Blockchains And Distributed Apps

# Glossary

* Dapp- Decentralised Applications
* DAO - Decentralized Autonomous Organizations
* EEA - Enterprise Ethereum Alliance
* UML - Unified Modelling Language
* ICO - Initial Coin Offerings

# Table of Contents

1. Chapter I: Introduction……………………………………………………………………………………………….Page 3
   1. What is a blockchain?. ………………………………………………………………………………………..Page 3
   2. How were blockchains Created? ………………………………………….……………………………..Page 3
   3. The Structure of a blockchain………………………………………………………………………………Page 4
   4. The main outlines of the project…………………………………………………………….……………Page 4
2. Chapter II: Methodology………………………………………………………………………………………..…..Page 5
   1. Methodology Used………………………………………………………………………………………….…..Page 5
   2. Tools used……………………………………………………………………………………………………………Page 5
   3. Risk Analysis………………………………………………………………………………………..………………Page 6
      1. Common Project Risks………………………………………………………………………………Page 6
      2. Understanding Risks…………………………………………………………………………….……Page 6
      3. Identifying Threats…………………………………………………………………………………….Page 6
      4. Assessing Risks……………………………………………………………………………….…………Page 7
      5. Managing Risks………………………………………………………………………………….……..Page 7
3. Chapter III: Research and Requirements……………………………………………………………….……Page 8
   1. Research Strategy………………………………………………………………………………………………………...Page 8
   2. Research Method………………………………………………………………………………………….……………...Page 8
   3. Quantitative Research vs. Qualitative Research…………………………………..………………Page 9
   4. Research approach……………………………………………………………………………………………Page 11
   5. The methods of data collection…………………………………………………………………………Page 11
   6. User Stories…………………………………………………………………………….…………………………Page 11
   7. Previous Case Studies………………………………………………………………………………..………Page 12
      1. The Cryptocurrency case study…………………………………………………….…………Page 12
      2. Smart Contracts case study………………………………….…………………………………Page 12
4. Chapter IV: Design and Results of the project……………………………………………..……………Page 14
   1. Introduction………………………………………………………………………………………………………Page 14
   2. Software architecture……………………………………………………………………………………….Page 14
      1. Architectural representation…………………………………………..…………………..…Page 14
      2. Architectural Goals and Constraints……………………………………………………….Page 15
      3. Use case view…………………………………………………………………………………..…….Page 15
5. Chapter V: Conclusion and next steps………………………………………………………………………Page 17
6. References………………………………………………………………………………………….…………………..Page 18

Chapter I: Introduction

This section has the purpose of outlining the main points of this report and provide the background of the subject that this project has been based on, the subject being blockchain technology and distributed apps.

## What is a blockchain?

A blockchain is a list of records that is constantly growing, called blocks and are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data.

The design of a blockchain is made to block any modification to the inputted data. It is "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way"(Iansiti, Marco; Lakhani, Karim R., January 2017).

For use as a distributed ledger, a blockchain is typically managed by a peer-to-peer network collectively adhering to a protocol for inter-node communication and validating new blocks. Once recorded, the data in any given block cannot be altered retroactively without alteration of all subsequent blocks, which requires consensus of the network majority.

## 1.2 How Blockchains were created

The first conceptualised blockchain has been created by an anonymous person or a group of people that goes by the name of Satoshi Nakamoto, with its core component being the use of a Hashcash-like method to add blocks to the chain without requiring them to be signed by a trusted party. The main purpose of this blockchain is to keep track of all the transaction made within the network when using the cryptocurrency bitcoin.

After the appearance of the Bitcoin platform, Vitalik Buterin, tried to popularize the technology in the early 2012. He also was one of the founders of the Etherium platform, with the vision of creating a developer community where they can create distributed apps (Dapps) that do not have only a Bitcoin currency as a core component. He referred to this concept as ‘smart contracts’ or even blockchain-based “[decentralized autonomous organizations](http://bitcoinmagazine.com/7050/bootstrapping-a-decentralized-autonomous-corporation-part-i/)” (DAOs).

Ethereum succeeded in gathering a strong developer community, enterprise support via the Enterprise Ethereum Alliance (EEA) and a establishing a true ecosystem in an extremely short time.

## 1.3 The structure of a blockchain

At its core, a blockchain is a decentralized, distributed and public digital ledger that is able to keep track and record transactions across multiple computers so that the data would not be altered without the alteration of all subsequent blocks. A blockchain database is managed autonomously using a peer-to-peer network and a distributed timestamping server.

