D3. ‘DESIGN SPECIFICATION WITH IMPLEMENTATION APPROACH’

REPUTABLE

Grant Agreement No.: 957338
Call: H2020-ICT-2020-1

Topic: ICT-54-2020
Type of action: RIA

REPUTABLE

D3. 'DESIGN SPECIFICATION WITH IMPLEMENTATION APPROACH'

DUE DATE	06/08/2021
SUBMISSION DATE	06/08/2021
TEAM	REPUTABLE
VERSION	1.0
AUTHORS	Dr Junaid Arshad, Dr Muhammad Ajmal Azad, and Alousseynou Prince

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

EXECUTIVE SUMMARY

The ONTOCHAIN initiative emphasizes trustworthy information exchange which requires a reliable and privacy-preserving reputation system. Through the state of the art analysis conducted as part of D1 of this project, we identified the fundamental requirements for an effective reputation system for ONTOCHAIN. These require a privacy-preserving, decentralised and verifiable reputation system which is able to preserve provenance of reputation information and the reputation scores deduced from this information. Further, the reputation data should be queryable to facilitate user verification as well as seamless integration with systems which may envisage leveraging such information to deliver trustworthy services.

REPUTABLE addresses these requirements and has the potential to deliver a cross-platform privacy-aware reputation system which leverages blockchain technology to achieve decentralised, verifiable calculation of reputation scores. Further it enables interaction with end users and systems through a secure, reputation analytics dashboard to facilitate user verification as seamless integration with other systems and services.

Through the work achieve so far in this project, we have highlighted the need for a trustworthy reputation system and have identified the following design goals for such system.

- User-centric reputation modelling and calculation
- Privacy-preserving user engagement
- Provenance-aware verifiable reputation modelling
- End-to-end decentralisation
- Interoperability with other services of the ecosystem

This document presents a detailed insight into the design specification for the REPUTABLE system in-line with the above design goals. The specification provides a blueprint for the prototype development of the REPUTABLE system for the remainder of the project.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

TABLE OF CONTENTS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

LIST OF FIGURES

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

LIST OF TABLES

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

ABBREVIATIONS

OC1	Open Call 1
PoS	Proof of Stake
PoW	Proof of Work
PoR	Proof of Reputation
REPUTABLE	A Provenance-aware Decentralized Reputation System for Blockchain-based Ecosystems
RDOS	Reputation Data Oracle Service

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

1 INTRODUCTION

1.1 THE PROBLEM

Blockchain is a disruptive paradigm which enables immutable transactions without the presence of a trusted third-party. Consequently, it has been adopted to achieve trustworthy applications across diverse domains such as healthcare, manufacturing, and finance. With such advancements, the need for blockchain-based applications to interact with external services and applications is increasingly evident which requires investigating challenges with respect to trustworthy interactions and efficient management of relevant data.

Reputation systems typically utilise user feedback to evaluate the reputation of a service, affecting its trustworthiness as perceived by the prospective users. Contemporary reputation systems involve a centralized authority to administer, aggregate and analyse user inputs to model the reputation of a service. Although reputation systems can help achieve trustworthy interactions with external services within a blockchain ecosystem, centralized approaches have inherent limitations. Furthermore, as reputation systems rely on user inputs, trustworthy management (collection, tracking, storage & processing) of such data feeds is critical to overall effectiveness of a reputation system. In this respect, although blockchain technology provides basic characteristics such as immutable transactions, there is need to strengthen these capabilities to achieve trustworthy management of external (off-chain) data feeds.

The ONTOCHAIN initiative emphasizes trustworthy information exchange which requires a reliable and privacy-preserving reputation system. Therefore, having a reliable and verifiable reputation system is critical to successfully achieving the objectives of the ONTOCHAIN initiative. In this respect, REPUTABLE facilitates achieving this through a provenance-aware decentralised reputation system which is able to preserve user privacy whilst ensuring unlinkability of the user feedback. A REPUTABLE-enabled ONTOCHAIN framework will be able to provide a mechanism to establish trustworthiness of services (verticals) hosted by the ONTOCHAIN framework leading to higher rate of adoption among users as they are able to rate services.

The primary objective of the REPUTABLE project is to investigate the challenge of achieving reliable and trustworthy reputation modelling for external services within the context of a blockchain-based system such as the ONTOCHAIN platform. We approach this challenge from two specific aspects: i) strengthening core blockchain capabilities to achieve provenance of external data feeds, and ii) exploring development of a

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

verifiable decentralized reputation system which utilizes trustworthy data feeds achieved earlier.

We envisage achieving trustworthy data feeds for reliable reputation modelling through development of a bespoke mechanism to record and store off-chain user input whilst preserving its provenance. With respect to reputation modelling, we propose developing a verifiable reputation modelling mechanism which will enable evaluating trustworthiness of external services whilst protecting user identities through homomorphic cryptography. Through use of cryptographic primitives, an adversary would not be able to learn how a particular stakeholder has rated a particular user. Furthermore, we will explore methods and mechanisms to protect against collusion among participants as well as the ability to publicly verify reputation score calculated by the reputation system.

1.2 ANALYSIS OF RELATED WORK

In this section, we attempt to highlight major conclusions derived from the state of the art and describe the potential of innovation captured by the REPUTABLE project. We specifically address this from the perspective of reputation systems and blockchain-based off-chain data provenance.

In an electronic marketplace, a consumer (consumer, user, or buyer) does not have an opportunity to physically inspect and evaluate the quality of products and services before purchasing them. Therefore, consumers are required to trust the centralized system that provides enough information to consumers to evaluate the trustworthiness of retailers of the marketplace and entities over connected systems. The marketplace (e-commerce, mobile edge or cloud marketplace) enables their consumers to evaluate the trustworthiness of others through the use of reputation systems.

Consumers make use of this available information to decide whether they should interact with the intended partner or hold its transaction until the next available party. The function of these reputation systems are dependent upon the set of users who already have had interaction with the providers that resulted in either negative or positive feedback. These feedbacks are aggregated together to compute the aggregate trust of the entities in the systems. Reputation systems can be classified into two types. 1) a content-driven system that computes the reputation of a retailer using text comments left by consumers for the retailer (retailer, seller, or service provider) (for example, whether the product is received on time, tracking is provided) [1]. 2) a user-driven system that utilizes feedback scores (like, dislike, rating score (0-5)) left by the consumer for their past transactions [2, 3, 4]. The system can also be implemented by combining both content-driven and user-driven systems.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

A typical reputation system for online marketplaces (e.g. eBay, Amazon, Airbnb, Stack-overflow, online dating applications) mainly uses user ratings stored in a trusted centralized setup [2, 3]. In such a case, the trusted system has to protect private information of consumers and ensure the privacy, security, and integrity of the consumer's data. However, consumers can be reluctant to trust the centralized system, especially in providing a negative rating to the particular entity because of fear of retaliation if their negative ratings are exposed to others [5, 6]. Although data anonymization approaches [7, 8, 9] could provide a privacy-preservation layer by hiding real identities of feedback providers, anonymization is prone to de-anonymization and de-identification attacks [10, 11, 5]. For example, Minkus et al. [5] were able to identify private information of eBay consumers by correlating feedback scores left by consumers for their purchases on the eBay network and information from their Facebook profiles. Further, the reputation system could protect privacy of feedback providers by using cryptographic systems [12, 13, 14, 15]. However, these systems not only require high system resources but also rely on a trusted group of users for privacy protection. We summarise the comparative analysis of the REPUTABLE system with existing prominent efforts within research community in Table 1 and those from the commercial reputation systems in Table 2.

Proposal	Architecture	Adversarial Model	Privacy	Complexity	Verifiable
Hassan et al. [14]	Decentralized	Malicious	depends on pre-selected peers	O(n)+O(log n)	No
Androulaki et al. [74]	Decentralized	Semi-honest	compromised if user colludes	O(n)	No
Gudes et al. [12] -1	Decentralized	Semi-honest	depend on witness peers	O(n2)+O(N)	No
Gudes et al. [12] -2	Decentralized	Semi-honest	depends on pre-selected peers	O(1)	No
Zhai et al. [8]	Distributed	Honest	depends on selected peers	O(log n) + O(logn)	No
Schaub et al. [75]	Decentralized	Malicious	Protects privacy	not provided	No
Bethencourt et al. [18]	Centralized	Malicious	depends on trusted party	O(1)	No
Stefanos et al. [76]	Decentralized	Semi-honest	Protects privacy	not provided	No
Clark et al. [77]	Decentralized	Semi-honest	protects privacy	not provided	No

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

REPUTABLE	Decentralized	Malicious/Semi-honest	protects privacy	O(n) + O(n)	Yes

TABLE 1: COMPARISON OF THE REPUTABLE SYSTEM WITH OTHER CENTRALIZED AND DECENTRALIZED REPUTATION SYSTEMS. N IS THE NUMBER OF USERS, AND N IS THE NUMBER OF PRESELECTED USERS FOR PRIVACY PROTECTION.

Broadly, reputation systems can operate in two settings: 1) systems that utilize trusted system for the collection and aggregation of participants feedback and ratings, [2, 3, 16], and 2) system that uses semantics of decentralization and distributed systems to allow user to report their feedback and aggregate the scores without holding the user data a single place [17, 18, 19, 20]. A trusted reputation system could ensure privacy, integrity and confidentiality of participants data through service level agreements, use of cryptographic approach and strong control measures, but the system still poses serious threat to availability of the system resources (reputation data) and privacy of user data. Furthermore, participants in real setup feel reluctant to trust the third-party system for their sensitive data which if leaked or linked to their transaction would bring catastrophic damage to the organization. Furthermore, the service providers could misuse the data for other purposes for which they have not collected this data for example personalized recommendation, falsely manipulating trust of specific entities for financial Gaines etc.

System	Anonymous Identities	Encrypted Rating	Rating Scale	Architecture	Verifiable
Amazon	√	X	(0-5)	Centralised	X
eBay	√	X	(0,1,-1)	Centralised	X
Uber	√	X	(0-5)	Centralised	X
AirBnB	√	X	(0-5)	Centralised	X
Epinios	X	X	(0-5)	Centr alised	X
OpenBazaar	√	X	(0,1 or 0-5)	Decentralised	X
Yelp	X	X	(0-5)	Centralised	X
REPUTABLE	√	√	(0,1 or 0-5)	Decentralised	√

TABLE 2: COMMERCIAL REPUTATION SYSTEMS AND THEIR ATTRIBUTES.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

Major commercial reputation systems have a centralized trusted system architecture [2, 3] responsible for the management and processing of the user's data. Table 1 presents different features of commercial reputation systems. Commercial reputation systems compute reputation of retailers, consumers, and sellers by adding or averaging the rating scores provided by consumers. For example, eBay, a popular auction site, allows buyers and sellers to rate each other as a positive, a negative or a neutral (represented as 1, -1, 0) score after the transaction. The aggregation process is centralized, where the eBay reputation engine computes the aggregated score of sellers and buyers by summing ratings together. The aggregated ratings are then displayed on the page of the seller and the buyer. Epinions.com4 is a general consumer product review site (owned by eBay) that allows users to have a review about the quality of different products before buying them. The Epinions registered users provide ratings (on the scale of 1 to 5 stars) to products and other users. Amazon is a popular website, starting business as an online bookstore in 1994, but now it has become the largest electronic marketplace in the world. Registered users of the site are allowed to rate retailers at the scale of 1-5 stars after the transaction. The system displays the average of all ratings on the web-page about the retailer. Early web search engines simply use the content of the search query to present the top pages to users. However, spammers can evade these systems by simply including the popular search queries in their content. Web search engines now use link-based reputation systems (for example, the PageRank used in a Google search engine) for suggesting reputed pages at the top of the searched query. In PageRank [73], the reputation of the web-page is computed as the number of reputed pages pointing links to the respective page.

