buildings

MDPI

Article

Construction Payment Automation Through Scan-to-BIM and Blockchain-Enabled Smart Contract

Hamdy Elsharkawi [1,2], Emad Elbeltagi [3], Mohamed S. Eid [2], Wael Alattyih [3,*] and Hossam Wefki [4]

[1] Civil Engineering Department, Faculty of Engineering, Horus University-Egypt, New Damietta 34517, Egypt; helsharkawi@horus.edu.eg or hamdy.elsharkawi@student.aast.edu
[2] Construction and Building Engineering, Arab Academy for Science, Technology, and Maritime Transport, Sheraton Heliopolis, Cairo 12577, Egypt; meid@aast.edu
[3] Department of Civil Engineering, College of Engineering, Qassim University, Buraydah 51452, Saudi Arabia; e.elbeltagi@qu.edu.sa
[4] Civil Engineering Department, Port Said University, Port Fouad 42523, Egypt; hossam.wefki@eng.psu.edu.eg
* Correspondence: w.alattyih@qu.edu.sa

Abstract: Timely approvals and payments to the project participants are crucial for successful completion of construction projects. However, the construction industry faces persistent delays and non-payments to contractors. Despite the desirable benefits of automated payments and enhanced access to digitized data progress, most payment applications rely on centralized control mechanisms; inefficient procedures; and documentation that takes time to prepare, review, and approve. As such, there is a need for a reliable payment automation system that guarantees timely execution of payments upon the detection of completed works. Therefore, this study used a cutting-edge approach to automate construction payments by integrating blockchain-enabled smart contracts and scan-to-Building Information Modeling (BIM). In this approach, scan-to-BIM provides accurate, real-time building progress data, which serve as the source of verifiable off-chain data. A chain-link is then used to securely relay these data to the blockchain system. Blockchain-enabled smart contracts automate the execution of payments upon meeting contract conditions. The proposed approach was implemented on a real case study project. The actual site scan was captured using a photogrammetry 360° camera, which uses a combination of structured light and infrared depth sensing technology to capture 3D data and create detailed 3D models of spaces. This study leveraged accurate, real-time building progress data to automate payments using blockchain-enabled smart contracts upon work completion, thus reducing payment disputes by tying payments to verifiable construction progress, leading to faster release of payments. The findings show that this approach provides a transparent basis for payment, enhancing trust and allowing precise project progress tracking.

Keywords: blockchain technology; smart contracts; Ethereum; scan-to-BIM; reality capture; digital technologies; construction industry

Academic Editor: Jurgita Antucheviciene

Received: 8 December 2024
Revised: 1 January 2025
Accepted: 8 January 2025
Published: 13 January 2025

Citation: Elsharkawi, H.; Elbeltagi, E.; Eid, M.S.; Alattyih, W.; Wefki, H. Construction Payment Automation Through Scan-to-BIM and Blockchain-Enabled Smart Contract. *Buildings* **2025**, *15*, 213. https://doi.org/10.3390/buildings15020213

1. Introduction

The architecture, engineering, and construction (AEC) industry faces challenges with delayed payments [1], information asymmetry, and low productivity as a result of inefficient contracting procedures. Processing payments and fulfilling contractual commitments may be difficult for the construction sector, which is known for its intricate projects and diverse stakeholder connections. Debates, mistakes, and delays can result from traditional techniques that rely on manual verification of finished work. In construction projects, the project owner pays the contractor regularly for the completed work, enabling the contractor to procure the necessary resources to complete the project. These progress payments are distributed after the owner verifies and approves the progress value reported by the contractor. Owner–contractor disputes regarding the value of progress may occur, especially if the contract's payment conditions are unclear, with inadequately detailed terms in the contract that leaves room for interpretation and potential disputes [2], or is poorly handled, referring to the ineffective administration and enforcement of the payment terms, leading to delays, errors, and dissatisfaction.

To this end, there is a need for an automated payment system that (1) captures the actual progress and quantities on site; (2) compares that to the owner-contractor agreements and settles payments; and (3) validates the transactions and progress via a decentralized and robust system. This can be achieved through combining the benefits of scan-to-Building Information Modeling (BIM), smart contracts (SCs), and blockchain technology (BCT). Scan-to-BIM technology offers a precise, 1:1 digital representation of the scanned environment, ensuring an accurate model of the existing conditions. Scan-to-BIM generates accurate 3D models of spaces by capturing millions of data points and converting them into 3D "point clouds", representing the physical layout of the space. Then, these scans are processed and converted into highly detailed BIM outputs. These can be used to check on the actual progress and compare it to the owner–contractor agreements via smart contracts.

SCs are automated agreements guaranteed to run by code on a decentralized blockchain, ensuring every transaction is recorded, transparent, and irreversible. Information in SCs is defined through coding, based on the principle that the code serves as the governing law. For instance, SCs can define a contractor as someone who gets paid a set payment upon completing a particular service. As a result, smart contracts utilize if–then rules to automatically process payments based on verified and approved contracted construction items. This can further be cross-validated through blockchain (BC). BCT is widely used in industries such as construction because of its significant potential [3]. BC has emerged as a distinctive technology in recent years, allowing a significant change in conventional business process management, which often relies on centralized systems that require trusted third parties [4]. BC can address these challenges by establishing a decentralized and transparent system where all the participants can access a shared database. This enables the tracking and monitoring of various project phases and can even automate specific processes, thereby enhancing efficiency and minimizing delays and rework. Through a decentralized consensus system, BC provides a secure, consistent global database without a central authority, supporting the execution of financial and database transactions. BC increases trust, security, transparency, decentralization, immutability, and traceability of the data shared across a business network and delivers cost savings with new efficiencies, which has sparked interest across various industries [5]. The advancement of BC has facilitated the creation of a platform that supports the implementation of smart contracts.

Therefore, decentralized BC-based SCs emerge as robust solutions for automating progress payments, addressing critical deficiencies in current payment mechanisms [6]. While many previous articles have mainly focused on framework-based solutions, there is a need for a more comprehensive approach that implements SC and BC to facilitate the payment transactions. This can be achieved by combining BC-based SCs with scan-to-BIM technology.

Research Goal and Objectives

The goal of this research is to facilitate the progress report submission and automate the payment based on the actual progress on site. This can be achieved through leveraging the benefits of scan-to-BIM and SC within a BC environment. To this end, the authors have the following three objectives: (1) utilize scan-to-BIM to generate accurate 3D models of construction progress on site, ensuring precise, real-time digital representation of the scanned site; (2) develop contract payment terms on SCs and apply them on a decentralized BC; and (3) implement the developed framework on a real case study to ensure accurate and efficient automation payment, thereby minimizing disputes and errors. Ultimately, the outcome of the aforementioned steps will result in semantically rich, as-built BIM for every utilization of a 3D scan camera for the construction site and estimating the actual quantities performed that have the potential to streamline payments and automate the transition from progress flow to payment flow.

The subsequent sections of this paper are structured as follows: A literature review of reality capture technologies, BC, and SC applications in the construction industry is provided in Section 2. Section 3 shows the research gap. Section 4 shows the proposed integration of scan-to-BIM and blockchain-enabled SC systems. The application of this integrated system to a real-world construction project is demonstrated in Section 5. Section 6 explores the benefits and limitations of the proposed system. Finally, concluding remarks are presented in Section 7.

2. Literature Review

2.1. Reality Capture Technologies

BIM technology is rapidly becoming the norm in many nations and has great potential to increase the automation of the AEC industry [7]. It also provides a digital option for mitigating contract payment ambiguity and poor progress payment administration. Despite the emergence of various BIM contract protocols and increasing interest in BIM, its utilization for contract administration remains limited [8]. Limited studies have investigated how BIM can improve the conventional process of administering progress payments.

As-built capture and modelling involve taking measurements and dimensions from a construction site and using them to create a model representing the site's finished state. This process used to be entirely manual, time-consuming, and prone to errors. However, thanks to advancements in design tools and technology, it has become more efficient through reality capture technologies and BIM, often referred to as the scan-to-BIM workflow. The workflow is divided into three primary stages: data collecting, processing, and BIM modelling [9]. In the last several years, remote sensing technologies have become essential tools for modelling as-built structures within the scan-to-BIM workflow. These technologies include laser scanning and photogrammetry, which are used for taking dense 3D measurements of a facility's state, and the point cloud that is produced, which can be processed to produce an as-built BIM. Scan-to-BIM recognizes numerous objects and obtains relevant geometric data from point clouds [10,11].

Golparvar-Fard et al. [12] aimed to automate the monitoring of building site activities by utilizing a combination of time-lapse and photogrammetry techniques to overlay the as-built point cloud model created from site photos over the as-planned 3D model using augmented reality (AR). This system was able to use color codes to show the progress status. Still, it was decided that color-blind employees would not benefit from reproducing the building process using a color-coding scheme. A different technique was presented by Roh et al. [13] in an attempt to address some of the shortcomings of earlier systems. It compares the as-planned BIM model with the as-built components, which were abstracted from site photos. In order to synchronize the items from BIM models with matching things from taken photographs, a classifier system was developed. The building elements' progress was successfully achieved with this approach in percentages. Critical data, including position and time, had to be manually entered due to extensive manual involvement, which made the data erroneous and unreliable. The algorithm's development requires manual synchronization of the construction elements; hence, the classifier accuracy heavily depends on human accuracy.

Behnam et al. [14] developed an automated system to create and display the status of repeated building tasks for linear infrastructure projects. By combining satellite remote sensing techniques, they created 3D models using satellite photos and areas identified by the Geographic Information System (GIS) web-based platform. The site images that were gathered enhanced the 3D model's accuracy. The progress status is further compared between the 4D BIM model and the 3D generated model. The suggested automated method partially overcame some of the drawbacks of earlier automation attempts to track the development of construction site operations. This approach, however, is restricted to linear projects with recurring operations, like pipelines, road construction, and railroads; therefore, it cannot be used for site activities in general.

Skrzypczak et al. [15] explored the integration of laser scanning and BIM technologies in construction, assessing the accuracy of 3D building models in inventory measurements. The study used terrestrial laser scanning and close-range photogrammetry from UAVs to create detailed 3D models of three buildings in Poland. It focuses on the accuracy of these models in representing actual building dimensions. The research highlights the benefits of combining these technologies for efficient and accurate architectural and structural assessments through enhancing the precision and efficiency of construction documentation and analyses. Integrating BIM with contemporary technology, such as laser scanning and photogrammetry, can effectively streamline the procedure and produce accurate data faster [16]. This BIM serves as a knowledge base to assist in the decision-making and problem-solving processes related to civil engineering and document as-built conditions through building information.

All the limitations discussed in the literature were motivated using the scan-to-BIM method with the 3D scanning camera. This technology addresses earlier limitations by accurately capturing and measuring the actual quantities performed and providing precise progress tracking for automating payments.

2.2. Blockchain and Smart Contracts

The construction industry features a highly intricate network of contractual relationships, including owners, consultants, primary contractors, subcontractors, and suppliers, along with complex documentation types and procedures [17]. Thus, BC is an appropriate solution for overcoming these issues. BC allows parties to be represented as users, and transactions can take many forms, including documents, materials, and cash flows. The financial transaction is the most crucial of these transactions, making it an attractive research area for cryptocurrencies [18]. A BC is a decentralized public ledger that keeps track of all digital events or transactions carried out and shared by network members [19]. A BC is a secure distributed ledger technology underlying cryptocurrency such as Bitcoin [20] and Ether [21]. Blockchains (BCs) employ cryptography, and peer-to-peer networks verify and synchronize records among decentralized peers without a central administrator [22]. In a BC-based network, each member can communicate or transact with one another on a peer-to-peer basis. Information sharing between members is based on a consensus method that eliminates the need for middlemen. This guarantees that information is distributed transparently among all network participants, with no one having complete control over the data [23].

BCs have progressed through three significant generations [24]. The first version of BC offers a public ledger to hold cryptographically signed financial transactions, such as Bitcoin. The second version of BC includes a general-purpose programmable infrastructure, SCs, and a public ledger that holds the computational results created by the programmable infrastructure. A SC is a digital contract that functions automatically when specific criteria are satisfied. Because the input conditions are retrieved from the BC network as immutable data and the code is likewise protected on the BC network, such contracts can serve as digitally binding contracts. The last generation is concerned with BC-based applications in many sectors and systems, and this indicates that BC 3.0 may be applied in a broader range of sectors outside of finance and economics.

SCs have been one of the most sought-after technologies since the debut of BCT. They are entirely digital and essentially encoded containers of code [25]. In a BC ecosystem, a SC is a novel technology that can autonomously negotiate, fulfill, and enforce the terms of an agreement [26]. SCs provide the advantages of lowering risk, administration, and service costs and boosting the efficiency of corporate operations compared to traditional contracts [27]. They are computer protocols that, once built and deployed on the BC network, may be self-executed and self-verified without human intervention [28]. In this regard, SCs are expected to offer a better solution to the current transaction mode in various industries.

