TÉCNICO LISBOA

# BlockRobot: Increasing Privacy in Human-Robot Interactions by using Blockchain - Healthcare Environment

**Viktor Vasylkovskyi**

Thesis to obtain the Master of Science Degree in

**Information Systems and Computer Engineering**

Supervisors: Prof. Sérgio Luís Proença Duarte Guerreiro
Prof. João Fernando Cardoso Silva Sequeira

## Examination Committee

Chairperson: Prof. David Manuel Martins de Matos
Supervisor: Prof. Sérgio Luís Proença Duarte Guerreiro
Member of the Committee: Prof. Miguel Nuno Dias Alves Pupo Correia

**Nov 2020**

# Acknowledgments

I would like to thank my parents for their friendship, encouragement and caring over all these years, for always being there for me through thick and thin and without whom this project would not be possible. I would also like to thank my aunts, uncles and my sister for their understanding and support throughout all these years. I would like to thank my girlfriend that always believed in me and motivated me with her love even in the most complicated moments.

I would also like to acknowledge my dissertation supervisors Sérgio Guerreiro and João Sequeira for their insight, support and sharing of knowledge that has made this Thesis possible.

Last but not least, to all my friends and colleagues that helped me grow as a person and were always there for me during the good and bad times in my life. Thank you.

To each and every one of you – Thank you.

# Abstract

Social Robots can record large streams of raw data in the form of images, audio, Radio Frequency Identification (RFID), among other sensory sources, which could be instrumental in enforcing Human-Robot Interactions. However, the emotional bonds between humans and robots can raise the problem of a robot accessing/inferring deeply private information, e.g., emotional states. Therefore, from the point of view of privacy, social robots may be a liability. Clarifying ownership of data collected by robots has been highlighted a concern in the European Union (EU) General Data Protection Regulation (GDPR), which requires 'privacy-by-design'. With this problem in mind, we present **BlockRobot** – a Blockchain-based Architecture to enforce privacy in Human-Robot Interactions. This architecture provides users with an identity management system for private data generated during interactions between humans and robots. The architecture explores the benefits of the Radio Frequency detection mechanism and Blockchain immutability property integrated with robotic events. The BlockRobot grants confidentiality, integrity, and non-repudiation of data transparently and fairly to every user. As proof of concept, we demonstrate the initial implementation of a Decentralized Application based on EOSIO Blockchain integrated with robotic events that contain private data. The data gathered in this Thesis result from the experiments conducted with a social robot in a non-lab environment.

# Keywords

Blockchain, Confidentiality, EOSIO Blockchain, Healthcare, Human Robot Interaction, Integrity, Privacy, Social Robot, Radio Frequency Identification

# Resumo

Os Robôs Sociais podem gravar grandes fluxos de dados na forma de imagens, áudio, RFID, entre outras fontes sensoriais, o que pode ser fundamental para reforçar as Interações Homem-Robô. No entanto, os laços emocionais entre humanos e robôs podem levantar o problema de um robô aceder ou inferir informações profundamente privadas, por exemplo, estados emocionais. Portanto, do ponto de vista da privacidade, os robôs sociais podem ser um risco. Esclarecer a propriedade dos dados recolhidos por robôs foi destacado como uma preocupação no Regulamento Geral de Proteção de Dados da União Europeia, que exige 'privacidade por desenho'. Com este problema em mente, apresentamos **BlockRobot** – uma arquitetura baseada em Blockchain para garantir a privacidade nas Interações Homem-Robô. Esta arquitetura fornece aos utilizadores um sistema de gestão de identidade para acesso aos dados privados gerados durante as interações entre humanos e robôs. A arquitetura explora os benefícios do mecanismo de detecção de radiofrequência e da propriedade de imutabilidade Blockchain integrada com eventos robóticos. BlockRobot concede confidencialidade, integridade e não repúdio de dados de forma transparente e justa a todos os utilizadores. Como prova de conceito, demonstramos a implementação inicial de uma applicação decentralizada baseada em EOSIO Blockchain integrado com eventos robóticos que contêm dados privados. Os dados recolhidos nesta Tese resultam de experiências com um robô social em um ambiente não laboratorial.

# Palavras Chave

Blockchain; Confidencialidade; EOSIO Blockchain; Cuidados de saúde; Interação Robô-Humana; Integridade; Privacidade; Robô social; RFID

# Contents


**1 Introduction** **1**

1.1 Motivation . . . 3

1.2 Problem Statement . . . 3

1.3 Contributions . . . 5

1.4 Research Methodology . . . 6

1.5 Document Outline . . . 7

**2 Theoretical Background** **9**

2.1 Social Robots . . . 11

2.2 Privacy . . . 12

2.2.1 Definition of Privacy . . . 12

2.2.2 Privacy in Human-Robot Interactions . . . 13

2.3 Blockchain . . . 14

2.3.1 Types of Blockchain . . . 14

2.3.2 Consensus Protocols . . . 15

2.3.2.A Proof of Work . . . 15

2.3.2.B Proof of Stake . . . 15

2.3.2.C Delegated Proof of Stake . . . 16

2.3.2.D Byzantine-Fault Tolerance . . . 16

2.3.3 Hybrid blockchain and off-chain storage . . . 17

2.4 Digital Signatures and Asymmetric cryptography . . . 18

2.5 Cryptographic Hash . . . 19

2.6 Radio Frequency Identification . . . 19

2.6.1 Types of RFID . . . 20

2.6.2 RFID integrated with Robots . . . 20

**3 Related Work** **21**

3.1 Privacy Requirements . . . 23

3.1.1 GDPR Legal Requirements . . . 23

3.1.2 Ethical Privacy in HRI . . . 24
3.2 Types of Private Data Management Structures . . . 24
3.3 Blockchain and Private Data Management . . . 25
3.4 Blockchain Integrated with Robots . . . 26
3.5 Blockchain and RFID . . . 26
3.6 Blockchain disadvantages . . . 27
3.7 Discussion . . . 28

**4 Architecture Design** **31**
4.1 Objectives of the Proposal . . . 33
4.2 Architecture Overview . . . 33
4.3 Design Guidelines . . . 35
4.3.1 Correct data outline . . . 35
4.3.2 Data Persistence . . . 35
4.3.3 User Interface and Blockchain Transactions . . . 36
4.4 Main Algorithms . . . 36
4.4.1 Data Subject Classification Algorithm . . . 36
4.4.2 Time-Pairing Algorithm . . . 38
4.5 Identity Management . . . 39
4.6 Discussion . . . 41

**5 Demonstration** **43**
5.1 Architecture of the Prototype High-level View . . . 45
5.1.1 MEAN Stack . . . 45
5.1.2 Blockchain, Smart Contracts and Crypto Wallet . . . 47
5.1.3 Angular Frontend Logical View . . . 47
5.1.4 ExpressJS Backend Logical View . . . 48
5.1.5 EOSIO Blockchain Client . . . 50
5.2 Sequence Views . . . 51
5.2.1 Registration Process . . . 51
5.2.2 Authentication Process . . . 51
5.2.3 Write Data Process . . . 52
5.2.4 Access Data Process . . . 54
5.2.5 Delete Data Process . . . 55
5.3 Demonstration Use Case . . . 56
5.3.1 Experimental Protocol . . . 57
5.3.2 Experiments . . . 57

5.4 Results . . . 59

**6 Evaluation** **63**

6.1 Objectives of the Evaluation . . . 65

6.1.1 DSRM evaluation methods to validate artifacts . . . 65

6.1.2 Design-oriented evaluation principles for conceptual model . . . 65

6.2 Descriptive Evaluation . . . 66

6.3 Design-oriented evaluation principles for conceptual model . . . 68

6.4 Discussion . . . 70

6.4.1 Privacy Validation in HRI . . . 70

6.4.2 GDPR and Right to be Forgotten . . . 71

6.4.3 Security and Privacy Threats . . . 71

6.4.3.A Hashes Security . . . 72

6.4.3.B BlockRobot Application Programming Interface (API) as a Trusted Server 72

6.4.4 Blockchain Evaluation . . . 73

**7 Conclusion** **75**

7.1 Communication . . . 78

7.2 Limitations . . . 78

7.3 Future Work . . . 79

# List of Figures


2.1 A social interface creates a social robot (taken from [1]) . . . 11
2.2 A Social Interface components (taken from [1]) . . . 12
2.3 Peer-to-Peer network . . . 14
2.4 Blockchain Off-chain and on-chain data relation . . . 18
2.5 RFID system (taken from [2]) . . . 19

4.1 High-level System Architecture of BlockRobot . . . 34
4.2 ROS middleware and BlockRobot interactions - Classification of data owner model . . . 37
4.3 Time Pairing model: Image detected out of bounds of RFID detection . . . 39
4.4 Time Pairing model: Image detected in bounds of RFID detection . . . 39
4.5 Identity Management Model – Registration and Authentication . . . 40

5.1 Application Layered View . . . 46
5.2 Angular Frontend View . . . 48
5.3 ExpressJS Components View . . . 49
5.4 Blockchain Layered View . . . 50
5.5 Registration Process Sequence Diagram . . . 52
5.6 Authentication Sequence Diagram . . . 53
5.7 Write Data Process Sequence Diagram . . . 54
5.8 Access Data Sequence Diagram . . . 55
5.9 Delete Process Sequence Diagram . . . 56
5.10 Multi-Robot Cognitive Systems Operating in Hospitals (MOnarCH) Robot – the social robot used in the experiments . . . 58
5.11 Red-Green-Blue-Depth (RGB-D) Image captured by robot with one RFID identified person. 58
5.12 RGB-D Image captured by robot without RFID identified person. . . . 59
5.13 Graph showing the execution of system events during experiment 1. . . . 60
5.14 Graph showing the execution of system events during experiment 2. . . . 61

5.15 Performance comparison from both experiments. The matrices present the source (rows) and target (columns) system's events. . . . . 62

6.1 Sample data gathered in Database during the experiments . . . . . 66
6.2 Privacy Dashboard - User accesses his/her private data in **BlockRobot** . . . . . 67
6.3 Private Data visualization screen . . . . . 67
6.4 Privacy Dashboard - User's data after several interactions with robot in **BlockRobot** . . . 68
6.5 Privacy Dashboard - User cannot access other user's data . . . . . 69
6.6 Privacy Validation – GDPR Requirements and BlockRobot Design Guidelines mapping . . 70

# Acronyms

| | |
|---|---|
| **SR** | Social Robot |
| **RF** | Radio Frequency |
| **RFID** | Radio Frequency Identification |
| **HRI** | Human-Robot Interactions |
| **EU** | European Union |
| **GDPR** | General Data Protection Regulation |
| **BC** | Blockchain |
| **DAP** | Decentralized Application |
| **RGB** | Red-Green-Blue (color model based on additive color primaries) |
| **RGB-D** | Red-Green-Blue-Depth |
| **CA** | Centralized Authority |
| **DSRM** | Design Science Research Methodology |
| **IS** | Information System |
| **DS** | Data Subject |
| **PET** | Privacy-Enhancing Tools |
| **UI** | User Interface |
| **SM** | Smart Contracts |
| **PoW** | Proof of Work |
| **PoS** | Proof of Stake |

| | |
|---|---|
| **DPoS** | Delegated Proof of Stake |
| **ROS** | Robot Operating System |
| **API** | Application Programming Interface |
| **DBMS** | Database Management System |
| **TXID** | Transaction ID |
| **UID** | Unique Identification |
| **NoSQL** | Non Structured Query Language |
| **JS** | Javascript |
| **MEAN** | Mongo Express Angular Node |
| **ECC** | Elliptic Curve Cryptography |
| **BE** | Backend |
| **RPC** | Remote Procedure Call |
| **MOnarCH** | Multi-Robot Cognitive Systems Operating in Hospitals |
| **TPS** | Transactions Per Second |
| **POMS** | Product Ownership Management System |
| **ECDSA** | Elliptic Curve Digital Signature Algorithm |
| **ECC** | Elliptic Curve Cryptosystem |
| **IS** | Information Science |
| **BFT** | Byzantine-fault tolerance |
| **IPOL** | Instituto Portugues de Oncologia – Lisbon |
| **Mbot** | MOnarCH Robot |
| **HICSS** | Hawaii International Conference on System Sciences |
| **IEEE** | Institute of Electrical and Electronics Engineers |
| **BCT4ROS** | Blockchain Technologies for Robotic Systems |
| **AI** | Artificial Intelligence |

**ML** Machine Learning

**MITM** Man-in-the-middle

**DOS** Denial of Service

**BPDIMS** Blockchain-based Personal Data and Identity Management System

# 1

# Introduction

## Contents


1.1 Motivation . . . . . 3
1.2 Problem Statement . . . . . 3
1.3 Contributions . . . . . 5
1.4 Research Methodology . . . . . 6
1.5 Document Outline . . . . . 7

## 1.1 Motivation

Social Robot (SR)s are designed to assist humans socially. The way a person interacts with a social robot is quite different from interacting with an autonomous robot. People prefer to interact with robots with basic communication and social skills. Social interactions imply the implementation of a new broad set of features that require multidisciplinary research [3]. Scientists need to understand the philosophical, ethical, and legal layers that motivate typical human behavior. To create social robots capable of genuine social behavior, they need to interact with humans at an emotional level [4] and act socially expected by humans. The long-term goal is to assist humans in tasks, such as education, health, entertainment, communication, and tasks requiring teamwork.

On the one hand, SR can collect data from (among many other sensors) Red-Green-Blue (color model based on additive color primaries) (RGB) and Red-Green-Blue-Depth (RGB-D) images, record audio data, react to human voices, or detect Radio Frequency Identification (RFID) signals. RGB-D images are depth images in which each pixel's value in the image represents the distance to the object. They provide the ability to accurately sense and track humans and objects without the possibility of complete identification [5]. RFID signals give information regarding the position, and proximity can be inferred [6–8]. Another example is the growing attention towards Robots with tactile skills [9]. On the other hand, people tend to bond with SR that look like humans and are likely to interact with them at an emotional level [10], in general, within the bounds of the uncanny valley paradigm [11]. Moreover, SRs are enabled with advanced technologies such as natural language processing and image recognition [12]. Powered by Big Data and Machine Learning, the best algorithms for facial recognition allow robots to identify a person from a digital image or video. Moreover, with direct access to the Internet, SR can instantly access all data relating to some individuals once identified in the image.

Due to the variety and unprecedented volume of private data that SR can collect, they raise questions about privacy [13] and security [4], subjecting them to the General Data Protection Regulation (GDPR) [14, 15] – a European Regulation that imposes legal compliance to all the public and private companies. GDPR requires *'privacy-by-design'*, whereby data protection safeguards are built into technology early on.

## 1.2 Problem Statement

Sophisticated technology such as artificial intelligence and big-data-driven algorithms, which are essential technology of SRs, are still very oblivious to humans [16], which lead people to a lack of understanding about the underlying technologies of SRs. Besides, a robot's humanoid form can mislead the individuals' knowledge of the robot's recording capabilities, facing people with a lack of awareness regarding what robots can do [17]. People may expect that robots are limited in some ways because they

create expectations based on their appearance. For example, one may think that a robot cannot see objects behind himself, while, in reality, that robot may detect things behind the walls [18]. The amount of data that the robot can collect and infer from an interaction is also unclear.

Also, the independence and autonomy of robots allow them to locate humans in real-time almost anywhere. They can access new places and locations as never before, such as people's houses and work environments like offices or hospitals. As a result, SRs can impose many different privacy risks. In extreme cases, the government could use SRs to spy on people for military purposes [19]. SRs can work as information extractors and persuaders because people bond emotionally [20]. These issues are even more dangerous if data processing happens in the cloud, as data collected in intimate settings is transmitted, analyzed, and stored beyond the users' control.

Besides, even if SR is not *"malicious,"* in other words, it is not operated by a third party with intentions to extract sensitive information, *people have the right to their private data and, in particular, transparency in privacy during Human-Robot Interactions (HRI).* To illustrate a problem with a use case, we consider a Multi-Robot Cognitive Systems Operating in Hospitals (MOnarCH) SR operating in the healthcare environment in Instituto Portugues de Oncologia – Lisbon (IPOL). MOnarCH Robot (Mbot) assists hospitalized children and staff by interacting with them. It can patrol and socially interact with the people in the hospital corridors and rooms, play with hospitalized children and even act as a teaching assistant in the pediatrics classroom by projecting videos related to the class content [21]. It is common for some people to interact with the robot spontaneously. However, some other people do not like the idea of robot recording information in real-time. As a result, they avoid interactions because they do not trust him. The hospital staff members or patients who sympathize with the robot and trust him also do not have any means to access their private data collected by the robot. They may request the erasure of the images by talking directly with the programmer who has access to robotic software. However, such requests are not very practical. They require the programmer's work and availability and are not immediate. Nevertheless, people remain oblivious to what the robot *"saw"* them doing or what the robot *"knows"* about them daily.

In this line of reasoning, we defined the problem as **lack of control and transparency in privacy in HRI**, for example, in a healthcare environment. In Chapter 4, we describe in detail the proposed design solution, which solves the problem identified.

Recent GDPR privacy legislation presents the privacy rules to comply with making an individual's privacy regulated. Today, **transparency is a core principle** enshrined in *Art. 5 (1)(a)* of the GDPR. In European data protection, **transparency law is an obligation** imposed on data controllers to communicate a series of pieces of information, and to communicate them *"in a concise, transparent, intelligible and easily accessible form, using clear and plain language" (Art. 12(1)).* Demands for transparency need to consider what information would be of value and interest to potential users. Significantly, a lack

of transparency may affect the perceived trustworthiness of those responsible for the provision of such information.

Apart from prospective transparency, where a data subject is informed about the data processing beforehand, transparency requires retrospective transparency, meaning the ability to follow the data processing step-by-step, for audit purposes [22]. *Art. 22* GDPR gives individuals the right not to be subject to a decision based solely on automated processing that significantly affects him/her. Moreover, *Recital 71* GDPR gives the subject the right to obtain human intervention, express his or her point of view, and obtain an explanation of the decision.

## 1.3 Contributions

From the Research Directions in Privacy-Sensitive Robotics [23], it is clear that work regarding privacy in Robotics is a new field, and there is a handful of privacy-by-design solutions in HRI. If robotics is still a young field, what we call privacy-sensitive robotics is even younger. For these reasons and similar ones, it is crucial to grant humans a correct privacy-sensitive design in SRs, both to guarantee the people's safety and increase the broad social robot acceptance.

Data collected by SR need to be stored in data storage accessible to the data owners to enforce their privacy rights. Current centralized organizations provide security for the individual's data, as long as the Centralized Authority (CA) is trusted. However, a CA can hinder trust, as the examples of government surveillance [24] or Facebook's large-scale scientific experiment conducted without explicitly informing participants [25]. Besides the privacy enforcing techniques employed by a CA, there is growing attention to the concept of decentralized private data management paradigm. Decentralized solutions such as Blockchain (BC) provide high-level properties of **transparency**, **tamper-resistance**, and **provenance**. For that reason, Blockchain is an excellent candidate to comply with GDPR in HRI.

In this line of reasoning, this thesis' contributions are focused on developing a 'privacy-by-design' architecture that may serve the GDPR in the way of enhancing privacy in HRI. **Concretely, this thesis contributes with the artifacts:**

- Software design and architecture for BC-based Decentralized Application (DAP) for HRI, considering the GDPR requirements.
- Development and test of a prototype – private permissioned EOSIO BC-based DAP integrated with a Social Robot.

Furthermore, we will show how BC, when integrated with SR, can improve users' privacy in HRI resulting in a better experience. Using a case study, we demonstrate the utility of a BC-based architecture encompassing an implementation of a DAP based on an EOSIO Blockchain, integrated with a SR

moving in a hospital. Our fundamental approach is that participants (in the case of the scenario used to collect data in this thesis, hospital staff, patients, and visitors) can monitor and verify what information was recorded during their interaction with the robots and guarantee that private data is outlined correctly to participants. Furthermore, when integrated with robotic events, with this proposed architecture, we will enforce individuals' privacy by giving them the possibility to manage their private data transparently and fairly. The experiments are performed in hospital corridors and rooms. The execution data logs are computed with process mining techniques [26] to analyze the performance and bottlenecks existing in the case study.

The software design was supported by rich text figures illustrating the high-level architecture and the main algorithms. All code and documentation necessary for the proposed solution is available on Github and is open-source.

- https://github.com/vvasylkovskyi/eos-web

## 1.4 Research Methodology

Design Science Research Methodology (DSRM) [27] was the methodology used for this thesis. It provides principles and procedures for conducting researches of Information Science (IS). DSRM is an iterative method constituted by six steps:

- **Problem identification and motivation** – Definition of the specific research problem and justify the value of a solution. In this thesis, the problem is the lack of control and transparency in privacy in HRI (Chapter 1).
- **Definition of the objectives for a solution** – Inference of the objectives of a solution from the problem definition and knowledge of what is possible and feasible (Chapter 3). Here we define what a solution should accomplish to be relevant to our research question and our identified problem. In other words, we present a description of new artifacts and how they should be used to solve the identified problem.
- **Design and development** – Detailed description of what artifacts must be developed, their characteristics, and desired functionality to solve the identified problem. This description is often accomplished using potential constructs, models, or even the creation of the artifact. In this thesis, the artifact is a software design and architecture for 'privacy-by-design' applications in HRI, and its design is described using rich text figures to illustrate the architecture and the main algorithms, ArchiMate Views and Sequence Processes (Chapter 4).
- **Demonstration** – Demonstration of the use of the artifact to solve one or more instances of the problem. In this thesis, we demonstrate the software design usefulness by developing and testing

a prototype based on the software design defined in Chapter 4 integrated with a social robot. The tests and experiments occur in a healthcare environment with one social robot and multiple users (Chapter 5).

- **Evaluation** – Observation and measurement of how well the artifact supports the solution to the problem. We accomplish this by comparing the proposed software design to the alternatives existent nowadays HRI. Also, we apply the qualitative analyses of security and GDPR compliance of the proposed prototype solution (Chapter 6).

- **Communication** – Communicate the problem and its importance, the artifact, its novelty and utility, the rigor of its design, and its effectiveness to researchers. The solution for the problem identified in this thesis was communicated to Institute of Electrical and Electronics Engineers (IEEE) international conference and Hawaii International Conference on System Sciences (HICSS), and a Blockchain Technologies for Robotic Systems (BCT4ROS) workshop (Chapter 7).