Through our literature review, we identified that most of the existing efforts related to this domain by the research community are focused at utilising blockchain's inherent characteristics (consensus, cryptographic hashes, tamper-proof storage, and trustworthy decentralised network) to store provenance of specific application domain/challenge. For instance, [59] presented a blockchain-based reference architecture to preserve completeness, consistency, and naturalness of archival records. Similarly, number of efforts (for instance, [55] and [56]) are aimed at utilising blockchain to maintain data provenance within supply chain systems benefitting from a trustworthy solution within typically trustless supply chain systems. Such efforts are limited in that these present tailored solutions to specific application domains and therefore do not consider challenges such as scalability, automation through smart contracts and oracles, on-chain and off-chain storage scenarios etc.

However, there have been efforts such as ProvChain [57] and DataProv [61] where authors have investigated use of blockchain to manage provenance of data generated in off-chain manner. For instance, ProvChain uses blockchain to store provenance about cloud data objects. In order to achieve

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

verifiability, authors introduce the concept of provenance data receipts which contain transaction ID and block number. Although authors use hashed IDs to achieve anonymous storage of provenance data, which only provide limited level of privacy. Furthermore, both ProvChain and DataProv propose verification mechanisms for data provenance which is achieved in ProvChain through an external auditor and in DataProv through automated scripts. We believe data receipts is an important concept and can be utilised within REPUTABLE system to enable users to confirm that their feedback has indeed been utilised in computation of a reputation score. However, implementing such receipts requires attributing specific transactions to users which may lead to compromise of user privacy and linkability.

With respect to blockchain-based reputation systems, two prominent efforts are Ethereum's node reputation mechanism and Proof of Reputation consensus algorithm [78]. Both these efforts are primarily focused on the reputation of nodes participating in the blockchain network with the view to identify and remove dishonest/lazy nodes. This highlights the fundamental difference between REPUTABLE project and these approaches i.e. REPUTABLE focuses on reputation of services external to blockchain whereas these mechanisms are focused on nodes participating in the blockchain network. Furthermore, Ethereum reputation score, to the best of our knowledge, has been short-lived (a user can start fresh by assuming new identity) which is understood to be addressed by the move to PoS as part of Ethereum 2.0. Similarly PoR also focuses on reputation of nodes participating in the network and utilises reward/punish strategies in to calculate reputation of nodes.

1.3 CONTRIBUTION TO RELATED WORK

ONTOCHAIN aims to develop a trustworthy platform for content sharing and information exchange utilising cutting-edge within blockchains, reputation systems, and semantic web technologies. Therefore, establishing trustworthy information exchange becomes a challenge which requires mechanisms to determine trustworthiness of external services. However, a fundamental challenge in achieving trustworthy systems and services is to advance methods and mechanisms which can ensure provenance of relevant data. The implications of mechanisms governing data provenance are vast and can influence the correctness and accuracy of systems using this data. For instance, marketplaces rely on accuracy of instantaneous as well as historical information to resolve disputes and claims, and therefore rely heavily on trustworthy data feeds. Similarly, reputation systems depend on reliable user inputs to model the reputation of service providers. Therefore, effective management (collection, tracking, processing &

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

storage) of user input is fundamental to trustworthy operation of reputation modelling systems.

Through the state of the art analysis conducted as part of D1 of this project, we identified the fundamental requirements for an effective reputation system for ONTOCHAIN. These are:

- Decentralised, verifiable reputation system to facilitate trustworthy data/information exchange with services external to blockchain ecosystem
- Reputation data collection in a secure, private and unlinkable manner
- Preserve provenance of reputation information to achieve reliable reputation calculation
- An ability for consumers and other third-parties (human or otherwise) to query reputation scores

Our proposed solution takes a holistic approach to achieve a trustworthy decentralized reputation model for blockchain-based ecosystems such as the ONTOCHAIN. Specifically, we envisage achieving trustworthy data feeds for reliable reputation modelling through development of a bespoke mechanism to record and store off-chain user input whilst preserving its provenance. With respect to reputation modelling, we propose developing a verifiable reputation modelling mechanism which will enable evaluating trustworthiness of external services whilst protecting user identities through homomorphic cryptography. Through use of cryptographic primitives, an adversary would not be able to learn how a particular stakeholder has rated a particular user. Furthermore, we will explore methods and mechanisms to protect against collusion among participants as well as the ability to publicly verify reputation scores calculated by the reputation system.

The REPUTABLE system is an innovative solution which addresses these requirements and has the potential to deliver a cross-platform, privacy-aware reputation system. It leverages blockchain technology to achieve decentralised, verifiable calculation of reputation scores. Further it enables interaction with end users and systems through a secure, reputation analytics dashboard to facilitate user verification as seamless integration with other systems and services.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

2 THE APPROACH

In this section, we present an overview of the REPUTABLE system, a high-level architecture and a use case scenario where REPUTABLE can be applied.

Our approach to developing the REPUTABLE system is focused at achieving the following design goals.

- User-centric reputation modelling and calculation
- Privacy-preserving user engagement
- Provenance-aware verifiable reputation modelling
- End-to-end decentralisation
- Interoperability with other services of the ecosystem

2.1 AN OVERVIEW OF THE REPUTABLE SYSTEM

Our proposed solution takes a holistic approach to achieve a trustworthy decentralized reputation model for blockchain-based ecosystems such as the ONTOCHAIN. Specifically, we envisage achieving trustworthy data feeds for reliable reputation modelling through development of a bespoke mechanism to record and store off-chain user input whilst preserving its provenance. With respect to reputation modelling, we propose developing a verifiable reputation modelling mechanism which will enable evaluating trustworthiness of external services whilst protecting user identities through homomorphic cryptography. Through use of cryptographic primitives, an adversary would not be able to learn how a particular stakeholder has rated a particular user. Furthermore, we will explore methods and mechanisms to protect against collusion among participants as well as the ability to publicly verify reputation scores calculated by the reputation system.

The high level architecture of the proposed solution is presented in Figure 1 which provides an insight into the functioning of the proposed system. In the interest of clarity, we describe our approach to reputation modelling and data provenance separately below.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

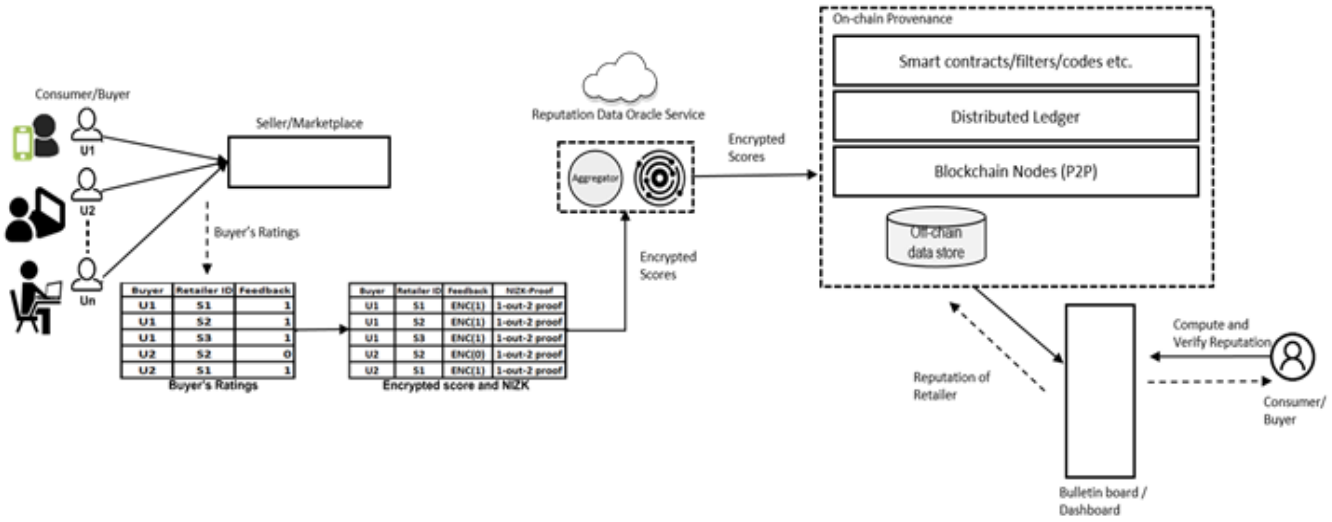

FIGURE 1: HIGH-LEVEL ARCHITECTURE OF THE REPUTABLE SYSTEM

Reputation data and its provenance

Our focus on strengthening core blockchain capabilities is to ensure efficient management of data provenance to support reputation modelling of external services across different phases of the reputation data lifecycle. We achieve this by implementing a dedicated provenance layer (implemented through smart contracts) which is envisioned to collect, track and store metadata associated with reputation data by initiating real-time provenance events.

Within REPUTBALE, the reputation data consists of two different types. Firstly, it is the individual user feedback i.e. the feedback provided by the users when contacted to share their experiences with a service/seller/marketplace. Secondly, it is the aggregate reputation score which is calculated using the individual user feedbacks. As these two types of data are linked with each other, we aim to preserve this linkage and utilise it to achieve verifiable reputation scores. In addition to these two data types, REPUTABLE aims to capture important provenance information such as number of participants, number of responses, and timestamp etc. Such data is crucial to achieve trustworthiness of the proposed reputation mechanism and verifiability of reputation scores.

Processing and storage of reputation data

With respect to processing and storage of reputation data and its provenance, our primary aims are to facilitate decentralised processing, privacy of user feedback, verifiability of reputation score, and secure and scalable data storage.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

Specifically, we plan to store the reputation data (aggregate reputation score) and its provenance in the form of transactions within the consensus blockchain through the execution of smart contracts. Further, the individual user feedback will be stored within an off-chain storage to enable user querying, interoperability with other services within the ONTOCHAIN ecosystem, and integration with external services such as reputation analytics.

As the reputation modelling system is expected to interact with external services and users in an off-chain arrangement, we envision leveraging decentralized oracles to integrate reputation data with the on-chain provenance mechanism. In developing oracles, we will explore strategies for reputation data aggregation supported by appropriate evaluation. The integration of off-chain reputation modelling with the blockchain ecosystem will be performed by developing bespoke smart contracts which will be used to push reputation data onto the blockchain. In this context, Provenance Data Capturing/Logging layer interacts with the smart contracts to enforce the association of provenance related metadata with reputation's data objects, built primarily from the attributes provided by oracles. After carrying out reputation's data analysis and processing, business intelligence scripts can execute data transformation services to rectify and generate metadata for off-chain provenance knowledge management.

Verifiable reputation modelling

With respect to reputation modelling, we will consider two types of users: Data providers (DP) who will provide data or services, and users who might use that data or services. The primary objective is to evaluate the trustworthiness of the stakeholders providing services to others so that the users and consumers can query information about the trustworthiness of stakeholders before initiating any transaction with the service provider.

A reputation system helps in evaluating the trustworthiness of service providers considering their past behaviour, however the development of reputation system comes with two fundamental challenges: 1) ensuring privacy and security of participants and 2) performing the computation in the decentralized setting. To this extent, in this project we will investigate strategies to develop the reputation system in a fully decentralized blockchain-based ecosystem setup with the inherent properties of privacy, security, accountability and unlinkability so that user participation in reputation modelling is anonymised as well as protecting confidentiality of their responses to the reputation query. To ensure user privacy, we plan to use homomorphic cryptographic constructs in the decentralized settings enabling us to compute the aggregated reputation without relying on the trusted system. Furthermore, the reputation scores are envisaged to be stored in off-chain storage and an

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

interface will be provided through implementation of a DApp for users to query reputation scores for specific service providers.

2.2 REPUTABLE FEATURES

In order to achieve the objectives outlined in the previous section, the REPUTABLE system will offer the following features/functionalities. We envisage using these to develop use-case descriptions, use-case diagrams and flow charts in the next section.

- Enable users to provide feedback for suppliers/service providers (user interfaces, communication between REPUTABLE & users etc)
- De-link user identity with the feedback
- Protect anonymity of user
- Aggregate user feedback to create a reputation score for service provider
- Capture and maintain provenance of the reputation scores through the use of oracles
- Achieve decentralised processing of gathering provenance of reputation scores through decentralised oracles
- Store reputation scores on Ethereum blockchain through the use of smart contracts
- Store individual user feedback on off-chain linked with on-chain records. For each campaign aggregate score is stored on-chain and individual feedback values are stored on off-chain. Verification done by comparing aggregate score calculated on off-chain and aggregated data stored on-chain.
- Query interface for reputation scores to enable users to query and verify scores
- Programmable interface to integrate reputation scores with other systems and service (within ONTOCHAIN and external).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

3 DETAILED SOFTWARE DESIGN

This section will present detailed system design, architecture, components, interfaces, and interaction.

