Enemies or Friends? Can GDPR and Blockchain technology be compatible regarding data privacy and protection?

UNIVERSITY OF
LINCOLN

Thomas Anderson

19701195@students.lincoln.ac.uk

School of Computer Science

College of Science

University of Lincoln

Submitted in partial fulfilment of the requirements for the

Degree of BSc(Hons) Computer Science

Supervisor: Yvonne James

May 2022

Contents

1. Acknowledgements

Firstly, I want to thank my wife Molly, for having supported me through all the ups and downs of this journey of university. You always believed in me, pushed me when I thought I couldn't go on and never gave up on me even when I had. As a mature student this degree was extremely hard for me to get back into education and I would not have completed without your support.

I also want to thank my father-in-law Steve, his fiancé Melissa, and their son Freddie for giving me and my wife a place to live whilst I was at university. I know it hasn't always been easy, but it has made my university life so much easier, to go to classes and the library being 20 minutes away has been a real help.

I want to thank my supervisor Yvonne James for the support she has provided throughout this project and her enthusiasm she has for students. Through lectures and this project, whenever I needed help you always made yourself available and was easily approachable which is a big deal when you find approaching people hard and for that I will be forever grateful.

And finally I want to thank my employers at Currys Repair Centre for allowing me to pursue my studies whilst reducing my working hours. Without a job, university would have been harder and more stressful for worrying about money and I doubt I would have been able to focus on my studies as much with that in mind.

2. Abstract

This research work aims to investigate Blockchain technology and GDPR compliance regarding data privacy. It will analyse the data privacy perspective with respect to distributed ledger technology (DLT). Blockchain has become one of the most frequently discussed technologies for its ability to allow for peer-to-peer transactions without a centralized intermediary. The GDPR was implemented in May 2018 for EU member states to maintain data privacy. DLT is the underlying technology of blockchain and is a decentralized system without any monetary authority. This research has been conducted through a thorough literature review on prior conducted research to investigate the problems and determine the disparities of GDPR compliance with blockchain technologies and discuss the technical solution that will allow blockchain to become more compliant regarding GDPR in terms of privacy.

3. Introduction

Due to data transparency individuals are becoming aware of the threats of data breaches and the use of their personal data for commercial purposes. The General Data Protection Regulation (GDPR) are data protection laws across the European Union and aims to give back the control of one's personal data. GDPR was drafted based on a world in which centralised and identifiable actors control personal data. Blockchain works completely differently. It aims to move power over personal data away from centralised entities by processing it in a decentralised environment. However, the decentralised nature of blockchain technology is not the only factor that causes both legal and compliance challenges. The near immutability of transactions potentially affects the rights of data subjects. (Kaurlartz et al, 2019) With GDPR, organizations with or without a physical market presence in the EU will still be required to comply with the GDPR. However, due to the lack of consensus as to how controllership should be defined, the allocation of responsibility and accountability is hindered. GDPR assumes that data can be modified or erased where necessary to comply with legal requirements, such as Articles 16 and 17 of the GDPR. (European Union, 2016). Blockchains, however, render the modification of data purposefully difficult to ensure data integrity. There are many literature studies summarizing blockchain research such

as “A Summary of Research on Blockchain in the Field of Intellectual Property” (Wang. J, et al, 2019), and its applications “A Critical Review of Blockchain and its Current Applications” (Tama. B.A, et al, 2017), however currently there is minimal content researching the integration of GDPR and blockchain. As this remains a critical legal and practical question for the adoption of blockchain technology in the European Union, the purpose of this project is to conduct a literature review on the topic and answer the question, Can GDPR and Blockchain technology be compatible regarding data privacy and protection?

This report will provide a background on the subject matter. It will provide a thorough in-depth literature review and its methodology. It will discuss the methodology of software development and the software that has been developed itself. It will then end with a conclusion and an evaluation of how the project went and what has been learned from the project.

4. Literature Review

4.1 Background

Blockchain is a distributed database that is shared among the nodes of a computer network. As a database, a blockchain stores information electronically in digital format. The goal of blockchain is to allow digital information to be recorded and distributed, but not edited (Hayes, 2022). In this way, a blockchain is the foundation for immutable ledgers, or records of transactions that cannot be altered, deleted, or destroyed. This is why blockchains are also known as a distributed ledger technology (DLT). What a blockchain does is to allow the data held in a database to be spread out among several network nodes at various locations. This not only creates redundancy but also maintains the reliability of the data stored. So if somebody tries to alter a record at one instance of the database, the other nodes would not be altered and thus would prevent a bad actor from doing so. If one user tampers with Bitcoin’s record of transactions, all other nodes would cross-reference each other and easily pinpoint the node with the incorrect information. This system helps to establish an exact and transparent order of events. This way, no single node within the network can alter information held within it. Due to this, the information and history (such as of transactions of cryptocurrency) are irreversible. A blockchain can also hold a variety of other information like legal contracts, state identifications, or a company’s product inventory. (Hayes, 2022)

GDPR is designed for the EU citizens to control their data and information. With the rapid growth of digital technologies and an increasing digitalization rate in all sectors like business, healthcare, and education, both governments and citizens are concerned about privacy issues. GDPR focuses on personal data in general and this regulation helps any users to understand how their data is being used. Additionally, the regulations guide businesses on how to manage and process personal data. Any sort of data that can identify the data subject is termed as personal data. It can be the name, phone number or email address tied to any identifiable information (Moubry et al, 2018, 222-233). A digital service needs to be compliant with GDPR recommendations, and the service provider should have responsibilities as a controller, collector, and processor (Almeida Teixeira, 2019). A GDPR compliant digital service must have data minimization, fairness, accuracy, transparency, and confidentiality (Hoofnagle, 2019).

Having discussed the characteristics of blockchains and recommendations of the GDPR, there is a need to harmonize between the two to establish common grounds for developing GDPR compliant blockchains.

4.2 Related Literature

To accomplish the task, an in-depth literature review has been made. To do these keywords were decided upon and then from the articles found the abstracts were read and then any articles that were not of any use were discarded. The keywords that were produced were generated from the research objectives and questions that relate to the objectives. For the literature review the search parameters would consist of two components linked together using the Boolean operator of "AND". The search strings used for this research are:

- Blockchain AND GDPR
- Digital Ledger Technology AND GDPR
- Data Protection Act AND Blockchain

These search strings are efficiently detailed so they do not produce thousands of unrequired pieces of literature, as when using the search keywords of "Blockchain" AND "Data Privacy" the results even when filtered were in the thousands. The first searches performed was done using the Scopus research database. Scopus is a well-recognised database of abstract and citation documents of peer-reviewed literature and searches for documents by using titles, abstracts, and keywords. To explore further and enlarge the search scope the IEEE Xplore Digital Library was used using the same search parameters as before. Once the searches completed, they were then filtered through inclusion and exclusion parameters which is explained in the next section.

Exclusion and Inclusion Parameters:

- The literature selected was based on exclusion and inclusion parameters. The parameters are listed below:
- Exclusion parameters:
- Review articles and conference reviews
- Duplicate articles
- Articles that do not discuss compliance issues but discuss more blockchain or GDPR in general

Inclusion parameters:

- The article is in the English Language
- Full text is available from digital databases
- The article includes a framework on GDPR compliance issues of blockchain

4.3 Results of Literature Review

Initially from the Scopus database 242 articles were found and additionally 97 were found from the IEEE database search. In total this amounts to 339 articles. Once filtered through the exclusion parameters the remaining articles left was 169. Table 1 shows the results data from the searches performed.

Database	Keywords	Results
Scopus	'Blockchain' AND 'GDPR' 'Digital Ledger Technologies' AND 'GDPR' 'Data Protection Act' AND 'Blockchain'	59 9 20
IEEE Xplore	'Blockchain' AND 'GDPR' 'Digital Ledger Technologies' AND 'GDPR' 'Data Protection Act' AND 'Blockchain'	65 5 11

Table 1: Search Result Data

To accomplish the first aim of the project which was to find disparities between blockchain and GDPR and prior research in this field, a paper published in the Journal of Physics: Conference Series was where I started. The paper *Blockchain and GDPR – A Study on Compatibility Issues of the Distributed Ledger Technology with GDPR Data Processing* by Suripeddi and Purandare (2021), was where I started as it starts with comparing GDPR and blockchain. It recognises that blockchain technology provides upside of transparency and immutability, however, these properties also cause significant conflicts with GDPR data protection regulation. The paper advises that any blockchain developers cautiously investigate the information proposed for capacity in blockchain and weigh up its circumstances and disservices on how blockchain is to be utilized.

The paper *Blockchain and the General Data Protection Regulation: Can Distributed Ledgers be Squared with European Data Protection Law* produced by Finck (2019) has discussed the application of the European Union's EU General Data Protection Regulation to blockchain technologies. In the study it has been observed that many points of tension between blockchains and the GDPR can be identified, and it can be maintained that these are due to two overarching factors. Firstly, the GDPR is based on the underlying assumption that in relation to each personal data point there is at least one natural person, the data controller, that data subjects can address to enforce their rights under EU data protection law. Blockchains, however, often seek to achieve decentralisation in replacing an actor with many different players. This makes the allocation of responsibility and accountability difficult. The GDPR assumes that data can be modified or erased where necessary to comply with legal requirements such as Articles 16 and 17 GDPR. Blockchains, however, render such modifications of data purposefully arduous in order to ensure data integrity and increase trust in the network. The study concluded that it can be easier for permissioned blockchains to comply with these legal requirements as opposed to permissionless blockchains. It has, however, also been stressed that the compatibility of these tools within the Regulation can only ever be assessed on a case-by-case basis. Blockchains are a class of technologies with contrasting technical features and governance arrangements. This implies that it is not possible to assess the compatibility between 'the blockchain' and EU data protection law. The key takeaway from this study should be that it is impossible to state that blockchains are, as a whole, either compliant or incompliant with the GDPR. Also highlighted in this study is that where there is a certain tension between various key features of blockchain technologies and various elements of European data protection law, many of the related uncertainties should be traced back to the specific features of DLT. Blockchains can offer benefits from a data protection perspective, however, they need to be purposefully designed in order for this to materialise. Where this is the case, they may offer new forms of data management that provides benefits

to the data-driven economy and enable data subjects to have more control over personal data that relates to them.

