future internet

MDPI

Article

A Self-Sovereign Identity–Blockchain-Based Model Proposal for Deep Digital Transformation in the Healthcare Sector

Luisanna Cocco * and Roberto Tonelli

Department of Mathematics and Computer Science, University of Cagliari, 09124 Cagliari, Italy; roberto.tonelli@unica.it

* Correspondence: cocco@unica.it

Abstract: The acceleration of the digital transformation process imposed by the pandemic in all the countries of the European Union, and in all sectors, has given way to a revolution that up until a couple of years ago would have been impossible even to imagine. Digital innovation has become a factor of competitiveness in all sectors. In this new scenario that has come to be, the Blockchain technology, the Self-Sovereign Identity paradigm, Internet of Things, and, in general, the new technologies that will emerge, will constitute enhancers of competitiveness and will have to aim for interoperability. In this context, this article develops and presents a model proposal in the healthcare field that aims to highlight how the combination of the Blockchain technology and the Self-Sovereign Identity paradigm restores full control over a person's identity and information, while ensuring the integrity of all medical reports, enabling secure communications between personal medical devices and patient/doctor applications on devices exploiting peer Decentralized Identifiers and ensuring data privacy, exploiting Zero Knowledge Proofs. The proposal relies on the Veramo platform, treating all medical reports as verifiable credentials and storing them in digital wallets owned by the patient. The article concludes by presenting a prototype designed and implemented for managing medication prescriptions, their issuance, and their exchange.

Keywords: self-sovereign identity; DID; blockchain; healthcare

Citation: Cocco, L.; Tonelli, R. A Self-Sovereign Identity–Blockchain-Based Model Proposal for Deep Digital Transformation in the Healthcare Sector. *Future Internet* **2024**, *16*, 473. https://doi.org/10.3390/fi16120473

Academic Editor: Petros Patias

Received: 2 November 2024
Revised: 11 December 2024
Accepted: 16 December 2024
Published: 19 December 2024

1. Introduction

The digital transformation triggered by the pandemic in all the countries of the European Union, and in all sectors, has created a new information technology (IT) scenario in which the Blockchain technology, the Self-Sovereign Identity paradigm, Internet of Things (IoT), and, in general, the new technologies that will emerge, will lead to an improvement in the economy and society in general. During the pandemic, entrepreneurs, workers, schools, and society as a whole in general have developed new digital habits to survive the crisis by trying in every way to carry on their usual activities.

In this new scenario, digital innovation will impact e-Government, business models, and society as a whole. Everyone will feel the impact. Therefore, it is necessary to spread greater awareness of these technologies, of the change taking place, and of the systems already in use or under development that aim at a more equitable and sustainable society by focusing on digital transformation.

The EU has launched a series of regulations that aim to create a more sustainable economy based on technology at the service of the community, such as the European Blockchain Service Infrastructure (EBSI) and European Self Sovereign Identity Framework (ESSIF) initiatives. The former aims at the creation of an institutional blockchain infrastructure, shared by all Member States and at the service of citizens. The latter aims at the creation and use of SSI models in order to facilitate cross-border interaction (refs. [1–3]).

In the short to medium term, the Self-Sovereign Identity paradigm will overturn all the IT processes with which users have interfaced in recent years in the most diverse fields,

from the financial world to that of healthcare. The principles on which an SSI is based are aligned with those of the General Data Protection Regulation (GDPR) and aim to protect and limit the use and processing of citizens' personal data. For example, the GDPR gives EU citizens the right to delete personal data and the right to request the correction of incorrect data, rights that can be implemented with an SSI system but which lead to inevitable incompatibilities if one considers the blockchain technology alone. In fact, the data present in a blockchain are mostly immutable; they can be kept without time limits, thus entering into conflict with the so-called right to be forgotten of the interested party, but also with the principle of minimization with regard to their use, given that the personal data entered in a blockchain are public and can be consulted by all participants even without the user having given consent [4].

The SSI–blockchain union can play a fundamental role in this context. In fact, if on the one hand, the SSI paradigm gives individuals control of their data, on the other hand, the blockchain guarantees their integrity, thus allowing for the creation of a single system capable of guaranteeing security and privacy, while enabling online identity management.

Nowadays, companies and organizations, including hospitals, and clinics, find themselves having to manage an ever-increasing amount of data that have to be both processed and archived using centralized systems that typically present problems related to security, privacy, data ownership, and data isolation. In the digital age, data represent us and constitute our only personal identity. Therefore cyber security, oriented towards the protection of personal data, is necessary to ensure that our data are not misrepresented, distorted, or used without our consent [5–8].

The SSI paradigm solves these problems and allows for the creation of a system in which digital identities can operate thanks to their corresponding Decentralized Identifier (DID), DID document (DDO), and wallet. Sensitive data are managed by Mobile Wallets, and therefore by applications installed on the user's own devices, and no longer outsourced to third-party information systems. This paradigm offers opportunities to improve privacy, security, and process efficiency in multiple sectors, thanks to the greater control it gives individuals over their personal information. For example, banks can use this paradigm to simplify customer onboarding processes, and governments can use this paradigm to improve citizen authentication, streamline online services and ensure personal data protection, or to issue and verify certificates and diplomas digitally, reducing fraud. In the healthcare sector, an SSI allows for the secure and private management of patients' personal information, facilitating data sharing between different healthcare entities and improving patient privacy, and in the IoT sector, an SSI can ensure that devices communicate securely and that users have control over the information they share. The model proposal presented in this article focuses on these last two sectors.

Specifically, this article presents a smart healthcare model proposal that is based on an SSI to return the control of data to individuals and that relies on blockchains to ensure data integrity. Focused on the most innovative information technologies, this article aims to propose an identity management model for all healthcare professionals and all entities that are involved with healthcare facilities, such as patients or general practitioners, without neglecting that of the medical devices connected through IoT technology to healthcare systems, and whose number has grown further with the COVID-19 pandemic to limit patient movements and risks for medical personnel.

Current data management in the healthcare system presents several challenges and issues, such as data security, interoperability among healthcare IT systems, data quality, and data access, which can affect the quality of care, patient safety, and operational efficiency.

All healthcare institutions can be vulnerable to cyber attacks, and different IT systems often do not communicate with each other effectively, making it difficult to transfer and share information. In addition, a lack of standardization or errors in data entry can lead to the creation of incomplete, inaccurate, or outdated databases, and the access to data by medical personnel necessary to make informed decisions can be difficult, reducing the effectiveness of care and increasing the risk of medical errors.

When investigating current data management in the healthcare system, we cannot ignore the data coming from IoT medical devices that are transforming the system by making it more efficient, accessible, and patient-centered, and all the challenges and issues associated with them. IoT medical devices can also be vulnerable to security breaches; the data collected by them must be treated in compliance with privacy regulations, and the quality and accuracy of the data must be guaranteed. Furthermore, to ensure the integration of data into a single healthcare information system, the use of common formats and protocols must be a priority.

The proposal aims to manage a massive amount of health data coming from the healthcare digitization process and that represent invaluable sources of information, both for a better delivery of health services and for medical research and prevention. The model proposal takes into account the creation and exchange of any type of medical documentation, such as medical reports issued by a specialist after a visit or data collected by a precise medical device. The owner of such medical reports can share them with other doctors through their DDO, the service endpoint that allows us to create a secure connection by using a DIDcomm protocol. Such medical documentation is shared as verifiable credentials (VCs) or verifiable presentations (VPs), which represent key concepts of the SSI paradigm. All stakeholders share data through secure channels thanks to encryption.

