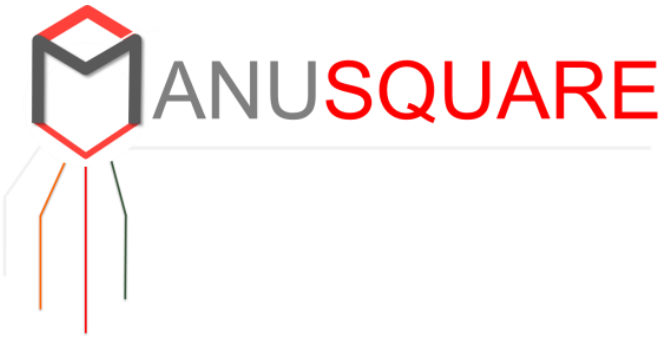
Horizon 2020 – The EU Framework Programme for Research and Innovation
Project Co-funded by the European Commission
Contract number: 761145
Call identifier: NMBP-22-2017
Project Start Date: 1st January 2018



MANUfacturing eco**S**ystem of **QUA**lified **R**esources **E**xchange

D3.5
Blockchain-based supply chain platform – final version

Dissemination Level	Public
Partners	IBM
Authors	IBM
Planned date of delivery	M30 – June 2020
Date of issue	30th June 2020
Document version	Final 1.1

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 761145

DOCUMENT HISTORY

Version	Issue date	Content and changes	Author
0.1	10/06/2020	Initial draft ToC for internal review	IBM
0.2	18/06/2020	Initial full version for internal review	IBM
0.3	21/06/2020	Post first internal review	IBM
0.4	22/06/2020	Post review from Henrique	IBM
0.5	23/06/2020	Post review from Marko	IBM
1.0	24/06/2020	Ready for QA and submission	IBM
1.1	30/06/2020	Quality assurance	SUPSI

Role	Partner	Person
Reviewer 1	INESC TEC	Henrique Diogo Silva
Reviewer 2	INNOVA	Marko Vujasinovic
Quality assurance	SUPSI	Andrea Bettoni

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

Acronym	Description
CA	Certificate Authority
DB	Database
IIoT/ Industrie 4.0	Industrial Internet of Things
IoT	Internet of Things
MANU-SQUARE	MANUfacturing ecoSystem of QUAlified Resource Exchange
REST	Representational State Transfer
RFI	Request for Information
RFP	Request for Proposals
RFQ	Request for Quotation
SDK	Software Development Kit
SME	Small and Medium Enterprises
VM	Virtual Machine
WP	Work Package

GLOSSARY

Term	Meaning
Certificate Authority	An entity that issues digital certificates to be used for authentication of the communicating party holding the certificate.
Chaincode	The implementation of a smart contract within Hyperledger Fabric. A programmatic manner which governs the interaction between a blockchain client and the blockchain network itself.
Endorsing peer	A member of the blockchain network that may be called to endorse transactions for a particular chaincode. The process of endorsing a transaction includes a speculative execution of the chaincode function included in the transaction proposal responding to the caller with the read and write sets (including versions) corresponding to the chaincode execution. In addition it indicates whether it supports the transaction.
MVCC	A multi-version concurrency control employed by Hyperledger Fabric to enable speculative concurrent execution of transactions without corrupting the underlying data storage.
Ordering service	A n Hyperledger Fabric entity that provides total order among incoming transactions and their inclusion into blocks.
Peer	A central entity in the Fabric network which is in charge of validating incoming blocks (and associated transactions) and committing the blocks to its own copy of the shared ledger.
REST	A common manner of interaction among different processes. (https://en.wikipedia.org/wiki/Representational_state_transfer)

1 EXECUTIVE SUMMARY

The main goal of this document is to present the blockchain based infrastructure underlying the trust aspects of the MANU-SQUARE platform. The document starts from a short description of the underlying blockchain technology and its specific relevance and usage within the MANU-SQUARE platform, briefly presenting 2 supply chain related constructs that were first introduced in D3.3 (Blockchain-based supply chain platform – first version), namely the components supporting the Request for Quotation (RFQ) process, and the reputation management process. The last of the blockchain supported constructs envisioned within the MANU-SQUARE platform, support for collaborative idea management (a collaborative tool to foster the join creation of innovative solutions), shall be presented in depth in this document.

The underlying blockchain technology used in MANU-SQUARE is Hyperledger Fabric (https://www.hyperledger.org/projects/fabric), a permissioned open source blockchain project maintained by the Linux Foundation.