Turk and Klinc [29] state that BC can be integrated with BIM to add new features. Mason and Escott [30] predict that by 2020, BIM integration with smart contracts will be possible because of the anticipated increase in the number of sensors in devices. BIM level 2 promises to reduce the number of paper-based exchanges and communications [31]. Ye et al. [32] developed a framework for automating and visualizing the delivery, acceptance, and payment (DAP) processes in construction projects using BC-based SCs integrated with BIM. It highlights the challenge of understanding and visualizing SCs for stakeholders, proposing a solution that combines BIM with the Business Process Model and Notation (BPMN) to make transactions and contract statuses more comprehensible. Ahmadisheykhsarmast et al. [33] propose a SC payment security system (SMTSEC) for improving payment securities in construction projects using BCT. The system automates payments through a decentralized protocol, ensuring secure, timely, and transparent transactions without intermediaries. The system aims to address common payment delays and insolvency issues in the construction industry by providing a structured and predictable payment process.

Elghaish et al. [34] explore the implementation of BCT in integrated project delivery (IPD) within the AEC industries where all financial operations, including costs, profits, and savings, are handled automatically by SCs coded within a BC network, explicitly using a Hyperledger fabric, as it aims to overcome financial barriers and improve transactional transparency and efficiency by automating financial interactions between project participants using BC-based SCs. This approach is intended to foster trust, reduce disputes, and streamline financial processes in construction projects. Ciotta et al. [35] examined the potential for integrating BCT and SCs within the common data environments (CDEs) used in the construction industry. The approach utilizes SCs of varying complexities, integrating them with Internet of Things (IoT) sensors to improve the tracking and verification of construction activities. The research aims to reduce human errors and enhance the transparency and reliability of managing construction documentation related to structural safety.

Sonmez et al. [36] present a novel BIM-integrated SC system for managing progress payments in construction projects, utilizing BCT, leveraging the detailed data from BIM and the automation capabilities of SCs to ensure transparent and timely financial transactions between stakeholders. The BIM model was not created by scanning an existing structure but was developed for the building process, specifically for the case study. Dakhli et al. [37] explored BC's potential to reduce non-value-adding transaction costs in building construction. Their study indicates that BC can significantly lower costs by reducing the need for intermediaries; using a case study from a real estate company, they demonstrate an 8.3% potential cost saving in residential construction by implementing BCT.

Integration of BIM and blockchain technologies into the construction industry has immense challenges that make its wide adoption difficult. Technically, BIM produces large datasets that blockchain networks struggle to handle due to scalability limitations and high transaction costs. Interoperability between BIM and blockchain platforms and the need for secure off-chain data relays add further complexity [6,34]. Legally, the enforceability of blockchain-based smart contracts is unclear under current regulatory frameworks, especially in international projects where jurisdictional differences further complicate adoption [33]. Organizationally, this integration disrupts traditional workflows; reduces the roles of consultants and project managers; and necessitates intensive training, which has often deterred small firms with limited resources [29]. High implementation costs, along with security risks such as data vulnerabilities and possible cyberattacks, present additional barriers [37]. Moreover, there is still a lack of awareness and the absence of standardized protocols for integration continue to impede widespread adoption. Addressing these challenges requires collaborative efforts to establish scalable, secure, and legally robust frameworks to support the integration of BIM and blockchain in construction.

The applications of BC SCs are still mostly theoretical, frequently focusing on creating conceptual models and frameworks, so more practical research is desperately needed in the near future [38]. Although SCs may be used in developing fields like digital twin technology, assessing how well they mesh with cutting-edge information and communication technology (ICT) solutions like BIM, IoT, and AI is also essential. In the upcoming years, such combinations may open the door to a transparent, safe, and intelligent building environment. Integrating BIM with BC is the basis of the new communication paradigm in the construction industry. BC secures this data, and BIM provides a platform for creating and managing data [39].

A new system that integrates BC, SCs, and robotic reality-capturing technology [6,40] has recently been introduced to automate the progress payment process. This approach utilizes robotic reality capture to facilitate progress payments to seven subcontractors involved in two real-world construction projects. These projects included building elements related to partitioning, plumbing, heating, ventilation, and air conditioning. One of the critical drawbacks is that the acquired data are processed with machine intelligence techniques to determine progress percentages for distinct scopes of work. The granularity of progress results varies depending on the machine intelligence algorithm used, and robotic reality capture systems are prone to inaccuracy and error and may produce disagreements. Hence, within this context, this research aims to create a novel, close-range photogrammetry-based method as scan-to-BIM for overseeing and managing construction site operations by integrating BC-enabled SCs. Scan-to-BIM allows for estimating actual quantities performed and provides them to the BC-enabled SC. These efforts yield a semantically rich as-built BIM integrated with a BC SC system.

3. Research Gap and Novelty

The explicit gap to be addressed in this study is the promotion of payment progress in construction projects. Integrating BIM with BCT in construction projects remains an underexplored domain. Despite the potential benefits, several critical gaps necessitate investigation. Existing research predominantly focuses on the individual merits of BIM or BC, often neglecting the synergy that can be achieved by combining these technologies. The construction industry continues to struggle with issues related to transparency, trust, and efficiency in payment processes. Traditional payment mechanisms are often plagued by delays, disputes, and administrative overheads, which can significantly impede project progress and profitability. The traditional construction payment process often involves significant delays, typically 60 to 70 days [41]. In the manual contracting system, the processing time for progress payments typically averaged around 20 days before the funds could be released to all project participants [42]. There are probable clashes between project participants whose benefits and viewpoints differ. In the construction industry, claims are inevitable and become disputes unless controlled. Researchers [43,44] identified one of the root causes of conflicts is poor payment, resulting in (1) not disbursing the payments received to the parties downstream, (2) rising cash flow problems due to late payments, (3) increasing insolvency risks, and (4) damaging trust [45].

Furthermore, the application of SCs to automate payments in construction projects is still in its nascent stages, frequently focusing on creating conceptual models and frameworks, so more practical research is desperately needed. Also, the granularity of progress results varies depending on the machine intelligence algorithm used, and robotic reality capture systems are prone to inaccuracy and error, which may produce disagreements. This gap underscores the need for a detailed exploration of how scan-to-BIM technology, coupled with BC SCs, can revolutionize payment initiation and management in construction projects.

The novelty of this research lies in its innovative integration of BC-enabled SCs with scan-to-BIM workflows, creating a seamless system that addresses inefficiencies in construction payment processes. Unlike prior studies that focus on individual components or theoretical frameworks, this work advances practical implementation by applying the integrated system to a real-world case study. The automation of payment mechanisms through BC SCs ensures transparency and efficiency, as payments are directly tied to verified progress data and executed autonomously upon meeting pre-defined conditions. The proposed framework would also be able to reduce administrative overhead and foster trust among stakeholders by utilizing immutable BC records, setting it apart as a pioneering effort. Furthermore, this study not only validates the feasibility of the integration in a practical setting but also lays a strong foundation for future exploration of BC and scan-to-BIM applications in construction payment automation, marking a significant leap from conceptual discussions to transformative real-world impact.

4. Methodology

The methodology implemented in this study involves a multi-step process deigned to integrate scan-to-BIM technology with BC-enabled SC to improve the construction project's payments and financial management. The system aims to be easily adopted by a diverse group of stakeholders, including owners, consultants, contractors, and subcontractors. It is particularly suited for design-bid-build projects that are contracted on a unit-price basis. This research adopted a multi-step process methodology, as depicted in Figure 1.

It should be noted that the authors assumed that the contract for the construction assignment of the works is signed between owner and contractor, and no change orders or variations are accounted for in the current model. In addition, the individual construction work elements, accounting units, and payments are included in the contract.

This study adopted a mixed-methods approach, using both qualitative and quantitative approaches to comprehensively evaluate the integration of scan-to-BIM technology with BC-enabled SCs for automating construction payment processes. The quantitative aspect involves the development and application of automating payment processes, supported by blockchain smart contracts and scan-to-BIM data. The qualitative aspect primarily involves the analysis and synthesis of literature, which informed the identification of inefficiencies in traditional payment workflows, as well as the design of the proposed framework. This analysis contextualized the challenges addressed by the integration of scan-to-BIM and blockchain-enabled smart contracts; hence, the framework is grounded in the realities of construction industry practices.

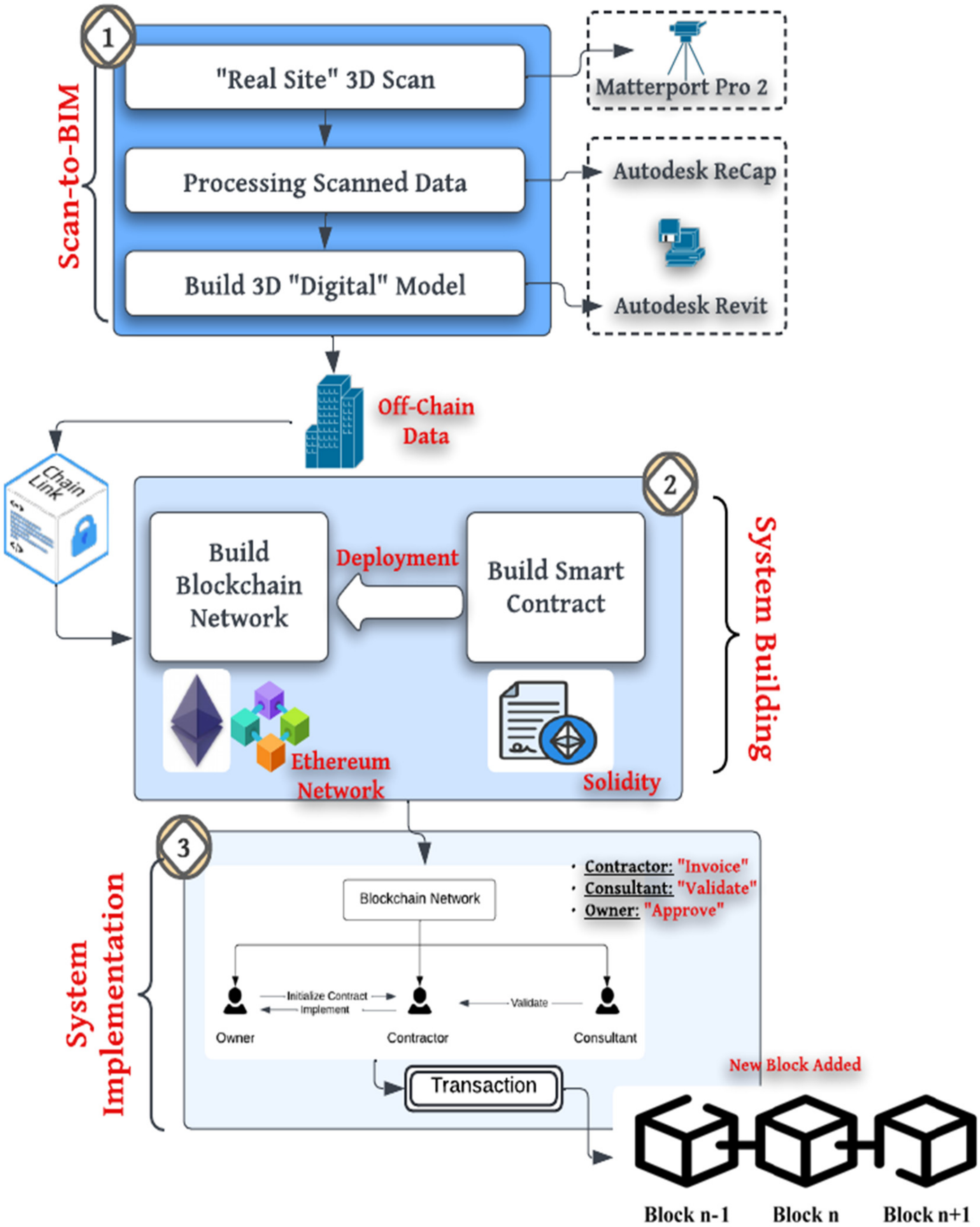

Figure 1. Proposed integration of scan-to-BIM and BC-enabled SC system.

4.1. Scan-to-BIM

Obtaining as-built data involves taking a point cloud representation of an object's shape and structure [46,47]. This study offers a scan-to-BIM workflow that uses 3D scanning, point cloud extraction technologies, and BIM intelligent tools as reality capture and BIM software (Revit 2020) to automate the modeling process further and deliver realistic visualization. This workflow can be applied across AEC and product design and manufacturing (PDM) industries [48]. Scan to BIM technology represents a powerful approach in the AEC industries for creating accurate digital representations of physical spaces. This process involves two essential technologies: laser scanning (LiDAR) and photogrammetry, which capture detailed information about a site or structure. Photogrammetry considers various factors such as project requirements, cost, and accessibility and tends to be more cost-effective, making it an attractive option for smaller projects or those with tighter budgets [49,50].

