IoT

MDPI

Article

A Reference Design Model to Manage Consent in Data Subjects-Centered Internet of Things Devices

Pankaj Khatiwada [1,*], Bian Yang [1], Jia-Chun Lin [1], Godfrey Mugurusi [2] and Stian Underbekken [3]

[1] Department of Information Security and Communication Technology (IIK), Norwegian University of Science and Technology (NTNU), 7034 Trondheim, Norway; bian.yang@ntnu.no (B.Y.); jia-chun.lin@ntnu.no (J.-C.L.)
[2] Department of Industrial Economics and Technology Management (IØT), Norwegian University of Science and Technology (NTNU), 7034 Trondheim, Norway
[3] IKOMM AS, 2624 Lillehammer, Norway; stian.undbekken@ikomm.no
[*] Correspondence: pankaj.khatiwada@ntnu.no

Abstract: Internet of Things (IoT) devices have changed how billions of people in the world connect and interact with each other. But, as more people use IoT devices, many questions arise about how these devices handle private data and whether they properly ask for permission when using it. Due to information privacy regulations such as the EU's General Data Protection Regulation (GDPR), which requires companies to seek permission from data subjects (DS) before using their data, it is crucial for IoT companies to obtain this permission correctly. However, this can be really challenging in the IoT world because people often find it difficult to interact with and manage multiple IoT devices under their control. Also, the rules about privacy are not always clear. As such, this paper proposes a new model to improve how consent is managed in the world of IoT. The model seeks to minimize "consent fatigue" (when people get tired of always being asked for permission) and give DS more control over how their data are shared. This includes having default permission settings, being able to compare similar devices, and, in the future, using AI to give personalized advice. The model allows users to easily review and change their IoT device permissions if previous conditions are not met. It also emphasizes the need for easily understandable privacy rules, clear communication with users, and robust tracking of consent for data usage. By using this model, companies that provide IoT services can do a better job of protecting user privacy and managing DS consent. In addition, companies can more easily comply with data protection laws and build stronger relationships with their customers.

Keywords: consent management; internet of things (IoT); data protection; privacy; GDPR; user control; transparency

check for updates

Citation: Khatiwada, P.; Yang, B.; Lin, J.-C.; Mugurusi, G.; Underbekken, S. A Reference Design Model to Manage Consent in Data Subjects-Centered Internet of Things Devices. *IoT* **2024**, *5*, 100–122. https://doi.org/10.3390/iot5010006

Academic Editor: Amiya Nayak

Received: 16 November 2023
Revised: 28 January 2024
Accepted: 1 February 2024
Published: 6 February 2024

1. Introduction

The Internet of Things (IoT) represents a vast network of interconnected smart devices. These devices are designed to autonomously organize, exchange information and resources, and respond to changes in the conditions in their environment, functioning within a structure akin to the Internet [1]. The IoT allows a wide range of applications, services, and interactions between people and objects and has had a significant impact on our world and daily life in many ways over the past decade. For example, numerous IoT-based applications are utilized in homes for energy cost savings, while in healthcare, the IoT aids in monitoring patients and the elderly more effectively. In industrial settings, the IoT is employed to streamline complex work processes, among many other applications. The number of IoT devices globally is projected to expand by roughly 12% every year, from 27 billion in 2017 to 125 billion in 2030 [2]. This highlights the limitless potential of IoT devices; however, it also underscores the challenges in managing the vast amount of data generated by their increased use. These devices collect and utilize personal data to tailor their services to the specific needs and preferences of each user they interact with [1]. As a result, there is growing concern about user privacy in the IoT domain due to the massive amount of personal information collected and exchanged in IoT environments [3]. The emergence of data protection regulations such as the General Data Protection Regulation (GDPR), California Consumer Privacy Act, etc., is aimed at giving users more control over their data and establishing new standards for how user data are handled. Personal information protection laws and regulations establish strict protection procedures for the entire life cycle of personal information, from creation to its collection, storage, processing, and disposal [2]. According to Kounoudes and Kapitsaki [2], when it comes to collecting and processing personal information, the first step is to obtain the consent of the owner, which must be obtained before any information is collected.

In plain language, consent is understood as being when one person freely agrees to the proposition or objectives of another [4]. But, in GDPR consent is considered to be one of the legal bases for data collection and processing of data subjects' (DS) data by any data controller (DC) and is considered valid only when it is a 'freely given, specific, informed, and unambiguous indication of a data subject's wishes by which he or she, by a statement or clear affirmative action, signifies agreement to the processing of personal data relating to him/her' [5]. Moreover, DS are given the ability to amend, withdraw, or revoke their consent at any point in time. While browsing the internet or installing apps, DS are prompted by a pop-up notification to provide their permission in the form of consent to collect and store information about them. In this case, DS can interact with the interface on their PC or smartphone to provide the consent and they know who the DC or organization operating them is. This consent-giving procedure is still not well communicated; many of the service providers' privacy standards are often vague and difficult to understand, leading DS to naively offer their consent [4]. This consent-giving process has become even more complicated in the IoT environment, which includes minimal interaction between DS and DC due to the lack of an interface in most IoT devices [6]. Most of the time, consumers who use IoT devices have no choice but to consent to the policies offered by the IoT provider or not use these devices at all. Furthermore, once a DS has given their consent to policies, there is no way for them to withdraw or change it. IoT manufacturers and service providers have a reputation for failing to explain how data are processed. The research conducted by the Global Privacy Enforcement Network (GPEN) found that 59% of IoT devices did not clearly describe how personal information from DS is collected, used, and shared. DS were not informed about how their personal information was stored and protected in 68% of IoT devices. More than two-thirds of the devices did not include device-specific instructions and they were unable to remove personal information from 72% of the DS devices [7].

The rapid increase in IoT devices has caused an increase in the complexity of managing consent and privacy settings, with significant challenges including consent fatigue and the need for granular control over data sharing [8]. Consent fatigue arises as users become overwhelmed with incessant data collection and usage requests from a multiplicity of devices, risking their privacy through quick approval or dismissal of these requests. To address this, a human-centered centralized consent management platform should be established with default consent or comparisons for similar devices or an AI consent assistant, and also by providing a unified interface for users to review and manipulate IoT device permissions. Such a system elucidates each device's data protocols, facilitating informed decision-making. Meanwhile, the heterogeneity of the data collected by IoT devices results in users wanting varying data sharing levels across devices, creating the daunting task of managing individual settings. A potential resolution involves enabling users to establish default privacy preferences for categorical groups of devices (e.g., home security, health monitoring, environmental monitoring), with facile modification options for individual devices. Furthermore, the use of AI algorithms could further ease the burden by discerning user preferences and suggesting tailored privacy configurations, significantly simplifying the complexity inherent in the consent management of IoT devices.

1.1. Consent Management

Consent is an innovative way to involve the DS or user in how their personal information is being collected and used [4]. The GDPR, which came into action in 2016, strictly set rules and regulations for the service providers to ensure that the personal data of EU individuals are collected, recorded, and stored properly both within the European Economic Area (EEA) and third-party countries who must obtain consent prior to the collection of personal information. This seeks to improve our understanding of DS in relation to privacy and promote their rights to regulate the legitimate collection, processing, and management of their personal data [9]. GDPR compliance in the IoT sector, and particularly in applications relating to smart health and smart homes, often requires the DC to acquire and manage relevant consents from DS, which can be difficult in some cases, as explained above. Therefore, in this paper, we seek to create a framework that enables users to provide consent to maintain and control their personal data in the IoT ecosystem, in accordance with the standards of the GDPR. Figure 1 illustrates the fundamental principles of a consent framework using a graphical representation.

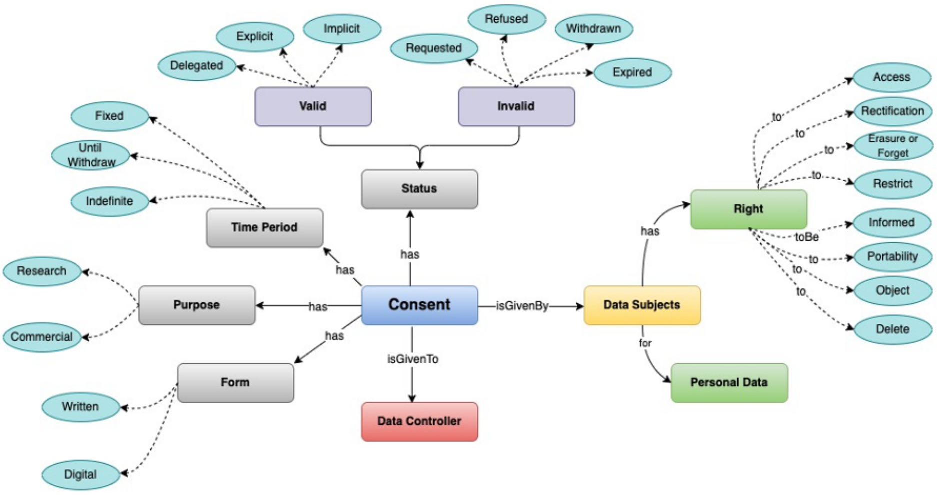

Figure 1. Concepts representing the context of consent.

