A DECENTRALIZED BIM DOCUMENT MANAGEMENT SYSTEM WITH BLOCKCHAIN AND IPFS INTEGRATION FOR CONSTRUCTION PROJECT DELIVERY

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

FURKAN KOÇ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
CIVIL ENGINEERING

SEPTEMBER 2022

Approval of the thesis:

A DECENTRALIZED BIM DOCUMENT MANAGEMENT SYSTEM WITH BLOCKCHAIN AND IPFS INTEGRATION FOR CONSTRUCTION PROJECT DELIVERY

submitted by **Furkan Koç** in partial fulfillment of the requirements for the degree of **Master of Science** i**n Civil Engineering, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Erdem Canbay
Head of the Department, **Civil Engineering, METU** ________________

Prof. Dr. Rifat Sönmez
Supervisor, **Civil Engineering, METU** ________________

Examining Committee Members:

Assist. Prof. Dr. Güzide Atasoy Özcan
Civil Engineering, METU ________________

Prof. Dr. Rifat Sönmez
Civil Engineering, METU ________________

Assist. Prof. Dr. Aslı Akçamete Güngör
Civil Engineering, METU ________________

Assist. Prof. Dr. Saman Aminbakhsh
Civil Engineering., Atılım University ________________

Assist. Prof. Dr. Furkan Uysal
Audit and Risk Management, Ankara SBÜ ________________

Date: 01.09.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Last name: Furkan Koç

Signature:

ABSTRACT

A DECENTRALIZED BIM DOCUMENT MANAGEMENT SYSTEM WITH BLOCKCHAIN AND IPFS INTEGRATION FOR CONSTRUCTION PROJECT DELIVERY

Koç, Furkan
Master of Science, Civil Engineering
Supervisor: Prof. Dr. Rifat Sönmez

September 2022, 102 pages

Construction project delivery has been evolving with digitalization of the industry. On the way of the digitalization, Building Information Modelling (BIM) is one of the leading developments such that most project deliverables are accomplished collaboratively among project participants through information models to successfully design and manage construction. The management of project deliverables in collaboration necessitates a common data environment in which each participant can access the permissioned project deliverables at any time, which means a network built on the web. In this manner, construction projects have begun to be managed in cloud-based document management systems (DMS). DMS helps the completion of a project systematically by providing an environment for storage, retrieval and management of the project documents. Despite its irreplaceable return, DMS constitutes remarkable issues related with security, serviceability and cost. Blockchain technology with its features of decentralization, transparency, immutability and security is well-matched with the issues of DMS. Thus, this study aims to develop a decentralized document management system (DDMS) with the integration of Ethereum blockchain which is a public blockchain and Interplanetary

File System (IPFS) for construction project delivery by deploying a smart contract. A decentralized application is designed for the proposed system to allow the participants to use it with ease while blockchain ensures immutability of document exchanges and IPFS stores the document on the distributed network securely. The contributions and limitations of the DDMS are demonstrated and discussed through a Metro Line project.

Keywords: Blockchain, Decentralized Storage, Information Security, Document Management, Building Information Modelling

ÖZ

İNŞAAT PROJE TESLİMLERİ İÇİN BLOKZİNCİRİ VE IPFS ENTEGRASYONLU MERKEZİYETSİZ BIM DOKÜMAN YÖNETİM SİSTEMİ

Koç, Furkan
Yüksek Lisans, İnşaat Mühendisliği
Tez Yöneticisi: Prof. Dr. Rifat Sönmez

Eylül 2022, 102 sayfa

İnşaat proje teslimleri sektörün dijitalleşmesi ile değişim geçirmektedir. Dijitalleşme yolunda Yapı Bilgi Modellemesi (BIM) öncülük eden gelişmelerden birisidir; öyle ki yapıyı başarıyla tasarlamak ve yönetmek için çoğu proje teslimi, bilgi modelleri üzerinden proje katılımcıları arasında iş birliği halinde tamamlanmaktadır. Proje teslimlerinin işbirliği içerisinde yönetilmesi, her paydaşın izinli olduğu belgelere her zaman erişebildiği ortak bir veri ortamını gerekli kılmaktadır; bu da Web üzerinde kurulmuş bir ağ anlamına gelmektedir. Bu bağlamda, inşaat projeleri bulut-tabanlı doküman yönetim sistemlerinde (DMS) yönetilmeye başlamıştır. DMS, proje dokümanlarının saklanması, alınması ve yönetilmesi için bir ortam sağlayarak projelerin düzenli bir şekilde tamamlanmasını sağlar. Yeri doldurulamaz getirilerine rağmen, DMS; güvenlik, hizmet ve maliyet açısından kayda değer sorunlar teşkil etmektedir. Merkeziyetsizlik, açıklık, değişmezlik ve güvenlik özellikleriyle Blokzinciri teknolojisi, DMS sorunlarıyla tam eşleşmektedir. Böylelikle, bu tez inşaat proje teslimi için halka açık bir blokzinciri olan Ethereum Blokzinciri ve Interplanetary File System (IPFS) bütünleşmesi ile akıllı sözleşme hazırlanarak merkeziyetsiz bir doküman yönetim sistemi (DDMS) geliştirmeyi hedeflemektedir. Arkaplanda blokzincir doküman paylaşımlarının değişmezliğini ve IPFS dokümanların dağınık bir ağda güvenle saklanmasını sağlarken, katılımcıların önerilen sistemi kolaylıkla kullanabileceği merkezi olmayan bir uygulama tasarlanmıştır. DDMS'nin katkıları ve kısıtları, bir Metro Hattı projesi üzerinden işaret edilmiş ve tartışılmıştır.

Anahtar Kelimeler: Blokzinciri, Merkeziyetsiz Depolama, Bilgi Güvenliği, Doküman Yönetimi, Yapı Bilgi Modellemesi

Dedication to my family

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisor Prof. Dr. Rıfat Sönmez for his revolutionary ideas, guidance, advice, criticism, encouragements and insight throughout the research. The author would also like to thank Mr. Salar Ahmadisheykhsarmast for his comments and helps to develop the system.

Further, the author is very grateful to his beloved wife, Mrs. Hale Nurefşan Koç, for both technical and moral support to be able to accomplish the research.

The author would like to extend his sincere thanks to Deputy Director General Serhat Kaş for his guidance, encouragements and provision of a suitable research environment and his teammates in Yüksel Proje for their sacrifice.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS

ABE	Attribute-Based Encryption
AEC	Architecture Engineering and Construction
API	Application Programming Interface
ASP	Application Service Provider
BIM	Building Information Modelling
CAD	Computer Aided Design
CDE	Common Data Environment
CID	Content Identifier
dApp	Decentralized Application
DDMS	Decentralized Document Management System
DLT	Distributed Ledger Technology
DMS	Document Management System
DDMS	Decentralized Document Management System
ECID	Encrypted Content Identifier
EDMS	Electronic Document Management System
IPFS	Interplanetary File System
ISO	International Organization for Standardization
P2P	Peer to Peer
PoA	Proof of Authority

PoS	Proof of Stake
DPoS	Delegated Proof Stake
PoW	Proof of Work
PSI	Publisher System Information
RFI	Request for Information
RFID	Radio Frequency Identification
RSI	Reviewer System Information

CHAPTER 1

INTRODUCTION

The delivery of a construction project is a sophisticated endeavor to manage and fund design, construction, operations and maintenance phases of the project pursued with the contribution of many parties that can be geographically dispersed. Project delivery methods are descriptive of the parties' roles and responsibilities together with the determination of an implementation plan to organize design, procurement and construction. There exist several project delivery methods such as Design-Bid-Build, Design-Build, etc. Each method is possessed of idiosyncratic pros and cons; hence, none of them is well-suited for all types of projects (Gordon, 1994). Project delivery methods have an unsparing impact on how project deliverables are managed by the parties involved in the project.

A deliverable is an output as part of the project. The construction deliverables include all documents and information created for the fulfillment of the project's objectives such as site investigation report, schedule, calculation reports, design manuals, Request for Information (RFI), design models and drawings, progress report, specifications, bill of quantities, material take-offs, shop drawings and so on. The deliverables are grouped as internal and external deliverables. The former are documents exchanged within a party for internal coordination while the latter are document delivered by the parties to each other. Considering the fact that a complex construction project might necessitate thousands of documents to be produced, revised and disseminated among parties, the process of document and information share should be managed in a well-organized manner. Hoła et al. (2014) and Tserng et al. (2004) indicate that a construction project can be prospering with efficient information management and systematical documentation respectively.

For a long time, the deliverables are designed and organized with the conventional method in construction industry, i.e. manual drafting of design documents such as plans and sections and archiving them in a physical room. They are organized by paper-based document management systems. Along with the implementation of Computer Aided Design (CAD), which emerges at the last half of the twentieth century, the deliverables are undergone a revolutionary transformation with digitalization. CAD provides opportunity for engineers and architects to design and draft buildings on computers with software instead of drawing and calculating manually. Then, the documents are started to be produced in computers and shared in a digital form such as diskettes and e-mails (Björk, 2006). The circulation of documents among project participants results in the complication in terms of organization of documents. As a solution for this, the next step on the way of digitalization becomes electronic document management systems. (Since the deliverables are the documents that are delivered to the responsible party, the terms of deliverable and document are used interchangeably.)

EDMS or simply DMS is a platform that stores documents in a secure manner, enables all participants to access valid documents from the single source under authorization (Ahmad et al, 2017) and coordinates documents with metadata, which is an information attached to documents such as the author of a document in order to facilitate document search and retrieval processes. Kao et al. (2013) indicate two purposes of the implementation of EDMS to construction industry. The purposes are availability of the data from anywhere at any time and convenience of sharing and collaboration. In addition to the availability and the convenience for project deliverables, EDMS brings about reduction in the use of paper-based documents which provides considerable benefits for organizations in terms of cost and for environment. Krishnan and Subramanian (2015) focus on “green-ness” of DMS and figure out the utilization of DMS contributes to remarkable decrease in the carbon footprint. Considering the number of paper-based documents produced in the construction industry, which is the one of major industries possessing of higher carbon footprints (Labaran et al, 2021), eco-friendly feature of DMS might be crucial for the future of the industry and the world.

Due to its benefits, there are numerous researches about electronic document management systems. Caldas and Soibelman (2003) propose a document management system by automatizing classification of unstructured documents by eliminating manual assignment of metadata. Bodhuin and Tortorella (2007) aim to build a document management service facilitated by RFID technology so that documents with legal value can be detected, linked and tracked. Konishi et al (2007) focus on digitizing handwritings and integrating them with paper and electronic documents. Fuertes et al. (2007) develop an ontology that structures a hierarchy such that data of documents are incorporated to documents for classification. Shehab et al. (2009) propose a barcode integrated system to increase productivity, decrease cost of document management and save time for storing data. Moon et al. (2018) focus on text mining; the developed prototype of DMS for international contracts to gather and organize the newest information in order to automatically create document information.

Early electronic document management systems are located on a dedicated network which is only for the use of an organization. Along with the enlargement of the Internet, the organizations start to transfer the systems to the Internet. Chassiakos and Sakellaropoulos (2008) reveal that information technology corporations provide commercial web-based document management systems for construction industry. This means the involvement of a third party, which named as Application Service Provider (ASP) to a project. ASP provides a service for document storage and management. Considering construction industry's fragmentation in nature, there are multiple stakeholders from different continents for multiple project at the same time. Thus, well-organized document management technique is necessary among the stakeholders for complex construction projects. In order to find a solution for sharing, storing and tracking large-sized documents, cloud computing technology is offered for the stand-alone systems or local networks (Chen, 2016). Cloud computing technology provides a network on the Internet for users to store files such as images, videos, documents etc. For managing construction documents in an environment that might be in local network or in a cloud, a systematic document sharing and storing schemes must be determined. Saraiva and da Silva (2009) develop a DMS that is designed with a component-based Content Management Systems providing document management and storage. Leukel, et al. (2011) propose a semantic DMS for detecting connections among documents by extending metadata. Parasuraman, et al. (2014) proposed a cloud based DMS to ensure integrity and security of documents by assigning a controller outside the organization.

As well as the systems designed in the literature, commercial document management systems are also developed extensively in industry for general purposes. The prominent document management platforms providers are listed as OpenText, Ricoh, Xerox, Oracle, Canon, Alfresco and IBM in Verified Market Research (2022). There are many platforms other than the prominent ones. Grover and Froese (2016) indicate that not all DMS's do conform with construction industry due to its dispersed and temporary cooperation for a project. In construction industry, the most common DMS's are Autodesk Construction Cloud and Oracle Aconex such that Aconex (2022) hosts more than 4 million projects and initiatives with more than 1.8 billion document exchanges. They are designed such that they meet the requirements of the industry such as project management, process management, issue management, etc. Apart from publishing, storing and tracking documents, these platforms provide displaying opportunities to examine the documents without opening a document on the related software. Further, the construction cloud enables participants work together in the same BIM model or a document simultaneously and collaboratively. This brings a new dimension for collaboration and document management to facilitate BIM-based design.

Although the DMS's provide irrefutable benefits for project management, it possesses critical threats for construction projects. Cloud-based DMS require an intermediary, ASP involvement to the project. All documents produced for a project are stored in ASP's servers such that the system is based on a trust to the third party. Even though there are agreements between ASP and parties such as confidentiality agreement, there might be situations invalidate the agreements and break the trust. For example, in case of the existence of compelling reasons like a war, the agreements may be invalid as experienced in Russia such that Oracle halts all its services in the country (Marcus, Poitiers and Weil, 2022). Even if there is not a compelling reason, the providers may abuse the project, which might be confidential such as military project, etc. through the back door on behalf of interest of them or their nations. These issues bring about doubts about trust for cloud-based document management system.

The centralization of storage may also pose a risk in terms of security and serviceability of the system. Cloud-based systems or any other document management system on the web is under threat of cyber-attack. McAfee (2019) which is a company providing security software published cloud adaptation and risk report clarifying that an ordinary company encounters nearly thirty threats monthly due to cloud-based storage. Due to the threats and the centralized structure, a probability of single source of failure exists for all project documents. The other threat on the systems is internal manipulation. In cloud DMS, there are admins responsible for managing project documents, participations and permissions. Das, et al. (2021) state that an admin or a participant can alter the document information or even delete critical documents deliberatively or accidentally like BIM models, which results in interruption of project delivery or even redesign. This is an important issue considering the dependency on an admin without a consensus in the project. Since the admin of the system which is generally from client has full authority to manage project documents. Thus, the rights of contractor and other parties might be under control of client. This demonstrates the need for a well-organized document management system protecting each party’s rights fairly and the documents’ permeance.

Since the DMS works on centralized servers, the data comes from the single source. The centralization causes not only a risk of single source of failure but also interruptions for project due to serviceability disruptions. Any service delay due to maintenance, renewal, update and even partial or complete failure on the system directly influence the project delivery durations; therefore, the project itself. In 2021, Amazon Web Services (AWS), which is cloud computing services of Amazon faced with a service failure affecting millions of users in large variety of industries (Giles, 2022). This evokes concerns about the continuity of service of cloud-based storages. In addition to service outages, there might be delay in the services due to excessive amount of participation at the same time. All in all, service quality may decrease or even service may fail because of aforementioned reasons, which affects the completion of a construction project negatively. The other issue related with the systems is cost. Document management systems require high cost of capital such that some of construction companies cannot afford. Therefore, they either cannot manage documents in DMS or use DMS by purchasing restricted number of accounts (Fernando et al, 2019). Thus, there must be alternative solution for DMS other than cloud-based ones to get rid of the issues.

With the adaptation of digital technologies to construction industry such as BIM and DMS, construction projects have started to be designed and shared by facilitating these technologies. The digitalization brings about some security issues in the industry. Confidential construction projects such as governmental buildings, industrial plants, hospitals, banks, prisons and other special buildings such as the ones designed for defense industry require full confidentiality in design and construction in order to protect any data leakage. There are specifications focusing on the security risk management to prevent alteration, misuse and corruption of confidential information. ISO 19650-5:2020 is one of them which designates requirements of security for confidential information within a built environment. In addition to the specifications, countries take distinctive precautions such as data localization to ensure security of data. Data localization is one of the prominent precautions for data residency. It means that the data is created and kept within certain borders due to security concerns (Taylor, 2020).

In General Data Protection Regulation (GDPR), European Union restricts the processing of even personnel data within its borders (Mishra, 2016). Similarly, Presidency of the Republic of Türkiye, Digital Transformation Office (2019) introduced a circular about information and communication security precautions that includes the condition of storing critical information such as the ones of population and health domestically to avoid data leakage. However, private companies even prefer not to store confidential project information in any server other than theirs. As a result of using internal solutions, confidential project might not be managed efficiently due to the absence of a common data environment.

Blockchain, which is a type of Distributed Ledger Technology (DLT), is a single digital ledger composed of chained transactions and distributed among the network participants with security measures ensured by cryptography in order to provide transparency, immutability, trust and security. Cloud-based document management systems suffer from aforementioned issues related with trust, security, serviceability, cost and confidentiality. Blockchain technology is facilitated in DMS a solution for the issues. Tao et al. (2022) state that blockchain ensures data reliability and integrity with decentralization, traceability and immutability of stored data. Although, document exchanges are recorded irrevocably with distributed and encrypted structure of blockchain, the blockchain is not suitable for storing documents for several reasons including block size limitation and costs (Sonmez et al, 2020). In this manner, decentralized file storage systems are developed to securely store documents. IPFS is one of that systems. IPFS makes document distributed through its network to prevent single point of failure and protect documents.

Considering the match of the issues of cloud-based DMS and the solutions provided by blockchain technology as shown in Figure 1.1, this study aims to develop a decentralized document management system with the integration of public blockchain and IPFS by deploying a smart contract in Ethereum in order to securely and immutably store project documents and transactions for construction project delivery. The proposed system consists of three subsystems namely Ethereum Blockchain, Interplanetary File System and Decentralized Application. The system is designed in a decentralized manner in order to prevent single source of failure of project documents in centralized storages. Blockchain is leveraged for tracking document publish immutably with its distributed network. IPFS is used to store documents in decentralized storages. The system has a display by decentralized application for project participants to publish, track and review documents without getting lost in the system details. In the end, the system is used in a project as a case study to present the functionality of the management system.

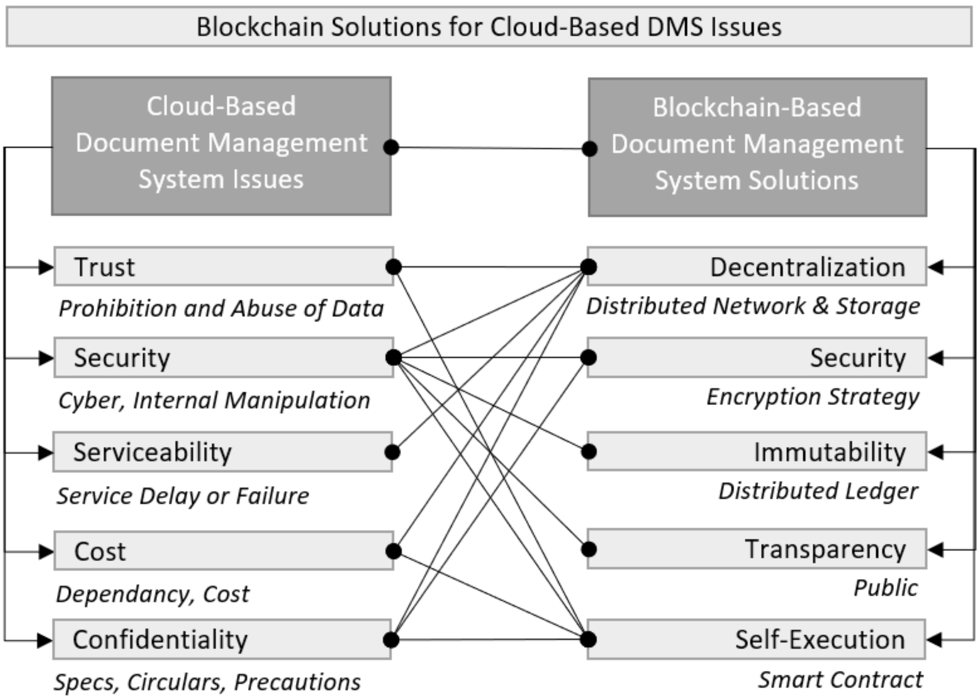


Figure 1.1 Blockchain Solutions for Cloud-Based DMS Issues

The paper is designed such that Chapter 1 introduces what is construction project delivery and project deliverables, how project parties are managed the deliverables for design and construction, what risks exist in the common method in order to identify the gap, followed by the introduction of the proposed system. Chapter 2 includes more detailed definitions for project design and delivery, then consists of the review of literature by focusing on document management systems and decentralized solutions in industries and specifically in construction industry.

Chapter 3 is the source of information for the leveraged technologies such that it explains blockchain, decentralized file storage system and smart contract together with a decentralized application comprehensively. Chapter 4 is for the description of proposed system with the implementation for real-life project as a case study. Chapter 5 consists of the comparison of the common method and proposed system in terms of cost and throughput. In Chapter 6, the proposed system is discussed in detail to demonstrate the benefits and limitations. Finally, Chapter 6 contains the conclusion and future research direction.

CHAPTER 2

LITERATURE REVIEW

2.1 Building Information Modelling

For decades, construction projects have been designed through Computer Aided Design (CAD). CAD provides opportunity for engineers and architects to design and draft buildings on computers with software instead of drawing and calculating manually. It can be used for producing two- or three-dimensional outputs depending on the demand for an industry. In AEC industry, CAD generally refers to two-dimensional drafting for building design. Construction project includes many disciplines; architecture, structure, mechanical, electrical and piping. Even all disciplines might be from different organizations. Maurer (2010) states that grand construction projects may need interorganizational partnerships. Especially in this case, it is burdensome to achieve collaboration among disciplines through two-dimensional architectural, structural, mechanical, electrical and piping drawings to create the project without errors. For the sake of better project management and delivery, the industry has started to substitute Building Information Modelling for Computer-Aided Design. Building Information Modelling (BIM) is one of the leading developments in Architecture, Engineering and Construction industry such that it is three-dimensional representation of a building incorporating graphical and non-graphical information within the body of the model in order to provide project stakeholders to plan, design and construct collaboratively for the life cycle of project. Kubba (2017) remarks that while CAD comprises of only graphical elements such as line, circle, etc. representing building components, BIM incorporates graphical and functional characteristic of a building. A model consists of schedules, plan, section and three-dimensional views, sheets as shown in Figure 2.1.

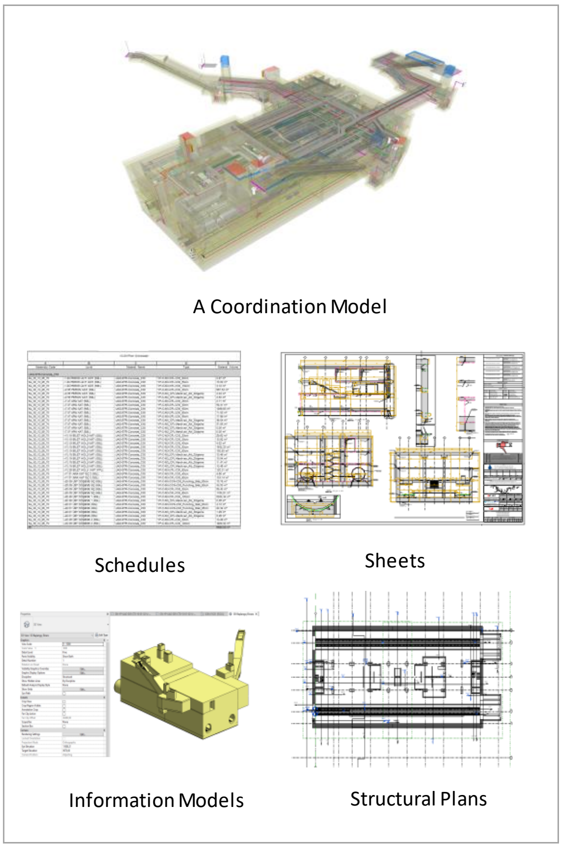


Figure 2.1 An BIM Coordination Model and Outputs of an Information Model

The models created have been used widely in building design, construction and operation phases. The effective adaptation of BIM to the phases results in project completion on time and within budget (Haron et al., 2010). The utilization of BIM can increase the efficiency in construction industry by providing accurate cost estimations, three-dimensional models (they are named as families for elements) together with detailed fabrication drawings, construction sequencing by considering spatial and timewise requirements and limitations as well as reports showing the clashes between different discipline models so that the problems can be resolved before the construction has started on the site (Azhar, 2011). Hussein et al. (2022) conduct a survey for the determination of merits and demerits of the BIM. Due to its unique advantages of Building Information Modelling, it is stated that BIM helps enhancing the quality of work, following the project schedule and cost and providing client satisfaction.