The technology utilizes digital cameras and drones, generally more affordable and accessible than the specialized sensors required for LiDAR scanning. This ease of access allows a broader range of users to undertake photogrammetry projects, making it possible to update data and scale projects as needed frequently. Beyond cost and accessibility, photogrammetry offers unique advantages in capturing visual details. It measures the geometry of a scene and captures its textures and colors, providing rich, detailed images that are invaluable in fields like the AEC industry. While LiDAR is unparalleled in its ability to perform in low-light conditions, photogrammetry's versatility and the richness of its visual information make it the preferred choice for projects where these attributes are prioritized.

This study used the photogrammetry technique in scanning and involved taking image measurements. Photogrammetry takes images as input and produces a map, a sketch, a measurement, or a 3D model of a real-world object as output. These images are then analyzed using specialized software that identifies common points in the photographs and calculates the distance between these points, effectively mapping the object or area in 3D.

4.2. Blockchain-Enabled Smart Contracts

BCT overlaps existing contracts by incorporating the details of agreements between two or more parties, but it outperforms them, owing to SCs, which automate the implementation of agreements in a distributed environment when circumstances are satisfied [51]. SCs are executable scripts that run on top of the BC to enable, implement, and enforce an agreement between two untrustworthy parties without the participation of a third party [52].

Choosing an appropriate BC for SC systems is a critical decision influenced by various application-specific factors. BC platforms are classified into two main categories: public BC platforms, including Ethereum [21] and Bitcoin [20], and permissioned BC platforms, such as Hyperledger [53]. Public BCs operate as fully decentralized systems, employing probabilistic consensus algorithms to add data blocks to a publicly accessible ledger. In contrast, permissioned BCs are more centralized, utilizing deterministic consensus algorithms to record information on private ledgers that are only available to specific authorized participants.

In this research, Ethereum is considered a public BC. Public platforms are particularly well-suited for the construction industry due to their transparency and decentralization. These features ensure that all parties involved in construction projects have equal access to information, which helps reduce fraud and enhance trust. Additionally, the decentralized nature of public BCs eliminates the need for intermediaries, streamlining processes and reducing costs. Ethereum's robust SC capabilities further automate and secure contract execution, making it an ideal choice for the construction industry's complex, multi-stakeholder environment.

The Ethereum BC platform (Figure 2) generates and deploys SC code. Ethereum [52] was the first BC platform to support the development of SCs. It allows complex and customizable SCs using the Ethereum virtual machine (EVM). EVM is the SC runtime environment, and every node in the Ethereum network runs an EVM implementation and performs the same instructions. SCs are written in Solidity, a high-level programming language inspired by C++, Python, and JavaScript, and the contract code is compiled down to EVM bytecode before being placed on the BC for execution. SC code was developed to automate payment processes, which are programmed to release payments upon meeting pre-defined conditions validated through the BIM model. Ethereum is now the most popular SC development platform, and it can be used to create various decentralized apps (DApps) in various disciplines. Therefore, Ethereum was chosen as the BC platform. Nonetheless, the proposed SC procedure can also be executed on other SC platforms.

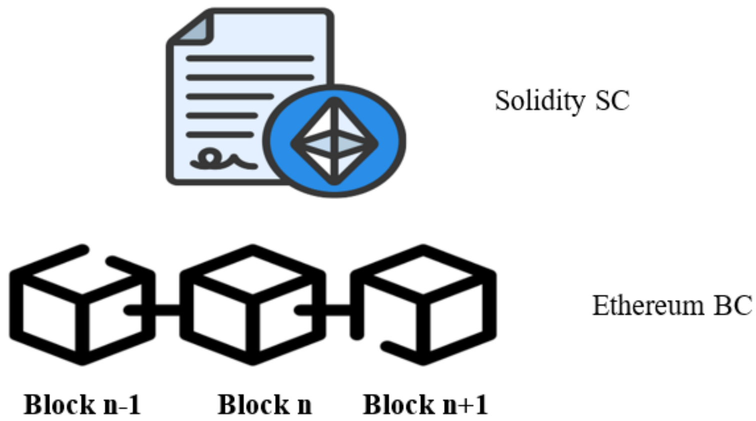

Figure 2. Solidity smart contract living on the Ethereum blockchain.

4.3. Automated Payment Workflow Using Blockchain and BIM Integration

Figure 3 outlines an automated payment process for construction projects integrating a BC SC system with a scan-to-BIM technology. The workflow initiates with a new SC in the BC environment, corresponding to the start of a construction contract. As the project progresses, contractors submit reports of completed work, which are then validated against a digital twin of the construction site, a BIM model updated through 3D scanning. The SC verifies the submitted work against the BIM model data through the BC network, ensuring that payment requests are consistent with actual progress. Upon confirmation, the BC mechanism triggers an automated payment transfer to the contractor, encapsulating a reliable, transparent, and efficient approach to managing construction finances. This process provides a significant leap toward optimizing project delivery by mitigating discrepancies and delays in construction payments.

The automated payment workflow in construction projects, powered by the integration of BC and scan-to-BIM systems, is underpinned by a robust mathematical model that facilitates SC creation. The model operates on data variables captured over time, as follows: (1) Scan-to-BIM data at time t, S(t). (2) Contractor reported quantity R(t). (3) Uses a validation function C(S(t), R(t)) to ensure reported quantities align with actual progress. If the reported quantities meet the specified criteria of the scanned data, they are deemed acceptable. (4) Approved quantities V(t) are then calculated, leading to the automatic triggering of payments P(t) based on the unit price (u). This mathematical model is critical in automating the entire payment process, from validating work completion to executing the payment, reflecting the actual progress of the construction project.

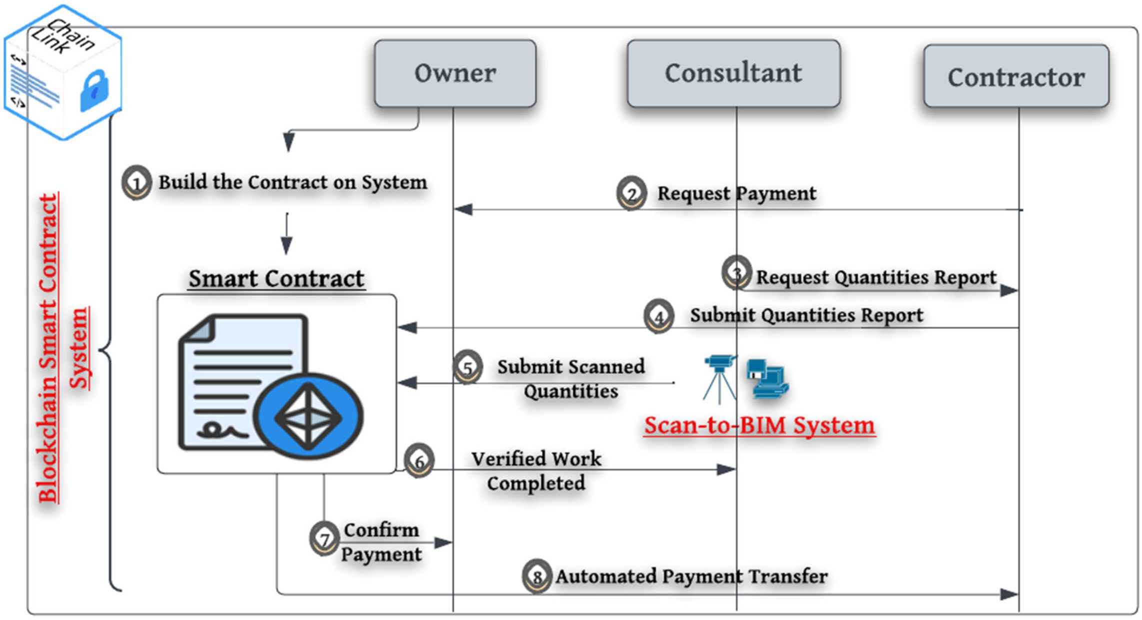

Figure 3. Automated payment workflow using BC, SC, and scan-to-BIM integration.

4.4. Mathematical Formulation

The reported quantity of item i at time t, R_i^t, is acceptable if it is up to 10% less than the scanned data, S_i^t. This 10% threshold is assumed based on the contract payment terms and can be adjusted based on the different case studies. Thus, the acceptance of reported quantities, $C_i^t(S_i^t, R_i^t)$ shown in Equation (1), is a binary term, which helps in validating the payments at time t.

$$C_i^t(S_i^t,\ R_i^t) = \begin{cases} 1 & if\ 0.9 \times S_i^t \le R_i^t \le S_i^t \\ 0 & otherwise \end{cases} \tag{1}$$

To this end, based on the approval of quantities at time t, the payment (P^t) is calculated as the sum of all of approved quantities by their unity price, u_i, as shown in Equation (2).

$$P^t = \sum_i C_i^t \times R_i^t \times u_i \tag{2}$$

This model leverages BC technology to automate the payment process in a transparent and immutable way. The process begins with data capture through 3D scanning, followed by steps that validate the reported work, approve quantities, and calculate payments, which are then executed and recorded using BC.

5. Case Study

This section presents a case study based on a real construction project, the Majid Al Futtaim (MAF) shopping mall, illustrating that the developed novel framework performs as anticipated. The case study project was selected based on several key criteria to ensure its relevance and suitability for applying the proposed framework. First, the project used traditional methods that were slow and error-prone, making it ideal for evaluating the automation of payment processes using BC-enabled SCs and scan-to-BIM. Secondly, the project was equipped with the necessary infrastructure, such as 3D-scanning tools, to implement scan-to-BIM, which was essential for capturing work progress. Additionally, the project provided complete access to necessary data, including contracts, progress reports, and payment records, which were crucial for applying the proposed approach. The traditional payment process in this project typically takes 2 to 3 weeks and involves several steps, each contributing to delays. It starts with the manual quantification and verification of the work performed on site, requiring detailed and precise measurements to ensure all work meets contractual requirements. Following this, the contractor submits an invoice, which is then manually reviewed by the owner's consultant for accuracy and completeness, confirming that the claimed work aligns with the actual completed work. The reviewed invoice undergoes multiple layers of approval within the project owner's organization, where each layer checks for compliance with financial policies and contractual terms, further extending the processing time. Finally, once approved, the payment order is issued and processed through centralized banks, adding additional delays due to the bank's transaction time and potential discrepancies or issues that may need resolution. This traditional process is inherently time-consuming due to the necessity of manual measurements, multiple approval stages, and the time required for bank transactions.

To address these delays, this study used a cutting-edge approach to automate construction payments by integrating BC-enabled SCs and scan-to-BIM, where scan-to-BIM behaves as a chain-link to connect real-world data from off-chain to on-chain sources. Regular scans of the construction site were conducted using a 3D scanning tool (Matterport Pro 2 camera), which employs structured light and infrared depth-sensing technologies to record point cloud data and build detailed 3D Revit models of actual site progress. The actual quantities derived from scan-to-BIM are inputted into SCs on the Ethereum BC. This SC is pre-programmed with terms that automatically validate the completion of work and trigger payments. The project owner, consultant, and contractor are the SC parties, ensuring transparency and trust in the automated payment process. The transactions were performed on the Ethereum test network, and a BC-enabled SC network was constructed among the project parties. To validate the effectiveness of this proposed system, the scenario was tested to measure the improvement in payment processing time compared to the traditional method.

This study used photogrammetry, an advanced 3D scanning technology, to significantly enhance the scan-to-BIM process in several ways. Photogrammetry is capable of producing detailed 3D scans of building interiors, ensuring digital models accurately depict actual spaces, which is critical for BIM. Meanwhile, relying on high-resolution scans, accurate measurements within 1% accuracy can be achieved. This is all implemented in a user-friendly approach and can be easily integrated to well-known BIM environments and software [54].

The scan-to-BIM workflow can be divided into three main steps: (1) Scan, which involves taking dense point measurements of the state using scanning camera; (2) data processing to produce a point cloud as an output, which is a discrete representation of reality; and (3) BIM modeling using the point cloud to create a BIM model that virtually represents the real-world surveyed object.

A point cloud is a 3D description of spatial data of geometric points in space representing the 3D shape of a physical object. Each point position has its set of Cartesian coordinates (X, Y, Z). Point clouds are most commonly used to capture the as-built conditions of a structure. They are obtained through photogrammetry and are highly accurate in terms of surveying non-regular geometries and saving time in as-built modeling of complex buildings and infrastructures. Creating a 3D model using the 3D scanning tool involves a comprehensive, multi-step process integrating various technologies and software applications. The initial phase focuses on preparation, ensuring the scanning area is clean, well-lit, and free from moving objects or individuals. Upon completing the scanning phase, the collected data undergo processing within the camera capture application. These processed data are then uploaded to the cloud platform, which is subjected to further processing and stitching to create a cohesive 3D model. Once the scan data are fully processed, an XYZ point cloud file is generated for further use.