1.2. Contributions

This paper makes the following contributions:

1. We propose a framework to enhance Internet of Things (IoT) consent management, aimed at reducing "consent fatigue" and empowering users with more control over their data sharing decisions.
2. The proposal of a consent tracker module that provides transparency for both involved parties by clearly indicating who is using the data (including a third party) and what data are being used.
3. Furthermore, the article utilizes this framework within a particular research project, "IoMT"; the details of which will be discussed later. This project specifically focuses on the management of consent related to data in electronic door locks.

The structure of this paper unfolds as follows. We discuss existing research relevant to our topic in Section 2, providing a comprehensive review of related work in this field. In Section 3, we then elaborate on the design elements crucial for creating a robust consent management framework suitable for a multitude of IoT devices. Further on, we introduce our enhanced framework, drawing upon insights gained from our review of the related work. Lastly, we conclude with a summary of our findings and an outline of future directions in the pursuit of building an effective consent management platform.

2. Related Work

Konstantinos Rantos et al. [10] proposed a user-centric solution for managing personal data in the IoT ecosystem in compliance with the General Data Protection Regulation (GDPR) rules. The authors addressed the challenge of applying GDPR rules to the IoT ecosystem, where users often lack understanding of how their data are collected and used.

The ADvoCATE framework by Rantos et al. [10], which includes a Consent Management Component, a Consent Notary Component, and an Intelligence Component, is designed to help users retain control over their personal data. The framework uses blockchain technology to protect the integrity and for versioning of user consents, while an intelligence component analyzes policy data to detect conflicts and provide recommendations. The study contributes to the field by offering a GDPR-compliant solution for managing personal data in the IoT ecosystem. The findings suggest that ADvoCATE can enhance data privacy and control for both data subjects and data controllers. Jaiman, V et al. [11] presented a blockchain-based consent model for data sharing, which addresses the challenge of compliance in data sharing practices. The model, implemented on the LUCE data sharing platform, offers users greater control over their data and reflects the specificity of purpose-of-use statements. The study reveals that the model could significantly reduce data collection costs. This research contributes to addressing the problem of data ownership and privacy in personal health data, a significant issue in the era of big data. However, the study acknowledges its limitations, including the assumption that the datasets are shared as a whole. The implications of the findings suggest that the proposed model could improve data sharing practices, particularly in personal health data, and potentially reduce associated costs. The findings could be applied in various domains, including genomic data sharing and other ontological solutions of health data.

Mathieu Cunche et al. [6] presented a novel framework for managing consent and information on the Internet of Things (IoT). The authors addressed the challenge of expressing and communicating consent in the IoT, proposing a system that is satisfactory both for data subjects (DS) and data controllers (DC). The framework includes both direct and indirect communication modes and is designed to minimize information fatigue while ensuring that data are not collected without proper consent. The key contribution of the study is the development of a prototype that meets these requirements and offers a potential solution to privacy issues in the IoT. However, the authors acknowledge that the prototype is not production-ready and that its effectiveness is based on organizational and regulatory measures. The findings have implications for improving the design of privacy policies in the IoT, potentially improving the awareness and decision making of DS. Ricardo et al. [12] present a novel framework for managing informed consent in the IoT. The author aims to address the limitations of the current End User License Agreements (EULAs) approach, which is expected to become more critical with the increasing complexity of IoT applications. The proposed framework uses usage control policies to regulate access to personal data, which can be tailored to the specific features of the user and the context. The key findings include the development of a privacy-preserving authentication solution, a model-based security toolkit for policy rule enforcement, and a user-centric mechanism for privacy policy management. This research contributes to the field by proposing a more sophisticated model of informed consent for the IoT, thus enhancing user privacy and data protection. However, the authors acknowledge limitations such as the need for a further more detailed analysis of identity attributes and the selection of presentation token policies for different service providers. The findings have significant implications for the development of IoT applications, providing a framework for managing informed consent and enhancing user privacy and data protection.

Harshvardhan et al. [13] presented GConsent, an innovative OWL2-DL ontology to represent consent information in compliance with the General Data Protection Regulation (GDPR). The authors addressed the limitations of existing work that primarily focuses on the ‘given’ aspect of consent. GConsent is designed to handle various states of consent, including ‘given’, ‘not given’, ‘refused’, and ‘withdrawn’, and also models events like delegation or associations with third parties. However, the authors acknowledge that GConsent does not provide information on or model compliance regarding various GDPR obligations and does not require the specification of legal justification for the purposes of processing. Despite these limitations, the study makes a significant contribution to the field of data protection and privacy, offering a more comprehensive consent model that could be instrumental in GDPR compliance. Merlec et al. [14] presented a novel framework for managing consent in compliance with the General Data Protection Regulation (GDPR) rules. The authors proposed a smart contract-based dynamic consent management system architecture that enables users to control their consent for organizations to collect and use their personal data throughout the data lifecycle. The system, backed by blockchain technology, achieved high transaction throughputs and low latencies with moderate resource consumption and utilization of the storage network bandwidth. However, the authors acknowledge that the experiments were performed on a single server with nodes running in containers, indicating a need for future scalability analyses using cluster or cloud computing services. Despite these limitations, the study makes a significant contribution to the field of data protection and privacy, offering a more comprehensive consent model that could be instrumental in GDPR compliance. The findings have significant implications for improving privacy policy design, potentially improving user control over personal data.

Alhajri et al. [15] presented a novel framework for managing consent in the sharing of fitness data. The authors proposed a blockchain-based dynamic consent mechanism to mitigate privacy concerns associated with fitness tracker privacy policies. The key findings include the successful design of a system architecture, requirements specification, and a formal proof model that closely follows the valid consent criteria within GDPR. The proposed system was evaluated using the Security Modeling Framework (SeMF) tool and identified blockchain assumptions, ensuring that it does not overwhelm the capabilities of a typical fitness tracker's functionality. However, the authors acknowledge limitations, including the need for in-depth performance and scalability analyses for a variable number of nodes and batch sizes. Despite these limitations, the study makes a significant contribution to the field of data protection and privacy, offering a more comprehensive model of consent that could be instrumental in GDPR compliance. The findings have significant implications for enhancing privacy policy design in the sharing of fitness data. Kounoudes et al. [2] addressed the gap in user-centric privacy protection on the Internet of Things (IoT). The authors conducted a systematic quantitative literature review to identify GDPR-compliant characteristics that can improve user privacy protection in the IoT. The study reveals that machine learning techniques are extensively used for this purpose, but more research is needed to exploit these techniques to specify user preferences based on past activity. The authors also identified open research challenges for studying privacy the in IoT from the end-user's perspective. The findings of this study are significant for researchers and practitioners in the field, providing them with a set of GDPR-compliant characteristics that can be incorporated into their systems or platforms to better protect users and comply with GDPR requirements. However, the study acknowledges that each paper can address more than one characteristic or challenge, which is a limitation of the total numbers displayed in their summary table.

The common research gap identified across all the reviewed literature is the need for more comprehensive and user-centric privacy protection mechanisms in the IoT domain. This includes the need for more advanced models of informed consent, better representation models to describe user consent, and more efficient privacy protection mechanisms that do not compromise the functionality of IoT devices. Additionally, more research is needed on the interaction of humans with their devices, through the provision of user interfaces, where users can express their privacy preferences. Also, there is a pressing need for the provision of appropriate rules and policies for data access in cloud services, the development of automated compliance-checking engines, and the optimization of privacy configurations based on the desired privacy level and utility preference of users. Therefore, our framework offers significant improvements compared to the models discussed above, focusing on a user-centric design. It prioritizes user privacy, enabling more control through predefined preferences and detailed consent options. The framework enhances flexibility, allowing users to easily modify or withdraw consent, and smoothly handles consent expiration and renewal. It reduces consent fatigue with predefined recommendations and unified interface for multiple IoT devices. In summary, the framework excels in user experience, flexibility, transparency, and legal compliance, marking a significant advancement in consent management.

