TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies

Henry Ifeanyi Anyanka 145789951VGM

Systematic Analysis of Methods/Frameworks for Developing Decentralized Applications on Blockchain

Master's Thesis

Supervisor: Alexander H. Norta
Ph.D. Associate
Professor

Co-supervisor: Chibuzor Udokwu

Tallinn 2020

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Henry Ifeanyi Anyanka 145789951VGM

Detsentraliseeritud Plokiahela Rakenduste Meetodite/Raamistike Süstemaatiline Analüüs

Magistritöö

Juhendaja: Alexander H. Norta
Ph.D. Associate
Professor

Kaasjuhendaja: Chibuzor Udokwu

Tallinn 2020

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Henry Ifeanyi Anyanka

15.05.2020

Abstract

Blockchain technology is novel and its exponentially increasing trend has the capacity to transform multiple sectors. Due to the introduction of decentralized application on the blockchain-based system, organizations and government sectors are moving due to security, transparency, privacy and interoperability function on a blockchain-based system. Decentralized application introduced the integration of several businesses on the blockchain which has enhanced trust and collaboration between business to business and government to government. Most decentralized applications built on the blockchain poses issues as method for development are challenging. Collaboration of businesses between stakeholders' processes are complex due to the use of different approaches and designs. Most tools and software do not capture modern functionalities between object because they are old and outdated. This paper aim to address techniques for developing a decentralized application on the blockchain. This research will explore existing design methods and frameworks for developing decentralized application on the blockchain, strength and weaknesses of existing methods/frameworks, scalability and trade-off. To achieve our aim, a case study research methodology was selected as the suitable research methodology for conducting this research while document review and interview were key instruments for data collection. The result attests to the fact that engineers and developers were not satisfied with the current approaches, designs and methods for modelling object-oriented system and collaboration among stakeholders during modelling and design processes. With respect to the study, the findings support a strategic collaboration in addressing modelling approaches and stakeholder partnership to attain a standardized framework/method for development of DApp.

Keywords: decentralized, blockchain, DApps, design, modelling, framework, technique, scalability, trade-off, security.

This thesis is written in English and is sixty-two (62) pages long, five (5) chapters, seven (7) figures and seven (7) tables.

Annotatsioon

Detsentraliseeritud plokiahela rakenduste meetodite/raamistike süstemaatiline analüüs.

[Tekst]

Lõputöö on kirjutatud [mis keeles] keeles ning sisaldab teksti [lehekülgede arv] leheküljel, [peatükkide arv] peatükki, [jooniste arv] joonist, [tabelite arv] tabelit.

List of abbreviations and terms

API	Application Programming Interface
BOS	Blockchain Oriented Software
BPMN	Business Process Model and Notation
C4	Context, Container, Components and Code
CAS	Complex Adaptive System
DHT	Distributed Hash Table
DLT	Distributed Ledger Table
DApps	Decentralised Applications
EHR	Electronic Health Record
IoT	Internet of Things
SC	Smart Contract
P2P	Peer-2-Peer
PoA	Proof-of-Authority
PoS	Proof-of-Stake
PoW	Proof-of-Work
TUT	Tallinn University of Technology
UML	Unified Mark-up Language
US	User Story

Table of Contents

1 Introduction 11

1.1 Problem Statement.. 13

1.2 Research Objectives 14

2 Related Work.. 15

2.1 Earlier Studies... 15

2.1.1 Underlying Peer-2-Peer and Blockchain Concept in a Distributed System. . 16

2.1.2 Technique and Methods for Developing a P2P Application on Distributed Systems. 18

2.1.3 Anthill: A Framework for the Design and Analysis of Peer-to-Peer. (Babaoglu, Meling and Montresor, 2002) 19

2.1.4 Integrating Multiple Perspectives in System Development. Viewpoint Framework (FINKELSTEIN et al., 1992). 22

2.2 Blockchain Technology 23

2.2.1 Components of Blockchain Technology 25

2.2.2 Key Attributes of the Blockchain Technology 30

2.3 Theoretical Frameworks 31

2.3.1 Privacy-Preserving Techniques (Ancile) By (Dagher et al., 2018)............... 31

2.3.2 Traceability Ontology-Driven Design By (Kim and Laskowski, 2018). 33

2.3.3 Strategy for Developing a DApps (Marchesi, Marchesi and Tonelli, 2019). 36

2.3.4 Summary.. 40

3 Research Methodology 41

3.1 Introduction 41

3.2 Case Study Design and Selection 41

3.3 Data Collection Procedure.. 43

3.4 Analysis of Procedure.. 45

3.5 Validity Procedure.. 47

3.6 Summary.. 47

4 Results 49

4.1 Introduction 49

4.2 Case Description.. 49

4.3 Subject Description... 50

4.4 Presentation of Findings 51

4.4.1 Current Practices in Development of Decentralised Applications on a Blockchain. 51

4.4.2 Existing Designs and Modelling Techniques in Software Development. 54

4.4.3 Measuring Correctness and Security Assessments. 57

4.4.4 Concerns on Scalability and Trade-Offs in DApps. 60

4.4.5 Summary... 62

5 Conclusion and Future Work... 63

5.1 Introduction 63

5.2 Summary of Findings and Recommendations.. 63

5.2.1 Standard Application and Integration... 64

5.2.2 Development Evaluation. 65

5.2.3 Security Assessment 65

5.2.4 Stakeholders Collaborative Approach. 66

5.2.5 Trade-Offs Implementation. 66

5.2.6 Scalability Design. 67

5.3 Impact/Implication... 67

5.4 Limitations.. 68

5.5 Future Work... 68

References 70

Appendix 1 – Interview Questions 77

Appendix 2 Thematic Map of all Categories and Codes 79

Appendix 3 Interview Recordings 80

Appendix 4 Interview Transcript.. 81

List of Figures

Figure 1. P2P network. Source: (Koteska, Karafiloski and Mishev, 2017) 19

Figure 2. Architect of a nest. source: (Babaoglu, Meling and Montresor, 2002)........... 20

Figure 3. Blockchain Overview. Source: (Ismail and Materwala, 2019)....................... 24

Figure 4: Layers of the blockchains source: (Lu, 2019).. 25

Figure 5. TOVE Traceability Ontology data (Kim and Laskowski, 2018).................... 34

Figure 6. User interaction with the blockchain (Kim and Laskowski, 2018)................ 35

Figure 7: A proposed UML activity diagram. Source: (Marchesi, Marchesi and Tonelli, 2019).. 38

List of Tables

Table 1. 1: Permissioned and Permission-less. Source 28
Table 1. 2 Proposed steps to developing a decentralized application on a blockchain .. 39
Table 1. 3: Agile framework strength and weakness in development of DApps 53
Table 1. 4: Strength and weakness of frequently utilised mark-up and notations.......... 57
Table 1. 5: Various testing on the decentralized application... 58
Table 1. 6: Different security measures in development of DApps. 60
Table 1. 7: Issues Of scalability in the DApps 61

1 Introduction

There is overwhelming progress in the expansion of cryptographic technology playing a major role as a catalyst in moving technology, business and governance forward in the nearest the future. Most companies are gradually moving from the era of a centralized unit to embrace decentralization. Due to the growing interest of companies as well as small grooming financial institutions, the renew technology branded blockchain has taken into several shapes not as it was first built by Satoshi Nakamoto in 2008 but rather to suit different needs (Nakamoto, 2008). According to Tschorsch and Scheuermann, (2016) "The blockchain is a peer-to-peer distributed ledger in which records called blocks are linked and secure using cryptographic hash". By nature, blockchains are decentralized, secured, immutable and tolerant which makes them appropriate for record management accomplishments such as financial dealings, data management, provenance and identity verification.

Blockchain commenced with the digital currency called Bitcoin which has been popular from its deployment in the financial market. Over time, the market capitalization of bitcoin from the last 5 years to date has been on a constant increase with a value of over $151 billion at present state (Chez, 2018). There is no doubt, blockchain technology has aided in the movement of funds from different part of the world in a secure manner with lesser cost than average. A unique feature of the blockchain has been the ability to convey digital assets (i.e. warranties, digital currencies, smart contracts, certifications, copyrights, licenses etc.) that contributes effectively to transparency, being that it is decentralized in nature with information distribution (Litke, Anagnostopoulos and Varvarigou, 2019). Blockchain could be seen as a joint ledger which is disseminated through connecting systems and servers, known as a node. Transactions and recordings of assets are stored in the ledger that makes tracking the source possible, with only one entity in its possession (Vitso et al., 2017). Due to immutability and auditability features the blockchain possesses, a validated transaction on the network that cannot be tampered nor changed which makes the platform consistent and trustworthy (Litke, Anagnostopoulos and Varvarigou, 2019).

Besides, most governmental organizations and financial institution have been victims of cybercrime in the past year losing a huge amount of fund due to laxity in the areas of security through attackers and hackers online. In 2018, Nigeria lost 15billion Naira ($41M approx.) in fraud and cyber-crimes with 37,817 fraud cases ('Banks lost N15billion to fraud, cyber-crime in 2018') other cases of attacks on multinational companies such as Capital One, First American Financial Corp., or PayID in 2019 or issues of attacks on e-government website with infiltration of SQL injections, cross-scripting (XSS) etc. to extract and manipulate data of citizens (Bui, 2016; Cameron Camp, 2016; Elisa et al., 2018). It is paramount the safety, privacy, reliability, accessibility and confidentiality of public details and records. One of the major causes of attacks and hacks has been the issues relating to the weakness of centralized management validation system which has made this a point of prominent attacks from foreign bodies (Elisa et al., 2018). According to Xu, (2016), Wang, Wu & He, (2019) stated on the bases of blockchain acclaimed adoption could appropriately mitigate most threats to fraud and malicious practices in the various sphere.

Due to several concerns and uprising in the single centralized management validation system, numerous companies are considering the option of migrating or building a decentralized application (DApps) on the blockchain, but most lacking a critical in-depth of this renewed technology and utilization (Ølnes, Ubacht and Janssen, 2017). Decentralized applications (DApps), are applications built on blockchain using smart contracts. Blockchain has been acknowledged with extensive significance and its realization a key to resolving difficulties encountered in countless institutions. It has found novel solutions to e-health care, e-governance agencies, sciences but more operational in finance and other aspects (Ølnes, Ubacht and Janssen, 2017).

The focus of this master thesis is to identify methods for developing a blockchain application. The research starts by evaluating different frameworks, methods and concept on which decentralized application (DApps) are built. Strategic approaches, technicality, steps, strength and weakness of development are outlined through evaluating and gathering of qualitative data, exploiting survey interviews and doing a thematic exploration through an open-source qualitative research software (RQDA)[1].

Above is a brief introduction of the main idea of this research, this chapter furthers to the problems this thesis focuses to resolve in detail. The next Section 1.2 will give a descriptive analysis of the research objectives.

1.1 Problem Statement

The adoption of a decentralized blockchain network by various enterprises and institution has been on persistent increase internationally with developed and developing nations embracing innovative methodologies to make development unified and translucent (Saveen A. Abeyratne, 2016). According to experts, decentralized applications have given way to more refined, advanced and hi-tech systems like digital signatures, smart contracts, X- road, etc., in so many countries (Martinovic, 2018).

Blockchain technology is considered in particular as a supporting platform for new services, products and business models with decentralized functionality (Lauslahti et al. 2016). So, expectations are increasing that blockchain might be a revolutionary technology aiming for use cases to potential advancement (Glaser 2017; Risius and Spohrer 2017). Properties and elements of upscaling centralized system have not been considered as the antidote to solving major issues such as security, privacy, transparency etc, due to its major vulnerability of one point, crippling an entire system nor like decentralized systems that are not prone to such threats (Saveen A. Abeyratne, 2016). Studies have shown that institutions, businesses, and government have gradually commenced their operations to a decentralized network, and such operations are mainly carried out on consortium and private blockchain networks (Jun, 2018; Casino, Dasaklis and Patsakis, 2019). Processes and transactions operating on a decentralized blockchain network are a bit expensive due to the related transaction fees (Buterin, 2018; Casino, Dasaklis and Patsakis, 2019). Permissioned blockchain networks that do not have fees are limited to challenges which include blockchain governance, restricted participants

---

[1] http://rqda.r-forge.r-project.org/

owing to the applicable consensus protocols (Atzori, 2017; Kormiltsyn, Thangalimodzi and Norta, 2018).

With respect to the above-mentioned, it is understandable that there is a need for an innovative methodology to address issues of availability, convenience and scalability implementation of DApp. Professionals from different works of life, in particular, are unsure about the effects blockchain could have on their organizations and are searching for a systematic approach to implement blockchain applications (Iansiti and Lakhani 2017). Based on a structured, systematic peer literature review and series of interviews, we plan to explore methods/framework for blockchain development. This thesis aims to provide a detailed understanding of software development methods/framework for developing DApps applications. We also try to outline challenges acknowledged in the relevant literature, especially limitations regarding the blockchain technology and possibility of proposing solutions.

1.2 Research Objectives

Although, there are other methods of developing a decentralized application, but our core emphasis is centered on blockchain technology. The general focus of this paper is to observe existing design methods and frameworks for developing applications on a decentralized and distributed network. Strength and challenges of existing methods/frameworks will be evaluated to determine if they are suitable for designing a scalable blockchain system.

The objectives of this research include:

1. Exploring current and existing software development methods/techniques for developing blockchain apps.
2. Examining different frameworks/methods, strength, weakness in a decentralized blockchain network.
3. Present current challenges confronting security, trade-offs or scalability.
4. How to improve software development methods to enable the development of DApps.

2 Related Work

For an extensive research to be conducted, it is of paramount to have a proper knowledge of past existing analysis and cases on the subject. The sole aim of this chapter is to initiate and present prior research which has led to this current master's research paper and likewise to scrutinize several applicable views and body of works connecting to the research objectives.

This section of the research paper is divided into three segments, the first part is concentrated on summarizing the definition, concepts and underlying technology of the blockchain, distributed system and P2P frameworks before blockchain. The second part focuses on the properties, component and attributes of the blockchain technology. The third section concretes on the theoretical frameworks, delving into the understanding of various existing methods of blockchain development with findings and research on different scholastic and scientific peer-reviewed literature in connection with the present study.

Additionally, during the exploration of this theoretical framework, we elaborate more in-depth on understanding of several existing methodologies for the development of blockchain technology while trying to analyse strengths and weakness to see towards a novel solution.

2.1 Earlier Studies

The application of blockchain technology to numerous business processes and start-up companies have adopted and established various methods of software development, which has been shallowly explored by researchers and therefore presents an ample opportunity for further research and studies. Examples of such methods could be found in these books (Langer, 2016; Dooley, 2017). There are blockchain use cases that were explored by researchers and scholars and present phenomenal guidance for various proposed businesses and transitional ideas in an organization. Prior studies include the use of blockchain as an alternative to traditional currencies and centralized banking schemes to cut exploitation in the finance sector (Nicholson, J., 2017). Kuo, T. Kim, H and Ohno-Machado, L., (2017) proffered the usage of blockchain-based technology for the managing and securing of hospital records, audit track record and protection of data due to its decentralized configuration.

In supply chain management according to Toyoda, K., Mathiopoulos, P.T., Sasase, I., and Ohtsuki (2017) utilizing the proof of work and validity of documents with the blockchain technology will enhance a decline in forgery, add accuracy in procedures, lessen time consumption and more efficiency. The utilisation of blockchain in the government sector will aid in the implementation of personal identification management, notary and validation of documents (Ølnes, S., Jansen, A., 2017). The blockchain technology which from the start was created for financial purposes has taken a shift in closing gaps coherent in information systems and business processes in several organizations. The digital currency bitcoin gave a clear consent about blockchain technology and chances of exploitation in other sectors from introduction into the commercial sector.

To completely understand the concept of the blockchain, we first need to get a proper understanding of other surrounding technologies that make up the core feature of the blockchain technology. An exploration of the concept of the peer-2-peer in a blockchain and methods for developing P2P applications in a distributed system.

2.1.1 Underlying Peer-2-Peer and Blockchain Concept in a Distributed System.

In an attempt to comprehend the concept of the Blockchain, it is of necessity to understand the coherent technologies that graft the core feature of the Blockchain technology.

As described earlier, "Blockchain" goes beyond the technology, by linking data blocks into an immutable chain. It operates as a fully distributed and decentralized network which allows all partaking peers to comply to clear rules in the blockchain to achieve data interoperability (Cai et al., 2018). Distributed simply implies that computation is dispersed across several nodes, rather than just one. Decentralized signifies all nodes are independent of themselves. A company, like Google, internally implemented a distributed architecture to improve computation and data latency (Raval, 2016). Bitcoin is an example of peer to peer (P2P), which indicates nodes can communicate uninterrupted to one another.

The P2P networks is not a new technology. The P2P technology pioneered the development of Distributed Hash Tables (DHTs) such as BitTorrent prior to the development of Blockchain. DHTs are useful for decentralized storage and streaming of data. The blockchain does not eliminate the necessity of a DHTs but acts as a supplement. The blockchain turns out to be spectacular at addressing DHT's biggest security problems (Raval, 2016).

