Karlsruhe Institute of Technology

HECTOR SCHOOL
OF ENGINEERING & MANAGEMENT

Executive Master Program:
Information System Engineering and Management

Master Thesis

TRANSPARENCY OF DATA PROCESSING WITHIN DATA TRUSTEE PLATFORM OF SLEEP STUDIES

FZI Research Center for Information Technology

Haid-und-Neu-Straße 10–14

76131 Karlsruhe

Supervisors: **Prof. Dr. rer. nat Wilhelm Stork** (Institute for Information Processing Technologies)

Prof. Dr. Stefan Nickel (Institute for Operations Research)

Advisor: **M.Sc. Christina Erler** (Institute for Information Processing Technologies)

Date of submission: 31.07.2023

Buwei Liao

Rastatter Str. 108A, Karlsruhe

Germany

Date of birth, Location: 23.11.1992, China

Number of matriculation: 2381740

I sincerely affirm to have composed this thesis work autonomously, to have indicated completely and accurately all aids and sources used and to have marked anything taken from other works, with or without changes. Furthermore, I affirm to have observed the constitution of the KIT for the safeguarding of good scientific practice, as amended.

Karlsruhe, 31.07.2023 _____

Acknowledgement

First, I am grateful of attending Hector School of KIT. I have received great master education; it will be an unforgettable journey for me. The courses were well organized; the teaching professors always did their best to enlighten us; and the classmates are all outstanding.

I would like to express my heartfelt gratitude to my supervisors; Prof. Wilhelm Stork and Prof. Stefan Nickel, for all the support they have given me throughout the entire process of pursuing my master's degree. Moreover, I am truly grateful to my advisor, Christina Erler, for her professional guidance, and expertise throughout the entire process of completing this master's thesis. Her invaluable insights, constructive feedback, and encouragement have been instrumental in shaping the direction and quality of this work.

I would also like to extend my gatitude to my loving wife, Baiyu Diao. Her understanding and sacrifices have made it possible for me to pursue my academic goals.

Lastly, I would like to acknowledge the contributions of all the individuals who have directly or indirectly supported me during this research endeavour. Their assistance, feedback have been invaluable.

Thank you all for being a part of this significant milestone in my academic journey.

Abstract

Using data analytics to investigate new healthcare choices has become popular. But don’t overlook that regulations must be followed while collecting data from EU citizens in accordance with the GDPR data protection law. Data trustee platforms for neutrally hosting data have appeared in this setting to regulate data usage behaviours with a controlled computing environment and prevent potential misuse of data. These platforms assure its donors that the information they provide will be used only in accordance with the terms of their agreement with the platform. From the data donators’ point of view, they lost track of their data after the act of contribution. How their data is processed, and who has used their data? Only if the platform respond well to these questions and become more transparent, it can accumulate trust from donators. Publicizing records of processing activities is also one major responsibility of these platforms (Art. 2 Para. 1 GDPR) [65]. In this thesis, we will closely look into a data trustee that hosts sleep data as an example. First, we present a theory framework for improving its transparency, then take the theory into detailed concept design and implementation, and finally evaluate the effectiveness of the work.

Keywords: transparency, data trustee, sleep research, blockchain, immutability, secure logging.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

GDPR	**G**eneral **D**ata **P**rotection **L**egislation
BDSG	**B**undes**D**aten**S**chutz**G**esetz
DGA	**D**ata **G**overnance **A**ct
REM	**R**apid **E**ye **M**ovement
POC	**P**roof of **C**oncept
MAC	**M**essage **A**uthentication **C**ode
DSRM	**D**esign **S**cience **R**esearch **M**ethodology
SUS	**S**ystem **U**sability **S**cale
WORM	**W**rite-**O**nce **R**ead-**M**any
OTS	**O**ff-**T**he-**S**helf
SGX	**S**oftware **G**uard **E**xtensions
TPM	**T**rusted **P**latform **M**odule
TEE	**T**rusted **E**xecution **E**nvironment
PoA	**P**roof **O**f **A**uthority
IPFS	**I**nter**P**lanetary **F**ile **S**ystem
PoW	**P**roof **O**f **W**ork
PoS	**P**roof **O**f **S**take
DApp	**D**ecentralized **A**pplication
ORM	**O**bject **R**elational **M**apping
JSON	**J**ava**S**cript **O**bject **N**otation
ABI	**A**pplication **B**inary **I**nterface

1 INTRODUCTION

This chapter will introduce the background of the topic of the thesis, the motives for researching the topic, the key research questions to be addressed, and the overall structure of the entire thesis.

1.1 Background

Sleep is essential for maintaining physiological health and cognitive performance during the day. Lack of high-quality sleep may negatively affect work-life balance, general health, and safety. Additionally, sleep difficulties are frequently linked to several serious health issues like diabetes mellitus, cardiovascular disease, dementia, mental illness, and chronic pain. Therefore, effective treatments must be developed to adequately care for patients who come with sleep-related morbidity [1].

In order to make achievements in the field of sleep research, researchers need to run their analysis on the relatively large amounts of sleep data to find the answers to the mechanisms and functions of sleep. Sleep disorders often have a complex etiology, making it difficult to establish a cause-and-effect relationships when observing only a few patients [2]. Big data analysis is expected to have a significant influence on the field of sleep studies and may help researchers to obtain more robust and stronger scientific evidence [3].

Currently, sleep data are sparsely stored in various sleep study labs or clinics. These data constitute the sources of an incredible richness of knowledge about patients. But how to gather this data and redistribute them according to legal regulations? Data trustees are currently considered by academia and industry as a promising solution to these problems, as they can establish the necessary pathway for trustworthy data exchange among multi-stakeholders, enabling data

sharing and processing. These data trustees improve the accessibility and quality of multidisciplinary big-data sources from diverse sectors in a trustworthy, fair, and responsible way [4]. In the field of sleep studies, data trustee platforms make big data analysis possible for clinical research and healthcare process improvement.

In Europe, data trustee platforms have to comply with General Data Protection Regulation (GDPR) [5]. The collected data can only be processed for use within the scope of registered purposes if the data subject has previously given consent. Moreover, data donors should be able to exercise all of their rights under the GDPR – accessing data, changing approvals, exporting data, having data erased, etc. – via the platform and see at any time for which services they have issued what kind of approvals.

Under this legal setting, data trustee need to figure out a way to enhance transparency of data processing within the platform. The increased transparency not only meet legal requirements but also earn more trust from data donators, hence the platform will be able to reach larger datasets and benefit more science researchers.

1.2 Motivation

As what we have introduced in the previous section, in order to promote the sustainable development of platforms, it is important for data trustee platforms to proactively disclose information about how donators' data is being processed. The benefits are manifold. On the one hand, third-party legal enforcement agencies can step in to intervene in a relatively early stage and examine the platform's compliance, rather than after privacy issues have already occurred [6]. On the other hand, it also allows donors to know that their data is being kept in a secure place and is not being used for anything other than its designated purpose [7]. In the meantime, of alleviating existing donors' concerns about data privacy, it can also have the effect of calling for more donors to join in [8].

Current data platforms do not pay sufficient attention to information disclosure. Donor's data may be used in many research projects, but donors do not know enough details [9]. This might potentially violate the GDPR requirement that donors "should have the right to be informed". In some cases of non-compliance, audit logs are often a key line of inquiry. Audit logs provide a chronological record of key events that occurred in the system and therefore provide some traceability to data privacy incidents. Most of the time, this log information is usually just recorded and stored in the database. Huge efforts are needed to make use of log data [10]. They are not perceptible to end users and do not fully serve their purpose. To some extent, the platform's data processing is almost opaque to the outside world. Donors can only blindly trust that the platform will not violate the rules, and third-party legal agencies can only step in after a breach has occurred.

We have already addressed the need for platform transparency at the regulatory level. There is also another scenario, reproduction of research results in the field of sleep research. A comprehensive review of research results by the research community requires replication of data results, which is often difficult for two reasons: firstly, other peers do not always have access to the same data sources; secondly, there is no way to know the detailed steps of data processing [69].

1.3 Research questions

The goal of this thesis is to develop the conceptual model and prototype of a log auditing system, which makes the processing of data within the data trustee platform more transparent. In order to achieve the transparency requirement, it is first necessary to collect critical log data and store them in a tamper-evident form. Then, to analyze different stakeholders' claims for platform transparency and determine types of information to disclose to promote the long-term sustainability of the platform. Finally, to analyze and visualize the log data, and present information that is easy to interpret for different roles.

To achieve the goal of this thesis, the following research questions should be answered in the course of the work:

RQ1. How to ensure that audit logs accurately reflect the data processing activities that occur in the platform?

RQ2. What exactly are the stakeholders' requirements for transparency, and what kind of information should be extracted from events happened within the platform?

RQ3. In which way the disclosure of the information is most effective, so that stakeholders can easily access and consume the log information?

1.4 Outline

After introducing the necessity of this topic and research directions to approach to solving the problems in Chapter 1, we will review the research methodology used in the thesis project in Chapter 2. Before going further, we will cover some basic concepts and offer concise explanations to technology terms related to this thesis in Chapter 3.

Chapter 4 will present the state of the art and develop a taxonomy in the area of secure logging. In Chapter 5, we revisit the research questions and do core concept design to solve these questions based on a data trustee for sleep research as an example. In Chapter 6, there will be major steps of implementation with technology and programming language of the previous concept design. Chapter 7 will show the evaluation methods and results of the implemented work, and discuss these results we got.

Finally, in Chapter 8, we draw the conclusion of our research findings and also point several potential future directions to explore further.

2 METHODOLOGY

We used a Design Science Research Methodology (DSRM) in this thesis. According to Peffers et al. [12] DSRM includes the following six steps:

1. **Problem identification and motivation**: Defining problem, conveying importance,
2. **Definition of the objectives for a solution**: Considering what would a better artifact accomplish,
3. **Design and development**: Implementing the artefact,
4. **Demonstration**: Finding the suitable context and using the artifact to solve problem,
5. **Evaluation**: Observing how effective the artifact is, iterating back to design,
6. **Communication**: The publication of the work.

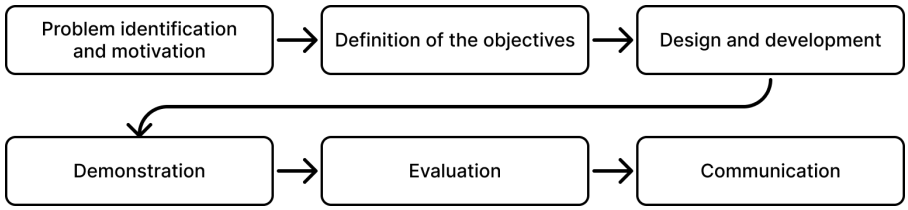

Fig. 2.1: The six steps of DSRM according to Peffers et al. [12].

In order to understand current achievements and best practices of the academia in addressing the transparency problems in data trustee platforms, we first got an understanding about the state of the art through extensive literature searching and reading. Then we developed a taxonomy according to Nickerson et al. [76]. Our taxonomy illustrates groups of technical terms related to our topic. The taxonomy sets a formal framework for analysing and categorizing current solutions and establishing benchmark of these solutions.

After literature review, we followed the user story method introduced by Sommerville [77] to formulate requirements of different stakeholders within the data trustee platform. We first understood how people are interacting with the data trustee platform for sleep research by observation and ethnography. Next, we used persona technique [70] to capture the features of different stakeholders. We developed use cases and scenarios to further refine their requirements. Then, the requirements are classified into functional requirement and non-functional requirements. At last, we used a MoSCoW technique [68] to prioritize the requirements.

We chose the foremost urgent requirements from stakeholders and design solutions in the form of interactive UI prototypes, which will later be used to create tasks and evaluate the effectiveness of our solutions [78]. Together with the UI prototypes is the implementation of a fully functional mock-up system. Different from the UI prototype, the mock-up system is backed by real code. The underlying technology architecture might not be sensible to average users without technology background, but it is the backbone that support the achievement of transparency, and it will be evaluated with more technical persons.

In the evaluation process, we used primarily the questionnaire and live interview sessions to measure the usability and degree of satisfaction of requirements. In the questionnaire a set of tasks and a System Usability Scale (SUS) [13].

3 BASICS

In this chapter, we will give short definitions and introductions to terminologies, technologies mentioned in the thesis.

3.1 Legal related terms

In the following paragraphs, we will introduce data protection laws in European Union countries, and also some important concepts from the laws.

3.1.1 GDPR

GDPR [65] stands for General Data Protection Regulations. It is a European Union (EU) law governs how we can use, process, and store personal data.

3.1.2 Personal data

In the definition of GDPR, personal data are any information which are related to an identified or identifiable natural person (Art. 4 GDPR) [65].

3.1.3 Right to be informed

GDPR gives individuals a right to be informed about the collection and use of their personal data (Art. 13 & Art. 14 GDPR) [65].

3.1.4 BDSG

Bundesdatenschutzgesetz [66] (BDSG in short), is Germany's Federal Data Protection Act supplements and specifies the GDPR in those areas that are left to the national regulations of the EU member states.

3.1.5 DGA

European Data Governance Act provides a framework to enhance trust in voluntary data sharing for the benefit of businesses and citizens [67].

3.1.6 Consent

Processing personal data is prohibited, unless law expressly permits it, or data user has acquired consent from the data subject. Consent must be given voluntarily, specific, informed and unambiguous. The basic requirements for a valid legal consent are defined in Article 7 and specified further in recital 32 of the GDPR [65].

3.2 Blockchain related terms

In the following paragraphs, we will introduce different types of Blockchain and discuss their differences. We will also introduce some key components and property of Blockchain.

3.2.1 Blockchain

A blockchain is a distributed ledger shared among a decentralized peer-to-peer network of computer nodes. It allows multiple parties to maintain the shared record of transactions in a secure and transparent manner. It is often referred to as a "chain of blocks" because it consists of a series of blocks, each containing a list of transactions or data. Blockchain technology has gained significant attention due to its potential applications beyond cryptocurrencies like Bitcoin [64]. It can be used in various industries, including finance, supply chain management, healthcare, voting systems, and more [79]. By providing a secure and transparent platform for recording and verifying transactions or information, blockchain has the potential to revolutionize how we exchange value and trust in the digital world.

3.2.2 Public Blockchain

Public Blockchain is open to anyone who wants to participate in the network. Anyone can create an address, send transactions, and be part of the process of adding new blocks to the chain.

One of the key features of a public blockchain is its transparency. All transactions recorded on the blockchain are visible to all participants, allowing for public scrutiny and verification. This transparency ensures accountability and trust among the participants, as any changes or additions to the blockchain can be easily detected and verified by anyone. [94]

3.2.3 Private Blockchain

Opposed to public Blockchain, in a private blockchain, only selected and verified participants may join activities of the Blockchain. And the operator of Blockchain has the rights to override, edit, or delete entries on the blockchain [94].

3.2.4 Consortium Blockchain

A consortium blockchain is a type of private blockchain network. Consortium blockchains are typically used by organizations that want to maintain a certain level of privacy and control over their data while still benefiting from the transparency and security provided by blockchain technology. [95]

3.2.5 Ethereum

Ethereum is a public Blockchain, anyone is free to join and participate in the core activities of the blockchain network [80]. Ethereum offers a turing-complete programming language: Solidity, and a integrated executing environment. This expands the capabilities of blockchain technology by providing a platform for developers to build and deploy smart contracts.

3.2.6 Smart contract

Smart contracts are self-executing contracts with the terms of the agreement directly written into code. They automatically execute actions when predefined conditions are met, without the need for intermediaries. [96]

3.2.7 Immutability

Immutability is a property of data. When we say the data is immutable, it cannot be modified or deleted in any way. [88]

3.2.8 Consensus algorithm

Consensus algorithm synchronize state machines on different servers, it ensures consistency among them. It helps to achieve trust and security across a decentralized computer network. [97]

3.3 Architecture related terms

3.3.1 Client-server

Client-server is an architecture model. In the client-server setting, the data processing is handled by the server, and the results are returned to the clients. While the client initiates requests for a service or resource from the server [93].

3.4 Sleep research related terms

3.4.1 Polysomnography

Polysomnography is a diagnostic test used to evaluate and diagnose sleep disorders. It involves monitoring and recording various physiological parameters during sleep, such as brain activity (EEG), eye movements (EOG), muscle activity

(EMG), heart rate, breathing patterns, and oxygen levels. This comprehensive assessment provides valuable information about sleep stages, sleep architecture, and the presence of abnormalities like sleep apnea, narcolepsy, insomnia, or restless leg syndrome. Polysomnography is typically conducted in a sleep laboratory or clinic, and the data collected helps healthcare professionals make accurate diagnoses and develop appropriate treatment plans for individuals experiencing sleep-related issues. [81]

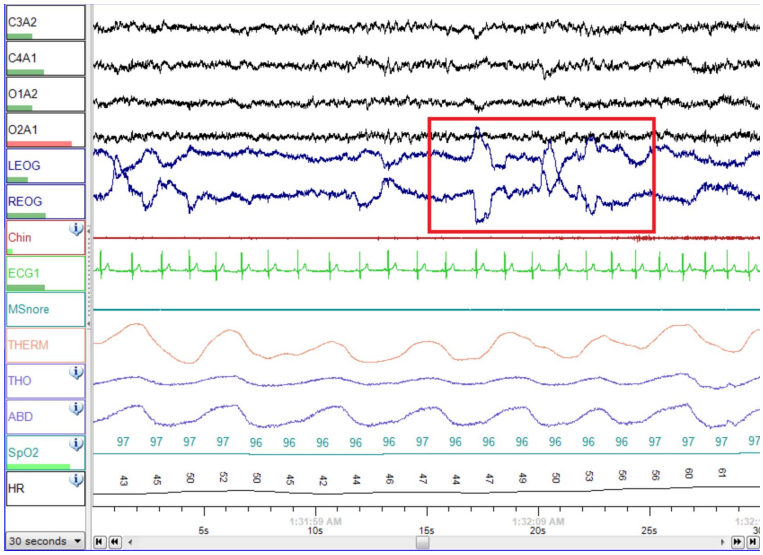

Fig. 3.1: Screenshot of a PSG of a person in REM sleep [82].

3.4.2 Hypnogram

A hypnogram is a simplified form of polysomnography. The graph of hypnogram shows multiple stages of sleep: rapid eye movement (REM), non-REM, deep sleep, etc. Each stage is represented by a specific pattern or waveform. The hypnogram helps to analyze sleep architecture and identify any disruptions or abnormalities in sleep patterns. It is commonly used in sleep studies and research to understand sleep quality and diagnose sleep disorders. [82]

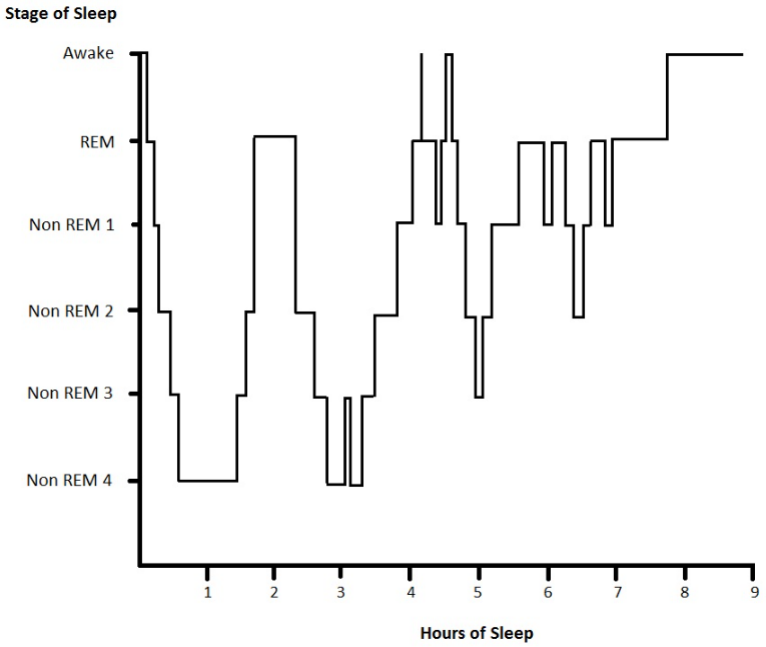

Fig. 3.2: Example hypnogram of a normal, healthy adult [82].

4 STATE OF THE ART

In this chapter, we will introduce how we first used a “problem space” thinking framework [14] to discuss the problem we were trying to solve. Then with the very first simple definitions of the problem space and solution space, we curated a short list of most relevant papers and got keywords inspiration from them. The problem space and solution space was then expanded after first round reading. We formulated the search string from it. The search string was used to connect to the most relevant literature available. These search results we got will go through two rounds of filtering and finally be grouped into several categories. We will discuss the different categories of approaches these papers are taking to solve the transparency problem and decide a most relevant path as our own approach.

4.1 Defining problem space and solution space

Before we march into the academic publishing websites, first think about the logic behind the transparency problem.

Inside a software system, there are multiple actions provided by the system. These actions could be performed by users. When each action is performed, there is contextual information generated. To make the system accountable, the context information has to be recorded. We often refer to the collected contextual information as logs. [84] (Fig. 4.1)

Fig. 4.1: The logic behind birth of logs.

The problem of lacking traspanrecy within data trustee platform is caused by lack of information on “data processing events occuring within the platfom” [85]. The data trustee platform is composed of the software and system users. The software can be programmed by the platform creator to automatically perform actions, or triggered by users to do so. Each action will result in an event. An event in its essence is contextual information about somebody did something in the past. The logs are the finalized form of events. They carry the information about a series of events. They can work together to reconstruct the scene and tell a story to those who wants to know what happened within the software platform. To enhance transparency of the platform, we need to publish log information to curious parties.

We can shape our basic understanding of the transparency topic into the “problem space” model. The concept of “problem space” was first introduced in 1979 [14]. The problem space theory uses the approach of defining the problem to find the solution. There is not much details about the final solution in the problem space, but instead it focus on steps and goals involved in working through the problem, it outlines what is needed to attain a solution.

The more modern discussion and application of the problem space concept [15],

> ***Problem space*** *is where all the customer needs that you’d like your product to deliver live … Whether it’s a customer pain point, a desire, a job to be done, or a user story, it lives in problem space.* ***Solution space*** *includes any product or representation of a product that is used by or intended for use by a customer.*

The fist version of problem space model (Fig. 4.2) is quite simple. What we want to do here is to improve current state of platform — no trace of data processing actions. And we use logs to address this problem. So we can draw the direct relationship between “logging” and “no trace of actions” (Fig. 4.2).

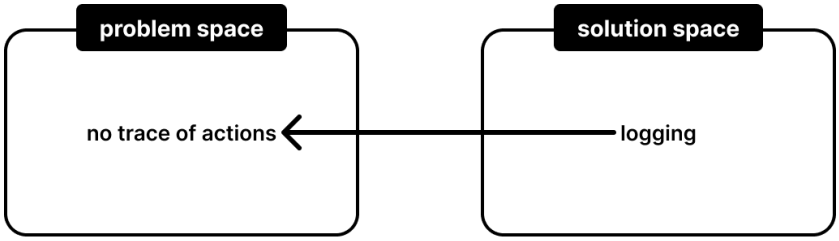

Fig. 4.2: The initial version of “problem space” model (extended from Fig. 4.1).

We used Google Scholar to search keywords “logging” and “transparency”, and read the most cited papers to get a simple understanding of logging syste. We found that logging is a widely adopted method in the software industry to accomplish following tasks: anomaly detection [17] [18], error debugging [19], performance diagnosis [20], workload measurement [21], system behaviour understanding [22], etc. In this thesis, we focus on logging as a major way to look into system behaviours. Logs may record the platform’s runtime actions and states that can directly reflect the platform’s runtime behaviours.

Log tampering, which means log information is deleted or modified, is a wide spread issue. In 72% of conducted cyber attack investigations, tampering evidence was uncovered [23]. There is some popular malware (e.g., BlackEnergy [24]) could automatically delete logs [25], [26] and hide attackers’ tracks. Once attackers get root access on a server, they can modify or delete system logs, thus obstructing after-event forensic analysis.

Having logging system in place is not enough, we need to make sure logs are secured, so that log information is not falsified by attackers. Furthermore, we need publicize log information the stakeholders and make log information verifiable to gain trust from the stakeholders. In Fig. 4.3, we extend the problem space and solution space model.