The components that are essential for a functional blockchain are:

* Block: This component is the list of the blockchain that keeps track of all the transactions within the system. Some blockchains do not have data security as a main priority, but all of them must keep track of any action that happened in the platform.
* Chain: The link that connects each block using hashes. The hash in blockchain is created from the data that was in the previous block. The hash is a fingerprint of this data and locks blocks in order and time.
* Network: The network is composed of “full nodes.” In other words, it is an algorithm that is run by a computer in order to achieve a secure network. Each node contains a complete record of all the actions that were ever recorded in that blockchain.

## 1.4 The main outlines of the project.

The purpose of this project is the creation of a prototype of a fully working project that uses blockchain technology in order to create a peer-to-peer(P2P) connection between two terminals, in this case being two command prompts, in order to play a game of rock-paper-scissors. The idea behind this blockchain is that of a game, playing with other users in a local area network (LAN), meaning that this blockchain is meant for entertainment purposes. In its first stages, the project was supposed to have many fore features such as the ability to search for multiple prompts and pick the terminal that the user wanted to, but in the ended up being too ambitious and most of it was scrapped.

The main objectives of the program are:

* To make a blockchain that connects to a network
* To be able to connect to another user
* To make the game feature playable

Chapter II: Methodology

As it is indicated in the title, this chapter includes the methodology used for the dissertation. In more details, this part outlines the methodology used, the tools and the risk analysis.

## 2.1 Methodology Used

In order to satisfy the objectives of the dissertation, the Agile practises were used. Agile is a discipline intended for project management and software development to help teams submit their deliverables faster to the clients without having any problems. If used correctly, it will make the programmers and the code itself more flexible.

Using Agile will lead to code that has little to no issues and having an efficient and compact design that will work with all the intended features. This methodology is also aiming to reduce risks by minimizing the "leap-of-faith" that is needed before any evidence of value can be obtained.

The core agile software programming practices are the following:

• Test-first programming (or perhaps Test-Driven Development)

• Rigorous, regular refactoring

• Continuous integration

• Simple design

• Pair programming

• Sharing the codebase between all or most programmers

• A single coding standard to which all programmers adhere

• A common “war-room” style work area

## 2.2 Tools Used

The tools that was used for this project was [Github](https://github.com/) [**Github Project Management**](https://github.com/).

Github is an online repository that is recording edits across an entire project in real time. It also integrates many other tools so multiple people–from your developers to the product owner–who can work in the same code at the same time making it truly a great tool for development teams.

Github Project Management is an Agile tool that is used for task management and helps you manage the workload of the project that you are currently developing.

## 2.3 Risk analysis

Risk management is a process that allows individual risks events and overall risk to be understood and managed proactively, optimising success by minimising threats and maximising opportunities (APM, 2014).

2.3.1 Common Project Risks

All projects are inherently risky as they are unique, constrained, based on assumptions, performed by people and subject to external influences (APM, 2014). Risk is managed in an organisation to protect the primary goal and assets of the firm, which are vital elements in business survival. Understanding specific risks to a project allows the management team to protect the activities from said risks. Charitable and not-for-profit organisations have extremely limited resources; therefore, risk can never be reduced to zero. Consequently, understanding risk and its magnitude allow the prioritisation of scarce resources.

The common project risks are:

* Cost – Often in the sense of spending more than originally budgeted for due too poor cost estimation and scope creep.
* Schedule – Project overrun and activities taking longer than expected. Slippages in schedule also increase costs and delay the receipt of project benefits
* Performance – This is the risk that the project will fail to produce the intended results originally specified

2.3.2 Understanding Risks

Risk is managed in an organisation to protect the primary goal and assets of the firm, which are vital elements in business survival. Understanding specific risks to a project allows the management team to protect the activities from said risks. Charitable and not-for-profit organisations have extremely limited resources; therefore, risk can never be reduced to zero. Consequently, understanding risk and its magnitude allow the prioritisation of scarce resources.

2.3.3 Identifying Threats

To assess threats accurately, the assessment must include the threat source. Common threat sources include natural, human, or environmental threats. Individual who understand the organisation, industry or type of project are key in recognising them; threats are commonly identified through experience of management or project members. It is valuable to document potential threats while in the preliminary stages of a project and update the document while project activities are ongoing.