Blockchains are data structures with write-only, without administrative rights to changing or erasing the records. The data structures are called blocks and are distributed across a P2P network (Tschorsch and Scheuermann, 2016). Every block retains the preceding block's cryptographic hash function and is used to form a connection among themselves. The blocks joined together to create a whole chain, hence the name blockchain. The hash function preserves the blockchain's privacy, authenticity and immutable. The blockchains operate across the Internet on a P2P network, with no central system therefore, no single point of failure or attack. Every peer adds value to the network maintenance storage and processing capacity (Tschorsch and Scheuermann, 2016). The author proceeds by providing a broader description for blockchain systems, that is a combination of the distributed P2P network, blockchain, and consensus model.

To understand the blockchain, it is therefore important to understand distributed systems, since blockchain is essentially a distributed system at its core (Bashir, 2017). Specifically, a blockchain is a distributed network that is decentralized. According to Van Steen and Tanenbaum, (2016) “A distributed system is a collection of autonomous computing elements that appears to its users as a single coherent system”. Distributed systems could also be defined “as one in which hardware or software components located at networked computers communicate and coordinate their actions only by passing messages” (Coulouris et al., 2011). One can rightly characterize distributed systems as a compendium of computing elements that are capable of independencies alone. Machines that make up the distributed systems perhaps computers, physical server, virtual machines, containers or nodes, which could be linked to the network, has a local memory and communicate through messages (Gibb, 2019). Every device has its end users with the distributed systems that enables participation resources, or interaction services (Gibb, 2019). A node is likely represented as a software or a hardware device. Nodes in their autonomous state collaborate to give users (individuals or programs) a single system. A single system, has the propensity to vary from small devices in sensor networks to high functioning workstation computers (Coulouris et al., 2011). A distributed system is built and supported by architectures that comprise of software and hardware. All devices are connected through the CPU to the network then to the requestor through an interactive method. Distributed systems could be divided into four model's architecture: Client-server, Three-tier, n-tier and Peer-to-peer. Being that blockchain is a sub tech of the distributed system it is, therefore, intelligent to examine existing method/framework for the development of a P2P application in comparison to our core aim on methods for developing a decentralised application on blockchain.

2.1.2 Technique and Methods for Developing a P2P Application on Distributed Systems.

Since the blockchain technology is built around the peer-to-peer topology distributed network that allows data storage universally on thousands of servers, that could be accessible by everyone on the network and other entries on real-time, making it challenging for a single user to take sole control. We look further into exploring existing methods for developing a P2P distributed application.

According to Coulouris et al, (2011) "Peer-to-Peer (P2P) is a way of structuring distributed applications such that the individual nodes have symmetric roles. Rather than being divided into clients and servers each with quite distinct roles. In a P2P application a node may act as a client and a server." Peer-to-Peer computing or networking could be described as a partitioned distributed application. An architecture that divide task or work between nodes. In contrast to a traditional client-server design of the Internet, P2P nodes are owned by diverse individuals which in return of services offered by the network, donate resources. Resource interchanged comprises of contents, P2P shared file, storage size, CPU drives and storage grid system (Babaoglu, Meling and Montresor, 2002). A peer could serve either as a provider and also a consumer, just as the master and the slave. As more nodes become inclusive in the P2P network, the demand increases as well and likewise the systems resources. Nevertheless, the P2P system is more economically conserved, scalable and robust compared to a traditional-client server which tolerates the whole capacity. Being that in a P2P network each peer serves as a service provider, makes the network distributed with redundancy leading to the robustness of the service. A P2P network which is rightly proposed could actively utilize the geographical and ISP diversity to analytically route content from source to the peer nodes and reduce traffic (Li, 2008).

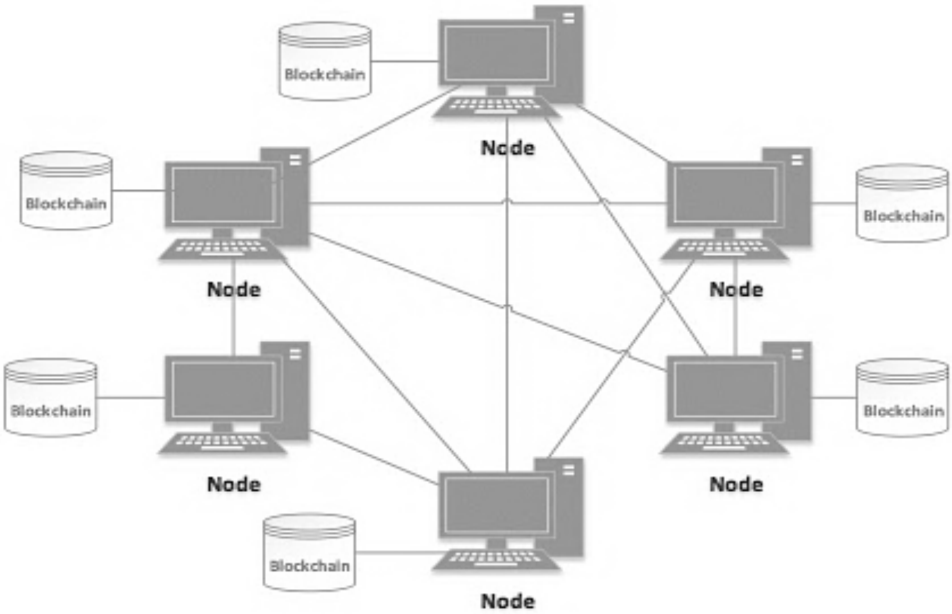


Figure 1. P2P network. Source: (Koteska, Karafiloski and Mishev, 2017)

Furtherly into this research, we tend to explore an existing P2P framework, its core design, mode of distribution, and operations in comparison to the blockchain technology.

2.1.3 Anthill: A Framework for the Design and Analysis of Peer-to-Peer. (Babaoglu, Meling and Montresor, 2002)

Anthill is a framework that supports the function and valuation of the P2P application based on the idea of multi-agent software design from CAS (complex adaptive systems) (Babaoglu, Meling and Montresor, 2002). Traditional methods for the development of a distributed application tend not to be suitable in handling scale and dynamism that makes up a recent P2P system. For instance, flooding style communication is regularly used by most file sharing applications which restrictions scalability (Oram, 2001). The method supported here is basically unlike those adopted in most P2P routing algorithms which complex protocols need reconfiguration of the routing tables where nodes connect or departure due to various reasons out of the system (Rowstron and Druschel, 2001).

We can state that to content the necessities of the P2P application development needed a change that will be inclusive of modification, resilences and self-organization as the base of interest(Babaoglu, Meling and Montresor, 2002). The basic aim of Anthill is to simplify the development and strategy of the P2P framework while using the complex adaptive systems to monitor and rate the functionality (Babaoglu, Meling and Montresor, 2002). We can suggest that this idea and methods derived from CAS have the concept to a new paradigm for structuring the P2P systems.

Anthill is a terminology that represents colonies of Ants. In Anthill a cluster of peer to peer node are staged as nests. Various nests deal with local users requests by generating ants which travel through the network, share computational and storage resources. The development and deployment of P2P applications are fully postulated by Anthill API through the availability of communication, storage management, topology and ant scheduling. P2P applications could be built by developers after a well erected P2P structure, following the ant algorithms and exploiting the Anthill API to resolve the problem of the application. The utilization of Anthill solves developer concerns of communication, security and scheduling strategies. A simulative environment exists in Anthill that assists developers to evaluate and examine the P2P systematic performance before it is deployed (Babaoglu, Meling and Montresor, 2002).

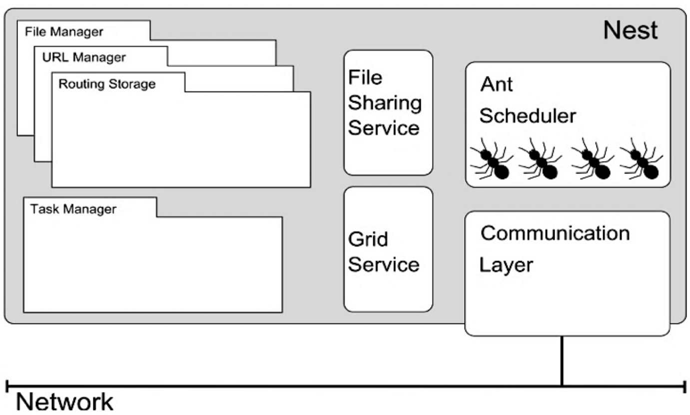


Figure 2. Architect of a nest. source: (Babaoglu, Meling and Montresor, 2002).

The nest architecture comprises of three components that are the: ant schedular, communication layer and resource managers. The schedular is in charge of security enabled by the use of a “sandbox” to limit the resources accessible and preventing dangerous actions that could be executed by the ants, e.g., local file access, etc. The communication layer helps network topology management, aid in uncovering new nests and movements between nests. Every node in the network has an exact identifier and for communication to take place with different remote nodes, this identifier has to be recognized. A set of nests recognized by a node is called a neighbour of a node. The resource manager is utilized by nests when offering resources to ants. For instance, a file in the CPU cycle or file-sharing system could be a resource and the disk-based storage or task schedule the resource manager. The routing storage is utilized in evaluation concerning routing. Routing storage is built on the notion of hashed keyword routing. The file manager aids shared files. The URL manager sustains the distributed index. The computational grid application executes task assigned in the task manager. Through Anthill, a file-sharing application was device called Gnutant, that emulates the ant behavioural. It creates a distributed index comprising of URLs that is acknowledged, by navigating the network for resources and keeping trace to be explored in the future (Babaoglu, Meling and Montresor, 2002). An open-source P2P project by Sun Microsystems JXTA was used as the basis for the runtime environment (Joseph D. Gradecki, 2002) and a Java prototype was developed for Anthill. JXTA is designed at launching a network programming platform for P2P systems. For instance, JXTA permits the use of various transport layers for communication, also TCP/IP & HTTP, and handles issues linked with firewalls and NAT (Network Address Translation).

Nevertheless, knowing that the blockchain-based system is built around the P2P distributed ledger, it evident that this method is suitable for development and deployment of P2P applications like Gnutant, Napster through the deployment of Anthill framework but not appropriate or suitable for the development of a decentralized blockchain application due to several reasons such as;

There framework incompetence to deploy a smart contract on the Anthill. The capability to integrate smart contract deals with the issues of security, privacy and automatization. Most P2P application built on a P2P framework could be distributed and not decentralized whereas a P2P on the blockchain technology is distributed as well as decentralized. P2P framework faces numerous issues with security that is one of the core features of blockchain technology. In general, it is possible to erase data due to storage constraint or the need the remove irrelevant info, which nullifies what the blockchain technology implies and destabilize the use of digital currency (Kozlovski, 2018) etc. Conversely, database transactions are rather complex to employ in distributed systems. Based on the limitation of the P2P system as mentioned above, the thesis focuses on the development of decentralized applications on a blockchain framework.

2.1.4 Integrating Multiple Perspectives in System Development. Viewpoint Framework (FINKELSTEIN et al., 1992).

The difficulty of coordinating and planning relating to different perspectives at numerous phases of development and strategies has contributed greatly to numerous challenges in development. Viewpoint approach explains the development of composite structures and approaches used to design complex systems. According to Finkelstein et al., (1992) “A viewpoint can be thought of as a combination of the idea of an “actor”, “knowledge source”, “role” or “agent” in the development process and the idea of a “view” or “perspective” which an actor maintains”. An agent in this context could be described as an entity can observe and react to its system. In certain situations, the system's agents involve both human and inanimate agents. In software terms, the viewpoint is a loosely coupled, locally controlled entity that encapsulates partial system and domain information, defined in a specific, acceptable representation scheme, and partial information of the design process. In software terms, the viewpoint is a loosely coupled, locally controlled entity that encapsulates partial system and domain information in a specific standard illustration scheme and partial information of the design method (Finkelstein et al., 1992).

Viewpoint consists of components called slots: which are styles, domain, specification, work plan and work record.

Style- The notation and method through which the viewpoint describes whatever it can do.

Domain- Which specifies the "world" aspect, illustrated in the style.

Specification- Statements presented in the form of the viewpoint outlining specific domains.

Work plan- Overview of the method by which specification could be constructed.

Work record- Previous records, and the present state of progress.

Modelling approaches which are in existents are believed to be restraining. One of their key weakness is frequent neglecting of the design representation in favour of design procedure. Perhaps we could presume that both representation and the processes are intimately connected. There is a use of a common data model, in viewpoint which is typically supported by a common database or server, supporting several "views." The framework supports countless existing methods in the development of systems. Numerous existing developments are performed according to systems, such as HOOD, CORE, SSADM, SADT etc., which proposes styles of representation and accompanying strategic plans. While the viewpoint framework provides a means to describe these approaches as components of viewpoint models using the "multiple perspectives enquiry" framework. Viewpoints have a flexible framework that represents the complexity of the 'real world' program creation process (Finkelstein et al., 1992). This framework could be integrated in the development of a distributed and decentralized application. Through this representation and processes, we could eliminate time wastage and cost disbursed on irrelevant procedures while developing.

2.2 Blockchain Technology

The Blockchain technology origination could be tracked to a set of unidentified persons or group called Satoshi Nakamoto in the year 2008 (Nakamoto, 2008) who extended aids in tackling the difficulties, liabilities, inadequacies and amount of the current transactional system practised by revolutionizing the financial sector through the emerging of Bitcoin in 2009.

The blockchain in a plain view seen as a data structure chain which comprises of blocks of information and data that stores the blocks in an encrypted form as a distributed ledger. The blockchain technology validates, utilising the block type data structure. Stored data in the blockchain technology uses distributed node consensus algorithms to create and reform data and utilising encryption to make sure data communication and access security (Lu, 2019). Blockchain stages could still be said to be under-utilized and still a lot to be exploited therein. The first version of the Blockchain 1.0 was built in 2008 (Swan, 2015). Bitcoin and cryptocurrencies were the most prominent here with bitcoin as a breakthrough to the financial market (Nakamoto, 2008).

Blockchain 1.0 was later upgraded with a Blockchain 2.0 in 2013 which made availability for the deployment of smart contracts, attainable agendas like collaborations of various entities and application into several industries (Luu et al., 2016). This strengthens the issue of trust and distribution of automated resource on a large scale through the implementation of blockchain technology. Blockchain 3.0 is an upgraded version of the of version 2.0 which is aimed at solving existing problems, faster, cheaper and proficient way of executing transactions. Blockchain 3.0 initiated the full operational functionality of DApps which support the deployment of backend code on the blockchain in a decentralised P2P network. This means user frontend code and interface on the DApps could virtual be written in any language making the structure flexible e.g. Ethereum Swarm (Blockchain evolution: from 1.0 to 4.0 - Unibright.io - Medium, 2017; Bitcoinik, 2019). It is renowned beyond applications dealing with currency, finance and markets, it, therefore, stretches to different works like the government, health etc.

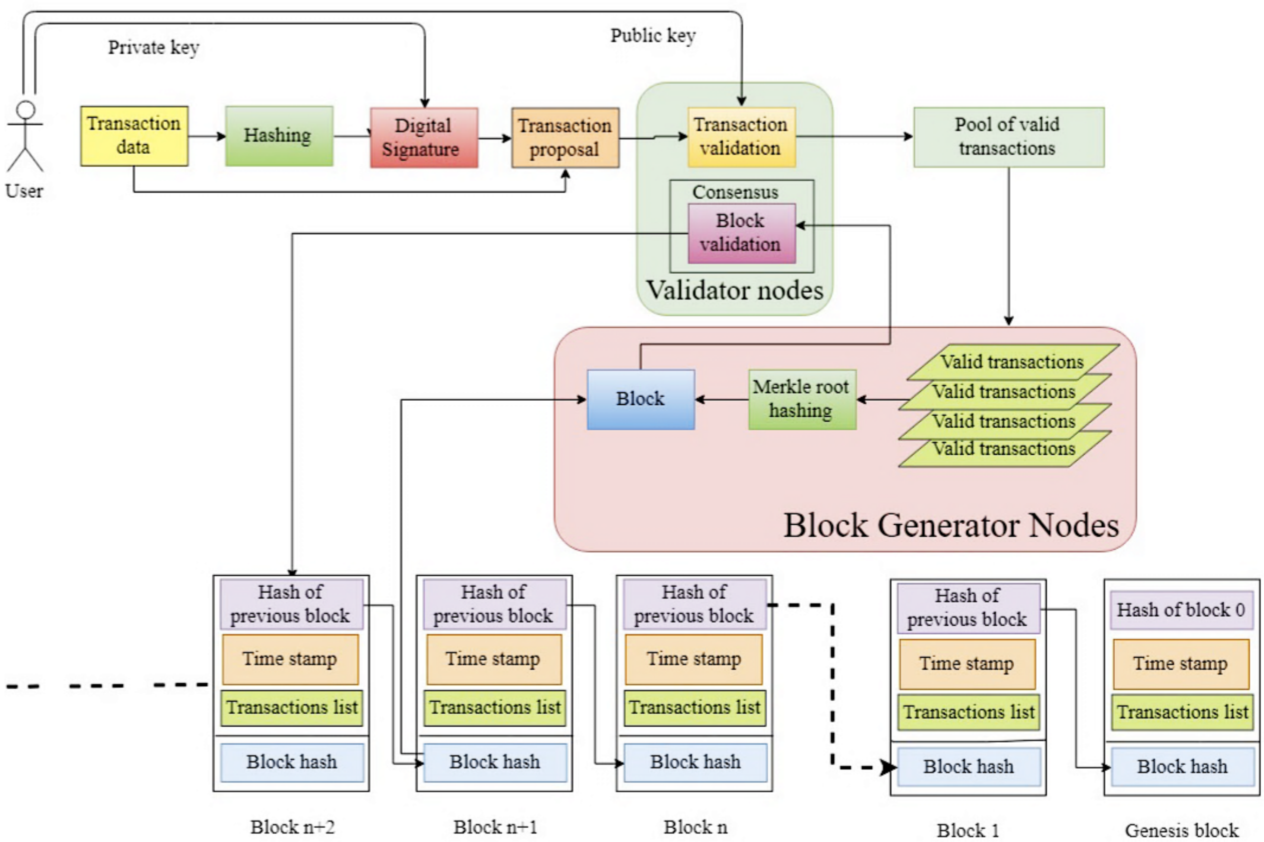


Figure 3. Blockchain Overview. Source: (Ismail and Materwala, 2019)

Figure (3) above illustrates the basic components and overview of blockchain technology. This simply describes how the blockchain technology is fully operational and taken advantage of by other software.