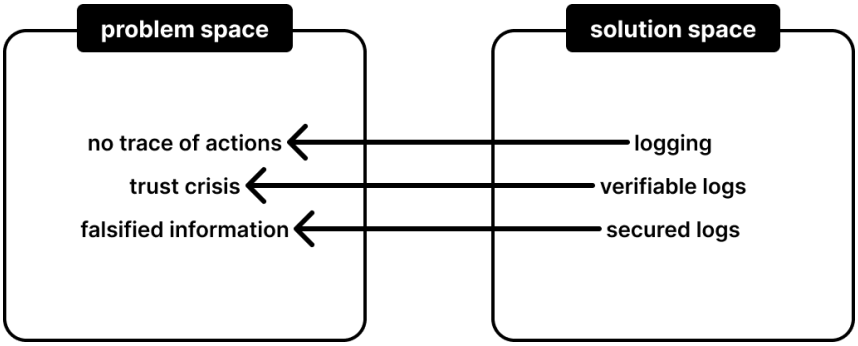

Fig. 4.3: The extended “problem space” model with verifiability and security of logs

A reliable logging system that could be trusted by the public will be the backbone of our solution. And in the literature research process, we first build the search string based on “logging system” and other properties related to the “logging system”.

4.2 Literature research

In this section, we will first develop search keywords based on CIA triad model. Then we present how we used keywords to develop a search string. The search string was used to connect all relevant literatures. At last, we present the result of filtering and grouping the literatures.

4.2.1 Developing keywords

As a starting point, we borrow the CIA triad model from [28] the area of information security to explore additional properties of a “reliable, trustable logging system”. See Fig. 4.4 for the visual expression of the mode. The CIA triad was first introduced to measure security level of classified military or government information. Here are the definitions of CIA from Samonas et al.:

1. **Confidentiality**: the system should prevent unauthorized information release. Non-related parties should not have access to secured information,

2. **Integrity**: the system should prevent unauthorized information modification. Either deleting or making changes to information is regarded as violation,

3. **Availability**: the system should prevent unauthorized denial of use. Intruders may obstruct the information flow, so that other system users cannot have timely, reliable accesses to the information. [28]

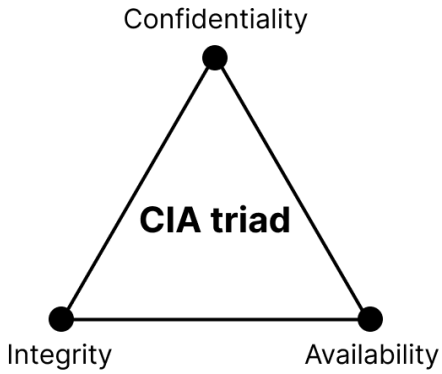

Fig. 4.4: CIA triad [28]

Among these three elements, integrity is the most intricate one. It packs following properties together:

1. **Authenticity**: the log information from the system is correct and genuine, it truly reflects to real history events.
2. **Non-repudiation**: it guarantees authenticity, but not vice versa. It means related parties are not able to contradict the facts supported by the log information.
3. **Immutability**: log information resides in the system are resistant to any change or deletion.
4. **Tamper-proof**: log information is not susceptible to change, if it happens there will be evident traces. Tamper-proof information could be deleted. Immutability assures tamper-proof, but not vice versa.

In the socio-technical aspect of security, we also need to examine the dynamics between different parties that are connected to the system [86]:

1. Accountability: parties within the system actively take the responsibility for acting honestly and ethically. Otherwise, that party will be held accounTable
2. Provenance tracking: log information about an event could help use to trace back to when it happened, so that we can understand the original context of the event.
3. Auditability: the ability of an auditor to get accurate results when they exam the system's log information.

After the event of violation, the logging system should be able to support cyber investigation:

1. **Forensic analysis**: forensic analysis is based on audit trails. Analysist relies on audit trails to detect known attack patterns, deviations from normal behavior, or security policy violations [29].

The following methods are often used to achieve information security:

4. **Distributed system**: distributing data for performance, availability and durability has been widely adopted in the file system and database communities [30]. With distributed system, we can avoid "single point of failure", thus achieve higher availability of information.
5. **Cryptography**: it is a technique used to achieve confidentiality of messages. Only authorized receiver can access the information [31]. Depends on different types or combinations of cryptography algorithm, different effects could be achieved.
6. **Blockchain**: a decentralized ledger, it uses cryptography and distributed consensus algorithms to implement user security and ledger consistency. Information stored in Blockchain is immutable and audiTable

4.2.2 Search string

We can group the keywords into two categories: the topic keywords and the descriptor keywords (See Fig. 4.5). Keywords in the topic group are essential and non-replaceable in a title, at least one topic keyword should be included in the target paper otherwise the paper is very likely irrelevant. Keywords in the descriptor group are properties or methods should be binded to the topic. At least one descriptor should be included in the title.

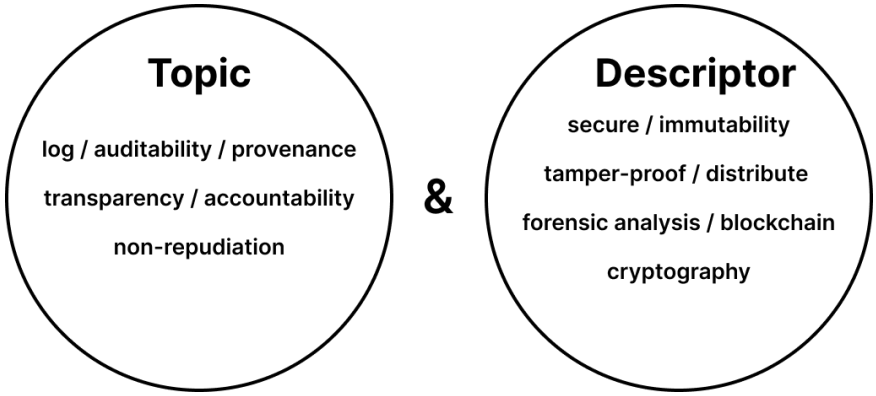

Fig. 4.5: Topic keywords and descriptors

Thus we can glue the keywords together with widely adopted boolean operators and formulate the search string as:

(log OR auditability OR provenance OR transparency OR accountability OR non-repudiation) AND (secure OR immutability OR tamper-proof OR distribute OR forensic analysis OR blockchain OR cryptography)

To make the search string more general and adaptive, we truncated the keywords a little bit:

(log OR audit OR provenanc OR transparen OR accountab OR repudia) AND (secur OR immutab OR tamper OR distribut OR forensic OR blockchain OR cryptograph)

In this way, both “immutable logging system” and “immutability of logging system” could be hit and matched by the search string.

4.2.3 Sources

We selected following academic publication databases as our primary literature sources: *Wiley Online Library, IEEE Xplore, ACM DL, ResearchGate, ScienceDirect, Scopus, Springer Link*.

4.2.4 Filtering and grouping the literature

We did the first round filtering based on the title, keywords and abstract. We got 114 items in total after deduplication.

In the second round, we skimmed through the full text and only picked the ones with high quality and relevance to our topic. The following standards are applied:

1. **Accessible**: PDF files are accessible.
2. **Cite counts**: papers with only 0~5 cites are considered as lack of authority..
3. **Page length**: short papers with only 3~5 pages don’t have much depth.
4. **Relevance**: papers do not put focus on log security are considered as irrelevant.
5. **Domain specific**: papers discuss logging security topics about IoT, network, or electronic patient record only are being too specific.
6. **Complete solution**: we favour papers that describe a complete solution rather than only sketch part of the problem.

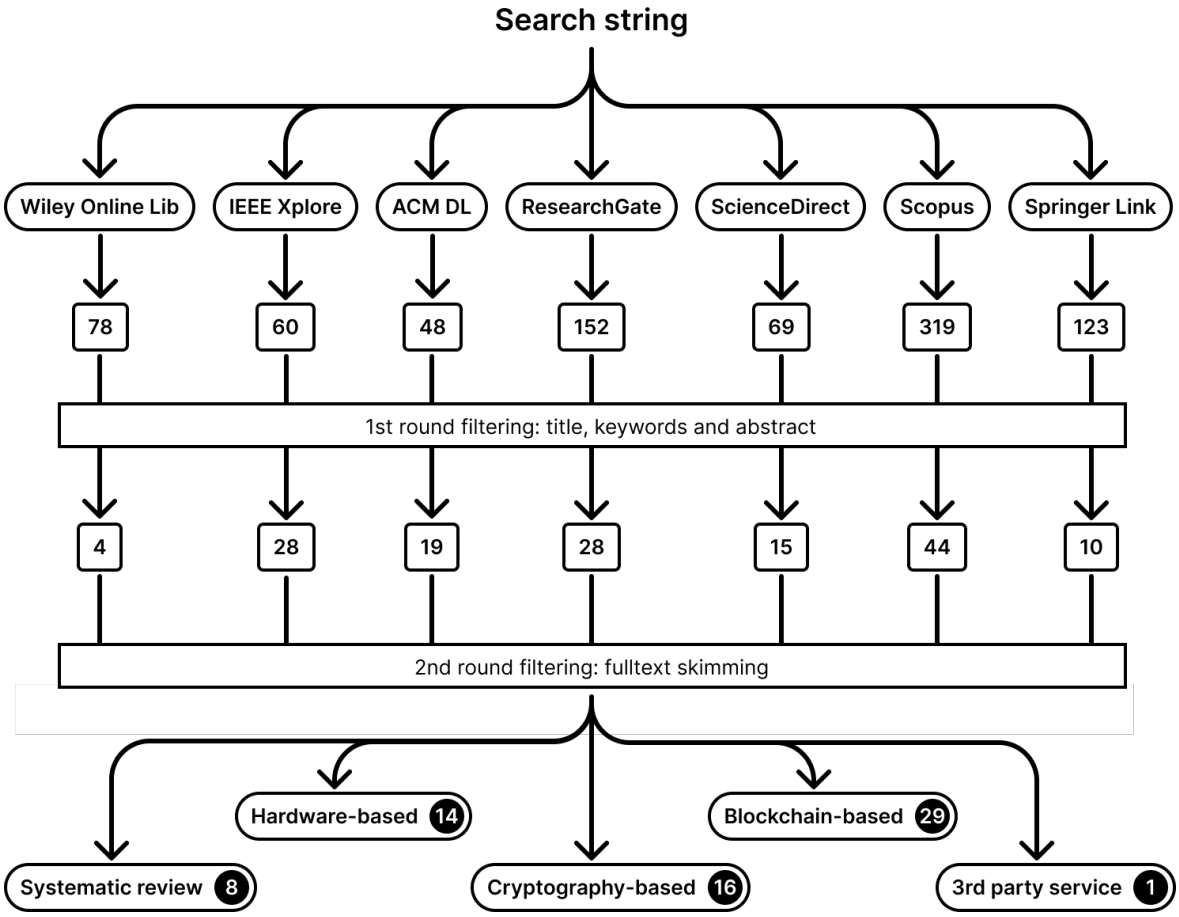

Fig. 4.6: The step by step literature searching, filtering and grouping processes.

After the final round of filtering, we grouped them into following categories:

1. **Systematic review**: papers in this group covers a range of solutions from different papers, they offers summarizations and comparisons of different solutions.
2. **Cryptography-based**: papers in this group focus on utilizing cryptographic method to achieve log information security.
3. **Hardware-based**: papers in this group use certain type of secure hardware to make sure security, and they often also integrate cryptographic methods.
4. **Blockchain-based**: papers in this group primarily focus on applying Blockchain and smart contract to make log information immuTable

5. **Third party service-based**: rather than coming up a security solution in the house, they seek help from a third party.

The step-by-step literature searching, filtering and grouping prcesses is expressed visually in Fig. 4.6.

4.3 Overview of relevant literature

In this section, we will organize solutions into different categories, and examine the advantages and disadvantages of each category.

4.3.1 Cryptography-based solution

This category of solutions used pure cryptographic algorithms to compose log items together in hope to make them tamper-proof.

Schneier & Kelsey [32] [33] demonstrated how Message Authentication Codes (MACs) combined with hashing algorithm can be used to generate a robust chain of log items. Ballare & Yee [34] discussed how MAC secret keys could evolve along the process (each new different key is derived from the previous key) and assures that:

1. Confidentiality of each log item.
2. The previously generated logs connot be modified.
3. Deletion of log items can be detected.

In such setting, the base MAC key and key evolvement algorithm are needed to verify log items, which makes it impossible to be publicly verified. Holt [36] proposed an alternative solution that uses a combination of public key cryptography and hash chain to achieve public verifiability. These approaches are further enhanced by Ma & Tsudik [35], who demonstrated how individual log item

signatures can be merged together into an aggregate signature. The previous verification process is to verify log items one by one, now it becomes chunk by chunk. This can solve truncation attack.

4.3.2 Secure hardware-based solution

This category of solutions resort to secure hardware to store log items or encryptions keys. They often also integrate cryptographic methods from previous category.

Wang & Zheng introduced a solution based on Write-Once Read-Many (WORM) storage device [41]. Hsu & Ong pointed out WORM is not enough; a holistic approach to store, manage and deliver is required [59].

Chong et al. [42] used resource constrained Java iButton as a tamper-proof hardware token, but due to some key sharing schemes, it does not bring better protection than cryptography-based solutions. Wouters et al. [61] and Pulls et al. [60] offer custom hardware design for secure logging. These kind of solutions are not universally applicable, because they completely avoid commercial off-the-shelf (COTS) hardware.

Later implementations like SGX-Log [44], EmLog [45], and BBox [46] used Intel Software Guard Extensions (SGX), Trusted Platform Module (TPM), or Trusted Execution Environment (TEE) to generate and store symmetric encryption keys. These keys are constantly evolving to guarantee forward security, this brings unbearable workload for resource constrained security hardware.

4.3.3 Blockchain-based solution

This category of solutions are based on Blockchain. They use Blockchain store log information or integrity proof of logs to make sure they are immuTable This kind of solutions are also the mainstream in recent industry practice.

Blockchain has its built-in nature of immutability. It is widely adopted as a secure storage option for medical records in recent years [62] [63]. The ultimate purpose of logging system we are discussing here is to protect data donators' data privacy. So it makes sense that technology used to protect donators' data could also be a sound option to protect log data.

BlockAudit 2.0 [51] used a proof of authority (PoA) Blockchain and database to achieve log integrity and availability. AuditTrust [52] used Hyperledger Besu Blockchain, IPFS to store metadata of log items and used database to store original log information. EngraveChain [54] used cloud storage and distributed storage system StorJ to store large log files, only hash of the files are written into the Blockchain.

Fig. 4.7: On-chain and off-chain collaboration.

Other implementations (e.g., BlockTrail [55], Medusa [56], BCALS [57]) also used a combination of on-chain and off-chain storage. The specific type of Blockchain they chose may vary, but the ideas of storing integrity proof to the Blockchain and original log files to off-chain storage options are the same (e.g., IPFS, database, cloud storage service, StorJ). On-chain storage and off-chain storage collaborates with each other while storing log information and also retrieving log information (Fig. 4.7).

4.3.4 Third party based solution

This category means the solution relies on a third party service for storing or notarizing log information. The service provider may also use other solutions (e.g., cryptography-based or Blockchain-based solutions) mentioned above.

Snodgrass et al. [48] used a notarization service that accepts a document needs to be notarized and returns a notary ID. The notary ID will then be stored in the database and later retrieved to verify the authenticity of logs.

4.4 Taxonomy development

We present the structure of taxonomy in six primary columns:

1. **Paper**: The detail properties of each paper is organized in a row. There are publication date, author names and reference number in this column. The publication date also help us to identify gradual evolution of technology.
2. **Scheme**: some papers made a name for their solution, if a scheme name is available it will be displayed in this column.
3. **Technology**: we can generally group different technologies into previously mentioned four categories. Some paper might have used a combination, for example iButton [42] used both specially designed secure hardware and also cryptography. In this case we mark it as “hardware” group.
4. **Security measures**: these are common approaches being used to protect log information security. Specific steps and details may vary, but as long as a similar method was mentioned to achieve same purpose, it will get a check mark.

    a) Forward security: after a key being compromised, the integrity of log entries generated by previous keys remain intact.

    b) Data encryption: logs are stored in cipher text, one cannot read the information without corresponding decryption keys.

    c) Secure retrieval: remote, authorised third parties shall be able to securely retrieve logs.

d) Public verifiable: the third parties shall be able to verify the integrity of logs without private keys.

5. **Defensible attacks**:

a) Truncation attack: the attacker may delete the trailing log items (records of malicious actions of attacker) before it is sent to storage system [42]. This is a special type of deletion attack.

b) Delayed detection attack: caused by time-consuming verification process (could be the verification algorithm or communication delay with verifier) the logging system is not able to detect log corruption promptly [38].

c) Reorder attack: the attacker may not able to create valid log signatures to generate new logs, but he may change the order of entries in a log sequence [87].

d) Insertion attack: the attacker may forge new items or duplicate items (by replaying) and insert it into or append it to the logs [87].

e) Modification attack: the attacker may change information of certain log items [87].

f) Deletion attack: the attacker deletes some or all log items [42].

6. **Security levels**: tamper-proofness is against modification, so it can prevent reoder attack, insertion attack and modification attack. And it makes modifications evident and discoverable, so it can also prevent delayed detection attack. However, the immutability property includes tamper proofness and adds deletion prevention as an extra layer of robustness.

a) Tamper-proof: it means logs are against modification. Some papers [89], [90] declaring their solutions as tamper-proof normally put emphasis on tamper detection (i.e., being tamper-evident). In addition, tamper-proofness does not prevent deletion attack.

b) Immutable: log data itself are immutable (i.e., can not be deleted or modified in any form) [88]. Immutability of integrity proof does not qualify as immutability here. If the data is immutable, then it is already tamper-proof.

Paper	Scheme	Technology	Security measures				Defensible attacks						Security level	
			forward security	data encryption	Secure retrieval	Public verifiable	Truncation attack	Delayed detection	Reorder attack	Insertion attack	Modification attack	Deletion attack	Tamper-proof	Immutable
(Schneier & Kelsey, 1999) [32] [33]	/	cryptography	✘	✔	✘	✘	✘	✘	✘	✔	✘	✘	✘	✘
(Bellare & Yee, 1997) [34]	/	cryptography	✔	✔	✘	✘	✘	✘	✘	✔	✔	✘	✘	✘
(Ma & Tsudik, 2009) [35]	FssAgg	cryptography	✔	✘	✘	✔	✘	✘	✔	✔	✔	✘	✘	✘
(Holt, 2006) [36]	LogCrypt	cryptography	✔	✔	✘	✔	✘	✔	✘	✔	✔	✘	✘	✘
(Yavuz et al., 2012) [37]	LogFAS	cryptography	✔	✘	✘	✔	✔	✘	✔	✔	✔	✘	✘	✘
(Yavuz et al., 2012) [38]	Fi-BAF	cryptography	✔	✔	✘	✔	✔	✔	✘	✔	✔	✔	✔	✘
(Kampanakis & Yavuz, 2015) [39]	BAFi	cryptography	✔	✔	✘	✔	✔	✔	✘	✔	✔	✔	✔	✘
(Hartung et al., 2017) [40]	/	cryptography	✔	✔	✘	✔	✔	✔	✘	✔	✔	✔	✔	✘
(Wang & Zheng,	/	hardware	✔	✔	✘	✘	✘	✘	✔	✔	✔	✘	✔	✘

2003) [41]														
(Chong et al., 2003) [42]	/	hardware	✘	✔	✘	✘	✘	✘	✘	✘	✔	✘	✔	✘
(Sinha et al., 2014) [43]	/	hardware	✔	✔	✘	✘	✘	✘	✘	✔	✔	✘	✔	✘
(Karande et al., 2017) [44]	SGX-Log	hardware	✔	✔	✔	✘	✔	✔	✘	✔	✔	✘	✔	✘
(Shepherd et al., 2017) [45]	EmLog	hardware	✔	✔	✔	✔	✔	✔	✔	✔	✔	✘	✔	✘
(Accorsi, 2011) [46]	BBox	hardware	✔	✔	✔	✔	✔	✔	✔	✔	✔	✘	✔	✘
(Ahmad et al., 2022) [50]	HardLog	hardware	✔	✔	✔	✔	✔	✔	✔	✔	✔	✘	✔	✘
(Snodgrass et al., 2004) [48]	/	3rd party	✔	✔	✔	✘	✔	✔	✔	✔	✔	✘	✔	✘
(Cucurull & Puiggalí, 2016) [49]	/	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
(Pawar et al., 2021) [51]	BlockAudit 2.0	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
(Sanchez et al., 2022) [52]	AuditTrust	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
(López Pimentel et al., 2021) [53]	RootLogChain	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

(Shekhtman & Waisbard, 2019) [54]	EngraveChain	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
(Ahmad et al., 2019) [55]	BlockTrail	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
(Wang et al., 2018) [56]	Medusa	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
(Ali et al., 2022) [57]	BCALS	blockchain	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

Table 4.1: Taxonomy table

4.5 Comparison

In this section, we will compare different generations of technology and discuss the reasoning process behind our choice.

4.5.1 Three generations of technology

If we look at the publication date of each paper (see Table 4.2), we can find the pattern that cryptography-based solutions were mostly discussed during 1997 and 2012, hardware-based solutions discussed during 2011 and 2017, and Blockchain-based solutions discussed since 2016 till to date:

	Cryptography-based	Hardware-based	Blockchain-based	3rd party service
1997	1			
1999	2			
2003	2	2		1
2006	1			
2009	1			
2011	1	1		
2012	2			
2014		1		
2015	1			
2016			1	
2017	1	2		
2018			1	
2019			1	
2021			2	
2022		1	2	

Table 4.2: Publication date of different solutions

In the table, only entries displayed in the taxonomy table (Table 4.1) are counted. If we include all search results, the pattern of different generations will be more evident.

4.5.2 Rationales for decision

Firstly, It is hard to pick an unbiased third party notarization service that could be trusted by the public, also the technology used by the third party is not transparent. So in our case we would avoid to use it.

Secondly, the cryptography-based solutions are widely adopted from 1998 ~ 2010, a lot of best practices could be found in the academic materials. But some recent researchers have successfully attack logging technology scheme (e.g., LogFAS [37], FssAgg [35]) once considered to be safe [58]. Because the security of cryptographic-based solutions come from the robustness of algorithms, once attackers found a smart way or have access to sufficiently abundant computing resources, the algorithm will be corrupted.

Hardware-based solutions add an extra layer to cryptographic-based solutions, but they either design a specific hardware or rely on secure hardware products. The managing of hardware causes inconveniences in the current trend of remote working. And also the hardware could be corrupted by physical attack.

We can compare different properties of these typical solutions (also see Table 4.3):

- **Security level**: Cryptography-based solutions are constantly facing new attack mechanisms, they could be corrupted when attacker find a way attack the cryptographic algorithms. Blockchain-based solutions promise immutability, they can achieve highest level of security. Hardware-based solutions used secure hardware platform to enhance cryptography-based solutions, so they are more secure than pure cryptography-based solutions. Technology used by third party service is not transparent, so it is regarded as low security level.
- **Ease of use**: third party service is ready to use, so it is the most convenient one. Blockchain offers immutability automatically, implementing extra functions with smart contract is also convenient. Developing cryptography-based

solutions require heavy cryptography background knowledgy, so they are harder than Blockchain-based solution. Hardware-based solutions require managing, maintaining, sometimes designing specialized hardware, so this type of solution are considered the most difficult to use.

- **Costs**: cryptography-based solutions are the most cost-effective, once a robust technology scheme is designed it can be used forever with nearly no extra cost. Costs are incurred for each storage when using third party services or Blockchain-based solutions. Depending on the pricing standards, third party services could be more expensive than Blockchain. Hardware-based solutions generate costs for every hardware deployed, so they are also more expensive than cryptography-based solutions.

- **Performance**: hardware-based solutions spend shorter time to process log information thanks to the integrated secure hardware, so they are more performant than pure cryptography-based solutions. Paid third party service offers ready to use APIs, it should also be quite performant. The bottle neck of Blockchain limits processing speed of Blockchain-based solutions, so they are less performant.

	Hardware-based	Cryptography-based	Blockchain-based	3rd party service
security level	★ ★	★	★ ★ ★	★
ease of use	★	★ ★	★ ★ ★	★ ★ ★
costs	★ ★	★	★ ★	★ ★ ★
performance	★ ★ ★	★ ★	★ ★	★ ★ ★

Table 4.3: Comparison of different solutions.

