

2022

# A Treatise on Blockchain and Healthcare

U1857977

ISMAIL KAMRAN

## Contents

<i>The Problem</i> .....	2
<i>A possible solution</i> .....	3
<i>The proposed solution</i> .....	3
<i>Understanding how and why it works</i> .....	3
<i>Web3 Architecture</i> .....	6
<i>Software Development Methodology for Prototype</i> .....	8
<i>Professional, Social, Legal, Ethical Issues:</i> .....	14
Professional: .....	14
Social: .....	15
Legal .....	18
Ethical: .....	18
<i>Implementation</i> .....	19
<i>Evaluation</i> .....	24
<i>Conclusion</i> .....	25
<i>Appendix</i> .....	25
<i>Bibliography</i> .....	29

## *The Problem*

Good healthcare data management is imperative for comprehensive medical care. The healthcare industry is undergoing a technological revolution. Internet of Things (IoT) devices are collecting data from patients more than ever before, this combined with the data collected from conventional procedures such as CT scans, MRIs, X-Ray reports and many other types of test reports, has amounted to the fact that the healthcare industry is generating 'approximately 36% of the world's data volume' (RBC Capital Markets, 2021) which is the largest out of any industry. As the amount of valuable medical data increases, the challenges for storing it securely, safely and accessibly has also increased considerably.

Currently, many healthcare organisations opt for on-site storage (RBC Capital Markets, 2021), they do this because it provides complete control over their data, and it has low risk of downtime and latency issues. However, there are many problems with this approach and these will be discussed as follows.

Healthcare organisations have to arrange for physical space within the premises to host the servers. In some larger cities, the cost of acquiring this physical space within the hospital can be very significant. As the amount of data needed to be stored increases, the amount of space needed will only increase, therefore increasing the cost. Furthermore, there is the cost of maintaining these servers, they require a continuous power supply, cooling mechanisms, backup batteries, and the cost of trouble shooting any faulty hardware. All of this, leads to a very considerable cost.

The on-premise data storage infrastructure implemented by most healthcare organisations may seem secure because all the data is on a local network, but in reality, there is still a risk of human error. Hundreds of staff work in such organisations, and one cannot ignore the huge risk of the highly sensitive data being exposed to phishing and malware attacks despite the regular upkeep of firewalls and anti-viruses. A single mistake, could lead to a data breach of an entire data centre.

The argument that healthcare organisations use to justify their implementation, as mentioned before is due to the apparent low risk of downtime and latency issues. However, it is important to be aware that any technical issue with the onsite hardware, or geographical disaster can create disastrous downtime issues, which may cost patient lives. Considering the healthcare organisations recovery from downtime depends on their very limited IT resources, it would likely be difficult for them to do so quickly. Moreover, the high volume of data on potentially out of date systems is likely to lead to latency issues, wasting the time of NHS staff, reducing the time they have to work with patients.

Moreover, the systems are simply not scalable. They require an upfront cost, and they don't usually have a future proof strategy for their infrastructure. When they do have some sort of future proof strategy, it is excessive as they have to purchase technology with high performance, only anticipating the future need to use it.

Healthcare data is critical for ensuring that the patient receives the most accurate medical care. Not having quick access to patient data can significantly impact the quality of care given to the patients.<sup>1</sup>

---

<sup>1</sup>

Therefore, it is vital that a solution is found in order to improve the healthcare industry and the service they provide as a whole.

### *A possible solution*

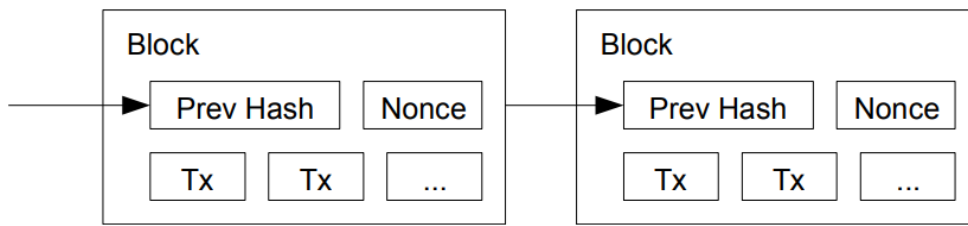
One solution that has been proposed is cloud storage. It is more scalable, if more data is needed to be stored, more cloud space can be purchased simply at a low cost. There is also a low maintenance and start-up cost. Data can become accessible more easily from multiple locations. However, there are issues with this storage infrastructure. There is a lack of control over the data, trusting a cloud storage provider such as AWS to implement the necessary protections to prevent the data from being exposed, and to ensure that all the necessary services experience no downtime is a risk. Furthermore, this solution does not protect against human error resulting in loss of data, or unintended/malicious alterations. The best that can be hoped for is backups, but even with regular backups it is still possible to lose some data.

### *The proposed solution*

The best and most innovative solution is Blockchain based. All of the above problems and issues can be easily solved by programming the back-end using smart contracts on a Blockchain network. There is no upfront cost for physical storage space, there are no concerns for downtime or latency issues, and it is virtually impossible for data to be lost, despite human error. Moreover, access to sensitive data can be tracked and monitored. There is no limit for storage either. The reason for all of these advantages is because Blockchains are immutable and are made up of a network of thousands upon thousands of nodes. If a single node has downtime, it will make a net zero impact on the network as a whole. Furthermore, some Blockchains confirm 50,000 – 60,000 Transactions Per Second (TPS) such as Solana (Cryptopedia, 2021), which translates to instant confirmations, resolving the concern of latency issues. There are many Blockchains that would be a good network to build such a healthcare data management system on in terms of speed and scalability, however 90% of Blockchains are built to be Ethereum Virtual Machine (EVM) compatible (Capitalgram, 2021). Therefore, it is most apt to build the application using Solidity (the programming language of the EVM) and then in the future the code can be deployed to the most suitable Blockchain or Layer 2 solution at the time.

### *Understanding how and why it works*

In order to understand Blockchain, one must first understand the first Blockchain that came into existence - Bitcoin, invented by the alias Satoshi Nakamoto (Satoshi Nakamoto 2021). Bitcoin works on the most secure and well-tested consensus mechanism in Blockchain known as 'Proof of Work'. The way this mechanism works is that each transaction is initially allocated to a block of transactions in the 'mempool'. The mempool is a collection of requested transactions that have yet to be confirmed by the network. Each transaction has a fee set by the sender of the transaction, the higher the fee, the faster the 'miners' will include the transaction as a part of their block. A block is approximately 1mb and contains about 4000 transactions. A miner node fills up a block of the most profitable transactions from the mempool and starts to hash them using the SHA256 encryption algorithm. The hashed block includes the hash of the previous block, and the details of the current block. The previous hash also includes the hash of the block before it and the chain continues till the very first block.



Satoshi Nakamoto (2021) Figure 1

Therefore, if a node was to try to claim that a certain transaction happened in the past, it would end up with a completely different hash, so the only way it would end up with the same hash is if it agrees to the same version of transaction history. There is also a possibility of malicious nodes entering the network and attempting to claim a completely different transaction history all together. The Bitcoin network's security mechanism for this is called 'Proof of Work' and is explained as follows.

Bitcoin has a certain 'difficulty' level. This means that a mining node does not simply hash a block once, rather it continues to hash the block repeatedly, until the hash has a variable number of leading zeroes called 'nonces'. This process of repeated hashing with the correct difficulty should take approximately ten minutes. Mining nodes only accept the longest chain available as it must have had the most computational hashing of the blocks (with the correct difficulty), and consequently must be the honest chain. The honest chain always grows the fastest, and therefore it is virtually impossible to create a longer dishonest chain with an incorrect transaction. The reason the honest chain grows the fastest is explained by Satoshi Nakamoto in his white paper:

*'The majority decision is represented by the longest chain, which has the greatest proof-of-work effort invested in it. If a majority of CPU power is controlled by honest nodes, the honest chain will grow the fastest and outpace any competing chains. To modify a past block, an attacker would have to redo the proof-of-work of the block and all blocks after it and then catch up with and surpass the work of the honest nodes.'* (Satoshi Nakamoto. 2021).

Thus, the longest chain is the chain which has the most CPU power invested in it, and it will be the fastest because the more CPU power there is, the faster the mining node will hash the transactions correctly with the given difficulty. Thus, the only way to successfully claim a false transaction on the network is to control the majority (51%) of the hash power in the network. However, with approximately one million individual Bitcoin miners across the world (Bitcoin Worldwide, 2021) the amount of CPU power required to mine a new block (and accordingly confirm a transaction) is 176.16M Transaction Hashes per Second (TH/s) (Ychart, 2021). It is virtually impossible to match 51% of such a high hash rate, whilst also hashing all of the previous blocks to create a chain from the 'genesis' (original) block. Even if it was possible to gain such a high level of computational power, there would be more of a financial incentive to secure the network rather than attack it. This is because there is a considerable financial reward for the fastest mining node which is how new Bitcoin is introduced into circulation.

Due to the large amount of Bitcoin nodes, and the fact that Bitcoin nodes can work on any form of communication (e.g. internet, radio etc) it is virtually impossible to ban it. Even if an authority managed to reduce the number of miners, and therefore the hash rate – similar to what happened in China (Sigalos, 2021), the Bitcoin difficulty would simply reduce in accordance to the hash rate, and that would incentivise new miners to start processing transactions due to an increase in the

likelihood of financial reward. Accordingly, when China banned the mining of Bitcoin, new miners from the US increased the hash rate back to what it was before in less than a few months. Even if the hash rate had not increased back to its highest, it would still be virtually impossible to successfully attack the network. Bitcoin, in its history has never been hacked nor shut down (Bitpanda, 2021) due to the technological aspects of it mentioned above. The purpose of this explanation is to clarify how the blockchain technology proposed in this treatise has effectively zero chance of failing or shutting down, and thus would be an ideal technology to build upon to avoid the current issues the healthcare system faces. This is a brief summary of the technological aspect of Bitcoin; however, it is encouraged to read the Bitcoin whitepaper (Nakamoto, 2021) to get a more in-depth overview.