On the other hand, when operating as a centralized service, some organizations or entities offers connectivity in terms of economy but also takes away our freedom. Using banks as a centralized organization, they can prevent transactions or confiscate a customer's account if deemed right by them. In the application of blockchain, such cannot happen as there is no centralized authority (SINGH, 2019). There is digital freedom with blockchain, and an individual can use and take their money whenever they want, without any authorization. Such digital freedom relies heavily on the backbone of blockchain technology (SINGH, 2019). Other importance of blockchain technology includes excellent use-cases, better security, and lesser transaction time.

2.2.1 Components of Blockchain Technology

The blockchain technology is a decentralised system which eliminates the concept of a centralized network. It comprises of six (6) layers which are, consensus, data, network, contract, application and service. Figure (7) below breaks and simplifies the components of these layers for understanding and comprehension.

Layers	Major technologies or components
Data layer	Data Block, chain structure, timestamp, Merkle tree, cryptography
Network layer	P2P network, verification mechanism, broadcast protocol
Consensus layer	PoW, PoS, DPoS, PBFT, …
Contract layer	Smart contract, script coding, incentive mechanism
Service layer	Ethereum, hyperledger, IBM Azure BaaS, …
Application layer	Cryptocurrency, healthcare, cloud service, …

Figure 4: Layers of the blockchains source: (Lu, 2019)

The blockchain contains three basic elements according to the structure: a time-stamp centred chain block structure, a distributed storage mechanism system on a P2P network function, and a consensus framework based on decentralized nodes. Data is validated and stored in the ledger which coordinates and executes all transactions, claims, and transfers (data layers). Partakers are connected in a peer-to-peer environment with no influence from a third party (network layer). Partakers process transaction based on a common consensus algorithm, exchanging information and interactions (consensus layers). (contract layer) Contract decisions, algorithms and smart contracts are built up and incorporated into the blockchain for contract executions (Lu, 2019).

Distributed ledger technology (DLT): is a rapidly evolving approach to data storage and sharing across several (or ledgers) data stores (Natarajan, Krause and Gradstein, 2017). All blockchain are DLT but all DLT are blockchain. Note that there are other technologies which are DLT but not blockchain such as DAGs (Direct Acyclic Graph) e.g., Iota[1], Nano. This technology enables for capturing, exchanging, and synchronizing transactions and data through a distributed network with multiple people involved in the network (Natarajan, Krause and Gradstein, 2017). Before entering transactions into the ledger, all the nodes in the network come to a consensus (using a consensus algorithm). This function essentially makes blockchain immutable and secured (Ramachandran and Krishnamachari, 2018).

Distributed Consensus: The consensus mechanism is located on the consensus layer of the blockchain. Blockchain is a distributed database that runs on the P2P network. Every node stores a copy of the validated ledger and unvalidated data which is required to be recorded and added to the ledger. For a transaction to be carried out on the blockchain, the nodes need to sync into an agreement on the ledger content and conditions which is accomplished through the aid of a consensus protocol which the data generated are validated based on chronological and order. These consensus protocols updates and confirms that all blocks in the chain are correct (Andrian, Kurniawan and Suhardi, 2018). Consensus frequently utilized are:

• **Proof-of-Work (PoW):** To prevent complication, the nodes must accomplish a set of mathematical algorithms to accept blocks. The reward is given to the earliest node capable of executing these algorithms. This consensus protocol entails a huge computational capability.

• **Proof-of-Stake (PoS):** The node responsible for creating new blocks is selected deterministically between the wealthiest nodes. The most rewarded nodes generate new blocks.

[1] https://docs.iota.org/

• **Proof of Authority (PoA):** This entails no required computational power, being that the nodes accountable for generating new blocks are nodes are delegated authority or permissions (Gil and Mu, 2019).

Smart Contract: The smart contract is located on the contract layer. it could be described as a code which conditions an executed transaction stored in the blockchain and has no sole control from any central authority. A smart contract is a series of digitally specified engagements. Smart contracts provide the conditions for execution and the logic for execution. If the condition is met, implementation logic is automatically executed. In the participant's view, a smart contract is an automated service program (Lu, 2019). Smart contract was introduced in blockchain 2.0 shared ledger (Christidis and Devetsikiotis, 2016). The smart contract is facilitated to process data, operate assets transactions and manage smart assets. The smart contract after execution can't be deleted or erased therefore it's trusted and implemented to secure transactions. Ethereum network could be cited on so many case study exploring smart contracts in development (Andrian, Kurniawan and Suhardi, 2018).

Asymmetric Encryption: is the technology utilised in fortifying the security of the blockchain network. The technology comprises of a public key and a private key (Truu, Buldas and Laanoja, 2017). It supports the pairing of public key and private key whereas providing users who are unidentified to exchange data which are encrypted. It has two basic functions that are data encryption and digital signature in the blockchain system. Asymmetric encryption based on studies goes beyond authentication and making signs of executed transactions thus also supports recorded data stored in the blockchain (Yli-Huumo et al., 2016). Nevertheless, the concepts of asymmetric encryption in the blockchain makes it transparent and efficient in recording transactions without fear of breaches or attackers.

Blockchain could be categorised into 3 major types, which are the public blockchain (permission-less), the private blockchain (permissioned) and the consortium (hybrid).

Public Blockchain: The public blockchain could also be referred to as a permission-less blockchain system. The public blockchain is fully decentralised that simply means every node on the distributed system can take part in consensus, analysing, examining aspects of data confined in the chain. Likewise, economic incentives could be attained through the means of contributions. The public blockchain like bitcoin was the first to be actualized before it was transformed to fit other spheres (Sankar, Sindhu and Sethumadhavan, 2017; Dinh et al., 2018). The consensus is reached only through an open process and everyone is welcomed to participate which nullifies the act of single control or ownership (Dimitri, 2017).

Private Blockchain: Permissioned is also referred to as a private blockchain. It is characterized as a centralised system. A central authority grants permission and access on data on the blockchain while the read permission is selected to the public. It is used by private industries and banking institutions for internal data management and audition. The private blockchain is not different from other blockchains in anyways but is constrained to fewer groups of people and organizations (Lin and Liao, 2017; Zheng et al., 2017)

The Consortium: is a partly distributed hybrid blockchain. Every block is mutually determined by pre-selected nodes. The consensus processes tend to be huge and controlled by some known servers that utilise rules accepted by all participants. Not all node take part in the consensus procedure, most only have access to carry out transactions. Over a consensus contract, several institutions can come collectively to form a consortium system for a mutual objective e.g. Hyperledger blockchain (Brousmiche et al., 2018; Gu et al., 2018).

Some major comparisons on different forms of blockchain technology have illustrated in the table (1) below:

Table 1. 1: Permissioned and Permission-less. Source: (Natarajan, Krause and Gradstein, 2017)

Features	Permissioned (Private)	Permission-less (Public)
Speed	Less partaking nodes in the network, executing transactions are faster.	A huge amount of time spent to disseminate transactions which make the procedure a bit slow.
Central	Some degree of control by a single group.	Everyone has access due to its decentralized.

Consortium	Only selected nodes can join the network.	Everyone can join the consortium process since the network is public.
Efficient	Only selected nodes can partake which makes validation processes easy and efficient	Lot of time to validate transactions being that there are high numbers of partaking nodes.
Access	Records are private and are visible only to the owner group	All records are visible to the public.
Security	Security by access control in a small-scale network	Security by a large-scale distribution network
Trust	The huge value of trust needed among participant.	Participants not required to trust each other
Sample	R3 Corda, Hyperledger Fabric.	Bitcoin, Ethereum

Table 1.1 above shows a listed number of attributes based on distinct features regarding different technology applied and unique depending on different platforms, usability and organizations. Some of the frequently used blockchain platforms based on the level of functionality and mechanism are as follows:

Ethereum: Established as a conceptual framework for programming, Ethereum is an open-source and public blockchain-based system which supports the full features of smart contracts (LeewayHertz, 2020). Ethereum utilize a proof-of-work algorithm known as *Ethash* that is demands a lot of memory making it difficult to mine using expensive ASICs-specialized mining chips. Smart contracts in Ethereum are implemented in ***Solidity*** programming language that is JavaScript subset (LeewayHertz, 2020).

Hyperledger Fabric: Hyperledger Fabric is a blockchain open-source framework, perfect for developing enterprise applications. It supports distributed ledger solutions on private networks where all participants have established and permitted identities (LeewayHertz, 2020).

Hyperledger Fabric is based on modular architecture and can advance flexibility and resilience in blockchain applications. Components of the pluggable architecture contain encryption, or consensus. Hyperledger Fabric facilitates smart contracts and considers this functionality as "Chain code."(LeewayHertz, 2020).

2.2.2 Key Attributes of the Blockchain Technology

Transparency: All participants through the aid of the blockchain can query and distribute data in decentralized system through the node. In the blockchain, members can query records which make the distributed system reliable and transparent. The recorded and transferred data information in the blockchain is open and accessible to participants at all times (Bonneau et al., 2015; Lin and Liao, 2017). “Participants know where the asset came from and how its ownership has changed over time” (Gupta, 2018). When it comes to the public blockchain, every member is knowledgeable about activities, transactions, authentication, or validation executed in the network. Permissioned blockchain utilises encryption of the messages which access is granted to only approved people in the system to the data.

Decentralization: Through a collection of peers to peer nodes in the blockchain, it is able to store and distribute transactions. The distributed system simply shares information on the blockchain network through nodes. Blockchain distributed ledger system transactions create trust and elude the excessiveness of a central body and middlemen in the industry (Perera et al., 2020) for example, financial institutions like a bank which most times turnout in deferments, or placing administrative blocks and restrictions. In the blockchain system, a third party is eradicated as consensus algorithms are utilised to sustain the consistency of data in the system.

Anonymity: Participant in the blockchain can choose to stay anonymous to safeguard their identity and privacy. Blockchain creates addresses which replace people physical identification which simply means members can transact without fear of identification disclosure to others (Sun, Yan and Zhang, 2016).

Security: The blockchain safeguard the validity of data stored and resist fraudulent attacks through the usage of encryption mechanism encompassing the asymmetric public-key cryptography (Tapscott & Tapscott 2017). The blockchain encryption helps to preserve the privacy of personal data. Digital signature aid in the authenticity and integrity of data.

Trust: Due to the decentralised system upon which the blockchain built, nodes interaction in the network create a sense of trust and confidence among participants. The blockchain is designed on principles of the peer to peer protocol with mathematical algorithms which employ the hash functionality and consensus to build a standard computing system. Along the year participant have learnt trust in the system (Narayanan et al., 2016).

Immutability: Transaction carried out in the blockchain technology cannot be undone nor changed. Blockchain supports, create and read process only (Lewis, McPartland and Ranjan, 2017). Stored data on the blockchain distributed system is immutable as this means past, or present transactions are stored on the ledger as a way of securing system (Atlam et al., 2018).

Auditability: The transaction executed in the blockchain through a timestamp is recorded and validated. Participants could easily trace any transaction through the node on the network to obtain valuable information of a transaction. It shows nothing could be erased, or deleted from the network after execution (Wang et al., 2018).

2.3 Theoretical Frameworks

In this segment of the research, we analyze various literature reviews to understand existing blockchain frameworks/methods in certain use cases. Comparing methods of developing a DApps with outlined strength and weakness based on our cases of references.

2.3.1 Privacy-Preserving Techniques (Ancile) By (Dagher et al., 2018)

Evolution in technology has likewise increased many techniques for violating digital privacy and security. Healthcare sectors has been the real focus of information thievery due to several factors to obtain private information such as patient names, social security number, and addresses etc. EHRs Mismanagement and theft are also on the rise due to lack of efficient security systems and fragile policy implementation. Most EHR systems are owned and maintained by individual owned companies, which are high-risk breaching privacy-preservation and security control. Records could easily get lost, or stolen without a trace. According to the Health Insurance Portability and Accountability Act (HIPAA) of 1996 (Califf and Muhlbaier, 2003), has four goals they tried to attain which were, the right to the transference of records, or rights to switch health insurance when changing a job, to reduce fraud and abuse, mandate standards for digital billing, and demand privacy & security of secured health data (PHI).

To achieve these goals a framework is designed on the blockchain technology for a secure, interoperable and proficient network through replicating all data on every node on the blockchain system. Ancile is a framework on an Ethereum-based blockchain platform that uses the smart contract to increase access control and the complication of data with enhanced security through the aid of cryptographic advancement. The major aim of this framework is to allow and grant access control of medical data to patients to oversee their records. The framework also addresses communication providers, third parties and patient while enhancing the privacy and security of medical records. The blockchain comprises of three kinds of nodes which are, the full, light and archives nodes (Go-Ethereum, 2017) to combat the issue of scalability. Full nodes stores every block and transaction executed in the blockchain, light nodes stores just block header which is the hash of preceding blocks, the hash of Merkle root and the archive node like the full node stores transactions, transaction receipts and help to retrieve info on the blockchain network (Buterin, 2015). The use and interchange of the three nodes the Ethereum blockchain can scale. In Ancile, the encrypted record, the query link and symmetric key are tools which must be accessed to get to the EHR, and they are placed in separate areas. Ancile comprises of three (3) major software components which are: Database manager, Cipher manger and the Ethereum go, client. Ancile is designed on a permissioned Ethereum blockchain. it comprises of six (6) smart contracts which are:

Consensus Contract, Classification Contract, Service History Contract, Ownership Contract, Permissions Contract and Re-encryption Contract. These different smart contracts benefit the patient to improve utility and reducing the necessity to communicate with all contract which increases privacy preservation and effectiveness. According to Samaria,(2016) great height for the storage of medical records on the cloud where it could be only seen by patients which transfers full ownership to the patient, we forget to highlight the necessity for access grant to a patient's doctors to have access as well of this records like in cases of emergency. In this case, multiple entities are taken into consideration. For example, consider a situation where a doctor is to keep a patient medical record undisclosed to them due to certain casualties. A company like JP Morgan's Quorum (JPMorgan, 2018) likewise deploys a private transaction for privacy but in comparison to ancile lacks procedures such as proxy re-encryption, deployed on ancile framework which helps to rationalize the secured transfer of EHRs. The act of mining incentive is not inclusive in this framework since it aimed at patient ownership of their records. It purely represents a simple system which utilizes the consensus algorithm rather than the frequently used proof of work.

Ancile, shows a typical blockchain system that could be highly decentralized with few nodes having a higher priority than others and so by initiating several smart contracts to diversify information while offering privacy, data integrity. Only a few nodes are needed for validation while executing transactions containing data hashes, as a part of a private transaction which reduces the storage and mining costs of the blockchain as it scales. To enhance scalability on the platform, the system only stores hashes and small records on the block-chain which means less storage space is been utilized.

The framework has shown limitations in the huge amount of computational power and also time is taken in the execution of transactions due to the deployment of several smart contract execution on full nodes. Furthermore, the time taken for patients' registration, adding a new node and also paperwork to kick start the whole process is quite cumbersome which could be an area of future improvement for a better framework. A tradeoff on gas cost for privacy and security being that transactions and steps in the blockchain are numerous attracts a higher performance cost.

