From automation to agency
Prototype for self-owning intelligent buildings enabled by blockchain

Wang, Hongyang; Hunhevicz, Jens J.; Hall, Daniel M.

DOI
10.1016/j.autcon.2025.106309

Publication date
2025

Document Version
Final published version

Published in
Automation in Construction

Automation in Construction 177 (2025) 106309

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

From automation to agency: Prototype for self-owning intelligent buildings enabled by blockchain☆

Hongyang Wang [a],*, Jens J. Hunhevicz [a,b], Daniel M. Hall [c]

[a] ETH Zurich, Institute of Construction and Infrastructure Management, Zurich, Switzerland
[b] Empa, Urban Energy Systems Lab, Dübendorf, Switzerland
[c] TU Delft, Department of Management in the Built Environment, Delft, Netherlands

ARTICLE INFO

Dataset link: https://github.com/no1s1labs/no1s1

Keywords:
Blockchain technology
Smart contracts
Cyber-physical systems
Artificial agent
Human-building interactions
Intelligent buildings
Self-owning buildings

ABSTRACT

While most research in human-building-interaction looks at the interaction between humans and building automation, few studies question the agency of the building itself. This paper explores how blockchain technology can be combined with intelligent buildings to achieve self-ownership and self-agency. Using a design science research approach, a blockchain-based smart meditation cabin, the "no1s1" prototype, is iteratively designed, tested and evaluated. no1s1 demonstrates that a building can autonomously manage access, finances, and operation with minimal human oversight. These findings suggest that blockchain can redefine technical system design by embedding ownership and agency into the building itself. The findings encourage further exploration into decentralized coordination mechanisms within intelligent environments, such as combining blockchain with artificial intelligence and advanced sensing environments, to rethink the coordination, ownership and agency of cyber–physical systems in the built environment.

1. Introduction

The introduction of new building technologies, such as the Internet of Things (IoT), digital twins, predictive maintenance, and building automation, has changed our understanding of buildings. Buildings have evolved from simple physical structures to ubiquitous living environments of dynamic cyber–physical-social systems [1,2]. The activities of both the building and the human occupants can be recorded and analyzed to simulate and optimize processes. This phenomenon has been described by scholars using terms such as "smart buildings" [3,4], "intelligent buildings" [5,6], and "autonomous buildings" [7]. Recently, the field has converged under the new terminology of "Human Building Interaction" (HBI) [8,9]. HBI scholars believe that as a building system becomes more sophisticated and interactive, the built environment also has a strong impact on human behavior and well-being [9].

This increasing intelligence has led scholars to emphasize the independence and autonomy of buildings. As technical systems enable more responsive real-time behavior of buildings [10,11], novel complex system characteristics emerge, such as sophisticated feedback loops that are similar to those observed in nature [12–14]. Therefore, scholars have designed buildings to achieve self-operation and management through real-time status detection and system response, such as self-sustaining drainage and energy systems for buildings [15,16]. Another example includes building skins and structures that can "self-heal" with real-time detection of the material behavior and corresponding predictive maintenance algorithms for repairs [17,18]. Scholars have even explored the emerging "awareness" of the cyber–physical systems in the built environment [19,20].

However, most of the existing research focuses mainly on technical systems, such as building operations automation, smart device integration, smart building data management, and cost-effective decision-making algorithms [21–24]. Research has not yet addressed how an intelligent built environment can have impacts beyond the operation and management of buildings; it is likely that there are also significant psychological, social, and economic implications [25]. There is a gap in understanding how intelligent buildings interact with their own self-agency. Traditionally, it is assumed that buildings must be owned by an external actor (be it a person or an organization) who is responsible for them. If an intelligent building could achieve self-sovereignty, it could overturn existing long-held assumptions about human, social, and economic systems.

Therefore, this paper investigates how buildings could not only be smart and intelligent, but how they might be conceptualized as sovereign, autonomous and self-owning agents. The main enabler is blockchain technology. Blockchain allows to encode the agency of

☆ To be noted, in this study, intelligent building serves as the overarching term that incorporates smart, intelligent, and autonomous buildings.

* Corresponding author.
E-mail address: wang@ibi.baug.ethz.ch (H. Wang).

https://doi.org/10.1016/j.autcon.2025.106309
Received 20 September 2024; Received in revised form 27 May 2025; Accepted 27 May 2025
Available online 20 June 2025

buildings by bootstrapping their operational and financial logic through so-called smart contracts. The resulting building operating system offers the possibility of replacing traditional economic intermediaries, such as landlords, who often control financial transactions to date, with a code-based, logic-centered system embedded in the intelligent building system itself. The resulting "artificial agent" [26] has many implications for economic systems, such as the ability to code the blockchain to be non-rent seeking.

To demonstrate the potential of this new system of value and ownership, this paper details the design and implementation of the prototype "no1s1" (pronounced "no-one's one"), the first self-owned house on the blockchain. no1s1 is designed to be a solar-powered meditation cabin that can operate autonomously, rent itself out to the users, and own its rental income on the blockchain. Our development of no1s1 follows the Design Science Research (DSR) methodology, chosen for its systematic approach, broad acceptance, and emphasis on artifact creation and evaluation. The development of no1s1 follows the sequential steps of DSR, including concept development, prototype system design, development, iteration, and evaluation.

To describe these steps, the paper proceeds as follows. First, the paper introduces necessary background on blockchain technology and its applications in the built environment, the concepts of smart and intelligent buildings, and engineered ownership in Section 2. In Section 3, the paper describes with more specific details the applied DSR methodology. Sections 4 through 6 then describe how the necessary functionality and technical system was designed and developed, and how the system was iteratively tested and demonstrated as an operational real-world prototype. Section 7 evaluates the design artifact no1s1 based on the results of the operation during the demonstration and the subsequent user feedback, while Section 8 discusses both implications and limitations related to the developed prototype, with the focus on topics relating to the future of automation in smart and intelligent buildings and potential ways to technically integrate the socio-economic aspects within these systems.

2. Background and conceptualization

2.1. Blockchain and the built environment

Blockchain uses cryptography and robust consensus algorithms in a peer-to-peer network to reach consensus and validate transactions, rather than using traditional intermediaries to store and validate transaction data [27,28]. The resulting transparent and immutable transactions, along with the pseudonymity of users identified only by their blockchain addresses, allow for more decentralized alternatives to securely facilitate transactions. The first blockchain application Bitcoin [29] demonstrated this as a digital currency for monetary systems. With the subsequent development of automatically executable and programmable digital scripts called smart contracts on the second largest blockchain network, Ethereum [30], blockchain enables customization of interaction logic between transactions and data in blockchain networks [31].

Because blockchain provides a codable and engineerable digital representation of anything of value [32–34], which can be exchanged in a peer-to-peer network, it holds the potential to profoundly transform economic systems [35,36]. The digital representation of value can be utilized and applied in many different areas, including digital currency for monetized value [29,37], non-fungible tokens (NFTs) for the digital art economy [38], the voting tokens for the governance of new digital organizations [39,40], or digital identity and certification for reputational value [33,41,42]. The rapid growth of Decentralized Autonomous Organizations (DAOs) is an example of ongoing experiments in organizing and governing blockchain-based online communities [43–45].

Alongside these real-world experiments, various scholars are further discussing blockchain as an institutional technology that could facilitate the establishment of new economic coordination and governance systems [46,47]. For example, the Finance 4.0 framework explores new incentive mechanisms for sustainability and more positive social-ecological outcomes [48,49]. Legal scholars Filippi and Hassan leverage blockchain as a regulatory technology with the potential to substitute traditional legal institutions [35]. The transparent nature of blockchain networks allows digital data or assets to be managed as a common resource, facilitating the digitization and scaling of the global commons [50,51]. Overall, blockchain technology has the potential to revolutionize digital governance by re-engineering the social and economic design of digital value [32,33,52].

In the built environment, however, blockchain technology is not often seen as a means to transform or reimagine economic systems. Instead, blockchain is mainly being explored as an efficiency-enhancing technology for existing systems and processes [53–58]. For example, researchers use blockchain technology to improve supply chain management by providing transparency, traceability, and immutability, allowing stakeholders to track the movement of goods, verify authenticity, and reduce fraud [59]. Other studies focus on improving material traceability to promote accountability and quality control throughout the material life cycle, supporting sustainable development [60]. Blockchain has also been utilized to improve privacy [61], streamline process automation [54,62], improve promotion strategy for building information modeling [58], facilitate payment execution [53,63,64], streamline the building permit trading process [55] and enhance information and communication between devices [65]. A notable trend in blockchain research is its integration with artificial intelligence, machine learning, or digital twin technologies [56,66,67]. Studies explored using blockchain technology to improve data ownership in digital twins [56], enhance construction AI data security [66], and add additional incentivization layer to promote digital-twin-based training.

While the above studies contribute significantly to blockchain research in the built environment, many of the practical applications often overlook the full potential of blockchain technology to reshape governance and value systems. Some recent research has begun to explore a more radical vision of blockchain technology to establish new social and economic coordination in the built environment [68–72]. One of the closest examples is the Block Foundation, which is using Solidity smart contracts to develop a blockchain-based land inventory and management system [73]. Land ownership and managerial coordination are essential for the implementation of self-owning buildings. However, existing research in AEC largely concentrates on data management, rarely focus cyber–physical systems or integration with non-human agents.

This research identifies a unique opportunity in blockchain technology to enable self-owning entities, such as smart buildings, in ways that traditional databases cannot [42,74,75]. As previously noted, blockchain technology uniquely combines immutability, autonomous value transfer, and trustless coordination. Unlike distributed databases or permissioned ledgers, public blockchains facilitate genuine autonomous ownership and financial asset transfer by non-human entities [27,76]. Distributed databases typically rely on centralized or trusted authorities for updates and lack a native asset and value transfer layer [74], whereas permissioned ledgers are governed by pre-approved actors, introducing higher custodial risk [77,78]. For the purpose of this research, smart contracts enable the distribution, accumulation, and usage of financial assets from a single blockchain address, owned and controlled by algorithms and their underlying system. To the best of the authors' knowledge, only blockchain technology, especially the permissionless blockchain networks, currently enables non-human agents to autonomously exchange value with human actors on an agent-to-agent level over a peer-to-peer network, without centralized human intermediaries.

2.2. Smart and intelligent buildings

With the growth of information and communication technologies, the concept of intelligence, or "smartness" in buildings has gained popularity [4,11,23,79]. The terms smart buildings and intelligent buildings are often used interchangeably in research [4,80]. Some researchers view smart systems as a subset of intelligent built environments [13,81], while others view smart buildings as a more advanced concept that is more adaptive and responsive than intelligent buildings [4].

While there is no universally accepted definition, several key concepts are generally accepted in the field [79,80]. First, buildings of the future will become more interactive, fostering greater engagement and interaction between occupants and the built environment [6]. Second, buildings can be adaptive, with the ability to respond dynamically to changes in occupancy, environmental conditions, and user preferences [81]. Third, building systems can be interconnected and networked, allowing seamless communication and integration between different components [10]. Finally, the control system should strike a balance between user control and automation, providing a flexible and sophisticated technical structure that accommodates individual preferences while taking advantage of automation [82]. Overall, the development of this research has run in parallel to a large body of research on building efficiency, automation and performance [22,83,84].

A key assumption found in past intelligent building literature is that buildings are mostly viewed as an adaptive system that satisfies human needs [5]. Under this research paradigm, humans are viewed more as *service receivers* from the building rather than *collaborators* or *interactors* [10] with the building. The adaptability of an intelligent building is achieved by implementing sensing devices so that the building system can adapt to changing human needs [23]. However, with the rise of Artificial Intelligence (AI), recent research is exploring human building interactions with intelligent buildings as collaborators [6,8,9]. Concepts such as the "cognitive city" view humans as part of the digital infrastructure.

When people and buildings work collaboratively, there is a need to create a suited digital governance system. Digital governance is the coordinating mechanism that manages the complexity of human-building-environment interaction. Scholars have called for more studies on governance in all phases of the building life cycle [79]. Research into a new mode of digital governance has been defined as one of the most important research directions to move the field forward, but it has been understudied at both the technical implementation and conceptual levels. Early work suggests that blockchain-based governance may be well suited to building such new systems in the built environment [70]. As digital governance and artificial intelligence advance, the future of governance increasingly points to collaboration between humans and non-human entities. While human control of the built environment is well-studied, investigating non-human control in smart built environments is crucial for shaping future human-machine governance.

2.3. Engineered ownership

Some scholars see blockchain technology as a means to govern and coordinate networks composed of both human and autonomous machine agents [12,46,85]. Taking it a step further, blockchain technology can potentially radically change the ownership system for human and machine agents. For example, artificial agents can interact with human agents on the blockchain network without revealing their identities [86]. This opens up the possibility of autonomous, self-executing artificial agents that retain sovereign ownership and control over their own digital identities and digital assets [42].

Subsequently, several blockchain-based projects offer a future vision of human-machine coordination with a radical ownership system. McConaghy coined the term "Nature 2.0" to envision an autonomous digital infrastructure network, empowered by blockchain technology and AI, that can be self-owned on the blockchain [12]. In this paradigm, the built environment becomes a digital version of nature with which humans can interact and live in symbiosis. There are also other studies of blockchain-based machine sovereignty. For example, researchers are proposing future logic-centric societies in which machines replace most of the work, leaving humans on the edge [87]. Other work implements blockchain-based life forms such as the *Plantoid* [88]. There have also been developments in blockchain frameworks to empower nature's self-agency, such as the Sovereign Nature Initiative [89] and terra0 [90].

Even without such futuristic visions, the automation in machines often leads to a decrease in human control and ownership over its actions and decision-making. However, for researchers in the built environment, automation has primarily been associated with increased efficiency, rather than a loss of ownership and control for human actors. This can be seen in the process of standardization and formalization of the level of automation.

The most widely accepted categorization for the level of automation by the U.S. federal government and the industry at large is for autonomous vehicles [91,92]. The SAE Levels of Driving Automation range from the lowest level 0 to the highest level 5 [93]. Level 0 vehicles require full human control at all times, while level 5 vehicles are capable of operating independently in all situations. An example in the built environment is the level of automation for the digital twin (DT) [94]. DT technology aims to create a virtual replica of a physical building or system, enabling real-time monitoring, analysis and optimization [95,96]. Researchers are increasingly aware of the importance of discussing the role of humans in DT automation [94,97]. By assigning roles to digital twins, such as analysts and decision makers, it systematically demonstrates the DT's ability to cause action. However, the lack of granularity in these roles undermine the ownership complexities inherent in human socio-economic coordination, limiting their broader applicability and impact.

The term "Engineered ownership" was created to address the above gaps [98]. Viewed from a systems engineering perspective [99,100], ownership is a complex system with interconnected and nested structures that intertwine with each other. The concept of ownership fundamentally relies on the provenance and acceptance of certain informational assertions; it can therefore be managed and engineered [101,102]. Therefore, the idea of "engineered ownership" on the blockchain refers to two potential levels of application: (1) the ability to enable a peer-to-peer network of autonomous human and machine agents, and (2) the ability to engineer the digital ownership for networked agents [98]. An engineered ownership approach provides a promising framework for understanding and facilitating human-machine agent interactions for the future development of intelligent agents.

Research on ownership systems for automation of machine agents in the built environment often remains either theoretical or abstract, mostly focusing on broad societal or citywide scales [25,101,103]. Applied research in this area tends to be limited to economic evaluations of construction processes or infrastructure planning, overlooking the deeper implications of ownership systems [104]. While some researchers have explored the intersection of ownership and the built environment, their focus is often on time-stamping or fragmented representations of ownership, primarily to facilitate storage and transfer [105]. Although blockchain technology fundamentally offers the opportunity to re-engineer ownership and value, it is rarely explored as an enabler of new human-machine coordination. Many of the previous studies remain conceptual and often lack practical implementation or integration with cyber–physical artifacts [106]. This absence of concrete examples limits our understanding of how self-owning buildings can shape and advance the future of smart and intelligent buildings and their communities. This shift, fundamentally enabled by blockchain technology, has the potential to transform the digital governance ecosystem for smart and intelligent buildings, establishing a foundation for studies on technology-based ownership and coordination.

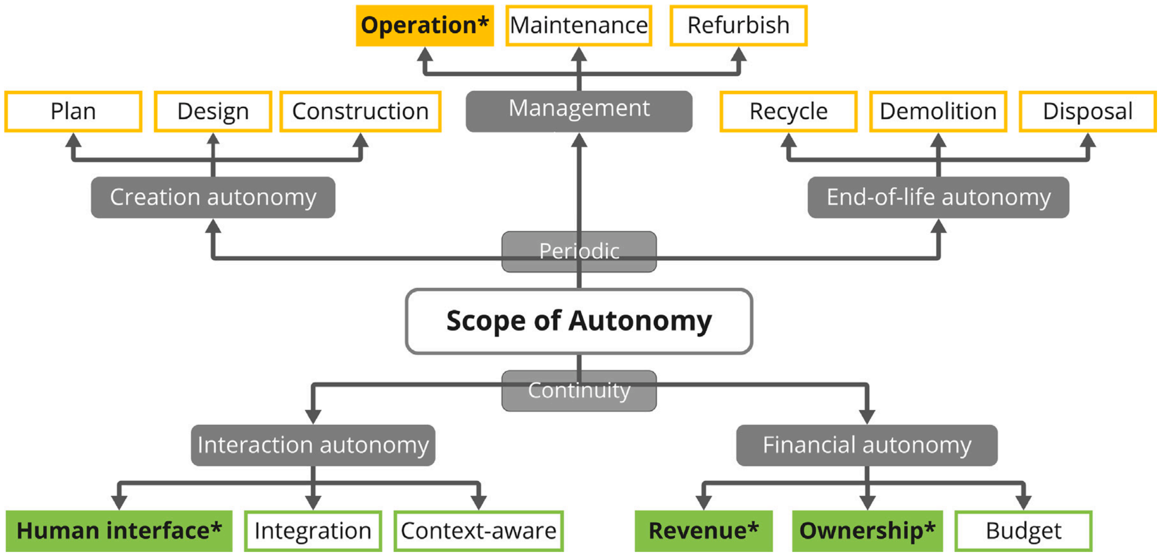

Fig. 1. Conceptualization of the scope of autonomy. The final selected autonomy aspects implemented in the no1s1 prototype are marked with *.

2.4. Conceptualization: Scope of autonomy

Building upon the above three sections, Fig. 1 clusters five distinct areas of autonomy that can be conceptualized for smart and intelligent buildings: creation autonomy, management autonomy, end-of-life autonomy, interaction autonomy, and financial autonomy [11,79,107]. Each cluster includes a range of specific functionalities that contribute to the overall intelligence and efficiency of the building. The first three clusters are periodic and occur at a specific time frame of a building lifecycle. The last two clusters represent continuous functions throughout the lifecycle of intelligent buildings.

- The *creation autonomy* category denotes the capacity for an intelligent building to self-create. This includes the ability to oversee and manage its own planning, design and construction activities. Planning autonomy refers to the ability to define the scope, concept and outline of the creation process itself. Design autonomy establishes guidelines to self-solicit design proposals and self-select a final design. Construction autonomy enables the intelligent building to request, authorize, execute, and supervise construction activities.
- The *management autonomy* category pertains to the capacity for an intelligent building to self-manage during facility operations. Operational autonomy relates to the proactive or reactive interaction between humans and technical systems (e.g., control systems, sensors, and smart devices). Maintenance autonomy ensures longevity of operations through fault detection, sensor feedback, or predictive maintenance data feedback. These serve to enhance system efficiency and maximize the system uptime.
- The *end-of-life autonomy* category refers to the ability of intelligent buildings to self-recycle, self-destruct, and self-dispose upon reaching the conclusion of it's lifespan. In the pursuit of a holistic approach to autonomy, it is essential to consider the entire lifecycle of intelligent building structures, including their eventual decommissioning. This consideration and shift of mindset from linear to circular symbolizes the forthcoming sustainable development in the built environment.
- The *interaction autonomy* category describes the communication capability between the intelligent building and outside actors or the environment. For instance, interaction with human actors traditionally occurs through an interface such as a screen. More immersive and advanced interfaces such as augmented reality, virtual reality, or extended reality allow for a more engaging and intuitive experience. Further, understanding the surrounding environment forms the basis of context awareness using popular methods such as computer vision. It also entails machine-to-machine interaction and the integration of their respective cyber systems. This also implies the possibility for mobility in the space. When certain interaction is needed, the space location and transportation can be re-arranged by itself.
- The *financial autonomy* category refers to mechanisms that enable the intelligent building to acquire, store, manage and spend its own funds. Most important is ownership autonomy, which necessitates a digital treasury managed by an artificial physical agent capable of storing funds; without such a treasury, self-ownership is not possible. Other functionalities such as revenue and budget autonomy are often integrated with the other autonomy categories. For example, revenue autonomy such as fundraising and budget autonomy such as spending within a set budget could be part of the creation phase, or for the refurbishment phase.

2.5. Point of departure

Blockchain technology has great potential to transform digital governance, but its applications in the built environment often focus on increasing efficiency rather than fully realizing its potential to reimagine governance and associated systems of ownership. The emergence of smart and intelligent buildings underscores the need for effective digital governance to address the complex interactions between the human and cyber–physical integrated built environment. Recent radical research envisions blockchain technology not only as a tool to govern interactions between human and autonomous machine agents, but also as a means to radically transform ownership from a social concept to engineer-able system. Autonomous agents could control their digital identities and assets, highlighting the need for further research into the emerging complexities of automation and ownership in the built environment.

Overall, to the best of our knowledge, no research has yet effectively demonstrated the potential of radical ownership transformation in intelligent buildings. The idea of a self-owning building remains highly conceptual and abstract, with no prototyping yet achieved. Therefore, there is a need to design, develop, and iterate a technical artifact that can demonstrate the possibility of (1) intelligent buildings acting as an integrated cyber–physical machine agent, (2) these agents possessing a level of operational autonomy, and (3) these agents possessing a level of self-ownership in managing their own financial assets and resources.