## 1.5 Document Outline

This document is structured as follows:

- **Introduction** (Chapter 1) – states the motivation, the problem of the research, what it is going to achieve, contributions, and the research methodology adopted.

- **Theoretical Background** (Chapter 2) – offers a comprehensive theory to help the reader to understand the technologies involved in a solution.

- **Related Work** (Chapter 3) – provides an overview of the present state of the art regarding social robots, privacy, privacy-sensitive robotics, Blockchain, and its utility in modern technology.

- **Design Architecture** (Chapter 4) – describes in detail the proposed design for privacy-by-design DAPs.

- **Demonstration** (Chapter 5) – illustrates the demonstration of the proposed design architecture usefulness in real human-robot interaction. As a proof of concept, this chapter also demonstrated the results of experience with a prototype based on the design of architecture proposed in Chapter 4

- **Evaluation** (Chapter 6) – illustrates the evaluations of the prototype and discusses the utility of the artifact through a series of analytical simulations and descriptive methods.

- **Conclusion** (Chapter 7) – concludes the thesis by summarizing the document content, this thesis scientific achievement, including the communication of the work to international conferences, and discusses the future work needed.

# 2

# Theoretical Background

## Contents


2.1 Social Robots . . . . . 11
2.2 Privacy . . . . . 12
2.3 Blockchain . . . . . 14
2.4 Digital Signatures and Asymmetric cryptography . . . . . 18
2.5 Cryptographic Hash . . . . . 19
2.6 Radio Frequency Identification . . . . . 19

This chapter presents the key background theory relatively to the Social Robots, Privacy, Blockchain, Cryptography and RFID technologies.

## 2.1 Social Robots

Among many definitions of the robot, the simplest one seems to be by definition of Kaplan [28], by whom a robot is an object that possesses three characteristics:

- It is a physical object,
- It is autonomous,
- it is situated in the environment.

In other words, a robot is a programmed physical entity that perceives and acts autonomously within a physical environment, which influences its behavior. Besides being a physical entity, it can manipulate not only information but also material things.

A robot is not a social robot as is. It needs specific communication capabilities to become social. In simple terms, the difference between a social and non-social robot is the *Social Interface* (Fig 2.1). In opposition to the non-social robots (e.g. factory robot), the social robots' main design feature is that they should engage with humans on social interactions. Therefore the SR should have basic notions about the humans behavior in society, circumstances of the communication, and other sociological variables that affect interactions. Not only SR needs to understand humans' social behavior, but it also has to behave socially expected by humans.

People expect SR to act like humans. This phenomenon is called **anthropomorphism** – the attribution of human-like qualities to non-human agents or objects [29]. Humans tend to anthropomorphize robots because this allows them to explain things that they do not understand. Based on that, the appearance of the social robot is essential. The more human-like robots can display a broader range of emotions by mimicking the individual's facial expression. In a study from [1], the robot's appearance is defined as a *Social Form*, and it produces the *Social Function*, meaning that the robot's appearance (form) represents his state of "mind" (function). The most basic example can be the robot's movements: for instance, a robot turning his head toward the human with whom he is interacting is an expression of


**Figure 2.1:** A social interface creates a social robot (taken from [1])

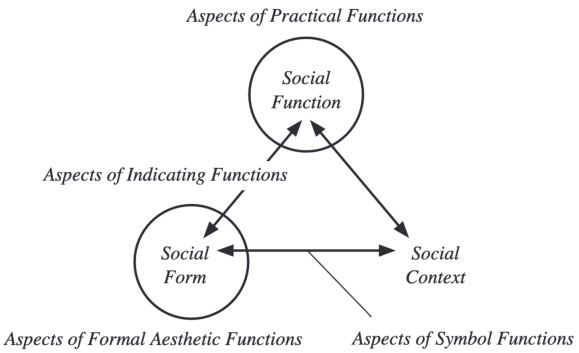


**Figure 2.2:** A Social Interface components (taken from [1])

attention. Furthermore, the *Social Context* determines the form and function. People tend to display a difference in social behavior according to social context. For example, in the bar, individuals expect the bartender to act in a certain way. So if the SR looks like a bartender, people will expect him to act like one. Fig. 2.2 presents the full illustration of Social Interface.

## 2.2 Privacy

Previous research indicates that the scope of privacy can span many disciplines varying from philosophical to legal discussions. However, the systematically reviewed papers and surveys on this subject enabled us to qualify the essential definitions and concepts. First question: *What is privacy*?

### 2.2.1 Definition of Privacy

Privacy comes from the Latin adjective *privatus*, meaning ”set apart, belonging to oneself (not to the state), peculiar, or personal.” This is used in contrast to *publicus* and *communis*, which have the expected meanings. However, there is no universal definition of the concept of privacy. One of the first and certainly one of the most influential definitions comes from Warren and Brandeis (1890) [30], who defined privacy as ”the right to be let alone.” Despite being simple to define, privacy definitions tend to be generalized. For this thesis's scope, we restrict the scope of privacy to the following, most prominent definitions [31]:

- **Privacy as Control over Information** – is defined as the prohibition of disseminating confidential information through eavesdropping, surveillance, and wiretapping, to name a few. (William Prosser, Edward J. Bloustein (1964)) [32],
- **Privacy as Intimacy and Social Relationships** – The protection of one's assets or interests or

protect a person against information leaks can improve the use of information for social relationships (Rachels (1975)) [33],

- **Privacy as Restricted Access** – provision of secrecy when no one has access to information (physical or mental) about the person or through anonymity when the identification is not possible (Ruth Gavison (1980)) [34].

### 2.2.2 Privacy in Human-Robot Interactions

Many social robots are equipped with recording devices such as kinetic cameras, microphones, and RFID lasers. When connected to the internet, they may raise questions about privacy [13] on legal and ethical grounds [14, 15]. We distinct several privacy types that robots can affect while interacting with users. At first, we consider the physical privacy that revolves around physical access to an individual. This type of privacy risk derives from the ability of robots to move physically in space. For example, in healthcare, the assistant robots proved to be very useful. However, by being autonomous, they can interfere in physical space with humans in places where we would not expect, e.g., bedrooms and bathrooms. As a result, they may be subject to witness sensitive or compromising situations. For instance, they could be used for government surveillance [19].

Furthermore, the Artificial Intelligence (AI) and Machine Learning (ML) algorithms allow robots to infer sensitive information beyond their devices' recording capacities. They could potentially collect "background" data without awareness or consent by users. There is evidence that people do not understand all the robot's recording capacities [18]. Moreover, people tend to anthropomorphize robots within the bounds of the uncanny valley paradigm [11], which increases the degree to which robots invade personal space. Therefore, robots can access sensitive information that affects individuals' informational privacy, meaning they can access restricted information, be information extractors, and persuaders. Moreover, people tend to interpret the robot's actions on an affective level, even if there are no emotional cues in the robot's outward appearance [35]. Robots can even access people's emotional and mental states [10].

In light of all the examples of privacy risks listed, it becomes clear that social robots may interfere with privacy. GDPR states in its principles rules on how to increase privacy in information systems [15]. However, the precise mapping between privacy while interacting with robots is in its very beginnings. Due to the recent advances in robotics, a new research area has emerged – Privacy-sensitive robotics [23], which aims to fill the gap between knowledge of privacy and robotics. Privacy can also be influential in the adoption of robots. The way that people perceive privacy affects their relationship with the robot. Low privacy in HRI may result in a lack of trust and non-willingness to engage with robots [36].

## 2.3 Blockchain

Blockchain is a distributed decentralized append-only database with a set of protocols or rules (known as the consensus system) that restrict the format and content of data that may be added. Blockchain architecture gives participants the ability to share a ledger updated through peer-to-peer replication each time a transaction occurs. *Peer-to-peer replication* (Fig. 2.3) means that each participant (also called a *node*) in the network acts as both a publisher and a subscriber [37]. The Blockchain network is economical and efficient because it eliminates duplication of effort and reduces the need for intermediaries. It's also less vulnerable because it uses consensus models to validate the information. Transactions are secure, authenticated, and verifiable.

A BC network has the following key characteristics:

- **Consensus** – for a transaction to be valid, all participants must agree on its validity.
- **Immutability** – it is not possible to change a transaction after it is recorded on the ledger. If a transaction is in error, a new transaction must be used to reverse the error, and both transactions are then visible.
- **Provenance** – participants know where the asset came from and how its ownership has changed over time.

### 2.3.1 Types of Blockchain

Blockchain networks are categorized based on their permission model, which determines who can maintain them (e.g., publish blocks). There are three variations of BC based on the permission model:

- **Permissionless** – any party can read from and write to the BC. Such BC types are suitable for the types of networks where there is no sensitive information, as all the transactions will be publicly available.

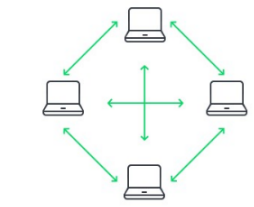


**Figure 2.3:** Peer-to-Peer network

- **Permissioned** – access to the BC is restricted to authorized parties. These types of BC networks are based on the initial identity establishment, and therefore to participate in the network, it is required to prove one's identity.

- **Partially Permissioned** – some actions, for example, read access, are granted to anyone, but other activities, for example, write access, may be limited to authorized parties. This type of configuration is suitable for enterprise-grade BC, especially in industries where data may not be entirely public. Furthermore, it is possible to configure the permissions on the fine-granular level about who can read/write based on different types of transactions.

### 2.3.2 Consensus Protocols

The consensus algorithm – is a set of rules that provide each member of a BC the guarantee that the transactions in the BC are valid [38]. It allows all the nodes on the network to agree on which transaction is valid and what block should be appended to the Blockchain.

#### 2.3.2.A Proof of Work

The most widely utilized consensus algorithm throughout the Blockchain ecosystem is the **Proof of Work (PoW)** algorithm. An example of a BC platform that uses PoW is a Bitcoin [39] a first and largest distributed digital currency system that allowed the performance of electronic transactions of value (Bitcoin) between two individuals without the intervention of a third party or a central authority to rule the transaction. The Bitcoin and other Bitcoin system crypto-currencies rely on a PoW to confirm chain transactions and create new blocks through particular nodes called miners. The miner nodes perform complex mathematical hash functions to find new, very correctly formatted, hashes. If the miner resolves a given complex mathematical function first, then the block is included in Blockchain, and the miner is rewarded. The probability of resolving the hash first is proportional to the amount of energy spent. The resolved mathematical function serves as proof that the required amount of energy has been spent (proof of work). Although the PoW algorithm is very straightforward and useful, there are several factors such as its overall cost, high power consumption, and security (in regards to some attacks such as the 51 percent attack [40]) – that downplay its overall functionality.

#### 2.3.2.B Proof of Stake

Another impressive BC technology that is worth mentioning is the Ethereum, introduced by Vitalik Butrin [41]. Ethereum overcomes some of the Bitcoin limitations, even though both technologies are unique in their application. The main contributions are full Turing-completeness, meaning that Ethereum

supports all types of computations, including loops [42], achieved by introducing the Smart Contracts (SM) – a set of cryptographic rules involved in transactions.

Ethereum even though initially adopted the PoW as its consensus algorithm has been attempting to move away from this consensus protocol toward the **Proof of Stake (PoS)** protocol. The core idea of PoS is to use a coin-based election of a block producer. The "coin" is a scarce and well-distributed resource that restricts the voting mechanism of permissionless blockchains. Unlike in PoW, where all miners have an equal probability of mining the next block, in PoS, each miner gets the chance to become the next block's miner proportional to the amount of money that he has staked (locked). Based on that, PoS has the advantages over PoW as it does not need to have the massive computational power to mine neither it needs to build large mining "farms."

#### 2.3.2.C Delegated Proof of Stake

One of the variations of PoS is a **Delegated Proof of Stake (DPoS)** – a consensus mechanism used by EOSIO Blockchain-based DAPs that implements a voting mechanism designed to prevent actors from controlling multiple nodes in the network. The consensus comes from the voting and delegation in the network.

Similarly to PoS, in DPoS, the voting power is represented by staking the resources. The difference is that instead of being elected to produce a block, the votes determine the representatives. The network nodes then vote with their voting power for other nodes to be elected as block producers. The top 21 block producers with the most votes become the block producers for the next epoch. If a block producer fails or acts maliciously, the DPoS algorithm intends to vote these block producers out.

DPoS's main benefits are speed, low energy usage, and incentives for honesty. However, its main weaknesses are centralization – number of people who can act as block validators is limited, and concentrating voting power – users who hold a large number of tokens will have significantly more voting power and thus more influence in deciding delegates.

One of the examples of the BC using DPoS consensus algorithm is the EOSIO [43] BC. EOSIO – is a decentralized enterprise system that executes industrial-scale DAPs. In contrast to Bitcoin [39] or Ethereum [41], EOSIO claims to scale to millions of transactions per second, and achieving far higher performance throughput, i.e., up to 8,000 Transactions Per Second (TPS) within a single thread, and unlimited for multiple-threaded cases [43].

#### 2.3.2.D Byzantine-Fault Tolerance

The permissionless BC protocols rely on the consensus by using each node working hard to produce a new block via spending his CPU energy like in a proof-of-work algorithm. It is a very costly algorithm. There is a lot of energy waste. The proof-of-stake solves the energy spending problem, but there are

still other significant problems, such as a potential risk of creating forks [44] of blocks, meaning that the BC eventually will grow into multiple BCs.

Permissioned BC protocols such as Hyperledger Fabric [45] or Tendermint [46] escape from such a costly consensus mechanism where each node works on block creation. They run a Blockchain among a set of known identified participants. These participants have a common business goal but do not trust each other. Therefore, they can rely on a traditional Byzantine-fault tolerance (BFT) consensus [47], i.e., the consensus where up to one-third of block validator nodes can be faulty.

Since the participants in a permissioned BC are all identified, they can rely on committee-based consensus protocols [48] – meaning that only a subset of participants decides on which block to produce. Note that the voting process dynamically determines the subset of participants once each block is appended. The voting process in BFT based algorithms usually consists of the PoS algorithm [49]. This step is also known as a *Selection Mechanism*, and it is responsible for selecting the being block validators. Once selected, they attempt to reach a consensus on one block at a time, where the block is a list of transactions. Each block has its proposer – the node that proposes a block. The validators then vote whether to accept the proposed block or not. If validators' decision is not to accept the block, then the next iteration of consensus begins where new block proposer emerge, and the process repeats itself. This consensus algorithm is known as the Repeated Consensus Algorithm [48].

### 2.3.3 Hybrid blockchain and off-chain storage

Most of the companies already have a storage system to keep their data. As these storages already exist (ex. Cloud Database), the data kept there only needs to be referenced by the Blockchain Application Programming Interface (API). Off-chain data is any non-transactional data that is too large to be stored in the BC efficiently, or requires the ability to be changed or deleted. Examples of off-chain data are PDF documents, images, video files, or any sensitive data that is not supposed to share.

Furthermore, in its silo, Blockchain with off-chain storage is not a platform where different business entities can see each other's data. Only the references of data are stored on-chain (representation Fig. 2.4). By placing only references to data on-chain, the following advantages present:

- **Integrity** – the data modifications are recorded on BC, and in case some modifications on some data occur, all business entities can see that,
- **Accountability** – the history transactions are kept on BC,
- **Privacy** – the companies do not have to share all their sensitive data. It is enough to share some data when requested, and BC provides the ability to verify data integrity. Besides, storage of the private data off-chain allows it to be deleted, as required by GDPR' "Right to be Forgotten" [14, p. 32].

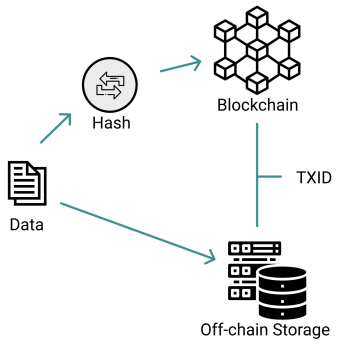


**Figure 2.4:** Blockchain Off-chain and on-chain data relation

Usually, the data stored on BC are the cryptographic hashes of data, and off-chain repositories keep the data itself [50].

## 2.4 Digital Signatures and Asymmetric cryptography

In Blockchain, the **digital signatures** are used for verifying the authenticity of digitally signed transactions. When there is a need to store some Blockchain data, a participant must create a new transaction, which requires authentication, accomplished by digital signature. The person proves his/her identity by digitally signing the transaction.

The underlying technology for the digital signature is asymmetric cryptography [51]. **Asymmetric cryptography** (also known as Public-key cryptography) is an encryption technique used for a long time, which uses a pair of keys: a *private* and a *public* key. It enables encryption and decryption of messages using two separate keys so that a message is encrypted with one key is later decrypted with the other key. *The private key is only known to the owner of the key and should remain secret, while the public key is known to everybody in the network*. Each member of the BC network is required to have private/public keys to be able to interact. In Blockchain's digital signature, the public-key cryptography ensures authenticity by encrypting the private key transaction data. Since only the owner knows the private key, decrypting the transaction with the owner's public key proves he initiated it. Therefore, the digital signature is proof that a user who is creating a transaction owns a private key that confirms his identity. By these properties, digital signatures provide non-repudiation of the actions and authenticity. The typical digital signature algorithms used in BC include Elliptic Curve Digital Signature Algorithm (ECDSA) [52].

## 2.5 Cryptographic Hash

The use of hash in Blockchain provides data integrity and the chaining of the blocks [37]. A successful hash function receives as input data and transforms that data into a fixed size hash digest. Traditionally, Blockchains use the *Sha-256* hash function. The main characteristics of hash functions, and Sha-256 in particular, are the following:

- **Determinism** – the same input will always produce the same fixed size digest (256 bits in case of Sha-256).
- **Preimage-resistance** – meaning that it is not possible to recover the original data from the hash.
- **Collision resistance** – meaning that two different inputs can not result in the same hash [53].

Given these properties, storing hash on-chain provides data verifiability because Blockchain is immutable.

## 2.6 Radio Frequency Identification

Radio Frequency Identification (RFID) is a wireless technology capable of automatic and unambiguous identification without a line of sight by extracting a unique identifier from microelectronic tags attached to objects [54]. RFID technology-enabled contact-less ID tags are useful in scenarios where identification features with minimal false positives are required.

There are three necessary components that all RFID contains. The RFID reader, antenna, and an RFID tag (Fig. 2.5). An RFID tag is a chip typically capable of carrying a few bytes of data [2].

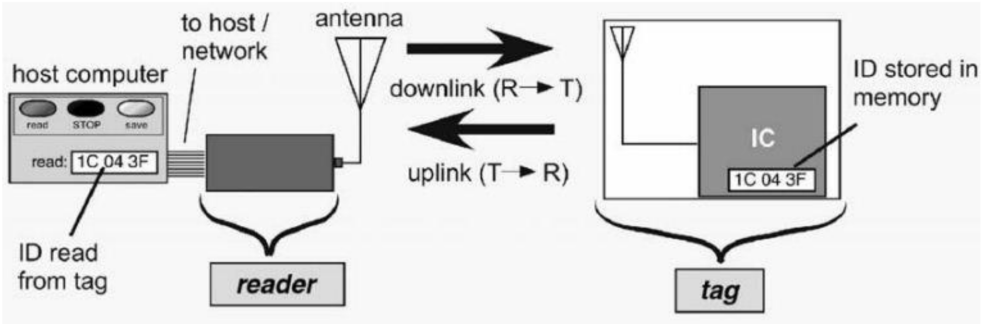


**Figure 2.5:** RFID system (taken from [2])

### 2.6.1 Types of RFID

There are many types of RFID, but at the highest level, we can divide RFID into two classes: *active* and *passive* [55]

- **Active** – Active tags required a power source, so they are either enabled with battery or connected to the source. One example of an active tag is the transponder attached to an aircraft that identifies its national origin.
- **Passive** – Passive tags, on the other hand, do not require batteries or maintenance. They are activated by the electromagnetic field generated by the RFID antenna. Passive RFID tags are small enough to fit into practical devices, such as a smart card or ID badges.

### 2.6.2 RFID integrated with Robots

Moreover, the RFID technology can enable the robot with an Radio Frequency (RF) reader to detect the RFID tags in the vicinity. Further, humans can use ID badges with embedded RFID tags. By detecting RFID tags, the robot can infer the proximity of the tags and adjust his behavior, such as assuming socially acceptable distance when interacting with humans or merely avoiding them by assuming a more significant distance than the social interactions proxemics [6–8]. .

Besides, by enabling people with RFID tags, social robots have a robust means to identify many people simultaneously. Moreover, the use of RFID tags for HRI purposes has the advantage of

- being a cheap technology, as passive tags are used;
- if necessary, preserve the anonymity of the users, e.g. of people wearing a tag, that is, the information being detected is simply that of the tag and may not convey any true information about the physical person carrying the tag [8].

# 3

# Related Work

## Contents


3.1 Privacy Requirements . . . . . 23
3.2 Types of Private Data Management Structures . . . . . 24
3.3 Blockchain and Private Data Management . . . . . 25
3.4 Blockchain Integrated with Robots . . . . . 26
3.5 Blockchain and RFID . . . . . 26
3.6 Blockchain disadvantages . . . . . 27
3.7 Discussion . . . . . 28

This chapter includes part of the **definition of the objectives for a solution** step of DSRM. Here we will present the current state of the problem and the existing solutions up to date. Further, with this information, we will infer the goals of the proposed solution (Chapter 4) to the problem defined (Chapter 1). Solving privacy in HRI has a broad and multidisciplinary discussion. First, we synthesize the GDPR legal privacy requirements and ethical privacy in robotics. Further, we exemplify related work regarding the benefits of BC technology for increasing privacy and robotics.

## 3.1 Privacy Requirements

According to *Art. 4* of the GDPR, *"personal data means any information relating to an identified or identifiable natural person Data Subject (DS); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors [...]."*, even when data are encrypted or hashed, it qualifies as personal data under European Union (EU) law [15].

In our research, The primary GDPR principle involved is the *identify the data subjects* related to the data classified as private data in HRI, specifically in Social Robots. Data subjects in HRI are all the people that interact with robots directly or indirectly. Even when humans walk by the robot, it should be considered as an HRI producing private data. SR development should comply with the GDPR privacy requirements, on both legal and ethical grounds. Next, we present the GDPR legal and ethical requirements for privacy in HRI.

