon Bridge, users can transfer tokens from one network to another, without having to go through a centralized exchange. The bridge offers fast and secure transfers, with minimal fees. ("Ethereum↔Polygon Bridge | Polygon Wiki," n.d.)

Polygon APIs

Polygon APIs are a set of tools that developers can use to interact with the Polygon network. The APIs include tools for accessing and analyzing blockchain data, as well as for building dApps on the Polygon network. With Polygon APIs, developers can easily build and deploy dApps on the Polygon network, without having to worry about the underlying blockchain infrastructure. The APIs are easy to use and offer a high level of flexibility and scalability. ("Polygon-SDK : Polygon Support," n.d.)

4. Existing Applications Case Study

There are many benefits to online bidding, but there are also drawbacks. Opportunistic behaviour on the part of the transaction partner can hurt buyers. Companies who wish to sell their products online will soon need to decide whether online auctioning platforms are a viable alternative to the standard web store. The fact that the participants in Internet auctions do not know one another is a fundamental issue. Therefore, trading between two people via online auctions demands high confidence. ("Analysis of the online auction platform eBay. Advantages and disadvantages - GRIN," n.d.) A Hyperledger-based system can remove this need for confidence; however, it is worth reflecting on some existing platforms to understand better how to achieve this.

4.1 eBay

eBay is a popular online marketplace recognised for its consumer-to-consumer transactions and auctions. It is also well-liked by internet retailers who use it as a sales channel. There are numerous nations where eBay is accessible. However, you can enter the zip code to look for things in your neighbourhood. ("What is eBay? Dow Does eBay Work?," n.d.) As an alternative, you can look for goods sold domestically or abroad. Anyone can create a free eBay account. Both buying and selling things are options. There are costs involved with having a seller account, such as listing fees. The expenses vary depending on the amount you sell your goods for and how long you offer them.

First, let us establish what an eBay shop is. A brand-named business page on eBay is what an eBay store is. Retailers can display all listings, regardless of format, in one location by setting up an eBay shop. Additionally, you can customise your store to represent your business better and get a special URL. ("How To Set Up An eBay Shop: Best Practices, Fees & Seller Tools.," n.d.) This is the equivalent of an organisation in HyperLedger in many respects.

The seller sets a beginning bid price in an auction, and customers place bids on the item. The seller must complete the deal even if there is just one offer. A reserve is a minimum asking price the seller must meet to sell an item at auction. It is concealed from the customer. Over time, reserve pricing has become less effective. Without running the danger of starting an auction too low, a reserve enables the seller to obtain their intended price. However, bidders may be discouraged by reserve pricing. ("Understanding eBay Buying Options," n.d.)

These systems are all core components of how eBay laid out its bidding and auctioning system, which has been copied repeatedly. However, a problem in recent years is eBay's sniping tools which, in plain terms, mean automatic bidding software.

The last few seconds of the auction are when snipers, and automated tools, place bids on things. Their goal is to bid automatically and covertly without the seller knowing the buyer's interest. Early bids stimulate other bids, increase the item's search results, and increase interest. One can use a free sniping service like Gixen while remaining undetected to win low-priced auctions. ("Understanding eBay Buying Options," n.d.)

4.2 SETAM

The Ukrainian government conducts state-owned property auctions using an electronic trading system called SETAM ("CETAM - setam.net.ua," n.d.). Any Ukrainian citizen is welcome to participate and place a bid.

However, many people needed more confidence in the internet system's complete security or transparency. Therefore, the system was transferred to an Exonum blockchain to increase public confidence and guarantee that the auctions were fair. ("Rebuilding citizen trust in government e-auctions," n.d.)

Exonum is an open-source, enterprise-grade blockchain framework that aids in the safe implementation of blockchain initiatives by corporations and governments, much like HyperLedger fabric. ("Exonum | Knowledge for policy," n.d.)

In this new method, third parties may monitor an auction's progress and might be contacted by users who have suspected fraud investigating. For SETAM, the East Europe Foundation worked as a neutral third-party auditor.

Following the addition of Blockchain to SETAM, the total number of auction participants and new partners and the average price of each lot increased. In addition, since there are no longer any intermediaries in the system, it is now more efficient and less likely to be corrupted.

Four thousand blockchain-based auctions with $24.5 million in sales were successfully held in the first five months following the launch of Exonum. ("Rebuilding citizen trust in government e-auctions," n.d.)

Although SETAM as a platform is accessible a language barrier exists for me to be able to fully grasp it and although Exonum is an open source platform, the SETAM specific version isn't as it's Ukrainian government property.

4.3 Walmart Food Tracing System

Walmart believed the decentralised food supply network and blockchain technologies might work well together. So the business developed a food traceability system based on Hyperledger Fabric to verify this theory. IBM, Walmart's technology partner, tested the system with two proof-of-concept projects. The goals of the two projects were to track pork sold in Walmart's China stores and mangos sold in its US stores, respectively. ("Walmart Case Study – Hyperledger Foundation," n.d.)

The food tracking system for the two products created on the Hyperledger Fabric blockchain functioned. It made it possible to upload authenticity certifications for pork in China to the Blockchain, increasing trust in a system where it was previously a significant problem—in addition, the time required to determine the origin of mangoes in the US decreased from 7 days to 2.2 seconds.

Walmart has been tinkering with it for a year and feels ready for wider implementation. The ultimate goal is to ensure that the food sold in the stores is safe to eat and, if there is a problem, to simplify auditing the supply chain. Using the Hyperledger Fabric-based system, Walmart can track the provenance of over twenty-five items from five vendors. ("Supply Chain Transparency," n.d.)

Like SETAM, sadly Walmart has not publicly released much information about this system for obvious security reasons, so I could not find any code, screenshots or even a logo for their application.

4.4 J.D Open Platform

The largest retailer in China, JD.com, is introducing a new blockchain technology platform to assist enterprise clients in creating, hosting, and utilising their blockchain apps for safer, more transparent, and more practical operations management.