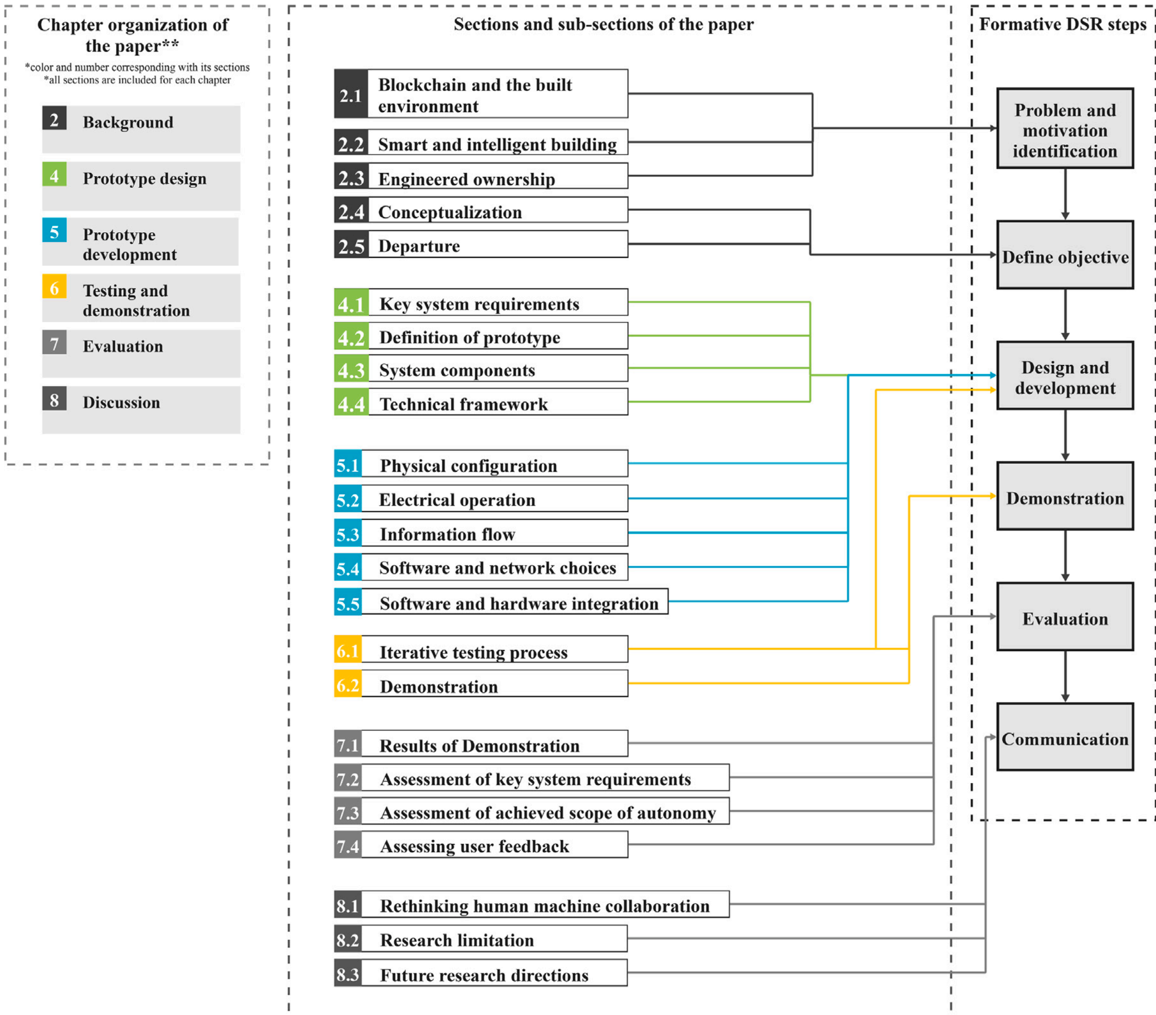

Fig. 2. Mapping the paper chapters and sections to the formative steps of DSR methodology.

3. Methodology

This paper uses the Design Science Research (DSR) methodology to design, develop, test, and demonstrate the above conceptualization. DSR is grounded in the scientific principle of creating innovative artifacts that address real-world problems and offer prescriptive scientific solutions [108,109]. This methodology aligns with this paper due to the futuristic nature of the concept, allowing for the formulation, implementation, and testing of a specific artifact. The artifact can be used to demonstrate technical feasibility, prompt user feedback and dialogue, and challenge present assumptions, in this case assumptions about the relationship between automation and ownership in the built environment.

The artifact for this DSR methodology is named no1s1, a self-owning meditation cabin designed to illustrate the conceptualization described above. Fig. 2 shows an overview of the process steps and associated sections in this paper used to apply the DSR methodology [110] for the no1s1 artifact. Below we elaborate further in Section 3.1 the steps of *problem and motivation identification* and *define objectives*, in Section 3.2 the step of *design and development*, and finally in Section 3.3 the steps of *demonstration*, *evaluation* and *communication*.

3.1. Problem, motivation and objective

The background and conceptualization as described in Section 2 establish the motivation and scope of this paper (see Fig. 2, black boxes). First, the researchers reviewed the existing literature to identify research gaps and situate the paper within the established academic context on the topics of blockchain and the built environment (Section 2.1), smart and intelligent buildings (Section 2.2), and engineered ownership (Section 2.3). Section 2.4 presents a conceptualization of the potential autonomy scope for intelligent buildings, synthesized from findings in Sections 2.1 to 2.3, alongside existing categorizations [11,79,80,107,111]. This is illustrated in Fig. 1 and guided the artifact's scope within the DSR research approach.

3.2. Design and development

Based on the established motivation and scope, Section 4 outlines the design of the prototype system (see Fig. 2, green boxes). In Section 4.1, a list of key system requirements has been established to guide the design of the prototype. The key requirements integrate the functional and nonfunctional requirements derived from (1) cyber–physical systems, emphasizing integrated operational functionality; (2) blockchain technology, highlighting the data requirements for machine self-ownership; (3) prototype users, underscoring an intuitive interface and seamless interaction. Subsequently, the use case of the artifact no1s1 was defined in Section 4.2 as a self-owning meditation cabin, with autonomous operation and renting experience for the user, allowing them to access and pay for their meditation usage. This choice reflects both the key system requirements and the chosen scope of autonomy (see Table 1 and Fig. 1). Next, the necessary system components for the prototype were defined in Section 4.3 and Fig. 3, referencing proposed architectures for cyber–physical systems of other work [112–115]. This provided an overview of the necessary system parts for both the physical and cyber environments. A technical framework was then created in Section 4.4 and Fig. 4 to illustrate and explain in more detail the interaction of the technical parts within the chosen system to guide the implementation of no1s1.

In the second part of the design and development phase, Section 5 describes the actual development of the prototype (see Fig. 2, blue boxes). Guided by the design phase, the development process unfolds along three main topics always with consideration of the necessary user experience. First, the *physical configuration and structure* describes the construction of the structure and spacial arrangement (see Section 5.1). Section 5.2 describes then the implementation of the electrical operation and power supply system of no1s1. Section 5.3 explains the necessary feedback loops and associated information flow and process logic to enable users to meditate in no1s1. Finally, Section 5.4 and Table 2 show the applied rationale of technology choice for the identified components and Section 5.5 explains the final integration of the software and hardware components with focus on the smart contract design and implementation crucial to facilitate the functioning of the prototype.

3.3. Testing, demonstration and evaluation

Section 6 describes the testing, demonstration, and evaluation of no1s1, with the aim to validate its operation, functionality, usability, and performance (see Fig. 2, yellow and light gray boxes).

The iterative testing process (see in Section 6.1 and Fig. 10) emerged from the system and technical design and allows for continuous refinement throughout different phases, enhancing the system's reliability and performance, including unit testing (see Section 6.1.1), integration testing (see Section 6.1.2), system testing (see Section 6.1.3), and user acceptance testing (see Section 6.1.4). Three user acceptance testings were carried out with various practitioners, researchers and users in different settings over the course of one year (see Section 6.1.5).

With the results of the above test phases, the final demonstration was performed (see Section 6.2). The demonstration was conducted with 31 participants over the course of three days with a designed onboarding session, testing session, and feedback session. The transaction data collected verifies the successful operation of no1s1 (see Section 7.1). The mapping of the achieved scope of autonomy proved that the prototype system design was according to key system requirements (see Section 7.2 and Section 7.3). The user survey was designed to provide insight into not only the usage of the prototype operation but also the future potential impact of the concept (see Section 7.4).

4. Prototype design

This section describes the key steps and considerations made during the design of the no1s1 artifact.

4.1. Key system requirements

To achieve the conceptualization, the researchers derived eight key system requirements to guide the prototype design and development:

1. **Blockchain Integration**. Utilize blockchain technology as the source of truth. The blockchain should record and store essential prototype status. It should also be used as input during the operation of the system and provide evidence to ensure operational security. Recorded data should include, but not be limited to, user transaction data and operational data.
2. **Comprehensive Metadata**. Include critical metadata (e.g., timestamp, user ID, transaction type, amount, and status) in all on-chain records to facilitate information verification and traceability. This will enable the prototype's historical operating data to appear on-chain.
3. **Self-agency**. Encode an operational treasury which can be accessed by the prototype. This treasury should be inaccessible to all human stakeholders. Ensure that automated financial operational processes are encoded and secured by on-chain smart contracts that require minimal human intervention.
4. **Financial Transactions**. Record financial transactions on the blockchain, utilizing native blockchain-based tokens to ensure immutability and security in economic interactions.
5. **Access Condition**. Provide pseudonymous access for users, ensuring broad accessibility while maintaining privacy and economic security.
6. **User Interaction**. Enable essential user-prototype interactions with a user-friendly interface for seamless identity verification and transaction status via smart contracts, ensuring that all recorded verifications and statuses are kept on-chain as much as possible.
7. **Physical Scale**. Design the physical prototype to accommodate human-scale testing by individuals, enabling real-world interactions and validating the prototype's functionality
8. **Functionality**. Minimize functionality to provide users with a cohesive experience that embodies the essence of space self-agency, thereby simplifying the complexity of the initial prototype. Functionality includes not only automated operational processes but also automated financial processes.

4.2. Definition of prototype use-case

To satisfy the key system requirements, the no1s1 prototype is designed to provide a simple space-rental experience. The primary use case selected is a meditation cabin. Users engage with the space by making a small cryptocurrency payment to the prototype. This design allows for a functional single-user prototype of feasible size and complexity to develop and test the concept. This focused approach helped develop the necessary core functionality, streamline implementation, and ensure a consistent user experience.

Within the above context, the no1s1 prototype focuses on three key autonomy functionalities:

- Management autonomy is demonstrated through partial *operation* of the meditation space,
- Interaction autonomy is demonstrated by the *human interface* that allows users to interact with the cabin for meditation,
- Financial autonomy is demonstrated by receiving *revenue* through selling meditation time, and *ownership* by transferring this revenue to the treasury.

These autonomy functionalities meet the eight key system requirements as outlined in Table 1.

4.3. Main system elements

The prototype consists of five main system elements: software, hardware, logic, information management, and physical configuration (see Fig. 3).

The *software* includes the programming languages, libraries, frameworks, packages, blockchain networks, and environment for both the front-end and back-end of the prototype. The software must enable the management and control of diverse building systems such as lighting, sound, security, and energy management. It should offer the possibility to update real-time data to further enable data-driven decisions. Importantly, blockchain networks should facilitate secure financial transactions and the storage of the prototype state. Additionally,

Table 1
Mapping of selected autonomy functionalities (see Fig. 1) to prototype system requirements (see Section 4.1).

System key requirements	Operation	Human interface	Revenue	Ownership
Blockchain integration	X	X	X	X
Comprehensive metadata	X	X	X	X
Self-agency	X	X	X	X
Financial transaction			X	X
Access condition	X	X	X	X
User interaction	X	X	X	X
Physical scale	X			
Functionality	X	X	X	X

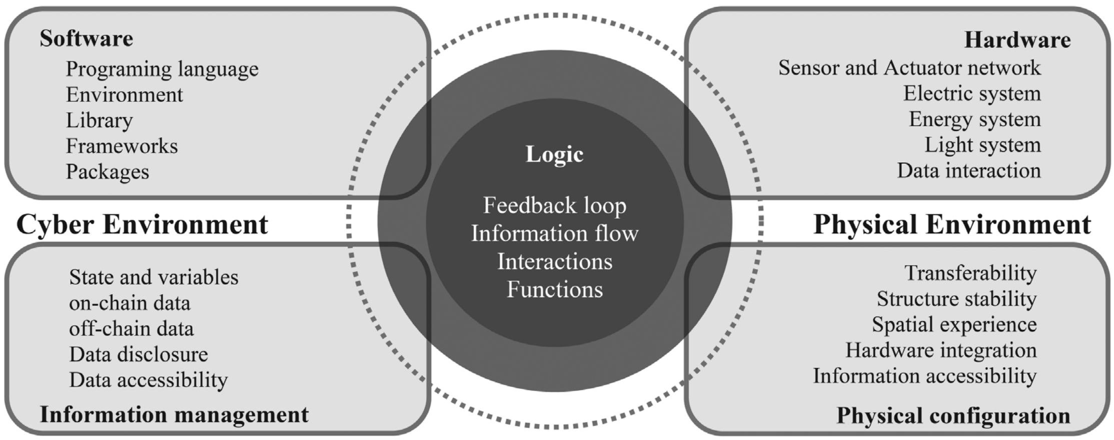

Fig. 3. Necessary elements bridging the cyber and physical environment.

smart contracts on the blockchain should allow coded interactions and transactions between humans and the prototype.

The *hardware* refers to the electronics integrated with the software components. For example, different IoT devices, like thermostats, security cameras, remote sensors, buttons, or LED lights, can be deployed to control and automate the operation of the prototype systems. These sensors and actuators act as the "muscle", "nerve" and "senses" of the prototype, consistently capturing and transmitting live data and initiating real-time responses.

The system operational *logic* determines the mode of interaction between the cyber environment, physical environment, and human actors. The rule-based logic defines how the prototype responds to various conditions and user actions and preferences based on real-time data captured by software and hardware components. For instance, the entry logic dictates the conditions for user access to the space.

The *information management* of the prototype system addresses questions such as what data is collected, where it is stored, and how accessible certain data is to specific actors. Decisions must be made to serve the objectives of the prototype, the request of the prototype system and the cohesion of operations.

The *physical configuration* integrates the hardware components and provides an interface between human users and between other components. The strategic placement of IoT devices and sensors is essential to the spatial design. These devices need to be positioned throughout the building to capture relevant data points, ensuring optimal coverage for data collection and control, and basic security for the system and user interaction. The design of the physical space also indicates stability, security, modularity, mobility and even scalability for future expansion.

While each element is discussed independently, significant interactions exist between the components. Hence it is essential to create iterative and systematic testing processes (see Section 6).

4.4. Technical framework

The technical framework that describes the relationship of these main system elements is shown in Fig. 4.

- Front-end component: a web-based user interface. This interface facilitates user interaction with the prototype system, through which human actors obtain on-chain and off-chain information and interact with the physical component. Users can make deposit payments, input pseudonymized identity, and specify prototype usage duration through the front-end to gain access to the physical space. It also displays information and facilitating system maintenance.
- Actor component: human actors who perform actions in the real world. In the context of the prototype, they are the renters and users of the space. Their actions include entrance, exit, payment and meditation inside the prototype.
- Physical component: the physical structure and the electrical components. The physical structure serves as the host environment for all systems. The electrical components include the solar panels, battery, sensors, camera and an electric lock. There are three main functions which are ensuring operation of the access and exit of users, providing steady energy supply and ensuring the security of the prototype.
- Back-end component: the computing component. This handles data processing, off-chain storage, broadcasting data to the blockchain, communicating between and with sensors and actuators, and controlling energy and access logic. With its ability to control energy usage and access permissions, the back-end helps optimize system performance and enhance the overall user experience. It also serves as the communication link between the local environment and the blockchain. For example, the remaining battery level is periodically and automatically updated to the blockchain, which allows the calculation of rental time availability for the user.

5. Prototype development

Building on the defined scope, system design, and technical framework from Section 4, this section now focuses on the more detailed implementation and system integration of the prototype, including

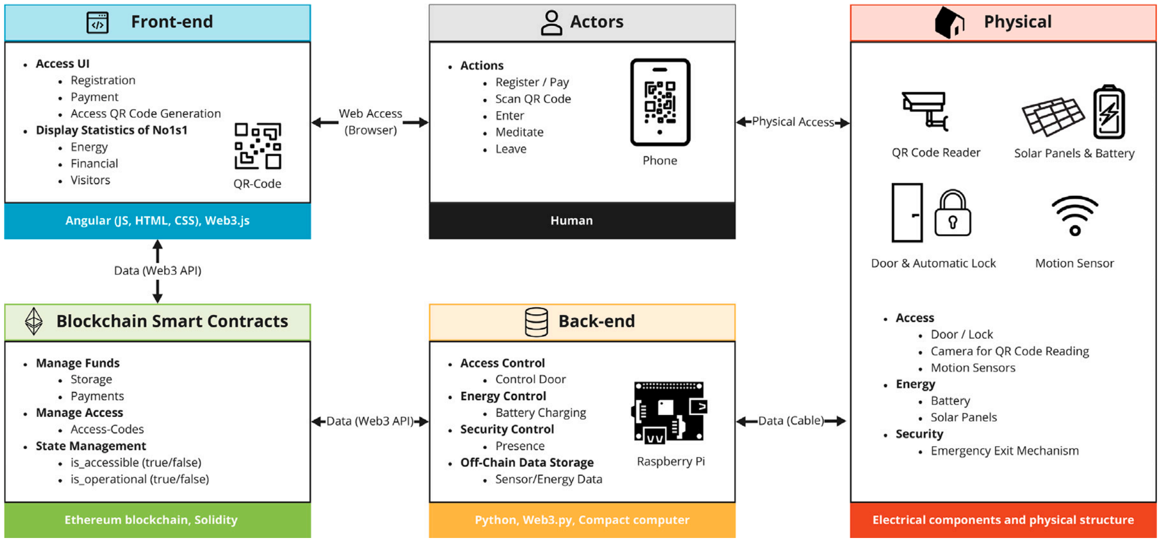

Fig. 4. Overview of no1s1's technical framework.

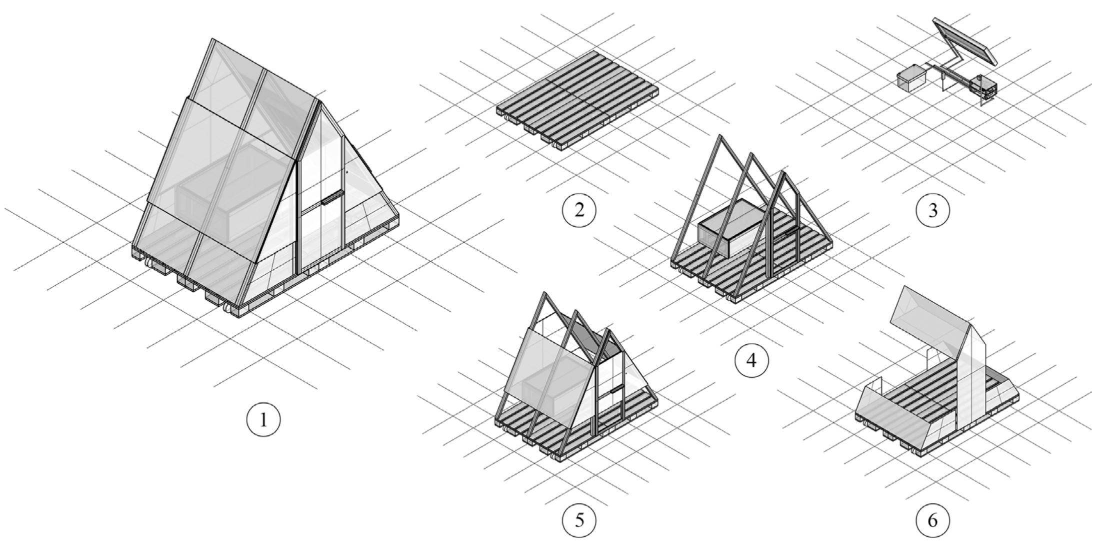

Fig. 5. Physical configuration of the prototype.

the physical configuration and structure (5.1), the power supply and electrical operation (5.2), the information flow and process logic (5.3), the software and network technical choices (5.4), and the software and hardware integration (5.5).

5.1. Physical configuration and structure

Physical design choices include the spatial layout, structure, and arrangement of different hardware and electrical components within the space to ensure that it provides a smooth experience for the user and serves the intended purpose of the prototype.

To meet the operational requirements of a meditation space, the minimum requirement is to provide an enclosed private area for an individual to sit comfortably. Simultaneously, the prototype must maintain a certain level of structural stability, flexibility, and mobility as it will be moved to several locations for demonstration and testing. Ultimately, a portable wooden A-frame structure was selected as the optimal structural solution for the prototype (see Fig. 5, Step 4). Then the A-frame structure is enclosed with side wood panels and semi-transparent PVC Vinyl folio (see Fig. 5, Step 5 and 6). The final design of the physical configuration is shown as a whole in Fig. 5 Step 1. Steps 2 through 6 show the step-by-step assembly process from the base (Step 2), electrical system (Step 3), A-frame structure and chair (Step 4), side wood panels (Step 5), to the enclosing folio (Step 6).

In addition to structural requirements, the physical space must accommodate the electrical system. RGB LED strips are incorporated into the door frame and chair frames to enable responsive interaction between the prototype and the user. Various sensors are embedded into the structure; for example, a distance sensor is integrated into the chair, and a camera is placed in the front panel next to the door for the user to scan their QR code. The wooden side panels provide structural support for a solar panel and battery, which is discreetly housed within the chair's structure. The design of the structure went through significant iterations during the system testing phase and user acceptance testing phase (see Sections 6.1.3 and 6.1.4). The detailed design of the electrical and power system is described in the next Section 5.2.

