future internet

MDPI

Article

Implementing GDPR-Compliant Surveys Using Blockchain

Ricardo Martins Gonçalves [1,2,*], **Miguel Mira da Silva** [1,2] and **Paulo Rupino da Cunha** [3]

[1] Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal; mms@tecnico.ulisboa.pt
[2] INOV INESC Inovação, R. Alves Redol 9, 1000-029 Lisbon, Portugal
[3] Department of Informatics Engineering, Faculty of Science and Technology, University of Coimbra, R. Sílvio Lima, Pólo II da Universidade de Coimbra, 3030-790 Coimbra, Portugal
* Correspondence: ricardo.martins.goncalves@tecnico.ulisboa.pt

Abstract: The immutability of data stored in a blockchain is a crucial pillar of trust in this technology, which has led to its increasing adoption in various use cases. However, there are situations where the inability to change or delete data may be illegal. European Union's General Data Protection Regulation (GDPR)—applying to any company processing personal data from European citizens—explicitly entitles individuals to the right to rectification and the right to be forgotten. In this paper, we describe the design of a system to deploy and process survey data in a GDPR-compliant manner. It combines an Hyperledger Fabric blockchain to ensure that data cannot be tampered with and InterPlanetary File Systems (IPFS) for storage. With the proposed arrangement, we reap several security benefits enabled by blockchain's immutability without running afoul of the regulations. Furthermore, the proof-of-concept is generic and can easily be adapted to various domains.

Keywords: blockchain; GDPR; personal information; sensitive data; data protection; implementation; hyperledger fabric; IPFS

check for updates

Citation: Martins Gonçalves, R.; Mira da Silva, M.; Rupino da Cunha, P. Implementing GDPR-Compliant Surveys Using Blockchain. *Future Internet* **2023**, *15*, 143. https://doi.org/10.3390/fi15040143

Academic Editor: Cheng-Chi Lee

Received: 3 March 2023
Revised: 28 March 2023
Accepted: 31 March 2023
Published: 7 April 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Blockchain has been the subject of great enthusiasm in several domains. Nowadays, several systems use blockchain to store data to guarantee characteristics such as verifiability [1–3], integrity [4–6], tamper resistance [7–9], transparency [10–12], and removing single points of failure [5,13,14].

The technology's versatility has led to blockchain being used for different applications, such as cloud authentication [15], secure sharing of health records [16], incentive mechanisms for machine learning [17], data sharing frameworks for IoT [18], control and traceability of food supply chains [19], and storing transaction details of pets' adoption process [20], among others.

However, when dealing with personal and especially sensitive data, it is necessary to have some precautions, in particular, to understand and comply with the applicable regulations. The protection of personal data is a transversal concern for every field of technology, from the IoT [21] to the cloud [22]. For example, the General Data Protection Regulation (GDPR) applies to all companies that process the personal data of European citizens, regardless of where they are headquartered [23]. The protections it affords are regarded as a benchmark and have been adopted in similar legislations worldwide.

GDPR entitles the users, or data subjects (DS), to request the deletion of their data (barring some exceptions), which is at odds with the immutability of blockchain, a key pillar for enabling trust in those systems. This conflict raises the need to investigate blockchain solutions for storing personal data without violating the law.

Previously, using a Systematic Literature Review [24], we identified the main challenges of using a blockchain-based system to store personal data and the methods to overcome them. Our main contribution in this paper is a general-use GDPR-compliant system for performing surveys with a variable number of fields involving personal data without compromising security. Our proposal segregates personal data (name, age, phone, email) from the answers to the survey and, consequently, enables its isolated processing. Anonymity is assured since records pertaining to the answers do not contain information that could identify the DS. Further, required separate consents for the two data types enable more flexible processing. The automatic deletion of the survey responses when a deadline expires further assures the correct handling of sensitive data. In summary, the proposed system:

- Complies with GDPR's article 16 (right to rectification);
- Complies with GDPR's article 17 (right to erasure, also known as ”right to be forgotten”);
- Enforces time constraints on the preservation of the data;
- Requires separate explicit consents for processing personal and survey data;
- Ensures processing of the survey data without disclosing the identity of the respondents;
- Ensures that only registered users can answer the survey.

The remainder of this paper is organized as follows: in the next section, we describe our research method, and in Section 3, we provide some common ground on key topics. Then, in Section 4, we address system design, followed by system implementation in Section 5 and evaluation in Section 6. We discuss related work in Section 7 before closing with conclusions that restate research contributions, main limitations, and future work.

2. Research Method

In designing our system, we followed a Design Science Research process conducted according to guidelines defined by [25,26], which is appropriate since design science can be used to create and evaluate “IT artifacts intended to solve identified organizational problems” [25].

DSR consists of six activities: (1) problem identification and motivation; (2) definition of the objectives for a solution; (3) design and development; (4) demonstration; (5) evaluation, and (6) communication [26].

For problem identification and motivation, Peffers et al. [26] state, “Define the specific research problem and justify the value of a solution.” In our case, the problem was to be able to use blockchain to process personal information due to its inherent security characteristics but in a manner compliant with GDPR.

Regarding defining the objectives for a solution, Peffers et al. [26] call on the designers to “Infer the objectives of a solution from the problem definition and knowledge of what is possible and feasible,” which, in our case, was to use blockchain to store surveys containing sensitive personal data while complying with GDPR. Design and development aim to “Create the artifact. Such artifacts are potentially constructs, models, methods or instantiations of new properties of technical, social and/or informational resources”. In our case, we designed and developed a software tool that uses blockchain and InterPlanetary File System (IPFS) to create, store, and access surveys while complying with GDPR. For demonstration, the goal is to “Demonstrate the use of the artifact to solve one or more instances of the problem” [26]. To this end, we created a tool and used fictional personal data to demonstrate and test all system functionalities.

As to evaluation, Peffers et al. [26] state that we should “Observe and measure how well the artifact supports a solution to the problem.” Our evaluation was performed ex post and artificial according to the principles recommended by [27–29]. We evaluated the system after the deployment, with artificial information, and evaluated both the fulfillment of requirements (compliance with GDPR) and performance. Finally, communication is intended to “Communicate the problem and its importance, the artifact, its utility and novelty, the rigor of its design and its effectiveness to researchers and other relevant audiences such as practising professionals, when appropriate” [26]. In our case, it is achieved with this paper.

The Design Science Research was conducted according to guidelines defined by two papers. Design Science "creates and evaluates IT artifacts, intended to solve identified organisational problems" [25]. The artifacts are represented in a structured form [25], in this case, a Software Tool.