2.3.4 Assessing Risks

Risk is assessed by identifying threats and vulnerabilities. The possibility and impact of each risk is also defined. Risk assessment is extremely complex and is usually based on imperfect information and assumptions are made, often on past experiences or industry knowledge. There are many methodologies aimed at risk assessment and in this project, a RAID Log will be used to document such risks.

2.3.5 Managing Risks

After identifying and assessing a risk, the next logical step is to plan how it will be dealt with. A risk management plan is crucial in ensuring project success and being prepared will reduce the impact or surprise of the risk.

There are four main ways to manage risk:

* Avoidance – Decision not to take on the risk by avoiding the actions that cause the risk.
* Reduction – Taking mitigating actions that reduce the risk probability, impact, or a combination of both.
* Transfer / Sharing – Transferring some or part of the risk to a third party or distributing the risk to multiple organisations or individuals.
* Acceptance – Deciding to face the risk. The risk needs to be budgeted for and included in the project resource planning.

.

Chapter III: Research and Requirements

As it is indicated in the title, this chapter includes the research and requirements of the dissertation. In more details, this section will outline the research strategy, the research method, the research approach, the methods of data collection, the selection of the sample, the research process, the type of data analysis, use cases in order to frame the requirements, user stories, previous case studies and the technology selection.

# 3.1 Research Strategy

The research that was used in this dissertation is nothing that any expert in this field would not know. Rather, numerous pieces of previous research can be found regarding the existence of blockchains and the process of them being created and also what types of blockchains are used by both individuals and organisations alike. As such, the proposed research took the form of a new research but on an existing research subject.

# 3.2 Research Method

The method that was used the most in this project was analysing the research data about blockchains from different sources, in order to have an understanding about what exactly is a blockchain, what platforms are they based on, what programming languages are available for the development of a blockchain and how to make your own blockchain.

Another method that was used in order to satisfy the objectives of the dissertation was to use the empirical research method. Empiricism is a philosophical theory that states that true knowledge can only arise by systematically observing the world. The empirical research has certain advantages. The advantages can be put forth in the following manner:

* It helps in authentication of traditional research by carrying out experiments and observing,
* It improves understanding and reverting back to dynamic situations by developing analytical and observation skills, and
* It assists in understanding the ways in which the difference in contexts results in difference in the outcome of research.

However, the empirical research methodology also has disadvantages. The disadvantages are as follows:

* Time consuming: It is time consuming because one has to collected data from different firms for carrying out the research.
* Cost: It is based on field research, which means a researcher has to visit different firms where they may need to stay that will involve expenses related to travelling, food and lodging, and other expenses.
* Admittance to firms: it is not always possible to get entry to the firms necessary for collecting data to carry out the research.
* Collecting data: Data collection is a very difficult task, as it is field research wherein the researcher sometimes does not get access to all firms and even if they get entry to the firm, they may not be able to get the correct data because the firm may be reluctant in providing it.

# 3.3 Quantitative Research vs. Qualitative Research

The type of empirical research that was used for this dissertation was the quantitative one. The main characteristic of quantitative research is that it is based on objective facts, statistics and numerical data. One major advantage that the quantitative research has over the qualitative research is that the outcome of quantitative research is easy to measure, and the results can be clearly shown through objective data. One drawback that this type of research has is that it cannot be used in non-numerical information, such as human emotions, beliefs or imaginations. The quantitative research may not be suitable for subjects such as psychology or sociology but due to the nature of this project, this is the best choice between the two types of research

# Features of Qualitative & Quantitative Research

|  |  |
| --- | --- |
| **Qualitative** | **Quantitative** |
| The aim is a complete, detailed description. | The aim is to classify features, count them, and construct statistical models in an attempt to explain what is observed. |
| Researcher may only know roughly in advance what he/she is looking for. | Researcher knows clearly in advance what he/she is looking for. |
| Recommended during earlier phases of research projects. | Recommended during latter phases of research projects. |
| The design emerges as the study unfolds. | All aspects of the study are carefully designed before data is collected. |
| Researcher is the data gathering instrument. | Researcher uses tools, such as questionnaires or equipment to collect numerical data. |
| Data is in the form of words, pictures or objects. | Data is in the form of numbers and statistics. |
| Subjective – individuals’ interpretation of events is important, e.g., uses participant observation, in-depth interviews etc. | Objective: seeks precise measurement & analysis of target concepts, e.g., uses surveys, questionnaires etc. |
| Qualitative data is more 'rich', time consuming, and less able to be generalized. | Quantitative data is more efficient, able to test hypotheses, but may miss contextual detail. |
| Researcher tends to become subjectively immersed in the subject matter. | Researcher tends to remain objectively separated from the subject matter. |