The platform, known as the JD Blockchain Open Platform, allows users to easily develop and modify smart contracts on both public and private enterprise clouds. It is based on several underlying technologies. The use of technology can assist businesses in improving operational processes such as authenticity certification, property evaluation, transaction settlements, digital copyrights, tracking and tracing of donated goods and donations to charities, and productivity. ("JD Launches Blockchain Open Platform - JD Corporate," n.d.)

The new platform is the most recent development in JD's Retail as a Service (RaaS) strategy, which sees the e-commerce behemoth make its innovative infrastructure and technology available to other businesses and sectors. ("JD.com launches new blockchain platform to help enterprise customers | Supply Chain Magazine," n.d.)

A BaaS platform called JD Blockchain Open Platform has an application store that offers several blockchain bottom layers, internal and externally produced tools, and applications (ISV). Users can securely and easily customise their applications to meet their business needs thanks to JD's strict quality control procedures applied to the app store's products. In addition, the company will

continuously improve its application offerings by adding additional ISV and building a strong ISV community on the platform.

China Pacific Insurance Company (CPIC) is the first JD partner to use the JD Blockchain Open Platform. CPIC used the platform to roll out a traceable system for e-invoices, or "fapiao," the legal receipts necessary in China for business. By giving each document a distinct blockchain ID and expediting the accounting process, the solution improves the security control of e-invoices. ("JD.com Launches Blockchain Open Platform," n.d.)

4.5 OpenIDL Insurance Software

Insurance firms are expected to submit considerable regulatory data regularly that must be shared securely with regulators and be subject to many different compliance requirements. OpenIDL (open Insurance Data Link), intended to streamline insurance regulatory reporting, was created by the American Association of Insurance Services, a nonprofit insurance advising organisation. OpenIDL can assist in simplifying regulatory and compliance requirements while enhancing efficiency and accuracy for both insurers and state insurance administrations because it is built on IBM Blockchain and powered by Hyperledger Fabric. ("Five Hyperledger Blockchain Projects Now in Production – Hyperledger Foundation," n.d.)

The OpenIDL Blockchain and distributed ledger technology were deployed for a COVID-19 regulatory data request, and it immediately produced previously unreachable insights while upholding data privacy and security. OpenIDL, initially created by AAIS specifically for our Members, is currently being improved upon by the Linux Foundation as an open-source ecosystem for the whole insurance sector. ("openIDL: an open blockchain network for insurers," n.d.)

OpenIDL is an expensive software often paid for by firms for access, so access to this for my research was only possible with an insurance firm license. However, this proves Hyperledger's effectiveness in keeping a blockchain system permissioned.

5. Relevant Technologies Overview

In order to complete this project, several software libraries and tools will be used to complete it. Things like Node.js and Docker will be used to configure and host test aspects of Docker and things like curl to pull the official Hyperledger Fabric modules down to my local machine. However, this section will also explore other technologies I could use depending on the solution I make the decision to use for my project.

5.1 Node JS

Built on the JavaScript Engine in Google Chrome, Node.js is a server-side platform (V8 Engine). Ryan Dahl created Node.js in 2009; version 19 is the most recent release. An open-source, cross-platform runtime environment called Node.js is used to create networking and server-side applications. Applications for Node.js can be created in JavaScript and run on Linux, OS X, and Microsoft Windows using the Node.js runtime. Additionally, Node.js offers a comprehensive library of different JavaScript modules, streamlining the creation of web applications utilising Node.js. ("Node.js - Introduction," n.d.)

An application written in Node.js operates in a single process rather than starting a new thread for each request. Blocking behaviour in Node.js libraries is the exception rather than the rule because libraries are typically created using non-blocking paradigms, and Node.js comes with a set of asynchronous I/O primitives that prevent blocking in JavaScript applications.

As Node.js conducts an I/O action, such as reading from the network, accessing a database, or using the filesystem, it will continue the operations when the response arrives rather than halting the thread and wasting CPU time waiting. (“Introduction to Node.js,” n.d.)

Because of this, Node.js can manage hundreds of connections at once on a single server without adding the hassle of controlling thread concurrency, which is a large potential source of errors.

Asynchronous refers to the ability for events to take place apart from the primary program flow. Every program on today's consumer computers runs for a set amount before stopping to allow another program to carry on with its execution. It is impossible to discern this thing's cycle because it moves so quickly. Our perception that our computers run multiple programs simultaneously is deceptive (except on multiprocessor machines). Internally, programs use interrupts, a signal sent to the CPU to get it to pay attention.("JavaScript Asynchronous Programming and Callbacks," n.d.) This is a vital component of the node and its functionality.

JavaScript would prove a huge advantage for this project as it’s well documented, widely used, regularly updated and has wide support. One substantial disadvantage it has in regards to other languages however, is its reliance on NPM. As so few packages and functionality are included in JavaScript by default (due to it being a scripting language), I would have to install node packages to expand the languages functionality. This is a bit annoying as my supervisor has said I can’t use many packages as this can be marked down but JavaScript has become such a core component of nearly all modern day software from web, desktop, android and iOS to backend web functionality, I feel that using NPM extensively is unavoidable regardless of the blockchain system used.

5.2 Docker

An open platform for creating, distributing, and running programs is Docker. You may divide your apps from your infrastructure with the help of Docker, allowing for rapid software delivery. You can manage your infrastructure using Docker the same way you manage your applications. You can shorten the time between writing and executing code in production by utilising Docker's methodology for shipping, testing, and deploying code quickly.("Docker overview | Docker Documentation," n.d.)

Fundamentally, Docker containers separate an application's functionality into a few parts that may be independently deployed, tested, or scaled as necessary. Consider an application's Docker-containerised database as an Example. A structure like this allows you to scale or maintain the database separately from other application modules or components without affecting the workloads of other crucial systems. (“Introduction To Docker: A Beginner’s Guide – BMC Software | Blogs,” n.d.)

Live instances of images are referred to as containers on which a program or its independent modules are executed. An image is a class in the object-oriented programming paradigm, and the container is an instance of that class. As a result, you may use a single image to create several containers, increasing operational efficiency. (“Introduction To Docker: A Beginner’s Guide – BMC Software | Blogs,” n.d.)