DSR consists of six activities: (1) problem identification and motivation; (2) defining the objectives for a solution; (3) design and development; (4) demonstration; (5) evaluation, and (6) communication [26].

Problem Identification: "Define the specific research problem and justify the value of a solution" [26]. In this matter, the global problem is to use a BC to store surveys with sensitive personal information since complying with GDPR is mandatory (If an entity is located or offering goods/services in the European Union [23]).

Define the objectives for a solution: "Infer the objectives of a solution from the problem definition and knowledge of what is possible and feasible" [26]. The objective of this project is to use a BC system to store surveys concerning sensitive personal data complying with GDPR.

Design and development: "Create the artifact. Such artifacts are potentially constructs, models, methods or instantiations of new properties of technical, social and/or informational resources" [26]. In this case, we designed and developed a software tool that uses BC to create, store, and access surveys complying with GDPR.

Demonstration: "Demonstrate the use of the artifact to solve one or more instances of the problem" [26]. We created a tool and used fictional personal data to demonstrate and test all functionalities of the system.

Evaluation: "Observe and measure how well the artifact supports a solution to the problem" [26]. The evaluation was performed ex post and artificial according to the principles recommended by three articles [27–29]. We evaluated the system after the deployment, with artificial information, and evaluated both the fulfillment of requirements (compliance with GDPR) and performance.

Communication: "Communicate the problem and its importance, the artifact, its utility and novelty, the rigour of its design and its effectiveness to researchers and other relevant audiences such as practising professionals, when appropriate" [26]. The communication is performed throughout this paper.

To evaluate the software tool, it performed a link between the challenges found and the approach used to confirm if each challenge was, in fact, overcome. The evaluation regards different aspects of the regulation: the right to rectification and the right to erasure, time constrains, purpose, anonymization, and quality of answers.

3. Background

3.1. General Data Protection Regulation

In recent decades, considerable amounts of personal data have been collected and processed online, frequently without proper authorization or adequate ethical considerations. This situation has led to the introduction of privacy-oriented legislation, such as the General Data Protection Regulation (GDPR) [23].

The GDPR took effect across all European Union member states (EU) in May 2018 [30]. Its purpose is to "protect the rights, privacy and freedoms of natural persons in the EU" and to reduce "barriers to business by facilitating the free movement of data throughout the EU" [31].

The regulation considers personal data as "any information which is related to an identified or identifiable natural person" [23] and sensitive data as "data revealing racial or ethnic origin, political opinions, religious or philosophical beliefs, or trade union membership [. . .] genetic data, biometric data, data concerning health or data concerning a natural person's sex life or sexual orientation" [23].

Sensitive data cannot be processed by default, although paragraph two of article 9 lists a few exceptions. The first is when the data subject gives explicit consent for one or more specific purposes. Other exceptions are narrower; most are connected to health/medical and legal purposes. Article 89 of the GDPR provides other exceptions (safeguards and derogations) relating to processing personal data for archiving purposes in the public interest, scientific or historical research, and statistical purposes [23].

GDPR identifies four stakeholders: the data subject (DS) (the person that the data refers to), the data controller (DC) (who determines the purpose and the processing that will be performed), the data processor (DP) (who processes the data), and the supervisor authority (an independent public authority that is responsible for the enforcement of the GDPR) [31]. It applies to all companies that process the personal data of European citizens, regardless of where they are headquartered [23], and includes hefty sanctions of up to 20 million Euros or 4% of a company's global revenue (whichever is higher)—see article 83 [23].

According to this regulation, the data subject (DS) must consent to the processing and can revoke consent or even request the deletion of the data if there is no legal purpose/reason for the organization to continue using it. Nowadays, any system within the scope of GDPR must be compliant. Blockchain is no exception. When using this technology to store and process personal data, it is imperative to make the required adaptations to comply with the regulation.

3.2. Blockchain

Blockchain was introduced by Satoshi Nakamoto [32] to support Bitcoin. This technology consists of a distributed and tamper-resistant ledger shared by a network of users. This ledger is append-only, which means that once information is entered, it can neither be deleted nor modified. New information can only be added with the consensus of the peers. Blockchains can generally be categorized into permissionless and permissioned. In the former, everyone can maintain the network (publish blocks); in the latter, only authorized users can publish blocks [33].

Some authors argue for four types of blockchain systems: public, private, consortium, and hybrid [34]. A public blockchain is an open platform that anyone can access. All participating nodes have the same authority to verify transactions and validate blocks. Private blockchains are closed networks owned by an entity or organization and restricted to specific users, i.e., new users can only join the network if the blockchain owner accepts them. The owner sets controlling nodes. The consortium mode rests on a community that enables more than one organization to manage a private blockchain. Finally, a hybrid blockchain is a cross of public and private that allows users to decide who can participate and which transactions can be made public [5]. Among the key properties of blockchain are immutability, transparency, availability, privacy, and consistency [35].

Since, due to GDPR, storing personal data directly on the blockchain is impractical, there is a need to use a different system to store this information. IPFS has very similar characteristics to the blockchain, such as full distribution, no need for a third party, and the fact that all the nodes have the same authority in the network.

3.3. IPFS

InterPlanetary File System (IPFS) is a distributed file system (DFS) and a peer-to-peer protocol designed to create decentralized, efficient, and robust data storage and distribution [36]. Conversely to the blockchain, it is possible to delete information and files from IPFS. It combines different technologies to achieve low latency and a content-addressable network. IPFS uses Distributed Hash Tables (DHT) to coordinate and maintain metadata and BitSwap (protocol inspired by BitTorrent) to coordinate networks of untrusting peers and cryptographically authenticated data structures such as git to supper file versioning [36]. A cryptographic hash code addresses all files to ensure tamper resistance and the removal of duplicated files [37].

When adding a file, it is split into smaller chunks, and the hash code of each chunk is calculated and given a Content ID (CID). When another node requests the file (by its CID), it downloads and stores a copy of the file. This makes the second node a provider of the file. The second node can "pin" the content to avoid losing the file. When a file is pinned, its CID is added to a list of CIDs whose files cannot be deleted, called the pinset. If the second node does not pin the file, it will be deleted after a certain amount of time or when a predetermined amount of storage is used. To avoid losing the file, the second node can pin the content [37].

The distribution of files between nodes, the content-addressing, and the use of DHT and BitSwap ensure good resilience, speed, and great censorship resistance to IPFS.

IPFS also enables the creation of private networks; these enable the peers to connect only to others that share the same private key and reject communication from nodes outside that network [38]. Another important feature of IPFS is the possibility of deploying a cluster, a distributed application that orchestrates data across a private network [39]. The cluster coordinates the pinsets (set of pins of each node), enabling collective pinning and unpinning. Furthermore, in an IPFS cluster (contrarily to a regular IPFS network), it is possible to ensure the deletion of a file from all the peers [36].

