Bachelor’s degree Project

Smart contracts for real estate transactions: a solution to existing problems?

Author: Rohaan Samdani
Supervisor: Daniel Toll
Semester: VT 2021
Subject: Computer Science

Abstract

Blockchain technology has become increasing popular with the advent of cryptocurrencies like Bitcoin. However, the technology lends itself to greater opportunities due to its inherent properties such as security and decentralization, and the possibility to transact without the need for intermediaries. Real estate is a massive industry which could greatly benefit from the use of blockchain technology. The current processes for purchasing property involve multiple third parties, all of whom charge a fee and delay the transaction process. Furthermore, once a property is purchased, the ownership records are insecurely recorded in centralized databases. These records are subject to loss, damage, or corruption due to negligence or fraudulent activity. This thesis investigates if blockchain smart contracts could be a feasible solution to these problems. The investigation consisted of a literature study to identify the potential benefits and limitations of smart contracts for property transactions. Based on the findings, a prototype was then developed to facilitate a key part of the property transaction process. The results demonstrated that blockchain smart contracts could address several crucial issues faced by the real estate industry, however, more work would need to be done before they could be commercially implemented on a large scale.

Keywords: blockchain, smart contracts, decentralization, real estate, transactions

Contents

1 Introduction

Since the inception of the first blockchain in 2008, the field has grown rapidly. Several new blockchain technologies have been developed, creating an industry worth over $US4b at the time of writing [1]. One prominent development in the field has been smart contracts. Smart contracts are a self-enforcing, digital contract managed by the blockchain [2]. Smart contracts have the potential to benefit a range of industries thanks to their various features such as efficiency, security, and transparency. Industries such as finance and healthcare have already begun to incorporate blockchain based smart contracts.

Another industry that could benefit from the use of smart contracts is real estate. At present, transactions between buyers and sellers of a property are typically facilitated by multiple third parties such as lawyers and brokers. Additionally, the ownership records of a property are stored insecurely in centralized databases. These are prone to loss, damage, or corruption. In such instances, it becomes difficult to verify the true ownership history of a property. This leads to property disputes and fraud [3].

This bachelor thesis in Computer Science aims to determine if blockchain based smart contracts could be a viable alternative to the current real estate transaction process. The investigation consisted of a literature study and an experiment conducted on a smart contract protype that was developed.

1.1 Related work

Numerous scientific papers have been published on the blockchain and smart contracts. However, in terms of smart contracts for real estate transactions, the publications are limited. A relevant piece of work is a conference paper by Bhanushali et al. [4] titled *Blockchain to prevent fraudulent activities: Buying and selling property using blockchain*. The paper proposed a system where buyers and sellers of a property could register themselves and conduct their transactions securely. The system would run on the blockchain and use smart contracts to facilitate the transactions. The result of the paper was a built front-end for the users with the ability to run smart contracts. The smart contract itself had not been implemented.

Another paper published by Karamitsos et al. [5] provided an overview of the blockchain and smart contracts. The paper then presented a use case for renting properties using blockchain smart contracts. The paper concluded that smart contracts and the blockchain could benefit real estate by allowing trustless transactions between entities.

Casino et al. [6] conducted a systematic literature review on blockchain based applications. The paper described various components of the blockchain before giving an overview of blockchain applications in different industries. Casino highlighted open issues with the blockchain and concluded that as the technology matures it would be incorporated into more and more industries. Parjuangan et al. [7] investigated the different smart contract platforms by means of a systematic literature review. They compared several popular smart contract platforms such as Ethereum and Cardano and concluded that there were five characteristics a platform should fulfil: speed, user-friendliness, stability, security, and reliability.

A detailed literature review is presented in Chapter 4 of this work. Of the papers mentioned, none have directly addressed the issues related to conventional property transactions between a buyer and a seller. The paper by Bhanushali et al. [4] made a start on the problem but did not develop a smart contract to complete the system. As it stands, a smart contract focused on buy/sell transactions for non-commercial properties is still needed.

1.2 Motivation

This thesis will advance the field of decentralized technologies. By investigating and providing a potential use case for blockchain smart contracts in one of the largest industries worldwide. The results should motivate further development in the field and identify areas for continued research. Although this work focuses on a particular area, the results should be applicable to others as well. From a societal perspective, the work could indicate a more cost effective, efficient, and secure method for conducting property transactions. This would also benefit the real estate industry by reducing the reliance on third parties and automating certain processes.

1.3 Problem formulation

Presently, real estate transactions are typically facilitated by multiple third parties such as lawyers or brokers. These parties charge a fee for their service which increases the cost for buyers and sellers. Additionally, ownership is not recorded or stored securely following the transaction. Should it somehow be lost or corrupted, it may be problematic to verify who the true owner of the property is. This opens the opportunity for unpleasant disputes and legal battles.

Could blockchain smart contracts be a viable alternative to the current approach by solving these issues? This project aims to investigate this problem. Three research question were proposed to guide the study. Chapter 2 describes these in more detail.

RQ1: How can blockchain smart contracts benefit property transactions?

RQ2: What are the limitations of using smart contracts to facilitate property transactions?

RQ3: What is the cost of using smart contracts to facilitate the down payment on a property?

1.4 Scope/Limitation

Due to the time constraints for this investigation, some limitations were needed. Firstly, there are several smart contract development platforms that could be investigated. The project focused only on Ethereum. This is because the Ethereum platform is well established and provides a web interface for deploying smart contracts on a virtual blockchain. This will negate the need to use real currency to test the artefact.

Moreover, since a front-end had already been developed by Bhanushali et al. [4], the project did not create any front-end. Additionally, the smart contract that was developed addressed a particular subset of the property transaction process. Namely, the down payment and subsequent transfer of ownership. Rental transactions and mortgage payments were not included. However, the results should be generalizable enough to provide insights into the applicability for other transaction types. The contract created was a prototype to prove the use case. As such, the functionality was limited and not intended for immediate commercial use.

Furthermore, the literature review was restricted to peer-reviewed scientific papers published in ACM or IEEE. A set of inclusion and exclusion principles were followed during the study collection. These are described in Chapter 4.

1.5 Target group

This work is targeted towards researchers and developers within the field of computer science. Specifically, those who may be interested in decentralized technologies and their potential applications in the real world. Individuals and organizations in the real estate industry may also benefit from this work.

2 Method

This chapter presents the research methodology that was used to investigate the problem. The first section gives an overview of the project and objectives. Sections *2.2* and *2.3* describe the research methodologies that were followed to answer the research questions. Section *2.4* describes factors relating to reliability and validity of the work.

2.1 Research Project

As stated in Chapter 1, the area of investigation for this project was the use of smart contracts for real estate transactions. The primary goal was to see if smart contracts could be a viable alternative to the current approach by solving certain issues. Three research questions were defined, to guide the investigation. A multimethod approach was chosen to answer these. This is where a combination of research methodologies are followed. Two methods were selected: a literature review to answer the first two research questions and design science for the third.

The result after applying the two research methods was a prototype in the form of a smart contract and a combination of textual and numeric data to answers the research questions. The following sections describe the methodologies in more detail.

2.2 Literature Review

The aim of a literature review is to collect data from published sources such as peer-review journals or articles on a particular subject [8]. The collected data is then synthesized to give an overview of the subject area. This is done following a reliable and commonly used methodology, hence why it was selected for this part of the project.

An alternative research method that was considered but deemed unsuitable was interviews. This was due to accessibility. Literature on the blockchain is widely available online and provides the necessary amount of detail required for this project. However, accessibility to experts in the field was much more limited and the process of conducting interviews is more time consuming. Additionally, the interview process itself may not have allowed for the same level of information retrieval to be achieved.

The literature review was conducted following these five activities [9]:

1. Identify research questions
2. Define search strategy
3. Primary study selection

4. Inclusion and exclusion criteria
5. Data extraction

During the first stage, the aim of the project was broken down into two research questions. In the second stage, a search strategy was determined to find relevant literature. A primary study selection was then conducted to remove any non-relevant literature. The fourth stage of the process was then concerned with filtering out the remaining literature based on specific inclusion and exclusion criteria. The final stage of review process was to extract relevant data from this subset of literature in order to answer the research questions.

The outcome of each stage is described in Chapter 4.

2.2.1 Theoretical Background

In addition to the literature review, a few more peer-reviewed articles were found. These were selected to be used as background information sources on the concepts necessary for this project. The concepts included were blockchain and smart contracts. They were found by searching the same digital libraries described in Chapter 4.1.1 using the search terms “blockchain” and “smart contracts”.

Other information sources that were utilized for the theoretical background were the official Ethereum website. The theoretical background is available in Chapter 3, before the literature review. Table 2.1 contains the objectives for the background study.

O1	Define blockchain and summarize how it works.
O2	Define smart contracts and summarize how they work.
O3	Give an overview of the Ethereum smart contract platform.

Table 2.1 *Objectives for Theoretical Background section*

2.3 Design science

To answer the third research question – ***What is the cost of using smart contracts to facilitate the down payment on a property?*** – an experiment was conducted whereby a smart contract prototype was developed and evaluated following the design science methodology. Peffers et al. [10] describes design science as a frequently used framework for developing artefacts. Therefore, design science was the most suitable approach for this part of the project. It consisted of the following six activities:

1. Problem identification and motivation
2. Defining objectives for a solution
3. Design and development of the artefact
4. Demonstration
5. Evaluation
6. Communication

Activity 1 was concerned with defining the problem and motivating the value of the solution. The second activity involved breaking down the problem into specific objectives for a solution. Activity 3 was the actual creation of the prototype which in this case, was a piece of software. The functionality was also defined here. For this thesis, a single iteration had been deemed sufficient for the development of the prototype. Activity 4 is where demonstration took place to show that it solved one or more of the problems. The evaluation stage compared the resulting prototype with the desired objectives. Finally, the importance of the problem and effectiveness of the solution were communicated to the relevant target audience [10].

The demonstration was done by running a simulation of a transaction using dummy data. The entire process was documented in Chapter 5.

2.4 Reliability and Validity

Given the two research methods outlined above, it should be possible for others to produce the same results that are found in this report. This would mean that the report is reliable. To achieve this, a detailed search strategy was defined. This included the databases searched, keywords used, date of search and reference to the literature. For the development of the artefact, all activities were recorded and described in Chapter 5. The steps for demonstration and evaluation were described, including the data used. Additionally, the source code of the artefact is available in Appendix B. To further reduce reliability issues only official and peer-reviewed literature was referenced. This includes the official Ethereum website. Information sourced from here is guaranteed to be accurate and reliable.

For this work to be valid, the results must be supported by the data collected. One potential threat in achieving this is human bias. This is especially true when it comes to the use of grey literature. This is because grey literature often has not gone through the stringent peer-review process. Therefore, it has a higher probability of including bias. The use of any bias material would significantly reduce the validity of this project. To minimize this risk, no grey literature was used in the literature study unless necessary.

Moreover, the implemented prototype was deployed and evaluated on a virtual blockchain. This may produce threats to the validity of the resulting data when compared to a real blockchain. However, it was not possible to test the contract using real gas. Nonetheless, the result should still give a clear and accurate indication of the cost.

2.5 Ethical Considerations

During this project, two research methodologies were used. Neither of which entailed any ethical dilemmas. For the literature study, only publicly available material sourced from online databases was be used. These materials are free for anyone to use, given that they are correctly referenced. For the design science approach, a new artefact was created from scratch and tested with dummy data. No ethical considerations were needed.

3 Theoretical Background

The following chapter contains the relevant background information needed to understand this investigation. This chapter realizes the objectives set in Table 2.1: to describe the blockchain, smart contracts, and Ethereum network.

3.1 The Blockchain

Blockchain is the technology that forms the basis of this research project. It is also the principle underlying technology behind well-known cryptocurrencies such as Bitcoin and Ethereum. Blockchains are a **decentralized** ledger which is essentially a database or a growing a list of digital records. They allow users to securely store and transfer data or currency. Users can do this directly, without the need for any intermediaries. This means that transactions can occur without requiring a third-party such as a bank to facilitate. Additionally, any data or transactions that are recorded on the blockchain are tamper-proof. This is because the blockchain is **immutable**, meaning that once something is recorded it cannot be changed [11].

The term blockchain stems from the idea that data is stored in information blocks which are securely linked together via cryptography to form a chain. As the volume of data increases, more blocks are appended to the chain. Each additional block makes it more difficult to alter previous blocks, thus making the chain more secure. There are several characteristics of the blockchain that make it useful for different industries. Primarily, they are decentralized, meaning there are no central figures who control the network. They are distributed across a network of computers, called nodes which maintain the network. Users can store digital assets such as cryptocurrencies or documents and have complete control over them using their **private key**. In effect, blockchains gives control back to individuals by removing the need for third parties to manage their assets [11].

Consensus mechanisms are another vital feature. This is how blockchains make decisions without a central authority. Consensus mechanisms are the protocols followed by all the nodes in the network to come to an agreement about the current state of the ledger. Before any block can be added, the nodes must agree and validate that all transactions in the block are legitimate. There are various consensus algorithms employed by different blockchains such as proof-of-stake and proof-of-work.

Although there are a multitude of different blockchains, they generally follow the same steps when completing transactions. These are as follows:

1. A transaction is recorded containing the digital signatures of the parties involved and other relevant information.

2. The nodes in the network check that the transaction is legitimate, using consensus mechanisms [11].

3. After verifying the transaction it is added to a block. The block has a unique hash code as well as the hash of the previous block.