3.1 REPUTATION

The fundamental concept within REPUTABLE is that of the *reputation*. In order to explain the concept of reputation as adopted within REPUTABLE, we present a high-level ontological structure for the reputation system in Figure 2.

In figure 1, the reputation is represented by an abstract entity *Reputation Object* which contains information about reputation of a service. The reputation of a specific entity is therefore an instance of the Reputation Object and has collection of attributes which together form the reputation score of a service at a specific time instance. These include *current value*, *historical values*, and *timestamp*.

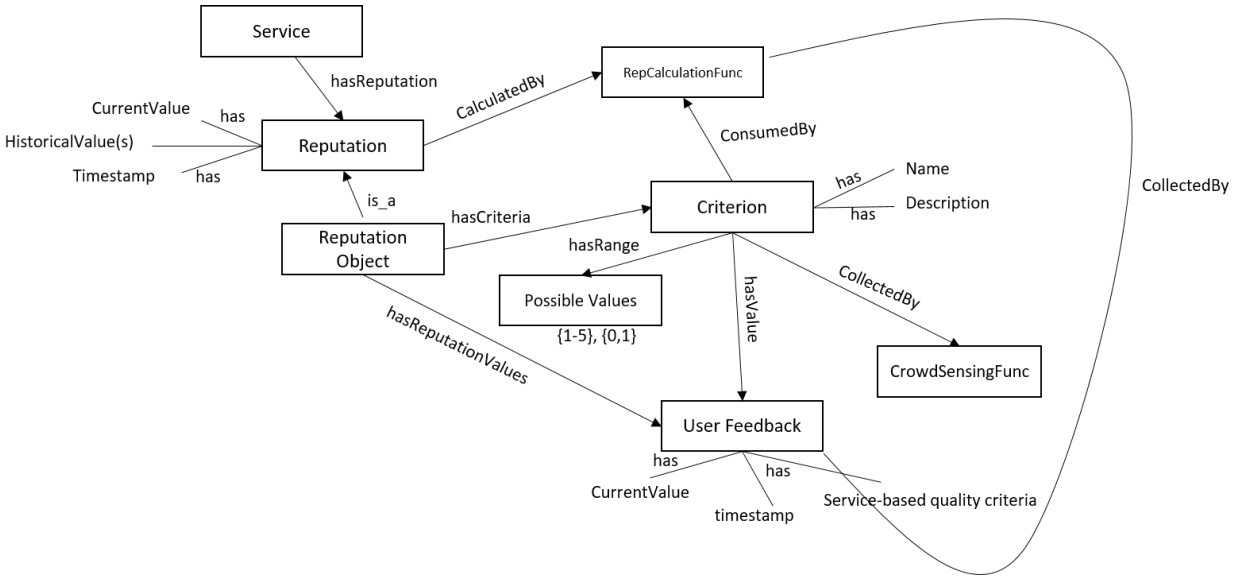

FIGURE 2: A HIGH-LEVEL ONTOLOGICAL STRUCTURE FOR THE REPUTATION CALCULATION MECHANISM

The Reputation Object comprises of a number of criterion which are structured information points and can represent elements of interest about a given service. Examples of criterion can be timeliness of a service,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

driving style of a driver, quality/taste of apples etc. Each user assigns a specific value to a criterion based on their experience and is represented as a user feedback. A user feedback has a value (chosen by the user), timestamp and service-based quality criteria. Each criterion has a set of potential values which in the case of REPUTABLE are {0,1}.

The user feedback is collected by a CrowdSensingFunction which is envisaged to be implemented in the form of a web form to facilitate usability. The CrowdSensingFunction relays the user feedback to the ReputationCalculationFunction which uses appropriate functions to calculate the reputation score for a service at a given time instance.

3.2 SYSTEM ARCHITECTURE

A detailed architecture of the REPUTABLE system is presented in Figure 3 which is an extension of the Figure 1. We present details of prominent components of this architecture below.

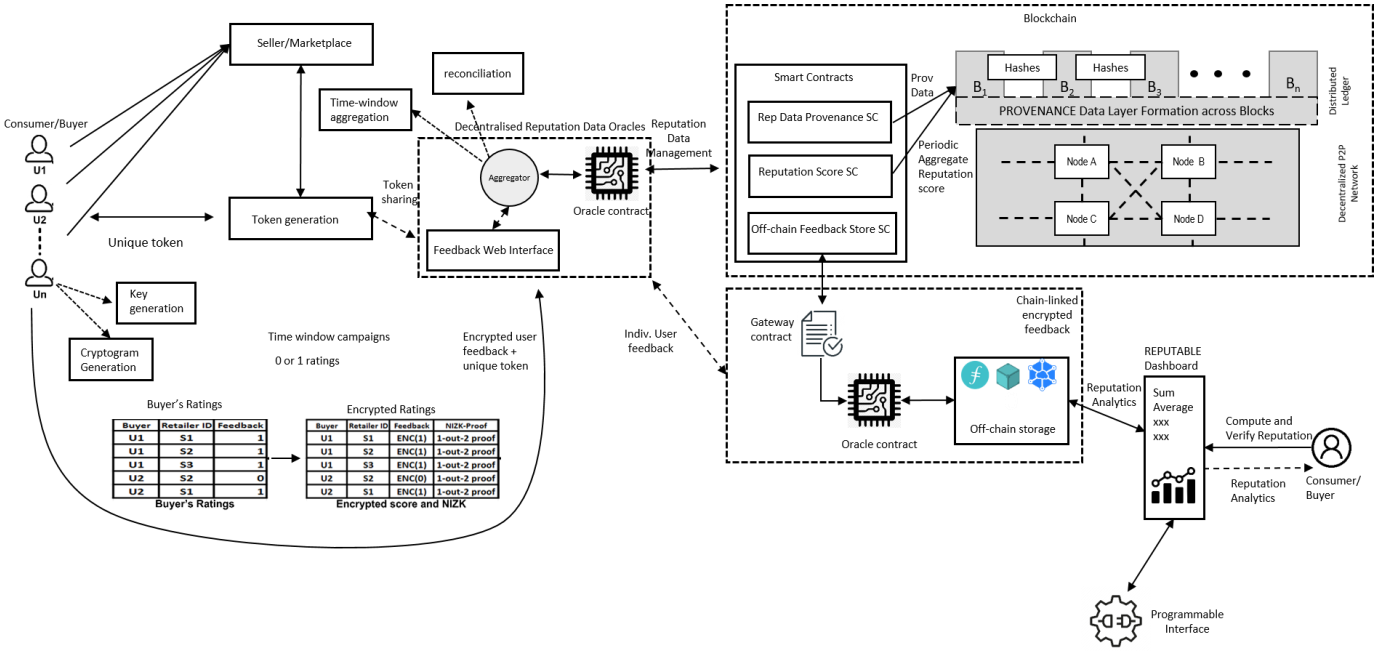

FIGURE 3: DETAILED SYSTEM ARCHITECTURE FOR REPUTABLE

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

Token Generation

One of the core concepts within any reputation system is the validity of the feedback or the question: How to determine whether a user is qualified to provide feedback on a service? Within REPUTABLE, we answer this question through the user of tokens. These tokens are issued by the seller against each valid purchase and are unique for a purchase. Once these tokens are generated, the seller distributes them to the qualifying customers through a communication medium such as email. As these tokens are generated by the seller, it makes this process susceptible to collusion attacks i.e. collusion between seller and the customer. We acknowledge this and consider this as an opportunity for future enhancements of REPUTABLE. For the current version of REPUTABLE, we assume seller and customer to be *honest but curious*.

User Engagement

Within the REPUTABLE system, a consumer of the marketplace is not required to anonymize his identity; instead, they hide their ratings by presenting cryptograms of ratings. The cryptograms, in this case, are encrypted feedback values where the encryption keys have been generated by the user. This features achieves end-to-end decentralisation for the REPUTABLE system whilst also avoiding susceptibility to collusion by a central authority which may be responsible for cryptogram generation. The value of the rating score (0 or 1, like or dislike, rating between 1 to 5 stars) is encrypted using cryptographic primitives as shown in Figure 3. To this extent, the adversary on the reputation system or the reputation system itself would not be able to learn how a particular consumer has rated a particular retailer or another interacted consumer. The REPUTABLE system could provide maximum privacy unless a maximum number of consumers (n-1) in the system collude to find the rating score of the target consumer. Furthermore, the design choice of REPUTABLE ensures two other properties: 1) it limits consumers to provide rating scores within the prescribed range, and 2) it provides public verification of the reputation score stated by the marketplace.

Reputation Calculation

Once consumers have submitted their cryptograms and NIZK proofs to the bulletin board, any entity (participant, marketplace, or analyst) can compute the aggregated reputation of the retailer. This can be done by simply multiplying the cryptograms from the bulletin board.

At this point, we already have the aggregate sum of positive ratings i.e. the sum of consumers who have shown trust (1) on the retailer. The number of negative rating can be computed by subtracting the positive ratings from the total number of users who have provided ratings. The simplest

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

approach to compute the reputation of the retailer is to use the negative and positive ratings together, i.e. subtracting negative ratings from the positive ratings. We use the beta reputation system to compute the final aggregated reputation of the business entity or the retailer E on the marketplace. Let n be the number of consumers providing ratings, P_E represents the number of consumers who provided positive ratings about entity E, and N_E represents the number of consumers who rated the entity E as non-trustworthy, then the final reputation RE_E of an entity can be computed as follows:

$$RE_E = (P_E - N_E) / (n + 2)$$

The system can be easily extended to other reputation systems, e.g. the average of ratings can be computed by simply averaging the sum of individual ratings over the number of users.

Dashboard

The dashboard is an important component in our proposed architecture. This component is envisaged to provide an interface to consumers (those providing feedback) and other interested third parties to query reputation scores. We envisage establishing this service off-chain potentially utilising cloud infrastructure both as a web-based end point as well as a programmable interface. The flexibility of having a programmable interface for dashboard enables collaboration with other components of the ONTOCHAIN ecosystem as well as external services (such as reputation analytics) which can benefit from the output of the REPUTABLE system. Furthermore, an off-chain implementation also means that users will not have to install specific modules/plugins (MetaMask etc.) to access and interact with this service.

Blockchain and on-chain storage

Blockchain is a core component of the REPUTABLE system. It enables end-to-end decentralisation whilst also providing immutable, tamper-proof storage for reputation data (thereby facilitating trustworthiness and verifiability of reputation data). Within REPUTABLE system, reputation data consists of two different types. Firstly, it is the individual user feedback i.e. the feedback provided by the users when contacted to share their experiences with a service/seller/marketplace. Secondly, it is the aggregate reputation score which is calculated using the individual user feedbacks. As these two types of data are linked with each other, we aim to preserve this linkage and utilise it to achieve verifiable reputation

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

scores. In addition to these two data types, REPUTABLE aims to capture important provenance information such as number of participants, number of responses, and timestamp etc. Such data is crucial to achieve trustworthiness of the proposed reputation mechanism and verifiability of reputation scores.

Specifically, we plan to store the reputation data (aggregate reputation score) and its provenance in the form of transactions within the consensus blockchain through the execution of smart contracts. Further, the individual user feedback will be stored within an off-chain storage to enable user querying, interoperability with other services within the ONTOCHAIN ecosystem, and integration with external services such as reputation analytics.

Off-chain storage and connectivity with blockchain

As highlighted earlier, we envisage storing raw user feedbacks on an off-chain storage to facilitate user verification, querying, and interoperability. In addition to the scalability benefit, this choice also enables implementing bespoke security layer to protect access to functions exposed by the REPUTABLE interfaces.