4. System Design

4.1. Challenges

The main challenges when storing personal data in a blockchain relate to GDPR's right to rectification (article 16) and right to erasure (article 17). These entitlements conflict with the principal characteristic of blockchains, i.e., maintaining permanent and immutable records.

However, compliance is possible without significant security compromises by storing personal data off-chain while keeping only validation information (such as hash codes) in the blockchain.

Processing sensitive data is not always necessary; however, it is indispensable in some cases. For example, to confirm that all ethnicities are treated equally or that there is no discrimination based on sexual orientation. Processing of sensitive information must be performed with GDPR constraints in mind and confirmed with a GDPR specialist.

Despite GDPR mandates, answering highly personal and sensitive questions raises other concerns. To receive the respondents' consent, the surveys must ensure anonymization and erasure of a person's answer if required. To achieve this, we decided to segregate the data provided on registration from the survey answers. This separation enables the deletion and modification of any of the datasets independently. This design choice also enforces anonymity since the identification data are only stored in the registration file, not together with the answers to the survey.

It is also essential that only people registered in the system can answer the surveys and that a person can only answer each survey once.

4.2. Methods

Our GDPR-compliant system stores all personal information off-chain using the InterPlanetary File System (IPFS). These data can thus be modified or removed from the IPFS at the DS request without compromising resilience, thanks to its distributed nature. Additionally, a hash of this data, together with a user ID, timestamp, and consent, is stored on a blockchain (immutable) record. This hash is a kind of proof of the validity of the off-chain data to maintain verifiability, auditability, and tamper resistance. Storing a hash of the personal data on the blockchain guarantees that, although the data can be deleted from the off-chain systems and consequently rectified, it cannot be tampered with. Any alteration to the data can be detected by comparing the hash stored in the blockchain with the hash of the actual file stored off-chain. When data are deleted from IPFS, the hash remains in the blockchain; however, recovering the original data it corresponded to is impossible. In practice, this approach satisfies GDPR's right to be forgotten.

To support surveys, we added (a) individual IPFS files for each answer of each user to each questionnaire, (b) a field to the user records in the blockchain with the surveys that the user answered, (c) a pointer to the IPFS file with the personal data (to enable edition), and (d) a separate database of surveys in the blockchain with the survey IDs, description, deadline, and pointers to all the IPFS files relative to the survey.

This architecture enables the users to choose whether to answer each survey or not and to consent to the processing of each after reading and agreeing with the description and purpose. Since these are stored in the blockchain, they cannot be altered or tampered with. All the surveys have a deadline, after which all the data relative to the survey is deleted. The data processor must store all the results of the processed data (but not the data itself) in another system (if desired) before the survey expires.

Each user in the system has different permissions: The DS can answer, edit, and delete the participation in each survey; the DC can create and delete the surveys; the DP can process all the collected data without knowing the identity of the users that answered the survey.

5. System Implementation

5.1. General Architecture

Our system comprises three main components: the blockchain, supported on Hyperledger Fabric (HLF); a distributed file system, supported on IPFS; and a REST Application Programming Interface (API). The API is a distributed application that communicates with the blockchain and IPFS and can be deployed by any node connected to HLF and IPFS. Users outside the system must access API functions by connecting to any nodes running it. The system supports three types of users with different authorizations and functionalities: data subject, data processor, and data controller.

Hyperledger Fabric stores metadata (user ID, consent, hash, pointer to the information residing off-chain, and timestamp) and administrators' authentication information. All interactions with the blockchain were performed via chaincode (also called a smart contract) that stores the metadata and authentication information in separate and segregated ledgers. The IPFS cluster is used to store all the personal data and the public keys of the user. Finally, an API provides a way to interact with the whole system. This API manages authentication and authorization, separating personal data from non-personal data and storing each in its respective system. After the user inserts the information in a form, the API stores the personal data in a single IPFS file, creating a Content ID (CID) and a hash, and then stores the metadata in the blockchain. The API can be accessed directly from the command line or using a browser to access the provided web interface.

5.2. Network Design

The ordering service and the peers form a Hyperledger Fabric network. In our implementation, the ordering service (machines that sort transitions and create blocks) is formed by three orderers connected. Conversely, peer nodes (machines that host ledgers and chaincode) only connect with other peers inside the same organization. The four peers (Alpha: Zeus, Poseidon; Beta: Hera, Demeter) can use any of the orderer nodes to connect to the ordering service as a whole. To interact with the ledger, the client application can invoke the chaincode using any of the four peers.

Figure 1 illustrates the architecture of the HLF network in our PoC. The "direct connection" represents machines that communicate with each other without any intermediate, while the "indirect connection" represents connections to the ordering service. A peer node can use any orderer to connect to the ordering service as a whole.

We use an IPFS private network with a cluster since these allow the nodes to connect only to other nodes that have the same shared key and reject communications from nodes outside that network [38]. Furthermore, the cluster enables the coordination of file operations across peers with the same secret keys, as well as collective pinning (protect a file from being deleted) and unpinning and, consequently, deletion of a file from all peers [36].

The same VMs that support the HLF peer nodes form Olympus's private cluster. The network consists of four peers that share a private key and a set of files. All peers are part of a cluster to make it possible to have a consistent file list across the network.

Figure 2 illustrates the Olympus IPFS network. Four peers (Zeus, Poseidon, Hera, and Demeter) are connected and share a private key to enable a private network and a cluster. Besides enabling the deletion of a file, this network architecture also enables the recovery of any file if a peer is lost.

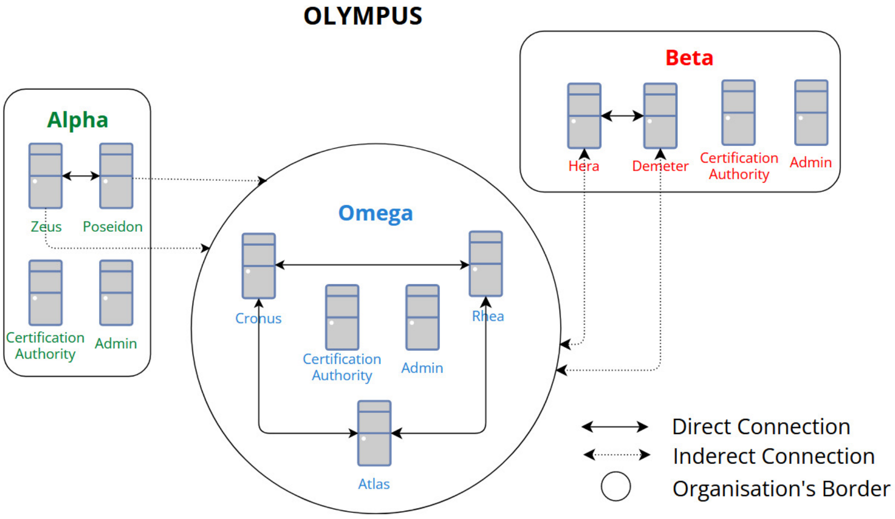


Figure 1. Olympus: HLF Network.