Beyond its great opportunities to manage project effectively and to visualize and analyze in three-dimension, BIM provides information database under single model that can be shared among participant. The information models created within the scope of BIM should be stored and published systematically such that each project participant can reach the latest and valid version of information models. It is not challenging to follow the different versions of documents and store that documents in a single organization. Considering construction industry's fragmentation in nature, there are multiple stakeholders even from different time-period for multiple project at the same time. Thus, well-organized document management technique is necessary among the stakeholders for complex construction projects. In order to find a solution for sharing, storing and tracking large-sized documents, cloud computing technology is offered for the stand-alone systems or local networks (Chen, 2016).

Cloud computing technology provides a network on the Internet for users to store files such as images, videos, documents etc. For managing construction documents in an environment that might be in local network or in a cloud, a systematic document sharing and storing schemes must be determined.

ISO 19650 organizes the way collecting, managing and distributing documents with a controlled mechanism under the term of Common Data Environment (CDE). CDE provides a single source of information stored in the shared area under four stages: work in progress, shared, published and archive. CDE is a database of outputs of a project through design, construction and operation phases. Figure 2.2 demonstrates that each participant might be confused with the validity of the data due to complicated and unsystematic data exchange in traditional information sharing method. On the other hand, CDE is the center of information that each participant can access the common data. Losev (2020) points out CDE is regarded as electronic document management system for the life cycle of buildings.

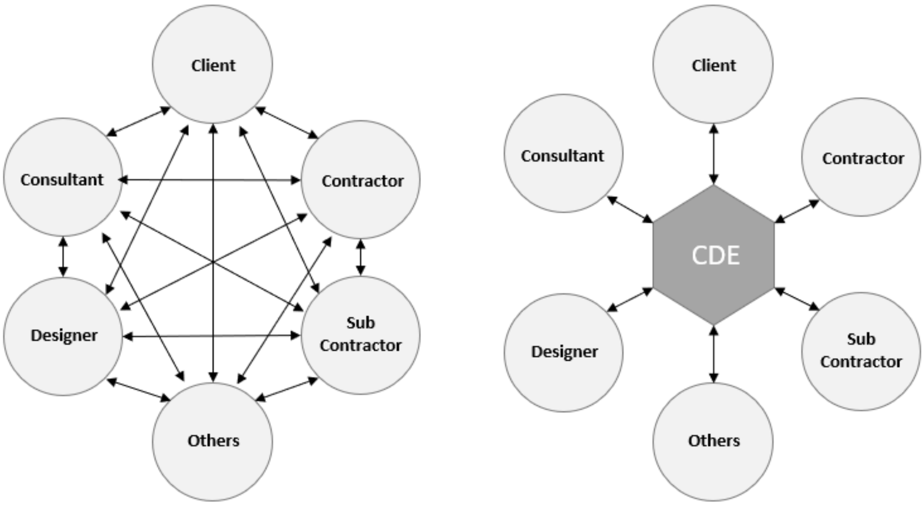


Figure 2.2 Traditional Information Sharing vs. Common Data Environment

2.2 Document Management System

Document Management System is the use of a computer system and software to store, manage and track electronic documents. They are mostly worked through cloud-based servers. In AEC industry, document management systems have been widely used with the adaptation of BIM-based design since BIM necessitates the coordination and collaboration of disciplines. This can be ensured by working collaboratively in a common environment.

Guo et al. (2019), Wormer et al. (2017) and Kao et al. (2013) state that document management system should have the following elemental features:

i. Document Approval Processes. Approval workflows for documents are designed in conformance with project processes such that the relations between project participants and permissions of them are defined in document management system properly.

ii. Document History. There should be a recordkeeping for documents in order to inspect document's confidence in necessary cases such as corruption of the document.

iii. Document Version Control. The latest version of documents together with all previous version should be stored in a systematical way. It should be noted that according to Perforce, around 90% of workers' waste time for problems related with versioning. This reveals the importance of a well-designed document version mechanism in DMS to save project time.

iv. Broad Access Opportunity. The data stored in DMS can be accessible from anywhere at any time with a basic structuring. Anyone can be able to use DMS.

Fernando et al. (2019) define document management systems as the coordination of information among project participants in order to let the participants spend less time and effort in storing, revising, retrieving and publishing documents. It is critical for a DMS to accelerate project delivery. Further, they remark the benefits of DMS as follows; decrease in the time needed for retrieving documents, acceleration of project completion, better coordination with other disciplines, availability of accurate and current information and increase in effectiveness with the components of DMS demonstrated in Figure 2.3.

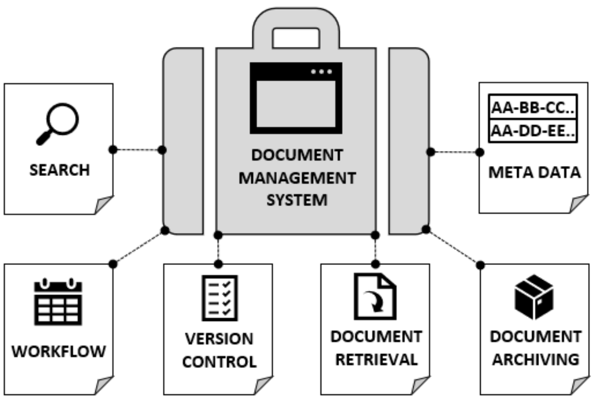


Figure 2.3 Document Management System Components

Although DMS provides irrefutable benefits, it possesses some disadvantages. The most distinct disadvantages of DMS are the dependency and security. The dependency to service provider brings about uncertainty in pricing. Security is another concern due to storage of documents in central servers. Ahmed et al. (2017) state that an obstacle for the adaptation of DMS is the necessity of capital investment in terms of time, cost and effort. All in all, the benefits prevail the disadvantages and the implementation of document management systems spread along the industry.

There are many electronic document management systems proposed in the literature. Fuertes et al. (2007) develop an ontology that structures a hierarchy such that data of documents are incorporated to documents for classification. Shehab et al. (2009) propose a barcode integrated system to increase productivity, decrease cost of document management and save time for storing data. Moon et al. (2018) focus on text mining; the developed prototype of DMS for international contracts to gather and organize the newest information in order to automatically create document information. Along with the development in the literature, many commercial applications are developed for document management such from Dropbox.

The prominent document management platforms providers are listed as OpenText, Ricoh, Xerox, Oracle, Canon, Alfresco and IBM in Verified Market Research. There are many platforms other than the prominent ones. However, some of them such as IBM are beneficial for the specific industries such as manufacturing and information technology, the suitability is a must for the implemented industry due to fragmentation structure of AEC industry (Das et al. (2022). The major platforms used in AEC are Autodesk Construction Cloud (formerly named as BIM360 Docs.) and Oracle Aconex, which are shown in Figure 2.4.

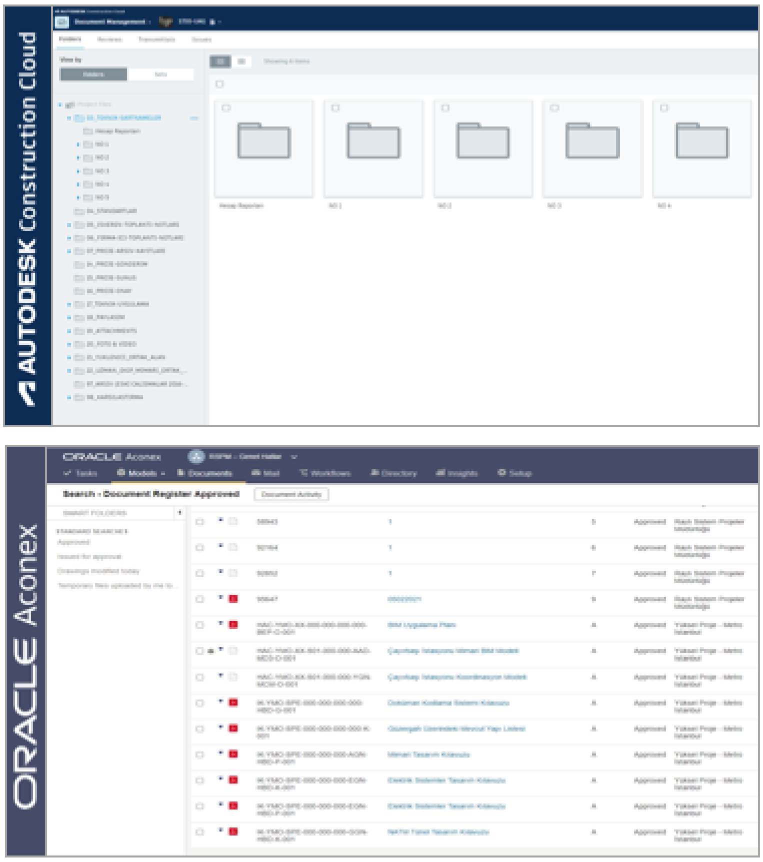


Figure 2.4 The user interfaces of Construction Cloud and Aconex

Apart from publishing, storing and tracking documents, these platforms provide displaying opportunities to examine the documents without opening the related software. Further, the construction cloud enables participants work together in the same BIM model simultaneously and collaboratively. This brings a new dimension for document management to facilitate BIM-based design.

Document management systems have implemented in various industries with its unique benefits. However, there are some issues about DMS to be improved and solved. As the new technologies emerge, DMS continues to evolve until the most optimum solution can be obtained. In this manner, DMS platforms are redesigned with blockchain technology to be decentralized in order to achieve more secure, transparent, immutable and traceable solutions.

2.3 Decentralization of Document Management System (DDMS)

Document management system is getting decentralized in two parts. The first one is to decentralize the way of tracking documents with the use of blockchains such as Ethereum, Hyperledger Fabric, etc. Blockchain used in the system might be public, private or consortium. The second one is the decentralization of storage by leveraging decentralized file storage systems such as IPFS, Filecoin, Storj, Swarm, etc. Both cases must be satisfied for obtaining decentralized document management system. There are several researches about tracking the documents by using blockchain and storing them in centralized servers. The researches satisfying the conditions for DDMS are reviewed in the following section. The ones adopted for document tracking are also examined thoroughly even if they do not have decentralized way of document storage. The reviewed researches are summarized in Figure 2.5. by separating the researches according to blockchain and file storage systems. The cell enclosed with red line is this research's focal point in which there exists limited number of researches.

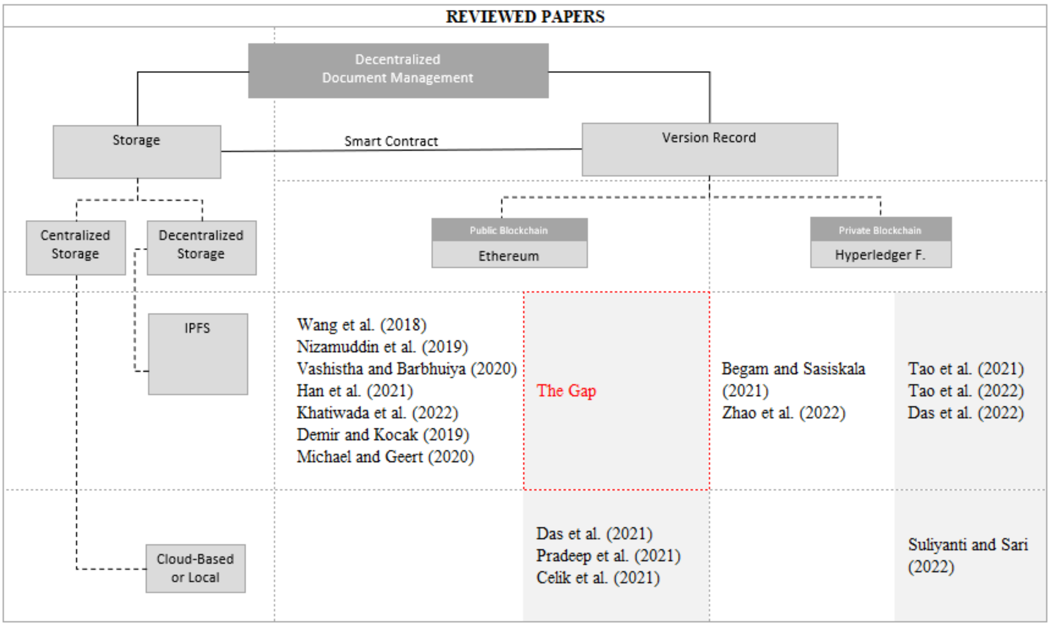


Figure 2.5 The summary of Reviewed Papers

2.3.1 Decentralized DMS in Literature

Decentralized document management systems aim to remove the centralization and ensure security. This necessitates a decentralized access control for users. Many approaches are developed for DDMS to construct and manage permissions through smart contracts.

Cloud storage systems employ attribute-based encryption (ABE) for insuring security, privacy and authorization. Begam and Sasiskala (2021) note that ABE can bring these features only if servers of cloud storage system are supplied by a trusted party. Not only the trusted supplier is enough, but also the private key creator for ABE system must be trusted. Wang et al. (2018) point out that the key creator can access to stored data in the cloud and may abuse the system, which poses a threat to security and privacy. Furthermore, trust for supplier cannot be fully ensured in cloud storage systems working in centralized servers. Therefore, they propose a decentralized storage (IPFS) and sharing system with attribute-based encryption. In this system, there are two parties; data owner and data user. Data owner sets the system and creates the key for the user. The key is embedded into an Ethereum blockchain as a transaction. This makes keys unalterable. Moreover, the system is promoted with the addition of keyword search algorithm.

Han et al. (2021) indicate that digital document is under the risk of being manipulated due to security vulnerability. They propose the distribution of access permissions for sensitive documents. Like the distribution keys into the blockchain for ABE schemes, the proposed DDMS creates a key for the document, breaks into parts and distributes the key via secret sharing scheme into public blockchain. When a user retrieves the document, all keys are gathered from each user with the distributed keys by smart contract. Smart contract detects permission of users. This brings high security to the system such that each user comes to an agreement to share the document.

The distribution of keys is essential for access control in DDMS, however as mentioned before it is crucial to keep documents in a decentralized network for the sake of achieving absolute decentralization. In this manner, Khatiwada et al. (2022) combine Ethereum blockchain, IPFS and ABE. For data publish and access control, hash and session key-dependent ABE (HS-ABE) is developed for the system. It encrypts the data and keep it in the cloud with attributes so that the cloud service provider cannot reach it. IPFS is considered as a cloud system in this framework. A blockchain is created for constructing and maintaining access policy.

Nizamuddin et al. (2019) propose a framework for publishing document and keeping track of versions. In the proposed framework, there are two parties namely developers and approvers whose addresses are defined by Ethereum blockchain. The relations between the parties are governed by smart contract. This framework is well-organized for a document management system. It is clear that the system must have appropriate workflows in order to be applicable for a specific industry.

2.4 DDMS Applications in Construction Industry

Decentralized document management implementation is closely related to the industry. In construction industry, there might be several parties involved in a project and complicated connections exists among them. Furthermore, design and review processes for the industry should be planned and organized to successfully track and manage documents. As mentioned beforehand, construction industry has benefited from building information modelling methods for improved management of information. BIM is generally managed through centralized databases, which causes a threat on confidentiality and accessibility.

The implementation of blockchain to cloud-based systems can prevent possible risks of confidentiality, accessibility and safety. Celik et al. (2021) state that monitorability of information models and coordination between project parties have to be ensured especially for the critical projects such as infrastructure projects and integrate BIM and blockchain technology in order to accelerate the automation in construction and provide transparency for the projects. Blockchain is used for displaying participants and their proceedings to diminish conflicts about responsibilities and assignments and securing project environment with its traceable and immutable structure. The framework is established with Ethereum blockchain and a cloud computing infrastructure. The smart contract, which is built in Ethereum, is designed to update design revisions. All in all, the framework is developed for enhancing cloud-based storage in terms of security, transparency and service availability. As a next step on the way of securing project documents, the following researches focus on the decentralization of storage.

Common Data Environment (CDE) is structured by ISO 19650 standards and it is described as an environment that stores, manages and transmits project documents including all graphical and non-graphical information at the life cycle of a project. In other words, it can be evaluated as a document management system. The environment is developed for communion and management of BIM models. However, current collaborative design platforms provide service through centralized servers, which means that it gives full authority to the service provider. Therefore, Tao et al. (2021) come up with a concept of distributed common data environment (DCDE) for collaborative design with BIM by integrating blockchain and IPFS. DCDE functions in the following order; a member uploads design files to IPFS, gets the uploaded files' CID, which is content identifier (a hash value), and records that CID to blockchain. IPFS networks work separately. Then, another member can download the files with the help of the CID and be the second provider of that files. Then client can reach design files from both provider and change the status of files in the end. It means that only the project members have the ledger and IPFS database. This is mainly due to the fact that the paper leverages a private blockchain, Hyperledger Fabric. The proposed framework is tested for three phases; initiation of design by uploading design files, performing collaborative design among project members and completion of design by changing status of model. The results for latency, throughput and storage cost are evaluated as acceptable.

After proposing a decentralized common data environment in 2021, Tao et al. (2022) indicate that the adaptation of blockchain to BIM-based collaboration processes needs accessibility arrangements. Therefore, they study a framework that integrates blockchain and IPFS with access control for BIM-base design collaboration. Like the previous study, there are two decentralized networks; one for storing large size of files, and the other for tracking the information exchanges. A consortium blockchain, Hyperledger Fabric, is selected for the framework owing to its permissioned architecture and fast transaction rates. In order to provide access control model, BIM models are divided as confidential and non-confidential parts. The confidential BIM data is encrypted in blockchain such that only permissioned members can reach that parts of the model. To be able to collaborate divided models, new design strategies are developed in the framework. For coordinating sensitive parts of a model, there exist two hash values; CID and ECID. While ECID is distributed to all members, only permissioned members are able to decrypt CID to download file from IPFS. Considering all processes defined in the framework, total latency is measured as approximately 0.1s. That is interpreted as negligible in design collaboration. On the other hand, total storage of the framework is 144 KB/day including blockchain and updated models' costs, which is also seen as acceptable.

The divisions of models to separate parts in sake of confidentiality might complicate the way that project is managed. If a part of model is sensitive, it is reasonable to assume that the model is sensitive completely since the party caring with design works of the divided parts of model is practically the same. Thus, an access control method should be applicable for entire documents in order to protect project's rights.

Das et al. (2022) propound a document management system providing document version integrity. The system utilizes blockchain for change recording, smart contract for approval processes and IPFS and Merkle Patricia Tree for versioning. However, a decentralized method for document storage is not leveraged for the framework, the preference of storage is up to client's choice. Only permissioned nodes can access to all of the transactions since the network functions through the private blockchain. Version integrity is supplied by using document coding, which is taken from Omniclass Classification System. It includes information about the document in order of domain, document type, document number and version number in the document name so that document and its version history can be searched and linked. A demo of the framework is examined in Hyperledger Fabric with a certain case. It comes up with a scalable framework in terms of latency and throughput. They recommend that the proposed framework should function publicly by maintaining document privacy and security.

Document coding is an essential part of tracking and classifying the documents. The utilization of a standard naming such as Omniclass Classification System is important for implementation; however, each construction project can have its own naming standards. Therefore, the coding can be created such that it is modifiable in smart contract for each project.

Zhao et al. (2022) highlight the importance of confidentiality in infrastructure projects and the necessity of a digital document management system with the utilization of blockchain technology. The system ensures documents' security and storage by IPFS and a private blockchain, Hyperledger Fabric. The authors are designed the system for tracking concrete construction process, i.e. facility management. The system includes three layers, application layer, blockchain layer and middle layer to establish the connection between them. Application layer is for users to access the system functions. Middle layer has Hyperledger SK and IPFS API to be run in the smart contract to evoke the core layer, which is made up with the super ledger and IPFS. The system enables construction projects to be traced and archived on chain digitally by providing security and integrity validation of information.

Due to limited number of participants, private blockchains are not fully distributed and decentralized networks. Permissionless blockchains are more secure than private blockchains in terms of widely extended and almost unalterable network structure. Therefore, it is preferred to review and include the decentralized document storage solutions that utilizes a permissionless blockchain. Pradeep et al. (2021) state that a public blockchain is preferred to a consortium or a private blockchain in order to reach better security. They develop a method for tracking exchanges in a secure manner under the public blockchain system, Ethereum. The aim of the research is to administer responsibilities of members by smart contracts and to increase security and trustworthiness for tracking by blockchain. Although the tracking of document changes is secured by blockchain, it is not considered to hold the main model in a decentralized file storage system. A prototype is created and tested for three scenarios; controlling design documents, design authoring and requesting information. The prototype shows that the method is a workable; nevertheless, its implementation is not considered by cost-benefit analysis.

A fully decentralized solution is necessary for a well-secured and transparent document management system. In this context, document management systems should function with the integration of decentralized file storage system and public blockchain.

Das et al. (2021) propose a decentralized construction document management system that employs distributed content-addressed storage and blockchain. Content addressable storage means that storage location is determined by content unlike the most popular equivalent, location-addressed storage (Wang and Wu, 2020). IPFS is also a content-addressed storage since content (document) establishes its location, which is hash, with cryptographic hash functions. In the paper, a smart contract is coded on Ethereum blockchain for workflows and authenticity. The framework includes automated workflows, searching function and data integrity. Request for Information workflow is shown as a case study. Radix tree and Merkle tree is used for indexing. Radix tree is utilized such that data structure includes document type, number and version number in order to be indexed and searched in the decentralized storage. Further, Merkle tree is adopted for ensuring integrity of the system by providing that each document contains the hash of the previous documents.

The researches using both blockchain and distributed file storage for document tracking and storing in distributed networks together or separately are examined in detail. In other industries, there are researches about the decentralized document management systems; however, they are not suitable for construction projects workflows. On the other hand, the industry does not have a full-decentralized document management system. Most of them leverages a permissioned blockchain. There is a gap in the literature for a decentralized BIM document management by using a public blockchain and a distributed file storage system for workflows and assignments suitable for construction project delivery. The study main objective of this thesis is to fill this gap and to design and develop a blockchain and IPFS based decentralized BIM document management system.

CHAPTER 3

BACKGROUND

3.1 BLOCKCHAIN

3.1.1 Blockchain Definition

Blockchain which is a type of Distributed Ledger Technology (DLT) is one and only decentralized digital database distributed among the participants with security measures ensured by cryptography. The term of blockchain simply comes from recording and connecting (i.e. chaining) blocks consisting of a data or information consecutively. Each block is validated through consensus mechanism and connected to the chain in which all participant has a copy of the current blockchain simultaneously. After validation, a block in the blockchain cannot be altered or damaged since a block is connected to the previous block with cryptographic hash functions. This validation process eliminates the need for a central authority and makes the blockchain immutable and transparent. Blockchain can be utilized as a solution for any central system.

It was firstly applied for a digital cash exchange system namely Bitcoin by Nakamoto (2008). Bitcoin is created with the aim of providing a payment system eradicating the need of intermediaries such as governments or banks to ensure security, namely deploying peer to peer (P2P) technology. All bitcoin transactions are stored in blocks securely where permissionless blockchain has copies on nodes which are servers around the world and verified by the consensus mechanism such as Proof of Work (PoW). The other leading deployment of blockchain is to store smart contracts and decentralized applications in blocks by Ethereum blockchain system. The utilization of blockchains for different purposes shows that any information can be recorded or stored in blockchains' decentralized and distributed, environment by ensuring trust. Unique attributes of blockchain are explained below.

Decentralization. In centralized solutions, there must be a central authority who is responsible for assuring that conditions of an agreement are performed truthfully among parties. For example, a transaction is carried out and verified by a bank and recorded in the bank's database. Figure 3.1. demonstrates that unlike centralized one, decentralized solution eliminates the involvement of third party for trust and records transactions with Peer to Peer (P2P) connections within the scope of a distributed network.