4. Blocks contain several transactions, once full they are appended to the chain.

3.1.1 Block

Figure 3.1 illustrates a typical blockchain architecture. The figure shows a succession of blocks, which together keep record of every transaction. Each block contains a header which stores the hash of the preceding block, called the *parent block*. A block can only have one parent block. The only exception is the *genesis block* which is very first block in the chain. This stops blocks being added to the center of the chain, preventing fraudulent activities [11].

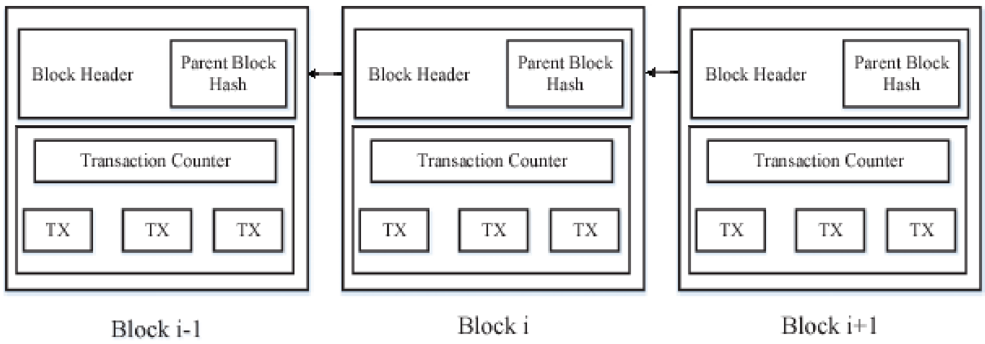

Figure 3.1 *Example of a blockchain, consisting of three consecutive blocks. Diagram taken from [11].*

3.1.2 Digital Signatures

The blockchain relies heavily on cryptography to maintain security. This is also true when it comes to authenticating transactions. Users of the blockchain are given a **public key** and **private key** [11]. Users can use these keys to digitally sign and authenticate transactions, such as when sending or receiving cryptocurrencies. Keys allow users to maintain full control of their digital assets and prove ownership.

The keys fit together as a pair. The private key, which must be kept confidential, can be used to authenticate a transaction. The public key, which may be known by anyone can be used to verify that this specific private key was truly the one who sent the transaction. This ensures transactions are legitimate and not sent fraudulently.

Public keys can be shared as they are essentially an address which people can use to send transactions to [11]. A private key is then used to unlock the transaction and verify the person is the true owner of the received data. However, if someone were to gain access to a user's private key, they would also have access to all the digital assets associated with that key. Users can store their private keys in secure wallets or write them down on paper.

3.1.3 Consensus Mechanisms

Consensus mechanisms are how nodes in a distributed network coordinate with each other. Independent nodes verify blocks and transactions following a predefined and self-enforcing set of protocols. This is how the blockchain establishes trust between unknown peers and comes to agreements on the current state of the network. It is the process which allows the blockchain to be decentralized [12].

Proof of Work (PoW) is the consensus mechanism used by the Bitcoin and Ethereum. It determines which node is selected as the next accepted block. The core idea is to solve and find solutions to complex mathematical algorithms. This process uses a lot of computing power and the first node to find a solution is selected to mine the next block [12]. The miners who operate these nodes are rewarded in cryptocurrency when their block is accepted.

In PoW, the calculations are to determine a hash value of the block header. For consensus to be reached, the calculated value must be within a given range. Once a node generates the requisite value, it is broadcasted to the whole network. The other nodes in the network must then validate the hash value. If it is deemed valid the block is appended to every node [12].

Proof of Stake (PoS) is the most common alternative to PoW. It is less energy intensive. Ethereum is planning to switch to PoS in 2021. PoS would turn mining into a virtual operation, with miners being replaced by validators. These validators would have to stake some of their currency since it is thought that individuals with more currency in the network are less likely to act malicously [12].

Validators are then chosen based on a combination of their stake size and other criteria depending on the specific PoS algorithm used. When a suitable block is found, validators essentially place a bet on it. If the block is selected and added to the chain, the validators are rewarded proportionate to their bet [12].

3.1.4 Types of Blockchains

The primary application of blockchain is to exchange information or execute transactions. The way in which this is done differs from case to case. To that end, there are different types of blockchains to accommodate varying use cases. Namely, public, and private blockchains [11].

One thing that every blockchain system has in common is that they all consist of a collection of nodes. These nodes work together as a peer-to-peer (P_2P) network and each node maintains a copy of the ledger [11].

Public blockchains such as Bitcoin and Ethereum are permission-less. Anyone can join the network and become an approved node. They can then access all records on the blockchain and verify transactions. This is one of the advantages of public blockchain systems. They are completely transparent since every node has a copy of the ledger and can be viewed by all participating nodes [11].

Another advantage of public blockchains is trust. Even though individual nodes in the network may not trust each other, consensus mechanisms ensure each node is authentic and all transactions are legitimate. Security is another benefit of public blockchains. This is because there can be any number of nodes in the network, unlike private blockchains. With more nodes, security increases as there is a greater distribution of the ledger and thus harder for hackers to attack the whole network [11].

Private blockchains operate in a closed network. Unlike public ones, they require permission to join. They are typically used by organizations where only a select few have access. The organization maintains full control of the network and it can be regarded as a centralized network. One advantage of private blockchains is their speed. Since there are a limited number of nodes, transactions can occur at a much faster rate [11].

The drawback of private blockchains is reduced security compared to public ones [11]. With fewer nodes in the network, it becomes easier for hackers to gain access to one node and subsequently control the whole network. Private blockchains are also centralized, which defeats one of the main purposes of blockchain technology.

To that end, public blockchains are more common. They are more secure and transparent [11]. However, there are still instances where a private blockchain may be more suitable.

3.2 Smart Contracts

Smart contracts are a type of digital contract which are deployed on a blockchain as part of its transactions [13]. They are designed to autonomously enforce agreements between non-trusting parties. Smart contracts are programmed by developers to execute when a set of pre-defined conditions are met. These agreements could for example involve the releasing of funds or transfer of ownership of a property or other asset.

Fundamentally, what smart contracts do is create trust between non-trusting parties via the execution of an agreement. Each involved party can be certain of the outcome, and it can be done without any intermediaries. Since there are no third parties involved, there is no paper-work or bureaucracy to go through. The contract is executed as soon as the conditions are met [13].

Another key point about smart contracts is that they are immutable. Once they are deployed on a blockchain they cannot be altered. This means anyone with fraudulent intent would not be able to tamper with the contract. Moreover, since the contract is executed on a blockchain, every transaction is digitally recorded on the blockchain ledger [13].

To establish the terms of a contract, participants must agree on a set of conditions. The contract can then be programed by a developer based on the user's needs. Smart contracts are deployed on the blockchain like any transaction. Users must pay a fee in the form of cryptocurrency to deploy and interact with a smart contract [13]. There are several platforms for developing smart contracts. The most popular are Ethereum and Hyperledger Fabric [14].

3.3 Ethereum

Ethereum is a permission-less blockchain with its own native cryptocurrency, Ether [14]. In addition to being used for value exchange, Ethereum aims to give users more control over their data and finances through decentralization and smart contracts [15]. Ethereum is one of the first blockchain-based platforms to introduce digital smart contracts.

The Ethereum blockchain maintains a history of every transaction and every smart contract. Each time a smart contract call is executed, the nodes in the Ethereum network process it and ensure the protocols are followed [15]. The Ethereum Virtual Machine (**EVM**) is the core of the Ethereum network. Each node has an EVM which is used to execute smart contracts. The smart contracts source code is first converted to bytecode which is then interpreted by the EVM [16].

For a transaction or smart contract to be succesful, users must pay a fee. This is paid in ether. When Ether is used to execute smart contracts, it is referred to as gas. To store Ether, Ethereum utilizes accounts of which there are two types of accounts: externally owned accounts (EOA) and contract accounts. EOA are for normal users to store and transact Ether. Users control these accounts via their private key. Contract accounts store smart contracts. Users can trigger these contracts by sending ether to the relevant contract account [13].

Ethereum gas prices are represented by **gwei**. Each gwei is equivalent to 10^{-9} ether. Table 3.1 compares the different units of measurement, where **wei** is the smallest unit. Gas is ultimately a measure of how much computation is done. These gas fees are essential in keeping the network secure. By setting a limit to the number of computational steps a transaction can execute, the network can prevent infinite loops which may break the system. Additionally, the requirement of a fee inhibits bad actors from spamming the network [17].

Unit	Number of Ether (ETH)
Ether	1
Gwei	1,000,000,000
Wei	1,000,000,000,000,000,000

Table 3.1 *Ether, gwei and wei price comparision.*

To calculate the cost of a transaction, the gas price is multiplied by the amount of the gas used [17]. It is represented by the formulae:

*Transaction Fee = Gas Used * Gas Price*

Gas price refers to the cost of one unit of gas. It is expressed in gwei. Gas used is the cost of executing each step in the contract plus a base fee of 21,000 gas for every transaction. When deploying and executing transactions on Ethereum, a gas limit is set. This is the maximum amount of gas the user is willing to pay for the transaction. If the gas limit is set too low, the transaction may not execute [17].

4 Literature Review

To determine if smart contracts are a viable solution for property transactions, the following two research questions were defined:

- **RQ1:** How can blockchain based smart contracts benefit property transactions?

- **RQ2:** What are the limitations of using smart contracts to facilitate property transactions?

4.1 Steps to Conduct Literature Review

As mentioned in Chapter 2.2, a literature review methodology was adopted to answer these two questions. This chapter presents the specific steps that were taken to conduct the review as well the results from each stage.

4.1.1 Search strategy

The goal of a search strategy is to identify all relevant literature regarding the research questions [9]. The search strategy specifies which sources were searched and what terms were used during the search. As this was not an exhaustive systemic literature review, only two online databases were searched: IEEE Xplore and ACM digital library. These are well established archives, primarily used in the field of computer science.

After an initial trial search, a set of key search terms were identified (Table 4.1). These terms were grouped into search strings and inputted into the search engine of the two databases. The search was conducted on the 14th of July 2021.

Search Terms	Group 1	Group 2
1	Smart contract	Real estate
2	Blockchain	Property

Table 4.1 *Search terms*

Table 4.1 represents the search terms as two groups that are either similar or synonymous with each other. By combining these terms using the AND (∧) and OR (∨) operator as noted by [9], a search string was defined as follows:

$$([G1,T1] \vee [G1,T2]) \wedge ([G2,T1] \vee [G2,T2])$$

During the search, terms from group 1 were limited to the document title only. Terms from group 2 were unrestricted to all metadata. The number of papers found in each library are summarized in Table 4.2.

Library	Number of papers found	Publication date
IEEE Xplore	142	2017 – 2021
ACM	336	2015 - 2021
Total	**478**	

Table 4.2 *Number of papers found after applying search string to each library*

4.1.2 Primary study selection

Following the search strategy from the previous subsection, almost 500 papers were found. Most of which were not relevant to the investigation. An initial filtering of papers was made by firstly, removing any duplicates and studies published in both searched libraries.

Secondly, the titles of each paper were checked manually. Papers with titles that were not at all relevant to the area of investigation were removed. Additionally, the reference list of the remaining papers were checked to retrieve any missing studies. Two more papers were found that were not from either of the searched libraries. This left us with a total of 24 papers. Table 4.3 summarizes the number of papers remaining from each library as well as the two additional ones found.

Library	Number of papers	Publication date
IEEE Xplore	19	2017 – 2021
ACM	3	2019 - 2021
Research Gate	2	2018 - 2020
Total	**24**	

Table 4.3 *Number of papers remaining after primary study selection*

4.1.3 Inclusion and exclusion principles

To further refine the search, the remaining papers were checked against a predefined set of principles. These inclusion and exclusion principles were defined as follows:

Inclusion criteria:

1. The paper is based on the use of smart contracts and/or blockchain technology
2. The paper is focused on the application of the above-mentioned technologies within the real estate/property industry
3. The study has a clearly defined aim

Exclusion criteria:

1. Studies that were not published in a reputable conference or journal
2. Studies that are not relevant to the application of smart contracts and/or blockchain technology in the real estate/property industry
3. Studies that make no contribution to the area of investigation
4. Studies not in English

After applying the above inclusion and exclusion criteria, **13 papers** were deemed appropriate for the area of investigation.

4.1.4 Data extraction

Having selected the papers for review, a protocol for extracting relevant data was defined. Two types of data sets were retrieved [18]: firstly, data connected to the quality of the publication such as the authors and publication type. Second, data relevant to the area of investigation such as the benefits and limitations. This data was necessary to answer the research questions. Table 4.4 details the type of data extracted.

Related Type	**Extracted Data Points**
Publication data	Title, author(s), publication year, publication type, publisher, country
Research questions	Aim of study, problem domain, proposed solution, benefits & limitations

Table 4.4 *Data points extracted from selected studies*

The extracted data regarding the publications are presented in Appendix A, Table 1. A summary of the selected papers is presented in the following subsection, with data related to the research questions. A full list of benefits and limitations are available in Appendix A, Table 2. The findings were synthesized into some visualizations, presented in Section 4.3.

4.2 Summary of Reviewed Literature

The following section summarizes the 13 papers selected for review. Data regarding each of the publications is presented in Section 4.3.1.

4.2.1 A Blockchain Based Land Registration System Proposal for Turkey

The paper by A. F. Mendi, et al [19] focuses on the land registration system in Turkey. The aim of the paper is to solve the issues faced by the current land registration system. The primary issue is that with the current system there is a chance for tax fraud. This is a result of how the system is designed. When the owner sells a property, he or she must declare the sale value and pay tax on the profit. To avoid paying this tax sellers declare the sale price lower than it really was [19].