3. Design Factors for Consent Framework

3.1. IoT Discovery

It is a key necessity in any IoT ecosystem that things or devices, as well as their resources, information, attributes, and capabilities, can be found and discovered. In order for DS to be able to identify the devices and understand what kinds of data are being gathered by those devices, the devices must be easily discoverable. Bröring et al. [16], in their article, discuss and present the four different categories of IoT discovery technologies. Every discovery interaction pattern has two essential roles: the client and the thing (IoT device). The client can be a user or DS operating the application to discover things, and the things can be an Internet- or a web-enabled device. The four different categories explained are: (a) Searching IoT Around Me, (b) Searching IoT on My Network, (c) Searching IoT in directories, (d) Accessing IoT Metadata. Application developers can adopt any one of these categories to decide which discovery technique for IoT to use according to their needs. Khalil et al. [17] also present various ways for the resource discovery in IoT environments and discovery protocols. He discusses different discovery approaches such as: (a) Distributed and centralized discovery, (b) semantic-based discovery, (c) search engine for resource discovery, (d) edge-centric distributed discovery. He concluded that resource discovery is a fundamental notion in the IoT environment, since it allows for more intelligent interactions and communications between different IoT devices and users. The ultimate goal of resource discovery in IoT environments is to locate devices and services that are of interest to the entity that initiated the request. A large amount of research is being conducted on a regular basis to find the most efficient method of discovering IoT devices, and we may choose from a variety of different techniques depending on our needs. It is also worth mentioning that there are already open-source home automation platforms available that we may use to discover IoT devices in our homes and surroundings. The most popular in use are (a) OpenHAB–compatible with over 1500 devices; (b) Domoticz–a great tool for Linux users; (c) Home Assistant–powerful privacy protection tools; (d) Calaos–powerful mobile apps; (e) OpenMotics–Google apps integration; (f) HomeGenie–customization tools; and (g) PiDome–advanced rules editor [18]. We can modify these platforms and configure them according to our needs to achieve our goal. We must also take into account the fact that every IoT service provider provides a mechanism for IoT devices through which they broadcast messages to be discovered within the IoT ecosystem.

3.2. IoT App/Web User Application

The applications in the IoT app/web user are required to provide an interface that satisfies the purposes of both the DS and the DC. The app should be able to be accessed on iOS or Android handheld devices such as smartphones and tablets, and the web app should be able to be accessed from any browser on PCs and laptops. The app can be mostly used by DS for IoT device discovery, to set privacy preferences, to provide and manage consent, and to use services provided by IoT devices. The app may come in various forms: (a) standalone app to fulfilll the purpose for a specific device (custom developed app); (b) open source app that could integrate many IoT devices on its platform (OpenHAB, Home Assistant); (c) propriety app that could integrate many IoT devices on its platform (Smart Things, Apple Home Kit); (d) apps that are being developed and provided by the IoT device provider itself (Philips Hue, Airthings Wave). The web application is mainly used by DCs to manage and store consents. It is combined with the backend server and database for the business logic, consent storage, and logs of the consent history provided by DS. Some providers may also provide a web app for DS, but it is replica of the app where they could manage their device consents.

3.3. Privacy Preferences and Policies

Today, a wide range of IoT devices are available from various providers. These providers gather, retain, and handle DS data in accordance with their privacy preferences to deliver their services. DS can choose a set of privacy settings and provide consent to better control how their personal data are handled and maintained by the service provider. Also, many services provided are context dependent, meaning that the type of service given is dependent on the user context (e.g., time, location, and so on). Since users may have varied privacy preferences depending on the situation, this complicates the meaning of the term "privacy preferences" (e.g., working hour, free time). A user can set up distinct privacy choices for each potential circumstance in order to give greater fine-grained controlled consent for their data, which could also be termed as a privacy model. However, since the user's context changes so frequently, it might be a time-consuming and difficult operation to obtain consent for every little change in a DS' privacy preferences. Hence, we can use machine learning and AI techniques which can predict new privacy preferences for a given scenario and set consent for DS [19,20].

Privacy policies are information or statements presented to DS by the DCs of organizations or companies that describe DS about their privacy practices. Due to the GDPR and the California Consumer Privacy Act (CCPA), organizations are increasingly being forced to disclose their data practices in privacy policies in plain terms. Privacy policies must be put in place by organizations or companies so that users are aware of their privacy rules and preferences before a DS selects or makes a decision. IoT devices have access to very sensitive data about their users, increasing the difficulty of enforcing privacy policies. Various initiatives have been taken to make privacy policies clearer in the wake of these issues.

3.4. Device Registration

As the number of Internet of Things (IoT) devices continues to grow, interoperability has become increasingly difficult due to the wide variety of protocols, interfaces, and hardware parameters used by devices, making it difficult to maintain effective communication between various devices, platforms, and users. Every IoT device that is used by DS should be able to be detected and registered on the platform so that DS can accurately define and track the consent given for each IoT device. DS can register the device in two ways, either manually or automatically. Manual registration is a simplified process and can be carried out by any DS who has some literacy towards technology for a few devices. But, as the number of devices grows, it becomes more complicated to register devices, and there is a need for the automatic registration of these devices. Using semantic ontology-based IoT device registration middleware, Wei et al. [21] developed an approach to automatically register IoT devices (IDRM). Consistent device information may be provided upwards, while heterogeneity can be shielded below using this technology. Device information can be automatically recorded in IDRM's device information database as an ontology file if available device triples can be retrieved from the banner of the application protocol. The device's ontology files can be retrieved by following the principles of information extraction if device vendor users submit device information-related files.

3.5. Consent Collection

Consent is collected from DS for the lawful processing of their sensitive personal data to be used in a specific way. There are several different methods available for obtaining consent from the DS, including written consent, email consent, verbal consent, and digital consent. Written consent is the oldest form of the consent collection method where DS are provided with a letter that contains the information about the project and the purpose of data collection. Usually, clinical trials and various research task studies use the written consent method. The DS reads the information and provides the written consent for their data collection and a DU or DC records the copy of the consent agreement. Verbal consent is also a legal way to collect consent from DS; Recital 32 Articles of the GDPR do not specifically prohibit providing oral consent. It is permissible under specific circumstances because there are many business and customer service activities that are performed over the telephone; therefore, client service representatives should be able to handle their accounts and process customer data over the phone without requiring major changes to current methods.

3.6. Consent Management

Consent management is a crucial aspect of any IoT consent framework, especially in the context of data privacy and compliance with regulations such as the General Data Protection Regulation (GDPR) or the California Consumer Privacy Act (CCPA). Consent management involves obtaining, storing, and managing user consent for the processing of their personal data [22,23]. Here are some key elements and considerations for consent management in a consent framework:

(a) Explicit Consent: Ensure that consent is obtained explicitly and that users have a clear understanding of what they consent to. Consent should be specific, informed, and unambiguous. Users should be able to easily give and withdraw consent.

(b) Transparency: Provide users with transparent information on the purposes for which their data will be processed, who will process it, and any third parties involved. This information should be easily accessible and understandable. Also, we should clarify the consent process which involves informing users comprehensively about the various reasons their data might be utilized. It is essential to provide clear examples to help users understand that giving consent for one purpose does not automatically mean they agree to all uses. For instance, if a user consents to their data being used for statistical analysis, this does not imply they have also agreed to have their data used for advertising purposes. This distinction is crucial in ensuring users are fully aware of what they consent to and helps maintain transparency and trust in the consent process. The goal is to ensure that users have a clear understanding and are making informed decisions about the use of their data.

(c) Granularity: Offer granular options for consent, allowing users to choose the types of data processing to which they agree. This means providing separate consent choices for different purposes rather than bundling them together.

(d) Age verification: Implement mechanisms to verify the age of users, especially if dealing with data from minors. In many jurisdictions, parental or guardian consent is required for data processing involving minors.

(e) Consent Records: Maintain detailed consent records, including the time, date, and version of consent provided by each user. These records are essential for demonstrating compliance in the event of regulatory audits.

(f) Consent Renewal and Expiry: Set clear periods in which consent remains valid. After the specified period, seek the renewed consent of the users. Additionally, allow users to revoke their consent at any time.

(g) User Interface: Design a user-friendly interface for obtaining and managing consent. The process should be intuitive, straightforward, and accessible across different devices.

(h) Withdrawal of Consent: Make it easy for users to withdraw their consent at any time. Provide clear instructions on how to do so and ensure that the withdrawal process is as simple as giving consent.

(i) Communication: Regularly communicate with users about their consent choices and any updates or changes to the consent framework or data processing practices.

(j) Security and Data Protection: Ensure that user consent data are stored securely and protected from unauthorized access or breaches.