A named Docker image distribution and storage system is called a Docker registry. Multiple variations of the same image may exist and be distinguished using tags. A Docker registry is divided into Docker repositories, each containing every iteration of an individual image. Users of Docker can submit new images to the registry and pull existing ones locally through the registry (given adequate access permissions when applicable). (“Docker Registries - Aqua,” n.d.)

The Docker engine communicates by default with Docker Hub, a public registry instance. The open-source Docker registry and distribution and a version maintained by a business called Docker Trusted Registry can be run on-premises. Online, there are other public registries.

The client-server platform Docker Engine creates and executes containers utilising Docker's components and services. Either Docker Engine consists of the Docker daemon, a REST API, and the CLI that communicates with the Docker daemon through the API, or Docker Inc., which provides different editions of containerisation technology based on Docker Engine, is meant when the term "Docker" is used. (“What is Docker Engine? - Definition from WhatIs.com,” n.d.)

The use of Docker across an application architecture has several advantages. First, by dockerising their programs into single or several modules, Docker enables development teams to save time, effort, and money.

A build cycle can prevent the ongoing challenge of having numerous versions of dependencies that may result in issues in production by making the first effort to produce an image specifically for an application. You can separately test each containerised application (or its component) using Docker without affecting other application components. Eliminating tightly connected dependencies and providing higher fault tolerance makes it possible to create a secure framework.

Docker ensures consistency in libraries and packages at every stage of the development process to reduce friction between teams. In addition, with an already-tested container, bugs are eliminated from the build process, allowing for an efficient migration to production. (“Introduction To Docker: A Beginner’s Guide – BMC Software | Blogs,” n.d.)

Sadly, Docker is the only service listed for virtual server hosting, however it is recommended to move docker onto a platform like digital ocean if I use Hyperledger. Docker is probably one of the Hyperledger related technologies I have the least experience with. Other than spinning some basic instances up on my home NAS I have no idea how to develop for it.

5.3 Python

At the National Research Institute for Mathematics and Computer Science in the Netherlands, Guido van Rossum created Python in the late 1980s and early 1990s.

ABC, Modula-3, C, C++, Algol-68, Smalltalk, the Unix shell, and other scripting languages are just a few that Python is derived. Copyright applies to Python. Python source code is now accessible under the GNU General Public License, just like Perl (GPL). ("Python - Overview," n.d.)

Although a core development team at the institution is now responsible for maintaining Python, Guido van Rossum continues to play a crucial role in guiding its development. First, you must compile a program you have written in C or C++. Compiling entails converting your human-readable code into Machine Code, which computers understand. Machine code is the lowest level of instructions that the CPU may directly execute. Your code creates an executable file after a successful compilation. The activities in your code are run step-by-step when this file is executed. ("What is the Python programming language?," n.d.)

Although compilation is a step, Python is primarily an interpreted language rather than a compiled one. Python code, written in.py files, is first translated into bytecode and then saved in .pyc or .pyo formats.

Python source code was converted to bytecode rather than machine code, like C++. An interpreter can carry out a low-level set of instructions known as bytecode. The Python interpreter is often installed on computers in /usr/local/bin/python3.8, and this is typically only on Linux, with Windows allowing for easy installation via the Microsoft store. Instructions written in bytecode are carried out on a virtual machine rather than a computer's central processing unit. ("How does Python work?. A simple explanation of how Python code… | by Dhruvil Karani | Towards Data Science," n.d.)

The fact that interpreted languages are platform-independent is a well-liked benefit. For example, python bytecode can be run on any platform as long as the Virtual Machine and the bytecode are the same versions which would greatly assist in making my application cross-platform.

Python is a language I have minimal experience with but it is a language I find extremely interesting and I would like to use it if possible.

5.4 Java

The object-oriented, class-based, general-purpose programming language Java was created to have fewer implementation requirements. It is an application development platform for computers. Java is consequently quick, safe, and trustworthy. It is commonly used to create Java applications for smartphones, game consoles, laptops, data centres, and other devices. ("What is Java? Definition, Meaning & Features of Java Platforms," n.d.)

OAK was the early name of the Java programming language. It was initially designed to handle portable electronics and set-top boxes. A colossal failure, Oak. Sun changed the name to "Java" in 1995 and made changes to the language to capitalise on the expanding www (World Wide Web) development market. Sun Microsystems was later acquired by Oracle Corporation in 2009, and as a result, three important Sun software assets—Java, MySQL, and Solaris—became part of Oracle's portfolio. ("What is Java? Definition, Meaning & Features of Java Platforms," n.d.)

The Java program was the first to use a Java Virtual Machine to integrate the two techniques (JVM). The Java Virtual Machine is the name given to the Java code compiler. To create bytecode, any Java file must first be compiled. Therefore, only the JVM supports running Java bytecode. The JVM subsequently translates the bytecode to be executed on the underlying hardware platform. Because of this, the JVM will interpret the program for Windows if it is executing on a Windows machine.

On the other hand, the JVM will interpret it for Linux if it operates on an open-source platform like Linux. ("What is Java? - Enterprise Java Beginner's Guide - AWS," n.d.) Like Python, this would be a huge advantage for the project as it ensures that cross-compatibility would be much easier to achieve.

Java is one of the only technologies relevant to this project that I have experience with from my degree course so if the documentation for it is good, I intend to use that.

5.5 Golang

The complexity of the codebases at Google led to the creation of the Go programming language. It was created by the trio of Robert Griesemer, Rob Pike, and Ken Thompson, whom all despise C++. Go was initially made public in 2009 and became open source in 2012 with the release of its first version, 1.0.

Due to its simplicity, readability, efficiency, and concurrent nature, Go quickly gained popularity and became the preferred language among many developers. It can perform numerous tasks simultaneously if they are concurrent. Go is used for cloud-based programming, game development, data science, and server-side (backend) programming. Making command-line tools is another common use for it. ("What is Go? Golang Programming Language Meaning Explained," n.d.)

Go works using "goroutines," lightweight processes that enable additional efficiency. Go also makes use of a number of packages to manage dependencies effectively. Google, Cloudflare, Dropbox, MongoDB, Netflix, SoundCloud, Twitch, and Uber are a few companies that use Go. ("What is the Go Programming Language?," n.d.)