5.2. Electrical operation and power supply

The operational and energy system design choices ensure reliable and efficient operation. They also enable the prototype to monitor

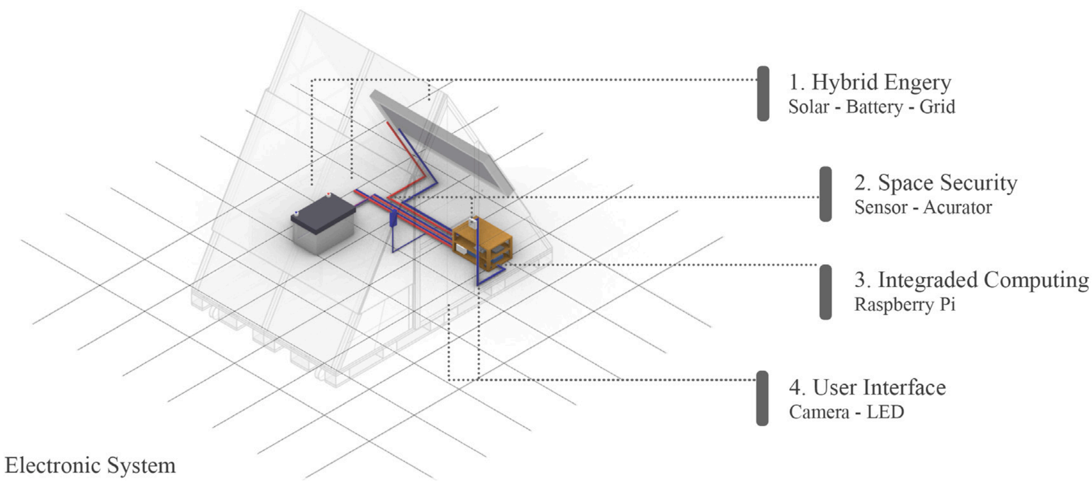

Fig. 6. Design of electronic system.

its own energy level. The electrical system has four main functions: ensuring power supply, providing space security, offering an interactive user interface, and controlling the computational power to operate the prototype.

The operation of the system is managed by the Raspberry Pi, a compact computer. It serves as the central computational hub for controlling the sensor and actuator network and the power network, overseeing space access, security, monitoring of the environment, and energy management. Python is chosen as the primary programming language for the electrical system for the reasons given in Table 2.

The prototype's electrical system uses a hybrid energy system as shown in Fig. 6. The battery is a 12 V, 120Ah deep-cycle Absorbent Glass Mat (AGM) battery and the solar panels are 17 V, 100 W solar panels. An Arduino-based MPPT solar charge controller is used to regulate the voltage difference and accommodate the irregular output of the solar panel. This controller offers benefits such as a higher solar conversion rate, multi-stage charging for battery longevity, increased charging efficiency, and scalability.

A smart shunt of 500 A/50 mV is implemented to accurately measure the energy flow from the battery. This shunt device provides real-time energy data output that can be transmitted to Raspberry Pi via a data cable without relying on the Internet or a built-in interface. This ensures that the energy system reliably updates the electrical system regarding its voltage and current output. The voltage output can be used to estimate the battery's state of charge (SoC), and the prototype's energy level can be broadcast to the blockchain and transparently displayed to users. When the prototype is indoors, there is not enough light to effectively charge the battery. To ensure a stable energy supply, an alternative connection to the power grid is implemented. The design of the electrical system went through significant iterations during all testing phases (see Section 6.1).

5.3. Information flow and process logic

Two information flows were identified as critical to the functioning of the prototype. The first information flow is the space access logic that allows a user to interact with the prototype. It necessitates the interaction of all five components identified in the technical framework: the front-end, the back-end, the blockchain, the human, and the physical space (see Fig. 4). The second information flow is the self-updating capabilities of the prototype, and involves only three cyber–physical components (the front-end, the back-end with feedback from the electronics, and the blockchain). Both are described in more detail below and in Fig. 7.

5.3.1. Space access logic

The primary information flow must enable the user to (1) register and pay for the meditation, (2) access the prototype, (3) exit the prototype, and (4) receive an escrow return, i.e., if the price of the meditation was less than the escrow amount paid, the user should be able to redeem the difference as a refund upon leaving the space. Fig. 7 shows in detail the information flow of the space access logic across these four processes. The process logic is triggered by physical events, mostly the user's interaction with no1s1.

The information flow occurs in two directions. It can be initiated by the user using the front-end, triggering a smart contract transaction with a change of state on the blockchain, captured by the back-end, resulting in an action on the physical no1s1 space. For example, this is what occurs in the case of the registration and payment (see Fig. 7, "registration and payment"). The user initiates the logic by executing a smart contract function with the desired meditation duration, making a deposit of cryptocurrency, which is registered on-chain. If successful, the smart contract generates a unique access ID based on the user's blockchain address and a chosen username. The front-end then creates a QR code based on this unique ID to provide secure and efficient access. Also the fourth step to redeem the difference between the actual cost after meditation and the amount paid upfront is following the same information flow by triggering a smart contract function by the user through the front-end (see Fig. 7, "escrow return").

The information flow can also occur in the other direction when it is triggered from events captured by electronic devices in the physical space, to the back-end. This triggers a smart contract function and state change on the blockchain, captured and displayed to the user in the front-end or through changing the physical appearance of no1s1. For example, this is applicable during the access and exit procedure (see Fig. 7, "access" and "exit"). Upon entry, the user must show the QR code to the prototype's camera to verify the user's identity and payment status. This time, not the user but the no1s1 back-end triggers a smart contract function that verifies this and, if successful, opens the door and turns the green light red to signal the successful status change. Afterwards, the user can enter the space and close the door. A proximity sensor built into the chair detects the user's movement, ensuring that the user has entered and that the door was locked. When access has occurred, the back-end again triggers a smart contract function to record successful access on-chain, the occupation of the prototype so that no one else can enter, and to trigger the countdown and meditative music in the headphones. The door automatically unlocks when the time is up, and the back-end triggers a third exit smart contract transaction to record on-chain the actual meditation time of the user and the availability of no1s1 for the next meditation.

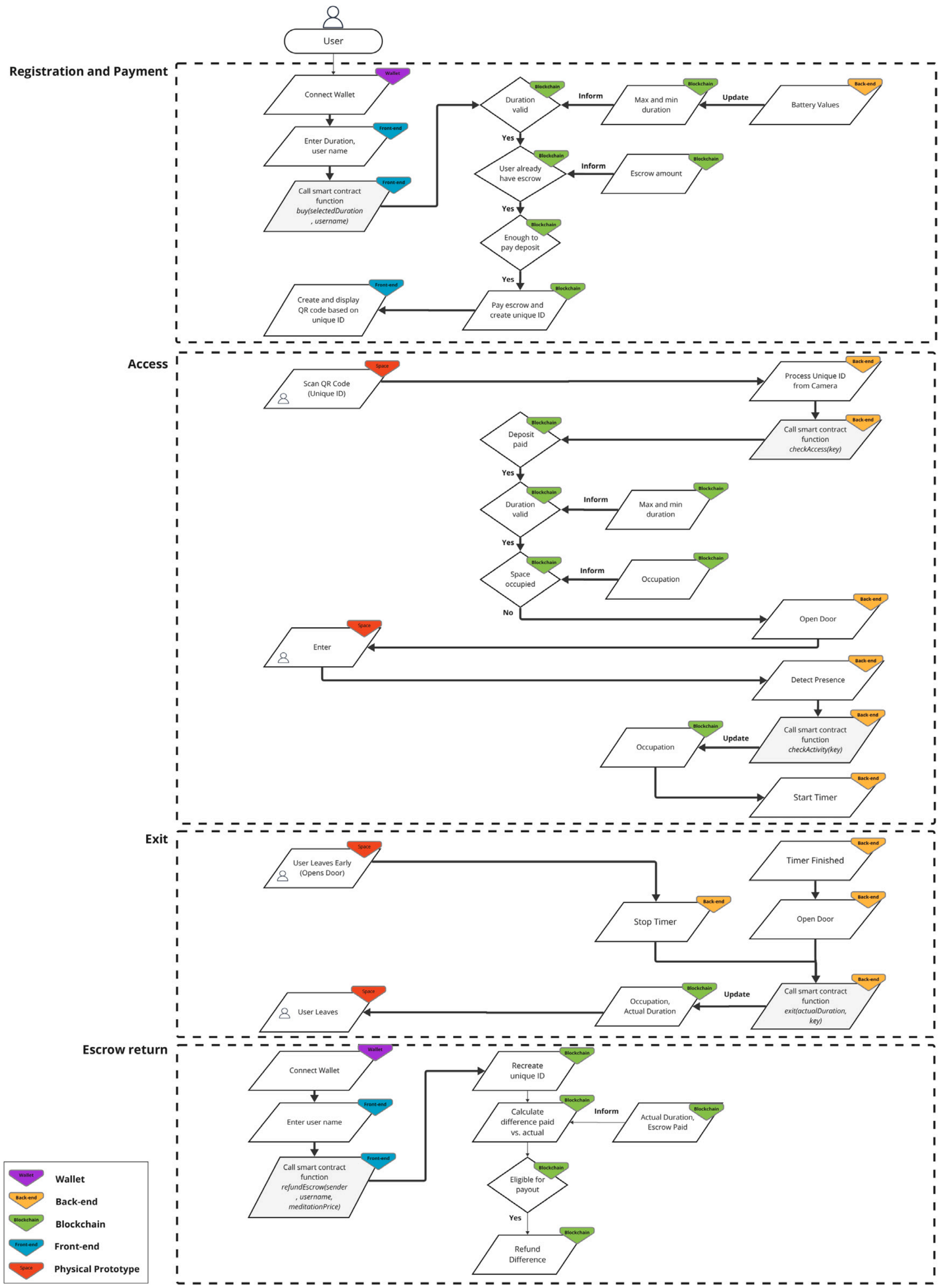

Fig. 7. Information flow of space access logic. (Note: interactions between technical components and the user are orchestrated by blockchain through updating essential on-chain state of the prototype.)

5.3.2. *System's self-updating logic*

The second type of information flow is autonomous and involves only no1s1 without the user. This is necessary to periodically update the blockchain state of no1s1, which informs the space access logic, for example, how long a user can purchase a meditation based on the available battery status of the prototype. The front-end can retrieve this information from the blockchain and display it to the user.

The main example implemented in this prototype is the energy management of the solar-battery system. The prototype needs energy to operate, thus the status of energy levels need to be updated on-chain to secure the immutability and self-sufficiency of the rest of the system. The battery energy level is transmitted from the smart shunt device to the back-end. The live data is then decoded, temporarily stored, and sent to the blockchain by triggering a smart contract function. This

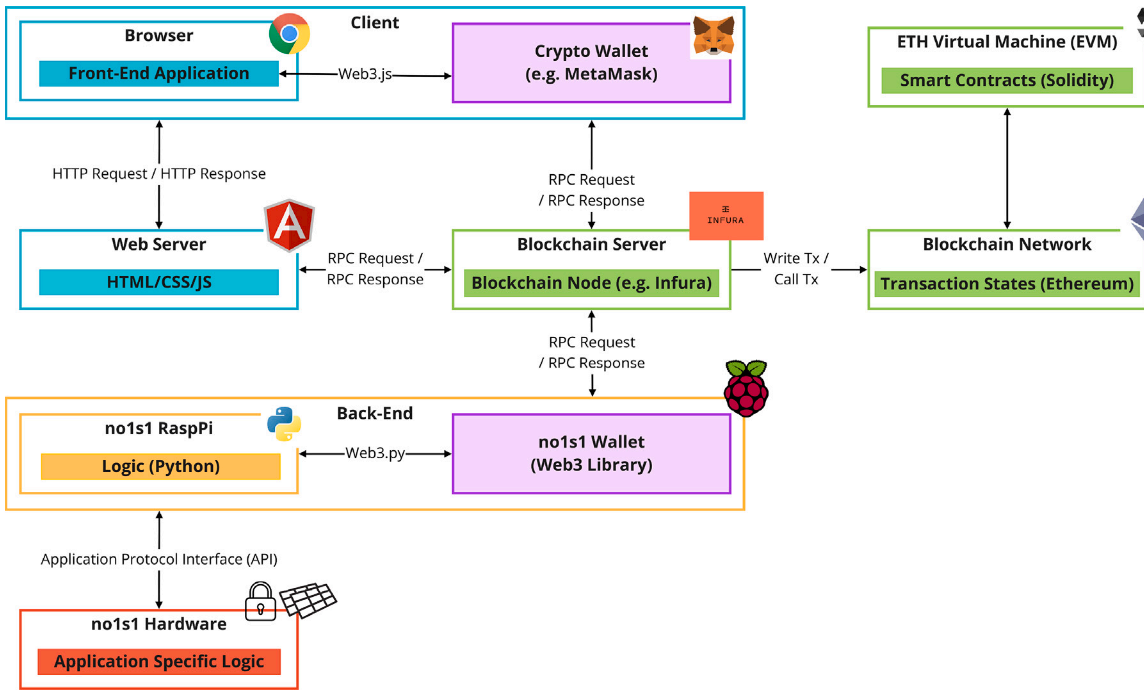

Fig. 8. Overview of hardware and software integration detailing the technical interactions between smart contracts, front-end, and back-end of the prototype.

update process is programmed to run every thirty minutes, ensuring the reliability and consistency of the data.

5.4. Software and network technical choices

The technical choices and rationales are detailed in Table 2. Network and software design choices affect system efficiency and performance by enabling communication and control of various components. These technical decisions impact digital governance, influencing data distribution across nodes, consensus mechanisms, and the maintenance of data immutability. These are not only technical decisions but also strategic choices that signal the ownership structure of non-human entities and the level of decentralization. However, this paper focuses on establishing an overarching discussion and a holistic framework. Technical choices are guided by the simplified system requirements outlined in Section 4.1. The following paragraph shows the simplified evaluation process of choosing a blockchain network.

The blockchain network configuration is responsible for data storage and transaction management, ensuring data integrity, security, and transparency. Given the scope and length limitations of this paper, the choice and use of the blockchain network are based on frameworks established by existing studies [74,77,78]. These analyses provide a clear decision framework for whether blockchain technology should be used and which type of blockchain network is suitable for implementation. This prototype (1) requires data storage, (2) has multiple writers, and (3) lacks an always online trusted third party (TTP). These three requirements and the native digital financial value transfer layer of blockchain technology made it a necessity for the artifact in this paper. However, given the systems complexity and scope, it remains unclear whether all participants are trusted or known, and whether public verifiability is required in all cases. Further investigation is needed with a finer granularity, considering every stage of the prototype lifecycle with different levels of human participation. Thus, to simplify the testing and demonstration with users, the network was initially chosen as a public permissionless blockchain network and later switched to a private blockchain network (effectively a fork of the previous public permissionless blockchain).

5.5. Software and hardware integration

The information flow and process logic described in Section 5.3 was implemented using the technologies and programming languages selected according to Table 2. The final technology stack and technical interactions are shown in Fig. 8. There are three main components that need to be implemented and coded: the front-end using the Angular framework with JavaScript, HTML, and CSS; the back-end using Python as the main programming language; and most importantly, the smart contracts using Solidity. Inter-component communication among the smart contract, front-end and back-end, is crucial to enable the coordination and operation of the prototype. Fig. 8 shows that there is no direct connection between the front-end and the back-end. Instead, the smart contracts take on the role of orchestrating the operations, thus ensuring the self-autonomy of no1s1. All code is available on Github at the URL https://github.com/no1s1labs/no1s1. Below we elaborate more on the role and implementation of the three components.

5.5.1. Blockchain smart contracts

The smart contracts coded in Solidity form the backbone of no1s1 and govern how the prototype operates. They encode the most important logic of no1s1 and store the on-chain *state* of no1s1, represented by variables that were defined to store key information that is required for the functioning of the prototype. An example is the variable *no1s1Occupation* that stores a boolean value whether the prototype is occupied or not. In addition to state variables about no1s1, the no1s1 smart contract also contains state variables about users, such as how much meditation time they have purchased, and whether and for how long they have already meditated. Since multiple state variables are associated with a user, a construct called *structs* is used to organize and store this related data together. To query the information about one specific user, the smart contract uses key–value pairs called *mappings* that associate each user-struct with a unique key created based on the users' blockchain-address and a chosen user name.

To modify or interact with the state variables, smart contract functions need to be executed. The functions in the smart contract define the actions that can be taken by users, the no1s1 back-end, or other

Table 2
Selection and rationale of technology choices for software and network.

Topic	Choices	Selection criteria	Selected	Rationale
DLT design	Public permissionless, Public permissioned, Private, Consortium, Hybrid	Open accessibility signals ownerless system, superior transparency, security and immutability, independent from third parties	Public permissionless	1. Public permissionless blockchains provide a high level of security and immutability due to their independence from third parties. 2. Smart contracts on public permissionless blockchains provide programmability of ownership by controlling all access rights to on-chain information and value. 3. High transparency signals the machine sovereignty and provides trust for users.
Type of public blockchain	Ethereum, Cosmos, Solana, Bitcoin, Stacks	Turing completeness, popularity adoption, resource availability	Ethereum	1. Ethereum is one of the first blockchains to achieve Turing Completeness. 2. The ecosystem is established and the community provides rich resources for building applications. 3. Its virtual machine is used in many other networks.
Testing network	Goerli, Rinkeby, Sepolia, local	Stability, ease of use, compatibility	Rinkeby	1. Rinkeby testnet was one of the largest, most active and supported testnets. 2. Test-ether can be easily obtained by request from the faucet.
Integrated development environment	Hardhat, Truffle, Ganache, Remix	Features, ease of use, stability, performance	Remix, Ganache, Truffle	1. Remix is one of the easiest to use platforms for beginners with a web-based interface. 2. Ganache and Truffle work together to create a more comprehensive development framework with additional testing features that result in better performance.
Smart contract languages	Solidity, Vyper, Clarity, Anchor, CosmWasm	Popularity and adoption, ecosystem and resources, compatibility, security	Solidity	1. Solidity is the native and most widely used smart contract language for the Ethereum network. 2. It has a mature tooling ecosystem with rich packages and libraries. 3. Its security has been proven in large-scale real-world applications.
User interface framework	Angular, React, Vue	Features, performance, ecosystem	Angular	1. Angular is a full-featured framework with strong support tools and scalable functionality. 2. Rich ecosystem and out-of-the-box templates.
Hardware programming language	Python, C, C++, Java, JavaScript	Compatibility, ease of use, ecosystem, resources	Python	1. Python has simple syntax, human readability, and is beginner friendly. 2. Python has a strong ecosystem of both libraries and frameworks to support hardware development and interface with blockchain networks.
Interface library with blockchain	web3.js, ether.js, web3.py	Compatibility, features, stability and security, ecosystem, resources	web3.js, web3.py	1. web3.js was the first library to interact with Ethereum with comprehensive functionality. 2. Widely adopted and offers rich resources and tools. 3. web3.js shares a similar syntax with web3.py

contracts to interact with the state variables, and therefore the physical prototype. The pseudocode of the most important smart contract functions are listed in Appendix A.1; they enable the space access logic shown in Fig. 7. For example the buy()-function allows a user to purchase meditation time and is executed by a user. If successful, the smart contract updates the user struct with the most recent values for the respective state variables. Similarly, the function checkAccess() is called by the no1s1 back-end when a user scans a QR code to ensure the code is valid. Upon success, the smart contract updates the respective state variables of both the user and no1s1, e.g. ensuring a code cannot be used twice. To prevent unauthorized execution of a function, *modifier functions* are defined that provide conditional information. For example, only users that have successfully meditated can call the refundEscrow() function. Other functions can only be called by the back-end of no1s1.

Finally, the smart contracts hold also the financial value of the prototype in cryptocurrency. To pay no1s1, users can send *ether*, the native cryptocurrency of Ethereum, to the smart contract by executing the buy() function. The smart contracts keep track of all payments through state variables and can refund users with the difference between the paid escrow amount and the actual amount used for meditation.

The smart contracts of no1s1 were coded with many common security practices in mind to avoid exploitation of the prototype, e.g. by stealing funds or meditating without paying. In addition, state variables like the *isOperational* boolean variable can be utilized to deactivate the entire smart contract by special admins in case the prototype would have needed to stop operation. The whole functionality described above was also divided into two sets of smart contracts: a data contract and an app contract. The data contract handles the storage and management of all sensitive state variables such as user information and financial transaction, holds the funds of no1s1, as well as core smart contract functions that interact with the previous two. On the other hand, the app contract holds state variables and functions that are likely to change over time, such as the meditation price. Having these two separated smart contracts ensures smart contract updatability, as changes in operational logic, e.g. to the meditation price, can be implemented through redeploying only the app contract without affecting the integrity of the data contract. If not structured in this manner, the stored prototype data in the data contract would be lost during such a smart contract update, resulting in the historical data of the prototype being erased. Moreover, this design adds a layer of security to the core on-chain data of the prototype stored in the data contract, since all functions are designed to be invoked through the app contract, and additional modifiers could be implemented also at a later point in case there would have been permission errors in the data contract.