In 2013, Vitalik Buterin suggested a change to the Bitcoin source code. He suggested that Bitcoin should be made to allow loops, and hence become Turing Complete. The Bitcoin development community rejected this idea mainly because it would pose a high risk of the network being spammed with loops. Theoretically, a transaction with a loop could run infinitely which would mispend CPU resources of the Bitcoin nodes. However, Vitalik came up with a solution where users pay for computation in 'gas' before it is executed on the network. This resolved the high risk of network spam. Still, many disagreed with Vitalik (Hacker News, 2018), so he released the 'Ethereum Whitepaper' (Vitalik Buterin, 2013), in 2013 he stated he wanted to change four 'limitations' about Bitcoin. The first of them was the lack of Turing Completeness. The others were 'value-blindness', 'lack of state', and 'blockchain blindness'. They essentially mean that Bitcoin is not transparent enough, as we cannot directly retrieve an account's balance from the blockchain on Bitcoin, rather we have to use oracles to follow UTXOs (Unspent Transaction Outputs), so Vitalik removed the whole concept of UTXOs. In 2015 Ethereum was released, as essentially the Turing Complete version of Bitcoin. The way this Turing Completeness was achieved was through the creation of the Ethereum Virtual Machine (EVM). Since then, the EVM has become the standard in blockchain development, despite other faster and cheaper blockchains arising, 90% of them use the EVM. Ethereum currently uses a proof of work consensus model similar to Bitcoin, however there are plans for it to move to Proof of Stake during 2022.

Proof of Stake is similar to Proof of Work as explained above, except the main difference is that Proof of Work mining requires huge amounts of energy consumption to fuel computational power; Proof of Stake gives mining rewards based on the percentage of coins held by a miner, not by actual continuous hashing. There are also other consensus mechanisms that have been developed, but ultimately, they are all modifications of Proof of Work.

Due to the high security, reliability and development support provided by Ethereum, it is ideal to build on there. However, currently Ethereum is slow and very expensive. The gas fees for a single transaction are \$23 (Ethereum Gas Tracker, 2013). This is far too expensive to run a healthcare data management system, thus there are three solutions to this problem.

The first solution is to deploy the smart contracts onto a Layer 2 solution such as Arbitrum or Polygon. These solutions work on top of the Ethereum network allowing transactions to become cheap and fast, all whilst retaining the same benefits of decentralisation.

The second solution is to deploy the smart contracts on to another EVM compatible blockchain. There are many options which would work perfectly well. Examples include Solana, Avalanche, Fantom, and the Binance Smart Chain. All of these are excellent options, but they are not as decentralised as Ethereum, so there could be a minute chance of security or downtime issues.

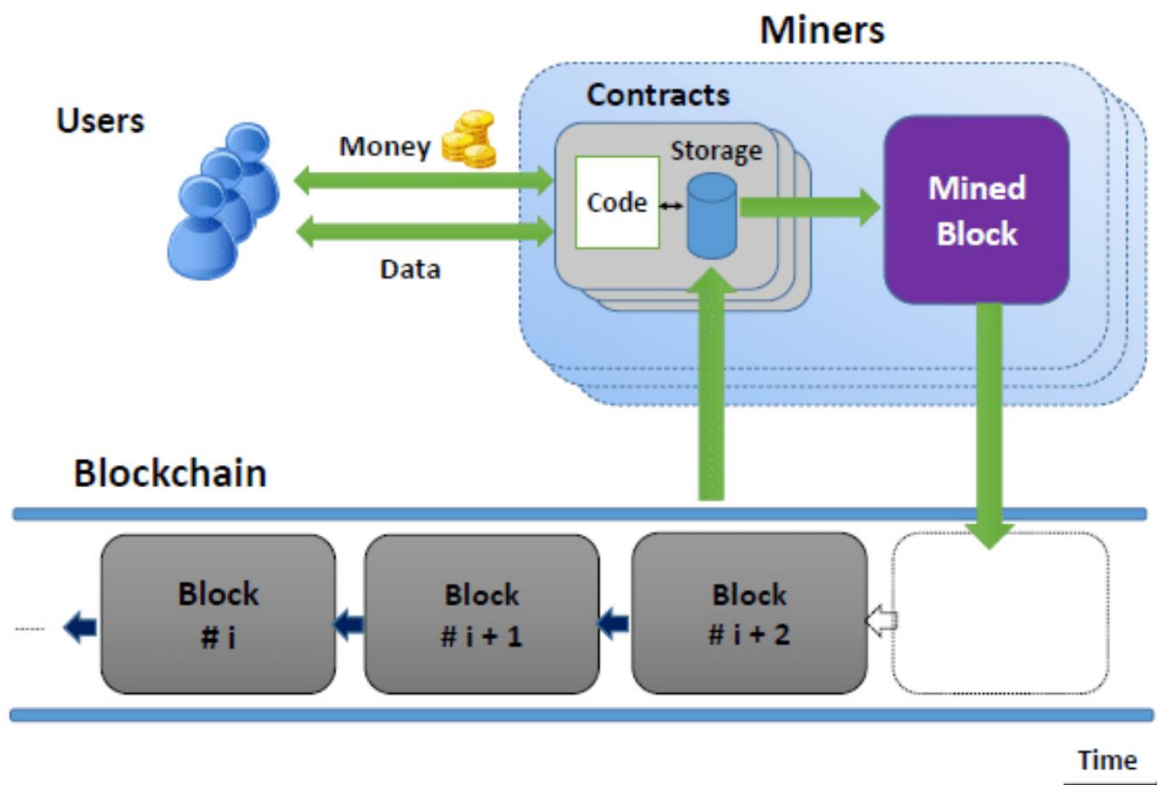
However, Solana is the only one that has ever experienced downtime from those listed (Bloomberg, 2021). So, overall that is positive for these alternative blockchain networks.

The third solution is to wait for Ethereum to move to Proof of Stake, before deploying the smart contracts onto the Ethereum Mainnet. Then, Ethereum will become fast and cheap enough to run a healthcare data management system.

For the purposes of this project, I will create the application on the Ethereum Testnet 'Görli. It is a Proof of Authority (another variation of proof of work) testnet run on Ethereum and it is the fastest and most stable of the testnets. The benefit of using a testnet is that Ether (the currency on Ethereum) is free, and so the application can be tested for free. Later, I can deploy to any of the above solutions as necessary should I deploy the application to production.

Finally, the intended cost of an individual transaction would be a small fraction of a cent per transaction. Polygon's gas fees are 0.002\$ per transaction (on average). The cost of running the application will ultimately depend on the computational complexity of each transaction, but assuming there are 1000 transactions per day from a fairly large healthcare institution, that would only cost them 2 dollars per day on average.

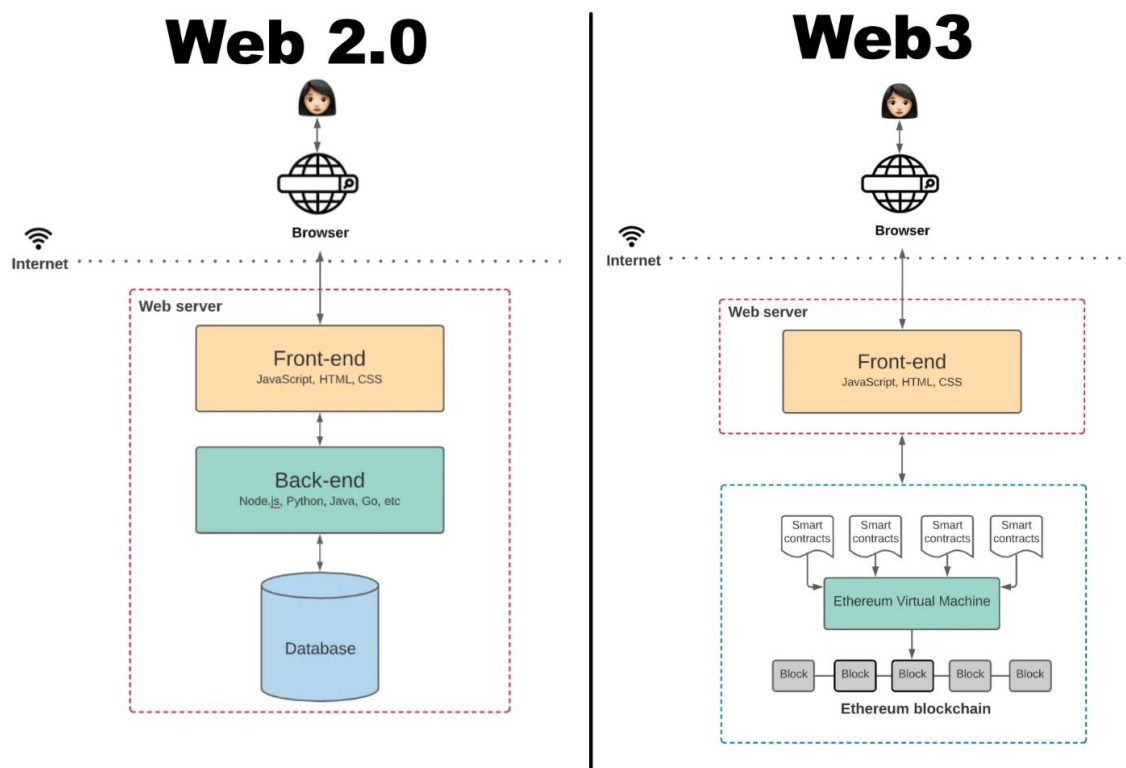
### *Web3 Architecture*



(Eze et al., 2017)

Figure 2

The structure in which smart contracts and decentralised applications are developed is described in figure 2 above. Users send data to a smart contract on the blockchain, which is then stored within the smart contract itself (not an external database). The new change in state of the smart contract is then hashed into the new block, continuing the chain.



(erwinkarim, 2021)

Figure 3

According to (Sharma, 2018) the progression of the world wide web is understood in three stages of development. The first stage is Web 1.0, which was essentially just static websites, providing information to users without any interaction. Then JavaScript was developed commencing the era of Web 2.0. This allowed content to be responsive to user input, as well as information to be dynamically displayed.

Consequently, due to the development of decentralised digital products, many hold the view that this is now the era of Web3 (ajammin, 2022). This is because value is now transactable over the internet without an intermediary. Web3 is the first time that a picture on a screen (i.e. NFTs) can hold value because there is now a decentralised way to verify ownership. This is a significant change to the web from Web 2.0.