Until taking on this project I didn't even know what Golang was and have no experience with it. Go is compulsory for the chaincode I need to write.

5.6 Rust

Rust is a systems programming language first introduced in 2010 by Mozilla Research. It has gained a lot of popularity in recent years due to its unique combination of safety, speed, and concurrency. Rust was designed to be a safe and reliable alternative to C and C++, focusing on preventing common programming errors such as null pointer dereferences, buffer overflows, and data races. Rust's safety features are enforced by its ownership and borrowing system, which allows for efficient memory management without needing a garbage collector.

In addition to its safety features, Rust is also known for its performance. Rust code can be optimised to run at near-native speeds, making it a popular choice for applications that require high performance, such as game engines and network servers. Rust's concurrency model is also

worth mentioning. It provides low-level control over threads and synchronisation primitives, making it easy to write highly concurrent code without sacrificing safety. (“Rust Programming Language,” n.d.)

Rust's popularity has proliferated recently, with many companies and organisations adopting it for their projects. Some notable examples include Dropbox, Microsoft, and Amazon. Rust is also used extensively in the blockchain and cryptocurrency space, with projects such as Parity and Polkadot (“paritytech/polkadot: Polkadot Node Implementation,” n.d.) written nearly entirely in Rust.

The performance friendly, memory safe structure makes this language perfect for use in a blockchain with Exonums SDK having support for Rust. Sadly Rust is complex and takes time to become proficient in, time I do not have with this project.

5.7 C++

C++ is a high-level, general-purpose programming language that was first developed in the 1980s by Bjarne Stroustrup as an extension of the C programming language. It is widely used for developing complex software applications, such as operating systems, database systems, video games, and scientific simulations. C++ offers a powerful combination of high-level abstractions and low-level control, making it suitable for a wide range of programming tasks. (“1.3 - Brief History of C++ - Object Oriented Programming with C++, Second Edition [Book],” n.d.)

One of the key features of C++ is its support for object-oriented programming (OOP), which allows developers to organise code into reusable modules called classes. C++ also offers support for generic programming, which allows developers to write code that works with different data types without having to write separate functions for each type. Other features of C++ include support for templates, which enable developers to write generic algorithms and data structures, and operator overloading, which allows operators such as + and - to be redefined for custom types. (“About : Standard C++,” n.d.)

C++ has a large and active community of developers and users, with many resources available for learning and using the language. For example, the C++ Standard Library provides a rich set of functions and data structures that can be used for everyday programming tasks, and there are also many third-party libraries available for specialised domains such as graphics, networking, and scientific computing.

C++ is also a language I had exposure to throughout my course work however I do not feel I’m nearly proficient enough in it to build a blockchain or use a C++ blockchain (“bitcoin/src at master · bitcoin/bitcoin,” n.d.)SDK. Also, in trying to tackle the P2P functionality in the project I would need to be able to handle socket programming which I’ve never used or encountered before.

6. Research Findings

Having researched the project specification thoroughly, its technologies and previous implementations, I feel Hyperledger Fabric is a broad and exciting technology to use in order to

build an auctioning system. Furthermore, the case study of Ukraine's currently implemented auctioning system is fascinating.

6.1 Previous implementations

A lot can be analysed and learned from the previous or other implementations of auction systems. For example, one of the main things that irritate people is using automatic bidding systems like Gixen, but this is extremely hard to counteract, especially when the base packages and libraries used are open sources.

However, I feel HyperLedger has the advantage of being significantly better in structure regarding policing auctions and users. The proposed way that HyperLedger and its official docs give to add organisations and users is much better than the system eBay uses.

eBay has become a slightly chaotic e-commerce situation as anyone can access the platform, its bids, and make shops or users. When blockchains took off initially, many GPU sales frauds took place on eBay, leading to surges in claims made by PayPal. By using Hyperledger to make and handle bids, it is much easier to revert or refund bids.

The Ukrainian implementation is also an interesting case study of the general public's lack of trust in blockchain-based systems. Although this technology had moved far from where it was when Bitcoin came into fruition, It still needs to be simplified to understand for most day-to-day users, and this can be a significant turnoff from using it.

6.2 Other Blockchains

Hyperledger is based on Ethereum and shares several of the same components and structures; however, it is its system and implementation. This is very important to note as it means HyperLedger shares several of the pros and Ethereum's cons.

The biggest pro to the Ethereum-based HyperLedger is its low wait times. One of the things that Bitcoin was criticised for was the fact that it wrote to the Blockchain in seven-second intervals when it was first started.

Another huge advantage it has over other blockchains is its permissioned nature. This makes it significantly easier to keep control of the Blockchain and the transactions that go on within it; however, this poses a disadvantage, often argued by critics that it is not truly traceable as it is not public.

It is also worth pointing out that this Blockchain, compared to others, is at a severe disadvantage as it currently does not have many use cases. Although many hypotheticals surround its use in the modern business world, few of these have come to fruition. It’s also somewhat limited as it orientates itself towards business applications as oppose a broad use case scenario.

For my application of this Blockchain, the biggest disadvantage I face compared to other blockchains is the need for existing examples of user-written code that I can learn from, with the official documents only taking me so far. However, over three years since its last official release, the official examples repository features some good examples to help me. I fear these will not be

sufficient to achieve the quality I know I can potentially achieve. Having researched other frameworks, their documentation, use cases, and example code are more extensive and clearer.

6.3 Technical Research

An important aspect of my research was trying to get a development environment working. I used the technologies listed in this document under the relevant technologies section. Getting a basic test or development environment running to observe HyperLedger Fabric's behaviours was extremely difficult. I followed the HyperLedger documentation; the first problem I encountered was with Docker.

There is no command listed to install Docker, so once I typed Sudo apt install docker, it installed; however, this installed the incorrect command, and I should have used docker-composer. From there, I had to add my user to the account to the docker group. Once I completed this command, I went on to try and begin setup; however, this also failed as although it is not mentioned anywhere, you have to either restart the terminal or type a command to reload the bashrc file for the changes to take effect.