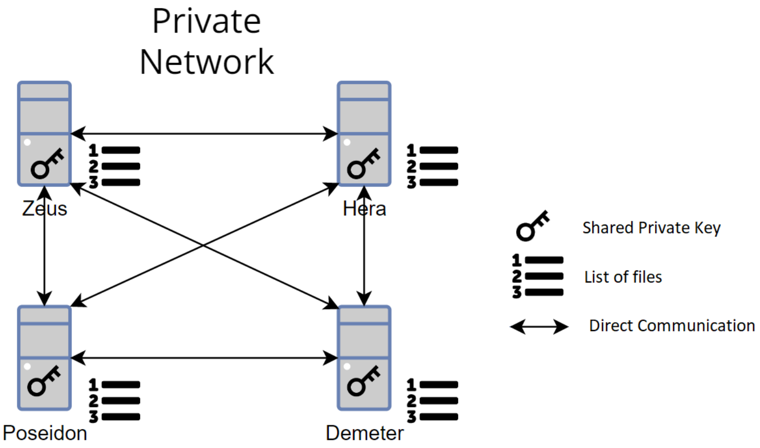


Figure 2. Olympus: IPFS Network.

5.3. Supporting Surveys

To support surveys, we added a ledger in HLF (to hold metadata of the surveys and IPFS CID of the answers), separated files in IPFS, created specific methods in the API and included a mechanism to delete the information of expired surveys.

In our Proof-of-Concept (PoC), the surveys are sets of questions created by the DC with an explicit purpose and a deadline. After the DC inserts the questions, the DS can read the survey description and decide whether to participate. To comply with article 9 of GDPR, the answer to each survey must be optional and time-constrained, and all the information about a survey must be deleted after the deadline [23].

Each answer is stored separately on IPFS to allow deletion and editing independently from the user information entered in the registration and other user answers. One answer to a survey corresponds to a single IPFS file. This method also enables processing information without knowing which user generated the data.

To ensure that only registered users answer the survey, they must authenticate themselves before answering. However, the authentication information is not stored with the answer to the survey. The CID of the answers is stored within the survey ID in the user record on the blockchain to ensure that a user can answer each survey only once, and they can only delete and edit their own files.

To comply with article 9 of GDPR, we request that the user provides specific consent for each survey after reading the description and the deadline. The consent information is stored along with the answer in the IPFS file.

To ensure the automatic deletion of the surveys after the deadline expires, a cronjob requests their list from the blockchain and iterates over each one, comparing the deadline with the current time. The frequency of these checks can be easily changed; we choose two minutes for demonstration purposes. The structure of the survey list is as follows:

{
[Survey1, Deadline1, Description1, [CID1, CID2, CID3]];
[Survey2, Deadline2, Description2, [CID4, CID5, CID6]];
}

When the cronjob detects an expired survey, it starts the deletion process: (1) iterates over the list of CIDs; (2) invokes the IPFS cluster to unpin each CID in the list; (3) invokes the IPFS cluster to run the garbage collector that removes every unpinned file; and (5) invokes the chaincode to delete the survey from their list. The list of answers in the users' records (with survey IDs and CIDs) remains intact.

5.4. Code

This subsection describes the main functions used in the chaincode (CC) and API. All the code used in this project can be found online (REF, accessed on 27 March 2023).

The structure presented in Listing 1 is used to hold the metadata of the surveys (ID, description, fields, and deadline) and the CIDs of the files that contain information relative to the survey.

Listing 1. Survey Structure.

```
type Asset struct {
ID              string ‘json:‘‘ID’’’
Description     string ‘json:‘‘description’’’
Fields          string ‘json:fields‘
CIDs            string ‘json:‘‘cids’’’
Deadline        string ‘json:‘‘dealine’’’
}
```

The code present in Listing 2 is used to create a survey. As with other examples below, data validations are not included here for brevity and comprehensibility. The function receives a survey's ID, description, fields, and deadline and creates an asset with an empty array of CIDs. This array will be populated with the CIDs of answers as the users respond to the questionnaires.

Listing 2. CC to create a survey.

```
func (s *SmartContract) CreateSurvey(ctx contractapi.
    TransactionContextInterface, id string, description string, fields
    string, deadline_str string) error {
asset := Asset{
ID:              id,
Description:     description,
Fields:          fields,
CIDs:            "",
Deadline:       deadline_str,
}
assetJSON, err := json.Marshal(asset)
if err != nil {
return err
}
return ctx.GetStub().PutState(id, assetJSON)
}
```

The code present in Listing 3 is used to read a survey. The function retrieves the asset and parses the data to JSON. The JSON object containing the surveys' metadata and the answers' CIDs is then sent to the API that obtains the IPFS files and presents them to the DC.

Listing 3. CC to read a survey.

```
// ReadAsset returns the asset stored in the world state with given id.
func (s *SmartContract) ReadSurvey(ctx contractapi.
    TransactionContextInterface, id string) (*Asset, error) {
assetJSON, err := ctx.GetStub().GetState(id)

var asset Asset
err = json.Unmarshal(assetJSON, &asset)
if err != nil {
return nil, err
}

return &asset, nil
}
```

The code presented in Listing 4 is used to delete a file across the IPFS cluster. First, the function removes the CID from the pinset of the cluster (unpin); second, it invokes the garbage collector that removes all the unpinned files (the file that was previously unpinned); and third, the function returns the result of the two operations (unpinning and calling the garbage collector).

Listing 4. Delete a file from the IPFS cluster.

```
def delete(cid):
command = ''~/gopath/bin/ipfs-cluster-ctl pin rm '' + str(cid)
proc = subprocess.Popen(command, shell=True, stdout=subprocess.PIPE)
lines = proc.stdout.readlines()
pinned = ''pin is not part of the pinset'' not in lines[2].decode()
#invokes the garbage collector
command = ''~/gopath/bin/ipfs-cluster-ctl ipfs gc''
proc = subprocess.Popen(command, shell=True, stdout=subprocess.PIPE)
lines = proc.stdout.readlines()
removed = ''-'' not in lines[2].decode()
if pinned and not removed:
return True
else:
return False
```

The code presented in Listing 5 is used to remove the survey metadata from the blockchain and the files containing answers relative to that survey. It starts by invoking the read function of surveys in the blockchain. After that, the function iterates over the list of CIDs and deletes each from the IPFS network. When all the IPFS files of that survey have been deleted, the function invokes the HLF to remove the survey metadata from the blockchain. This function can be easily changed to delete only the answers of a survey (the IPFS files), leaving the metadata in the blockchain (ID, description, fields, and deadline) if desirable.