### 3.1.1 GDPR Legal Requirements

By analyzing the GDPR requirements from official source [56] and reducing its scope to the context of robotics, we present in detail the requirements:

- **Fairness and Transparency** – Data subjects should be aware of the processing purposes and provided with proper notification and information regarding their scope. Regarding fairness, M. Fink in [14, p. 64] defines fairness as reasonableness, meaning that personal data is processed in a manner that would be expected by data subjects and that they are not "tricked."
- **Principle of purpose limitation** – data controllers are compelled to use collected data for specific and well-defined purposes.
- **Data minimization** – only necessary data are collected, and redundant data or the one beyond the scope are not collected.
- **Storage limitation** – the personal data should be stored no longer that is necessary for the purpose for which the personal data are processed.

Besides, the GDPR states in its principles and various *Articles (e.g. 6, 24, 32-34)* that **'privacy-by-default'** and **'privacy-by-design,' anonymization**, **encryption** and other **Privacy-Enhancing Tools (PET)** should be used. Developing robots that support PET should increase user trust and public acceptance of robots and enhance user privacy.

### 3.1.2 Ethical Privacy in HRI

Although from an *ethical* perspective, we foresee that users' should be given both *privacy by opacity* and *information transparency*.

- **Privacy by opacity** – meaning that data subjects should be assured that there will not be unauthorized private data disclosure to other data subjects. This requirement is followed by sub-requirements, such as:

  1. **Integrity** – refers to the possibility to verify the accuracy and consistency (validity) of data, and that it was not compromised in some way.
  2. **Confidentiality** – meaning that the data remains undisclosed to the public.

- **Information Transparency** – Regarding transparency in privacy, a critical assessment of the provenance and accountability is required:

  1. **Provenance** – Users should be assured in some way that the data they have access to indeed comes from their interaction with the robot.
  2. **Accountability** – The data controller should notify the data subject of all the data processing activities made not only in *prospective*, the future data processing but also in *retrospective*, possible audit of data processing activities.

The above principles should altogether assist in providing individuals with **accuracy** regarding data records, in the sense that the data they have access to is indeed trustworthy and correctly outlined to participants. Further, individuals should be able to *access* their private data by providing a token of access to the data in an accessible way. Especial attention is deemed on the *"Right to be Forgotten"* – the user **can withdraw the consent or delete** his/her private data seamlessly.

## 3.2 Types of Private Data Management Structures

Given the sensitive nature of the data generated by a robot, it is worth considering all the different data storage systems available and evaluating their degree of security and privacy. Nowadays, companies have at their disposal a vast diversity of services to store their data. For instance, cloud-based storage

is a quite popular solution among enterprise organizations because of its rapid provision of service and minimal effort with service interaction [57].

However, this approach reveals substantial security risks, especially in private data, as there is a massive possibility of losing data control [57]. Besides, extensive formularies about privacy policies that centralized data storage entities provide are usually not well understood by the participants or not read.

One particularly interesting concept is the MyData Human-Centric Personal Data Management [58]. The MyData proposal aims to give the privacy back to users by shifting the organization's infrastructure to the personal data management focus. Concretely, in MyData's scope, users have a better overview of where their data is stored, who uses it, and who can access it, and ultimately enable users to manage and control their data. It is a consent-based paradigm in which users should manage the permissions over their private data separately from the data itself.

## 3.3 Blockchain and Private Data Management

In this section, we exemplify how Blockchain improves data management when integrated with traditional databases. Furthermore, we present some related work regarding enterprise solutions on private data management to integrate BC technology.

In the scope of access control, Guy Zyskind et al. propose a decentralized privacy solution using a BC to enforce a self-sovereign access-control manager [59]. In their system, the Blockchain, by its immutability and consequent trust property, serve as a replacement for a centralized entrusted authority, which usually manages the private data access. The decentralization of trust places private data back into the hands of data owners who can control the data while an external database – off-chain repository – stores the private data, which enables the participants to delete their private information. Further, several studies conducted active research about the benefit of Blockchain technology for medical data access in the healthcare domain. In MedRec [60], a decentralized medical record management system was proposed. The integration of electronic medical records with Blockchain intends to increase the verifiability of the information.

Binh Truong et al. [61] propose a solution that tackles the ongoing problem of data integrity when centralized authorities store the information. Here, the Blockchain-based solution enhances data transparency by providing the immutable audit of data accesses on Blockchain, while the data itself is protected on off-chain storage. A Blockchain-based Personal Data and Identity Management System (BPDIMS) researched Blockchain-based solution for Personal Data and Identity management system [62]. To align private data management with GDPR compliance, the BC with digital signature verification proved to be useful to enhance the identity management for data access.

## 3.4 Blockchain Integrated with Robots

This section exemplifies how the Blockchain can assist robotic systems and present some of the use cases. Furthermore, we are going to describe why and how Blockchain can enhance privacy in HRI.

E. Castelló Ferrer et al. [63] attempts to solve privacy matters related to using personal information with blockchain technology and sharing data and machine learning models. The architecture proposed by them illustrates a different approach regarding private data generated during HRI. While robots generate personal data from patients' healthcare records, these data references are recorded on the blockchain ledger, giving the user transparency and the ability to verify his private data and manage that data.

A research conducted by Bruno Degarding and Luís A. Alexandre [64] shows how Blockchain can assist in auditing the robotic events. Furthermore, the Blockchain smart contracts allow us to use the information from different robots and create action-triggers based on the contracts that are stored and verified on the Blockchain.

The following contributions made by Miguel Fernandes and Luís A. Alexandre using Tezos's technology [65] propose using Blockchain to audit robotics events. This study contemplates the use of BC technology to solve the problem of keeping accurate, immutable records of robotic actions in a factory environment. The event records, which will be stored in a BC, can then be used for further goals such as understanding and improving manufacturing productivity.

Also, more studies regarding the integration of a BC and robots illustrate that robots will become more efficient in cooperation between themselves because the Blockchain will provide them a comprehensive knowledge share base by being a verifiable and immutable single source of truth [66].

## 3.5 Blockchain and RFID

The combination of RFID with Blockchain also presented impressive results. Namely, RFID and Blockchain has gained popularity in improving security and transparency in supply chain management. For example, in [67], The combination of RFID and BC technologies seem to assist in accountability in the business of building materials' supply chains and logistics systems in general. Placing RFID data on BC immutable ledger has a promising future. It reduces losses from the influence of the human factor and intentionally false information and resolves trust between the participants.

A new case proposed the RFID integration with Blockchain for the Product Ownership Management System (POMS). K. Toyoda et al. [68] proposed to persist in BC all the RFIDs related to the product under the retailer's identification. In this way, when the customer is buying a product, he can verify on the BC platform whether the RFID attached to the product indeed belongs to the retailer, preventing the product copying in the post supply chain by duplicating of RFID tag.

The one-way identification method of RFIDs in supply chain management uses Blockchain to record the one-time pad authentication tokens for RFID tag authentications [69]. The tokens can be immutably recorded and timestamped on the Blockchain. Consequently, the Blockchain acts as the go-to point to verify the previous usage of the RFID tag. It improves the security of RFID tags by checking on duplication detections and preventing the serial number's counterfeit on RFIDs.

In this paper, we use RFID tags embedded in smart cards for the correct human identification by robots in HRI. Further, we propose to combine the RFID uniqueness and anonymity features to identify humans with the Blockchain immutability property for identity management purposes.

## 3.6 Blockchain disadvantages

Besides the clear advantages that Blockchain can bring into robotic systems and privacy altogether, it is worth to notice that Blockchain technology is still very new and is yet to mature, and have several disadvantages.

Regarding permissionless Blockchains, the main disadvantage is the high energy consumption. The experience with Bitcoin-like [39] Blockchains demonstrated that high power consumption is needed to keep a real-time ledger, and the more the nodes, the more is the waste of power because each node works hard to validate a block. Whenever the new node appears, it connects to the rest of the nodes in the network and initiates reaching a consensus. Moreover, the signature verification challenges the Blockchain because each transaction must be signed with a cryptographic digital signature, requiring a significant amount of computing power. Blockchain scalability is another problem. Ethereum [41] and the distributed ledgers based on the PoS consensus may happen to overcome the high energy consumption limitation, but due to the nature of PoS algorithm, BC forks can happen. Besides, the most popular crypto-currencies still have a low processing transaction speed compared to centralized solutions [70]. When more nodes are participating in the network, more nodes will have to verify each transaction, which potentially implies more time to verify each transaction.

Regarding permissioned BC platforms, the consensus protocols are mostly committee-based. Mainly because everybody in a permissioned network knows the other participants' identities. By being committee based, the subset of BC members can be elected as block validators. As a result, there is no need to have all the nodes working hard to validate blocks, and there is no energy waste. However, this may imply a partial centralization of control over the network, and, additionally, participants have to reveal their identities, which may imply some extra concern regarding participants' privacy.

Besides, the immutability property, while advantageous because of the transparency feature, storing the private data on Blockchain may not be such a good idea. GDPR requires the possibility to erase private data when the data owner decides to do so [14, 50], and it is not possible to erase nor modify the

data on Blockchain by design.

## 3.7 Discussion

Through this chapter, we exposed the present state of the art regarding the private data and GDPR compliance in HRI, more specifically, involving SR. As opposed to nonsocial robots, social robots interact with people. To interact with humans naturally, many social robots rely on sophisticated AI and collect large amounts of data from their users and their environment. Current generations of social robots are equipped with connected sensors, cameras, rangefinders, accelerometers, and GPS sensors (Calo, 2010b) [19]. Besides, current social robots are wireless and connected to computers or the Internet, thus transmitting data in real-time. As a result, users' informational privacy is endangered by social robots' increased capacity for data collection.

Researchers developed several techniques to enhance personal data protection, including data anonymization from a security perspective. Moreover, developing robots that support PET is encouraged by GDPR and should increase user trust and public acceptance of robots and enhance user privacy.

Besides the privacy enforcing techniques employed for the centralized data management authorities, there is growing attention to the concept of decentralized private data management paradigm. Decentralized solutions such as Blockchain, at first, seem to be incompatible with privacy requirements due to its immutability property, which implies that there is no way to delete private data from Blockchain. However, if designed with privacy in mind, it is possible to create a privacy-by-design application based on Blockchain technology. First, the Blockchain provides immutability in the records. Therefore, if connected to robotic events, it becomes possible to audit data recorded by robots, including private data. Second, if integrated with the off-chain database, Blockchain can assist in the verifiability of private data. The core idea here is to place private data on the off-chain database and store that personal data references on-chain so that the data can be further verified against the BC immutable ledger. The data subject can delete the private data itself at any time because it is stored off-chain. Third, by being immutable, BC gives the ability to audit all the accesses to the private data. As a result, the end-user has transparency regarding all access to his data since its very beginnings. Finally, BC provides distributed consensus, which means that no CA will have to manage the identities and secret keys. Without CA, BC eliminates the risk of secret keys leakage.

Considering all this information, we see a Blockchain-Based DAP as the best possible option for the privacy-by-design solution to solve the problem of lack of transparency in HRI.

Having evaluated the technical aspects of BC technology, we decided that private permissioned Blockchain is the most appropriate to succeed in improving the privacy of individual data because it

provides restricted access to the network. While in the public Blockchains, everyone can join the network in the BC, which does not guarantee any data privacy. Bear in mind that when using private permissioned BC, the identities of participants are known in the network. The knowledge of individuals' identities can be a privacy risk. Therefore, the architecture design will address these questions to avoid any security breach.

After a careful assessment of the design of the DAP for increasing privacy in HRI, we have concluded that the solution will have to be integrated with robotic events. These happen at a very high frequency. Therefore the design choices have to be leveraged on some BC platform that supports fast transactions. For this reason, we decided to use EOSIO Blockchain-based solution. With its DPoS consensus algorithm, it is possible to achieve fast transaction throughput [43]. To the best of our knowledge, we are the first to introduce a privacy-by-design DAP to audit robotic events, including privacy-sensitive information, by using EOSIO Blockchain technology.

The next chapter will illustrate in detail the design of the proposed DAP.

# 4

# Architecture Design

## Contents


4.1 Objectives of the Proposal . . . . . 33
4.2 Architecture Overview . . . . . 33
4.3 Design Guidelines . . . . . 35
4.4 Main Algorithms . . . . . 36
4.5 Identity Management . . . . . 39
4.6 Discussion . . . . . 41

The present chapter corresponds to the **design and development** phase of DSRM, in which are presented the DSRM artifacts. The artifacts aim at solving the identified problem (Chapter 1).

The chapter is structured as follows. First, we provide the objectives of the solution inferred from the related work. Later, we present an architecture overview, starting with the high-level design that depicts how all the software components work together, illustrating with functional diagrams. Then we explain the main design guidelines to apply for the development of privacy-by-design architecture. Further, follow the presentation of the main algorithms involved in the private data classification and Identity Management for private data access. We finalize the chapter with the discussion.

## 4.1 Objectives of the Proposal

This section identifies the objectives of this thesis proposal inferred from the state of the art of the problem identified in Chapter 1, which is the **lack of control and transparency in privacy in HRI**. We exposed the problems observed currently in interactions between humans and robots regarding individuals' privacy. We set as an objective the provision of a solution on this matter by using blockchain technology.

In this line of reasoning, the present proposal aims to:

- Design and model in detail Blockchain-based DAP for privacy in HRI. This model will include the description of each DAP's layer with functional diagrams to illustrate the mental model.
- Describe the main algorithms used to classify the owner of private data and the correct private data outline.
- Describe how users' identity management is handled by RFID pseudo-identities, Blockchain, and Crypto-Wallets for the seamless data access without breaching security or privacy of individuals.

## 4.2 Architecture Overview

This section describes the design architecture of the **BlockRobot** for privacy-by-design DAPs. BlockRobot harmonizes the mechanism to identify robotic events with private data and classify the data subjects (Fig. 4.1).

We depict three significant layers of the system:

1. **HRI layer**: presents the interaction between the social robot and an individual. Here, the individual is identified with an RFID tag – hardware (for instance, embedded into a smart card). The robot is equipped with a depth camera – a tool to retrieve streams of data, and an RFID sensor – equipment that allows the robot to detect RFID tags. Both the RFID sensor and RFID tag enable

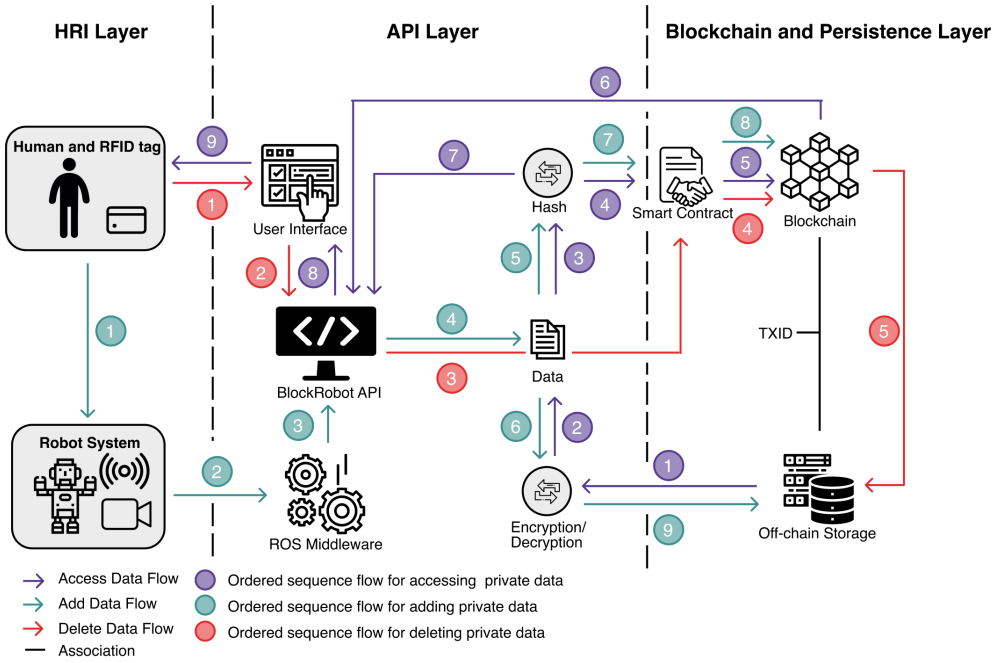


**Figure 4.1:** High-level System Architecture of BlockRobot

identifying individuals by the robot unambiguously and preserving the anonymity [8]. Once an individual is detected and his image is being recorded, the information flow continues in the API layer.

2. **API layer**: After the primary interaction between a human and robot, both streams of images and the RFID tags are published as separate Robot Operating System (ROS) events. Furthermore, the BlockRobot API can access that data API layer through the ROS Middleware – software that allows us to read robotic events in the *publish-subscribe pattern*. Therefore, the publishing of events on ROS topics happens seamlessly, and no additional overhead is delegated to the robot. The BlockRobot API is a trusted server that handles data transformations of ROS events into private data and operates the off-chain storage by synchronizing the on-chain transactions with off-chain database updates.

3. **Blockchain and Persistence layer**: Once the BlockRobot API main algorithms retrieve private data from ROS events, the data can be persisted in Blockchain and Persistence Layer, which represents the storage of data – through BC and off-chain repository, and the access of data – managed by the smart contracts. To modify BC's state, individuals and robots interact with BC as peers by issuing the transactions. Moreover, with smart contracts, it is possible to establish access control management on a transaction level.

## 4.3 Design Guidelines

The fundamental goal of the system is to provide a mechanism for seamless private data access in HRI. We believe that to achieve that, the following design guidelines should be applied:

### 4.3.1 Correct data outline

- The Robot should communicate all the robotic events relevant to the correct private data outline to BlockRobot API. Such robotic events should include events with RFID tags and RGB-D camera recordings.
- The individuals' identities should be anonymized to ensure data confidentiality and security.

### 4.3.2 Data Persistence

- The private data needs to be tamper-resistant. Therefore, it should be hashed, and hash should be stored on BC for verifiability.
- The original private data should be persisted in the external data repository, such as a cloud storage system (e.g., AWS or Azure), Database Management System (DBMS) (e.g., PostgreSQL, MongoDB), or any other data storage repositories, or a distributed storage system (e.g., IPFS, [71]). The external repository, also called by off-chain repository, should be mutable, and it is essential in order to be able to delete private data [14, p. 32].
- The private data stored off-chain should be encrypted to avoid direct access by third parties. The encryption should be done with symmetric or asymmetric encryption keys schemes, though in some cases, symmetric encryption is a better option due to performance reasons. Naturally, the encryption keys should be the ones owned by the users to whom private data belongs. The possible algorithms for encryption should be considered, such as Elliptic Curve Cryptosystem (ECC) or RSA, [72]. The encrypted data will become confidential because the encryption algorithm makes data unreadable until it is decrypted.
- The private data (stored off-chain) and its hash (stored on-chain) should be relatable. Both the block with the hash and the data structure where the private data is kept should contain a Transaction ID (TXID). Further, the BlockRobot system lookup the hash by TXID while retrieving private data from the off-chain repository.
- The data provenance should be specified by design. Robots should provide their digital signature to mark the data records. The hash and robot's digital signature should be placed on-chain together and provide the data provenance.

- The off-chain and on-chain should be synchronized. Once a new block is validated in the BC, the off-chain database should be triggered to align the change. Accordingly, if a participant or a robot updates the BC state, then the off-chain state will update as well.
- Robots and participants should act as peers in Blockchain to provide the provenance and non-repudiation of private data.

### 4.3.3 User Interface and Blockchain Transactions

- The user should visualize data records on the intuitive user interface – privacy dashboard – in a human-readable format. Such an interface should present the user with all the data where he/she was identified in HRI. The human-readable format may depend on the nature of the data, per ex. an image, video, audio, or plain text.
- It should be possible for the users to request the access or erasure of data by User Interface (UI). Either access or erasure of data should create the transaction on the BC for audit purposes.
- The user should provide a digital signature to prove their identity and sign the transaction on each access and request of erasure. The UI should be integrated with secure crypto wallets for the digital signatures management.
- Users should visualize the history of data (summary of data processing actions, p. ex. access, erasure of data). Blockchain is immutable, and for that reason, it should be possible to retrieve the previously audited data accesses from it. Therefore storing all data processing actions on BC should provide data accountability.

## 4.4 Main Algorithms

The present section describes algorithms that attribute the private data to the individuals to whom the data belongs. There are two main algorithms: *Data Subject Classification* and *Time-Pairing*.

### 4.4.1 Data Subject Classification Algorithm

During the interactions between a robot and a human, the robot may record large streams of images. To illustrate, we present the interaction design in Figure 4.2, where the robot records RGB-D images while a person walks by the robot. A person is equipped with a smart card with an embedded RFID tag. As the interaction goes, and streams of RGB-D images are published on the RGB-D Topic, the RFID laser embedded in the robot detects an RFID tag from the person's smart card. It is worth to note that the

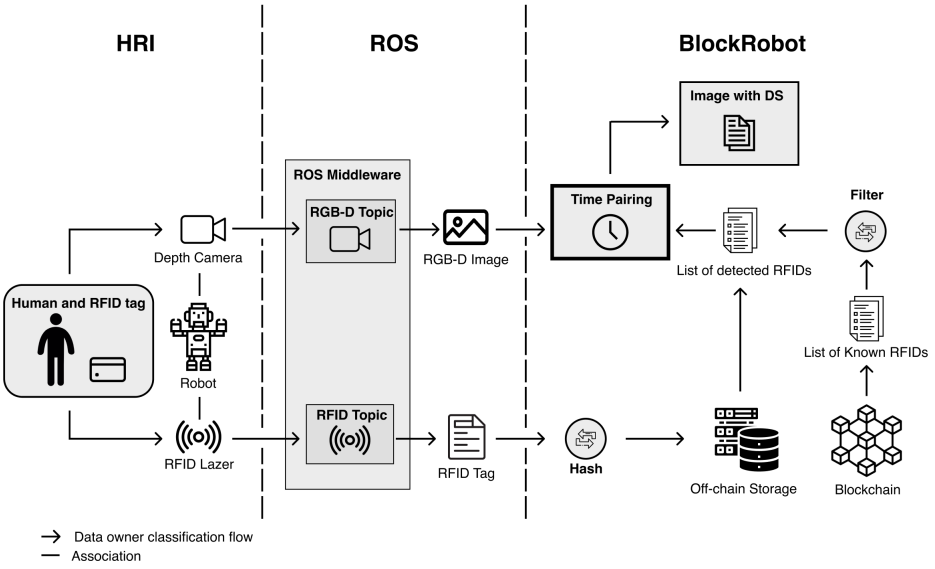


**Figure 4.2:** ROS middleware and BlockRobot interactions - Classification of data owner model

RFID tag detection and RGB-D image are asynchronous events because the RFID laser and the kinetic camera are separate devices. Therefore, the data from both events require additional coordination.