In this thesis we will first design our solution based on the assumption that cost and performance are not issues. Later in the "**8.2 Future improvements**" chapter we will re-visit these issues and provide potential solutions.

4.5.3 Why Blockchain is secure?

Quite similar to what the cryptography-based solutions were doing, Blockchain also connects a sequence of data blocks together using a hash algorithm. However, there is a huge decentralized miner network maintaining the data blocks [64]. Nodes (peer miners) in the network communicate with each other using consensus algorithm. Only when an attacker controls more than half of the nodes, it is possible to tamper the newly appended data blocks [91]. This is barely impossible in the real world. Bitcoin used proof of work (PoW) as the base of consensus algorithm [64], while some other Blockchains use proof of authority (PoA) or proof of stake (PoS) [80].

4.5.4 Which type of Blockchain to use?

The two big genre of Blockchains are: private Blockchain and public Blockchain (for more details, see **3.2 Blockchain related terms**). In this thesis we will use public Blockchain out of our request for public verifiability. Any user can easily query block information from the Blockchain, without the need for being authorized by the Blockchain first.

4.5.5 Which public Blockchain to choose?

We favour secure public Blockchain over others. The security of Blockchain comes from its nature of decentralization. To certain degree, the more participants join in the Blockchain network's activities, the more secure it is. Total Value Locked (TVL) is one of the main metrics investors and analysts used to represents the dollar value of digital assets staked in the Blockchain. The higher the TVL, the more trustworthy the platform or DApp is perceived to be. Ethereum ranks number one among all Blockchains and reaches more the 50% of entire TVL [92]. Ethereum Blockchain is one of the safest public Blockchain. It provides a complete ecosystem for developing decentralized applications (DApp).

4.5.6 What are the drawbacks of Ethereum Blockchain?

The major deficiency of Ethereum Blockchain is the cost of transactions. When doing the concept design and implementation, we assume that Ethereum Blockchain is cheap enough. We will propose potential solution to solve this issue in "**8.2 Expensive transaction cost**".

5 CONCEPT DESIGN

In this chapter, we first review the legal basis of data trustee platform, then discuss the underlying mechanism of an representative data trustee platform for sleep research, and do requirement engineering for different stakeholders connected to the platform.

5.1 Legal basis

The legal enforement pressure is one of the main reasons that the data trustee platform need to enhance their transparency of data processing. So it is essential to understand how the laws regulate the platform. First let's take a look how the primary mechanism — dynamic consent takes shape (Fig. 5.1).

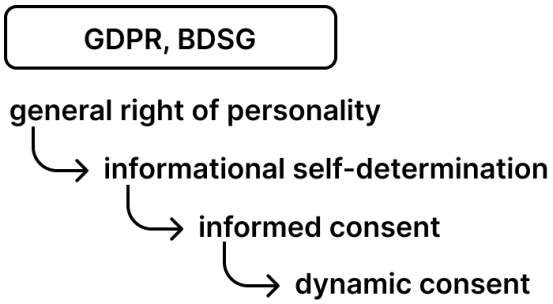

Fig. 5.1: How "dynamic consent" takes shape.

One of the aims of the General Data Protection Regulation (GDPR) is to empower individuals and give them control over their personal data (Art. 1 Para. 1 GDPR) [65]. GDPR Information self-determination is a special expression of the general right of personality (Art. 2 Para. 1 GDPR) [65]. Whenever an organization uses one's personal data, it should get informed consent from this person frist. "Informed" means data subjects are fully informed when making their decisions to give away personal data.

There are three types of informed consent (Fig. 5.2):

1. **Narrow consent**: tends to protect one's data, and decides to share only in cases involving your own medical treatment.
2. **General consent**: tends to contribute their data in the aim of accelerating medical research of certain field.
3. **Dynamic consent**: dynamically respond to whether contribute or not in real time with the help of information system.

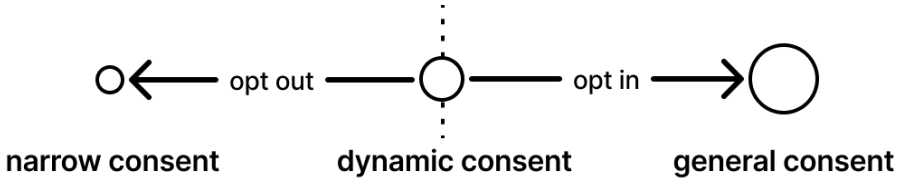

Fig. 5.2: Different types of consent.

Because of always changing nature of dynamic consent, the data trustee platform comes to gather and manage consent. This is one primary reason and responsibility of these kind of platforms. The data trustee platform acts as an intermediatary between data donators and data users. On the one hand, it allows data donators to flexibly express their preference of donation; on the other hand, it allows data users to conveniently get access to prcessed, large amount of data.

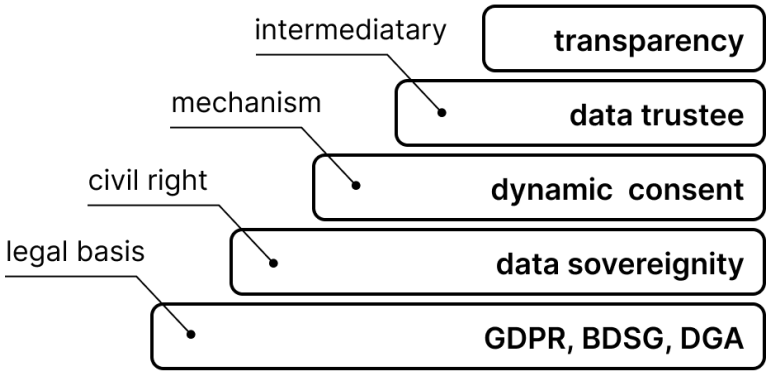

Fig. 5.3: Pyramid of legal ground.

Our transparency topic is based on the data trustee platform. The mechanism of data trustee platform can trace back to the consent type that well supports data sovereignty. Good support for data sovereignty realizes the requirements from the laws. (Fig. 5.3)

5.2 How data trustee for sleep data works

In the case of data trustee for sleep data, the sleep data was previously scattered among different clinics or sleep labs. Sleep researchers either only have access to a relatively small amount of data, or need to spend a lot of time and effort to gather various datasets from different clinics. The data trustee platform connects to all clinics that agreed to corporate, and get the data consent from the donators. When a researcher needs to use data to do experiment, the platform will gather data from available clinics and send the required data set a black box (normally a docker container) (Fig. 5.4).

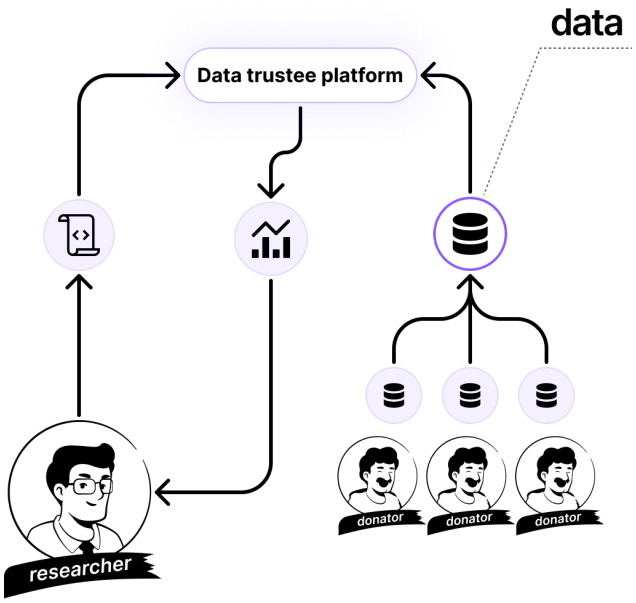

Fig. 5.4: Pipeline of data gathering

The interactions between data users (e.g., the researchers) and the platform are limited. To use the data from donators, the researcher need to sign the usage

policy first. The usage policy requires researcher to agree on the scope of use, any other usage behaviours obey the usage policy are not possible. When using the data, researchers send the data analysis algorithm or machine learning model-training algorithm to the platform. The platform will run the algorithm in the black box and return the analysis or training result to researchers. (Fig. 5.5)

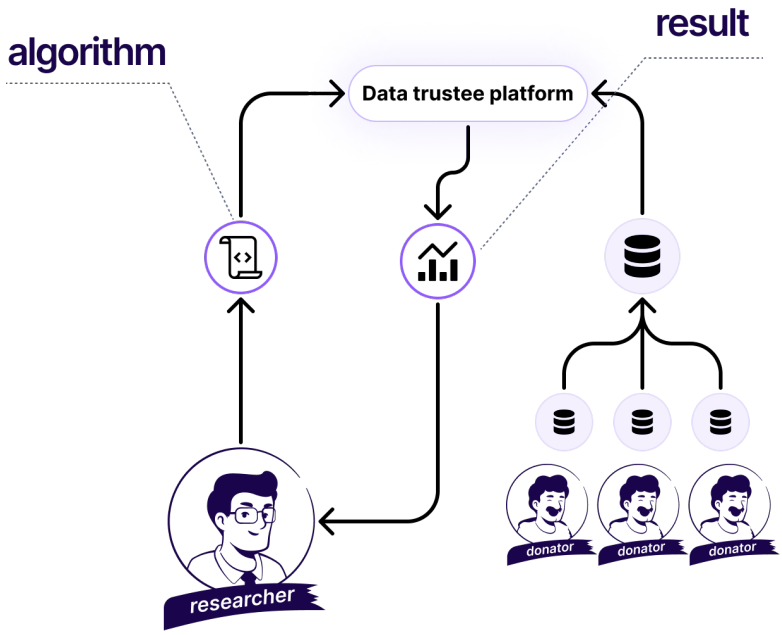

Fig. 5.5: Restricted interactions with platform

In such way, researchers do not have direct access to the data; they are only using data in an indirect way. Another security measure the platform takes is to cache the data from multiple locations only during a limited period. There is no single central database built by the platform to host all data. Donators' data remains at where they were.

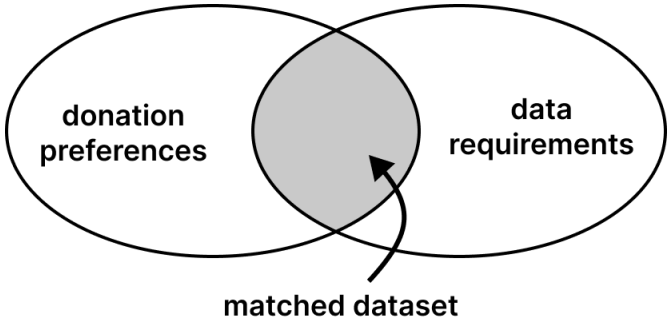

Fig. 5.6: Data matching

One big part of data trustee for sleep data's job is match data for researchers. The platform receives data requirements from sleep researchers, then gathers available data from donators and make sure do not break the preferences of those donators. (Fig. 5.6)

5.3 Overview of all stakeholders

Now that we have introduced the legal basis and operating mechanisms of data trustee for sleep data, we can go further to look at all the connected stakeholders to the platform and the requirements from their perspective (Fig. 5.7).

We already know that the platform sits in between data donator and researchers; it works like a bridge connecting those two parties. What we did not cover is doctors from the labs (or sleep clinics), they help data donator to publish their data to the platform. Only after data being published, the platform is able to scan and match the datasets.

In addition, there is a group of managers from the platform itself. They developed and deployed the platform, and are responsible for the daily operating of the platform. One of the biggest responsibility for them is to honour the data protection laws. They need to observe activities happening within the platform and discover any potential misbehaviours. We can also address them as the platform auditor.

There are also auditing pressure from the outside: the legal enforcement agencies with a government background, and the research community. These two parties do not directly involved with the data consuming events, but they oversees the platform and data users on the platform to make sure donators' data is in good hands. We will refer to these two types of stakeholders collectively as third parties.

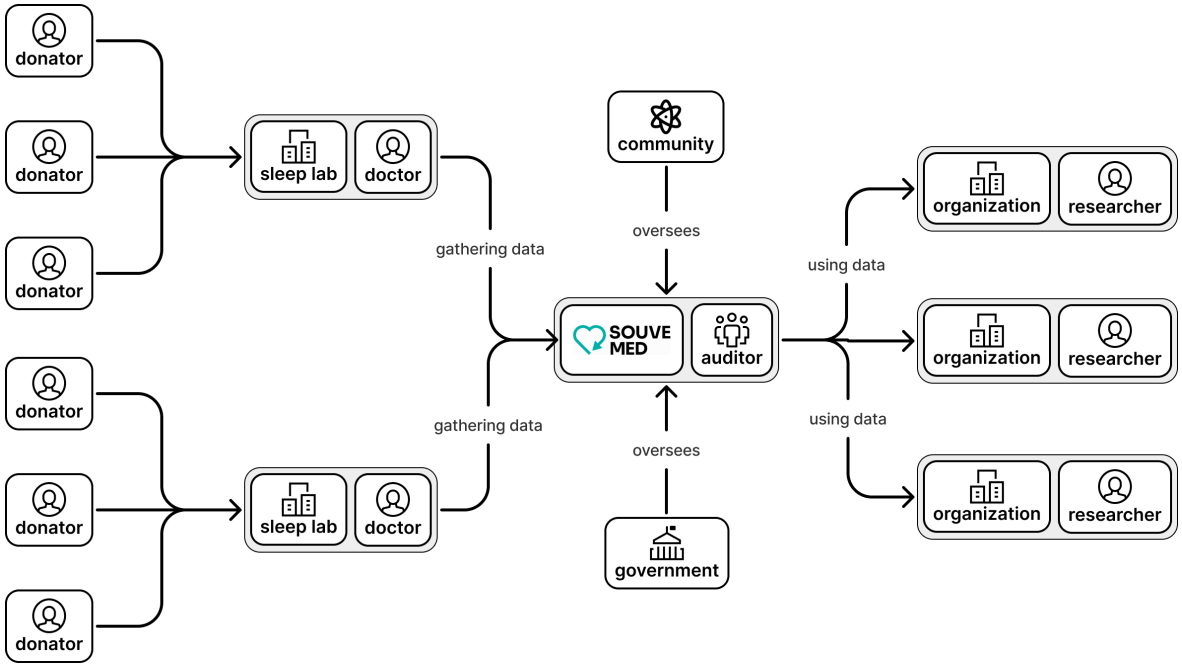

Fig. 5.7: Full view of all stakeholders

One big part of requirements of each stakeholder could be inferred from how the platform works. There are only a limited set of event and information exchange happen with the ecosystem. Requirements of stakeholders should not go beyond the context they exist. Another part of requirements comes from literature research.

5.4 Personas

In this section, we will build a simplified persona for each stakeholder. The persona technique [70] can summarize their representative characteristics and embody the image of them.

- **Researcher**: I value data privacy. It will be great if the platform can store proofs of our usage records, so that we have something to depend on when facing disputes.

- **Platform auditor**: I can already query all the log data in the database, but it is not that convenient. Moreover, we need to make sure the log data should never be modified in any circumstances.
- **Donator**: I hope the researchers are doing meaningful science study with my data. The platform should keep my data safe, and also explain how it's processed to me.
- **Doctor**: I encourage my patients to upload their datasets. I feel responsible for keeping their data safe.
- **Third parties**: Misuse of data happens all the time. We need to be able to examine usage records and other information to make sure we are doing it right.

Platform auditors are part of the team that develops and manages the platform; they already have access to all necessary information. Doctors are not direct contributor and owner of the data, so the priority of this type of stakeholder is also lower. (Fig. 5.8)

Fig. 5.8: User priority

The two most important stakeholders are data donator and third parties. The major reason researcher is not included here is that data trustee for sleep data platform has already cover the requirements for querying and displaying usage records generated by themselves, and checking integrity proof of their usage records can be achieved through the website prepared for the third parties.

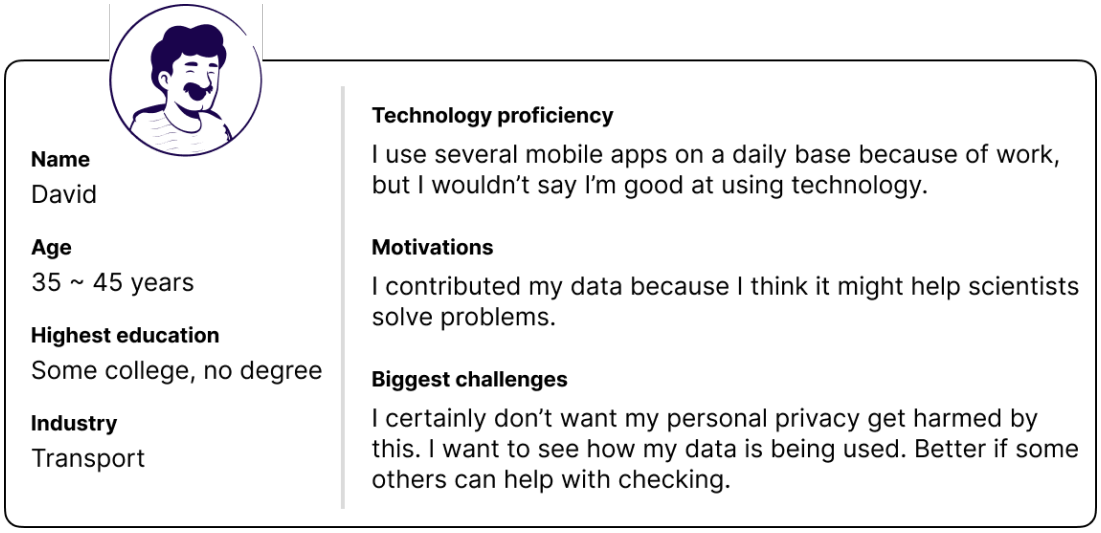

Fig. 5.9: Persona of donator

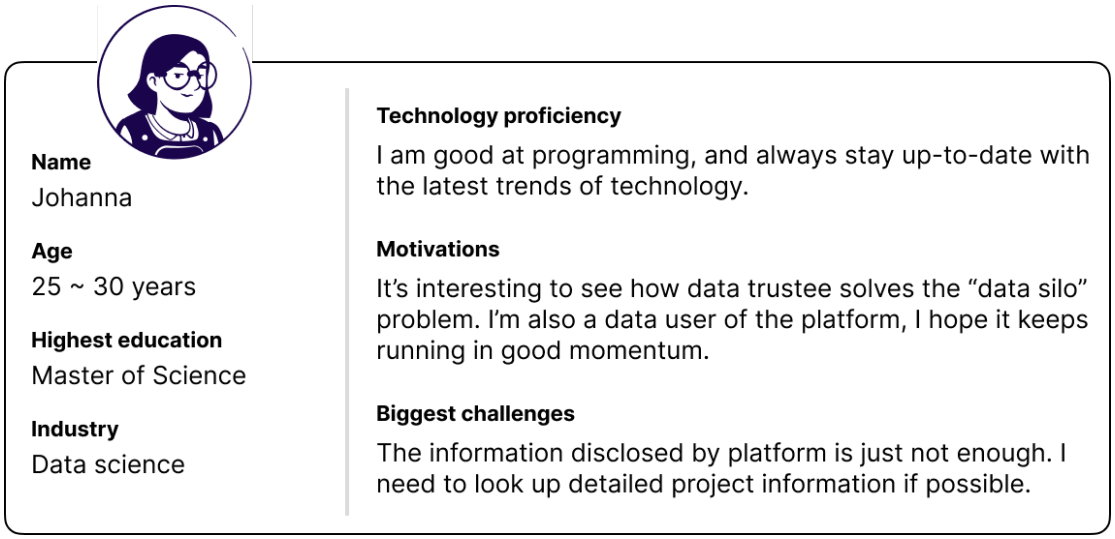

Fig. 5.10: Persona of researcher

5.5 Classification of requirements

In the following sub-sections, we will describe user stories in non-technical natural language. By creating the user stories (according to definition by Sommerville [77]), we are putting stakeholders at the center and providing context about how they

might want to interact with the system. At the same time, we use a MoSCoW technique [68] to prioritize the user stories of each stakeholder based on the following criteria:

- "Must Have" (**M**): defines the requirements that must be included.
- "Should Have" (**S**): high priority requirement should be included if possible.
- "Could Have" (**C**): desirable or nice to have requirements, could be included without incurring too much effort or cost.
- "Won't Have This Time" (**W**): requirements that could be meaningful in the future but not urgent at all.

5.5.1 Data donator

Data donators are the direct owner of the contributed data. They have the right to actively contribute their data; to express their preferences of sharing; to examine how their data is being used; and also decline data request.

- **M1**: As a data donator, I want to look up the detailed information of datasets I have published to the platform, so that I know exactly what I am contributing.
- **M2**: As a data donator, I want to review my consent settings of datasets, so that I know who I am contributing to.
- **M3**: As a data donator, I want to browse usage records of my datasets, so that I know who have used my data in what kind of ways.
- **S1**: As a data donator, I need easy-to-understand explainations about what my data has went through, so I can understand how the platform processes my data.
- **S2**: As a data donator, I want to browse the detailed information of the organization that researcher belongs, so that I know who benefits from my data.

- **S3**: As a data donator, I want to browse all the history usage records of the organization that researcher belongs, so that I know the overall research target of the organization.
- **C1**: As a data donator, I want to adjust my consent settings, so that I can express my preferences for sharing data.
- **C2**: As a data donator, I want to upload and publish my datasets, so that I can actively contribute my data.
- **C3**: As a data donator, I want to get reward from the researchers, so that I can be motivated to do contribution.
- **C4**: As a data donator, I want to allow doctors to manage my consent settings, so that I do not have to learn to manage the data.
- **W1**: As a data donator, I want to get notified when non-compliant use happens, so that I respond to it quickly.
- **W2**: As a data donator, I want report potential violation to the platform when I see suspicious activities.

5.5.2 Researcher (data user)

Researchers are direct users of the data. They use the data to do big data analysis or train machine-learning models with a target of making progress in the area of sleep research.

- **M4**: As a researcher, I want to look up my own usage records, so that I know my recorded activities on the platform.
- **S4**: As a researcher, I want to be able to check the integrity proof of usage records, so that I can defend myself when being reported as violating regulations.

- **C5**: As a researcher, I want to be able to answer auditing questions from the third parties or the donators, so that I can avoid misunderstanding and legal dispute.
- **C6**: As a researcher, I want to publicize the algorithms used to analyse the data and results of data analysis, so that my research activities are more open and helpful to the public.

5.5.3 Doctor

Since current the datasets are generated from sleep monitoring devices in the clinics or lab, doctors are involved as well. Doctors form the clinic or professionals from the sleep laboratory help data donators to upload the datasets to the platform.

- **M5**: As a doctor, I want to look up the datasets I have helped with uploading, so that I can answer queries about the contribution from donators.
- **C7**: As a doctor, I want to adjust the consent settings of datasets on behalf of donators when I am authorized by donators to do so.

5.5.4 Platform auditor

Different from third party auditors, platform auditors are a group of internal auditors. They bear the largest responsibility of making sure the transparency of the platform and preventing violations within the platform. Data breaches or violations can ruin the platform's reputation and bring higher audting pressure from third parties. Only if donators' data is well protected, the platform may enter a benign cycle of growth in the long run.

- **M6**: As a platform auditor, I want to browse all usage records generated within the platform, so that I can audit researchers' activities.
- **M7**: As a platform auditor, I want to look up detail information of a specific usage record, so that I can investigate context information of the research project.

- **M8**: As a platform auditor, I want to be able to check integrity proof of usage records, so that I can be sure whether a researcher violated donators' consent.
- **S5**: As a platform auditor, I want to analyse the statistics of data usage events, so that I can have an overview of activities within the platform.
- **C8**: As a platform auditor, I want to receive violation reports from the third parties or donators, so that I can uncover non-compliant data use and stop further violations immediately.

5.5.5 Third parties

Requirements from this group of stakeholders are mostly about auditing. Although expressions of "M9 ~ M11" are the same as "M6 ~ M8", they should be listed as different requirements from the perspective feature implementation. "C9" is related to reproducibility of research result.

- **M9**: As a third party auditor, I want to browse all usage records generated within the platform, so that I can audit researchers' activities.
- **M10**: As a third party auditor, I want to look up detail information of a specific usage record, so that I can investigate context information of the research project.
- **M11**: As a third party auditor, I want to be able to check integrity proof of usage records, so that I can be sure whether a researcher violated donators' consent.
- **S6**: As a third party auditor, I want to report violation events to the platform and data donators, so that further violations could be stopped in time.
- **C9**: As a third party auditor, I want to be able to follow up steps taken by a research project, so that I can reproduce the research result and validate researcher's publications.