2.3.2 Traceability Ontology-Driven Design By (Kim and Laskowski, 2018).

The act of provenance in modern times has taken a paradigm shift since the involvement of blockchain into conception. The adoption of blockchain in the area of the supply chain has given better potential at provenance. This use case illustrates and analysis the traceability ontology design while translating it to a smart contract on an Ethereum blockchain network which implements provenance trace and executes traceability controls. "Provenance" according to Merriam-Webster, (2016) means "source or origin or the history of ownership of a valued object, or work of art or literature." According to Gruber, (1993) Ontology can be defined as "an explicit specification of a conceptualization." In ontology conceptual, data need to hold a common interpretation in a shared database. Ontology could be formal, Informal or a combination of both which are: lightweight (informal), Tove (formal) or in-between semiformal. An informal or semi-formal ontology modelling approach could enhance an improved data standards, processes and practices in business migrating and operating on the blockchain-based platform.

Database distributed among different organizations, ought to have a common standard of communication which could be enabled for automated inference and verification within a program that runs throughout these organizations. Therefore, ontologies embed in the blockchain is believed to advance the supply chain provenance and delivery (Frey et al., 2003; Grubic and Fan, 2010) considering into further studies. The Tove Traceability Ontology illustrates a bigger concept in traceability perspective as created after essential enterprise modelling construct as a proof of concept stands as the design model in the blockchain implementation.

The ontology-based blockchain application as a proof of concept that was built on an Ethereum platform to enable the use of a smart contract. Solidity which is an object-oriented language was used as the programming language for development. The platform is connected to a web browser interface. A Truffle framework by consensys was used to create a JavaScript-based (Web3 ABI) interface for communicate with the smart contract in the system to define and identify the state of objects and also perform the trace.

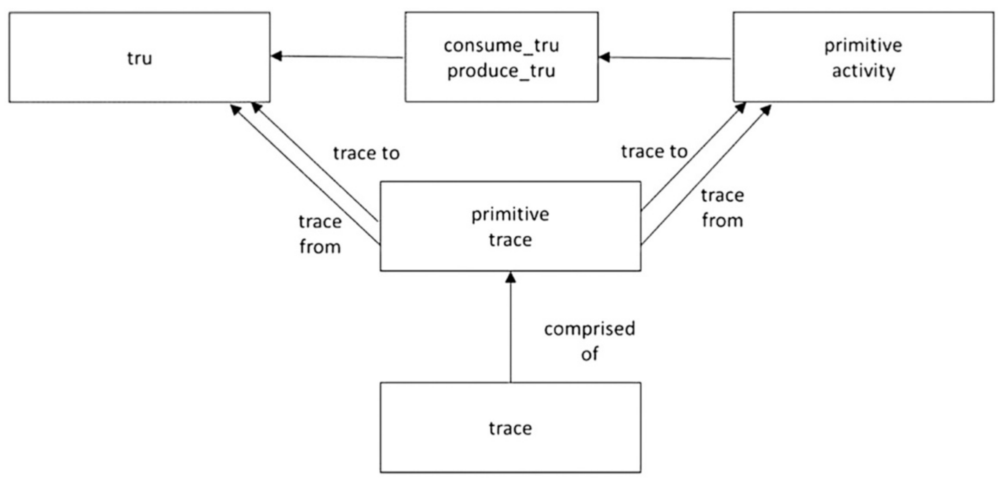


Figure 5. TOVE Traceability Ontology data (Kim and Laskowski, 2018)

The above image simply shows a traceability ontology data model illustrated with a UML methodology frequently used to represent or translate business processes into an objected-oriented language in software engineering. Here the ontological enterprise model and a simplified TOVE ontology were modified and implemented as a tracking real-time system, able to trace the provenance of TRU in return to whichever TRU in the provenance chain which includes:

Under (Kim and Laskowski, 2018) "Axiom 1. A TRU is produced only once. Axiom 2. A TRU (resource) can only be consumed if it is available (exists and is not consumed). Axiom 3. A Primitive Activity consumes one TRU and produces one TRU. Axiom 4. A TRU is first known to exist at the time it is first consumed or produced. Axiom 5. Once consumed, a TRU cannot become unconsumed." In first-order logic these axioms were conveyed into the smart contracts as a proof of concept, performing provenance trace and administering traceability constraints on the blockchain platform.

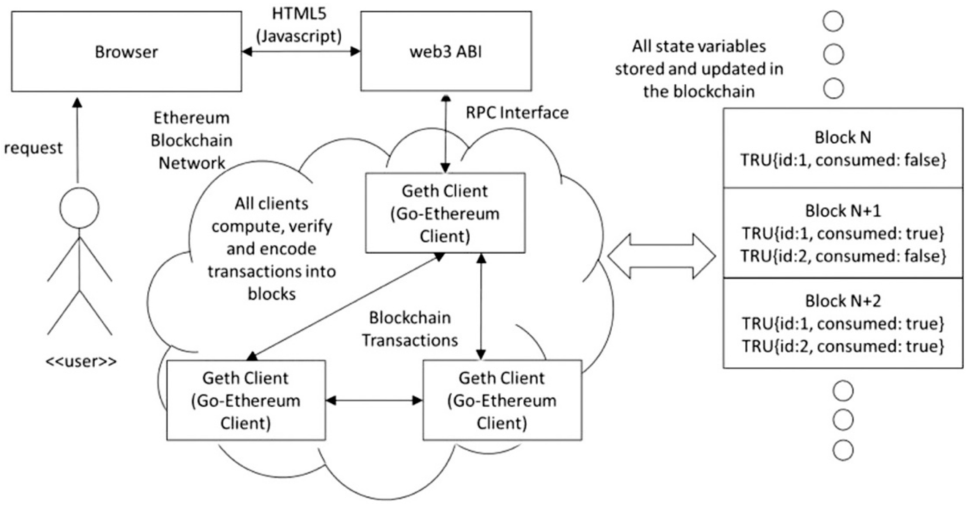


Figure 6. User interaction with the blockchain (Kim and Laskowski, 2018)

The diagram above is an activity chart with interactions between the users and the blockchain application in an ontology-driven design. As blockchain advances, more businesses take advantage this innovation and so researches and scholars go deeper to finding more opportunities to exploit as ontology is believed to greatly contribute to the design on the blockchain and more especially the supply chain enterprise. Based on Grubic and Fan, (2010); Kim and Laskowski, (2018) analysis there are still lot of opportunities and research to be carried out to explore the full potentials of the ontology application especially with advancements of the internet of things (IoT) coming into full usability in supply chain.

2.3.3 Strategy for Developing a DApps (Marchesi, Marchesi and Tonelli, 2019).

In this section, we describe the step and process used in software development. Taking the Agile method as the proposed technique in implementing the User story (US) to develop with a formal notation such as UML diagrams a representation in a Sequence, and Class diagrams showing the processes of the design with addition concept represented in blockchain development. In this use case, we use the UML diagrams to interpret the smart contract (SC) specificity in the content of solidity. According to (Rocha and Ducasse, 2018) the smart contract design could be complemented with three basic software engineering tool which is: UML, BPMN and Relationship diagram. The proposed design method focuses on the application of an Ethereum blockchain platform with solidity language. Outlined below are the processes are taken into consideration during development.

Unified Modelling Language (UML): is a universal resolution standard to detail, describe, design, visualize, and document software systems (Rocha and Ducasse, 2018). Since the 1980 UML has been the standard integration of numerous object-oriented modelling notations. Consequently, UML was particularly designed for modelling software programmed using the object-oriented paradigm (Scott et al., 1999). UML outlines 13 diagrams classified into three categories which are: The Structure, the Behaviour, and the Interaction. UML diagrams comprise of the Class Diagram, Object Diagram, Component Diagram, Composite Structure Diagram, Package Diagram, and Deployment Diagram (Scott et al., 1999). Behaviour diagrams comprise of Use Case Diagram, Activity Diagram, and State Machine Diagram. Lastly, the interaction diagrams consist of the Sequence Diagram, Communication Diagram, Timing Diagram, and Interaction Overview Diagram (Scott et al., 1999).

Business Process Model and Notation (BPMN): Could be described as a graphical notation to detail business process (Mendling and Weidlich, 2012). The BPMN shows a flow-oriented illustration needed by the software which portrays and expresses the behaviour of the practical necessity (Dumas et al., 2009).

Container, Context, Component and Code (C4)[1]: The C4 model is an abstraction-first model for software architecture flowcharting, focused on abstractions that signify how computer systems engineers imagine concerning software and development. The limited collection of different kinds of abstractions and illustrations enables the C4 model understandable and useable.

Corda Design Language (CDL)[2]: is a mark-up language used to precisely characterize and model on a CorDapp. This modelling language is an inbuilt modelling tool for corda blockchain platforms and cannot be extended.

Modelling: Consequently, we try to introduce modelling approaches for a Blockchain-Oriented Software (BOS) which utilizes smart contracts for integrating part of the business logic properly into the blockchain. Firstly, we describe an outlined sample of a BOS and its smart contract for model sampling. Secondly, to model our sample we propose a data-driven method. Thirdly, we use the UML class diagram to display a structure-oriented approach. And Lastly, we introduce a modelling process for exploring BPMN. The object-oriented UML notation comprises of six diagrams for shaping processes design. Been that smart contracts are quite specific to classes; we could use the UML diagrams. So, by utilising the UML, we could model an object-oriented application and the blockchain framework. The benefit of utilizing the UML design diagrams is its simplicity to model the smart contracts and describe their functions as well as data attributes. Besides, UML is common for modelling processes, it's easy to grasp for software engineers and developers. The drawback of structure modelling is the inability to define the business processes operational behavioural in other words a need for process-focused behaviour modelling or a methodology necessity (Rocha and Ducasse, 2018). various developers and software engineers have numerous mark-up language or notations they use in modelling and documenting while developing a DApp.

---

[1] https://c4model.com/
[2] https://solutions.corda.net/corda-modelling-notation/overview/overview-overview.html

▯

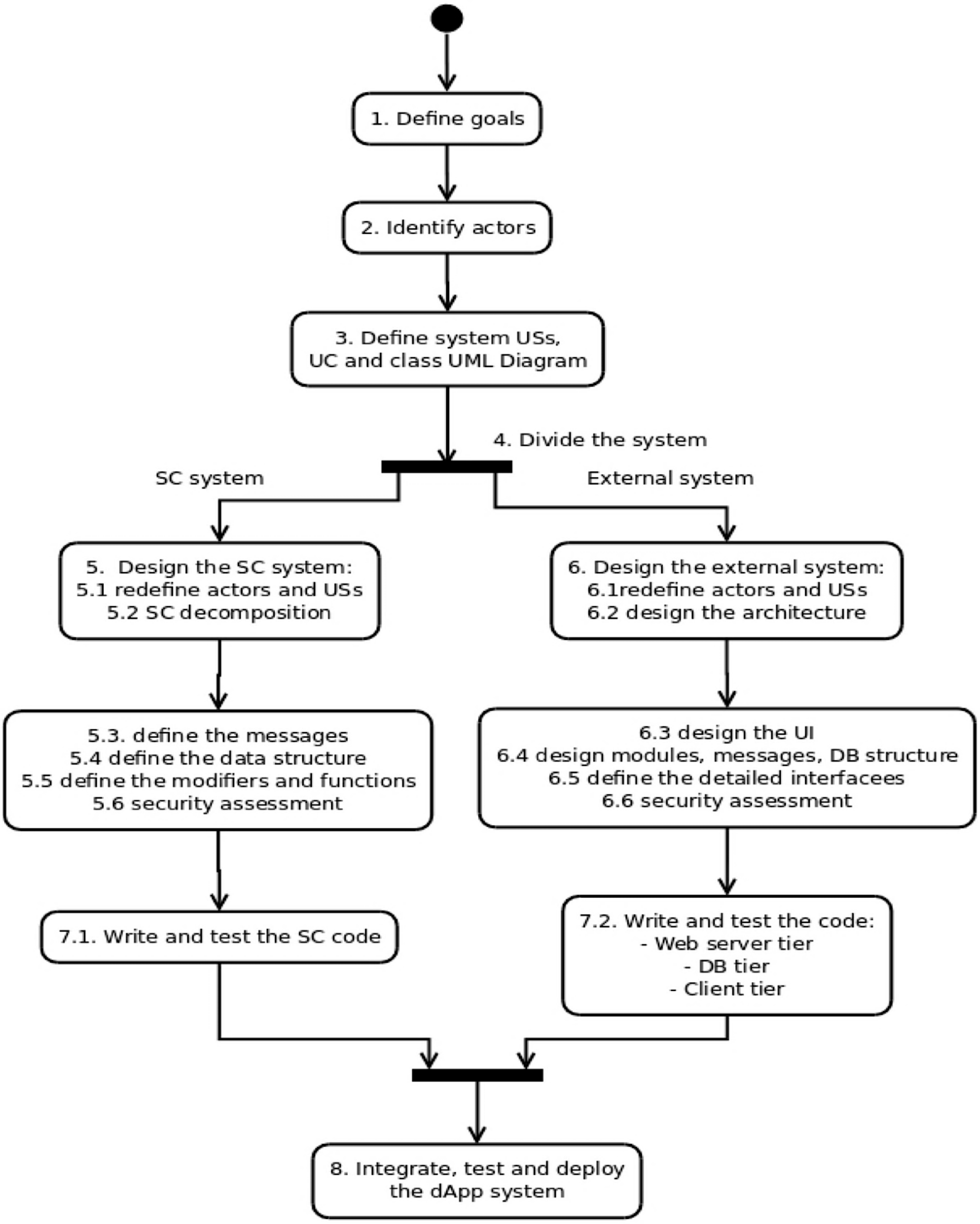


Figure 7: A proposed UML activity diagram. Source: (Marchesi, Marchesi and Tonelli, 2019)

The above diagram shows a proposed UML activity diagram illustrating several steps, processes in executing the proposed use case in developing the blockchain technology. These steps are broken into 9 phases undertaking in achieving this project which are:

Table 1. 2 Proposed steps to developing a decentralized application on a blockchain system

1. Define the goals	The purpose and aim of on what you plan to achieve from the blockchain system
2. Identify the actors	People involved in the process of development. from the project manager down to the clients.
3. Define the User stories;	User concept, class UML diagram, notations. The user's story or a personal on the case study. What modelling approach or method is used in designing the blockchains framework. e.g. petri net, class diagrams, sequence diagram, state, components, behavioural, BPMN, ArchiMate etc
4. Divide the system;	The system is divided into two sub sections • The smart contract on the blockchain • The App system that interacts with the blockchain
5. Design the Smart contract system;	Redefine the actors and User stories, Smart decomposition, define the messages, define the data structure, define the modifier and functions, security assessment.
6. Write and test the smart contract code;	Write and test the User stories, security assessment, write and run unit and acceptance test.
7. Design the App system;	Redefine actors and User stories, design the architecture, design the UI, design modules, messages, DB structure, define the detailed interfaces, security assessment.

8. Write and test the code;	Write and test the User stories, security assessment, write and run automated unit test (UTs) and acceptance test (ATs)
9. Integrate, test and deploy the DApp system	The integration of the smart contract and the App system with testing as load, correctness verification, operational, functional etc.

2.3.4 Summary

This section is divided into two parts, the earlier studies and theoretical framework. The first section starts with essential explanation of terminologies such as, blockchain technology, decentralized application, distributed systems, peer-2-peer, distributed ledger to help comprehend several concepts as related to this research. This section went further to offer a thorough description on the underlying concept of distributed systems, decentralization, P2P and the blockchain in the research. Use cases from peer-review literature are analysed to derive substantial data on framework and methods for developing a P2P application and a decentralised application on the blockchain. Comparison to show the underlying concept of the peer-2-peer application framework adoption to develop a DApp.

The second section emphases on the current practices in modelling, approach, designs and tools used in the development. Fundamental understanding on step and procedures for development are outlined. The next chapter introduces the utilized research methodology exploited in the course of this study.

3 Research Methodology

3.1 Introduction

This chapter emphasizes the selected design and comprehensive analysis on the parameters for this research study. This section is apportioned into six different parts. The first section is focused on providing a detailed evaluation of the use of a case study design technique which is been utilized in this research. The core aim is to understand the implementation of this design and the motivations of this research method. The second section presents an outline of the research questions designated for this paper in view to address the core objectives of this research with an extensive description of the case and justification for each question and how it can aid to actualize the objective goals. Finally, the third section expands on the collection of data processes utilized in this research paper followed the by two sections which describe the analysis and validity procedure explored with a later summary of the complete sections in full.

After an exact review of the outlined foundation and systematic progression of this research through a befitting literature overview, analysis and comparative study clearly stated, there is a need to undergo empirical research to close the existing gaps in refers to a systematic evaluation of several frameworks/methods for developing blockchain on a decentralized application. So, while supported by reviewed literature and hypotheses, it is still important to check and validate the above proposition. This part of the study deals with testing the proposition. The procedure begins by specifying the case as well as the research questions which motivate research study. It also describes the methodological approach selected to obtain data and how to interpret the data.