In SSI systems, individuals have complete control of their data and of the devices associated with them and choose in a completely autonomous way how, when, where, and with whom to share it, also choosing which specific data to share. These systems allow us to carry out more smoothly management and verification processes which in traditional systems are typically slow, expensive, and not entirely reliable. In addition to exploiting VCs, these systems allow us to develop scalable and reliable solutions that guarantee data privacy while managing the complexity and overhead of IoT medical device administration.

The proposal is conceived as a decentralized application that includes two subsystems: one on chain and another off chain. On the blockchain side, the proposal includes two smart contracts. One implements the so-called verifiable registry to make the entities operating as issuers publicly known, to guarantee the integrity of the VCs' claims, and to make the credential status publicly accessible. The other implements the DID register and allows for the building of the DDO associated with each DID. On the off-chain side, the proposal exploits Veramo, a Javascript framework, for identity management and data sharing as VCs and the InterPlanetary File System (IPFS) for storing VCs' schemas. A prototype of the model proposal for the use case related to the creation and exchange of medication prescriptions and a schema for prescriptions are described in Section 4.5.

The novelty of the model proposal lies in exploiting information technologies that promote communication and integration among different systems, facilitating the exchange of information and resources to amplify operational efficiency, reduce information silos within individual structures, and, as a result, to promote collaboration among the various stakeholders in the sector that can be both public and private. All this is thanks to the creation and the exchange of all medical records as VCs, which represents the element of differentiation from other works present in the literature as detailed in Section 2.

The model proposal is based on open standards and common protocols to encourage innovation, allowing the entire healthcare system to adapt to changes by integrating new tools and services more easily without having to completely redesign their IT infrastructures. In this regard, as detailed in the following sections, the model proposal relies on the Veramo platform, on protocols such as DIDcomm, on peer DID connections, and on data sharing platforms such as the IPFS.

It is necessary to highlight that the implementation of a healthcare system that fully embraces the potential of the model proposal requires collaboration between institutions, technology providers, healthcare professionals, and patients. It is also essential to address issues related to governance, regulation, and user education to ensure an effective and safe transition to this new paradigm. Data privacy laws and regulations, such as the GDPR in Europe, place stringent requirements on the management of clinical data, but as already highlighted, the SSI paradigm is compliant with this regulation by definition.

Today, many organizations such as Trinsic ref. [9], Connect ref. [10], Dock ref. [11], Esatus ref. [12], Talao ref. [13], Veramo [14–16], and the ESSIF refs. [1,3,17] are working on solutions for storing and managing credentials, hence for managing an SSI (see works by [18–20]), but governance models that define principles, terminology, standards, and responsibilities and trust frameworks that define specifications and rules need further investigation to allow individuals to operate through their SSI in several specific sectors [4,21,22].

2. Related Work

Regarding the literature on SSIs and healthcare, as also underlined by Satybaldy et al. [23], who, through a comprehensive systematic literature review, presented a taxonomy identifying and analyzing the open challenges in SSIs, there are not many research works in the healthcare domain.

Bai et al. [5] and Harrel et al. [7], presented an SSI management system based on Hyperledger Indy/Aries for managing all the stakeholders, including the medical devices. The former used a Hyperledger Aries blockchain to implement the model and presented a performance analysis using Hyperledger Indy to gather data. In this model, the valid data retrieval requests generated by a doctor imply the creation by the patient of a download link that allows the doctor to download necessary data. The latter presented the application MediLink, which allows patients to interface with clinics autonomously through a custom-built web and mobile application realized in ReactJS and React Native. MediLink manages six credential types, a health ID with patient demographics, insurance, a medication list including a COVID-19 vaccination status, a credit card, medical power of attorney (MPOA) for guardians of pediatric or geriatric patients, and research consent, and leverages blockchain technologies to establish the patient-centric and patient-controlled sharing of medical records across isolated institutions. Also, Siqueira et al. [6] presented a blockchain-based architecture for the management of an SSI in the healthcare sector and validated it using the Indy/Aries architecture and the Trinsic wallet. Siqueira et al. [8] and Houtan et al. [24] presented a review about SSI and blockchain technology in the healthcare sector, investigating the link between these technologies and the management of health records. Shuaib et al. [25] discussed the applicability of blockchain-based SSI solutions and presented a use case in which a patient presents a health insurance attestation in such a way as to gain access to dental services from a doctor.

An interesting white paper by the Sovrin Foundation [26], in agreement with the model proposal, presents the benefit of an SSI in the healthcare sector through some interesting use cases. Specifically, it illustrates a use case in which a 55-year-old male has Type 1 diabetes and dementia and uses a glucose meter implanted under his skin and shoe inserts that recognize him by his gait. Using an app, the male stores his glucose levels and shares these data with his endocrinologist, and his parents track his physical location. In this use case, DIDcomm and peer DID connections were proposed to ensure that the glucose meter connects securely to a smartphone. Thanks to peer DIDs, unique secure identifiers known only to the meter and the smartphone are created, and no one can impersonate the male. Thanks to DIDcomm, each message exchanged is encrypted with the public key of the receiver and decrypted by the sender with his private key. Thus, the meter and smartphone exchange data securely. This use case also leverages VC concepts; a legal guardian can be appointed and, in case the male goes missing, the guardian can require the issuance of a VC to law enforcement granting access to the male's location data and his glucose meter readings for the duration of the search operation.

Note that, unlike the literature articles cited above, the peculiarity of this article is that it manages not only the personal information of digital identities but also all medical reports as VCs and, to provide an in-depth understanding of the capabilities, resources and opportunities offered by a healthcare system based on an SSI and blockchain, which aims at the realization of a healthcare information system designed to integrate and decentralize all information coming from different sources, making data accessible and usable in a unified way. In such a system, interoperability and data integrity become crucial factors to improve

operational efficiency, facilitate informed decisions, and ensure a holistic view of information. The definition of the model proposal was also based on best practices, models, and principles of software engineering, typical of the development of decentralized applications.

As a result, the proposal was designed using the ABCDE methodology, a method of software engineering specially created for decentralized application development and relying on the Veramo framework, which on its web page https://veramo.io/docs/basics/introduction/ (accessed on 15 October 2024) underlines the following:

> *The verifiable data landscape has grown exponentially in the past few years and continues to grow at an accelerated rate. Where you have accelerated innovation, you will also have a lot of competing standards. This anomaly makes interoperability challenging. Veramo was designed to be flexible, modular, and scalable. It allows developers to orchestrate a custom system from a growing list of standards without worrying about interoperability and fast-changing specs. . . . At Veramo, we work closely with the W3C and DIF to build compatibility across many projects and initiatives in the space.*

In this work, interoperability and the use of open standards were a priority, so W3c credentials that follow the standards set by the W3C, ref. [27], communication protocols, such as DIDcomm by the Decentralized Identity Foundation, ref. [28], or decentralized systems, such as the IPFS for storing VCs' schemas and an Ethereum blockchain to ensure the integrity of the claims declared in the credentials and to make the VC status publicly accessible were used and proposed, and the Veramo platform was adopted since its modular architecture allowed us to add specific functions as needed.