The second aim was to research how Blockchain stores and manages data and how it infringes the GDPR. One of the sources used to understand how blockchain stores data was an article by Sahu (2020) *How to Use Blockchain and Store Data*, which explains multiple ways to store data on the blockchain. This is where the idea to use IPFS originated over databases as a system holding multiple file types was seen as more productive. Research was then started in ways that data can be protected on the blockchain.

Protection of Personal Data in Blockchain Technology is a paper by Wallace (2018) which is a paper on the protection of personal data on the blockchain. It informs that encryption is not a method on anonymisation unless the original data is erased, and the bar is set out low on what constitutes as indirect personal data. The conclusion must be that natural persons may be identified in the blockchain. It goes on to declare that the GDPR is adapted only for cases where there is at least one actor who holds some sort of power over data subjects and their personal data and that the principles of the GDPR cannot either be fulfilled in a public blockchain. This further strengthens the use of permissioned blockchains. The purposes cannot be identified unless there is a controller responsible to begin with, and the principles of data minimisation, accuracy and storage limitation cannot be met since the data entered on the blockchain is practically impossible to tamper with and the possibility of erasure would undermine the very purpose of the blockchain. The GDPR might succeed in being implemented where there is someone in power of the personal data. However, it does not succeed in a public blockchain, which would mean that the GDPR fails in its attempt on being technologically neutral. The paper further explains that the principle of transparency can be fulfilled, and the assessment has to be made on a case-by-case basis. It goes on to say even the French data protection authority, the Commission nationale de l'informatique et des libertés (CNIL), seem to avoid giving advice on the public blockchains. The recommendations of the CNIL overall runs counter to the purposes of public blockchains since such networks intend to be borderless. The EU's opinion on blockchain also seems to be unclear (Wallace, 2018). In one way it favours the rights of the data subjects and in another way, it invests millions of euros into blockchain projects. The EU's intention must logically be that both GDPR and the blockchain investment forms part of the digital market strategy and eventually a mutual tipping point will be met that satisfies both sides. However, behind the GDPR and blockchain issue, lies the balancing of the concept of privacy and the concept of transparency. Privacy within the EU requires that personal data is kept away to some extent whereas the principle of data minimisation, accuracy and storage limitation is fundamental in the GDPR (Wallace, 2018). Transparency on the other hand, is needed to ensure that the authorities who process personal data comply with these obligations. The blockchain favours transparency where it should not necessarily be a threat on the GDPR, but rather a tool for achieving it. Regulators may need to change how to balance data transparency with the right to erasure, especially in cases where a decentralized technology is used. The paper continues to say decryption keys can be kept outside the blockchain, to ensure no personal data are processed directly in the blockchain. Thereby, it would still constitute personal data, but by deleting the decryption key and ensuring no one will ever get access to that data again, the right to erasure could be fulfilled. However, the reason this is not done already is because it requires a lot more capacity, and such solution could just as well mean the blockchain is not useful anymore. It would undermine the purpose of blockchain since it would not be as transparent as it is today. From this it is also apparent that some legal changes have to be made in the EU to solve this conflict. The discussion has to continue until balance is found between privacy and blockchain technology. This would help organisations in fulfilling their accountability obligations and showing that they try to comply with the GDPR.

5. Methodology

5.1 Software Development

Software Development is the process of taking a set of requirements (a problem statement), analysing them, designing a solution to the problem, and then implementing that solution on a computer (Dooley, 2017). The software development life cycle (SLDC) (Figure 1) is a process that produces software with the highest quality in the shortest time which includes a detailed plan for how to develop, alter, and maintain a software system (Stackify, 2017). There are many different models of the SLDC, and the Waterfall method (Figure 2) is one of them.

(Figure 1, The SLDC, Synotive, 2017)

The core methodology chosen for this project is the traditional Waterfall method however, elements and ideas from iterative and agile processes were also considered throughout. Figure 2 shows The Waterfall Model (Adobe Experience Cloud, n.d.) and show how it is a step-by-step process of development that follows a strict path with defined stages along the way. The first stage of the Waterfall method is the Requirements phase. This is where the requirements of the project are gathered and analysed. As there is no defined way that requirements should be gathered using the waterfall method, various techniques from different methodologies could be used at this stage. This is an advantage because it allows the freedom of choosing the most suitable methods of requirements elicitation for the project at hand. Since the requirements stage is fully completed before the development takes place, it ensures that potential problems are considered early and also avoids unexpected issues as requirements will not be continuously added along the way. The requirements stage of the waterfall method ensures that all requirements are clearly defined and documented, and that they define exactly what the system should do. The next stage is the design phase. This is where the requirements gathered from the first stage are carefully reflected upon in order to design a system that will meet the defined requirements. An advantage of completing the design phase thoroughly before beginning development is that it helps to decide how the different components of a system will work together all at once. This is another way of avoiding unexpected issues as it does not leave an opportunity to add new features into a design that would slow down the development and thus potentially missing target milestones. The implementation stage is where the development and coding take place. By the end of the implementation stage, a working piece of software is complete. Once the system is developed, it must then be tested. The testing phase will be where a series of tests to spot bugs and errors will be run, this is to ensure that all requirements are being met. The final stage at the bottom of the waterfall is the maintenance/operations stage. This is where any issues

identified during the testing phase are addressed in order to get the system ready for release. While the waterfall model has seen a slow phasing out in recent years in favour of more agile methods, it can still provide a number of benefits, particularly for projects that require the stringent stages and have strict deadlines (Powell-Morse, 2016). One benefit of the waterfall method is that it forces structured organisation, and this helps to meet targets and stay on track. Due to the well-defined requirements and design stages pre-development, the processes of the project have been made easier to document than if large amounts of improvisation and adaptation took place. The waterfall method is suited to milestone-focused development. Due to the strict deadlines and documentation requirements of this project, the waterfall method was very suitable in aiding to meet any milestones and deadlines. As well as this, most agile processes require constant involvement of users, and this was not applicable for this project. For these reasons, it is clear that the waterfall method was the most appropriate choice of methodology for this project.

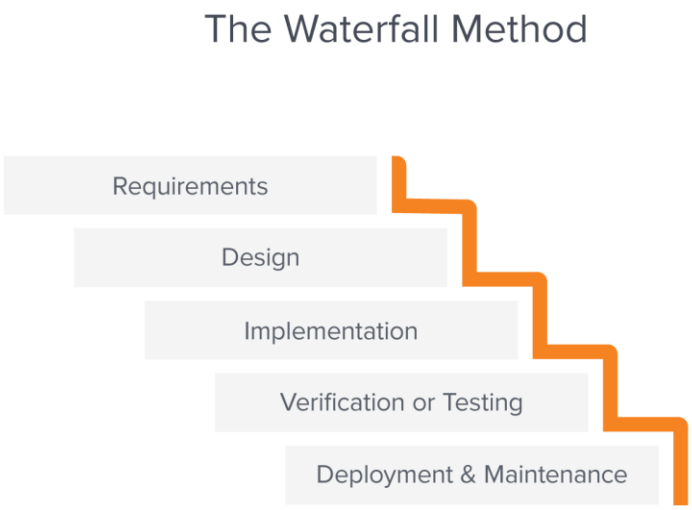


Figure 2. The Waterfall Method (Adobe Experience Cloud, n.d.)

5.2 Project Management

Project management techniques refer to the tools and processes used to aid in carrying out a project whilst maintaining scope and timeframes.

5.2.1 Gannt Chart

As discussed in the Project Proposal and Interim Report documents, a Gantt chart was created to plan the schedule of this project. Gantt charts are a visual representation of tasks against time in the style of a bar chart. They represent critical information such as the duration of tasks and overlapping activities. They are useful because they are relatively simple to create and understand, and they serve as a timeline that illustrates how the project will progress (Kashyap, 2019). The Gantt Chart is a very important and suitable tool for the waterfall methodology since a complete chart often takes the same shape as the steps in the waterfall and is therefore in sync with the methodology. An advantage of making a Gantt chart before beginning development is that all necessary tasks and activities are considered, decided, and planned.

This meant that it was less likely to forget a step or to carry out tasks in an illogical order. Furthermore, it allowed for estimates to be made regarding the timeframe of the project overall and the individual tasks necessary to complete. However, the Gantt chart for this project was a failure. While it was useful in making initial estimates, deciding tasks, and identifying milestones, it was not used to its fullest potential. Some of the timeframes set proved to be too ambitious which led to the project falling behind and large amounts of stressed caused by the build-up of tasks. The Gantt chart was updated and remade to adapt to these new timescales midway through the project which were meant to help with catching up the original timeline. However again due to over-ambition as well as other assessments to complete, meant that some tasks were still not completed within timeframes set. The entire project overall was completed within the whole timeframe, but the timelines never went according to plan. Appendix 1 contains the adapted Gannt chart to show the actual timeline where red will indicate a milestone not hit on time and yellow indicates more time required.

5.2.2 Project Supervisor Meetings

Upon receiving feedback from the Interim Report that was submitted, one piece of feedback that was given to me was to have more supervisor meetings to help stay on track and to manage the processes of a project this size. Upon this feedback I believe I managed to arrange meetings bi-monthly. If I had no updates or issues, e-mail communication was made. I believe this change did keep me on track and helped me understand more of what is required of me for this project and how to stay on course.

5.3 Toolsets and Machine Environments

A toolset relates to any kind of software tool that is helpful in any aspect of the project whether that be management or development. This section will focus on those tools which aided development of the project. Many of the different tools used throughout this project, as is to be expected of a project this size, and have been listed in the table below and which part they aided in. Some tools are very straightforward and down to simply personal preference or have no major competitors therefore only a selection of the toolsets chosen will be discussed. Most the tools listed are open-source tools and therefore will not be using this as an advantage or comparison.

Design	Documents	Programming	Browser	Video
Figma	Microsoft Word, Microsoft PowerPoint	PyCharm IDE Python HTML CSS JavaScript jQuery Bootstrap JSON Flask SocketIO IPFS AES Encryption	Microsoft Edge, Google Chrome	YouTube

Table 2: Toolsets and Environments Used

5.3.1 Python

Python language is a simplistic scripting language which allows the development of many different types of applications. Variable types in the Python programming language do not need to be declared as it is a dynamically typed language and allows code to be run uncompiled which made debugging and fixing issues that arose generally easier, rather than having to compile, stop the code, fix the code, compile again, and restart the application. Albeit a language such as C++ on the other hand is a compiled language and the evaluation of the code happens at compile time. The output of this process is a machine language executable where lines of code can execute in a handful of clock cycles. On a multi-GHz processor, this ends up being a lot faster than Python. However, due to the features listed as well as the ease of use, simplistic syntax and a more experience personally was the reasoning why Python was chosen over a compiling language such as C++.