This is issue further perpetrated by the fact that transfer of money between the buyer and seller is primarily a physical transaction. This makes it difficult for the land registry office to get an accurate sales value. The paper proposed a blockchain based solution built on the Hyperledger system to try and solve these issues. The system incorporates all the parties involved; land registry office, bank, municipality, buyer, and seller into a digital system built on the blockchain. The result is a completely transparent and traceable alternative to the current method of land registration [19].

Another benefit of the blockchain system is that it reduces the physical contact between parties since everything is done digitally. The paper highlights this as an important advantage, especially during times of a pandemic like COVID-19. However, the system has limitations. Namely, that it cannot be implemented on a wide scale due to legal and legislative requirements that must be updated first. Additionally, the solution does nothing to reduce the number of third parties involved in the transaction [19]. A full list of benefits and limitations can be found in Table A.2.

4.2.2 A secured land registration framework on Blockchain

The study proposes a land registration framework built on the Ethereum blockchain. The purpose of the system is to solve the issues faced by the Indian land registration system. The first of these issues is that properly tracking land ownership records is very difficult. The current system is built on paper trails which can be easily forged or by centralized systems that can be altered. Fraudulent activities and incomplete records escalate these issues. This leads to property ownership disputes which can last for several years [20].

The proposed solution utilizes smart contracts built on the Ethereum network to securely track and transfer land ownership records. The benefits the system are increased security and safety of the records, due the blockchains immutable characteristic. The solution makes land registration records more transparent as they can be viewed and verified on the blockchain. The number of third parties involved in the process are also reduced via the use of smart contracts [20].

One limitation of the system is that transaction costs are uncertain due to the fluctuating price of ether. The system was evaluated on a small-scale test network and not implemented in a real-world situation. Based on the studies evaluation of 36 transactions, it found that each one would cost 0.36 USD. This is 1886 times less than it would cost using the traditional system [20].

4.2.3 A Transparent and Trusted Property Registration System on Permissioned Blockchain

The paper by T. Ali, et al [21] focuses on the issues presented by the Saudi Arabian property registration system. The first issue is that the current process is manual. Certain parts of it are digital, however, it is still centralized and does not maintain the complete ownership history. The system also does not have any mechanism for detecting malicious record tampering [21].

The paper proposes that a decentralized system, built on the blockchain could solve these issues [21]. The study offers a solution using the Hyperledger Fabric blockchain in which users can buy, sell, and update property records via the use of smart contracts. Due to the blockchains characteristics, this new system provides a tamper-proof and trusted solution. The system maintains a history regarding all previous buyers and sellers of a property, which can be verified on the blockchain. No limitations have been mentioned.

4.2.4 Blockchain to Prevent Fraudulent Activities: Buying and Selling Property Using Blockchain

The study by Bhanushali, et al [4] proposes a blockchain based solution to prevent fraudulent activities in property transactions. The solution uses smart contracts to facilitate the purchase and selling of a property. By digitally uploading the relevant information to the smart contract via a front-end, it would act as a self-enforcing third party to verify that the terms have been met. When the terms of the smart contract are met and verified the transaction is complete. This transaction is recorded on the blockchain and provides a secure and transparent method of verifying the owner of a property [4].

The proposed solution aims to solve the problem of property fraud that arises due to a lack ownership records. Currently, ownership records or title deeds are managed by a central authority such as a government office. These records are not securely stored and may be lost or damaged. In such a case, there may not be a concrete way of proving who the owner of a property is [4]. The solution offered by Bhanushali, et al provides a client-side interface for registering properties to be sold. The system would use the cryptocurrency ether to pay for the smart contract transactions between the buyer and seller. However, the solution is incomplete as the smart contract was not implemented.

4.2.5 Blockchain-based Land Record Management in Pakistan

Similar to the issues highlighted in the paper by Nandi, et al [20], Pakistan's current method of land record management makes it difficult to get an accurate history of property ownership. The current approach is error-prone and subject to data tampering. Additionally, it requires buyers and sellers of a property to make visits to multiple locations in order to transfer and register ownership [22].

The solution proposed to solve these issues is a simple web application where users can create an account and register or transfer ownership of their property. The data is then recorded on the Hyperledger blockchain which can then be verified in a transparent and secure manner. The implemented solution was built on a private blockchain to reduce costs as only a couple nodes were required to test the system. The system was tested using false data. To implement the solution in a real-life scenario, additional permissions are needed from the relevant departments [22]. Table A.2 provides a list of the benefits and limitations of the proposed system.

4.2.6 Business Process Models of Blockchain and South African Real Estate Transactions

The study aims to solve the following issues with the South African real estate transaction process: inefficiency, high transaction costs and long transaction times due to the reliance on multiple third parties [23]. Further issues stem from the need to manually review and update multiple systems. This increases the risk of error and fraudulent activities.

By conducting a case study of blockchain technology being used for real estate transactions, the study was able to produce a conceptual model for the South African market. The findings showed that smart contracts could solve the issues mentioned by increasing the speed of transactions and reducing costs. This was due to the smart contract reducing the reliance on third parties. The smart contracts also reduce the risk of fraud as it allowed for payments to be made concurrently. Transparency was another key benefit. Every stage of the transaction was publicly available for all stakeholders [23].

The study also found that there were five challenges or limitations. Firstly, adoption of the technology. For this to occur, there would need to be trust and support from every stakeholder before a finalized system could be developed. Additionally, the reliance on digital systems would reduce the flexibility as compared to human involved systems. Like many other papers, a legal framework for blockchain technology in real estate transactions would also need to be implemented [23]. Table A.2 summarizes the list of benefits and limitations.

4.2.7 Cheat Proof Escrow System for Blockchain

The study proposes an algorithm for a cheat proof escrow system. The algorithm can be applied to property transaction between a buyer and seller. The study highlights that during property transactions, the involved parties do not want to transfer payment or property until the other party has done so first. Escrow system have been created as a solution to this problem. However, these systems require a trusted mediator to facilitate the transaction. This mediator may act maliciously and transfer funds to their own account [24].

As such, the algorithm proposed uses three private keys to circumvent this. One key for the buyer, one for seller and one shared key between them. This shared key prevents the mediator from making any decisions with the consent of both buyer and seller. The algorithm provides security and guarantees a cheat proof escrow system. The algorithm could be implemented using blockchain technology to provide the encrypted keys [24].

4.2.8 Digitalization of Land Records using Blockchain Technology

The aim of this study was to determine if blockchain technology was a feasible solution to the existing land record management business. The study identified benefits of using blockchain as follows: accelerating the process of land registration, reducing the risk of fraud, and allowing transparency with the use of smart contracts [25].

Furthermore, the immutable and traceable characteristics of the blockchain allow for a full record of land ownership history to be recorded and maintained. This solves the issue of disputes occurring regarding who owns a property due to loss or inconsistent records in the current system [25].

4.2.9 Land Registry Using Blockchain - A Survey of existing systems and proposing a feasible solution

The study proposes a system architecture for a land registry application in India. The proposed system utilizes the Ethereum blockchain to store data regarding land ownership. Specifically, the system uses the Inter Planetary File System (IPFS) to securely store the property documents. IPFS is a decentralized database, which produce a unique hash value for each document. This hash value is then recorded on Ethereum blockchain following the verification of the smart contract conditions [26].

The mentioned solution solves the issues of fraud, data integrity and persistence faced by the Indian property market. For a full list of benefits, see Table A.2.

4.2.10 LandLedger: Blockchain-powered Land Property Administration System

The study identifies four primary issues with the current land administration system in India: accountability, transparency, security, and efficiency. Because of these issues, land records are often damaged or incomplete. The process is also slow due the records being maintained by multiple departments using their own separate systems. The study proposes a land administration system using the Hyperledger blockchain to address these issues [27].

LandLedger system incorporates the relevant departments such as the income tax department and registrar's office, as well as the buyer and seller of the land. The system allows users to register properties and verify ownership while maintaining traceability, accountability, and security. The system is also designed to easily integrate into the current property management systems [27].

4.2.11 Real Estate Management System based on Blockchain

This study takes a slightly different approach to the others regarding the Indian real estate system. It proposes a hybrid centralized/decentralized system for managing land registry. The system provides a solution to the previously mentioned issues of the current system: transparency, multiple third-parties, fraud, data corruption/loss, efficiency, and security [28].

The system works by incorporating all the necessary departments for land management into a centralized system for synchronous and real time update of data - this is in addition the buyer and sellers of a property. The actual data is stored and accessed from a decentralized blockchain network making it secure and immutable. The system uses smart contracts and digital signatures to ensure integrity and trust [28]. Although the system does make some progress to reduce the number of intermediaries involved in transactions, it is still not a direct transaction between buyers and sellers of a property.

4.2.12 Design of the Blockchain Smart Contract: A Use Case for Real Estate

The paper provides a design methodology for developing smart contracts as well as a use case for smart contracts in real estate. The use case involves the Ethereum blockchain and two primary actors: the property owner/landlord and the tenants who must pay rent. The contract is designed to ensure a rental agreement has been signed, rent is paid on time, and termination of the contract is properly done [5].

The landlord initiates the smart contract with the terms. Once terms have been agreed and verified by the tenants, all interactions such as signatures, payment and termination are conducted securely via smart contract. The goal of the paper was to provide one possible use case for blockchain smart contracts in the real estate industry. It concludes that real estate individuals must determine themselves if the technology is ready to be implemented within their organization [5].

4.2.13 Constraints and benefits of the blockchain use for real estate and property rights

The final paper from the review aims to highlight the benefits and constraints of blockchain in real estate from a more general perspective. First, it describes some possible uses of blockchain in real estate [29]:

1. Cryptocurrency for payments or to deploy smart contracts using gas
2. As a method of immutable data storage for property rights
3. Smart contracts for automated transactions
4. Tokenization of property titles
5. Decentralized applications (dApps)

This list is not exhaustive. The paper goes on to mention several constraints of blockchain use in real estate. One such constraint is immutability. The author states that this feature of the blockchain would be an issue as it would be difficult change or update mistaken data. He suggests that an architecture with overlaid technologies would be required to account for this [29].

Anonymity is another highlighted constraint, as governments would likely need to know who initiates a transaction to prevent criminal activities. Additionally, personal data published on the blockchain is there forever. Users would therefore need to be given a choice in which data they provide [29].

Price volatility of cryptocurrency and scalability of blockchain networks are two more concerns. Another major constraint is legal compliance and enforceability of smart contracts. Government approval would be needed which may not be possible in countries with high rates of corruption [29].

The author concludes by saying blockchain *could* meet the legal demands of property registries, but it would not be possible to transfer the entire system overnight. Instead, the traditional system would have to work in parallel with a blockchain based system. This would benefit users by giving them a choice in which system they want to use when conducting property related activities [29].

4.3 Results from Literature Review

Results from the literature review are presented in the following section. The first subsection reports data regarding the reviewed publications. Subsequent sections present answers to the first two research questions.

4.3.1 Publication Data

The initial search for literature regarding blockchain smart contracts and real estate, returned 478 papers as shown in section 4.1.1. Following the primary study selection and application of inclusion and exclusion criteria, **13 papers** were selected for review. Table 1, Appendix A contains a list of all 13 papers.

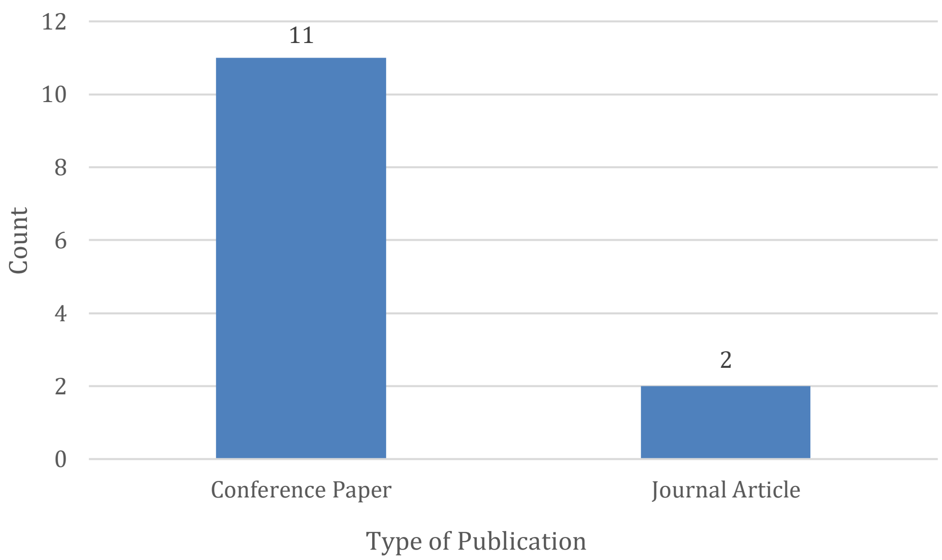

Figure 4.1 *Number of studies reviewed and their publication type.*

Most of the selected studies were IEEE conference proceedings as shown in Figure 4.1. Figure 4.2 shows the publication dates of the papers ranging from

2018 to 2021. Figure 4.3 contains the geographical location from which each study was published. The figure shows most papers were published in India. The remaining six countries published one paper each.