3. Self-Sovereign Identity: An Overview

An SSI model is a model for managing digital identities made possible thanks to distributed ledger technologies, including the blockchain. It operates in a similar way to real identities and thus it contrasts with traditional account-based models. In this model, a direct relationship is created between the entities that want to communicate or exchange data through so-called VCs. VCs are the digital representation of a person's attributes or other information, such as a driver's license, a school diploma, or a birth certificate.

VCs are digitally represented through the use of credential schemas, which are json-ld schemas. These schemas represent the data template for a credential. They allow the issuer to structure the data within a credential in a very precise way so that the holder and verifier can correctly interpret them. Typically, a credential includes the following fields: *contexts*, *identifiers*, *types*, *the credential subject*, *the issuer*, *the issuance date*, and *proofs* (signatures) that define the terminology to be used to communicate, identify the credential, describe the type of credential, identify the holder of the credential, identify the issuer of the credential, identify the date of issuance, and present the cryptographic proof to verify the validity of the credential, respectively.

In the SSI model, there is a trilateral relationship in which a holder, who is the one who possesses a given credential, an issuer, who is the one who issued the credential, and a verifier, who is the one to whom the credential is presented and who has to verify whether the credential is valid, operate. In an SSI system, the data are not stored in a centralized system but are archived in digital wallets owned by the users, where the user also stores his/her keys.

In such a system, two entities communicate with each other by creating a connection and operating thanks to the DIDs and DDO which allow the parties to identify themselves and to prove their identity to the other party. A protocol for communication between DIDs widely used in the community is DIDComm, [29], which creates a secure communication channel in which the control of the private key by a DID proves the authenticity of the DID entity.

The *DID method* specifies the precise operations by which DIDs and the DDO are created, resolved, updated, and deactivated (refs. [30,31]). *DID:ethr:* is a DID method based on the Ethereum blockchain. In the model proposal, which uses the local blockchain, Ganache, a DID is, for example, *did:ethr:1337:0x0278facce82103 dd0bd079f235ed93f29187117a10e9e8e*

a6dfc4dcff517d06666, where 1337 identifies the local network and the hexadecimal string is the public key that identifies a specific entity.

4. Model Proposal

The architecture of the model proposal is complex and requires several components to ensure security, privacy, and interoperability. The first module allows stakeholders (such as patients, doctors, nurses, and administrators) to create, manage, and control their digital identities, using blockchain-based technologies to ensure immutability and security. A second module manages the storage of VCs and the keys used to exchange encrypted messages. A third module manages the interaction with IoT medical devices. A fourth module manages the issuance of VCs, such as medical reports, vaccinations, and treatments performed, storing their status and the hash of their claims on the blockchain. A fifth module manages the storage of VCs' schemas on the IPFS. Another module allows for the verification of the validity of a VC, exploiting the proofs on chain and on the IPFS. Yet another module manages the web and/or mobile application that must guarantee access to all the system functionalities, including on-chain functionalities. Through these applications, the different entities store and select the information they wish to share.

In such a system, typical interactions can be the registration of stakeholders, the issuance of VCs having health data as claims, and the sharing of VCs between a patient and a doctor. So, a stakeholder, for example, a patient, creates their digital identity through an app in their smartphone and receives a public/private key with which he/she proves his/her identity and issues a VC and VP; a stakeholder, for example, a doctor or an IoT device, issues VCs containing health information and sends them to the patient, who must store them safely in his/her digital wallet; a patient shares their data with a verifier (for example, another doctor), presenting a VP through his/her wallet; and through a user-friendly web or mobile application, a verifier validates the credentials, performing a check on chain of the identity of the issuer and a check on chain and on the IPFS of the validity of the information and the VC's schema.

Figure 1 shows the SSI data flow in the model proposal in which all stakeholders, such as patients and doctors and IoT medical devices, operate through their DID. In this proposal, the blockchain and IPFS represent the so-called verifiable registries. In the first registry, the DDO and VC's proofs and status are stored; on the other hand, in the second, verifiable credentials (VCs)' schemas are stored.

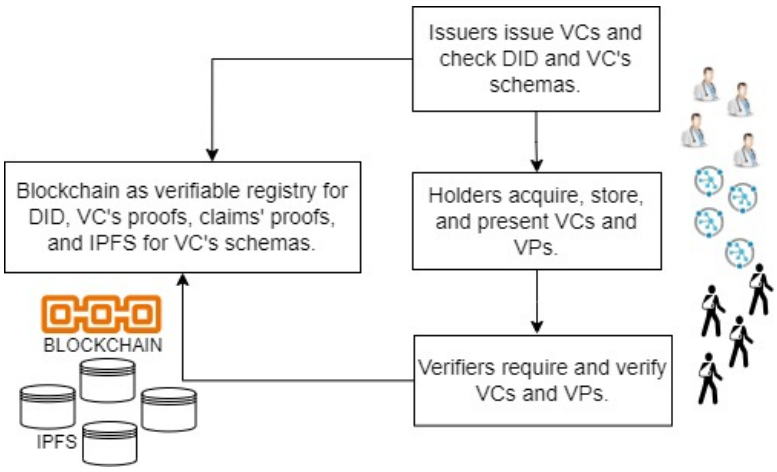

Figure 1. SSI data flow.

The identity of the issuers is declared using the *identity management* pattern with the address published on chain. Precisely these identities are stored in a register that associates their DID with their identification code in such a way that the DIDs of all the entities authorized to issue medical reports are publicly known and associated with a specific individual/organization.

All stakeholders can issue VCs. When a credential is issued, the issuer signs the credential and delivers it to the holder [4,32], who stores it in their digital wallet. The credential issued contains all the data necessary to verify the validity of the credential. It contains the proofs which guarantee that a specific issuer has issued a given document on a specific date to a specific entity and contains the smart contract's address in which the hashes of the VC's claims are stored. So, the model proposal can guarantee the integrity of all medical reports. The combination of SSI concepts with the storage on chain of the clinical information flow of a patient allows for the exchange of reports with third parties with the certainty of data integrity and of the correct time flow since the verification of the validity of exchanged clinical reports by relying on the proofs left on chain.

VCs can also be used to grant temporary authorization to another entity to access the credential in its entirety or only some of its claims, and can also be issued by IoT medical devices to transmit data as VCs to the patient, for example, in the case of a blood pressure meter that transmits data to the mobile application of the patient.

The proposal can be applied for enabling secure communications between personal medical devices and patient/doctor devices. It guarantees secure communications between personal medical devices such as blood pressure meters and user/doctor devices. In the SSI paradigm, so-called peer DIDs allow us to establish a secure connection between the blood pressure meters and the applications on the patient's mobile phone or on the doctor's terminal. This is because, in addition to the secure communication channel between meters and applications provided by the DIDcomm protocol, thanks to peer DIDs, the two parties can use keys shared only between them. In this way, no one can impersonate the doctor and read the blood pressure data.

In addition, the proposal can ensure data privacy, exploiting Zero Knowledge Proofs (ZKPs) and in particular the BBS+ signature scheme. Let us consider a use case where a patient gives temporary permission to a doctor to access their credentials. The credentials can be related to personal information and clinical data. The patient can allow the doctor to read all the claims in the credentials or only some subsets of the claims. In emergency situations, in case the patient is unable to give consent for access to their data by themselves, the doctor could access the data thanks to the delegation by the patient issued as a VC previously. In turn, the patient is sure that the doctor can only access precise data, exploiting ZKPs as defined in the VC. The patient has full control over granting and revoking access to clinical data and other personal data.