5.3.2 PyCharm IDE

PyCharm was the environment of choice to develop the application. PyCharm is one of the most popular Python IDEs. It is developed by JetBrains, the developer behind the popular IntelliJ IDEA IDE that is one of the big 3 of Java IDEs (Arora, 2022). It comes pre-packaged with many modules that help and hasten development as they decrease the effort required to do the same task at a greater extent. PyCharm also has a large open-source community that develops modules that are easy to install. There are numerous modules installed such as the 'ipfsclient' module that was developed by the community that have been used. PyCharm was used over Visual Studio due to its plethora of community built libraries that can be imported, some of which were unavailable on Visual Studio. Python language was used to develop the blockchain in blockchain.py, along with the various functions the website requires. Uploading and downloading the files are in my_constants.py file and then in the server.py file there is the API for the website, functions for saving the file, retrieving the file, hashing the file as well as encryption and decryption functions.

5.3.3 HTML and CSS

HTML is the language used to create the layout and structure of the website and CSS is the language used to add styling such as colours, fonts, and advanced positioning. HTML and CSS are standards in the web industry and therefore no decision needed to be made regarding which programming languages to use for the front end of the website. Nevertheless, it was important to be aware of the current standards and make use of the newest features in order to futureproof the site. For example, many new tags were introduced in the latest version of html, HTML5, that were important to learn and utilise. Using these tags enrich the semantic value of the site and make the code cleaner and easier to understand. For example, using the tag makes the purpose of the section far clearer than a generic. Additionally, HTML5 forms were made much more elegant and offer new capabilities with new types of inputs and fields which reduced the need to use the much more complicated JavaScript for a relatively basic feature. Many JavaScript scripts running on a page can slow down response times so since forms were an important feature of the site, it was beneficial to be able to avoid large amounts of JavaScript since HTML alone can now handle advanced forms.

5.3.4 JavaScript

JavaScript has also become a standard of modern websites with any advanced features or dynamic information. It can be used to update and change both HTML and CSS based on different conditions and can also calculate, manipulate, and validate data (W3schools.com, n.d.). JavaScript can create responsive and interactive elements for web pages which is essential in enhancing user experience. However, JavaScript can be considered relatively complex, especially when compared with the simplicity of HTML and CSS. Therefore, for this project, jQuery was used. jQuery is a lightweight, "write less, do more",

JavaScript library which offers an easier way of implementing JavaScript. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish and wraps them into methods that you can call with a single line of code (W3schools.com, n.d.). There are many JavaScript Libraries, but jQuery seems to be the most popular by far. 77.3% of websites use jQuery and that it has a 94.8% market share (W3Techs. N.d.). As well as its simplicity and popularity, jQuery offers many other benefits such as interoperability, speed, and efficiency. Therefore, jQuery was a clear choice for this project.

5.3.5 IPFS (Inter-Planetary File System)

IPFS (Inter-Planetary File Storage) is a versioned file system that can store files and folders. It is a protocol and peer-to-peer network for storing and sharing data in a distributed file system. As this project is looking at how GDPR and blockchain can be compatible, an off-chain storage system that is compatible with blockchain was required. It uses Distributed Hash Tables (DHT). In DHT the data is spread across a network of computers, and efficiently coordinated to enable efficient access and lookup between nodes.

5.3.6 Infura

The Infura API provides developers with easy-to-access Ethereum-based infrastructure to build decentralized applications (dApps). Using the Infura Ethereum API, builders can connect applications in just a few seconds using a single line of code. The micro service-driven architecture that powers Infura is designed to dynamically scale with the Infura Ethereum API. Furthermore, using the Infura Ethereum API enables developers to connect to the InterPlanetary File System (IPFS) via WebSocket and HTTPS. This makes it extremely simple to build dApps using existing infrastructure. Below in Figure 3 shows how Infura works.

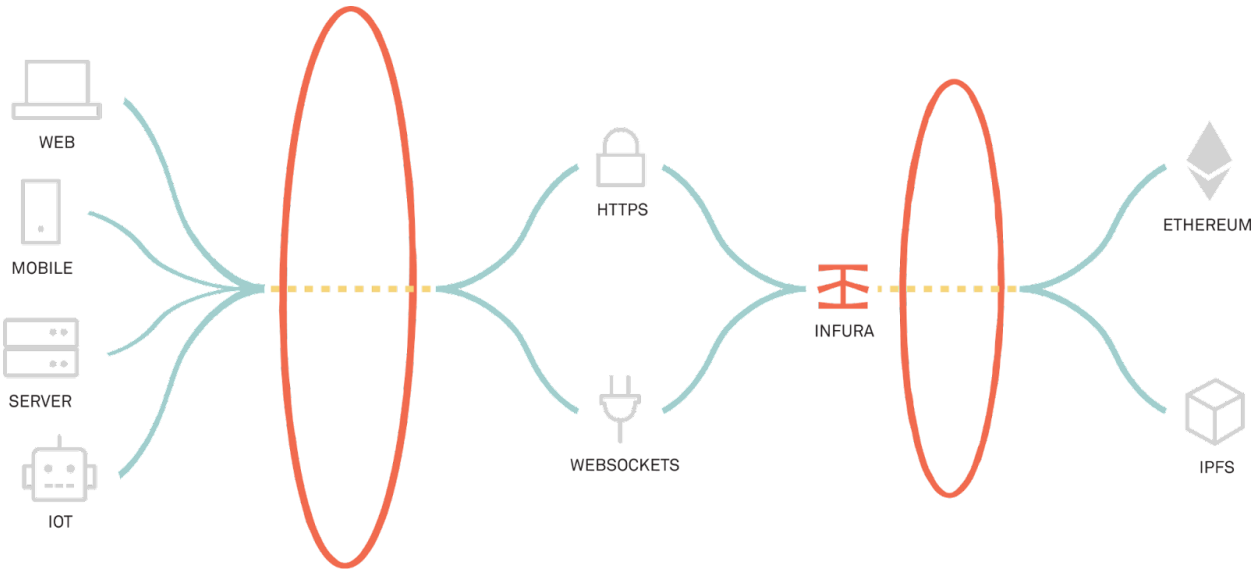


Figure 3. How Infura Works (Ivanotech, 2021)

5.4 Research Methods

Research methods are the strategies, processes or techniques utilised in the collection of data or evidence for analysis in order to uncover new information or create better understanding of a topic (Libguides.newcastle.edu.au, 2019). As this project does not focus on users per se, no quantitative or qualitative data was collected initially. The research that was carried out has been strictly literature as no user data was required to build the artefact. The results of project could be classed as quantitative data, as quantitative data is a measure of values or counts and are expressed as numbers (Australian Bureau of Statistics, n.d.), as it is numerical, and analysis can be made from the results of the project. There is also a counter for number of nodes on the chain and number of blocks. So an argument could be made that the result through usage provide quantitative data. However, as per the artefact, the research has mainly come from theoretical ideas where no artefact has been made. After researching this project on what was required for GDPR and blockchain to be able work together, different technologies have been brought together to create the solution that this project was taken on for.

6. Design, Development and Evaluation

6.1 Requirements

Requirements are arguably the most important phase of the Software Development Life Cycle (SDLC) since it defines exactly what is to be developed and can avoid potential issues later in development if done thoroughly. A study by the Standish Group noted that the three most commonly cited root causes of project failures, responsible for more than a third of projects running into problems are; a lack of user input (13% of projects), incomplete requirements and specifications (12% of projects) and changing requirements and specifications (12% of projects) (Mohapatra, 2010). All three of these issues can be addressed and avoided during Requirements Elicitation. Requirements Elicitation is the first stage of the Requirements phase in which potential users or stakeholders can inform developers regarding what they would like the system to do for them and how they might use it.

From the literature review and literature read, an easy to use off-chain blockchain storage solution is required. The blockchain will need to be a permissioned blockchain to ensure the control of the data on the blockchain. Permissioned blockchains are blockchain networks that require access to be part of. In these blockchain types, a control layer runs on top of the blockchain that governs the actions performed by the allowed participants. They are crafted to take advantage of blockchains without sacrificing the authority aspect of a centralized system. A permissioned system is also known to have a restriction on the participants, making permissioned networks highly configured and controlled by the owners (Iredale, 2019). Below is a table with the advantages and drawbacks of permissioned blockchains.

Advantages	**Disadvantages**
Efficient performance: When compared permissioned blockchains to permissionless blockchains, they offer better performance. The core reason behind this is the limited number of nodes on the platform. This removes the unnecessary computations required to reach consensus on the network, improving the overall performance. On top of that, permissioned networks have their own pre-determined nodes for validating a transaction.	**Compromised security:** A public blockchain has better security as the nodes participate in a consensus method properly. But, in the case of permissioned blockchains, this might not hold true. The security of a permissioned network is as good as the member's integrity. This means that a small section of a permissioned system can work together to modify the data stored within the network. In this way, the integrity of the network can be compromised. To resolve it, the system should have proper permissions set so those bad

	actors cannot merge together to cause the desired effect.
Proper governance structure: Permissioned networks do come with an appropriate structure of governance. This means that they are organized. Administrators also require less time to update the rules over the network, which is considerably faster when compared to public blockchains. The public blockchain network suffers from the consensus problem as not all nodes work together to get the new update implemented. These nodes might place their self-interest above the needs of the blockchain, which, in return, means slower updates to the whole network. In comparison, permissioned blockchain doesn't have the problem, as the nodes work together to move the updates faster.	**Control, Censorship, and Regulation:** Permissioned blockchains should work as that of a public blockchain, but with regulations. However, the regulations bring censorship to the network, where the authority can restrict a transaction or control it from happening. These are a threat to any organization that is using the permissioned network. This approach also stops the permissioned network from making the most out of the whole blockchain ecosystem.
Decentralized storage: Permissioned networks also make proper use of blockchain, including utilizing its decentralized nature for data storage.	
Cost-Effective: Permissioned blockchains reduce costs because they remove intermediaries, which becomes unnecessary in the blockchain protocol.	