The integration with BIM technology begins by importing the cloud file into Autodesk Recap. This crucial step converts the point cloud data into a compatible format for 3D modeling, paving the way for the final stage of BIM model creation. The operator creates a detailed 3D model by tracing over the point cloud data to construct architectural elements such as walls, columns, and floors using the Autodesk Revit tool. This model is progressively refined and detailed, adding materials, finishes, and other architectural details, culminating in a highly accurate and richly detailed digital twin of the scanned space. This intricate process, from site preparation to the final detailing model, exemplifies the integration of advanced scanning and modeling technologies to capture and digitize physical spaces with remarkable precision.

The generated BIM model enables the extraction of accurate quantities and dimensions of building components. Finally, the actual quantities performed are integrated into a BC-based Ethereum platform that establishes a peer-to-peer network that securely executes and verifies application code. This platform uses SCs to automate payments. The SCs are programmed to execute transactions automatically when certain pre-defined conditions are met, such as completing a work segment. This ensures transparency, security, and efficiency in the payment process.

The authors utilized Autodesk ReCap for point cloud processing by importing the XYZ point cloud data file. This phase was critical for transforming the rich spatial data acquired by the scanning camera into a complete point cloud dataset, which enabled modeling and visualization of the scanned area (Figure 4). The scanning procedure resulted in a point cloud of around 27.16 million points. This large information represents the complete and comprehensive capture of the scanned 3548 m^2 area's spatial dimensions and textures, offering a high-resolution 3D representation suited for a precise space measurement, converting this data into a BIM format, using the generated XYZ point cloud file.

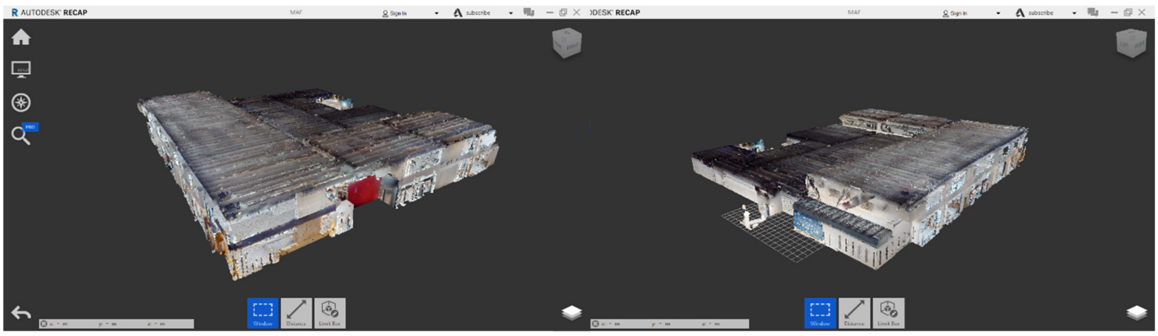

Figure 4. 3D point cloud model in Autodesk ReCap.

The processed point cloud in Autodesk ReCap (Figure 5a) was then integrated into a fully functional 3D model in Autodesk Revit (Figure 5b), marking a significant milestone in the BIM process. This includes developing a detailed structural model and extracting the actual quantities the contractor performs and measurements straight from the model. This functionality significantly streamlines the payment workflow, making it faster and more accurate.

(a) (b)

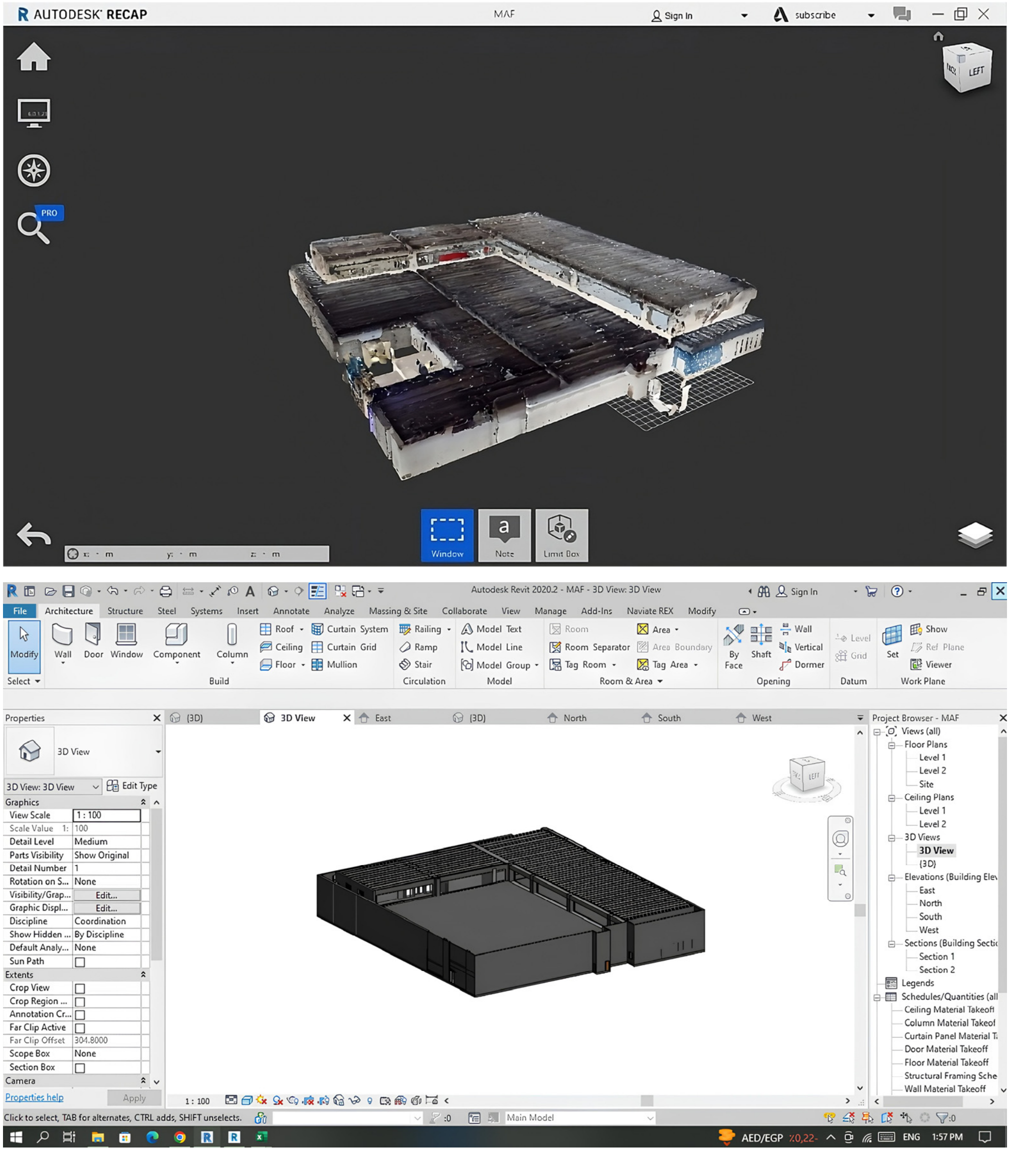

Figure 5. (**a**) Three-dimensional model in Autodesk ReCap; (**b**) 3D model in Autodesk Revit.

5.1. Ensuring Dimensional Accuracy in the Scan-to-BIM Process

A crucial part of the scan-to-BIM process is ensuring the correctness of elements by comparing their measurements in both the point cloud data and the associated Revit model. Using a column as an example, Figure 6a describes how to guarantee that the dimensions received from the detailed scan are appropriately reflected in the Revit model (Figure 6b). Also, by comparing the architectural CAD drawings (Figure 7a) with the point cloud data (Figure 7b) of the project, it is evident that the dimensions captured were very close to the measurements depicted in the CAD drawings. The scan results show that the accuracy of the 3D camera's measurements fell within a margin of +/− 1%, confirming that the scanning technology provides precise and reliable dimensional data. This level of accuracy is critical for ensuring the validity and trustworthiness of the captured measurements for progress verification and payment automation.

(**a**)

(**b**)

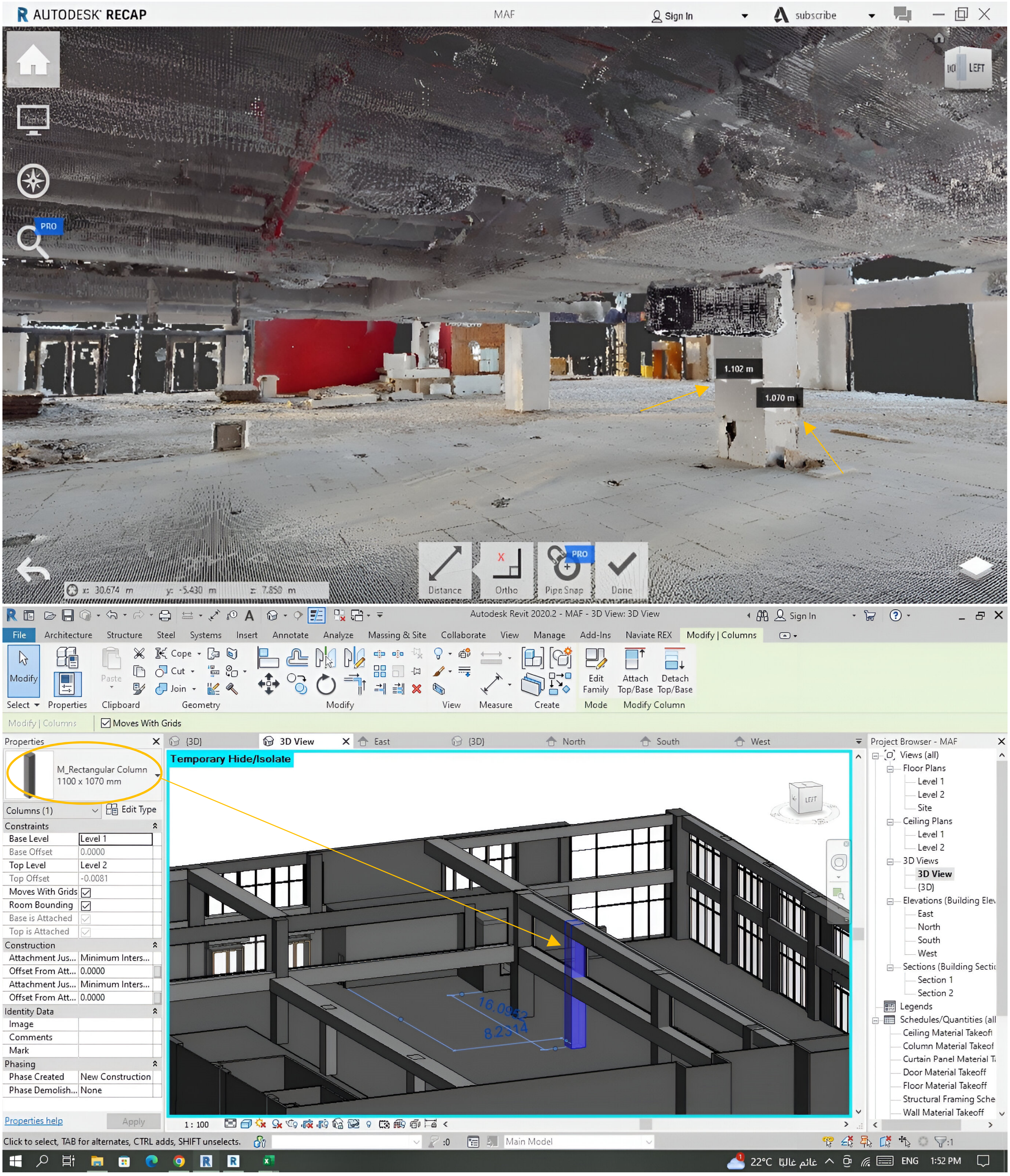

Figure 6. (**a**) Column dimensions in ReCap point cloud; (**b**) column dimensions in the Revit model.