In figure 3 above, it is clear that there is a significant architectural difference in the back-end of both architectures. Web3 does not have a separate database, rather it is integrated into the backend inside the smart contracts. The smart contracts are then run on the EVM, and the EVM is essentially a virtual operating system, but for all the nodes on the network.



The EVM operated as a stack machine (The ENS Team, 2022). The EVM runs as a stack machine with 1024 elements in depth. Each item is a 256-bit word that was chosen for its compatibility with 256-bit encryption.

The EVM retains transitory memory (as a word-addressed byte array) during execution, which does not remain between transactions. Contracts, on the other hand, include a word-addressable word array called Merkle Patricia which is connected with the account in question and is part of the global state. Compiled smart contract bytecode runs as a series of EVM opcodes that perform conventional stack operations like as XOR, AND, ADD, and SUB. The EVM also implements some blockchain-specific stack operations, such as ADDRESS, BALANCE, and BLOCKHASH.

Each node has a copy of the entire blockchain, and therefore all smart contracts, therefore many parties will be able to view the encrypted bytecode of the smart contracts, however it is unlikely that bytecode can be reverse engineered to reveal original code, unless it is a simple application where basic opcodes are translated back into Solidity. Therefore, copyright and security can be preserved better.

## *Software Development Methodology for Prototype*

(UpTech, 2022) explains the prototype model. It is a model which allows developers to work on a prototype version of the eventual product rather than producing the complete software. Customers can then test, evaluate, and provide comments on the prototype.

The prototype goes through multiple stages of development based on the feedback received until the buyer is satisfied. The appeal of the prototype technique is its thorough review, which identifies potential problems before actual development begins.

The effectiveness of this technique depends not just on the development team, but also on the communication with potential stakeholders.

There are many positives of choosing this model:

- Before beginning the genuine development work, it is beneficial to make sure the healthcare institutions are pleased with the product.
- Good in identifying and resolving possible problems early in the development process, lowering the chance of product failure. This is important as a blockchain application failing could be devastating, a prototype lowers this risk.
- Early on in the project, establish a connection with the healthcare institutions through talks.
- With the prototype, collect specific information that will be utilised to construct the final version.

(UpTech, 2022) mentions that there some negatives of the prototype model such as:

- Excessive back and forth with the client when testing the prototype might cause the development timetable to slip.
- Customers' expectations of the finished product may differ from the prototype.

- Because the developer frequently pays for prototype work, there is a possibility of cost overrun.

Since there are many unknowns with the proposed prototype software, it is most suitable to choose the prototype model, as the positives far outweigh any potential negatives.

## System Design

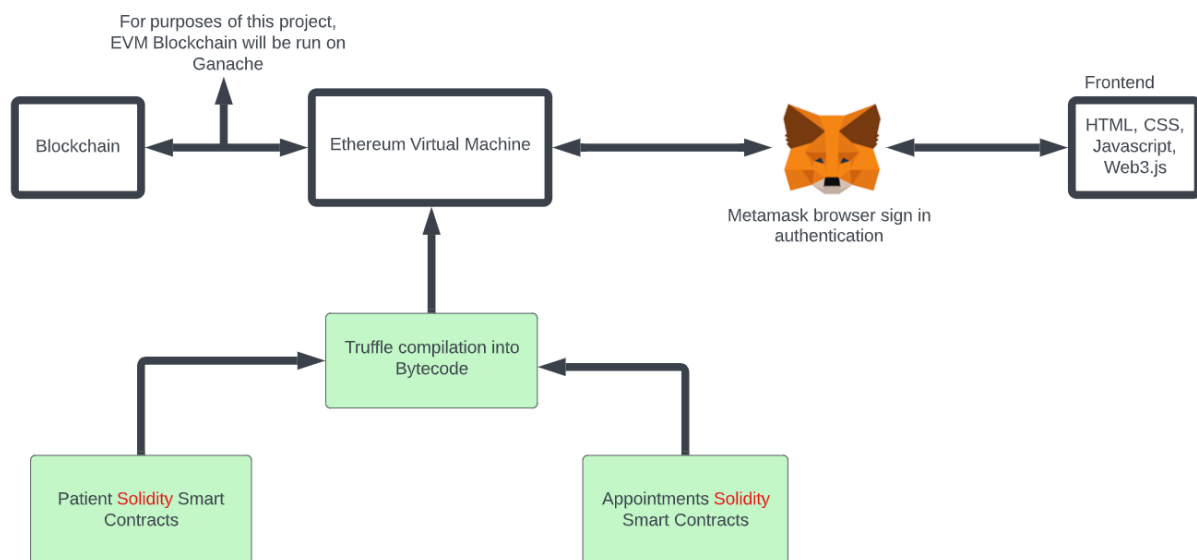


Figure 4 – Architecture of proposed prototype

For the backend, there will be two smart contracts that will act as classes. They are the Patient Contract and the Appointment Contract. All the data will be contained within these two contracts in structs and mappings. There will also be CRUD (Create, Read, Update, Delete) functions in these contracts. Truffle will compile these contracts into bytecode, and store them in JSON files. These JSON files will deploy onto the blockchain. Once deployed, they cannot be changed; such is the design of the immutable blockchain. Any updates to the contracts would require redeployment.

The deployments will be to Ganache. Ganache starts a local blockchain node on the development PC, which essentially imitates a public live blockchain node. It 'enables development, deploying and testing in a safe and deterministic environment' (Truffle Suite, 2022).

The Ganache EVM will use MetaMask as the authentication gateway between the front end and the smart contracts.

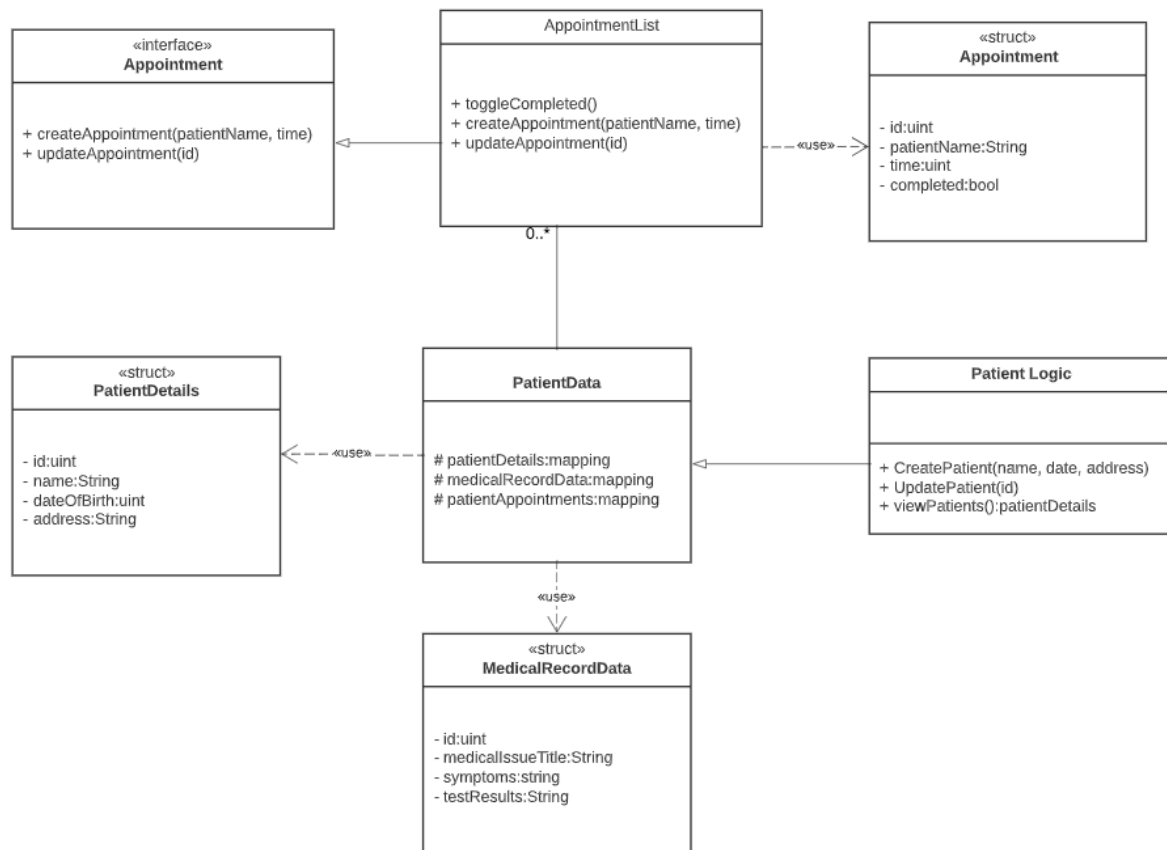


Figure 5 – Contract/Class diagram

The Appointment contract from figure is be divided into an interface contract and AppointmentList contract. This is to allow other external systems to implement their own way of handling appointments, by connecting their own contracts and inheriting from the interface. Not all healthcare institutions may want to handle their appointments in the exact same way, so this design feature allows flexibility for other implementations. This is possible because interfaces in Solidity are external, and can be inherited by any smart contract on the blockchain (Ethereum Foundation, 2022).

The AppointmentList contract will then use the Appointment struct, and store it in a mapping. The PatientData contract will then access the public mapping and use the data to match patients with appointments using patient details in the Appointment struct.

The Patient contract from figure 4 is also split into two contracts. The PatientLogic contract inherits from the PatientData contract. Solidity compiles inherited contracts together into a single contract in the bytecode. The purpose of this split is to produce better organised code.

The PatientData contract uses two different structs, medicalRecordData, and PatientDetails. The PatiendLogic contract will be able perform Create and Update operations on the mappings which hold these structs.