5.6 Requirements selection

Together with considerations from "**5.4 Personas**", we will first focus on fulfilling the high-priority requirements from the two most important stakeholders: data donator and third party auditor.

Donators' requirements	Information display	Data management	Data consent managent	Feedback mechanism	Event notification
Requirement coding	M1, M2, M3, S1, S2, S3	C2	C1, C4	C3	W1

Table 5.1: Dividing donators' requirements into groups.

Third party auditors' requirements	Information display	Auditing	Research community	Event reporting
Requirement coding	M9, M10	M11	C9	S6

Table 5.2: Dividing third party auditors' requirements into groups.

In Table 5.1 and 5.2, we grouped their requirements into sub-categories. Our UI prototype in an early stage will first solve "information display" requirements first. And also implement "data management" and "data consent management" for donators. "Auditing" is a core requirement for auditors, so we will also cover this one. "Feedback mechanism" and "Research community" are less connected to our transparency topic, so they will be partially supported. "Event notification" and "Event reporting" could be implemented in later iterations.

5.7 UI prototype

We used Figma to design interactive UI prototype and implemented requirements from previous section. In this section we will review some key pages taken from the UI prototype and discuss how we have realized the requirements with the functions and information presented by UI.

The list of screenshots:

- **Fig. 5.13 (donator's app)**: In this screenshot, menu "My dataset" is selected, donators can browse dataset information here. This screenshot is related with requirement **M1**,
- **Fig. 5.14 (donator's app)**: In this screenshot, menu "Consent management" is selected, donators can manage data consent here. This screenshot is related with requirement **M2**,
- **Fig. 5.15 (donator's app)**: In this screenshot, menu "Usage records > by dataset" is selected, donators can browse usage records of selected dataset here. This screenshot is related with requirement **M3**,
- **Fig. 5.16 (donator's app)**: In this screenshot, menu "Usage records > project list" is selected, donators can browse usage records of with filter options here. This screenshot is also related with requirement **M3**,
- **Fig. 5.17 (donator's app)**: In this screenshot, menu "About Souvemed" is selected, donators can see illustration of platform mechanism here. This screenshot is related with requirement **S1**,
- **Fig. 5.18 (third party auditor's app)**: In this screenshot, menu "Project list" is selected, auditors can browse usage records here. This screenshot is related with requirement **M9**,
- **Fig. 5.19 (third party auditor's app)**: In this screenshot, menu "How to audit" is selected, auditors can see auditing method here. This screenshot is related with requirement **M11**,
- **Fig. 5.20 (third party auditor's app)**: In this screenshot, detail information of a usage record is displayed. Auditors can see what kind of data is used in the project and detailed experiment setting here. This screenshot is related with requirement **M10**,

- **Fig. 5.21 (third party auditor's app)**: In this screenshot shows the detail page of a usage record. Auditors can check integrity proof of usage record here. This screenshot is related with requirement **M11**.

5.7.1 Overall structure

All the UI pages for data donators follows similar page structure. A header on the top, menu bar on the left side, the space left belongs to main content. Often, the main content is further divided into two columns. The left column may used for secondary navigation or holding filter options. (Fig. 5.11)

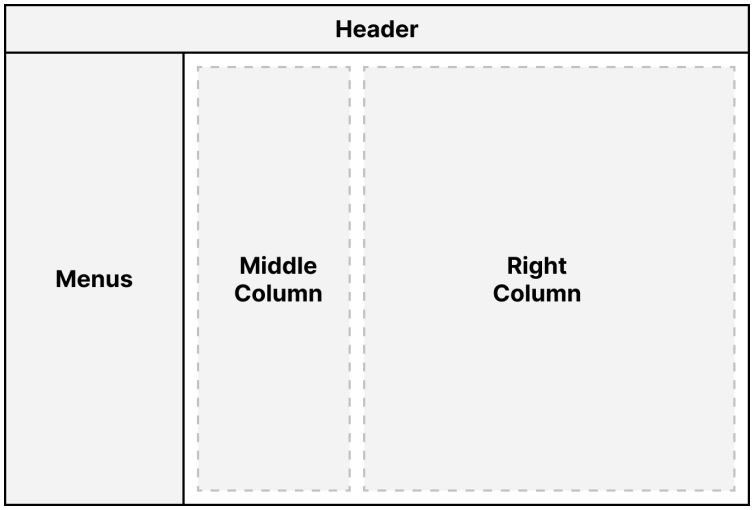

Fig. 5.11: Three-column layout (donator's app)

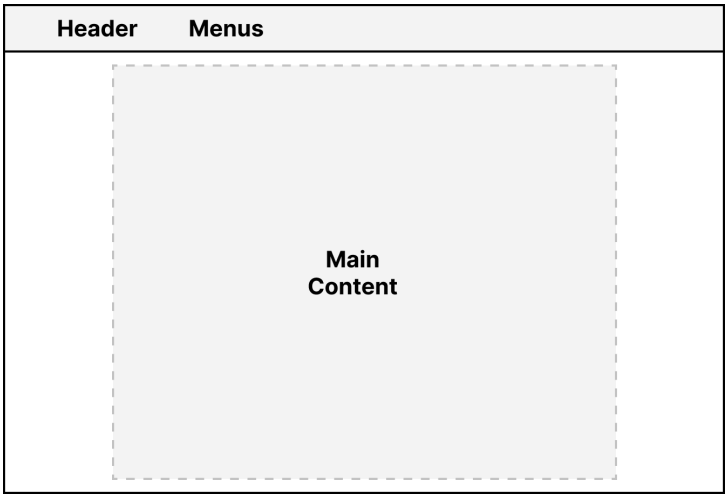

Fig. 5.12: Simpler layout (third party auditor's app)

The layout of third party auditor's app (Fig. 5.12) is a bit more simpler than data donator's. Since currently we only prepared limited features, so the menus are merged into the header area, and vast majority of page space is contributed to main content for ease of use.

5.7.2 Donator's app

The donator can see all datasets he uploaded to the platform. What data each dataset includes is displayed here. (Fig. 5.13)

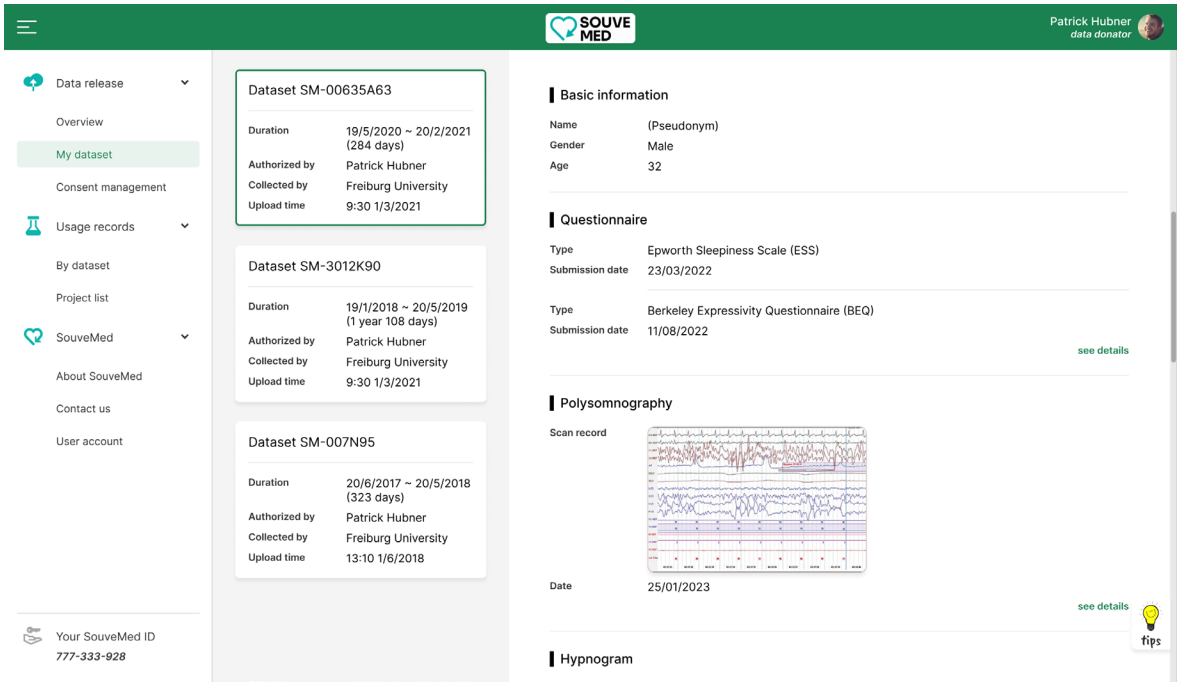

Fig. 5.13: Browsing a list of datasets and detail information of the selecteddataset (donator's app)

Donator can publish their datasets and set their preference of data sharing. Public research project, or private project; whether the research result will be shared with the donator or not are some possible setting options. Donator can always review and change settings in this page. (Fig. 5.14)

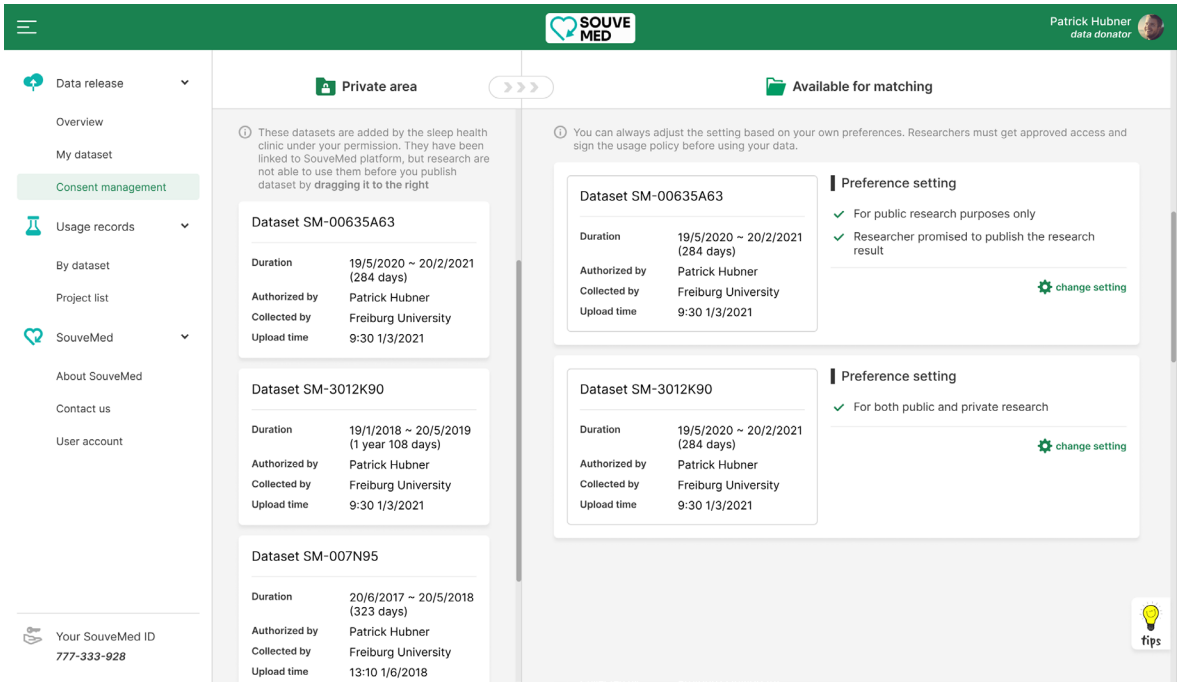

Fig. 5.14: Managing the preference setting for the dataset (donator's app)

There are two browsing modes. Donator can browse usage records by dataset or search through the projects if they want to find out some particular information. The project list mode provides convenient filtering options, but requires the user to be more proficient at using the app.

Why certain dataset is used in a research project is also explained with "preference setting" and "data requirements" information in the detail page. If the researcher promised to share search results, it will also be display here. This is a way donators get rewarded for their contributions.

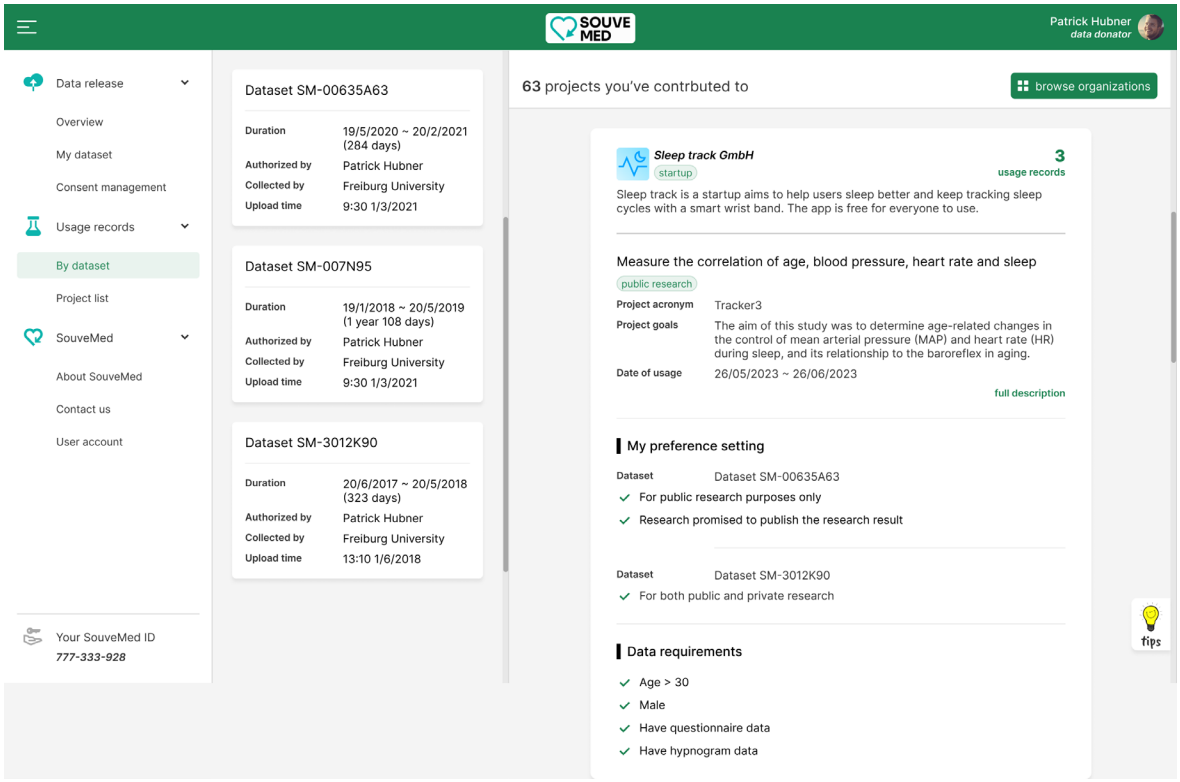

Fig. 5.15: Browsing usage records by dataset (donator's app)

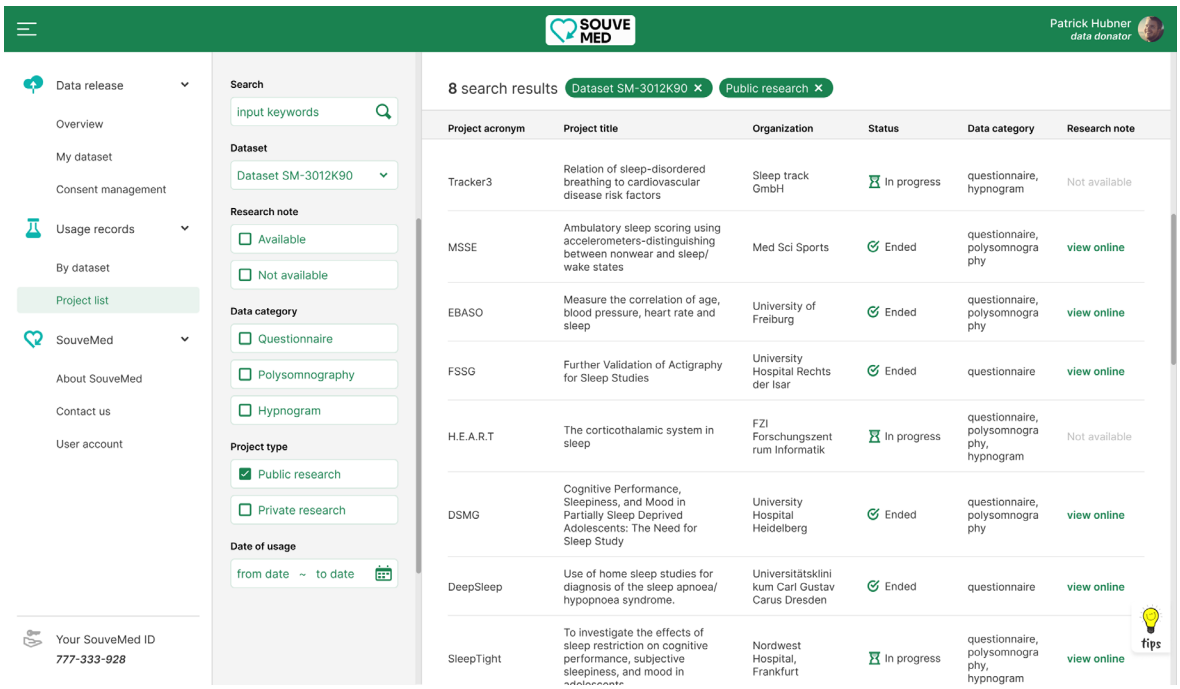

Fig. 5.16: Filtering usage records with abundant filter options (donator's app)

Donators can grasp how data trustee for sleep data handles their data through easy-to-understand illustrations and explanations written in plain language. We avoided using technical terms and exhaustive writing, instead we used flash cards to hold bite-sized information. In each flash card, there is title, a short description of the card and vivid graphic drawing. (Fig. 5.17)

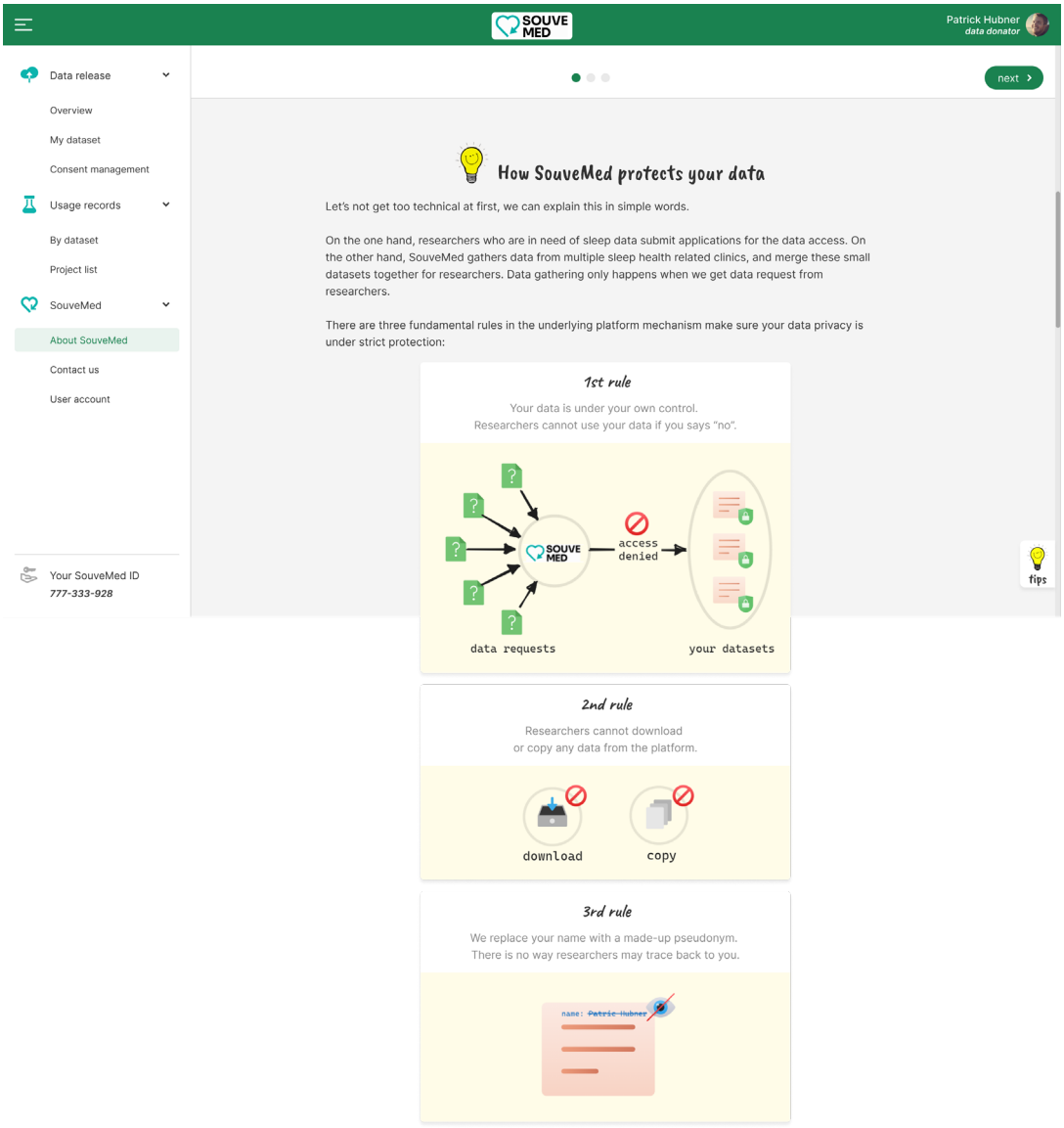

Fig. 5.17: Easy-to-understand illustrations (donator's app)

5.7.3 Third party auditor's app

The features of this page (Fig. 5.18) is quite similar to "project list" mode of donator's app (Fig. 5.16). Third party auditors can quickly find a usage record they hope to look in detail with convenient filter options on the left.

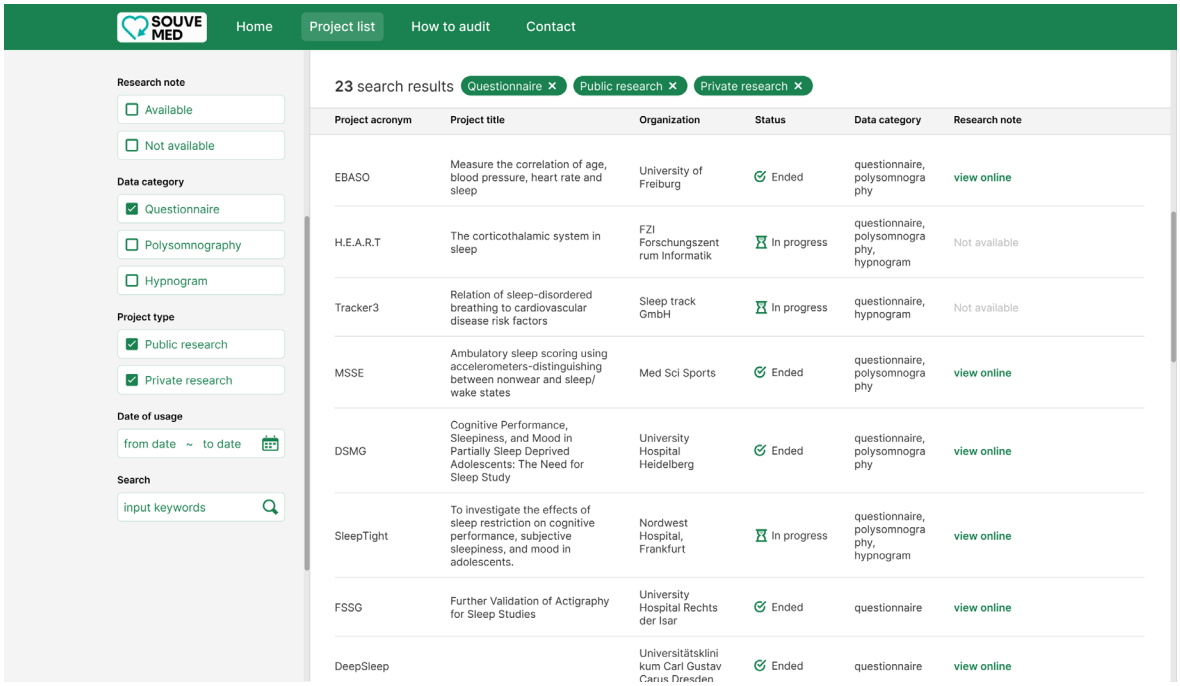

Fig. 5.18: Browse all project usage records (third party auditor's app)

This is a succinct guide about how to use this system to audit usage records (Fig. 5.19). The underlying mechanism of how the Blockchain helps to protect integrity of usage records will be explained with more details in the "**6 Implementation**" chapter.