(**a**)

(**b**)

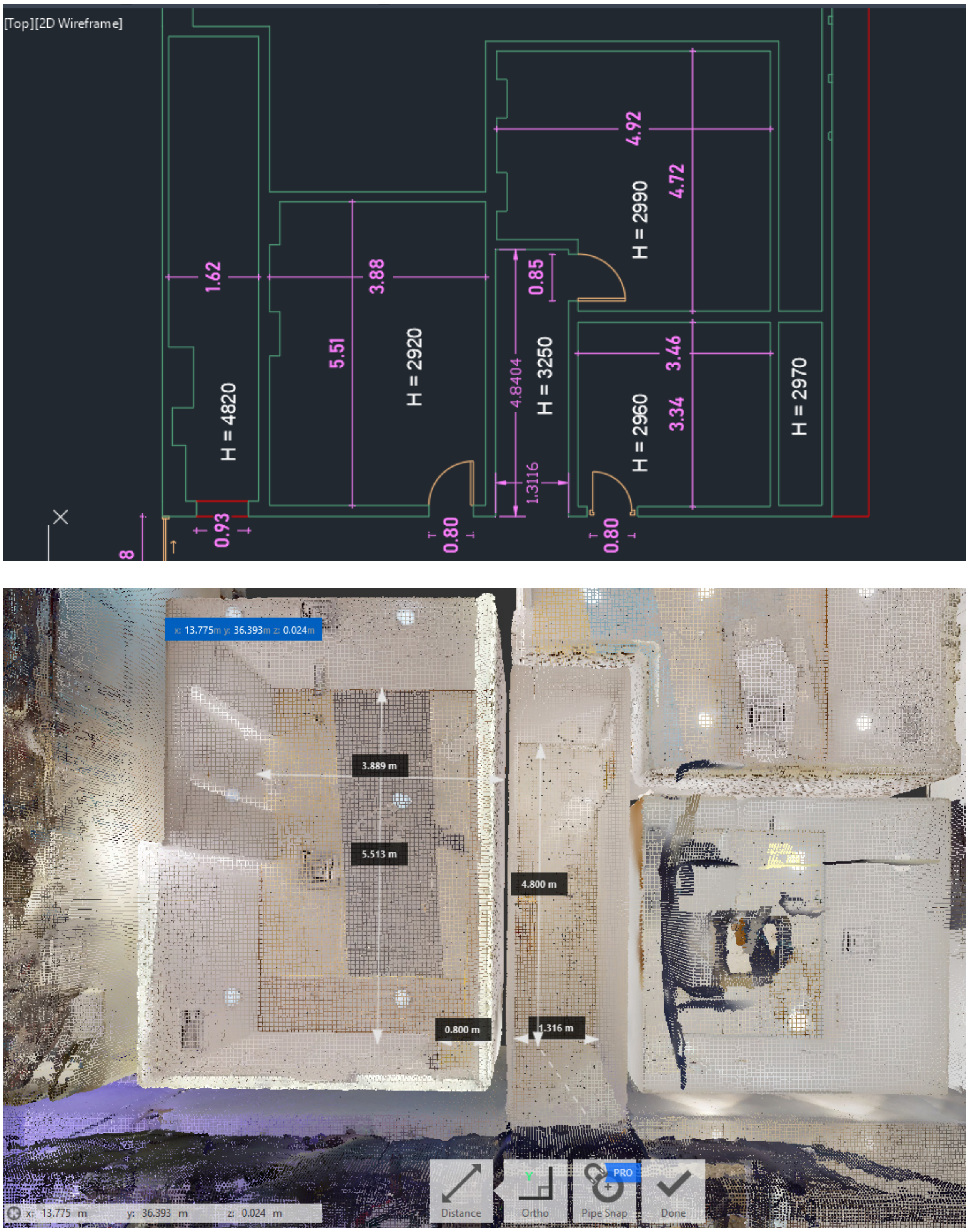

Figure 7. (**a**) Space dimensions in the 3D scanned data; and (**b**) in the architectural CAD drawings.

5.2. Model Automation

The use of SCs in this BC network automated the payment process using validated Revit data from consultant scans of actual quantiles performed to trigger payments from owner to contractor. The BC system architecture developed for the project includes a simplified and transparent process involving project stakeholders as the owner, contractor, and consultant. This system manages and validates transactions using the accuracy of scanned quantities of data and BC security.

The proposed method for automating construction payments within the blockchain framework was thoroughly built in Solidity, a programming language created exclusively for creating SCs on the Ethereum platform. The SCs were written with pragma solidity > 0.7.0 < 0.9.0 (Appendix A provides further clarification). to ensure compliance with Ethereum BC capabilities and security requirements. The SC includes various structures (data structures) (Figure 8) that represent the different entities and states in the construction payment process.

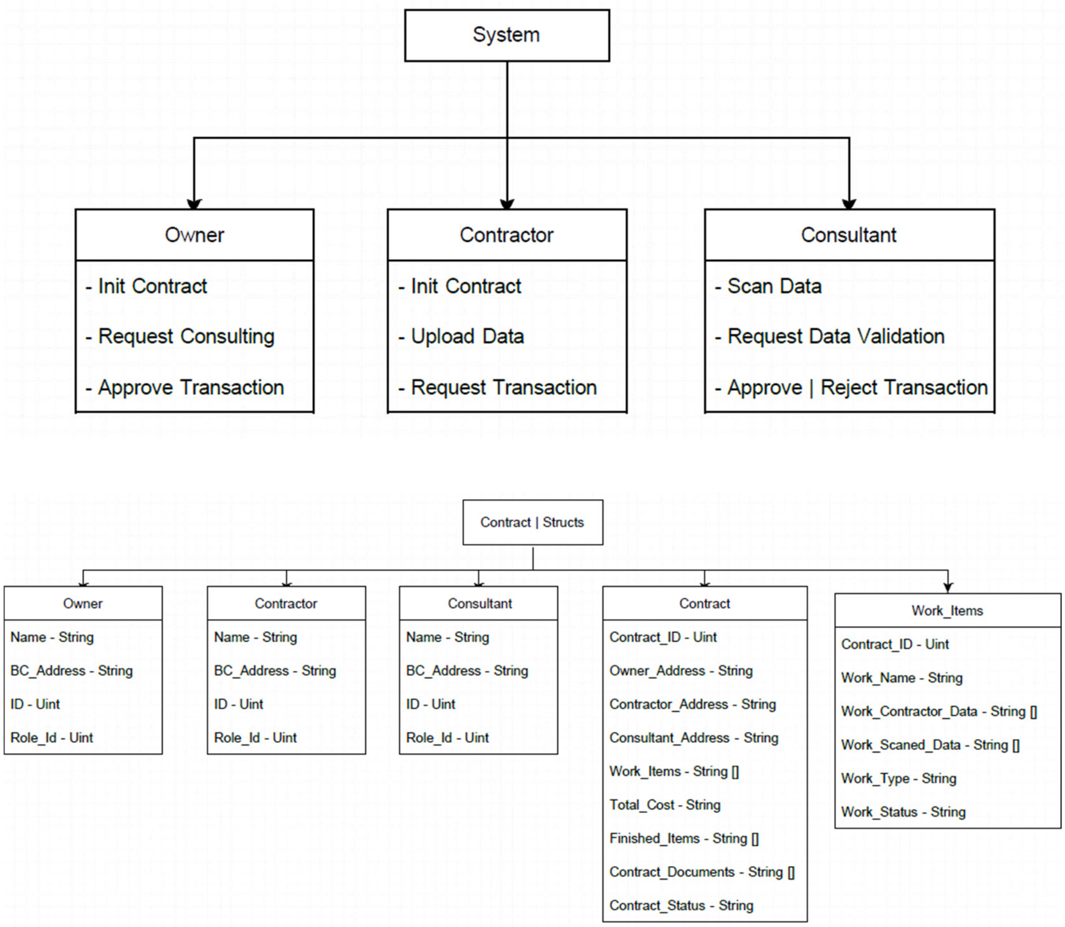

Figure 8. System and structs of the smart contract in Solidity.

The contractor begins the process by submitting a full report on the quantities, done as part of the construction project to the BC SC system. This report simply requests the invoice outlining the work elements completed and the price needed. Upon receipt of the contractor's report, the consultant plays a crucial role in verifying the claimed work through SC. The consultant conducts a site scan using the scanning camera that captures the current state of the construction site in detail. The data from the 3D scan camera are then used to accurately assess the actual quantities of work completed by the contractor. This step provides an objective, data-driven basis for verifying the contractor's invoice request. Once it is confirmed through a SC that the contractor's reported quantities align with the actual work completed on site with a tolerance percentage of 10% less, as evidenced by the camera's scanned data, they approve the transaction. The approval triggers a SC within the BC system. This contract is pre-programmed with terms agreed upon by the owner, contractor, and consultant. Upon meeting the specified criteria in the SC, the system automatically initiates payments from the owner to the contractor, reducing the need for manual intervention and verification. The entire transaction, from the contractor's initial invoice request to the final payment, is recorded on the BC to complete the first invoice with eight blocks created. This provides a transparent, immutable record of all transactions, ensuring trust and accountability among all parties.

Remix and Visual Studio Code with Solidity extensions were used to create the SCs in a Solidity-friendly environment. This setting offered a stable foundation for creating, testing, and implementing the contracts. The SC designed to encapsulate the essential functionalities required for automating the payment process in the construction project includes project participant's registration, validation of work elements, and payment processing. Setting up and operating a BC network on a server for development and testing is more accessible with Ganache. Ganache is part of the Remix Integrated Development Environment (IDE), a tool for Ethereum development. It is a personal BC for Ethereum development that mimics a BC environment on the local PC or server, making it simple to create and test SCs.

Figure 9a shows the compilation and deployment of the Ethereum SC using Remix IDE. Once the contract is deployed, a distinct and immutable contract address (Figure 9b) is automatically produced and presented. This contract address is a definitive identification for the deployed contract within the BC network, and the contract can be interacted upon by calling its functions directly from Remix.

(**a**)

(**b**)

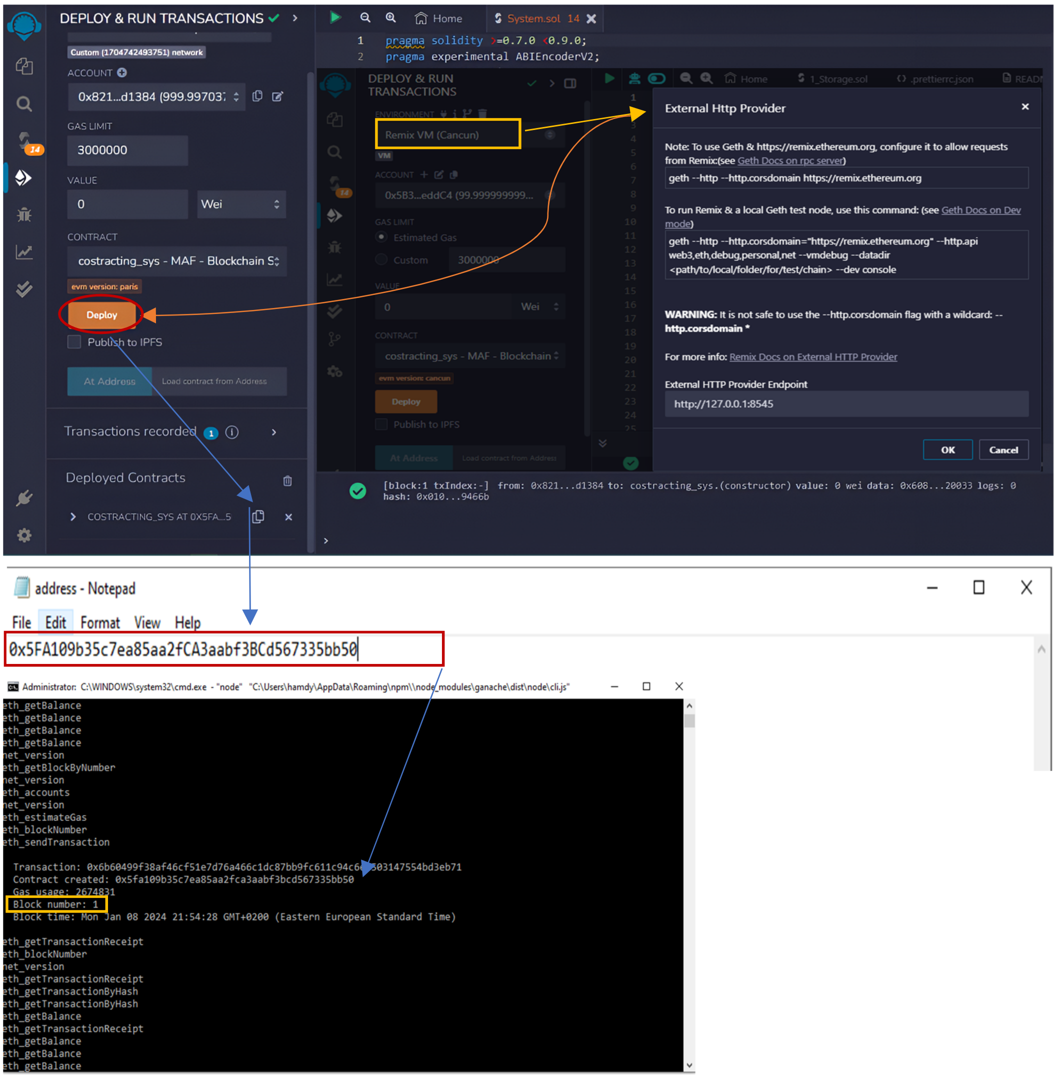

Figure 9. (**a**) Remix IDE showing smart contract deployment. (**b**) Contract address.