(k) Integration with Systems: Integrate consent management functionalities with other systems that process personal data to ensure that data processing activities are aligned with user consent preferences.

(l) Auditing and Compliance Monitoring: Regularly audit the consent management process to ensure compliance with the relevant regulations. Continuously monitor consent-related activities and address any issues or non-compliance promptly.

Also, we should consider that the specific requirements for consent management may vary on the basis of the data protection regulations which are applicable in the region and the nature of the organization's data processing activities. Therefore, it is always good to have legal experts to ensure that the consent framework is aligned with the relevant laws and regulations.

3.7. Provenance Manager

In an IoT consent framework, a Provenance Manager [24] refers to a component or system responsible for capturing, recording, and managing the history and lineage of user consent and related data processing activities. It ensures that there is a transparent and traceable record of how and when user consent was obtained, as well as any subsequent changes or updates to that consent. The Provenance Manager is an essential part of a consent framework as it helps organizations adhere to data protection regulations, such as GDPR and other privacy laws. It allows organizations to demonstrate compliance with these regulations by maintaining an auditable trail of user consent and data processing activities. Here are the key functions and roles of a Provenance Manager in a consent framework:

(a) Consent History: The Provenance Manager maintains a comprehensive log of all consent interactions with users. It records the time, date, and specific consent choices made by each user, along with any supporting metadata.

(b) Consent Versions: As consent agreements may evolve over time due to changes in policies or user preferences, the Provenance Manager keeps track of different consent versions. This enables organizations to show the progression of consent from initial collection to subsequent updates.

(c) Consent Lifecycle Management: It manages the entire lifecycle of consent, from the initial collection to the revocation or expiration of consent. This ensures that organizations honor user preferences throughout the data processing journey.

(d) Data Processing History: In addition to consent, the Provenance Manager tracks the history of data processing activities related to each user's consent. This includes information on how and where the data were used, accessed, or shared.

(e) Consent Revocation and Expiry: When a user withdraws their consent or when consent expires after a defined period, the Provenance Manager updates the records accordingly to reflect the current status of consent for each user.

(f) Third-Party Consent Management: If data is shared with third-party processors based on user consent, the Provenance Manager ensures that these third parties have valid consent for data processing.

(g) Audit and Compliance: The Provenance Manager provides a reliable source of information for internal and external audits. It allows organizations to demonstrate compliance with data protection regulations by showing a history of user consent and data processing activities.

(h) Data Access and Deletion Requests: When users exercise their rights to access their personal data or request its deletion, the Provenance Manager facilitates the retrieval of consent history and associated data processing information to fulfill these requests.

(i) Security and Access Control: As the Provenance Manager deals with sensitive consent-related information, it must implement robust security measures to protect the integrity and confidentiality of the data.

(j) Integration with the consent management system: The Provenance Manager should integrate seamlessly with the broader consent management system to ensure that consent-related data is consistently and accurately recorded.

By maintaining a reliable record of consent and data processing activities, the Provenance Manager improves transparency, accountability, and user trust within the consent framework. It plays a critical role in meeting the legal and ethical requirements of data privacy regulations while enabling organizations to build positive relationships with their users.

4. The Case of Internet Of my Things (IOmT) Research Project

The IOmT project [8] aims to develop a privacy-enabled platform that seamlessly integrates privately owned technology like electronic locks and tablets with municipal health and care services. The platform will be developed by IKOMM AS and connect private devices to municipal systems securely following legal and privacy requirements. Eidsiva Bredbånd AS will develop an IoT platform to interact with the main integration platform. Research partners including NTNU and Innlandet University College will conduct key research activities to translate legal and privacy regulations into the platform's technical design, develop standards for workflow integration between all partner systems, and analyze expected benefits for end users as well as municipal and corporate service providers. Lillehammer Municipality will provide access to real users and employees to test the platform. The project brings together partners from industry, academia, and the public sector in Norway's Innlandet region. Expected results include more sustainable and user-friendly welfare technologies, reduced costs for municipalities through optimizing investments, improved quality and accessibility of health and care services, and strengthened regional industry competitiveness and collaboration in the digital health sector. Overall, the project aims to proactively address the increasing need for the digital renewal of public health services and the development of sustainable care models considering demographic changes such as aging populations in the Innlandet region. So, in this paper, we investigate the development of an improved consent management framework to perform the research project task, addressing the privacy challenges inherent in the technical design of the platform. The emphasis is on introducing a framework tailored for effectively managing consent in Data Subjects (DS) Centered Internet of Things (IoT) Devices.

5. Proposed Model Description and Result

In this section, we will provide a detailed explanation of our consent framework model, including algorithmic details. Figure 2 illustrates the foundational components of our model.

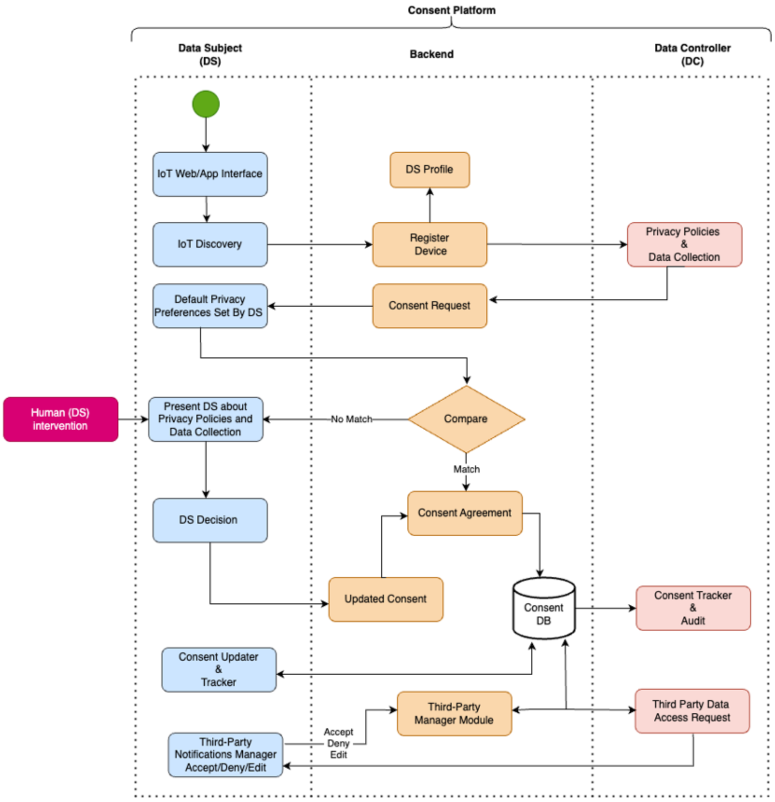

Figure 2. Consent framework.

Entities for the Model

Table 1 shows the description and notation used in the consent process:

- DS (Data Subject): Refers to an individual whose personal data is being processed or collected.

- DC (Data Controller): Refers to the entity or organization that determines the purposes and means of processing personal data.
- D (IoT Device): Represents an Internet of Things device, which is a physical object connected to the Internet that is capable of collecting and transmitting data.
- DP (DS Profile): Describes the characteristics, preferences, and attributes of a data subject.
- $\mathrm{PP_{DS}}$, $\mathrm{PP_{DC}}$ (Privacy Preference of DS, Privacy Policies DC): Represents the privacy preferences of a data subject and the privacy policies of a data controller, respectively.
- $\mathrm{D_{type\ i:}}$ Represents a diverse range of data generated by various IoT devices operating in different environments. When providing consent, the data subject (DS) has the option to selectively choose the specific types of data for which they wish to grant consent, based on their default privacy preferences. If the data controller (DC) requests access to data that falls outside the selected nature of data, the DS must intervene and approve the consent before the DC can commence data collection.
- PI (Personal Identifier): Represents a unique identifier associated with an individual, such as a name, email address, or identification number.
- CR, $\mathrm{CR_{id}}$ (Consent Request, Consent Request ID): Refers to a request made to a data subject by a data controller to obtain consent for the processing of personal data. CRid is the unique identifier for a specific consent request.
- TP, P, CS (Time Period, Purpose, Consent Status): These variables describe different aspects of consent. TP refers to the time period during which consent is valid, P represents the purpose for which consent is being requested, and CS indicates the status of the consent (e.g., granted, revoked, expired).
- R (Region): Represents a geographic region or jurisdiction, which may have specific regulations or laws related to data protection and privacy.

Table 1. Notation and Description.