3.2 Case Study Design and Selection

From the start of this research, the major objective is to explore different techniques of building a decentralized application with strength and weakness to adopt or improve a new form and method of development with ease. Stating by literature review, the study denotes in clear facts of constant changes in the technology sphere, the gradual progression of businesses venturing into the adaptation of blockchain as a foundation for security, trust, transparency, availability and convivence. Due to the future possibilities that could be accomplished through the help of blockchain technology, it is of the essence to pinpoint the existing gaps on the methods/framework for developing a blockchain application and provide a framework that can help improve scalability and development processes. The major research question for this master thesis are engrafted from the existing gap.

How to identify key techniques for developing decentralized applications on the blockchain that ensures scalability, trade-offs and security?

To carry out a well-detailed interview which would adequately provide sufficient answers for to the major research, it is broken into 3 sub-questions which are been stated below:

- **SRQ1**: How to identify key criteria for classifying application development techniques of blockchain applications?

- **SRQ2**: How to identify key strengths and weaknesses of application development techniques for blockchain applications?

- **SRQ3**: How to improve software development approaches to ensure scalability, trade-offs and security of decentralized blockchain applications.

The sub questions are advanced into several questions for the sake of a constructive interview. Through this means of a comprehensive approach, the research objectives could attain a deeper study.

Sub research question one as stated below, help to recognize and obtain knowledge of major components of a decentralized application with analysing several methodologies of development and languages in use. The outcome will help us know and understand several easeful means of development and how time and resources could be used efficiently to achieve optimum functionality while providing a befitting result.

- What are the existing modelling techniques for deriving requirements of DApps?

- What are the existing software methods and frameworks for developing DApps?

- What are the existing programming languages that support the identified framework?

The second sub-question two is promptly compared different techniques of developing a blockchain application on various platforms. The outcome with help to identify strength and weakness in different techniques and methodologies. Combination of different techniques to form a better method of development as the case may be.

- What are the key properties for analysing software development techniques for blockchain applications?
- What are the merits of the identified software development techniques?
- What are the demerits of the identified software development techniques?

Lastly, the final sub-question which is the third go in-depth with the possibilities, impact and principles of trade-offs in the development of blockchain application. This question helps if scalability possibilities bring advancement in the development of an application.

- What are the principles of scalability, transparency and security design?
- What are the key characteristics of software development techniques that bolster scalability, transparency and security?
- What will be regarded as an effective balance between scalability, trade-offs and security?

3.3 Data Collection Procedure

Undergoing this research, the author had a conclusive end to engage the first-degree data collection practice. A semi-structured interview is utilized as the major source (primary data collection) and peer review literature articles (secondary data collection). This would be the evidence upon which this research will be subjected. Runeson et al., (2012) described this as a method whereby the researcher come into engagement with the outlined respondent(interviewee) in which real-time data is being extracted for analysis.

The validity of research carried out could be improved through the usage of different data sources servicing as multiple source of evidence to bound the clarification of a single data source (Runeson et al., 2012). The procedure of data collection could be described as the process whereby the materials and sources needed for research is gathered. Sapsford and Jupp, (2006) define "Data collection as the process of gathering the desirable information carefully, with least possible distortion, so that the analysis may provide answers that are credible and stand to logic." Through the help of data triangulation, a wider picture of the phenomenon studies is carried out and results acquired are reliable.

3.2.1 Interviews

During this process, the researcher also tries to validate the resources collected during the evaluation of several documents. The data collected are qualitative, and the sample will be designed to cover software development techniques. To deliver unbiased and accurate results, survey interviews would be initiated with actors from the governmental and private sector of engineering and development from different geographical region. The use of a semi-structured process for the interviews will be employed during this exploration as it is rightly appropriate for an exploratory research design.

A compilation of several open-ended questions is offered to the respondents as a means for them to better express their feelings regarding the phenomenon based on occurrences and knowledge. The questions handpicked for the studies are critically selected to derive answers to the research. The questions are designed to explore the research objectives at optimum. Being that it is a semi-structured interview, the questions or directions of the respondent's answers and proceeding questions from the researcher is not limiting to the interview questions alone. As described in the interview section above, the researcher has less control over the flow of the questions for validity reasons. A total number of fourteen (14) respondents agreed to an interview. The interviews are conducted either through skype or zoom and some in a face to face contact with the respondent. A tape recorder is placed to record all conversations carried out during the interview sessions and after which they were transcribed and coded with an RQDA.

According to Runeson et al., (2012), which expressed an interview process of extracting data as the most frequently used technique by people to gather resources. Considering Gill et al., (2008) which stated, "Interviews are also convenient to gather data when the research deals with more sensitive topics, about which the participants may not feel very comfortable to talk openly in a group environment.". It is in constant use because facts about most phenomenon researched can be richly extracted from individuals who are directly connected with the case study and cannot be assumed. This of great paramount to extract additional information from the respondent which the researcher might not be aware existed in the studies.

3.2.2 Document Review

During data collection, the practice of document review will be utilized. The practice could be exercised as the gathering of documents relevant to research such as managerial, business, legal and personal documents e.g. bulletins, program logs, applications, reports, financial statements, portfolios, letters, emails, journals, magazines etc. This process aids in gathering data collaboratively with background evidence from several sources (Yin, 2003). We dive deeper in the exploration of diverse documents in this thesis varying from articles to peer-reviewed journals to reports textbook, legal documents etc. are been utilized to find data on the profound topic and hypothetical background.

Major discussions in the sphere of software engineering, covering terminologies such as; decentralized applications, distributed systems, blockchain technology, scalability, methods, notations, security, samples of DApps built on different blockchain platforms with a systematic analysis of several methods of development.

3.4 Analysis of Procedure

This section explores the essence and utilization of data gathered for this study. We should bear fully in mind, that since we are using a case study research method, we embark on the usage of the qualitative analysis to get adequate and well-defined substantiation of the study. The data gathered around from several respondents will be examined utilizing an RQDA which is a qualitative open source program. According to Runeson et al., (2012) which expatiated on the core focus of this step as a method of having an enhanced knowledge on what and how the collected data is put into use. A thematic analysis is employed with specific categories implemented accompanied by transcribed responses from interviews in codes, labelling the ideas from the responses of the respondents that best explain the phenomenon examined and forging ahead into generalization depending on their components(Maguire, Maguire and Delahunt, 2017). According to Saldana, (2009) that enlightened several coding methods on which data collected could be studied. Based on our research we have decided to implore the use of three to four coding methods, which are the pattern, descriptive, viva and evaluation coding techniques.

RQDA is a free open source program which could run on several operating systems as compared to other CAQDAS (Computer aided qualitative data analysis software) program and meets most qualitative requirements and enables the analysis of textual data. Mathematical and statistical programs could be incorporated with the program for analysis. According to Maguire, Maguire & Delahunt, (2017) descriptive in carrying out an analysis based on the data collected, we decided to follow six steps to get a conclusive result. These steps offer strategies in fragments and outlining how information can be dissected utilizing the RQDA. They are: Become familiar with the data, generate initial codes, Search for themes, Review themes, Define themes, Write-up.

Firstly, this step has to do with reading and comprehending the transcribed data to get accustomed to the information. Jottings and records are made of initial exclamation and events on the case with repetitive statements and significant responses that are useful to the study (Maguire, Maguire and Delahunt, 2017). Then after the data collected will be structured in a consequential and efficient way that is the second procedure. Code extraction is utilized on a repetitive account which capture relevance to the research objectives. Thirdly, this part has to do with the choice of codes and themes use in labelling or identifying different answers important to the research e.g. Patterns, relevance, facts etc. The fourth and fifth stage covers investigation and scrutiny of the various theme on relevance to the study. Codes are then grouped according to defined themes. And lastly, a descriptive of the result gotten from the analysis will obtainable based on the structured information grouped.

3.5 Validity Procedure

Yin (2003) express validity as the length to which the trustworthiness of the outcomes derived are centred on the investigator's subjective opinion. According to Bordeianu and Morosan-Danila, (2013) "validity is the extent to which the instrument measures what it is intended to measure.". He went ahead in dividing validity into four different parts which are Construct, Content, Criteria and Factors validity. Our research will be based on Construct and Reliability.

Reliability Validity: This test simply guarantees the accuracy and dependability of the information gathered by the researcher. It endeavours to limit the blunders and predispositions in the analysis. The goal is to guarantee that if the case is directed once more by a subsequent scholar, following same techniques as illustrated by the previous researcher, the subsequent scholar ought to show up at similar discoveries at the end. To support this exactness and unwavering quality of this proposition, the rules in carrying out a case investigation-based research created by Yin (2003) was followed.

Construct Validity: This tests the adequacy of the functioning measures, that determine which data is to be collected and procedure of gathering to represent what the researcher and respondent are trying to describe, it attempts to build up a typical comprehension between the scientist and interviewees. This test guarantees that the targets of the exploration logically links with the study objectives and investigative questions.

3.6 Summary

This section was divided introductory part of a case study research, the study design and selection, data procedures which comprised of the interview style & document review, the data analysis procedure and finally, validity procedure. This part offers a thorough description of the nature of case study methodology, as it was the approach selected for the research. The major research question was sub-divided into three parts in other to properly answer the research objectives. A semi structured interview was conducted for 14 respondents in the field of software engineering. The source of our findings was depended on the semi-structured interview as the primary source of data collection and peer review literature articles the secondary source. Tools and means of conducting the interview were outlined. RQDA was the major tool for analysis which was convenient in generating a thematic analysis through the extraction of codes. The reliability and validity assessment of the data retention methods were also examined to match the required measure.

4 Results

4.1 Introduction

A detailed overview of the case selected and examined for this thesis is given in this section of the research. it also delves deep into qualitative analysis obtained through the RQDA program, and also a step further to provide some thorough explanation of our analytical findings.

4.2 Case Description

To comprehend the existing method/framework for the development of DApps, this research examines different steps and components during development. The blockchain in comprises blocks of encrypted information organised in chronological order and stored in a distributed fashion (Du *et al.*, 2019). Blockchain technology is still largely under-utilized with a lot more power to be explored.

The first wave of blockchains; collectively known as Blockchain 1.0 started off with Bitcoin in 2008 (Swan, 2015). Bitcoin and cryptocurrencies were the most prominent here with bitcoin being a major breakthrough in the world of digital currencies (Nakamoto, 2008). Blockchain 2.0 came in 2013 made availability the deployment of smart contracts, attainable agendas like collaborations of various entities and application into several industries (Perera *et al.*, 2020). Blockchain version 3.0 initiated the full operational functionality of DApps which support the deployment of backend code on the blockchain in a decentralised P2P network. This means user frontend code and interface on the DApps could virtually be written in any language making the structure flexible e.g. Ethereum (Perera *et al.*, 2020).

In a decentralized system, high performance, scalability, and security necessities cannot be met except the system is well designed. Software development techniques and tools along with research methods designed specifically to tackle the novel features of decentralized applications must be identified. Weak architectural design choices and outdated design methods could create loopholes in the system with serious consequences. Therefore, it is recommended that the development team follow best practices in software engineering, modelling, development, bug amputation and testing before the program is deployed into production. Meanwhile, blockchain technology advances each day and creative tooling is being created to better address development best practice and scaling. Scaling trade-offs have turned out to be a problem in the development of DApps. Latency, speed and cost of running DApps has limited its adoption.

Thus far, there is not widely standardized approach to modelling DApps that addresses all concerns. To provide the current practices used in the development of decentralised applications, we aim to perform a semi-structured interview to investigate the software development practices including design modelling review, methodologies, testing, verification, scalability, trade-offs and security coherent in the developing a DApp (Agbo, Mahmoud and Eklund, 2019).

4.3 Subject Description

To enhance our understanding of the case study mentioned earlier, in regards to information on the existing processes and challenges we ultimately seek to provide a clear path in developing a framework/method that will guide the implementation of a DApps on the blockchain application. This research intends to gather the viewpoints and perspective of few skilled respondents over a semi-structured interview.

A maximum of (14) fourteen respondents was contacted from different locations for an interview. Along the cause of our research, we found out that not all software developers, engineers, or specialists are blockchain inclined as we carefully made selections in picking our respondents. Most of the respondents had different involvements, in-depth knowledge and technical know-how of developing a DApp with vast experience in software engineering and development. At the moment most are fully specialized in the development of blockchain, smart contract engineering, DApp consultation/advisory. The concept over a broad selection of participants was to acquire from several software engineers and developers, a collective overview and ideas on existing method/framework as well as problems to provide a path to advance a better strategy.

4.4 Presentation of Findings

In this part of our studies, we would identify and interpret the results obtained by analysing data gathered on behalf of this study. Upon carrying out a semi-structured interview, the processes were divided into several sections for us to obtain the right analysis. Firstly, by collecting enough information from the respondent on their background on the subject matter for the sake of a proper validity regarding their answers and furtherly proceed to gather responses related to attaining the goal of the research on possible ways to improve the existing methods. The response to the interviews was transcribed into text to carry out an efficient analysis with the aid of the RQDA. As initiated in the previous chapter, the use of the six (6) phase guide while performing a thematic analysis which is, - getting familiar with transcript, generating initial codes, searching for themes, reviewing themes, defining and naming themes and producing a report were employed respectively. Code is extracted from the interview based on the research goal and categorised into themes according to the connection based on the evolving points. Below are the selected categorises for identifying the key findings to our research objectives:

- Current practices in the development of decentralised applications on a blockchain.
- Existing designs and modelling techniques in software development.
- Measuring correctness and security assessments.
- Concerns on scalability and trade-offs in DApps.

4.4.1 Current Practices in Development of Decentralised Applications on a Blockchain.

In the earlier phase of this research, observations and inferences about the current practices in developing a decentralized application on the blockchain were made from previous pieces of literature. Although those notions may be true, in this research they remain unproven and rather questionable. This concept of theme and coding category is drawn from respondents, comments, the accuracy of this argument and further information on existing programming standards as applied to the case studied.

The results support our statement that blockchain is a novel cutting-edge technology that is forecasted to cause a disruptive change in technology, business, organization, and collaborative spheres. Most respondents ascertained to the future of blockchain as believed to be indefinite since the full potential of the technology has not yet been unravelled. With new protocols, developments and probabilities emerging every day, the blockchain technology is evolving rapidly. There is no doubt that blockchain projects ought to follow certain standard in practices for swift adaptation in information system. Otherwise, they risk falling behind enterprise value.

DApps being the most explored functionality on the blockchain at the moment has erupted major concerns on methods and models of integration with other programs on several blockchain platforms. Most blockchain platforms utilised by our respondents range from permissioned (private) to permission-less (public) platforms such as Ethereum, Hyperledger, R3 Corda, IBM blockchain with programming languages such as Solidity, JavaScript, Golang, Kotlin, Python, Java etc. Most DApps are built on an Ethereum blockchain. 85% of our respondent worked and prefer the Ethereum platform due to ease of developing and/or integrating with previously deployed DApps. Ethereum does have shortcomings, mostly regarding scalability.

With tremendous years spent in software engineering, respondents attested to numerous methods utilised as development process are dynamic and not static but utilized in daily activity for R&D (research and development) under the roles they occupied during their years of development. 93% of our respondents attested to the fact of practising fully Agile methodology in terms of the development of DApps due to its flexibility processes. Of course, this process ranges from initiation, planning, execution, control to closure of the project. Two respondents obliged to utilising a mixture of development processes which fitted their process and they felt was more conducive. As stated,

"Yes, we do have a timeline. This is what we call sprint in Agile unlike the common 2 weeks which is allocated, where I present work, we have 6 weeks which is like a month and a half. it gives us enough time to do a lot of work and deliver. Like I said we do a lot of Agile, but we don't do agile fully, its rigid. We pick so good things out of agile, then some other flexibility forms other methodologies as well."

Two respondents pinpointed the effective of mixed processes such as Scrumban, a combination of Scrum and Kanban agile method. A respondent deliberated on agile as a framework being viable in the market due to speed of delivery. With agile workflows, functioning products can be showcased to stakeholders with short period of time. He stressed that with agile methodology, plans (which may not comprise the full scope) may frequently change. Also, procedures that lead to the end goal are less standardized. So, diagrams and flows could be thrashed in no time to adopt new insight. The respondent pointed out that a well-structured development plan would be better to get a good flow organised from the start to the end of the project like the waterfall framework. He points out:

"Having a good core plan of what is required is best, so development can be done with a blueprint. An Example is having a plan to build a house of 5 storey building and end up building a 10 storey".

One respondent stressed even as much as Agile is so lucrative in Information system development based on his years of experience, waterfall methodology has been capitalized in development in pharmaceutical, ports & waterways industries as plans have to be put out before processes commence which is rigid and has no soft landing for mistakes in development.

Table 1. 3: Agile framework strength and weakness in development.

Strength	Weakness
Fast and efficient	Lack necessary documentation
Flexibility	Project lacks definite plan
Less expensive to run	Could easily go off track
Quick response	More time spent on development

There is still more research to be carried out in R&D departments in several institutions to discover best practices in development and adoption DApp.