Furthermore, in order to achieve connectivity between on and off-chain components, we envisage using oracles for effective interoperability and linkage between on and off-chain storages. In this regard, the reputation data oracle service (RDOS) is envisaged to be responsible for managing the process of interacting with users to gather their feedback. RDOS achieves this by generating encrypted election data in accordance with the reputation model (0-5, 0,1, …).In terms of the choice for rating scales and compatibility with diverse rating scales, work is in progress and we aim to address this as part of the design specifications and on-going discussions with the ONTOCHAIN coaching team.

Although RDOS acts as an aggregator bordering the blockchain ecosystem, we aim to develop it as a service which can be implemented as a decentralised oracle or hosted on the cloud. In this regard, we have explored the dOracle service offered by iExec or Chainlink and aim to investigate this further as part of the design specification phase. Our motivation here is to achieve complete decentralisation and dOracles by iExec facilitate this. Similarly, if implemented on the cloud, the RDOS service can benefit from the security and reliability offered by the infrastructure to avoid it becoming a central bottleneck in otherwise completely decentralised architecture.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

3.3 UML DIAGRAMS

In this section, we present further insight into the design of the REPUTABLE system. This is achieved through different UML diagrams including sequence diagram, class diagrams, use-case diagram, and flow charts of different components of the REPUTABLE system.

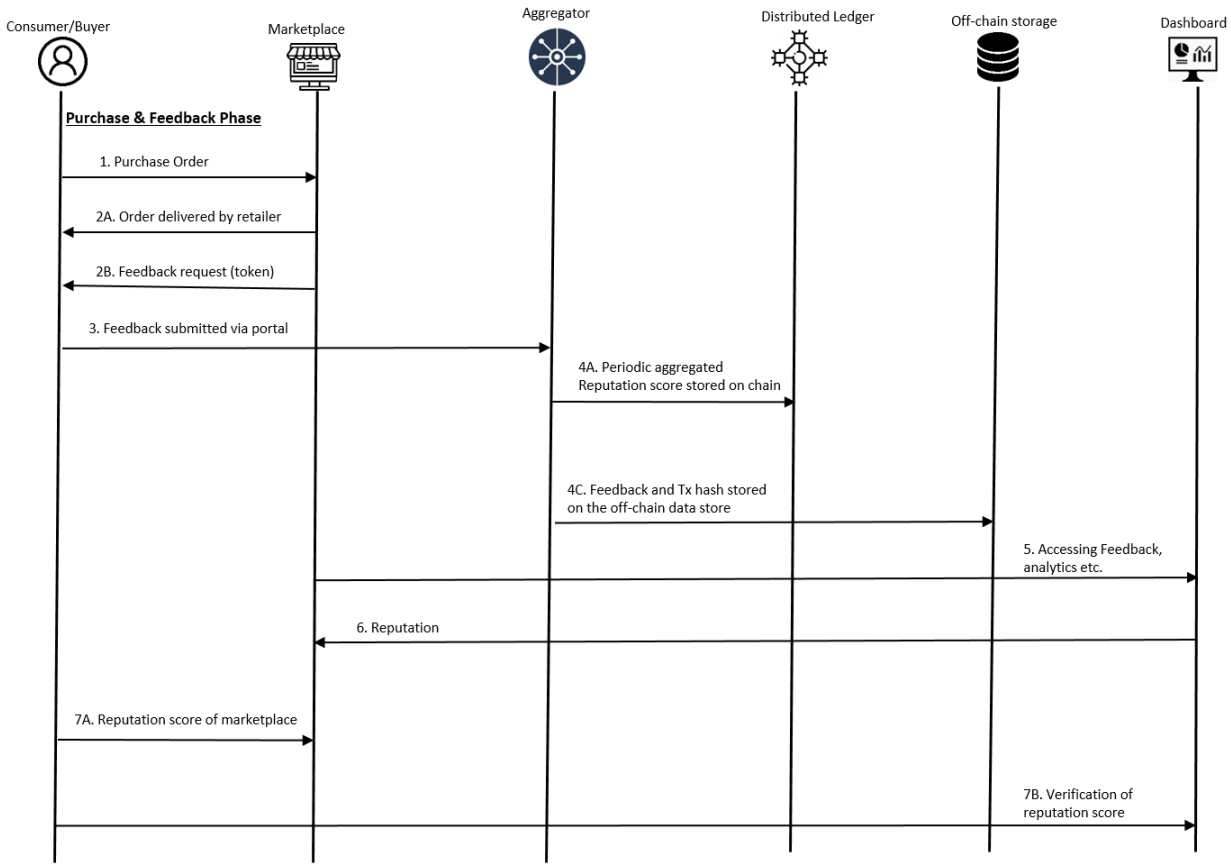

FIGURE 4: SEQUENCE DIAGRAM FOR THE REPUTABLE SYSTEM

Figure 4 presents the sequence diagram for the REPUTBALE system highlighting interactions between different components. In the interest of readability and understanding, we have presented it at a high-level and is augmented by the detailed system architecture presented in Figure 3.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

Figure 5 and 6 present the class diagrams for the REPUTABLE system where Figure 5 is focused on the user engagement and on-chain interaction whereas Figure 5 is focused on the off-chain storage and dashboard feature of the system. Details of specific classes for Figure 5 are presented below.

Web interface:

The web interface class includes attributes such as campaignId, token, user_rating and user_addr. The methods of the class are submit_feedback, generate_keys, generate_cryptograms, rating_submitted_event. These methods are used in allowing the buyer to enter provide a rating to the seller.

Aggregator:

The aggregator class has attributes and methods for allowing individual scores to be turned into aggregated scores. To achieve this, the aggregator has an attribute called ind_scores which is a mapping which consists of a struct of user_score, seller_address and campaignId. The aggregator also has a seller_tokens mapping of a struct which consists of user_token, used (a Boolean to determine if the token was already used), exp_responses – the expected amount of responses for a specific campaign and seller.

The aggregator also has methods such as store_ind_score, retrieve_ind_score, isTokenValid, aggregate and getResponseCount.

Gateway:

The gateway class is responsible for handling data, whether that is fetching aggregate score, storing individual scores or fetching individual scores. It achieves that through functions such as: fetch_aggr_score, store_ind_scor_off_chain and fetch_ind_scores. The gateway also has two attributes. The first one called ind_scores keeps a mapping of a struct of user's score, seller address and campaignId, the other attribute keeps a mapping of seller's hash with a struct of user token, a Boolean of whether the token was used as well as the expected number of responses.

OnchainReputationData:

The onchain reputation data class is involved with storing reputation scores on the onchain storage. It has four methods: add_rep_score, rep_score_added_event and get_rep_score_event. These methods respectively add reputation data onchain, notifies the web interface when the data has been added to the onchain storage and returns the data necessary to allow the web interface to show the rating of a particular seller.

The scores attribute is a mapping that consists of a struct of seller score, campaignId and number of responses.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

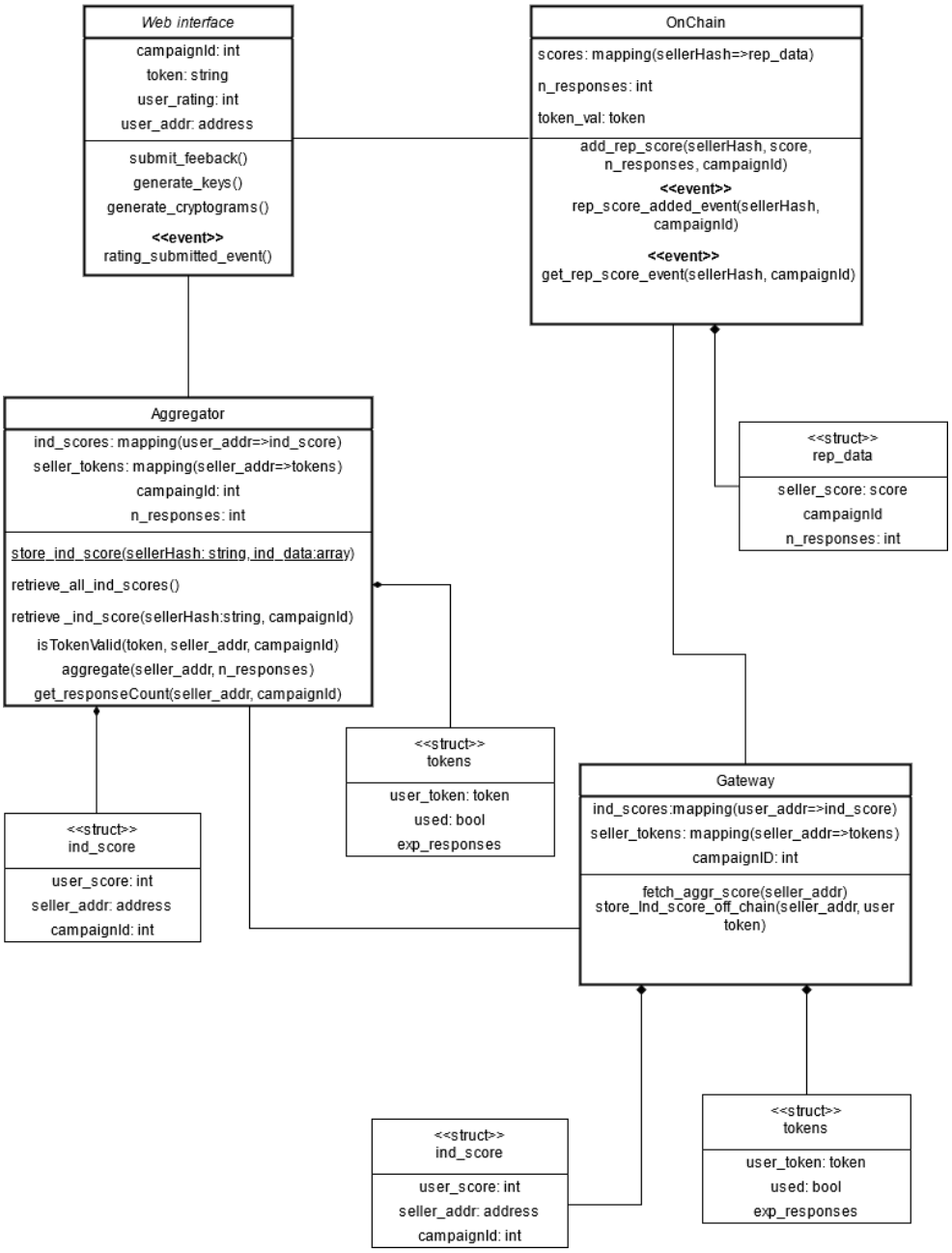

FIGURE 5: CLASS DIAGRAM FOCUSING ON FEEDBACK GATHERING, AGGREGATION AND ON-CHAIN STORAGE

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

OnchainProvenanceData:

The onchainProvenance data class has two methods, add_prov_data and concat_ind_scores. The former stores data such as campaignId, timestamp, number of responses, tokens and individual scores into the onchain provenance data storage. The latter method (concat_ind_stores) concatenates individual scores onto a string.

The scores mapping stores the campaignId, timestamp, number of responses, an array of tokens and an array of individual scores.

ReputableAPI:

The reputable API class exposes the system with methods such as register_tokens, store_ind_score_off_chain, aggregate, add_prov_data_on_chain, add_score_on_chain, generate_pkis and generate cryptograms.