RFID tags provide information regarding the proxemics and time relating to the RFID laser. Previous research [8] points out that RFID tags are subject to background noises, depending on the environmental conditions, which may reduce the precision of the time and distance of an individual carrying tags. Still, they provide knowledge that some individuals are in the robot's vicinity, which is valuable for identification purposes. The RFID signal is composed of the Unique Identification (UID) – unique identifier, and the timestamp when the RFID signal was detected. Furthermore, the UID gets anonymized by hashing it, with the intuit to difficult the individual's possible re-identification. In real-world scenarios, many people may walk by the robot. Therefore, many RFID signals will be identified simultâneously. For that reason, all the anonymized UIDs and timestamp retrieved from each RFID tag are temporarily stored in the off-chain repository. The storage of the UIDs will allow us to consult the information about all the individuals detected in HRI, and consequently, who is present in the RGB-D image recorded at a specific time.

On the other hand, to identify an individual in the network, we persist his/her anonymized UID on the BC (Further explained). Besides being immutable, placing RFID tags on BC provides an indirect relation between known participants and their RFID tags. Finally, the algorithm has enough data to know:

- who are the known members of the network – UIDs stored on-chain,
- who are the individuals detected in a specific time interval – UIDs and timestamps temporarily stored off-chain.

Then two lists of UIDs are reduced into a list of known members detected in a given interval. For

each RGB-D image, this list will be used to understand whether an individual in a list is present in the RGB-D image. If yes, then we can create a unique data structure containing the RGB-D image and individuals identified in the image. The details are discussed in the *Time-Pairing Algorithm*.

### 4.4.2 Time-Pairing Algorithm

After supplying the list of RFID tags and an RGB-D image frame, with respective timestamps, it is possible to calculate the image's data subject by the process that we denominate as time-pairing. This process, in essence, consists of that *if some RFID tag is detected during the image recording, then the carrier of an electronic card with that RFID tag is the data subject (owner) of the recorded data.*

The Time-pairing algorithm consists of examining each RGB-D image and the list of RFID tags, and give as output whether that RGB-D image is a private data or not. If it is a private data, meaning there are individuals identified in the image, it also says who the individuals are. Each RFID detected is a single event in the time-pairing algorithm. Therefore if there are many participants near the robot, many RFID events will be published. One RGB-D image is also an event. Different inputs, such as a list of RFIDs, one RFID, or zero RFIDs collected from events during the HRI, will trigger different outcome scenarios. Furthermore, for different RFIDs detected, based on the timestamps of detection, we estimate how long the participant identified shall be considered to remain close to the robot. $\Delta$ **denominates this estimate**.

We define $\Delta$ as a certain interval for which the person carrying a tag should be considered in the robot vicinity. For simplicity, here and after, we will assume that when RFID tag is detected, then the individual is detected. For each individual detected by the robot, we take this event's timestamp and enclose it with $\Delta$, which will give us the interval of time when the person remains near the robot. *Two* possible outcomes may happen:

- **The out of bound time-pairing** (Fig. 4.3) – the estimated interval of the person's presence near the robot does not intersect with the time when the image is recorded. In other words, during the time of the recording of the image frame, there are no DS in the robot's vicinity, so the data is not private. In this case, we do not output the data object, as it has no interest in BlockRobot.

- **The inbound time-pairing** (Fig. 4.4) – the estimated time $\Delta$ when the robot sees the participant intersects with the timestamp of the RGB-D image recording. In such a case, the data object will be created with the RGB-D image and an RFID reference. This data object represents a private data with the data subject identified by the reference, and private data is the RGB-D image. The creation of the data object is the output and finalization of the iteration. For both use cases, if there are more RFIDs on the list, than the process will repeat for the rest of RFID signals.

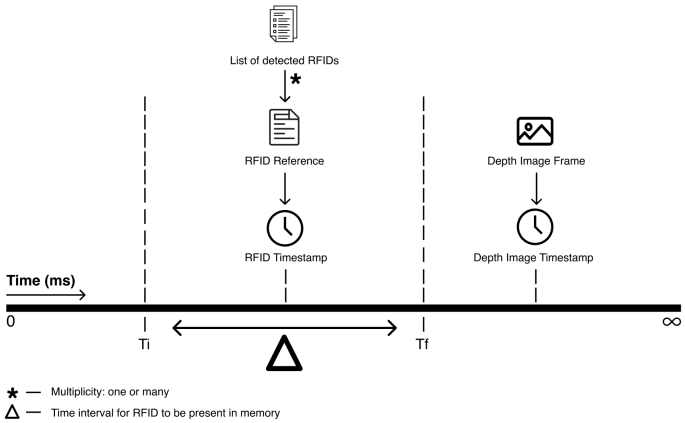


**Figure 4.3:** Time Pairing model: Image detected out of bounds of RFID detection

If more that one person is identified during HRI, meaning that some RFID intervals overlap, then more than one participant is in the image. In such a case, the picture is the private data belonging to all these persons, and anonymization algorithms can be applied so that individuals cannot recognize other people in the image. Further, a person can request the deletion of the link between him and the image. Given the sufficient anonymity level, if the individual is no longer recognizable on the image, that image is no longer considered that participant's private data. Once the last participant requests deletion, then the image can be deleted.

## 4.5 Identity Management

Following the logic of the algorithm of time pairing, each image has a corresponding paired **UID**, which identifies the DS present in the image. Furthermore, *the private data that belongs to the user is only the*

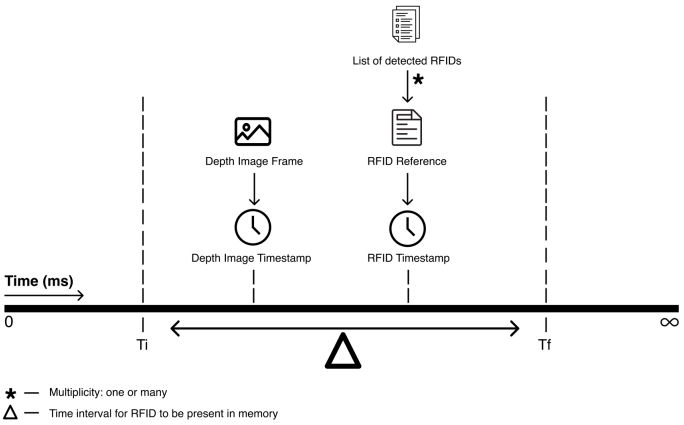


**Figure 4.4:** Time Pairing model: Image detected in bounds of RFID detection

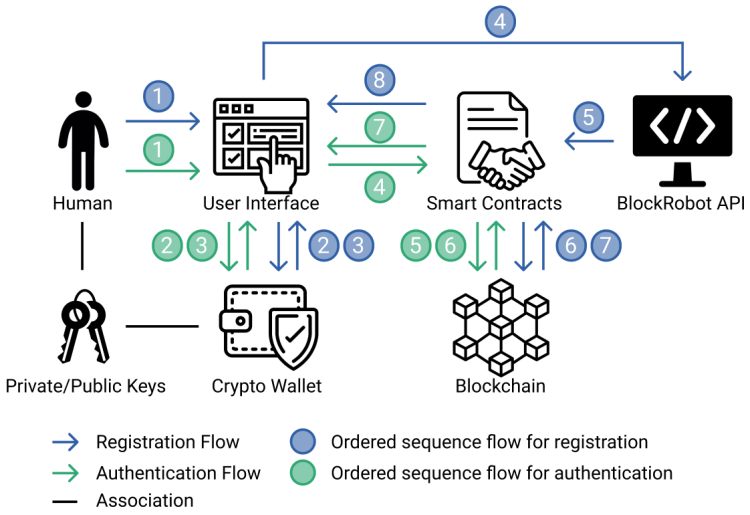


**Figure 4.5:** Identity Management Model – Registration and Authentication

*images with a paired UID by which the individual is identified*.

Identity management comprises two phases – proof of ownership of the UID (retrieved from RFID tag) and proof of identity by a digital signature. Once proven the identity, a user may access all the images with his/her associated UID. We illustrate the *Registration* and the *Authentication* on Fig. 4.5.

- **Registration:** to interact with a private permissioned BC user must belong to the network. In our architecture, the proof of membership to the BC is managed by the smart contracts. As a first step, a user should provide his UID and a public key through a user interface and issue a new transaction to the BC with his UID (hashed) and public key as a payload. Further, once the transaction is initiated, the smart contract will require the user to sign the transaction with his/her private key. Once signed, the smart contract performs a public key recovery to prove that the digital signature belongs to the user's public key. Given the cryptographic relation between private and public keys, and the digital signature inferred from the private key, it is possible to recover a public key from the digitally signed transaction. Once the public key is confirmed to belong to the user, and the transaction is valid, the UID with the user's public key will be installed on the Blockchain, and the user can begin the process of authentication for future access the data under that same UID.

- **Authentication:** To access private data, the user needs to authenticate to the network by providing his previously registered UID. Once provided, the UID is hashed, and the smart contract search for the UID on the BC. If any hash of UID installed on BC matches the UID that an individual claims to own, then the smart contract performs a public key recovery algorithm to recover a public key from the digitally signed transaction. Once the public key matches the public key installed on the

BC under the provided UID, the user is the UID owner. Therefore, the proof of UID ownership is successful, and the user can access his private data.

## 4.6 Discussion

The proposed design includes the guidelines that should provide the individuals' protection and control for private data. Robots should be able to identify many humans at the same time, flawlessly and efficiently. By using RFID technology, it is possible to attribute the individuals with some RFID tags so that robots can identify humans unambiguously. Such is possible by providing participants with some smart cards with embedded RFID tags.

After the correct identification, the system encrypts the data to avoid direct access to the information. Only the person in possession of the key can access who is naturally the data owner. Furthermore, The hash of private data will be stored on the BC for verifiability, thus BC is a trust anchor. Using BC as a trust anchor has the advantages of improving the scalability because only hashes represent a minimal amount of data compared to the storage of the actual images (each image is approximately 2 MB). Moreover, the private data stored off-chain can be deleted and comply with GDPR' right to be forgotten. Finally, the proposed architecture improves transparency in privacy during HRI, since by design, each action is recorded on BC, creating a history of data processing since the first interaction.

Since we are dealing with the real stakeholders – end-users without technological background, the intuitive UI should further be developed to present the private data to the end-user.

Regarding the algorithms involved, both Data Subject Classification and Time-Pairing work together to correctly outline the private data. The first algorithm extracts the list of detected (and known) UIDs that refer to the system's members. The second algorithm takes the timestamps and verifies whether detection time is close to the image recording time by examining the timestamps. Therefore, the algorithm will disregard some of the detected identifiers, keep the correct ones attached to the image, and assign individuals.

Finally, from an identity management perspective, data access is managed by proof of ownership of RFID tag and proof of identity by digital signature. The users can prove their identity by providing the correct crypto keys and the digital signature that crypto-wallets manage securely. Since the users have RFID, they can prove the ownership, and the data that belongs to each individual is the data resulting in assignment calculations from the main algorithms. The proposed design of the Blockchain-based architecture provides the complete flow of the private data without the disclosure of the real identity of the user.

In the next chapter, we will demonstrate the utility of the proposed architecture in a real-world experience.

# 5

# Demonstration

## Contents


5.1 Architecture of the Prototype High-level View . . . . . 45

5.2 Sequence Views . . . . . 51

5.3 Demonstration Use Case . . . . . 56

5.4 Results . . . . . 59

The present chapter corresponds to the **Demonstration** and fifth phase of the DSRM that we adopted in this thesis. It's main objective is to demonstrate the use of the artifact to solve one or more instances of the problem. As a proof of concept, we establish an implementation with a SR. The experiments are performed in hospital corridors and rooms. The execution data logs are computed with process mining techniques [26] to analyze the performance and bottlenecks existing in the case study.

Results obtained in a real scenario pointed to the use of EOSIO Blockchain-based solution. To the best of our knowledge, we are the first to introduce a privacy-by-design DAP to audit robotic events, which include privacy-sensitive information, by using EOSIO Blockchain technology.

The chapter contains four sections. First, we describe the technical implementation of EOSIO-based Blockchain DAP based on the design guidelines from Chapter 4. Then we present sequential diagrams to demonstrate the information flows. After that, we will cover a series of experiences as proof of concept and the applicability of solutions to real-world problems. Finally, we present results of the experiences.

## 5.1 Architecture of the Prototype High-level View

This section provides an overview of an architecture of the EOSIO-based permissioned Blockchain DAP [73]. In this DAP, personal data identification and off-chain storage mechanisms combined with database querying tools offer users access to their private data cleanly and transparently through the user interface.

We developed the Mongo Express Angular Node (MEAN) stack Javascript (JS) Application [74] – ExpressJS for Backend API service layer, Angular 7 Frontend UI, and MongoDB - the Non Structured Query Language (NoSQL) database on top of the NodeJS runtime environment. MEAN stack integrates with EOSIO Blockchain that maintains BC shared ledger and serves smart contracts logic for the business layer (Fig. 5.1).

### 5.1.1 MEAN Stack

Architectural components listed above, such as MongoDB, Express, Angular, and NodeJS, are usually referred to as a Collection of MEAN Stack. MEAN is an acronym for MongoDB, ExpressJS, Angular, and NodeJS. From client to server to database is a full-stack JS.

- **Angular Frontend** serves three essential services for the data management process: *Authentication Service*, *Read Data Service*, and *Delete Data Service*.
  - Authentication Service – serves as a starting point for the application since any operation performed requires the previous authentication.
  - Read Data Service – serves the list of all personal data related to the user.

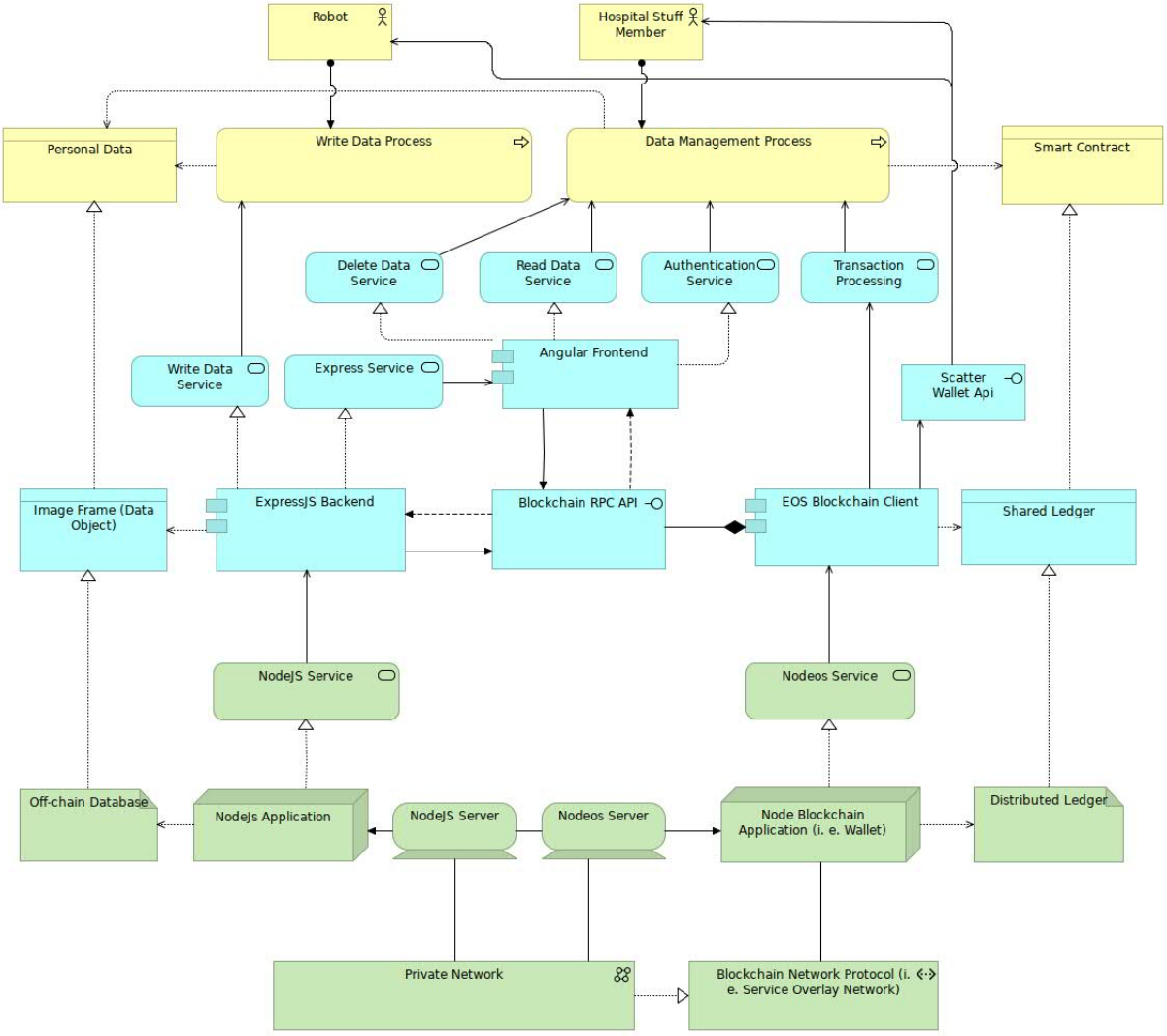


**Figure 5.1:** Application Layered View

  - Delete Data service – gives the user the possibility to delete data from the database.

- **ExpressJS Backend** realizes two primary services: *Write Data Service* and *Express Service*.
  - Write Data Service – serve the Write Data Process, which is performed by robots collecting personal data. It mainly offers services for the robots to write data to the off-chain database.
  - Express Service – is a middleware API to query data on behalf of the frontend component. It provides all the services realized on Frontend.

- **MongoDB** – is an open-source document-based NoSQL-Database and, in our DAP, is an off-chain database storage for private data. MongoDB integrated with ExpreesJS will enable users to query their data securely.

- **NodeJS** – is a runtime environment that includes everything that is needed to execute javascript programs. It is an application server that hosts ExpressJS backend.

The prototype foundation on the MEAN stack rises from the requirement of integration to both Blockchain and robotics. For the Frontend, Angular was a comprehensive solution in order to integrate with EOSIO Blockchain client and with Scatter Wallet. Furthermore, Express demonstrated to be an outstanding middleware for the connection with ROS middleware and with MongoDB because of its simplicity in deployment and integration.

### 5.1.2 Blockchain, Smart Contracts and Crypto Wallet

- **Local EOSIO Blockchain** – is composed of smart contracts and a shared ledger. EOSIO BC integrates with the MEAN stack, maintains Blockchain shared ledger, and serves smart contracts logic for the business layer. It runs on *NodeOS*.

- **NodeOS** – is a lightweight operating system using NodeJS as userspace. NodeOS is an EOSIO Blockchain middleware.

- **Crypto Wallet** – Each Blockchain has its signature provider and provides an easy way to set up application. We use **Scatter** as a signature provider. Scatter is a wallet used to sign transactions and manage user's keys. Each user who is a member of a network must have a Scatter Wallet in order to access his data.

### 5.1.3 Angular Frontend Logical View

Angular Frontend serves the user interface to access private data (Fig. 5.2), thought *Authentication* and *Data Access* components, *eosjs-ecc* and *scatter-js* libraries.

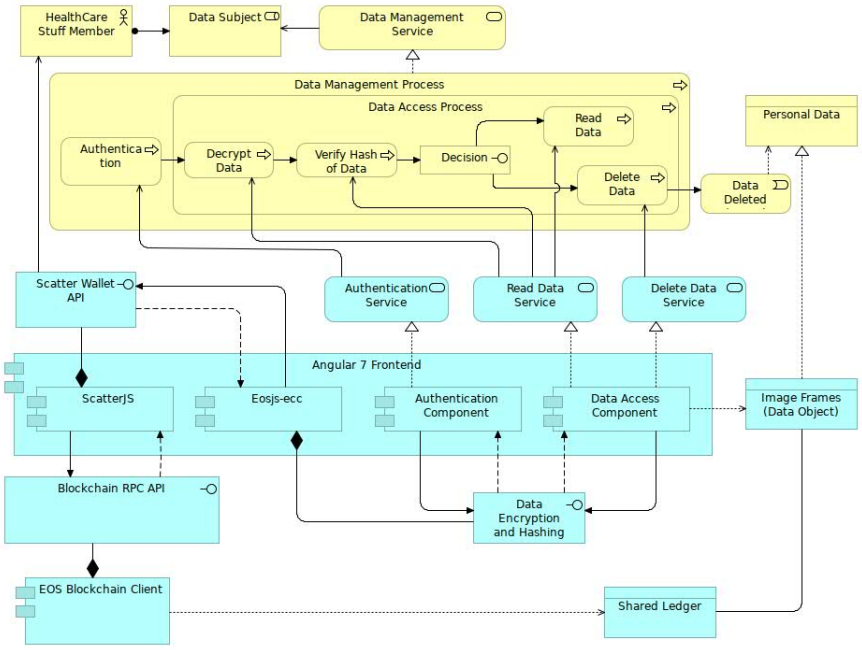


**Figure 5.2:** Angular Frontend View

- **Authentication component** – To access private data, users need to prove their identity through authentication service. Users need to supply their UID through the form. Once submitted:
  - **Scatter-js** – on behalf of Scatter, the user provides his digital signature, and crypto-wallet verifies his identity by the digital signature recovery algorithm.
  - **Eosjs-ecc** – ECC primitives supply us the Sha-256 hash function that will hash the UID and verify if the user is a member of the network. Further, if the hashed UID matches with one stored on the Blockchain – then he is a member of the network.
- **Data Access Component** – provides users with control over their data. The data access process consists of:
  - **Eosjs-ecc** – provides the decryption function - for user to decrypt his private data issued from Backend (BE), and hashing function to verify data integrity by comparing the hash of data against the hash stored on Blockchain.
  - **Scatter-js** – securely provides a user's digital signature to realize a transaction on a Blockchain.