Notation	Description
DS, DC	Data Subject, Data Controller
D	IoT Device
DP	DS Profile
$\mathrm{D_{type\ i}}$	Data Type
$\mathrm{PP_{DS}}$, $\mathrm{PP_{DC}}$	Privacy Preference of DS, Privacy Policies DC
PI	Personal Identifier
CR, $\mathrm{CR_{id}}$	Consent Request, Consent Request ID
TP, P, CS	Time Period, Purpose, Consent Status
R	Region

$$CR = \{CR_{id}, D_{type}, PP_{DC}, PI, TP, P, CS, R\} \tag{1}$$

where,

$$\begin{aligned}
CR_{id} &= \{CR_1, CR_2, CR_3, \dots, CRn\}, \quad \text{where } id \in (1 \le id \le n) \\
D_{type_i} &= \{D_{type1} = \text{personal}, D_{type2} = \text{medical}, D_{type3} = \text{home}, D_{type4} = \text{environment}, \\
&\quad D_{type5} = \text{industrial}, D_{type6} = \text{vehicular}, D_{type7} = \text{Other}\} \\
PI_i &= \{PI_1 = \text{personal information}, PI_2 = \text{health data}, PI_3 = \text{location}, PI_4 = \text{environment}, \\
&\quad PI_5 = \text{internet}, PI_6 = \text{device}, PI_7 = \text{other}\} \\
TP_i &= \{TP_1 = \text{start:end}, TP_2 = \text{until withdrawn}, TP_3 = \text{indefinite}\} \\
P_i &= \{P_1 = \text{Research}, P_2 = \text{Commercial}, P_3 = \text{Other}\} \\
CS_i &= \{CS_j = \text{Valid}, CS_k = \text{invalid}\}, \quad \text{where (implicit, explicit, delegated)} \in CS_j, \\
&\quad \text{(requested, refused, withdrawn, expired)} \in CS_k \\
R_i &= \{R_1 = \text{EU}, R_2 = \text{USA}, R_3 = \text{other}\} \\
PP_{DS} &= \{D_{type}, PI, TP, P, CS, R\}
\end{aligned}$$

Here, we are mainly dealing with the consent request framework and data, but privacy policies can vary according to organization; this is beyond this paper, but common policies may include Data Collection Methods (DCM), Data Usage (DU), Data Storage and Security Measures (DSSM), User Rights and Options (URO), and Data Sharing Entities (DSE) which can be in a textual form to be shown to DS.

The algorithm can be found in Algorithm 1 and an explanation of the model is given below.

Algorithm 1 Consent Process for IoT

```
procedure CONSENT_REQUEST_IOT(IoT_App, D, DP, Info, DS_PK, DS_PRK,)
DC_PK, DC_PRK, CRid, Dtype, PP$_{DC}$, TP, P, PI, DC_Sign, DS_Sign
    // Step 1: Discovery and Registration of Device //
    $d \leftarrow$ Discover(IoT_App)
    reg_status1 $\leftarrow$ Initiate$(d)$
    reg_status2 $\leftarrow$ Complete$(d, Info)$
    $dp' \leftarrow$ Update$(DP, d)$
    // Step 2: Formulating and Sending Encrypted Consent Request by DC //
    $CR \leftarrow \{CRid, Dtype, PP_{DC}, TP, P, CS, PI\}$
    $CR\_Encrypted \leftarrow$ Encrypt$(\{CR, DC\_Sign\}, DS\_PK)$
    // Step 3: Decryption of Consent Request by DS //
    $CR \leftarrow$ Decrypt$(CR\_Encrypted, DS\_PRK)$
    // Step 4: Verify Consent Request Signature by DS //
    $CR, DC\_Sign \leftarrow$ Decrypt$(CR\_Encrypted, DS\_PRK)$
    Validation_Status_CR $\leftarrow$ Verify_Signature$(CR, DC\_Sign, DC\_PK)$
    // Step 5: Review of Consent Request by DS //
    $CS\_Predefined \leftarrow$ PredefinedConsentFunction$(DS, Dtype)$
    if $CS\_Predefined \neq$ null then
        $CS\_Final \leftarrow CS\_Predefined$
    else
        $CS\_Final \leftarrow$ ManuallyGrantConsent$(DS, CR)$
    end if
    // Step 6: Generation and Encryption of Consent Token by DS //
    $CT \leftarrow$ GenerateToken$(DS)$
    $CT\_Encrypted \leftarrow$ Encrypt$(\{CT, DS\_Sign\}, DS\_PRK, DC\_PK)$
    // Step 7: Decryption and Validation of Consent Token by DC //
    $CT\_Decrypted \leftarrow$ Decrypt$(CT\_Encrypted, DC\_PRK)$
    Validation_Status_CT $\leftarrow$ Verify_Signature$(CT, DS\_Sign, DS\_PK)$
    Validation_Status $\leftarrow$ Validate$(CT\_Decrypted)$
    // Step 8: Acknowledgement of Consent Status and Enablement of IoT Device by
DC //
    $CS\_Encrypted \leftarrow$ Encrypt$(\{CS\_Final, DC\_Sign\}, DS\_PK)$
    if Validation_Status == "Success" then
        $D$.processData $\leftarrow$ true
        $D$.storeData $\leftarrow$ true
    end if
    Association_Status $\leftarrow$ Associate$(CS\_Encrypted, CRid)$
    // Step 9: Recording of Consent Status //
    Record_Status $\leftarrow$ RecordConsent$(CRid, CS\_Final, DS, DC)$
    // Step 10: DS Edits or Deletes Consent if Required //
    if DS decides to edit consent then
        Edit_Status $\leftarrow$ EditConsentFunction$(DS, CRid, newCS)$
    end if
    if DS decides to delete consent then
        Delete_Status $\leftarrow$ DeleteConsentFunction$(DS, CRid)$
    end if
end procedure
```

5.1. Registration Process

A data subject (DS) uses an IoT web/app to discover the IoT device (D) and initiates the registration process.

- D: The set of all IoT devices.
- DP: The DS's profile, containing information about the DS and their associated devices.
- *Info*: The set of all possible information that DS can provide.
- *Reg_Status*: The set of possible registration statuses (e.g., "initialized", "completed", "failed").
- IoT_App: The IoT application interface used by DS.

i. Discovery of Device: The DS discovers and selects an IoT device through the IoT web/app interface. The Discover function: $IoT\_App \rightarrow D$ describes this process, where the IoT application is taken as input, and the selected IoT device is output.

$$\text{Discover}(IoT\_App) = d \in D$$

ii. Initiation of Registration: The Initiate function: $D \rightarrow$ Reg_Status signifies the initiation of the registration process. It takes the selected device as input and outputs a registration status.

$$\text{Initiate}(d) = \text{reg\_status}_1 \in \text{Reg\_Status}$$

iii. Completion of Registration: This process is represented by the function Complete: $D \times$ Info $\rightarrow$ Reg_Status. It takes the selected device and the necessary information provided by the DS as input, and outputs a new registration status.

$$\text{Complete}(d, \text{info}) = \text{reg\_status}_2 \in \text{Reg\_Status}$$

iv. Updating DS Profile: The Update function: $DP \times D \rightarrow DP$ updates the DS profile with the registered device. It takes the existing DS profile and the selected device as the input, and outputs an updated DS profile.

$$\text{Update}(dp, d) = dp' \in DP$$

Then, the entire process for the DS can be represented as:

$$\text{Update}(\text{Complete}(\text{Initiate}(\text{Discover}(IoT\_App)), d, \text{info}), dp, d) = dp'$$

In this model, $dp' \in DP$ is the updated profile for the DS after discovering the device, initiating the registration, providing the necessary information to complete the registration, and updating the DS profile with the registered device. The 'prime' symbol denotes the updated status.

5.2. Encrypted Consent Request

The data controller (DC) sends an encrypted consent request (CR) to the data subject (DS) using secure communication channels:

- Consent Request ID (CRid): a unique identifier for the consent request;
- Device Type (Dtype): the type of IoT device for which the consent is being requested;
- Data controller's Privacy Policies (PP_{DC}) or terms and conditions associated with the device and data processing;
- Time Period (TP) for which the consent is valid;
- Purpose (P) for which the data controller (DC) is requesting consent;
- Consent Status (CS) of the current consent request;
- Categories of Personal Identifiers (PI) to be collected, represented as $\{PI_1, PI_2, PI_3, PI_4, PI_5, PI_6, PI_7\}$

To send an encrypted consent request, the data controller uses the data subject's public key. The data controller (DC) formulates a Consent Request (CR) as:

$$CR = \{CRid, Dtype, PP_{\text{DC}}, TP, P, CS, PI, R\}$$

The encrypted Consent Request (CR_Encrypted) is then:

$$CR_{\text{Encrypted}} = Encrypt(\{CR, DC_{\text{Sign}}\}, DS_{\text{PK}})$$

where:

- *Encrypt* is the encryption function.
- *CR* is the consent request.
- *DS_PK* is the Data Subject's public key.