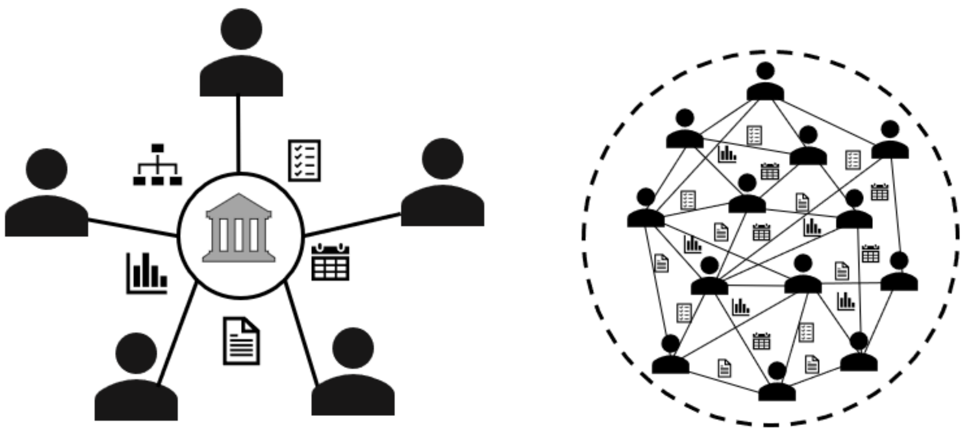


Figure 3.1 Centralized vs. Decentralized Systems

Transparency. Due to its distributed structure, the ledger is replicated at each node and open to public. Anyone can follow the details of transactions with some privacy measures. Zheng et al. (2017) used audibility instead of transparency for high openness and easy tracking of transactions. Since the architecture of blockchain, all blocks include the parent block information, in other words blocks are linked together. This brings about the fact that tracing the blocks is accomplishable till the very first block, genesis block, which also corresponds to traceability feature of blockchain.

Immutability. Blocks are validated by nodes and the blockchain is synchronized at each node of the network. Blocks with invalid transactions are not accepted by nodes. After transaction is recorded in the blockchain it is almost impossible to alter or remove it. The data in blocks are time stamped and encrypted by cryptographic hash functions. Since all blocks are chained subsequently, altering even a word or a number in a block results in disruption of the following blocks and can be detected immediately.

Anonymity. Each user has a generated address and uses this address for transactions instead of real identity (Lin and Liao, 2017). It is impossible to find the identity of a node from its address. In this manner, public blockchain provides privacy for its users. Xie et al. (2019) call this feature as pseudonymity since blockchain ensures privacy with a public address distributed publicly.

Security. Since asymmetrical cryptography is assigned to the system and using public and private keys, the system is secure. The distributed characteristics of the network prevents single point of failure.

3.1.2 Blockchain Architecture and Functioning

Blockchains are made up of single unalterable growing sequence of blocks. The very first block is named as genesis block and the previous block is called parent block. Each block has following information inside; certain data, nonce, block hash and parent block hash as shown in Figure 3.2.

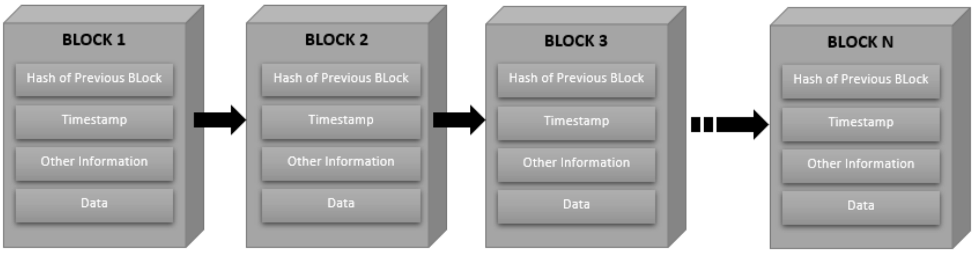


Figure 3.2 Blockchain Architecture

Bhutta et.al. (2021) indicates that the blockchain consists of layers namely data, network, consensus layers and incentive, contact, application layers as listed in Table 3.1. The basic layers are data, network and consensus layers. Each blockchain use different methods. For example, while Bitcoin uses public blockchain, Hyperledger Fabric leverages private blockchains in a network layer.

Table 3.1 Block Layers

Layers	Description
Application Layer	IoT, Health Records, Documen Management
Contact Layer	Smart Contract
Incentive Layer	Issuance Mechanism, Allocation Mechanism
Consensus Layer	PoW, PoS, PoA, etc.
Network Layer	P2P Network, Com. and Verification Mechanism
Data Layer	Data Block, Chain Structure, Time Stamp

Hash values are created by cryptographic hash functions which are also named as hashing algorithms such that hash is a fixed size number that is transformed from an arbitrary size data. For example, some words and sentences are converted to hashes with SHA-256, which is a cryptographic hash function generating a 256-bit value and demonstrated in the Table 3.2. Cryptographic hash functions are one-way, meaning they are designed such a way that the value entered cannot be obtained from generated hash.

Table 3.2 The Hashes of Given Words in SHA-256

Arbitrary Size Value	**Hash**
Furkan Koç	0eec0f3836430309185c862a4cfa7baf4fc86ac440032f64f5c1d3271ea3d2ce
The hash of this sentence is	9357fb3aba9532eaf96329c38cf13de30696a59e1f27e6456a192d4356fa429d
07.08.2021	d5179f77ebbba57748ae72721692f9f7cff1da8cb76cef87b6dc75f0242c009a
decentralization	b38e227d5deacc3533df413ecb4371a3a1bbd20a3ff1bdb58dd5f035a36982bf

Hash is a representation of a block. It is a unique 256-bit number that is generated with data, nonce and parent block hash. A new block can be added to the blockchain only if it has a hash that starts with four zeros. Data in a block and parent block hash is known, the only variable is nonce. The data inside the block represents the information that the blockchain aims to store. For instance, while a Bitcoin blocks pack data concerning buyer, sender and amount of cryptocurrency, a Ethereum blockchain can store a bytecode for an entire smart contract. Suitable nonce value, 32-bit number, must be found out in order to be able to create a valid block which has a valid hash. This time-consuming process is called mining. Miners run against each other to solve complex mathematical problems to find the nonce and get the prize of creation a new block.

Hashes help to detect whether the data in the block is altered or not. In the block architecture, there is also parent block information is attached so that blocks can be connected together and form the chain. Any modification in a block brings about alteration of block hash for the concerned block and all the following blocks. Not only finding the valid hashes for the concerned blocks is enough to manipulate the blockchain, but also consensus of the network must be reached among the participants of the network, which are called as nodes. It means the manipulator must also change the enough number of replicas of blockchain that is distributed among the nodes. The manipulation of blockchain is almost impossible due to necessity of great amount of time and computing power to change all linked blocks hashes for majority of blockchain copies, which is called majority attack, considering the enormous number of nodes.

A node can be any type of electronic device that keeps a copy of the blockchain and keeps the network running. As blockchain is distributed among nodes, it is transparent. Every transaction in the network is traceable public. Due to the transparent structure, nodes' privacy might be considered as violated; however, each node has its own address as an identity.

3.1.3 **Blockchain Limitations**

In the previous chapters, Blockchain Classification and Consensus Mechanism, the limitations of blockchain are mentioned briefly since selected type of blockchain and consensus mechanism are indispensable part of a blockchain, and selected type and consensus constitute or directly influence the weaknesses of the blockchain.

As mentioned before, blockchains, generally referred to public blockchains, capitalize on Proof of Work in order to reach consensus among parties by demonstrating computational power. Miners perform great amount of computational effort to solve mathematical problems and proves the validity of block and the majority in the network accept and validate that the block is valid. In Bitcoin, it is easier to mine a block with even a private computer at the beginning; however, the effort made for mining changes as the network becomes wide and computers owning the latest hardware are operated to win the prize. Bitcoin aims to keep the time needed for a block generation at around ten minutes by stabilizing the difficulty of mining for constant intervals. Considering that the time is fixed but the number of miner and their computing power increases and develops respectively, the amount of energy required increases. Total Bitcoin electricity consumption which is prepared by Cambridge Center for Alternative Finance, CCFA (2022), is demonstrated in Figure 3.3 and it is seen that electricity consumption ascends exponentially with each passing year.

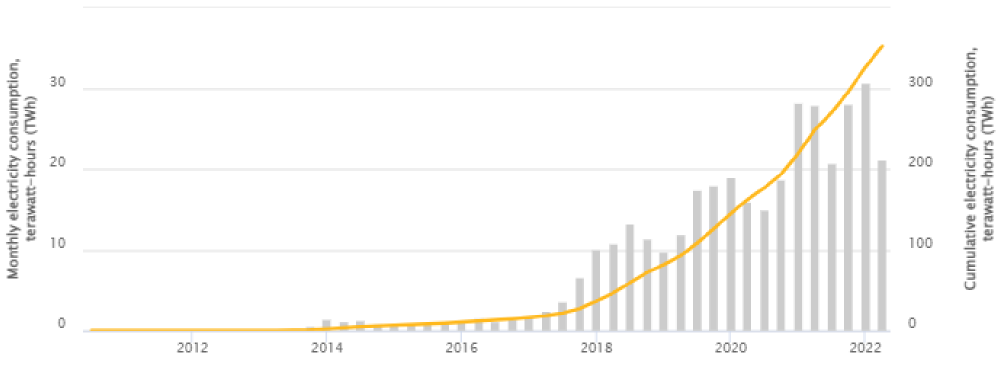


Figure 3.3 Total Bitcoin Electricity Consumption Adapted from CCFA (2022)

Bitcoin currently consumes around 150 terawatt-hours of electricity in a year, which corresponds to electricity consumption of nearly 45 million people. This amount of consumption for the operation of network leads to remarkable impacts on the environment such as air pollution and climate change. Therefore, the way that blockchain works is regarded as a weakness.

Along with the growth in load of transactions as well as in number of nodes, blockchain network's ability to progress is called scalability. Croman et al. (2016) indicates that scalability is essential and exigent concern for blockchain. The main obstruction for Bitcoin blockchain to scale is block size. Bitcoin has block average size of 0.9 MB and needs nearly 10 minutes to verify transactions, meaning that it can allow 7 transactions per second (Nakamoto, 2008) as maximum throughput, which is the maximum rate blockchain can verify transactions. Since the maximum throughput keeps constant despite of the increase in transaction demands, Bitcoin blockchain, which is representative public blockchain operating with Proof of Work, is seen as incapable of performing immense rate of transactions, i.e., not scalable. Furthermore, Considering the fact that the popular payment processing networks such as Visa and Mastercard can perform 1700 transactions per second averagely, Bitcoin fells behind in terms of throughput. Hafid et al. (2020) state that decentralization is the basis of blockchain, security is the necessary part, while stability is the major issue. This is described as scalability trilemma by Buterin (2020). It means that only two of decentralization, security and scalability can be achieved. In order to provide all of them at the same time, some solutions are found such as sharding.

Security is another concern for blockchain. Even though it is unlikely for well-distributed blockchains. the probability of a majority attack still exists. Provide that an individual or a group of people controls %51 of the nodes of the network, the network can be corrupted, meaning that modification of the ledger can be possible for attackers. However, it is nearly impossible to take possession of majority of a million nodes owing to enormous amount of computing power. On the other hand, Otte et.al. (2020) regards the Sybil attack as the most compelling attack in public blockchain structures. While a majority attack is carried out by owning the sufficient nodes legally, Sybil attack is carried out by pretending to be more modes to assure majority.

Above mentioned limitations of blockchain belongs to permissionless blockchain with Proof of Work. To avoid or prefer some limitations, many types of blockchain and consensus mechanisms are developed for blockchains. Permissioned blockchains offer solutions for energy, scalability and security issues of permissionless blockchain with several versions of consensus mechanisms such as Proof of Stake.

3.1.4 **Blockchain Classification**

Blockchain structures are roughly grouped into four types; private (permissioned) blockchain, public (permissionless) blockchain, consortium blockchain and hybrid blockchain to determine participation of nodes into the network consensus of blockchain. All types have their own advantageous and disadvantageous in terms of blockchain features such as transparency, anonymity, rate of transaction. Before going into detail, the types are demonstrated in Figure 3.4.

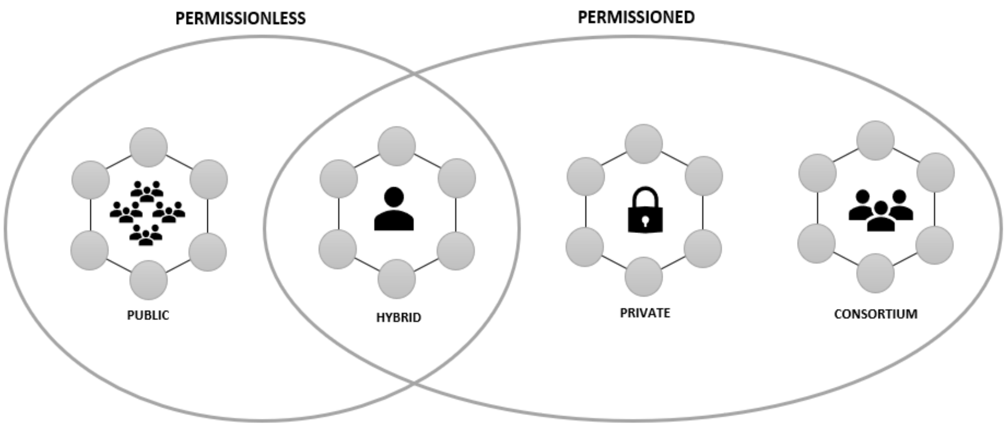


Figure 3.4 Blockchain Types

3.1.4.1 **Public (Permissionless) Blockchain**

In this type of blockchain, there is no permission for nodes to be part of the network. Anyone satisfying hardware requirements can attend and be an authorized node, which makes the blockchain transparent. As shown in Figure 3.5, the network does not have a central authority, but the consensus of the node in the network. A node in a public blockchain has ability to follow the ledger by owning a copy of it, verify transactions and create a block by making consensus among the network.

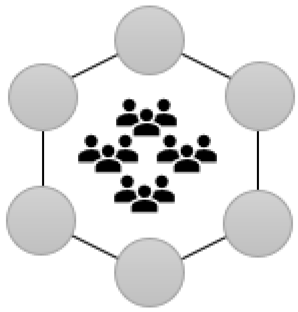


Figure 3.5 Public Blockchain

Proof of Work (PoW) is the consensus protocol used in most of public blockchains such that nodes aim to solve a complex mathematical problem first in order to prove that the transaction is valid by using excessive processing power. Since this process having huge amount of effort prevents any attack or manipulation to public blockchains dissuasively and many nodes exist to validate transactions, public blockchains are more secure distributed ledgers. However, it should be noted that it brings long processing times for a transaction. It comes to be known with high energy consumption so that anonymity and transparency of blockchain features are provided for the network. This type of blockchain is generally used in cryptocurrency. The very first public blockchain, Bitcoin and Ethereum can be given as an example.

3.1.4.2 **Private (Permissioned) Blockchain**

Private blockchain, which is also called as managed blockchain, is generally created and controlled by an author. The author determines who can join the network and which permission level each node has as demonstrated in Figure 3.6. Only the permissioned nodes can validate a transaction. Even different validation methods can be structured for a blockchain by the author like establishing strict rules for the process. As a consensus mechanism, mostly Proof of Authority (PoA) is used for private blockchains. The anonymity of nodes is not applicable for private blockchain in which identity is a stake but it provides transparency for the ledger.

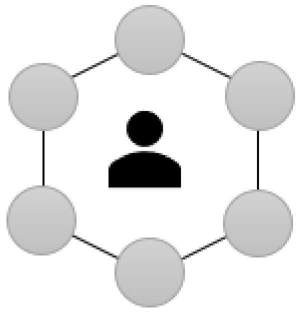


Figure 3.6 Private Blockchains

Considering the power of the authority, it can be said that private blockchain diverges from decentralized structure, yet it is distributed. In other words, it is stated that private blockchains do not fully represent the blockchain features of decentralization. Although trust is an issue due to centralized network, restricted validation permissions might enable hundreds of transactions in a second. In this manner, private blockchains are ideally suited for internal solutions for an organization. Hyperledger Fabric whose objective is to produce blockchain-based solutions and applications might be the most popular private blockchain.

3.1.4.3 **Consortium Blockchain**

Consortium Blockchain is a combination of public and private blockchains in which the latter is more dominant than the former. The blockchain network is managed by not a single author but a group of predefined nodes (Figure 3.7).

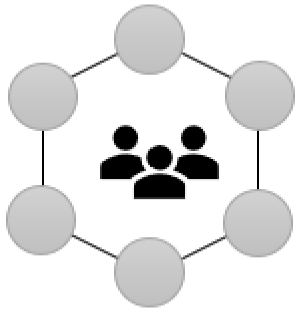


Figure 3.7 Consortium Blockchain

The blocks can be validated via this group only. Additionally, the group can set rules, arrange or cancel invalid transactions. The transactions are not open to public to ensure security. Since the nodes in the network is limited, for this type of blockchain, consensus or agreement is easily reached among the network, which brings about faster transactions and improved scalability. However, the centralized structure of consortium blockchain may lead to misuse the authorization. Therefore, it raises questions about trust. The utilization of a shared database assures more security than a single organization or a consortium.

3.1.4.4 **Hybrid Blockchain**

Like Consortium blockchain, Hybrid blockchain incorporates the features of private and public blockchains to reach an ideal form. The network is controlled by a single entity and transactions and verifications can be performed via permissioned nodes or a smart contract in case of need (Figure 3.8).

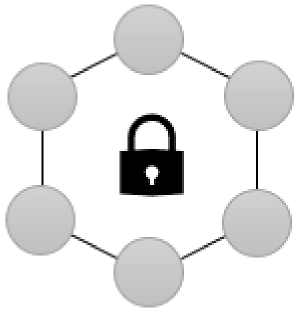


Figure 3.8 Hybrid Blockchain

Until a node is a part of a transaction, its identity is anonymous. After involving a transaction, its identity is revealed to other parties. In this type of blockchain, the transactions cannot be altered by the entity, meaning that the blockchain is immutable. While access control can be defined to restrict the permissions of nodes to particular information in the blocks as a part of permissioned blockchain, the nodes can settle upon which transactions are recorded publicly as a part of permissonless blockchain. In other words, permission and transparency features of mentioned types of blockchain are combined to structure a hybrid one. Therefore, risk of majority attack, which is also called as %51 attack, by outsiders is prevented due to the enclosed network structure of hybrid blockchain.

3.1.4.5 **Comparison of Blockchains**

After above overall definitions, all types of blockchain are compared broadly and listed in Table 3.3. over blockchain attributes. While all of them are clarified with above definitions, there might be a confusing terminology used in the table such as trust and security. Trust stands for elimination of intermediary and development of reliability among participants to transact with each other. On the other hand, security defines maintainability of the network, in other words, how much secure to external attacks. Although trust is high for public blockchain, security is evaluated as moderate in the table since it uses PoW consensus which means that it might be attacked by the majority of the network. Scalability describes the ability of network to support increment of transaction as well as the number of nodes. Performance is measured by the transaction rate per second. While, permissioned blockchains are faster, cheap, more scalable, secure and sustainable than permissonless blockchain, the latter is more decentralized, immutable, transparent, trustworthy and anonymous than the former.

Table 3.3 The Comparison of Blockchain Types

Attribute	**Public**	**Private**	**Consortium**	**Hybrid**
Decentralization	High	None	Moderate	Moderate
Access Control	None	High	High	High
Immutability	High	None	Moderate	High
Transparency	High	Low	Low	Moderate
Trust	High	Low	Moderate	Moderate
Security	Moderate	High	High	High
Anonymity	High	None	None	Moderate
Performance	Low	High	High	High
Scalability	Low	High	High	High
Energy Consumption	High	Low	Low	Moderate
Cost	High	Low	Low	Moderate

All types and their advantages and disadvantage are clearly indicated above. Public blockchain can be selected due to decentralization, immutability, transparency, trust, anonymity by satisfying the basics of blockchain definitions. The disadvantages of public blockchain are access control, performance, energy consumption and cost. They can be handled with the utilization of further developments. For example, access control can be ensured by a smart contract without requiring a private blockchain to restrict the access of users.

Wust and Gervais (2018) create a flow chart as shown in Figure 3.9 in order to decide whether a blockchain is needed and what type of blockchain should be selected for a specific application. Sonmez et al. (2021) also develop a blockchain decision framework for project management applications covering all types of blockchains.

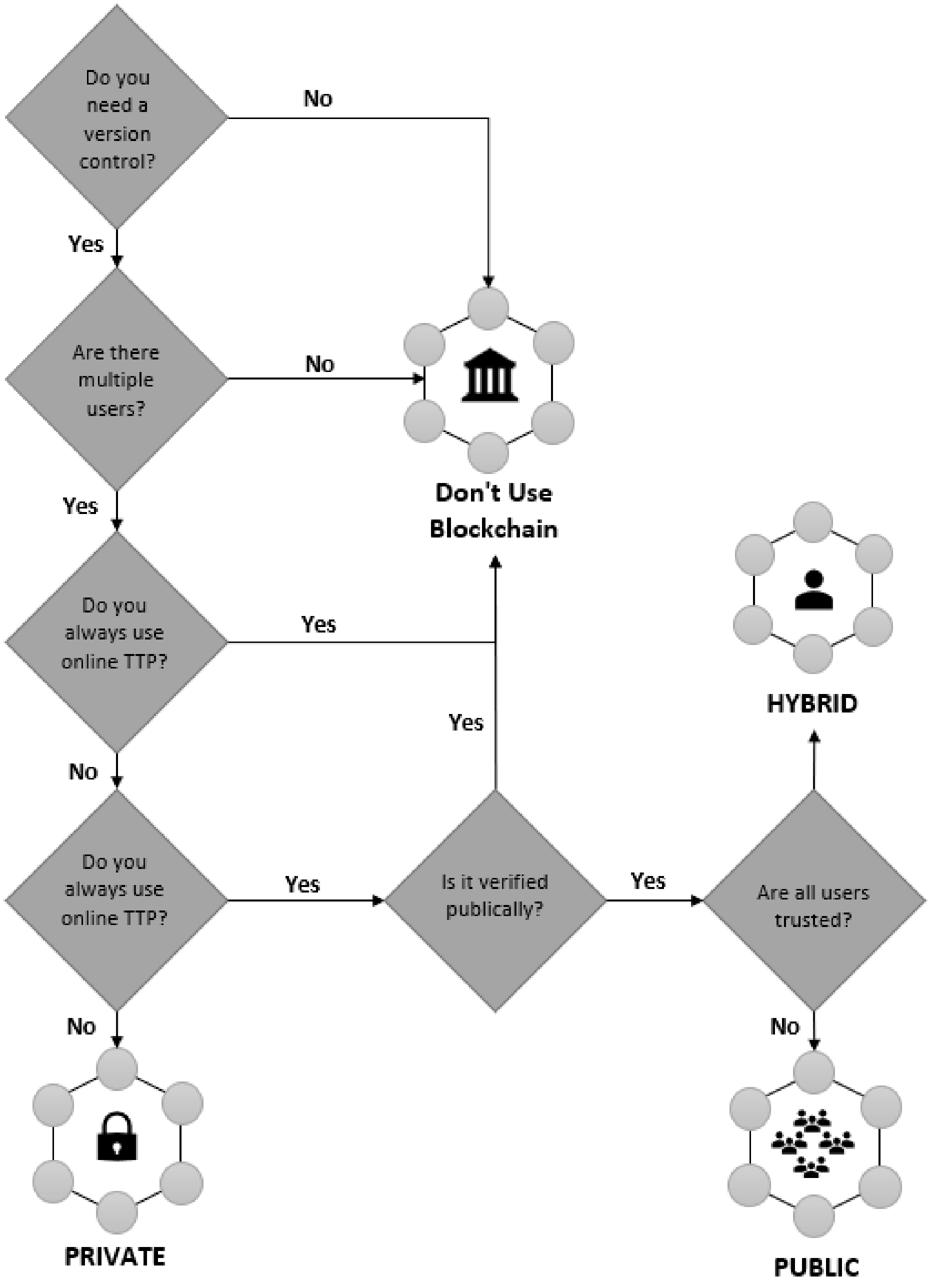


Figure 3.9 Decision Making Flow Chart Adapted from (Wust and Gervais, 2018)

3.1.5 Blockchain Consensus Mechanisms

In blockchain, the consensus mechanism or sometimes called as consensus protocol is used to control all the nodes which gets involved in transactions on the network. This ensures that all nodes in the network are in sync with each other and that the transaction is valid to be added to the blockchain. These mechanisms play an important role in the blockchain network. The transaction is checked and validated at any moment by all nodes using a consensus mechanism.