Table 3: Advantages and Disadvantages of Permissioned Blockchains

To integrate with the blockchain an off-chain file storage will be need as well as a way to integrate this into the blockchain without compromising the data integrity. This will also need to be able to upload to and download from. All files will need to be hashed and encrypted for security reasons. GDPR also needs to be considered and that any and all data is following the correct regulations and that all regulations such as Article 17, the Right to Removal (European Union, 2016).

The requirements for the web application are:

- The system must avoid difficult navigation or long load times by having an optimal number of pages
- Information must be easy to access and easy to view and read
- The system must have a how-to guide to ensure correct usage of the application
- Easy upload/download usability
- The blockchain must be viewable for data integrity and call-back
- Be GDPR compliant

6.2 Design

The design phase is the second stage of the Software Development Lifecycle in which the established requirements are used to decide the structural and aesthetic properties of the system. During the design phase structure and use of the artefact are decided upon. This can be done in a multitude of ways. Firstly, a navigational diagram of the website has been made to show the content of the page and its navigational features. This defines the structure of each page and reflects how users get to each page and where they could go from it. This is useful for deciding which pages must be designed and the required navigational features to navigate between them. The structure helps to define how users interact with the product and how the system behaves when a user interacts with it. The structure has been split into two elements: interaction and informational design. Interaction design defines how user interaction affects the system and the user (Interaction Design Foundation, n.d.). Good interaction design helps users complete the tasks the aim to complete, makes the user aware of interactivity and functionality and helps to prevent user error or mistakes. Below is the navigational diagram of the website.

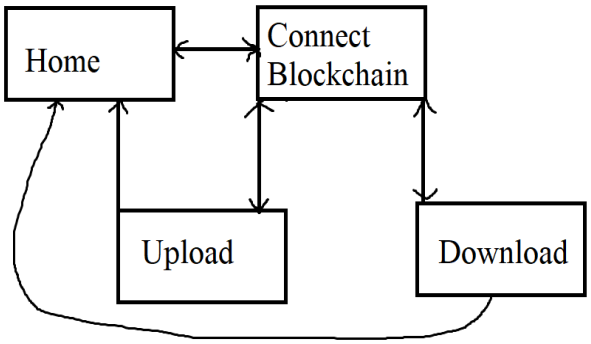


Figure 4: Navigational Diagram of Website

Informational design defines the order of the content and how it is arranged. Good informational design organises, categorises, and prioritises information based on user needs, makes it easy for users to understand and navigate through the information, and is appropriate for the audience (Elgabry, 2016). At this stage, the pages, the type of content to be displayed on each page, and the navigation features can be decided upon and designed. A helpful way of displaying this is by creating a diagram to show what the pages will be, how users can get to each of them and where they can go from them.

Figma was used to design the boxes that would eventually become the website. The design for me to work on is very simplistic but has the key elements of what each box is for. Without the design being complicated or detailed it allowed for more creativity when in the development of the project. For the homepage the design for the top half of the page was for navigation. Quick links to further sections of the page are in the header as to not overload the user with information. Get Started is a button that takes the user to instructions and the Demo video link will be a link to the demonstration of the video which will display a video on how to use the software. This will help demonstrate the project and allow users who use the software a video help guide in case they struggle understanding text. The ‘Connect Blockchain’ will take the user to the ‘Connect Blockchain’ page to begin uploading or downloading data. The image used is for aesthetics to make the website look professional. Figure 5 shows the design of the top half of the page.

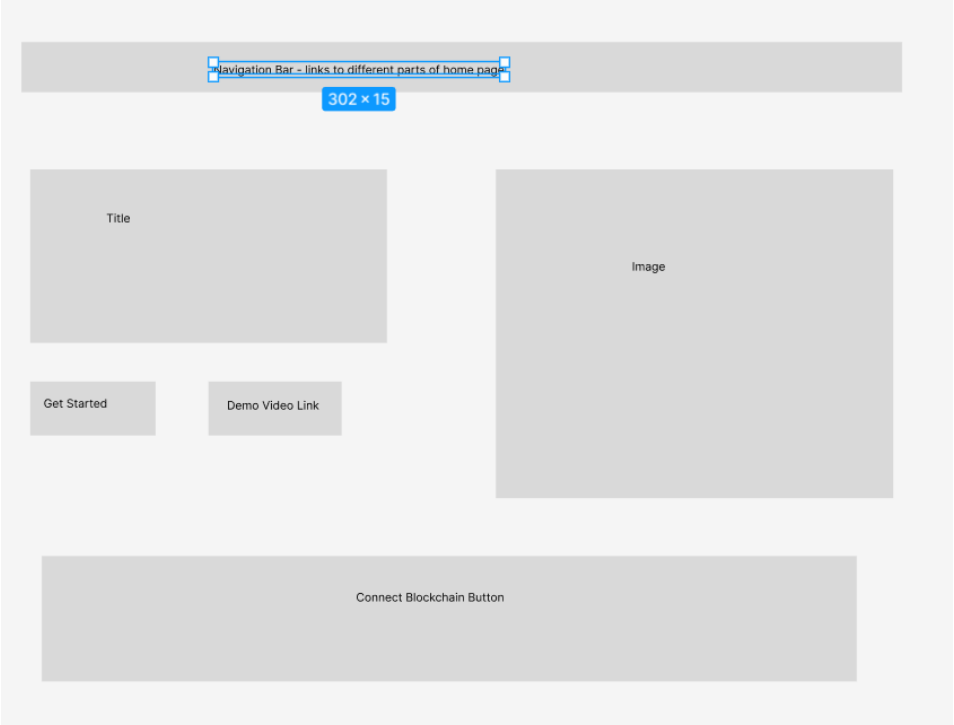


Figure 5: Top half of home page design

The lower half of the home page contains a textual instruction guide next to another image which has been placed for aesthetic reasons again. Following this will be information on the technologies used and a contact box for queries or help. This page is just to help the user understand the operation of the application and the technologies involved.

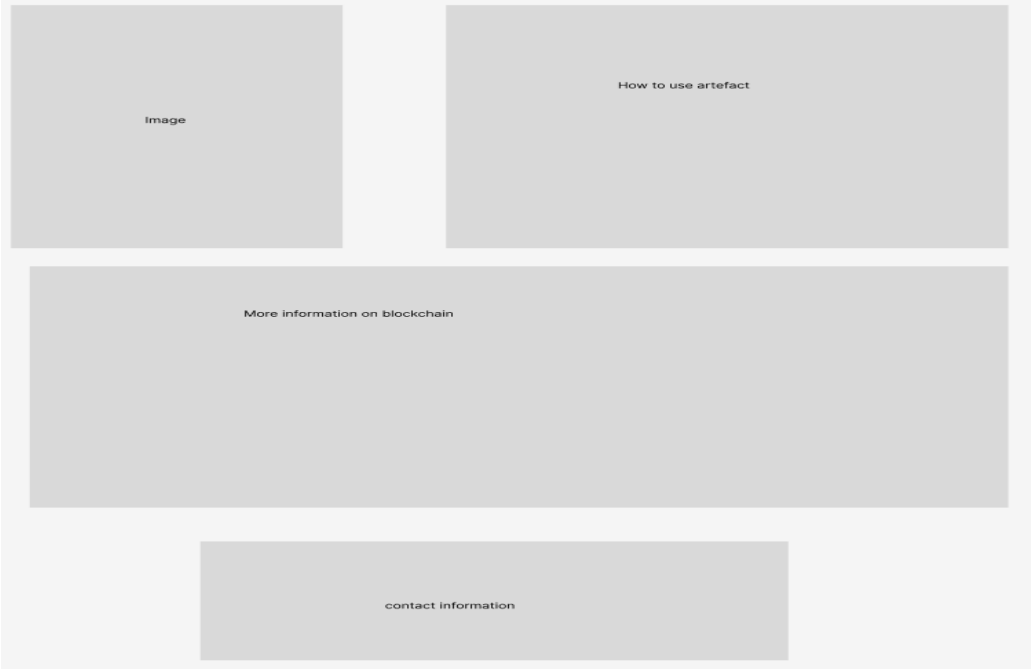


Figure 6: Lower half of home page design

Next the ‘Connect Blockchain’ page was designed. As this was the main operation page, the ease of use for this page required the most diligence. From Figure 4 below the style is simplistic yet easy to follow. It contains buttons that are self-explanatory, upload, download and disconnect. Upload and Download buttons link to their respective pages as shown in Figure 4 and the Disconnect button returns a user the home page. A counter was added to the application code to keep count of the number of nodes on the chain and the number of blocks the chain has. These two pieces of information are clearly marked out and will in a font and font size that is easy to read. Below this is will a display of information from each block on the chain.