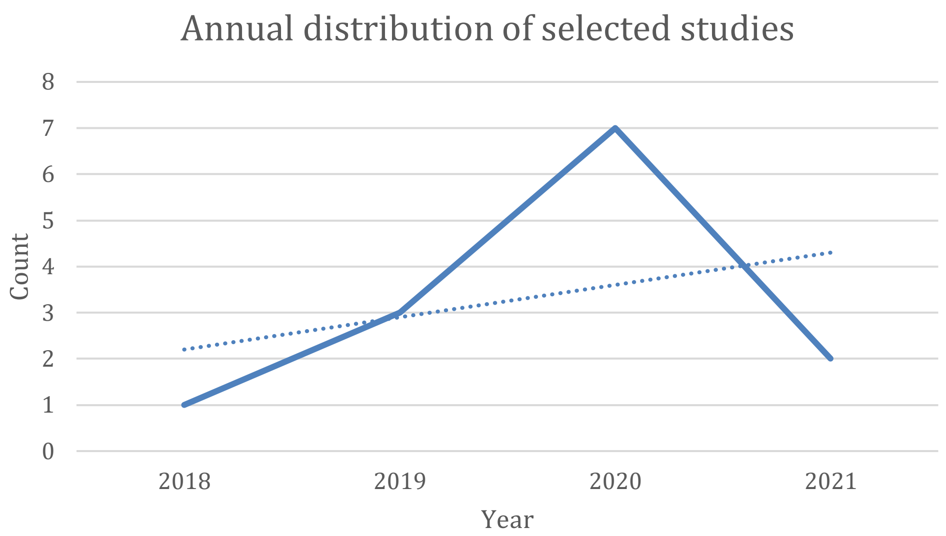

Figure 4.2 *Publication year of reviewed literature, with trend line showing a steady increase.*

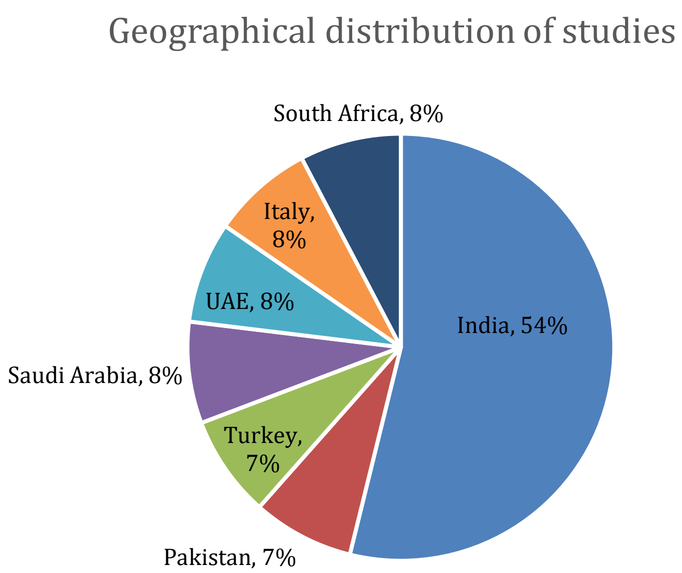

Figure 4.3 *Distribution of reviewed studies, based on geographical origin.*

Figure 4.4 below, outlines the main area of investigation for each selected study. Of the 13 papers reviewed, 8 were primarily concerned with management of ownership records using the blockchain smart contracts. Four studies focused on property transactions, and one was dedicated to the constraints and benefits of blockchain use in real estate. However, in most papers, a combination of the three instances were included to some degree. Thus, it was possible to extrapolate data regarding the benefits and limitations of blockchain smart contracts for property transactions from each study. This data is presented in the following sections.

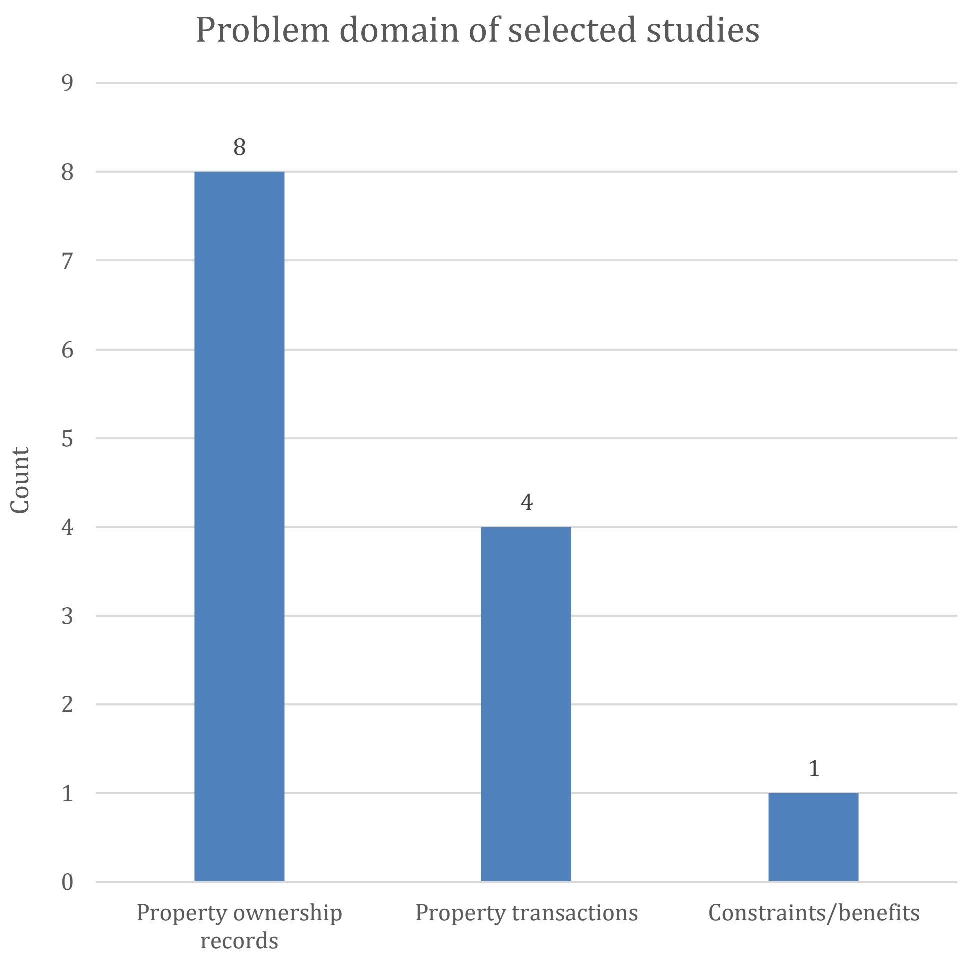

Figure 4.4 *Primary problem domain for each selected study regarding blockchain smart contracts in real estate.*

4.3.2 RQ1: Benefits

This section contains the results from the literature study regarding the first research question:

How can blockchain based smart contracts benefit property transactions?

Table 4.5 summarizes the benefits extrapolated from the selected studies. Figure 4.5 provides a visualization to emphasize the number of studies that highlighted each benefit.

Benefit	Explanation
Increased security	The blockchain provides increased security as compared to traditional property management systems and even banking systems. Smart contracts used for property transactions, inherit this security [4], [19], [20], [22], [24], [26], [27], [28].
Increased transparency	Transactions are recorded on the blockchain which can be viewed and verified by anyone [4], [19], [20], [21], [23], [25], [27], [28].
Increased traceability	Transactions can be traced on the blockchain preventing fraud and property ownership disputes [19], [27].
Reduced physical contact	Smart contracts allow property transactions to be done digitally and thus reduce physical contact which may be important during a pandemic [19], [22].
Reduce risk of fraud	Transactions and ownership records cannot be altered once recorded on the blockchain [19], [23], [25], [26].
Reduce intermediaries	Smart contracts remove the need for multiple intermediaries/third parties during the transaction process [4], [19], [20], [23], [25], [29].
Reduce transaction time	Smart contracts execute functions as soon as pre-defined conditions are met and cryptocurrency payment are transferred almost instantly [20], [23].

Table 4.5 *Summary of results regarding the benefits of using smart contracts for real estate property transactions.*

Benefit	Explanation
Potentially reduce transaction cost	By removing the need for third party intervention and thus the associated fees. Only the smart contract gas fees must be paid which are likely to be cheaper [20], [23].
Trust	Smart contracts act as a trusted third party. Buyers and sellers do not have to trust each other, only the contract [5], [21].
Decentralization	Smart contracts are inherently decentralized which provides a new method of conducting transactions without having to relying on centralized authorities [21], [22], [26].
Tamper proof	Once data is recorded on the blockchain, it can be tampered with [21], [22], [23], [26].
Provide integrity of records	Property ownership can be recorded securely on the blockchain which is tamper-proof and traceable this providing integrity [4], [21], [22], [25], [26], [28].
Increased efficiency	Smart contracts function autonomously and execute when pre-defined conditions are met. This allows for a more efficient transaction process that does not rely on paperwork to be filed or bureaucracy [5], [22], [25], [27], [28].
Scalability	Blockchains systems can be integrated into current property management systems which can then be scaled digitally [27].
Immutability	Once data is recorded on the blockchain it cannot be altered making it tamper proof and reducing the risk of property fraud [29].

Table 4.5 Continued.

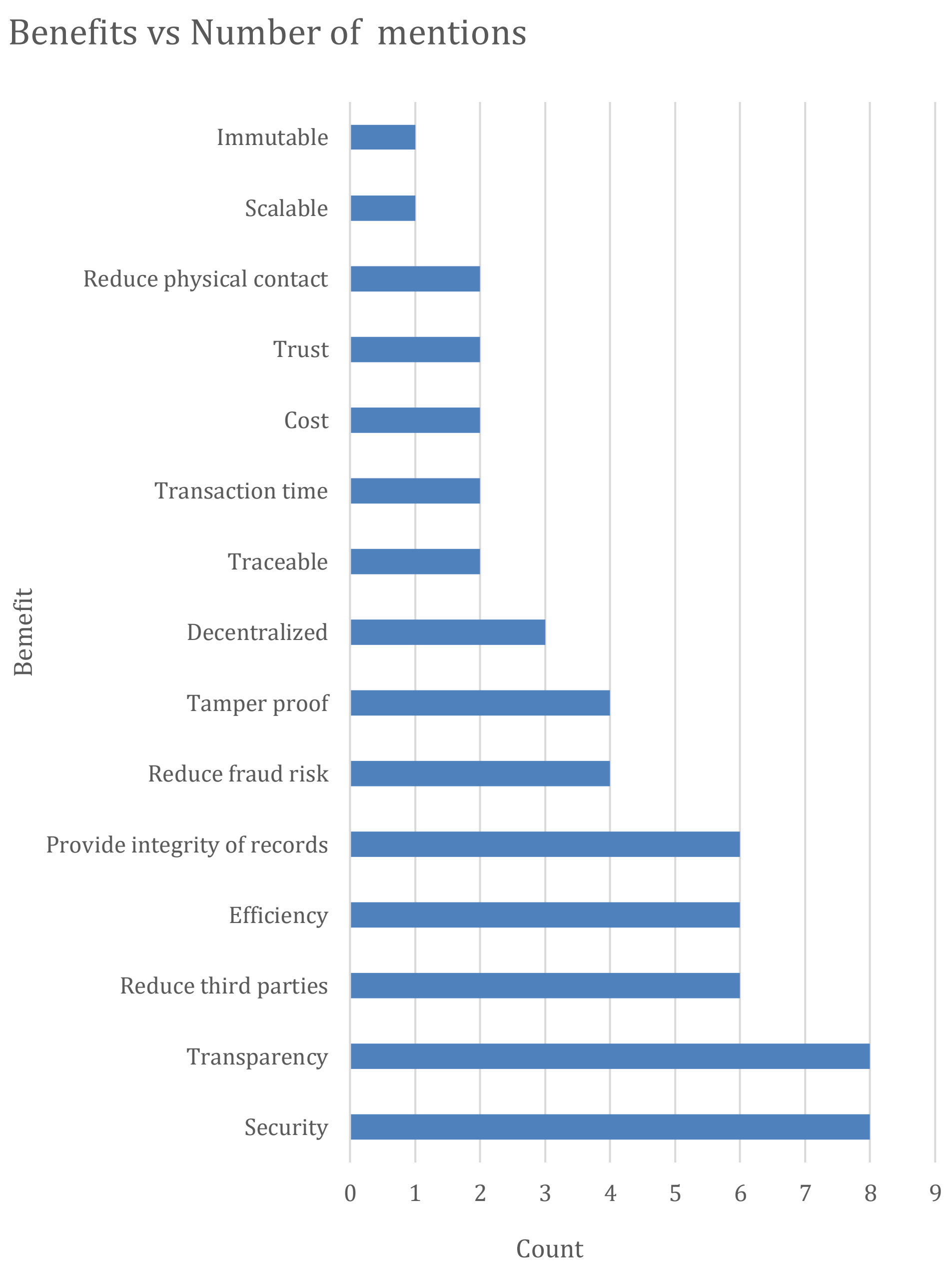

Figure 4.5 *Number of studies that mentioned each benefit of using smart contracts.*

4.3.3 RQ2: Limitations

This section presents the findings from the literature study regarding the second research question:

What are the limitations of using smart contracts to facilitate property transactions?