Aggarwal et al. (2020) state that consensus mechanism is a fault-tolerant mechanism that is utilized as a main part of blockchain with the aim of the fact that nodes agree upon the single state of blockchain and compile many types of consensus mechanism used in cryptography; proof-of-work (PoW), proof-of-stake (PoS), delegated proof-of-stake (DPoS), practical Byzantine fault tolerance (PBFT), proof-of-capacity (PoC), proof-of-activity (PoA), proof-of-publication, proof-of-retrievability, proof-of-importance, proof-of-burn, proof-of-elapsed time, and proof-of-ownership. Some consensus mechanisms commonly used in blockchains are explained below.

3.1.5.1 Proof of Work (PoW)

Jakobsson (1999) characterizes the notion of Proof of Work. In 2009, Proof of Work is used in Bitcoin, which functions with permissonless blockchain, in order to reach consensus among parties by demonstrating computational power. A node (a miner) performs great amount of computational effort to solve mathematical problems and proves the validity of block and the others accept and validate it. The computational effort and energy make the system operate quite slowly and harm the environment. Moreover, Law (2017) states that when the network of blockchain broadens and the number of nodes increases, the amount of time needed for validation of transactions increase accordingly.

3.1.5.2 **Proof of Stake**

Proof of Stake reduces the computation effort to verify blocks and transactions and provide security of blockchain. The verification of blocks in PoS is performed by the computers of users owning token. The users employ tokens as deposit to be able to verify blocks. Owners having staked coins are called as validators. Proof of Stake randomizes the determination of miners and validators. Among users with token, a miner is selected by chance and the others become validators.

It is created to improve PoW in terms of scalability and sustainability. In PoW, spends a lot of money for electricity, equipment, etc. to be the miner and it requires great amount of energy. PoS proposes staking instead of computational power in order to lower energy consumption for consensus.

3.1.5.3 **Delegated Proof of Stake**

In addition to Proof of Stake, nodes not only stake their coins but also cast votes for delegates. A restricted number of delegates are appointed for each block. An elected delegate may not attend the next block. The contribution of a node's vote to the election depends on how much token the node stakes. The delegate which receives the highest vote becomes entitled to create a new block and obtains transaction fee. On the other hand, reward of the block is shared among each node in compliance with the stake. In other words, if a node stake coin which is one tenth of the total stake, then the nodes get rewarded at that rate. As a type of PoS, Delegated Proof Stake is energy efficient. Due to limited number of validators, the transactions performed by PoS mechanisms faster than PoW. However, it is more likely to be exposed to an attack since the majority of the election can be reached for manipulation due to the limitation.

3.1.5.4 **Proof of Authority**

In this consensus mechanism, identity is used as stake. Validators have to be the part of the network fairly in order to maintain or spread their reputation. Validators are determined beforehand, and they verify blocks and transactions, which removes the necessity of miners and mining. It is more convenient for private organizations whose aim is to find internal solution and not to interfere with outsiders. While organizations make use of blockchain technology, they keep privacy by using Proof of Authority. Therefore, Proof of Authority does not pertain to a permissionless blockchain, but a permissioned blockchains. As identity of nodes becomes known in the network, anonymity cannot be preserved. However, this structure of the mechanism lets transactions be generated fast and the network scalable. In this manner, PoA may be evaluated as a better solution for permissioned blockchains.

3.1.6 **The Evolution of Blockchain**

Mukherjee and Pradhan (2021) review the blockchain evolution in four major versions. Blockchain 1.0 to Blockchain 4.0 corresponds to cryptocurrency, smart contracts, decentralized applications (dApps) and business-usable-platforms respectively. Additionally, Choi and Siqin (2022) interpret Blockchain 5.0 version as the utilization of machine intelligence and data analytics for automatization of decentralized applications. All evolutionary versions are described below in detail.

Blockchain 1.0

This version is the genesis of the blockchains. It is started with the peer to peer cash exchange system, Bitcoin, which is created by Nakamoto (2008). However, the basics of blockchain had been studying for about two decades. Haber and Stornetta (1991) propose that hashes of documents are linked together and distributed to clients in order to make sure that time stamp is unalterable using one-way hash functions and digital signatures. This looks very similar to the way how Blockchain works. In blockchain, each block includes transactions inside. Merkle Tree, which was developed by Merkle (1988), helps to gather all transactions hashes in one block hash. After a block is created, it must be validated by the network. This is done through consensus of the network. Proof of Work is the consensus mechanism of Bitcoin. It was introduced as Reusable Proof of Work and adopted to secure digital money by Finney (2004). All accumulation of knowledge over the years provided a basis for blockchain.

Blockchain 1.0 is inspired by Distributed Ledger Technology. Distributed Ledger Technology (DLT) is one and only decentralized digital database distributed among the participants with security measures ensured by cryptography. The first application of DLT is cryptocurrencies, which enables digital payment platforms providing financial transactions without the need of an intermediary.

Blockchain 2.0

As the first evolution, Blockchain 2.0 shows that blockchains are more than a currency. Buterin (2014) create Ethereum blockchain whose blocks contain running computer codes, which is called as smart contracts. Smart contracts are self-executing computer programs that are coded based upon the predetermined conditions among parties. In other words, it is simply the digital form of a conventional contract with a renewal of the fact that it carries out the conditions automatically. Due to its automatization which cannot be interfered once a smart contract is established, it can be used in various field such as e-voting (Yavuz et al.),

Blockchain 3.0

The main limitations of the previous blockchain versions are scalability and the great amount of effort and time required for validation. Blockchain 3.0 is evolved in order to overcome the mentioned limitations with the development of new consensus mechanisms such as Proof of Stake and Proof of Authority and the utilization of sharding. This version offers Decentralized Apps. They can run on each node's computer in blockchain network without relying on a single service provider. It uses decentralized storage and communication. dApps enables sharding which means that information is fractionated, and parts of that information is distributed among the network. This makes the network process faster and safer. Swan (2015) states that blockchain applications are not limited to financial markets only, but it can be suitable for many fields; government, health, etc.

Blockchain 4.0 and Blockchain 5.0

Blockchain 4.0 aims to make blockchain prepared for industries with beneficial applications, while Blockchain 5.0 aspires to automate decentralized applications by using machine intelligence and data analytics. They are out for integrating different platforms and accelerating the process respectively. Nonetheless, they are not fully independent versions yet.

3.2 SMART CONTRACTS

3.2.1 Smart Contract Definition

The contracts are the guaranty between two or more parties to the execution of the terms the parties agreed on. However, the terms cannot always be followed by the parties and there might exist conflicts between parties. In case of conflict, the solution is not always as it is agreed on the contract. In order to automatize the execution of contracts, Szabo (1996) indicates that the self-executing contracts reduce the efforts companies make for contract management. Deploying smart contracts in a decentralized blockchain instead of a centralized server makes smart contracts accurate, timely and tamper-proof.

In Blockchain 1.0, Bitcoin are not capable of storing conditions for transactions. Bitcoin blockchain is not equipped with this feature due to its aforementioned limitations. However, in the next evolution of the blockchain, Ethereum makes the definition of conditions in a blockchain possible. Ethereum is created for providing programmability functions on its blockchain to build applications for any use without control of central authority (Buterin, 2014).

Smart contracts are self-executing computer programs that are coded based upon the predetermined conditions among parties. In other words, it can be referred as the digital form of a conventional contract runs on a blockchain with a renewal of the fact that it carries out the conditions automatically. The smart contract is deployed on a blockchain in order to ensure the interaction between parties. There are many use cases of smart contracts. For example, Ahmadisheykhsarmast and Sonmez (2020) develop a payment security system with a smart contract to prevent payment problems in the construction industry.

3.2.2 Smart Contract Architecture and Functioning

The coder writes the conditions of smart contract on one of the programming languages the blockchain platform supports and compiles the code with a suitable complier for blockchain in order to acquire the byte code. The byte code is published on the blockchain network. As soon as it is published, it cannot be altered. When there is a revision for the contract, the coder publishes the new version on the network and inform the users to proceed with that. In Ethereum blockchain, access to the contract published on the network is got through unique address of the contract. The coder and the users keep interaction with the contract with that address. The users with the contract addresses can send transactions in which each transaction compromised of a function in the contract. The created transaction is validated through the nodes of Ethereum network in conformance with the consensus mechanism. When it is validated, the transaction is stored in a block and the block is added to the chain. The smart contract deployment and execution processes are demonstrated in Figure 3.10.

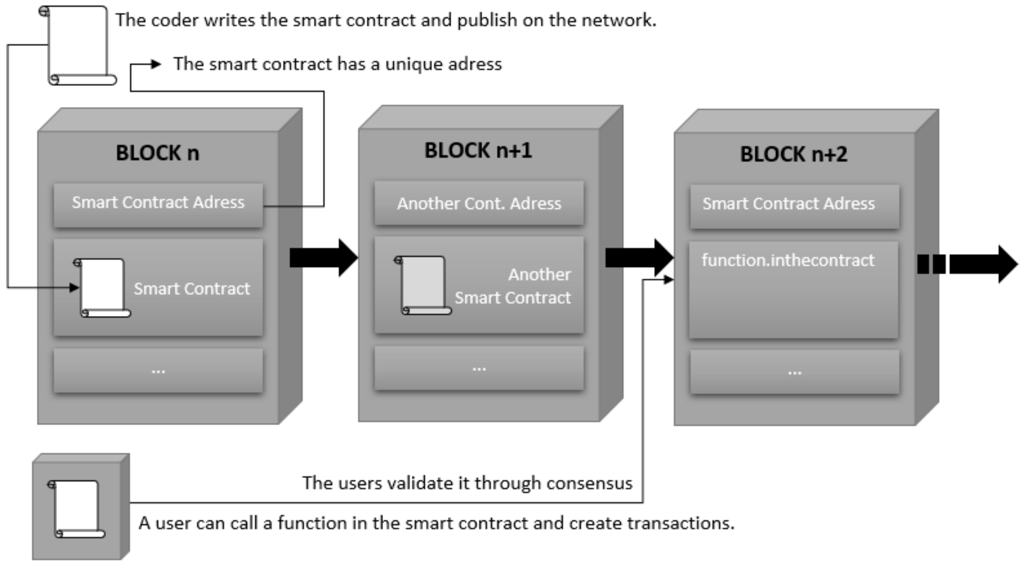


Figure 3.10 The Smart Contract Deployment and Execution processes

There exists a cost for each transaction on the blockchain. Each function in the smart contract and each transaction require Gas to be executed. The computational effort of the users to execute transaction is called Gas amount. Cost of a transaction is calculated through Gas amount. Each transaction using the smart contract requires the call of a specific function or an instruction of the contract. As the difficulty of the function to be performed increases, the Gas price ascends. Gas price depends on the Ethereum cryptocurrency, ETH by measuring it in Gwei in which 1 ETH equals to 10^9 Gwei. Gas amount and gas price are critical factors for the determination of transaction cost.

3.2.3 Smart Contract Limitations

Although the smart contracts are the digital version of conventional contracts, there are not enough policies to accept and regulate them. Governments do not approve the utilization of the smart contracts yet. Therefore, companies might not adopt the smart contracts due to legal concerns. A regulation for the smart contract is needed to legitimize and promote the adaptation of the smart contracts. Privacy might be another limitation. Due to its transparent network structure of blockchain, the smart contracts are prone to privacy breach. On the other hand, since the contract is written by a coder, there might be errors affecting the success of the smart contract. This can be seen as lack of technical expertise in the technology.

The main obstacle for the smart contracts, the limited application fields. The traditional paper-based contracts cannot be fully transformed into a digital one due to restricted operations such as payments. Besides to the limited operations, the unalterable feature of the smart contracts does not let any change in the contract once it is published on the network. This bring about inflexibility for the smart contracts. Further, the applications in payments directly depend upon the cryptocurrency. The fluctuation of cryptocurrency mat result in inoperability of the contract.

CHAPTER 4

A DECENTRALIZED BIM DOCUMENT MANAGEMENT SYSTEM

In Chapter 4, the decentralized document management system designed for secure and fast construction project delivery is desribed by examining its three subsystems namely Ethereum Blockchain, Interplanetary File System and Decentralized Application. The system is designed in a decentralized manner in order to prevent single source of failure of project documents in centralized storages. Blockchain is leveraged for tracking document publish immutably with its distributred network. IPFS is used to store documents in decentralized storages. The system has a display by decentralized application for project participants to publish, track and review documents without getting lost in the system details. In the end, the system is used in a project as a case study to present the functionality of the management system.

4.1 Research Method

The strategy followed in this research is demonstrated in Table 4.1. As a beginning the prevailing project design and document submission procedure among project parties is identified in construction industry. The current document management systems in literature and industry are indicated so that the limitations and differentiations between them is revealed. For the limitations, both overall and industry-specific blockchain-based solutions are reviewed in this section. After literature review, the technologies to be used in the system is explained in detail.

In the light of the extensive research, it is revealed that blockchain-based solutions for document storage or/and document tracking are available in the literature. However, the solutions must conform with the specific requirements of the industry to be adapted. Therefore, the solution must be adapted to the construction project delivery methods. There are similar approaches to the proposed system in the construction industry; however, they do not fully provide the features of the technologies leveraged for the solution. As a result, lack of fully implemented blockchain-based solution for document management system for construction project delivery is designated as the gap. In this manner, it is described that public blockchain and decentralized file storage is integrated to manage project documents. Then, the system is tested in a real-life project as a case study. The proposed and the current systems are compared in terms of cost and throughput for the case study. Lastly, the proposed system is discussed to indicate the contributions and limitations.

Table 4.1 Research Strategy

1. Problem Identification & Motivation	Literature Review
▪ Identify the prevailing project design and document submission procedure ▪ Research current applications in academia and industry ▪ Reveal the limitations of the applications ▪ Investigate the blockchain-based solutions for whole literature ▪ Investigate the blockchain-based solutions for construction industry	
2. Define Principles of Solution	Background
▪Describe the principles of the subsystems	
3. Design & Demonstration	System Statement & Proof of Concept
▪ Describe the proposed system ▪ Test the system with real life project as a case study	
4. Detection of Advantages & Disadvantageous	Comparison & Discussion
▪ Evaluate the comparisons in terms of cost and serviceability ▪ Discuss about security ▪ Validate the proposed system through the case study ▪ Clarify the limitations	
5. Evaluation of the System	Conclusion
▪ Summarize the study ▪ Demonstrate the future directions	

4.2 The Decentralized Document Management System

4.2.1 System Overview

A decentralized document management system is created to provide secure and trustworthy document storage together with open and immutable document tracking for construction projects. The system compromises of three subsystems namely blockchain, Interplanetary File System and decentralized application as demonstrated in Figure 4.1. Blockchain is the core subsystem such that it has smart contract inside to maintain operability of the system. IPFS is the distributed database where the system is decentralized. IPFS and blockchain is integrated through the smart contract for the sake of the decentralization of the published documents. The last subsystem is the decentralized application compromising of user interfaces that enables project participants use the document management system without getting involved and lost in the complex structures of blockchain and IPFS subsystems.

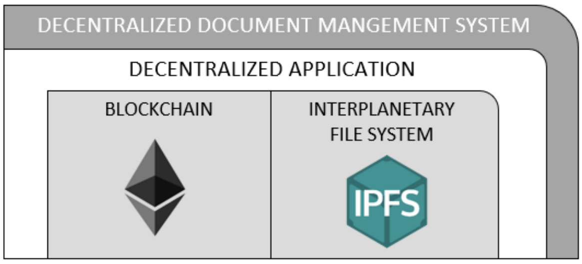


Figure 4.1 The Simplified System Architecture

The detailed system architecture is demonstrated in Figure 4.2. Each transaction prepared in the decentralized application is recorded as blocks in Ethereum blockchain network while IPFS keeps all versions of documents and issues. The participants can sign into the system via MetaMask by using blockchain accounts. All functions and general information about the project are code in the smart contract. The smart contract of the system is deployed in a virtual blockchain. All documents published and reviewed are listed in the decentralized application's document tracking page while transactions are accomplished Reviewer and Publisher pages.

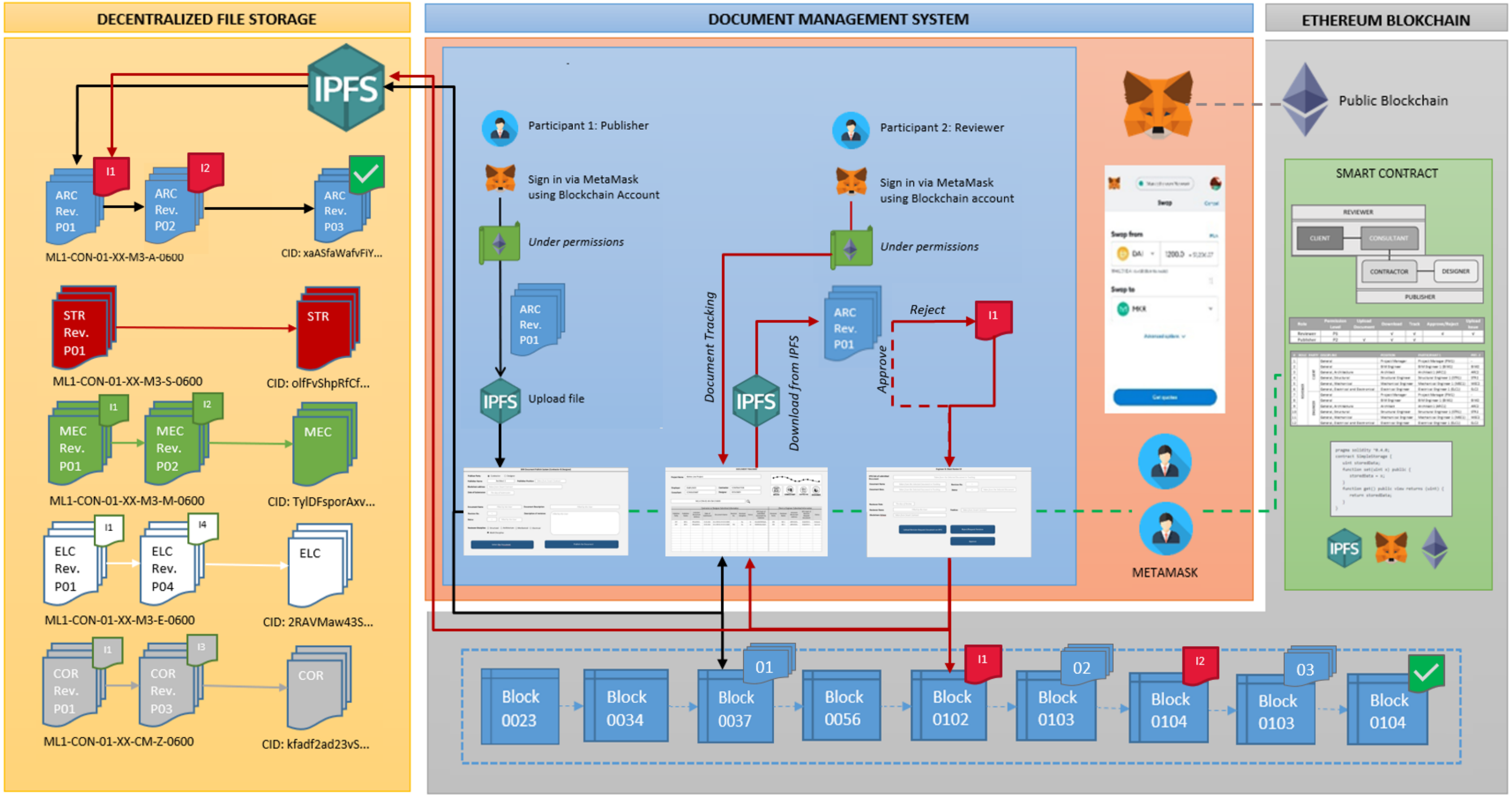


Figure 4.2 The Detailed System Architecture

4.2.2 Blockchain and Smart Contract

Blockchain is leveraged in the system to deploy the smart contract, which is self-operated and to record published documents in an immutable way. As a blockchain, Ethereum blockchain is preferred in the system design since Ethereum blockchain is a public blockchain successfully satisfying all mentioned features; decentralization, transparency, immutability, anonymity and security. Besides, Ethereum blockchain is utilized to be able to deploy a smart contract and create a decentralized application to automatize the document tracking due to the fact that it is blockchain-based smart contract platform, which provides defined rules and standards for developers to create smart contracts with ease.

The smart contract includes two parts namely information and functions. It has general information about the project that cannot be altered such as project name, parties involved, project description, account addresses, etc. In addition to general information, the smart contracts have functions to determine permissions for roles, to integrate IPFS and blockchain and to keep the system operable.

Blockchain has the smart contract as the genesis block and the information of published project documents among parties are chained in rows under the control of the smart contract. The smart contract has predefined project conditions and relations of parties involved in the project. The relations of parties vary depending upon the construction project delivery methods.

Party, Position and Participant

In the system, Design Build is selected as construction delivery method since Design Build is better than the other project delivery methods in terms of cost and schedule advantageous according to Plusquellec et al. (2017). As Design Build method incorporates, there are two major participants in the system namely, Client and Contractor with single-point responsibility. Contractor is responsible for designing and constructing the project with its Subcontractors. On the other hand, client is the party that controls and validates the project outputs. However, Client may not have required knowledge about a project on occasion especially requiring project specific know-how. Therefore, an additional party might be involved in the project; Consultant. Consultant undertakes Client assignments. Similarly, Contractors may not be possessed of a design-driven culture; therefore, it might be in need of a Designer to design the project without relinquishing the control of design part while it is in charge of construction part. The number of parties are not constrained with two parties, Client and Contractor. There exist four parties incorporated into the system to represent the complex structure of the involvement of many parties in a construction project. They are namely Client, Consultant, Contractor and Designer. Moreover, Contractor and Designer might have subcontractors for specific design parts; however, more branching is not preferred since the system are well-designed for multi-participants for a position.

As demonstrated in Figure 4.3, there are two roles for four parties to manage project documents. The first one is Publisher, which is the role of Contractor and Designer. Publisher can publish documents to the relevant parties so that they can review the documents. However, Publisher does not have authority to review and give comments to published documents. The second role for the parties is Reviewer. This role is assigned to Client and Consultant. Reviewer has permission to review published documents by Publisher and upload reviewed documents with the comments. Only Reviewer can change the status of the document.

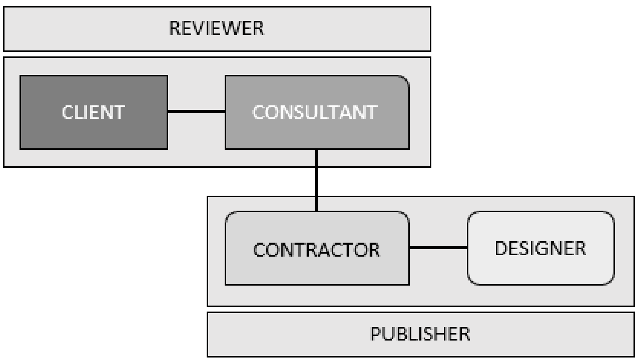


Figure 4.3 The Parties in the Project

Each party -Client, Consultant, Contractor and Designer- has its own participants such as Architect, Structural Engineer, Electrical Engineer and Mechanical Engineer. These participants are recorded in the smart contract with their positions in the project. There are 24 positions for all parties' participants. Each position is recorded in the smart contract; however, participants can get involved in the system with the same position account when more than one participant is needed for the position. For example, the smart contract has one account for Designer Architect (ARC1) as shown in Table 4.2 and the accounts for Reviewer parties can be seen in Table 4.3.