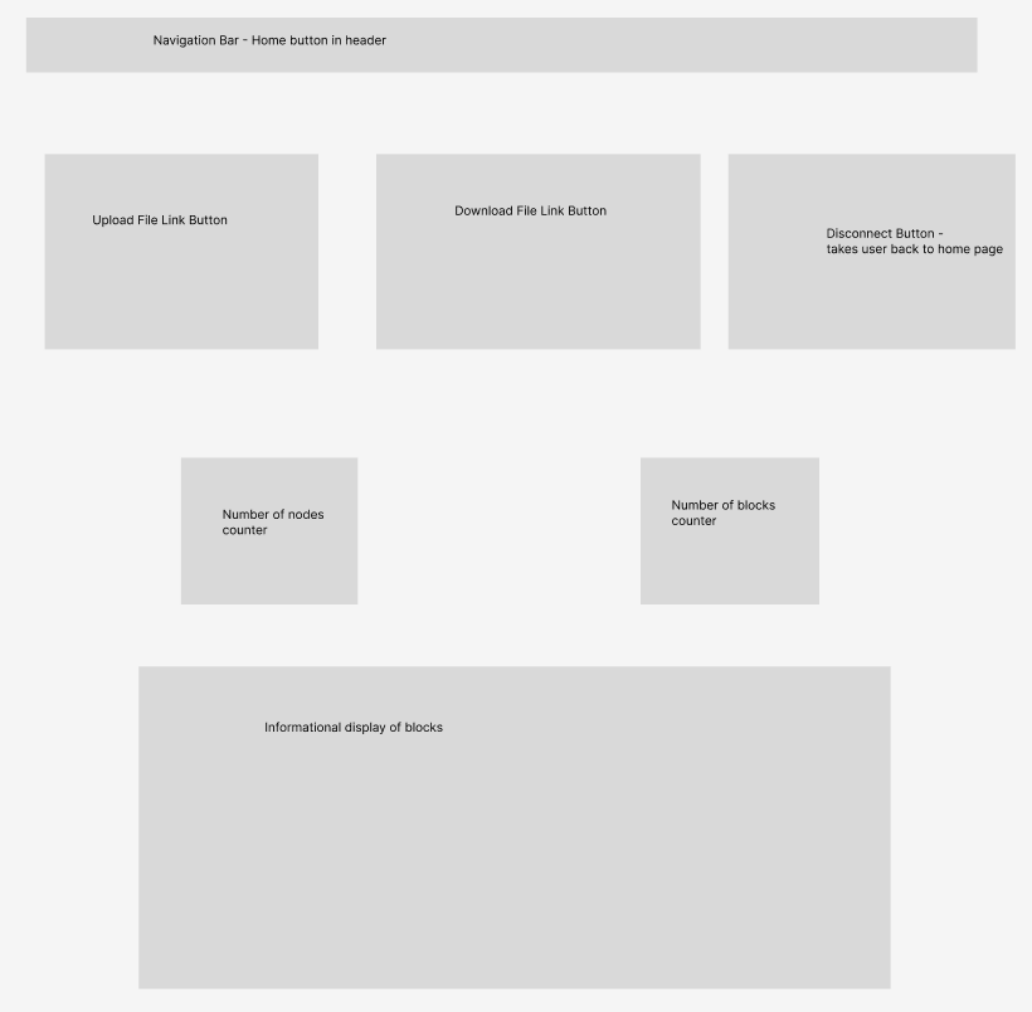


Figure 7: Connect Blockchain page design

The ‘Upload’ page was designed with simplicity in mind. So there is a basic welcome message box. Then there is a sender’s and receiver’s field box which will accept any name and will be displayed on the block information on the ‘Connect Blockchain’ page. The file key will be a user-derived key has no minimal limit. A ‘Choose file’ button will allow the user to select a file and upload it. Submit will then add the hash of the file to the blockchain and will store the file off-chain using IPFS. The view chain button brings the user back to the ‘Connect Blockchain’ page that will have been updated with the new block. Below is the design for this page.

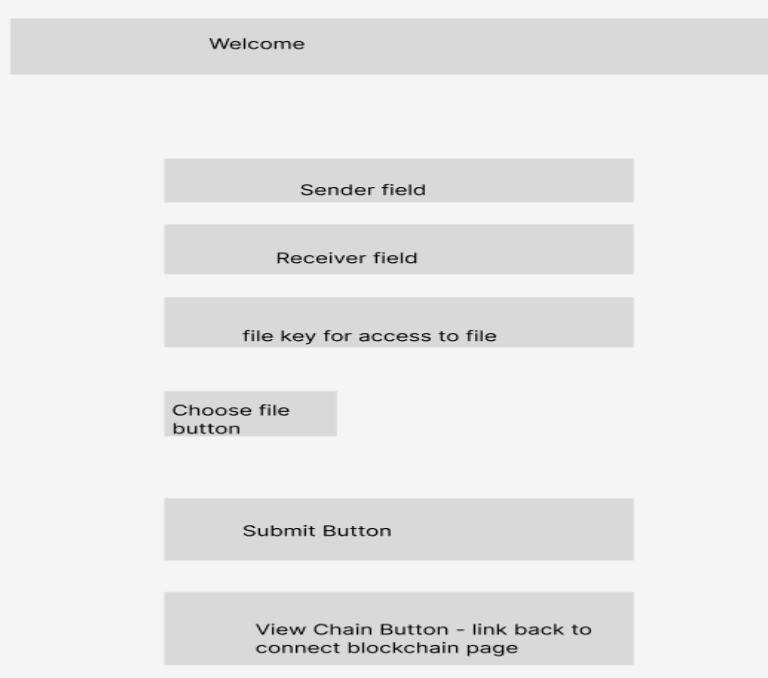


Figure 8: Upload page design

The ’Download’ page will retrieve a file on the chain. It has a field for the user-generated hash key that the uploader would share to the whomever requires the file stored. The shared hash of the file can be taken from the chain information from the ‘Connect Blockchain’ page. The submit button will download the file from IPFS and save it locally. The view chain button brings the user back to the ‘Connect Blockchain’ page. Figure 7 shows the design for this page.

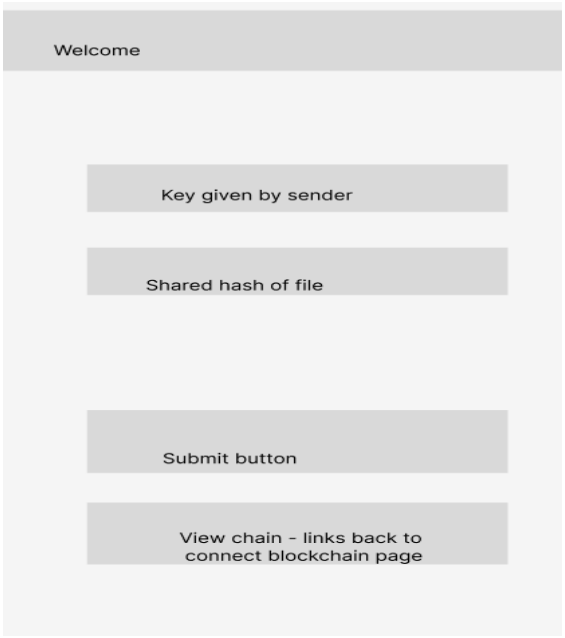


Figure 9: Download page design

The design for the interactive parts of the application lacks a finesse that could have been applied. Due to time constraints, I decided to focus more on the development of the artefact and left the design looking very simplistic and dull. However, it achieves what was wanted and the desired design from which the artefact was built.

Three program files will be developed to handle the computing side of the artefact. One will be the new blockchain, another will handle file uploads and downloads and the third program file will be responsible for website interactions including the API. For the website design, Figma, which is a has been used to design a user-friendly, professional website. For optimal efficiency there will be 4 webpages that will be able to perform all tasks required.

6.3 Development

The development/implementation phase is the central stage of the SLDC Waterfall Model. This is where the actual system is created through programming. This is usually the longest stage of the process as it is often the largest and most complex task. For this reason, it is often broken down into manageable subtasks as shown in the Project Management section. These subtasks are often referred to as units which can then be used to carry out unit testing which is where a single piece of the system is tested to ensure it works by itself before testing how it integrates with the rest of the system. Once the implementation is complete, all of the requirements should have been met which is why it is essential to keep the requirements in mind at all times. Furthermore, factors such as experience can also impact what is done, what is prioritised, and how long is spent on each subtask.

As the design section was carried out carefully and thoroughly, few decisions needed to be made regarding aesthetics and layout as these had already been made however, the whole process took longer than expected overall due to limited prior knowledge and several functionality features requiring more careful planning.

The back-end programming was done first as this would need to be implemented into the front-end of the artefact.

6.3.1 Blockchain.py

The development of the blockchain in blockchain.py was where the development started. A genesis block was created as this is requirement of the blockchain. After this a function called ‘create_block’ was made to create additional blocks that will be needed.

```
def create_block(self, proof, previous_hash, sender, receiver, file_hash):
    block = {'index': len(self.chain) + 1,
             'timestamp': str(time.strftime("%d %B %Y , %I:%M:%S %p", time.localtime())),
             'proof': proof,
             'previous_hash': previous_hash,
             'sender': sender, #########
             'receiver':receiver, #########
             'shared_files': file_hash}
    self.chain.append(block)
    return block
```

Figure 10: Function to create a block on the blockchain

This function creates a block when something is uploaded, each block will have a number to identify the block, a timestamp to show when the file was uploaded. The proof is a nonce, which stands for "number only used once," which is a number added to an encrypted block in a blockchain that, when rehashed, meets the difficulty level restrictions for example by varying the proof we can vary the hash generated so that a new block can be created. This was added as it acts as another layer of security further proofing that the chain is secure and can be seen below.

```
def proof_of_work(self, previous_proof):
    new_proof = 1
    check_proof = False
    while check_proof is False:
        hash_operation = hashlib.sha256(str(new_proof**2 - previous_proof**2).encode()).hexdigest()
        if hash_operation[:4] == '0000':
            check_proof = True
        else:
            new_proof += 1
    return new_proof
```

Figure 11: Function that creates a proof of work (Nonce)

The previous hash represents the hash of the previous block. The hash of the entire block is generated using the SHA-256 hashing algorithm. This field creates a chain of blocks and is the main element behind blockchain architecture's security. The sender is the uploader of the file and will be the decider of the key that is used to download the file. The receiver downloads the file via the hash that was generated and the uploader's key. The Shared file field is the uploaded file when it is first encrypted with the file key given by the uploader using the AES encryption mechanism and subsequently using the SHA-256 hashing algorithm when it is uploaded to IPFS. The hash, then received from the IPFS after the encryption is the hash of the shared file which is added to the block. The chain is then appended with each new block.

Also in this part of the program, a hash function has been made to hash the data using JSON and SHA256 hashing. This is to keep any hashes secure, even the slightest change to the data would change the hash allowing for transparency as well as security, as if someone has managed to get into the block to change anything it will be obvious as the hash would not deliver the expected data. Before files are added to the blockchain, a function was developed to ensure security of any data and that the chain that the file was going to be uploaded to is the valid chain and it does this by reviewing the blockchain and ensuring that it is the next block to be added. This feature ensures that data is not lost into cyberspace on a chain that does not exist. Next, to allow files to be added to the blockchain an add file function has been developed to allow files to be added to the chain. This function takes the information in the 'create_block' function and adds it to the file uploaded and then creates a block from this. The replace chain function is there to check if the chain that is being added too is the longest chain. The longest chain rule is a rule used in public blockchains that I thought I would adapt to this permissioned chain as it provides a logical way to decide what chain should be added to. The longest chain is what individual nodes accept as the valid version of the blockchain. Nodes that adopt the longest chain of blocks allows every node on the network to agree on what the blockchain looks like, and therefore agree on the same transaction history (Walker, n.d.). It allows computers acting independently over a network to maintain a globally shared view of a file and this also adds to the transparency of the blockchain developed. The code for all functions developed can be found in Appendix 2 via supporting documentation.