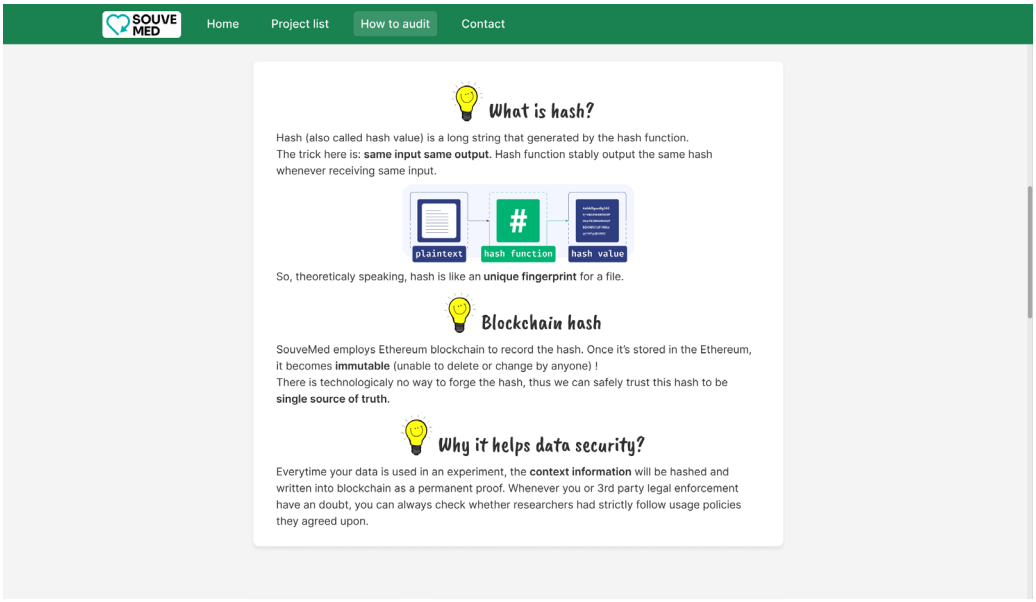

Fig. 5.19: Explanation about checking integrity of usage records (third party auditor's app)

The detail page of a usage record (Fig. 5.20) includes data requirements of this project, and also the experiment setting. Third party auditors can know what kind of data the platform has prepared for this project. Researchers from sleep research community can try to reproduce the research result with the algorithm file, project goals, and research notes here.

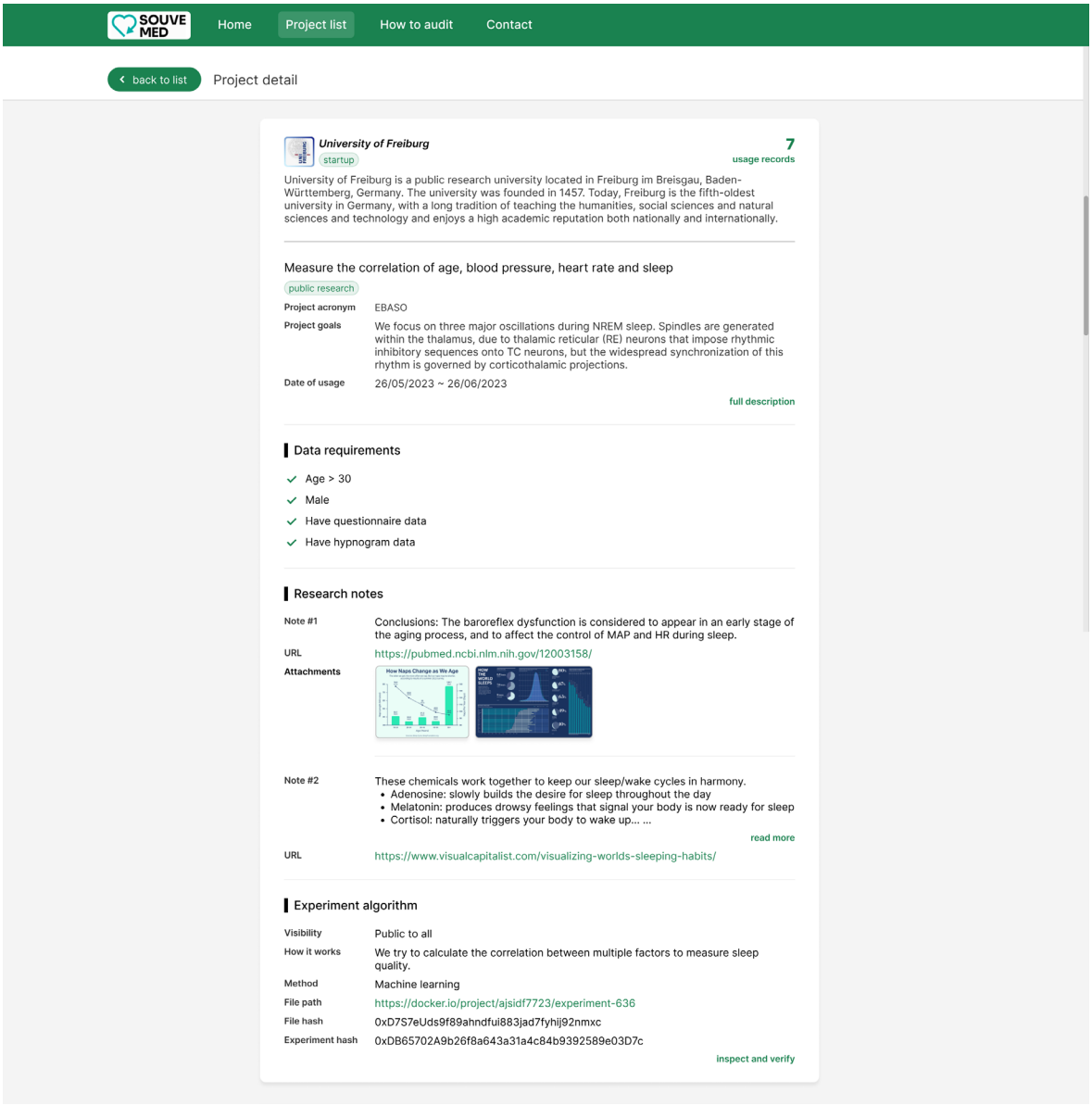

Fig. 5.20: Detail page of one usage record (third party auditor's app)

If clicking "**inspect and verify**" in the bottom of screenshot Fig. 5.20, the page jumps to Fig. 5.21. Here auditors can verify whether the core information of a usage record is authentic.

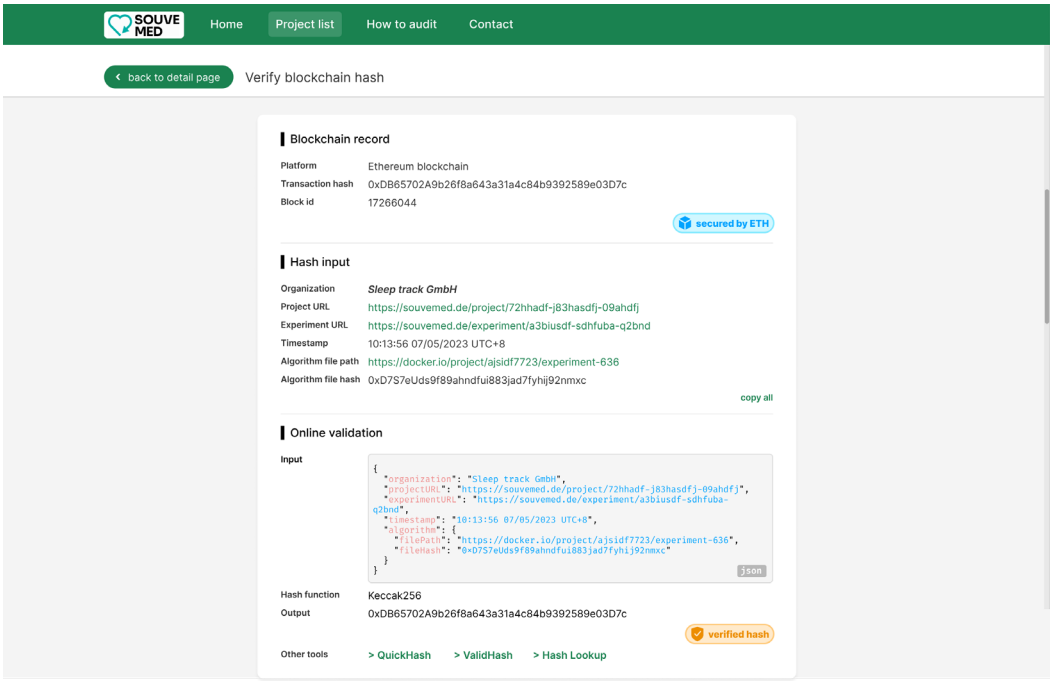

Fig. 5.21: Checking integrity of usage records with the help of Blockchain (third party auditor's app)

6 IMPLEMENTATION

In this chapter, we will introduce underlying system architecture of the mockup system, different technologies we used to build the system, how different system components work with each other. We will also highlight the design of smart contract and explains why it is the backbone of our logging system.

6.1 System architecture

With the three research questions bear in mind, we designed a three-layer system architecture: the frontend layer, the backend layer and the Blockchain layer. The frontend layer is implemented with web technology. It is responsible for displaying information and offering interactivity to users. The backend is responsible for handling data usage events on the one the hand, retrieving data usage information and integrity proof on the other hand. The Blockchain is responsible for storing and checking integrity proof of log information.

6.1.1 Frontend layer

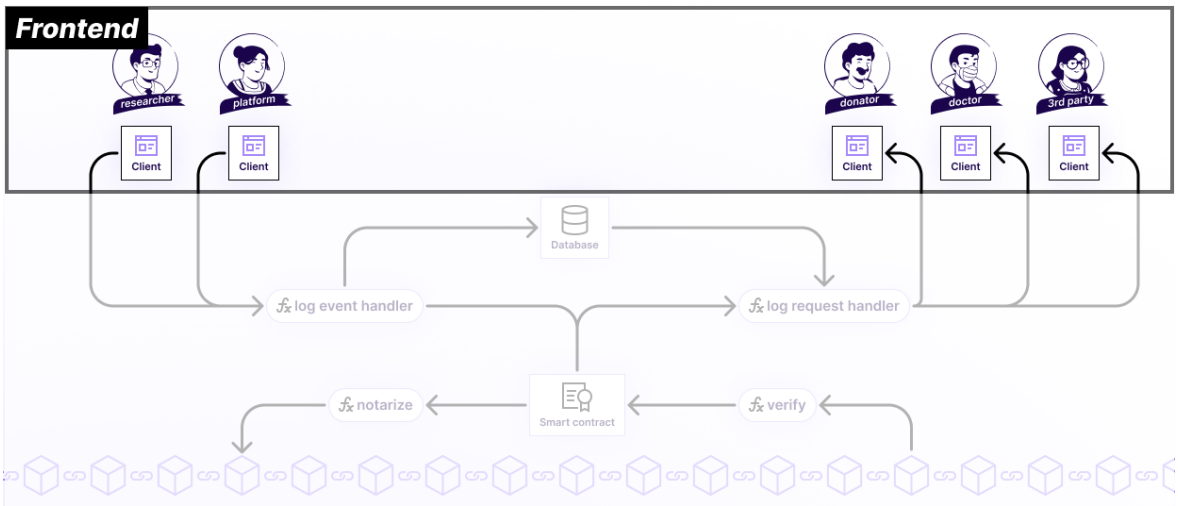

Fig. 6.1: Dedicated client apps for different stakeholders.

As previous chapter "**5 Concept design**" has discussed, each type of stakeholder has their own specific sets of requirement. The information they hope to see, features they need to use, habits of using application are different from other stakeholders. To convey the information effectively, we prepare dedicated client apps for each of them (Fig. 6.1).

The angle of each client app is different. Client app for donators focuses on visualizing usage records and other important log information with user-friendly interfaces. Client app for auditors focuses on offering tools facilitate auditing.

6.1.2 Backend layer

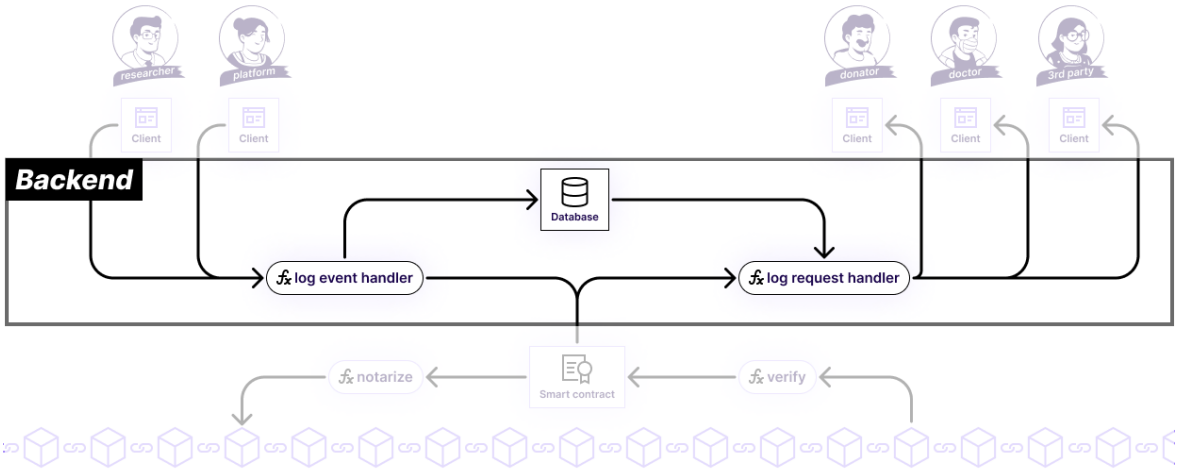

Fig. 6.2: Handling new log events and log retrieval request.

The backend receives log events on the one hand, and responds to log view request on the other hand. In addition, it stores core information into the database and immutablize the integrity proof of core information with Blockchain (Fig. 6.2). Whenever the system needs to notarize a new usage record or verify the integrity of usage record, the backend will interact with Blockchain.

6.1.3 Blockchain layer

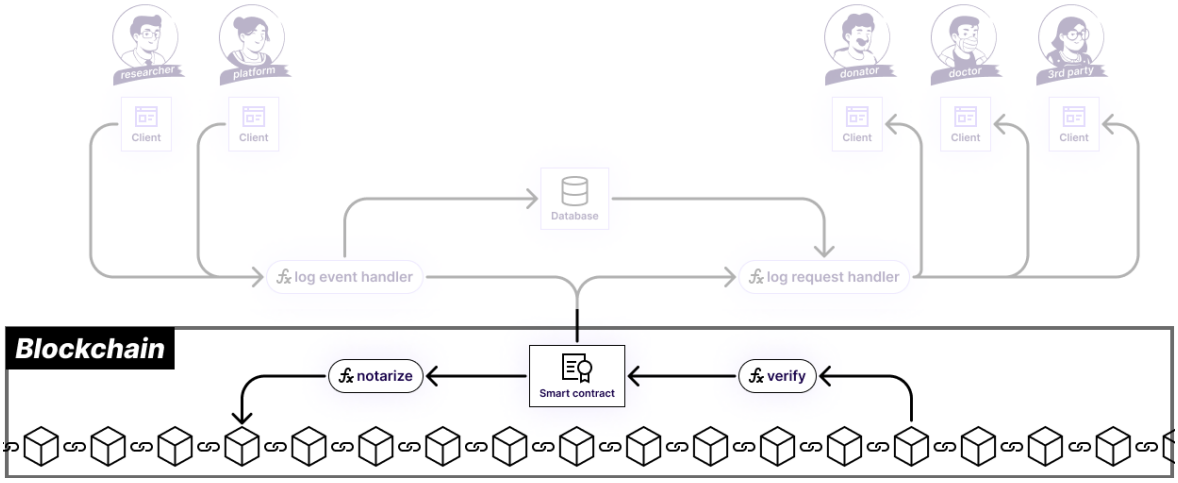

Fig. 6.3: Smart contract hosted in the Blockchain.

We have deployed a smart contract in Ethereum Blockchain. The smart contract exposes two important API to the backend layer (Fig. 6.4). The blockchain layer collaborates with database in the backend layer to make sure the integrity and authenticity of usage records.

6.2 Data storage

6.2.1 Design of database

Since this is only a mockup system, we only created one table in the database (Fig. 6.3). This table will store all related information about the usage records, each table record equals to a log item. The Log table in the database is comprised of five parts:

- **Meta data**: this includes meta information of the table record, e.g., uinique id of the record (**id**), creation time of the record (**createdAt**), update time of the record (**updatedAt**),

- **Project information**: this includes various information about the research project, e.g., type of the project (**projectType**), title of the project (**projectTitle**), acronym of the project (**pojectAcronym**), and URL of the project information page (**projectURL**). Possible data categories are questionnaire data, polysomnography data and hyponogram data,

- **Experiment information**: this includes various information about the experiment, e.g., execution time point of the experiment (**timestamp**), hardware or software resources used to run the experiment (virtualResource), algorithm code used to process data (**code**), file path of the algorithm (**filePath**), hash value of the algorithm file (**fileHash**), and different data categories used in this experiment (**dataCategory**). To accomplish a research project, the researcher may run different experiments multiple times,

- **Core information**: there is a field called "**hashInput**" in the table, this field summarizes all important fields of each experiment into a JSON object. More details about core information in the next section "**6.2 Immutable core information**",

- **Transaction hash**: every transaction of smart contract will generate a unique transaction hash. This transaction hash (**transactionHash**) is also recorded for later use.

```
model Log {
  id              String   @id @default(cuid())
  createdAt       DateTime @default(now()) @map("created_at")
  updatedAt       DateTime @updatedAt @map("updated_at")
  hashInput       Json
  organization    String
  projectType     String
  projectTitle    String
  projectAcronym  String
  timestamp       String
  virtualResource String
  code            String
  filePath        String
  fileHash        String
  projectURL      String
  experimentURL   String
  dataCategory    String
  transactionHash String
}
```

Fig. 6.4: Design of the log table (defined with Prisma's schema language)

6.2.2 Core information

This fragment of JSON code (Fig. 6.4) is the core information that will be notarized by the Blockchain. It includes detailed context information of a usage record:

- **Who**: the organization behind researcher,
- **When**: the time when usage event (e.g., execution time of the experiment) happened,
- **What**: the specific ways researchers took to process the data,
- **Why**: the objectives of research project, data requirements and usage policy researchers signed before getting access to data.

Fig. 6.5: Data structure of core information

The core information is generated according to the context when researchers run the data analysis experiment. Our smart contract notarizes the core information and generates the integrity proof (e.g. a hash value stored in the blockchain) of each log item. (Fig. 6.5)

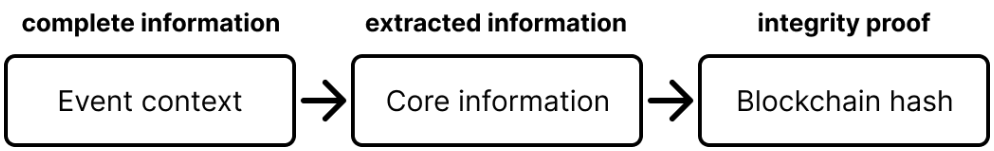

Fig. 6.6: The generation processes of integrity proof.

6.3 Smart contract

The two exposed functions available for external systems to call are “notarize” and “verify”, other functions inside the smart contract are utility functions used by these two functions.

6.3.1 Notarize

Notarize function receives the input string and transform the input string into a hash value. The result hash value will be recorded in the “proofs” array.

Only the contract deployer can call the “notarize” function. This prevents unwanted fake log to be created. To be recognized as deployer the function, caller need to have the private key of the wallet used to deployed the contract. After creating the log, the function will also emit an event in the Ethereum Blockchain. The event information could be accesed through Etherscan [71].

6.3.2 Verify

Anybody can use the “verify” function. It checks whether a log item has been notarized. The input string (e.g., core information) will be turned into hash string first, then compared with the hash values in the proofs array. The function returns false if the hash value is not found in the array.

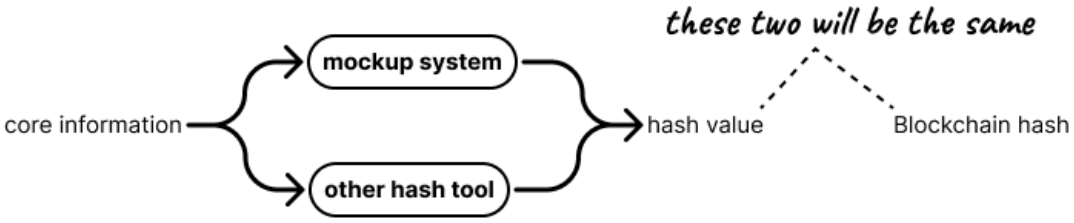

Fig. 6.7: Compare the hash value

6.3.3 Source code

The code of smart contract (Fig. 6.8) is publicly available; anyone could look up the information with the following smart contract address in Etherscan website [71]:

0x0Dee0f40573ec25E5Db34BAAF507e24C3EDa49fe

The development of smart contract is separated from development of our application. After we tested the smart contract in the local environment using Hardhat, we have deployed it to the Goerli test net of Ethereum Blockchain. After the deployment, the smart contract will be transformed into Application Binary Interface (ABI). The ABI file is necessary for calling API of the smart contract. It is put under our project folder.

```
contract LogProof {
    address public immutable owner;
    bytes32[] public proofs;
    event NewLogNotarized(bytes32 indexed proof);

    constructor() {
        owner = msg.sender;
    }

    function notarize(string memory _hashInput) public returns (bytes32) {
        require(msg.sender == owner, "Only the owner can notarize");
        bytes32 proof = getProof(_hashInput);
        storeProof(proof);
        emit NewLogNotarized(proof);
        return proof;
    }

    function verify(string memory _hashInput) public view returns (bool) {
        bytes32 proof = getProof(_hashInput);
        return hasProof(proof);
    }

    function getProof(string memory _hashInput) public pure returns (bytes32) {
        return keccak256(abi.encodePacked(_hashInput));
    }

    function storeProof(bytes32 proof) internal {
        proofs.push(proof);
    }

    function hasProof(bytes32 proof) internal view returns (bool) {
        for (uint256 i = 0; i < proofs.length; i++) {
            if (proofs[i] == proof) {
                return true;
            }
        }
        return false;
    }
}
```

Fig. 6.8: Smart contract code

6.4 File structure of the system

The complete list of files in our code project is quite long; we will divide the list into three parts and introduce them one by one.

6.4.1 Part 1

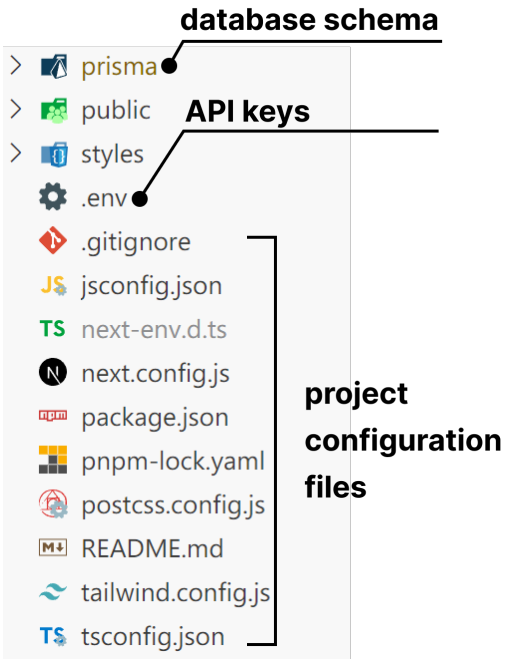

Fig. 6.9: Part 1 of the file list

As we used Prisma as ORM to connect to Postgres database, so there is a "**prisma**" folder holding schama information. The definition of "Log" table exists in here.

The "**.env**" file stores credentials like database URL, password we used to connect to the database. In addition, we keep private key of wallet (used to deploy smart contract) and Alchemy API keys in this file.