The Encrypt function can be any encryption function such as RSA or any other secure and widely accepted public-key encryption algorithm. The particular choice of encryption function would depend on the specific security needs and capabilities of the system in question.

So, the final expression for the process would be:

$$CR_{\text{Encrypted}} = \text{Encrypt}(\{CRid, Dtype, PP_{\text{DC}}, TP, P, CS, PI, R\}, \text{DS\_PK})$$

This notation succinctly expresses the process of constructing a consent request, encrypting it using the data subject's public key, and then sending it from the data controller to the data subject.

5.3. Decryption of Consent Request

After the data controller (DC) sends the encrypted consent request ($CR_{\text{Encrypted}}$) to the data subject (DS) using the DS's public key (DS_{PK}), the DS then needs to decrypt the consent request using their own private key (DS_{PRK}).

In mathematical notation, the decryption process is represented as:

$$CR = Decrypt(CR_{\text{Encrypted}}, DS_{\text{PRK}})$$

where:

- *Decrypt* is the decryption function.
- $CR_{\text{Encrypted}}$ is the encrypted consent request sent by the DC.
- DS_{PRK} is the data subject's private key.

Verify Consent Request Signature by DS:

After the DS decrypts the *CR*, they can verify the signature using the DC's public key (DC_{PK}). This step confirms that the *CR* was indeed sent by the DC and was not altered.

$$CR, DC_{\text{Sign}} \leftarrow \text{Decrypt}(CR_{\text{Encrypted}}, DS_{\text{PRK}})$$

$$\text{Validation\_Status\_CR} = \text{Verify\_Signature}(CR, DC_{\text{Sign}}, DC_{\text{PK}})$$

After this decryption process, the data subject (DS) will have access to the consent request (CR) in its original unencrypted form. This can include details like the Consent Request ID (CRid), Device Type (Dtype), data controller's privacy policies (PP_{DC}), the Time Period (TP) for which the consent is valid, the Purpose (P) of data collection, the Consent Status (CS), and the Categories of Personal Identifiers (PI). The DS verifies the integrity and authenticity of the Consent Request (CR) and its associated information. This ensures that the DS is the only one able to view the consent request, preserving the privacy and security of the information contained within.

5.4. Consent Review

The data subject (DS) reviews the Purpose (P), Time Period (TP) and Personal Information Categories ($\mathrm{PI_i}$) for which consent is requested.

Let us define:

- $CR = \{CRid, Dtype, PP_{\mathrm{DC}}, TP, P, CS, PI\}$ where CR is the Consent Request decrypted by the DS.
- *PredefinedConsentFunction* is a function that retrieves predefined consent settings based on DS and $Dtype$.

Next, the DS checks if there are any predefined consent settings (CS) for this type of device (Dtype).

$$CS_{\text{Predefined}} = \textit{PredefinedConsentFunction}(DS, Dtype)$$

If the data subject's privacy preference ($\mathrm{PP_{DS}}$) aligns with the data controller's privacy policies ($\mathrm{PP_{DC}}$), and the Consent Request (CRid) is accepted for the Purpose (P) within a Time Period (TP) and in a specific region (R), the Consent Status (CS) is set to "granted".

This can be represented as:

$$\text{if } (PP_{\text{DS}} \cap PP_{\text{DC}} \neq \varnothing \text{ AND } CR[P, TP, R] = \text{accepted}) \text{ then } CS = \text{granted}$$

This indicates that if the intersection of the DS's privacy preference and the DC's privacy policies is not empty (meaning they have some common elements), and the Consent Request is accepted for a given Purpose, Time Period, and region, then the Consent Status is set to "granted".

The above can also be represented as:

$$\text{if } (CS_{\text{Predefined}} \neq \text{null}) \text{ then } CS_{\text{Final}} = CS_{\text{Predefined}}$$

This can be interpreted as: If the predefined consent is not null (meaning it exists), then the final Consent Status CS_{Final} is equal to the predefined consent $CS_{\text{Predefined}}$.

If the privacy preference of the data subject ($\mathrm{PP_{DS}}$) does not align with the privacy policies of the data controller ($\mathrm{PP_{DC}}$), or the Consent Request (CRid) is not accepted, then the Consent Status (CS) is set to "denied".

This can be represented as:

$$\text{if } (PP_{\text{DS}} \cap PP_{\text{DC}} = \varnothing \text{ OR } CR[P, TP, R] = \text{denied}) \text{ then } CS = \text{denied}$$

This indicates that if the intersection of DS's privacy preference and DC's privacy policies is empty (meaning they have no common elements), or if the Consent Request is denied for a given Purpose, Time Period, and region, then the Consent Status is set to "denied".

The above can also be represented as,

$$\text{if } (CS_{\text{Predefined}} = \text{null}) \text{ then } CS_{\text{Final}} = \text{ManuallyGrantConsent}(DS, CR)$$

This can be interpreted as: If the predefined consent is null (meaning it does not exist), then the final Consent Status CS_{Final} is obtained through a manual consent grant.

5.5. DS Manual Consent Review

If there are no predefined consent settings or the consent is not automatically accepted, the data subject manually grants consent.

$$\begin{aligned}&\text{if } (CS_{\text{Predefined}} = \text{null OR}\\&\quad CS_{\text{Predefined}} \neq \text{accepted})\\&\quad \text{then } CS_{\text{Final}} = \text{ManuallyGrantConsent}(DS, CR)\end{aligned}$$

Here ManuallyGrantConsent(DS, CR) is the process by which the data subject (DS) manually grants consent, based on the Consent Request (CR).

This implies: If there are no predefined consent settings ($CS_{\text{Predefined}} = \text{null}$) or the predefined consent is not automatically accepted ($CS_{\text{Predefined}} \neq \text{accepted}$), then the final consent status (CS_{Final}) is determined by manually granting consent (`ManuallyGrantConsent(DS, CR)`).

5.6. Consent Token

The data subject (DS) generates a unique consent token (CT) and encrypts it with the data controller's (DC) public key:

$$CT = \text{GenerateToken}(DS)$$

Here `GenerateToken(DS)` is a function that generates a unique consent token for the data subject (DS). Next, the data subject (DS) encrypts this token using the data controller's public key (`DC_PK`). This can be represented as:

$$CT_{\text{Encrypted}} = \text{Encrypt}(\{CT, DS_{\text{Sign}}\}, DC_{\text{PK}})$$

This implies that the $CT_{\text{Encrypted}}$ is the encrypted form of the consent token (CT), obtained using the encryption function `Encrypt` that takes the original CT, and the public key from the data controller DC_{PK}.

5.7. Data Controller (DC) Decrypts

The data controller (DC) decrypts the consent token (CT) using its private key and validates its authenticity and integrity.

The data controller uses his/her private key to decrypt the encrypted consent token:

$$CT_{\text{Decrypted}} = \text{Decrypt}(CT_{\text{Encrypted}}, DC_{\text{PRK}})$$

The data controller then validates the decrypted consent token, which we can model as a function *Validate*($CT_{\text{Decrypted}}$):

$$\text{Validation\_Status\_CT} = \text{Verify\_Signature}(CT, DS_{\text{Sign}}, DS_{\text{PK}})$$

$$\text{Validation\_Status} = \textit{Validate}(CT_{\text{Decrypted}})$$

5.8. DC Acknowledgment

The data controller acknowledges the Consent Status and enables the IoT device: First, the DC acknowledges the Consent Status by encrypting it with the DS's public key and the DC signature.

$$CS_{\text{Encrypted}} = \text{Encrypt}(\{CS_{\text{Final}}, DC_{\text{Sign}}\}, DS_{\text{PK}})$$

If the validation status is successful (*Validation_Status* == *Success*), the IoT device is enabled to process and store data:

$$\text{if } (\textit{Validation\_Status} = \textit{Success}) \text{ then } (D.\text{processData} = \text{true AND } D.\text{storeData} = \text{true})$$

The Consent Status is associated with the Consent Request ID (CRid):

We can model the association of Consent Status with CRid as a function.

$$\text{Associate}(CS_{\text{Encrypted}}, CRid) :$$

$$\text{Association\_Status} = \text{Associate}(CS_{\text{Encrypted}}, CRid)$$

5.9. Consent Status (CS) Association

Consent Status (CS) is associated with the Consent Request ID (CRid) for future reference and auditing purposes, using cryptographic hashes. The consent tracker function records the Consent Status.The recording of the Consent Status can be modeled as a function.

$$\text{RecordConsent}(CRid, CS_{\text{Final}}, DS, DC) :$$

$$\text{Record\_Status} = \text{RecordConsent}(CRid, CS_{\text{Final}}, DS, DC)$$

The data subject (DS) edits or deletes consent if required:

If the data subject decides to edit or delete the consent, these can be represented as functions.

$$\text{EditConsentFunction}(DS, CRid, \text{newCS}) \quad \text{and} \quad \text{DeleteConsentFunction}(DS, CRid)$$

$$\text{respectively:}$$

$$\text{if (DS decides to edit consent) then Edit\_Status = EditConsentFunction(DS, CRid, newCS)}$$

$$\text{if (DS decides to delete consent) then Delete\_Status = DeleteConsentFunction(DS, CRid)}$$

5.10. Consent Tracker

The consent tracker can be used to retrieve the Consent Status and other relevant information for auditing or verification purposes.