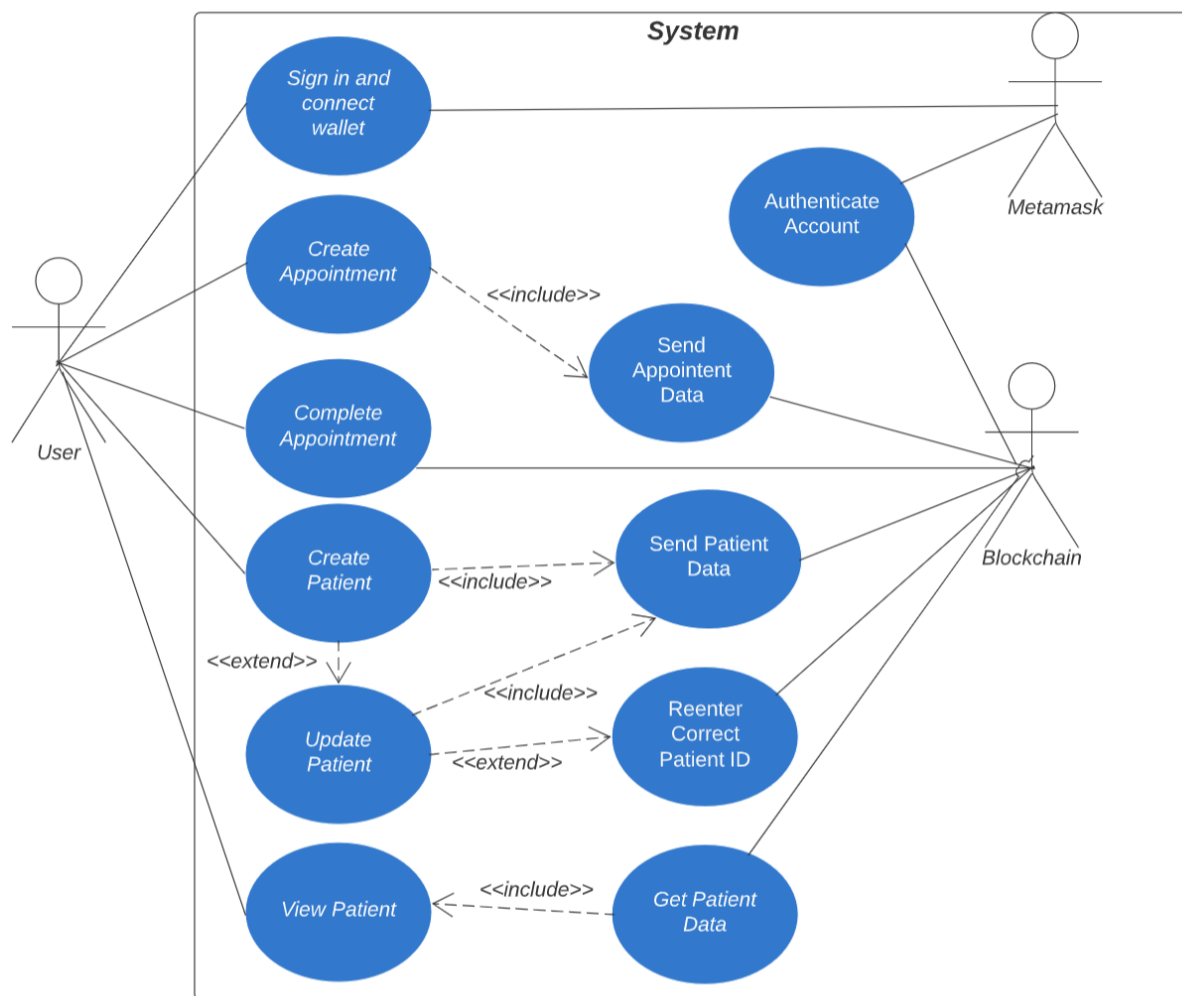


Figure 6 – Use case Diagram

The user signs in and connects his wallet to the blockchain. Metamask confirms the authentication, and fulfils the connection. Then the user is able to carry out operations such as creating an appointment, completing it, creating a patient, updating a patient, and viewing all of the data.

The create and update functions require sending the data to the blockchain (i.e. Ganache). It is possible that the Ganache update patient function may have an incorrect ID entered. In such case, the user will be prompted to enter a valid ID.

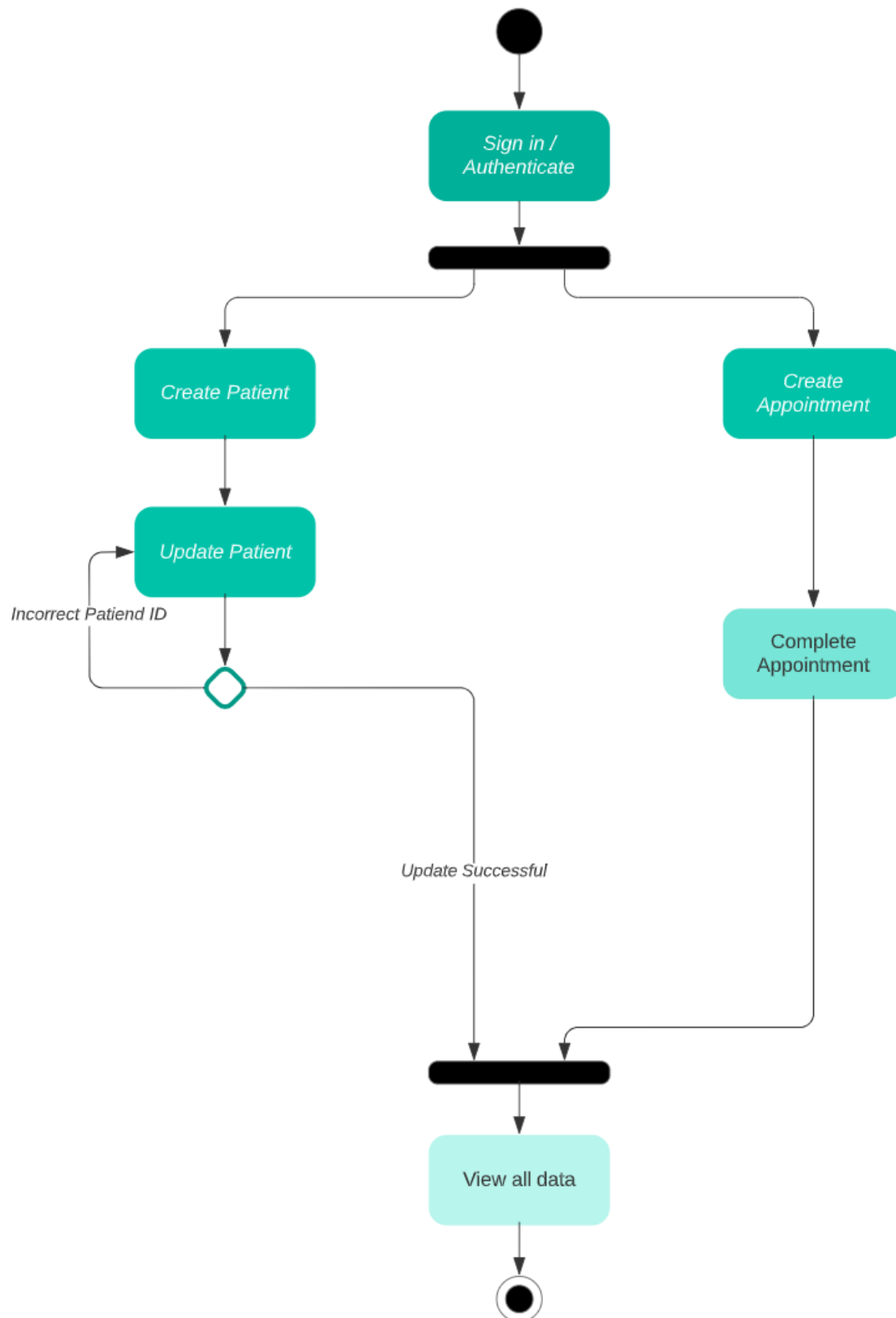


Figure 7 – Activity Diagram

The user can create appointments and create patients, however the webpage will display the functions in different sections, hence why there is a separate workflow for each.

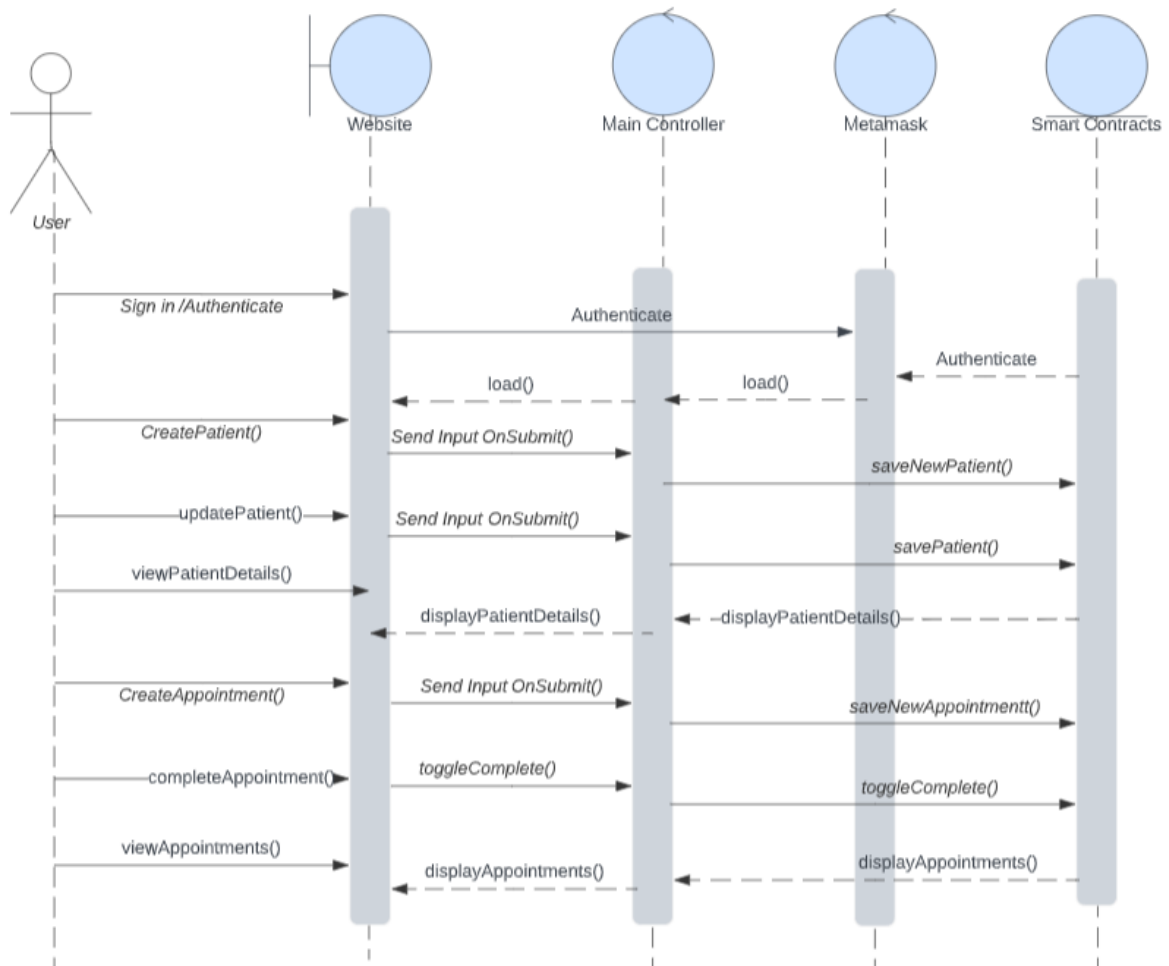


Figure 8 – Sequence Diagram

The key difference between figure 8 and figure 6 is the demonstration of the Model View Controller architecture within the application. The main controller is a JavaScript file, which runs on the web3.js framework. Metamask also acts as a controller because it is essentially intermediary logic and authentication between the front end and wallet balances on the blockchain.