Finally, the code presented in Listing 6 is invoked by a continuously running observer to verify if there are any expired surveys and, if so, invokes the function that deletes surveys (see Listing 5). The observer requests the list of surveys from the blockchain and iterates over it, comparing the deadline of each survey with the present time and deletes the survey if it expired.

In addition to the chaincode, we have created several methods in the API and web pages to support the surveys, namely a page with the available options for each user and a specific page for each method. The API manages the authentication and authorization and separates information into three groups, the answer, the metadata to store in the surveys ledger, and the metadata to store in the user record.

Listing 5. Delete all information of a survey.

```
def delete_survey(survey_id):
survey = get_survey(survey_id)
cids = survey['cids'].split('';'')
cids.pop()
for cid in cids:
IPFS.delete(cid)
Surveys_HLF.delete(survey_id)
```

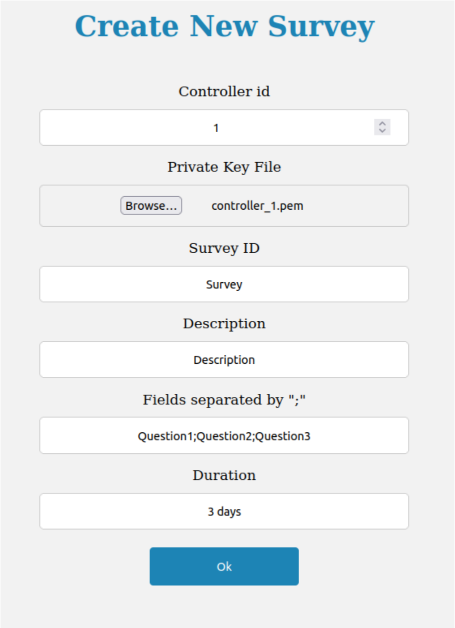
Listing 6. Cronjob to delete expired surveys.

```
def observer():
surveys = get_survey_list()
for survey in surveys:
deadline_str = survey['dealine']
deadline = datetime.strptime(deadline_str, '%Y−%m−%d␣%H:%M:%S')
now = datetime.now()
#if deadline has passed, deletes the survey
if deadline < now:
delete_survey(survey['ID'])
```

5.5. Functionalities by User Type

The data controller can list all surveys (page with the metadata of each survey), create and delete surveys, and add or remove user participation. The data subjects can participate in any survey, edit their answers, remove any participation, and list all the answers given to the current surveys. After a survey is deleted, they can access the former CID of the file with the answer but not the answer given since it was removed from the system. Finally, the data processor can read the information about one or more surveys at once. This function returns the metadata of the survey, followed by the answers.

Figure 3 illustrates the creation of a new survey. This form can only be accessed and filled by the data controller after authentication.



Create New Survey

Controller id

1

Private Key File

Browse... controller_1.pem

Survey ID

Survey

Description

Description

Fields separated by ";"

Question1;Question2;Question3

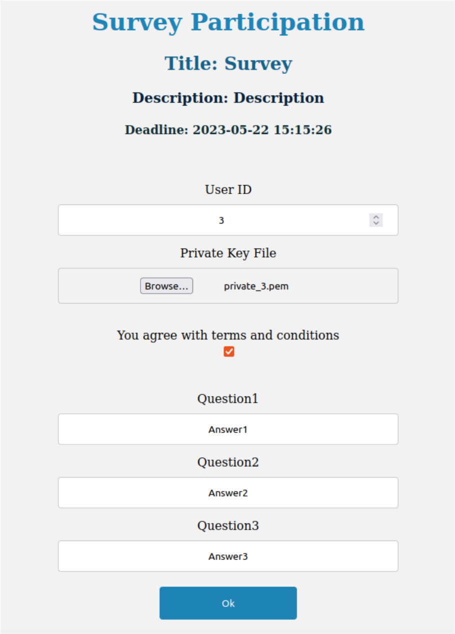
Duration

3 days

Figure 3. Create Survey Web Page.

The delete option of the data controller is very similar to the deletion process invoked by the cronjob. The main difference is the trigger; instead of being invoked when the survey expires, the deletion function is invoked when the data controller has a reason to delete the survey before the deadline, for example, if the survey has a mistake or the necessary information to pursuit the original objective has been collected. After the data controller chooses the survey to delete, the API invokes the chaincode to get the CIDs of the related files and then asks the IPFS cluster to unpin those files and run the garbage collector.

The function used by the DS to answer a survey interacts with HLF and IPFS. After a user selects the survey, they read the description and deadline and check the consent checkbox to proceed to answer the questions. Figure 4 shows an example of an answer to a questionnaire.



Survey Participation

Title: Survey

Description: Description

Deadline: 2023-05-22 15:15:26

User ID

3

Private Key File

Browse... private_3.pem

You agree with terms and conditions

☑

Question1

Answer1

Question2

Answer2

Question3

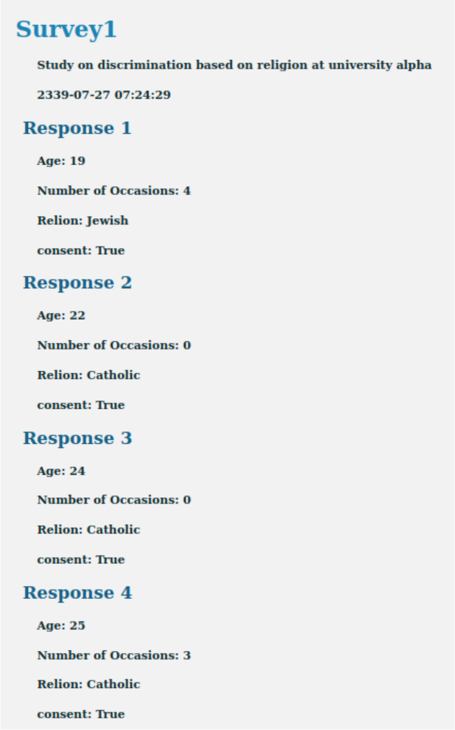
Answer3

Figure 4. Answer Page.