5.5.2. Front-end

The front-end (see Fig. 8, blue) ensures the user interaction with the no1s1 smart contracts. The no1s1 decentralized application was coded using the Angular framework with JavaScript, HTML and CSS. The website was hosted on a free static Github server and made accessible through the URL https://no1s1.space. This allowed users to access the no1s1 front-end with a browser of their choice, both on desktop and mobile. In addition, support for a blockchain wallet such as Metamask

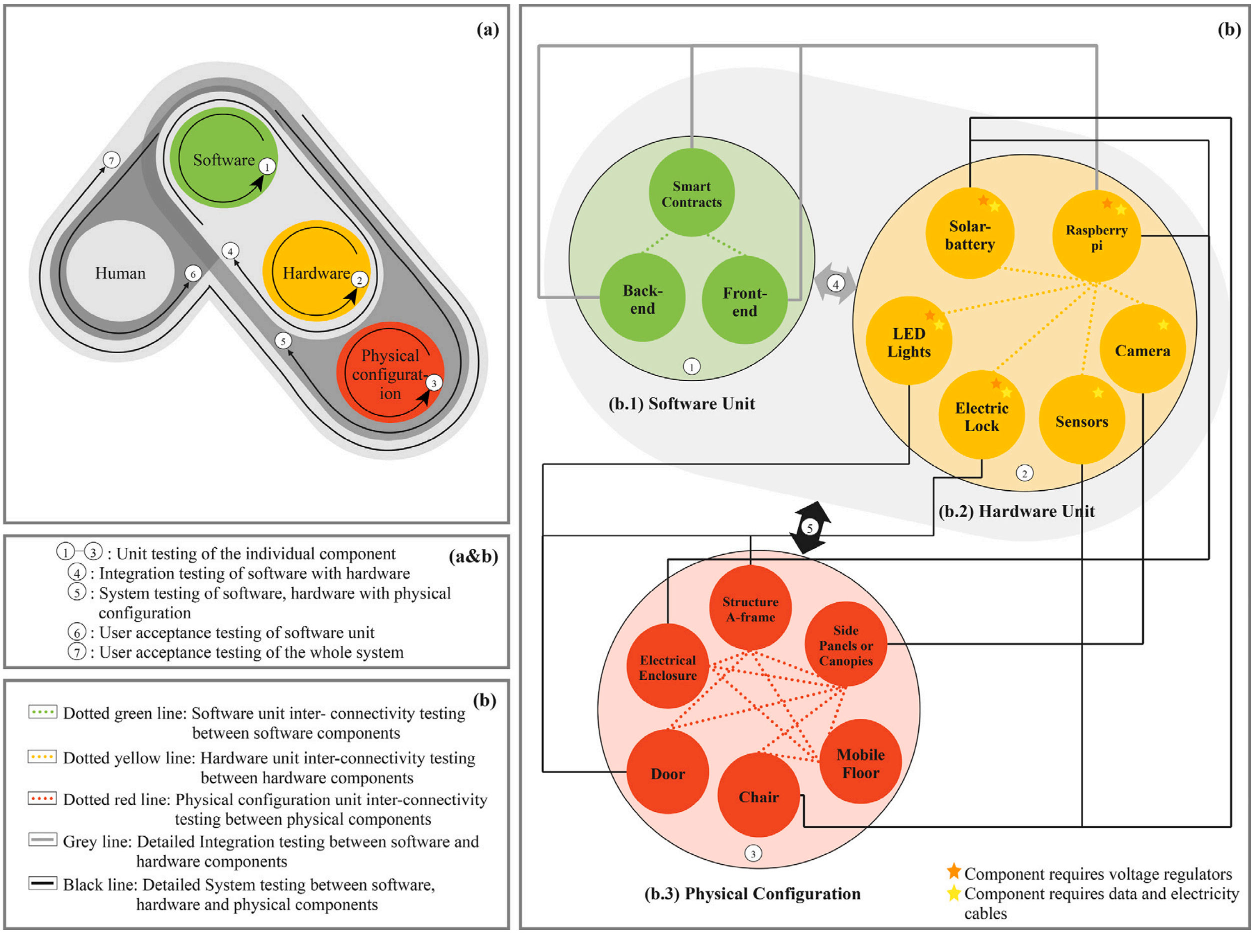

Fig. 9. (a) Overview of iterative testing process; (b) Detailed testing processes 1–5.

was integrated using the web3.js library. The wallet provides a connection to one or more blockchain nodes, which can then interact with the no1s1 smart contracts by executing transactions, such as function calls (see Fig. 8, green). With this, users could execute transactions such as buy() to purchase meditation time conveniently from their web browser by connecting and unlocking their crypto wallet and pressing one of the buttons implemented for this purpose.

5.5.3. Back-end

The back-end (see Fig. 8, orange) ensures the response of the physical no1s1 space to state changes in the smart contract, or in turn to change the state in the smart contract upon events in the physical space. This logic is coded in Python and deployed on a Raspberry Pi, which is capable of connecting to other physical hardware of no1s1 as well as to the Internet. Using the Web3.py library, the no1s1 back-end also has its own wallet to connect to the Ethereum blockchain and listen for state changes related to no1s1, as well as to execute functions. For example, if the distance sensor successfully detects the user entering the space and closing the door, the back-end triggers the checkActivity() transaction to change the state of the *isOccupied* variable to indicate that a user is meditating and no other user can enter. It can also trigger the functionality of the physical prototype. For example, by listening to the smart contract's events, the back-end will catch the successful execution of the checkAccess() function and trigger the opening of the door.

6. Testing and demonstration

To assess the viability of the conceptualization and design of the prototype, the final artifact no1s1 is tested as the first prototype of a self-owning intelligent building. Based on multiple tests, the prototype was further improved especially with regards to the user experience, and was thereafter comprehensively demonstrated. This chapter shows first the iterative testing processes during the construction of no1s1, and subsequently, describes the final demonstration of the artifact with the collected user feedback.

6.1. Iterative testing process

Graph (a) in Fig. 9 shows that the iterative development process comprises seven sub-processes, which fall into four main categories: unit testing, integration testing, system testing, and user acceptance testing (UAT). Graph (b) in Fig. 9 illustrates in detail the components involved in sub-processes 1 to 5.

The unit testing is where software, hardware and physical configurations, are tested independently to ensure proper functionalities within each unit (see sub-processes 1 to 3 in Fig. 9). Within the unit, different components require to be tested for their inter-connectivity, as shown by the dotted lines in (b) of in Fig. 9. Integration testing focuses on evaluating the interactions between software and hardware components (see sub-processes 4 in Fig. 9). During this stage, the goal is to identify interface defects and ensure the functionality from combined components works together as intended. The gray lines in graph (b) of Fig. 9 represent these integration tests between software and hardware components. System testing (see sub-processes 5 in Fig. 9) assesses security, spatial arrangement, and connectivity once hardware components are integrated with the prototype's physical structure. The black lines in graph (b) of Fig. 9 indicate the expansion of system testing from software and hardware components to physical prototype

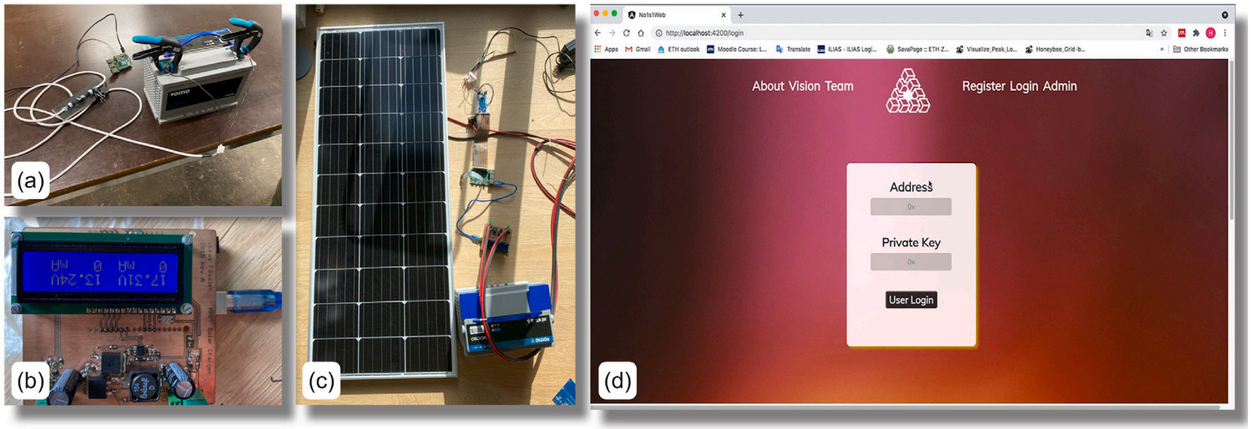

Fig. 10. Examples of unit testing. (a): Battery unit; (b): Solar Panel and battery controller; (c): Solar-battery, controller and shunt units; (d): Front-end. (Note: numerous components in the photo were upgraded after unit testing.)

configurations. UAT (see sub-processes 6 in Fig. 9) is the final step when users are involved. The end-users validate the software's functionality in real-world scenarios, ensuring that it meets user needs and is ready for implementation.

6.1.1. Unit testing

The unit testing process is the isolated testing of a single component, as represented in processes one, two and three in Fig. 9. The processes for each hardware, software, and physical configuration unit are outlined in graph (b) of Fig. 9, with real examples of the prototype shown in Fig. 10. Software unit tests, (process 1 in (a) and (b) of Fig. 9) are carried out in 5 parts: (1) front-end software units; (2) smart contract units; (3) back-end units; (4) front-end and smart contract unit inter-connectivity; (5) back-end and smart contract unit inter-connectivity.

For the front-end unit, it was important to test the data interactions, display and logging. The process typically involves debugging in the code editor, followed by deployment in a local environment to ensure that the encoded components display correctly and the data logging functions as intended (as shown in picture (d), Fig. 10). Smart contracts follow a comparable initial process before being deployed in a local blockchain simulation or a web-based integrated development environment. The back-end units, encoded in Python, are tested in an integrated development environment or a code editor, similar to the front-end. Once the front-end, smart contracts and back-end are confirmed to operate independently, their unit connectivity to each other is tested in a local deployment environment using libraries like web3.js, web3.py, with both a local server for the front-end and a local blockchain environment for the smart contracts. These processes are illustrated as dotted green lines in (b.1) of Fig. 9.

Hardware units (process 2 in (a) and (b) of Fig. 9) undergo basic functionality testing through the following steps. Step 1 connects the components to electricity without a software interface, as shown in the first three images of Fig. 10. Step 2 tests the inter-connectivity of multiple hardware components, indicated by yellow dotted lines in graph (b.2) of Fig. 9. The picture (c) in Fig. 10 illustrates connectivity testing with solar-battery power units, verifying that the solar panel is charging the battery and that data is being read accurately from the controller board. The controller in (b) Fig. 10 exhibited inconsistent performance and unstable connectivity, so it was replaced with a new industrial-grade MPPT controller.

Unit testing of the physical structure requires testing the physical components before integrating hardware systems (process 3 in Fig. 9). The whole physical structure is described in detail in the first paragraph of Section 5.1, and also illustrated in Fig. 5. Its unit testing covers the following five parts: (1) the mobile floor system; (2) the A-frame structure and its attachment to the floor; (3) the side panels and canopies and their attachments to the A-frame; (4) the door unit and its attachment to the wood canopies; (5) the chair unit and its attachment to the floor and back canopies. The attachments of physical components are represented as red dotted lines in graph (b.3) of Fig. 9.

6.1.2. Integration testing

Integration testing is shown in Fig. 9 as process number 4, and this process refers to the integration of software and hardware units. In the last Section 6.1.1, software and hardware units were examined separately, however, most hardware requires a software to be controlled, interacted with, or accessed by a user, and the records of the hardware need to be updated to the blockchain through software interfaces. These testing processes are shown as gray lines in graph (b) of Fig. 9.

Detailed examples of controlling hardware via software are shown in Fig. 11, where a screen is connected to the Raspberry Pi, controlling the LED lights (picture (a) in Fig. 11), a motion sensor (picture (b) in Fig. 11) and a camera (picture (b) in Fig. 11). For these three components, the LED lights and the motion sensor are mainly used as a signal for the users, while the camera is integrated into the access logic (more details in Section 5.3.1) and has direct interactions with both the user, the back-end and the blockchain. These testing processes are mainly related to electronic sub-systems 2,3 and 4 in Fig. 6.

6.1.3. System testing

The main evolution from integration testing to system testing is to fit the hardware and software systems into the physical structure configuration and spatial arrangements (see process 5 in Fig. 9). All hardware components require system testing and are illustrated as black lines in graph (b) of Fig. 9. Examples of integrating and fitting electrical systems, the electric door lock and other sensors into the physical structure are shown in Fig. 12.

Picture (a) in Fig. 12 represents an initial effort to integrate electronic components directly into the structure. However, as the system grew in complexity, the placement of the Raspberry Pi made iterative testing and updates to the hardware and software difficult due to limited access. While the electric lock and camera needed to remain in their designated positions, the Raspberry Pi and other electrical components were later organized into electrical boxes. Picture (b) in Fig. 12 tests the hardware and software components while the hardware components are mounted in the physical space. It is clear that the hardware placement and cable organization must be carefully designed. The stars in graph (b.2) of Fig. 9 depict this complexity. The components marked with a yellow star require a voltage regulator. Various

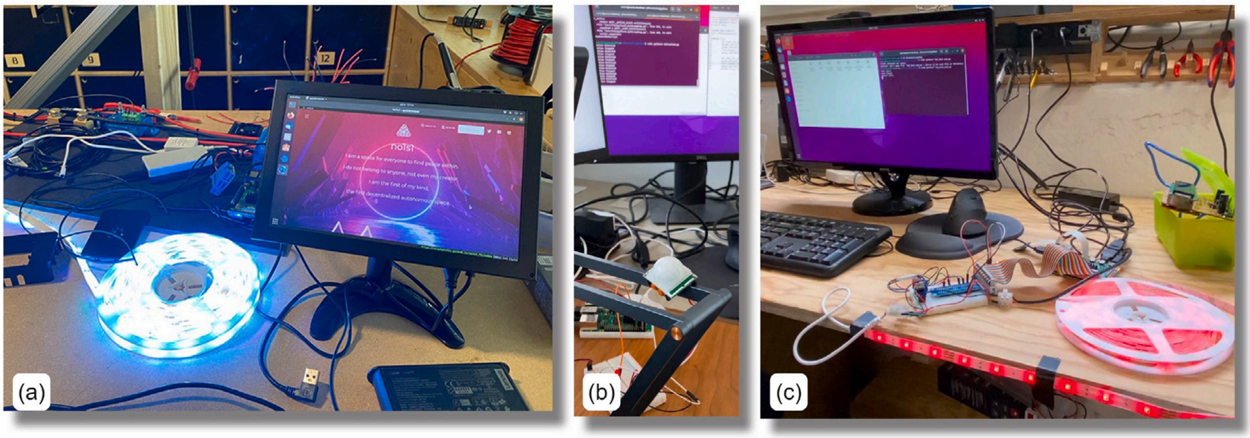

Fig. 11. Examples of integration testing. (a): LED light control testing; (b): Distance sensor control testing; (c): Camera unit, LED light and button control testing.

components demand different voltage levels, such as 5 V, 12 V, and 24 V for input and 24 V, 12 V, and 240 V for output. The orange star indicates data and power cables; while some components can use a single cable for both functions, it is more common for separate cables to be used for data and electricity.

Pictures (c) and (d) in Fig. 12 illustrate the evolution of electronics enclosures. Key components such as the Raspberry Pi, camera, motion actuator, battery shunt, voltage regulators and MPPT were initially organized and mounted in a plastic box for easier mobile testing, as shown in picture (a) of Fig. 12. When integrated into the structure of the prototype, it was found that such an enclosure strategy created difficulties for cable passage, which hampered connectivity between the energy system and other components. As the complexity of the electrical system increased, the enclosure iterated into the picture (b) in Fig. 12, which is a custom-built wooden casing. The hardware components and cables were reorganized and rearranged according to the no1s1's physical structure and spatial requirements.

6.1.4. User Acceptance Testing (UAT)

While previous testing processes have verified that the software, hardware units and physical configuration operate as intended, UAT ensures the no1s1 prototype meets the needs of the audience in different settings (illustrated as the sub-processes (6) and (7) in Fig. 9). By allowing human interaction, unique user-specific issues are exposed and areas for improvement were identified. There are three main areas of impact: (1) hardware and physical configurations, (2) software user interface, (3) procedures involving user interactions.

Hardware requires basic safety measures, such as casings and enclosures to prevent damage and accidents during interactions and to ensure visibility for demonstration purposes. Additionally, the modular design, color-coded wiring and labeled ports allow for future upgrades and adaptations, ensuring that the system can evolve alongside technological advancements. This can also be seen in the iterative testing process of the electronic enclosures (in Fig. 12 from (c) to (d)). In addition to the rationale for system testing discussed previously, this iteration addresses user needs by providing a more open casing that enhances system visibility.

Furthermore, UAT can also be implemented with only the software component, e.g., to improve the web-front as shown in the left image in Fig. 13. Integrated with the electrical system and the blockchain through previous testing processes, the web-based front-end has improved user accessibility, ease of use, and reduced waiting time, with additional user-friendly functions such as displaying the real-time status of the energy system to increase the data transparency.

One of the most important process improvements that cannot be achieved without the user is the space access processes (see Section 5.3.1 and picture (b) of Fig. 13). User feedback on the transaction wait time, actuator response time, sensor accuracy, and user interface friendliness provides valuable insights for refining the overall usability and ensuring that the system meets user expectations. The series of improvements during this testing phase not only provides system integrity but also extends the life of the system through improved security and accessibility.

6.1.5. UAT case studies

After the internal iterative testing phase described in Section 6.1, the prototype was evaluated as a complete artifact in real-world deployments (shown in Fig. 14). Benefiting from the modular design and mobility of the structure, the case studies were conducted in three different locations with different user profiles over the course of more than a year. This real-world evaluation process is part of the final testing step (process 7 in Fig. 9).

Case 1 (picture (a) of Fig. 14). no1s1 was deployed on the Ethereum testnet, Rinkeby. Users interacted with the prototype with guided steps using their mobile phones for registration and payment.

- Location: ETH Zurich Student Project House, Zurich, Switzerland
- Duration: 1 day
- Users: practitioner, students, and researchers

Case 2 (Picture (b) of Fig. 14). no1s1 was deployed on the Ethereum testnet, Rinkeby. Users interacted with the prototype using an iPad, following explanations and guided steps to minimize human intervention in the process.

- Location: ETH Zurich Pavilion at the World Economic Forum, Davos, Switzerland
- Duration: 3 days
- Users: more than 40 users with various academic and business background that are attending World Economy Forum

Case 3 (Picture (c) of Fig. 14). As a concept showcase, museum visitors can learn about no1s1 through an on-screen video narrated by AI voices. Feedback has been gathered through unstructured discussions with users. The prototype is powered on but not open for registration or payments.

- Location: House of electronic Arts, Basel, Switzerland
- Duration: 2 month
- Visitors: people with an artistic interest in digital installations that move through the museum

During this user evaluation phase, several critical issues emerged. Firstly, there is a delicate balance between user-friendliness and decentralization in the access mechanism. Many users do not have a crypto

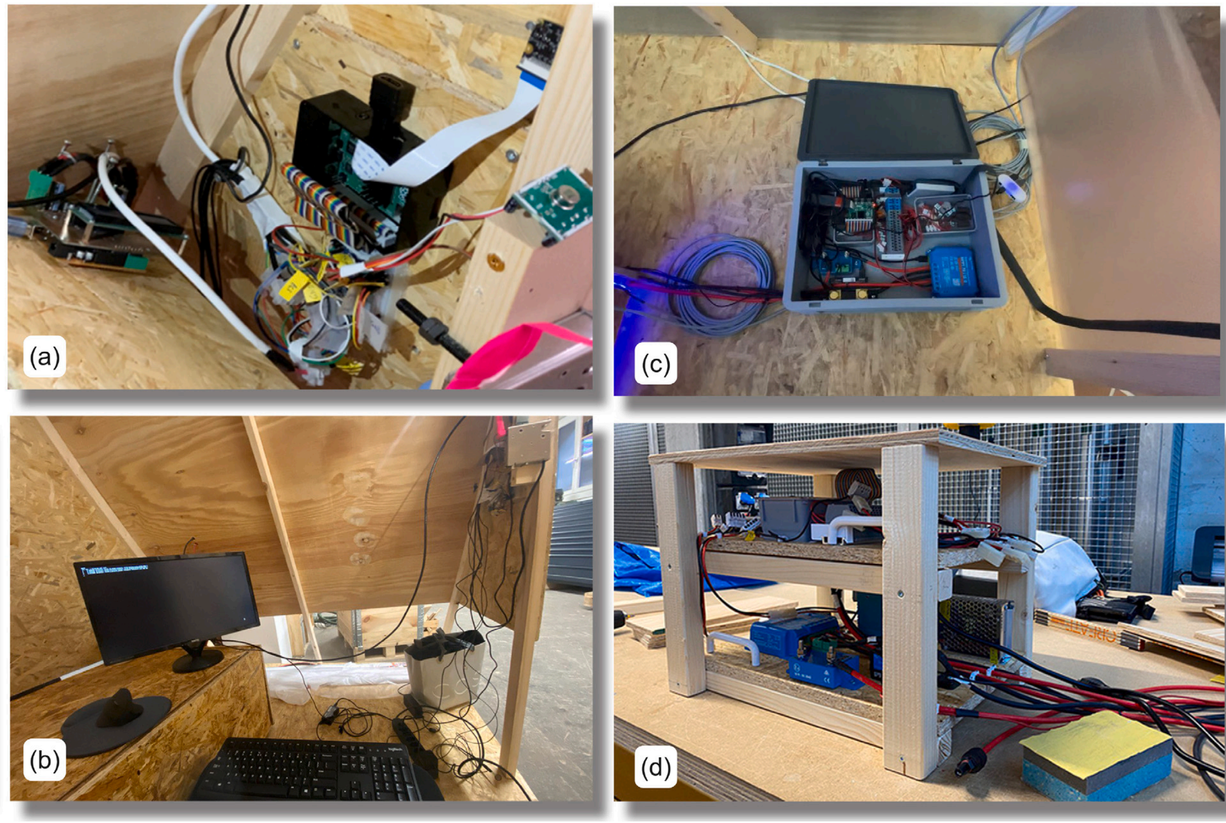

Fig. 12. Examples of system testing. (a): Raspberrypi testing; (b): Electric cable placement testing; (c): Electronic plastic casing box testing; (d): Electronic wood casing testing.