### 5.1.4 ExpressJS Backend Logical View

The Backend server's primary accountability is to write data and classify its access control. The Backend communicates with the robot through ROS Topics. At the same time, robots record some private data,

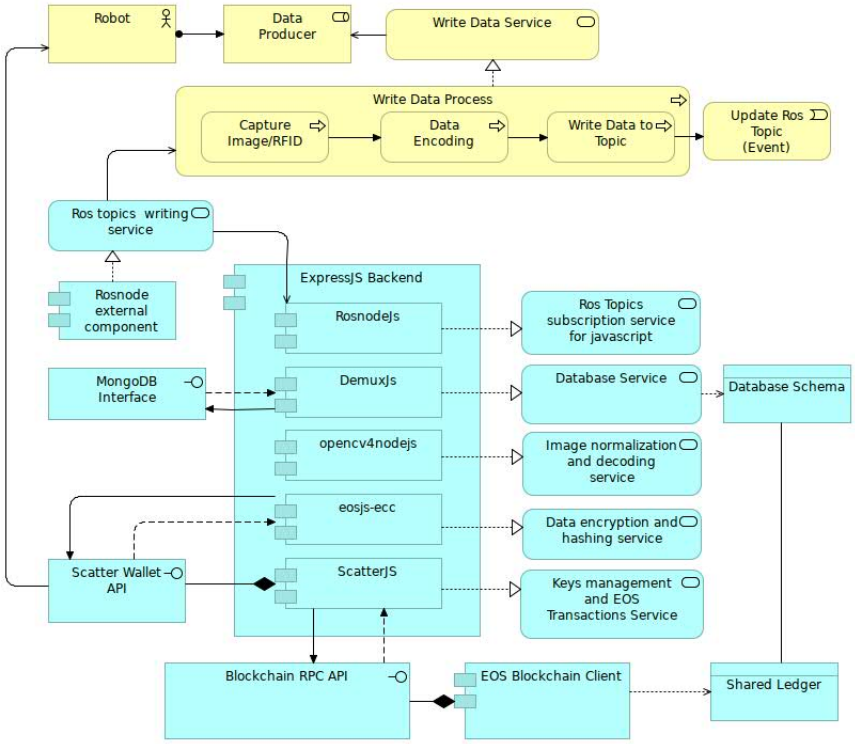


**Figure 5.3:** ExpressJS Components View

which flow passes to the ROS node external component, which has a list of ROS topics. After publishing the data on the topic, the Backend can reach it. ExpressJS Backend uses five libraries (Fig. 5.3) to accomplish that task, each one with its employment:

- **RosnodeJS** – is a javascript library for subscription to ROS topics. Given some known ROS node address with ROS topic name, this library lets us to subscribe to the data that robots record and publish to the ROS topic.

- **Opencv4nodejs** – allows us to use the native OpenCV library in NodeJS. It's a C++ library that offers image processing features. Its main utilization is image frames decoding and normalization.

- **Eosjs-ecc** – ECC functions. This library provides data encryption and data hashing service.

- **Scatter-js** – is a javascript library to interact with a Scatter wallet interface. It grants secure keys management and realizes EOSIO Blockchain transactions service through EOSIO Blockchain Remote Procedure Call (RPC) API.

- **Demux-js** – is a BE infrastructure pattern for sourcing BC events. It synchronizes BC on-chain with database off-chain data.

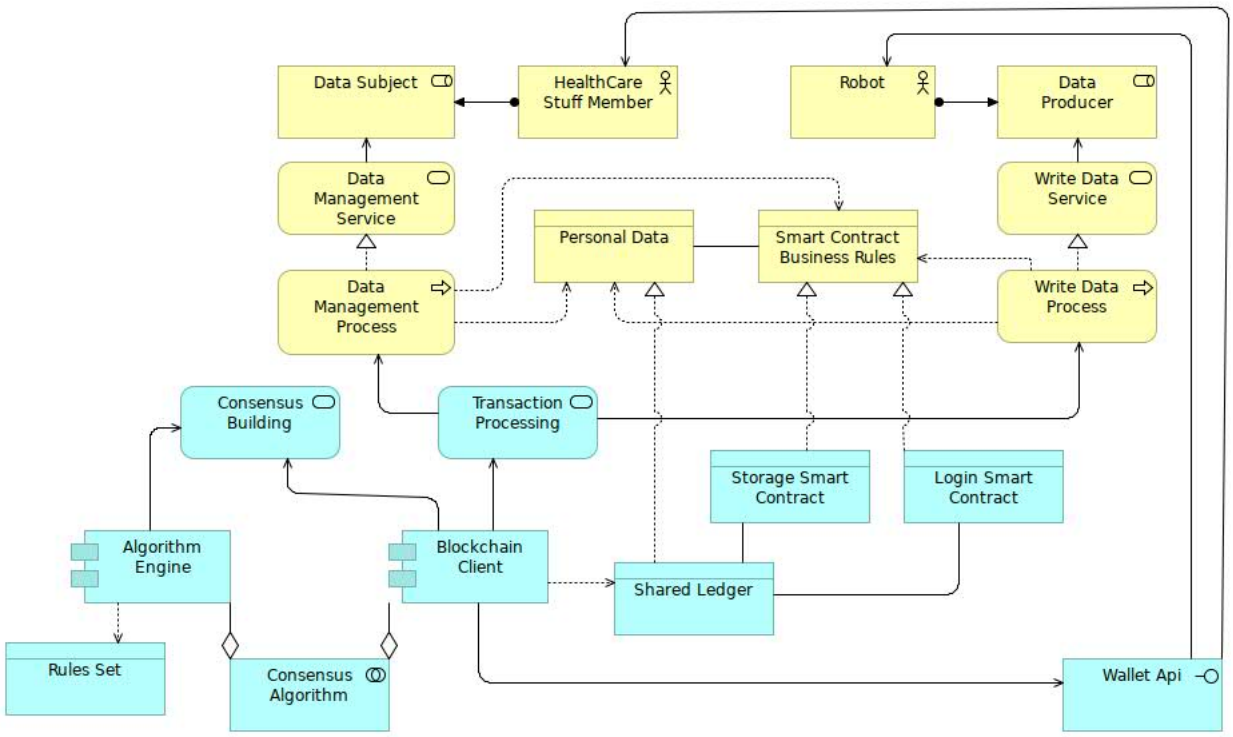


**Figure 5.4:** Blockchain Layered View

### 5.1.5 EOSIO Blockchain Client

EOSIO Blockchain client exposes transaction processing service for private data processing activities (Fig. 5.4). It is based on a consensus-building service that utilizes a consensus algorithm and rules set. Users can access BC through wallet API, as long as they have private keys. Our EOSIO Blockchain includes two smart contracts: Storage Smart Contract and Login Smart Contract.

- **Login Smart Contract** – handles the authentication of the users. It has one multi-index table with two columns: *username* and *UID*, both previously hashed. The authentication in this smart contract is accomplished by using two actions: *Upsert* and *Prove*.
  - Upsert – users register in the network by performing upsert action with a username, UID, and signature as a payload. As a result, a new transaction is created, representing the event of registration.
  - Prove – is a ”fake” contract action. It retrieves the user’s signature from BC, without performing an actual transaction (that is why we call it fake). Therefore this action does not cost the user any resource and proves his identity by recovering the public key from users’ signature.
- **Storage Smart Contract** – handles the metadata relating to personal data records. This smart contract has one multi-index table with two columns: *Hash* and *TXID*. Whenever some private

data is inserted by writing the data or is modified by accessing it, the BC transaction is triggered. The Storage smart contract consists of two actions: *Insert* and *Erase*.

- Insert – creates a new transaction that represents new data addition; thus, BC adds a unique hash of private data to its shared ledger.
- Erase – creates a new transaction representing the event of deletion of the data; thus, a new block is added to the BC with the information about the deletion of the corresponding off-chain data.

## 5.2 Sequence Views

The current section will illustrate the execution of the significant processes' atomic operations sequentially along with the corresponding sequence diagrams. Each of these processes is executed based on the integrated work of the previously presented components.

### 5.2.1 Registration Process

The Registration Sequence diagram is illustrated in Figure 5.5. To register, the user needs to introduce his username and his medical card UID (corresponding to his UID). The UID is then hashed with the eosjs-ecc sha256 algorithm, and the new transaction is realized with the hashed UID and username as payload to Login Smart Contract with upsert action. ScatterJS signs the transaction. On the BC side, BC verifies if the current username is available and whether UID has not already been taken. If validation is successful, then the new row is inserted in the Login multi-index table, which means a user's successful registration.

### 5.2.2 Authentication Process

The Authentication Sequence diagram is illustrated on Figure 5.6. After being registered in a network, the user should authenticate himself to prove his identity and access some data. In order to do that, he has to introduce his username and initiate the process of authentication. All usernames belonging to the members of the network are hashed and stored in Login Smart Contract in a multi-index table. The next step is to retrieve all usernames publicly available in Login Smart Contract. After retrieving that list, the next step is to loop all the usernames and wait for a match. If there is no match, then authentication is failed, and a user does not belong to the network. Else, the current username belongs to the network, and the next step is to verify if users have a signature that proves his identity by the EOSIO signature verification process.

The process of verifying signature goes through the following steps:

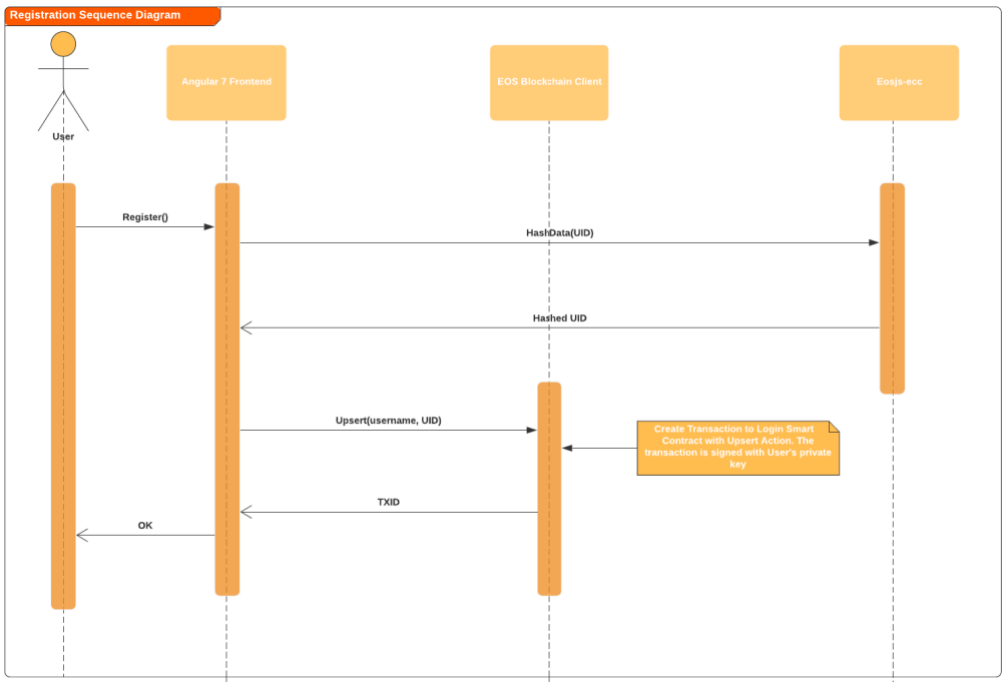


**Figure 5.5:** Registration Process Sequence Diagram

- The user performs a ”fake” transaction to the Blockchain to the login smart contract with prove action. While doing it, ScatterJS sign transaction. As the transaction is fake - it never happens to be broadcast to the Blockchain. Instead, the user's signature is recovered based on his private key.

- The last step is to recover the user's public key with the eosjs-ecc library. If the user indeed is who he claims to be, that the recovered public key is going to be the same as the user's (given the signature properties).

- If the previous statement is true that the authentication is successful; otherwise, authentication failed.

### 5.2.3 Write Data Process

The Write Data Process is illustrated by sequence diagram on Figure 5.7. When new data arrives, the ROS topic event is triggered. As stated before, RosnodeJS is a javascript library that allows us to subscribe to ROS topics. So each time a new Image Frame is recorded, the notification of that event will be received on RosnodeJS subscription to the topic. The Image frame received needs to be stored on the database, so we need a way to encrypt it with an encryption key. The Image Frame represents private data that requires encryption for security purposes.

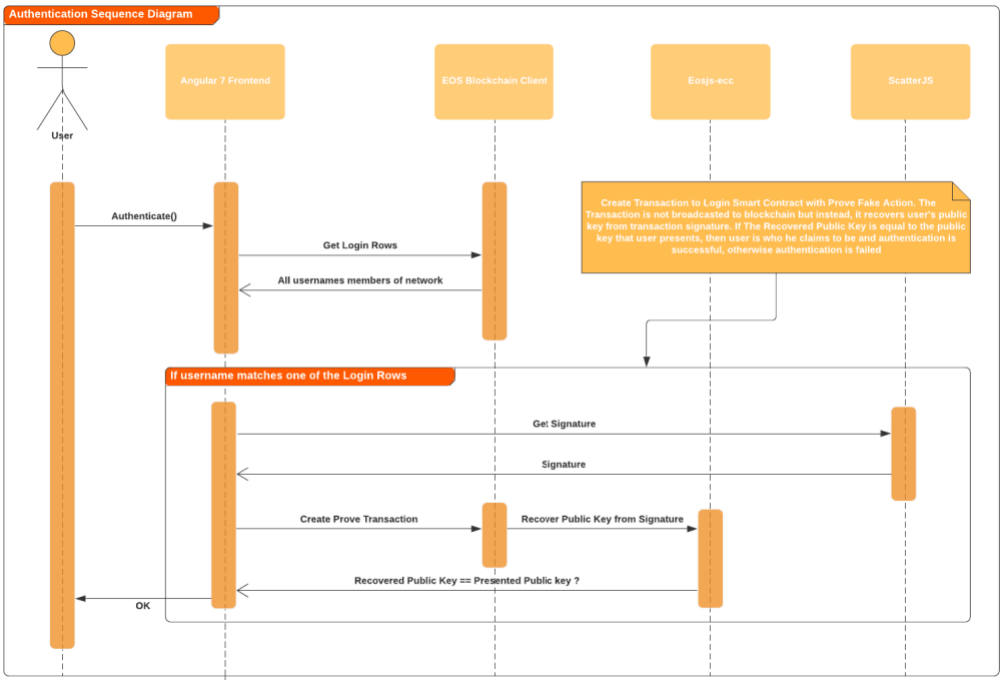


**Figure 5.6:** Authentication Sequence Diagram

In the first place, we need to decide whose private data is that. As we know, each user member of the network is identified by its UID, also known as the RFID tag attached to the user. In order to know whether there is any RFID detected in the vicinity, we consult RFID listener and check if there are any currently detected RFIDs. If not, then there are no humans in the vicinity, and that data is not private, and consequently, there is no need to protect it, and the process ended.

If the RFID's List is not empty, then there is human in the vicinity, and so we continue the process. The next step is to obtain the user's public key. To retrieve users' public key, we retrieve an account associated with the UID (Stored on Login Smart Contract). Once we get an account, we can lookup for its public key on the BC. Now with the private/public key pair, ScatterJS can generate an encryption key. This key is generated from the producer's private key and the user's public key.

The next step is to hash the data with eosjs-ecc library sha256 function, which receives data as a parameter and returns its hash. After having the hash of data, the data can be encrypted with the previously obtained encryption key. We are going to use symmetric encryption with the encryption key. Now only the entity who possesses the encryption key can access the private data (In this case, only user whose public key was used for the encryption).

Once the data is encrypted, and its hash is computed, the next step is to create a transaction to the Blockchain and store the hash of data on the multi-index table of a Storage Smart Contract. Given the immutability properties of Blockchain, the data can be verified against the hash on the Blockchain in the next process (The Access Data Process). The transaction is signed with producer node's private key on

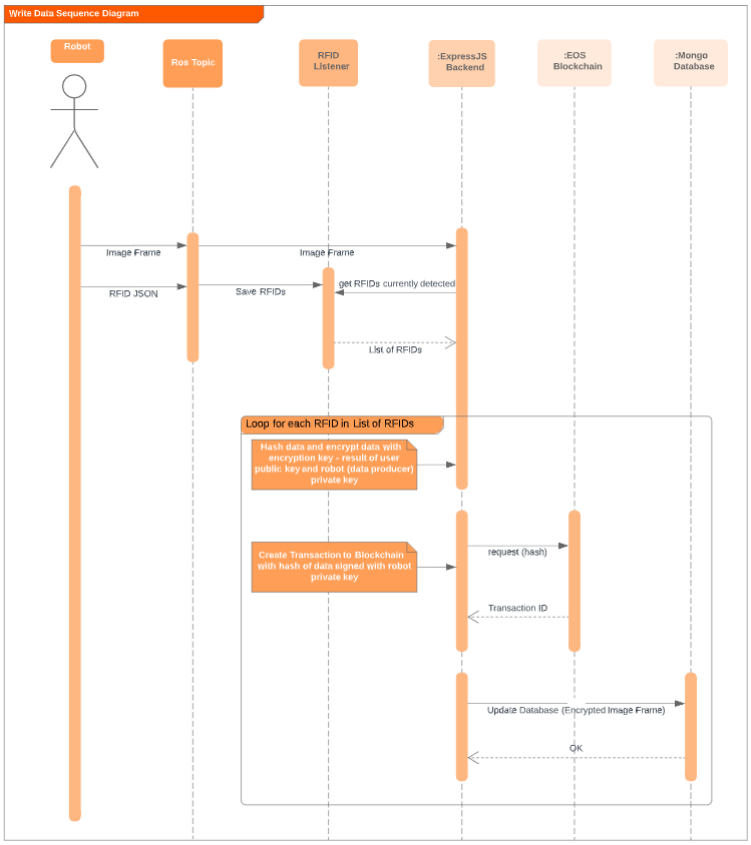


**Figure 5.7:** Write Data Process Sequence Diagram

ScatterJS, which access scatter wallet API. Once the transaction is signed, ScatterJS forwards it to the BC, which processes the transaction. After a successful transaction, a BC return TXID.

Given TXID and encrypted data, we can finally store it on a database with the DemuxJS component's interface. This sequence flow shall repeat itself for each RFID in the RFIDs List returned by RFID Listener.

### 5.2.4 Access Data Process

The Access Data Sequence diagram is illustrated on Figure 5.8. Whenever a previously authenticated user requests access to his data, our DAP queries to the BC all the records that reference the user's data via RPC API. Each one of these records contains a hash of private data and with time reference when that data was captured. Here user decides what private data record he wants to access. By choosing a record, the user requests Image Frames to ExpressJS, which queries MongoDB for the data. The data returned from the database is a list of encrypted Image Frames. Once data is retrieved, on the BE, ExpressJS will now have to decrypt all the encrypted frames. So the next step in the process is to get the producer's account public key and compute the encryption key with the ScatterJS. The encryption key

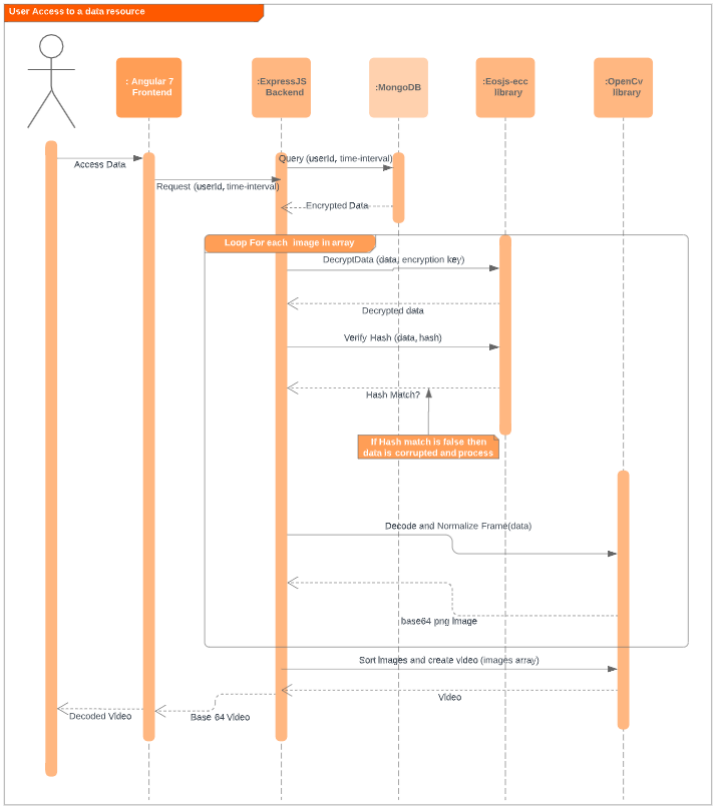


**Figure 5.8:** Access Data Sequence Diagram

is computed with the producer node's public key and the user's private key and is the same encryption key that was computed when the data was encrypted. After obtaining the encryption key, we shall start to decrypt each frame. After Successful decryption, the data can be verified on its integrity. To do that, we compare the hash of data with the hash stored on BC. This verification shall execute for each Image Frame in the sequence.

If the integrity verification is successful, the next step is to decode image with Opencv4nodejs c++ library introduced before. Each frame has its encoding, which is handled by this library. After decoding the frame, it follows the process of normalization of the image. In the end, we have a list of normalized frames. The last step is to order the frames by frame step number, and once finalized, we can create a video by representing each frame after another with a fixed delay. When finished, the data is ready to be displayed on the Data Access Component on the frontend.