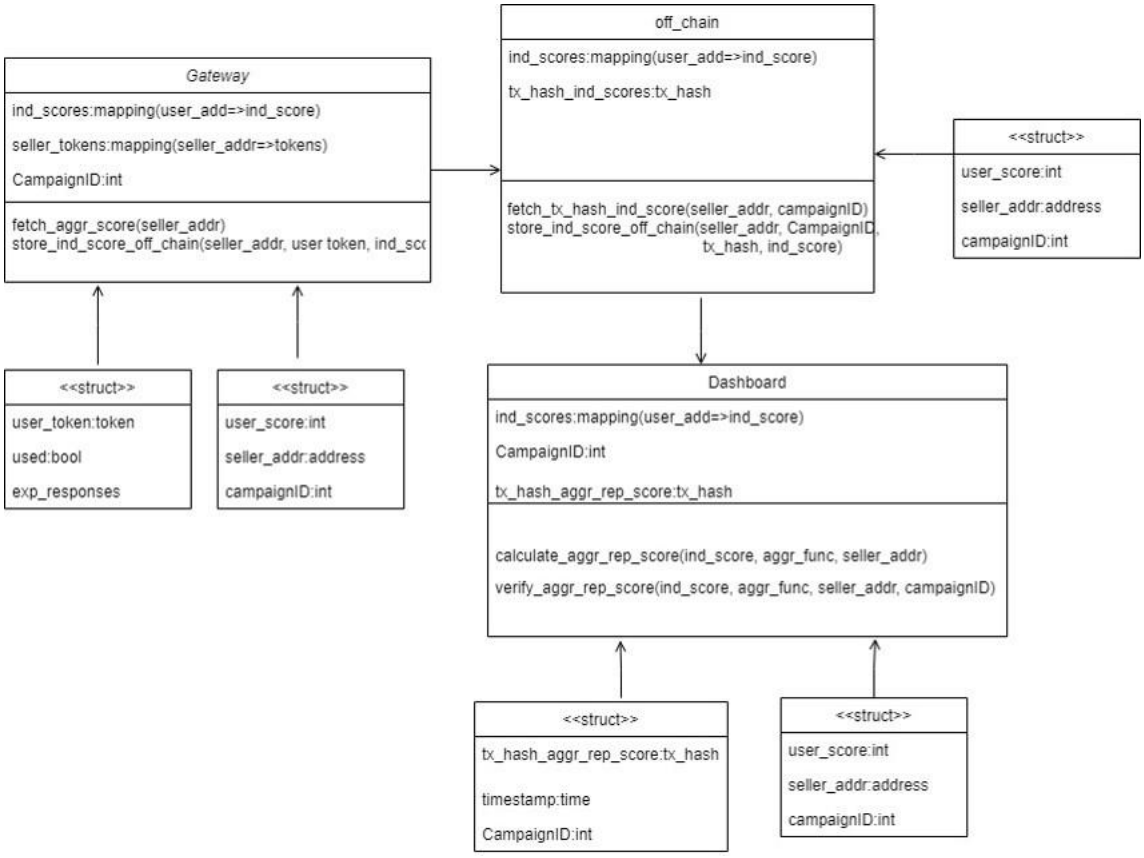

FIGURE 6: CLASS DIAGRAM OFF-CHAIN STORAGE AND REPUTATION DASHBOARD

As represented in Figure 6, we envisage developing a gateway smart contract which will act as a bridge between the on-chain and off-chain storage of

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

reputation data. With respect to off-chain storage, this contract will fetch individual user feedback for a specific seller for a given campaign and relay these to an oracle contract responsible for off-chain storage. In addition to the user feedbacks, this contract also allows access to campaign ID and seller address so as to uniquely identify a feedback campaign for a seller.

The *Off-chain* contract represents an oracle contract which is responsible for storing individual reputation data to an off-chain storage such as cloud, storJ or IPFS. This is an important contract as it achieves linkage between on and off-chain storage. This is achieved by getting the transaction id (hash) of the on-chain storage of individual feedbacks for a given campaign. By doing so, it enables integrity as any changes to a user feedback can be tracked by comparing the hash of the raw feedback with that stored on the chain. Furthermore, it also facilitates as the hash stored on the chain is generated using all user feedbacks as an input and ay missing feedback will generate a mismatch with the hash stored on the chain.

Dashboard is an important feature of the REPUTABLE system. It enables querying of reputation data by end users as well as by other systems and services. The dashboard class has access to individual user feedback, campaignID, and transaction hash for user feedbacks. It enables users to conduct predefined queries such as those used by the aggregator to calculate aggregated reputation score and to query historical reputation scores.

Use-case Diagram

A use-case diagram for REPUTABLE system is presented in Figure 7. The actors involved in the use case diagram are the buyer and the seller who will perform actions such as view rating and rate seller.

The smart contracts involved in this system also have use cases. The web interface through which the user interacts with the system has a user rating a seller which includes submitting feedback which in turn belongs to the web interface.

The web interface has links to the get_rep_added_event as well as rating_submitted_event which simply allow the web interface to stay in the loop of actions occurring in the system.

The web interface smart contract has use cases such as generate_cryptograms, generate_keys and submit_feedback.

The aggregator has a lot of use cases in this system. These use cases are store_ind_score, retrieve_ind_score, aggregate rating, store_all_ind_scores, fetch_ind calculate_aggr_rep_score,

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

fetch_ind_scores, verify_aggr_rep_scores, getResponseCount and isTokenValid.

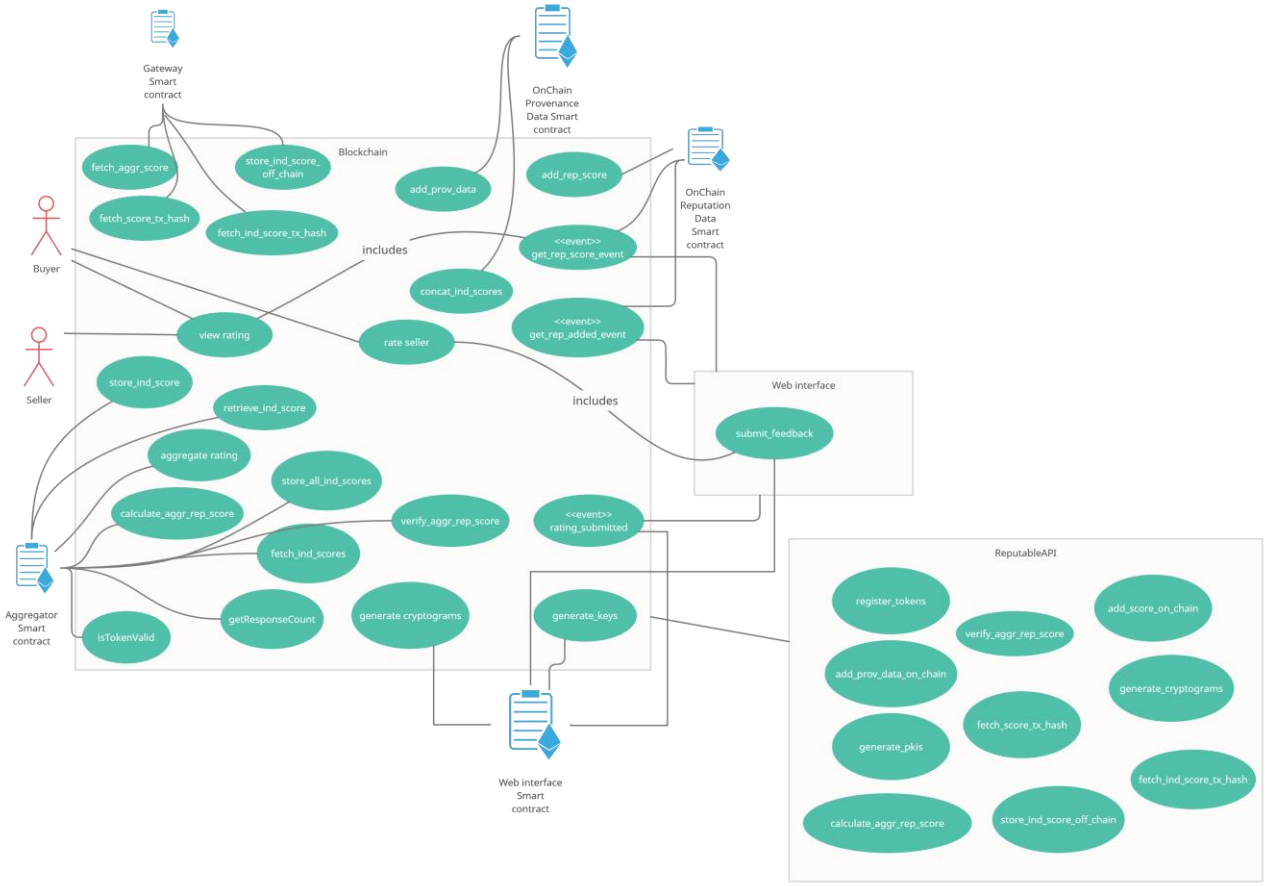

FIGURE 7: USE-CASE DIAGRAM FOR REPUTABLE

The gateway smart contract's use cases are fetch_aggr_score, store_ind_score_off_chain, fetch_score_tx_hash and fetch_ind_score_tx_hash.

The onchain provenance data smart contract has two use cases: add_prov_data and concat_ind_scores.

The Onchain reputation data smart contract has several use cases which are involved in adding the reputation score to the onchain storage as well as events for alerting the web interface when certain changes are made. The use cases are: add_rep_score, get-rep_score_event and get_rep_added_event.

The view rating use case includes the get_rep_score_event as that use case is responsible for returning the data required to view the rating.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

The use case is also connected to an API which interacts with the entire system through six use cases – register_tokens add_prov_data_on_chain, generate_pkis, store_ind_score_off_chain, add_score_on_chain and generate_cryptograms.

3.4 PRIMARY USE CASES AND THEIR DETAILS

In this section, we present details of primary use cases support by use-case description and flow charts to provide insight into the challenges and design choices.

Enable user feedback: Associates a set of questions to a set of answers and asserts that the buyer bought a product or a similar item from the seller before being able to respond to survey questions. It then gathers the feedback through the dashboard

Name	**Enable user feedback**
Purpose	Rating Questions sent to buyers once he buys product from the seller.
Prerequisites	The set of questions and associate set of answers.
Main Narrative	Only user having buy product and associate token could response to the survey questions.
Alternative	
Outcome	Feedback value are collected on the dashboard
Alternative Narrative	-
Outcome	Encrypted feedback scores.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

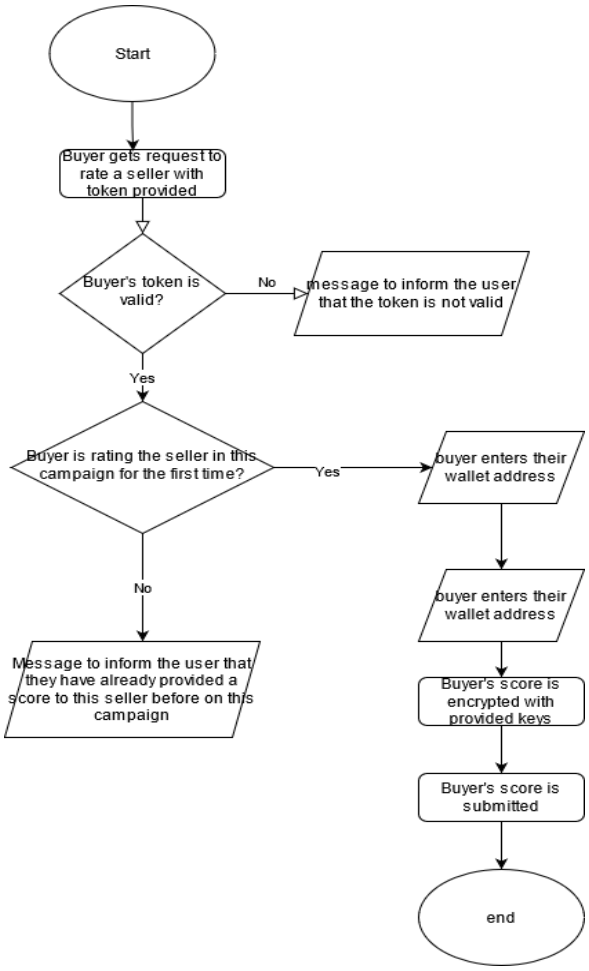

FIGURE 8: FLOW CHART FOR USER INTERACTION WITH WEB INTERFACE

Rating aggregation: Aggregates the seller's score (rating) that is encrypted with zero knowledge proof protocol.