4.1. ABCDE Method: Goal, Actors, and User Stories

The development of the blockchain–SSI model proposal was performed following the ABCDE method [33]. This method outlines eight steps for the development of blockchain software systems. The first three steps involve the definition of the requirements of the system, and hence the definition of the goal and the actors of the system, and the description of the requirements in terms of user stories. In the fourth step, the system is divided into on-chain and off-chain components, and in the last four steps, the design, the testing, the integration, and the release are carried out. In the following, some of the steps are described.

The goal of the proposal is to present an SSI–blockchain-based model that allows for the management of all medical reports as VCs.

The following user stories describe the functional requirements that the model proposal has to achieve.

- The system administrator initializes the system and deploys the smart contracts. He publishes the identity of the issuers authorized to operate it.
- All stakeholders have their own digital identity.
- All stakeholders communicate with each other by identifying and authenticating themselves autonomously, without resorting to third parties.
- Medical reports are issued by doctors and/or delegation reports are issued by the patient/legal guardian.
- The patient keeps his/her medical reports and presents them as VPs when necessary.

- IoT medical devices communicate data to the app in the patient's and/or in the doctor's device.
- Before considering the presented credentials valid, a verifier has to be verify the VC's claims and the VC's schema against the blockchain and IPFS, respectively.

The actors are the doctor, the patient, the legal guardian, the IoT medical devices, the administrator, and the smart contracts.

4.2. ABCDE Method: On-Chain Component and Its Design

The proposal comprises two smart contracts, the *EthereumDIDRegistry.sol* extracted from https://github.com/uport-project/ethr-did-registry (accessed on 11 October 2024) and the *CredentialIssuerRegistry.sol* proposed in this article and also reported in the description of the prototype.

Two smart contracts, *CredentialIssuerRegistry* and *EthereumDIDRegistry*, were developed. The first implements the logic to make publicly accessible the identity of the issuers and the logic to store the status and the hash of the VC's claims. The second implements the logic for creating the DDOs. This is because in the world of SSIs, DDO registries, lists of certification authorities, lists of certificates issued by the so-called issuers, and the status of certificates, as well as lists of VCs' schemas that allow for the standardization and simplification of the creation and management of credentials, play a key role. These elements contribute to creating a secure and reliable ecosystem for digital identity management. The use of blockchain registries and smart contracts or other types of decentralized systems such as the IPFS are essential for the management of such lists and allow for accessibility to information even if an issuer decides not to operate. The lists remain in the blockchain or in the IPFS, and not in a concentrated way in a digital repository exposed by the issuer.

Note that developing a system that relies on a blockchain means implementing applications that reside on chain, the so-called smart contracts, which have unique characteristics that distinguish them from traditional software applications. Once deployed on the blockchain, the smart contract's code is publicly accessible. Further, both deploying smart contracts on blockchains and performing blockchain transactions imply a cost. For this reason, these applications have to be designed and implemented following patterns and best practices right from the early stages of their development (see works by [33–35] for more details).

Regarding access to the code, it is managed by *mapping* of addresses and by *modifiers*, allowing access only to authorized users and protecting the system from unwanted access. Concerning costs, a smart contract to be deployed on a blockchain is compiled and converted into a sequence of operation codes. Each code is associated with a gas consumption, and hence with a precise cost (see work [36] for more details). For this reason, patterns and best practices for saving gas (see work by [35]) have to be adopted during the design and development of a smart contract. For example, the storage of data on chain has to be limited in the so-called *storage* memory, and the so-called *event log* memory has to be preferred. The latter contains data from events issued by the smart contract that can be accessed through applications external to the blockchain.

4.3. ABCDE Method: Off-Chain Component, Its Design, and SSI Management Information Flow

As already mentioned, the proposal relies on the Veramo framework, and all stakeholders operate in the model through the Veramo agent, which is the entry point to the Veramo framework.

Figure 2 shows a possible configuration of the relationship between the Veramo agent, the interfaces, the plugins, and the external components.

The Veramo agent exploits several classes, such as DIDComm, KeyManager, KeyStore, KeyManagementSystem, PrivateKeyStore, DIDManager, DIDStore, EthrDIDProvider, DIDResolverPlugin, Resolver, CredentialIssuer, MessageHandle, JwtMessageHandler, W3cMessageHandler, DIDCommMessageHandler, DataStore, and DataStoreORM, that are imported from several plugins to orchestrate the whole information flow described

through the user stories just listed. Note that as a default, Veramo uses SQlite databases to represent the digital wallets of the various stakeholders.

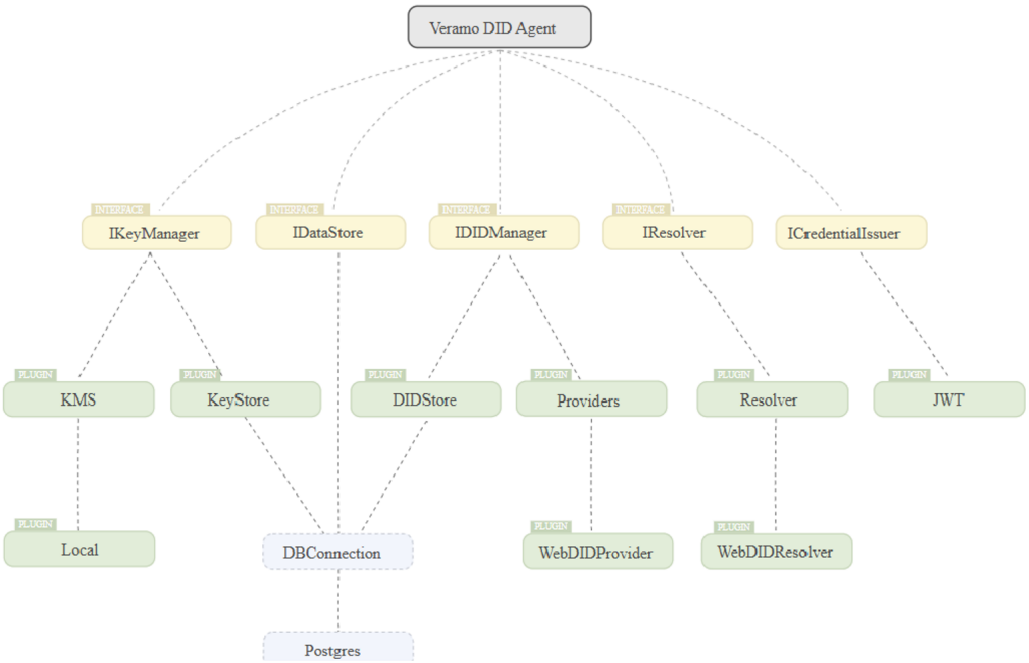

Figure 2. Veramo framework (image extracted from https://veramo.io/docs/veramo_agent/plugins (accessed on 10 October 2024)).

4.4. DID Document and Medical Reports as Verifiable Credentials

The proposal relies on the smart contract *EthereumDIDRegistry.sol*, based on ERC1056, a standard for creating and updating SSIs. This contract uses the blockchain, using its resources in a lightweight way (refs. [37,38]).