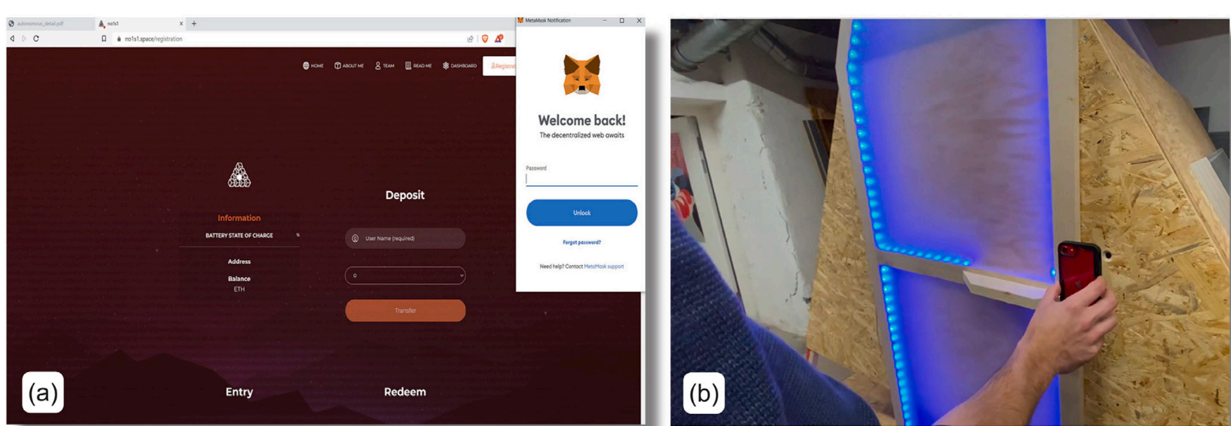

Fig. 13. Eamples of UAT testing. (a): Front-end user interaction testing; (b): Access interaction testing.

wallet for direct payments, and in exhibition setting, time constraints make on-boarding these users with a crypto wallet challenging. Thus, for UAT, a pre-funded crypto wallet was programmed for all the users to make quick payments. However, in the final demonstration, this procedure is improved due to the user's concern about putting his own assets at risk. Secondly, the wait time for on-chain data transactions exceeded the typical attention span of the average user, prompting consideration of network transaction throughput. To reduce transaction time, the demonstrations used a locally hosted public network instead of Ethereum testnet. Thirdly, the mechanism for operating the door had some complexities that required calibration. For example, during testing, it was observed that users frequently took longer to enter the space, so the door's programmed wait time was adjusted based on actual interactions. Despite these challenges, it is important to highlight that the overarching concept of the prototype has demonstrated its viability and functionality, providing a positive indication of its potential to reshape the domain of building access mechanisms.

6.2. Demonstration

The final implementation of the prototype was tested with 31 users at the University of Zurich (UZH) in the open lobby of an institutional building (see Fig. 15). The selection of this location ensured security for the physical structure, enabled continuous access to grid electricity

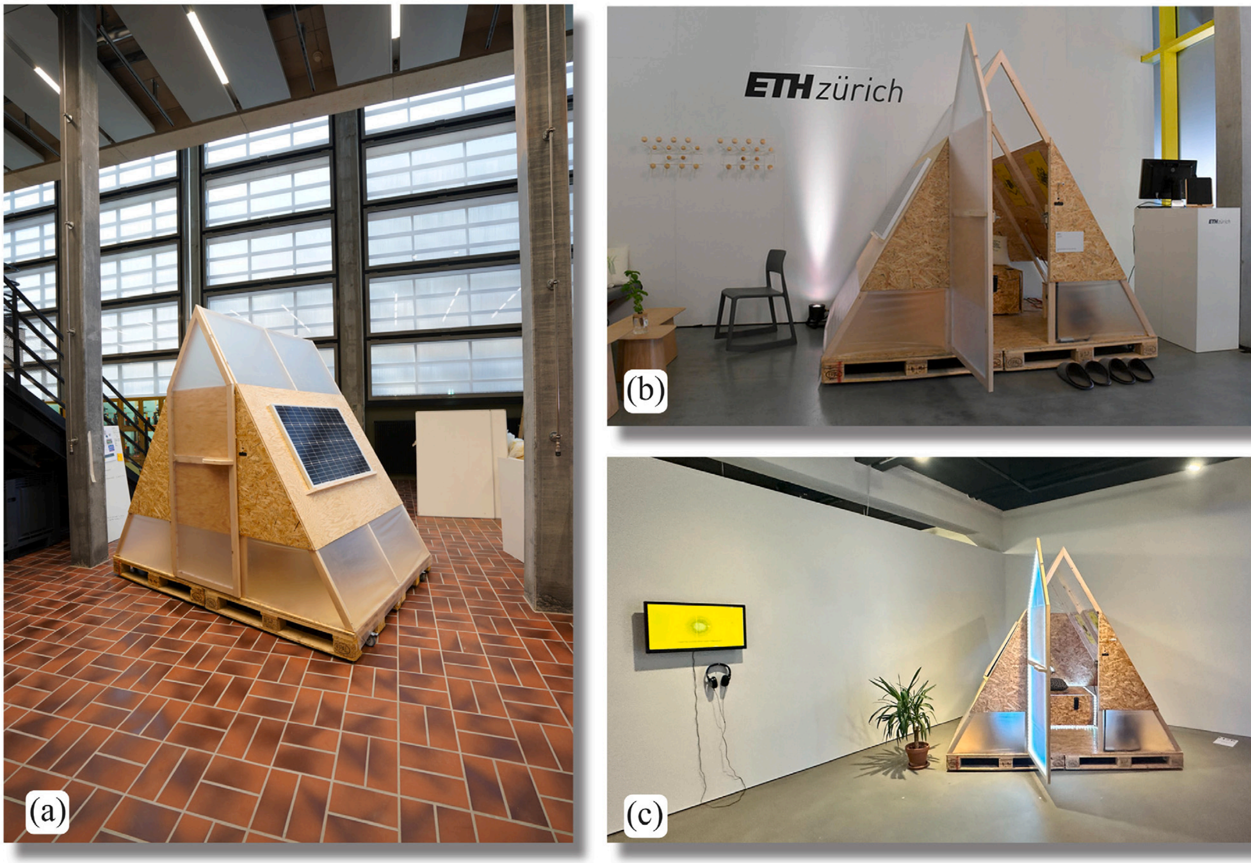

Fig. 14. The three UAT case studies: (a) ETH student project house; (b) ETH pavilion at WEF; and (c) HeK exhibition.

and a reliable Wi-Fi connection, all of which were essential for the prototype's operation. User profiles ranged from students, researchers from ETH Zurich and the University of Zurich, industry practitioners, and even curious passersby. Users pre-registered by choosing a 30-min time slot and signing a pre-agreement form prior to the testing period.

6.2.1. Technical set-up

Based on the observations of the testing phases, adjustments were made for the final demonstration. A new UZH Ethereum local blockchain network was selected as the implementation environment, a private derivative of the Ethereum chain. Therefore, the blockchain is compatible with Ethereum smart contracts, ensuring a seamless transition and operation. The basic information of the network is as follows:

- Network name: UZHETH
- Network URL: https://vm-216.s2it.uzh.ch
- Chain id: 5
- accessibility: Public

This choice offers: (1) increased transaction throughput and reduced user waiting times, (2) an easy-to-use faucet for all users to acquire and utilize their own test cryptocurrency, (3) maintenance of a similar system security level to the testnet, and (4) cost-effective transactions due to the low monetary values of these tokens. Furthermore, to ensure that users could onboard seamlessly with the automated and independent testing process, they were required to spend time to set up a crypto wallet and pre-fund it. Each individual's testing period was 30 min, inclusive of the time required for onboarding prior to the test and a subsequent 10-min questionnaire session.

6.2.2. Demonstration procedure

The demonstration procedure was refined during the series of testing phases (see Sections 6.1.1, 6.1.3, and 6.1.2) especially the UAT case studies (see Section 6.1.5). It was found that many users have no experience with crypto-currency or a wallet. In UAT case study 2, a pre-funded wallet was implemented that users could access through a custom-designed iPad app. While this setup was convenient, it diminished the sense of giving real funds to the house. Thus, a new demonstration procedure was designed as follows:

- Pre-Demonstration: An advertisement about the demonstration was circulated to potential users. Interested participants were required to register for a specific time slot to attend the demonstration. Once registered, users received links to additional information regarding the prototype, ensuring they were properly prepared for the demonstration.
- Demonstration: During the allocated 30-min time slot, the demonstration began with a 5-min introduction to the prototype's concept and operation. Following this, users spent the next 10 min installing the Metamask crypto wallet app. After successfully setting up Metamask, users received UZHETH currency for testing no1s1. Participants interacted then with the no1s1 platform by registering through the web interface on their phone and depositing funds into the no1s1 wallet. To streamline the testing process, the no1s1 usage time was pre-configured to 1 min.
- Post-Demonstration: After the demonstration, users were encouraged to provide constructive feedback by completing an online questionnaire. This questionnaire served multiple purposes: (1) validate the feasibility of the no1s1 platform, (2) gather user

Fig. 15. Final demonstration of no1s1 at UZH.

Table 3
Exemplary user transaction details.

Title	Data
Blockhash	0 × 42c001dabe70ad1302bc85fd44e923c1d6b541894be50ad4d08b367bd9bd9f79
Blocknumber	2 787 692
From(address)	0 × 2f383f9704cBAE28E2b1FC4d620D9e8eC34B9524
Gas	110 218
Gasprice	1 000 000 000
Hash(transaction)	0 × 3fe4c28377dcfcbc77b0e30f57f7b698595e5c14714af5b9d12dc0664b4faabe
To(address)	0xE7C1FBCE16D2f88F890A6a2c27B93B4636A80f5D
Nonce	10
Value	5E+17
Timestamp	1 660 661 845
Time	2022-08-16 16:57:25+02:00

insights on the concept, and (3) identify potential areas for improvements and extensions.

7. Evaluation

The evaluation of the design artifact is based on the results of the demonstration, the user feedback, and a critical reflection on the fulfillment of key system requirements as reflected in the final version of the artifact.

7.1. Results of demonstration

During the final demonstration, 31 users successfully interacted with the prototype. An example of the transaction data is shown in Table 3. The participants' experience directly confirmed both the completion of the transactions and the codification structure between the app contract and the data contract (explained in Section 5.5). Each transaction adhered to the encoded access logic, as detailed in Section 5.3.1, ensuring a secure and well-structured execution.

7.2. Assessment of key system requirements

After the demonstration, the researchers critically assessed the performance of the artifact against the eight key system requirements: blockchain integration, comprehensive metadata, self-agency, financial transaction, access condition, and user interaction (see Section 4.1). The results are summarized in Table 4. Upon successful transactions and collection of users metadata, the verification of the prototype is conducted against the system requirement developed under Section 4.1. The Table 4 outlines the fulfillment levels of the prototype against key system requirements. The "Blockchain Integration" requirement is mostly fulfilled, as most critical statuses of the prototype are recorded

Table 4
Verification based on system requirements.

Key requirements	Detailed description for each of the four autonomy features	Fulfilled?
Blockchain integration	**Operation autonomy**: Three main types of data, namely user, prototype occupancy, and security, were all updated and recorded on the blockchain. The energy data was working effectively, but the indoor solar input was missing in the last demonstration. **Human interface autonomy**: User identity and occupancy data was recorded on-chain and retrieved by the front-end and back-end interfaces to verify the user's identity. **Revenue and ownership autonomy**: User payment and escrow return data was recorded on-chain and retrieved by the front-end and back-end to verify all payment processes.	Mostly
Comprehensive metadata	**Operation autonomy**: Operation-related transactions were recorded. **Human interface autonomy**: Human-interface-related transactions were recorded. **Revenue and ownership autonomy**: Revenue and financial transactions were recorded.	Yes
Self-agency	**Operation autonomy**: Achieved autonomous access and exit processes, security, occupancy time and experience control, and partial energy control without human intervention. **Human interface autonomy**: Users were able to interact with the system with a minimal onboarding process, largely due to their unfamiliarity with blockchain technology. **Revenue and ownership autonomy**: Established self-ownership of financial assets through cryptocurrency without any human involved having access.	Mostly
Financial transaction	**Operation autonomy**: N/A **Human interface autonomy**: N/A **Revenue and ownership autonomy**: Revenue generation from space rentals was encoded in smart contracts using cryptocurrency.	Partially
Access condition	**Operation autonomy**: Allowed pseudonymous access for users. **Human interface autonomy**: Simplified user access with web-based front-end, QR code scanning, and digital wallets. **Revenue and ownership autonomy**: Ensured that the revenue belonged to no1s1.	Yes
User interaction	**Operation autonomy**: Hardware operations related to the physical prototype were mainly performed in the back-end to reduce latency and wait time. **Human interface autonomy**: Simplified the process with a web-based front-end, QR code scanning, and escrow process. **Revenue and ownership autonomy**: Users interacted directly with the prototype system for financial and operational purposes.	Partially
Physical scale	**Operation autonomy**: Designed for single user interaction, ensuring focused operational efficiency. **Human interface autonomy**: N/A **Revenue and ownership autonomy**:N/A	Yes
Functionality	**Operation autonomy**: Users participated autonomously in core functionalities (e.g. meditation). **Human interface autonomy**: Users only had to register on the front-end and scan their QR code using a physical camera built into the device. The rest of the interaction was automated. **Revenue and ownership autonomy**: The prototype has accumulated revenue and owns assets.	Yes

on-chain; however, the indoor solar input data was missing in the last demonstration. The "Comprehensive Metadata" requirement is fully met, with all transactions recorded alongside 11 critical metadata points. Similarly, the "Self-agency" requirement is also mostly satisfied, as the system allows users to maintain ownership of their financial assets without any access rights granted to human stakeholders, and minimized level of operational autonomy is achieved. Human can interact with the system directly with minimized guidance. The "Financial Transaction" requirement is partially fulfilled through encoded revenue generation for space rentals in smart contracts. However, more complicated financial mechanisms tied to operation and users can be developed. The "Access Condition" requirement is fully met, allowing pseudonymous access for users without identity restrictions. The "User Interaction" requirement is partially fulfilled, as some users still demanded guidance during interactions. Lastly, both the "Physical Scale" and "Functionality" requirements are fully met, enabling single-user interactions and allowing users to autonomously engage in core functionalities, such as meditation, through a streamlined process. This mapping ultimately showcased the successful realization of prototype's intended functionalities.

7.3. Assessment of achieved scope of autonomy

Based on the assessment of the key system requirements, the following sections critically assess the degree to which the four scopes of autonomy are achieved.

7.3.1. Operation autonomy within the management autonomy category

Achieving operation autonomy requires the fulfillment of 7 system requirements (excluding financial transactions). To reduce maintenance needs and emphasize operational autonomy, the prototype is specifically designed as a meditation space (see Section 4.2). Users in this space only need to sit with minimal activities. This significantly decreases the complexity of user activity within the prototype, leading to less maintenance such as frequent reoccurring cleaning. Additionally, it simplifies operational requirements by eliminating complex building systems such as drainage or insulation. The operation system successfully enables autonomous access, payment and exit processes, supports pseudonymous access through web-based front-end interfaces and QR code scanning, and optimizes latency by streamlined hardware operations in the back-end. The essential operational data is recorded

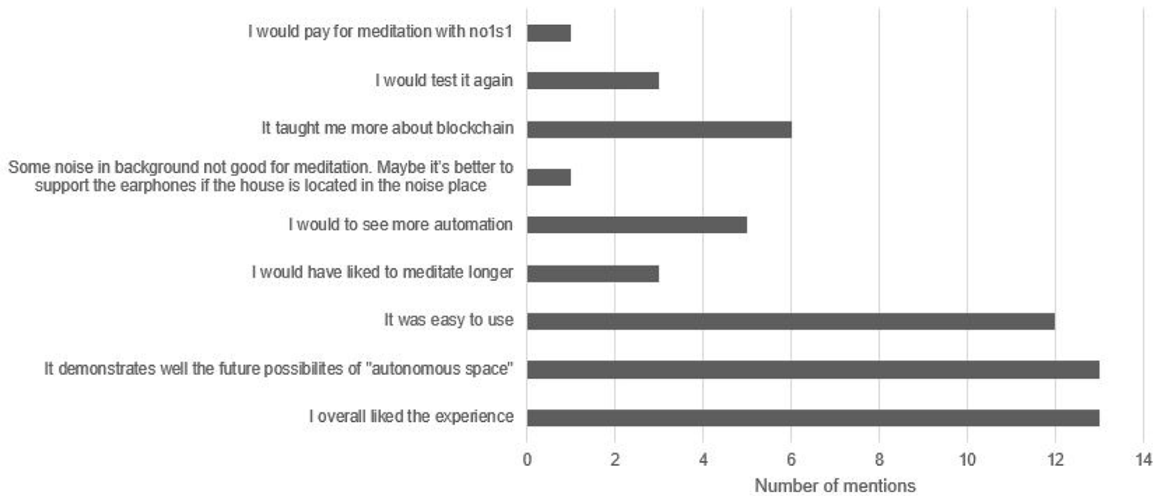

Fig. 16. User feedback of no1s1 artifact during the final demonstration.

on the blockchain with its metadata and retrieved as truth for verifying all important operational steps. Users can engage in core functionalities like meditation with minimal on-boarding aid, reinforcing the prototype's autonomous operational model. The general operational autonomy processes meet completely or partially all 7 key requirements that were intended to be achieved (see detailed description in Table 4).

7.3.2. Human interface autonomy within interaction autonomy category

Achieving human-interface autonomy requires implementation of six key system requirements (excluding financial transactions and physical scale). The developed interfaces ensure that user identity and occupancy data are recorded on-chain and easily retrieved by both front-end and back-end interfaces for user identity verification. The system simplifies user access, enhancing the overall user experience. Human interface autonomy meets completely or partially the 6 key requirements (see detailed description in Table 4).

7.3.3. Revenue and ownership autonomy within financial autonomy category

Achieving revenue and ownership in financial autonomy requires the fulfillment of 7 key system requirements (excluding physical scale). The electric system of the building is designed to hold its own crypto-wallet, where the access of the funding in the wallet is only granted to the prototype system itself. This exclusivity of fund access by the system itself marks the self-ownership of the prototype. The logic of revenue generation processes are encoded through smart contracts. The smart contracts are designed to be modular and robust, allowing for future updates to rental pricing models and other financial mechanisms to be integrated into the existing code base.

Additionally, the revenue and ownership of the prototype are designed to be closely linked to its energy consumption. Energy serves as the only value for the machine. The energy consumption during operation determines the rental price of the space, which in turn becomes the revenue generated.

All financial records and transaction are on-chain, ensuring transparency, security and decentralized access. The system allows the prototype to accumulate and hold its crypto-assets without human intervention. Overall, revenue and ownership autonomy meets the 8 requirements (see detailed description in Table 4), demonstrating a new possibility for cyber–physical systems to digital ownership.

7.4. Assessing user feedback

User feedback was collected using a survey, to which 19 of the 31 participants responded. By exploring the users' satisfaction levels and views on concept relevance, the survey offers insights into the current artifact's performance and establishes a basis for making informed decisions to guide future improvements.

In Fig. 16, each bar represents the number of times a specific feedback point was mentioned. A majority of users expressed positive views about the prototype, praising its overall experience. Users widely agreed that the prototype effectively embodies the concept of an autonomous self-owning house, offering a glimpse into future. Additionally, users also found the prototype to be user-friendly, contributing to its overall ease of use. Notably, six users reported an enhanced understanding of blockchain technology following their testing experience. Other positive feedback included a desire to "meditate longer" and "see more automation". However, a noteworthy critique emerged concerning the level of automation; users expressed a desire for increased automation in certain aspects. Furthermore, a minor criticism pertained to the limited meditation length imposed by the testing setup. The least selected feedback points were related to the background noise during meditation and the user's desire to pay for meditation. This multifaceted feedback not only provides valuable insights for refining the prototype but also serves as validation that the prototype operated smoothly and for the most part met user expectations.

In addition to gathering direct feedback on users' experiences with the prototype, participants were asked to share their perspectives on the benefits and challenges of blockchain technology in the context of autonomous buildings as in Figs. 17 and 18. While not conducted in a structured interview format, this supplementary inquiry provides insights and, to a certain extent, validates users' demands and concerns regarding the conceptualization and future prospects of blockchain-enabled autonomous buildings. This approach sought to gain insights directly from users, allowing their input to contribute meaningfully to the potential future application of the proposed concept.

According to Fig. 17, users prioritize transparency and reduced intermediaries as the primary benefits of implementing blockchain technology in autonomous buildings, with 12 and 10 selections respectively. Other notable benefits, each receiving more than 7 selections, include fractional ownership, altering ownership structures of existing real estate, and new incentives through token usage.

Interestingly, the main challenges perceived by users (see Fig. 18) in adopting blockchain technology are not technical; rather, they relate

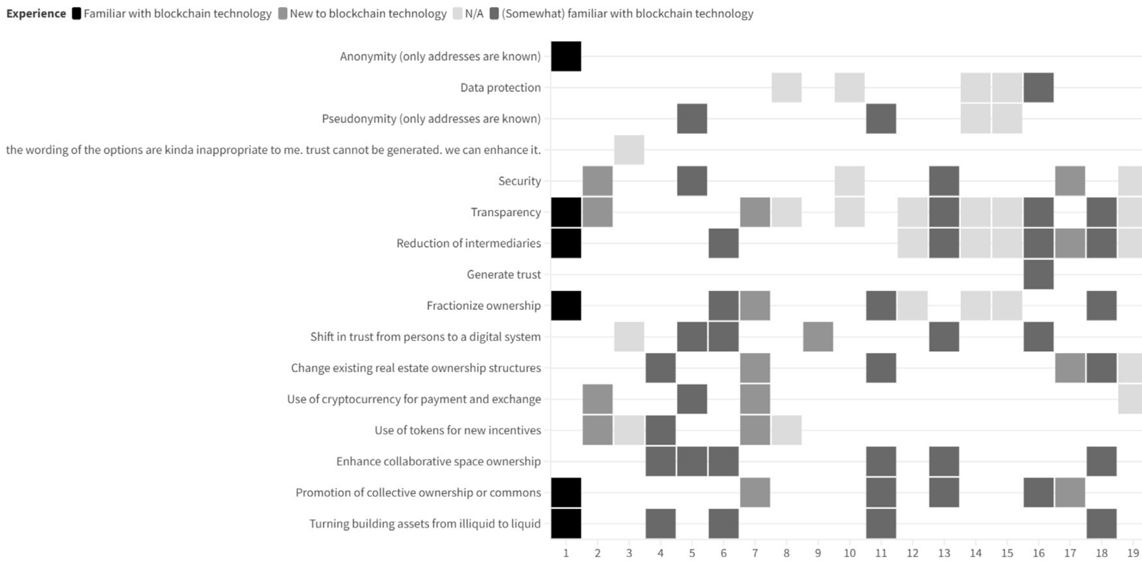

Fig. 17. Blockchain technology's benefits for autonomous buildings.

to user literacy in technical tools (11 selections), market volatility of cryptocurrencies (10 selections), and legal uncertainties (10 selections). Technical concerns, such as the adaptability of smart contracts and the maturity of technology, rank as secondary challenges with 7 and 8 selections.