Name	**Aggregate Rating**
Purpose	Aggregate Encrypted Ratings of Seller or Raters
Prerequisites	Encrypted feedback score and associate Zero Knowledge proofs
Main Narrative	Participants will have aggregated score of the seller or raters
Alternative	
Outcome	• The seller aggregated score is computed • The prove has been done to ensure participants are providing honest feedback
Outcome	Aggregated reputation of the Seller

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

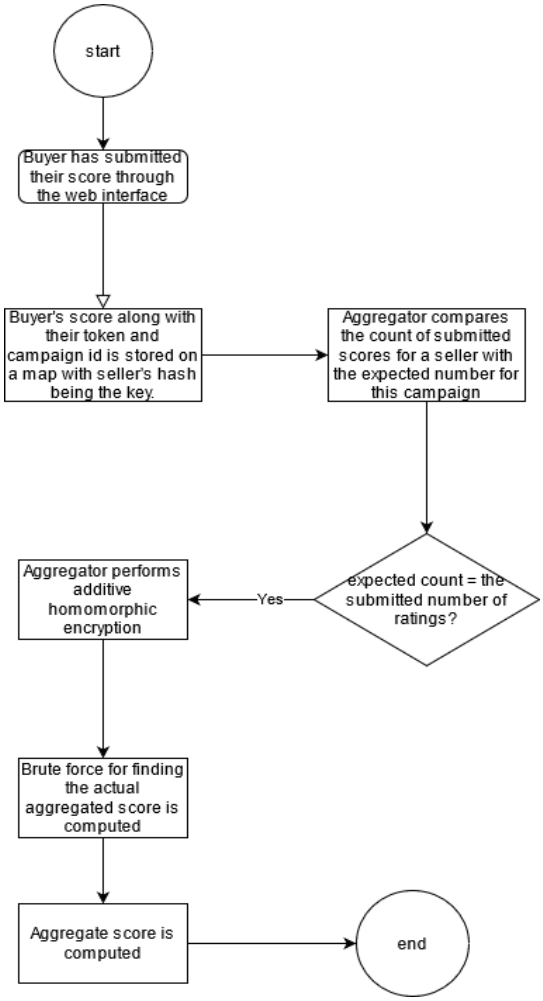

FIGURE 9: FLOW CHART FOR GENERATING AGGREGATE REPUTATION SCORE

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

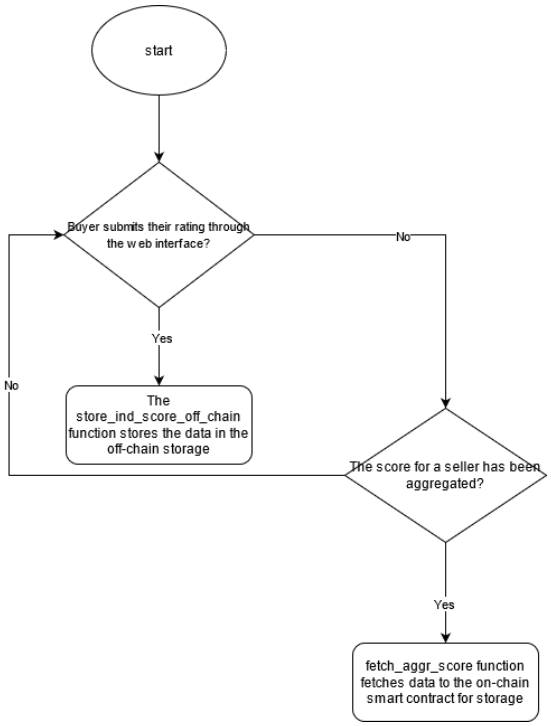

FIGURE 10: FLOW CHART FOR GATEWAY CONTRACT

Add score on-chain: Though the smart contract, pushes the aggregated reputation score which will be submitted onto the block before being validated and broadcast onto the blockchain.

Name	**add_score_on_chain**
Purpose	Adds the reputation score to the blockchain
Prerequisites	• The buyer must have rated the seller prior to it being added into the blockchain • The aggregator has received scores from all contacted buyers • The reputation score is aggregated
Main Narrative	• Aggregator (decentralised oracle service) pushes the aggregated reputation score through the smart contract • Smart contract submits the aggregated score onto the block
Alternative	
Outcome	• The transaction is submitted for validation in the blockchain
Alternative Narrative	-
Outcome	-

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

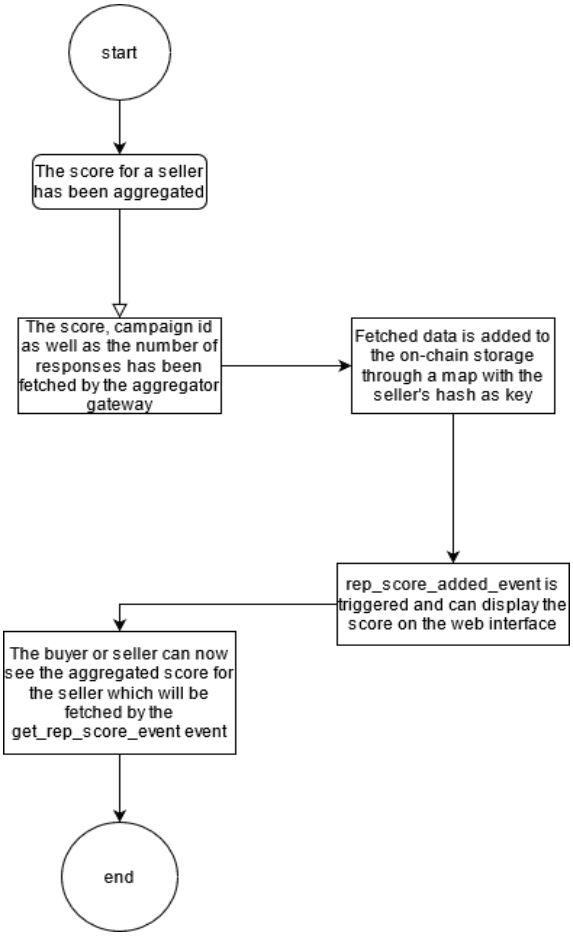

FIGURE 11: FLOW CHART FOR STORING AGGREGATE FEEDBACK ON-CHAIN

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

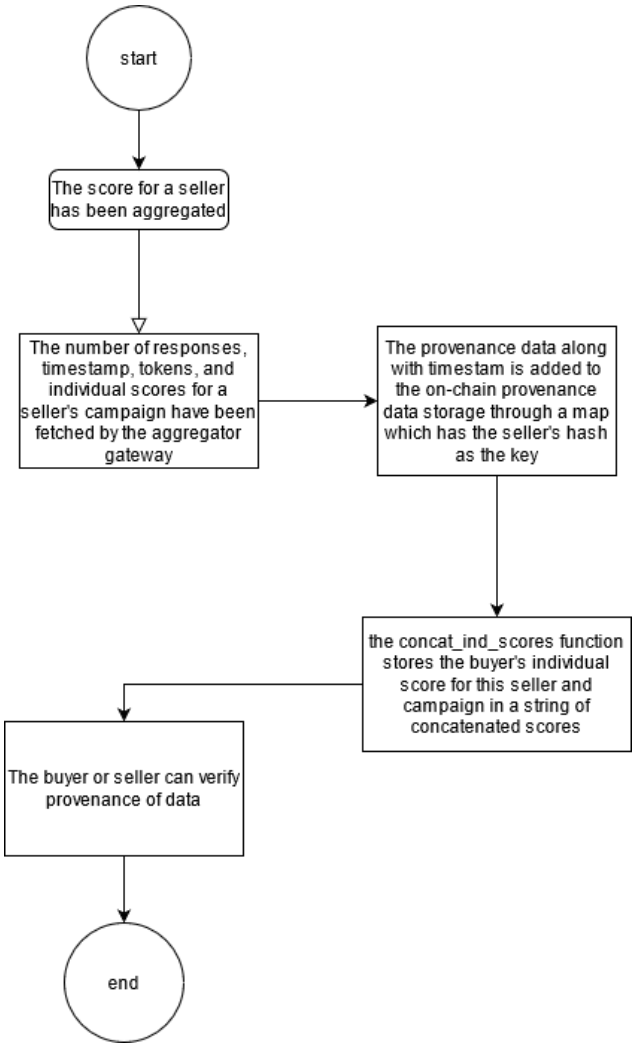

FIGURE 12: FLOW CHART FOR STORING PROVENANCE DATA ON-CHAIN

View rating: Presents the buyer with the reputation score of a seller that they searched for.

Name	**view_rating**
Purpose	To allow a buyer and seller to view the reputation score of a seller
Prerequisites	• The seller whose reputation score is about to be viewed must have been rated previously • A transaction is created • The transaction is approved and added to the blockchain
Main Narrative	• Buyer or seller searches for a seller • Clicks on view reputation score
Alternative	
Outcome	• Shows the reputation score of the seller

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

Alternative Narrative	Seller has not been rated or no reputation score of the seller exists
Outcome	• Reputation score is not shown

Validate transaction: Ensures the legitimacy of a transaction before it is put into the blockchain.

Name	**validate_transaction**
Purpose	To make sure that a transaction is legitimate
Prerequisites	• A buyer rates a seller • A transaction is created • A transaction is about to be added to a block
Main Narrative	• A Miner takes the transaction and ensures its legitimacy. • Transaction is legitimate
Alternative	
Outcome	• The transaction is put into a block
Alternative Narrative	Transaction is not legitimate
Outcome	• The transaction is not put into the blockchain

Broadcast transaction: When a transaction is validated, it will then be broadcast onto the blockchain.

Name	**broadcast_transaction**
Purpose	Broadcasts the transaction onto the blockchain
Prerequisites	• A buyer rates a seller • A block for the transaction is created • The transaction is verified
Main Narrative	• Nodes receive reward for proof of work • The block is added to the blockchain
Alternative	
Outcome	Transaction is broadcast onto the blockchain
Alternative Narrative	Transaction is not legitimate
Outcome	Transaction is not broadcast onto the blockchain

Retrieve transaction data: Retrieves transaction data from the off-chain data through the smart contract making a request to the off-chain storage

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

provider. Upon reception of the request from the smart contract, the off-chain storage service provider will respond with the computed result and hence the transaction data is returned.

Name	**retrieve_transaction_data**
Purpose	To retrieve transaction data off-chain
Prerequisites	• Buyer has rated a seller • The aggregator has received scores from all the contacted buyers • The aggregator has computed the reputation score for this campaign • Transaction has been committed to the blockchain • The transaction id, block number, hash and individual scores, supplier id and campaign id are stored on the off-chain storage
Main Narrative	• The smart contract requests transaction data stored on the off-chain storage • The off-chain storage service provider picks up the request • The service provider responds with computed result
Alternative	
Outcome	Transaction data is returned
Alternative Narrative	-
Outcome	-

Add score off-chain: Adds the individual scores of sellers along with transaction data (transaction id, block number, hash of aggregated score, etc.) on the off-chain storage.