Through the interactions with the smart contract *EthereumDIDRegistry.sol*, and through the events emitted by it, the DDO of a given DID is built using the *ethr-did-resolver*. This resolver, given an ethereum address, builds the DDO corresponding to that address (ref. [39]), hence allowing us to retrieve all useful data to authenticate the DID.

The DDO contains both attributes and authentication methods, and transactions on the blockchain are required to update its data. The DDO is updated by the DID thanks to its private key, and this update involves gas consumption. The model proposal relies on Ganache, and Ganache accounts are imported through the method *didManagerImport*, managed by a core plugin of Veramo in which the public and private key have to be set, as shown in Figure 3 (ref. [40]).

```
aDoctor = await veramoAgent.didManagerImport({
    controllerKeyId: _alias,
    did: pubKey2,
    provider: 'did:ethr:1337',
    alias: _alias,
    keys: [
      {
        privateKeyHex: '0a83044e94fa255e68a3bd41914bc4c10eea615134b8b88d0b3680394b615679',
        kms: 'local',
        type: 'Secp256k1',
        kid: _alias,
      },
    ],
  })
```

Figure 3. Method *didManagerImport*.

In Figure 4, the default DDO, built when the DID is created, is shown. The DID document is built using the contract *EthereumDIDRegistry.sol* deployed on the Ganache network, and the method *resolveDid* returns the document of the DID, passed as an argument to the method mentioned (refs. [41,42])


```
{
  '@context': [
    'https://www.w3.org/ns/did/v1',
    'https://identity.foundation/EcdsaSecp256k1RecoverySignature2020/lds-ecdsa-secp256k1-recovery2020-0.0.jsonl
d'
  ],
  id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666',
  verificationMethod: [
    {
      id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controller',
      type: 'EcdsaSecp256k1RecoveryMethod2020',
      controller: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666',
      blockchainAccountId: '0x7BF1AB34418180F5C784dA4BC9705e9DcC734d60@eip155:1337'
    },
    {
      id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controllerKey',
      type: 'EcdsaSecp256k1VerificationKey2019',
      controller: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666',
      publicKeyHex: '0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666'
    }
  ],
  authentication: [
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controller',
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controllerKey'
  ],
  assertionMethod: [
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controller',
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controllerKey'
  ]
}
```


Figure 4. Default DID document.

The DDO is associated with the DID via the *id* field. There is also a *verificationMethod* field, an *authentication* field, and an *assertionMethod* field. The *verificationMethod* field contains verification material properties, that is, parameters to be used to verify a proof. The field *controller* defines the entity that controls the keys. In the case of a DID associated with an individual, the entity that controls the keys can be the individual themselves. In the case of a DID associated with an organization, the entity that controls the keys can be a member of the organization itself. In the case of a DID associated with a device, the entity controlling the keys can be the individual whom the device is associated with.

Thanks to this smart contract, a DID can delegate the performance of a specific function to a third party. In this case, the third party signs on behalf of the DID as long as the owner of the DID does not revoke the issued proxy.

Through interactions with the smart contract *EthereumDIDRegistry.sol*, a DID can also set off-chain attributes' values. For example, it can set the endpoints through which it can be reached by another DID, thus making communication between DIDs using the DIDcomm protocol possible.

Figure 5 shows the DID document containing the added endpoint.

As regards the creation of the medical report, the issuer, and hence a doctor or an IoT medical device, starts with the creation of a credential message, which conforms to a precise VC schema and builds the verifiable credential through the method *createVerifiableCredential* (ref. [43]), passing on as arguments the credential message and the proof format to be used.

In addition, since the proposed model relies on a blockchain to ensure the integrity of the claims declared in the credentials and to make the VC status publicly accessible, the issuer has to store on chain the proofs to ensure claims' integrity and the VC's status. A detailed description of the smart contract that implements these functionalities is provided in the next section.

The Veramo default platform only manages the proof format JWT (refs. [44–46]). The proof in the VC is a Json Web Token (JWT), which is a way to securely transmit json objects, and hence our credentials. The transmitted information is signed using a public/private key

pair and ES256K or ECDSA. By using the signature, it is possible to verify if the information has been tampered with and if the issuer of the JWT is that indicated in the credential.

Note that the W3C has developed the Verifiable Credentials Data Model v1.1, which includes two distinct types of verifiable credentials. The first is the JSONLD VC, which uses Linked Data Proofs. The second is the JSON-LD with BBS+ signatures, which supports the selective disclosure of claims. Since Veramo is built on a modular architecture, specific functions can be added as needed, including those supporting selective disclosure.


```
{
  '@context': [
    'https://www.w3.org/ns/did/v1',
    'https://identity.foundation/EcdsaSecp256k1RecoverySignature2020/lds-ecdsa-secp256k1-recovery2020-0.0.jsonl
d'
  ],
  id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666',
  verificationMethod: [
    {
      id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controller',
      type: 'EcdsaSecp256k1RecoveryMethod2020',
      controller: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666',
      blockchainAccountId: '0x7BF1AB34418180F5C784dA4BC9705e9DcC734d60@eip155:1337'
    },
    {
      id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controllerKey',
      type: 'EcdsaSecp256k1VerificationKey2019',
      controller: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666',
      publicKeyHex: '0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666'
    }
  ],
  authentication: [
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controller',
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controllerKey'
  ],
  assertionMethod: [
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controller',
    'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#controllerKey'
  ],
  service: [
    {
      id: 'did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666#service-1',
      type: 'DIDCommMessaging',
      serviceEndpoint: 'http://localhost:13903/messaging'
    }
  ]
}
```


Figure 5. DID document with added service endpoint.

4.5. The Model Proposal: A Prototype for Managing Medication Prescriptions

In the following section, a prototype of the proposal, related to the creation and exchange of the medication prescriptions as VCs, is described in detail. In this scenario, the list of medicines prescribed can be bought in any pharmacy provided that the pharmacist is able to verify the holder's, as well as the issuer's, proofs, and hence the prescription itself.

The credential schema proposed for medication prescription is that shown in Listing 1. The credentials' schemas are stored on the IPFS to support the credential verification process.

When a patient/holder presents a prescription to a pharmacist/verifier, the latter will have all the information that allows them to verify the validity of the prescription presented to them.

The functional requirements that the prototype has to achieve are defined through the following user stories, to which the first four user stories previously listed have to be added.

- Medication prescriptions are issued by an issuer, the doctor, and delivered to the patient.
- The patient keeps the prescription and presents it to the pharmacist when purchasing the prescribed medicines.
- Before selling the prescribed items, the pharmacist has to ensure that the prescription is valid.
- Before selling prescribed items, the pharmacist has to ensure that the person buying the medicines is the one to whom the prescription was issued.
- A doctor asks the patient to see the list of prescribed medicines.