### 5.2.5 Delete Data Process

The Delete Data Sequence diagram is illustrated on Figure 5.9. The user should be able to delete the data, according to GDPR. Once he decides to delete, the authentication component verifies the user's EOSIO signature, like described in the authentication process. If the signature verification is positive,

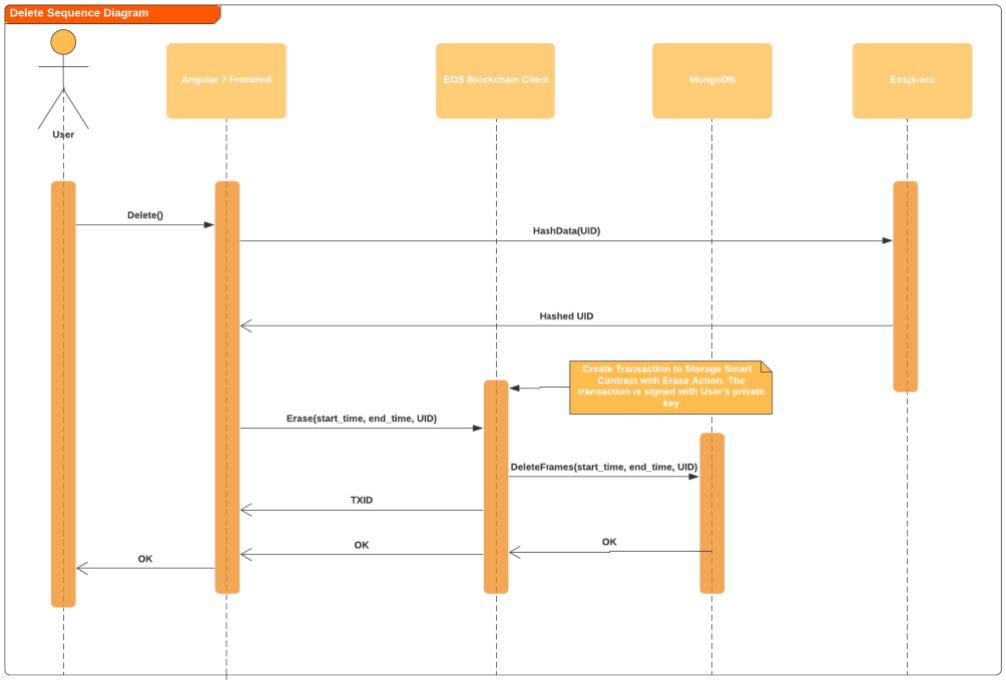


**Figure 5.9:** Delete Process Sequence Diagram

user proceed and create a transaction which is signed by ScatterJS. The rows which correspond to the personal data record are deleted. Consequently, DemuxJS updates the database state by deleting all the frames referenced by just deleted rows in Blockchain.

## 5.3 Demonstration Use Case

Robots need robust means of human recognition. However, with multi-person interaction it is challenging to develop robots that work in daily life, such as inside homes and offices, using only visual and auditory sensors [75]. Concerning audition, many people may be talking at the same time. For vision, lighting conditions are unpredictable, and the shapes and colors of objects in a real scene are not simple. When dealing with data privacy, a different accurate human identification system is required. Mistakes in identification may grant access to personal data by a third party and consequently violate data privacy rights.

Robots operating in public places may have to distinguish between hundreds of humans at once and simultaneously identify the ones nearby. To tackle the human recognition problem, we use wireless sensors. With the use of wireless sensors, robots will be able to identify the people that interact with them in a specific environmental context. Consequently, for the sake of our research, people should wear a wireless ID tags embedded in nameplates in order to be correctly recognized by social robots that are enabled with a RFID reader. Several usage protocols can be applied, for example upon check-in

at the building, each person (staff member or patient) receives a medical card with an embedded RFID tag. Therefore, robot is able to identify many people simultaneously.

### 5.3.1 Experimental Protocol

As an experimental protocol, we define the following hypothesis: *"Is it possible for individuals to access their private data generated during their interactions with the robot without any security breach for other participants?"* In other words, we want to know whether users can control **only** their data, and personal data is correctly outlined.

Further, we we consider the following three stages:

1. Pre-experiment:
    - Configuration of the two users smart cards identified by RFID tags,
    - Preparation of one social robot with its dedicated camera for image recording and RFID sensor used for detecting and positioning of RFID tags,
    - Reset of the Blockchain and database, and create 1 (or 2) wallet key pair(s) (one for each user),
    - Register each user in the network with his/her RFID and key pair,
    - Deploy the BlockRobot API and a Mongo database (off-chain repository).
2. Experiment execution:
    - The individuals and the robot act as peers and are connected to the local network,
    - Users walk near the robot, and the robot detects a person in less than 5 seconds (in the range of 3 meters).
3. Post-experiment:
    - Extraction of data logs and process mining techniques are performed to evaluate the system's performance.

### 5.3.2 Experiments

We performed many different experiments of humans interacting with Mbot illustrated on Fig. 5.10 producing RGB-D images. Figure 5.11 exemplifies an image captured with a person carrying a RFID, and Figure 5.12 an image without an identified person. We summarised them in the four main scenarios.


**Figure 5.10:** MOnarCH Robot – the social robot used in the experiments

1. One person walking by the robot carrying the RFID card. The robot records streams of RGB-D images and detects an RFID tag. At the end of the scenario, the person can access his/her data and visualize the images recorded by the robot.

2. The same scenario as before, with one person and one robot recording RGB-D images and detecting the person with its RFID laser. However, in this scenario, the data is corrupted. An example of corrupted data may result from a malicious party gaining access to the off-chain repository and modifying the recorded data. This configuration shows the *Integrity* property of the personal data assured by the proposed framework because of the use of Blockchain as a trust anchor. If some data is corrupted, then the integrity test will fail. Still, the user can access his/her personal data

**Figure 5.11:** RGB-D Image captured by robot with one RFID identified person.

**Figure 5.12:** RGB-D Image captured by robot without RFID identified person.

and have the information that the data was corrupted.

3. A third scenario manifest slightly different configuration. Here, two users walk by the robot, one at the time, without intersecting their images. This experiment demonstrates the feasibility of the BlockRobot algorithms to outline the private data to the participants correctly. As a result of the experiment, two individuals could access their private data without causing any security breach to the other participant.

4. In the last experiment, two individuals walk by the robot at the same time. In the real-world scenario, this is possibly the most common situation because it is common to have many people walking in the hospital corridors simultaneously. In this scenario, both individuals were identified by the robot, and their RFID intervals intersected, meaning that two participants are identified in the same image. Both can access the images because it is their private data. However, the other's individuals shapes are blurred as an attempt to anonymize them in the image. In such a case, the participants' confidentiality is guaranteed because it is not possible to detect them; hence their identities are covered by applying the anonymization techniques, e.g., blurring faces. Therefore the user can access data or revoke access (delete data) without breaching of data privacy of other users as no information is given about other data subjects' identities in the recording.

## 5.4 Results

Regarding the performance of the system, Figures 5.13 and 5.14 depict the results of logs analysis performed by process mining. The data was generated from experiment 1 and 2 execution, respectively. Further data is presented in Figure 5.15. Each operation corresponds to the following system's event.

- IMAGE_RETRIEVE – the generation of a new image capture by the robot;

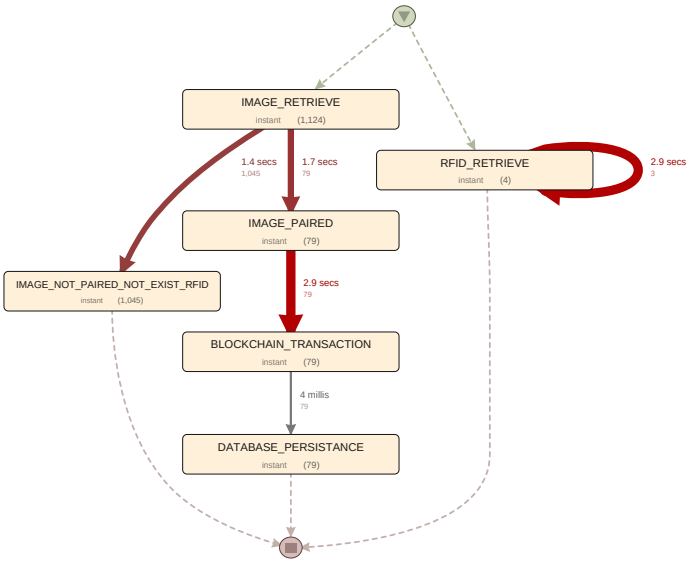


**Figure 5.13:** Graph showing the execution of system events during experiment 1.

- IMAGE_NOT_PAIRED_NOT_EXIST_RFID – no RFID can be associated with a given image;
- RFID_RETRIEVE – the generation of a new RFID capture by the robot;
- IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID – an RFID is identified, but it is not provisioned, therefore it cannot be associated with any image;
- IMAGE_PAIRED – a pair between image and RFID is found;
- BLOCKCHAIN_TRANSACTION – the hash of the image is stored in Blockchain;
- DATABASE_PERSISTANCE – the encrypted image is stored in the off-chain database.

In both experiments, after the initial setup of the RFIDs (with RFID_RETRIEVE event), the most time consuming event occurs when IMAGE_PAIRED execute successfully and BLOCKCHAIN_TRANSACTION is triggered. Showing that Blockchain transaction plays a cost-effective part of the system. Moreover, all the IMAGE_PAIRED instances have a correspondingly BLOCKCHAIN_TRANSACTION instance, therefore, no transaction has been considered corrupted. In the first experiment, the difference from the activity frequency of IMAGE_RETRIEVE to the frequency of DATABASE_PERSISTANCE shows that only 7% of the images are being paired and stored encrypted. With the increase of RFIDs, in second experiment, this value increase to 13%. In the second experiment (Figure 5.14), a more complex network of events shows that in some instances are identified some other RFIDs that were not involved directly in this

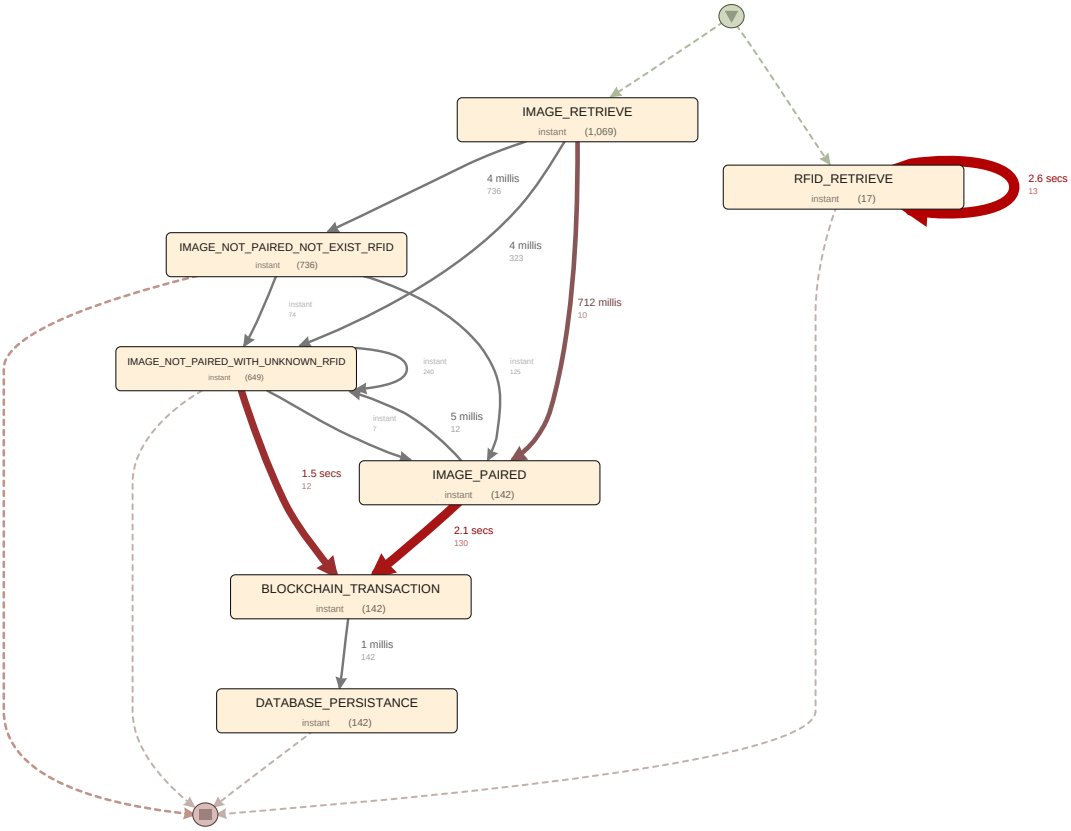


**Figure 5.14:** Graph showing the execution of system events during experiment 2.

experiment (*e.g.*, other health care personnel). In that situations, only the identified RFID is considered and a BLOCKCHAIN_TRANSACTION triggered.

All the code relative to the current implementation is open-source and available on Github [1] altogether with the datasets from discussed experiences [2].

| | | IMAGE_RETRIEVE | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | RFID_RETRIEVE | IMAGE_PAIRED | BLOCKCHAIN_TRANSACTION | DATABASE_PERSISTANCE |
|---|---|---|---|---|---|---|---|
| Absolute Frequency | IMAGE_RETRIEVE | | 145 | | 79 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 3 | | | |
| | IMAGE_PAIRED | | | | | 79 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 79 |
| | DATABASE_PERSISTANCE | | | | | | |
| Case Frequency | IMAGE_RETRIEVE | | 145 | | 79 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 1 | | | |
| | IMAGE_PAIRED | | | | | 79 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 79 |
| | DATABASE_PERSISTANCE | | | | | | |
| Maximum Duration [ms] | IMAGE_RETRIEVE | | 1411 | | 1672 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 2883 | | | |
| | IMAGE_PAIRED | | | | | 298 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 4 |
| | DATABASE_PERSISTANCE | | | | | | |
| Maximum Repetition | IMAGE_RETRIEVE | | 1 | | 1 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 3 | | | |
| | IMAGE_PAIRED | | | | | 1 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 1 |
| | DATABASE_PERSISTANCE | | | | | | |
| Mean Duration [ms] | IMAGE_RETRIEVE | | 12 | | 687 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 2597 | | | |
| | IMAGE_PAIRED | | | | | 1995 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 1 |
| | DATABASE_PERSISTANCE | | | | | | |
| Median Duration [ms] | IMAGE_RETRIEVE | | 3 | | 675 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 2559 | | | |
| | IMAGE_PAIRED | | | | | 23 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 1 |
| | DATABASE_PERSISTANCE | | | | | | |
| Minimum Duration [ms] | IMAGE_RETRIEVE | | 1 | | 3 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 235 | | | |
| | IMAGE_PAIRED | | | | | 231 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | |
| | DATABASE_PERSISTANCE | | | | | | |
| Total Duration [ms] | IMAGE_RETRIEVE | | 1326 | | 54279 | | |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | | | | | | |
| | RFID_RETRIEVE | | | 7792 | | | |
| | IMAGE_PAIRED | | | | | 157682 | |
| | BLOCKCHAIN_TRANSACTION | | | | | | 95 |
| | DATABASE_PERSISTANCE | | | | | | |

**((a))** Performance results from experiment 1.

| | | IMAGE_RETRIEVE | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | RFID_RETRIEVE | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | IMAGE_PAIRED | BLOCKCHAIN_TRANSACTION | DATABASE_PERSISTANCE |
|---|---|---|---|---|---|---|---|---|
| Absolute Frequency | IMAGE_RETRIEVE | 0 | 736 | 0 | 323 | 10 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 74 | 125 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 13 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 240 | 7 | 12 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 12 | 0 | 130 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 142 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Case Frequency | IMAGE_RETRIEVE | 0 | 736 | 0 | 323 | 10 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 74 | 125 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 240 | 7 | 12 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 12 | 0 | 130 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 142 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Maximum Duration [ms] | IMAGE_RETRIEVE | 0 | 1731 | 0 | 1111 | 1343 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 8991 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 1 | 1 | 1838 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 7 | 0 | 3373 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 5 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Maximum Repetition | IMAGE_RETRIEVE | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 11 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Mean Duration [ms] | IMAGE_RETRIEVE | 0 | 125 | 0 | 14 | 741 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 3048 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 0 | 0 | 1386 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 5 | 0 | 2058 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Median Duration [ms] | IMAGE_RETRIEVE | 0 | 4 | 0 | 4 | 712 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 2558 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 0 | 0 | 1538 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 5 | 0 | 2097 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Minimum Duration [ms] | IMAGE_RETRIEVE | 0 | 2 | 0 | 1 | 426 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 1931 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 0 | 0 | 238 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 4 | 0 | 502 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Total Duration [ms] | IMAGE_RETRIEVE | 0 | 92412 | 0 | 4565 | 7410 | 0 | 0 |
| | IMAGE_NOT_PAIRED_NOT_EXIST_RFID | 0 | 0 | 0 | 10 | 8 | 0 | 0 |
| | RFID_RETRIEVE | 0 | 0 | 39633 | 0 | 0 | 0 | 0 |
| | IMAGE_NOT_PAIRED_WITH_UNKNOWN_RFID | 0 | 0 | 0 | 30 | 1 | 16639 | 0 |
| | IMAGE_PAIRED | 0 | 0 | 0 | 62 | 0 | 267631 | 0 |
| | BLOCKCHAIN_TRANSACTION | 0 | 0 | 0 | 0 | 0 | 0 | 163 |
| | DATABASE_PERSISTANCE | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**((b))** Performance results from experiment 2.

**Figure 5.15:** Performance comparison from both experiments. The matrices present the source (rows) and target (columns) system's events.

[1] https://github.com/vvasylkovskyi/eos-web

[2] https://github.com/vvasylkovskyi/eos-web/tree/master/experiments

# 6

# Evaluation

## Contents


6.1 Objectives of the Evaluation . . . . . 65
6.2 Descriptive Evaluation . . . . . 66
6.3 Design-oriented evaluation principles for conceptual model . . . . . 68
6.4 Discussion . . . . . 70

This chapter corresponds to the fifth step of DSRM: **Evaluation**. Evaluation is the process of observing and measuring how well the artifact supports a solution to the problem. The chapter structures as following: first, we provide the objectives of the evaluation; second, we apply the Österle et al. principles on the conceptual model proposed in Chapter 4; third, we evaluate the artifact. The chapter terminates with discussion.

## 6.1 Objectives of the Evaluation

The current section intends to explain the basis for the evaluation of both the conceptual model and the developed artifact.

### 6.1.1 DSRM evaluation methods to validate artifacts

In order to accurately assess the artifact proposed in Chapter 4, the DSRM framework suggests different design evaluation methods to validate artifacts: **Observational**, **Analytical**, **Experimental**, **Testing**, and **Descriptive** [76].

We decided to use the Descriptive method to evaluate privacy features:

- **Descriptive** – Simulate scenarios to evaluate the artifact.

Also, despite the **Experimental** evaluation being interesting in the context of the proposed software design, we did not cover it in this thesis scope. However, performing such an evaluation would be a natural step because it would assess the individuals' perception of privacy, and we suggest it as a future work in the conclusions.

### 6.1.2 Design-oriented evaluation principles for conceptual model

To address the evaluation of the conceptual model proposed in Chapter 4, we are going to apply the Österle et al. principles for design-oriented IS research [77]:

- **Abstraction**: Each artifact must be applicable to a class of problems.
- **Originality**: Each artifact must substantially contribute to the advancement of the body of knowledge.
- **Justification**: Each artifact must be justified in a comprehensible manner and must allow for its validation.
- **Benefit**: Each artifact must yield benefit – either immediately or in the future – for the respective stakeholder groups.

storages

| | _id ObjectId | txid String | hash_uid String | stamp_secs Int32 | stamp_nsecs Int32 | seq Int32 | height Int32 | width Int32 |
|---|---|---|---|---|---|---|---|---|
| 1 | 5e63c26f9cc99202fde4f70f | "677a7fd10347c15aa20c106eb3a0cf9a3e1 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 22996313 | 2434946 | 480 | 640 |
| 2 | 5e63c26f9cc99202fde4f710 | "2d0045274b4837085053962011fa4d0ca4 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 54906290 | 2434947 | 480 | 640 |
| 3 | 5e63c2709cc99202fde4f711 | "62f1147c72d322b120865f7dd113b75469 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 86692870 | 2434948 | 480 | 640 |
| 4 | 5e63c2709cc99202fde4f714 | "7d4070942cb55d6f2d459d0e008c93053d | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 154653271 | 2434950 | 480 | 640 |
| 5 | 5e63c2709cc99202fde4f713 | "779c28139fe1c7fa7c161f19d177c9f5ce3f | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 223026380 | 2434952 | 480 | 640 |
| 6 | 5e63c2709cc99202fde4f712 | "1ac373451ed47c1dfc2d6c4f2327a460665 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 254663428 | 2434953 | 480 | 640 |
| 7 | 5e63c2719cc99202fde4f716 | "04e3bc21ea3749abc308b8ae2a975722306 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 322710422 | 2434955 | 480 | 640 |
| 8 | 5e63c2719cc99202fde4f715 | "6e0d21947b0b96dc1875ae4e5a76188c07 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 354729917 | 2434956 | 480 | 640 |
| 9 | 5e63c2719cc99202fde4f717 | "707b6420d73fc523946557798d58f90e5b3c | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 387436225 | 2434957 | 480 | 640 |
| 10 | 5e63c2719cc99202fde4f719 | "68a459dd19d38b6219a629247d281acf8d | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 431382281 | 2434958 | 480 | 640 |
| 11 | 5e63c2719cc99202fde4f718 | "cbd83022368f4d40eb5d2c76244a2a32903 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 487017999 | 2434960 | 480 | 640 |
| 12 | 5e63c2729cc99202fde4f71a | "262c415886d2af30bb54091608bc9b9873 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 555103241 | 2434962 | 480 | 640 |
| 13 | 5e63c2729cc99202fde4f71b | "8a7fb0943b504caf8361d50aef582cf814c6 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 586719918 | 2434963 | 480 | 640 |
| 14 | 5e63c2729cc99202fde4f71d | "be2c57416826b806ca57e90af30a589979f | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 623059620 | 2434964 | 480 | 640 |
| 15 | 5e63c2729cc99202fde4f71c | "bed72ce1fe082b9b3b69897ea73aa1e864b | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 686740037 | 2434966 | 480 | 640 |
| 16 | 5e63c2739cc99202fde4f71e | "32c44ec32ed684a7f66c89763001eab88ab | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 754802261 | 2434968 | 480 | 640 |
| 17 | 5e63c2739cc99202fde4f71f | "27fa65a4f31ee6d640706c3b399e21073d | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 786782024 | 2434969 | 480 | 640 |
| 18 | 5e63c2739cc99202fde4f720 | "dc4316262f1a6fc0c26c2951e7e96d8d637 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 854700746 | 2434971 | 480 | 640 |
| 19 | 5e63c2739cc99202fde4f722 | "241f4cf8e67be766d0f948a0c1c109006455 | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 922645727 | 2434973 | 480 | 640 |
| 20 | 5e63c2739cc99202fde4f721 | "6b98896e39786d12df93405d46cca26c50ff | "7d3d3351fad06ecf58ecea911154c22636ff | 1583346683 | 954851203 | 2434974 | 480 | 640 |

**Figure 6.1:** Sample data gathered in Database during the experiments

## 6.2 Descriptive Evaluation

The descriptive evaluation should illustrate the artifact's ability to satisfy this thesis's objectives – **enhancing control and transparency of private data in HRI**.