Adapted from: Miles & Huberman (1994, p. 40). Qualitative Data Analysis, available at http://wilderdom.com/research/QualitativeVersusQuantitativeResearch.html

# 3.4 Research Approach

The research approach that was followed for the purposes of this research was the deductive one. According to this approach, researchers begin with formulating a set of hypotheses at the start of the research. Then, relevant research methods are chosen and applied to test the hypotheses to prove them right or wrong. However, the main weakness of the deductive approach is that it relies heavily upon the initial hypotheses being correct. If one or more hypotheses are incorrect, the argument is invalid and necessarily unsound.

# 3.5 The methods of data collection

For collecting the data required for the primary research, information regarding blockchains and Dapps has been gathered from multiple websites that were knowledgeable about these subjects such as [Etherium](https://www.ethereum.org/). Based on the information that has been found, user stories can be made in order to create use cases, requirements analysis and being able to choose the technology that is best suited for the project.

Blockchain technology has been developing at an incredibly fast rate. Alongside the rate of technological development comes adoption from individuals and companies, the growing number of blockchain jobs, government pilots and the proliferation of Initial Coin Offerings (ICOs).

# 3.6 User Stories

User stories are part of the Agile methodology with the purpose of delivering the highest value of a product by focusing on small and immediate customer needs. All agile user stories include a written sentence or two and, more importantly, a series of conversations about the desired functionality.

User stories usually follow a simple structure like the following template:

As a < type of user >, I want < some goal > so that < some reason >.

Some of the user stories that can be used for the development of this project are the following:

* As a client, I want to be able to connect to the blockchain network so that I can play the specified game.
* As a client, I want my connection to be secure so that my personal data would not be in danger.
* As a client, I want the application’s interface to be simple in order to have a better user experience.
* As an admin, I want the system to have low maintenance so that I can update the application with ease
* As an admin, I want the system to have maximum automation in order to better maintain the application.

# 3.7 Previous Case Studies

There are a range of different ways blockchain technologies can be used to generate business benefits, such as improved visibility and near real-time reporting. Some applications are built around the synchronicity of the ledger and its ability to simplify reconciliation, while others are focused on removing middleman from systems, reducing cost and bias.

# 3.7.1 The Cryptocurrency case study

This case study may be one of the most common case studies related to blockchains. The prime example for this case study is the Bitcoin platform. Bitcoin is a cash-related cryptocurrency that has been announced in the late 2008 and launched in early 2009. The purpose of Bitcoin was to use electronic cash that could be sent peer-to-peer without the need for a central bank

Bitcoin is attractive to users for several reasons:

* payer-borne transaction costs are low;
* the valuation of the currency has generally been growing strongly since its creation
* the system is much less restricted than traditional banking.

Another benefit in using bitcoin is that there are no regulations between international borders, meaning that transfer between territories is no different from any other payment.

# 3.7.2 Smart Contracts case study

Smart contracts allow for transactions to be made automatically and without the need to rely on a central party to adjudicate the operation of the contract terms. Blockchain offers opportunities in this arena, because smart contract code can be written directly onto a block and is examinable by the contracting parties ahead of time, just like a traditional legal contract.

Smart contracts help you exchange money, property, shares, or anything of value in a transparent, conflict-free way while avoiding the services of a middleman.

One of the advantages that smart contracts provide is the fact that they can reduce counterparty risk. Traditional contracts can be most of the time ambiguous and they can also provide loopholes that are beneficial for only one party. A smart contract due to the fact that it is a Dapp, it is just a set of computer codes, with pre-set conditions, which would be executed on the present conditions. The contract will carry out the one and only meaning of its code.

However, smart contracts in the present day are more of an ideal rather than something that can be implemented. In terms of legal contracts, courts would have to recognise that the operations of smart contracts are legitimate ways to transfer ownership and value between parties, and that the terms of smart contracts are enforceable in case a breach somehow does occur.

Another issue would be that the contract can be overridden if it is badly designed. An example of this issue would be hen the DAO (a smart contract-driven investment vehicle created for the Ethereum blockchain) had much of its funding hijacked through a loophole in a poorly-written smart contract.