Listing 1. Credential schema for medication prescription.

```
{
    {"$schema": "http://json-schema.org/draft-04/schema#",
    "type": "object",
    "properties": {
        "credentialSubject": {
            "type": "object",
            "properties": {
                "id": {
                    "type": "string"
                },
                "prescription": {
                    "type": "object",
                    "properties": {
                        "name": {
                            "type": "string"
                        },
                        "surname": {
                            "type": "string"
                        },
                        "address": {
                            "type": "string"
                        },
                        "state": {
                            "type": "string"
                        },
                        "prescription": {
                            "type": "string"
                        },
                        "codePrescription": {
                            "type": "string"
                        }
                    },
                    "required": [
                        "name",
                        "surname",
                        "address",
                        "state",
                        "prescription",
                        "codePrescription"
                    ]
                }
            },
            "smartContract": {
                "type": "object",
                "properties": {
                    "Address": {
                        "type": "string",
                        "description": "The address to the smart contract on Blockchain"
                    },
                    "hash": {
                        "type": "string",
                        "description": "The unique hash of the claim in the credential"
                    },
                    "status": {
                        "type": "string",
                        "description": "The status of the claim in the credential"
                    }
                },
                "required": [
                    "Address",
                    "hash",
                    "status"
                ]
            },
            "required": [
                "id",
                "prescription"
            ]
        }
    },
    "required": [
        "credentialSubject"
    ]
}
```

Figure 6 shows the UML diagram of the use cases, in which by the Veramo agent, the doctor, the patient, and the pharmacist are represented. The use cases shown in the figure represent the user stories described above.

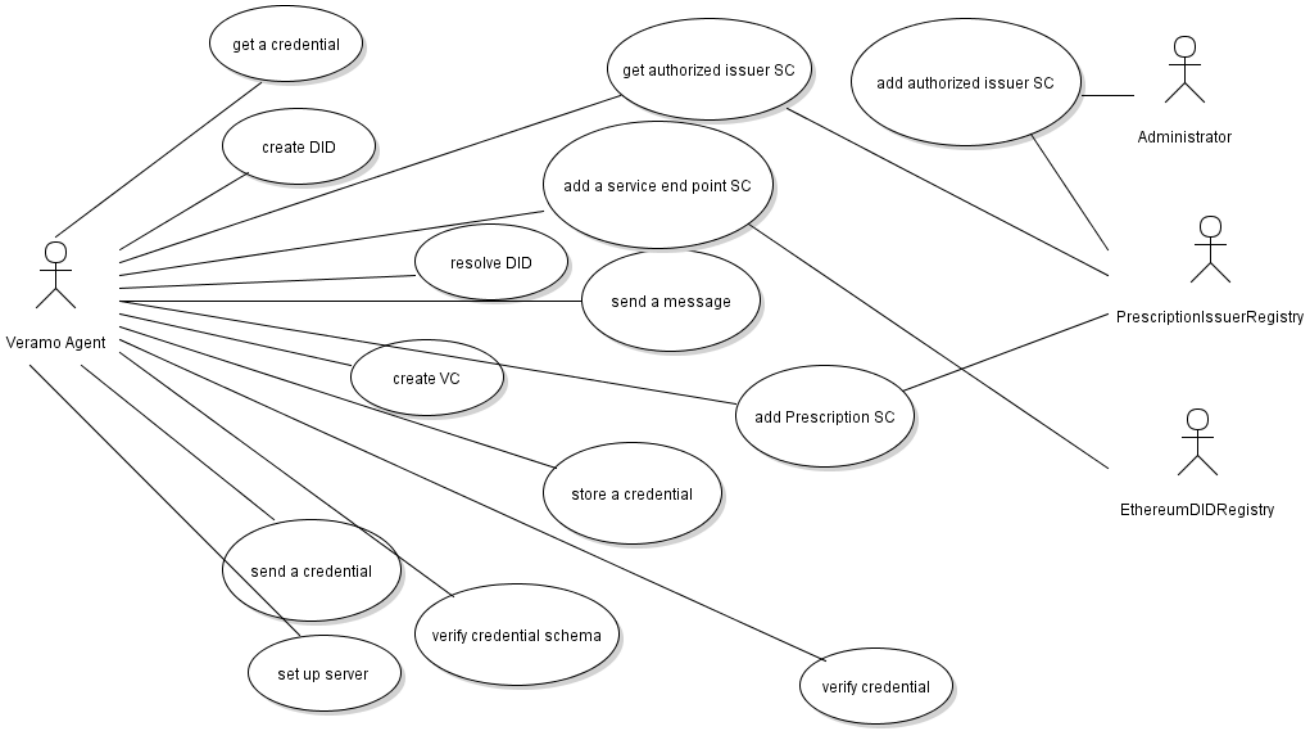

Figure 6. Use case diagram.

The smart contract, *CredentialIssuerRegistry*, in this prototype is re-named *PrescriptionIssuerRegistry*. Figure 7 shows its UML class diagram.

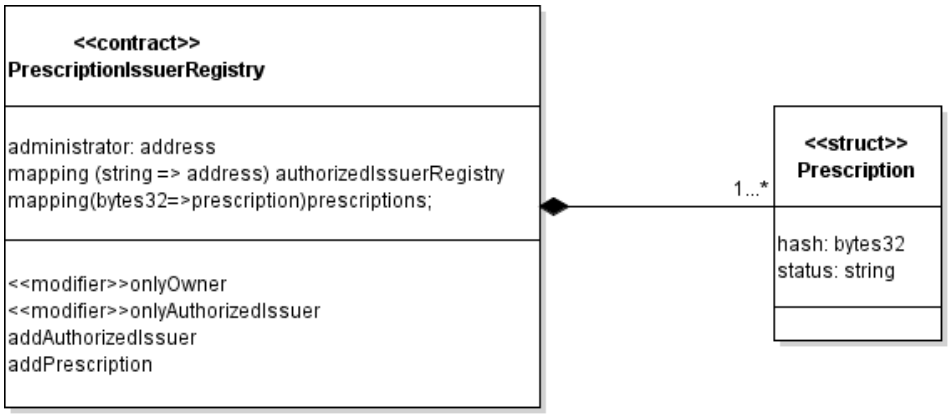

Figure 7. UML class diagram of the *CredentialIssuerRegistry.sol* contract.

The smart contract contains two *mappings*, *authorizedIssuerRegistry* and *prescriptions*, a *struct*, two methods, and two *modifiers*. The *mapping authorizedIssuerRegistry* stores the identification code of the authorized issuers, associating each code with a DID, and hence with an address. This mapping is populated using the *addAuthorizedIssuer* method, whose calls are managed through the *onlyOwner modifier*, which allows only the contract owner to run the method just mentioned. In turn, in the *mapping prescriptions*, the authorized issuers can store the *struct* named *prescription*, which contains two variables, the hash and the status of the issued credential. The first variable is of the type *bytes32* and the second is of the type *string*. The mapping is populated using the *addPrescription* method, whose calls are managed through the *onlyAuthorizedIssuers modifier*.

As regards the creation of the verifiable prescription, the issuer, and hence the doctor, starts with the creation of a credential message following the prescription schema shown in Listing 1 and builds the VC with the proof as shown in Listing 2.