In the previous Chapter 5, we discussed the setup of the prototype EOSIO-based Blockchain DAP integrated with the MOnarCH robot. We described two experiments:

- The robot detects one person,
- The robot detects two persons simultaneously.

The robot acts as a peer in Blockchain, and for each robotic event with DS present, the DAP automatically creates a transaction, which follows the database transaction. For our experiment's simplicity, our local EOSIO Blockchain has three nodes performing transactions: the robot and two individuals in the experiment.

During the first experiment, an individual previously registered to our DAP walks by the robot. The robot detects a person and starts performing BC transactions with images as a payload, resulting in 79 images of 1MB each. Then, the images are hashed and their hashes stored on BC ledger, which means 79 BC transactions were created and 79 new blocks appended to the BC. Further, the images are encrypted and stored on an external database ex. of data stored on a database in Fig. 6.1. These experiments demonstrate that the robotic events can be audited by storing them on the BC. Most importantly, we have observed that despite the Blockchain transactions playing a cost-effective part in the system, i.e., taking more time; no image skipped the on-chain storage, even given the high frequency of robotics events. Moreover, only hashes of the images are stored on-chain, resulting in better scalability because hashes do not occupy much memory.

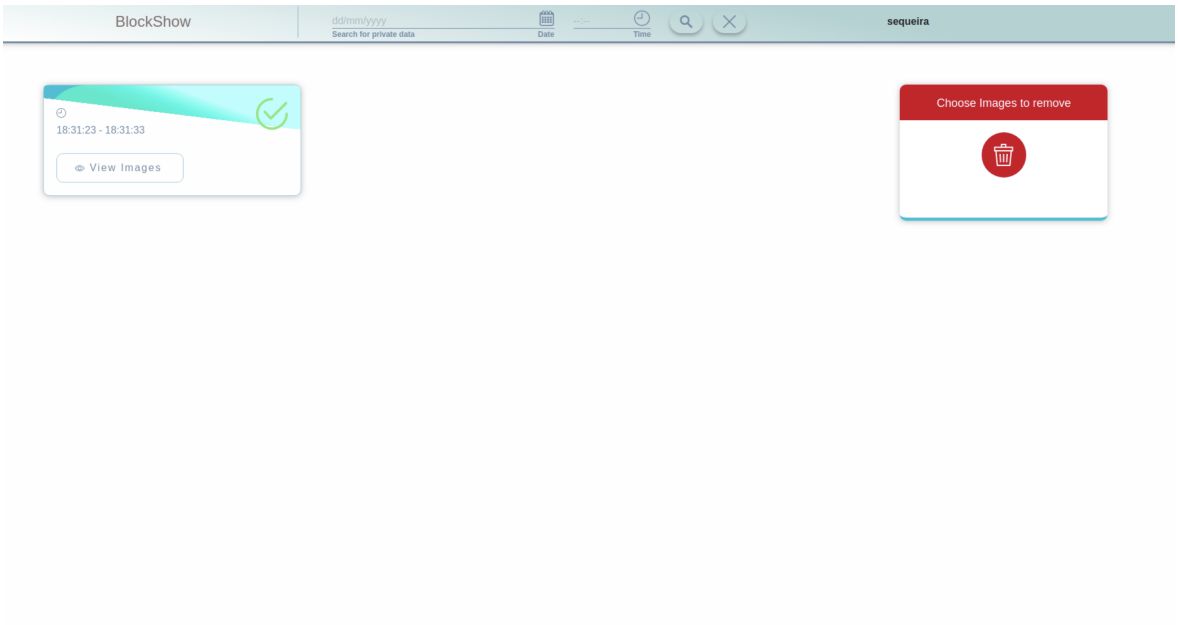


**Figure 6.2:** Privacy Dashboard - User accesses his/her private data in **BlockRobot**

With the proposed privacy-by-design DAP, the user can access the DAP with his/her crypto-wallet and see the *privacy dashboard* as an example of a prototype in Fig. 6.2. This privacy dashboard gives the user a clear vision of his private data collected during HRI. Further, the user can access that data by selecting the personal data card from the list. As he/she does that, he/she creates a transaction to the BC, which will create a new block with an event that represents new access to that piece of private data – which further will give him the **transparency** regarding the previous accesses on data and **accountability**.

As the user moves forward, the next screen Fig. 6.3 shows him the RGB-D images that the robot had recorded. Besides, the user can also see the data's current status (top-left corner) to understand


**Figure 6.3:** Private Data visualization screen

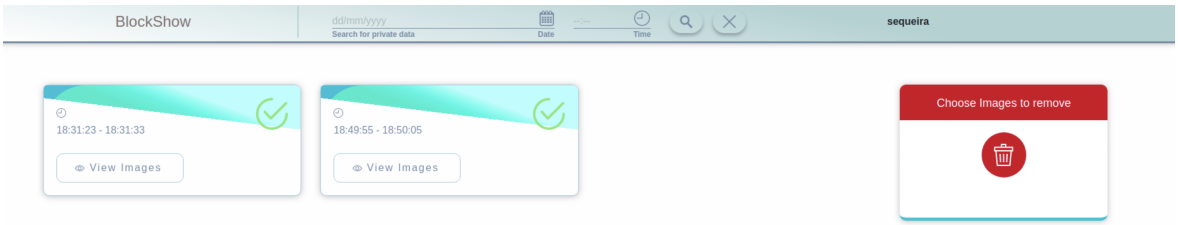


**Figure 6.4:** Privacy Dashboard - User's data after several interactions with robot in **BlockRobot**

whether the data verification against Blockchain was successful and assert the **integrity** of data. On the bottom, the user can see the list of previous accesses, which gives him/her the ability to see his/her data processing activities in retrospective.

During the second experiment, two individuals passed by the robot simultaneously, and both were detected and had images recorded (Fig. 6.4). Further, we can see from the images that the first user, the same as visualizing his data in the first experiment, now has two private data cards, and therefore, he can choose what data to access or erase.

As for the second user, we can notice that he can only see one card (Fig. 6.5), which means that he cannot access the data he was not identified. Therefore, by this experiment, we prove the **privacy by opacity** – meaning that individuals can only see the private data that belongs to them.

This descriptive evaluation method proves that the artifact satisfies the problem identified in this thesis – users have more control and transparency of their private data in HRI.

## 6.3 Design-oriented evaluation principles for conceptual model

Based on the Österle et al. [77] concept model evaluation principles, the results are as follows:

- **Abstraction** – The conceptual model solves the privacy problem as robots are present in many different industries. The idea can be applied not only to the SR but to other types of robots, such as an assistant or factory robots. Further, it solves the problem of having CA managing private

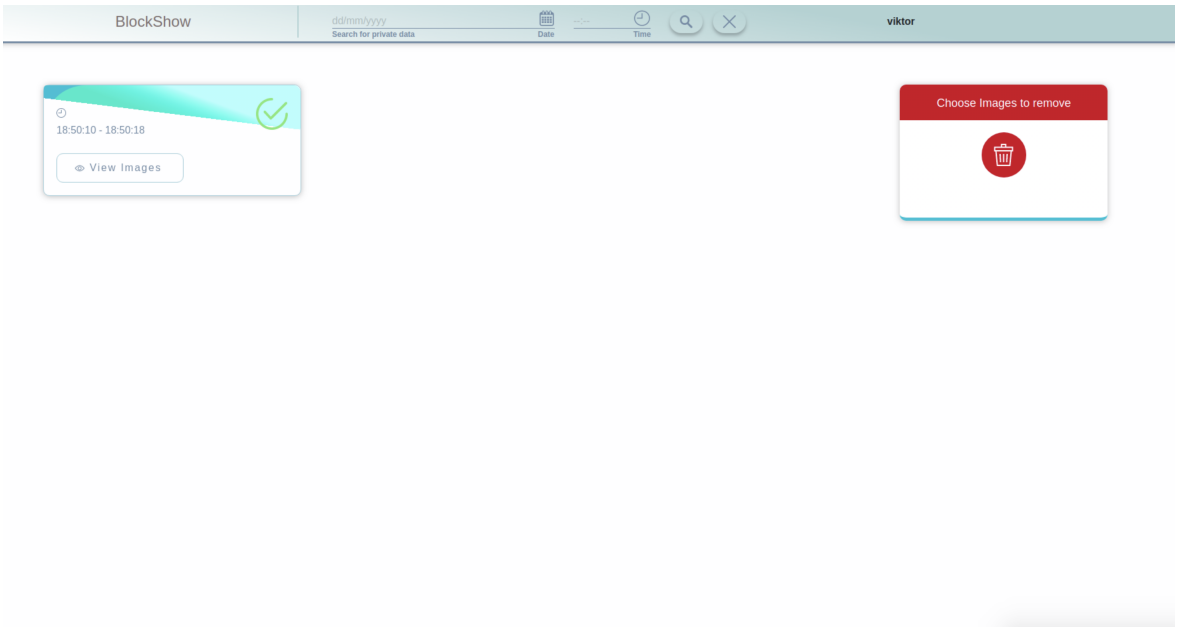


**Figure 6.5:** Privacy Dashboard - User cannot access other user's data

data in HRI, including it can extend to the classes of problems of private data management in other areas.

- **Originality** – The conceptual model is original. To the best of our knowledge, there is no existing privacy-by-design architecture for the HRI based on BC and RFID technology to the date of writing of this thesis. Moreover, the use of BC with RFID and crypto-wallets for identity management provides access to the private data in HRI without actually revealing individual's identity. Instead, the use of anonymity protects individual's identity, which is also an innovative approach.

- **Justification** – The conceptual model is justified because the privacy requirements were presented and can be resolved by applying the design. The developed prototype justifies the design by demonstrating the privacy features in Chapter 5 identified as essential in the privacy requirements listing.

- **Benefit** – The conceptual model yields benefit because the stakeholders (end-users) can have better privacy in HRI after integrating the SR with software based on the conceptual model. Besides, the BC improves private data security by providing the verifiability and accountability and elimination of a centralized party.

All four principles of Österle et al [77] where accomplished which validates the conceptual model.

| Privacy Requirements | Design Guidelines |
|---|---|
| Purpose limitation | SR moving in environment |
| Data minimization | Recording of the Depth images |
| Storage limitation | Destruction of data in SR |
| Integrity | Hashing of data on-chain |
| Confidentiality | Encryption of data off-chain |
| Provenance | SR interacting as a peer in BC |
| Accountability | Audit history on-chain |
| Accuracy | Identity Pseudonimyzation/RFID |
| Access | Identity Management/Crypto Wallet |
| Withdraw the concent or deletion | Storage of data off-chain |

**Figure 6.6:** Privacy Validation – GDPR Requirements and BlockRobot Design Guidelines mapping

## 6.4 Discussion

In this section, we will discuss the work done in the Thesis. We start by validating the privacy requirements and evaluating the privacy improvement by qualitative metrics, evaluating the security and privacy threats, and assessing the GDPR' Right to be Forgotten. Finally, we will discuss the Blockchain competence in solving the presented thesis issue.

### 6.4.1 Privacy Validation in HRI

Figure 6.6 maps the GDPR imposed requirements with the BlockRobot design options. The **principle of purpose limitation** is fulfilled as the SRs record the data only when they need to move around in the environment. **Data minimization** is reached with the depth-images, only shapes are seen, and these shapes do not provide complete human identification (cf. Figures 5.11 and 5.12). **Storage Limitation** is enforced with data destruction after it is no longer necessary [78].

In the previous section, we have presented the *Descriptive evaluation* method by simulating scenarios that are used to evaluate the proposed solution. Results of experiences validate one of the private data's dualities in HRI stated in Chapter 3, the *privacy by opacity*, meaning that no private data is outlined incorrectly to other data subjects. Moreover, with presented privacy-by-design DAP for HRI, users are ensured with **"Right of access"**, **"Right to be Forgotten"** and with **accurate** data. Also, it becomes possible to envision the **provenance** of the data by robot interacting with the Blockchain as a peer. The blockchain ledger is immutable; accordingly, it is possible to **verify the data integrity** by hashing private data and placing hash on-chain. The private data remains off-chain encrypted, and therefore, it is **confidential**. Moreover, the solution provides the possibility to audit history of the data, which could potentially resolve the **accountability** requirements. Individuals can perceive the data processor's activities in retrospective.

These results corroborate the related work stating that robots can support the basic principles of privacy, e.g. MOnarCH [78].

### 6.4.2 GDPR and Right to be Forgotten

GDPR' **"Right to be Forgotten"** requirement imposes the ability for the data subject to request erasure of the private data. Therefore, it is mandatory to avoid placing the private data on BC due to its immutability properties. It is technically impossible to erase the data from BC. Conversely, by storing the private data off-chain, the user can erase his/her private data [50]. However, if the hash of private data that remains on-chain is considered as a private data, then an open question remains [14]. On the other side, even if the malicious user can see the publicly available hash of RGB-D image on BC, it seems very unlikely that given that hash, he/she may be able to reconstruct or guess the binary file from the hash. Therefore, hashes of binary files, such as RGB-D images, when stored on BC, at first glance, may seem unlinkable to any outer information. Additionally, the vague definition of the term private data *"erasure"* in GDPR imposes certain flexibility. If the data on BC is sufficiently anonymized, then the DS' identification is not possible. In such a case, in some interpretations, this data, by being inaccessible / unreadable, shall be considered as destroyed or erased. Whether this fully satisfies the GDPR is still an open question for future research [14, p. 76].

### 6.4.3 Security and Privacy Threats

Identity management based on RFID and BC is an innovative approach. In BlockRobot, identity management is based on placing the RFID tags on the BC, which any network member can visualize. One of the privacy-sensitive topics to discuss is the degree to which the placing of RFID tags on the BC is secure. If the owner of the RFID tag happens to be discovered, then everybody in the network knows who the individual that has been *"seen"* by the robot is. However, it is worth noting that they will not access the individuals' images even in his/her identity disclosure. For that reason, it is crucial to anonymize the RFID tags before placing them in BC. In the definition of the GDPR [14], the anonymity is the degree to which it is impossible to re-identify a data subject from the data. We attempt to anonymize that data by hashing the UIDs extracted from the tags; however, a sufficiently motivated malicious entity may re-identify the hash's actual UID. The platform might be prone to pattern analysis despite the use of hashed UIDs. Whether the storage of hashed RFID tags on Blockchain provides the irreversibility in identification is the specific question to each system/business involved, in our case, HRI.

#### 6.4.3.A Hashes Security

There are many different ways to crack the hash. Some of the attacks that can be applied are pre-image attack, second pre-image attack, collision attack, birthday attack, length extension attack, and many more [79]. We do not intend to evaluate the strength of the hashed messages in this thesis; however, we leave some directions for future investigations.

Brute Force attack is the only attack that works on all cryptographic hash functions. It is the exhaustive search for a message. Similarly to cryptographic encryption/decryption systems, where this attack intends to find a secret key, this attack finds the actual message in hash functions. The measure of the resistance to the Brute Force attack is the function of the length *(n)* of the message. Given the message of length *(n)*, the effort required to succeed is $2^n$, i.e., the attacker tries $2^n$ possible messages until he/she finds the match with the actual hash. Besides the length of the message, another factor influencing the hashed message's security is the strength of the hash function itself. Some of the highly used hash functions are *MD(x), JH, SHA-1, SHA-2, SHA-3* (and the rest of *SHA*-family hash functions). Different hash functions offer trade-offs such as security of the message (the more secure the hash function means, that more time it will take to the hacker to crack it) versus the speed of hashing. Usually, if the hash function is more secure, then slower is the hashing process. We use *SHA-256*, which is an example of a highly secure hash function.

Based on these properties of hashes, we assume that the security of depth images is much better than the security of hashed RFID tags. A Depth image can be seen as the message size of a binary file. Therefore the length *(n)* is high, and the hash is secure. Therefore, we assume that the hashes of depth images on BC are a secure option. On the other hand, an RFID tag in our experiment is a message of length $3 < n < 6$. Therefore, we suggest additional validations of security of storing RFID tags on-chain.

#### 6.4.3.B BlockRobot API as a Trusted Server

The use of the BlockRobot API as a trusted intermediary impose some security risks. The evident one is that BlockRobot API is the single point of failure in the system. One could perform a Denial of Service (DOS) attack, in which an attacker sends a stream of traffic data to a targeted system to overload the system and stop it from regular functioning [80]. In our case, the target system is the BlockRobot API.

Moreover, the robotic events transferred through the network are subject to a *Man-in-the-middle (MITM)* attack. A MITM attack includes an unauthorized actor positioning its system in transmissions between a user and a trusted party to capture or eavesdrop the messages [81]. In our system, a malicious entity can position itself between the robot and a BlockRobot API and, therefore, intercept all the private data leaving the robot before it is stored on BC. These types of attacks have happened previously, even in the context of a distributed system where there is no central authority [82]. Additionally,

the data produced by the robot is stored on ROS topics in raw format and can result in unauthorized access and disclosure of data from the source.

Hence, in this thesis, the reason for using the BlockRobot API as an intermediary is to simplify the experimental environment to demonstrate the concept's feasibility. As future work, We suggest enforcing the security by removing the BlockRobot API. As a result, the robots should interact with BC directly by issuing transaction as peers. Accordingly, the encryption should happen within the robotic software before issuing a transaction. Finally, the decryption of the private data, plus all the necessary data transformations, should happen in the client, where it is secure to perform the decryption operations.

### 6.4.4 Blockchain Evaluation

Blockchain is still an emerging technology. The technology demands evaluating its cost, performance, security, functionality, scalability, and decentralization. In particular, properties such as security, performance, and decentralization are currently a "Blockchain Triangle dilemma" [83]. There is still a great progression margin regarding performance. Each image captured by robots needs to be classified by the data owner and verified on BC, including robots taking hundreds of images per minute.

Additionally, security and consistency are concerns to maximize, granting the trustability of images. Therefore, EOSIO BC seems promising regarding the increase of transaction throughput by using the DPoS consensus algorithm in favor of sacrificing a degree of decentralization [43]. DPoS also resolves the problem of high energy consumption as in the PoW algorithm, which favors the potential of scalability. However, by being a relatively new platform, many uncertainties exist. The most recent studies [84, 85] attempted to analyze EOSIO data, identified the most critical security concerns.

Besides the EOSIO Blockchain, the previous work indicates there are more Blockchain-based solutions capable of interoperating with Robots, such as *Tezos* [63, 65] or *Aitheon* [86]. Not to mention, there is a great variety of BC-based platforms at disposal. Among several mentioned, Hyperledger Fabric [45], Tendermint [46], Quorum [87] are permissioned BC based on BFT consensus would likely be highly useful in context of robotics by allowing an arbitrary faulty behaviour. We did not analyze the possibility of corrupt private data due to the robotic's sensors' corruption in this thesis's scope but would be a worthy investigation. For example, some robots could have corrupted sensors and produce incorrect data. Therefore, having more robots interacting as peers could increase data reliability because the byzantine robot's data would not cross the block validation phase.

Moreover, it is crucial to evaluate the performance of the system with more nodes (robots). We assume that having robots participating in the Blockchain will result in more images being produced. Additionally, it will result in slower block validations for some consensus algorithms. Nevertheless, integrating robots with the BC is still a new research area, and some other exciting consensus protocols can be proposed. In particular, Proof-of-Sensing is the consensus protocol that could fit for multiple

robots detecting RFID tags (only robots that can produce a specific sensory output can send/validate transactions) [88].

In conclusion, the adoption of BC technology seems to be highly beneficial when integrating with robotic events. In this thesis, in particular, we discussed how BC offers privacy-friendly qualities such as accountability, verifiability, and provenance because BC is immutable. Another benefit is that decentralization and the elimination of centralized authority enable the data subjects to control their private data. Furthermore, the BC interoperability is at its growing pace. BC Interoperability is the ability of two or more BC to cooperate, despite differences in interface, execution platform, or language [89]. In this line of reasoning, we foresee that soon it will be possible to have several Blockchains interoperating. In such a scenario, BlockRobot may assist in improving privacy in HRI, not excluding the future integration with other BC that can assist in robotics for another purpose [89].

# 7

# Conclusion

## Contents


7.1 Communication . . . . . 78

7.2 Limitations . . . . . 78

7.3 Future Work . . . . . 79

This chapter corresponds to the **conclusion** of the work developed in this thesis.

SRs bear potential in areas such as personal assistant robots, healthcare, manufacturing industry, education assistants, defense agents, public space hosting, and many more. In particular, robots with humanoid traces are likely to interact with humans at an emotional level because humans tend to bond when emotions are involved. Individuals have limited knowledge about the robots' inference abilities and what information they may happen to know during their HRI. We need to provide them with adequate privacy-by-design features to comply with GDPR requirements, both legally and ethically, enforcing transparency, as humans may have limited knowledge regarding how their private data is collected during HRI. An increase in human awareness could improve the trust and acceptance of robots by humans. BlockRobot aims at being a step in the right direction while stimulating the discussion on privacy-by-design in SR.

We did extensive research on the topic of robotics and privacy and, in particular, privacy-sensitive robotics. It turns out that there are no many results regarding the privacy-by-design solutions for interactions between robots and humans. We have observed that Blockchain and the GDPR seem profoundly incompatible at a conceptual level at first sight, but concluded that if appropriately designed, Blockchain may serve the GDPR in the way of enhancing privacy. One possible solution to increase privacy between robots and humans was to provide a privacy-by-design DAP based on Blockchain technology, as it provided valuable benefits. The benefits of being immutable will provide a history audit for data that presents individuals with transparency regarding their private data processing. Another clear benefit is removing a trust authority, which removes the risk of having a malicious intermediary.