6.3.2 My_constants.py

This part of the application adds anything uploaded or downloaded to the relevant folders. It saves the encrypted and decrypted data in the relevant folder. If a user uploads a file, it will label it to whatever it is currently saved as. Once it is uploaded to the file system the encrypted image is then saved to the destination the program tells it too but encrypted. When the file is downloaded the encrypted data is downloaded and named by the hash that was generated and then once decryption has finished the decrypted image is saved under the same naming mechanism. To ensure this works I have developed it so that it saves to my local file. This was to ensure that the feature works.

```
from flask import Flask

UPLOAD_FOLDER = '/Users/Tom/Desktop/Blockchain/main_server/uploads'
DOWNLOAD_FOLDER = '/Users/Tom/Desktop/Blockchain/main_server/downloads'

app = Flask(__name__)
app.secret_key = "secret key"
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER
app.config['DOWNLOAD_FOLDER'] = DOWNLOAD_FOLDER
app.config['ALLOWED_EXTENSIONS'] = set(['txt', 'pdf', 'png', 'jpg', 'jpeg', 'gif'])
app.config['BUFFER_SIZE'] = 64 * 1024
app.config['MAX_CONTENT_LENGTH'] = 16 * 1024 * 1024
```

Figure 12: my_constants.py code

6.3.3 Server.py

With Blockchain.py focused on the creation of the blockchain and subsequent blocks and the my_constants.py focused on the file handling. The last program to create was the functions of the application. With this part of the program, functions for the functions of the application were made. From the screenshots below, the functions created necessary for functionality were the ‘allowed file’ function checks if the files are on the allowed extensions list that lies in my_constants.py. The next function uploads the file to the chain and saves the upload to the destination set. The next two functions encrypt and decrypt the file and saves this where the destination has been set too with the AES encryption that has been used. The hash_user_file function is a function that hashes the file using SHA-256 hashing algorithm to IPFS. This function is followed by the retrieve_from_hash function to retrieve the file from the blockchain using the hash and to save it to the destination set in my_constants.py.

```
def allowed_file(filename):
    return '.' in filename and filename.rsplit('.', 1)[1].lower() in app.config['ALLOWED_EXTENSIONS']

def append_file_extension(uploaded_file, file_path):
    file_extension = uploaded_file.filename.rsplit('.', 1)[1].lower()
    user_file = open(file_path, 'a')
    user_file.write('\n' + file_extension)
    user_file.close()

def decrypt_file(file_path, file_key):
    encrypted_file = file_path + ".aes"
    os.rename(file_path, encrypted_file)
    pyAesCrypt.decryptFile(encrypted_file, file_path,  file_key, app.config['BUFFER_SIZE'])

def encrypt_file(file_path, file_key):
    pyAesCrypt.encryptFile(file_path, file_path + ".aes",  file_key, app.config['BUFFER_SIZE'])
```

Figure 13: Code snippet from server.py showing functions created for webpage functionality

```
def hash_user_file(user_file, file_key):
    encrypt_file(user_file, file_key)
    encrypted_file_path = user_file + ".aes"
    client = ipfshttpclient.connect('/dns/ipfs.infura.io/tcp/5001/https')
    response = client.add(encrypted_file_path)
    file_hash = response['Hash']
    return file_hash

def retrieve_from_hash(file_hash, file_key):
    client = ipfshttpclient.connect('/dns/ipfs.infura.io/tcp/5001/https')
    file_content = client.cat(file_hash)
    file_path = os.path.join(app.config['DOWNLOAD_FOLDER'], file_hash)
    user_file = open(file_path, 'ab+')
    user_file.write(file_content)
    user_file.close()
    decrypt_file(file_path, file_key)
    with open(file_path, 'rb') as f:
        lines = f.read().splitlines()
        last_line = lines[-1]
    user_file.close()
    file_extension = last_line
    saved_file = file_path + '.' + file_extension.decode()
    os.rename(file_path, saved_file)
    print(saved_file)
    return saved_file
```

Figure 14: Code snippet from server.py showing functions created for webpage functionality

The rest of the program contains the API for routing throughout the application. This has been done using Flask and is due to the size and number of API routes this has been added to the supporting documentation via Appendix 2.

Flask was chosen over a framework such as Django due to my inexperience developing API's and Flask was less complicated over Django. Whilst many reviews online including one by Singh (2022) have Django as the better of the two it also comes with a much deeper learning curve. Therefore for this application Flask was chosen as the framework.

Using Socket programming, the blockchain is setup so that only those with permission to view the chain are able to view the entire chain. Socket programming enables the communication for sending and receiving the data between the socket endpoints by using the code logic. An example of a socket would be a node (Pedamkar. N.d.). In the artefact, socket programming has been used via imported libraries. The image below shows how this has been implemented.

```
# Getting the full Blockchain
@app.route('/get_chain', methods = ['GET'])
def get_chain():
    response = {'chain': blockchain.chain,
                'length': len(blockchain.chain)}
    return jsonify(response), 200

@socketio.on('connect')
def handle_connect():
    print('Client connected')
    print(request)

@socketio.on('add_client_node')
def handle_node(client_node):
    print(client_node)
    blockchain.nodes.add(client_node['node_address'])
    emit('my_response', {'data': pickle.dumps(blockchain.nodes)}, broadcast = True)

@socketio.on('remove_client_node')
def handle_node(client_node):
    print(client_node)
    blockchain.nodes.remove(client_node['node_address'])
    emit('my_response', {'data': pickle.dumps(blockchain.nodes)}, broadcast = True)

@socketio.on('disconnect')
def handle_disconnect():
    print('Client disconnected')
    print(request)
```

Figure 15: Code snippet showing how socket programming was used

6.3.4 Website Development

Beginning with the home page of the site was a good place to start development as any. The first step of development was using CSS to make features such as the background, font, and font colour representative of the colour scheme desired. Choosing an easy-to-read font as well as background and font colours that complemented each other were chosen so that anyone using the site can read it. For this reason, this was an important requirement to meet. Furthermore, an established layout and colour scheme can help to position and style content appropriately in a way that complements the theme and does not clash. The next requirement for the website was to avoid difficult navigation or long wait times by having a minimal number of pages. So, the next stage in development was to create all the necessary HTML pages and create a navigation feature. As shown in the design, the navigation bar was to remain constant on the 'home' and 'connect blockchain' pages. This was to ensure a user could get back to the home page as well as get to the information required on the home page. There is no navigation menu for the 'Upload' and 'Download' pages as they link back to the 'connect blockchain' page.

With the aesthetics and navigation done, the next job was to add all the information to the 'home' page. This included setting up a link the demonstration video as well as setting up the images with the text and the Connect Blockchain button. The 'Connect Blockchain' page was next to be setup. The first step in doing this was making sure the three boxes in the design were made and able to link to the correct pages. After this a group list was made to ensure that the blockchain could be viewed easily. Counts were also added to show as user number of nodes or blocks.

The 'upload' and 'download' pages were developed next and as these pages were relatively simple in design did not take as much time as first thought. The 'Upload' page was developed first. The first step was to create a html form with the fields that the users will see and be required to fill in. After this a button to choose a file was added to allow file uploading. The 'submit' and 'view chain' buttons were also then added. The 'Download' page was developed last and acts very similar to the 'upload' page with a html form just with different fields, 'submit' and 'view chain' buttons just no file select button. It is a little simpler to the 'upload' page as it only required the fields for the uploaders file key and the hash generated from the upload. The final step of development was then to implement a feature in which the chain that is viewable is then updated to represent the fie that has been uploaded.

6.4 Testing

Testing is the penultimate phase of the waterfall model in which the fully developed system is tested in a variety of ways to assess its functionality, identify bugs, and validate its usefulness to the user.

If no testing takes place until development is fully completed, it is very likely that there will be many bugs that were not identified whilst coding. However, although the process was planned to be linear, development and testing were carried out iteratively. As an fairly inexperienced programmer, tests were performed continuously throughout development to check that a piece of code did what it was expected to do. This was done because it is easier to identify what part of the code is causing the issue if it is tested immediately, rather than having to search through a completed set of code. For this reason, the tests carried out in the table below were largely successful as all bugs had been identified and fixed during development. As the sole developer of the project white box testing was used for testing. White box testing is a software testing method in which the internal structure, design and execution of the item being tested is known to the tester. The tester chooses inputs to exercise paths through the code and determines the appropriate outputs, and programming know-how and the implementation knowledge is essential (Software Testing Fundamentals, n.d.). Whilst the first stage of documented testing is usually unit testing,

this had largely been completed during development and therefore was not fully documented or done linearly. Unit testing is a level of software testing where individual parts of a software are tested. The purpose is to validate that each part of the software performs as designed. In order to save some time the first documented testing phase combined both integration tests and functionality testing. Integration testing is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units (Software Testing Fundamentals, n.d.). Functionality testing was also done, and this is when a system is tested for its compliance with the requirements and therefore, its functionality that were defined in the requirements stage. Testing is carried out to ensure product quality, security, and user satisfaction.