Listing 2. Verifiable credential, and hence the verifiable medication prescription.

```
{
    "@context": [
        "https://www.w3.org/2018/credentials/v1"
    ],
    "issuer": {
        "id": "did:ethr:1337:0x0278facce82103dd0bd079f235ed93f29187117a10e9e8ea6dfc4dcff517d06666db"
    },
    "type": [
        "VerifiableCredential",
        "PrescriptionVC"
    ],
    "credentialSubject": {
        "id": "did:ethr:1337:0x026e205a3fefa81f8e51cf7c284ec234fa667280740ca8e394ec608edad3680bfe",
        "prescription": {
                "name": "Mario",
                "surname": "Rossi",
                "address": "L. Amat 56",
                "state": "Italy",
                "prescription": "Paracetamol 1000mg capsules",
                "codePrescription": "2000AA136779354",
            "smartContract": {
                "Address": "www.pdfs.com/pdf0.pdf",
                "hash": "8743b52063cd84097a65d1633f5c74f5",
                "status": "Valide"
            }
        }
    },
    "issuanceDate": "2022-02-18T10:02:18.000Z",
    "proof": {
        "type": "JwtProof2020",
        "jwt": "eyJhbGciOiJFUzI1NksiLCJ0eXAiOiJKV1QifQ.eyJ2YyI6eyJAY29udGV4dCI6WyJ.....      }
}
```

4.6. Final Considerations: Potentiality, Challenges, and Future Works

The blockchain and SSI union in our model proposal has several advantages but also implies several challenges. Patients gain control over their health data and manage access to their information. They can share data without giving up ownership since the data remain stored in their wallets. Patients can provide or withdraw consent to the use of their data. The SSI paradigm therefore promotes transparency and trust in how data are used. Blockchains, on the other hand, facilitate interoperability between different health systems while maintaining data integrity. The proofs that guarantee data integrity and the status of VCs as well as the DDOs that allow DID entities to interact are stored on chain, in appropriate data structures implemented in the so-called smart contracts. By distributing the proofs of claims on the blockchain, the risk of centralized data breaches is reduced, and patient data remain secure and tamper-proof. The use of DIDcomm and peer DID connections through the use of cryptographic techniques improves data protection by ensuring that sensitive information is accessible only to authenticated parties. The immutability of blockchain records can increase trust among stakeholders, who can verify the authenticity of data, track their history, and attribute responsibility for statements in VCs to individual healthcare providers or third-party institutions.

Note that the proposed Ethereum blockchain within the proposal is permissioned. Hence, the model proposal relies on the deployment of a reliable and effective blockchain solution, to be selected after having examined critical metrics such as the transaction latency, or the amount of time a transaction needs to be in the network from the moment it is submitted, the transaction throughput, or the rate at which valid transactions are committed to the blockchain, and scalability.

Among the challenges that the model proposal must address, let us mention the following. It is essential that the blockchain–SSI-based model proposal can integrate as much as possible with existing infrastructure already in use in the healthcare sector.

Each country has specific regulations regarding the management of healthcare data, so implementing a model proposal that complies with these regulations is a significant challenge. Blockchain-based systems must be scalable to handle a large number of transactions. It is essential to find solutions that ensure high performance without compromising security, as mentioned above. The widespread adoption of the model proposal requires acceptance by all stakeholders. A lack of understanding or trust in the technology can hinder adoption. In addition, creating a new system based on the model proposal requires significant upfront investments in technology development, training and integration with existing systems, the definition of common standards for the formats of all types of data in healthcare, and overcoming resistance to change in established practices in the healthcare system today.

Based on the very early research presented in this article, several avenues for future work emerge that could significantly enhance the applicability and robustness of our model proposal. First, a critical area for future investigation is the data storage mechanism, particularly concerning how digital wallets manage VCs. Another critical area is the integration of ZKPs, which represent a pivotal step in reinforcing the privacy and security of our proposal. ZKPs allow one party to prove to another that a statement is true without revealing any additional information beyond the validity of that statement. Finally, the comprehensive testing and evaluation of the proposed prototype, and of prototypes related to other use cases, is essential for validating the model proposal.

5. Conclusions

Focusing on interoperability and on best practices and principles of software engineering, this article presents a model proposal for deep digitization in the healthcare sector, aiming to give individuals back the full ownership and control of their data, and hence of their digital identity. In this model, all stakeholders operate thanks to their DIDs and DDOs, exploiting VCs to transmit data by using DIDcomm and peer DID connections, and give third parties the right to act on their behalf. In addition, by exploiting SSI and blockchain technologies, the model is able to manage the overhead associated with the large amount of IoT medical devices in need of robust, reliable, and scalable solutions.

It is worth underlining that to allow the model proposal to fully realize its potential, it is essential to involve all stakeholders. Everyone has to embrace the changes. Not only does the whole of society have to be aware of the transformation taking place, it is necessary to encourage the growth of a widespread digital culture, and in this context, the role of institutions becomes fundamental. Both at a European and national level, institutions have to protect trust in the use of digital technologies. Only in this way will the entirety of society accept the change and will digital transformation be able to produce the expected benefits. So, the goal of this article is also to provide food for thought and a better understanding of the potential of SSIs, Blockchain technology, and IoT and of digital transformation in general.

Author Contributions: Conceptualization, L.C.; methodology, L.C.; software, L.C.; validation, L.C.; formal analysis, L.C.; investigation, L.C.; resources, L.C.; data curation, L.C.; writing—original draft preparation, L.C.; writing—review and editing, L.C.; visualization, L.C.; supervision, L.C. and R.T.; project administration, R.T.; funding acquisition, R.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been developed within the framework of the project e.INS, the Ecosystem of Innovation for Next Generation Sardinia (cod. ECS 00000038), funded by the Italian Ministry for Research and Education (MUR) under the National Recovery and Resilience Plan (NRRP)—MISSION 4, COMPONENT 2, "From research to business"; INVESTMENT 1.5, "Creation and strengthening of Ecosystems of innovation"; and the construction of "Territorial R&D Leaders". This work was partially funded under the National Recovery and Resilience Plan (NRRP), Mission 4, Component 2, Investment 1.3, call for tender No. 1561 of 11.10.2022 of the Ministero dell'Università e della Ricerca (MUR), funded by the European Union, NextGenerationEU, project code PE0000021, Concession Decree No. 1561 of 11.10.2022, adopted by the Ministero dell'Università e della Ricerca (MUR), CUP F53C22000770007, according to attachment E of Decree No. 1561/2022, project title: "Network 4 Energy Sustainable Transition–NEST".

Data Availability Statement: No new data were created or analyzed in this study.

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

References

1. EBSI. Available online: https://ec.europa.eu/digital-building-blocks/sites/display/EBSI/Home (accessed on 12 October 2024).
2. EBSI and eSSIF. Available online: https://decentralized-id.com/government/europe/eu/ebsi-essif/ (accessed on 12 October 2024).
3. NGI eSSIF-LAB. Available online: https://essif-lab.eu/ (accessed on 12 October 2024).
4. López, M.A. *Self Sovereign Identity the Future of Identity: Self-Sovereignity, Digital Wallets, and Blockchain*; Technical report; IDB, IDB LAB, LACCHAIN: Washington, DC, USA, 2020.