8. Discussion

8.1. Moving from automation to agency

The growing discourse of human-building interaction suggests rethinking of the role of buildings and non-human entities not only as passive players but as active participants in the built environment. This becomes particularly relevant due to the rapid development of artificial intelligence, enabling a form of "building cognition" [6,10,116–118]. These existing studies form an early basis to move towards an "agentic" understanding of intelligent buildings as autonomous cyber–physical agents, in contrast to previous decades of work focused on building automation logic that serves as a passive recipient of human interaction [5]. A crucial aspect missing from this discussion is a clear understanding of the full economic self-agency and "self-ownership" of these agents. Without it, future discourse is at risk of being irrelevant or focused narrowly on human-controlled scenarios. This paper uniquely explores how it is possible to also grant buildings ownership autonomy via self-custodial blockchain-based financial assets.

One key contribution in this shift is the creation of a tangible artifact. The no1s1 prototype garnered significant attention due to its tangibility, as most blockchain research lacks systematic physical implementations [119]. Furthermore, the process of creating the physical artifact also facilitates a deeper understanding of the matter and raised new questions from participants. For example, what is the relationship between automation and ownership? To what extent do the machine agents of intelligent building possess decision-making rights? These questions are rarely asked in built environment research beyond property rights and ownership management [120]; when raised, it is often in the context of digital governance [79] or user ownership [121], without considering machine self-agency. Understanding machine self-agency and its impact on humans is likely an essential next step in the future of both human-building interaction and more generally human-machine collaboration in the built environment [8,122].

Furthermore, this paper offers a potential technical foundation for future research discourse on human-building interaction by demonstrating, which could be iterated upon to show how smart and intelligent buildings can co-create value and eventually manage assets alongside users. The technical framework (see Fig. 4) illustrates the relationship between the various components required for technical implementation of self-owning physical objects. In the case of no1s1, the cyber–physical system needs a web-based front-end to achieve (1) digital user interaction, and (2) the physical response from/to the back-end and other technical infrastructure. The chosen interaction successfully facilitates the coordination of intelligent buildings with humans without intermediaries, shifting the required trust related to counterparty risk with non-human agents to the technical peer-to-peer blockchain network.

8.2. Research limitations

This paper explores the possibility of establishing a self-owning system by constructing and iterating the artifact using the design science methodology. To ensure the viability of building a practical artifact, it is essential to identify key requirements that limit the scope to only the critical components while allowing for future expansion and full implementation. Furthermore, this paper emphasizes developing overarching concepts and frameworks rather than optimizing system efficiency or technical configurations, and therefore did not aim to gather data or metrics for improving efficiency or enabling meaningful comparisons with technically optimal self-owning intelligent buildings. As a result, the research design faces two main types of limitations and challenges: those specific to the artifact (Sections 8.2.1, 8.2.2, and 8.2.3) and those general to the study (Section 8.2.4). Together, these limitations and challenges highlight critical areas not only for technical optimization of the artifact but also for broader implementation challenges, both of which are essential for advancing toward a fully autonomous, self-owning intelligent building system at scale.

8.2.1. Scope limitations

The no1s1 artifact was well received according to the user feedbacks and it demonstrated the feasibility of a small-scale, self-owned building. However, implementing a fully autonomous, self-owning building system is significantly more complex and challenging at scale and throughout the building's entire life cycle. First, the artifact suffers from

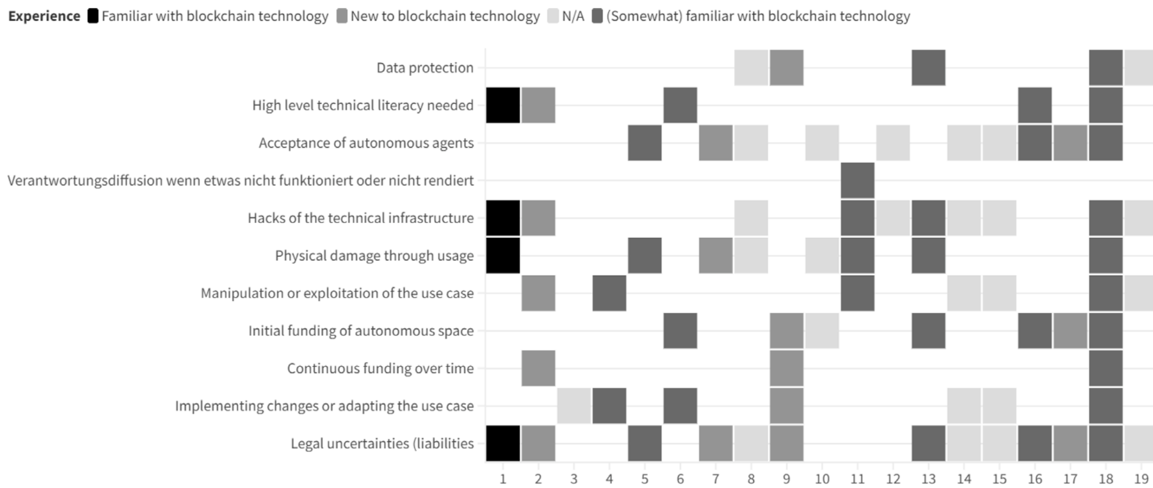

Fig. 18. Blockchain technology's challenges for autonomous buildings.

scope constraints. The envisioned scope of autonomy (shown in Fig. 1) is only partially achieved, raising concerns about its scalability to large-structure buildings. Beyond the limited scope of autonomy, the prototype's design does not account for the entire life cycle or the changing stakeholder groups throughout different stages.

8.2.2. Technical limitations

Due to resource and time constraints, the prototype also had to make technology and tooling compromises that prioritized the practicality of demonstrating the defined key system requirements. Moving forward, technical system optimization of the artifact for real-world implementation should be prioritized. For example, the choice of blockchain network will significantly affect the system's overall technical efficiency, latency, and security, and should be compared and discussed in detail for future implementation. Observation from this prototype indicates that cost-efficiency analysis is critical when scaling this system for real-world application due to the volatility of cryptocurrency prices and the potential high transaction fees. However, cost-effective analysis is complex and requires optimization in multiple aspects, including smart contracts, blockchain network selection, system logic, and on-chain data volume.

To conclude, the following technical research directions can be identified to improve the system technical performance: (1) comparing different blockchain types—private, public, and consortium—for various functionalities; (2) evaluating communication protocols, IoT system, data schema and processing; (3) conducting cost-efficiency analyses for transaction gas fees; (4) developing a true autonomous key management system for autonomous non-human entities; and (5) assessing system cybersecurity risks.

8.2.3. Testing set-up and environment limitations

Frequent prototype relocation was time-consuming due to disassembly and reassembly; future research should enhance transportability. Although three UAT cases and a demonstration (see Sections 6.1.5 and 6.2) were performed to demonstrate the success of the concept, the number of user interactions still can hardly support a statistically relevant analysis. Users were also often highly educated, familiar with the technology, and given a tutorial to help them navigate the process.

Although the conditions of the test sites were as close to real-world application as possible, the researchers were constantly monitoring the process to guarantee the system was working, so it was not fully evaluated in an environment completely without human assistance. Additionally, test sites were carefully selected with secure, guarded areas, power and connectivity. This prevents a more comprehensive assessment of the scalability and robustness of the prototype's full autonomy in random locations.

8.2.4. General challenges

The level of decentralization should be evaluated for the whole process, considering both the technical and human factors. For example, the current implementation relies on a middle-ware layer based on a Raspberry Pi as a back-end (see Section 5.5), which processes and formats the sensor data before transacting it to the blockchain. The middle-ware layer can be viewed also as a single point of failure. Its failure would bring no1s1 to a halt. Multiple network devices would increase security through decentralization and redundancy. Beyond technical level of decentralization, the management and ownership decentralization need to be considered in real-world human-machine interactions.

Scalability also remains a significant challenge in the system design context. Scaling up the system would for example lead to a higher transaction volume on the public blockchain due to more stakeholders and increased operational complexity. The system may then face bottlenecks in transaction processing times and high transaction costs. To address these challenges, future research could explore the integration of layer-2 blockchain scaling solutions [123,124].

Scaling the system would also increase the number of sensors and actuators required to provide the necessary feedback to the smart contract about the physical state. Therefore, ensuring standardized, synchronized, and integrated data streams would become increasingly complex. Without proper standardization, discrepancies in data formats or synchronization issues could compromise the system's functionality and autonomous decision-making feedback loops. Future iterations of the system should explore the adoption of standardized data models and IoT protocols to ensure seamless interoperability. Additionally, implementing advanced data integration techniques, such as semantic data models or machine learning algorithms, could further enhance the system's ability to process and utilize sensor data effectively, ensuring reliable and accurate operations in a blockchain-enabled environment.

There are already inherent cybersecurity risks associated with the use of public blockchain systems, which would also multiply as system interactions scale. Risks include data transparency of sensitive information, potential smart contract exploits, and susceptibility to network instability or attacks such as Sybil attacks. The security measures implemented in the prototype's smart contracts, such as separating logic and data into multiple interacting contracts, ensured successful protection during testing. (see Section 5.5). More research will be required on robust smart contract auditing processes.

The above limitations provide a starting point for implementing a more advanced system at scale. Continuous innovation and adaptation to the evolving technology landscape will be required to improve the potential limitations associated with the current state of blockchain technology.

8.3. Future research directions

This section focuses on the high-level future research directions for self-owning buildings in the built environment. The findings of this paper advocate for a paradigm shift in the perception of ownership within the built environment. The following chapters outline the implications and suggest future research directions following the three main objectives of this paper (see Section 2.5). Future studies could expand beyond these objectives, integrating insights from other disciplines such as social science [48,125], economics and finance [126–128], AEC management [70,129,130], value engineering [49,131,132], digital twin [56,118], urban planning and governance [68,133,134], organizational science [135,136], law [35,86,137], and real estate [73,138,139].

8.3.1. Cyber-physical integrated agents and systems

Cyber-physical systems face increasing complexity, especially regarding their interface with human and social issues [1,114]. These issues have inspired research on cyber–physical-human systems and cyber–physical-social systems [2]. For example, for the proof-of-work mechanism, physical infrastructure is at the core of computing power to maintain blockchains decentralized network [29]. The trend of integrated cyber–physical applications with incentive mechanisms is often referred to as DePin (Decentralized Physical Infrastructure) [140]. In addition, new research linking intelligent agents to DePin has created theories around intelligent cyber–physical agents [141]. Even though limited in its scope, this paper pioneers the integration of the disciplines of cyber–physical systems, smart buildings, and blockchain technology in the built environment and hopefully inspires further research along these lines.

8.3.2. Automation, ownership and self-agency

Automation in the built environment is closely tied to ownership due to the embedded user engagement and coordination involved. It is critical that researchers and engineers become aware that technical design decisions can directly alter the dynamics of decision rights and ownership between human and machine agents. This relationship is reciprocal, dynamic, and complex, and requires close examination and analysis [98].

For future studies, the above quest should be considered in the context of advances in AI. The agent-hood of AI and its moral and ethical dimension have long been critically explored [142,143]. Scientists seek to define, describe, and evaluate the value and self-agency of technology relative to its human counterparts [144–146]. Establishing clear ownership mechanisms for agents will become increasingly important as we move to more complex, interconnected systems where machine agency is not just an extension of human control, but a co-dependent relationship. This presents a unique opportunity for studies in the built environment, as human and machine agents often coexist in the same digital and physical environment.

8.3.3. Peer-to-peer coordination and digital governance

Blockchain-based digital governance can be managed in a unique transparent manner that aligns with principles of common pool resources [51,147–149]. The data recorded on-chain could be the source of truth to assist the coordination of human and machine agents in cyber–physical systems [150]. Drawing also on experiments in decentralized autonomous organizations [43], the potential lies in further exploring how on-chain coordination mechanisms could regulate the interactions and transactions of machine and human agents at scale. Further large-scale experiments in the built environment would certainly provide interesting research insights for peer-to-peer digital governance of networked human and machine agents in the built environment.

9. Conclusion

This paper highlighted how blockchain technology can enable self-agency in intelligent buildings, allowing them to achieve financial ownership. By utilizing DSR methodology and introducing a physical artifact, no1s1, the feasibility of this concept and the ability of intelligent buildings to participate in the digital economy is highlighted. The successful implementation demonstrates the technical feasibility of creating self-agency of things in the built environment, disrupting traditional models of ownership and governance, and confirming theoretical work outlining this potential shift towards new economic systems based on decentralized peer-to-peer participation. This paper emphasizes the potential of re-engineering embedded ownership systems within smart and intelligent buildings using peer-to-peer networks, offering new pathways for blockchain applications for the future of intelligent built environment. Notably, the no1s1 artifact intends to explore a new model of asset ownership and resource distribution that incorporates both human and machine agents in the built environment, in the hope of a positive symbiosis and collaborative future. It also enriches the discourse of autonomous and cyber–physical systems in the built environment with a focus on the impact of ownership on human, social, and economic aspects. Future research should expand interdisciplinary studies to include social sciences, economics, law, management, and governance to better understand autonomous buildings as active agents. Investigating machine self-agency, AI ethics, and decision-making mechanism is also crucial to fostering scalable human-machine collaboration in the built environment. Last but not least, the discourse and technical framework of blockchain-based engineered ownership can be applied across different cyber–physical systems in other domains such as manufacturing.

CRediT authorship contribution statement

Hongyang Wang: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Data curation, Conceptualization. **Jens J. Hunhevicz:** Writing – review & editing, Visualization, Supervision, Software. **Daniel M. Hall:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors gratefully acknowledge the financial support from ETH Research Grants for this project. The human experiment received approval from the ETH Zurich Ethics Commission, whose guidance was invaluable throughout. Gratitudes also go to the ETH Pavilion at the World Economic Forum, House of Electronic Arts Basel, and UZH Blockchain Center for hosting prototype testing and feedback collection. The authors thank Dezentrum for conceiving the idea and name of no1s1 and for their generous resource support in building the physical prototype. Appreciation is also extended to ETH Student Project House for hosting and maintaining the prototype during its development. The authors are grateful for the constructive feedback from the anonymous reviewers and the editor, which improved the manuscript. Finally, special thanks to Ranjith Kuttantharappel Soman, Alexander Nikolas Walzer, Florian Spychiger, and Philip Blaser for their invaluable help with prototype preparation and manuscript development.

Appendix

A.1. Smart contract pseudocode and interactions

The algorithms show in pseudocode the smart contract space access logic of the five main functions mentioned in Fig. 7. The complete code is available on Github at the URL https://github.com/no1s1labs/no1s1.

Algorithm 1 Smart contract function that allows a user to purchase meditation time.

```
function Buy(selectedDuration, txSender, username, escrowAmount, maxDuration, goodDuration, lowDuration)
    if selectedDuration > maxDuration then
        Throw Error: "Requested duration exceeds maximum allowed duration."
    end if
    if value of funds sent < escrowAmount then
        Throw Error: "Insufficient funds provided for escrow."
    end if
    if selectedDuration ≥ goodDuration then
        Ensure battery level is Full.
    else if selectedDuration ≥ lowDuration then
        Ensure battery level is Full or Good.
    else if selectedDuration > 0 then
        Ensure battery level is Full, Good, or Low.
    end if
    key ← HASH(txSender, username)
    if key exists in no1s1Users with non-zero boughtDuration then
        Throw Error: "User has already purchased meditation time."
    end if
    Store no1s1Users[key] with:
      boughtDuration ← selectedDuration
      accessed ← FALSE
      actualDuration ← 0
      left ← FALSE
      paidEscrow ← amount of funds sent
    Increase escrowBalance by the amount of funds sent.
    Emit newQRcode(key).
end function
```

Algorithm 2 Smart contract function that allows `no1s1` to check user access for a scanned key.

```
function CheckAccess(key, goodDuration, lowDuration)
    allowedDuration ← no1s1Users[key].boughtDuration
    if allowedDuration = 0 then
        Throw Error: "No meditation time purchased for this key."
    end if
    if allowedDuration ≥ goodDuration then
        Ensure battery level is Full.
    else if allowedDuration ≥ lowDuration then
        Ensure battery level is Full or Good.
    else if allowedDuration > 0 then
        Ensure battery level is Full, Good, or Low.
    end if
    Update no1s1Occupation ← FALSE.
    Emit accessSucceeded(allowedDuration).
end function
```

Algorithm 3 Smart contract function that allows `no1s1` to record user presence on-chain upon detection.

```
function CheckActivity(pressureDetected, key)
    if no1s1Occupation = TRUE then
        Throw Error: "Space is not occupied, access not checked."
    end if
    if no1s1Users[key].accessed = TRUE then
        Throw Error: "Access has already been registered for this user."
    end if
    if pressureDetected = TRUE then
        no1s1Users[key].accessed ← TRUE
        Emit userActive(TRUE).
    else
        Update no1s1Occupation ← TRUE
        Emit userActive(FALSE).
    end if
end function
```

Algorithm 4 Smart contract function that allows `no1s1` to record user exit upon re-opening the door.

```
function Exit(doorOpened, actualDuration, key)
    if no1s1Users[key].accessed = FALSE then
        Throw Error: "User has not accessed the space."
    end if
    if doorOpened = FALSE then
        Throw Error: "User has not left the space."
    end if
    Update no1s1Occupation ← TRUE.
    Increment counterUsers ← counterUsers + 1.
    Add actualDuration to counterDuration.
    Update no1s1Users[key] with:
      actualDuration ← actualDuration
      left ← TRUE
    Emit exitSuccessful(actualDuration).
end function
```

Algorithm 5 Smart contract function that allows the user to refund unused funds paid.

```
function RefundEscrow(sender, username, meditationPrice)
    key ← HASH(sender, username)
    if no1s1Users[key].accessed = FALSE then
        Throw Error: "User has not accessed the space."
    end if
    if no1s1Users[key].left = FALSE then
        Throw Error: "User has not left the space."
    end if
    actualDuration = no1s1Users[key].actualDuration
    price = actualDuration × meditationPrice
    escrow = no1s1Users[key].paidEscrow
    amountToReturn = escrow − price
    Remove escrow from escrowBalance.
    Reset no1s1Users[key].
    if price ≥ escrow then
        amountToReturn = 0
    else
        Send amountToReturn to sender.
    end if
    Emit refundSuccessful(price, amountToReturn).
end function
```

Data availability

Source code available at https://github.com/no1s1labs/no1s1.

References

[1] C.J. Anumba, N. Roofigari-Esfahan (Eds.), Cyber-Physical Systems in the Built Environment, Springer International Publishing, Cham, 2020, http://dx.doi.org/10.1007/978-3-030-41560-0, ISBN: 978-3-030-41559-4 978-3-030-41560-0. URL: http://link.springer.com/10.1007/978-3-030-41560-0.

[2] P. Li, Y. Lu, D. Yan, J. Xiao, H. Wu, Scientometric mapping of smart building research: Towards a framework of human-cyber-physical system (HCPS), Autom. Constr. 129 (2021) http://dx.doi.org/10.1016/j.autcon.2021.103776, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580521002272.

[3] E.I. Batov, The distinctive features of "smart" buildings, Procedia Eng. 111 (2015) 103–107, http://dx.doi.org/10.1016/j.proeng.2015.07.061, URL: https://linkinghub.elsevier.com/retrieve/pii/S1877705815013107.

[4] A. Buckman, M. Mayfield, S. B.M. Beck, What is a smart building? Smart Sustain. Built Environ. 3 (2) (2014) 92–109, http://dx.doi.org/10.1108/SASBE-01-2014-0003, URL: https://www.emerald.com/insight/content/doi/10.1108/SASBE-01-2014-0003/full/html.

[5] T. Derek, J. Clements-Croome, What do we mean by intelligent buildings? Autom. Constr. 6 (5–6) (1997) 395–400, http://dx.doi.org/10.1016/S0926-5805(97)00018-6, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580597000186.

[6] M. Manic, K. Amarasinghe, J.J. Rodriguez-Andina, C. Rieger, Intelligent buildings of the future: Cyberaware, deep learning powered, and human interacting, IEEE Ind. Electron. Mag. 10 (4) (2016) 32–49, http://dx.doi.org/10.1109/MIE.2016.2615575, Conference Name: IEEE Industrial Electronics Magazine.

[7] G. Ramanathan, Autonomous buildings, in: Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, ACM, Coimbra Portugal, ISBN: 978-1-4503-9114-6, 2021, pp. 246–247, http://dx.doi.org/10.1145/3486611.3492388, URL: https://dl.acm.org/doi/10.1145/3486611.3492388.

[8] H.S. Alavi, E.F. Churchill, M. Wiberg, D. Lalanne, P. Dalsgaard, A. Fatah gen Schieck, Y. Rogers, Introduction to human-building interaction (HBI): Interfacing HCI with architecture and urban design, ACM Trans. Comput.- Hum. Interact. 26 (2) (2019) 1–10, http://dx.doi.org/10.1145/3309714, URL: https://dl.acm.org/doi/10.1145/3309714.

[9] B. Becerik-Gerber, G. Lucas, A. Aryal, M. Awada, M. Bergés, S.L. Billington, O. Boric-Lubecke, A. Ghahramani, A. Heydarian, F. Jazizadeh, R. Liu, R. Zhu, F. Marks, S. Roll, M. Seyedrezaei, J.E. Taylor, C. Höelscher, A. Khan, J. Langevin, M.L. Mauriello, E. Murnane, H. Noh, M. Pritoni, D. Schaumann, J. Zhao, Ten questions concerning human-building interaction research for improving the quality of life, Build. Environ. 226 (2022) http://dx.doi.org/10.1016/j.buildenv.2022.109681, URL: https://linkinghub.elsevier.com/retrieve/pii/S0360132322009118.