Name	**add_score_off_chain**
Purpose	Adds the individual reputation score to the off-chain storage
Prerequisites	• The buyer must have rated the seller • The aggregator has received scores from all the contacted buyers • The aggregator has computed the reputation score for this campaign • Transaction has been committed to the blockchain • The aggregator has received the transaction id, block number and hash of aggregated score
Main Narrative	• Aggregator (decentralised oracle service) pushes the individual reputation score along with the transaction id, block number, hash of aggregated reputation score to the off-chain storage • Acknowledgement of the record being added to the off-chain storage
Alternative	
Outcome	• The individual scores are stored on the off-chain storage
Alternative Narrative	The off-chain storage is unavailable
Outcome	Individual scores are not added to the off-chain storage

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

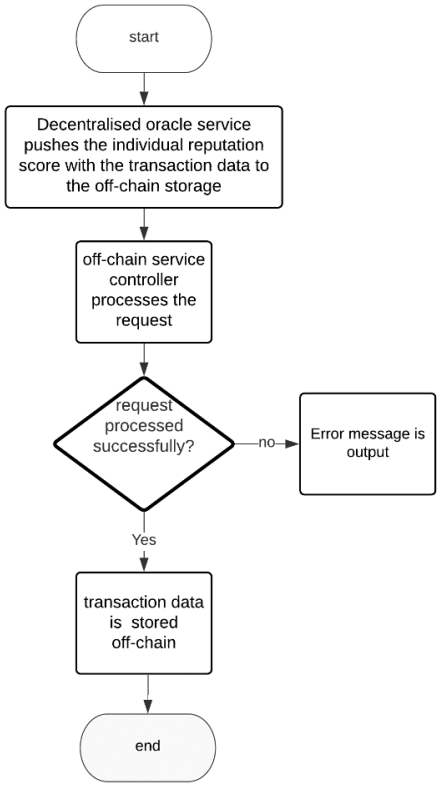

FIGURE 13 FLOWCHART OF OFF-CHAIN DATA STORAGE

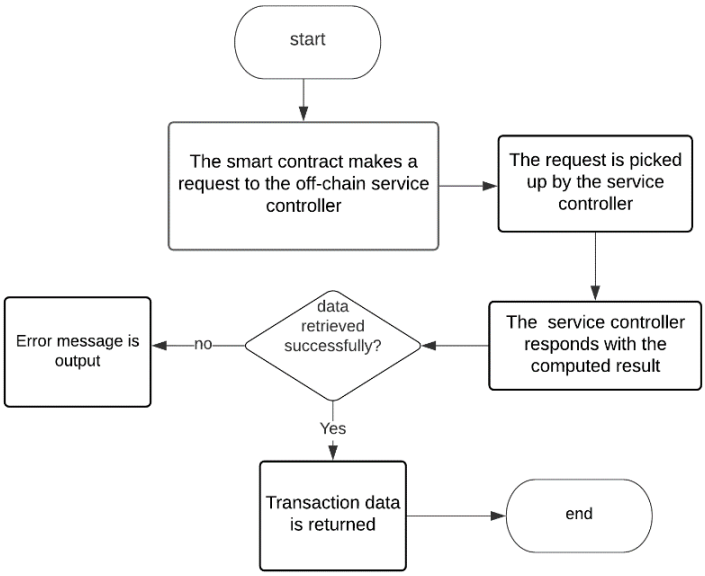

FIGURE 14: FLOWCHART OF OFF-CHAIN DATA RETRIEVAL

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

Query interface for reputation scores: In this use case the user/consumer can send a query submit a query through the dashboard to get information about the trustworthiness of the service providers, for instance, before initiating any transaction with the service provider.

Name	**User requests reputation information**
Purpose	To allow user to get information about the trustworthiness of the service providers
Prerequisites	• The user makes a query through the dashboard before making any transaction. • The service providers scores and information are already on the blockchain. • The reputation data (individual user feedback) are stored on the off-chain
Main Narrative	• The user submits a query via a web-based front end utilising predefined queries • The web application formulates the backend query and uses a REST API to connect to the off-chain storage • The query is executed at the off-chain storage and results are returned to the web-based application • The results are displayed to the user through the function available at the web application.
Alternative	/
Outcome	• The information and score are presented to the user on the dashboard.
Alternative Narrative	• There is no data about the service provider. • Reputation still being updated version given. • The process is not completed.
Outcome	• The query returns no data, and the user notified.

Programmable interface for reputation system/scores: In this use case, the API is an important component to complete the interaction between the front-end and off chain. The iExec SDK is a library that is used to setup the interactions with Decentralised Marketplace to run computations off-chain.

Name	**Programmable interface for reputation system/scores**
Purpose	To enable seamless integration/consumption of reputation data with other applications
Prerequisites	• The reputation data is stored and accessible at an off-chain storage • The reputation data is queryable through programmable interfaces
Main Narrative	• An API is made available which offers different methods for querying reputation data • Using iexec SDK to initiate the connection between the marketplace and off-chain computations.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

	• Initiate the connection between the front-end and off-chain.
Alternative	-
Outcome	API for the reputation system/scores
Alternative Narrative	-
Outcome	-

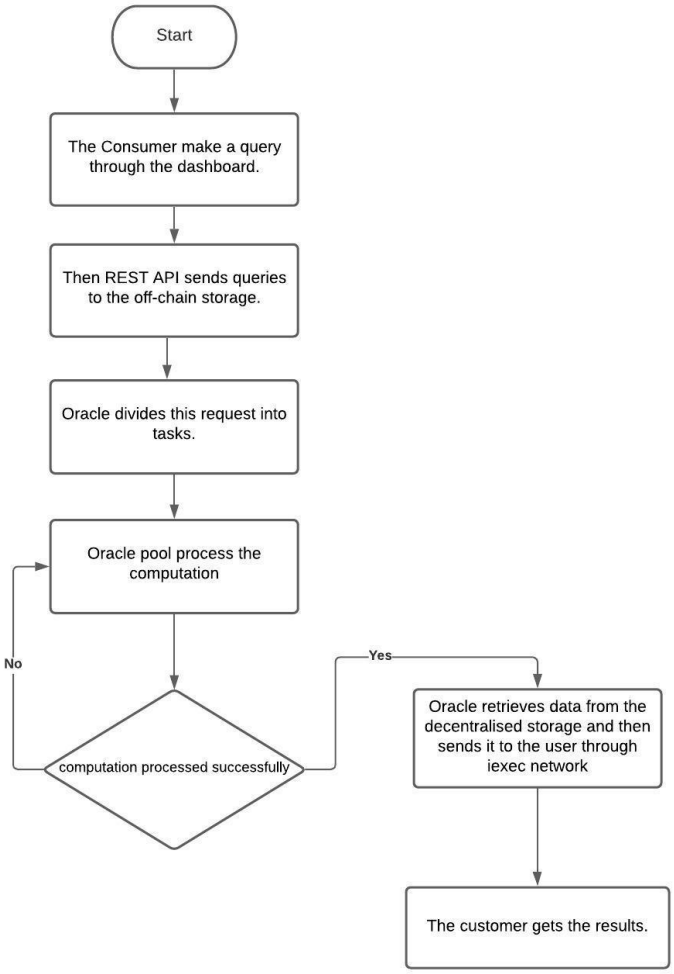

FIGURE 15: FLOW CHART FOR PROGRAMMABLE ACCESS TO REPUTATION DATA

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

3.5 DETAILED DESIGN OF SOFTWARE COMPONENTS

In this section, we present detailed pseudocodes for the primary software components of the REPUTABLE system. This will act as a blue print for the prototype development to be conducted during the rest of the project.

Pseudocodes for the software components

```
BEGIN
        initialise campaignId variable
        initialise token variable
        initialise user_rating variable
        initialise user_addr variable
        initialise Aggregator aggr instance with defined tokens

        IF   user token is valid
             generate_keys()
             generate_cryptograms()
             user_rating = user input
             #encrypt user rating
             submit_feedback()
             rating_submitted_event() #returns value that lets the web interface know to show a message to
let the user know that the
             rating has ben successfully submitted.
        ENDIF
END
```

FIGURE 16: PSEUDOCODE FOR THE WEB INTERFACE COMPONENT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

```
BEGIN
        initialise ind_scores mapping
        initialise seller_tokens mapping
        initialise campaignId variable
        initialise n_responses variable
        ind_score = new struct{
                user_score = user submitted value
                seller_addr = seller address
                campaignId = id
                }
         ind_scores[sellerHash] = ind_score
        store_ind_score(seller1, ind_score)
        IF   get_responseCount(seller1, campaign=2) equals n_responses
        retrieve_ind_scores(seller1, campaign=2)
        aggregate(seller1, campaign=2)

END
```

FIGURE 17: PSEUDOCODE FOR THE AGGREGATOR COMPONENT

```
BEGIN
        initialise scores mapping
        initialise token_val variable
        initialise n_responses variable
        fetch_aggr_score(seller1)
        add_rep_score()
        rep_score_added_event()

END
```

FIGURE 18: PSEUDOCODE FOR ON-CHAIN STORAGE SMART CONTRACT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

```
BEGIN
        initialise scores mapping
        initialise ind_scores_string mapping
        IF rep_score_added_event(seller1, campaign=2)

            fetch_ind_scores(sellller1)
            add_prov_data()
            concat_ind_scores(seller1, scores)
        ENDIF

END
```

FIGURE 19: PSEUDOCODE FOR THE DATA PROVENANCE STORAGE SMART CONTRACT

```
BEGIN
        initialise struct user_scores (user_score, seller_addr, campaignID)
        initialise struct user_token(user_token, used, exp_responses)
        initialise seller_tokens:mapping(seller_addrs=>tokens), ind_scores:mapping(user_add=>ind_score)

        For seller_i in seller_list
        fetch aggregate_reputation_score (seller_addr)
        End For

        For user_score_i in user_scores(seller_addr==seller_j)
        append (user_score,user_score_collated)
        End For

        store_ind_score_on_chain(seller_addr, user_score_collated, Campaign_k)

        accept tx_hash

END
```

FIGURE 20: PSEUDOCODE FOR THE GATEWAY SMART CONTRACT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

```
BEGIN
        initialise struct user_scores (user_score, seller_addr, campaignID)
        initialise ind_scores:mapping(user_add=>ind_score)
        initialise tx_hash_ind_scores

        fetch tx_hash_ind_scores(seller_i, campaignID)
        accept tx_hash

        For user_score_i in user_scores(seller_addr==seller_i && campaignID==campaignID_j)
        store_ind_score_off_chain (seller_addr, campaignID_j, tx_hash, user_score)
        End For


END
```

FIGURE 21: PSEUDOCODE FOR THE OFF-CHAIN STORAGE CONTRACT

```
BEGIN
        initialise struct user_scores (user_score, seller_addr, campaignID)
        initialise struct seller_aggr_score(tx_hash_aggr_score, timestamp, campaignID)
        initialise ind_scores:mapping(user_add=>ind_score)
        initialise tx_hash_aggr_score

        fetch tx_hash_aggr_scores(seller_i, campaignID)
        accept tx_hash


        calculate_aggr_rep_score(seller_addr, ind_scores, aggr_func, campaignID)
        accept seller_aggr_score

        verify_aggr_rep_score(ind_scores, aggr_func, seller_addr, campaignID)
        accept verify_outcome
END
```

FIGURE 22: PSEUDOCODE FOR THE DASHBOARD COMPONENT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

4 EVALUATION METHODOLOGY

With respect to evaluation of the REPUTABLE system, we envisage achieving a subset of the following criteria.

- Computation complexity analysis: We envisage presenting the computation costs in terms of time require for creating the cryptograms (the encrypted feedback and the non-interactive zero-knowledge proof) at the client side, and the time require for aggregating the cryptograms from the dash board

- Communication complexity analysis: The communication overhead depends on the size of data sent by the feedback provider to the bulletin board. The most expensive data unit in the protocol design is the NIZK proof of well-formedness, which consumes most of the communication bandwidth. This analysis will highlight communication efficiency and identify specific overheads.

- Time analysis for aggregated score: As the validity (and in some use-cases real-time computation of aggregated reputation score) is crucial to trustworthiness of the REPUTABLE. In this respect, we envisage evaluating the time required by the aggregator module to calculate aggregate reputation score for a seller.

- Time analysis for time-window (periodic) mode: Through continuous discussions within the team and the ONTOCHAIN coaches, we have explored the option of a time-window mode for feedback aggregation. This can introduce flexibility of operation and ability to model different strategies with respect to how valid the reputation score can be. However, identifying an optimum time-window is a challenge and therefore we aim to explore evaluating with different time-windows to help this decision.

- Scalability: The scalability analysis can be two-fold i.e. with respect to the number of users that can be accommodated by the REPUTABLE system for their feedbacks, and the number of blockchain transactions that can be achieved with respect to storage of reputation data. In this respect, we wish to develop a diverse evaluation strategy which can assess different aspects of the REPUTABLE system with different parameters.

- Cost: An important factor we have considered is the cost of adding data to blockchain with respect to Ethers. This is important aspect as it can influence the frequency with which aggregate reputation score can be updated. We would therefore aim to explore cost evaluation of REPUTABLE's smart contracts to understand the optimal setup.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

- Time required for verification of reputation score (query time): The REPUTABLE dashboard is aimed at facilitate querying reputation scores for specific sellers by end users and other services. Therefore, the response time for such requests is important to adoption and usability of REPUTABLE service. In this context, we aim to evaluate different scenarios to identify and analyse the performance of dashboard interfaces.

Key Performance Indicators:

With respect to the demonstrator for the REPUTABLE system, we aim to fulfil a subset of the following KPIs.