4.4.2 Existing Designs and Modelling Techniques in Software Development.

This category was fashioned to evaluate design and modelling techniques frequently used in the development of DApps. As consistent with other parts of this research, respondents were interviewed to get some knowledge with regards to this following their experience. The research question that correctly tied to these views were: How designs are documented in their practices? Preferred mark-up languages or modelling notations? Their shortcomings and strength. Also, in getting a holistic understanding of this aspect of the research, respondents' views weren't just limited to base on their practices but other which they are knowledgeable about.

Documentation of designs helps keep track of various aspects of software development and functionalities. Participants agree that documentation is of great importance in development. During observation, it came to notice most respondents had diverse means and programs used in the documentation of DApps during development such as GitHub, Jira, Trello, Structurizr etc. Two respondents claimed that designs were for physical representations and these were moulded with Adobe or wireframe and later move to Jira when the processes, split of task and documentation were rolled out during development. He highlighted:

"We use platforms like Jira, to break all the requirements into backlogs and assign different backlogs to different engineers and put the stories and epic in each of those backlogs. That forms the basis for the sprint that we have and if engineers want to pass on information among themselves, that will usually come up on our collaboration platform which is Jira."

To know what styles were most convenient in documenting designs, a question about frequently utilised mark-up languages or modelling notations used in development was asked. Most frequent tools used by our respondents in development of DApps were UMLs, C4, BPMN, ArchiMate, Microsoft Visio. Four respondents emphasised that so private blockchain platforms such as R3 Corda comprise a personalised design language for the development called Corda modelling notation that could be used to document designs and workflows. Five respondents spoke on not using a certain modelling notation or mark-up language during development as most flow charts were done either through Text with sketched out diagrams on different flow.

"For modelling notation recalling from the past firm we worked on the Corda platform, it has its prefixed modelling notation which was called Corda modelling notation. It was a customized UML to have features are similar to the platform. Where I work now, we don't do any standard modelling using UML."
"At the moment we use a lot of text and freehand sketch a few diagrams if they are needed."

A respondent highlighted the importance of text/free sketch over UML on several levels especially when it came to keeping stakeholders in line to comprehend what developers plan to execute. As expressed,

"For the notations for example if your stakeholders are not technically inclined. I could remember back at Interswitch I was the only one who understood the Corda mark language very well. The other didn't because they didn't understand UML if they understand they would have known this is the class, the representation of inheritance, the function, a method curve etc. if you aren't technical enough to know or understand UML, it will be difficult to read. But the text makes it understandable, easier and faster couple with a few diagrams attached to it. So, for me, I prefer the text."

85% of the respondents were unsatisfied relating their designs approach to the stakeholders. Most stakeholders without technical backgrounds find mark-up languages and modelling notations difficult to understand. Designs that can be easily understood by a wider audience range is considered strong. Most respondents employed the help of professional in explaining each step to different stakeholders. This process turns out to be time consuming educating people on objects represented in a chart, functionality and spin-off. Three respondents utilize power points and Adobe diagrams to simplify these procedures for the business side and other stakeholders who are not inclined technically. Five respondents commented about a modelling notation program called C4 which comprises a hierarchical set of software architectural diagrams for context, containers, components and code. Three respondents felt it was a better replacement for UML as it considers all stakeholders on different levels and self-explanatory. Another respondent commended BPMN. He stated, it was easier and understandable by both the business side and the technical side during development which lead to a collaboration with simplicity, but it lacked an in-depth view of architectural and component functionalities. Most respondents outlined the weakness of the UML, though still used as the standard mark-up language but has its weakness as to being old and not being able to express all a developer wishes to implement during modelling. A respondent expressed that UML are not to be thrown away as it is the standard language for modelling and could still be used in object-oriented programming such as in Hyperledger platform, although not sufficient.

"We have used UML and C4 and they tend to be generic which is their strength, but the weakness is that there are things that cannot be represented which was why we chose Corda."

Another respondent disagreed with the use of text as in comparison to others, the time and number of words which has to be written to convene a message. He stated:

"Freehand sketch can be successful but takes time and is not flexible in terms of updating and changes"

75% of the respondents stated they were not satisfied with their current mark-up or modelling notations due to several shortcomings and opted for object-oriented programming which could be frequently updated and has all functionalities to express different workflow views without problems from any stakeholders on understanding the concept for better collaboration. Most respondents attested to using a combination of tools to enable cover weakness of the others.

"So, we use the Corda modelling notation to document the design, with a combination of markdown (for completion). But for the actual design. We use the CDL (Corda design language)."

These feedbacks illustrate that respondents were fully aware of the issues facing the documentation of designs, mark-up and modelling notations in developing DApps and are interested in knowing which programming, or a tool that could be a solution to these issues. Some essential and far-reaching conclusions were met.

Table 1. 4: Strength and weakness of frequently utilised mark-up and notations for modelling based on respondent's view.

Components	UML. Unified mark-up language	BPMN. Business process management notation	C4. Context, Container, Comprises & Code	CORDA. Corda design language.	Text/Free Sketch/ Photo typing
Modelling flexibility	+	A	A	+	++
Modelling object-oriented systems	A	A	+	++	-
Models for the application of security	A	-	++	+	- -
Modelling for interaction to all stakeholders	+ (class, sequence diagrams somehow understood e.g. product owners)	+ BPMN show more strength & analytical on the business side.	+ More efficient in architectural designs (ADL) Bird-eye view for other components	+ Restricted to only R3 Corda platform, (class, sequences, objects	+ Understandable by all stakeholder, just overview, not precise, text too long to write.
Scalability	Doesn't address	Doesn't address	Doesn't address	+	Doesn't capture

A average, / - - much worse than average, /- worse than average, /+ better than average, /++ much better than average.

4.4.3 Measuring Correctness and Security Assessments.

This is one of the most important sections in this research as it has to deal with what the respondents think as an efficient way to verify the correctness and ensure security. Thus, the section provides an answer to the research question "What are the principles of transparency and security design?" "What are the key characteristics of software development techniques that bolster transparency and security?"

A close look at the results gotten from the interviews conducted indicated that various respondents had different practices in verifying the correctness of the DApps and security checks. The correctness of the DApps has to do with rigorous testing of functionalities before a viable product is rolled out for mass utilization. The correctness of a DApps has a lot to play in the transparency of the program, e.g., immutability and smart contract functionality.

"*At times when you go to production you see some errors; you have to make changes. Operationally we have to do a lot of manual testing. Functionally we do it unit by unit, using unit testing to know what work and doesn't and what need to be improved. So yes, we write lot of test*"

54% of our respondents classified verification of correctness into functionality correctness and operational correctness. According to all of our respondents, the correctness of DApps is simply matching the functionality to the requirements expected of the software. Most correctness is done by unit testing, integrated testing in the blockchain e.g. G-Unit.

"Functional testing authorises a product, service, meets to ensure it conforms to the actual requirement".

Table 1. 5: Various testing on the decentralized application

Functional Correctness	Functional testing authorizes a product service, meet to ensure it conforms to the actual requirement.	This test is executed before operational test. Manual testing or automation tools can be used for functional testing.
Operational Correctness. **e.g. Performance, Load, volume, security,**	Operational testing is a sort of non-functional approval test that authorises a product, service, process or system meets operational requirements. This test checks if a product, service, system or document reaches the operational necessities.	It is performed after functional testing Tools are more effective for this type of testing.

"The blockchain technology we use has an in-build testing system. For example, we can verify our smart-contract and in combination with other testing frameworks. For example, J-unit is a common testing framework and other testing frameworks"

Our respondent expressed operational correctness as simply checking the activities in the program corresponds as expected without errors. The respondent furtherly explained that a lot of manual tests are implemented during operational verification and are functionally implemented in unit by unit to ensure correctness. Most respondents felt sufficient about their testing procedures as they derived the actual result, they wanted and spoke on the huge importance of the process before rolling out. A respondent who acts as a Project Manager in Microsoft describe their workflow to ensure correctness with clients after testing as stated:

"All work items are laid out in the KANBAN board. And this KANBAN board stations out a column of how many work items are done and are featured in Trello, etc. Once it hits the state, or their claim resolved, we usually go back to the customers and make sure they have a copy. We do a fresh implementation at the customer site as a sign that is resolved or built to their satisfaction then we go back to the dev team and tell them to close the job."

Most respondents felt sufficient about their testing procedures as it produced the actual result they wanted and spoke on the huge importance of the process before rolling out. Security in DApps is to be taken seriously. After a good observation of the blockchain and components, you will agree it was built on the bases of security. Doing the interview, respondents commented about their dealings with security during development and how security is ensured at its best before deployment in the market.

40% Of the respondents attested to using standard security practices that are certified and tested in the past by engineers as a standard measure while developing a DApps. 90% employed the hands of certified cybersecurity companies to do a security audit their smart contract and application to search for loopholes. A respondent expressed that no system is completely secured as tactics used by attackers keep changing each day. 90 % of the respondents pay attackers to find bugs or loopholes into their system. Based on frequent attack threats two respondent highlighted they made constant security updates.

Table 1. 6: Different security measures in development of DApps.

Security Methods	Strength	Weakness
Bug bounty	Very effective. Covers up logical and application specific mistake	Expensive for small companies. Not trusted to evaluate the full system.
Security audit	Trust. Security auditing company smart contracts developed	Some small companies can't hire. Generalize in checks
Manual testing & testnet deploy	Cheap and convenient	Not sufficient and enough.
Automated testing / Unit testing	Reduces excesses issues for the testers, developers, auditors. Covers routine case of general software utility easily.	Not fully reliable for the safety of the full system. Depends on the correctness of the implemented software. Not good handling logical mistakes or flaws in business model
Formal practices e.g. Libraries, adhering to coding standards.	Already known and tested standards. Save time to implement	Most known Libraries hoard vulnerabilities which are known by attackers.
Formal Verifications. e.g. clean codes	Reduces excesses issues for the testers, developers, auditors	Not sufficient. Could be relied on for security practices

This section is directed towards discovering standard major practices observed by most software developers. As can be seen from the data gathered, different developers for availability and convenience use different techniques and procedure for verifying correctness, load assessment and security measures. Most of these measures, especially for security checks, e.g., bug bounty, external security audit is expensive and not affordable for small scale firms. Most end up relying on clean codes, formal practices of coding and manual testing. Security practices and assessments are identified as compulsory measures before the DApps is deployed, to ensure security, privacy-preservation and transparency in the blockchain against damages.

4.4.4 Concerns on Scalability and Trade-Offs in DApps.

Over the years, it has become evident that solutions to scalability problems of the Blockchain are far from accomplished. This section will attempt to analyse respondents' opinions on the issues of scalability implementation on the blockchain as well as their thoughts on how these subjects can be addressed. As already stated previously in the present state of scalability in the development of DApp, it is noted that there are lots of restricting issues with the current blockchain platforms and having a proper understanding of these issues and seeking respondent's idea on tackling them is an excellent way. During the interview, respondents commented on the scalability of the DApps and challenges encountered. Being that 95% of DApps are built on the Ethereum blockchain, respondents have raised issues of gas cost, low throughput capacity and slow confirmation to speed as a major issue.

Four respondents expressed that most blockchain platforms were built solely for the sake of decentralization and security. Two respondents highlighted the blockchain was not created with scaling in consideration but to tackle the current issues with security. 60% expressed Ethereum traded scalability for security. According to one of the respondents he highlighted:

"*The most important factor to think of is security and not scalability, which is why you have Ethereum and bitcoin product having scalability issues before they focus more on security and not scalability, which is something they would solve later."*

One respondent described the issue of capacity as a big problem as the blockchain is heavy with the number of transactions and has to store balances of account, contract, storage etc. A comparison was made by one of the respondents between transactions occurring within a centralized application against transactions occurring within the blockchain. Noting that the velocity of blockchain transactions is far lower. One respondent highlighted:

"Ethereum trades scalability for security that is why miners do proof of work before transactions could be mined into the blockchain. It's not a debate and already a given that the platform traded scalability for security. There are plans for a new deployment of the Ethereum blockchain"

Table 1. 7: Issues of scalability, trade-offs in the DApps

Scalability Issues	Speed	Computational cost
	Latency	Throughput

"That is why I talked about more encryption, hashing algorithm, which adds more security and slows down time request. Those are trade-off. And blockchain applications basically trades-off speed for security."

Speed: The blockchain is faced with issues of transactional functionality. Users have to get all their nodes verified and that decreases the speed on the platform. Proof-of-work is utilized in Ethereum and Bitcoin heavily and it negatively affects the scalability of the system and general throughput.

Computational cost: With Proof-of work, gigantic electricity consumption is required to solve complex mathematical computations before blocks are added to the Blockchain. For public blockchain to be scalable, security is at risk. 65% of the respondents acknowledge the issues of scalability as a bottleneck in public blockchains in contrast to private blockchains, which have a limited number of users and greater control over blockchain participants.

4.4.5 Summary

This chapter began with a brief description of the case selection and viewpoints for this study. In an attempt to provide an appropriate answer to the research questions chosen for the study, it offered a thorough review of the findings obtained from the interviews. Respondents commented on processes and several issues relating to the development of Dapps. Most responses were correlating to numerous challenges faced from the modelling techniques, to design, methodologies, testing, scalability and trade-offs. The research stressed a lot more on modelling techniques while leaving the respondents to comment on their approach and design most utilized in relation to all stakeholder's collaboration. 90% of respondents use the agile methodology for development of DApp. The agile methodology is seen an umbrella with several processes and combinations of approaches. Respondent answered all questions as they were presented in order to fulfil the research goal although leaving openings for some unanswered opinions. Also, we see groupings created from the specified research questions and afterwards classified into sub-sections to provide the questions under review with a well-explanatory answer. The explanation for this was to make sure each research issued were adequately answered.

5 Conclusion and Future Work

5.1 Introduction

This section gives a description based on the findings on the research and also some suggestions are taken from the literature reviewed and the results of the study. This also addresses the impact of results on the field of study and describes the research limitations and possible areas for future studies.

5.2 Summary of Findings and Recommendations

From inception, this study pursued a thorough understanding of existing practices in the development of decentralized blockchain applications. To consider the overall structure, dynamics and complexities encountered in development and afterwards proposal a possible way to improve the existing methods. Our research goals influenced the study approach through which questions were established that effectively formed the interviews questions on which respondents provided appropriate responses.
From the results outlined, the researcher put together a set of recommendations on addressing these issues. From these suggestions, we plan to propose a framework that will aid in the development of DApps on a blockchain-based system. The key focus of this study based on finding is to compare a few modelling framework and development and likewise offered a framework to tackle issues on scalability implementation of decentralised application on blockchain technology. This recommendation as follows:

- Standard Application and Integration
- Development Context.
- Security Assessment
- Stakeholders Collaborative Approach.
- Trade-offs Implementation
- Scalability Design

5.2.1 Standard Application and Integration

One of the main points to consider in having a functional and highly effective modelling design system is the combination and incorporation of different modelling designs and methods that are related to the blockchain-oriented systems (BOS). As earlier highlighted in this study, there are still existing gaps in identifying standards to modelling measures in the development of a DApps. Nevertheless, as the blockchain technology expands, it becomes more complex to model with old software or tools used previously. Tools for modelling designs such as UMLs, Flowcharts etc., are still fully in utilization but have shortcomings when it applies to BOS. Observing from interviews, different developers and firms use different approach when it comes to modelling designs and processes, depending on what is planned to achieve which implies to the fact that there are no means to approximate the questions on how sufficient a modelling is, varying form countless variables. For instance, many organizations still utilise BPMN in modelling business activities which is quite sufficient but exist certain restrictions to other stakeholders while modelling a BOS. Consequently, there are open source tools such as C4, etc., developed by programmers to close this gap in modelling designs. The R3 corda blockchain platforms developed a self-integrated mark-up language for the corda blockchain platforms which fits into their users experience e.g. Corda design mark-up language but stands impossible to extend to any other platform than the Corda blockchain. Modelling blockchain-based systems with new representation and models on a system that does not adequately reflect the conventional use case diagram, component diagram, state diagram, etc., could be challenging.

Currently, being that there is no specific notation available for designing or modelling. DApps may need a standardized notation for representation. The absence of a specified notation will make implementation or relocation complex and complicated, as the relationship between the blockchain and the application are not suitable.

My recommendation will be for various blockchain platforms to create personalised notations which could be extended to other platforms, also a combination of tools as expressed by the respondents could be used to bridge this gap in modelling design and approach. This would create the availability for openness, ease, synergy, transparency, accountability in the workflow process involving several stakeholders from different spheres.

5.2.2 Development Evaluation.

One of the emerging factors when it comes to techniques for developing a DApp has been the methodology or style used in development. There are numerous methodologies with diverse approaches and are useful in different projects for development. For reviewed literature and interviews we could finally conclude that Agile methodology of development is the most suitable in software development. Agile is proven to be different from other methodologies as it has a various process that could be utilize or combine to fit the desired approach, for instance, Scrumban[1], is a combination of Kanban and Scrum. Scrumban, is a mixture of two processes in the agile methodology. This is frequently practiced by developers compared to other methodology due to how efficient agile response to requests, time and changes.