From there, I used curl to pull from the repository. Then, after receiving several warnings about code depreciation, I set up a variable to my install path in my current terminal session as per the documentation to allow me to call various binaries free of the installation directory. Thankfully, this worked without any errors.

I then moved into the test network directory only to be met with an assortment of errors regarding missing modules in various lines of the test network-scripts. I eventually figured out through trial and error that the node defaults on CLI to version 12.0, so I had to install NVM via node to update the node version to the latest v19.0 manually.

Upon running the network script again using node, I received fewer errors. Still, I received one about the scripts being incapable of finding a particular fabric module, and after much trial and error, I failed to fix it.

I rolled my virtual machine of Ubuntu 22.04 back to an older snapshot and started again, and thankfully this time, it worked as expected. I then tried to deploy the basic chaincode onto the test network, which errored every single line of the file.

As it turned out, the software needs to have to go installed to run; however, this is not explicitly stated in the pre-requisites section of the official documentation, and I only figured it out by reading over the command a few times as listed in the official documentation and breaking down what each parameter was doing.

Upon attempts to submit my assets to the network, I was plagued with docker errors about connections being refused. So, I restarted the VM without a full snapshot reset and decided to see if this fixed the issue. The remainder of setting up the test network was smooth sailing, with my assets successfully committed to the test network without any errors.

The deal breaker here was attempts to change the test scripts. Specifically trying to add new organisations. Although I could eventually add them, they were added statically and not dynamically, meaning other organisations could not be added or signed later.

7. Conclusions

A convincing argument can be presented as to why Hyperledger Fabric is a suitable blockchain for an auctioning system. It utilises some very secure and manageable design principles which better equip the user to correctly set up and control an enterprise blockchain system.

Having fallen victim to an eBay scam in the past, I feel this would help mitigate many issues regarding scams. Although completely preventing them may be impossible, a HyperLedger system could make revert the damages they cause easier. I also feel this system can be popular quickly using the lessons learned in the Ukrainian public auction system.

Another big reason presented from my research that this is suitable for this application is its low latency compared to other blockchains. All these factors present a much smaller set of possible headaches when the platform has launched.

However, I have serious concerns regarding several aspects of this project. First, the documentation needs to be completed, as the documentation I have used in the past is much more extensive in the specifics of the setup process. Second, online support is lacking due to the very select number of developers working with this framework.

Many support requests on Stack Overflow and other platforms like Reddit or YouTube are unfulfilled. Udemy courses are completely outdated as one of the most popular tools for developing on this Blockchain, composer, is depreciated. Although HyperLedger is popular in concept by many institutions, simply partaking in the setup process proves that it is not popular in application or support. Although it preaches a modularity and to make application specific blockchain systems easier to develop and use this is clearly not the case with those promises being encased in blunderingly bad documentation, poor overall community support and very little sample code to work from that’s up to date.

Ethereum, Cardano, Monero and Polygon although technically complex are also viable solutions. The best documented is Ethereum and to not waste further time I had directed my attention there. This also proved a disadvantage as I have to develop a full system and to just get through learning solidity, it’s a minimum time investment of 25 hours.

Hence this research has allowed me to conclude that the most flexible, time friendly option is to use Node JS and its package manager to build the application specific blockchain that Hyperledger had proposed using my own blockchain solution. Given the research I have collected above I have more than enough information to lay out a basic framework and to build on this framework into my own fully fledged system.

8. References

1.3 - Brief History of C++ - Object Oriented Programming with C++, Second Edition [Book] [WWW Document], n.d. URL https://www.oreilly.com/library/view/object-oriented-programming/9789332503663/xhtml/head-0045.xhtml (accessed 4.11.23).

5 Basic Components of A Blockchain Network [WWW Document], n.d. URL https://vietnamblockchain.asia/post/5666316/5-basic-components-of-blockchain (accessed 11.10.22).

About Monero | Monero - secure, private, untraceable [WWW Document], n.d. URL https://www.getmonero.org/resources/about/ (accessed 4.10.23).

About : Standard C++ [WWW Document], n.d. URL https://isocpp.org/about (accessed 4.11.23).

About Us | Polygon [WWW Document], n.d. URL https://polygon.technology/about (accessed 4.11.23).

Analysis of the online auction platform eBay. Advantages and disadvantages - GRIN [WWW Document], n.d. URL https://www.grin.com/document/503758 (accessed 11.22.22).

Awesome Duplicate Photo Finder - Find and Remove Duplicate or Similar Images [WWW Document], n.d. URL https://www.duplicate-finder.com/photo.html (accessed 11.11.22).

bips/bip-0039.mediawiki at master · bitcoin/bips [WWW Document], n.d. URL https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki (accessed 4.11.23).

Bitcoin - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Bitcoin (accessed 11.8.22).

bitcoin/src at master · bitcoin/bitcoin [WWW Document], n.d. URL https://github.com/bitcoin/bitcoin/tree/master/src (accessed 4.11.23).

Blockchain & Role of P2P Network | Blockchain Council [WWW Document], n.d. URL https://www.blockchain-council.org/blockchain/blockchain-role-of-p2p-network/ (accessed 11.9.22).

Blockchain - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Blockchain (accessed 11.7.22).

Blockchain: A Very Short History Of Ethereum Everyone Should Read | Bernard Marr [WWW Document], n.d. URL https://bernardmarr.com/blockchain-a-very-short-history-of-ethereum-everyone-should-read/ (accessed 4.11.23).

Blockchain Facts: What Is It, How It Works, and How It Can Be Used [WWW Document], n.d. URL https://www.investopedia.com/terms/b/blockchain.asp (accessed 11.8.22).

Blockchain Technology History: Ultimate Guide [WWW Document], n.d. URL https://101blockchains.com/history-of-blockchain-timeline/ (accessed 11.7.22).

Blockchain Use Cases in 2021: Real World Industry Applications | ConsenSys [WWW Document], n.d. URL https://consensys.net/blockchain-use-cases/ (accessed 11.8.22).