Table 4.2 Publisher Participants

#	ROLE	PARTY	DISCIPLINE	POSITION	PARTICIPANT 1	PRT. 2
13	REVIEWER	CONTRACTOR	General	Project Manager	Project Manager (PM1)	-
14			General	BIM Engineer	BIM Engineer 1 (BIM1)	BIM2
15			General, Architecture	Architect	Architect 1 (ARC1)	ARC2
16			General, Structural	Structural Engineer	Structural Engineer 1 (STR1)	STR2
17			General, Mechanical	Mechanical Engineer	Mechanical Engineer 1 (MEC1)	MEC2
18			General, Electrical and Electronical	Electrical Engineer	Electrical Engineer 1 (ELC1)	ELC2
19		DESIGNER	General	Project Manager	Project Manager (PM1)	-
20			General	BIM Engineer	BIM Engineer 1 (BIM1)	BIM2
21			General, Architecture	Architect	Architect 1 (ARC1)	ARC2
22			General, Structural	Structural Engineer	Structural Engineer 1 (STR1)	STR2
23			General, Mechanical	Mechanical Engineer	Mechanical Engineer 1 (MEC1)	MEC2
24			General, Electrical and Electronical	Electrical Engineer	Electrical Engineer 1 (ELC1)	ELC2

Table 4.3 Reviewer Participants

#	ROLE	PARTY	DISCIPLINE	POSITION	PARTICIPANT 1	PRT. 2
1	REVIEWER	CLIENT	General	Project Manager	Project Manager (PM1)	-
2			General	BIM Engineer	BIM Engineer 1 (BIM1)	BIM2
3			General, Architecture	Architect	Architect 1 (ARC1)	ARC2
4			General, Structural	Structural Engineer	Structural Engineer 1 (STR1)	STR2
5			General, Mechanical	Mechanical Engineer	Mechanical Engineer 1 (MEC1)	MEC2
6			General, Electrical and Electronical	Electrical Engineer	Electrical Engineer 1 (ELC1)	ELC2
7		ENGINEER	General	Project Manager	Project Manager (PM1)	-
8			General	BIM Engineer	BIM Engineer 1 (BIM1)	BIM2
9			General, Architecture	Architect	Architect 1 (ARC1)	ARC2
10			General, Structural	Structural Engineer	Structural Engineer 1 (STR1)	STR2
11			General, Mechanical	Mechanical Engineer	Mechanical Engineer 1 (MEC1)	MEC2
12			General, Electrical and Electronical	Electrical Engineer	Electrical Engineer 1 (ELC1)	ELC2

Nevertheless, when the project necessitates two or more architects, ARC2 and ARC3 can access to the system via ARC position of Designer. These means that many participants can access through one account. As an example, one participant for

Client's positions and three participants for Consultant, Contractor and Designer's positions result in roughly sixty participants. As illustrated in the example, the system can be scalable for different participations rates of projects due to its position-based accounts.

The position accounts must be controlled and managed by the responsible party. The accounts are created for positions instead of participants since long-termed projects might host many participants for a position from design to construction. Considering the immutableness feature of smart contracts, assigning one account to a participant that might leave the project, is not quite reasonable. Therefore, accounts are attached to the positions.

Permissions

Unlike private or consortium blockchains, public blockchains lack restriction of access of participants to the blockchain. Instead of creating permissions in the blockchain, which means that blockchain ledger is open for only project participants, permissions are created in the smart contract so that both decentralization and privacy are satisfied. Decentralization is limited in private blockchains, however, public blockchains are more secure than that due to its distributed and decentralized network. Although private blockchains are better in terms of privacy, the system protects privacy by keeping information of position accounts in the smart contract and enabling only the permissioned accounts to access the decentralized application through MetaMask. Thus, while block information i.e. document information such as document name, revision number, etc. are distributed among public blockchain network, the documents are available for only project participants.

In the decentralized document management system, permission levels are created so that the roles cannot interfere with each other. There are two permission levels, P1 and P2. Whilst P1 corresponds to Reviewer permission level, P2 represents Publisher's permissions as indicated in Table 4.4. Each project participant without role differentiation can download and track the documents. However, only Publisher can publish the very first document and the following revised documents.

On the other hand, the status of the published documents can only be changed i.e. approved or rejected by Reviewer. Reviewer can also upload an issue document, which includes comments about the published documents so that Publisher revises the document.

Table 4.4 Permission Levels

Role	**Permission Level**	**Upload Doc.**	**Download**	**Track**	**Approve /Reject**	**Upload Issue**
Reviewer	P1		√	√	√	√
Publisher	P2	√	√	√		

Permission levels could be extended to P3 such that only view features is open for an external party. However, the project documents are restricted for the project participants only in order to ensure privacy and prevent data leakage.

Smart Contract Conditions

The smart contract has a function for calling IPFS hashes i.e. CID to the block as soon as the document is uploaded. When the hash is recorded to the block with entered information by Publisher or Reviewer, the block is created and added to the public ledger (Figure 4.4).

4.2.3 **IPFS**

It is indicated that central servers are used in traditional document management system as cloud-based storage solutions such as Autodesk Construction Cloud and Oracle Aconex. Interplanetary File System is leveraged in the proposed system for storing project documents in the distributed network so that single source of failure and the other mentioned problems can be prevented for the document management. In the system, IPFS is well-integrated to the smart contract such that a participant has no direct interaction with IPFS while uploading or downloading documents.

IPFS stores published documents in encrypted and multiplied pieces distributed among network. This provides not only the secure storage but also fast access to the documents for participants.

Considering the other decentralized file storage providers such as Swarm, IPFS is selected since its functioning is proven by several researches and studies and the sources for the utilization and adaptation of IPFS in the smart contract are widespread.

In the proposed system, the IPFS-API is leveraged in order to provide the connection between the proposed system and IPFS. Further, the ipfs.io gateway is added so that the project documents can be stored and accessed. The ipfs.io gateway is a community gateway that facilitates IPFS implementation for various applications. As soon as the document is uploaded by Publisher through Publisher interface, IPFS-API sends the document to IPFS and obtain the generated CID in return. The obtained CID is recorded in the system; thus, project participants can reach the documents. The web.js is used in the system for the purpose of enabling smart contract to store and call the documents' CID values into the blockchain as a transaction. CID is unique to the document. Any alteration on the document means that the CID changes. This features of IPFS brings about traceability of the documents. Thus, data integrity and security are ensured for the stored documents.

4.2.4 **Decentralized Application**

Decentralized application web user interfaces are designed by using HTML5, CSS3, PHP and JavaScript programming languages. On the other hand, Remix IDE with Solidity 0.4.0 language and Web3.js are leveraged for matching HTTP web page with the blockchain and the smart contract. There exist three user interfaces created for facilitating the use of the decentralized management system namely Document Publish System for Publisher, Document Tracking and Document Review System for Reviewer. The user interfaces are explained in detail in the following sections.

4.2.4.1 **Document Publish System for Publisher**

The page has a simple interface that is designed for Publisher to publish documents to Reviewer as shown in Figure 4.4. Publisher party can be selected from the page as either Contractor or Designer. An account is associated with the user position and blockchain address information in the smart contract. Thus, Publisher position and blockchain address come from the smart contract since the participant signs in the system by using blockchain account via MetaMask.

Publisher enters the document name, revision number, document description, revision description and Reviewer disciplines. Document name, description and revision must conform with the project's document naming convention. While document description uncloses the document name, revision description is used for noting the revisions, changes or additions made in the document by Publisher to inform Reviewer. Moreover, the relevant disciplines can be selected so that that disciplines are aware of the fact that they must review the published document.

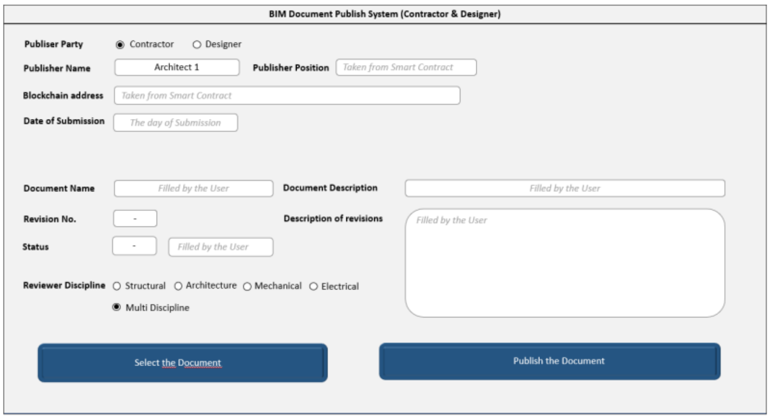


Figure 4.4 Document Publish System for Publisher

After the document information is entered properly, the document is selected by using "Select the Document" button. Then Publisher clicks on "Submit the document and Upload the document on IPFS so that the document is stored in IPFS network and the publish is recorded in the blockchain network with the entered information together with the hash taken from IPFS. The process is shown in Figure 4.5.

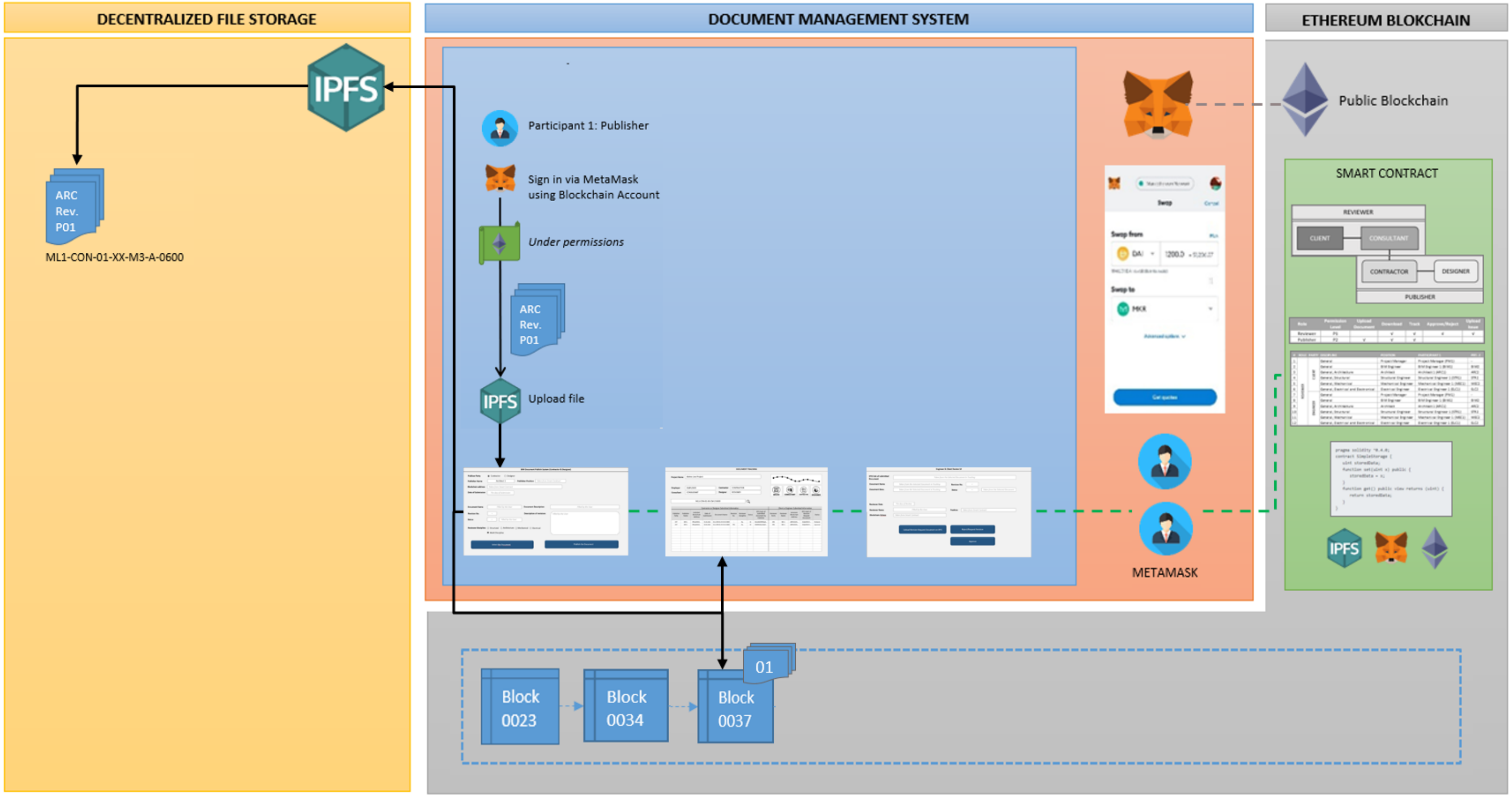


Figure 4.5 Document Publish in the DDMS

4.2.4.2 **Document Tracking**

The documents submitted by Publisher are chained and listed in the document tracking page. The page consists of two parts namely search and tracking. Search part enables user to find the documents with a document naming in a breeze. Next, tracking is the part that all project participant can follow the project. It includes several columns at two groups as shown in Figure 4.6.

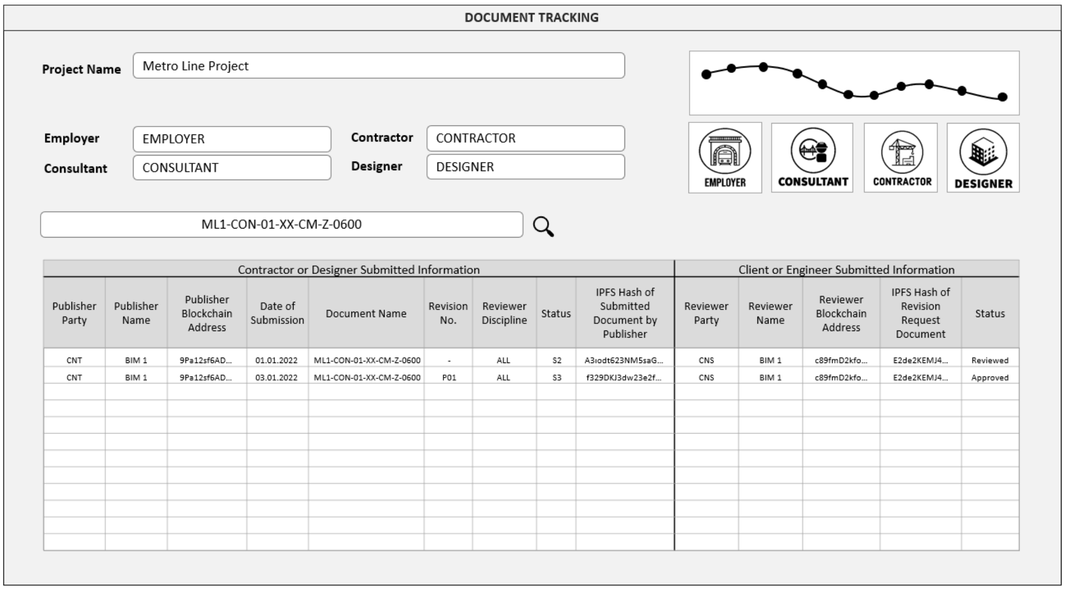


Figure 4.6 Document Tracking

The first group of columns, which are called as Publisher Submitted Information (PSI), have publish information of the document; Publisher party, Publisher name, Publisher Blockchain Address, Date of Submission, Document Name, Revision Number, Reviewer Discipline, Description of Revisions, IPFS Hash. The second group of columns, which is named as Reviewer Submitted Information (RSI), are Reviewer Party, Reviewer Name, Reviewer Blockchain Address, IPFS Hash for Revision Document and Status of Document.

When the document is published into blockchain and IPFS, the first group of columns are filled with the corresponding information from the created block. Each transaction in the blockchain is also recorded in the tracking list in order to provide efficient document management. The columns in PSI define the Publisher's identity and the Document's information. Furthermore, IPFS hash is automatically located in the tracking list. Reviewer can detect which document should be controlled and which ones have already been controlled. Once it is detected, the document is downloaded by Reviewer with IPFS hash. Then RSI is located after Reviewer submits the block. It means the first line or cycle is filled.

The first revision cycle is over when Reviewer uploads issue. If the document does not require any revision, Reviewer changes the status of document as "Approved". Thus, the document is closed for a change and locked in the blockchain. The document is recorded as final document. However, if the document does have revisions, the new cycle starts. The review process is explained in Document Review System for Reviewer.

From the beginning of the project to the end, each document publish is appeared in the tracking list. This is very important feature for accessing the single source for the document tracking since each party keeps the record of the project on its own unless a document management system provides it. In other words, each project participant is aware of who is the responsible party for the document at any time. This might bring about the acceleration of project delivery.

4.2.4.3 **Document Review System for Reviewer**

This is the user interface (Figure 4.7) for Client or Consultant to review documents. Reviewer detects the published documents in the tracking list and select the document. After the selection, the document is appeared in the Document Review System for Reviewer with IPFS Hash, Document Name, Document Description and Revision Number. The document information is automatically located in the page, which results in the protection of the chain. The revision block is chained into the published document.

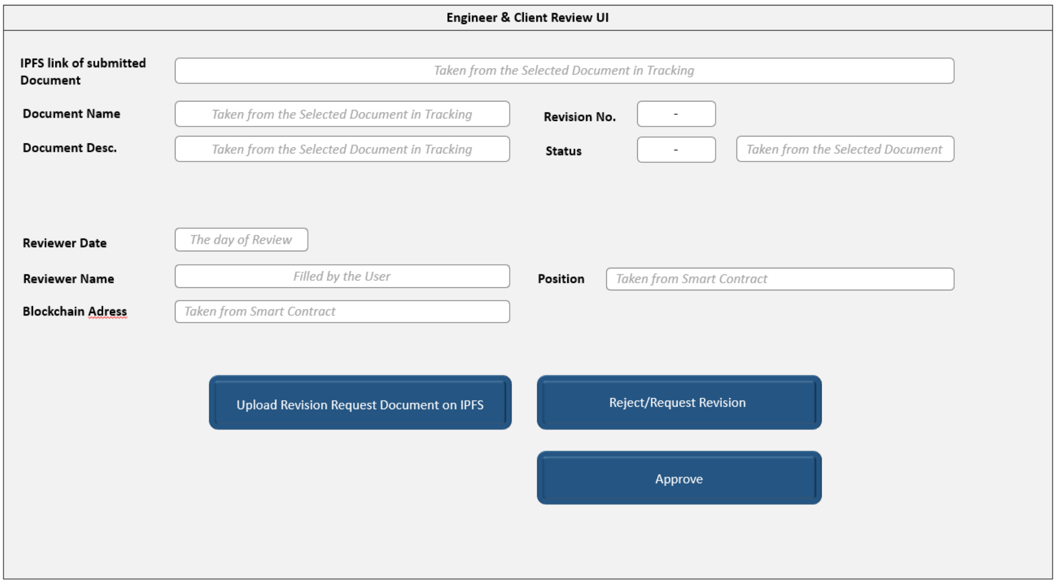


Figure 4.7 Document Review System for Reviewer

Reviewer enters the date and the name after the completion of controlling the document. If Reviewer does not detect any mistake, Reviewer clicks on “Approve” button and confirms the document. Otherwise, Reviewer rejects the document. In that condition, the issue document including revisions must be uploaded to the system. When uploaded, the document is stored in IPFS and the block is recorded in the blockchain. As indicated in the previous section, this is the end of the first document cycle. The next cycle starts with publish of revised document and ends with issue document. As shown in Figure 4.8, this process is repeated till the approval.

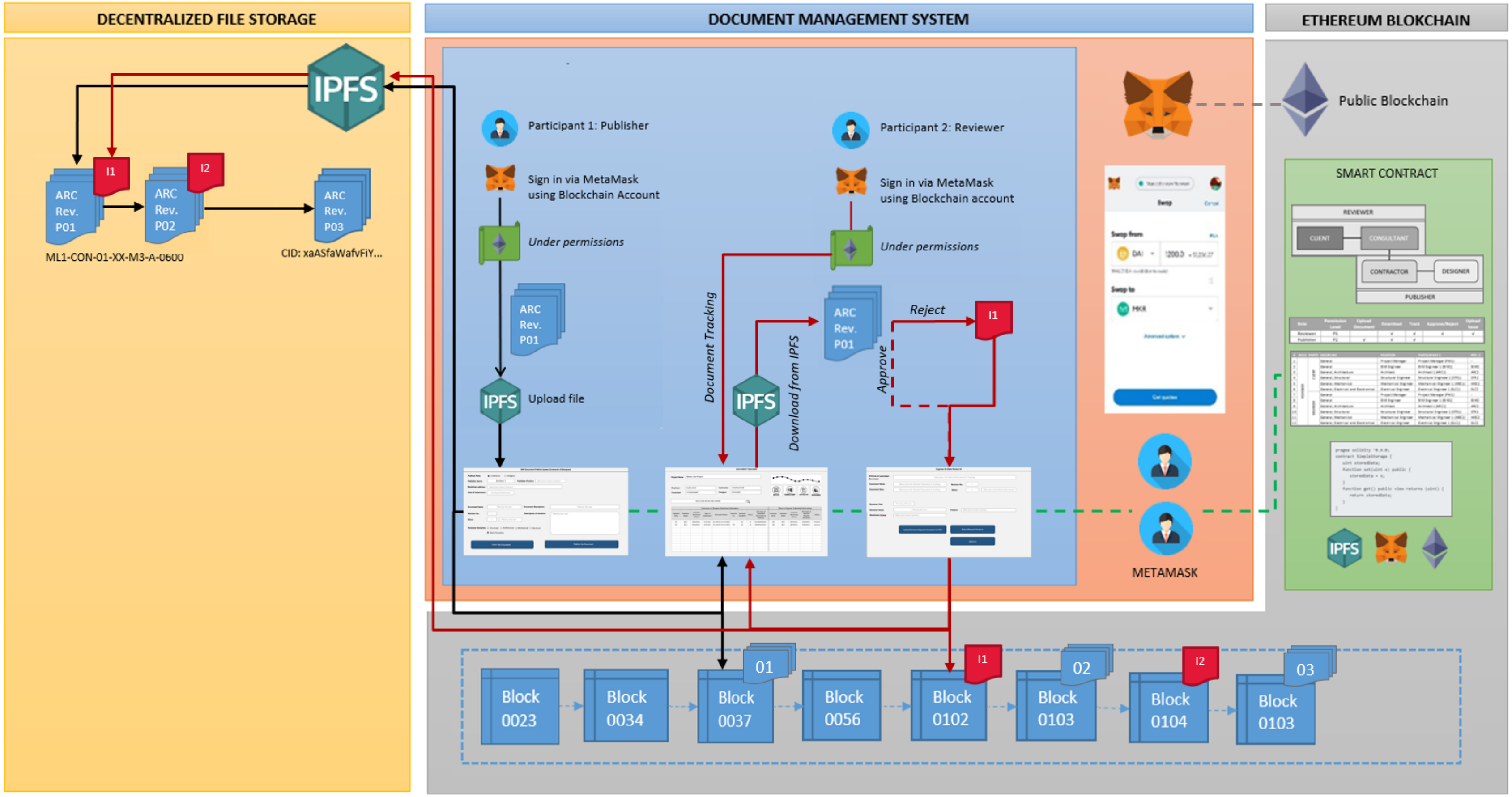


Figure 4.8 Document Review in the DDMS

4.3 Case Study

A metro line project is selected for the case study. The project is constructed with Design Build method of project delivery and BIM-based design. There are four parties involved in the project namely Client, Consultant, Contractor and Designer. It is well-suited to the proposed system. In order to define the complexity, the scope of the project is explained below:

The project overall budget is around $650 million. The capacity of the line is 44.000 passenger per hour per direction. The metro project consists of 11 stations, 1 train maintenance depot and 13 NATM and TBM tunnels along 13 km. For each station and tunnel, 6 information models are created; geotechnical, structural, architectural, mechanical, electrical, coordination models. It means that 150 information models to be coordinated and tracked excluding the other submitted documents such as technical specifications, reports, etc. For one station, Architectural, Structural, Mechanical, Electrical and Coordination BIM models are demonstrated in Figure 4.9.

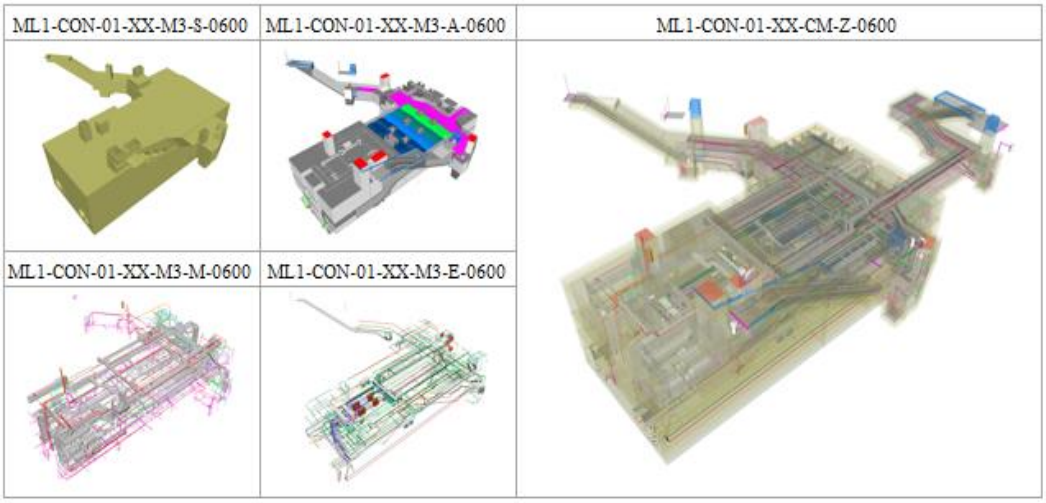


Figure 4.9 BIM Models of the Disciplines

File Naming Convention

Naming conventions help facilitating the storage and retrieval of records. BS EN ISO 19650 File Naming Conventions aims to manage information such as storing, revisioning and arranging for a whole cycle of a building from planning to operation. Thus, it makes document efficiently manageable. The standard is created to be adapted to any projects. For the project, document naming including all types of documents, drawing, models and data files is structured to be compatible with BS EN ISO 19650 File Naming Conventions. The document naming convention defined in the standard is shown in Table 4.5 and explained below.