Limitation	Explanation
Security testing	Smart contracts require rigorous security testing before they can be deployed as any error that is found later cannot be fixed. This becomes even more important for property transactions as large amounts of money are being handled via the contract [4].
Uncertain transaction costs	Due to the volatile nature of cryptocurrencies, it may be difficult to determine the exact cost of a transaction ahead of time. Moreover, impairment loss may occur resulting in lost funds due this volatility [20], [29].
Implemented systems only work for individual countries and needs to be redesigned for use in others	Each country has their own specific requirements when buying/selling property. A blockchain smart contract system designed for one country may not work in another and therefore would need to be redesigned for each location [19].
Governmental compliance needed to implement on wide scale	In order for smart contracts to be implemented on a wide scale, government approval would be needed [19], [29].
Legal requirements must be accounted for	Smart contracts must be compliant with legal requirements before they can be used for property transactions [22], [23], [29]

Table 4.6 *Summary of results regarding the current limitations of using smart contracts for real estate property transactions*

Limitation	Explanation
Mass adoption	Achieving mass adoption of blockchain and smart contracts in the real estate industry may be difficult. This would limit the progression and utility of the technology [23].
Less flexibility	Smart contracts digitalize the transaction process which reduces the flexibility as compared to human involved systems. This may be an issue for property transaction which are subject to last minute changes [23].
Blockchain smart contracts may not cover entire organizations requirements	Real estate organizations have multiple systems and transaction processes. Smart contracts may not provide the flexibility needed to incorporate all of those systems [5].
Immutability	Once data is recorded on the blockchain, it cannot be changed even if there was an error, and the wrong data was recorded. Likewise, smart contracts cannot be updated once they are deployed. If an error is found, a new contract would have to be written and deployed. This contributes to the reduced flexibility and rigorous security testing that is needed [29]
Cost of storing data on Ethereum blockchain	Ethereum gas prices can often be very high. Conducting single transaction may not cost a lot, however, storing large data on the Ethereum blockchain such as ownership records can be very costly [4], [20].

Table 4.6 Continued.

5 Experimentation

As outlined in Chapter 2.3, an experiment was conducted to collect data with the aim of answering the third and final research question. The experiment involved the creation of a novel smart contract that was developed following the design science approach. The contract was executed and evaluated in a virtual environment using dummy data. The results of the experiment are detailed in section 5.4. The activities carried out to conduct the experiment are also described.

5.1 Problem

As mentioned previously, the traditional process for purchasing property involves multiple intermediaries. These intermediaries can slow down the transaction process and increase the cost burden for the buyer and seller. Endurthi et al [24] highlighted that they are necessary for the current real estate system to act as trusted mediators between the two parties. This is due to the fact that neither parties want to transfer payment or property until the other party has done so first.

Secondly, there is a risk of property fraud and disputes due to the insecure and centralized record management systems mentioned by [19], [20], [21], [22], [23], [25], [26], [27], [28]. These methods do not allow for data integrity to be retained.

To solve these issues, a smart contract prototype was created. Since real estate is a large industry with various forms of transactions, it was not feasible to create a solution involving all of them for this bachelor thesis. Instead, the focus was narrowed down to one important aspect. The initial down payment and subsequent transfer of ownership. By focusing on this phase of the transaction process, we could prove a potential use case for smart contracts in real estate and answer the final research question.

The findings could also provide a basis for further research and development. Additionally, the current research based on the literature review was lacking with regards to the initial down payment. As such, it was deemed a necessary research gap to investigate. The results should be generalizable for other transaction types as well.

5.2 Prototype

Having identified the problem, the objectives for a solution were outlined in the following subsection including the requirements and workflow of the protype. Subsequently, the tools necessary to conduct the experiment including development, evaluation, and demonstration are described.

5.2.1 Requirements

Following the literature review, a set of six requirements were identified. The ensuing section delineates the requirements that the prototype should fulfil. As the prototype was not intended for immediate commercial use, a limited functionality was implemented. The requirements identified were deemed necessary for the experiment and to collect data to answer the third research question.

Requirement 1

To solve the issues caused by multiple intermediaries, as identified in the literature study, the contract was designed to function with minimal stakeholders. Chiefly, the buyer and seller of a property. As such, we assume a scenario where the buyer and seller agree upon purchase terms externally. These terms are uploaded to the smart contract as part of its deployment parameters by the seller (**REQ1**).

Requirement 2

As mentioned in Chapter 3.1.2, every user of the blockchain has a public and private key. Public keys act as addresses which are used to send and receive digital assets. Private keys are needed to access and control these assets [11]. This is how **REQ2** was implemented. The public addresses of the buyer and seller were uploaded as part of the deployment terms. Consequently, they were the only ones who could interact with the contract. This was to maintain decentralization and remove the need for intermediaries. Furthermore, as identified in the literature study fraud was a major concern for property related transactions. If the buyer and seller retain full control of the contract and transaction process, the risk of fraud and malicious activity is reduced [3], [4].

Requirement 3

REQ3 to **REQ6** were deemed necessary to solve the issues of trust during property transactions. As stated by Endurthi, et al [24], current transaction processes require third party mediators such as lawyers to provide trust between the buyer and seller. This is to ensure the buyer does not loose their payment and the seller does not transfer ownership without receiving payment. To allow a decentralized solution to this problem, it is important that the buyer and seller can safely conduct the transaction. To ensure this, **REQ3** was identified. After the contract is deployed with the terms (keys, price, property address, deadlines etc.), the buyer must verify they are correct before the transaction can progress to the next stage. If the terms are not correct, the contract is rejected and goes no further.

Requirement 4 & 5

REQ4 and **REQ5** maintain this further throughout the transaction process. Firstly, the smart contract acts a secure escrow system [24]. The buyer can safely pay the down payment into the smart contract, without having to trust the seller or any third party. Secondly, the contract holds the funds until certain conditions are fulfilled. If those conditions are not met or the buyer does not verify, the payment is automatically returned to his/her wallet. This allows the buyer to retrieve the down payment at any stage of the transaction process.

Requirement 6

The final requirement maintains a trusted and decentralized solution for both the buyer and seller. Before the down payment is released from the smart contract to the sellers account, the buyer must make a final verification that all terms have been fulfilled. Likewise, before ownership is transferred to the buyer, the seller must also verify all conditions have been met. Only when both parties have done so, does the transaction complete. If either party does not verify, the transaction is cancelled, and payment is returned to the buyer whilst ownership is retained by the seller.

Table 5.1 summarizes the six requirements.

REQ1	Seller must deploy contract with the agreed terms
REQ2	Only buyer and seller are authorized to interact with contract
REQ3	Buyer must be able to verify or reject contract terms
REQ4	Buyer must be able to safely pay down payment
REQ5	Buyer must be able to retrieve payment if sellers does not fulfil terms of agreement
REQ6	Both buyer and seller must verify all terms met before payment and ownership transferred

Table 5.1 *Functional requirements for smart contract*

Figure 5.1 illustrates the workflow of the prototype.

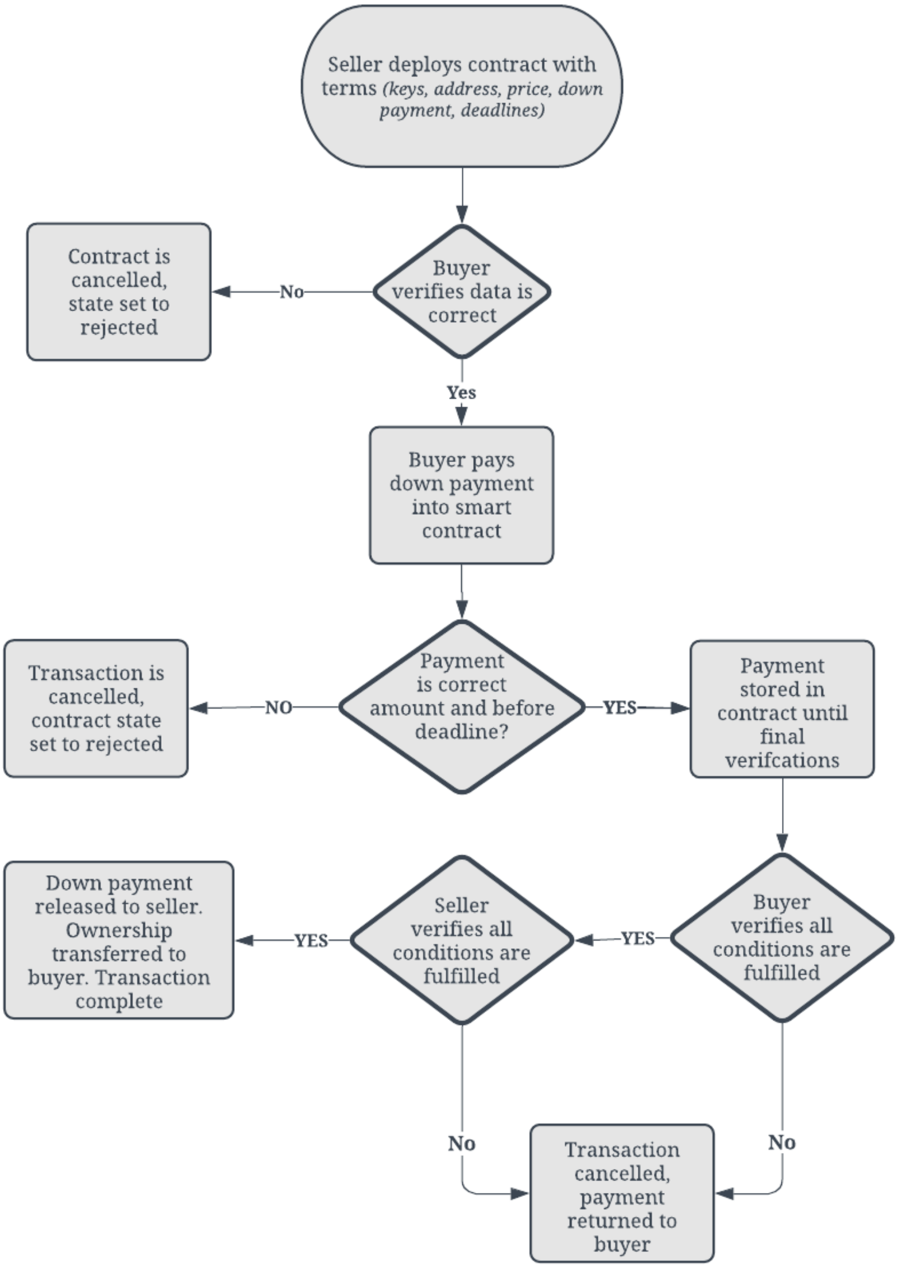

Figure 5.1 *Workflow diagram for the prototype, showing the outcome of each step.*

5.2.2 Development Tools

Ethereum was chosen as the smart contract development platform. Ethereum was the first platform introduced for smart contract development and remains one of the most popular. As such, it has proven to be a trusted and secure platform with a large community of developers.

Furthermore, Ethereum is a public blockchain [15] so anyone can become a user. Ethereum also maintains a complete transaction history, which is very difficult to tamper with [14]. This feature can be utilized for recording the ownership of a property to prevent fraud and disputes. Users can check the blockchain and be certain of the integrity of the information.

Another reason for choosing Ethereum, is that it provides a purpose-built programming language for writing smart contracts. Solidity [30], is Ethereum's native smart contract development language. Developers can write contracts in Solidity and deploy them on the Ethereum blockchain directly. Solidity has its own compiler which translates source code into bytecode, which can be read by the Ethereum virtual machine [17]. However, as mentioned in Chapter 3.3, deploying smart contracts costs gas.

Fortunately, Ethereum comes with an open-source web IDE called Remix [31]. Remix IDE allows developers to write, test, and deploy Solidity smart contracts in a virtual environment [31]. This meant we could deploy our smart contract on a virtual Ethereum blockchain without spending real gas.

5.3 Experiment Demonstration

This section describes the steps taken to demonstrate the prototype and collect data. The source code is found in Appendix B. The contract was deployed in a virtual environment using the Remix IDE, so it could be tested without requiring the use of real ether. The Remix compiler settings were set as shown in Figure 5.2.

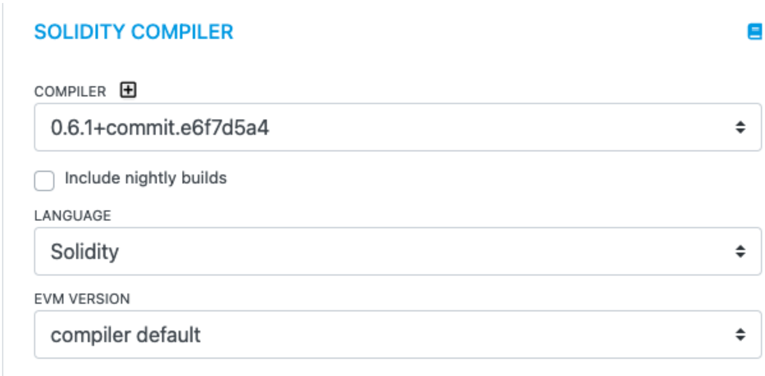

Figure 5.2 *Remix IDE compiler setting used for demonstration and evaluation.*

Figure 5.3 shows the environment settings where the JavaScript virtual machine was used. This allowed the simulation of a virtual blockchain within the browser. The contract could now be run.

First, the terms of the agreement were uploaded to the constructor before deployment as shown in Figure 5.3. The crucial part was the two wallet addresses. One for the seller and one for the buyer. Remix IDE provides several virtual addresses that could be selected at any time by clicking on the account tab. Only the authorized addresses may interact with the contract, otherwise the transaction would fail. Note that both the accounts start with 100 Ether.

- **Seller address**: *0xCA35b7d915458EF540aDe6068dFe2F44E8fa733c*
- **Buyer address**: *0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db*

The remaining deployment fields were filled with dummy data as shown in Figure 5.3. The value of the property and down payment were given in terms of ether. For testing purposes, a final price of 100 Ether was used and a down payment amount equivalent to 20% of the price. The down payment and final transfer deadlines were expressed in terms of Unix timestamps. These are the default time mechanisms used by Solidity smart contracts [30]. The dates given were for the 10th of October and November 2021, respectively. Finally, the gas limit was set to the default 3,000,000. Sending more gas than necessary is safe, as any unused are returned to the sender [17].