The application's user interface simplifies onboarding in a blockchain environment. The process begins with the owner registering on the platform (Figure 10), creating their unique blockchain address to establish their secure identity. Similarly, each participant, such as the contractor and consultant, registers and receives their own unique address. This decentralized system ensures that every stakeholder has a distinct and tamper-proof identity, enabling secure and transparent interactions throughout the project lifecycle.

Figure 10. Owner registration on the blockchain platform.

The following section shows the project participants joining the network, starting with the owner who initiates a contract (Figure 11) to choose the remaining parties and their number depending on whether there is a main contractor or group of sub-contractors. Then, we determine how many work elements, for example, the following will test finishing the structural concrete columns and inserting the total price of the contract. After that, the contractor joins the network, uploads the quantities (Figure 12) of finished concrete columns, and requests the first invoice. Finally, the consultant joins and verifies the work (Figure 13) using the scanned data for the actual quantities performed, and the owner transacts money for the accepted elements through the BC network. The system loops back to the operation and validation phase to indicate the ongoing nature of the project until completion.

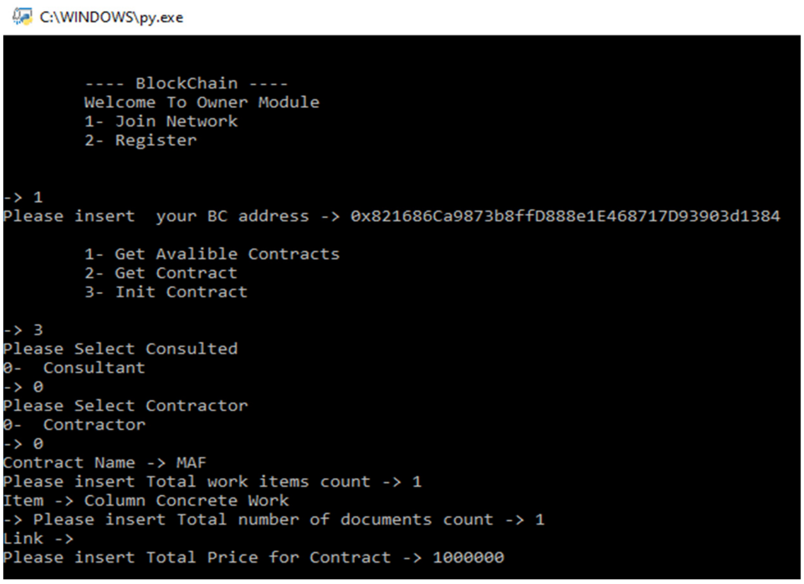

Figure 11. Owner joining and initiating the contract.

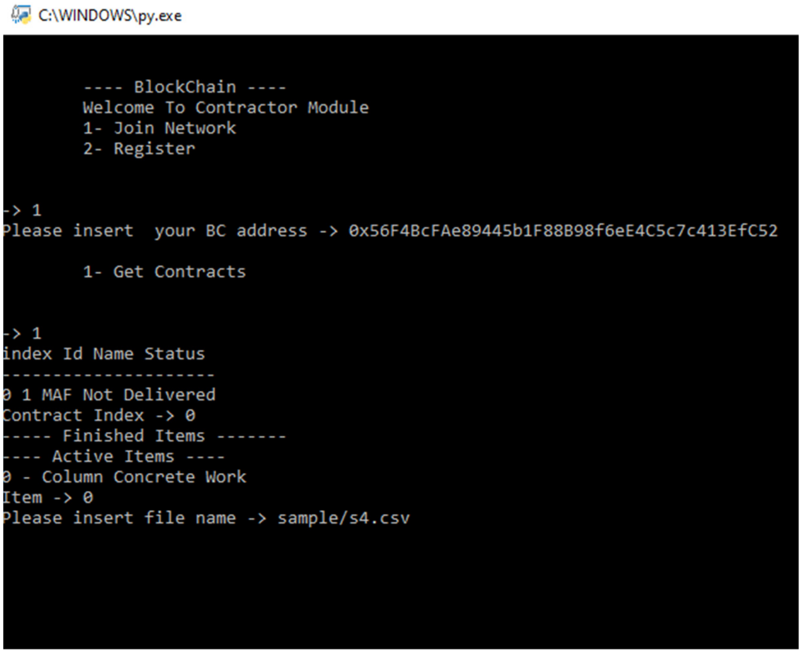

Figure 12. Contractor joining and uploading the quantities report.

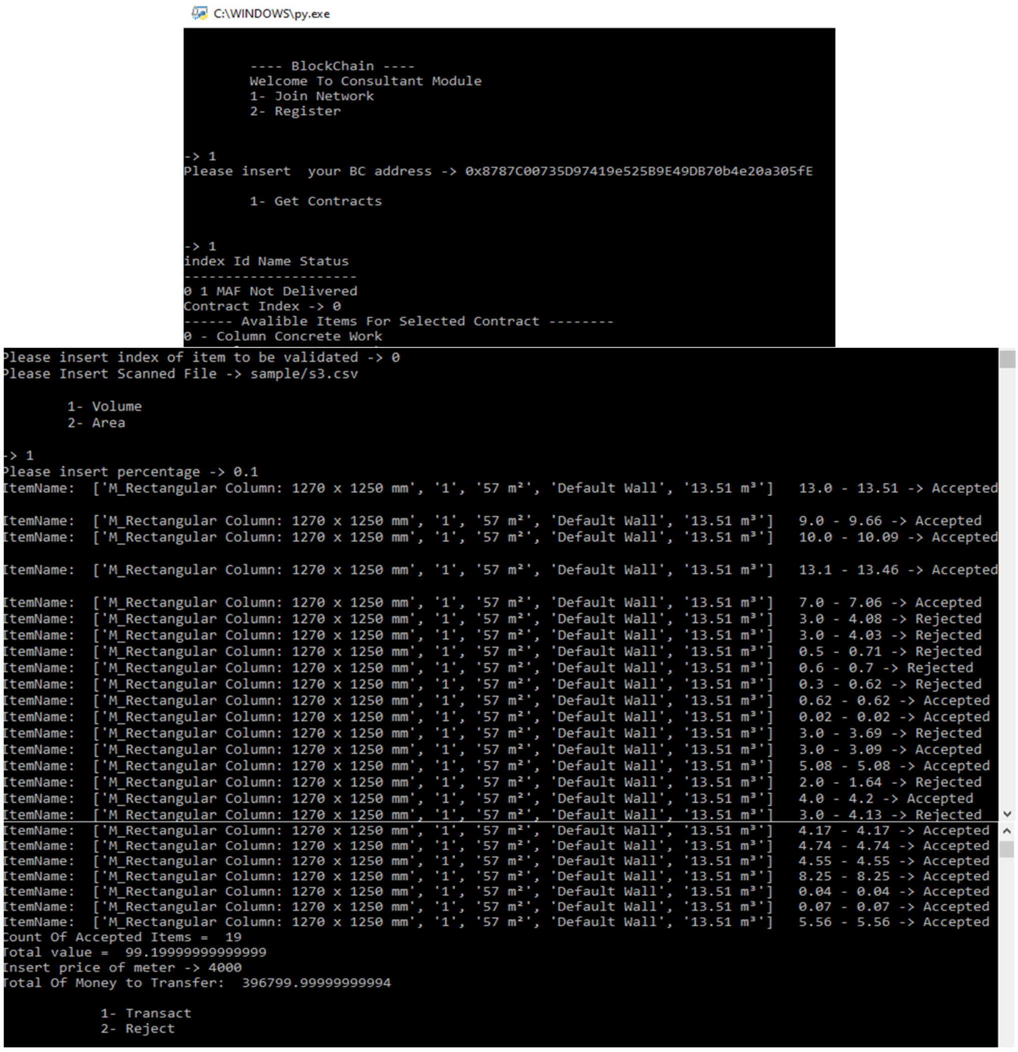

Figure 13. Consultant verification and transaction process.

6. Discussion and Limitations

This study showcased a combination of scan-to-BIM technology with BC-based SCs to revolutionize the construction payment process, directly addressing the research objectives. Specifically, by utilizing scan-to-BIM, the system delivered precise, real-time digital representations of the scanned site, ensuring that payments were initiated only after verified work completion, thereby enhancing transparency and minimizing disputes. To evaluate the effectiveness of the proposed BC-enabled SC system, it was applied it to a real-world construction project previously managed with traditional payment methods. This application demonstrates how the developed framework can ensure accurate and efficient automation of payments, thereby minimizing errors and disputes. The traditional process for processing and releasing payments for completed milestones typically required 2–3 weeks due to manual handling of contracts and multiple layers of approval. In contrast, the proposed model drastically reduced the payment processing time to 1 to 2 days. This significant improvement was due to the automation of contractual obligations and the immediate release of payments upon completing pre-defined milestones. Several vital benefits evidence the effectiveness of the proposed system. This practical implementation validates the framework's capability to minimize errors, enhance efficiency, and ensure accurate payment processes in a real-world setting.

Firstly, it promotes enhanced transparency and trust by using the BC component to ensure all transactions are transparent and immutable, thereby boosting confidence among all stakeholders. Secondly, it reduces payment disputes as payments are automated based on objective data from scan-to-BIM, linking them directly to independently verifiable physical progress. Additionally, the system streamlines the payment process, thereby decreasing the amount of administrative work required and eliminating delays, improving cash flow and financial planning for all parties involved. Continuous monitoring through scan-to-BIM also allows for better project management and informed decision making.

Compared to traditional methods (Table 1), the proposed system in the case study reduces the workload significantly by data capture through scan-to-BIM, thus reducing the manual effort required for on-site measurement and documentation. The traditional method involves the contractor submitting payment requests based on daily progress monitoring and work reports, as well as the consultant reviewing the contractor's workload with occasional on-site checks for disputed areas. Also, the bank centralization delays the payment process due to manual verification and reconciliation. In the proposed system, contractors continue to submit payment requests, and consultants use scan-to-BIM to verify and confirm all completed work using an automated BC-enabled SC platform. This automated verification reduces the manual effort required for on-site verification of progress and minimizes the need for manual documentation and reporting.

To introduce this system into a project, staff need to undertake several actions. Initially, there will be a requirement for training and education to understand the operations of scan-to-BIM technology and BC-based SCs. This includes learning to use the scanning equipment, interpret the data, and interact with the BC system. Furthermore, the necessary equipment, scanning cameras, need to be procured and set up, along with the installation and configuration of relevant software for data processing and BC transactions. Staff must also adjust their workflows to incorporate the new system, including new procedures for capturing and verifying progress data and submitting payment requests through the BC platform.

The introduction of this system impacts project costs in various ways. Initially, there are increased costs due to the upfront investment required for purchasing scanning equipment, BC software (Ethereum IDE (Integrated Development Environment)), and training staff. There are also ongoing costs for system maintenance, support, and potential upgrades. However, these costs are offset by significant savings in administrative labor due to the reduced need for manual verification and approval of payment requests and the decrease in payment disputes and delays, which improves cash flow and financial planning. Additionally, centralizing banks to authorize payments in the traditional payment process adds more cost and time for the owner. The involvement of banks means that invoices must go through a series of checks and authorizations, which can incur banking fees and lead to significant delays. This centralization increases the administrative burden and imposes financial costs on the owner for processing invoices.

Table 1. Comparative analysis of traditional paper-based contracts vs. blockchain-enabled smart contract with scan-to-BIM.

Aspect	Traditional Paper-Based Contracts	Blockchain-Enabled Smart Contract with Scan-to-BIM
Time consumed	High: 2–3 weeks per payment cycle due to manual measurement, verification, and the presence of a central administrative body.	Low: 1–2 days per payment cycle as automated measurement, verification, and smart contracts expedite the process.
Progress verification	Manually based on daily progress monitoring and work reports and is verified through on-site inspections.	Automated, using scan-to-BIM technology to capture real-time, accurate data on construction progress.
Efficiency	Time-consuming and prone to human error, leading to potential delays and inaccuracies.	Significantly reduces verification time and minimizes human error, leading to faster and more accurate payments.
Transparency	Limited; relies on manual reports and inspections, which might not always be objective.	Enhanced with immutable BC records and objective, independently verifiable data from scan-to-BIM technology.
Administrative overhead	High due to manual verification, documentation, and coordination between multiple parties.	Lower due to automation of progress verification and payment processes.
Payment authorization	Centralized through banks or financial institutions, adding extra steps, fees, and potential delays.	Decentralized payments are triggered automatically upon verification of progress data on the BC, reducing reliance on banks.
Payment disputes	More frequent due to subjective progress reports and manual verification processes.	Less common due to objective and verifiable data and transparent, immutable records.
Legal and regulatory validity	Well-established legal frameworks ensure the enforceability and compliance of its provisions across jurisdictions.	Lacks a fully developed legal infrastructure. Current systems do not have established mechanisms for resolving disputes related to smart contracts in courts or arbitration. Conflicts are challenging to address due to the decentralized and automated nature of the technology [33].