Agile has proven to be efficient and sufficient in the development of Information systems. it has also shown weakness in the areas of documentation and a proper execution plan in development that is trade-off for speed and flexibility in production.

5.2.3 Security Assessment

Security measure is the most important aspect of the blockchain. Blockchain became popular and first considered for its security qualities before other features. Potential problems can arise with deployments of DApp and the use of blockchain nodes, when penetration tests are not properly initiated, thereby leaving vulnerabilities when deployed. A smart contract when deployed on the blockchain becomes impossible to change or modify. Discovery of vulnerabilities on the blockchain and smart contract after deployment can cause loopholes and bugs on the DApp framework. Bugs and loopholes of such cannot be modified or reverse but can only be rectified by hard forking which is harmful in DApp framework.

Being that the blockchain is immutable it is advisable to conduct a thorough code review before deployment for early recognition of bugs and avoidance to potential damages on the DApp.

---

[1] https://www.agilealliance.org/what-is-scrumban/

5.2.4 Stakeholders Collaborative Approach.

Collaboration with all parties involved in the development process is a critical part of implementing DApps. There is a concern to close the gaps between stakeholders especially the project sponsor, project manager and project team which comprises of the different people from various works of life in development, e.g., business unit, software engineers, advisory, legal unit etc., to have a common understanding and approach regarding development. Most appealing models for the business unit do not align with the prerequisite of the engineers and vice-versa. Most product owners or project sponsors get confused whenever engaged with technical terminologies and diagrams whereby requiring the assistance of professionals to explain this process which is time exhausting and frustrating.

Collaboration is essential for all stakeholders to have an agreement and understanding of all processes. These collaborations would involve a face-to-face meeting with stakeholders over the goal modelling aspects of the technology to ensure collaboration. According to findings, we believe a combination of approaches could save time and ease development processes, e.g., power points, text, wireframe etc. to be utilized for non-technically inclined personnel for faster comprehension. UML, C4 etc utilized for engineers, project managers, developers etc.

5.2.5 Trade-Offs Implementation.

With issues of compromise in the blockchain being eminent and major trade-offs are connected to the scalability, there have been so many successful cases of trade-off and there exist cases where security could be affected if authorized. For instance, formally the blockchain traded speed for security but at the moment there are Acyclic directed-graph technology that addresses this issue, e.g., IOTA[1], although has its own flaws. No firm is ready to compromise the security of the DApp especially for financial institutions for any additional feature such as privacy, speed, capacity etc. The major feature of the blockchain is security after which any other feature could be included.

[1] https://medium.com/@ercwl/iota-is-centralized-6289246e7b4d

Ethereum as a case study has issues with speed due to all users are verified but helps in security measures on the platform[1].

5.2.6 Scalability Design.

This section addresses the issues of scaling in development of DApps. Based on revealed literature review and findings, we categorically highlighted that most DApps are built on the Ethereum framework, which is faced with issues of scalability such as computational cost, latency, throughput. Scalability has been a bottleneck for Ethereum the same as Bitcoin. Nevertheless, there were suggestions by scholars on solutions to these flaws relating to speed, with reduction in the verification of nodes possibly advance the swiftness on the platform, lightning network transactions per seconds, the use of a different blockchain application for storage and capacity connected to the parent blockchain through an API. Compared to a traditional application, the blockchain still lacks behind in fundamental abilities to match a normal transaction. We can fully highlight as mentioned by the respondents that the blockchain was not built with scalability in consideration. Currently, there have been other frameworks built to compliment the Ethereum platform in the development of DApps which should be fully considered in development.

5.3 Impact/Implication

This study explored the existing design methods/frameworks for developing DApps by identifying strength, weakness, methodologies, challenges confronted in development process. Understanding these underlying points, the study addressed the current practices, steps, process and techniques during development a decentralized application on the blockchain. The outcome of this study is to propose a solution critical at solving issues in areas of modelling relating to BOS as considered in the development of blockchain application, collaboration of stakeholders to close the existing gaps in the areas of modelling designs and approach.

---

[1] https://arxiv.org/pdf/2004.12768.pdf

The issue of scalability and trade-off concerns were addressed under the current practices and possibilities of scaling. The study offers a feasible explanation aimed at providing current understanding of software development, modelling, design, techniques, approaches, and challenges in developing decentralised applications on the blockchain.

5.4 Limitations

The constraints of this study would be primarily related to the limitations of the reviewed case study included in this research. One key limitation was the question of generalizability, the author can never be sure if the conclusion is far reached on this case can indeed be applied towards a universal approach. According to Hyde et al., (2012), "the case study research design is meant to prove a theory and not to generalize findings to a population." The point considered by the author here is that the case study works well in making the phenomenon under study a theoretical generalization, but not sure if it could be universally allying.

A case study is said to lack the scientific rigour required to arrive at a vast reasonable deduction. It could be highlighted that a case study examiner could be inaccurate too many times and has allowed contradictory facts or skewed views to affect the course of the results and conclusions (Yin, 2003). Whereas this research attempted to extract the data needed from multiple sources to get there at such a conclusion, the limitation emphasized here cannot be overlooked.

Lastly, we should not forget that most respondents or interviews carried out were based on their personal experiences and personal impact derived from the phenomenon could be an outstanding limitation on the validity of the data collected.

5.5 Future Work

Some future research areas have been identified in the course of this research, and it's important to state that core research could be done to explore existing design methods and frameworks for developing information systems on decentralized and distributed networks with a holistic view to integrating a wider population and pragmatic approach on different framework/methods. It is strongly recommended, as it will aid towards a broader and diverse sample of respondent's visibility into the research.

This study also emphases on the impact of introducing a collective approach to the problems in the development of a DApp especially with methods and approaches. A related research will be to develop a possible framework for stakeholder's collaborative project planning to close the gap in development. Nevertheless, it is vital to reminisce that collaborative effort doesn't in any way guarantee success in execution.

Lastly, since one of the biggest issues in the development of DApps on the blockchain is scalability, it is therefore advisable to seek future mitigating term to tackle the issue properly for the advancement of the blockchain.

References

Agbo, C., Mahmoud, Q. and Eklund, J. (2019) 'Blockchain Technology in Healthcare: A Systematic Review', *Healthcare*, 7(2), p. 56. doi: 10.3390/healthcare7020056.

Andrian, H. R., Kurniawan, N. B. and Suhardi (2018) 'Blockchain Technology and Implementation : AA Systematic Literature Review', *2018 International Conference on Information Technology Systems and Innovation, ICITSI 2018 - Proceedings*. IEEE, pp. 370–374. doi: 10.1109/ICITSI.2018.8695939.

Atlam, H. F. *et al.* (2018) 'Blockchain with Internet of Things: Benefits, challenges, and future directions', *International Journal of Intelligent Systems and Applications*, 10(6), pp. 40–48. doi: 10.5815/ijisa.2018.06.05.

Atzori, M. (2017) 'BLOCKCHAIN TECHNOLOGY AND DECENTRALIZED GOVERNANCE: IS THE STATE STILL NECESSARY?', *Journal of Governance and Regulation*, 6(1), pp. 620–630. doi: 10.22495/jgr_v6_i1_p5.

Babaoglu, Ö., Meling, H. and Montresor, A. (2002) 'Anthill: A framework for the development of agent-based peer-to-peer systems', *Proceedings - International Conference on Distributed Computing Systems*. IEEE, pp. 15–22. doi: 10.1109/icdcs.2002.1022238.

'Banks lost N15billion to fraud, cyber-crime in 2018' | (no date). Available at: https://nairametrics.com/2019/08/02/banks-lost-n15billion-to-fraud-cyber-crime-in-2018/ (Accessed: 27 January 2020).

Bashir, I. (2017) *Mastering Blockchain: Deeper insights into decentralization, cryptography, Bitcoin, and popular Blockchain frameworks*, *Packt Publishing*.

Blockchain evolution: from 1.0 to 4.0 - Unibright.io - Medium (2017). Available at: https://medium.com/@UnibrightIO/blockchain-evolution-from-1-0-to-4-0-3fbdbccfc666 (Accessed: 1 April 2020).

Blockchain Version 1.0, 2.0, 3.0 And Future - Bitcoinik (2019). Available at: https://bitcoinik.com/blockchain-evolution-1-0-to-3-0/ (Accessed: 3 April 2020).

Bonneau, J. *et al.* (2015) 'SoK: Research perspectives and challenges for bitcoin and cryptocurrencies', in *Proceedings - IEEE Symposium on Security and Privacy*. IEEE, pp. 104–121. doi: 10.1109/SP.2015.14.

Bordeianu, O. and Morosan-danila, L. (2013) 'Development and Validation of Research Instruments for Cross-', (March 2016).

Brousmiche, K. L. *et al.* (2018) 'Digitizing, Securing and Sharing Vehicles Life-cycle over a Consortium Blockchain: Lessons Learned', in *2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, pp. 1–5. doi: 10.1109/NTMS.2018.8328733.

Bui, K. (2016) 'Nasdaq uses blockchain-based technology to reduce risk and prevent fraud', *Feedzai*. Available at: https://feedzai.com/blog/nasdaq-uses-blockchain-based-technology-to-reduce-risk-and-prevent-fraud/ (Accessed: 25 April 2020).

Buterin, V. (2018) 'Blockchain Resource Pricing', pp. 1–32. Available at: https://ethresear.ch/uploads/default/original/2X/1/197884012ada193318b67c4b777441e4a1830f49.pdf (Accessed: 25 April 2020).

Cai, W. *et al.* (2018) 'Decentralized Applications: The Blockchain-Empowered Software System', *IEEE Access*. IEEE, 6(September), pp. 53019–53033. doi:

10.1109/ACCESS.2018.2870644.

Califf, R. M. and Muhlbaier, L. H. (2003) ‘Health Insurance Portability and Accountability Act (HIPAA)’, *Circulation*, 108(8), pp. 915–918. doi: 10.1161/01.CIR.0000085720.65685.90.

Cameron Camp (2016) *BankThink Bitcoin May Help Criminals, But Blockchain Can Help Thwart Fraud*. Available at: https://www.americanbanker.com/opinion/bitcoin-may-help-criminals-but-blockchain-can-help-thwart-fraud (Accessed: 25 April 2020).

Casino, F., Dasaklis, T. K. and Patsakis, C. (2019) ‘A systematic literature review of blockchain-based applications: Current status, classification and open issues’, *Telematics and Informatics*. Elsevier, 36(May 2018), pp. 55–81. doi: 10.1016/j.tele.2018.11.006.

Chez, B. (2018) ‘Cryptocurrency Market Capitalizations | CoinMarketCap’, *Coinmarketcap*, 924, p. 20000000. Available at: https://coinmarketcap.com/ (Accessed: 25 January 2020).

Christidis, K. and Devetsikiotis, M. (2016) ‘Blockchains and Smart Contracts for the Internet of Things’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., pp. 2292–2303. doi: 10.1109/ACCESS.2016.2566339.

Coulouris, G. *et al.* (2011) *Distributed Systems: Concepts and Design*. 5th edn. USA: Addison-Wesley Publishing Company.

Dagher, G. G. *et al.* (2018) ‘Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology’, *Sustainable Cities and Society*. Elsevier, 39(August 2017), pp. 283–297. doi: 10.1016/j.scs.2018.02.014.

Dimitri, N. (2017) ‘Working Paper No . 2017 / 03 The Blockchain Technology : Some Theory and Applications’. Available at: https://www.msm.nl/resources/uploads/2017/10/Working-Paper-No.-2017-3.pdf.

Dinh, T. T. A. *et al.* (2018) ‘Untangling Blockchain: A Data Processing View of Blockchain Systems’, *IEEE Transactions on Knowledge and Data Engineering*, 30(7), pp. 1366–1385. doi: 10.1109/TKDE.2017.2781227.

Dooley, J. F. (2017) *Software Development, Design and Coding*, *Software Development, Design and Coding*. doi: 10.1007/978-1-4842-3153-1.

Du, W. (Derek) *et al.* (2019) ‘Affordances, experimentation and actualization of FinTech: A blockchain implementation study’, *Journal of Strategic Information Systems*. Elsevier, 28(1), pp. 50–65. doi: 10.1016/j.jsis.2018.10.002.

Dumas, M. *et al.* (2009) ‘From Business Process Models to Process-Oriented Software Systems’, *ACM Transactions on Software Engineering and Methodology*, 19(1), pp. 1–37. doi: 10.1145/1555392.1555395.

Elisa, N. *et al.* (2018) ‘A framework of blockchain-based secure and privacy-preserving E-government system’, *Wireless Networks*. Springer US, 0. doi: 10.1007/s11276-018-1883-0.

FINKELSTEIN, A. *et al.* (1992) ‘Viewpoints: a Framework for Integrating Multiple Perspectives in System Development’, *International Journal of Software Engineering and Knowledge Engineering*, 02(01), pp. 31–57. doi: 10.1142/s0218194092000038.

Gibb, R. (2019) *What is a distributed system ?*, *Computer Engineering*. Available at: https://blog.stackpath.com/distributed-system/ (Accessed: 25 March 2020).

Gil, A. M. and Mu, J. L. (2019) ‘Deploying Ethereum Infrastructure and Dapps’, pp. 2018–2019.

Gill, P. *et al.* (2008) 'Methods of data collection in qualitative research: Interviews and focus groups', *British Dental Journal*. Nature Publishing Group, 204(6), pp. 291–295. doi: 10.1038/bdj.2008.192.

Glaser, F. (2017). "Pervasive Decentralisation of Digital Infrastructures: A Framework for Blockchain enabled System and Use Case Analysis," in Proceedings of the 50th Hawaii International Conference on System Sciences, Waikoloa, Hawaii, USA, pp. 1543–1552.

Go-ethereum (2017) *go-ethereum/README.md at master · ethereum/go-ethereum · GitHub*. Available at: https://github.com/ethereum/go-ethereum/blob/master/README.md (Accessed: 11 April 2020).

Gruber, R. T. (1993) *KNOWLEDGE S A Translation Approach to Portable Ontology Specifications A Translation Approach to Portable Ontology Specifications*.
Grubic, T. and Fan, I. S. (2010) 'Supply chain ontology: Review, analysis and synthesis', *Computers in Industry*, 61(8), pp. 776–786. doi: 10.1016/j.compind.2010.05.006.

Gu, J. *et al.* (2018) 'Consortium blockchain-based malware detection in mobile devices', *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 6, pp. 12118–12128. doi: 10.1109/ACCESS.2018.2805783.

Ismail and Materwala (2019) 'Article
 A Review of Blockchain Architecture and Consensus
 Protocols: Use Cases, Challenges, and Solutions', *Symmetry*, 11(10), p. 1198. doi: 10.3390/sym11101198.

Joseph D. Gradecki (2002) *Mastering JXTA: Building Java Peer-to-Peer Applications - Joseph D. Gradecki - Google Books*. Available at: https://books.google.ee/books?id=KtnIL3v3TNgC&pg=PA467&lpg=PA467&dq=www.jxta.Org&source=bl&ots=tpqbdV0D_m&sig=ACfU3U3NGjRHz5FosDFGXfPWHGoFuIYEvQ&hl=en&sa=X&ved=2ahUKEwjP0Mz6293oAhVMxaYKHaUKBxcQ6AEwAXoECAsQKA#v=onepage&q=www.jxta.Org&f=false (Accessed: 10 April 2020).

jpmorgan (2018) *quorum-docs/Quorum Whitepaper v0.1.pdf at master · jpmorganchase/quorum-docs · GitHub*. Available at: https://github.com/jpmorganchase/quorum-docs/blob/master/Quorum Whitepaper v0.1.pdf (Accessed: 11 April 2020).

Jun, M. S. (2018) 'Blockchain government - A next form of infrastructure for the twenty-first century', *Journal of Open Innovation: Technology, Market, and Complexity*. Journal of Open Innovation: Technology, Market, and Complexity, 4(1). doi: 10.1186/s40852-018-0086-3.

Kim, H. M. and Laskowski, M. (2018) 'Toward an ontology-driven blockchain design for supply-chain provenance', *Intelligent Systems in Accounting, Finance and Management*, 25(1), pp. 18–27. doi: 10.1002/isaf.1424.

Kormiltsyn, A., Thangalimodzi, K. and Norta, A. (2018) 'An Exploration of Blockchain enabled Smart-Contracts Application in the Enterprise'. doi: 10.13140/RG.2.2.36464.97287.

Koteska, B., Karafiloski, E. and Mishev, A. (2017) 'Blockchain implementation quality challenges: A literature review', *CEUR Workshop Proceedings*, 1938(September).

Kozlovski, S. (2018) *A Thorough Introduction to Distributed Systems*, *freeCodeCamp*. Available at: https://www.freecodecamp.org/news/a-thorough-introduction-to-distributed-systems-3b91562c9b3c/ (Accessed: 29 March 2020).

Langer, A. M. (2016) *Guide to Software Development*, *Guide to Software Development*.

London: Springer London. doi: 10.1007/978-1-4471-6799-0.