Other than these two files are mostly configuration files automatically generated by Next.js, the framework we used for making full stack application development easier. (Fig. 6.9)

6.4.2 Part 2

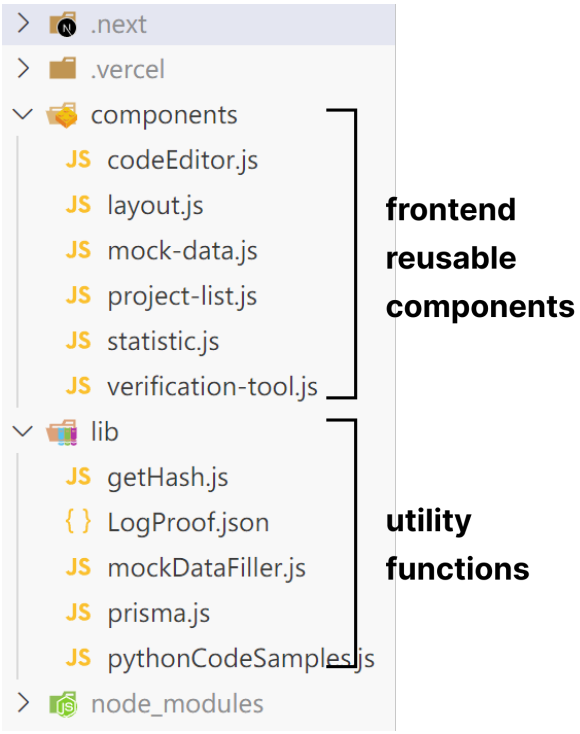

Fig. 6.10: Part 2 of the file list

In this part, there are two important folders: reusable components in the "**components**" folder and utility functions in the "**lib**" folder (Fig. 6.10). Component is a concept used in React to refer reusable user interface parts. The "**codeEditor.js**" beautifies display of code, the "**layout.js**" stipulates overall page layout of our mockup system. Each of "**mock-data.js, project-list.js, statistic.js, verification-too.js**" correspond to tab panel in the mockup system.

The "**LogProof.json**" in the "**lib**" folder is the ABI file used as an intermediary to interact with smart contract. "**getHash.js**" receive a string and output a hash value generated with Keccack256 hash method. The two files "**mockDataFiller.js, pythonCodeSamples.js**" are used for generating mock data of experiments. "**prisma.js**" is used to get the prisma client. We need this client before doing any operations to the database.

6.4.3 Part 3

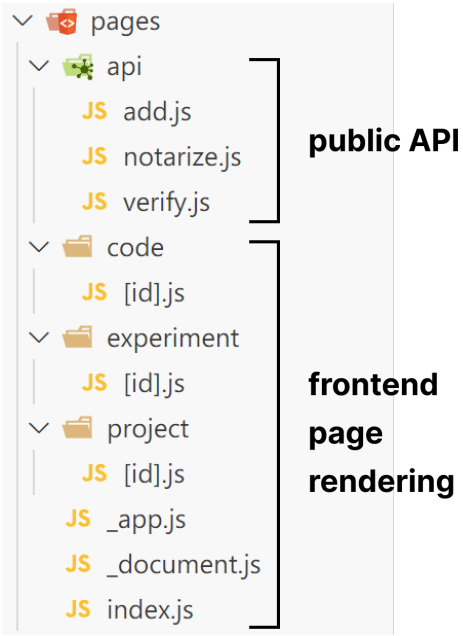

Fig. 6.11: Part 3 of the file list

In this part, there are several sub-folders under the "**pages**" folder (Fig. 6.11). The "**api**" folder is comprised of three publicly available APIs. The "**add.js**" is used for adding log item to the database. Database credentials are required to call this API. The "**notarize.js**" can notarize a log item by calling the "notarize" function of smart contract. The private key of contract deployer is required to call this API. The "**verify.js**" can verify whether the core information has been notarized. Everyone can call this API without restriction as long as he pays gas fee. Gas fee of verification happens within our system is paid by the platform developer.

The "**code, experiment, project**" sub-folders correspond to pages display information of code, experiment and project. The "**index.js**" page is the entry file of entire system.

6.5 Features of the mockup system

In this section, we will go through the key components of the mockup system. First, main tab panels: Project list, Mock data, Verification tool. Then the detail pages: Project detail, Experiment detail, Code detail.

This mockup system is developed to serve only as a proof of concept (POC); it is not integrated with the production environment of data trustee for sleep research yet. For this reason, we have implemented some features that should belong to the data trustee platform. These extra features complement the logging features as a whole, and make it easier to observe how the logging features work.

6.5.1 Project list

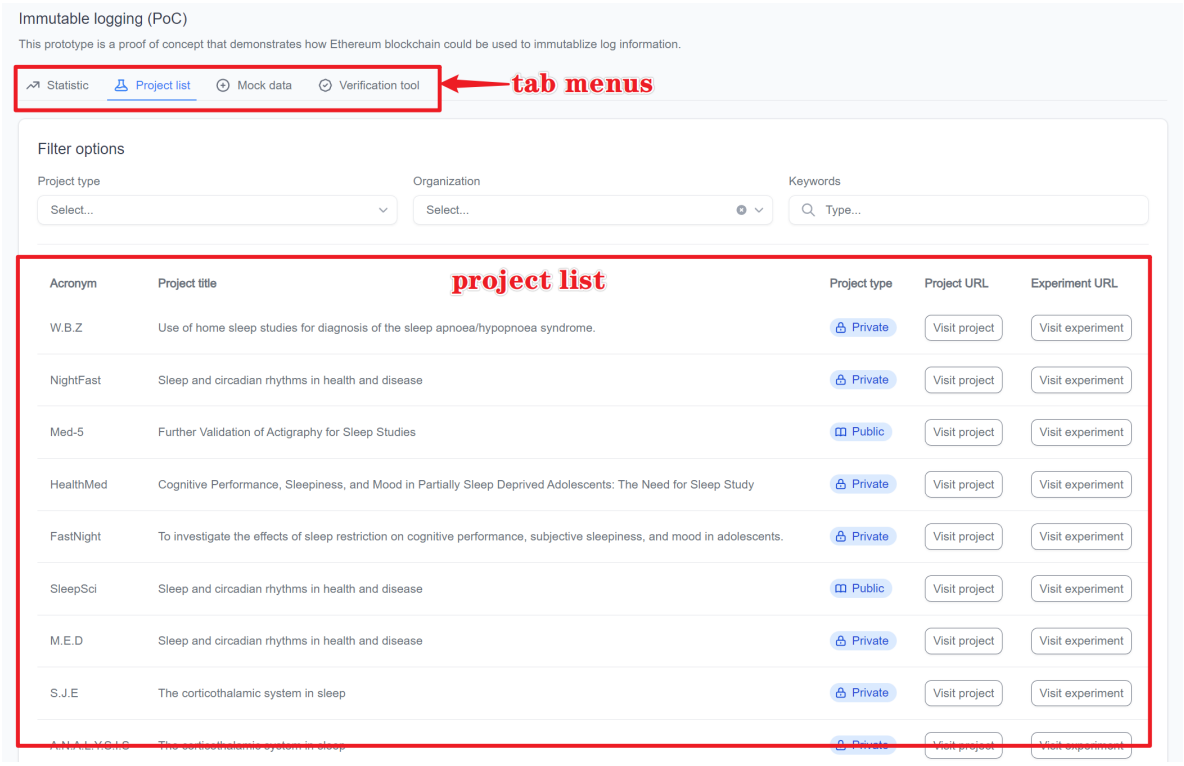

Fig. 6.12: Project list (screenshot of mockup system)

The project list displays all the research projects on the platform. Each project has its project detail page and experiment detail page. These pages can be reached with button in the list. (Fig. 6.12)

6.5.2 Mock data

In the left column of “Mock data” panel is the project setting and algorithm will be used to process the data. In the right column is the corresponding log item and core information generated from the log. When submitted, the core information will be sent to Blockchain to be notarized. (Fig. 6.13)

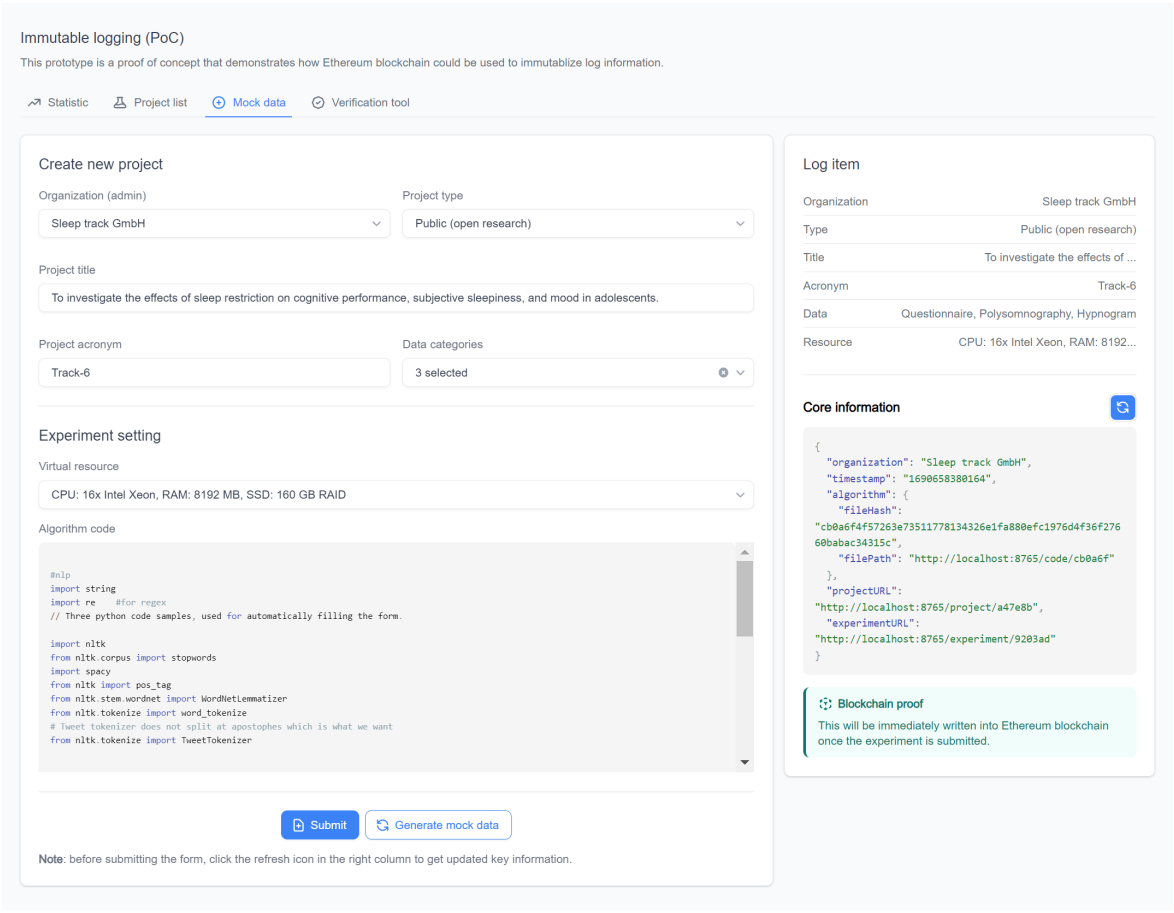

Fig. 6.13: Generating usage record (screenshot of mockup system)

6.5.3 Verification tool

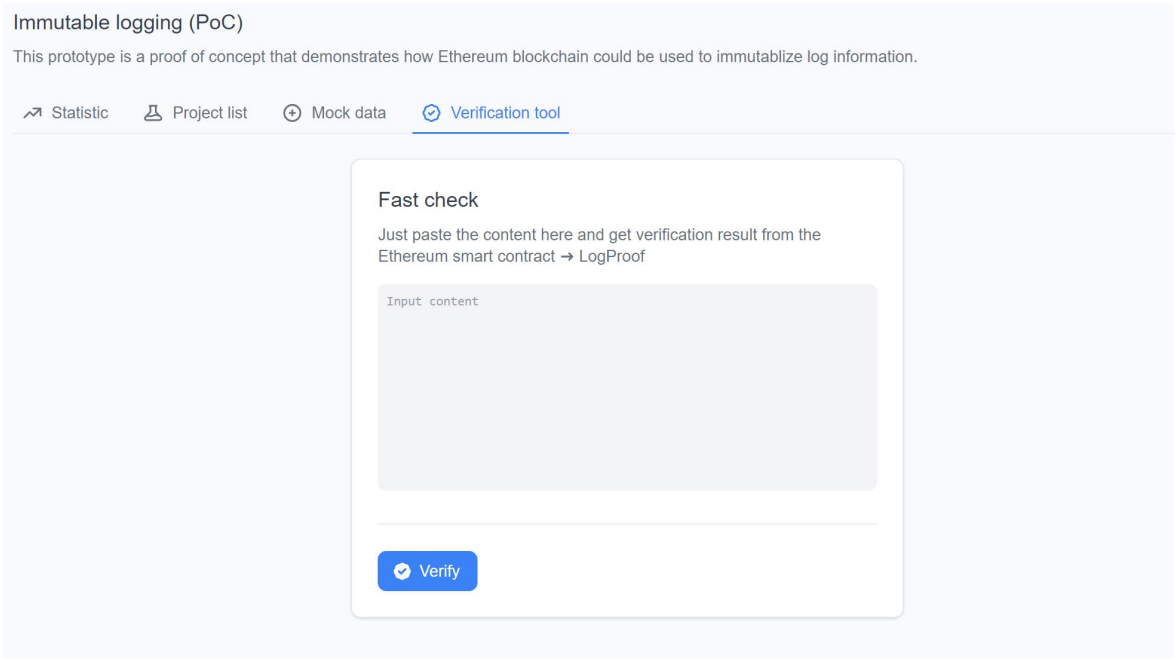

Fig. 6.14: Verification tool (screenshot of mockup system)

Whenever we want to check whether a piece of core information is authentic, we can use the verification tool here (Fig. 6.14). The system will send the information to the Blockchain and call the “verify” function of our smart contract. The hash value of this core information must match with the hash value of previously notarized core information. The hash method here is Keccak256, this is also used in smart contract. Any other online hash tools generates the same hash value as long as they use the same core information and hash method.

6.5.4 Project detail

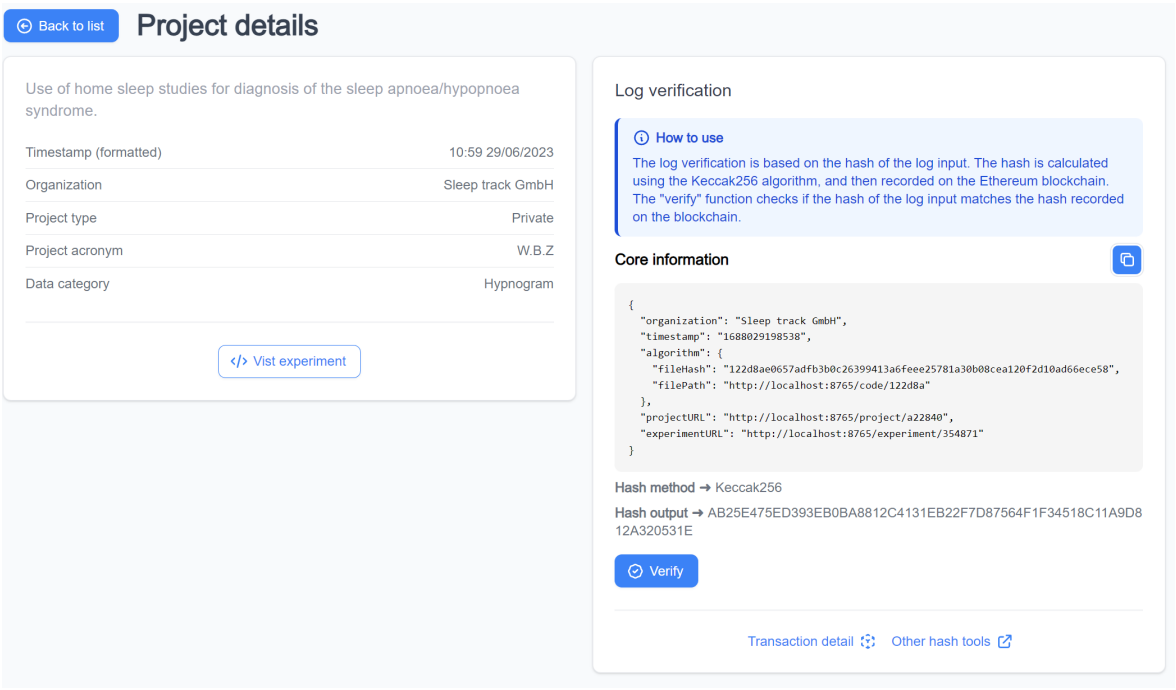

Fig. 6.15: Verify a specific project (screenshot of mockup system)

The project detail page shows all information about the research project. In addition, the core information of this project and hash value of this core information are displayed on the right column. (Fig. 6.15) We can verify whether the core information has been notarized to check its authenticity.

6.5.5 Experiment detail

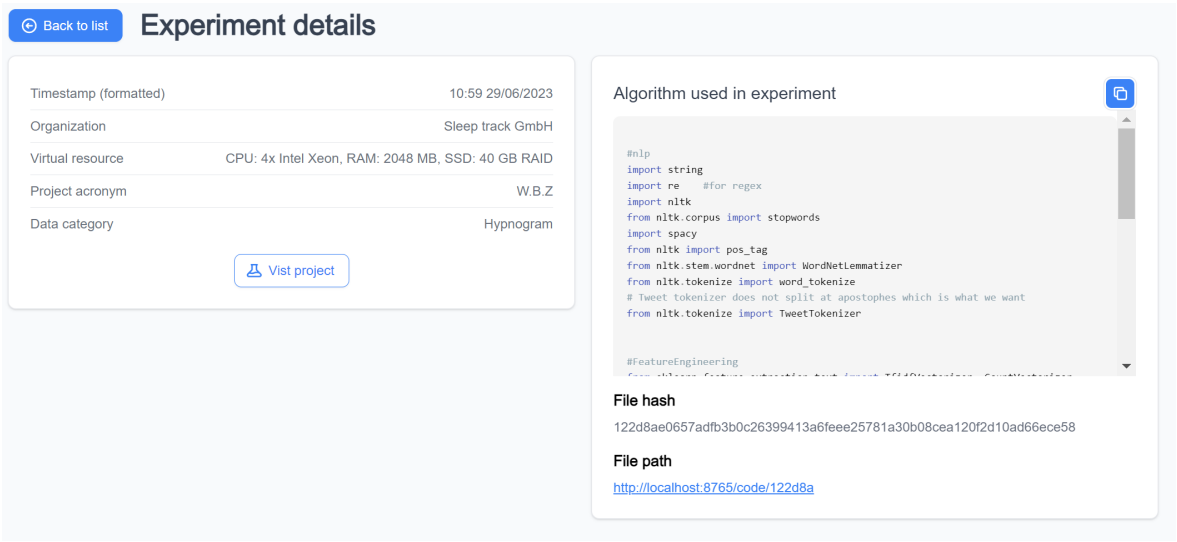

Fig. 6.16: Look up details of an experiment (screenshot of mockup system)

In the left column is the detail information of experiment setting. In the right column is the algorithm code preview, file path to the code and also hash value of the algorithm file. The hash value is stored in the database. It is used to prove the algorithm code has not change since it is first stored in the database.

6.5.6 Code file

```
#nlp
import string
import re    #for regex
import nltk
from nltk.corpus import stopwords
import spacy
from nltk import pos_tag
from nltk.stem.wordnet import WordNetLemmatizer
from nltk.tokenize import word_tokenize
# Tweet tokenizer does not split at apostophes which is what we want
from nltk.tokenize import TweetTokenizer


#FeatureEngineering
from sklearn.feature_extraction.text import TfidfVectorizer, CountVectorizer, HashingVectorizer
from sklearn.decomposition import TruncatedSVD
from sklearn.base import BaseEstimator, ClassifierMixin
from sklearn.utils.validation import check_X_y, check_is_fitted
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
from sklearn.metrics import log_loss
from sklearn.model_selection import StratifiedKFold
from sklearn.model_selection import train_test_split

def cramers_corrected_stat(confusion_matrix):
    """ calculate Cramers V statistic for categorial-categorial association.
        uses correction from Bergsma and Wicher,
        Journal of the Korean Statistical Society 42 (2013): 323-328
    """
    chi2 = ss.chi2_contingency(confusion_matrix)[0]
    n = confusion_matrix.sum().sum()
    phi2 = chi2/n
    r,k = confusion_matrix.shape
    phi2corr = max(0, phi2 - ((k-1)*(r-1))/(n-1))
    rcorr = r - ((r-1)**2)/(n-1)
```

Fig. 6.17: Details of algorithm code used to process data (screenshot of mockup system)

This page displays algorithm code used by researcher to process sleep data in the experiment. (Fig. 6.17)

6.6 Call graph

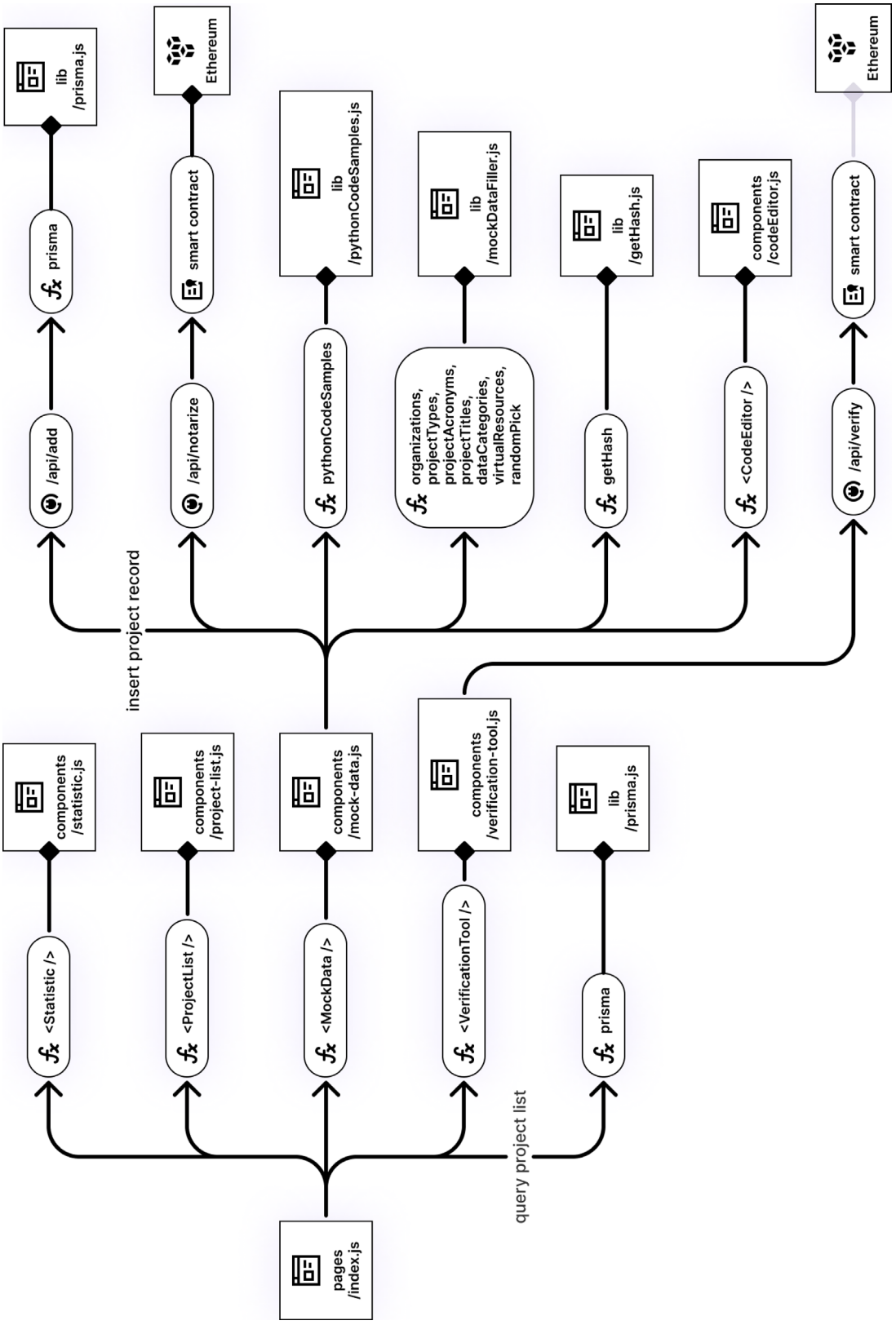

Fig. 6.18: Call graph of the entire system.