- Anonymity of feedback
- Completeness of aggregation (all feedbacks are included)
- Well-formedness of the feedback (so that feedback is not out of range)
- Immutable storage of aggregate reputation score (achieved through blockchain)
- Verifiability of reputation score (through dashboard by users)
- Query of reputation scores for a seller (through dashboard)
- Link between on and off-chain (through oracles)
- Provenance preservation for the user feedback (achieved through smart contracts and oracles)

5 DETAILED DEMONSTRATION DESIGN AND WORKFLOW

In order to assess the effectiveness of REPUTABLE system, we plan to conduct a thorough evaluation of the system which will focus on performance overhead (memory, bandwidth, storage, processing), complexity of reputation system, scalability, accuracy of the aggregation algorithm, and response time for the reputation system.

Demonstration/Experimentation use case

To aid evaluation, we plan to use the use case of a typical e-commerce transaction. Typically, the consumer transacts directly with the marketplace or the retailer (for example buying products from the retailer over the Amazon and eBay networks, or buying products from the independent

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

online store). Once the product is received by the consumer, he is then asked for the feedback about his recent transaction. The consumer provides his feedback to the bulletin board in an encrypted form. The bulletin board acts as a platform for a public authenticated channel, where authenticated consumers can post data, say with a digital signature to prove the data authenticity. In particular, the bulletin board stores the identity of the consumer providing rating, tokens issued by the marketplace (to ensure it is a legitimate transaction), encrypted feedback scores, and the associated zero-knowledge proof to prove the well-formedness of feedback ciphertext. Once the information is published on the BB, the marketplace could use this information to compute the aggregated reputation of the retailer or seller. The marketplace can then put this aggregated reputation score on the web page of the retailer in order to provide information about the trustworthiness of the retailer. Further, a new or old user can also verify the stated reputation score by accessing the information from the bulletin board in a secure and private way.

Demonstration setup

In order to conduct the evaluation, we plan to implement the REPUTABLE system on a testbed hosted at Birmingham City University. In this respect, we have access to dedicated high performance resources and a blockchain testbed at the university. We wish to utilise these resources to develop the REPUTABLE solution. Further, we plan to host the prototype REPUTABLE solution on a public website to facilitate evaluation through user engagement. In this regard, we plan to use students, academics at Birmingham City University and Derby University as well as the general public to attract a considerable volume of users. In doing so, we will seek guidance from Ethics procedures adopted at the partner universities and follow them when engaging with the users. Further, we will engage with the selected projects of the ONTOCHAIN OC1 to engage diverse user-base whilst also exploring opportunities for collaboration with respect to the application.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

6 SOFTWARE AS PART OF THE ONTOCHAIN ECOSYSTEM

Interoperability is one of the critical goals for the REPUTABLE system. In this respect, we envisage exposing REPUTABLE functionality to end users, other components of the ONTOCHAIN ecosystem, and external services which may be interested in querying reputation data.

Specifically, we aim to have three different interfaces for REPUTABLE:

- Query reputation score for a seller
- Query historical reputation score for a seller
- Verify reputation calculation for a user

We have developed a draft API docs for the above interfaces which are included below. We aim to enhance these in the remainder of the project.

```yaml
openapi: 3.0.0
info:
  title: The REPUTABLE API
  description: A simple description of REPUTABLE API
  contact:
    name: REPUTABLE Team
    url: reputable.io
    email: contact@reputable.io
  version: '1.0'
paths:
  /get_reputation:
    description: access reputable api
    get:
      description: query reputation of a service/seller
      parameters:
        - in: query
          name: seller_id
          required: true
          schema:
            type: string
            example: seller_1
        - in: query
          name: campaign
          description: provide campaignID or empty to get the latest reputation score
          required: false
          schema:
            type: string
            format: integer
      responses:
        '200':
          description: Success response
          content:
            application/json:
              schema:
                type: number
                description: the reputation of the seller for the specified campaign
```

FIGURE 23: DESCRIPTION OF QUERY_REPUTATION API

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

```yaml
/verify_reputation:
  summary: to verify reputation of a service/seller
  description: to verify reputation of a service/seller
  get:
    description: to verify reputation
    parameters:
      - in: query
        name: seller_id
        required: true
        schema:
          type: string
          example: seller_1
      - in: query
        name: user_token
        required: true
        schema:
          type: string
          description: user token issued by the seller/marketplace
    responses:
      '200':
        description: Success response
        content:
          application/json:
            schema:
              type: string
              description: receipt highlighting on-chain record/hash of user_feedback
```

FIGURE 24: DESCRIPTION OF VERIFY_REPUTATION API

```yaml
/get_historical_reputation:
  summary: get historical reputation data of a seller
  get:
    description: query historical reputation of a service/seller
    parameters:
      - in: query
        name: seller_id
        required: true
        schema:
          type: string
          example: seller_1
    responses:
      '200':
        description: Success response
        content:
          application/json:
            schema:
              type: object
              description: the historical reputation of the seller
```

FIGURE 25: DESCRIPTION OF QUERY_HISTORICAL_REPUTATION API

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

7 CONCLUSIONS

The ONTOCHAIN initiative emphasizes trustworthy information exchange which requires a reliable and privacy-preserving reputation system. Through the state of the art analysis conducted as part of D1 of this project, we identified the fundamental requirements for an effective reputation system for ONTOCHAIN. These require a privacy-preserving, decentralised and verifiable reputation system which is able to preserve provenance of reputation information and the reputation scores deduced from this information. Further, the reputation data should be queryable to facilitate user verification as well as seamless integration with systems which may envisage leveraging such information to deliver trustworthy services.

REPUTABLE addresses these requirements and has the potential to deliver a cross-platform privacy-aware reputation system which leverages blockchain technology to achieve decentralised, verifiable calculation of reputation scores. Further it enables interaction with end users and systems through a secure, reputation analytics dashboard to facilitate user verification as seamless integration with other systems and services.

The REPUTABLE system is a continuation of the team's existing research which has been published at high quality venues. As part of our existing research, we have conducted formal analysis of fundamental properties of the system such as verifiability, anonymity, and completeness. Further, as part of this project, we have conducted a comprehensive state of the art with respect to decentralised reputation systems leveraging blockchains. Furthermore, we have conducted detailed design specification of the REPUTABLE system including class diagrams, sequence diagrams, use-case diagrams and pseudocode for primary software components of the system.

Leveraging work conducted so far, we plan to conduct the development of the REPUTABLE prototype solution in-line with the design goals highlighted earlier in this document. We envisage making the REPUTABLE service available to other applications and components within the ONTOCHAIN ecosystem to achieve provision of trustworthy marketplace.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

REFERENCES

[1] L. De Alfaro, A. Kulshreshtha, I. Pye, and B. T. Adler, Reputation systems for open collaboration," ACM Communication, vol. 54, no. 8, pp. 81- 87, 2011.

[2] A. Josang, R. Ismail, and C. Boyd, A survey of trust and reputation systems for online service provision," Elsevier Decision Support Systems, vol. 43, no. 2, pp. 618-644, 2007.

[3] F.Hendrikx, K.Bubendorfer, and R.Chard, Reputation systems: A survey and taxonomy," Journal of Parallel and Distributed Computing, vol. 75, pp. 184 - 197, 2015.

[4] S. Bag, M. A. Azad, and F. Hao, A privacy-aware decentralized and personalized reputation system," Computers & Security, vol. 77, pp. 514 - 530, 2018.

[5] T. Minkus and K. W. Ross, I Know What You're Buying: Privacy Breaches on eBay. Cham: Springer International Publishing, 2014, pp. 164 - 183. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-08506-7 9

[6] P. Resnick and R. Zeckhauser, Trust among strangers in internet transactions: Empirical analysis of eBay's reputation system," in The Economics of the Internet an E-Commerce, ser. Advances in Applied Microeconomics, M. Baye, Ed., 2002, vol. 11.

[7] S. Clau, S. Schiner, and F. Kerschbaum, K-anonymous reputation," in Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security, 2013, pp. 359-368.

[8] E. Zhai, D. I. Wolinsky, R. Chen, E. Syta, C. Teng, and B. Ford, Anonrep: Towards tracking-resistant anonymous reputation," in Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation, ser. NSDI'16. Berkeley, CA, USA: USENIX Association, 2016, pp. 583 - 596.

[9] J. Bl• omer, J. Juhnke, and C. Kolb, Anonymous and publicly linkable reputation systems," in Proceedings of 19th Financial Cryptography and Data Security, R. B• ohme and T. Okamoto, Eds., pp. 478-488.

[10] A. Narayanan and V. Shmatikov, Robust de-anonymization of large sparse datasets," in 2008 IEEE Symposium on Security and Privacy (sp 2008), May 2008, pp. 111{125.

[11] ||, De-anonymizing social networks," in Proceedings of the 2009 30th IEEE Symposium on Security and Privacy, ser. SP '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 173-187. [Online]. Available: http://dx.doi.org/10.1109/SP.2009.22

[12] E. Gudes, N. Gal-Oz, and A. Grubshtein, Methods for computing trust and reputation while preserving privacy," in Proceedings of 23rd Annual IFIP WG 11.3 Working Conference Data and Applications Security, 2009, pp. 291-298.

[13] N. Gal-Oz, E. Gudes, and D. Hendler, A robust and knot-aware trustbased reputation model," in Proceedings of IFIPTM Conferences on Privacy, Trust Management and Security, 2008, pp. 167-182.

[14] O. Hasan, L. Brunie, E. Bertino, and N. Shang, A decentralized privacy preserving reputation protocol for the malicious adversarial model," IEEE Transactions on Information Forensics and Security, vol. 8, no. 6, pp. 949 -962, 2013.

[15] O. Hasan, L. Brunie, and E. Bertino, Preserving privacy of feedback providers in decentralized reputation systems," Computer & Secur., vol. 31, no. 7, Oct. 2012.

[16] L.Liu and M.Munro, Systematic analysis of centralized online reputation systems," Elsevier Decision Support Systems, vol. 52, no. 2, pp. 438 - 449, 2012.

[17] M. Kinateder and S. Pearson, A Privacy-Enhanced Peer-to-Peer Reputation System. Springer Berlin Heidelberg, 2003, pp. 206-215.

[18] J. Bethencourt, E. Shi, and D. Song, Signatures of reputation: Towards trust without identity," in Proceedings of the 14th International Conference on Financial Cryptography and Data Security, ser. FC'10. Berlin,Heidelberg: Springer-Verlag, 2010, pp. 400-407.

[19] S. Schiner, S. Clau, and S. Steinbrecher, Privacy and Liveliness for Reputation Systems. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 209 - 224.

[20] M. A. Azad, S. Bag, and F. Hao, M2m-rep: Reputation of machines in the internet of things," in Proceedings of the 12th International Conference on Availability, Reliability and Security, 2017, pp. 28:1-28:7.

[55] C. Pham, A. Adamopoulos, and E. Tait, Towards a triple bottom line perspective of blockchains in supply chain," in Australasia Conference of Information Systems (ACIS) 2019, 2019.

[56] A. Shwetha and C. Prabodh, Auction system in food supply chain management using blockchain," in Proceedings of International Conference on Advances in Computer Engineering and Communication Systems. Springer, 2021, pp. 31-40.

[57] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla, Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability," in 2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID). IEEE, 2017, pp. 468-477.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338

[59] V. L. Lemieux, Blockchain and distributed ledgers as trusted recordkeeping systems," in Future Technologies Conference (FTC), vol. 2017, 2017.

[61] A. Ramachandran, D. Kantarcioglu et al., Using blockchain and smart contracts for secure data provenance management," arXiv preprint arXiv:1709.10000, 2017.

[73] L. Page, S. Brin, R. Motwani, and T. Winograd, The pagerank citation ranking: Bringing order to the web." Stanford Info-Lab, Technical Report 1999-66, November 1999. [Online]. Available: http://ilpubs.stanford.edu:8090/422/

[78] Gai, F., Wang, B., Deng, W. and Peng, W., 2018, May. Proof of reputation: A reputation-based consensus protocol for peer-to-peer network. In International Conference on Database Systems for Advanced Applications (pp. 666-681). Springer, Cham.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957338