The developed integrated BC-enabled SC system, coupled with scan-to-BIM technology, holds great promise as a groundbreaking solution for government agencies in the construction sector. It can revolutionize traditional practices by streamlining processes, enhancing transparency, and reducing inefficiencies. This innovation represents a significant opportunity for government entities to embark on a transformative journey toward modernization within the construction industry. More research is needed to develop strategies to manage the transition to this new system and address resistance to change among stakeholders. Companies should establish training and education programs for industry professionals to familiarize them with the new technologies and processes.

Although there are many benefits to the suggested approach, there are also several drawbacks that call for more investigation. Adopting advanced technologies like BC and scan-to-BIM requires specific expertise and could face resistance from small firms due to the high upfront costs and the complexity of integrating new technologies into existing systems. The demonstration conducted in this study was based on a controlled test environment using Ethereum blockchain and predefined conditions, which may not be able to represent the complexity of real-world projects, including unexpected changes, or diverse stakeholder dynamics. Moreover, ideal conditions regarding equipment and light were required for the process of scan-to-BIM, which limits the practicality in a diverse construction environment. Also, other factors such as dust, vibrations, or frequent site changes, which are common in the construction industry, were not considered and could impact the reliability of the scan-to-BIM process. This study did not consider potential resistance from stakeholders unfamiliar with blockchain or scan-to-BIM technologies, which may resist their adoption in practice. The interoperability of this system with existing financial and project management software is critical for its widespread adoption and necessitates additional research into seamless integration methods. Moreover, legal and regulatory challenges in applying SCs to construction payments need to be addressed to provide a solid legal foundation for the technology. Finally, to accommodate various project sizes and types, the system must be scalable and customizable, which calls for developing adaptable models suitable for a broad range of construction projects.

7. Conclusions and Future Work

This paper presented a novel approach for automatic construction payments by merging BC-enabled SCs and 3D scans with BIM. Thus, the model can measure actual quantities performed and automate payments through a BC platform. This integration is a huge step forward in tackling the construction industry's long-standing inefficiencies and trust difficulties. This research has demonstrated the feasibility and effectiveness of utilizing scan-to-BIM to accurately estimate work completed on site, which is then verified against contractor-submitted quantities through SCs. These SCs serve as an automated intermediary, ensuring that payment is triggered only upon validating the actual work performed, thus minimizing disputes and enhancing transactional transparency. This technique promises to simplify payment processes, improve transparency, and reduce conflicts by combining the precision of scan-to-BIM with the security of BC technology. The research findings indicate that this developed system can significantly reduce the time and resources spent on payment processing while providing a more objective and transparent basis for payment release. BCT ensures that all stakeholders have access to an immutable and transparent record of transactions, fostering a higher level of trust among parties. Additionally, the real-time data provided by scan-to-BIM allows for more accurate project progress tracking, ensuring that payments are closely aligned with the work completed.

In light of these findings, future research may investigate the scalability and cost-effectiveness of scan-to-BIM technology and BC adoption in construction projects of varying sizes and complexities. Additionally, research on SCs' legal and regulatory implications in construction payments is essential to address potential legal challenges and ensure compliance with local and international regulations. Moreover, future research should emphasize the cross-validation for quantities performed derived by contractors, ensuring an integrated examination from both consultant and contractor standpoints using scan-to-BIM to assure the process. Also, exploring more advanced BC platforms that offer higher scalability, better security, and lower operational costs can be a significant area of research. Creating intuitive, user-friendly interfaces is crucial for scan-to-BIM and BC platforms, facilitating easier adoption, especially for stakeholders not well-versed in these technologies.

Author Contributions: Conceptualization, H.W. and H.E.; methodology, H.W. and H.E.; software, H.E.; validation, H.W., E.E., M.S.E., W.A. and H.E.; formal analysis, H.W. and H.E.; investigation, H.W., E.E., M.S.E., W.A. and H.E.; writing—original draft preparation, H.W. and H.E.; writing—review and editing, H.W., E.E., M.S.E., W.A. and H.E.; visualization, H.W., E.E., M.S.E. and H.E.; supervision, H.W., E.E., M.S.E. and H.E.; project administration, H.W., E.E., M.S.E. and H.E.; funding acquisition, W.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The Researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support (QU-APC-2025).

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Smart Contract Code Components

Two critical components of smart contract code used are the defining of structs (Figure A1) and the coding of "operation functions" (Figure A2). The "structs" were meticulously constructed to contain the framework for the smart contract's data management. Meanwhile, the operation functions were created to express the essential functionality of our smart contracts. These functions provided the logic for starting transactions, verifying conditions, and carrying out the contract terms.

F: > MSc Thesis Project > Blockchain & Smart Contracts Project > Blockchain & Smart Contracts Project 2 > smartcontract > system.sol

```
pragma solidity >=0.7.0 <0.9.0;
pragma experimental ABIEncoderV2;
contract costracting_sys
{
    // owner details struct;
    struct Owner
    {
        string Own_name;
        string Own_BC_Address;
        uint Own_ID;
        uint Own_Role_ID;
    }

    // contractor  details struct
    struct Contractor
    {
        string con_name;
        string con_BC_Address;
        uint con_ID;
        uint con_Role_ID;
    }
    // consulted  details struct
    struct Consulted
    {
        string cons_name;
        string cons_BC_Address;
        uint cons_ID;
        uint cons_Role_ID;
    }
```

F: > MSc Thesis Project > Blockchain & Smart Contracts Project > Blockchain & Smart Contracts Project 2 > smartcontract > system.sol

```
{
    //work items details struct
    struct Work_Items
    {
        uint id;
        uint WI_Contract_ID;
        string Work_Name;
        string[] Work_Contractor_Data;
        string[] Work_Scaned_Data;
        string Work_Type;
        string Work_Status;

    }
    //contract details struct
    struct Contract
    {
        uint Contract_ID;
        string Contract_Name;
        string Owner_Address;
        string Contractor_Address;
        string Consulted_Addresss;
        string[] Work_Items;
        string Total_Cost;
        string[] Finished_Items;
        string[] Contract_Documents;
        string Contract_Status;
    }
    // Structs Arrays Section
    Owner[] owner;
    Contractor[] contractors;
    Consulted[] consulted;
    Work_Items[] work_items;
    Contract[] contracts;
```

Figure A1. Solidity smart contract structs codebase.

```
                                        //operation section
    //function to set ownerdata
    function setOwner_Data(string memory Oname,string memory OBC_address,uint role_id) public returns (uint)
    {
        uint _id = owner.length + 1;
        Owner memory temp=Owner(Oname,OBC_address,_id,role_id);
        owner.push(temp);
        return 1;
    }
    //cout data
    function getOwnerCount() public returns(uint)
    {
        return owner.length;
    }
    //function to get ownerdata
    function getOwnerData(uint _id) public returns(uint,uint,string memory,string memory)
    {
        Owner memory temp = owner[_id];
        return (temp.Own_ID,temp.Own_Role_ID,temp.Own_name,temp.Own_BC_Address);
    }
    function getOwnerDataByAddress(string memory _Address) public returns(uint,uint,string memory,string memory)
    {
        for(uint i = 0; i < owner.length; i++)
        {
            Owner memory temp = owner[i];
            if(compareStringsbyBytes(temp.Own_BC_Address,_Address))
            {
                return (temp.Own_ID,temp.Own_Role_ID,temp.Own_name,temp.Own_BC_Address);
            }
        return(0,0,"0","0");
    }
    //function to set contractordata
    function setConstractor_Data(string memory Cname,string memory CBC_address,uint C_role_id) public returns (uint)
    {
        uint C_id = contractors.length + 1;
        Contractor memory temp=Contractor(Cname,CBC_address,C_id,C_role_id);
        contractors.push(temp);
        return 1;
    }
    //cout data
    function getContractorCount() public returns(uint)
    {
        return contractors.length;
    }
    //function to get contractor data
    function getContractorData(uint C_id) public returns(uint,uint,string memory,string memory)
    {
        Contractor memory temp = contractors[C_id];
        return (temp.con_ID,temp.con_Role_ID,temp.con_name,temp.con_BC_Address);
    }
    function getContractorDataByAddress(string memory C_address) public returns(uint,uint,string memory,string memory)
    {
        for(uint i = 0; i < contractors.length; i++)
        {
            Contractor memory temp = contractors[i];
            if(compareStringsbyBytes(temp.con_BC_Address,C_address))
            {
                return (temp.con_ID,temp.con_Role_ID,temp.con_name,temp.con_BC_Address);
        }

        return(0,0,"0","0");
    }
    //function to set consulted
    function setConsultedData(string memory Consname,string memory ConsBC_address,uint Cons_role_id) public returns (uint)
    {
        uint Cons_id = consulted.length +1;
        Consulted memory temp=Consulted(Consname,ConsBC_address,Cons_id,Cons_role_id);
        consulted.push(temp);
        return 1;
    }
    //cout data
    function getConsultedCount() public returns(uint)
    {
        return consulted.length;
    }
    //function to get consulted data
    function getConsultedData(uint Cons_id) public returns(uint,uint,string memory,string memory)
    {
        Consulted memory temp = consulted[Cons_id];
        return (temp.cons_ID,temp.cons_Role_ID,temp.cons_name,temp.cons_BC_Address);
    }

    function getConsultedDataByAddress(string memory _Address) public returns(uint,uint,string memory,string memory)
    {
        for(uint i = 0; i < consulted.length; i++)
        {
            Consulted memory temp = consulted[i];
```

Figure A2. *Cont*.

```
            if(compareStringsbyBytes(temp.cons_BC_Address,_Address))
            {
                return (temp.cons_ID,temp.cons_Role_ID,temp.cons_name,temp.cons_BC_Address);
            }
        }

        return(0,0,"0","0");
    }
    //function to set work items
    function setWorkitemsData(uint wi_contract_id,string memory work_name,string[] memory work_contractor_data,string[] memory work_scaned_data,s
    {
        uint id = work_items.length + 1;
        Work_Items memory temp = Work_Items(id,wi_contract_id,work_name,work_contractor_data,work_scaned_data,work_type,work_status);
        work_items.push(temp);
        return 1;
    }
    //cout data
    function getworkitemsCount() public returns(uint)
    {
        return work_items.length;
    }
    //function to get work items
    function getWorkitemsData(uint id) public returns(uint,string memory,string[] memory,string [] memory,string memory,string memory)
    {
        Work_Items memory temp = work_items[id];
        return (temp.WI_Contract_ID,temp.Work_Name,temp.Work_Contractor_Data,temp.Work_Scaned_Data,temp.Work_Type,temp.Work_Status);
    }
```

```
    //function to set contract
    function setContractData(string memory Owneraddress,string memory contractoraddress,string memory consultedaddress,string[] memory workitems,
    {
        uint id = contracts.length + 1;
        Contract memory temp = Contract(id,contractname,Owneraddress,contractoraddress,consultedaddress,workitems,totalcost,finisheditems,contrac
        contracts.push(temp);
        return 1;
    }
    //cout data
    function getcontractCount() public returns(uint)
    {
        return contracts.length;
    }
    //function to get contract data
    function getContractData(uint contract_id) public returns(uint,string memory,string memory,string memory,string memory,string[] memory,string
    {
        Contract memory temp = contracts[contract_id];
        return (temp.Contract_ID,temp.Contract_Name,temp.Owner_Address,temp.Contractor_Address,temp.Consulted_Addresss,temp.Work_Items,temp.Total
    }



    function compareStringsbyBytes(string memory s1, string memory s2) public pure returns(bool)
    {
        return keccak256(abi.encodePacked(s1)) == keccak256(abi.encodePacked(s2));
    }
```

Figure A2. Solidity smart contract operation function codebase.