## *Professional, Social, Legal, Ethical Issues:*

Overall, the purpose of this entire project is to preserve the data of patients. To keep them private, and ensure that sensitive patient data does not go to potentially malicious individuals who profit from medical data. Therefore, the aim of this project is based on Professional, Legal, and Ethical motives. However, there are some social challenges; all of which will be explained below.

### *Professional:*

(The General Medical Council, 2022) have stated that 'Every health and care professional must be open and honest with patients and people in their care'. This quote is the first line of the professional duty of candour, and thus clearly asserts that honesty and openness are key professional values in the healthcare industry. It would then be pertinent to mention that 'Bitcoin was designed to be the foundation for an open and honest system' (Porras & Daugherty, 2021). Therefore, the intended purposes behind blockchain itself, is in line with the professional goals of this industry.

However, professionalism is not limited to worthy goals. It must be addressed that there are still some concerns with blockchain technology. The greatest of these concerns is smart contract security. Despite the fact that it is 'virtually impossible for any third party to interfere with Ethereum' (What Is Ethereum, 2022) due to the inherently secure design of the blockchain, the same cannot be said for smart contracts and DApps (Decentralised Applications).

The largest smart contract security breach was a DeFi (Decentralised Finance) hack on Poly-Network where \$600 million dollars was stolen (Browne, 2021). The issue here is not the underlying blockchain technology, but it is the substandard coding of smart contracts. 'Ethereum has reportedly over 32,000 smart contracts that are vulnerable to hacking due to poor coding' (Capitol Technology University, 2020). It is unprofessional for software engineers to produce substandard code. Rather, from the definitions of a professional software engineer is that 'a professional programmer does high quality work' (Greenspun, 2000). Consequently, if this project is to be delivered to market, it is necessary that smart contract security auditing is carried out by independent smart contract security experts during the final implementation, alongside the successful hiring of professional software engineers.

Additionally, the Ethereum team have been improving the Solidity programming language, releasing new versions regularly (*Solidity — Solidity 0.8.13 Documentation*, 2021). From the breaking changes there has been an emphasis on the removal potential vulnerabilities in the code. For example, before Solidity v0.5, integer variables had an overflow vulnerability and users had to import an 'OpenZeppelin SafeMath' contract to avoid the issue. Another example is the 'reentrancy attack' where the fallback function of a smart contract is called repeatedly as a way to steal funds. This vulnerability has now also been fixed, but security is increased by using the set standards set by OpenZeppelin, whom have created standard contracts like the ERC-20 contract for tokens, and an ERC-721 contract for NFTs. When developers use these thoroughly tested contracts, rather than coding their own, it aids overall smart contract security.

All current smart contract security issues are tracked, see (*Overview · Smart Contract Weakness Classification and Test Cases*, n.d.). There is a greater focus on smart contract security over security

issues in other languages, and that is because smart contracts interact with finances. One slight loophole, could lead to very large sums of money lost. In reality, this robust attitude towards constantly improving security, could actually make smart contracts an excellent place to store sensitive information, especially in the coming years.

Even though the healthcare data itself would be encrypted on the blockchain, the transactions to and from the system are usually not encrypted. These transactions do not expose data, but they can potentially leave the system open to a front running attack (Ekanem, 2022). This means that malicious actors could see the pending transactions in the memepool, and cause some minor interference (Ekanem, 2022). However this can be resolved by encrypting the data, one such solution is currently being developed by 'zk-SNARKs, a zero-knowledge-proof<sup>2</sup> technique, to achieve encryption' (Ekanem, 2022).

### Social:

According to the (Deloitte, 2019) Global Blockchain Survey, 53% of senior executives surveyed said blockchain technology has become a critical priority for their organisation'. Furthermore, 86 percent of respondents believed that the technology would eventually become mainstream, with 83 percent of the sample pool admitting that their company has a compelling business case for blockchain application.

With major corporations such as IBM (IBM, 2019) and Microsoft (Azure, 2022) now offering Blockchain as a Service (BaaS) at a low cost the technology's applications are now within reach of many small to medium-sized businesses. While the former is currently working on a blockchain healthcare system, Microsoft is powering blockchain applications ranging from artificial intelligence to capital markets.

One of the greatest misunderstandings in the blockchain industry is that cryptocurrencies are largely utilised for illicit purposes. This is partially due to the fact that Bitcoin payments were used for the 'Silk Road'<sup>3</sup>. Digital assets, on the other hand, have shown to be useful for cross-border transactions, ease, and long-term wealth storage (Alluva, 2019). According to (McGovern, 2022) 'it is estimated that the average ownership rate of cryptocurrency is 3.9%, which means that there are more than 300 million people using cryptocurrency around the world'. This means that on average, in a room of 25 people, approximately one of them is a cryptocurrency user. Thus, it is clear from the high number of blockchain users, as well as the participation from major corporations, that there has been a shift in recent years in the perception of blockchain technology, towards a positive light.

To illustrate this shift in perception, below are some results of the blockchain survey carried out by (Deloitte, 2019):

---

<sup>2</sup> A cryptographic protocol that enables a party to demonstrate possession of information, such as a private key, without disclosing it. To convey information from a prover to a verifier, zero-knowledge proofs provide "proofs of validity." They are used to quickly verify transactions on a blockchain. (Definition of Zero-Knowledge Proof, 2022)

<sup>3</sup> Silk Road was a digital black market platform that was popular for hosting money laundering activities and illegal drug transactions using Bitcoin (Frankenfield, 2021).



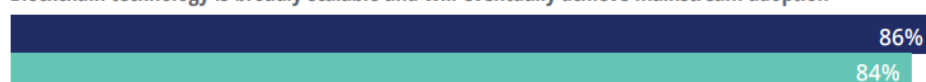
## Survey respondents' attitudes on blockchain and its adoption (2019 vs. 2018)

There was a general improvement in attitudes about blockchain over the past year

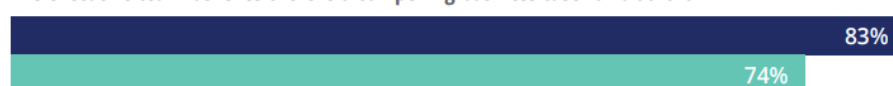
*Survey question: What is your level of agreement or disagreement with each of the following statements regarding blockchain technology?*

■ 2019 ■ 2018

Blockchain technology is broadly scalable and will eventually achieve mainstream adoption



The executive team believes there is a compelling business case for blockchain



Our suppliers, customers, and/or competitors are discussing or working on blockchain solutions to current challenges in the value chain that serves my organization



We are planning to replace current systems of record



We will lose a competitive advantage if we don't adopt blockchain technology



Blockchain will disrupt our industry



Blockchain is overhyped



Figure 9

N=1,386 (2019 global enterprise); N=1,053 (2018 global enterprise).

Note: Percentages indicate respondents who strongly or somewhat agree with each statement.



Source: Deloitte's Global Blockchain Survey, 2018 and 2019.

It is evident from the results of the above questionnaire that attitudes from businesses have improved significantly from 2018 to 2019. 83% of executive teams believe that there are compelling use cases for blockchain, up from 74% in 2018. This coupled with Bitcoin's significant price increase in 2020<sup>4</sup>, and Tesla's investments into Bitcoin at a high price of \$32,000 (Root, 2021), provide a lot of credibility to many people who are not able to appreciate the technology itself.

In countries with poor or struggling economies, such as Venezuela and Zimbabwe, digital currencies have proven to be an important resource. These nations' citizens have been suffering with the repercussions of hyperinflation for a long time and have begun to distrust their government-issued fiat currency. As a result, Bitcoin trade volume in Venezuela has been steadily increasing, which has been reported by (Meredith, 2019).

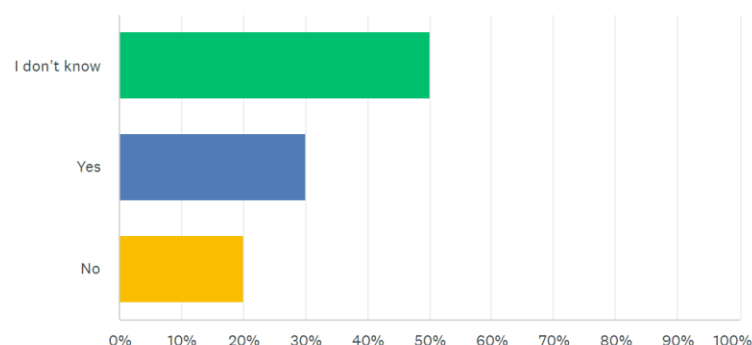
When the BBC interviewed numerous Venezuelans (Mathew Di Salvo, 2019), it was found that they had resorted to the cryptocurrency market for stability, an overwhelming percentage of locals stated that Bitcoin and other digital currencies helped them attain financial independence and security.

Q4

 Customize  Save as

Do you believe that Blockchain would be a more secure data structure to store data than your current local healthcare system?