Test Number	Test Description	Input	Expected Outcome	Actual Outcome	Status
1	Connect to Blockchain button loads the correct page	Click the 'Connect to blockchain button'	Connect Blockchain page loads	Connect Blockchain page loads	Success
2	Upload file button goes to the correct page	Click the 'upload file button'	upload page loads	upload page loads	Success
3	Download file button goes to the correct page	Click the 'download button'	download page loads	download page loads	Success
4	Upload page input fields work	Sender Name 'Tom', Receiver Name 'Bob' File key 'block'	Fields are filled in and displayed on the block information	Fields are filled in and displayed on the block information	Success
5	Choose file loads the correct file	File chosen	File chosen and is ready to be hashed	File chosen and is ready to be hashed	Success
6	Submit button works	Submit	File is hashed and the hash is uploaded to the blockchain with success message	File is hashed and the hash is uploaded to the blockchain with success message	Success
7	Blockchain information is all correct in block	N/A	if the upload has worked the information displayed will be correct	if the upload has worked the information displayed will be correct	Success
8	Download page input fields work	Shared key and hash of data	Data downloaded to downloads folder	Data downloaded to downloads folder	Success
9	Error message if wrong key or hash is entered	Any data other than the correct	Error message appears	Error message appears	Success

10	Count of nodes and blocks are correct	N/A	Correct node and block count displayed	Correct node and block count displayed	Success
11	Disconnect button brings user back to home page	Clicking disconnect	Brings user back to home page	Brings user back to home page	Success
12	Retrieve file if file key is lost or forgotten	N/A	N/A	No option available	Fail – no option available to retrieve a forgotten file key
13	Multiple users can access chain	Change port numbers as to appear as two nodes	Chain updates on both users sites	Node number did not change but both users could access the hash	Partial success

Table 4: Testing Results

6.5 Operation

To use the artefact the first thing a user sees is the home page. To connect to the blockchain they click the 'connect to blockchain' button which will then load a page which will show the current chain. The user can then upload or download a file on the chain. If a user wants to upload, then they select the file they want to upload and decide on a file key that only they know. They will also fill in the sender and receiver names. After which the file will be encrypted and added to the blockchain. After uploading a file the user can click on 'View Chain' and this will take the user back to the previous page with the options and the updated chain. If a user wishes to download a file, they will need the hash that was generated and the file key that the uploader made. If both pieces of information are correct, then the file will be downloaded and decrypted. This process can be done as many times as a user likes. To disconnect a user simply hits disconnect and it will bring the user back to the home page.

7. Conclusions

In order to conclude this project, an evaluation of its success must first be carried out. The table below shows all original requirements and an explanation on how each of them has been met. This can help to evaluate the overall success of the project, based on how well the implemented system met requirements.

Requirement	Status	Explanation
Permissioned blockchain developed		The blockchain developed is a permissioned blockchain as described
Off-chain data storage integration	Success	IPFS off-chain storage has been implemented
Upload files successfully	Success	Successful upload completed
Download files successfully	Success	Successful download completed
The system must avoid difficult navigation or long load times by	Success	Optimal web design has been incorporated and an optimal

having an optimal number of pages		number of webpages have been produced
Information must be easy to access and easy to view and read	Success	Font and font size choices have been made to ensure easy readability
The system must have a how-to guide to ensure correct usage of the application	Success	How-to textual guide and demo video has been produced
Easy upload/download usability	Success	Upload and download options are easy to use and instructional guide/text has been provided
The blockchain must be viewable for data integrity and call-back	Success	Blockchain is viewable including all the information such as hashes
Be GDPR Compliant	Success	All GDPR regulations have been followed

Table 5: Requirements Met Results

The system has met the requirements set out at the start of the project. The question asked though at the start of this project was 'Can GDPR and Blockchain technology be compatible regarding data privacy and protection?' and this artefact has shown that yes it can be. Through a permissioned blockchain granting only authorised user even access to the chain prevents unwanted users from access. The user generated file key when uploading allows only permitted users who have gained access to the file key access to the file. This adds another layer of security as you need to contact the owner of the file to have any access. This could be made better with randomised phrases added to ensure the file key is not too easy to be brute forced. This was designed with good intentions however if the key is forgotten then there is no way of retrieving said key. This could lead to human error in getting the file and thus not being able to retrieve the file without re-uploading. SHA-256 hashing algorithm used with a nonce, which in the application is called proof, makes reverse encryption extremely difficult as hashes themselves are referred to as digital fingerprints, the nonce just adds another layer of security. As the sole user and controller nodes within this system I can control the data. If this was to go live however I it would be recommended that some changes are made regarding data security such as logins and where all the data would be saved too. Depending on the number of nodes also would mean more data controllers would also be required. To ensure that the blockchain views for everyone data was uploaded to client server as well as a main server and both version of the site could see what each other had uploaded. This shows the peer-to-peer network working and the artefact acting as intended.

With regards to GDPR and personal data there are usually three articles that come up that conflict with blockchain within GDPR. Article 15 – Right of Access, Article 16 – Right to Rectification and Article 17 Right to be forgotten (European Union, 2016). Right of access is upheld as to upload and download you have to have the right to access the blockchain, through this any files would be available it would just be a case of having the file key. In a real-life use case if anyone was to upload your information then the hash and file key would be given to the user as to not breach this. Right to be rectification is also upheld as the sole controller of node if a file requires rectifying then the chain can be rehashed or forked to a new chain. If more nodes were added that were not under my control, then it could become difficult for everyone to agree to fork to the new chain. Finally the Right to be forgotten, similarly to the right of rectification the chain can be forked and rehashed to not include that data. The files that are saved when uploaded can be deleted also so there is no data upon said user.

Overall, this project was largely successful in many ways including meeting requirements, following the development cycle, and creating a product usable and usable to its end users. Furthermore, the development and testing phases were accidentally and naturally carried out iteratively, revealing why it is becoming the industry standard to develop software in this way. Whilst there are many improvements that could be made especially if hosted live that would need to be made. The artefact shows that GDPR and blockchain can work together and friends and not enemies Nevertheless, this project resulted in a fully functioning artefact that functions as intended, whilst it may have felt more complete if a live web link was obtained in time rather than relying on local hosting alone, the aim and requirements of this project have been met.

7.1 Demonstration Video

Please follow this link to the demonstration video: https://youtu.be/ihz23hmzPjE

8. Reflective Analysis

Whilst all stages of this project have been critically analysed throughout, I will now take this opportunity to reflect on this process from a personal perspective, as the sole developer on the project. As expected, this project has been a challenge however, as the result of this project is a functioning website that meets the aim of the project, it is clear that it has been successful overall. However, this section will explore my opinions on what went well during this process, what could be improved ideas for potential future development

Key strengths of mine during this process proved to be determination and perseverance. As an unconfident programmer, coding an entire system from scratch for this solo project felt like an overwhelming task at times especially as the project uses a variety of languages it really challenged me. However, I feel like I learned a lot through experience throughout this process and I am now a stronger and more confident programmer, better equipped to hopefully begin a career within a Cyber/Information Security role. One success of this project was the very choice of project to begin with as blockchain is an emerging technology that will only get bigger. This, with the skills I have learned and experience I have gained has improved me in my skillset and cemented the career that I desire. Choosing programming languages that I have no doubts will be used in the future and one that I already knew well, has helped keep me motivated throughout the project. This project was not only completed in order to complete my final year at university and successfully graduate, but to also serve as a useful tool in my future to prove my capabilities to potential employers.

Time management I have previously talked about and whilst the Gantt chart timeline did not materialise in the end, the project was completed on time. As mentioned before making less ambitious goals and timeframes would greatly help and also using more project management tools such as Trello. As said in the conclusion, if a live web link was obtained for hosting, then it may have worked out even better however, due to time management this did not materialise. Project time management will be something I definitely look to improve upon going forward as it seems to have been a major weakness.

Overall, I am satisfied with my completion of this project and the product that I have developed. Although it wasn't done perfectly, the reasons for this have been carefully reflected upon and therefore I have learned valuable lessons through this process. With a year of hard work coming to an end, I can simply hope that this project can help me to achieve the 1st class honours degree I have worked hard towards.

9. References

Adobe Experience Cloud. N.d. *Waterfall Methodology* Available from: https://www.workfront.com/en-gb/project-management/methodologies/waterfall#:~:text=The%20waterfall%20methodology%20is%20a,detailed%20documentation%2C%20and%20consecutive%20execution. [Accessed 12 March 2022].

Almeida Teixeira, G., Mira da Silva, M. and Pereira, R. (2019), The critical success factors of GDPR implementation: a systematic literature review. *Digital Policy, Regulation and Governance*, 21(4) 402-418. Available from https://www.emerald.com/insight/content/doi/10.1108/DPRG-01-2019-0007/full/html [Accessed 23 February 2022]

Arora. S. (2022) What is PyCharm? Features, Advantages and Disadvantages. Hackr.io. Available from: https://hackr.io/blog/what-is-pycharm [Accessed 23 April 2022]

Australian Bureau of Statistics. N.d. Statistical Language – Quantitative and Qualitative Data. Available from: https://www.abs.gov.au/websitedbs/D3310114.nsf/Home/Statistical+Language+-+quantitative+and+qualitative+data [Accessed 30 April 2022]

Dooley. J. (2017). *Software Development, Design and Coding*. Apress, Chapters 1 & 2 pg.1-29.

Elgabry, O. (2016). UX - A quick glance about The 5 Elements of User Experience (Part 2). Medium. Available at: https://medium.com/omarelgabrys-blog/ux-a-quick-glance-about-the-5-elements-of-user-experience-part-2-a0da8798cd52 [Accessed 4 March 2022].

European Union (2016) Regulation (EU) 2016/679 of the European Parliament and the Council of 27April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data and repealing Directive 95/46/EC (General Data Protection Regulation) L119, 39-46. Available from https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679 [Accessed 10 February 2022]

Finck. M. (2019*) Blockchain and the General Data Protection Regulation: Can Distributed Ledgers be Squared with European Data Protection Law?* Brussels, Belgium: Scientific Foresight Unit (STOA) on behalf of the European Parliamentary Research Service. Available from https://www.europarl.europa.eu/RegData/etudes/STUD/2019/634445/EPRS_STU(2019)634445_EN.pdf [Accessed 10 November 2021]

Hayes. A. (2022) *Blockchain Explained.* Dotdash. Available from https://www.investopedia.com/terms/b/blockchain.asp [Accessed 28 February 2022]

Hoofnagle. C., van der Sloot. B., Borgesius. F. (2019) The European Union General Data Protection Regulation: What it is and What it Means. *Information and Communications Technical Law.* 28(1) 65-98. Available from https://www.tandfonline.com/doi/full/10.1080/13600834.2019.1573501 [Accessed 24 January 2022]

Interactive Design Foundation. N.d. *Interaction Design.* Available from: https://www.interaction-design.org/literature/topics/interaction-design#:~:text=Interaction%20Design%20(IxD)%20is%20the,output%20to%20suit%20precise%20demands [Accessed 24 February 2022]

Iredale. G. (2019) Introduction to Permissioned Blockchain. Available from https://101blockchains.com/permissioned-blockchain/ [Accessed 01 March 2022]

Ivanotech. (2021) What is Infura? Available from: https://academy.moralis.io/blog/infura-explained-what-is-infur [Accessed 02 April 2022]

Kaurlartz, M., van Kranenburg-Hanspians, K., Sanders, S., Domokos, M., Horvath, K., Runte. C., Kamps. M., Martin. B. (2019) The Tension between GDPR and the Rise of Blockchain Technologies, CMS Legal. Available from https://cms.law/en/int/publication/the-tension-between-gdpr-and-the-rise-of-blockchain-technologies [Accessed 8 November 2021]

Libguides.newcastle.edu.au. 2019. Libguides: Research Methods: What Are Research Methods? Available at: https://libguides.newcastle.edu.au/researchmethods [Accessed 21 April 2022].