As the entry point, "**index.js**" loads all tab panels at once: statistic, project list, mock data, verification tool. It also loads Prisma client for later use. In the "verification tool" panel, users can paste the core information they hope to verify with the "**verify**" API.

The most complex one is "mock data" panel. After manually or automatically filling the project information and experiment setting, all the information will be stored into the database with "**add**" API. At the same time, the context information will be transformed into more succinct core information. The core information is then sent to the Blockchain to be notarized.

6.7 Technology stack

We used React.js to build responsive UI. Tremor is a out-of-the-box react component library, and TailwindCSS is the CSS utility library. Both of them make frontend UI development faster and more maintainable.

Next.js is a fullstack framwork that works well with React. We used Prisma as the Object Relational Mapping (ORM) manager to connect to the Postgres database. Ethers is a node module, it hleps us to interact with the deployed smart contract. Solidity is a programming language for building smart contract of Ethereum Blockchain.

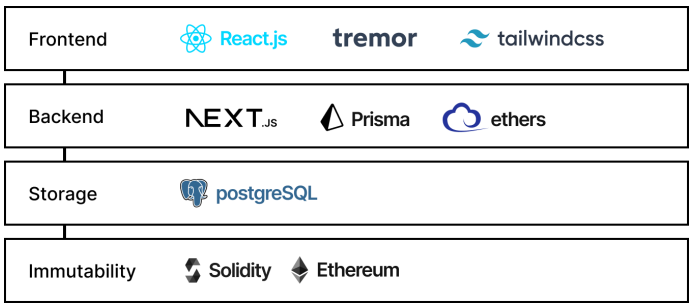

Fig. 6.19: Technology stack

7 EVALUATION AND DISCUSSION

In this chapter, we will show how we designed the survey used to evaluate the UI prototype, and then we will show the analysis of recycled survey results, at last we will discuss the degree of fulfillment of requirements from "**5 Concept design**" in the next section.

7.1 Design of survey

The survey is comprised of four parts (see **Appendix: Survey questions**):

1. **General information**: the first part asks about general information (e.g., age, gender and proficiency in using software) of participants,
2. **Tasks to be solved**: the second part have designed four group of tasks for the participants to solve,
3. **Transparency score**: asking participants to give a transparency score based on their experience with the UI prototype during completing the survey,
4. **Questions from SUS** [72]: the ten questions are directly taken from "system usability scale". A final score will be calculated to measure the usability of the prototype.

The four groups of tasks can be organized as such (Table 7.1):

Task index	Question index	Menu of the prototype	Tested requirements
Task 1	Q1	About SouveMed	S1
	Q2	My dataset	M1

Task 2	Q3	Consent management	M2, C1, C2
Task 3	Q4	Usage records > By dataset	M3
Task 4	Q5	Usage records > Project list	M3
	Q6	Usage records > Project list	M3

Table 7.1: Four groups of tasks and the tested requirements.

7.2 Analysis of survey results

We distributed the survey through mailing list of FZI, social media, and also asked friends who are interested in privacy protection topic to join in the survey. The starting date of distribution was 2023.06.21, after about one week, we have successfully collected 32 participants' feedbacks.

7.2.1 General information of participants

Half of the participants are male; nearly half of them are female (Fig. 7.1).

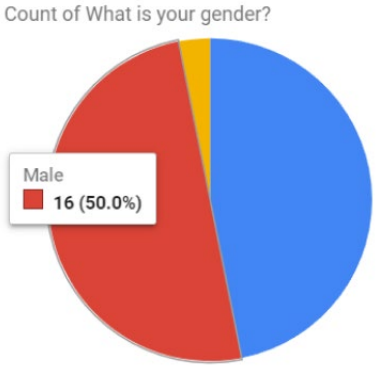

Fig. 7.1: Gender distribution of participants.

About 80% of participants are older than 18 and younger than 39. About 20% of them are older than 40 (Fig. 7.2).

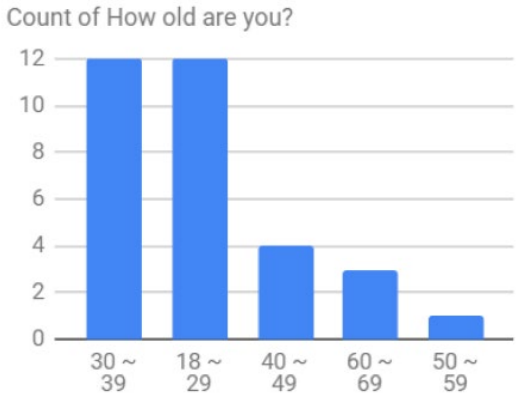

Fig. 7.2: Age distribution of participants.

About 30% of participants are of high proficiency of using software technology. About 40% of them are of medium level proficiency. Less than 20% of them have only limited proficiency. (Fig. 7.3)

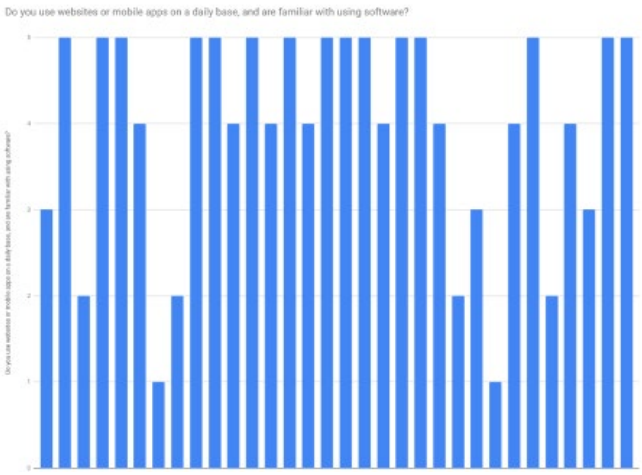

Fig. 7.3: Software proficiency distribution of participants.

As this questionnaire is designed to evaluate UI prototype of client app for donators. The evaluation result could be more relevant if the participants' average age is a little bit larger and proficiency of using software technology a little bit lower than current participants.

7.2.2 Usability score

John Brooke published system Usability Scale (SUS) in 1986. It is one of the most well-known and established methods for testing usability [72]. It is a standardized questionnaire for measuring the usability of software product perceived by the user. The survey consists of ten statements using a Likert scale.

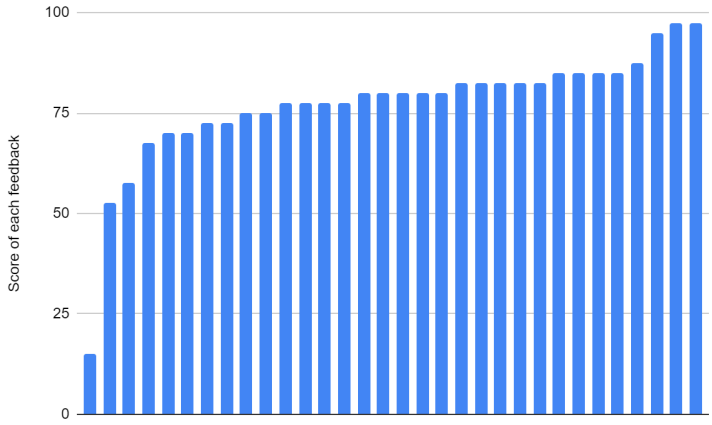

Fig. 7.4: Results of SUS scores.

Scores higher than 75 are labelled as good feedback. More than 70% of participant gave scores higher than 75. (Fig. 7.4)

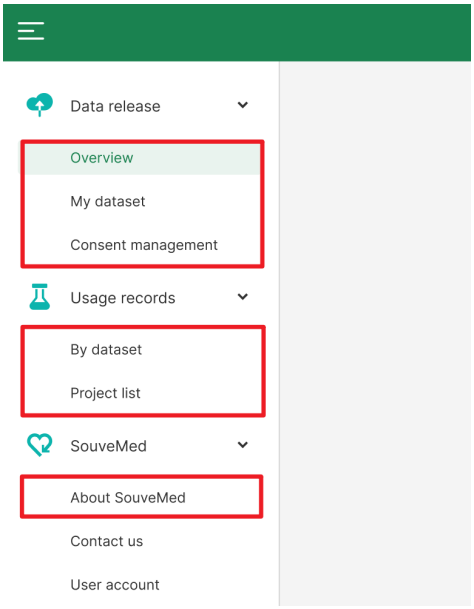

Fig. 7.5: Tested menus of UI prototype (donator's app).

Fig. 7.6: Final score of usability according to SUS's algorithm.

According to official standard of SUS, an average score higher than 72 is considered to be accepTable In our case, we get a score of 77.1. It means under the SUS evaluation framework, there is no evident usability issues might hamper participants experience when testing the prototype.

7.2.3 Success rate of each task

We have designed four group of tasks. In each task, the participants are given a short guidance text, then a set of single choice or multiple-choice questions to answer (Fig. 7.7). The questions are designed in such way that it is impossible to guess the right answer without clicking through the interactive UI prototype. Since the entire process of survey takes online, it is not possible to observe how participants fill out the survey. But the options of each task question are closely related to the UI prototype, participants have to precisely avoid incorrect options and choose the right ones then their answer to the question will be labeled as "correct".

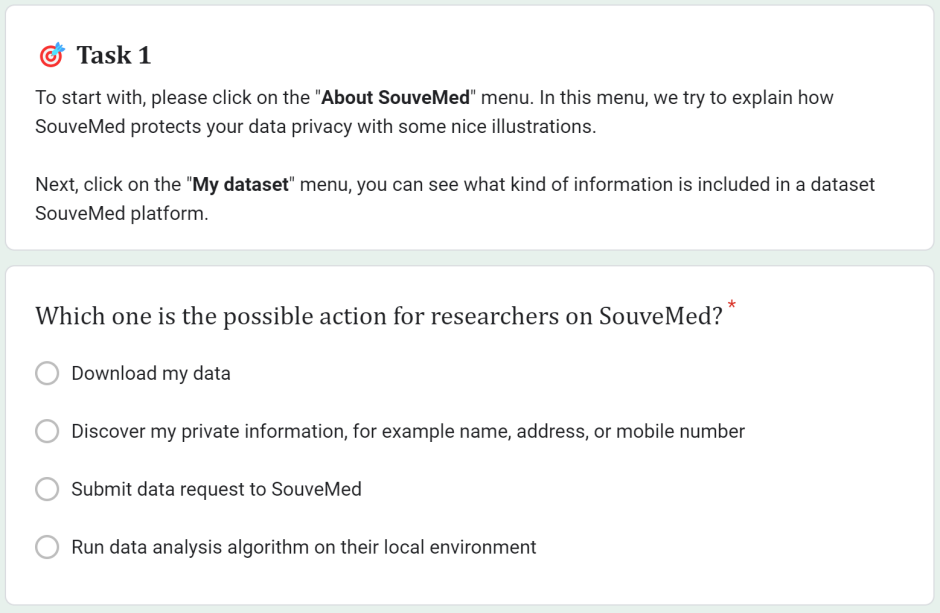

Fig. 7.7: Example of a task in the questionnaire

7.2.3.1 TASK 1

This task is designed to evaluate whether donators can understand how the platform processes their data. 81.2% of participants have submitted right answers to all questions in this task.

7.2.3.2 TASK 2

This task is designed to evaluate whether donators know how to review and adjust their consent according to their own preferences. 84.3% of participants have submitted right answers in this task.

7.2.3.3 TASK 3

Both of task 3 and task 4 are designed to evaluate functions under “Usage records” menu. Task 3 focus on testing whether donators can understand the connection between usage record and dataset. In addition, it has tested whether donators understand information displayed in the detailed page of usage record. 82.2% of participants have submitted right answers in this task.

7.2.3.4 TASK 4

Task 4 focus on usage records displayed in a “project list” mode of view. It has tested whether donators can quickly locate a usage record with filter options. 82.2% of participants have submitted right answers to all questions in this task.

7.2.4 Transparency score

From a scale one to ten, this question asks participants for transparency score based on their experience with the UI prototype. A score of “one” represents “Lots of information are still missing”, a score of “ten” represents “I know exactly what’s going on with my data” (Fig. 7.8).

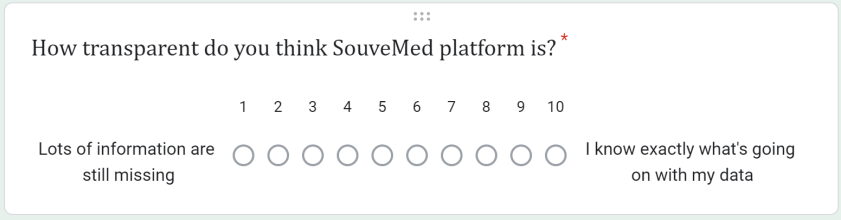

Fig. 7.8: Likert scale of transparency score

More than 80% of participants gave a score of 9 or 10. The average score of 32 participants is 9.37.

7.3 Degree of fulfillment of the requirements

In this section, we will review how much the requirements of donator and third party auditor are fulfilled according to the evaluation results (Table 7.2).

Code	Status	Stakeholder	Discussion
M1	fulfilled	Donator	Donators can browse a list of their datasets and also look into details of each dataset. This requirement is also tested in the survey, over 80% of participants

			have selected the right answer (Q2).
M2	fulfilled	Donator	Donators can see their current preference setting for each dataset. This requirement is also tested in the survey, over 80% of participants have selected the right answer (Q3).
M3	fulfilled	Donator	Donators can browse usage records connected to their datasets in two different view modes. This requirement is also tested in the survey, over 80% of participants have selected the right answer (Q4, Q5, Q6).
S1	fulfilled	Donator	Donators can understand the basic operating mechanism of the platform with the help of flash card illustrations. This requirement is also tested in the survey, over 80% of participants have selected the right answer (Q1).
S2	N/A	Donator	Not tested in the survey.
S3	N/A	Donator	Not tested in the survey.
C1	N/A	Donator	Not tested in the survey.
C2	N/A	Donator	Not tested in the survey.
C3	N/A	Donator	Not tested in the survey.
M9	fulfilled	Third party auditor	Third parties auditors can browse all usage records generated within the platform. This requirement has some similarities with M3, the survey result of M3 can partially prove the effectiveness of our solution to this requirement.
M10	fulfilled	Third party auditor	Third parties auditors can browse all usage records generated within the platform. This requirement has some similarities with M3, the survey result of M3 can partially prove the effectiveness of our solution to this requirement.
M11	N/A	Third party auditor	Not tested in the survey.

C9	N/A	Third party auditor	Given the project objective, experiment setting and algorithm information, it is possible to replicate the procedures and reproduce the results. This requirement is not tested in the survey.

Table 7.2: How much donators and third party auditors' requirements are fulfilled.

Other requirements not listed in Table 7.2 are no implemented in the concept design phase. Those implemented requirements but not evaluated with survey are makred with the status of "N/A" (not available).

8 CONCLUSION AND OUTLOOK

In this chapter we will review how well our research questions from the beginning is met. And also discuss some weak points could be augmented in the future.

8.1 Review of the research questions

In this section, we will review our research questions and discuss how much we have fulfil these questions with our solutions and research findings.

RQ1. How to ensure that audit logs are authentic and accurately reflect to the data processing events that happens within the platform?

Discussion: We have built the mockup logging system based on Ethereum Blockchain. All the context information is kept in safe place and notarized by the smart contract. Any modifications to the log data will be discovered. In this way we make sure audit logs are authentic and honestly reflects to data processing events.

RQ2. What exactly are the stakeholders' requirements for transparency, and what kind of information should be extracted from events happened within the platform?

Discussion: First we applies persona technique to grasp the overall image of the stakeholders. Then we listed all the stakeholders connected to the platform, and requirements of each type of stakeholder. At last, all requirements are further classfied according to priority and relevance to our transparency topic.

RQ3. In which way the disclosure of the information is most effective, so that stakeholders can easily access and consume the log information?

Discussion: We have built the UI prototype for the two most important stakeholders: donator and third party auditors. The UI prototype is designed to be user-friendly so that log information is easy to explore and understand. The

effectiveness is evaluated with several sets of tasks and the results proved to be excellent.

8.2 Expensive transaction cost

Every time we notarize a log item with the smart contract, there will be a cost. The cost of executing smart contract varies depends on the complexity the smart contract and current gas fee [73]. In our case, notarizing a usage record costs about €1 ~ €2.5 (fluctuates depending on the real time price of gas fee). If the platform generates many usage records, the cost will increase proportionately.

A Layer 2 Blockchain like Polygon is much cheaper than Ethereum Blockchain. If the usage records have reached more than twenty per day, the platform should start to consider combining Layer 2 Blockchain and Ethereum Blockchain together. The Layer 2 Blockchain could first notarize every usage record, and then the logging system aggregates a small batch of log integrity proofs periodically, at last write the aggregated proof into Ethereum Blockchain. (Fig. 8.1)

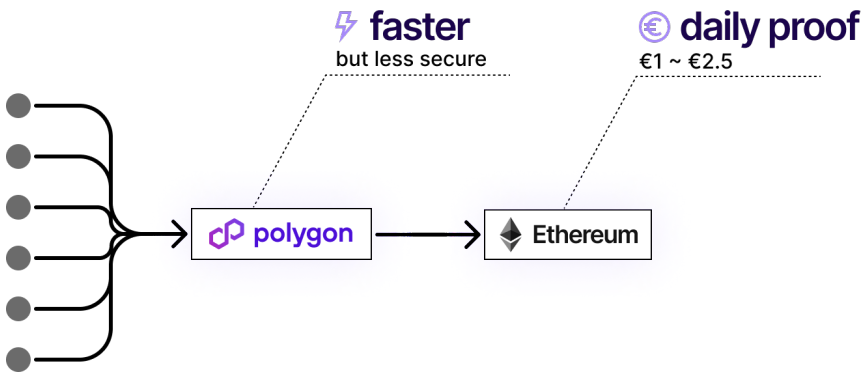

Fig. 8.1: Adding a Layer 2 Blockchain in between

Layer 2 Blockchains are faster because they use a different approach to reach consensus. This also make them more vulnerable than Ethereum Blockchain. As long as we still frequently store integrity proof to Ethereum Blockchain, security issues can be minimized.

8.3 Technical improvements

There are some technical shortcomings exist in our logging system. It could be enhanced with more robust technology solutions. Here we will list three major problems and present the potential solutions to solve them.

8.3.1 Partial proof

In this thesis, we focus on the log event of consuming data. We should not overlook other important events:

- **Gathering data consent**: the intrinsic nature of data trustee platform is to gather data consents from individual donators then redistribute those consents to data users. Recording the data consent set by donator is important.
- **Signing usage policy**: data users have to sign the usage policy before getting access to data. The usage policy regulates the data usage behaviours. The action of signing means data users are committed to follow the rules. Recording the signing event is helpful for resolving potential disputes.
- **Building data pipeline**: only after the platform has built the data pipeline according to data users' requirements and data donators' preferences, data users can start using data. This is the last step before data usage event happens. It is important to record the steps platform took to prepare the data.

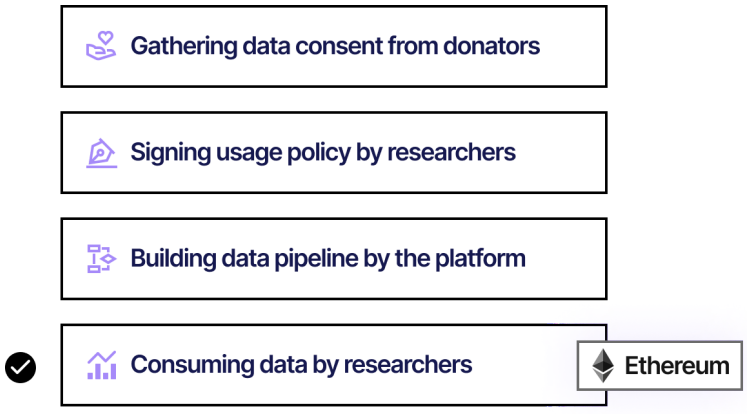

Fig. 8.2: All scenarios that should be secured

8.3.2 Insider attack

Considering the transaction cost, we only store the hash value of core information to the Blockchain. Database in the backend is essential for the logging system to recreate the scene. Imagine if a platform manager has the access to manipulate database, the context information of data usage records will be lost.

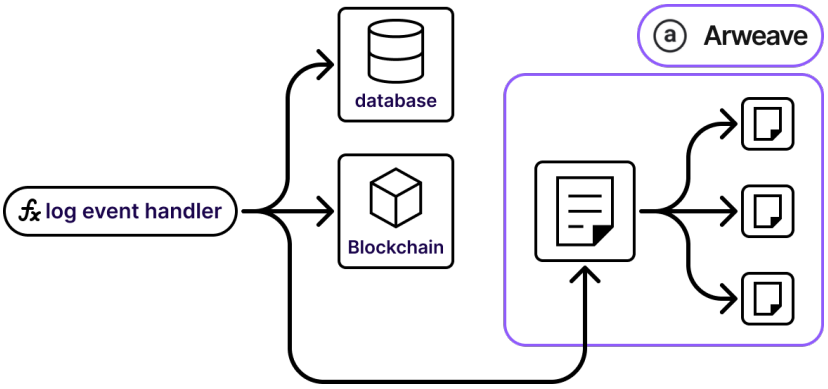

Fig. 8.3: Adding Arweave as backup storage

To solve this problem, we can rely on Arweave as a third storage location. Arweave is a decentralized network that promised to store files for 200 years [75]. It is a paid service, currently $10/GB.

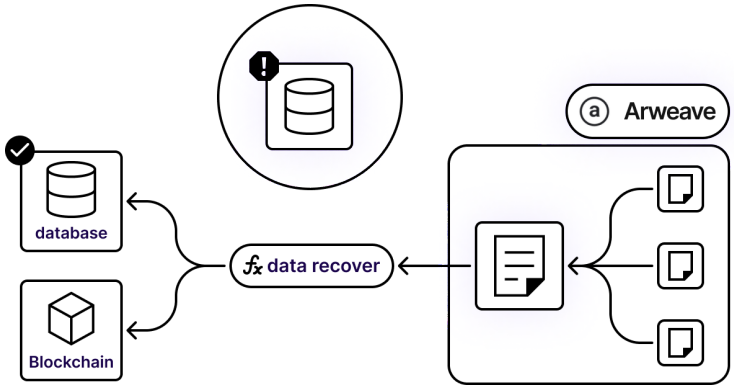

Fig. 8.4: Rebuild the database after-the-event

In case the database is corrupted, it is always possible to rebuild the database from zero and also able to check validity of log items with the help of Blockchain

LITERATURE

[1] Skaer, T. L., & Sclar, D. A. (2010). Economic implications of sleep disorders. Pharmacoeconomics, 28(11), 1015-1023.

[2] Panossian, L. A., & Avidan, A. Y. (2009). Review of sleep disorders. Medical Clinics of North America, 93(2), 407-425.

[3] Bragazzi, N. L., Guglielmi, O., & Garbarino, S. (2019). SleepOMICS: how big data can revolutionize sleep science. International journal of environmental research and public health, 16(2), 291.

[4] Blankertz, Aline, Patrick von Braunmühl, Pencho Kuzev, Frederick Richter, Heiko Richter und Martin Schallbruch. 2020. Datentreuhandmodelle. https://www.ip.mpg.de/de/publikationen/details/datentreuhandmodellethemenpapier.html.

[5] Li, H., Yu, L., & He, W. (2019). The impact of GDPR on global technology development. Journal of Global Information Technology Management, 22(1), 1-6.

[6] Garfinkel, S. L. (2010). Digital forensics research: The next 10 years. Digital Investigation, 7, S64-S73.

[7] Caine, K., & Hanania, R. (2013). Patients want granular privacy control over health information in electronic medical records. Journal of the American Medical Informatics Association, 20(1), 7-15.

[8] Skatova, A., & Goulding, J. (2019). Psychology of personal data donation. PloS one, 14(11), e0224240.

[9] Mori, I. (2017). The one-way mirror: public attitudes to commercial access to health data.