Table 4.5 File Naming Conventions from BS EN ISO 19650

PR1	XYZ	V1	01	M3	A	15:05:15	0001	S1	P02
Project Name	Responsible Party	Volume or System	Level	File Type	Discipline	Classification	Document Number	Status	Revision Number

Project name, Responsible Party, Volume, Level, File Type, Discipline and Document Number are defined as required codes for naming. While classification is evaluated as optional, Status and Revision Number can be used if it is necessary. The required codes are used; however, optional code, classification, is not included in the naming in sake of simplicity. Status or Revision Number changes, the document name also changes, which might interrupt the tracking. Therefore, Status and Revision Number is excluded from the naming. Nevertheless, they are located in the system as separate codes.

Project name is called as ML1 representing Metro Line 1. Responsible party is the party that is responsible for the document. In this case, it is named as CON for Contractor. Since Designer is contained within Contractor. Volume is defined for stations starting from 01 to 11. The following codes are defined in the standard as follows. Level code defines level. For example, GF and 01 correspond to ground floor and base level respectively. XX is used to indicate that no level is applicable for the document. File types includes combined model (CM), clash rendition (CR), 2D drawing (DR), 2D model (M2), 3D model (M3), and so on. Discipline has one letter for the representation of roles. A, C, E, K, M, S, W, X, Z letters are for Architect, Civil Engineer, Electrical Engineer, Client, Mechanical Engineer, Structural Engineer, Contractor, Sub-contractor and General respectively.

Apart from the document name, status and revision number are given for the document tracking. Status is S0 for the Work in Progress. In Shared, there are four status; suitable for coordination (S1), suitable for information (S2), suitable for review (S3), and suitable for approval (S4). Published documents' status are named as A1 when approved by Reviewer. Next, revision number starts with a letter which is either P (Preliminary Revisions) and C (Construction Status), the following numbers start with 01 and continue. For the case study, the document can be shared in three status; S2, S3 and S4.

The BIM model names for disciplines used in the case study are listed in Table 4.6. There are five BIM models such that each of them has its own unique names so that they can be easily tracked and managed. As an example, Architectural BIM Model of Station 1 has the document name of ML1-CON-01-XX-M3-A-0600 with status and revision of S4 and P01 respectively.

Table 4.6 File naming convention of the Case Study

Document Name							**Document Description**	**Sta.**	**Rev.**
ML1	CON	01	XX	M3	A	0600	Station 1: Arch. BIM Model	S4	P01
ML1	CON	01	XX	M3	S	0600	Station 1: Struc. BIM Model	S4	P01
ML1	CON	01	XX	M3	M	0600	Station 1: Mech. BIM Model	S4	P01
ML1	CON	01	XX	M3	E	0600	Station 1: Elec. BIM Model	S4	P01
ML1	CON	01	XX	CM	Z	0600	Station 1: Coord. Model	S4	P01

Project Participants

The party, position and participant assignments for the project are the same as that of proposed system. As indicated previously each position has an account and related participant can use the account. The accounts are assigned to the positions in the smart contract. In the system, the address cannot be changed after signing into the system, which ensures security among the project participants. Role and permission assignments are completed with the deployment of the smart contract. The assigned accounts for the related positions are listed in Table 4.7 together with the permissions.

Table 4.7 Project Position Accounts

Blockchain Accounts		**Level of Permission**
Client	Project Manager	P1
Client	Architect	P1
Client	Structural Engineer	P1
Client	Mechanical Engineer	P1
Client	Electrical Engineer	P1
Client	BIM Engineer	P1
Consultant	Project Manager	P1
Consultant	Architect	P1
Consultant	Structural Engineer	P1
Consultant	Mechanical Engineer	P1
Consultant	Electrical Engineer	P1
Consultant	BIM Engineer	P1
Contractor	Project Manager	P2
Contractor	Architect	P2
Contractor	Structural Engineer	P2
Contractor	Mechanical Engineer	P2
Contractor	Electrical Engineer	P2
Contractor	BIM Engineer	P2
Designer	Project Manager	P2
Designer	Architect	P2
Designer	Structural Engineer	P2
Designer	Mechanical Engineer	P2
Designer	Electrical Engineer	P2
Designer	BIM Engineer	P2

Document naming, project participants and their roles and permissions are determined for the project. Thus, the decentralized document management system can be used for publishing and review of the project documents. In construction projects, Contractor can publish documents to Client or Consultant for many purposes. It might for information, review or approval. These three purposes of document share are examined in the case project.

All in all, the operability of the system is tested in case scenarios. Cases are summarized in Table 4.8 and defined in the following sections.

Table 4.8 The Case Scenarios

Cases	**Description**
Case 1	Document Publish for Information
Case 2	Document Publish for Review/Approval

4.3.1 **Case 1: Document Publish for Information**

Many documents are created and shared among project participant during construction project life cycle such as bidding documents, drawings, models, specifications, bill of quantities, schedules, work orders, agreements, forms, reports and so on. Some documents are shared for information so that the project participants are aware of the document. The following flow chart (Figure 4.10) is used in this condition.

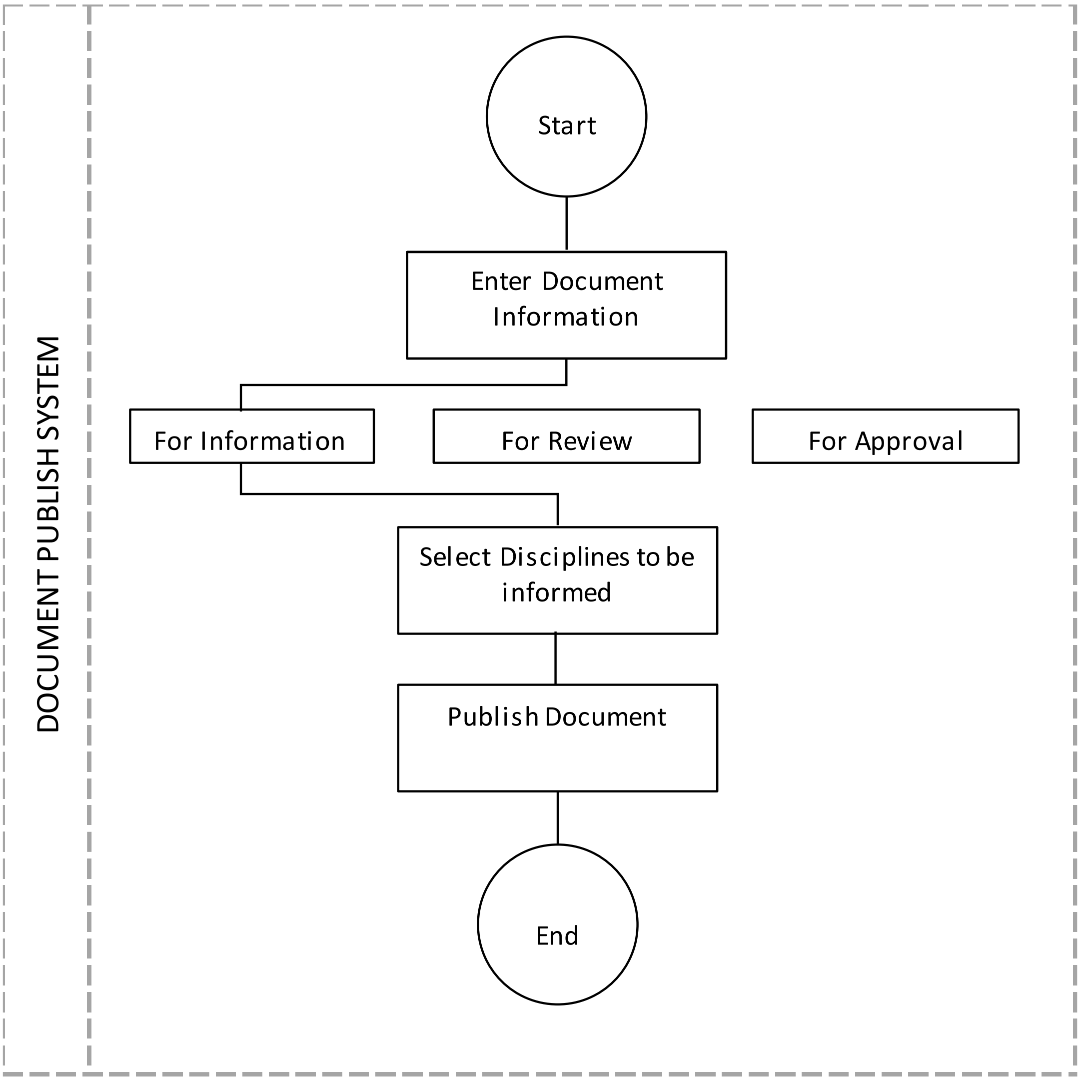


Figure 4.10 The Flow Chart of Case 1

As an example, The Document Coding System (ML1-CON-00-XX-SP-Z-0601) is demonstrated in Figure 4.11. The document is published by Contractor to all disciplines of Client and Consultant for information. The documents published for information does not require a revision number and; therefore, description of revisions. However, status of the document is indicated to be able to tracked easily.

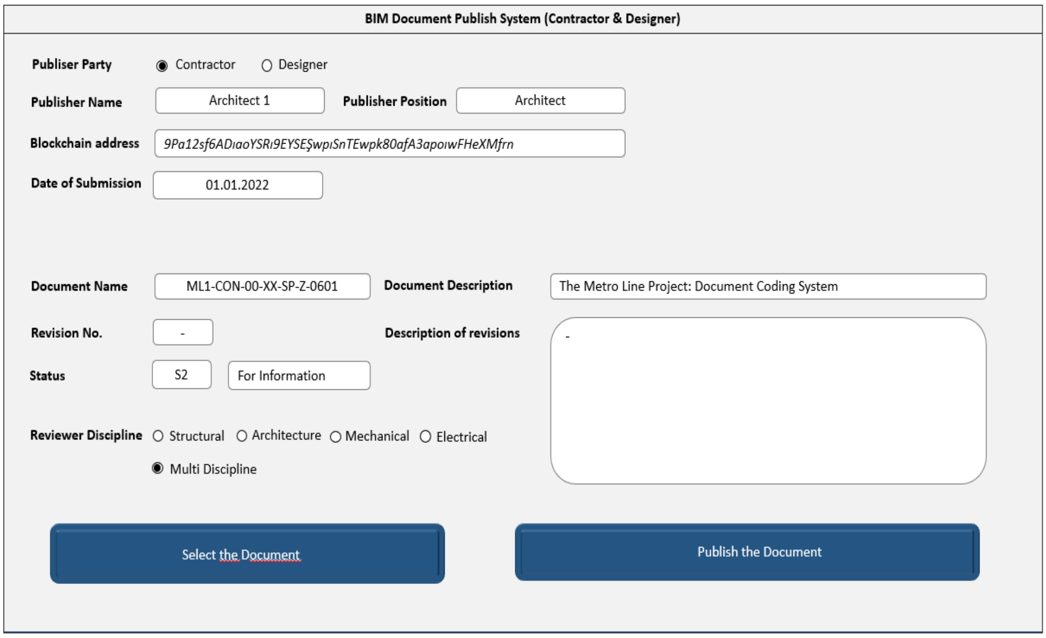


Figure 4.11 Document Publish for Information

After the publish of the document, it is listed in the document tracking (Figure 4.12).

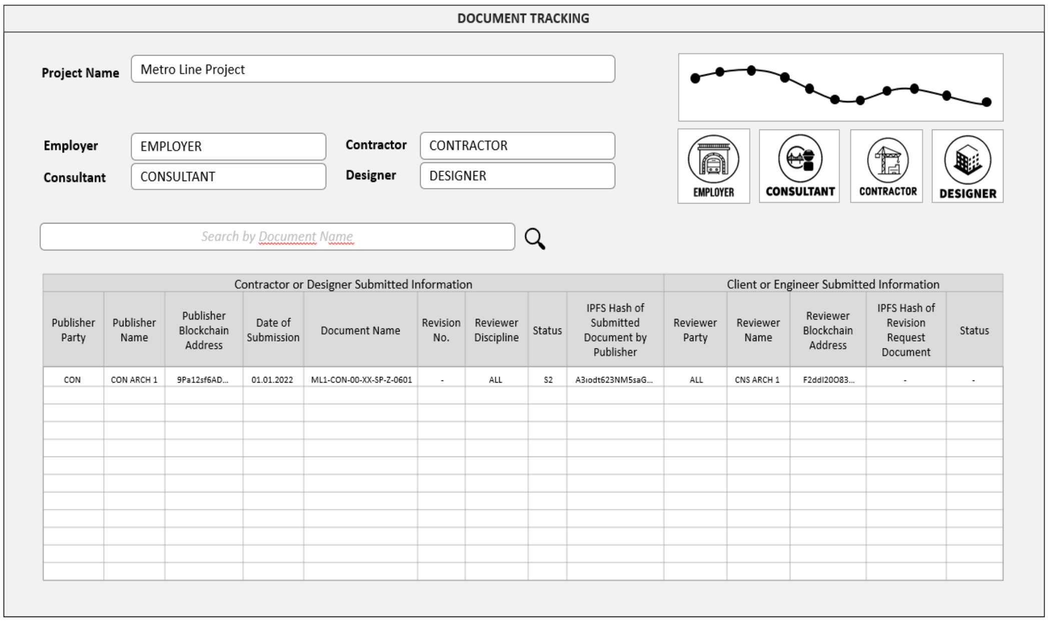


Figure 4.12 Document Tracking for the Published Document

4.3.2 Case 2: Document Publish for Review and Approval

The internal collaboration of parties can be ensured by local solutions since the process does not require external involvement. The system is developed mainly for document approval and archiving. For example, a Designer architect must be in interaction with other Designer participants namely Structural, Mechanical and Electrical engineers for design collaboration. However, the management of internal document share might not be part of document management systems in which all project parties are there. Therefore, Case 2 focuses on document share for review and approval among only Publisher and Reviewer. However, the design procedure is also explained since it is the part of document management.

Case 2 covers the publish and review processes of the coordination model which is the superposition of disciplines' information models. The model is used for many purposes. The most important use of the coordination model is the detection of physical clashes and design related deficiencies among the disciplines' information models and the resolution of the issues before construction. After Designer teams accomplish their models under the control of Consultant, BIM Engineer superposes and analyses the models. The clashes and deficiencies are reported to the teams. Then, the teams redesign their models. This process continues until the clashes and deficiencies are resolved by the teams. Once the design works for the related model are completed, the proposed document management system can be used for the design approval. The flow chart in Figure 4.13. demonstrates the path to be followed.

BIM Engineer from Contractor or Designer enters the necessary information for the coordination model namely Publisher Name, Date of Submission, Document Name, Document Description, Revision No, Status together with the Reviewer Disciplines which is responsible for the review of the model. The rest of information comes from the system itself. The coordination model has the document name of ML1-CON-01-XX-CM-Z-0600 with status of S2 and revision of P01 as shown in Figure 4.14. The document is published, means that it is uploaded to IPFS and recorded as a block in the blockchain.

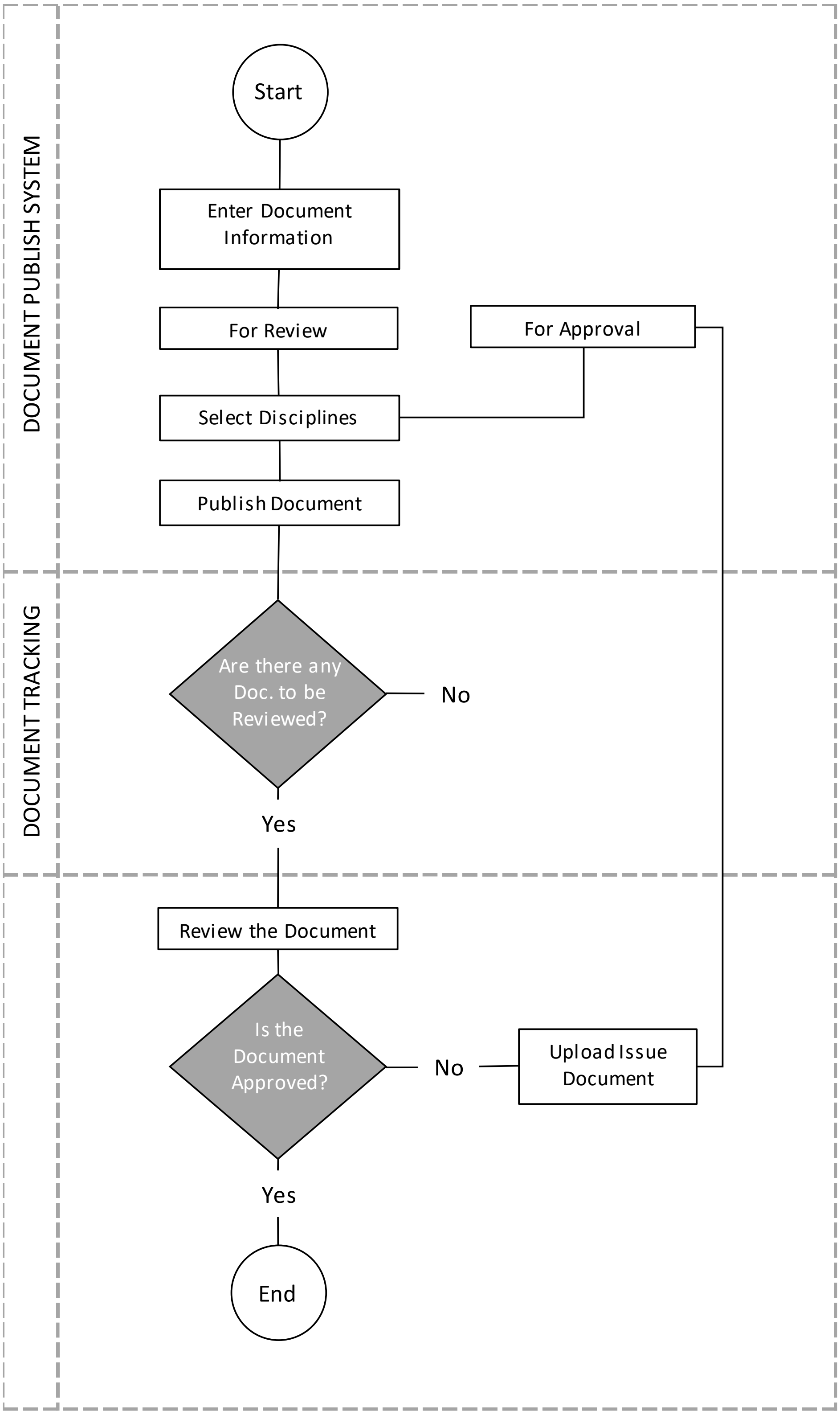


Figure 4.13 The Flow Chart of Case 2

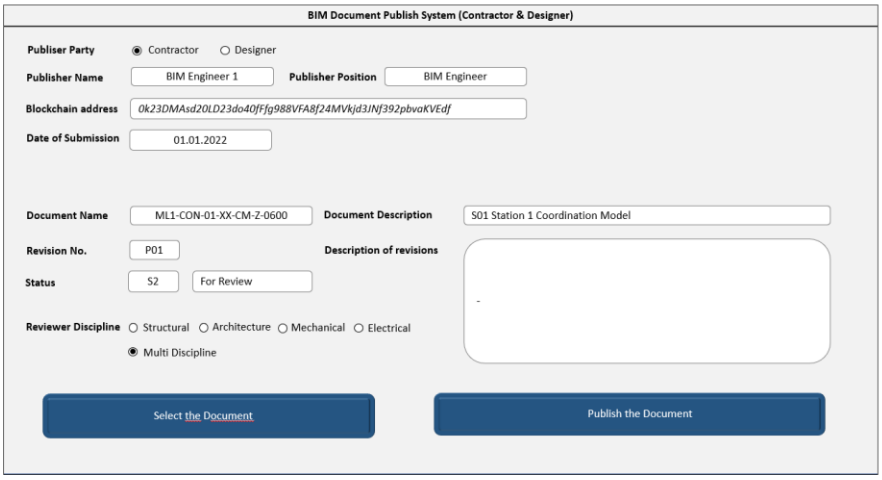


Figure 4.14 Publish of the Coordination Model for Review

As observed in Case 1, the model is listed in the document tracking with PSI. The model is assigned to multi disciplines since it requires the control and approval of all disciplines. Each participant from Reviewer parties downloads the model through the system. Unless the model is approved, the issues detected by them is combined in one issue document. Then, the issue document published on the system (Figure 4.15). The issue document is attached to the model by uploading it into the system with "Upload Issue Document on IPFS". Then, the model is rejected as demonstrated in Figure 4.17. The rejection is also recorded as a block and listed in the tracking in RSI section along with PSI. This is the end of the first revision cycle.

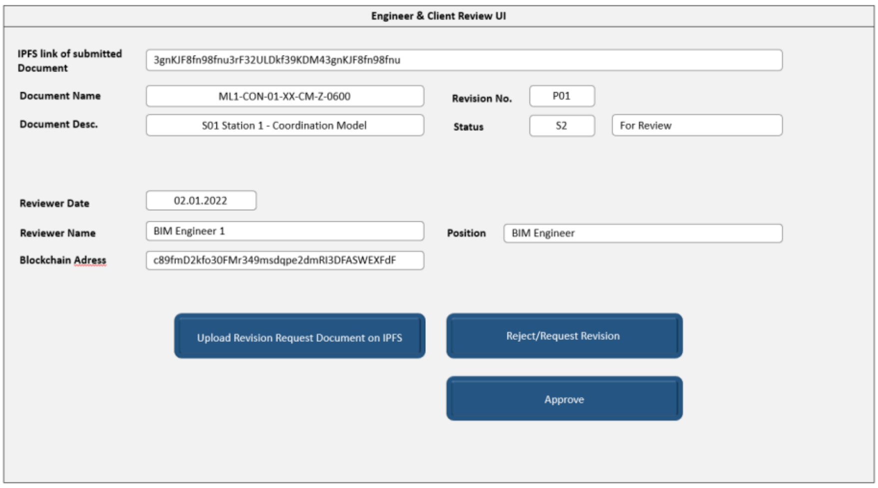


Figure 4.15 Review of The Coordination Model

The completion of first cycle is recorded in the first line on the document tracking interface. When the document is published, it fills the information in PSI from the transaction (Figure 4.16). Then, Reviewer approves or rejects the document and the information in RSI is filled automatically with the transaction (Figure 4.17).