Chapter IV: Design

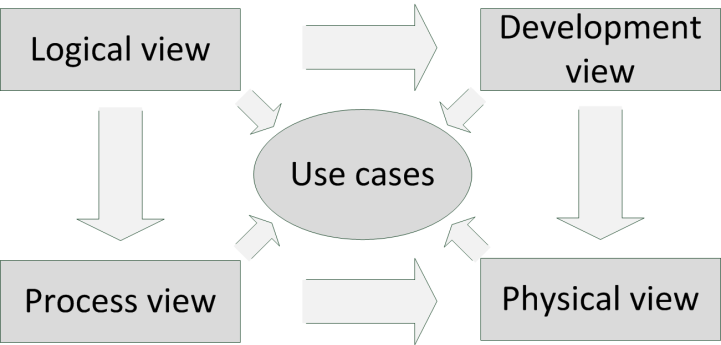
# 4.1 Introduction

As stated in the chapter title, this section will provide the design of the blockchain application for this project and what resulted from it. In more details, this section will outline the software architecture, the usability, functionality, reliability, portability, efficiency and maintainability of the software.

# 4.2 Software architecture

The purpose of this software is to establish a P2P connection between two terminals in order to play a game of rock-paper-scissors.

In order to depict the software as accurately as possible, the structure of this document is based on Philippe Kruchten’s “4+1” model view of architecture [Kruchten].



# 4.2.1 Architectural representation

This section details the architecture using the views defined in the “4+1” model [Kruchten]. The views to document the system are:

* Use case view
  + Audience: all the end users
  + Area: describes the set of scenarios and/or use cases that represent some significant, central functionality of the system. Describes the actors and use cases for the system, this view presents the needs of the user and is elaborated further at the design level to describe discrete flows and constraints in more detail
  + Related artifacts: Use-Case Model
* Logical view
  + Audience: Designers
  + Area: Functional Requirements: describes the design's object model
* Data view
  + Audience: Data specialists, Administrators
  + Area: describes the architecturally significant persistent elements in the data model as well as how data flows through the system.
* Deployment view
  + Audience: Deployment managers
  + Area: Describes potential deployment structures, by including known and anticipated deployment scenarios

# 4.2.2 Architectural Goals and Constraints

There are some key requirements and system constraints that have a significant bearing on the architecture. They are:

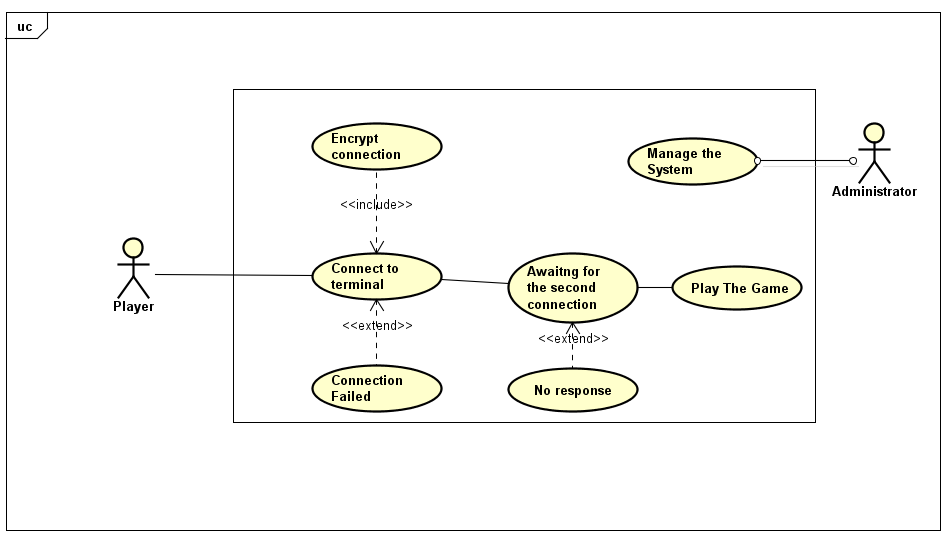
* The system is meant as a proof of concept for a more complete project prediction system to be built in the future
* The system will be written using GO programming language and will also be hosted on [LibP2P](https://github.com/libp2p/go-libp2p).
* Both user’s terminals are up and running
* Both users are connected to each other
* Both users can interact with each other
* The system will run in command prompts, therefore, no other interface will be implemented

# 4.2.3 Use Case view

The purpose of a use case is to describe a function a user will perform using the target system. A user can be a human user or another system that is connected to the target system. To make our lives easier we call a human user and those other systems actors. Use case diagrams are considered for high level requirement analysis of a system. When the requirements of a system are analysed, the functionalities are captured in use cases (www.tutorialspoint.com, 2019).