Buterin, V., n.d. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform.

Cardano | Home [WWW Document], n.d. URL https://cardano.org/ (accessed 11.8.22a).

Cardano | Home [WWW Document], n.d. URL https://cardano.org/ (accessed 4.11.23b).

Cardano (blockchain platform) - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Cardano_(blockchain_platform) (accessed 11.8.22).

Cardano Components | Cardano Developer Portal [WWW Document], n.d. URL https://developers.cardano.org/docs/get-started/cardano-components/#docusaurus_skipToContent_fallback (accessed 4.11.23).

Chaincode Tutorials — hyperledger-fabricdocs master documentation [WWW Document], n.d. URL https://hyperledger-fabric.readthedocs.io/en/release-1.3/chaincode.html (accessed 11.16.22).

Channels — hyperledger-fabricdocs main documentation [WWW Document], n.d. URL https://hyperledger-fabric.readthedocs.io/en/latest/channels.html (accessed 11.22.22).

Connecting your local wallet to your own daemon over Tor | Monero - secure, private, untraceable [WWW Document], n.d. URL https://www.getmonero.org/resources/user-guides/tor_wallet.html (accessed 4.10.23).

Consensus Algorithms in Blockchain - GeeksforGeeks [WWW Document], n.d. URL https://www.geeksforgeeks.org/consensus-algorithms-in-blockchain/ (accessed 11.8.22).

Consortium Blockchain | Bybit Learn [WWW Document], n.d. URL https://learn.bybit.com/glossary/definition-consortium-blockchain/ (accessed 11.11.22).

cryptogen — hyperledger-fabricdocs main documentation [WWW Document], n.d. URL https://hyperledger-fabric.readthedocs.io/en/latest/commands/cryptogen.html (accessed 11.15.22).

CryptoKitties - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/CryptoKitties (accessed 11.8.22).

Decentralized applications (dapps) | ethereum.org [WWW Document], n.d. URL https://ethereum.org/en/dapps/ (accessed 4.10.23).

Definition of Digital Assets - Gartner Finance Glossary [WWW Document], n.d. URL https://www.gartner.com/en/finance/glossary/digital-assets (accessed 11.9.22).

Digital Asset Definition [WWW Document], n.d. URL https://www.investopedia.com/terms/d/digital-asset-framework.asp (accessed 11.9.22).

Docker overview | Docker Documentation [WWW Document], n.d. URL https://docs.docker.com/get-started/overview/ (accessed 11.22.22).

Docker Registries - Aqua [WWW Document], n.d. URL https://www.aquasec.com/cloud-native-academy/docker-container/docker-registry/ (accessed 11.22.22).

Ethereum↔Polygon Bridge | Polygon Wiki [WWW Document], n.d. URL https://wiki.polygon.technology/docs/develop/ethereum-polygon/getting-started (accessed 4.11.23).

Ethereum - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Ethereum (accessed 11.7.22).

Ethereum Virtual Machine (EVM) | ethereum.org [WWW Document], 2023. URL https://ethereum.org/en/developers/docs/evm/ (accessed 4.10.23).

Exonum | Knowledge for policy [WWW Document], n.d. URL https://knowledge4policy.ec.europa.eu/foresight/tool/dlt4good/exonum_en (accessed 11.22.22).

Explained: What Is Hashing in Blockchain? | Bybit Learn [WWW Document], n.d. URL https://learn.bybit.com/blockchain/what-is-hashing-in-blockchain/ (accessed 11.11.22).

Fabric Gateway — hyperledger-fabricdocs main documentation [WWW Document], n.d. URL https://hyperledger-fabric.readthedocs.io/en/latest/gateway.html (accessed 11.22.22).

fabric-gateway | Go, Node and Java client API for Hyperledger Fabric v2.4+ [WWW Document], n.d. URL https://hyperledger.github.io/fabric-gateway/ (accessed 11.22.22).

Fanti, G., Venkatakrishnan, S.B., Bakshi, S., Denby, B., Bhargava, S., Miller, A., Viswanath, P., 2018. Dandelion++: Lightweight Cryptocurrency Networking with Formal Anonymity Guarantees. Performance Evaluation Review 46, 5–7. https://doi.org/10.1145/3219617.3219620

File:Bitcoin-Genesis-block.jpg - Wikimedia Commons [WWW Document], n.d. URL https://commons.wikimedia.org/wiki/File:Bitcoin-Genesis-block.jpg (accessed 11.16.22).

Five Hyperledger Blockchain Projects Now in Production – Hyperledger Foundation [WWW Document], n.d. URL https://www.hyperledger.org/blog/2018/11/30/six-hyperledger-blockchain-projects-now-in-production (accessed 11.25.22).

Formal Specification of the Plutus Core Language, n.d.

Get started | Polygon Wiki [WWW Document], n.d. URL https://wiki.polygon.technology/docs/develop/ethereum-polygon/matic-js/get-started (accessed 4.11.23).

Gossip data dissemination protocol — hyperledger-fabricdocs main documentation [WWW Document], n.d. URL https://hyperledger-fabric.readthedocs.io/en/latest/gossip.html (accessed 11.22.22).

Gossip Protocol | Binance Academy [WWW Document], n.d. URL https://academy.binance.com/en/glossary/gossip-protocol (accessed 11.22.22).

Guide to Hybrid Blockchain, Benefits and Use Cases [WWW Document], n.d. URL https://www.zeeve.io/blog/guide-to-hybrid-blockchain-benefits-and-use-cases/ (accessed 11.11.22).

Hashing Algorithms | Jscrambler Blog [WWW Document], n.d. URL https://blog.jscrambler.com/hashing-algorithms (accessed 11.11.22).

How Does Blockchain Work? Everything You Need to Know [Updated] [WWW Document], n.d. URL https://www.simplilearn.com/tutorials/blockchain-tutorial/blockchain-technology#the_process_of_transaction (accessed 11.11.22).

How does Python work?. A simple explanation of how Python code… | by Dhruvil Karani | Towards Data Science [WWW Document], n.d. URL https://towardsdatascience.com/how-does-python-work-6f21fd197888 (accessed 11.24.22).