[10] F. Mofidi, H. Akbari, Intelligent buildings: An overview, Energy Build. 223 (2020) http://dx.doi.org/10.1016/j.enbuild.2020.110192, URL: https://linkinghub.elsevier.com/retrieve/pii/S0378778819338289.

[11] D. Clements-Croome (Ed.), Intelligent Buildings: Design, Management and Operation, Thomas Telford ; ASCE Press, distributor, London : Reston, VA, ISBN: 978-0-7277-3266-8, 2004.

[12] T. McConaghy, Nature 2.0. the cradle of civilization gets an upgrade, 2018, URL: https://blog.oceanprotocol.com/nature-2-0-27bdf8238071. (Accessed 19 April 2025).

[13] A. Kaklauskas, E.K. Zavadskas, J. Naimavicienė, M. Krutinis, V. Plakys, D. Venskus, Model for a complex analysis of intelligent built environment, Autom. Constr. 19 (3) (2010) 326–340, http://dx.doi.org/10.1016/j.autcon.2009.12.006, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580509001952.

[14] A. Espinosa, R. Harnden, J. Walker, A complexity approach to sustainability – stafford beer revisited, European J. Oper. Res. 187 (2) (2008) 636–651, http://dx.doi.org/10.1016/j.ejor.2007.03.023, URL: https://linkinghub.elsevier.com/retrieve/pii/S0377221707003591.

[15] W.M. Kroner, An intelligent and responsive architecture, Autom. Constr. 6 (5–6) (1997) 381–393, http://dx.doi.org/10.1016/S0926-5805(97)00017-4, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580597000174.

[16] P. Medici, Autonomous houses and architecture of cybernetics in the 1970s: Towards limits and undeveloped potentials of the sustainable, Sustainability 14 (10) (2022) http://dx.doi.org/10.3390/su14106073, URL: https://www.mdpi.com/2071-1050/14/10/6073.

[17] A. Kamariotis, E. Chatzi, D. Straub, A framework for quantifying the value of vibration-based structural health monitoring, Mech. Syst. Signal Process. 184 (2023) http://dx.doi.org/10.1016/j.ymssp.2022.109708, URL: https://www.sciencedirect.com/science/article/pii/S0888327022007841.

[18] N. De Belie, E. Gruyaert, A. Al-Tabbaa, P. Antonaci, C. Baera, D. Bajare, A. Darquennes, R. Davies, L. Ferrara, T. Jefferson, C. Litina, B. Miljevic, A. Otlewska, J. Ranogajec, M. Roig-Flores, K. Paine, P. Lukowski, P. Serna, J.-M. Tulliani, S. Vucetic, J. Wang, H.M. Jonkers, A review of self-healing concrete for damage management of structures, Adv. Mater. Interfaces 5 (17) (2018) http://dx.doi.org/10.1002/admi.201800074, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/admi.201800074. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/admi.201800074.

[19] L. Gurgen, O. Gunalp, Y. Benazzouz, M. Gallissot, Self-aware cyber-physical systems and applications in smart buildings and cities, in: 2013 Design, Automation & Test in Europe Conference & Exhibition, DATE, 2013, pp. 1149–1154, http://dx.doi.org/10.7873/DATE.2013.240, (ISSN: 1530-1591).

[20] K. Nabben, H. Wang, M. Zargham, Decentralised governance for autonomous cyber-physical systems, 2024, arXiv:2407.13566. URL: https://arxiv.org/pdf/2407.13566.

[21] N. Aste, M. Manfren, G. Marenzi, Building automation and control systems and performance optimization: A framework for analysis, Renew. Sustain. Energy Rev. 75 (2017) 313–330, http://dx.doi.org/10.1016/j.rser.2016.10.072, URL: https://www.sciencedirect.com/science/article/pii/S1364032116307365.

[22] T. O'Grady, H.-Y. Chong, G.M. Morrison, A systematic review and meta-analysis of building automation systems, Build. Environ. 195 (2021) http://dx.doi.org/10.1016/j.buildenv.2021.107770, URL: https://www.sciencedirect.com/science/article/pii/S0360132321001773.

[23] G. Lilis, G. Conus, N. Asadi, M. Kayal, Towards the next generation of intelligent building: An assessment study of current automation and future IoT based systems with a proposal for transitional design, Sustain. Cities Soc. 28 (2017) 473–481, http://dx.doi.org/10.1016/j.scs.2016.08.019, URL: https://www.sciencedirect.com/science/article/pii/S2210670716302414.

[24] E.N. Farsangi, M. Noori, T.T.Y. Yang, P.B. Lourenço, P. Gardoni, I. Takewaki, E. Chatzi, S. Li, Automation in Construction toward Resilience: Robotics, Smart Materials and Intelligent Systems, CRC Press, ISBN: 978-1-00-091290-6, 2023.

[25] D. Helbing, F. Fanitabasi, F. Giannotti, R. Hänggli, C.I. Hausladen, J. van den Hoven, S. Mahajan, D. Pedreschi, E. Pournaras, Ethics of smart cities: Towards value-sensitive design and co-evolving city life, Sustainability 13 (20) (2021) http://dx.doi.org/10.3390/su132011162, URL: https://www.mdpi.com/2071-1050/13/20/11162.

[26] B. Friedman, Human Values and the Design of Computer Technology, Cambridge University Press, ISBN: 978-1-57586-081-7, 1997.

[27] A. Wright, P. De Filippi, Decentralized blockchain technology and the rise of lex cryptographia, SSRN Electron. J. (2015) http://dx.doi.org/10.2139/ssrn.2580664, URL: https://www.ssrn.com/abstract=2580664.

[28] Z. Zheng, S. Xie, H. Dai, X. Chen, H. Wang, An overview of blockchain technology: Architecture, consensus, and future trends, in: 2017 IEEE International Congress on Big Data, BigData Congress, 2017, pp. 557–564, http://dx.doi.org/10.1109/BigDataCongress.2017.85.

[29] S. Nakamoto, Bitcoin: A peer-to-peer electronic cash system, 2008, URL: https://bitcoin.org/bitcoin.pdf. (Accessed 18 April 2025).

[30] V. Buterin, Ethereum white paper, 2014, URL: https://static.peng37.com/ethereum_whitepaper_laptop_3.pdf. (Accessed 17 April 2025).

[31] N. Szabo, Smart contracts: Building blocks for digital markets, 1996, URL: https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html. (Accessed 31 January 2022).

[32] G. Ishmaev, Blockchain technology as an institution of property, Metaphilosophy 48 (5) (2017) 666–686, http://dx.doi.org/10.1111/meta.12277, URL: https://onlinelibrary.wiley.com/doi/10.1111/meta.12277.

[33] Y. Teng, Towards trustworthy blockchains: normative reflections on blockchain-enabled virtual institutions, Ethics Inf. Technol. 23 (3) (2021) 385–397, http://dx.doi.org/10.1007/s10676-021-09581-3, URL: https://link.springer.com/10.1007/s10676-021-09581-3.

[34] D.W.E. Allen, C. Berg, A.M. Lane, J. Potts, Cryptodemocracy and its institutional possibilities, Rev. Austrian Econ. 33 (3) (2020) 363–374, http://dx.doi.org/10.1007/s11138-018-0423-6, URL: http://link.springer.com/10.1007/s11138-018-0423-6.

[35] S. Hassan, P. De Filippi, The Expansion of Algorithmic Governance: From Code Is Law to Law Is Code, SSRN Scholarly Paper, Social Science Research Network, Rochester, NY, 2017, URL: https://papers.ssrn.com/abstract=3117630.

[36] S. Basly (Ed.), Decentralized Finance: The Impact of Blockchain-Based Financial Innovations on Entrepreneurship, in: Financial Innovation and Technology, Springer International Publishing, Cham, 2024, http://dx.doi.org/10.1007/978-3-031-49515-1, ISBN: 978-3-031-49514-4 978-3-031-49515-1. URL: https://link.springer.com/10.1007/978-3-031-49515-1.

[37] P. De Filippi, Bitcoin: a regulatory nightmare to a libertarian dream, Internet Policy Rev. 3 (2) (2014) http://dx.doi.org/10.14763/2014.2.286, URL: https://policyreview.info/node/286.

[38] Q. Wang, R. Li, Q. Wang, S. Chen, Non-fungible token (NFT): Overview, evaluation, opportunities and challenges, 2021, URL: http://arxiv.org/abs/2105.07447. arXiv:2105.07447 [cs]. (Accessed 31 January 2024).

[39] A.E. Fezzazi, A. Adadi, M. Berrada, Towards a blockchain based intelligent and secure voting, in: 2021 Fifth International Conference on Intelligent Computing in Data Sciences, ICDS, IEEE, Fez, Morocco, ISBN: 978-1-66544-238-1, 2021, pp. 1–8, http://dx.doi.org/10.1109/ICDS53782.2021.9626751, URL: https://ieeexplore.ieee.org/document/9626751/.

[40] G. Sun, M. Dai, J. Sun, H. Yu, Voting-based decentralized consensus design for improving the efficiency and security of consortium blockchain, IEEE Internet Things J. 8 (8) (2021) 6257–6272, http://dx.doi.org/10.1109/JIOT.2020.3029781, URL: https://ieeexplore.ieee.org/document/9219122/.

[41] O. Dib, K. Toumi, Decentralized identity systems: Architecture, challenges, solutions and future directions, Ann. Emerg. Technol. Comput. 4 (5) (2020) 19–40, http://dx.doi.org/10.33166/AETiC.2020.05.002, URL: http://aetic.theiaer.org/archive/v4/v4n5/p2.html.

[42] F. Wang, P. De Filippi, Self-sovereign identity in a globalized world: Credentials-based identity systems as a driver for economic inclusion, Front. Blockchain 2 (2020) http://dx.doi.org/10.3389/fbloc.2019.00028, URL: https://www.frontiersin.org/article/10.3389/fbloc.2019.00028/full.

[43] V. Buterin, DAOs, DACs, DAs and more: An incomplete terminology guide, 2014, URL: https://blog.ethereum.org/2014/05/06/daos-dacs-das-and-more-an-incomplete-terminology-guide/.

[44] N. Diallo, W. Shi, L. Xu, Z. Gao, L. Chen, Y. Lu, N. Shah, L. Carranco, T.-C. Le, A.B. Surez, G. Turner, Egov-DAO: a better government using blockchain based decentralized autonomous organization, in: 2018 International Conference on EDemocracy & EGovernment, ICEDEG, IEEE, Ambato, ISBN: 978-1-5386-2521-7, 2018, pp. 166–171, http://dx.doi.org/10.1109/ICEDEG.2018.8372356, URL: https://ieeexplore.ieee.org/document/8372356/.

[45] P. Van Vulpen, S. Jansen, Decentralized autonomous organization design for the commons and the common good, Front. Blockchain 6 (2023) http://dx.doi.org/10.3389/fbloc.2023.1287249, URL: https://www.frontiersin.org/articles/10.3389/fbloc.2023.1287249/full.

[46] S. Davidson, P. De Filippi, J. Potts, Disrupting governance: The new institutional economics of distributed ledger technology, SSRN Electron. J. (2016) http://dx.doi.org/10.2139/ssrn.2811995, URL: https://www.ssrn.com/abstract=2811995.

[47] S. Davidson, P. De Filippi, J. Potts, Blockchains and the economic institutions of capitalism, J. Inst. Econ. 14 (4) (2018) 639–658, http://dx.doi.org/10.1017/S1744137417000200, URL: https://www.cambridge.org/core/product/identifier/S1744137417000200/type/journal_article.

[48] M.C. Ballandies, M.M. Dapp, B.A. Degenhart, D. Helbing, Finance 4.0: Design principles for a value-sensitive cryptoecnomic system to address sustainability, 2021, http://dx.doi.org/10.48550/arXiv.2105.11955, URL: http://arxiv.org/abs/2105.11955. arXiv:2105.11955 [cs]. (Accessed 30 April 2023).

[49] M.C. Ballandies, To incentivize or not: Impact of blockchain-based cryptoeconomic tokens on human information sharing behavior, IEEE Access 10 (2022) 74111–74130, http://dx.doi.org/10.1109/ACCESS.2022.3189774, URL: http://arxiv.org/abs/2206.03221. arXiv:2206.03221 [cs].

[50] D. Rozas, A. Tenorio-Fornés, S. Díaz-Molina, S. Hassan, When ostrom meets blockchain: Exploring the potentials of blockchain for commons governance, SAGE Open 11 (1) (2021) 1–14, http://dx.doi.org/10.1177/21582440211002526.

[51] F. Fritsch, J. Emmett, E. Friedman, R. Kranjc, S. Manski, M. Zargham, M. Bauwens, Challenges and approaches to scaling the global commons, Front. Blockchain 4 (2021) http://dx.doi.org/10.3389/fbloc.2021.578721, URL: https://www.frontiersin.org/articles/10.3389/fbloc.2021.578721/full.

[52] D. Wright, Quadratic voting and blockchain governance, 2019, URL: https://irlaw.umkc.edu/faculty_works/723. (Accessed 19 April 2025).

[53] J.J. Hunhevicz, M. Motie, D.M. Hall, Digital building twins and blockchain for performance-based (smart) contracts, Autom. Constr. 133 (2022) http://dx.doi.org/10.1016/j.autcon.2021.103981, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580521004325.

[54] D. Lee, S.H. Lee, N. Masoud, M. Krishnan, V.C. Li, Integrated digital twin and blockchain framework to support accountable information sharing in construction projects, Autom. Constr. 127 (2021) http://dx.doi.org/10.1016/j.autcon.2021.103688, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580521001394.

[55] H. Gao, B. Zhong, L. Ding, A blockchain-based engineering design review service trading scheme for digital building permits, Autom. Constr. 165 (2024) http://dx.doi.org/10.1016/j.autcon.2024.105496, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580524002322.

[56] H. Naderi, A. Shojaei, Digital twin non-fungible token (DT-NFT): Enabling data ownership in the AEC industry, Autom. Constr. 168 (2024) http://dx.doi.org/10.1016/j.autcon.2024.105777, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580524005132.

[57] M. Heydari, A. Shojaei, Blockchain applications in the construction supply chain, Autom. Constr. 171 (2025) http://dx.doi.org/10.1016/j.autcon.2025.105998, URL: https://linkinghub.elsevier.com/retrieve/pii/S092658052500038X.

[58] X. Yan, X. Liao, S. Cheng, T. Wang, BIM-focused incentive-driven adoption of information management technology in bridge construction, Autom. Constr. 172 (2025) http://dx.doi.org/10.1016/j.autcon.2025.106052, URL: https://www.sciencedirect.com/science/article/pii/S0926580525000925.

[59] C.G. Schmidt, S.M. Wagner, Blockchain and supply chain relations: A transaction cost theory perspective, J. Purch. Supply Manag. 25 (4) (2019) http://dx.doi.org/10.1016/j.pursup.2019.100552, URL: https://linkinghub.elsevier.com/retrieve/pii/S1478409218301298.

[60] A. Widyasningrum, Blockchain for a Circular Economy. An Explorative Research Towards The Feasibility of Blockchain in Enhancing The Implemented Material Passport and Life Cycle Assessment Methodology Within Construction Sector in the Netherlands (Master's thesis), Universitas Islam Indonesia, 2021, URL: https://dspace.uii.ac.id/handle/123456789/34055.

[61] A. Dorri, S.S. Kanhere, R. Jurdak, P. Gauravaram, Blockchain for IoT security and privacy: The case study of a smart home, in: 2017 IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops, 2017, pp. 618–623, http://dx.doi.org/10.1109/PERCOMW.2017.7917634.

[62] K. Christidis, M. Devetsikiotis, Blockchains and smart contracts for the internet of things, IEEE Access 4 (2016) 2292–2303, http://dx.doi.org/10.1109/ACCESS.2016.2566339, Conference Name: IEEE Access.

[63] D. Scott, T. Broyd, L. Ma, Conceptual framework of a project bank account (PBA) blockchain payment application for the construction industry, 2022, http://dx.doi.org/10.35490/EC3.2022.188, URL: https://ec-3.org/publications/conference/paper/?id=EC32022_188.

[64] M. Afzal, J. Li, W. Amin, Q. Huang, K. Umer, S.A. Ahmad, F. Ahmad, A. Raza, Role of blockchain technology in transactive energy market: A review, Sustain. Energy Technol. Assess. 53 (2022) http://dx.doi.org/10.1016/j.seta.2022.102646, URL: https://www.sciencedirect.com/science/article/pii/S2213138822006956.

[65] B. Zhou, W. Li, K.W. Chan, Y. Cao, Y. Kuang, X. Liu, X. Wang, Smart home energy management systems: Concept, configurations, and scheduling strategies, Renew. Sustain. Energy Rev. 61 (2016) 30–40, http://dx.doi.org/10.1016/j.rser.2016.03.047, URL: https://www.sciencedirect.com/science/article/pii/S1364032116002823.

[66] C. Zheng, X. Tao, L. Dong, U. Zukaib, J. Tang, H. Zhou, J.C. Cheng, X. Cui, Z. Shen, Decentralized artificial intelligence in construction using blockchain, Autom. Constr. 166 (2024) http://dx.doi.org/10.1016/j.autcon.2024.105669, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580524004059.

[67] Y. Zuo, Exploring the synergy: AI enhancing blockchain, blockchain empowering AI, and their convergence across IoT applications and beyond, IEEE Internet Things J. 12 (6) (2025) 6171–6195, http://dx.doi.org/10.1109/JIOT.2024.3507746, URL: https://ieeexplore.ieee.org/document/10769427/.

[68] I. Gloerich, M. De Waal, G. Ferri, N. Cila, T. Karpinski, The city as a license. implications of blockchain and distributed ledgers for urban governance, Front. Sustain. Cities 2 (2020) http://dx.doi.org/10.3389/frsc.2020.534942, URL: https://www.frontiersin.org/articles/10.3389/frsc.2020.534942/full.

[69] P.F. Wong, F.C. Chia, M.S. Kiu, E.C. Lou, Potential integration of blockchain technology into smart sustainable city (SSC) developments: a systematic review, Smart Sustain. Built Environ. 11 (3) (2022) 559–574, http://dx.doi.org/10.1108/SASBE-09-2020-0140, URL: https://www.emerald.com/insight/content/doi/10.1108/SASBE-09-2020-0140/full/html.

[70] J. Hunhevicz, T. Dounas, D.M. Hall, The promise of blockchain for the construction industry: A governance lens, in: T. Dounas, D. Lombardi (Eds.), Blockchain for Construction, in: Blockchain Technologies, Springer Nature, Singapore, ISBN: 978-981-19375-9-0, 2022, pp. 5–33, http://dx.doi.org/10.1007/978-981-19-3759-0_2.

[71] E. Yu, W. Yue, S. Jianzheng, W. Xun, Blockchain-based AI agent and autonomous world infrastructure, in: 2024 IEEE Conference on Artificial Intelligence, CAI, IEEE, Singapore, Singapore, ISBN: 979-8-3503-5409-6, 2024, pp. 278–283, http://dx.doi.org/10.1109/CAI59869.2024.00061, URL: https://ieeexplore.ieee.org/document/10605324/.

[72] M.-L. Marsal-Llacuna, Future living framework: Is blockchain the next enabling network? Technol. Forecast. Soc. Change 128 (2018) 226–234, http://dx.doi.org/10.1016/j.techfore.2017.12.005, URL: https://linkinghub.elsevier.com/retrieve/pii/S0040162517310727.

[73] L.B. Van Vianen, Solidity-self-owning-building, 2023, URL: https://github.com/block-foundation/solidity-self-owning-building. (Accessed 07 January 2025).

[74] M.J.M. Chowdhury, A. Colman, M.A. Kabir, J. Han, P. Sarda, Blockchain versus database: A critical analysis, in: 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/ 12th IEEE International Conference on Big Data Science and Engineering, TrustCom/BigDataSE, 2018, pp. 1348–1353, http://dx.doi.org/10.1109/TrustCom/BigDataSE.2018.00186, URL: https://ieeexplore.ieee.org/document/8456055/. (ISSN: 2324-9013).

[75] M. Raikwar, D. Gligoroski, G. Velinov, Trends in development of databases and blockchain, in: 2020 Seventh International Conference on Software Defined Systems, SDS, 2020, pp. 177–182, http://dx.doi.org/10.1109/SDS49854.2020.9143893, URL: https://ieeexplore.ieee.org/document/9143893/.

[76] S. Davidson, P. De Filippi, J. Potts, Economics of blockchain, SSRN Electron. J. (2016) http://dx.doi.org/10.2139/ssrn.2744751, URL: http://www.ssrn.com/abstract=2744751.

[77] K. Wüst, A. Gervais, Do you need a blockchain? in: 2018 Crypto Valley Conference on Blockchain Technology, CVCBT, 2018, pp. 45–54, http://dx.doi.org/10.1109/CVCBT.2018.00011.

[78] J.J. Hunhevicz, D.M. Hall, Do you need a blockchain in construction? Use case categories and decision framework for DLT design options, Adv. Eng. Inform. 45 (2020) http://dx.doi.org/10.1016/j.aei.2020.101094, URL: https://linkinghub.elsevier.com/retrieve/pii/S147403462030063X.

[79] A. Ghaffarianhoseini, H. AlWaer, A. Ghaffarianhoseini, D. Clements-Croome, U. Berardi, K. Raahemifar, J. Tookey, Intelligent or smart cities and buildings: a critical exposition and a way forward, Intell. Build. Int. 10 (2) (2018) 122–129, http://dx.doi.org/10.1080/17508975.2017.1394810, URL: https://www.tandfonline.com/doi/full/10.1080/17508975.2017.1394810.