[10] Hussain, T., Asghar, S., & Masood, N. (2010, June). Web usage mining: A survey on preprocessing of web log file. In 2010 International Conference on Information and Emerging Technologies (pp. 1-6). IEEE.

[11] Baskerville, R., Baiyere, A., Gregor, S., Hevner, A., & Rossi, M. (2018). Design science research contributions: Finding a balance between artifact and theory. Journal of the Association for Information Systems, 19(5), 3.

[12] Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. Journal of Management Information Systems, 24(3), 45–77. https://doi.org/10.2753/MIS0742-1222240302

[13] Brooke, J. (1995). SUS: A quick and dirty usability scale. Usability Eval. Ind., 189.

[14] Newell, A. (1979). Reasoning, problem solving and decision processes: The problem space as a fundamental category.

[15] Olsen, D. (2015). The Lean Product Playbook: How to Innovate with Minimum Viable Products and Rapid Customer Feedback. John Wiley & Sons.

[16] Fu, Q., Lou, J.-G., Wang, Y., & Li, J. (2009). Execution Anomaly Detection in Distributed Systems through Unstructured Log Analysis. 2009 Ninth IEEE International Conference on Data Mining, 149–158. https://doi.org/10.1109/ICDM.2009.60

[17] Lou, J.-G., Fu, Q., Yang, S., Xu, Y., & Li, J. (n.d.). Mining Invariants from Console Logs for System Problem Detection.

[18] Glerum, K., Kinshumann, K., Greenberg, S., Aul, G., Orgovan, V., Nichols, G., Grant, D., Loihle, G., & Hunt, G. (2009). Debugging in the (very) large: Ten years of implementation and experience. Proceedings of the ACM SIGOPS 22nd Symposium on Operating Systems Principles, 103–116. https://doi.org/10.1145/1629575.1629586

[19] Structured Comparative Analysis of Systems Logs to Diagnose Performance Problems | USENIX. (n.d.). Retrieved July 14, 2023, from https://www.usenix.org/conference/nsdi12/technical-sessions/presentation/nagaraj

[20] Sambasivan, R. R., Zheng, A. X., Rosa, M. D., Krevat, E., Whitman, S., Stroucken, M., Wang, W., Xu, L., & Ganger, G. R. (n.d.). Diagnosing performance changes by comparing request flows.

[21] Sharma, B., Chudnovsky, V., Hellerstein, J. L., Rifaat, R., & Das, C. R. (2011). Modeling and synthesizing task placement constraints in Google compute clusters. Proceedings of the 2nd ACM Symposium on Cloud Computing, 1–14. https://doi.org/10.1145/2038916.2038919

[22] Fu, Q., Lou, J.-G., Lin, Q., Ding, R., Zhang, D., & Xie, T. (2013). Contextual analysis of program logs for understanding system behaviors. 2013 10th Working Conference on Mining Software Repositories (MSR), 397–400. https://doi.org/10.1109/MSR.2013.6624054

[23] Hackers are increasingly destroying logs to hide attacks. (n.d.). ZDNET. Retrieved July 14, 2023, from https://www.zdnet.com/article/hackers-are-increasingly-destroying-logs-to-hide-attacks/

[24] BlackEnergy APT Attacks in Ukraine. (2023, April 19). Www.Kaspersky.Com. https://www.kaspersky.com/resource-center/threats/blackenergy

[25] Indicator Removal: Clear Windows Event Logs, Sub-technique T1070.001—Enterprise | MITRE ATT&CK®. (n.d.). Retrieved July 14, 2023, from https://attack.mitre.org/techniques/T1070/001/

[26] Indicator Removal: Clear Linux or Mac System Logs, Sub-technique T1070.002—Enterprise | MITRE ATT&CK®. (n.d.). Retrieved July 14, 2023, from https://attack.mitre.org/techniques/T1070/002/

[27] Anderson, J. M. (2003). Why we need a new definition of information security. Computers & Security, 22(4), 308–313. https://doi.org/10.1016/S0167-4048(03)00407-3

[28] Samonas, S., & Coss, D. (n.d.). THE CIA STRIKES BACK: REDEFINING CONFIDENTIALITY, INTEGRITY AND AVAILABILITY IN SECURITY.

[29] Mukkamala, S., & Sung, A. H. (2003). Identifying Significant Features for Network Forensic Analysis Using Artificial Intelligent Techniques. 1(4).

[30] Lakshman, A., & Malik, P. (2010). Cassandra: A decentralized structured storage system. ACM SIGOPS Operating Systems Review, 44(2), 35–40. https://doi.org/10.1145/1773912.1773922

[31] Qadir, A. M., & Varol, N. (2019). A Review Paper on Cryptography. 2019 7th International Symposium on Digital Forensics and Security (ISDFS), 1–6. https://doi.org/10.1109/ISDFS.2019.8757514

[32] Schneier, B., & Kelsey, J. (1999). Secure audit logs to support computer forensics. ACM Transactions on Information and System Security, 2(2), 159–176. https://doi.org/10.1145/317087.317089

[33] Schneier, B., & Kelsey, J. (1998). Cryptographic Support for Secure Logs on Untrusted Machines. 7th USENIX Security Symposium (USENIX Security 98). https://www.usenix.org/conference/7th-usenix-security-symposium/cryptographic-support-secure-logs-untrusted-machines

[34] Bellare, M., & Yee, B. (1997). Forward Integrity For Secure Audit Logs.

[35] Ma, D., & Tsudik, G. (2009). A new approach to secure logging. ACM Transactions on Storage, 5(1), 1–21. https://doi.org/10.1145/1502777.1502779

[36] Holt, J. E. (2006). Logcrypt: Forward security and public verification for secure audit logs. Proceedings of the 2006 Australasian Workshops on Grid Computing and E-Research - Volume 54, 203–211.

[37] Yavuz, A., Ning, P., & Reiter, M. (2012). LogFAS: Efficient, Compromise Resilient and Append-only Cryptographic Constructions for Digital Forensics.

[38] Yavuz, A. A., Ning, P., & Reiter, M. K. (2012). BAF and FI-BAF: Efficient and Publicly Verifiable Cryptographic Schemes for Secure Logging in Resource-Constrained Systems. ACM Transactions on Information and System Security, 15(2), 1–28. https://doi.org/10.1145/2240276.2240280

[39] Kampanakis, P., & Yavuz, A. A. (2015). BAFi: A practical cryptographic secure audit logging scheme for digital forensics. Security and Communication Networks, 8(17), Article 17. https://doi.org/10.1002/sec.1242

[40] Hartung, G., Kaidel, B., Koch, A., Koch, J., & Hartmann, D. (2017). Practical and Robust Secure Logging from Fault-Tolerant Sequential Aggregate Signatures. In T. Okamoto, Y. Yu, M. H. Au, & Y. Li (Eds.), Provable Security (pp. 87–106). Springer International Publishing. https://doi.org/10.1007/978-3-319-68637-0_6

[41] Wang, Y., & Zheng, Y. (2003). Fast and Secure Magnetic WORM Storage Systems. Second IEEE International Security in Storage Workshop, 11–11. https://doi.org/10.1109/SISW.2003.10002

[42] Chong, C. N., Peng, Z., & Hartel, P. H. (2003). Secure Audit Logging with Tamper-Resistant Hardware. In D. Gritzalis, S. De Capitani di Vimercati, P. Samarati, & S. Katsikas (Eds.), Security and Privacy in the Age of Uncertainty (pp. 73–84). Springer US. https://doi.org/10.1007/978-0-387-35691-4_7

[43] Sinha, A., Jia, L., England, P., & Lorch, J. R. (2014). Continuous Tamper-Proof Logging Using TPM 2.0. In T. Holz & S. Ioannidis (Eds.), Trust and Trustworthy Computing (pp. 19–36). Springer International Publishing. https://doi.org/10.1007/978-3-319-08593-7_2

[44] Karande, V., Bauman, E., Lin, Z., & Khan, L. (2017). SGX-Log: Securing System Logs With SGX. Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, 19–30. https://doi.org/10.1145/3052973.3053034

[45] Shepherd, C., Akram, R. N., & Markantonakis, K. (2017). EmLog: Tamper-Resistant System Logging for Constrained Devices with TEEs.

[46] Accorsi, R. (2011). BBox: A Distributed Secure Log Architecture. In J. Camenisch & C. Lambrinoudakis (Eds.), Public Key Infrastructures, Services and Applications (pp. 109–124). Springer. https://doi.org/10.1007/978-3-642-22633-5_8

[47] Zawoad, S., Dutta, A. K., & Hasan, R. (2013). SecLaaS: Secure Logging-as-a-Service for Cloud Forensics. 219–230. https://doi.org/10.1145/2484313.2484342

[48] Snodgrass, R. T., Yao, S. S., & Collberg, C. (2004). Tamper Detection in Audit Logs (pp. 504–515). VLDB Endowment.

[49] Cucurull, J., & Puiggalí, J. (2016). Distributed Immutabilization of Secure Logs. In G. Barthe, E. Markatos, & P. Samarati (Eds.), Security and Trust Management (pp. 122–137). Springer International Publishing. https://doi.org/10.1007/978-3-319-46598-2_9

[50] Ahmad, A., Lee, S., & Peinado, M. (2022). HARDLOG: Practical Tamper-Proof System Auditing Using a Novel Audit Device. 2022 IEEE Symposium on Security and Privacy (SP), 1791–1807. https://doi.org/10.1109/SP46214.2022.9833745

[51] Pawar, A., Barthare, D., Rawat, N., Yadav, M., & Shirole, M. (2021). BlockAudit 2.0: PoA blockchain based solution for secure Audit logs. 2021 5th International Conference on Information Systems and Computer Networks (ISCON), 1–6. https://doi.org/10.1109/ISCON52037.2021.9702378

[52] Sanchez, H. L., Tysebaert, S., Rath, A., & Rivière, E. (2022). AuditTrust: Blockchain-Based Audit Trail for Sharing Data in a Distributed Environment. In S. Marrone, M. De Sanctis, I. Kocsis, R. Adler, R. Hawkins, P. Schleiß, S. Marrone, R. Nardone, F. Flammini, & V. Vittorini (Eds.), Dependable Computing – EDCC 2022 Workshops (pp. 5–17). Springer International Publishing. https://doi.org/10.1007/978-3-031-16245-9_1

[53] López Pimentel, J., Morales Rosales, L., & Monroy, R. (2021). RootLogChain: Registering Log-Events in a Blockchain for Audit Issues from the Creation of the Root. Sensors, 21, 7669. https://doi.org/10.3390/s21227669

[54] Shekhtman, L., & Waisbard, E. (2019). EngraveChain: Tamper-Proof Distributed Log System. 8–14. https://doi.org/10.1145/3362744.3363346

[55] Ahmad, A., Saad, M., Njilla, L., Kamhoua, C., Bassiouni, M., & Mohaisen, A. (2019). BlockTrail: A Scalable Multichain Solution for Blockchain-Based Audit Trails. ICC 2019 - 2019 IEEE International Conference on Communications (ICC), 1–6. https://doi.org/10.1109/ICC.2019.8761448

[56] Wang, H., Yang, D., Duan, N., Guo, Y., & Zhang, L. (2018). Medusa: Blockchain Powered Log Storage System. 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 518–521. https://doi.org/10.1109/ICSESS.2018.8663935

[57] Ali, A., Khan, A., Ahmed, M., & Jeon, G. (2022). BCALS: Blockchain-based secure log management system for cloud computing. Transactions on Emerging Telecommunications Technologies, 33(4), e4272. https://doi.org/10.1002/ett.4272

[58] Hartung, G. (2017). Attacks on Secure Logging Schemes. In A. Kiayias (Ed.), Financial Cryptography and Data Security (pp. 268–284). Springer International Publishing. https://doi.org/10.1007/978-3-319-70972-7_14

[59] Hsu, W. W., & Ong, S. (2007). WORM storage is not enough [Technical Forum]. IBM Systems Journal, 46(2), 363–369. https://doi.org/10.1147/sj.462.0363

[60] Pulls, T., Wouters, K., Vliegen, J., & Grahn, C. (2012). Distributed Privacy-Preserving Log Trails. Karlstads universitet. http://urn.kb.se/resolve?urn=urn:nbn:se:kau:diva-13309

[61] Wouters, K., & Preneel, B. (2012). Hash-Chain based Protocols for Time-Stamping and Secure Logging: Formats, Analysis and Design (Hash-keten gebaseerde protocollen voor tijdszegels en beveiligde logbestanden: formaten, analyse en ontwerp).

[62] Sun, J., Yao, X., Wang, S., & Wu, Y. (2020). Blockchain-Based Secure Storage and Access Scheme For Electronic Medical Records in IPFS. IEEE Access, 8, 59389–59401. https://doi.org/10.1109/ACCESS.2020.2982964

[63] Chen, Y., Ding, S., Xu, Z., Zheng, H., & Yang, S. (2018). Blockchain-Based Medical Records Secure Storage and Medical Service Framework. Journal of Medical Systems, 43(1), 5. https://doi.org/10.1007/s10916-018-1121-4

[64] Nakamoto, S. (n.d.). Bitcoin: A Peer-to-Peer Electronic Cash System.

[65] Radley-Gardner, O., Beale, H., & Zimmermann, R. (Eds.). (2016). Fundamental Texts On European Private Law. Hart Publishing. https://doi.org/10.5040/9781782258674

[66] BDSG - nichtamtliches Inhaltsverzeichnis. (n.d.). Retrieved July 21, 2023, from https://www.gesetze-im-internet.de/bdsg_2018/

[67] Regulation (EU) 2022/868 of the European Parliament and of the Council of 30 May 2022 on European data governance and amending Regulation (EU) 2018/1724 (Data Governance Act) (Text with EEA relevance), 152 OJ L (2022). http://data.europa.eu/eli/reg/2022/868/oj/eng

[68] Achimugu, P., Selamat, A., Ibrahim, R., & Mahrin, M. N. (2014). A systematic literature review of software requirements prioritization research. Information and Software Technology, 56(6), 568–585. https://doi.org/10.1016/j.infsof.2014.02.001

[69] Xu, C., Doi, S. A. R., Zhou, X., Lin, L., Furuya-Kanamori, L., & Tao, F. (2022). Data reproducibility issues and their potential impact on conclusions from evidence syntheses of randomized controlled trials in sleep medicine. Sleep Medicine Reviews, 66, 101708. https://doi.org/10.1016/j.smrv.2022.101708

[70] Marshall, P. D., Moore, C., & Barbour, K. (2019). Persona Studies: An Introduction. John Wiley & Sons.

[71] etherscan.io. (n.d.). TESTNET Goerli (GTH) Blockchain Explorer. Ethereum (ETH) Blockchain Explorer. Retrieved July 23, 2023, from http://goerli.etherscan.io/

[72] Lewis, J. R. (2018). The System Usability Scale: Past, Present, and Future. International Journal of Human–Computer Interaction, 34(7), 577–590. https://doi.org/10.1080/10447318.2018.1455307

[73] Pierro, G. A., & Rocha, H. (2019). The Influence Factors on Ethereum Transaction Fees. 2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB), 24–31. https://doi.org/10.1109/WETSEB.2019.00010

[74] Rouhani, S., & Deters, R. (2017). Performance analysis of ethereum transactions in private blockchain. 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), 70–74. https://doi.org/10.1109/ICSESS.2017.8342866

[75] Can data really be stored forever? (n.d.). Retrieved July 23, 2023, from https://ardrive.io/can-data-really-be-stored-forever/

[76] Nickerson, R. C., Varshney, U., & Muntermann, J. (2013). A method for taxonomy development and its application in information systems. European Journal of Information Systems, 22(3), 336–359. https://doi.org/10.1057/ejis.2012.26

[77] Sommerville, I. (2011). Software Engineering, 9/E. Pearson Education India.

[78] Bertram, L., & Dahm, M. (n.d.). Conceptual design and implementation of an automated metrics and model-based usability evaluation of UI prototypes in Figma.

[79] Mohanta, B. K., Panda, S. S., & Jena, D. (2018). An Overview of Smart Contract and Use Cases in Blockchain Technology. 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 1–4. https://doi.org/10.1109/ICCCNT.2018.8494045

[80] Ethereum Whitepaper. (n.d.). Ethereum.Org. Retrieved July 27, 2023, from https://ethereum.org

[81] Markun, L. C., & Sampat, A. (2020). Clinician-Focused Overview and Developments in Polysomnography. Current Sleep Medicine Reports, 6(4), 309–321. https://doi.org/10.1007/s40675-020-00197-5

[82] Polysomnography. (2023). In Wikipedia. https://en.wikipedia.org/w/index.php?title=Polysomnography&oldid=1146272792

[83] Hypnogram. (2023). In Wikipedia. https://en.wikipedia.org/w/index.php?title=Hypnogram&oldid=1166812466

[84] Butin, D., Chicote, M., & Le Métayer, D. (2013). Log Design for Accountability. 2013 IEEE Security and Privacy Workshops, 1–7. https://doi.org/10.1109/SPW.2013.26

[85] Schaefer, C., & Edman, C. (2019). Transparent Logging with Hyperledger Fabric. 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), 65–69. https://doi.org/10.1109/BLOC.2019.8751339

[86] Ferreira, A., Huynen, J.-L., Koenig, V., & Lenzini, G. (2014). A Conceptual Framework to Study Socio-Technical Security. In T. Tryfonas & I. Askoxylakis (Eds.), Human Aspects of Information Security, Privacy, and Trust (pp. 318–329). Springer International Publishing. https://doi.org/10.1007/978-3-319-07620-1_28

[87] Ali, Ahmed, M., & Khan, A. (2021). Audit Logs Management and Security—A Survey. Kuwait Journal of Science, 48. https://doi.org/10.48129/kjs.v48i3.10624

[88] Helland, P. (2015). Immutability changes everything. Communications of the ACM, 59(1), 64–70. https://doi.org/10.1145/2844112

[89] Vaithianathan, G. (2015). A Tamper Proof Log Architecture for Cloud Forensics. A Tamper Proof Log Architecture for Cloud Forensics, 9, 722–727.

[90] Wahid, K. F., Kaufmann, H., & Jones, K. (2017). Tamper Resistant Secure Digital Silo for Log Storage in Critical Infrastructures. In G. Havarneanu, R. Setola, H. Nassopoulos, & S. Wolthusen (Eds.), Critical Information Infrastructures Security (pp. 226–238). Springer International Publishing. https://doi.org/10.1007/978-3-319-71368-7_19

[91] Sayeed, S., & Marco-Gisbert, H. (2019). Assessing Blockchain Consensus and Security Mechanisms against the 51% Attack. Applied Sciences, 9(9), Article 9. https://doi.org/10.3390/app9091788

[92] DefiLlama. (n.d.). DefiLlama. Retrieved July 27, 2023, from https://defillama.com/chains

[93] Sulyman, S. (2014). Client-Server Model. IOSR Journal of Computer Engineering, 16, 57–71. https://doi.org/10.9790/0661-16195771

[94] Guegan, D. (2017). Public Blockchain versus Private blockhain. https://shs.hal.science/halshs-01524440

[95] Dib, O., Brousmiche, K.-L., Durand, A., Thea, E., & Hamida, E. (2018). Consortium Blockchains: Overview, Applications and Challenges.

[96] Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R., & Wang, F.-Y. (2018). An Overview of Smart Contract: Architecture, Applications, and Future Trends. 2018 IEEE Intelligent Vehicles Symposium (IV), 108–113. https://doi.org/10.1109/IVS.2018.8500488

[97] Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W., & Qijun, C. (2017). A review on consensus algorithm of blockchain. 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2567–2572. https://doi.org/10.1109/SMC.2017.8123011

APPENDIX

Software used

Figma

Figma is a powerful design tool used by a lot of UI/UX designers. It supports online collaboration, vector editing, interactive prototyping. What's more, it can be further enhanced by an abundant source of plugins from the design community.

Google Form

Google Form is free online survey management tool. It not only facilitates survey creation, distribution but also display analysis result graphically. A google account is needed before a participant filling the survey; this greatly prevents duplication of participation and makes sure the recycled feedbacks are authentic.

Developer tools used

The following software was used during this thesis project.

Hardhat

Hardhat is a development environment that helps developers in testing, compiling, deploying, and debugging dApps on the Ethereum blockchain. It comes built-in with Hardhat Network, a local Ethereum network designed for development. This greatly improves developer experience when implementing smart contract.

Alchemy

Alchemy is a web3 development platform. We generated an API key from alchemy and used alchemy's "provider" feature to make interacting with Ethereum Blockchain much easier.

Survey questions

The following is the content of the survey. We used this survey to test the UI prototype and gathered 32 responses from participants.

About this survey

Hello,

My name is Buwei Liao, I study information system engineering and management at Karlsruhe Institute of Technology. I'm currently writing a master thesis about "Transparency of data processing within data trustee platform of sleep research" at FZI (Research Center for Information Technology).

For this purpose I created a web application (prototype), and hope to gather real feedbacks with the help from you.

Note: The prototype is designed for web, so it makes sense if you test them on a relatively larger screen, a normal size laptop should be enough.

There are 4 parts and 21 questions in total included in this survey.

1. The first part contains general information about yourself.
2. The second part you will test the clickable prototype and solve some tasks.

3. The third part contains 10 questions about usability of the data trustee plattform.
4. At last, you'll leave a short comment about the system.

This survey might take you about 20 minutes to complete, and the questions are in English.

All collected data will only be used in pseudonymized form for my master thesis.

Part 1: General information

How old are you?

- ○ 18 ~ 29
- ○ 30 ~ 39
- ○ 40 ~ 49
- ○ 50 ~ 59
- ○ 60 ~ 69

Do you use websites or mobile apps on a daily base, and are familiar with using software?

	1	2	3	4	5	
No experience at all	○	○	○	○	○	Professional user

Part 2: Let's solve some tasks!

Scenario introduction: SouveMed is a data trustee platform that helps sleep researchers to get access to data on the one hand, and enables the Individual data donators and sleep research labs to contribute their data through our platform on the other hand. The main idea of SouveMed is to honor your personal data sovereignty and help you take control of your own data.

Now, please assume you are a data donator called Patrick Hubner. You have already uploaded several datasets to the platform.

You can solve the tasks one by one. Here are the suggested setps to solve each task:

1. Read the task description

2. Go ahead to the prototype

3. Jump back here to answer question

4. Then repeat until you finish all four tasks

The prototype is composed of some mock-up pages, so some features might not work perfectly. If you find some problem, please jump back and follow the task descriptions.

Task 1

To start with, please click on the "About SouveMed" menu. In this menu, we try to explain how SouveMed protects your data privacy with some nice illustrations.

Next, click on the "My dataset" menu, you can see what kind of information is included in a dataset SouveMed platform.

Q1: Which one is the possible action for researchers on SouveMed?

- ○ Download my data
- ○ Discover my private information, for example name, address, or mobile number
- ○ Submit data request to SouveMed
- ○ Run data analysis algorithm on their local environment

Q2: What kind of information are included in a dataset? (multiple choice)

- ☐ Questionnaire
- ☐ Polysomnography
- ☐ Hypnogram
- ☐ Gender
- ☐ Age

Task 2

Try to change the setting of your datasets in the "Consent management" menu according to your own preference.

Q3: What are the available setting options for you to use? (multiple choice)

- ☐ Project type: private, public or both
- ☐ Research result will be shared or not
- ☐ Duration of data access

Task 3

Q4: All the usage events that relate to your data are recorded by the platform, you can browse them in the "Usage records" menu.

There are two browsing mode, try "By dataset" menu first

How many projects have used your "Dataset SM-007N95"?

- ○ 21
- ○ 22
- ○ 23
- ○ 24

Task 4

Click on the "project list" menu. Use the filter options in the middle column to find a project called "Tracker3". Click that project and see the details of it.

Note: "Tracker3" is a public project and uses your "Dataset SM-3012K90".

Q5: Why your "Dataset SM-3012K90" is being used by "Tracker3"? (multiple choice)

- ☐ My preference setting allows access from public project
- ☐ The project promised to share research result
- ☐ The project is a public project
- ☐ Data included in the dataset matches with data requirements from researcher

Q6: What is the organization behind the "Tracker3" project?

- ○ University of Freiburg
- ○ FZI
- ○ Sleep track GmbH

Transparency score

How transparent do you think SouveMed platform is?

	1	2	3	4	5	6	7	8	9	10	
Lots of information are still missing	○	○	○	○	○	○	○	○	○	○	I know exactly what's going on with my data

Part 3: SUS usability test

We have applied the original ten questions in our survey (see [13] for detailed questions).