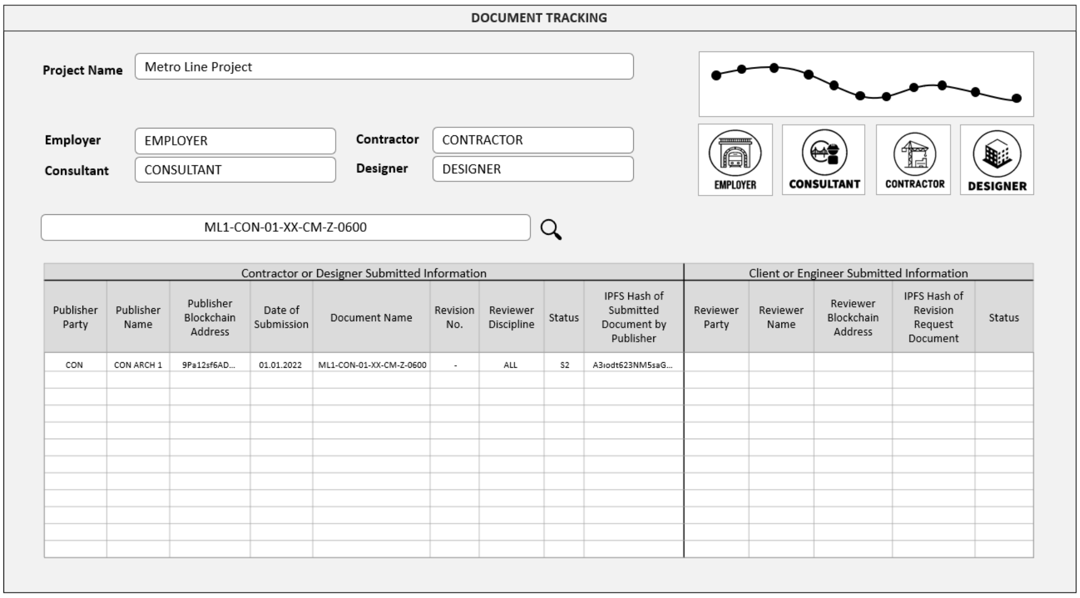


Figure 4.16 Before Document Review

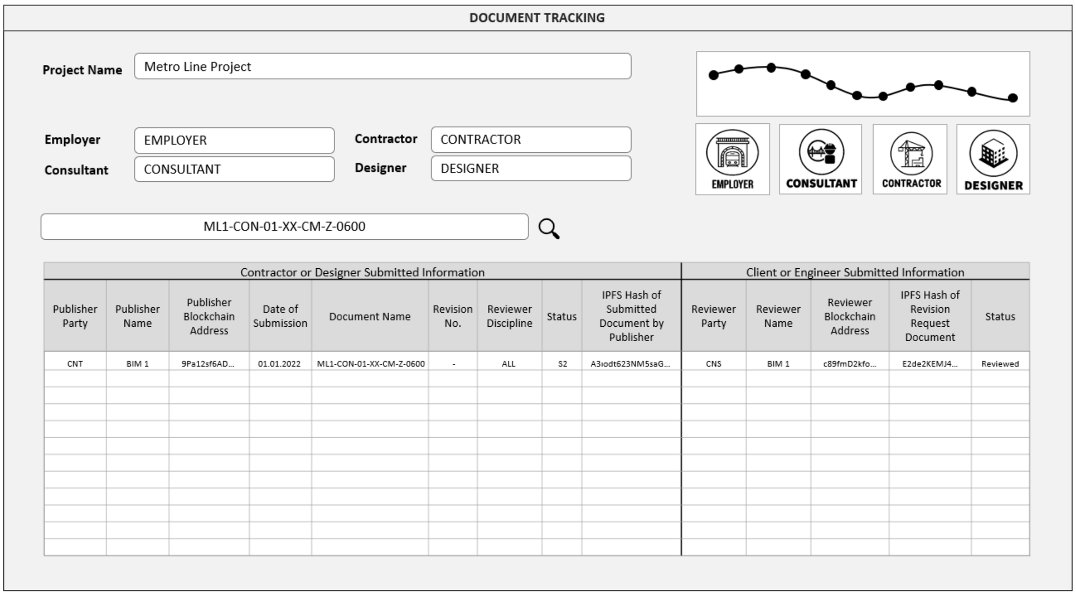


Figure 4.17 After Document Review

Publisher parties follow the document tracking for the issue document for the model (Figure 4.17). When it is uploaded by Reviewer, the responsible participants of Publisher download and provide the needful for the issues. The revised coordination model is uploaded into the system by changing the status as S3 (For Approval) and revision numbers (P02) and adding the descriptions for the revision (Figure 4.18).

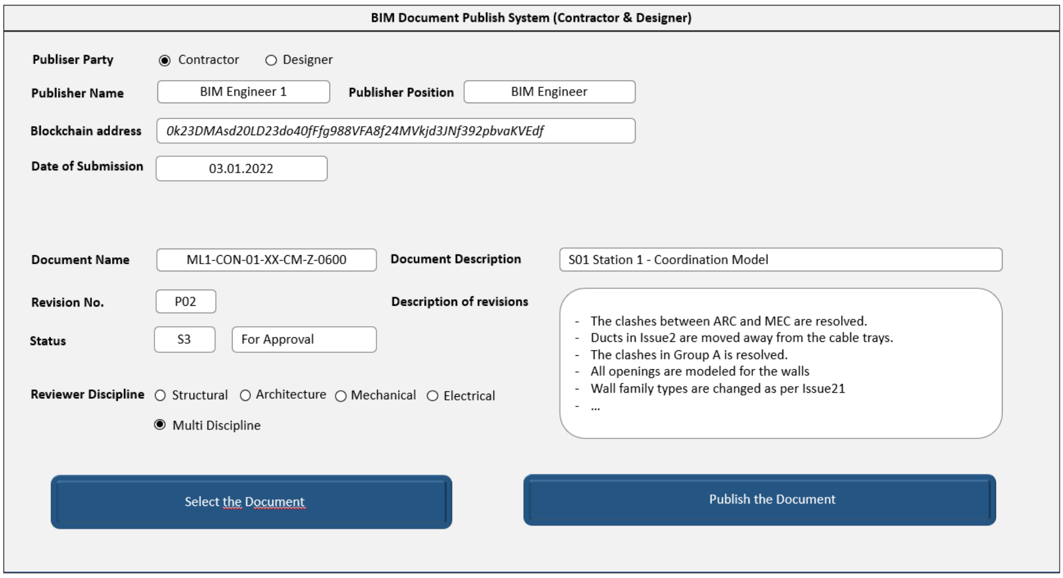


Figure 4.18 Publish of Revision of the Coordination Model

Description of Revisions section is a substantial feature for reviewing and tracking the document's revisions. This is the section that Reviewer can state the responses. The responses can either be located on the Description of Revisions section or the issue document uploaded by Reviewer. The revised coordination model is published on the system for approval of Reviewer with revisions. This brings about the issue and response tracking on the system as well. For each published document, the system produces a new IPFS hash and record it on the blockchain with the provided information.

As illustrated in Figure 4.19, the model is again under the control of Reviewer. Provided that Reviewer detects new issues, the third revision cycle is required for the model. In absence of new issue, Reviewer clicks on "Approve" button and approve the coordination model.

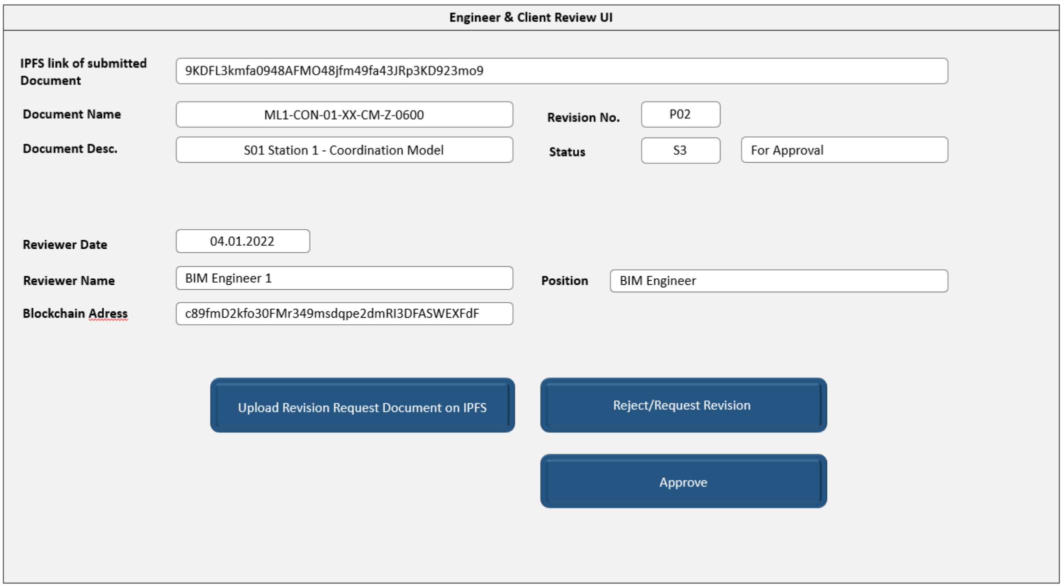
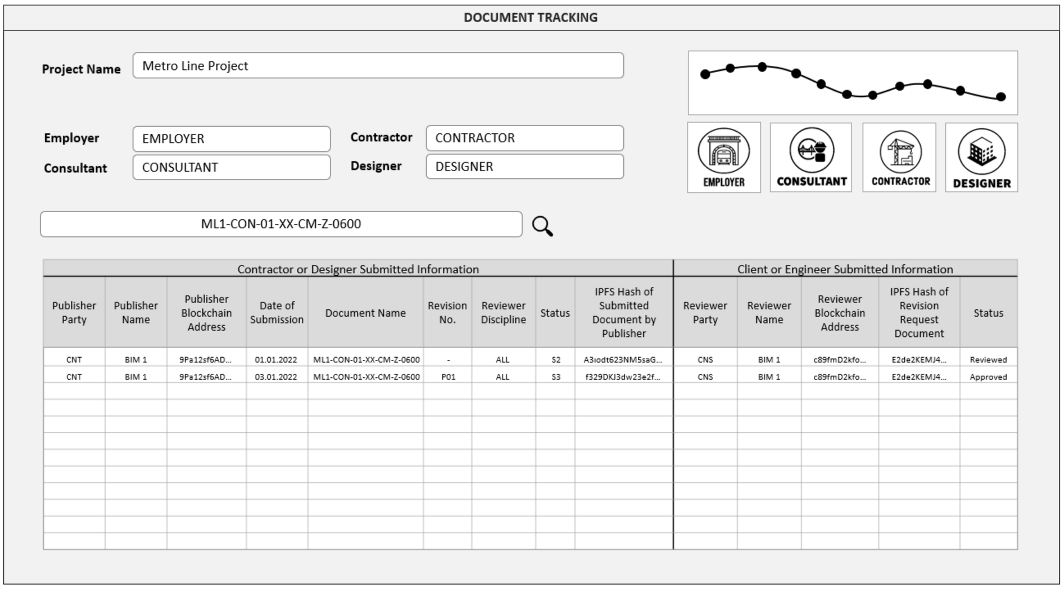


Figure 4.19 Review of the Coordination Model in the 2nd Cycle

When the model is approved by Reviewer, the status of the model changes "Approved" from "For Approval" as demonstrated in Figure 4.20.



Contractor or Designer Submitted Information									Client or Engineer Submitted Information				
Publisher Party	Publisher Name	Publisher Blockchain Address	Date of Submission	Document Name	Revision No.	Reviewer Discipline	Status	IPFS Hash of Submitted Document by Publisher	Reviewer Party	Reviewer Name	Reviewer Blockchain Address	IPFS Hash of Revision Request Document	Status
CNT	BIM 1	9Pa12sf6AD...	01.01.2022	ML1-CON-01-XX-CM-Z-0600	-	ALL	S2	A3ıodt623NM5saG...	CNS	BIM 1	c89fmD2kfo...	E2de2KEMJ4...	Reviewed
CNT	BIM 1	9Pa12sf6AD...	03.01.2022	ML1-CON-01-XX-CM-Z-0600	P01	ALL	S3	f329DKJ3dw23e2f...	CNS	BIM 1	c89fmD2kfo...	E2de2KEMJ4...	Approved

Figure 4.20 The Approval of the Model

CHAPTER 5

COMPARISON & DISCUSSION

The proposed document management system will be compared to the existing document management systems and discussed with its contribuitons and limitations in terms of cost and security analysis in this section.

5.1 Cost Analysis

The proposed system is empowered by blockchain to record transactions and IPFS to store documents. Therefore, there two main costs for recording a document publish and storing the published document. The former one is the cost spent for the blockchain utilization while the latter is paid to IPFS for decentralized storage.

In Ethereum blockchain, a smart contract is developed at first glance and each following transaction are originated from the smart contract. There are two costs for the operations on the blockchain namely contract deployment/storage and contract execution costs. In Ethereum blockchain, transaction fees are paid per each transaction for the network to validating and adding the transaction to the blockchain. Transaction fees are calculated through Gas which is the unit corresponding to the computational effort to complete a transaction on the Ethereum network. Gas prices are stated in gwei instead of ETH due to extremely low prices of Gas with respect to ETH since 1 ETH is 10^9 gwei. Transaction fee is the multiplication of Gas units and base fee plus tip. Base fee is determined with a block space and tip is for miners to execute the transaction. Higher gas prices determination for a transaction attracts the miners' attention more, which results in faster transactions. Gas prices are the keystone for security of the network where it dissuades bad actors from sabotaging.

The smart contract deployment of the system on the network requires 2,191,186 gas amount which corresponds to 0.06573560 ETH or $111.85. The other costs are derived from each transaction for using functions in the smart contract. The gas amount and transaction costs of the functions namely "Publish of the Document by Publisher", "Approve the Document by Reviewer" and "Reject and Upload Issue Document for the Document by Reviewer" are gathered in "Publish and Reject/Approval" function and shown in Table 5.1.

Table 5.1 Contract Deployment and Execution Costs for the Proposed System ($/ETH=1701.55 and Gas Price: $30*10^{-9}$ ETH)

Function	Gas Amount	Transaction Cost (ETH)	Transaction Cost ($)
Deployment of the Smart Contract	2191186	0.06573560	111.85
Publish and Reject/Approval	21544	0.00064632	1.10

For the decentralized file storage system, IPFS is selected since it does not require a fee for the storage and it has a great amount of knowledge for development and integration to other systems. However, it should be noted that the stored documents in IPFS might not be permanent if a node cannot hold the document. Fortunately, Filecoin, which is supported by IPFS, provides permanent and distributed storage in exchange for a fee. In this manner, the proposed system leverages the employability of IPFS considering Filecoin pricing. The storage cost of documents on Filecoin is calculated with respect to document size and it is $0.0000002 per GB per year. It means that 1,000,000 GB costs $1 for a 5 year-project. Thus, the cost for decentralized storage is neglected in this study.

The common document management system's pricing is determined per user annually. For example, Autodesk Construction Cloud costs $500 annually per user. Since the other DMSs have project-specific pricing such as Aconex and Procore, the proposed system is compared to Autodesk Construction Cloud in terms of cost.

While DMS cost is per user, the system' cost is per transaction. Therefore, a common criterion should be developed to compare the systems. The comparison is carried out through the case study, which is a Metro Line project with the determination of the number of participants and documents published till the completion of the project.

The planned duration of the project is planned as 5 years with total of 80 participants from each party. However, the project duration is delayed about 2 years due to some interruptions. For the non-workdays of project, the cost is still paid for DMS as per the agreement. At the 5th year of the project, there are 6400 documents published by Publisher in which 30% percent of the documents are approved and the remaining is rejected by Reviewer. The number of rejections for the document is taken as 3 on average. The cost of DMS is simply expressed as:

$$Cost_{DMS} = Cost\ of\ DMS\ ((\$/user)/year) * the\ number\ of\ users \\ * the\ project\ duration\ (year)$$

$$Cost_{DMS} = 500\ (\$/user/year)\ * 80\ user\ * 5\ years\ = \$200{,}000$$

However, there might be a reduction in bulk purchase such that it might be $100,000 for the project under the explained conditions. Therefore, the price of DMS for the project is determined as $100,000. On the other hand, the cost of the proposed system is calculated as follows:

$$Cost_{DDMS} = Contract\ Deployment\ Cost + Contract\ Execution\ Cost\ (\$/tr) \\ = Contract\ Storage\ Cost + \\ * \ The\ number\ of\ transactions\ of\ Publisher \\ + Transaction\ Cost\ (\$/tr) \\ * \ The\ number\ of\ Approved\ transactions\ of Reviewer \\ + The\ number\ of\ Revision\ Cycle \\ * The\ number\ of\ Rejected\ transactions\ of Reviewer \\ * Transaction\ Cost\ (\$/tr)$$

$$Cost_{DDMS} = 111.85\ (\$) + 6400\ (tr) * 1.1(\$/tr) + 0.3 * 6400\ tr * 1.1\ (\$/tr)$$
$$+ 3 * 0.7 * 6400\ (tr) * 1.1\ (\$/tr)$$
$$= 111.85 + 7040 + 2112 + 14784\ (\$) = \$24047.85$$

The results of a real-life project show that the cost of proposed system is $24,047.85 while the considered DMS costs $100,000 for 5 years of the ongoing project. The proposed system consists of contract deployment and contract execution costs for the project at the time of investigation are $111.85 and $23,936.00 respectively. Most of the cost is due to contract execution i.e. transactions. The system's total cost corresponds to the quarter of total cost for the document management system. The total cost of the proposed system is 24% of that of the DMS. Moreover, provided that the number transactions executed in the project should be more than 26,708 transactions to make the DMS surpass the proposed system.

$$111.85\ + 1.1\ * X + 1.1\ * 0.3 * X + 1.1 * 0.7 * 3 * X\ = 100{,}000$$

$$X\ = 26708\ tr.$$

As in the case study, it should be also noted that the project delays cannot affect the cost of the proposed system since it does require a purchasing per use instead of a time limitation. All in all, the proposed decentralized document management system is cheaper than the considered DMS.

ETH exchange rate in which both systems' costs are equal is determined by below calculations:

$$Cost_{DDMS} = Gas\ Amount\ for\ Contract\ Deployment * Gas\ Price$$
$$+ The\ number\ of\ Transaction\ of\ PublisherG$$
$$* as\ Amount\ for\ Approved\ transaction * Gas\ Price$$
$$* +The\ number\ of\ Approved\ transactions\ of Reviewer$$
$$* Gas\ Amount\ for\ Approved\ transaction * Gas\ Price$$
$$+ The\ number\ of\ Revision\ Cycle$$
$$*\ The\ number\ of\ Rejected\ transactions\ of Reviewer$$
$$* Gas\ Amount\ for\ Rejected\ transaction * Gas\ Price$$

$$Cost_{DDMS} = 2191186 * 30 * 10^{-9} + 6400 * 21544 * 30 * 10^{-9} + 0.3 * 6400 * 21544 * 30 * 10^{-9} + 3 * 0.7 * \ 6400 * 21544 * 30 * 10^{-9} = 14.1296\ ETH = \$\ 24042.32$$

When ETH exchange rate is \$7077 or gas price $124.77*10^{-9}$ ETH, the proposed system's cost is equal to that of the cloud-based document management system. Although even the maximum value of ETH, which was \$4426 on November, 2021, is considered, the system still advantageous in terms of cost. However, the fact that the increase in gas prices is 316% excluding the increase in ETH might make the system more expensive than the other one.

5.2 Serviceability Analysis

Since the DMS works on centralized servers, the data comes from the single source. The centralization causes not only a risk of single source of failure but also interruptions for project due to serviceability disruptions. Any service delay due to maintenance, renewal, update and even partial or complete failure on the system directly influence the project delivery durations; therefore, the project itself. Service quality may decrease or even service may fail because of aforementioned reasons, which may affect the completion of a construction project negatively. On the other hand, the developed system runs on many nodes. It means that the system works unless most of the nodes breaks down together. As the network expands, the probability of service outages approaches to disappear. Further, speed of service is higher for the system due to data storage and retrieval from many nodes.

5.3 Security Analysis

In this section, the proposed system is evaluated with its contributions in terms of security in order to reveal a cloud-based document management system's deficiency. The contributions are explained under three subsections namely Security, Immutability, Trust and Authentication in which all are related with each other correlatively. Each feature benefits from the decentralization and encryption of blockchains.

Security and Immutability

Each document publishes and review operations are recorded in blocks of the Ethereum blockchain. Blocks are validated by each blockchain nodes through the consensus mechanism of the blockchain and the ledger is synchronized and copied at each node of the network. After a transaction is recorded in the blockchain it is almost impossible to alter or remove it. The data in blocks are time stamped and encrypted by cryptographic hash functions. Since all blocks are chained subsequently, altering even a word or a number in a block results in disruption of the following blocks and can be detected immediately. This brings about the immutability feature for the blockchain-based document management systems. Like the document recording's immutability with the blockchain, the document storage of the system is immutable by IPFS.

IPFS makes the published document fragmented into many encrypted pieces and distributed them to its network with several copies. In other words, each encrypted piece is circulated to many nodes. The document has also unique CID value such that any change in the document results in the alteration of the CID. Furthermore, the CID is recorded to the corresponding block in the Ethereum blockchain. Therefore, it is very difficult to alter a transaction and document itself due to enormous amount of computational, monetary and temporal efforts. This immutability feature of the system also ensures security of the system.

The decentralized structure of existing DMS solutions has possibility of a single source of failure due to physical and cyber-attacks or service outages. Although cyber-attacks might be minimized with advanced precautions, the risk of physical damages on the storages exists. Any successful attack on the cloud-based system may bring about the loss of data permanently due to central storage of project documents. The proposed system keeps decentralization for both storage of documents and their publish information; therefore, single source of failure is not applicable for the system. Each node in Ethereum and IPFS protects the ledger and documents independently from each other respectively.

Trust and Authentication

The system is open for only the project participants. Each participant has its position blockchain account in order to reach documents. The participants sign in to the system via MetaMask using public key cryptography together with the private keys for publish or review a document. Further, Publisher or Reviewer accounts with their names are also recorded in the transaction in order to detect the responsible party for an operation to satisfy transparent management.

DMS requires a third-party involvement, ASP into a project. All documents produced for a project are stored in ASP's servers such that the system is based on a trust to the third party. Unlike DMS, the proposed system does not necessitate a trust of single party for control of documents. The nodes of the system do not manipulate or abuse the documents since they are all fragmented and encrypted meaninglessly. They are gathered when it is retrieved with proper permissions. However, they are under the initiative of ASP as its original and meaningful form in DMS. The providers may abuse the project, which might be confidential such as military project, etc. through the back door on behalf of interest of them or their nations. In this manner, the proposed system annihilates the need of a trusted party.

Like a trust for a third party, DMS also require to be managed by an admin, which can be seen as an internal trust, among the project participants. The author can manage assignments such as permissions etc. and the author may fail to do so. The proposed system is controlled and managed by the smart contract with self-execution. Therefore, the procedure is applied till the completion of project as specified at the beginning of the project without any trust. In addition to misleading of the author, the author or a participant may delete or alter the document in DMS by using their permissions. As indicated above, once the document has published, the system does not let the data deletion or modification by a project participant accidentally or deliberatively, which results in interruption of project delivery or even redesign. This is an important issue considering the dependency on an admin without a consensus in the project participants. Since the admin of the system which is generally from client has full authority to manage project documents. Thus, the rights of contractor and other parties might be under control of client. The proposed system protects each party's rights fairly and the documents' permeance.

Many researches reviewed in Chapter 2 leverage a permissioned blockchain in order for authentication. However, permissioned blockchains are lack of basic features of permissionless blockchain such as decentralization. The system records a transaction, which includes document publish information, throughout the determined consensus mechanism for the project participants. However, the transactions might be altered by majority of the participant or the authority due to the consensus mechanisms; Proof of Work or Proof of Authority respectively. The proposed system uses public blockchain to make the transactions unchangeable together with permission limitations encoded in the smart contract. In other words, the permissions are not applied by the type of blockchain but the smart contract in order to access the project information via the decentralized application.

5.4 Limitations

Public blockchains might use Proof of Work in order to reach consensus among parties by demonstrating computational power. Miners perform great amount of computational effort to solve mathematical problems and proves the validity of block and the majority in the network accept and validate that the block is valid. The effort made for mining changes as the network becomes wide and computers owning the latest hardware are operated to win the prize. In decentralized file storages, nodes providing storages might be always online to be able to serve. As a result of these, there needs to be a considerable amount of energy consumption, which finally harms the environment. However, Ethereum blockchain is expected to move to PoS consensus very soon which will reduce energy consumption significantly.

Most of the document management systems provide display feature on their web interface to enable project parties to review and give comments without downloading the documents such as information models. This is a quite useful feature such that it decreases the review process of documents and it eliminates the use of a software that opens the concerned document. For example, when an information model is published for information, review or approval in the decentralized document management system, Reviewer must have a software to open and review the model. All project parties generally have licenses for detailed design and control purposes. Therefore, in terms of cost burden, this might not be an issue for the parties, however, the convenience of displaying documents on the interface cannot be ignored.

Furthermore, a document management system might be in interaction with a design software to complete design works through information models in collaboration. The disciplines can work on a single model simultaneously in a central BIM model such that many participants can revise a specific part of the model at the same time. This is mainly possible due to existence of a single and centralized model. Since the proposed system has distributed the model by fragmenting it into meaningless parts with encryption. It is not possible to work on the same model on the system. However, the proposed system mainly focuses on the publish and review of documents between Publisher and Reviewer, not the ones for Publisher itself.