[80] A. Ghaffarianhoseini, U. Berardi, H. AlWaer, S. Chang, E. Halawa, A. Ghaffarianhoseini, D. Clements-Croome, What is an intelligent building? Analysis of recent interpretations from an international perspective, Archit. Sci. Rev. 59 (5) (2016) 338–357, http://dx.doi.org/10.1080/00038628.2015.1079164, URL: http://www.tandfonline.com/doi/full/10.1080/00038628.2015.1079164.

[81] D. Clements-Croome, Sustainable intelligent buildings for people: A review, Intell. Build. Int. 3 (2) (2011) 67–86, http://dx.doi.org/10.1080/17508975.2011.582313, URL: https://www.tandfonline.com/doi/abs/10.1080/17508975.2011.582313. Publisher: Taylor & Francis _eprint: https://www.tandfonline.com/doi/pdf/10.1080/17508975.2011.582313.

[82] F. Topak, M.K. Pekeriçli, Collaborative building control: a conceptual mixed-initiative framework, Intell. Build. Int. 14 (4) (2022) 487–498, http://dx.doi.org/10.1080/17508975.2021.1941731, URL: https://www.tandfonline.com/doi/full/10.1080/17508975.2021.1941731.

[83] D. Ramsauer, M. Dorfmann, H. Tellioğlu, W. Kastner, Human perception and building automation systems, Energies 15 (5) (2022) http://dx.doi.org/10.3390/en15051745, URL: https://www.mdpi.com/1996-1073/15/5/1745. Number: 5 Publisher: Multidisciplinary Digital Publishing Institute.

[84] P. Domingues, P. Carreira, R. Vieira, W. Kastner, Building automation systems: Concepts and technology review, Comput. Stand. Interfaces 45 (2016) 1–12, http://dx.doi.org/10.1016/j.csi.2015.11.005, URL: https://linkinghub.elsevier.com/retrieve/pii/S0920548915001361.

[85] J. Grey, O. Seneviratne, I. Godage, Blockchain-based mechanism for robotic cooperation through incentives: Prototype application in warehouse automation, in: 2021 IEEE International Conference on Blockchain, Blockchain, 2021, pp. 597–604, http://dx.doi.org/10.1109/Blockchain53845.2021.00090.

[86] P.D. Filippi, S. Hassan, Blockchain technology as a regulatory technology: From code is law to law is code, First Monday (2016) http://dx.doi.org/10.5210/fm.v21i12.7113, URL: https://firstmonday.org/ojs/index.php/fm/article/view/7113.

[87] K. Nabben, Imagining human-machine futures: Blockchain-based "decentralized autonomous organizations", SSRN Electron. J. (2021) http://dx.doi.org/10.2139/ssrn.3953623, URL: https://www.ssrn.com/abstract=3953623.

[88] P. De Filippi, Plantoïde: Une forme de vie fondee sur la blockchain, 2020, URL: https://hal.science/hal-03098591/document. (Accessed 10 march 2024).

[89] SNI, Sovereign nature initiative, 2019, URL: https://sovereignnature.com. (Accessed 12 January 2023).

[90] terra0, 2016, URL: https://terra0.org. (Accessed 12 January 2023).

[91] M.R. Endsley, D.B. Kaber, Level of automation effects on performance, situation awareness and workload in a dynamic control task, Ergonomics 42 (3) (1999) 462–492, http://dx.doi.org/10.1080/001401399185595, URL: https://www.tandfonline.com/doi/full/10.1080/001401399185595.

[92] U.S. Department of Transportation, Federal automated vehicles policy, 2016, URL: https://www.transportation.gov/sites/dot.gov/files/docs/AV%20policy%20guidance%20PDF.pdf.

[93] J. Van Brummelen, M. O'Brien, D. Gruyer, H. Najjaran, Autonomous vehicle perception: The technology of today and tomorrow, Transp. Res. Part C: Emerg. Technol. 89 (2018) 384–406, http://dx.doi.org/10.1016/j.trc.2018.02.012, URL: https://linkinghub.elsevier.com/retrieve/pii/S0968090X18302134.

[94] A. Agrawal, R. Thiel, P. Jain, V. Singh, M. Fischer, Digital twin: Where do humans fit in? Autom. Constr. 148 (2023) http://dx.doi.org/10.1016/j.autcon.2023.104749, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580523000092.

[95] A. Thelen, X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu, A comprehensive review of digital twin – part 1: Modeling and twinning enabling technologies, Struct. Multidiscip. Optim. 65 (2022) http://dx.doi.org/10.1007/s00158-022-03425-4, URL: http://arxiv.org/abs/2208.14197.

[96] A. Thelen, X. Zhang, O. Fink, Y. Lu, S. Ghosh, B.D. Youn, M.D. Todd, S. Mahadevan, C. Hu, Z. Hu, A comprehensive review of digital twin—part 2: roles of uncertainty quantification and optimization, a battery digital twin, and perspectives, Struct. Multidiscip. Optim. 66 (1) (2023) http://dx.doi.org/10.1007/s00158-022-03410-x, URL: https://link.springer.com/10.1007/s00158-022-03410-x.

[97] K. Lyu, A. Brambilla, A. Globa, R. De Dear, An immersive multisensory virtual reality approach to the study of human-built environment interactions, Autom. Constr. 150 (2023) http://dx.doi.org/10.1016/j.autcon.2023.104836, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580523000961.

[98] H. Wang, J. Hunhevicz, D. Hall, What if properties are owned by no one or everyone? foundation of blockchain enabled engineered ownership, in: EC3 Conference 2022, European Conference on Computing in Construction, 2022, http://dx.doi.org/10.35490/EC3.2022.213, URL: https://ec-3.org/publications/conference/paper/?id=EC32022_213.

[99] M. Donella, Leverage points, places to intervene in a system, 1999, URL: https://www.donellameadows.org/wp-content/userfiles/Leverage_Points.pdf. (Accessed 17 April 2025).

[100] E. Schlager, E. Ostrom, Property-rights regimes and natural resources: A conceptual analysis, Land Econ. 68 (3) (1992) http://dx.doi.org/10.2307/3146375, URL: http://www.jstor.org/stable/3146375?origin=crossref.

[101] J. Fairfield, Bitproperty, 2015, pp. 805–874, URL: https://scholarlycommons.law.wlu.edu/wlufac/492. (Accessed 18 April 2025).

[102] H.E. Smith, Property as the law of things, 2012, URL: https://papers.ssrn.com/abstract=2012815. (Accessed 28 January 2022).

[103] E. Nagel, J. Kranz, Is 'chained' property the future? A review and synthesis of literature on blockchain technology and property rights, in: ECIS 2021 Research Papers, 2021, URL: https://aisel.aisnet.org/ecis2021_rp/88.

[104] D. Lorenz, P. Dent, T. Kauko, Value in a Changing Built Environment, John Wiley & Sons, ISBN: 978-1-119-07365-9, 2017, Google-Books-ID: Pso9DwAAQBAJ.

[105] R.M. Garcia-Teruel, H. Simón-Moreno, The digital tokenization of property rights. a comparative perspective, Comput. Law Secur. Rev. 41 (2021) http://dx.doi.org/10.1016/j.clsr.2021.105543, URL: https://linkinghub.elsevier.com/retrieve/pii/S0267364921000169.

[106] M.S. Kiu, F.C. Chia, P.F. Wong, Exploring the potentials of blockchain application in construction industry: a systematic review, Int. J. Constr. Manag. 22 (15) (2022) 2931–2940, http://dx.doi.org/10.1080/15623599.2020.1833436, URL: https://www.tandfonline.com/doi/full/10.1080/15623599.2020.1833436.

[107] J.K.W. Wong, J. Zhou, Enhancing environmental sustainability over building life cycles through green BIM: A review, Autom. Constr. 57 (2015) 156–165, http://dx.doi.org/10.1016/j.autcon.2015.06.003, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580515001211.

[108] W. Beitz, Design science—The need for a scientific basis for engineering design methodology, J. Eng. Des. 5 (2) (1994) 129–133, http://dx.doi.org/10.1080/09544829408907878, URL: http://www.tandfonline.com/doi/abs/10.1080/09544829408907878.

[109] J. Holmström, M. Ketokivi, A.-P. Hameri, Bridging practice and theory: A design science approach, Decis. Sci. 40 (1) (2009) 65–87, http://dx.doi.org/10.1111/j.1540-5915.2008.00221.x, URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1540-5915.2008.00221.x.

[110] M. Chu, J. Matthews, P.E. Love, Integrating mobile building information modelling and augmented reality systems: An experimental study, Autom. Constr. 85 (2018) 305–316, http://dx.doi.org/10.1016/j.autcon.2017.10.032, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580517301218.

[111] J. Wong, H. Li, S. Wang, Intelligent building research: a review, Autom. Constr. 14 (1) (2005) 143–159, http://dx.doi.org/10.1016/j.autcon.2004.06.001, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580504000536.

[112] A. Bonci, A. Carbonari, A. Cucchiarelli, L. Messi, M. Pirani, M. Vaccarini, A cyber-physical system approach for building efficiency monitoring, Autom. Constr. 102 (2019) 68–85, http://dx.doi.org/10.1016/j.autcon.2019.02.010, URL: https://www.sciencedirect.com/science/article/pii/S0926580518308793.

[113] X. Guan, B. Yang, C. Chen, W. Dai, Y. Wang, A comprehensive overview of cyber-physical systems: from perspective of feedback system, IEEE/ CAA J. Autom. Sin. 3 (1) (2016) 1–14, http://dx.doi.org/10.1109/JAS.2016.7373757, URL: http://ieeexplore.ieee.org/document/7373757/.

[114] O.C. Madubuike, C.J. Anumba, Potential for the integration of cyber-physical systems with intelligent buildings, in: Construction Research Congress 2020, American Society of Civil Engineers, Tempe, Arizona, ISBN: 978-0-7844-8286-5, 2020, pp. 697–705, http://dx.doi.org/10.1061/9780784482865.074, URL: https://ascelibrary.org/doi/10.1061/9780784482865.074.

[115] E.R. Griffor, C. Greer, D.A. Wollman, M.J. Burns, Framework for Cyber-Physical Systems: Volume 1, Overview, Technical Report NIST SP 1500-201, National Institute of Standards and Technology, Gaithersburg, MD, 2017, http://dx.doi.org/10.6028/NIST.SP.1500-201, URL: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1500-201.pdf.

[116] A. Mostashari, F. Arnold, M. Maurer, J. Wade, Citizens as sensors: The cognitive city paradigm, in: 2011 8th International Conference and Expo on Emerging Technologies for a Smarter World, CEWIT 2011, 2011, http://dx.doi.org/10.1109/CEWIT.2011.6135861.

[117] M.R.A. Berkani, A. Chouchane, Y. Himeur, A. Ouamane, S. Miniaoui, S. Atalla, W. Mansoor, H. Al-Ahmad, Advances in federated learning: Applications and challenges in smart building environments and beyond, Computers 14 (4) (2025) http://dx.doi.org/10.3390/computers14040124, URL: https://www.mdpi.com/2073-431X/14/4/124.

[118] A. Nechesov, I. Dorokhov, J. Ruponen, Virtual cities: From digital twins to autonomous AI societies, IEEE Access 13 (2025) 13866–13903, http://dx.doi.org/10.1109/ACCESS.2025.3531222, URL: https://ieeexplore.ieee.org/document/10844277/.

[119] F. Elghaish, F. Pour Rahimian, T. Brooks, N. Dawood, S. Abrishami, Blockchain of Things and Deep Learning Applications in Construction: Digital Construction Transformation, Springer International Publishing, Cham, ISBN: 978-3-031-06828-7, 2023, http://dx.doi.org/10.1007/978-3-031-06829-4, URL: https://link.springer.com/10.1007/978-3-031-06829-4.
[120] S. Horlitz, Housing Beyond Profit: A Comparison of U.S. and German Alternative Ownership Models, German Institute for Urban Affairs, 2022, ISBN: 978-3-87994-477-7.
[121] C. Altun, B. Tavli, H. Yanikomeroglu, Liberalization of digital twins of IoT-enabled home appliances via blockchains and absolute ownership rights, IEEE Commun. Mag. 57 (12) (2019) 65–71, http://dx.doi.org/10.1109/MCOM.001.1900072, Conference Name: IEEE Communications Magazine.
[122] A. Abdelmaboud, A.I.A. Ahmed, M. Abaker, T.A.E. Eisa, H. Albasheer, S.A. Ghorashi, F.K. Karim, Blockchain for IoT applications: Taxonomy, platforms, recent advances, challenges and future research directions, Electronics 11 (4) (2022) http://dx.doi.org/10.3390/electronics11040630, URL: https://www.mdpi.com/2079-9292/11/4/630.
[123] C. Sguanci, R. Spatafora, A.M. Vergani, Layer 2 blockchain scaling: a survey, 2021, http://dx.doi.org/10.48550/arXiv.2107.10881, URL: http://arxiv.org/abs/2107.10881. arXiv:2107.10881 [cs]. (Accessed 24 January 2025).
[124] A. Gangwal, H.R. Gangavalli, A. Thirupathi, A survey of layer-two blockchain protocols, J. Netw. Comput. Appl. 209 (2023) http://dx.doi.org/10.1016/j.jnca.2022.103539, URL: https://linkinghub.elsevier.com/retrieve/pii/S1084804522001801.
[125] D. Helbing, S. Mahajan, R. Hänggli, A. Musso, C.I. Hausladen, C. Carissimo, D. Carpentras, E. Stockinger, J. Argota Sánchez-Vaquerizo, J. Yang, M.C. Ballandies, M. Korecki, R.K. Dubey, E. Pournaras, Democracy by design: Perspectives for digitally assisted, participatory upgrades of society, 2022, http://dx.doi.org/10.2139/ssrn.4266038, URL: https://papers.ssrn.com/abstract=4266038. (Accessed 03 January 2023).
[126] M. Takemiya, Sora: A decentralized autonomous economy, in: 2019 IEEE International Conference on Blockchain and Cryptocurrency, ICBC, 2019, pp. 95–98, http://dx.doi.org/10.1109/BLOC.2019.8751489, URL: https://ieeexplore.ieee.org/abstract/document/8751489.
[127] B. Nguyen Thanh, H.X. Son, D.T.H. Vo, Blockchain: The economic and financial institution for autonomous AI? J. Risk Financ. Manag. 17 (2) (2024) http://dx.doi.org/10.3390/jrfm17020054, URL: https://www.mdpi.com/1911-8074/17/2/54.
[128] J.A. Fehrer, S. Benoit, L. Aksoy, T.L. Baker, S.J. Bell, R.J. Brodie, M. Marimuthu, Future scenarios of the collaborative economy: Centrally orchestrated, social bubbles or decentralized autonomous? J. Serv. Manag. 29 (5) (2018) 859–882, http://dx.doi.org/10.1108/JOSM-04-2018-0118, URL: https://www.emerald.com/insight/content/doi/10.1108/JOSM-04-2018-0118/full/html.
[129] M. Asif, Z. Aziz, M. Ahmad, A. Khalid, H.A. Waris, A. Gilani, Blockchain-based authentication and trust management mechanism for smart cities, Ital. Natl. Conf. Sens. (2022) http://dx.doi.org/10.3390/s22072604, URL: https://www.semanticscholar.org/reader/30d22bea5fe07307093df266bd7039ae2dc173ed.
[130] J. Patacas, N. Dawood, M. Kassem, BIM for facilities management: A framework and a common data environment using open standards, Autom. Constr. 120 (2020) http://dx.doi.org/10.1016/j.autcon.2020.103366, URL: https://linkinghub.elsevier.com/retrieve/pii/S0926580520309468.
[131] N. Montes, N. Osman, C. Sierra, M. Slavkovik, Value engineering for autonomous agents, 2023, URL: http://arxiv.org/abs/2302.08759. (Accessed 25 October 2023).
[132] P. De Filippi, M. Mannan, W. Reijers, Blockchain as a confidence machine: The problem of trust & challenges of governance, Technol. Soc. 62 (2020) http://dx.doi.org/10.1016/j.techsoc.2020.101284, URL: https://linkinghub.elsevier.com/retrieve/pii/S0160791X20303067.
[133] A. Francisco, N. Mohammadi, J.E. Taylor, Smart city digital twin–enabled energy management: Toward real-time urban building energy benchmarking, J. Manag. Eng. 36 (2) (2020) http://dx.doi.org/10.1061/(ASCE)ME.1943-5479.0000741, URL: https://ascelibrary.org/doi/10.1061/%28ASCE%29ME.1943-5479.0000741.
[134] P. Parker, S. Schmidt, Enabling urban commons, CoDesign 13 (3) (2017) 202–213, http://dx.doi.org/10.1080/15710882.2017.1355000, URL: https://www.tandfonline.com/doi/full/10.1080/15710882.2017.1355000.
[135] J. Pitt, J. Schaumeier, A. Artikis, Axiomatization of socio-economic principles for self-organizing institutions: Concepts, experiments and challenges, ACM Trans. Auton. Adapt. Syst. 7 (4) (2012) 1–39, http://dx.doi.org/10.1145/2382570.2382575, URL: https://dl.acm.org/doi/10.1145/2382570.2382575.
[136] S. Van Kerckhoven, U.W. Chohan, Decentralized Autonomous Organizations: Innovation and Vulnerability in the Digital Economy, first ed., Routledge, London, ISBN: 978-1-00-344960-7, 2024, http://dx.doi.org/10.4324/9781003449607, URL: https://www.taylorfrancis.com/books/9781003449607.
[137] W. Reijers, I. Wuisman, M. Mannan, P. De Filippi, C. Wray, V. Rae-Looi, A. Cubillos Vélez, L. Orgad, Now the code runs itself: On-chain and off-chain governance of blockchain technologies, Topoi 40 (4) (2021) 821–831, http://dx.doi.org/10.1007/s11245-018-9626-5, URL: https://link.springer.com/10.1007/s11245-018-9626-5.
[138] A. Saari, J. Vimpari, S. Junnila, Blockchain in real estate: Recent developments and empirical applications, Land Use Policy 121 (2022) http://dx.doi.org/10.1016/j.landusepol.2022.106334, URL: https://linkinghub.elsevier.com/retrieve/pii/S0264837722003611.
[139] R. Levine-Schnur, From blackstone to blockchain: Theorizing property law in the age of cryptography, in: A. Lehavi, R. Levine-Schnur (Eds.), Disruptive Technology, Legal Innovation, and the Future of Real Estate, Springer International Publishing, Cham, ISBN: 978-3-030-52387-9, 2020, pp. 29–34, http://dx.doi.org/10.1007/978-3-030-52387-9_2.
[140] M.C. Ballandies, H. Wang, A.C.C. Law, J.C. Yang, C. Gösken, M. Andrew, A taxonomy for blockchain-based decentralized physical infrastructure networks (depin), 2023, http://dx.doi.org/10.13140/RG.2.2.18553.85605, URL: http://arxiv.org/abs/2309.16707. arXiv:2309.16707 [cs]. (Accessed 15 April 2024).
[141] A. Nash, Decentralized intelligence network (DIN), 2024, arXiv preprint arXiv:2407.02461. URL: https://arxiv.org/abs/2407.02461.
[142] B. Friedman, C. College, Bias in computer systems, ACM Trans. Inf. Syst. 14 (3) (1996) 330–347, http://dx.doi.org/10.1145/230538.230561.
[143] L. Floridi, J.W. Sanders, Artificial evil and the foundation of computer ethics, SSRN Electron. J. (2001) http://dx.doi.org/10.2139/ssrn.3848534, URL: https://www.ssrn.com/abstract=3848534.
[144] L. Floridi, J.W. Sanders, On the morality of artificial agents, Minds Mach. (2004) http://dx.doi.org/10.1023/B:MIND.0000035461.63578.9d.
[145] S. Franklin, A. Graesser, Is it an agent, or just a program?: A taxonomy for autonomous agents, in: J.G. Carbonell, J. Siekmann, G. Goos, J. Hartmanis, J. Van Leeuwen, J.P. Müller, M.J. Wooldridge, N.R. Jennings (Eds.), Intelligent Agents III Agent Theories, Architectures, and Languages, in: Lecture Notes in Computer Science, vol. 1193, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997, pp. 21–35, http://dx.doi.org/10.1007/BFb0013570, ISBN: 978-3-540-62507-0 978-3-540-68057-4. URL: http://link.springer.com/10.1007/BFb0013570.
[146] K.E. Himma, Artificial agency, consciousness, and the criteria for moral agency: what properties must an artificial agent have to be a moral agent? Ethics Inf. Technol. 11 (1) (2009) 19–29, http://dx.doi.org/10.1007/s10676-008-9167-5, URL: http://link.springer.com/10.1007/s10676-008-9167-5.
[147] P. Poux, P. de Filippi, S. Ramos, Blockchains for the governance of common goods, in: Proceedings of the 1st International Workshop on Distributed Infrastructure for Common Good, ACM, Delft Netherlands, ISBN: 978-1-4503-8197-0, 2020, pp. 7–12, http://dx.doi.org/10.1145/3428662.3428793, URL: https://dl.acm.org/doi/10.1145/3428662.3428793.
[148] E. Ostrom, Governing the Commons: The Evolution of Institutions for Collective Action, in: Political Economy of Institutions and Decisions, Cambridge University Press, Cambridge, 1990, http://dx.doi.org/10.1017/CBO9780511807763, URL: https://www.cambridge.org/core/books/governing-the-commons/7AB7AE11BADA84409C34815CC288CD79.
[149] J.J. Hunhevicz, D.M. Hall, P.-A. Brasey, M.M. Bonanomi, M. Fischer, Decentralized project delivery on the crypto commons: Conceptualization, governance mechanisms, and future research directions, Proj. Leadersh. Soc. 5 (2024) http://dx.doi.org/10.1016/j.plas.2024.100132, URL: https://linkinghub.elsevier.com/retrieve/pii/S2666721524000176.
[150] I. Murtazashvili, J.B. Murtazashvili, M.B.H. Weiss, M.J. Madison, Blockchain networks as knowledge commons, Int. J. Commons 16 (1) (2022) 108–119, http://dx.doi.org/10.5334/ijc.1146, URL: https://www.thecommonsjournal.org/articles/10.5334/ijc.1146/.