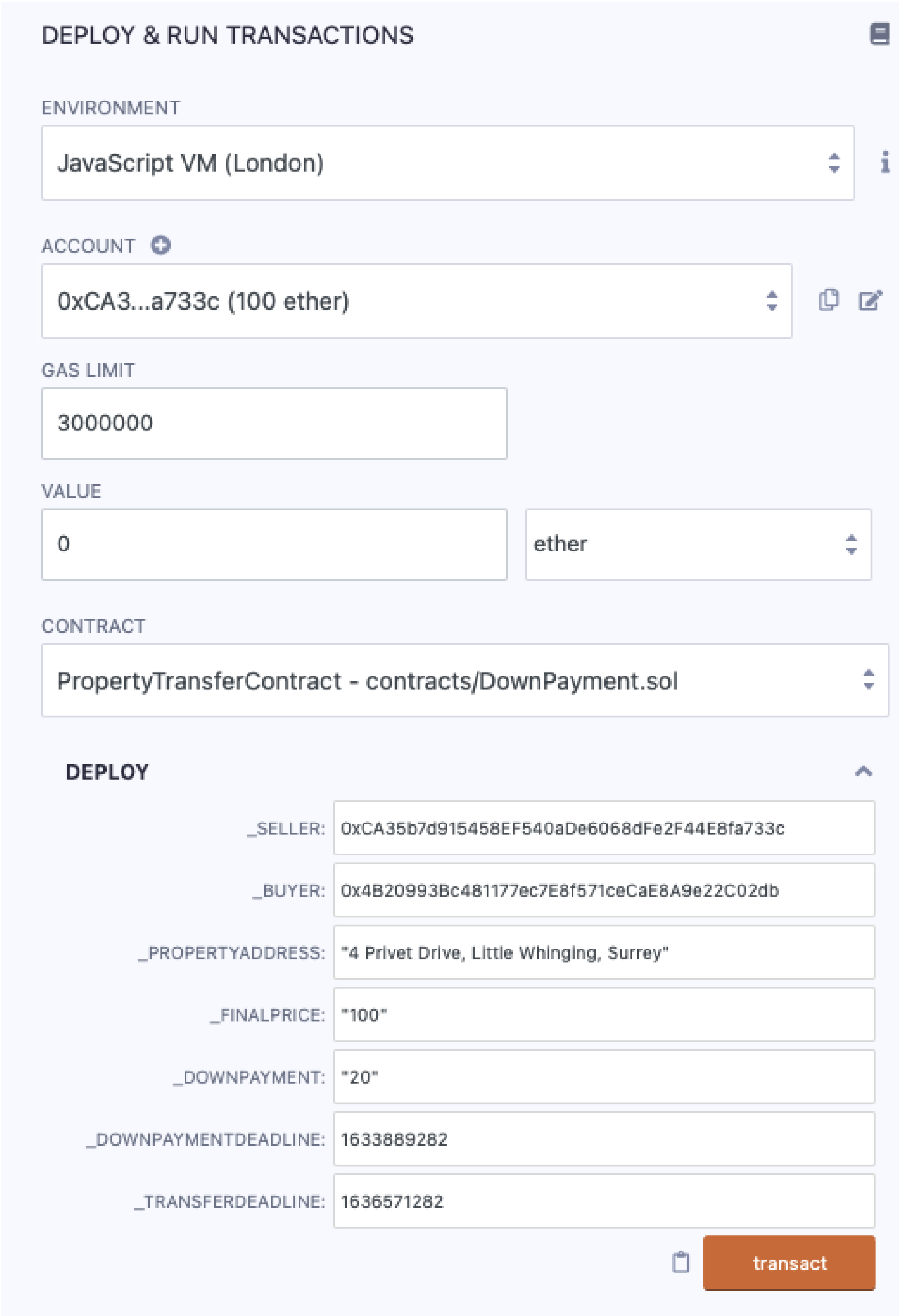

Figure 5.3 *Screenshot of smart contract deployment settings. Terms such as keys, price and deadlines are added to the constructor fields as shown.*

With the sellers account selected (***0xCA3****...)*, the contract was deployed. A succesful output log was printed in the console as shown in Figure 5.4. Clicking on the transaction log provided data regarding the cost of each transaction which is presented in the Results chapter.

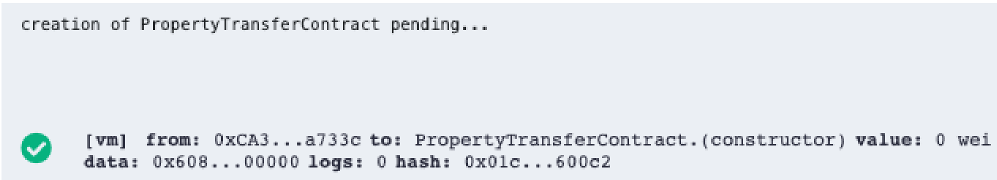

Figure 5.4 *Output log for contract deployment showing it was succesfully deployed.*

After the contract was successfully deployed, the account was changed to that of the buyers (***0x4B2…****)*. The buyer must verify the terms uploaded by the seller are correct. We assumed they were and entered 'true' in the field next to the function *buyerVerifiesContractTerms(bool)* as shown in Figure 5.5.

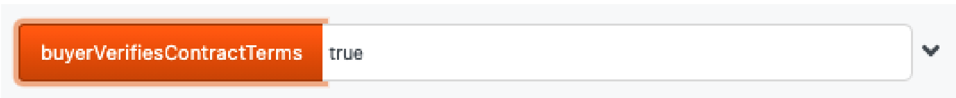

Figure 5.5 *Buyer verifies the terms of the contract are correct.*

The console showed the transaction was succesful. The buyer was now able to pay the down payment. To do this, we entered 20 ether into the value field and ran the function *buyerPaysDownPayment()*. The transaction was succesful and the buyers account showed 20 ether had been deducted as shown in Figure 5.6.

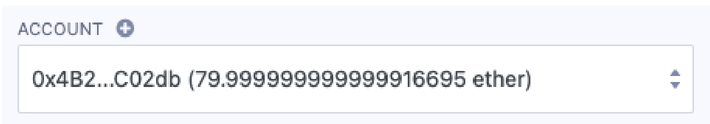

Figure 5.6 *Buyers account showing 20 ether deducted after succesfully paying down payment into smart contract*

The final stage of the transaction was for the buyer and seller to both verify all the terms of the agreement had been fulfilled. We entered ‘true’ in the field next to the function *buyerVerifiesConditionsMet(bool)* and *sellerVerifiesConditionsMet(bool)*, making sure to use the right accounts when executing the functions. The transactions were succesful and the contract was finalized. The down payment was transferred to the sellers account and ownership of the property was transferred to the buyer.

Figure 5.7 *Seller’s account showing an additional 20 ether, after contract was succesfully completed.*

Figure 5.7 shows the sellers account received an additional 20 ether. Figure 5.8 shows the result of calling the *getOwner()* function. It confirmed ownership of the property had been transferred to the buyer by displaying the address beginning **0x4B2…**

Figure 5.8 *Calling function getOwner shows ownership of the property now belongs to buyer.*

Each smart contract function call is executed as a transaction. A transaction can either be succesful or failed. The demonstration showed the prototype was succesfully deployed. Each function was run and tested succesfully. By viewing the output log for each transaction, we could see exactly how much gas each one required. This is how data was collected to answer the final research question. The data is reported in section 5.5. The next subsection describes how the prototype was evaluated.

5.4 Evaluation

Six requirements were identified in section 5.2.1. The prototype was manually tested to determine if those requirements had been fulfilled. Table 5.2 shows the results of the evaluation. Only one of the requirements was not fulfilled. This section describes how the prototype was evaluated to meet these requirements.

REQ1	REQ2	REQ3	REQ4	REQ5	REQ6
	☑	☑	☑	☑	☑

Table 5.2 *Results of manually testing prototype. Check box shows a requirement succesfully fulfilled.*

REQ1. Seller must deploy contract with the agreed terms

This requirement was the only one that was not fulfilled. However, it had no impact on the intended use of the prototype to facilitate the down payment and transfer of ownership on a property. This was because as long as the correct seller and buyer address was uploaded, the contract would work as intended. As such, anyone could deploy the contract: the buyer, seller, an agent, or web application for example. Once deployed, only the buyer and seller would be able to interact with it, given that the correct addresses were uploaded. If the wrong addresses were uploaded, then the contract would be redundant anyway and have no impact on security. Maintaining the contract functionality this way would in fact make it more adaptable for future use. Thus, this requirement can be disregarded for this prototype.

REQ2. Only buyer and seller are authorized to interact with the contract

This requirement was succesfully implemented and tested. The address of the buyer and seller were uploaded to the contract as shown in the demonstration, Figure 5.3. Once the contract was deployed on the virtual blockchain, only these two addresses could interact with it. This was tested manually, by executing function calls using different Ethereum addresses. If an address that was not the buyer or seller attempted to initiate a transaction, it would fail immediately. This requirement was achieved by using the ***require*** function of the Solidity programming language. This function ensures that the defined conditions (in this case, the invoking address being the buyer or seller) are met, otherwise the transaction fails [30]. The only functions that could be called by anyone were *getOwner()* and *getTransactionState()*. These are public, view only functions that were reserved as such so that ownership of said property could be verified. The code for these can be viewed in Appendix B.

REQ3. Buyer must be able to verify or reject contract terms

Following deployment of the contract, the buyer is required to verify that the terms such as addresses, property address, price, deadlines etc. are correct. If verified the contract continues to the next stage. If the buyer sees the terms are incorrect, he/she is able to reject the contract and thus cancel the entire transaction. This was tested manually as shown in the demonstration. When the contract terms were verified, the contract proceeded to the next state. When the terms were rejected, the contract was cancelled and could no longer be interacted with.

REQ4 Buyer must be able to safely pay down payment

This requirement was succesfully fulfilled. The buyer was able to pay the down payment into the smart contract as opposed to directly to the seller. This solved the issue of mistrust between buyers and sellers. The smart contract acted as a secure escrow system which held the payment until final conditions were met and verified by both parties. No one was able to access the funds and commit fraud, other than the buyer. This was tested manually. The seller could only access the funds after the buyer made the final verification. Thus, the buyer could safely pay the down payment knowing that he/she was the only one that could interact with it.

REQ5 Buyer must be able to retrieve payment if seller does not fulfil terms of agreement

This requirement carries on from the previous one and was tested manually. During the penultimate stage of the transaction, before the down payment was released to the seller and the contract was finalized, the buyer could make a final verification. This was to confirm that the seller had fulfilled his end of the agreement and the transaction could complete. If verified, as shown in Figure 5.6, the payment was released. When a scenario was run where the buyer did not verify the terms had been met, the *cancelTransfer()* function was called. This resulted in the payment being returned to the buyers account and the transaction being cancelled. The same occurred when predefined terms were not met such as the seller missing deadlines. The contract automatically returned the down payment back to the buyer.

REQ6 Both buyer and seller must verify all terms met before payment and ownership transferred

The final requirement was succesfully implemented. Before payment could be released to the sellers account and ownership could be transferred to the buyer, both parties were required to verify all terms of the agreement had been met. This was tested manually. When both parties verified conditions were met, the transaction completed succesfully as shown in the demonstration. However, when either party did not verify, the entire transaction was cancelled. Payment was returned to buyer and ownership was retained by the seller.

5.5 Experiment Results

The results collected from the experiment on the prototype are presented here. The third research question is answered.

What is the cost of using smart contracts to facilitate the down payment on a property?

To answer this research question, a smart contract was developed on the Ethereum blockchain. The contracts' main function was to facilitate the down payment on a property and record the transfer of ownership on the blockchain. The contract was evaluated against six identified requirements. Only one was not fulfilled, however, it was later found that it was unnecessary and could be disregarded. The source code of the contract is available in Appendix B.

The contract was run multiple times using the parameters described in section 5.3 and returned the same transaction cost each time. The gas costs of deploying the contract and executing each key function are reported in Table 5.3 below as well as the transaction time. All functions were succesfully executed.

Function	Transaction cost (gas)	Cost (USD)	Time (s)
Deploy smart contract	1,037,815	$163.00	<1
buyerVerifiesContractTerms()	29,060	$4.59	<1
buyerPaysDownPayment()	54,245	$8.57	<1
buyerVerifiesCondtionsMet()	30,928	$4.89	<1
sellerVerifiesCondtionsMet()	45,592	$7.20	<1
Total	**1,197,640**	**$188.25**	

Table 5.3 *Transaction cost and time of deploying and executing functions of implemented smart contract to succesfully pay down payment on a property.*

Every transaction on the Ethereum blockchain incurs a base cost of at least 21,000 gas [17]. This is in addition to the execution cost of a function. Thus, the total transaction cost of a function is the sum of these two values. From

Table 5.3 we can see the total cost of deploying and using the implemented smart contract to pay the down payment on a property was **1,197,640 gas**. This was equal to **$USD 188.25** at the time of writing.

6 Analysis

The following section attempts to draw conclusions to answer the research questions based on the results from the literature study and data collected from the experiment.

6.1 Literature

In Chapter 4.2 of this investigation, a summary of the reviewed literature was presented. The review highlighted several issues with traditional real estate systems. Common issues across all studies stemmed from the need for multiple third parties to facilitate transactions. As stated in the problem statement and supported by the literature, these third parties increased the cost of property transactions as well as the time to complete the purchase [4], [5], [23], [28], [29].