The implementation of a decentralized document management system requires some technical competence. Lack of expertise for the implementation might be appeared as a limitation at this point. However, as the need for the utilization of blockchain technologies is proven, there would be increase in the number of the experts in blockchain systems. The cost is not an issue for operation of the proposed system since it comes out to be 4 times cheaper than the DMS. However, ETH exchange rate and gas prices can fluctuate instantly. When ETH or gas prices increase 316%, the proposed system is not favorable in terms of cost. The system is developed on the test network for the exhibition of its contributions and limitations. To be able to fully comprehend the system, it should be implemented in the Ethereum network.

CHAPTER 6

CONCLUSION

The delivery of a construction project is a sophisticated endeavor to manage and fund design, construction, operations and maintenance phases of the project pursued with the contribution of many parties that can be geographically dispersed. A construction project has several project deliverables including all documents and information created for the fulfillment of the project's objectives such as design drawings, information models, specifications, design basis etc. The deliverables are the dynamic documents circulated among parties for approval. They can be published several times by revising according to the comments to be able to reach the final version. In complex project, there are thousands of documents and many versions of them to be managed systemmatically for the successful completion of projects. This is ensured by an electronic document management system in construction industry.

Document management systems help design, construction and operation of project by providing a common data environment such that participants can access the latest and valid document to collaborate with each other. They are mostly leveraged with cloud computing due to instant accessibility. The cloud-based document storages, which is supplied by a third party, have security and serviceability issues threatening the project success. There are brand new technologies started to be adopted in many industries for mainly security purposes. Blockchain, smart contract and distributed file storages are one of them. Blockchain is an emerging technology to securely transact between peers eliminating an intermediary with decentralization and encryption. Smart contract is a self-executing script carrying out conditions of an agreement among parties. IPFS makes document distributed through its network to prevent single point of failure and protect documents.

With the utilization of all aforementioned technologies namely public blockchain, smart contract and IPFS, this study presents a decentralized BIM document management system for construction project delivery by providing better security, immutability, trust, authentication, serviceability measures than the traditional document management systems. The proposed system integrates advantages of public blockchain, smart contract and distributed file storage on a user-friendly decentralized application in order to eliminate security and serviceability issues of the traditional systems. Blockchain is used as the core subsystem such that it has smart contract inside to maintain operability of the system for recording document publish immutably with its decentralized network. IPFS is the distributed database where the storage of documents is distributed with encryption. IPFS and blockchain is correlated with the smart contract. The system gains a display with decentralized application compromising of user interfaces that enables project participants use the document management system without getting involved and lost in the complex structures of blockchain and IPFS subsystems. The system can be used in the document management either for work in progress or archieving purposes. It should be noted that cost of the storage of documents in the proposed system is lower compared to the traditional one.

The contributions and limitations of the proposed system is observed through a real construction project by comparing the well-known cloud-based document management system. The foremost objective and contribution of the proposed system is enhancing security. The traditional document management systems are susceptible for data loss or modification due to internal or external attacks, service outages and loss of trust to the application service provider. Since the system is developed on distributed networks with encryption strategies, data loss and any modification on data are almost impossible due to immutability unlike the centralized ones. The system ensures the secuirty and permanence of stored document and its publish information on the distributed storage system and the distributed ledger respectively by benefiting from the corresponding technologies.

In addition to security, the system ensures uninterrupted serviceability due to its extensive distributed network together with the cost advantage. Since the cloud-based systems offer service through central servers, any delay in the system might obstacle the project works. However, the proposed system does not rely upon a single server but a distributed network. Therefore, it enables uninterrupted and fast transactions. Moreover, cost analysis of the case project demonstrates that the decentralized system is 4 times cheaper than the traditional system.

The proposed system is developed for Design-Build, which is one of the common construction delivery method, and the case study is selected accordingly. The system can be structured for any project delivery method by arranging or recoding the terms in the smart contract. Project parties and participants, their permissions and relations between them can be organized according to the requirements of a project at the beginning.

Although the proposed system has unique advantageous in terms of security and serviceability, there are still potential to advance the system. The future direction of research will focus on the following issues. The traditional systems are more developed in terms of practicability. The practical features of them such as the one that enables displaying and working documents on the system without a need of external applications might be added to the proposed system to make the system functional and practical. Moreover, the central models that enable participants work on the same model simultaneously is also a practical feature of the traditional document management systems. Since the proposed system concentrates on the document exchanges between parties, not party itself. This may not be seen as a problem since a party can already use its local network to benefit from the central model feature securely. However, the proposed system can be improved such that parties can use it for its internal document exchanges.

Since the implementation require technical knowledge. There might be smart contract templates developed for specific construction delivery methods in which the terms can be arranged for a project so that technical expertise cannot hinder the implementation. Since the templates are provided, any project participant with a basic knowledge of coding can easily implement the decentralized systems to a specific project. Considering the environmental damage of the system for the sake of security, less harmful blockchain consensus mechanisms as preferred in Ethereum blockchain and well-organized file distributed system methods can be investigated to decrease the energy consumption of the system.

The other improvements might focus on the cost of the system. It is indicated that ETH exchange rate and gas prices can fluctuate instantly and when ETH or gas prices increase in calculated amount, the proposed system is not favorable in terms of cost. In that condition, some precaution or optimizations might be applied for the system. The transaction cost is directly related with the gas amount. Gas amount is the computational effort to accomplish a transaction, which is affected by the block size. An optimization method can be performed to accomplish the same transaction with smaller block size. Furthermore, the other blockchains with lower transaction costs can be utilized in the system. However, it should be noted the main purpose of the system is not to decrease document management costs. The aim of the proposed system is to ensure security in first place. Therefore, cost might be a secondary issue.

The decentralized BIM document management system ensures immutable document tracking and secure document storage together. The former provides project parties to follow the document exchanges on a single unalterable ledger to eliminate the possible conflicts. However, if project parties are capable of managing document in old-fashioned way in which each party has its own document tracking system, only the latter can be used for a document management system together with the development of centralized application. Considering the storage in IPFS is extremely cheap, documents are stored in decentralized file storage system and tracking is provided by an application yet it is prone to be altered or lost. As the integrations of public blockchain and IPFS with the smart contract and decentralized application, the proposed system has great potential to help design, construction and operation of a facility securely and more practically with further developments.

REFERENCES

Aggarwal, S., & Kumar, N. (2021). Cryptographic consensus mechanisms. *Advances in Computers*, 211-226. doi: 10.1016/bs.adcom.2020.08.011

Ahmad, H., Bazlamit, I., & Ayoush, M. (2017). Investigation of Document Management Systems in Small Size Construction Companies in Jordan. Procedia Engineering, 182, 3-9. doi: 10.1016/j.proeng.2017.03.101

Ahmadisheykhsarmast, S., & Sonmez, R. (2020). A smart contract system for security of payment of construction contracts. Automation In Construction, 120, 103401. doi: 10.1016/j.autcon.2020.103401

Autodesk, BIM360, Available at, https://www.autodesk.com/bim-360/. Last accessed on 12 April 2022

Autodesk. (2016). Autodesk collaboration for Revit security overview. Retrieved 20 March 2022, from https://microsolresources.com/wp-content/uploads/2016/07/Collaboration-for-Revit-Whitepaper-Security.pdf

Azhar, S. (2011). Building Information Modeling (BIM): Trends, Benefits, Risks, and Challenges for the AEC Industry. Leadership and Management in Engineering, 11(3), 241-252. doi: 10.1061/(asce)lm.1943-5630.0000127

Begam, B., & Sasiskala, M. (2021). Attribute based Encryption in Cloud Computing – A Review. International Journal of Computer Applications, 174(19), 36-38. doi: 10.5120/ijca2021921084

Bhutta, M., Khwaja, A., Nadeem, A., Ahmad, H., Khan, M., & Hanif, M. et al. (2021). A Survey on Blockchain Technology: Evolution, Architecture and Security. IEEE Access, 9, 61048-61073. doi: 10.1109/access.2021.3072849

Björk, B. (2006). Electronic document management in temporary project organisations. Online Information Review, 30(6), 644-655. doi: 10.1108/14684520610716144

Bodhuin, T., Preziosi, R., & Tortorella, M. (2007). Supporting document management by using RFID technology. International Journal of Internet Protocol Technology, 2(3/4), 165. doi: 10.1504/ijipt.2007.016218

BS EN ISO 19650-3, (2020). Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM). Information management using building information modelling. https://www.iso.org/obp/ui/#iso:std:iso:19650:-3:ed-1:v1:en

Buterin, V. (2014). Ethereum white paper, 2014. A next-generation smart contract and decentralized application platform. Retrieved from www.github.com/ethereum/wiki/wiki/White-Paper%0Ahttps://s3.us-east-2.amazonaws.com/bci-static/downloads/ethereum_whitepaper.pdf%0Ahttps://ethereum.github.io/yellowpaper/paper.pdf

Caldas, C., & Soibelman, L. (2003). Automating hierarchical document classification for construction management information systems. Automation in Construction, 12(4), 395-406. doi: 10.1016/s0926-5805(03)00004-9

Cambridge Bitcoin Electricity Consumption Index (CBECI). (2022). Retrieved 28 May 2022, from https://ccaf.io/cbeci/index

Celik, Y., Petri, I., & Rezgui, Y. (2021). Leveraging BIM and Blockchain for Digital Twins. 2021 IEEE International Conference On Engineering, Technology and Innovation (ICE/ITMC). doi: 10.1109/ice/itmc52061.2021.9570246

Chassiakos, A., & Sakellaropoulos, S. (2008). A web-based system for managing construction information. Advances in Engineering Software, 39(11), 865-876. doi: 10.1016/j.advengsoft.2008.05.006

Choi, T., & Siqin, T. (2022). Blockchain in logistics and production from Blockchain 1.0 to Blockchain 5.0: An intra-inter-organizational framework. Transportation Research Part E: Logistics and Transportation Review, 160, 102653. doi: 10.1016/j.tre.2022.102653

Croman, K., Decker C., Eyal, I., Gencer A., et al. (2016). On Scaling Decentralized Blockchains (A Position Paper). 3rd Workshop on Bitcoin and Blockchain Research, Barbados.

Das, M., Tao, X., & Cheng, J. (2021). BIM security: A critical review and recommendations using encryption strategy and blockchain. Automation in Construction, 126, 103682. doi: 10.1016/j.autcon.2021.103682

Das, M., Tao, X., Liu, Y., & Cheng, J. (2022). A blockchain-based integrated document management framework for construction applications. Automation in Construction, 133, 104001. doi: 10.1016/j.autcon.2021.104001

Document Management Software Market Size, Share, Trends & Forecast. (2022). Retrieved 14 July 2022, from https://www.verifiedmarketresearch.com/product/document-management-software-market/

Document Management Software Market Size, Share, Trends & Forecast. (2021). Retrieved from https://www.verifiedmarketresearch.com/product/document-management-software-market/

Erdoğan, R. T. (2019). 2019/12 Sayılı Cumhurbaşkanlığı Bilgi ve İletişim Güvenliği Tedbirleri Genelgesi [Presidential Information and Communication Security Measures Circular No. 2019/12]. Türkiye Cumhurbaşkanlığı. Retrieved from https://cbddo.gov.tr/mevzuat/2019-12-sayili-bilgi-guvenligi-tedbirleri-cumhurbaskanligi-genelgesi/

Erri Pradeep, A., Yiu, T., Zou, Y., & Amor, R. (2021). Blockchain-aided information exchange records for design liability control and improved security. Automation In Construction, 126, 103667. doi: 10.1016/j.autcon.2021.103667

Fernando, H., Hewavitharana, T., & Perera, A. (2019). Evaluation of Electronic Document Management (EDM) systems for construction organizations. 2019 Moratuwa Engineering Research Conference (Mercon), 273-278. doi: 10.1109/mercon.2019.8818768

Finney, H. (2004). RPOW - Reusable Proofs of Work. Retrieved from https://nakamotoinstitute.org/finney/rpow/index.html

Fuertes, A., Forcada, N., Casals, M., Gangolells, M., & Roca, X. (2007). Development of an Ontology for the Document Management Systems for Construction. Complex Systems Concurrent Engineering, 529-536. doi: 10.1007/978-1-84628-976-7_59

Giles, M. (2022). A Major Outage at AWS Has Caused Chaos at Amazon's Own Operations, Highlighting Cloud Computing Risks. Retrieved 23 July 2022, from https://www.forbes.com/sites/martingiles/2021/12/07/aws-outage-caused-chaos-at-amazon-underlining-cloud-computing-risks/?sh=6e8c669e68347

Gordon, C. (1994). Choosing Appropriate Construction Contracting Method. Journal of Construction Engineering and Management, 120(1), 196-210. doi: 10.1061/(asce)0733-9364(1994)120:1(196)

Grover, R., & Froese, T. (2016). Knowledge Management in Construction Using a SocioBIM Platform: A Case Study of AYO Smart Home Project. Procedia Engineering, 145, 1283-1290. doi: 10.1016/j.proeng.2016.04.165

Guo, F., Jahren, C., & Turkan, Y. (2019). Electronic Document Management Systems for the Transportation Construction Industry. International Journal of Construction Education and Research, 17(1), 1-16. doi: 10.1080/15578771.2019.1685612

Haber, S., & Stornetta, W. (1991). How to time-stamp a digital document. Journal of Cryptology, 3(2), 99-111. doi: 10.1007/bf00196791

Hafid, A., Hafid, A., & Samih, M. (2020). Scaling Blockchains: A Comprehensive Survey. IEEE Access, 8, 125244-125262. doi: 10.1109/access.2020.3007251

Han, J., Kim, H., Eom, H., & Son, Y. (2021). A decentralized document management system using blockchain and secret sharing. Proceedings Of The 36Th Annual ACM Symposium On Applied Computing. doi: 10.1145/3412841.3442077

Haron, A., Marshall-Ponting, A., Aouad, G., 2010. Building information modelling: Literature review on model to determine the level of uptake by the organisation. CIB World Building Congress 2010. Salford, UK

Hoła, B., & Sawicki, M. (2014). Tacit Knowledge Contained in Construction Enterprise Documents. Procedia Engineering, 85, 231-239. doi: 10.1016/j.proeng.2014.10.548

Hussain, M., Memon, A., & Bachayo, A. (2022). Building Information Modeling in Construction Industry of Pakistan: Merits, Demerits and Barriers. Journal of Applied Engineering Sciences, 12(1), 43-46. doi: 10.2478/jaes-2022-0007

IPFS, IPFS. https://ipfs.io/. Last accessed on 12 June 2022

Kao, C., & Liu, S. (2013). Development of a Document Management System for Private Cloud Environment. Procedia - Social and Behavioral Sciences, 73, 424-429. doi: 10.1016/j.sbspro.2013.02.071

Khatiwada, P., & Yang, B. (2022). An access control and authentication scheme for secure data sharing in the decentralized cloud storage system. 2022 5Th Conference on Cloud and Internet of Things (Ciot). doi: 10.1109/ciot53061.2022.9766634

Konishi, K., & Ikeda, N. (2007). Data model and architecture of a paper-digital document management system. Proceedings of the 2007 ACM Symposium On Document Engineering - Doceng '07. doi: 10.1145/1284420.1284429

Krishnan, S., & Subramanian, N. (2015). Evaluating Carbon-Reducing Impact of Document Management Systems. 2015 Seventh Annual IEEE Green Technologies Conference. doi: 10.1109/greentech.2015.32

Kubba, S. (2017). Building Information Modeling (BIM). Handbook of Green Building Design and Construction, 227-256. doi: 10.1016/b978-0-12-810433-0.00005-8

Labaran, Y., Mathur, V., & Farouq, M. (2021). The carbon footprint of construction industry: A review of direct and indirect emission. Journal of Sustainable Construction Materials and Technologies, 6(3), 101-115. doi: 10.29187/jscmt.2021.66

Law, A. (2017). Smart Contracts and their Application in Supply Chain Management.

Leukel, J., Schuele, M., Scheuermann, A., Ressel, D., & Kessler, W. (2011). Cooperative Semantic Document Management. In Business Information Systems, 14th International Conference, BIS 2011 (pp. 254-265). Poznan, Poland.

Lin, I.-C., Liao T.-C., (2017). A survey of blockchain security issues and challenges, Int. J. Netw. Secur, 19(5), 653-659. doi: 10.6633/IJNS.201709.19(5).01)

Losev, K. (2020). The common data environment features from the building life cycle perspective. IOP Conference Series: Materials Science and Engineering, 913(4), 042012. doi: 10.1088/1757-899x/913/4/042012

M. Jakobsson, A. Juels, Proofs of Work and Bread Pudding Protocols (Extended Abstract), Springer US, Boston, MA, 1999, pp. 258–272. doi:10.1007/978-0-387-35568-9 18.

Marcus, S., Poitiers, N., & Weil, P. (2022). The decoupling of Russia: software, media and online services. Retrieved 7 April 2022, from https://www.bruegel.org/blog-post/decoupling-russia-software-media-and-online-services

Maurer, I. (2010). How to build trust in inter-organizational projects: The impact of project staffing and project rewards on the formation of trust, knowledge acquisition and product innovation. International Journal of Project Management, 28(7), 629-637. doi: 10.1016/j.ijproman.2009.11.006

McAfee (2019). Cloud Adoption and Risk Report. Retrieved 18 May 2022, from https://www.mcafee.com/enterprise/en-us/assets/skyhigh/white-papers/cloud-adoption-risk-report-2019.pdf

Merkle, R. (1988). A Digital Signature Based on a Conventional Encryption Function. Advances in Cryptology — CRYPTO ’87, 369-378. doi: 10.1007/3-540-48184-2_32

Mishra, N. (2016). Data localization laws in a digital world: Data protection or data protectionism?. The Public Sphere: Journal of Public Policy, 4(1), 135–158.

Moon, S., Shin, Y., Hwang, B., & Chi, S. (2018). Document Management System Using Text Mining for Information Acquisition of International Construction. KSCE Journal of Civil Engineering, 22(12), 4791-4798. doi: 10.1007/s12205-018-1528-y

Mukherjee, P., & Pradhan, C. (2021). Blockchain 1.0 to Blockchain 4.0—The Evolutionary Transformation of Blockchain Technology. Intelligent Systems Reference Library, 29-49. doi: 10.1007/978-3-030-69395-4_3

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System.

Nizamuddin, N., Salah, K., Ajmal Azad, M., Arshad, J., & Rehman, M. (2019). Decentralized Document Version Control Using Ethereum Blockchain and IPFS. Computers &Amp; Electrical Engineering, 76, 183-197. doi: 10.1016/j.compeleceng.2019.03.014

Oracle, Aconex, Available at, https://www.oracle.com/industries/construction-engineering/aconex/. Last accessed on 15 April 2022

Otte, P., de Vos, M., & Pouwelse, J. (2020). TrustChain: A Sybil-resistant scalable blockchain. Future Generation Computer Systems, 107, 770-780. doi: 10.1016/j.future.2017.08.048

Parasuraman, K., Srinivasababu, P., Angelin, S., & Devi, T. (2014). Secured document management through a third party auditor scheme in Cloud Computing. 2014 International Conference on Electronics, Communication and Computational Engineering (ICECCE). doi: 10.1109/icecce.2014.7086643

Perforce, The case for better document collaboration, knowledge worker survey, http://info.perforce.com/rs/perforce/images/versioning-report.pdf,Perforce Software, Last Accessed 21 May 2022.

Ping Tserng, H., & Lin, Y. (2004). Developing an activity-based knowledge management system for contractors. Automation in Construction, 13(6), 781-802. doi: 10.1016/j.autcon.2004.05.003

Plusquellec, T., Lehoux, N., & Cimon, Y. (2017). Design-Build and Design-Bid-Build in Construction- a Comparative Review. 25Th Annual Conference of the International Group for Lean Construction. doi: 10.24928/2017/0057

Saraiva, J., & Silva, A. (2009). WebC-Docs: A CMS-based Document Management System. KMIS 2009 - Proceedings of the International Conference on Knowledge Management and Information Sharing, Funchal - Madeira, Portugal

Shehab, T., Moselhi, O., & Nasr, E. (2009). Barcode-Assisted System for Document Management of Construction Projects. International Journal of Construction Education and Research, 5(1), 45-60. doi: 10.1080/15578770902717592

Sonmez, R., Ahmadisheykhsarmast, S., & Güngör, A. (2022). BIM integrated smart contract for construction project progress payment administration. Automation in Construction, 139, 104294. doi: 10.1016/j.autcon.2022.104294

Sonmez, R., Sönmez, F.Ö. & Ahmadisheykhsarmast, S. (2021). Blockchain in project management: a systematic review of use cases and a design decision framework. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-021-03610-1

Swan, M. (2015) Blockchain Blueprint for a New Economy, Sebastopol, CA, USA:O'Reilly Media

Szabo, N. (1996). Smart Contracts: Building Blocks for Digital Markets. EXTROPY: The Journal of Transhumanist Thought, 16(18), 1–11.

Tao, X., Das, M., Liu, Y., & Cheng, J. (2021). Distributed common data environment using blockchain and Interplanetary File System for secure BIM-based collaborative design. Automation in Construction, 130, 103851. doi: 10.1016/j.autcon.2021.103851

Tao, X., Liu, Y., Wong, P., Chen, K., Das, M., & Cheng, J. (2022). Confidentiality-minded framework for blockchain-based BIM design collaboration. Automation in Construction, 136, 104172. doi: 10.1016/j.autcon.2022.104172

Taylor, Richard. (2020). "Data localization": The internet in the balance. Telecommunications Policy. 44. 102003. 10.1016/j.telpol.2020.102003.

V. Buterin. Ethereum Sharding FAQ. Accessed: Jan. 28, 2020. [Online]. Available: https://github.com/ethereum/wiki/wiki/Sharding-FAQ

Van Wormer, G., & Larkin, M. (2017). Document management for design engineering, construction, and owner operators for the new enterprise. 2017 Annual Pulp, Paper and Forest Industries Technical Conference (PPFIC), 1-8. doi: 10.1109/ppic.2017.8003856

Wang, F., & Wu, Y. (2020). Keyword Search Technology in Content Addressable Storage System. 2020 IEEE 22Nd International Conference on High Performance Computing and Communications; IEEE 18Th International

Conference On Smart City; IEEE 6Th International Conference On Data Science and Systems (HPCC/Smartcity/DSS). doi: 10.1109/hpcc-smartcity-dss50907.2020.00095

Wang, H.-W., (2020). On Sharding Blockchains, Available at: https://github.com/ethereum/wiki/wiki/Sharding-FAQ.

Wang, S., Zhang, Y., & Zhang, Y. (2018). A Blockchain-Based Framework for Data Sharing with Fine-Grained Access Control in Decentralized Storage Systems. IEEE Access, 6, 38437-38450. doi: 10.1109/access.2018.2851611

Wust, K., & Gervais, A. (2018). Do you Need a Blockchain?. 2018 Crypto Valley Conference on Blockchain Technology (CVCBT). doi: 10.1109/cvcbt.2018.00011

Xie, J., Tang, H., Huang, T., Yu, F., Xie, R., Liu, J., & Liu, Y. (2019). A Survey of Blockchain Technology Applied to Smart Cities: Research Issues and Challenges. IEEE Communications Surveys &Amp; Tutorials, 21(3), 2794-2830. doi: 10.1109/comst.2019.2899617

Yavuz, E., Koc, A., Cabuk, U., & Dalkilic, G. (2018). Towards Secure e-voting Using Ethereum Blockchain. 2018 6th International Symposium on Digital Forensic and Security (ISDFS). doi: 10.1109/isdfs.2018.8355340

Zhang, P., & Zhou, M. (2020). Security and Trust in Blockchains: Architecture, Key Technologies, and Open Issues. IEEE Transactions On Computational Social Systems, 7(3), 790-801. doi: 10.1109/tcss.2020.2990103

Zhao, X., Zhang, Z., Hu, R., Liu, J., Yang, X., Zhang, R., & Gao, H. (2022). Blockchain Technology Based Digital Document Management System Design. 2022 7th Asia Conference on Power and Electrical Engineering (ACPEE), 440-446. doi: 10.1109/ACPEE53904.2022.9783819

Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017). An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. In 2017 IEEE International Congress on Big Data (BigData Congress) (pp. 557–564). https://doi.org/10.1109/BigDataCongress.2017.85