Answered: 10 Skipped: 0



ANSWER CHOICES	RESPONSES
I don't know	50.00% 5
Yes	30.00% 3
No	20.00% 2
TOTAL	10

Figure 10

After conducting a survey to 10 healthcare professionals, 50% of them did not know that blockchains could hold data more securely than the traditional databases that are being used in the industry at the moment. However, there were more positive responses than negative responses, which shows a slightly overall positive view of blockchain. Having said that, one cannot deny that there were 2

<sup>4</sup> 'December 2020, Bitcoin's price had increased by over 300% since January. The year ended at a price of about \$29,374 — the highest it had ever been' (DeMatteo, 2021)

negative responses to this question, demonstrating that current views on blockchain still need to be addressed.

(See figures 18 – 21 in the appendix for the remainder of the survey).

### Legal:

'Censorship-resistance is considered to be one of the main value propositions of Bitcoin' (Binance, 2022). This is because it operates on millions of nodes across the world. China spent several months attempting to ban bitcoin in stages (QUIROZ-GUTIERREZ, 2022), and despite having the biggest hash rate in the world, today the bitcoin security remains unaffected by the temporary drop in hash rate that was caused by the ban. Despite the regulatory pressure from the Chinese government, it is very difficult to track cryptocurrency users and miners as they are simply using a computer to connect to the blockchain, which can be done covertly at home. Nevertheless, tough regulations would make buying cryptocurrency on exchanges more difficult.

Under GDPR, medical and healthcare data stored on a blockchain is considered "personal data" and "special category data" (Finck, 2018). The GDPR applies to "cryptographically changed data kept on a distributed ledger, in addition to public keys". This data is not considered anonymous in European Union data protection law. To put it another way, even if data is encrypted or hashed, it is still considered personal data under GDPR (GDPR Art 4, 2019). This is a problem, but it can be overcome if the patient receives explicit permission to upload his data to the Blockchain. By virtue of the authorization being provided, the data is no longer considered personal or sensitive, and it can be uploaded/encrypted to the Blockchain. Thus, there should be professional considerations of informing patients about their data, and the GDPR implications of consent, regarding their data.

Automated consent is a significant problem in healthcare, and with the public's lack of understanding of such a new technology, it is important they understand what they are consenting to, and that it is actually proven that they gave consent.

"In the instance of the DeepMind collaboration with Royal Free London NHS Foundation Trust, the lack of patient consent was flagged as one of the most significant issues, despite the positive effect Google's product suite had on patient diagnosis and treatment" (Hern, 2017)

Therefore, it is crucial that patients are educated about what they are consenting to, in order to avoid automated consent. If patients are not actively consenting for their data to be used on the blockchain, legal issues could arise due to blockchain data not being considered anonymous in regulation yet. Another option would be proposing for the GDPR to be revised, once regulators are convinced that storing data in such a manner is indeed secure.

Also, there is a legal issue of liability. If a smart contract is miscoded and fails to achieve the parties' goal, or if the oracle makes a mistake or wilful error, then the issue of responsibility for such faults must be addressed. The law must be specific with regards to penalties on hackers, as well as projects which fail to design secure systems, causing loss to public funds. Moreover, smart contracts are not recognised as a proof of agreement between parties in most countries, so this is another legal issue which will hopefully be resolved in the future.

### Ethical:

As mentioned earlier, blockchains are censorship resistant (QUIROZ-GUTIERREZ, 2022); this is due to their decentralisation. Many perceive this as positive due to the fact it encourages personal and

economic freedom. However, Decentralization of authority means that there is no single authority in the network that can enforce law and order. There are no moderators, leaders, or governing agency. This can often allow for scams, and unethical activity on the network, such as unfair loans and gambling. Creating ethical projects such as the subject of this treatise, will encourage blockchain developers to work on more ethical projects, rather than working on decentralised applications that have lost many users money.

Centralised system consumes less energy than Proof of Work (PoW) blockchain technology. Their redundancy not only causes them to consume more power than a typical centralised cloud-based system, but their transaction validation mechanism also plays a significant role. Hence why Tesla refused to continue their plan to accept Bitcoin payments (BBC, 2021).

In addition, they demand more storage than other systems. The number of nodes added to a blockchain multiplies the amount of electricity required. Each node stores and processes nearly as much data as any other system's central body. However, a viable solution would be to release decentralised applications onto Proof of Stake and other efficient networks. The energy that is required by a Proof of Stake node is equivalent to that of an average laptop (Conway, 2022).

## Implementation

The prototype has been successfully implemented with full functionality. See below for the successful run of the feature tests:

```
> truffle test
Using network 'development'.

Contract: Patients
  ✓ deploys successfully
  ✓ lists patients (312ms)
  ✓ creates patients (547ms)
  ✓ updates patients (416ms)

Contract: AppointmentList
  ✓ deploys successfully
  ✓ lists appointments (270ms)
  ✓ creates appointments (473ms)
  ✓ toggles task completion (521ms)

8 passing (3s)

/mnt/c/Users/ismai/OneDrive/Documents/fyp3
```

Figure 11 See figures 16 and 17 for the code to the tests

Healthcare System | Appointments and Patient Database

0x77f2edbd1b49495fd8223bde82336076a4cb8e9b

### Appointments

Add Appointment...

☐ Appointment with Ismail Kamran at 18:00

☒ Appointment with John Doe today at 16:15

☒ Appointment with Faris at 17:30 today

### Add New Patient

Name

Your name..

Date of Birth: dd/mm/yyyy

Medical Issues

Healthcare System | Appointments and Patient Database

0x77f2edbd1b49495fd8223bde82336076a4cb8e9b

### Add New Patient

Name

Your name..

Date of Birth: dd/mm/yyyy

Medical Issues

Medical Issues...

Submit

### Patient Records

Patient ID	Name	D.O.B	Medical Record
1	John Doe	1978-02-25	Pre diabetic with athritus
2	Ismail Kamran	1976-10-10	Fit as fiddle
3	Faris Razaq	1999-10-16	Fast Metabolism, W3 Schools addiction

Figure 12 and 13

When a task is completed, it is ‘toggleCompleted’ in the smart contract, which then causes the front end to cross it out. This increases immutability, and reduces the chance of data loss. Patient records are updated on refresh.

The system is fully functional. However, there were some design simplifications made. There are now only two smart contracts, a Patient contract and Appointment contract. The Patient contract contains a Patient struct, and the Appointment contract contains an Appointment struct. The Patient contract, and Appointment contract are linked in the controller but not the contracts are not linked. This is because appointment records are usually stored separately to medical records, so there was

no need to link them in this prototype. The code is now more simply written compared to the class diagram (figure 5) whilst maintaining all functionality, and this is in accordance with the K.I.S.S design principle.

(Interaction Design Foundation, 2022) states “Keep it simple, stupid (KISS) is a design principle which states that designs and/or systems should be as simple as possible. Wherever possible, complexity should be avoided in a system—as simplicity guarantees the greatest levels of user acceptance and interaction”. Hence, it would be appropriate to reduce complexity when possible.

All other diagrams apart from figure 5 have been implemented accurately. See the code for the smart contracts below:

```
contract Patients {
    uint public patientCount = 0;

    struct Patient {
        uint id;
        string name;
        string dob;
        string medicalRecord;
    }

    mapping(uint => Patient) public patients;

    event PatientCreated(
        uint id,
        string name,
        string dob,
        string medicalRecord
    );

    event PatientUpdated(
        uint id,
        string name,
        string dob,
        string medicalRecord
    );

    constructor() public {
        createPatient("John Doe", "1978-02-25", "Pre diabetic with athritus");
    }

    function createPatient(string memory name, string memory dob, string memory _medicalRecord) public {
        patientCount++;
        patients[patientCount] = Patient(patientCount, name, dob, _medicalRecord);
        emit PatientCreated(patientCount, name, dob, _medicalRecord);
    }

    function updatePatient(uint id, string memory name, string memory dob, string memory _medicalRecord) public {
        emit PatientUpdated(id, name, dob, _medicalRecord);
        patients[id] = Patient(patientCount, name, dob, _medicalRecord);
    }
}
```

Figure 14 Patient smart contract

```

pragma solidity ^0.5.0;

contract AppointmentList {
    uint public taskCount = 0;

    struct Task {
        uint id;
        string content;
        bool completed;
    }

    mapping(uint => Task) public tasks;

    event TaskCreated(
        uint id,
        string content,
        bool completed
    );

    event TaskCompleted(
        uint id,
        bool completed
    );

    constructor() public {
        createTask("Appointment with John Doe today at 16:15");
    }

    function createTask(string memory _content) public {
        taskCount ++;
        tasks[taskCount] = Task(taskCount, _content, false);
        emit TaskCreated(taskCount, _content, false);
    }

    function toggleCompleted(uint _id) public {
        Task memory _task = tasks[_id];
        _task.completed = !_task.completed;
        tasks[_id] = _task;
        emit TaskCompleted(_id, _task.completed);
    }
}

```

Figure 15 Appointment Smart Contract

Note that they have been called tasks in the code, as healthcare staff could use the system to set other tasks in their schedule, not just appointments.