Lauslahti, K., Mattila, J., and Seppälä, T. (2016). 'Smart Contracts–How will Blockchain Technology Affect Contractual Practices?' The Research Institute of the Finnish Economy. Legner,

Lauslahti, K., Mattila, J., and Seppälä, T. (2016) 'Smart Contracts–How will Blockchain Technology Affect Contractual Practices?' The Research Institute of the Finnish Economy.

LeewayHertz (2020) *A Complete Guide to Blockchain Development Process | LeewayHertz*. Available at: https://www.leewayhertz.com/guide-to-blockchain-development-process/ (Accessed: 10 April 2020).

Lewis, R., McPartland, J. W. and Ranjan, R. (2017) 'Blockchain and Financial Market Innovation - Federal Reserve Bank of Chicago', *Economic Perspectives*, 41(7). Available at: https://www.chicagofed.org/publications/economic-perspectives/2017/7#ftn5.

Li, J. (2008) 'On peer-to-peer (P2P) content delivery', *Peer-to-Peer Networking and Applications*. Springer Science and Business Media LLC, 1(1), pp. 45–63. doi: 10.1007/s12083-007-0003-1.

Lin, I. C. and Liao, T. C. (2017) 'A survey of blockchain security issues and challenges', *International Journal of Network Security*, 19(5), pp. 653–659. doi: 10.6633/IJNS.201709.19(5).01.

Litke, A., Anagnostopoulos, D. and Varvarigou, T. (2019) 'Blockchains for Supply Chain Management: Architectural Elements and Challenges Towards a Global Scale Deployment', *Logistics*, 3(1), p. 5. doi: 10.3390/logistics3010005.

Lu, Y. (2019) 'The blockchain: State-of-the-art and research challenges', *Journal of Industrial Information Integration*, 15(January), pp. 80–90. doi: 10.1016/j.jii.2019.04.002.

Luu, L. *et al.* (2016) 'Making smart contracts smarter', in *Proceedings of the ACM Conference on Computer and Communications Security*. New York, New York, USA: Association for Computing Machinery, pp. 254–269. doi: 10.1145/2976749.2978309.

Maguire, M., Maguire, M. and Delahunt, B. (2017) 'Doing a thematic analysis: A practical, step-by-step guide for learning and teaching scholars.', *AISHE-J - The All Ireland Journal of Teaching and Learning in Higher Education*, 9(3). Available at: https://ojs.aishe.org/aishe/index.php/aishe-j/article/view/335 (Accessed: 11 April 2020).

Marchesi, L., Marchesi, M. and Tonelli, R. (2019) 'ABCDE -- Agile Block Chain Dapp Engineering'. Available at: http://arxiv.org/abs/1912.09074.

Martinovic, I. (2018) 'Blockchains: Design Principles, Applications, and Case Studies', 7(7). Available at: http://www.egov.ee/media/1374/martinovic-blockchains-design-principles-applications-and-case-studies.pdf%0Ahttp://egov.ee/media/1374/martinovic-blockchains-design-principles-applications-and-case-studies.pdf.

Mendling, J. and Weidlich, M. (2012) *Business Process Model and Notation*, *Lecture Notes in Business Information Processing*. Edited by J. Mendling and M. Weidlich. Berlin, Heidelberg: Springer Berlin Heidelberg (Lecture Notes in Business Information Processing). doi: 10.1007/978-3-642-33155-8.

Merriam-Webster (2016) *Provenance | Definition of Provenance by Merriam-Webster*. Available at: https://www.merriam-webster.com/dictionary/provenance (Accessed: 11 April 2020).

Nakamoto, S. (2008) ‘Bitcoin: A Peer-to-Peer Electronic Cash System’, *SSRN Electronic Journal*. doi: 10.2139/ssrn.3440802.

Narayanan, A. *et al.* (2016) *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. USA: Princeton University Press. Available at: https://bitcoinbook.cs.princeton.edu/.

Natarajan, H., Krause, S. and Gradstein, H. (2017) *Distributed Ledger Technology and Blockchain*, *Distributed Ledger Technology and Blockchain*. World Bank. doi: 10.1596/29053.

Ølnes, S., Ubacht, J. and Janssen, M. (2017) ‘Blockchain in government: Benefits and implications of distributed ledger technology for information sharing’, *Government Information Quarterly*, 34(3), pp. 355–364. doi: 10.1016/j.giq.2017.09.007.

Oram, A. (2001) *Peer-to-peer : harnessing the benefits of a disruptive technology*.
Peer-to-peer - Wikipedia (no date). Available at: https://en.wikipedia.org/wiki/Peer-to-peer (Accessed: 27 March 2020).

Overview of CDL (no date) Overview of CDL - Corda Solutions Site October 2018 documentation. Available at: *https://solutions.corda.net/corda-modelling-notation/overview/overview-overview.html.*

Perera, S. *et al.* (2020) ‘Blockchain Technology: Is it Hype or Real in the Construction Industry?’, *Journal of Industrial Information Integration*. Elsevier, 17(June 2019), p. 100125. doi: 10.1016/j.jii.2020.100125.

Ramachandran, G. S. and Krishnamachari, B. (2018) ‘Blockchain for the IoT: Opportunities and Challenges’. Available at: http://arxiv.org/abs/1805.02818.

Raval, S. (2016) ‘Decentralized Applications Harnessing Bitcoin’s Blockchain Technology’, in Tim McGovern, Lobner, C., and Fraleigh, J. (eds) *O’Reilly Media*. 1 edition. 1005 Gravenstein Highway North, Sebastopol, CA 95472.: O’Reilly Media, p. 118. Available at: https://www.oreilly.com/catalog/errata.csp?isbn=9781491924549.

Risius, M., and Spohrer, K. (2017). “A Blockchain Research Framework,” Business & Information Systems Engineering (59:6), pp. 385–409.

Rocha, H. and Ducasse, S. (2018) ‘Preliminary steps towards modeling blockchain oriented software’, *Proceedings - International Conference on Software Engineering*, pp. 52–57. doi: 10.1145/3194113.3194123.

Rowstron, A. and Druschel, P. (2001) ‘Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Verlag, pp. 329–350. doi: 10.1007/3-540-45518-3_18.

RUNESON, P. *et al.* (2012) *CASE STUDY RESEARCH IN SOFTWARE ENGINEERING.* Available at: http://www.worldcat.org/title/case-study-research-in-software-engineering-guidelines-and-examples/oclc/828789615&referer=brief_results.

Saldana, J. (2009) ‘Summary for Policymakers’, in Intergovernmental Panel on Climate Change (ed.) *Climate Change 2013 - The Physical Science Basis*. Cambridge: Cambridge University Press, pp. 1–30. doi: 10.1017/CBO9781107415324.004.

Samaria, A. (2016) *Improving Enterprise Business Process Management Systems: Electronic Health Records (#EHR) implementation with #blockchain, #BPM, #ECM and #platform*. Available at: http://improving-bpm-systems.blogspot.com/2016/07/electronic-health-records-ehr.html (Accessed: 11 April 2020).

Sankar, L. S., Sindhu, M. and Sethumadhavan, M. (2017) 'Survey of consensus protocols on blockchain applications', in *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE, pp. 1–5. doi: 10.1109/ICACCS.2017.8014672.
Sapsford, R. and Jupp, V. (2006) *Data Collection and Analysis*. 1 Oliver's Yard, 55 City Road, London England EC1Y 1SP United Kingdom: SAGE Publications Ltd. doi: 10.4135/9781849208802.

Saveen A. Abeyratne (2016) 'Blockchain Ready Manufacturing Supply Chain Using Distributed Ledger', *International Journal of Research in Engineering and Technology*, 05(09), pp. 1–10. doi: 10.15623/ijret.2016.0509001.

Scherer, M. (2017) 'Performance and Scalability of Blockchain Networks and Smart Contracts', p. 46. Available at: http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1111497&dswid=8567.

Scott, K. *et al.* (1999) *UML Distilled Second Edition A Brief Guide to the Standard Object Modeling Language Martin Fowler UML Distilled Second Edition A Brief Guide to the Standard Object Modeling Language UML Distilled Second Edition A Brief Guide to the Standard Object Modelli*, *Software Design*. Available at: www.awl.com/cseng/ (Accessed: 12 April 2020).

Steen, M. and Tanenbaum, A. S. (2016) 'A brief introduction to distributed systems', *Computing*. Springer Vienna, 98(10), pp. 967–1009. doi: 10.1007/s00607-016-0508-7.

Sun, J., Yan, J. and Zhang, K. Z. K. (2016) 'Blockchain-based sharing services: What blockchain technology can contribute to smart cities', *Financial Innovation*. SpringerOpen, 2(1), p. 26. doi: 10.1186/s40854-016-0040-y.

Swan, M. (2015) *Blockchain: Blueprint for a New Economy - Melanie Swan - Google Books*. Available at: https://books.google.ee/books?hl=en&lr=&id=RHJmBgAAQBAJ&oi=fnd&pg=PR3&ots=XQyBI02Oj4&sig=1IjNsDSRMLREAqGuyiQJl78A2WQ&redir_esc=y#v=onepage&q&f=false (Accessed: 15 February 2020).

Truu, A., Buldas, A. and Laanoja, R. (2017) 'Keyless signature infrastructure and PKI: hash-tree signatures in pre- and post-quantum world', *International Journal of Services Technology and Management*, 23(1/2), p. 117. doi: 10.1504/IJSTM.2017.10002708.

Tschorsch, F. and Scheuermann, B. (2016) 'Bitcoin and beyond: A technical survey on decentralized digital currencies', *IEEE Communications Surveys and Tutorials*. Institute of Electrical and Electronics Engineers Inc., 18(3), pp. 2084–2123. doi: 10.1109/COMST.2016.2535718.

Vitso, M. *et al.* (2017) 'A study on blockchain technology as a resource for competitive advantage', (June), p. 90. doi: 10.1177/0956797610371344.

Wang, H. *et al.* (2018) 'Blockchain challenges and opportunities: a survey', *International Journal of Web and Grid Services*, 14(4), p. 352. doi: 10.1504/IJWGS.2018.10016848.
Wang, T., Wu, X. and He, T. (2019) 'Trustable and Automated Machine Learning Running with

Blockchain and Its Applications’. Available at: http://arxiv.org/abs/1908.05725.

Xu, J. J. (2016) ‘Are blockchains immune to all malicious attacks?’, *Financial Innovation*. Financial Innovation, 2(1). doi: 10.1186/s40854-016-0046-5.

Yin, R. K. (2003) ‘Case Study Methods: Design and methods’, *Sage publications*. Available at: https://books.google.ee/books?id=BWea_9ZGQMwC&printsec=frontcover#v=onepage&q&f=false (Accessed: 11 April 2020).

Yli-Huumo, J. *et al.* (2016) ‘Where Is Current Research on Blockchain Technology?—A Systematic Review’, *PLOS ONE*. Edited by H. Song. Public Library of Science, 11(10), p. e0163477. doi: 10.1371/journal.pone.0163477.

Zheng, Z. *et al.* (2017) ‘An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends’, in *2017 IEEE International Congress on Big Data (BigData Congress)*. IEEE, pp. 557–564. doi: 10.1109/BigDataCongress.2017.85.

Appendix 1 – Interview Questions

Section 1 - Segmentation Questions.

1. What is your role in your organization?
2. How long have you been in your position?
3. Classify your software development experience? Beginner, advanced, senior developer.
4. How long have you been involved in the development of blockchain apps?
5. How would you rate your knowledge of DApps development? Novice () Average () Experienced ()
6. What are your major responsibilities?

Section 2 – Questions on modelling techniques and challenges during development processes

7. Can you give a brief explanation of how you document your designs? any mark-up languages or modelling notations in this organization?
8. Are the project stakeholders able to understand the requirements outlined in the modelling notations?
9. What programming language(s) do you use in developing DApps?
10. In your opinion, do you think an analysis of several DApps development processes will lead to an improved standard for developing? Yes () No ()
11. If Yes, please kindly explain how; If No, please give reasons
12. What do you think might be a major challenge in the development of DApps?
13. How do you think these challenges can be addressed?
14. Can the outcome of the DApp development processes be predicted and evaluated from the knowledge of the system's functionality?
15. Is the modelling notation you currently using sufficient?
16. If No, what are some shortcomings? what are its strengths?

Section 3 – Questions about Methodologies and approach of development.

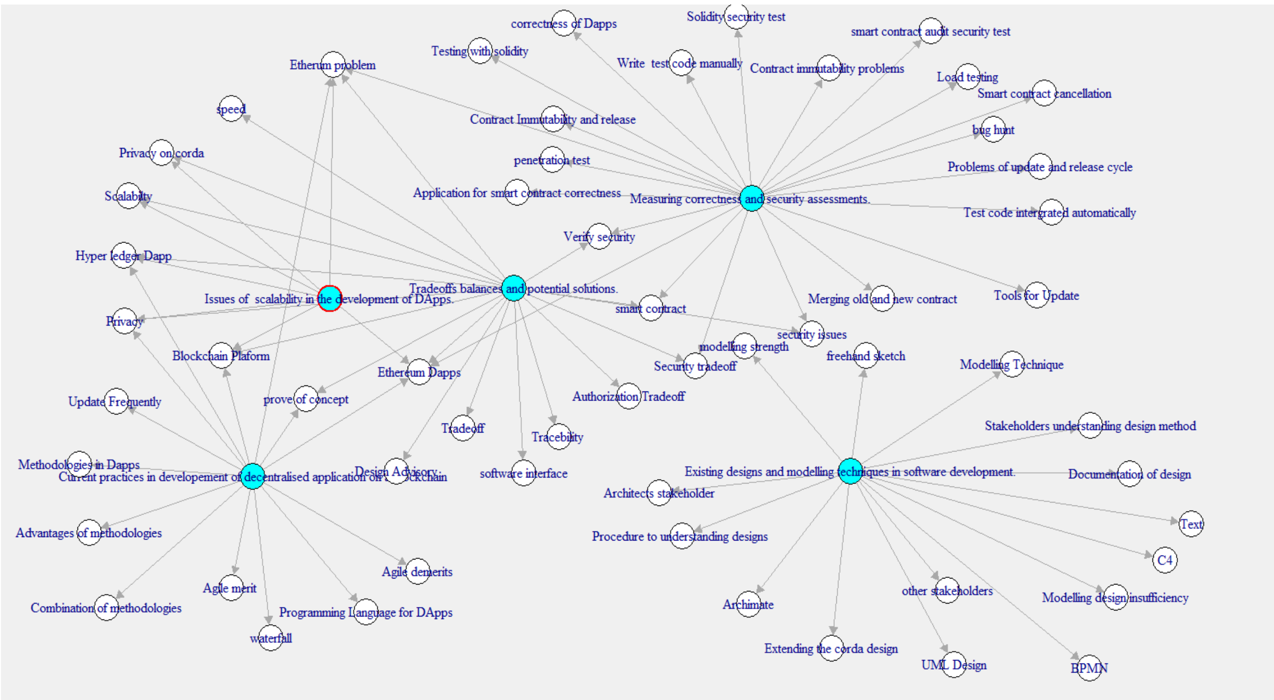
17. what development methodologies did you most frequently use before you got involved in DApps?
18. Are those methodologies still relevant to DApps?

19. If YES, what parts of those methodologies are most relevant while building DApps?
20. What are their strengths? and relevance?
21. Could this methodology be substituted by any, based on practice?
22. If YES, which?
23. An effective block-chained based process will take into account a clear definition of the working techniques.
24. Would you say that there is a detailed process or technique of how the blockchain system is the developer?
25. Do you have a role in ensuring that the steps or processes to achieving this system are clearly understood?
26. If YES, how do you contribute to that?
27. Would you say that your current development technique produces the desired outcome of the overall DApp necessity?

Section 4 – Questions about the Correctness, Security, Load And Scalability.

28. If Yes, how do you verify the correctness of your DApps?
29. How do you verify the security of DApps?
30. Do you frequently make updates in response to user requests? Yes () No ()
31. If Yes, How often?
32. Please can you give a brief explanation of measures considered before deliberating updates?
33. Are they aspects of the current system that you think should be improved on? Yes () No ()
34. If Yes, please give a brief description of these aspects
35. Is there a relationship between contract immutability and release cycles?
36. If Yes, please explain?
37. Do you perform load tests on your DApps? Yes () No ()
38. If Yes, how?
39. Have you done any trade-off security to Scalability?
40. What are the issues of scalability?
41. In your opinion, do you think the introduction of blockchain technology will lead to an improvement in several aspects of software development processes?
42. Yes () If Yes, please explain how, If No, why?

Appendix 2 Thematic Map of all Categories and Codes.



Appendix 3 Interview Recordings

Find the link to the interviews recorded for this thesis below:

https://1drv.ms/u/s!Al8UGhPBhicZuR8ZbWQLeCBxzxFU?e=7jdq5d

Appendix 4 Interview Transcript

Find the transcripts to interviews conducted by clicking on the link below:
https://1drv.ms/u/s!Al8UGhPBhicZuSLO8WTiYpF7ZeW6?e=0Msz14