5. Bai, P.; Kumar, S.; Aggarwal, G.; Mahmud, M.; Kaiwartya, O.; Lloret, J. Self-Sovereignty Identity Management Model for Smart Healthcare System. *Sensors* **2022**, *22*, 4714. [CrossRef]
6. de Siqueira, A.G.; da Conceição, A.F.; Rocha, V. User-Centric Health Data Using Self-sovereign Identities. *arXiv* **2021**, arXiv:2107.13986.
7. Harrell, D.T.; Usman, M.; Hanson, L.; Abdul-Moheeth, M.; Desai, I.; Shriram, J.; de Oliveira, E.; Bautista, J.R.; Meyer, E.T.; Khurshid, A. Technical Design and Development of a Self-Sovereign Identity Management Platform for Patient-Centric Healthcare Using Blockchain Technology. *Blockchain Healthc. Today* **2022**, *5*, 196.
8. Siqueira, A.; Da Conceição, A.F.; Rocha, V. Blockchains and Self-Sovereign Identities Applied to Healthcare Solutions: A Systematic Review. *arXiv* **2021**, arXiv:2104.12298.
9. Trinsic. Available online: https://trinsic.id/about/ (accessed on 17 October 2024).
10. Connect.Me. Available online: https://github.com/evernym/ConnectMe (accessed on 16 October 2024).
11. Self-Sovereign Identity: The Ultimate Guide 2024. Available online: https://www.dock.io/post/self-sovereign-identity (accessed on 22 October 2024).
12. Esatus. Available online: https://esatus.com/index.html%3Fp=7663&lang=en.html (accessed on 11 October 2024).
13. Saas and in-House Solution to Integrate Digital Identity Wallets in Applications. Available online: https://talao.co/sandbox/saas4ssi (accessed on 20 October 2024).
14. ETHR DID Method Specification. Available online: https://github.com/decentralized-identity/ethr-did-resolver/blob/master/doc/did-method-spec.md (accessed on 4 October 2024).
15. Beregszaszi, A.; Braendgaard, P.; Zoltu, M.; Entriken, W.; eth bot. Ethereum Verifiable Claims. 2021. Available online: https://github.com/ethereum/EIPs/blob/master/EIPS/eip-1812.md (accessed on 4 October 2024).
16. Bugyis. Introducing Veramo. 2021. Available online: https://medium.com/uport/introducing-veramo-5a960bf2a5fe (accessed on 4 October 2024).
17. Available online: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A52021PC0281 (accessed on 10 October 2024).
18. Lipińska, A. uPort Serto Ecosystems: Creating Trusted Data Networks Between Businesses and Individuals. 2019. Available online: https://medium.com/uport/uport-serto-ecosystems-creating-trusted-data-networks-between-businesses-and-individuals-ff21c9368d3b (accessed on 14 October 2024).
19. Braendgaard, P. Different Approaches to Ethereum Identity Standards. 2018. Available online: https://medium.com/uport/different-approaches-to-ethereum-identity-standards-a09488347c87 (accessed on 14 October 2024).
20. Coutts, V. The Who's Who of Decentralized Identity Systems. 2018. Available online: https://medium.com/linum-labs/the-whos-who-of-decentralized-identity-systems-433b2dd9a195 (accessed on 14 October 2024).
21. Sroor, M.; Hickman, N.; Kolehmainen, T.; Laatikainen, G.; Abrahamsson, P. How modeling helps in developing self-sovereign identity governance framework: An experience report. In Proceedings of the International Conference on Industry Sciences and Computer Science Innovation, Gaia, Portugal, 9–11 March 2022; Cruz-Cunha, M.M., Mateus-Coelho, N., Eds.; Elsevier: Amsterdam, The Netherlands, 2022; pp. 267–277.
22. Doerk, A. The Trust Infrastructure of Self-Sovereign Identity Ecosystems. 2020. Available online: https://ssi-ambassador.medium.com/the-trust-infrastructure-of-self-sovereign-identity-ecosystems-551f46ed9e2c (accessed on 14 October 2024).
23. Satybaldy, A.; Ferdous, M.S.; Nowostawski, M. A Taxonomy of Challenges for Self-Sovereign Identity Systems. *IEEE Access* **2024**, *12*, 16151–16177. [CrossRef]
24. Houtan, B.; Hafid, A.S.; Makrakis, D. A Survey on Blockchain-Based Self-Sovereign Patient Identity in Healthcare. *IEEE Access* **2020**, *8*, 90478–90494. [CrossRef]
25. Shuaib, M.; Alam, S.; Shabbir Alam, M.; Shahnawaz Nasir, M. Self-sovereign identity for healthcare using blockchain. *Mater. Today Proc.* **2021**, *81*, 203–207. [CrossRef]
26. Self-Sovereign Identity and IoT. Available online: https://sovrin.org/wp-content/uploads/SSI-and-IoT-whitepaper.pdf (accessed on 11 September 2024).
27. Verifiable Credentials Data Model v1.1. Available online: https://www.w3.org/TR/vc-data-model/ (accessed on 17 October 2024).
28. DIF-Decentralized Identity Foundation. Available online: https://identity.foundation/ (accessed on 17 November 2024).
29. DIDComm Messaging v2.x Editor's Draft. Available online: https://identity.foundation/didcomm-messaging/spec/ (accessed on 2 October 2024).
30. ethr-did. Available online: https://github.com/uport-project/ethr-did (accessed on 17 October 2024).
31. DID Methods. Available online: https://veramo.io/docs/veramo_agent/did_methods (accessed on 17 October 2024).
32. Kiat, Y.Y. Effecting Data Portability in Our Healthcare Ecosystem: A Data Standards Problem. 2022. Available online: https://medium.com/singapore-gds/effecting-data-portability-in-our-healthcare-ecosystem-a-quick-primer-2c6c80d0ef8b (accessed on 14 October 2024).
33. Marchesi, L.; Marchesi, M.; Tonelli, R. ABCDE—Agile block chain DApp engineering. *Blockchain: Res. Appl.* **2020**, *1*, 100002. [CrossRef]
34. Marchesi, L.; Marchesi, M.; Pompianu, L.; Tonelli, R. Security checklists for Ethereum smart contract development: Patterns and best practices. *arXiv* **2020**, arXiv:2008.04761.

35. Marchesi, L.; Marchesi, M.; Destefanis, G.; Barabino, G.; Tigano, D. Design Patterns for Gas Optimization in Ethereum. In Proceedings of the 2020 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE), London, ON, Canada, 18 February 2020; pp. 9–15.
36. Wood, G. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Proj. Yellow Pap.* **2014**, *151*, 1–32.
37. Ethereum Registry for ERC-1056 Ethr Did Methods. Available online: https://github.com/uport-project/ethr-did-registry (accessed on 17 October 2024).
38. ERC: Lightweight Identity. Available online: https://github.com/ethereum/EIPs/issues/1056 (accessed on 17 October 2024).
39. DID Resolver for Ethereum Addresses with Support for Key Management. Available online: https://github.com/decentralized-identity/ethr-did-resolver (accessed on 14 October 2024).
40. IDIDManager Interface. Available online: https://veramo.io/docs/api/core-types.ididmanager (accessed on 14 October 2024).
41. IResolver Interface. Available online: https://veramo.io/docs/api/core-types.iresolver (accessed on 14 October 2024).
42. DIDManager.didManagerAddService. Available online: https://veramo.io/docs/api/did-manager.didmanager.didmanageraddservice/ (accessed on 15 October 2024).
43. CredentialPlugin.createVerifiableCredential. Available online: https://veramo.io/docs/api/credential-w3c.credentialplugin.createverifiablecredential/ (accessed on 15 October 2024).
44. Create Credential. Available online: https://veramo.io/docs/cli_tutorials/cli_create_vc (accessed on 15 October 2024).
45. Create and Verify W3C Verifiable Credentials and Presentations in JWT Format. Available online: https://github.com/decentralized-identity/did-jwt-vc (accessed on 15 October 2024).
46. did-jwt package. Available online: https://veramo.io/docs/api/did-jwt (accessed on 15 October 2024).