Additionally, 10 of the 13 studies also supported the problem statements issue regarding the security and integrity of property ownership records. Studies [4], [19], [20], [21], [22], [24], [25] [26], [27], [28] stated that the current methods of recording and storing ownership records were not secure. This was due the centralized nature of these systems which could be damaged, corrupted, altered or lost. This would lead to disputes regarding property ownership and fraud.

6.2 RQ1: Benefits of smart contracts for property transactions

From the results, it is evident that blockchain smart contracts can benefit real estate property transaction in several ways. Table 4.5 lists 15 benefits reported by the reviewed literature.

D. Bhanusha li, et al [4] highlighted that a mistrust exists between buyers and seller of a property. Hence, the need for trusted third parties such as lawyers and notaries. The decentralized nature of blockchain smart contracts provide a trustless solution for direct transactions between two parties [5]. Smart contracts, therefore, remove the need for multiple intermediaries. This reduced reliance on third parties creates a more cost effective, efficient, and faster transaction process as supported by the data in Table 4.5.

Furthermore, blockchains are immutable and difficult to tamper with [21], [22], [23], [26], [29]. Once data is recorded, it is highly unlikely to be lost. Thus, property transactions conducted via smart contracts can securely record property ownership. This reduces the risk of property fraud [19], [23], [25], [26] and disputes as ownership can be verified on the blockchain which provides data integrity [4], [21], [22], [25], [26], [28]. This data is also traceable, as all transactions are permanently recorded on the blockchain ledger.

6.3 RQ2 Limitations of smart contracts for property transactions

The results showed there are also certain limitations of using smart contracts for property transactions. One such issue highlighted by [29] is the immutable nature of the blockchain and smart contracts. While this can be a benefit, the downside is that once a contract is deployed, it cannot be changed. This could prove problematic if any errors were made or an update was required.

Furthermore, smart contracts would also need to be rigorously tested for security flaws before deployment [4]. Due to their immutability, if a security flaw was found after deployment, the entire contract would likely be unusable as no patches could be uploaded. A new contract would have to be deployed in its place.

Additionally, because of the digitalized and autonomous nature of smart contracts, there would be less room for the flexibility that human intervention allows for [23]. Smart contracts are designed to enforce pre-encoded conditions automatically and autonomously [13]. Property transactions are subject to last minute changes which are manageable in traditional systems but less so with smart contracts.

Smart contracts also introduce uncertainty regarding cost [20], [29]. The volatile nature of cryptocurrencies such as ether make it harder to determine exactly how much a transaction will cost beforehand. Storing ether for prolonged periods of time in preparation for a transaction could result in impairment loss due if ether’s price were to decrease.

Another issue is achieving mass adoption of the technology [23]. A legal framework would need to be implemented and approved by the government [19], [22], [23], [29].

A list of limitations found in the literature review are shown in Table 4.6.

6.4 RQ3: Cost of using smart contracts to facilitate property down payment

For this research question an experiment was conducted on a novel smart contract prototype that was implemented. The prototype solved two of the key issues elucidated in the problem formulation and supported by the literature review. Namely, the issue of having multiple intermediaries to provide trust and the insecure methods of maintaining ownership records.

The contract was developed and evaluated against six requirements. One of the requirements was not fulfilled, however, it had no impact on the succesful use of the prototype. The prototype could still be used for its intended purpose and maintained the necessary characteristics of trust and security. As such, the requirement could be disregarded for this investigation. The contract allowed direct peer-to-peer transactions between the buyer and seller of a property, without any intermediaries. The contract could be used to securely pay the down payment for a property. The subsequent transfer of ownership could also be recorded securely on the blockchain. This was proven by calling the *getOwner()* function after the transaction was complete. It showed that ownership of the property now belonged the public key of the buyer.

The contract created trust between two untrusting parties. The buyer could pay the down payment safely, knowing that it could be retrieved if certain conditions were not met. The seller could also trust the contract to only record a transfer of ownership if and when the down payment and specific conditions had been met.

The smart contract was deployed on a virtual Ethereum blockchain and provided with some data to simulate a down payment transaction. The total cost of deploying and using the contract was $USD188.25 as shown in Table 5.3. Based on this result, it is possible to say that cost-wise, smart contracts are a feasible alternative to traditional approaches.

However, the prototype was implemented with limited functionality. It would likely need additional functionality and security measures to be implemented before being viable for commercial use. This would increase the cost as Ethereum smart contract costs are based on the number functions executed [17].

7 Discussion

The problem formulation for this thesis asserted that there were certain issues with the current real estate system. These issues included high transaction costs, prolonged transaction times and insecure management of ownership records. A hypothesis was made to suggest blockchain based smart contracts could solve some of these issues. To determine if smart contract technology could be a viable alternative to the current methods of transacting property, three research question were proposed.

To answer these research questions, a literature study was conducted. Additionally, an experiment was carried out to further prove the potential use case for smart contracts in real estate and to collect data for the final research question. The investigation was succesful in answering all three research questions.

RQ1 - How can blockchain smart contracts benefit property transactions?
The findings show there are several benefits smart contracts can offer real estate transactions. Primarily, they remove the need for multiple intermediaries which is the cause of many issues in the industry presently [4], [20], [22], [23], [25], [29]. This is due to the smart contracts decentralized nature, which is inherited from the blockchain, allowing it to execute a pre-defined set of conditions autonomously [13]. A smart contract can provide trust between untrusting parties such as the buyer and seller of a property. No middlemen needed.

The knock-on effect of this, is that several issues are solved or at least reduced in severity. Individuals can buy and sell property in a secure manner, without having to trust anyone. Buyers can be certain that their funds are safe and can be retrieved promptly if any conditions are not fulfilled. Furthermore, the cost of the transaction can be much lower due to not having to pay third-party fees. The transaction process is streamlined as there is less reliance on bureaucracy and human processes such as filing paperwork. Thus, transactions are more efficient and less time consuming [5], [20], [22], [23], [25], [27], [28]. The contract can act autonomously and execute as soon as pre-defined conditions are fulfilled. Although this project focused on down payment transaction, the results can be generalized for other transaction types as well.

The investigation also confirmed that smart contracts provide a more secure method of recording property ownership [4], [19], [20], [22], [24], [26], [27], [28]. The data is recorded on the blockchain which is immutable and tamper-proof [21], [22], [23], [26], [29]. Therefore, the data is unlikely to be lost or corrupted, as is the case with current methods. Due to the blockchains decentralized characteristics, there are multiple copies of the transaction history. Even if one node is damaged or corrupted, the rest of the network will still maintain an accurate history. This reduces the risk of property fraud and ownership disputes as the data can be verified on the blockchain [19], [23], [25], [26].

RQ2 - What are the limitations of using smart contracts to facilitate property transactions

The results also showed that there are certain limitations of using smart contract technology for real estate transactions. One issue is due to immutability. Once a smart contract is deployed, it cannot be changed. If a security flaw is present, it could have major consequences as large sums of money are at risk. Therefore, every contract must be rigorously tested before deployment. Fortunately, there are a few firms that perform security audits on smart contracts. *Solidity.finance* [32] and *Certik.io* [33] are two such companies.

Other limitations mainly stem from the fact that blockchain and smart contracts, are a relatively young and developing field. Similarly, research into the application of these technologies for real estate is younger yet. This is supported by Figure 4.2, which shows there were only 13 papers relevant to this investigation, the earliest of which was from 2018. However, the trend line showed the number of papers generally increasing year on year, suggesting that the subject is gaining traction. As more people become aware of the technology and research and development continues, it should soon be ready for mass adoption.

RQ3 - What is the cost of using smart contracts to facilitate the down payment on a property?

At present it does not seem likely that smart contracts could be implemented on a broad scale. Though the developed prototype was shown to be cost effective, it would not meet legal or institutional requirements. The contract would need to be developed further with more comprehensive security measures followed by rigorous testing. This is needed as the contracts would handle large sums of money and must be fail-safe to gain approval from legal authorities and institutions.

Ideally, a set of standardized protocols would be developed, similar to those in the banking industry. Smart contracts would adhere to these standards in order to be compliant with security and legal regulations. Doing so would give smart contracts the possibility to be adopted on a wide scale. Moreover, the low-level functionality of these contracts should be abstracted from the end users. For the general population, it is not feasible to expect them to learn how the technology works. Simple web interfaces and applications would be needed.

As an example, D. Bhanusha li, et al [4] created a UI which could connect to the smart contract. Users could then interact with the contract via the interface, without requiring knowledge of the technology on the back end. This would provide a more seamless transition from the current property transaction processes and allow adoption to be as streamlined as possible.

8 Conclusion and Future Work

For this thesis on smart contracts for real estate transactions, three research question were proposed to determine if they could be a feasible solution. To answer these questions, a smart contract prototype was developed in addition to a literature study. It was found that smart contracts could provide several benefits. The main benefit being the removal of third parties to facilitate transactions which in turn produced auxiliary benefits such as reduced transaction costs and increased efficiency. The prototype that was implemented to facilitate down payment transactions had a total cost of $USD188.25, which is very cost effective.

The findings also showed there were certain limitations of using smart contracts for property transactions. One of these limitations is the price volatility of cryptocurrencies, such as ether, which was used for this projects prototype. However, a possible solution for this, could be the use of stable coins. Stable coins are cryptocurrencies that are pegged to the value of an asset such as USD or GBP [34]. There are several stable coins available today, which could be used instead of ether. This would mitigate the risk of impairment loss which happens when the market price of your cryptocurrency asset decreases.

The findings also showed that the immutable nature of blockchain smart contracts could be an issue for property transactions, where last minute changes are likely. Other limitations mainly stem from the fact that smart contracts in real estate are a young field and as such, more research and development is needed. Based on the findings from this investigation, smart contracts are not yet ready to be implemented as a solution to real estate property transactions on a large scale.

Regarding future work, the prototype had been developed to prove a potential use case for smart contracts in the real estate industry which had not been mentioned in any of the reviewed literature. As such, only the basic functionality was implemented. The contract allowed users to pay the down payment on a property and record the transfer of ownership. For it be commercially viable, it would need to have additional security features implemented and comply with legal regulations. Another area the contract lacks in, is the recording of property ownership. At present, it simply updates the owner field with the public key of the buyer. In future, it would be better to add a link in the contract to a written and signed agreement such as the title deed. This could be done using IPFS which is a decentralized file storage system [26].

Implementing these additional features would raise the cost of using the smart contract but it is likely to still be a cheaper alternative. The contract was also tested on a virtual blockchain within the browser. To get more statistically accurate data regarding cost and transaction times, the contract would need to be tested on a real blockchain, using real cryptocurrency.

The contract used public/private key pairs which ensured only the relevant parties could interact with it. However, in the case of lost or forgotten private keys, some sort of back up or loss prevention would be required. At present, this responsibility lies with the individual, as is the nature of decentralization. However, for government approval, an alternative method would likely be needed. This knowledge gap could form the basis of future research into the field.

Furthermore, the contract was developed solely for down payment transactions. Though, the findings could be generalized for other transaction types as well such as mortgage or rental payments. For these to be viable, the contract would need to integrate banks and/or mortgage lending companies. This would increase the number of involved parties; however, the benefits of a decentralized smart contract would remain. This would require financial institutions to come onboard and create an infrastructure for blockchain and smart contract transactions. Additionally, for the technology to have mass appeal, a simple web or mobile interface could be developed to abstract the functionality of the technology.

References

[1] Market Data Forecast. (2021). Global Blockchain Market | Size, Trends, Forecast | 2020 – 2025. [online]. Available: https://www.marketdataforecast.com/market-reports/blockchain-market [Accessed 9 March 2021].

[2] B. K. Mohanta, S. S. Panda and D. Jena, "An Overview of Smart Contract and Use Cases in Blockchain Technology," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2018, pp. 1-4.

[3] M. Nandi, R. K. Bhattacharjee, A. Jha and F. A. Barbhuiya, "A secured land registration framework on Blockchain," 2020 Third ISEA Conference on Security and Privacy (ISEA-ISAP), 2020, pp. 130-138.

[4] D. Bhanushali, A. Koul, S. Sharma and B. Shaikh, "BlockChain to Prevent Fraudulent Activities: Buying and Selling Property Using BlockChain," 2020 International Conference on Inventive Computation Technologies (ICICT), 2020, pp. 705-709.

[5] Ioannis Karamitsos, Maria Papadaki, and Nedaa Barghuthi. “Design of the Blockchain Smart Contract: A Use Case for Real Estate”. *Journal of Information Security*, 2018, pp. 177-190.

[6] Fran Casino, Thomas K. Dasaklis, Constantinos Patsakis, “A systematic literature review of blockchain-based applications: Current status, classification and open issues”, *Telematics and Informatics*, Volume 36, 2019, Pages 55-81.

[7] S. Parjuangan and Suhardi, "Systematic Literature Review of Blockchain based Smart Contracts Platforms," *2020 International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung, Indonesia, 2020, pp. 381-386.

[8] Kitchenham, Barbara. (2004). “Procedures for Performing Systematic Reviews,” Keele, UK, Keele University. 33.

[9] Kofod-Petersen, Anders. (2015). “How to do a Structured Literature Review in computer science”.

[10] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A Design Science Research Methodology for Information Systems Research,” *Journal of Management Information Systems*, vol. 24, no. 3. pp. 45–77, 2007.

[11] Z. Zheng, S. Xie, H. Dai, X. Chen and H. Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," 2017 IEEE International Congress on Big Data (BigData Congress), 2017, pp. 557-564.

[12] Changqiang Zhang, Cangshuai Wu, and Xinyi Wang. 2020. Overview of Blockchain Consensus Mechanism. In Proceedings of the 2020 2nd International Conference on Big Data Engineering (2020). Association for Computing Machinery, New York, NY, USA, 7–12.