Below is the code for the tests:

```
it('deploys successfully', async () => {
  const address = await this.apptList.address
  assert.notEqual(address, 0x0)
  assert.notEqual(address, '')
  assert.notEqual(address, null)
  assert.notEqual(address, undefined)
})

it('lists appointments', async () => {
  const taskCount = await this.apptList.taskCount()
  const task = await this.apptList.tasks(taskCount)
  assert.equal(task.id.toNumber(), taskCount.toNumber())
  assert.equal(task.content, 'Appointment with John Doe today at 16:15')
  assert.equal(task.completed, false)
  assert.equal(taskCount.toNumber(), 1)
})

it('creates appointments', async () => {
  const result = await this.apptList.createTask('A new task')
  const taskCount = await this.apptList.taskCount()
  assert.equal(taskCount, 2)
  const event = result.logs[0].args
  assert.equal(event.id.toNumber(), 2)
  assert.equal(event.content, 'A new task')
  assert.equal(event.completed, false)
})

it('toggles task completion', async () => {
  const result = await this.apptList.toggleCompleted(1)
  const task = await this.apptList.tasks(1)
  assert.equal(task.completed, true)
  const event = result.logs[0].args
  assert.equal(event.id.toNumber(), 1)
  assert.equal(event.completed, true)
})
```

Figure 16 Appointment tests



```

contract('Patients', (accounts) => {
  before(async () => {
    this.patientList = await Patients.deployed()
  })

  it('deploys successfully', async () => {
    const address = await this.patientList.address
    assert.notEqual(address, 0x0)
    assert.notEqual(address, '')
    assert.notEqual(address, null)
    assert.notEqual(address, undefined)
  })

  it('lists patients', async () => {
    const patientCount = await this.patientList.patientCount()
    const patient = await this.patientList.patients(patientCount)
    assert.equal(patient.id.toNumber(), patientCount.toNumber())
    assert.equal(patient.name, 'John Doe')
    assert.equal(patientCount.toNumber(), 1)
  })

  it('creates patients', async () => {
    const result = await this.patientList.createPatient("James Rider", "1978-02-27", "Full diabetic with athritus")
    const patientCount = await this.patientList.patientCount()
    assert.equal(patientCount, 2)
    const event = result.logs[0].args
    assert.equal(event.id.toNumber(), 2)
    assert.equal(event.name, 'James Rider')
  })

  it('updates patients', async () => {
    const result = await this.patientList.updatePatient(2, "Alex Rider", "1978-02-27", "Full diabetic with athritus")

    const event = result.logs[0].args
    assert.equal(event.id.toNumber(), 2)
    assert.equal(event.name, 'Alex Rider')
  })
})

```

Figure 17 Patient tests

## Evaluation

The objective of this treatise and prototype, was to propose blockchain solution to the healthcare interoperability problem, as well as safeguarding patient data. Due the successful creation of said blockchain system, it is correct to say that the overall objectives have been met.

A prototype with almost complete CRUD operations has been created for both patient data and appointments. There is however, one CRUD operation missing which is the deletion of patient data. In order to develop this operation, it would be necessary to redesign the way that data is kept within the Patient contract. One cannot simply delete data from a mapping, rather it is incumbent to maintain a record of which data is active within an array. Then delete the item in the array, and shift all elements left. When creating the full software this should be considered, however for the prototype this is not a problem. As (Lunn, 2003) mentions 'prototypes should be thrown away once they have served their purpose', thus a redesigning for the final version would be necessary.

Within the code, there are events for each function. These events can be tracked indefinitely on the blockchain, so if there any data issues or concerns, then the events are there to track data changes. See figure 16 and 17 where events are used to test the application is working correctly.

## Conclusion

The subject of blockchain is very rich with information, but the fundamentals of the entire technology are all built upon Computer Science concepts. Everything from the programming language, to the EVM, even the mathematical operations on the hardware level – the basics of them – have been taught across my modules throughout my degree.

Finally, key sections requested during feedback were the acceptability of a blockchain solution to healthcare system users. Blockchain security, and demonstration of blockchain success (i.e. Bitcoin and Ethereum). As well as completing the objectives of this treatise, I hope the reader finds all topics covered in a well manner.

## Appendix

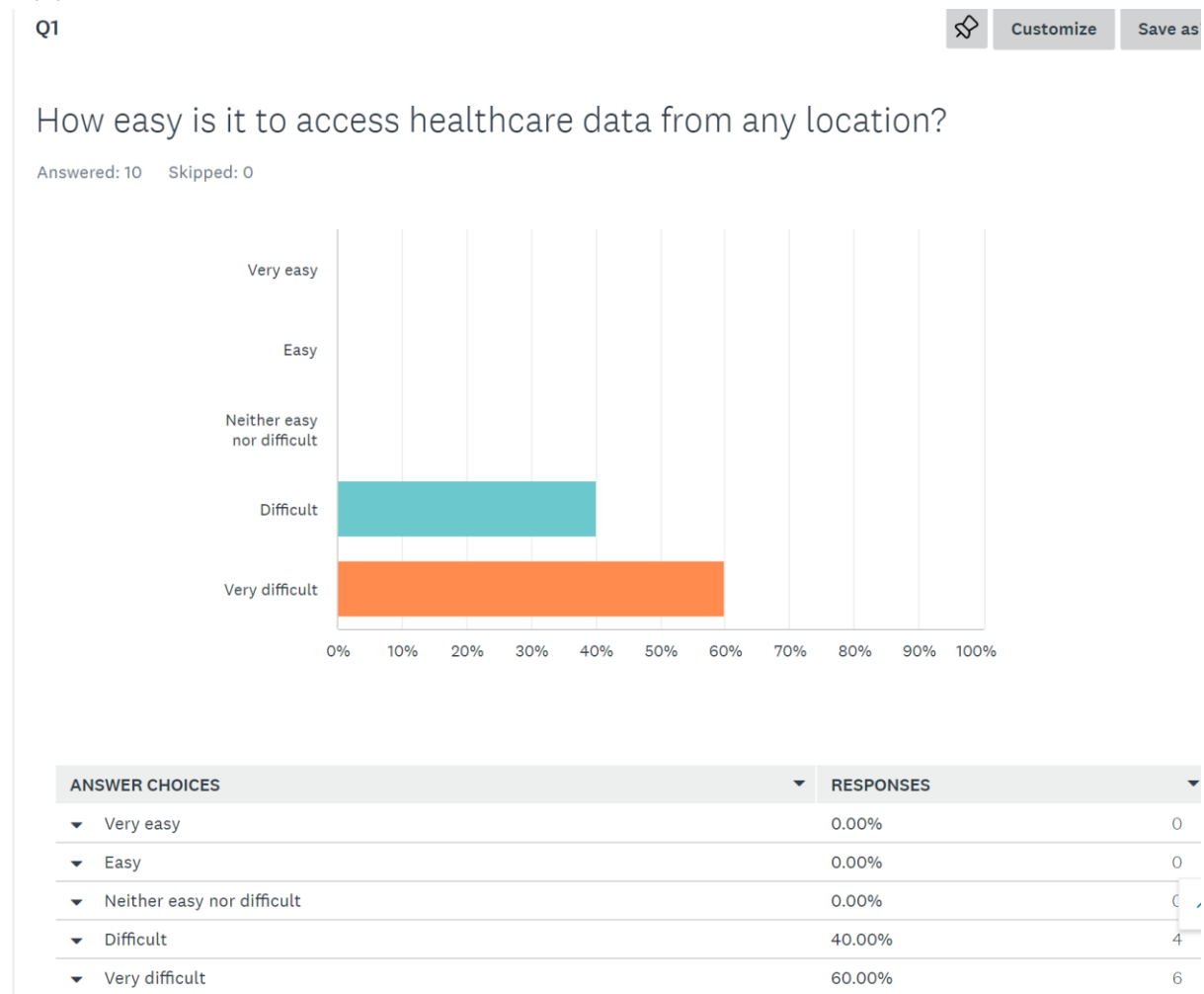
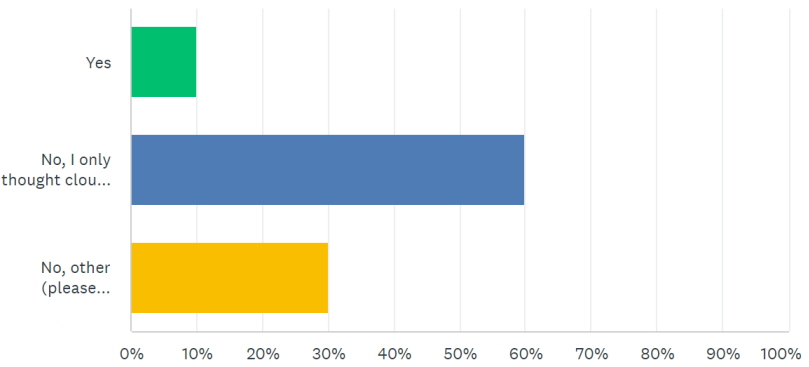


Figure 18

Are you aware that Blockchain can offer a valid solution to the healthcare interoperability problem?

Answered: 10    Skipped: 0

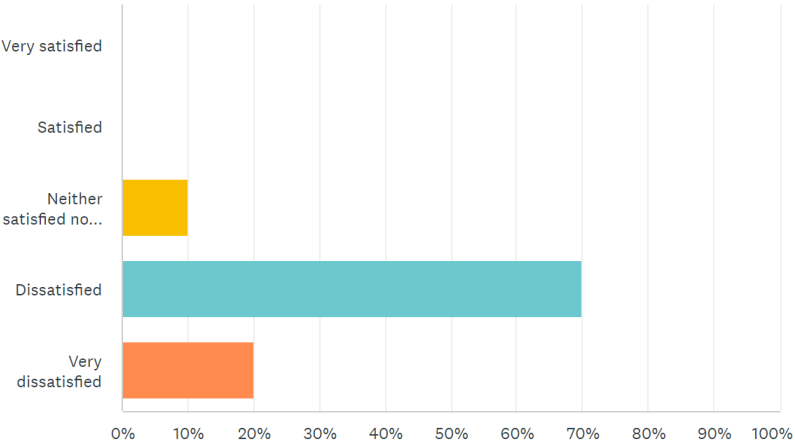


ANSWER CHOICES	RESPONSES	
Yes	10.00%	1
No, I only thought cloud could achieve this	60.00%	6
No, other (please specify)	30.00%	3
TOTAL		10

Figure 19

# How satisfied are you with your current healthcare management system

Answered: 10    Skipped: 0



ANSWER CHOICES	RESPONSES	
Very satisfied	0.00%	0
Satisfied	0.00%	0
Neither satisfied nor dissatisfied	10.00%	1
Dissatisfied	70.00%	7
Very dissatisfied	20.00%	2
TOTAL		10

Figure 20

Q5

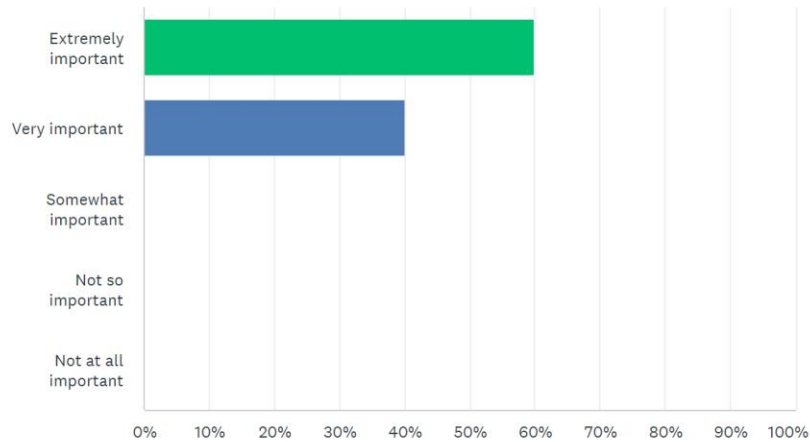


Customize

Save as

## How important is data interoperability in healthcare?

Answered: 10 Skipped: 0



ANSWER CHOICES	RESPONSES	
Extremely important	60.00%	6
Very important	40.00%	4
Somewhat important	0.00%	0
Not so important	0.00%	0
Not at all important	0.00%	0
<b>TOTAL</b>		<b>10</b>