5.11. Third Party Data Access

Also, we have a third party (like a business or organization) which expresses the desire to gain access to the data pertaining to the data subject (DS) that are stored within a Consent Database (CD). This CD is maintained and managed by a separate data controller (DC). When the DC receives a data access request from the TP, the DC initiates a notification procedure to alert the DS about this request. The notification encompasses detailed information, including the identity of the TP and the specific data they seek to access. Upon receiving the notification, the DS is endowed with several options: they may choose to accept or deny the request outright, or they may opt to edit the selection of data that are to be shared. The DS's response is sent back to the CD and stored there, and subsequently sent to the DC and data sharing is initiated according to DS's response. If the DS decides to accept to the request, the relevant data are then shared with the TPar. In case of denial, no data are shared. If the DS decides to edit the data to be shared, only the approved selection are sent to the TPar. This scenario requires a formulation that encapsulates the various stages of the consent process, starting from the initial request for data access to the final decision of the DS regarding data sharing. The model will also incorporate elements such as data encryption, validation, and periodic review of consents, all from the perspective of the DS. The simple algorithm flow for the process is given below in Algorithm 2.

Algorithm 2 Third Party Handle Consent Process

```
Require: Request (REQ: TP, DC , DATA), DS, CD
Ensure: Share decision (SHARE: DS x DC x TP x DATA)
Begin
for each REQ in Request do
    $TP, DC, DATA \leftarrow REQ$
                                                    ▷ Step 1: Request for data access
    if Req($TP, DC, DATA$) == 1 then                ▷ Step 2: Notification to DS
        if Notify($DC, DS, TP, DATA$) == 1 then     ▷ Step 3: Response from DS
            $RESP \leftarrow$ Resp($DS, DC, TP, DATA$)
            if $RESP$ == accept then                ▷ Step 4: Update Consent Database
                if Update($DS, DC, TP, DATA$) == 1 then   ▷ Step 5: Data sharing decision
                    Share($DS, DC, TP, DATA$) := 1
                end if
            else if $RESP$ == deny then
                Share($DS, DC, TP, DATA$) := 0
            else if $RESP$ == edit then             ▷ DS updates the data to be shared
                $DATA' \leftarrow DS$.EditData($DATA$)   ▷ Recursive call with the updated data
                HandleConsentProcess(($TP, DC, DATA'$), $DS, CD$)
            end if
        end if
    end if
end for
End
```

6. Electronic Door Lock Scenario

Here is a scenario of how the consent flow might work for an electronic door lock using the Internet of Things. This is a simplification and could vary depending on specific circumstances and systems:

6.1. Data Subject (DS)

This is the individual who is using the electronic door lock. This could be a homeowner, a renter, or anyone else who needs to access the secure premises secured by the lock.

6.2. Data Controller (DC)

The entity that decides why and how personal data will be processed. This could be the company that manufactures and operates the electronic door lock.

6.3. IoT Device (D)

The electronic lock on the door itself. This device collects data (like lock/unlock times, and potentially who is using the lock if it has capabilities like fingerprint scanning or face recognition).

6.4. DS Profile (DP)

The set of data that the lock has collected about the DS. This could include information about when and how often the DS uses the lock.

6.5. Privacy Preference of DS (PP_{DS}) and Privacy Policies of DC (PP_{DC})

The DS may have specific preferences about their data (such as not wanting their data to be stored for more than a certain period of time). The DC will have privacy policies that dictate how they handle these data.

6.6. Personal Identifier (PI)

The DS's unique identifier that the electronic door lock uses to recognize them. This could be a digital key, a fingerprint, face recognition data, or some other form of identification.

6.7. Consent Request (CR) and Consent Request ID (CRid)

Before the DC can process the DS's data, they need to get the DS's consent. They send a consent request (CR), which is tracked using a unique Consent Request ID (CRid).

6.8. Time Period (TP), Purpose (P), Consent Status (CS)

The DS gives consent for a specific Time Period and Purpose. The Consent Status tracks whether the DS has given, denied, or withdrawn their consent.

6.9. Region (R)

The geographic location where data processing is taking place. The data protection laws can vary by region, and so the region can affect how the DC is allowed to process the DS's data.

For example, when setting up the door lock for the first time, the user (DS) might receive a notification from the manufacturer's app (DC) requesting consent (CR, with a unique CRid) to collect and process user's data. The Purpose (P) would be to ensure the security of the premises and user convenience, and the Time Period (TP) would be defined as the length of time the device is in use. The user can accept or deny the consent request, setting the Consent Status (CS). All information is stored and processed according to the privacy preferences of the DS (PP_{DS}) and the privacy policies of the DC (PP_{DC}). The user's data are then stored with a personal identifier (PI), which might be a user ID, linked with their DS Profile (DP) in the system. Data regulations and management could change according to the user's location or region (R). A simple JSON store for the data is given below.

JSON Snippet

```
{
  "IoT_App": "Smart Lock System",
  "Device": {
    "id": "D12345",
    "type": "Electronic Door Lock",
    "model": "ABC Model",
    "manufacturer": "ABC Locks",
    "description": "This is a smart lock...",
    "status": "Active",
    "reg_status": "initialized"
  },
  "Data_Subject": {
    "name": "John Doe",
    "email": "johndoe@email.com",
    "location": "USA",
    "devices": [
      {
        "device_id": "D12345",
        "type": "Electronic Door Lock",
        "reg_status": "completed"
      }
    ],
    "DS_key": "ds_public_key",
    "consent_reviews": [
      {
        "CRid": "CR987",
        "review_status": "pending",
        "review_timestamp": null
      }
    ]
  },
  "Data_Controller": {
    "name": "LockController Inc.",
    "DC_key": "dc_public_key",
    "consent_requests": [
      {
        "CRid": "CR987",
        "Dtype": "Access Control",
        "TP": "1 year",
        "P": "Processing of access control data for security",
        "CS": "pending",
        "encrypted_CS": "fghijklm==",
        "PI": ["access_times", "access_location"],
        "consent_validation": {
          "validated_by": null,
          "validation_timestamp": null
        }
      }
    ]
  },
  "Consent_Tracker": {
    "CRid": "CR987",
    "DS": "John Doe",
    "DC": "LockController Inc.",
    "CS": "pending",
    "TPShare": "True",
    "Track": {
      "Edit_Consent": false,
      "Delete_Consent": false.
    }
  }
}
```

7. Conclusions

The exploration of consent frameworks for use in the IoT in this article underscores the essential elements required to ethically and fairly handle user consent, ensuring privacy protection. Any effective consent mechanism must embed clear, knowledgeable, and have unequivocal consent protocols. It is of paramount importance that users can effortlessly provide, reject, retract, or adjust their consent throughout all phases of data aggregation and manipulation. Emphasizing a user-focused approach is indispensable. This includes reducing the overload of consent requests through pre-set privacy settings, offering comparative tools, and delivering enhanced suggestions that are tailored to individual preferences. Furthermore, the interfaces for acquiring consent must be user-friendly on various gadgets and should consistently update users on the data handling procedures. Despite current consent models facing challenges in terms of scalability, efficacy, adherence to legal standards, and centering on human needs, they present valuable foundational elements for ethical consent management. As we envision the future of IoT systems, it is imperative to instill privacy-focused design ideologies. Such a move will facilitate transparent consent procedures, well grounded in legal imperatives, and focused on human priorities, granting users a more dominant role in data management. This progression is not only about compliance; it is about cultivating trust and empowering individuals in an era where IoT is becoming increasingly intertwined with our everyday experiences.