[13] M. Alharby, A. Aldweesh and A. v. Moorsel, "Blockchain-based Smart Contracts: A Systematic Mapping Study of Academic Research (2018)," 2018 International Conference on Cloud Computing, Big Data and Blockchain (ICCBB), 2018, pp. 1-6.

[14] S. Parjuangan and Suhardi, "Systematic Literature Review of Blockchain based Smart Contracts Platforms," *2020 International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung, Indonesia, 2020, pp. 381-386.

[15] ethereum.org. 2021. *What is Ethereum? | ethereum.org*. [online] Available at: <https://ethereum.org/en/what-is-ethereum/>

[16] Samiur Khan, Md. Al-Amin, Hrittik Hossain, Nawshad Noor, and Md. Wahid Sadik. 2020. A Pragmatical Study on Blockchain Empowered Decentralized Application Development Platform. In Proceedings of the International Conference on Computing Advancements (ICCA 2020). Association for Computing Machinery, New York, NY, USA, Article 82, 1–9.

[17] ethereum.org. 2021. *Gas and fees | ethereum.org*. [online] Available at: <https://ethereum.org/en/developers/docs/gas/>

[18] Vimal Dwivedi, Vishwajeet Pattanaik, Vipin Deval, Abhishek Dixit, Alex Norta, and Dirk Draheim. 2021. “Legally Enforceable Smart-Contract Languages: A Systematic Literature Review,” ACM Comput. Surv. 54, 5, Article 110 (June 2021), 34 pages.

[19] A. F. Mendi, K. K. Sakaklı and A. Çabuk, "A Blockchain Based Land Registration System Proposal for Turkey," 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), 2020, pp. 1-6.

[20] M. Nandi, R. K. Bhattacharjee, A. Jha and F. A. Barbhuiya, "A secured land registration framework on Blockchain," 2020 Third ISEA Conference on Security and Privacy (ISEA-ISAP), 2020, pp. 130-138.

[21] T. Ali, A. Nadeem, A. Alzahrani and S. Jan, "A Transparent and Trusted Property Registration System on Permissioned Blockchain," 2019 International Conference on Advances in the Emerging Computing Technologies (AECT), 2020, pp. 1-6.

[22] M. Aquib, L. D. Dhomeja, K. Dahri and Y. A. Malkani, "Blockchain-based Land Record Management in Pakistan," 2020 3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 2020, pp. 1-5.

[23] J. L. Tilbury, E. de la Rey and K. van der Schyff, "Business Process Models of Blockchain and South African Real Estate Transactions," 2019 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD), 2019, pp. 1-7.

[24] A. Endurthi and A. Khare, "Cheat Proof Escrow System for Blockchain," 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), 2021, pp. 294-298.

[25] I. Mishra, Supriya, A. Sahoo and M. Vivek Anand, "Digitalization of Land Records using Blockchain Technology," 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2021, pp. 769-772.

[26] D. Shinde, S. Padekar, S. Raut, A. Wasay and S. S. Sambhare, "Land Registry Using Blockchain - A Survey of existing systems and proposing a feasible solution," 2019 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA), 2019, pp. 1-6.

[27] N. Gupta, M. L. Das and S. Nandi, "LandLedger: Blockchain-powered Land Property Administration System," 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), 2019, pp. 1-6.

[28] A. Mittal, B. Sharma and P. Ranjan, "Real Estate Management System based on Blockchain," 2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), 2020, pp. 1-6.

[29] Konashevych, Oleksii. (2020). Constraints and benefits of the blockchain use for real estate and property rights. Journal of Property, Planning and Environmental Law. ahead-of-print.

[30] Solidity. n.d. Solidity — Solidity 0.7.4 documentation. [online] Available: https://docs.soliditylang.org/en/v0.7.4/ [Accessed 24 March 2021].

[31] Remix. (2021). Remix - Ethereum IDE & community. [online] Available: https://remix-project.org/ [Accessed 24 March 2021].

[32] Solidity.finance. 2021. *Smart Contract Auditing Services - Solidity.finance*. [online] Available at: <https://solidity.finance/>

[33] Certik.io. 2021. [online] Available at: <https://www.certik.io/security-audits/>

[34] Investopedia. 2021. *Stablecoin*. [online] Available at: <https://www.investopedia.com/terms/s/stablecoin.asp/>

Appendix A

Table A.1 *List of selected publications and their extracted data*

P No.	Authors	Title	Country*	P. Type*	Year	Ref
1	A. F. Mendi, et al	A Blockchain Based Land Registration System Proposal for Turkey	Turkey	CP (IEEE)	2020	[19]
2	M. Nandi, et al	A secured land registration framework on Blockchain	India	CP (IEEE)	2020	[20]
3	T. Ali, et al	A Transparent and Trusted Property Registration System on Permissioned Blockchain	Saudi Arabia	CP (IEEE)	2020	[21]
4	D. Bhanushali, et al	BlockChain to Prevent Fraudulent Activities: Buying and Selling Property Using BlockChain	India	CP (IEEE)	2020	[4]
5	M. Aquib, Et al	Blockchain-based Land Record Management in Pakistan	Pakistan	CP (IEEE)	2020	[22]
6	J. L. Tilbury, et al	Business Process Models of Blockchain and South African Real Estate Transactions	South Africa	CP (IEEE)	2019	[23]
7	A. Endurthi, et al	Cheat Proof Escrow System for Blockchain	India	CP (IEEE)	2021	[24]
8	I. Mishra, et al	Digitalization of Land Records using Blockchain Technology	India	CP (IEEE)	2021	[25]

Table A.1 *Table A.1 continued*

9	D. Shinde, et al	Land Registry Using Blockchain - A Survey of existing systems and proposing a feasible solution	India	CP (IEEE)	2019	[26]
10	N. Gupta, et al	LandLedger: Blockchain-powered Land Property Administration System	India	CP (IEEE)	2019	[27]
11	A. Mittal, et al	Real Estate Management System based on Blockchain	India	CP (IEEE)	2020	[28]
12	I. Karamitsos, et al	Design of the Blockchain Smart Contract: A Use Case for Real Estate	UAE	JA	2018	[5]
13	O. Konashevych	Constraints and benefits of the blockchain use for real estate and property rights	Italy	JA	2020	[29]

*P No: Publication number.
*Country: refers to the Country of origin of the study.
*P. Type: Publication type. CP: Conference Proceeding, JA: Journal Article.
*Ref: Reference

Table A.2 *List of benefits and limitations of smart contracts for real estate transactions, extracted from reviewed literature.*

P No.	Benefits	Limitations
1	▪ Reduce property tax fraud ▪ Increased security ▪ Increased transparency ▪ Increased traceability of transactions ▪ Reduce physical contact	▪ Multiple third parties ▪ Legislation must be updated before system can be used ▪ System only works for one country (Turkey)
2	▪ Increased transparency ▪ Reduced number of third parties ▪ Increased security ▪ Reduced transaction time ▪ Reduced transaction cost	▪ Uncertain transaction cost
3	▪ Increased transparency ▪ Trust ▪ Decentralized ▪ Tamper-proof ▪ Integrity of records	None mentioned
4	▪ Security ▪ Increased transparency ▪ Integrity of records ▪ Reduced number of third parties	▪ Security of contract must be tested further ▪ Solution is incomplete and lacks functionality
5	▪ Security ▪ Integrity of records ▪ Tamper-proof ▪ Decentralized ▪ Reduced number of third parties ▪ Reduced physical contact ▪ Increased efficiency	▪ Legal requirements not accounted for ▪ Permissions needed to test system in real-life scenario
6	▪ Reduced number of third parties ▪ Reduce risk of fraud ▪ Cost effective ▪ Increased transparency ▪ Increased speed ▪ Tamper-proof data	▪ Adoption ▪ Initial information capture ▪ Less human involvement leading to reduced flexibility ▪ Legal framework needed ▪ Blockchain has unique vulnerabilities
7	▪ Secure ▪ Privacy	None mentioned

Table A.2 *Table A.2 continued.*

8	▪ Transparency ▪ Reduced fraud risk ▪ Increasing speed of process ▪ Integrity of records ▪ Reduce number of third parties	None mentioned
9	▪ Decentralized ▪ Tamper-proof ▪ Secure ▪ Integrity of data ▪ Reduced fraud risk	None mentioned
10	▪ Transparent ▪ Efficient ▪ Secure ▪ Scalable ▪ Traceability ▪ Easy integration with conventional systems	None mentioned
11	▪ Transparent ▪ Secure ▪ Efficient ▪ Data integrity	▪ Multiple third parties
12	▪ Multiple parties can interact with the system ▪ Provides a trustless solution between untrusting parties ▪ Smart contract acts as a trusted intermediary removing need for notaries/brokers etc. ▪ Increased transaction efficiency	▪ Only covered one use case ▪ Blockchain technology may not cover their entire organizations processes
13	▪ Immutability ▪ Multiple use cases ▪ Reduce number of third parties	▪ Immutability ▪ Anonymity ▪ Privacy ▪ Scalability ▪ Price volatility ▪ Legal compliance ▪ Government approval needed

Appendix B

```
pragma solidity 0.6.12;

contract PropertyTransferContract {

    //stakeholders
    address payable public buyer;
    address payable public seller;

    enum TransactionState {
        Initiated, DataVerified,
        ConditionsConfirmedBuyer,
        Complete, Rejected }

    TransactionState public state;

    struct PropertyDetails {
        string propertyAddress;
        //same as seller, transferred to buyer once transaction terms fulfilled.
        address payable propertyOwner;
        uint finalPrice;
        uint downPayment;
        uint downPaymentDeadline;
        uint transferDeadline;
    }

    PropertyDetails property;

    constructor(
        address payable _seller,
        address payable _buyer,
        string memory _propertyAddress,
        uint _finalPrice,
        uint _downPayment,
        uint _downPaymentDeadline,
        uint _transferDeadline)
          public {
            require(_downPayment <= _finalPrice,
            "Down payment must be less than the final price of the property.");

            seller = _seller;
            buyer = _buyer;

            setPropertyDetails(_propertyAddress, _seller, _finalPrice,
            _downPayment, _downPaymentDeadline, _transferDeadline);

            state = TransactionState.Initiated;
    }
```

Figure B.1 *Source code of implemented artefact part 1.*

```
function setPropertyDetails(
    string memory _propertyAddress,
    address payable _seller,
    uint _finalPrice,
    uint _downPayment,
    uint _downPaymentDeadline,
    uint _transferDeadline)
    private {
        property = PropertyDetails(_propertyAddress, _seller, _finalPrice,
        _downPayment, _downPaymentDeadline, _transferDeadline);
}

function buyerVerifiesContractTerms(bool confirmed) public {
    require(msg.sender == buyer, "Buyer must verify terms.");

    require(state == TransactionState.Initiated, "Contract must be initiated.");

    if(confirmed) {
        state = TransactionState.DataVerified;
    } else {
        state = TransactionState.Rejected;
    }
}

uint downPaymentRecieved;

function buyerPaysDownPayment() public payable {
    require(msg.sender == buyer, "Down payment must be sent from buyer.");

    require(state == TransactionState.DataVerified, "Contract conditions must be verified.");

    //The value sent should be between the down payment and final price.
    require(msg.value/(1 ether) >= property.downPayment && msg.value/(1 ether) <= property.finalPrice,
    "The value sent should greater or equal to the down payment and less than or equal to the final price");

    //The payment should be sent before the agreed deadline.
    require(block.timestamp <= property.downPaymentDeadline,
    "Deadline for down payment has passed, transaction is cancelled.");

    downPaymentRecieved = msg.value;
}

function buyerVerifiesCondtionsMet(bool confirmed) public payable {
    require(msg.sender == buyer, "Buyer must verify condtions are fulfilled.");

    require(block.timestamp <= property.transferDeadline, "Transfer deadline passed.");

    if(confirmed) {
        state = TransactionState.ConditionsConfirmedBuyer;
    } else {
        //return down payment to buyer if conditions have not been met.
        buyer.transfer(downPaymentRecieved);
        state = TransactionState.Rejected;
    }
}
```

Figure B.2 *Source code of implemented artefact part 2.*

```
    function sellerVerifiesCondtionsMet(bool confirmed) public payable {
        require(msg.sender == seller && state == TransactionState.ConditionsConfirmedBuyer,
        "Seller must verify condtions are fulfilled after buyer has confirmed.");

        require(block.timestamp <= property.transferDeadline, "Transfer deadline passed.");

        if(confirmed) {
            finalizeTransfer();
        } else {
          cancelTransfer();
        }
    }

    /**
     * Private helper method which is called when all conditions are met.
     * Down payment is transferred to seller and ownership transffered to buyer.
     * Transaction state is set to complete.
     */
    function finalizeTransfer() private {
        seller.transfer(downPaymentRecieved);
        property.propertyOwner = buyer;
        state = TransactionState.Complete;
    }

    /**
     * Private helper method which is called if conditions are not met.
     * Down payment is transferred back to buyer and transaction is cancelled.
     * Transaction state is set to rejected.
     */
    function cancelTransfer() private {
        buyer.transfer(downPaymentRecieved);
        state = TransactionState.Rejected;
    }

    /**
     * Public view only method which can be called to see who the current
     * owner of the property is.
     */
    function getOwner() public view returns(address) {
        return property.propertyOwner;
    }

     /**
     * Public view only method which can be called to see what state the transaction is in.
     */
    function getTransactionState() public view returns(TransactionState) {
        return state;
    }
```

Figure B.3 *Source code of implemented artefact part 3.*