Mohapatra, P. (2010). *Software engineering*. *A Lifecycle Approach* New Delhi: New Age International, Chapter 3 pg.63-92.

Moubry. M., Mackey. E., Elliot. M., Gowans. H., Wallace. S., Bell. J., Smith. H., Aidinlis. S., Kaye. J. (2018) Are 'pseudonymised' data always personal data? Implications of the GDPR for administrative data research in the UK. *Computer Law and Security Review.* 34(2) 222-233. Available from https://www.sciencedirect.com/science/article/pii/S0267364918300153#:~:text=There%20has%20naturally%20been%20a,forthcoming%20General%20Data%20Protection%20Regulation.&text=The%20definition%20of%20pseudonymisation%20under,this%20definition%20are%20personal%20data [Accessed 12 March 2022]

Pedamkar. P. n.d. Socket Programming in Python. Available from: https://www.educba.com/socket-programming-in-python/ [Accessed 18 April 2022]

Powell-Morse, A., 2016. Waterfall Model: What Is It And When Should You Use It? Airbrake Blog. Available at: https://airbrake.io/blog/sdlc/waterfall-model [Accessed 23 February 2022]

Sahu. M., (2020). How to Use Blockchain to Store Data. Upgrad Education. Available from https://www.upgrad.com/blog/how-to-use-blockchain-to-store-data/ [Accessed 02 February 2022]

Singh. V., 2022. Flask vs. Django in 2022. Which Framework to Choose? Hackr.io. Available from: https://hackr.io/blog/flask-vs-django#:~:text=Django%20is%20considered%20to%20be,to%20the%20ones%20in%20Django [Accessed 04 March 2022]

Software Testing Fundamentals. n.d. White Box Testing. Available at: https://softwaretestingfundamentals.com/white-box-testing [Accessed 06 2022]

Stackify. 2017. What Is SDLC? Understand The Software Development Life Cycle. Available at: https://stackify.com/what-is-sdlc/ [Accessed 23 February 2022].

Suripeddi. M. K. S. and Purandare. P. (2021) Blockchain and GDPR – A Study on Compatibility Issues of the Distributed Ledger Technology with GDPR Data Processing. *Journal of Physics: Conference Series* 1964. Available from https://iopscience.iop.org/article/10.1088/1742-6596/1964/4/042005/pdf [Accessed 07 February 2022]

Synotive. 2017. 10 Questions To Ask When Developing Software. Available at: https://www.synotive.com/blog/software-development-client-questionnaire [Accessed 13 March 2022]

Tama. B. A., Kweka B. J., Park. Y, Rhee. K. (2017) A critical review of blockchain and its current applications, In: 2017 International Conference on Electrical Engineering and Computer Science

(ICECOS), Palembang, Indonesia, 22-23 August 2017. IEEE, 109-113, Available from https://ieeexplore.ieee.org/document/8167115/citations#citations [Accessed 9 November 2021]

W3schools.com. N.d. What Is Javascript. Available from: https://www.w3schools.com/whatis/whatis_js.asp [Accessed 12 April 2022]

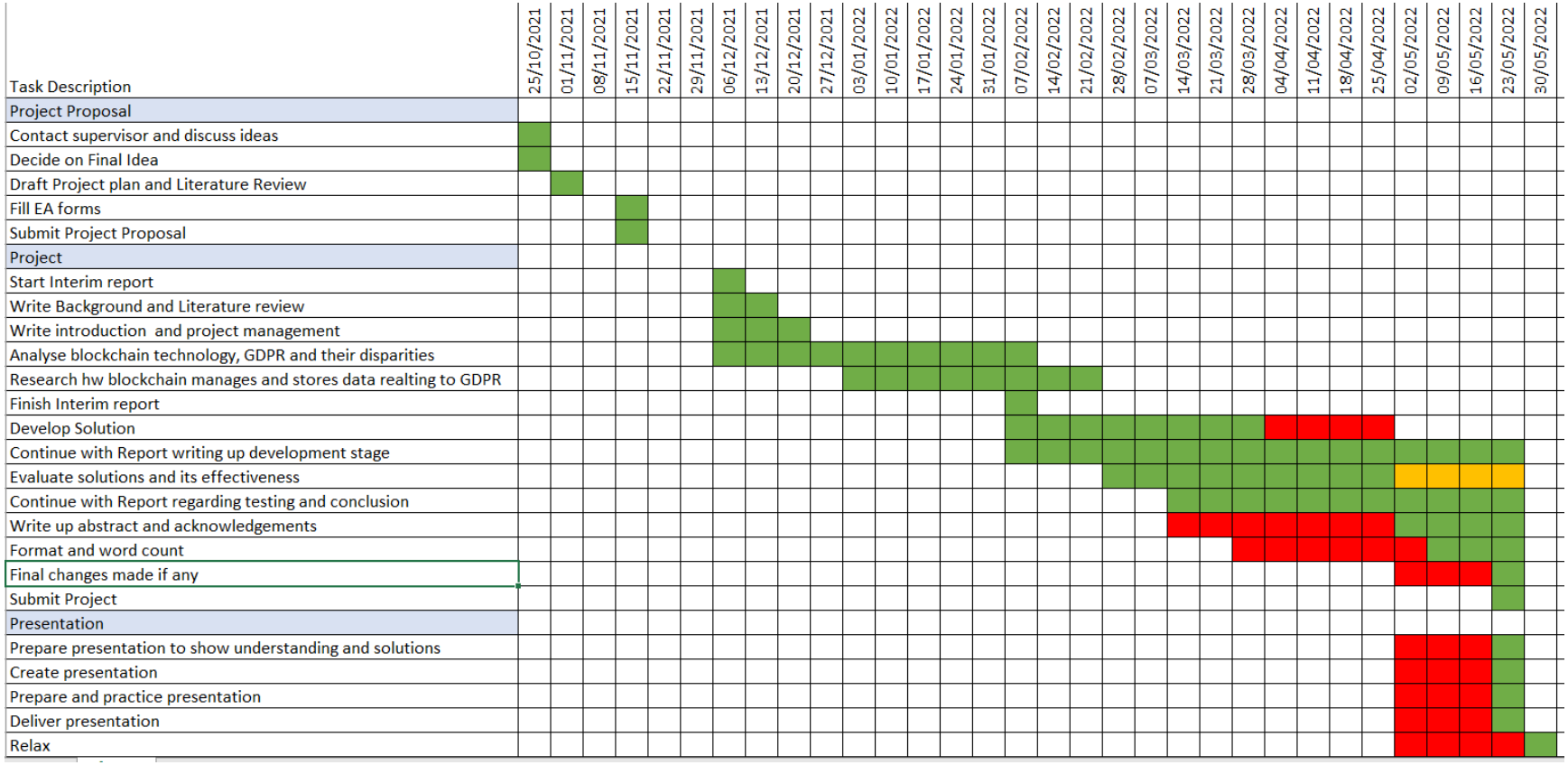
W3Techs. N.d. Usage Statistics of JavaScript Libraries for websites. Available from: https://w3techs.com/technologies/overview/javascript_library [Accessed 01 May 2022]

Walker. G. n.d. Learn Me A Bitcoin. Available from: https://learnmeabitcoin.com/technical/longest-chain#:~:text=The%20rule%20that%20nodes%20adopt,shared%20view%20of%20a%20file. [Accessed 15 April 2022]

Wallace. A., (2018) *Protection of Personal Data in Blockchain Technology.* Masters. Stockholm University. Available from http://www.diva-portal.org/smash/get/diva2:1298747/FULLTEXT01.pdf [Accessed 10 February 2022]

Wang, J., Wang S., Guo. J., Du. Y., Cheng. S., Li. X, (2019) A Summary of Research on Blockchain in the Field of Intellectual Property. Procedia Computer Science, 147(1) 191-197, Available from https://reader.elsevier.com/reader/sd/pii/S187705091930239X?token=DE804BBFDFB58A0BEA0C1C5EC79D6BB4F7BAD828D2B6D4BE6312EDCC3F9350FC2FBF6C514C97578F4EEBD94B737E7F1D&originRegion =eu-west-1&originCreation=20211115145437 [Accessed 9 November 2021]

Appendix 1


Task Description
25/10/2021
01/11/2021
08/11/2021
15/11/2021
22/11/2021
29/11/2021
06/12/2021
13/12/2021
20/12/2021
27/12/2021
03/01/2022
10/01/2022
17/01/2022
24/01/2022
31/01/2022
07/02/2022
14/02/2022
21/02/2022
28/02/2022
07/03/2022
14/03/2022
21/03/2022
28/03/2022
04/04/2022
11/04/2022
18/04/2022
25/04/2022
02/05/2022
09/05/2022
16/05/2022
23/05/2022
30/05/2022
Project Proposal
Contact supervisor and discuss ideas
Decide on Final Idea
Draft Project plan and Literature Review
Fill EA forms
Submit Project Proposal
Project
Start Interim report
Write Background and Literature review
Write introduction and project management
Analyse blockchain technology, GDPR and their disparities
Research hw blockchain manages and stores data realting to GDPR
Finish Interim report
Develop Solution
Continue with Report writing up development stage
Evaluate solutions and its effectiveness
Continue with Report regarding testing and conclusion
Write up abstract and acknowledgements
Format and word count
Final changes made if any
Submit Project
Presentation
Prepare presentation to show understanding and solutions
Create presentation
Prepare and practice presentation
Deliver presentation
Relax

Appendix 2

All coding that has been submitted into the supporting documents folder

Appendix 3

The PowerPoint presentation has been submitted into the supporting documents folder also

Appendix 4

Word Count: 12167