Figure 21

### STATEMENT BY THE STUDENT

**I believe that the information I have given in this form on ethical issues is correct.**

Signature: Ismail Kamran Date: 17/05/2022

### AFFIRMATION BY THE SUPERVISOR

**I have read this Ethical Review Checklist and I can confirm that, to the best of my understanding, the information presented by the student is correct and appropriate to allow an informed judgement on whether further ethical approval is required.**



Signature: G. ALLEN Date: 17/05/2022

Figure 22 – Passed risk assessment

ACTIVITY:			Name:
LOCATION:			Date:
Hazard(s) Identified	Details of Risk(s)	People at Risk	Risk management measures
Security issues/ potential security breaches	There have been many smart contract security breaches, some of which have costed projects hundreds of millions of dollars	Patients and healthcare staff	Smart contract security audits, and using well established coding practices (i.e. <u>OpenZeppelin</u> contracts)
Lack of patient consent to use data	There has been some negative perception of Blockchain in the past, partially due to security issues, as well as illicit use	Stakeholders	Educate patients on the benefits of this technology for their data
Interviewing healthcare staff may lead to a breach in patient confidentiality	Healthcare staff are required not to reveal patient information to non-healthcare staff	Patients	Ask general questions relating to healthcare and blockchain, and do not ask about individual patient experiences. Also remind staff, to only reveal unidentifiable information on patients <u>where</u> beneficial.

Figure 22 - Project risk assessment

## Bibliography

Eze, P., Eziokwu, T., & Okpara, C. (2017). A Triplicate Smart Contract Model using Blockchain Technology. *Circulation in Computer Science*, DC CPS 2017(01), 1–10. <https://doi.org/10.22632/ccs-2017-cps-01>

Lunn, K. (2003). *Software development with UML*. Palgrave Macmillan.

erwinkarim. (2021, December 30). Web3 Explained. Techjourneyman.com.  
<https://techjourneyman.com/blog/web3-explained/>

The General Medical Council. (2022). The professional duty of candour. Wwww.gmc-Uk.org.  
<https://www.gmc-uk.org/ethical-guidance/ethical-guidance-for-doctors/candour---openness-and-honesty-when-things-go-wrong/the-professional-duty-of-candour#:~:text=Every%20health%20and%20care%20professional>

Porras, E. R., & Daugherty, B. (2021). Bitcoin and Ethics in a Technological Society. In www.intechopen.com. IntechOpen. <https://www.intechopen.com/chapters/75905>

Browne, R. (2021, August 23). Hacker behind \$600 million crypto heist returns final slice of stolen funds. CNBC. <https://www.cnn.com/2021/08/23/poly-network-hacker-returns-remaining-cryptocurrency.html>

Capitol Technology University. (2020, October 19). Smart Contract Hacking: What is it and What Does it Affect? Wwww.captechu.edu. <https://www.captechu.edu/blog/smart-contract-hacking-what-it-and-what-does-it-affect>

Greenspun, P. (2000). Redefining Professionalism for Software Engineers. Philip.greenspun.com. <https://philip.greenspun.com/ancient-history/professionalism-for-software-engineers>

Solidity — Solidity 0.8.13 documentation. (2021). Docs.soliditylang.org.  
<https://docs.soliditylang.org/en/v0.8.13/>

Definition of zero-knowledge proof. (2022). PCMAG.  
<https://www.pcmag.com/encyclopedia/term/zero-knowledge-proof>

IBM. (2019). IBM Blockchain - Enterprise Blockchain Solutions & Services. Ibm.com.  
<https://www.ibm.com/blockchain>

Azure, M. (2022). Azure Blockchain Workbench | Microsoft Azure. Azure.microsoft.com.  
<https://azure.microsoft.com/en-gb/features/blockchain-workbench/>

McGovern, T. (2022, January 31). Cryptocurrency Statistics 2022: How Many People Use Crypto? CRYPTOCURRENCY STATISTICS 2022. <https://earthweb.com/cryptocurrency-statistics/#:~:text=of%202021%20million->

Ekanem, D. (2022, May 13). A developer's guide to smart contract security audits. LogRocket Blog. <https://blog.logrocket.com/developers-guide-smart-contract-security-audits/>

Meredith, S. (2019, February 14). Bitcoin trading in crisis-stricken Venezuela has just hit an all-time high. CNBC. <https://www.cnn.com/2019/02/14/venezuela-crisis-bitcoin-trading-volumes-hit-an-all-time-high-.html>

Mathew Di Salvo. (2019, March 19). Why are Venezuelans seeking refuge in crypto-currencies? BBC News. <https://www.bbc.co.uk/news/business-47553048>

State of the DApps. (2019). State of the DApps — DApp Statistics. Stateofthedapps.com.  
<https://www.stateofthedapps.com/stats>

Binance. (2022). Censorship-resistance. Binance Academy.  
<https://academy.binance.com/en/glossary/censorship-resistance>

Finck, M. (2018). Blockchains and Data Protection in the European Union. European Data Protection Law Review, 4(1), 17–35. <https://doi.org/10.21552/edpl/2018/1/6>

QUIROZ-GUTIERREZ, M. (2022, January 4). Crypto is fully banned in China and 8 other countries. Fortune. <https://fortune.com/2022/01/04/crypto-banned-china-other-countries/#:~:text=When%20it%20banned%20crypto%20last%20year%2C%20China%20did%20so%20in>

Hern, A. (2017, July 3). Royal Free breached UK data law in 1.6m patient deal with Google's DeepMind. The Guardian; The Guardian.  
<https://www.theguardian.com/technology/2017/jul/03/google-deepmind-16m-patient-royal-free-deal-data-protection-act>

BBC. (2021, May 13). Tesla will no longer accept Bitcoin over climate concerns, says Musk. BBC News. <https://www.bbc.co.uk/news/business-57096305>

Conway, L. (2022, February 18). Proof-of-Work vs. Proof-of-Stake: Which Is Better? Blockworks.  
<https://blockworks.co/proof-of-work-vs-proof-of-stake-whats-the-difference/#:~:text=stake%20pros%2C%20explained->

Sharma, M. (2018, September 24). Web 1.0, Web 2.0 and Web 3.0 with their difference - GeeksforGeeks. GeeksforGeeks. <https://www.geeksforgeeks.org/web-1-0-web-2-0-and-web-3-0-with-their-difference/>

ajammin, S. (2022, April 12). Web2 vs Web3. Ethereum.org.  
<https://ethereum.org/en/developers/docs/web2-vs-web3/>

The ENS Team. (2022, April 12). Ethereum Virtual Machine (EVM). Ethereum.org.  
<https://ethereum.org/en/developers/docs/evm/>

Truffle Suite. (2022). Ganache | Overview - Truffle Suite. Trufflesuite.com.  
<https://trufflesuite.com/docs/ganache/>

Ethereum Foundation. (2022). Contracts — Solidity 0.8.13 documentation. Docs.soliditylang.org.  
<https://docs.soliditylang.org/en/v0.8.13/contracts.html#interfaces>

Interaction Design Foundation. (2022). What is Keep It Simple, Stupid (Kiss)? The Interaction Design Foundation. [https://www.interaction-design.org/literature/topics/keep-it-simple-stupid/#:~:text=Keep%20it%20simple%2C%20stupid%20\(KISS\)%20is%20a%20design%20principle](https://www.interaction-design.org/literature/topics/keep-it-simple-stupid/#:~:text=Keep%20it%20simple%2C%20stupid%20(KISS)%20is%20a%20design%20principle)

Capitalgram. (2021, July 12). Scaling EVM (Ethereum Virtual Machine). Capitalgram.  
<https://capitalgram.com/posts/scaling-evm/#:~:text=Today%20EVM%20powers%20~90%25%20of>



Satoshi Nakamoto. (2021). The Bitcoin Whitepaper. <https://bitcoin.org/en/bitcoin-paper>

Bitcoin Worldwide. (2021). How Many Bitcoins Are There? (Circulating Supply - Live).  
Www.buybitcoinworldwide.com. <https://www.buybitcoinworldwide.com/how-many-bitcoins-are-there/#:~:text=How%20Many%20Bitcoin%20Miners%20Are>

Ychart. (2021). Bitcoin Network Hash Rate. Ycharts.com.  
[https://ycharts.com/indicators/bitcoin\\_network\\_hash\\_rate](https://ycharts.com/indicators/bitcoin_network_hash_rate)

Sigalos, M. (2021, December 10). Bitcoin mining has totally recovered from Chinese ban. CNBC.  
<https://www.cnbc.com/2021/12/10/bitcoin-network-hashrate-hits-all-time-high-after-china-crypto-ban.html>

Bitcoin Worldwide. (2021). How Many Bitcoins Are There? (Circulating Supply - Live).  
Www.buybitcoinworldwide.com. <https://www.buybitcoinworldwide.com/how-many-bitcoins-are-there/#:~:text=How%20Many%20Bitcoin%20Miners%20Are>

Satoshi told everyone his fork was Bitcoin and Vitalik told everyone his fork wa... | Hacker News.  
(2016). News.ycombinator.com. <https://news.ycombinator.com/item?id=16588708>

(Vitalik Buterin, 2013). <https://ethereum.org/en/whitepaper/>.  
<https://ethereum.org/en/whitepaper/>

(Ethereum Gas Tracker, 2013). Gas Tracker [www.etherscan.io/gastracker](http://www.etherscan.io/gastracker)

(Bloomberg, 2021). Solana Promises 'Detailed Post-Mortem' After 17-Hour Outage.  
<https://www.bloomberg.com/news/articles/2021-09-16/solana-network-of-top-10-sol-token-applies-fixes-after-outage>