References

1. Durdyev, S.; Hosseini, M.R. Causes of delays on construction projects: A comprehensive list. *Int. J. Manag. Proj. Bus.* **2020**, *13*, 20–46. [CrossRef]
2. Chan, E.E.; Nik-Bakht, M.; Han, S.H. Sources of Ambiguity in Construction Contract Documents, Reflected by Litigation in Supreme Court Cases. *J. Leg. Aff. Disput. Resolut. Eng. Constr.* **2021**, *13*, 04521031. [CrossRef]
3. Lu, W.; Li, X.; Xue, F.; Zhao, R.; Wu, L.; Yeh, A.G.O. Exploring smart construction objects as blockchain oracles in construction supply chain management. *Autom. Constr.* **2021**, *129*, 103816. [CrossRef]
4. Casino, F.; Dasaklis, T.K.; Patsakis, C. A systematic literature review of blockchain-based applications: Current status, classification and open issues. *Telemat. Inform.* **2019**, *36*, 55–81. [CrossRef]
5. Hughes, L.; Dwivedi, Y.K.; Misra, S.K.; Rana, N.P.; Raghavan, V.; Akella, V. Blockchain research, practice and policy: Applications, benefits, limitations, emerging research themes and research agenda. *Int. J. Inf. Manag.* **2019**, *49*, 114–129. [CrossRef]
6. Hamledari, H.; Fischer, M. Construction payment automation using blockchain-enabled smart contracts and robotic reality capture technologies. *Autom. Constr.* **2021**, *132*, 103926. [CrossRef]
7. Biolek, V.; Hanak, T.; Hanak, M. Proposed Interconnecting Database for BIM Models and Construction-Economic Systems in the Czech Republic. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *471*, 112079. [CrossRef]
8. Heap-Yih, C.; Su-Ling, F.; Monty, S.; Shang-Hsien, H.; Ching-Mei, T. Preliminary Contractual Framework for BIM-Enabled Projects. *J. Constr. Eng. Manag.* **2017**, *143*, 04017025. [CrossRef]

9. De Gaetani, C. Scan-to-BIM Workflow: An Overview and Case Study. 2020. Available online: https://www.politesi.polimi.it/retrieve/90a1b203-1fb2-46af-99f0-0434508edfc3/BRUMATTI%20PINHO%20LETICIA%2010665567.pdf (accessed on 8 January 2024).
10. Wang, Q.; Guo, J.; Kim, M.K. An application oriented scan-to-BIM framework. *Remote Sens.* **2019**, *11*, 365. [CrossRef]
11. Bosché, F.; Ahmed, M.; Turkan, Y.; Haas, C.T.; Haas, R. The value of integrating Scan-to-BIM and Scan-vs-BIM techniques for construction monitoring using laser scanning and BIM: The case of cylindrical MEP components. *Autom. Constr.* **2015**, *49*, 201–213. [CrossRef]
12. Golparvar-Fard, M.; Peña-Mora, F.; Arboleda, C.; Lee, S. Visualization of Construction Progress Monitoring with 4D Simulation Model Overlaid on Time-Lapsed Photographs. *J. Comput. Civ. Eng.* **2009**, *23*, 391–404. [CrossRef]
13. Roh, S.; Aziz, Z.; Peña-Mora, F. An object-based 3D walk-through model for interior construction progress monitoring. *Autom. Constr.* **2011**, *20*, 66–75. [CrossRef]
14. Behnam, A.; Wickramasinghe, D.C.; Ghaffar, M.A.A.; Vu, T.T.; Tang, Y.H.; Isa, H.B.M. Automated progress monitoring system for linear infrastructure projects using satellite remote sensing. *Autom. Constr.* **2016**, *68*, 114–127. [CrossRef]
15. Skrzypczak, I.; Oleniacz, G.; Leśniak, A.; Zima, K.; Mrówczyńska, M.; Kazak, J.K. Scan-to-BIM method in construction: Assessment of the 3D buildings model accuracy in terms inventory measurements. *Build. Res. Inf.* **2022**, *50*, 859–880. [CrossRef]
16. Qiu, Q.; Zhou, X.; Zhao, J.; Yang, Y.; Tian, S.; Wang, J.; Liu, J.; Liu, H. From sketch BIM to design BIM: An element identification approach using Industry Foundation Classes and object recognition. *Build. Environ.* **2021**, *188*, 107423. [CrossRef]
17. Wang, W.; Chen, Y.; Zhang, S.; Wang, Y. Contractual Complexity in Construction Projects: Conceptualization, Operationalization, and Validation. *Proj. Manag. J.* **2018**, *49*, 46–61. [CrossRef]
18. Ibrahim, R.; Harby, A.A.; Nashwan, M.S.; Elhakeem, A. Financial Contract Administration in Construction via Cryptocurrency Blockchain and Smart Contract: A Proof of Concept. *Buildings* **2022**, *12*, 1072. [CrossRef]
19. Bitcoin, B.; Crosby, M.; Nachiappan, G.; Pradhan Pattanayak, Y.; Verma, Y.S.; Kalyanaraman, V.; Semiconductor, F. BlockChain Technology. 2015. Available online: https://scet.berkeley.edu/wp-content/uploads/BlockchainPaper-1.pdf (accessed on 8 January 2024).
20. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 8 January 2024).
21. Wood, D.D. Ethereum: A Secure Decentralised Generalised Transaction Ledger. 2014. Available online: https://api.semanticscholar.org/CorpusID:4836820 (accessed on 7 January 2024).
22. Lu, Y. The blockchain: State-of-the-art and research challenges. *J. Ind. Inf. Integr.* **2019**, *15*, 80–90. [CrossRef]
23. Penzes, B. *Blockchain Technology in the Construction Industry*; Institution of Civil Engineers: London, UK, 2018. [CrossRef]
24. Xu, X.; Lu, Q.; Liu, Y.; Zhu, L.; Yao, H.; Vasilakos, A.V. Designing blockchain-based applications a case study for imported product traceability. *Future Gener. Comput. Syst.* **2019**, *92*, 399–406. [CrossRef]
25. Macrinici, D.; Cartofeanu, C.; Gao, S. Smart contract applications within blockchain technology: A systematic mapping study. *Telemat. Inform.* **2018**, *35*, 2337–2354. [CrossRef]
26. Rouhani, S.; Deters, R. Security, performance, and applications of smart contracts: A systematic survey. *IEEE Access* **2019**, *7*, 50759–50779. [CrossRef]
27. Feng, T.; Yu, X.; Chai, Y.; Liu, Y. Smart contract model for complex reality transaction. *Int. J. Crowd Sci.* **2019**, *3*, 184–197. [CrossRef]
28. Cong, L.W.; He, Z. Blockchain Disruption and Smart Contracts. *Rev. Financ. Stud.* **2019**, *32*, 1754–1797. [CrossRef]
29. Turk, Ž.; Klinc, R. Potentials of Blockchain Technology for Construction Management. *Procedia Eng.* **2017**, *196*, 638–645. [CrossRef]
30. Mason, J.; Escott, H. Smart Contracts in Construction: Views and Perceptions of Stakeholders. 2018. Available online: https://uwe-repository.worktribe.com/output/868722 (accessed on 7 January 2024).
31. Gibbs, D.-J.; Emmitt, S.; Lord, W.; Ruikar, K. BIM and construction contracts—CPC 2013's approach, Proceedings of the Institution of Civil Engineers—Management. *Procure. Law* **2015**, *168*, 285–293. [CrossRef]
32. Ye, X.; König, M. Framework for Automated Billing in the Construction Industry Using BIM and Smart Contracts. In *Lecture Notes in Civil Engineering*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 824–838. [CrossRef]
33. Ahmadisheykhsarmast, S.; Sonmez, R. A smart contract system for security of payment of construction contracts. *Autom. Constr.* **2020**, *120*, 103401. [CrossRef]
34. Elghaish, F.; Abrishami, S.; Hosseini, M.R. Integrated project delivery with blockchain: An automated financial system. *Autom. Constr.* **2020**, *114*, 103182. [CrossRef]
35. Ciotta, V.; Mariniello, G.; Asprone, D.; Botta, A.; Manfredi, G. Integration of blockchains and smart contracts into construction information flows: Proof-of-concept. *Autom. Constr.* **2021**, *132*, 103925. [CrossRef]
36. Sonmez, R.; Ahmadisheykhsarmast, S.; Güngör, A.A. BIM integrated smart contract for construction project progress payment administration. *Autom. Constr.* **2022**, *139*, 104294. [CrossRef]
37. Dakhli, Z.; Lafhaj, Z.; Mossman, A. The potential of blockchain in building construction. *Buildings* **2019**, *9*, 77. [CrossRef]

38. Wefki, H.; Salah, M.; Elsharkawi, H.M. Deciphering the Nexus: Blockchain-Smart Contracts and Their Transformative Potential in the Construction Industry. *J. Eng. Res.* **2023**, *7*, 19–30. [CrossRef]
39. Darabseh, M.; Martins, J. Risks and Opportunities for Reforming Construction with Blockchain: Bibliometric Study. *Civ. Eng. J.* **2020**, *6*, 1204–1217. [CrossRef]
40. Hamledari, H.; Asce, S.M.; Fischer, M.; Asce, A.M. Role of Blockchain-Enabled Smart Contracts in Automating Construction Progress Payments. *J. Leg. Aff. Disput. Resolut. Eng. Constr.* **2020**, *13*, 04520038. [CrossRef]
41. Bageis, A.S. An Investigation into the Causes of Payment Delays and Deliberate Delay Tactics in Public Construction Projects in Saudi Arabia. *Buildings* **2024**, *14*, 1792. [CrossRef]
42. Wahab, A.; Wang, J.; Shojaei, A.; Ma, J. A model-based smart contracts system via blockchain technology to reduce delays and conflicts in construction management processes. *Eng. Constr. Archit. Manag.* **2023**, *30*, 5052–5072. [CrossRef]
43. Davis, P.; Ellis, J.; Cheung, S. Dispute causation: Identification of pathogenic influences in construction. *Eng. Constr. Archit. Manag.* **2010**, *17*, 404–423. [CrossRef]
44. Fenn, P.; Lowe, D.; Speck, C. Conflict and dispute in construction. *Constr. Manag. Econ.* **1997**, *15*, 513–518. [CrossRef]
45. Mahmudnia, D.; Arashpour, M.; Yang, R. Blockchain in construction management: Applications, advantages and limitations. *Autom. Constr.* **2022**, *140*, 104379. [CrossRef]
46. Dai, F.; Rashidi, A.; Brilakis, I.; Vela, P. Comparison of Image-Based and Time-of-Flight-Based Technologies for 3D Reconstruction of Infrastructure. *J. Constr. Eng. Manag.* **2013**, *139*, 929–939. [CrossRef]
47. Corso, J.; Marco, A.; Casals, J.; Garcia-Almirall, P. SCAN to BIM Beyond a Final BIM: Why, When and How. *IOP Conf. Ser. Mater. Sci. Eng.* **2019**, *603*, 042090. [CrossRef]
48. Yunos, S.; Baynes, B. Refining the Scan to BIM Workflow for Further Automation and Visualisation. 2020. Available online: https://www.autodesk.com/autodesk-university/article/Refining-Scan-BIM-Workflow-Further-Automation-and-Visualization-2021 (accessed on 6 December 2023).
49. Alshawabkeh, Y.; Baik, A. Integration of photogrammetry and laser scanning for enhancing scan-to-HBIM modeling of Al Ula heritage site. *Herit. Sci.* **2023**, *11*, 147. [CrossRef]
50. Yucel, B.; Liu, J.; Alathamneh, S. A Comparative Analysis of Point Clouds Acquired from UAV Photogrammetry and UAV-based LiDAR in Built Environment. In Proceedings of the 60th Annual Associated Schools of Construction International Conference, Auburn, AL, USA, 3–5 April 2024; pp. 584–592. [CrossRef]
51. Khan, S.N.; Loukil, F.; Ghedira-Guegan, C.; Benkhelifa, E.; Bani-Hani, A. Blockchain smart contracts: Applications, challenges, and future trends. *Peer Peer Netw. Appl.* **2021**, *14*, 2901–2925. [CrossRef] [PubMed]
52. Buterin, V. A Next Generation Smart Contract & Decentralized Application Platform. 2015. Available online: https://api.semanticscholar.org/CorpusID:19568665 (accessed on 12 December 2023).
53. Androulaki, E.; Barger, A.; Bortnikov, V.; Muralidharan, S.; Cachin, C.; Christidis, K.; De Caro, A.; Enyeart, D.; Murthy, C.; Ferris, C.; et al. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. In Proceedings of the 13th EuroSys Conference, EuroSys 2018, Porto, Portugal, 23–26 April 2018; Association for Computing Machinery, Inc.: New York, NY, USA, 2018. [CrossRef]
54. Shults, R.; Levin, E.; Aukazhiyeva, Z.; Pavelka, K.; Kulichenko, N.; Kalabaev, N.; Sagyndyk, M.; Akhmetova, N. A Study of the Accuracy of a 3D Indoor Camera for Industrial Archaeology Applications. *Heritage* **2023**, *6*, 6240–6267. [CrossRef]