After the user submits the answers to the survey, the API stores the answers and consent in an IPFS file and obtains the respective CID. The CID is then added to the CIDs list of the survey record in the blockchain, and both the survey ID and CID of the answer are stored in the user record in the blockchain.

Users can delete their participation in any survey in two ways. They can invoke the delete function directly or ask the data controller to invoke the same function. The only difference is the authorization process. After the user is authenticated, the API invokes a smart contract (SC) to receive a list of the surveys answered by the user. First, the API verifies if the user has answered the survey; second, it removes the IPFS file from the system; and third, the API deletes the CID from both the survey record and the user record in the blockchain.

Figure 5 illustrates the information that the data processor can access when reading a survey. In this case, the DP chose to read the answers to Survey 1. The API invokes the HLF to retrieve the metadata of the survey and the list of CIDs that correspond to the files with answers to that survey. Then, it invokes IPFS to read all the files whose CID appeared in the list. The web page displays the survey metadata (ID, description, and deadline) and then each participant's answers. However, since the identity of the participants is not stored in the IPFS, the API does not show that information, and the DP can process all the data without knowing which people participated or to whom each answer belongs.



Survey1

Study on discrimination based on religion at university alpha

2339-07-27 07:24:29

Response 1

Age: 19

Number of Occasions: 4

Relion: Jewish

consent: True

Response 2

Age: 22

Number of Occasions: 0

Relion: Catholic

consent: True

Response 3

Age: 24

Number of Occasions: 0

Relion: Catholic

consent: True

Response 4

Age: 25

Number of Occasions: 3

Relion: Catholic

consent: True

Figure 5. Survey's Answers.

6. Evaluation

We performed our evaluation by contrasting the challenges previously identified for GDPR-compliant blockchain systems and our proposed solution to confirm that each challenge was successfully met. We considered different aspects of the regulation: the right to rectification, right to erasure, time-constrains, purpose, anonymization, and the restriction of respondents to authorized users.

The answer to any question is optional to guarantee that users are not forced to enter personal data. The time constraint of the survey and the automatic deletion of the data ensures the data will be deleted when it is not required anymore.

Although the data are not fully anonymized when collected (since the header of the post request contains data such as IP address, browser, and operating system), the data are anonymized immediately after being collected. In particular, the DC and DP cannot know who submitted an answer.

Our proposal also ensures that only registered users can answer the surveys and that they can do it only once.

The separation between the user data provided on registration and the answers to the survey provides a more flexible environment (since the data can be processed, deleted, and modified separately) and ensures greater anonymization (since the answers do not contain information about the respondent). This separation also allows the user to give separate consent to each survey.

Although article 9 of the GDPR has some exceptions when processing sensitive data, these exceptions require knowledge and application of the principle of the lawfulness of processing (article 6) [23] and of the principle of data minimization. In particular, personal data must be "adequate, relevant and limited to what is necessary for relation to the purposes for which they are processed" (article 5) [23].

The data controller must consider all these factors when creating a new survey, as the consent is not absolute. For example, the DC cannot ask for the tax identification number if there is no payment—even if the DS consents. The DC must justify all the fields in a survey.

There are also some precautions to consider about the consent itself, as the consent must be accurate and specific and must be given after justifying the need for the data. For example, the DC cannot ask the DS to give generic consent to process all sensitive data.

The proposal separates the roles of DC and DP for grouping functionalities. However, it is not possible to fully separate these roles. The data processing must be covered by a written contract and carried out by the DP exclusively by permission of the DC.

In the prototype, we added the pointers to the answers to the user record. However, a malicious engineer can link the answers to the DS in a production environment. This challenge can be solved by one of the following methods:

- Remove the pointer from the DS record. In this case, the DS needs to store the pointer manually and cipher the answer with a secret key shared only by the DS, DP, and DC;
- Cipher the answer so that no one (beyond DS, DP, and DC) has access to the answer;
- Completely remove the pointer to the answer. In this case, the DS can no longer access, edit, or delete the answer;
- Use a method, such as a jump server, that limits the operations that an administrator can perform;
- Last but not least, all engineers that access the data must sign a confidential disclosure agreement.

For surveys with a few ranges of possible answers and a small number of questions, a nonce/salt value should be added to the file before uploading to the IPFS network to prevent brute-force attacks. Another caution to consider is that the smaller the group, the bigger the chance of deanonymizing the answers. If the survey has only a few answers, it may be possible to identify the DS. In a production system, the DC and DP could only access surveys with a minimum number of answers to ensure better anonymization.

The software tool overcomes all the major challenges we found in the literature. Regarding the right to be forgotten and the right to rectification [23], the tool uses IPFS private cluster to enable the deletion of files [36] and re-writing of personal data, thus achieving the objective. The surveys are time constrained and automatically deleted after the deadline, assuring accordance with the GDPR principle of storage limitation [23].

The survey also has an explicit (legal) purpose, created by the data controller, that is shown and accepted by the data subject before answering any questions, according to the principle of lawfulness, fairness, and transparency in article 5 of GDPR [23]. Finally, there is no link between the data subject and the answer to assure anonymization and only registered users can access and answer the questions in the survey to ensure data quality.

7. Related Work

This section presents an overview of other projects that use blockchain-based systems to store sensitive information complying with GDPR and compares the solutions found in the literature and our proposal to support surveys.

Some projects already use blockchain to store personal data and comply with GDPR [9,40], but most are meant to solve a specific problem for a specific domain.

Rotondi et al. [40] used DLT to store data concerning work activity. The project stored workers' information as well as the work hours of each employee. The authors used off-chain storage to write personal information and had several cautions concerning GDPR, such as complying with the right to be forgotten, right to rectification and right to data portability. However, the authors did not separate the data types and used the same method to process all the information since all data stored in that system has the same "sensitivity" level.

Some authors separated the types of information before storage in the blockchain system. For example, Truong et al. [9] separated personal data into types (such as logs, information necessary to authenticate, and sensitive information). The authors also separated the DS, DP, and DC functionalities. The project stored consent to process the personal information and used HLF as a blockchain platform. However, conversely to our system, they did not implement a survey functionality.

Onik et al. [41] proposed and implemented a blockchain system to store personal data securely to comply with GDPR. The authors also used off-chain storage to save personal information and required consent before processing personal information. However, the distinction between data types is not the same as in our system. They separated personal information into sensitive and non-sensitive (metadata) and stored the non-sensitive data directly in the blockchain. Furthermore, they also defined two types of personal data, PII and Potential Personally Identifiable Information (PPII), also known as indirect identifiers or quasi-identifiers. The authors stored both PII and PPII off-chain and did not separate the functionalities of each role of GDPR (DS, DP, and DP).