* Actors
  + User: The user will drive all operation of the software. No distinction is made in regards to type of user.
  + Administrator: The administrator will proceed with maintaining the system

The following use case outlines the requirements for the project:



Chapter V: Conclusion and next steps

Like the title says above, this chapter has the purpose of concluding the dissertation, having a summary of the contents of this dissertation

The main purpose of this dissertation was to create a software based on blockchains and decentralised apps. To achieve this objective, research on this subject has been made while also identifying the risks that come with the development and also choosing the suitable methodologies in terms of project management and in terms of conducting the research.

This dissertation outlines gives a proper background explanation of what blockchains and distributive apps are while also explaining how they are created and where are they originating from. The variety, popularity and usability of blockchains has also increased up until the present day. However, due to the fact that blockchains are relatively a new concept, they still present many flaws in terms of security as seen in section 3.7.2 and also in terms on energy consumption.

Overall, despite the risks and current limitations of blockchains and decentralised apps, many developers are still creating new content in the community, the proof being that Etherium, one of the platforms that hosts decentralised apps, is currently hosting over two thousand apps and hosting dozens of new Dapps each month.

# References

En.wikipedia.org. (2019). *Blockchain*. [online] Available at: https://en.wikipedia.org/wiki/Blockchain [Accessed 10 Apr. 2019].

Hacker Noon. (2019). *A brief history in the evolution of blockchain technology platforms*. [online] Available at: https://hackernoon.com/a-brief-history-in-the-evolution-of-blockchain-technology-platforms-1bb2bad8960a [Accessed 10 Apr. 2019].

dummies. (2019). *The Structure of Blockchains - dummies*. [online] Available at: https://www.dummies.com/personal-finance/the-structure-of-blockchains/ [Accessed 10 Apr. 2019].

APM (2014) Introduction to Risk Management. *APM* [online]. Available from: <https://www.apm.org.uk/body-of-knowledge/delivery/risk-management/>.

Ukdissertations.in. (2019). *Burdens and benefits of Empirical Research | UK Dissertations*. [online] Available at: http://www.ukdissertations.in/blog/burdens-and-benefits-of-empirical-research/ [Accessed 11 Apr. 2019].

Explorable.com. (2019). Deductive Reasoning. [online] Available at: https://explorable.com/deductive-reasoning [Accessed 11 Apr. 2019].

Study.com. (2019). [online] Available at: https://study.com/academy/answer/what-is-quantitative-research-what-are-the-advantages-and-disadvantages-of-quantitative-research.html [Accessed 11 Apr. 2019].

Softwaretestinghelp.com. (2019). Types of Risks in Software Projects. [online] Available at: https://www.softwaretestinghelp.com/types-of-risks-in-software-projects/ [Accessed 11 Apr. 2019].

Wilderdom.com. (2019). Qualitative versus Quantitative Research: Key points in a classic debate. [online] Available at: http://wilderdom.com/research/QualitativeVersusQuantitativeResearch.html [Accessed 11 Apr. 2019].

Getzephyr.com. (2019). How to identify and manage software testing risks | Zephyr. [online] Available at: https://www.getzephyr.com/insights/how-identify-and-manage-software-testing-risks [Accessed 11 Apr. 2019].

Cohn, M. (2019). User Stories and User Story Examples by Mike Cohn. [online] Mountain Goat Software. Available at: https://www.mountaingoatsoftware.com/agile/user-stories [Accessed 11 Apr. 2019].

Agile For Growth. (2019). 6 Benefits of Good User Stories - Agile For Growth. [online] Available at: http://agileforgrowth.com/blog/userstory-benefits/ [Accessed 11 Apr. 2019].

www.tutorialspoint.com. (2019). UML - Use Case Diagrams. [online] Available at: https://www.tutorialspoint.com/uml/uml\_use\_case\_diagram.htm [Accessed 11 Apr. 2019].

Icaew.com. (2019). Blockchain case studies. [online] Available at: https://www.icaew.com/technical/technology/blockchain/blockchain-articles/blockchain-case-studies [Accessed 11 Apr. 2019].

Utm.mx. (2019). Guideline: Outline the Architecture. [online] Available at: http://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/guidelines/outline\_the\_architecture\_35F1530A.html [Accessed 11 Apr. 2019].