How To Set Up An eBay Shop: Best Practices, Fees & Seller Tools. [WWW Document], n.d. URL https://www.linnworks.com/blog/how-to-set-up-an-ebay-shop (accessed 11.22.22).

Hybrid Blockchain- The Best Of Both Worlds [WWW Document], n.d. URL https://101blockchains.com/hybrid-blockchain/ (accessed 11.11.22).

Hyperledger Fabric - Components Overview - Hyperledger India Regional Chapter - Hyperledger Foundation [WWW Document], n.d. URL https://wiki.hyperledger.org/display/HIRC/Hyperledger+Fabric+-+Components+Overview (accessed 11.11.22).

input-output-hk/cardano-ledger: The ledger implementation and specifications of the Cardano blockchain. [WWW Document], n.d. URL https://github.com/input-output-hk/cardano-ledger#cardano-ledger (accessed 4.11.23).

input-output-hk/cardano-node: The core component that is used to participate in a Cardano decentralised blockchain. [WWW Document], n.d. URL https://github.com/input-output-hk/cardano-node#cardano-node-overview (accessed 4.11.23).

input-output-hk/ouroboros-network: An implementation of the Ouroboros family of consensus algorithms, with its networking support [WWW Document], n.d. URL https://github.com/input-output-hk/ouroboros-network/#ouroboros-network (accessed 4.11.23).

input-output-hk/plutus: The Plutus language implementation and tools [WWW Document], n.d. URL https://github.com/input-output-hk/plutus#plutus-core (accessed 4.11.23).

Introduction To Docker: A Beginner's Guide – BMC Software | Blogs [WWW Document], n.d. URL https://www.bmc.com/blogs/docker-101-introduction/ (accessed 11.22.22).

Introduction to Node.js [WWW Document], n.d. URL https://nodejs.dev/en/learn/introduction-to-nodejs/ (accessed 11.22.22).

JavaScript Asynchronous Programming and Callbacks [WWW Document], n.d. URL https://nodejs.dev/en/learn/javascript-asynchronous-programming-and-callbacks/ (accessed 11.22.22).

JD Launches Blockchain Open Platform - JD Corporate [WWW Document], n.d. URL https://jdcorporateblog.com/jd-launches-blockchain-open-platform/ (accessed 11.25.22).

JD.com Launches Blockchain Open Platform [WWW Document], n.d. URL https://coverager.com/jd-com-launches-blockchain-open-platform/ (accessed 11.25.22).

JD.com launches new blockchain platform to help enterprise customers | Supply Chain Magazine [WWW Document], n.d. URL https://supplychaindigital.com/technology/jdcom-launches-new-blockchain-platform-help-enterprise-customers (accessed 11.25.22).

Kiayias, A., Russell, A., David, B., Oliynykov, R., 2019. Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol.

LevelDB - Database of Databases [WWW Document], n.d. URL https://dbdb.io/db/leveldb (accessed 11.16.22).

maticnetwork/whitepaper: Matic whitepaper [WWW Document], n.d. URL https://github.com/maticnetwork/whitepaper/ (accessed 4.11.23).

May, W.E., 2012. FIPS PUB 180-4 FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION Secure Hash Standard (SHS) CATEGORY: COMPUTER SECURITY SUBCATEGORY: CRYPTOGRAPHY. https://doi.org/10.6028/NIST.FIPS.180-4

Monero price: XMR to USD chart | Ledger [WWW Document], n.d. URL https://www.ledger.com/coin/price/monero (accessed 4.10.23).

Möser, M., Soska, K., Heilman, E., Lee, K., Heffan, H., Srivastava, S., Hogan, K., Hennessey, J., Miller, A., Narayanan, A., Christin, N., 2018. An Empirical Analysis of Traceability in the Monero Blockchain. Proceedings on Privacy Enhancing Technologies 2018, 143–163. https://doi.org/10.1515/POPETS-2018-0025

Nakamoto, S., n.d. Bitcoin: A Peer-to-Peer Electronic Cash System.

Node.js - Introduction [WWW Document], n.d. URL https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm (accessed 11.22.22).

openIDL: an open blockchain network for insurers [WWW Document], n.d. URL https://openidl.org/ (accessed 11.25.22).

paritytech/polkadot: Polkadot Node Implementation [WWW Document], n.d. URL https://github.com/paritytech/polkadot (accessed 4.11.23).

Peer-to-peer - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Peer-to-peer (accessed 11.9.22).

performance - Fastest algorithm to detect duplicate files - Stack Overflow [WWW Document], n.d. URL https://stackoverflow.com/questions/53314863/fastest-algorithm-to-detect-duplicate-files (accessed 11.11.22).

Polygon PoS | The most efficient blockchain protocol [WWW Document], n.d. URL https://polygon.technology/polygon-pos (accessed 4.11.23).

Polygon-SDK : Polygon Support [WWW Document], n.d. URL https://support.polygon.technology/support/solutions/articles/82000886636-polygon-sdk (accessed 4.11.23).

Python - Overview [WWW Document], n.d. URL https://www.tutorialspoint.com/python/python_overview.htm (accessed 11.24.22).

Rebuilding citizen trust in government e-auctions [WWW Document], n.d. URL https://exonum.com/story-ukraine (accessed 11.22.22).

research-lab/whitepaper.pdf at master · monero-project/research-lab [WWW Document], n.d. URL https://github.com/monero-project/research-lab/blob/master/whitepaper/whitepaper.pdf (accessed 4.10.23).

Role of Peers in Hyperledger Fabric Blockchain | by Chetan Gadgilwar | Medium [WWW Document], n.d. URL https://medium.com/@chetan.gadgilwar/role-of-peers-in-hyperledger-fabric-blockchain-fe383d8ee84c (accessed 11.11.22).

Rust Programming Language [WWW Document], n.d. URL https://www.rust-lang.org/ (accessed 4.11.23).

Satoshi Nakamoto - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Satoshi_Nakamoto (accessed 11.8.22).

Smart contract - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Smart_contract (accessed 11.8.22).