The healthcare sector and the COVID-19 Pandemic inspired several blockchain projects [6,42,43]. Although these authors used blockchain systems to store personal data and take precautions to comply with GDPR, projects related to health and medicine have different criteria and exceptions to comply with GDPR. Personal data are processed with different requirements more suited to the healthcare domain. For example, if someone had COVID-19 and the system registered that information, the DS could not request the DC to delete that information. Furthermore, all types of healthcare-related information can be broadly considered sensitive, i.e., a different definition from the one provided by GDPR.

The projects found in the literature had similar cautions and methods to our proposal, such as separating different types of personal data and independent processing. However, we found no project that used blockchain and off-chain storage to process surveys.

8. Conclusions

In this paper, we proposed and implemented a solution to support surveys based on a blockchain that stores personal data. Although the proposal takes advantage of Hyperledger Fabric and IPFS, the same proposal can be implemented with any blockchain with off-chain storage.

8.1. Research Contributions

Apart from proposing and demonstrating the proposal with a prototype, we evaluated the proposal. Furthermore, we proposed a solution to support surveys that collect personal data and store that data in a blockchain, including the improvements needed for running the proposal in a production environment.

8.2. Main Limitations

Processing and storing personal data needs knowledge of the regulations, not only GDPR but also the laws of each member state. The proposal does not address this challenge that can only be solved by lawyers in each member country.

Although the proposal separates the DC and DP roles, this separation may not be straightforward in practice because it requires interpreting and applying the GDPR.

There is also a need for a trade-off between anonymization and flexibility. If a DS can change and delete his/her data, that means there is a link between the data and the user, so the data are not fully anonymous. The alternative would be to prevent users from accessing their data.

Another main limitation is the impossibility of guaranteeing complete anonymity in this or any other proposal, for example, in an HTTP request (GET or POST), a variety of data can be used to identify the user. Anonymity is always relative.

8.3. Future Work

Higher levels of anonymity of personal data stored in a blockchain is undoubtedly an interesting research topic for future work. In particular, guaranteeing that personal data submitted to questionnaires could not be linked to the user would have tremendous benefits in many business domains. For example, whistleblower platforms (in which anonymity is a significant concern) already have some security, but today it is still impossible to guarantee total anonymity.

Another interesting topic for future research in this area would be improvements to our proposal to guarantee full GDPR compliance, namely defining the DC and DP roles.

Author Contributions: Conceptualization, R.M.G. and P.R.d.C.; Methodology, R.M.G, M.M.d.S. and P.R.d.C.; Software, R.M.G. and P.R.d.C.; Writing—original draft, R.M.G.; Writing—review & editing, M.M.d.S. and P.R.d.C.; Supervision, M.M.d.S. and P.R.d.C.; Project administration, M.M.d.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Guest Intelligence Chain: ALG-01-0247-FEDER-047399 and Fundação para a Ciência e Tecnologia: CISUC R&D Unit—UIDB/00326/2020, project code UIDP/00326/2020.

Data Availability Statement: The algorithms and smartcontracts used in this paper can be found at https://github.com/ricardo-martins-goncalves/olympus.

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

References

1. Hofman, D.; Lemieux, V.L.; Joo, A.; Batista, D.A. The Margin Between the Edge of the World and Infinite Possibility. *Rec. Manag. J.* **2019**, *29*, 240–257. [CrossRef]
2. Alboaie, S.; Ursache, N.C.; Alboaie, L. Self-Sovereign Applications: Return Control of Data Back to People. *Procedia Comput. Sci.* **2020**, *176*, 1531–1539. [CrossRef]
3. Lodha, G.; Pillai, M.; Solanki, A.; Sahasrabudhe, S.; Jarali, A. Healthcare System Using Blockchain. In Proceedings of the 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 6–8 May 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 274–281. [CrossRef]
4. Kolan, A.; Tjoa, S.; Kieseberg, P. Medical Blockchains and Privacy in Austria—Technical and Legal Aspects. In Proceedings of the 2020 International Conference on Software Security and Assurance (ICSSA), Altoona, PA, USA, 28–30 October 2020. [CrossRef]
5. Javed, I.T.; Alharbi, F.; Margaria, T.; Crespi, N.; Qureshi, K.N. PETchain: A Blockchain-Based Privacy Enhancing Technology. *IEEE Access* **2021**, *9*, 41129–41143. [CrossRef]
6. Abid, A.; Cheikhrouhou, S.; Kallel, S.; Jmaiel, M. Novidchain: Blockchain-Based Privacy-Preserving Platform for COVID-19 Test/Vaccine Certificates. *Softw. Pract. Exp.* **2021**, *52*, 841–867. [CrossRef]
7. Chenthara, S.; Ahmed, K.; Wang, H.; Whittaker, F.; Chen, Z. Healthchain: A novel framework on privacy preservation of electronic health records using Blockchain Technology. *PLoS ONE* **2020**, *15*, e0243043. [CrossRef] [PubMed]
8. Wu, G.; Wang, Y. The Security and Privacy of Blockchain-Enabled EMR Storage Management Scheme. In Proceedings of the 2020 16th International Conference on Computational Intelligence and Security (CIS), Guangxi, China, 27–30 November 2020; pp. 283–287. [CrossRef]