8. Limitations and Future Work

This article presents a framework for consent management, with an acknowledgment that there are areas for improvement that will be addressed in future updates. These enhancements include elaborating on the technical specifics of implementation and validation to demonstrate the framework's scalability, highlighting the novel functionalities of a centralized, user-centric consent platform with features like automation and compliance, and incorporating advanced security measures and privacy infringement monitoring tools. Furthermore, future iterations will provide detailed examples of how the framework can address multifaceted privacy threats in IoT ecosystems. Furthermore, the proposed model faces challenges in dealing with DCs using deceptive tactics to manipulate consent, particularly when misleading requests imply that all data are necessary. This raises doubts about its effectiveness in situations where DCs do not follow 'data protection by default,' and particularly when DS lack awareness of privacy. Plans to use homomorphic encryption are also discussed, with the aim of keeping consent and data from data subjects (DS) encrypted, even when shared with third parties, ensuring security. This article does not dive into the formal proof of security or conduct a thorough threat analysis, including the mitigation of replay attacks. These elements will be included in future work, particularly when homomorphic encryption is integrated into the model.

Author Contributions: Conceptualization and Writing—original draft by P.K., Writing—review & editing by B.Y., J.-C.L., G.M. and S.U. All authors have read and agreed to the published version of the manuscript.

Funding: It was funded by Regionalt Forskningsfond Innlandet(RFF), grant number: 321136; and Research Council of Norway, Health Democratization project, grant number: 288856.

Data Availability Statement: This research has not gathered any data, and there has been no involvement with any living subjects.

Acknowledgments: The IoMt (Internet of My Things) project is funded by the Regional Research Fund (RFF) Innlandet of Norway, and we are deeply grateful for their generous support. We extend our special thanks to RFF Innlandet for their invaluable contribution to our research efforts. We also acknowledge the dedicated consortium working on the project, without whom this endeavor would not have been possible. Our sincere appreciation goes to IKOMM AS, Eidsiva Bredbånd, KeyFree AS, Safe4 Security Group AS, HelseInn, Lillehammer Kommune, NTNU, and Høgskolen i Innlandet for their collaboration and expertise.

Conflicts of Interest: The authors declare no conflict of interest. Stian Underbekken is employed in IKOMM AS. The funder Regionalt Forskningsfond Innlandet(RFF) and Health Democratization project, had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

IoT	Internet of Things
DS	Data Subject
DC	Data Controller
DP	DS Profile
PP_(DS)	Privacy Preference of DS
PP_(DC)	Privacy Policies of DC
PI	Personal Identifier
CR	Consent Request

CR_(id)	Consent Request ID
TP	Time Period
P	Purpose
CS	Consent Status
R	Region
DCM	Data Collection Methods
DU	Data Usage
DSSM	Data Storage and Security Measures
URO	User Rights and Options
DSE	Data Sharing Entities
GDPR	General Data Protection Regulation
CCPA	California Consumer Privacy Act
GPEN	Global Privacy Enforcement Network
IDRM	IoT Device Registration Middleware
OWL2-DL	Web Ontology Language 2 Description Logic
EULA	End User License Agreement
SeMF	Security Modelling Framework
JSON	JavaScript Object Notation
EEA	European Economic Area
PK	Public Key
PRK	Private Key
RSA	Rivest–Shamir–Adleman encryption algorithm
TPar	Third Party
CD	Consent Database
AI	Artificial Intelligence
LUCE	Logical Unified Consent Expressed

References

1. Madakam, S.; Lake, V.; Lake, V.; Lake, V. Others Internet of Things (IoT): A literature review. *J. Comput. Commun.* **2015**, *3*, 164. [CrossRef]
2. Kounoudes, A.; Kapitsaki, G. A mapping of IoT user-centric privacy preserving approaches to the GDPR. *Internet Things* **2020**, *11*, 100179. [CrossRef]
3. Psychoula, I.; Singh, D.; Chen, L.; Chen, F.; Holzinger, A.; Ning, H. Users' privacy concerns in IoT based applications. In Proceedings of the 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), Guangzhou, China, 8–12 October 2018; pp. 1887–1894.
4. Pardo, R.; Le Métayer, D. Analysis of privacy policies to enhance informed consent. In Proceedings of the Data And Applications Security And Privacy XXXIII: 33rd Annual IFIP WG 11.3 Conference, DBSec 2019, Charleston, SC, USA, 15–17 July 2019; pp. 177–198.
5. Regulation (EU) 2016/679 of the European Parliament and of the Council. *Off. J. Eur. Union* **2016**, *679*, 1–88.
6. Morel, V.; Cunche, M.; Le Métayer, D. A generic information and consent framework for the IoT. In Proceedings of the 2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), Rotorua, New Zealand, 5–8 August 2019; pp. 366–373.
7. Chikukwa, G. A Consent Framework for the Internet of Things in the GDPR Era. *Dak. State* **2021**.
8. Internet of My Things. (n.d.). Internet of My Things. Available online: https://www.internetofmythings.no (accessed on 11 October 2023).
9. Voigt, P.; Bussche, A. The eu general data protection regulation (GDPR). In *A Practical Guide*, 1st ed.; Springer International Publishing: Cham, Switzerland, 2017; Volume 10, pp. 5510–5555.
10. Rantos, K.; Drosatos, G.; Demertzis, K.; Ilioudis, C.; Papanikolaou, A.; Kritsas, A. ADvoCATE: A consent management platform for personal data processing in the IoT using blockchain technology. In Proceedings of the Innovative Security Solutions for Information Technology and Communications: 11th International Conference, SecITC 2018, Bucharest, Romania, 8–9 November 2018; pp. 300–313.
11. Jaiman, V.; Urovi, V. A consent model for blockchain-based health data sharing platforms. *IEEE Access* **2020**, *8*, 143734–143745. [CrossRef]
12. Neisse, R.; Baldini, G.; Steri, G.; Miyake, Y.; Kiyomoto, S.; Biswas, A. An agent-based framework for informed consent in the internet of things. In Proceedings of the 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), Milan, Italy, 14–16 December 2015; pp. 789–794.
13. Pandit, H.; Debruyne, C.; O'Sullivan, D.; Lewis, D. GConsent-a consent ontology based on the GDPR. In Proceedings of the Semantic Web: 16th International Conference, ESWC 2019, Portorož, Slovenia, 2–6 June 2019; pp. 270–282.

14. Merlec, M.; Lee, Y.; Hong, S.; In, H. A smart contract-based dynamic consent management system for personal data usage under GDPR. *Sensors* **2021**, *21*, 7994. [CrossRef] [PubMed]
15. Alhajri, M.; Rudolph, C.; Shahraki, A. A blockchain-based consent mechanism for access to fitness data in the healthcare context. *IEEE Access* **2022**, *10*, 22960–22979. [CrossRef]
16. Bröring, A.; Datta, S.; Bonnet, C. A categorization of discovery technologies for the internet of things. In Proceedings of the 6th International Conference on the Internet of Things, Stuttgart, Germany, 7–9 November 2016; pp. 131–139.
17. Khalil, K.; Elgazzar, K.; Seliem, M.; Bayoumi, M. Resource discovery techniques in the internet of things: A review. *Internet Things* **2020**, *12*, 100293. [CrossRef]
18. Top 7 Open Source Home Automation Software in 2023. (n.d.). FixThePhoto.com. Available online: https://fixthephoto.com/best-open-source-home-automation-software.html (accessed on 11 October 2023).
19. Alom, M.; Carminati, B.; Ferrari, E. Helping users managing context-based privacy preferences. In Proceedings of the 2019 IEEE International Conference on Services Computing (SCC), Milan, Italy, 8–13 July 2019; pp. 100–107.
20. Lee, H.; Kobsa, A. Privacy preference modeling and prediction in a simulated campuswide IoT environment. In Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications (PerCom), Kona, HI, USA, 13–17 March 2017; pp. 276–285.
21. Yue, W.; Liu, X. Strategies for Intelligent Registration of IoT Devices. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021; pp. 1–7.
22. Geller, S.; Müller, S.; Scheider, S.; Woopen, C.; Meister, S. Value-based Consent Model: A Design Thinking Approach for Enabling Informed Consent in Medical Data Research. In Proceedings of the 2022 HEALTHINF—15th International Conference on Health Informatics, Vienna, Austria, 9–11 February 2022; pp. 81–92.
23. Utz, C.; Degeling, M.; Fahl, S.; Schaub, F.; Holz, T. (Un) informed consent: Studying GDPR consent notices in the field. In Proceedings of the 2019 ACM Sigsac Conference on Computer and Communications Security, London, UK, 11–15 November 2019; pp. 973–990.
24. Hu, R.; Yan, Z.; Ding, W.; Yang, L. A survey on data provenance in IoT. *World Wide Web* **2020**, *23*, 1441–1463. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.