It is also worth mentioning that there are some drawbacks to adopting Blockchain technology to assist in privacy in HRI. On one side, Blockchain requires expensive computations of hashes in order to verify the transactions. On the other side, robotic events happen at a very high frequency. Together, there is a risk of privacy being very expensive on an industrial level. Besides that, Blockchain is a new technology, and, therefore, very immature, and organizations have little knowledge about the technical aspects of it, plus the knowledge is sparse, and the labor is expensive. All these factors difficult the adoption of Blockchain technologies.

To tackle the identified problem in this thesis, we designed a conceptual model for the generic Blockchain-based DAP and presented an artifact, instance of the designed model. The latest further assisted us as a proof of concept to demonstrate the increase in privacy in HRI in the healthcare environment case study. We evaluated the proposal by the following methods:

- Evaluation of the conceptual model by applying Österle et al. principles [77],
- A descriptive evaluation of the prototype measuring completeness, in other words, how well the artifact solves the problem [27].

These evaluations demonstrated that the conceptual model applies to many use cases – it is

abstract and justifies its existence since it provides the intended benefit – an increase of privacy in HRI. It is original because privacy-sensitive robotics is a new field, and there are no many solutions existent to the date of writing this thesis. Moreover, applying Blockchain to tackle privacy issues in HRI is an original approach.

The experiments with the prototype have shown that it is possible to achieve the desired result by applying the proposed BC-based design model. Furthermore, by being an outstanding technology among BC-based DAPs, EOSIO Blockchain provides the advantage of being fast – a well-suited option for the integration with robotic events that happen at high velocity.

## 7.1 Communication

The solution to the problem identified in this thesis was communicated to two international conferences and one workshop and **accepted** by each one of them:

1. *The 2nd International Workshop on BCT4ROS 2020*:
    - V. Vasylkovskyi, S. Guerreiro and J. Sequeira, "BlockRobot – Blockchain-based Architecture to Enforce Privacy in Human Robot Interaction"
2. *The 3rd IEEE Internation Conference on Blockchain (Blockchain - 2020)*
    - V. Vasylkovskyi, S. Guerreiro and J. Sequeira, "BlockRobot: Increasing Privacy in Human Robot Interaction by Using Blockchain"
3. *The 54th HICSS on minitrack Human-Robot Interactions*
    - V. Vasylkovskyi, S. Guerreiro and J. Sequeira, "Designing and Validating a Blockchain-based Architecture to Enforce Privacy in Human Robot Interaction"

## 7.2 Limitations

The limitations identified in this thesis have several sources. For starters, it is relevant to know that for the sake of simplicity, we performed test cases on one specific social robot that belongs to the MOnarCH. In theory, it should be possible to apply the current solution to all social robots that belong to the MOnarCH system. However, the lack of tests with several robots working in the same blockchain network and lack of a variety of robotic systems (other than MOnarCH), it is not possible to ultimately declare that this solution is appropriate to all robots.

Furthermore, a limitation regarding the technological aspects of robots directed our investigation towards developing secure algorithms for correct private data outline to the participants. MOnarCH [78]

robots are equipped with RFID lasers which allow them to identify individuals near by [8]. Current, up-to-date image recognition algorithms possibly would provide the correct identification of individuals without the need of RFID technology. These technical aspects are important because they shape the research directions. RFID detection sensors and depth image recording devices in social robots are separate devices, and there is no direct association between identified RFIDs and image captions, and to establish such an association, we used the logical connections between the timestamps intervals and RFID detection. Nevertheless, we attempted to design software with a sufficient level of abstraction so that in the future, it will be possible to perform similar evaluations with robots that do not have RFID lasers, but instead image recognition techniques to identify participants – owners of private data.

Importantly, RFID tags are subject to background noises, depending on the environmental conditions [8]. Despite proved as working in our supervised experiments, the results could be different regarding the completeness of the information in a real-world scenario. Hospital wards vary in construction, which can impact RFID signals detection and, therefore, slow down the data subject identification [8]. Despite our lack of evaluating the solution in terms of scalability and latency, hopefully, in the future, the advancements in RFID technologies will improve the quality data detection, and consequently, our algorithms, based on RFID, will also improve.

## 7.3 Future Work

With the proposed privacy-by-design architecture model, developed and tested prototype, we demonstrated the BC-based solutions could pursue GDPR compliance. We leave for future research some suggestions regarding the improvements. In particular, performing demonstrations with real stakeholders to evaluate the perceived **fairness** and **transparency** by individuals during HRI is a natural step. Transparency will be validated by performing the experiments in a controlled environment with real users because it is up to them to perceive the transparency requirement's subjective nature. We believe that the same principle applies to the fairness validation, as it results from user awareness regarding what happens to his/her data. Moreover, from the *ethical perspective*, as users tend to bond with anthropomorphic robots [10], we are keen to think that giving information transparency to the individuals regarding what data is recorded, where it is stored, and how to access and manage that data should increase awareness and trust. Furthermore, to increase the user experience, one of the possible improvements is to apply the stream processing to the images presented to the individuals, currently presented in the RAW format. Using stream processing could increase the user experience and the quality of visualization of the pictures.

Another important task would deem to remove the trusted API for data transformations and delegate the responsibility to the robots, wherein the business logic is written by smart contracts and robots

perform transactions on their own. As seen in the previous sections, the usage of central API for data conversions and transformations serves the sole purpose of simplifying experiments. In a real-world scenario, trusted authority is a subject to be a single point of failure of the system.

After that, it is crucial to mitigate risks regarding privacy breaches. Rather than promising that personal data can always be protected, it is more satisfying to communicate realistic expectations of the extent to which users' data can be protected. Furthermore, the possibility of notifying individuals (e.g. by email) regarding their private data access should be deemed future work. Also, performing the evaluations regarding the anonymity of the hashes stored on-chain, as these have special attention under GDPR scope in [14, p. 31]. Since Blockchain performs the user's identity control, it could have a likelihood to be discovered due to a collision resistance or pattern analysis attack. A possible improvement of anonymity can be achieved by using salted hash or peppered hash, [14, p. 31].

Moreover, other sensors may be more invasive to users' privacy, e.g., audio and emotional state. The proposed architecture could be extended to include these privacy-sensitive data. In such a case, the hashes of other data types should be evaluated on security. We believe that the matter of HRI privacy is a complex issue and requires more work to be done and that our research will serve as a ground point to further investigations.

# Bibliography


[1] F. Hegel, C. Muhl, B. Wrede, M. Hielscher-Fastabend, and G. Sagerer, "Understanding social robots," in *2009 Second International Conferences on Advances in Computer-Human Interactions*. IEEE, 2009, pp. 169–174.

[2] M. M. Z. Tanim, "How does passive rfid works, briefly explained." *Researchgate*, 01 2016.

[3] E. Cucco, M. Fisher, L. Dennis, C. Dixon, M. Webster, B. Broecker, R. Williams, J. Collenette, K. Atkinson, and K. Tuyls, "Towards robots for social engagement," *Department of Computer Science, University of Liverpool, UK*, 2017.

[4] C. Lutz and A. Tamò, "Privacy and healthcare robots–an ant analysis," in *We Robot 2016: the Fifth Annual Conference on Legal and Policy Issues relating to Robotics*, 2016.

[5] M. Dziergwa, P. Kaczmarek, and J. Kedzierski, "Rgb-d sensors in social robotics," *Journal of Automation Mobile Robotics and Intelligent Systems*, vol. 9, 2015.

[6] F. Hegel, M. Lohse, A. Swadzba, S. Wachsmuth, K. Rohlfing, and B. Wrede, "Classes of applications for social robots: A user study," in *RO-MAN 2007-The 16th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2007, pp. 938–943.

[7] F. Alonso-Martín and M. A. Salichs, "Integration of a voice recognition system in a social robot," *Cybernetics and Systems: An International Journal*, vol. 42, no. 4, pp. 215–245, 2011.

[8] J. Sequeira and D. Gameiro, "A probabilistic approach to RFID-based localization for human-robot interaction in social robotics," *Electronics*, vol. 6, no. 2, p. 32, 2017.

[9] J. Parviainen, T. Turja, and L. Van Aerschot, "Social robots and human touch in care: The perceived usefulness of robot assistance among healthcare professionals," in *Social Robots: Technological, Societal and Ethical Aspects of Human-Robot Interaction*. Springer, 2019, pp. 187–204.

[10] J. Vitale, M. Tonkin, S. Herse, S. Ojha, J. Clark, M.-A. Williams, X. Wang, and W. Judge, "Be more transparent and users will like you: A robot privacy and user experience design experiment," in

*Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 379–387.

[11] A. Richert, S. Müller, S. Schröder, and S. Jeschke, "Anthropomorphism in social robotics: empirical results on human–robot interaction in hybrid production workplaces," *AI & SOCIETY*, vol. 33, no. 3, pp. 413–424, 2018.

[12] U. Pagallo, "Robots in the cloud with privacy: A new threat to data protection?" *Computer Law & Security Review*, vol. 29, no. 5, pp. 501–508, 2013.

[13] M. Finck, "Blockchains and data protection in the european union," *Eur. Data Prot. L. Rev.*, vol. 4, p. 17, 2018.

[14] ——, *Blockchain and the General Data Protection Regulation: Can Distributed Ledgers be Squared with European Data Protection Law?: Study*. European Parliament, 2019.

[15] R. Jay, W. Malcolm, E. Parry, L. Townsend, and A. Bapat, *Guide to the General Data Protection Regulation (GDPR)*. Sweet & Maxwell, 2017.

[16] F. Pasquale, *The black box society*. Harvard University Press, 2015.

[17] H. Draper and T. Sorell, "Ethical values and social care robots for older people: an international qualitative study," *Ethics and Information Technology*, vol. 19, no. 1, pp. 49–68, 2017.

[18] M. K. Lee, K. P. Tang, J. Forlizzi, and S. Kiesler, "Understanding users' perception of privacy in human-robot interaction," in *2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2011, pp. 181–182.

[19] M. R. Calo, "12 robots and privacy," *Robot ethics: The ethical and social implications of robotics*, p. 187, 2011.

[20] M. Rueben and W. D. Smart, "Privacy in human-robot interaction: Survey and future work," *We robot*, 2016.

[21] V. González-Pacheco, Á. Castro-González, M. Malfaz, and M. A. Salichs, "Human robot interaction in the monarch project," in *Proc. 13th Workshop Robocity2030*, 2015, pp. 1–8.

[22] H. Felzmann, E. Fosch-Villaronga, C. Lutz, and A. Tamo-Larrieux, "Robots and transparency: the multiple dimensions of transparency in the context of robot technologies," *IEEE Robotics & Automation Magazine*, vol. 26, no. 2, pp. 71–78, 2019.

[23] M. Rueben, A. M. Aroyo, C. Lutz, J. Schmölz, P. Van Cleynenbreugel, A. Corti, S. Agrawal, and W. D. Smart, "Themes and research directions in privacy-sensitive robotics," in *2018 IEEE workshop on advanced robotics and its social impacts (ARSO)*. IEEE, 2018, pp. 77–84.

[24] J. Ball, “Nsa’s prism surveillance program: how it works and what it can do,” *The Guardian*, vol. 8, 2013.

[25] V. Goel, “Facebook tinkers with users’ emotions in news feed experiment, stirring outcry,” *The New York Times*, vol. 29, 2014.

[26] W. Van Der Aalst, “Data science in action,” in *Process mining*. Springer, 2016, pp. 3–23.

[27] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A design science research methodology for information systems research,” *Journal of management information systems*, vol. 24, no. 3, pp. 45–77, 2007.

[28] F. Kaplan, “Everyday robotics: robots as everyday objects,” in *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies*, 2005, pp. 59–64.

[29] N. Epley, A. Waytz, and J. T. Cacioppo, “On seeing human: a three-factor theory of anthropomorphism.” *Psychological review*, vol. 114, no. 4, p. 864, 2007.

[30] S. D. Warren and L. D. Brandeis, “The right to privacy,” *Harvard law review*, pp. 193–220, 1890.

[31] J. DeCew, “Privacy–from stanford encyclopedia of philosophy (first published tue may 14, 2002; substantive revision mon sep 18, 2006), 2006.”

[32] E. J. Bloustein, “Privacy as an aspect of human dignity: An answer to dean prosser,” *NYUL rev.*, vol. 39, p. 962, 1964.

[33] J. Rachels, “Why privacy is important,” *Philosophy & Public Affairs*, pp. 323–333, 1975.

[34] R. E. Gavison, “privacy: Legal aspects,” 1987.

[35] N. Tschöpe, J. E. Reiser, and M. Oehl, “Exploring the uncanny valley effect in social robotics,” in *Proceedings of the Companion of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, 2017, pp. 307–308.

[36] C. Lutz and A. Tamó-Larrieux, “The robot privacy paradox: Understanding how privacy concerns shape intentions to use social robots,” 2020.

[37] T. Laurence, *Blockchain for dummies*. John Wiley & Sons, 2019.

[38] M. S. Ferdous, M. J. M. Chowdhury, M. A. Hoque, and A. Colman, “Blockchain consensuses algorithms: A survey,” *arXiv preprint arXiv:2001.07091*, 2020.

[39] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” Manubot, Tech. Rep., 2019.

[40] S. Sayeed and H. Marco-Gisbert, “Assessing blockchain consensus and security mechanisms against the 51% attack,” *Applied Sciences*, vol. 9, no. 9, p. 1788, 2019.

[41] V. Buterin, “A next-generation smart contract and decentralized application platform. ethereum white paper,” *Ethereum Project White Paper*, 2014.

[42] D. Vujičić, D. Jagodić, and S. Ranđić, “Blockchain technology, bitcoin, and ethereum: A brief overview,” in *2018 17th international symposium infoteh-jahorina (infoteh)*. IEEE, 2018, pp. 1–6.

[43] I. Grigg, “Eos-an introduction,” *White paper. https://whitepaperdatabase.com/eos-whitepaper*, 2017.

[44] F. Schär, “Blockchain forks: A formal classification framework and persistency analysis,” 2020.

[45] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich *et al.*, “Hyperledger fabric: a distributed operating system for permissioned blockchains,” in *Proceedings of the thirteenth EuroSys conference*, 2018, pp. 1–15.

[46] J. Kwon and E. Buchman, “A network of distributed ledgers,” *Cosmos, dated*, pp. 1–41, 2018.

[47] M. Correia, “From byzantine consensus to blockchain consensus,” *Essentials of Blockchain Technology (2019)*, vol. 41, 2019.

[48] Y. Amoussou-Guenou, A. Del Pozzo, M. Potop-Butucaru, and S. Tucci-Piergiovanni, “On fairness in committee-based blockchains,” *arXiv preprint arXiv:1910.09786*, 2019.

[49] A. Wahab and W. Mehmood, “Survey of consensus protocols,” *arXiv preprint arXiv:1810.03357*, 2018.

[50] I. S. Michael Ault, “Why new off-chain storage is required for blockchains,” *NIST National Institute of Standards and Technology*, 2019.

[51] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[52] D. Johnson, A. Menezes, and S. Vanstone, “The elliptic curve digital signature algorithm (ecdsa),” *International journal of information security*, vol. 1, no. 1, pp. 36–63, 2001.

[53] J. L. Carter and M. N. Wegman, “Universal classes of hash functions,” *Journal of computer and system sciences*, vol. 18, no. 2, pp. 143–154, 1979.

[54] S. Ajami and A. Rajabzadeh, “Radio frequency identification (rfid) technology and patient safety,” *Journal of research in medical sciences: the official journal of Isfahan University of Medical Sciences*, vol. 18, no. 9, p. 809, 2013.

[55] R. Want, “An introduction to rfid technology,” *IEEE pervasive computing*, vol. 5, no. 1, pp. 25–33, 2006.

[56] G. D. P. Regulation, “General data protection regulation (gdpr),” *Intersoft Consulting, Accessed in October 24*, vol. 1, 2018.

[57] S. Aldossary, W. Allen *et al.*, “Data security, privacy, availability and integrity in cloud computing: issues and current solutions,” *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 4, pp. 485–498, 2016.

[58] A. Poikola, K. Kuikkaniemi, and H. Honko, “Mydata a nordic model for human-centered personal data management and processing,” *Finnish Ministry of Transport and Communications*, 2015.

[59] G. Zyskind, O. Nathan *et al.*, “Decentralizing privacy: Using blockchain to protect personal data,” in *2015 IEEE Security and Privacy Workshops*. IEEE, 2015, pp. 180–184.

[60] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, “Medrec: Using blockchain for medical data access and permission management,” in *2016 2nd International Conference on Open and Big Data (OBD)*. IEEE, 2016, pp. 25–30.

[61] N. B. Truong, K. Sun, G. M. Lee, and Y. Guo, “Gdpr-compliant personal data management: A blockchain-based solution,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1746–1761, 2019.

[62] B. Faber, G. C. Michelet, N. Weidmann, R. R. Mukkamala, and R. Vatrapu, “Bpdims: a blockchain-based personal data and identity management system,” in *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 2019.

[63] E. C. Ferrer, O. Rudovic, T. Hardjono, and A. Pentland, “Robochain: A secure data-sharing framework for human-robot interaction,” *arXiv preprint arXiv:1802.04480*, 2018.

[64] B. Degardin, “Blockchain for robotic event recognition,” *Covilha: Universidade da Beira Interior*, 2018.

[65] M. Fernandes and L. A. Alexandre, “Robotchain: Using tezos technology for robot event management,” *Ledger*, 2019.

[66] A. Brown, “Rise of the machines? amazon’s army of more than 100,000 warehouse robots still can’t replace humans because they lack ‘common sense’,” *Daily Mail (accessed 9 March 2019) https://www. dailymail. co. uk/sciencetech/article-5808319/Amazon-100-000-warehouse-robots-company-insists-replace-humans. html*, 2018.

[67] A. Lanko, N. Vatin, and A. Kaklauskas, “Application of rfid combined with blockchain technology in logistics of construction materials,” in *Matec Web of conferences*, vol. 170. EDP Sciences, 2018, p. 03032.

[68] K. Toyoda, P. T. Mathiopoulos, I. Sasase, and T. Ohtsuki, “A novel blockchain-based product ownership management system (poms) for anti-counterfeits in the post supply chain,” *IEEE access*, vol. 5, pp. 17 465–17 477, 2017.

[69] K. Finlow-Bates, “RFID tag one-way identification method through blockchain verification and throttling,” *Researchgate*, 2019, [Online].

[70] M. Bez, G. Fornari, and T. Vardanega, “The scalability challenge of ethereum: An initial quantitative analysis,” in *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*. IEEE, 2019, pp. 167–176.

[71] J. Benet, “Ipfs-content addressed,” *Versioned, P2P File System*, vol. 2, 2014.

[72] J. Izaguirre, J. Furgeson, and Q. Ma, “Proceedings of cse 331, data structures, department of computer science and engineering, university of notre dame, notre dame, indiana,” pp. 63–67, 2000.

[73] A. Gervais, G. O. Karame, K. Wüst, V. Glykantzis, H. Ritzdorf, and S. Capkun, “On the security and performance of proof of work blockchains,” in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 3–16.

[74] C. J. Ihrig and A. Bretz, *Full stack JavaScript development with MEAN*. Sitepoint, 2014.

[75] T. Kanda, T. Hirano, D. Eaton, and H. Ishiguro, “Person identification and interaction of social robots by using wireless tags,” in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)(Cat. No. 03CH37453)*, vol. 2. IEEE, 2003, pp. 1657–1664.

[76] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design science in information systems research,” *MIS quarterly*, pp. 75–105, 2004.

[77] H. Österle, J. Becker, U. Frank, T. Hess, D. Karagiannis, H. Krcmar, P. Loos, P. Mertens, A. Oberweis, and E. J. Sinz, “Memorandum on design-oriented information systems research,” *European Journal of Information Systems*, vol. 20, no. 1, pp. 7–10, 2011.

[78] J. Sequeira, P. Lima, A. Saffiotti, V. Gonzalez-Pacheco, and M. A. Salichs, “Monarch: Multi-robot cognitive systems operating in hospitals,” in *ICRA 2013 workshop on many robot systems*, 2013.

[79] A. Sharma, S. Mittal, and S. Mittal, “Attacks on cryptographic hash function and advances,” *IJICS*, vol. 5, no. 11, 2018.

[80] P. C. Hershey and C. B. Silio, “Procedure for detection of and response to distributed denial of service cyber attacks on complex enterprise systems,” in *2012 IEEE International Systems Conference SysCon 2012*. IEEE, 2012, pp. 1–6.

[81] M.-H. Chiu, K.-P. Yang, R. Meyer, and T. Kidder, “Analysis of a man-in-the-middle experiment with wireshark,” in *Proceedings of the International Conference on Security and Management (SAM)*. Citeseer, 2011, p. 1.

[82] Y. Perwej, N. Akhtar, and F. Parwej, “A technological perspective of blockchain security,” *International Journal of Recent Scientific Research*, vol. 9, no. 11, 2018.

[83] M. Garriga, M. Arias, and A. De Renzis, “Blockchain and cryptocurrency: A comparative framework of the main architectural drivers,” *arXiv preprint arXiv:1812.08806*, 2018.

[84] Y. Huang, H. Wang, L. Wu, G. Tyson, X. Luo, R. Zhang, X. Liu, G. Huang, and X. Jiang, “Characterizing eosio blockchain,” *arXiv preprint arXiv:2002.05369*, 2020.

[85] L. Quan, L. Wu, and H. Wang, “Evulhunter: Detecting fake transfer vulnerabilities for eosio’s smart contracts at webassembly-level,” *arXiv preprint arXiv:1906.10362*, 2019.

[86] Aitheon, “Aitheon whitepaper,” 2018, [Online].

[87] J. M. Chase, “A permissioned implementation of ethereum,” *Accessed Feb*, vol. 20, 2018.

[88] V. Strobel, E. Castelló Ferrer, and M. Dorigo, “Managing byzantine robots via blockchain technology in a swarm robotics collective decision making scenario,” 2018.

[89] R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, “A survey on blockchain interoperability: Past, present, and future trends,” *arXiv preprint arXiv:2005.14282*, 2020.