9. Truong, N.B.; Sun, K.; Lee, G.M.; Guo, Y. GDPR-Compliant Personal Data Management: A Blockchain-Based Solution. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 1746–1761. [CrossRef]
10. Teperjian, R. The Puzzle of Squaring Blockchain with the General Data Protection Regulation. *Jurimetrics* **2020**, *60*.
11. Sharma, B.; Halder, R.; Singh, J. Blockchain-Based Interoperable Healthcare Using Zero-knowledge Proofs and Proxy Re-Encryption. In Proceedings of the 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS), Bengaluru, India, 7–11 January 2020. [CrossRef]
12. Tatar, U.; Gokce, Y.; Nussbaum, B. Law Versus Technology: Blockchain, GDPR, and Tough Tradeoffs. *Comput. Law Secur. Rev.* **2020**, *38*, 105454. [CrossRef]
13. Marcinkowska, E. Tracking of Clinical Documentation Based on the Blockchain Technology—A Polish Case Study. *Sustainability* **2020**, *12*, 9517. [CrossRef]
14. Parmar, M.; Shah, S. Reinforcing Security of Medical Data Using Blockchain. In Proceedings of the 2019 International Conference on Intelligent Computing and Control Systems (ICCS), Madurai, India, 15–17 May 2019; pp. 1233–1239. [CrossRef]
15. Deep, G.; Mohana, R.; Nayyar, A.; Sanjeevikumar, P.; Hossain, E. Authentication Protocol for Cloud Databases Using Blockchain Mechanism. *Sensors* **2019**, *19*, 4444. [CrossRef]
16. Abouali, M.; Sharma, K.; Ajayi, O.; Saadawi, T. Blockchain Framework for Secured On-Demand Patient Health Records Sharing. In Proceedings of the 2021 IEEE 12th Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON), New York, NY, USA, 1–4 December 2021; pp. 35–40. [CrossRef]
17. Gao, L.; Li, L.; Chen, Y.; Xu, C.; Xu, M. FGFL: A blockchain-based fair incentive governor for Federated Learning. *J. Parallel Distrib. Comput.* **2022**, *163*, 283–299. [CrossRef]
18. Yang, L.; Zou, W.; Wang, J.; Tang, Z. EdgeShare: A blockchain-based edge data-sharing framework for Industrial Internet of Things. *Neurocomputing* **2022**, *485*, 219–232. [CrossRef]
19. Pandey, V.; Pant, M.; Snasel, V. Blockchain technology in food supply chains: Review and bibliometric analysis. *Technol. Soc.* **2022**, *69*, 101954. [CrossRef]
20. Gururaj, H.; Manoj, A.A.; Kumar, A.A.; Nagarajath, S.; Kumar, V.R. Adoption of pets in distributed network using blockchain technology. *Int. J. Blockchains Cryptocurrencies* **2020**, *1*, 107–120. [CrossRef]
21. Laghari, A.A.; Wu, K.; Laghari, R.A.; Ali, M.; Khan, A.A. A review and state of art of Internet of Things (IoT). *Arch. Comput. Methods Eng.* **2021**, *29*, 1395–1413. [CrossRef]
22. Yang, P.; Xiong, N.; Ren, J. Data Security and Privacy Protection for Cloud Storage: A Survey. *IEEE Access* **2020**, *8*, 131723–131740. [CrossRef]
23. European Commission. EU General Data Protection Regulation (GDPR): Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). *Off. J. Eur. Union* **2016**, *679*, 2016.
24. Gonçalves, R.M.; Silva, M.M.d.; Cunha, P.R.d. Using blockchain to store personal information: A systematic literature review. *Int. J. Blockchains Cryptocurrencies* **2022**, *3*, 235–255. [CrossRef]
25. Hevner, A.R.; March, S.T.; Park, J.; Ram, S. Design Science In Information Systems Research. *MIS Q.* **2004**, *28*, 75–105. [CrossRef]
26. Peffers, K.; Tuunanen, T.; Rothenberger, M.A.; Chatterjee, S. A Design Science Research Methodology for Information Systems Research. *J. Manag. Inf. Syst.* **2007**, *24*, 45–77. [CrossRef]
27. Prat, N.; Comyn-Wattiau, I.; Akoka, J. Artifact Evaluation in Information Systems Design-Science Research—A Holistic View. In Proceedings of the PACIS, Chengdu, China, 24–28 June 2014.
28. Pries-Heje, J.; Baskerville, R.; Venable, J.R. Strategies for Design Science Research Evaluation. In Proceedings of the 16th European Conference on Information Systems, ECIS 2008, Galway, Ireland, 9–11 June 2008.
29. Venable, J.; Pries-Heje, J.; Baskerville, R. A Comprehensive Framework for Evaluation in Design Science Research. In *Design Science Research in Information Systems: Advances in Theory and Practice, Proceedings of the 7th International Conference, DESRIST 2012, Las Vegas, NV, USA, 14–15 May 2012*; Peffers, K., Rothenberger, M., Kuechler, B., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 423–438.
30. Rieger, A.; Guggenmos, F.; Lockl, J.; Fridgen, G.; Urbach, N. Building a Blockchain Application That Complies with the EU General Data Protection Regulation. *MIS Q. Exec.* **2019**, *18*, 263–279. [CrossRef]
31. Schwerin, S. Blockchain and Privacy Protection in the Case of the European General Data Protection Regulation (GDPR): A Delphi Study. *J. Br. Blockchain Assoc.* **2018**, *1*, 1–77. [CrossRef] [PubMed]
32. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. Available online: https://bitcoin.org/bitcoin.pdf (accessed on 15 October 2021).
33. National Institute of Standards and Technology. *Blockchain Technology Overview*; Technical Report Federal Information Processing Standards Publications (FIPS PUBS), 2018; U.S. Department of Commerce: Washington, DC, USA, 2018.
34. Kaur, A.; Nayyar, A.; Singh, P. Blockchain: A path to the future. In *Cryptocurrencies and Blockchain Technology Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2020; pp. 25–42.
35. Karthika, V.; Jaganathan, S. A quick synopsis of blockchain technology. *Int. J. Blockchains Cryptocurrencies* **2019**, *1*, 54–66. [CrossRef]

36. Politou, E.; Alepis, E.; Patsakis, C.; Casino, F.; Alazab, M. Delegated Content Erasure in IPFS. *Future Gener. Comput. Syst.* **2020**, *112*, 956–964. [CrossRef]
37. IPFS Community. IPFS Powers the Distributed Web. Available online: https://ipfs.tech/ (accessed on 1 March 2021).
38. IPFS Community. Experimental Features of Go IPFS. 2021. Available online: https://github.com/ipfs/kubo/blob/release-v0.9.0/docs/experimental-features.md#private-networks (accessed on 1 March 2021).
39. IPFS Community. IPFs Cluster. Available online: https://ipfscluster.io/ (accessed on 1 March 2021).
40. Rotondi, D.; Saltarella, M.; Giordano, G.; Pellecchia, F. Distributed Ledger Technology and European Union General Data Protection Regulation Compliance in a Flexible Working Context. *Internet Technol. Lett.* **2019**, *2*, e127. [CrossRef]
41. Onik, M.M.H.; Kim, C.S.; Lee, N.Y.; Yang, J. Privacy-aware blockchain for personal data sharing and tracking. *Open Comput. Sci.* **2019**, *9*, 80–91. [CrossRef]
42. Agbo, C.C.; Mahmoud, Q.H. Design and Implementation of a Blockchain-Based E-Health Consent Management Framework. In Proceedings of the 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 11–14 October 2020; IEEE: Piscataway, NJ, USA; pp. 812–817.
43. Barati, M.; Buchanan, W.J.; Lo, O.; Rana, O. A privacy-preserving platform for covid-19 vaccine passports. In Proceedings of the 14th IEEE/ACM International Conference on Utility and Cloud Computing Companion, Leicester, UK, 6–9 December 2021. [CrossRef]

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