Solidity — Solidity 0.8.19 documentation [WWW Document], n.d. URL https://docs.soliditylang.org/en/v0.8.19/ (accessed 4.10.23).

Supply Chain Transparency [WWW Document], n.d. URL https://thegemba.com/article/how-walmart-used-blockchain-to-increase-supply-chain-transparency (accessed 11.25.22).

tevador/RandomX: Proof of work algorithm based on random code execution [WWW Document], n.d. URL https://github.com/tevador/randomx (accessed 4.10.23).

Triptych Prioritization Meeting: 21 April 2021, 17:00 UTC · Issue #568 · monero-project/meta [WWW Document], n.d. URL https://github.com/monero-project/meta/issues/568 (accessed 4.10.23).

Two Ways to Generate Crypto Materials in Hyperledger Fabric: Cryptogen and CA Server | by KC Tam | Medium [WWW Document], n.d. URL https://kctheservant.medium.com/two-ways-to-generate-crypto-materials-in-hyperledger-fabric-cryptogen-and-ca-server-36d3c3e2daad (accessed 11.15.22).

Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP [WWW Document], n.d. URL https://www.foley.com/en/insights/publications/2021/08/types-of-blockchain-public-private-between (accessed 11.16.22a).

Types of Blockchain: Public, Private, or Something in Between | Blogs | Manufacturing Industry Advisor | Foley & Lardner LLP [WWW Document], n.d. URL https://www.foley.com/en/insights/publications/2021/08/types-of-blockchain-public-private-between (accessed 11.10.22b).

Understanding eBay Buying Options [WWW Document], n.d. URL https://www.liveabout.com/understanding-ebay-auction-options-1140295 (accessed 11.22.22).

Walmart Case Study – Hyperledger Foundation [WWW Document], n.d. URL https://www.hyperledger.org/learn/publications/walmart-case-study (accessed 11.25.22).

What Are Blockchain Nodes and How Do They Work? | Built In [WWW Document], n.d. URL https://builtin.com/blockchain/blockchain-node (accessed 11.8.22).

What are smart contracts on blockchain? | IBM [WWW Document], n.d. URL https://www.ibm.com/topics/smart-contracts (accessed 11.8.22a).

What are smart contracts on blockchain? | IBM [WWW Document], n.d. URL https://www.ibm.com/topics/smart-contracts (accessed 11.8.22b).

What are the 4 different types of blockchain technology? [WWW Document], n.d. URL https://www.techtarget.com/searchcio/feature/What-are-the-4-different-types-of-blockchain-technology (accessed 11.10.22).

What Does Proof-of-Stake (PoS) Mean in Crypto? [WWW Document], n.d. URL https://www.investopedia.com/terms/p/proof-stake-pos.asp (accessed 4.11.23).

What Is a Blockchain Consensus Algorithm? | Binance Academy [WWW Document], n.d. URL https://academy.binance.com/en/articles/what-is-a-blockchain-consensus-algorithm (accessed 11.8.22).

What is Blockchain? | Oracle Middle East Regional [WWW Document], n.d. URL https://www.oracle.com/middleeast/blockchain/what-is-blockchain/ (accessed 11.8.22).

What is blockchain and how does it work? [WWW Document], n.d. URL https://www.techtarget.com/searchcio/definition/blockchain (accessed 11.11.22).

What is CouchDB and Why Should I Care? [WWW Document], n.d. URL https://www.infoq.com/articles/warner-couchdb/ (accessed 11.16.22).

What is Docker Engine? - Definition from WhatIs.com [WWW Document], n.d. URL https://www.techtarget.com/searchitoperations/definition/Docker-Engine (accessed 11.22.22).

What is eBay? Dow Does eBay Work? [WWW Document], n.d. URL https://ecommerce-platforms.com/glossary/what-is-ebay (accessed 11.22.22).

What is ether (ETH)? | ethereum.org [WWW Document], n.d. URL https://ethereum.org/en/eth/ (accessed 4.10.23).

what is Genesis Block and why Genesis Block is needed? | by Tecra Space | Medium [WWW Document], n.d. URL https://tecracoin.medium.com/what-is-genesis-block-and-why-genesis-block-is-needed-1b37d4b75e43 (accessed 11.16.22a).

what is Genesis Block and why Genesis Block is needed? | by Tecra Space | Medium [WWW Document], n.d. URL https://tecracoin.medium.com/what-is-genesis-block-and-why-genesis-block-is-needed-1b37d4b75e43 (accessed 11.16.22b).

What is Go? Golang Programming Language Meaning Explained [WWW Document], n.d. URL https://www.freecodecamp.org/news/what-is-go-programming-language/ (accessed 11.25.22).

What is Java? - Enterprise Java Beginner's Guide - AWS [WWW Document], n.d. URL https://aws.amazon.com/what-is/java/ (accessed 11.25.22).

What is Java? Definition, Meaning & Features of Java Platforms [WWW Document], n.d. URL https://www.guru99.com/java-platform.html (accessed 11.24.22).

What is Matic Network? (MATIC) Blockchain Scaling Platform [WWW Document], n.d. URL https://blockonomi.com/matic-network-guide/ (accessed 4.11.23).

What is MD5 (MD5 Message-Digest Algorithm)? [WWW Document], n.d. URL https://www.techtarget.com/searchsecurity/definition/MD5 (accessed 4.11.23).

What is the Go Programming Language? [WWW Document], n.d. URL https://www.techtarget.com/searchitoperations/definition/Go-programming-language (accessed 11.25.22).

What is the Python programming language? [WWW Document], n.d. URL https://www.techtarget.com/whatis/definition/Python (accessed 11.24.22).

Zcash - Wikipedia [WWW Document], n.d. URL https://en.wikipedia.org/wiki/Zcash (accessed 11.7.22).

Zcash's Zero Knowledge Proofs, ZK Snarks, and More | Gemini [WWW Document], n.d. URL https://www.gemini.com/cryptopedia/zcash-zero-knowledge-proof-zk-snarks-mining (accessed 11.10.22).

CETAM - setam.net.ua [WWW Document], n.d. URL https://setam.net.ua/ (accessed 4.11.23).