A blockchain based infrastructure provides many benefits for the MANU-SQUARE platform. The chief among these are the establishment of trust and clarity into shared processes, in an otherwise thrustless and fragmented environment.

This document should be perceived as a continuation of D3.3 (Blockchain-based supply chain platform – first version), providing more details on the third supported MANU-SQUARE process, and the overall deployment of the blockchain infrastructure including the supporting smart contracts and applications.

This document further relies on developments and architectural design presented in D3.1 (Connecting IoT devices to blockchain services), and D3.2 (Security and privacy services). In order for this document to be self-contained, a concise summary of the project and its objectives is presented in § 2.1, while aspects of the blockchain technology and its application within the MANU-SQUARE project are presented in § 2.2. For completeness, being the final deliverable of Task 3.3, some elements concerning support for RFQ and reputation management that were introduced in D3.3, were added into this deliverable as well.

As the final report of the activities carried out within Task 3.3, this document details the technology used, deployment aspects, and details on all developed capabilities. As planned, in the last stage of this task the emphasis was on providing support to the idea management tool, thus this capability is placed at the heart of the current document. This deliverable documents the last stage of the design, development, and deployment of the blockchain support within MANU-SQUARE. This stage comprised the last 6 months of the task duration, and as such relies mostly on all the development that has been carried out during the first two years of the project evolution.

The description of the work is organized into the following sections:

- § 2 briefly introduces the main concepts behind the MANU-SQUARE platform, the blockchain essentials, and the intersection between the two within a supply chain environment.
- § 3 dives into the uses of Blockchain within MANU-SQUARE; the architecture, capabilities, and manifestation, placing emphasis on constructs to support the collaborative ideas development tool.
- § 4 details the blockchain based underlying support specifically in the project context.
- § 5 provides more details on the currently implemented and deployed blockchain based supply chain constructs.
- Finally, § 6 provides conclusions.

2 INTRODUCTION

2.1 MANU-SQUARE in a nutshell (with a blockchain twist)

The MANU-SQUARE project aims at fostering an ecosystem that acts as a virtual marketplace in which surplus of industrial resources can easily meet corresponding shortage, thus bringing the available capacity (such as production capacity), as well as other virtual and physical assets, closer to the demand to obtain the optimal match (See Figure 1). This scheme has two main advantages:

- The rapid and efficient creation of local distributed value networks for innovative providers of product services;
- The reintroduction and optimization in the loop of unused capacity and potential that would otherwise be lost.

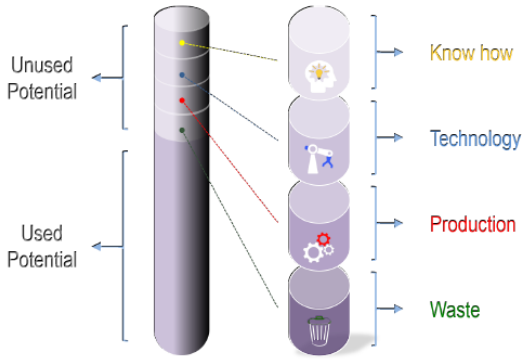


Figure 1: Composition of the unused potential

MANU-SQUARE establishes an ecosystem that is organized to match the needs of buyers with the availability of sellers in terms of know-how, technology, manufacturing capacity, and waste (or by-product). The blockchain technology provides trust, transparency, and security to the MANU-SQUARE platform, thereby serving as a single source of truth and distributed trust in an otherwise trust less environment amongst the different stakeholders of the platform.

Throughout the document we mainly refer to the permissioned flavour of a blockchain which is more suitable for business scenarios. The focus is mainly on the Hyperledger Fabric implementation (https://www.hyperledger.org/projects/fabric). A permissioned network ensures that members agree to enter a joint network and it can control the identity of participants in the network. This stands in contrast to public networks (such as bitcoin) in which anyone can join in any role, and the identities of participants are concealed. Nevertheless, not all information on the blockchain is visible to all consortium participants, but rather there is a built-in flexibility supporting differential visibility constructs.

The main objective of the project is to match shortage with surplus on a wide spectrum of areas. The main reason for integrating blockchain technology at the basis of the platform is to provide security, privacy, and trust in the process. In a multi-sided platform, such as MANU-SQUARE aspires to be, there are many entities with different kinds of relationships between them. That leads to different levels of privacy and data visibility requirements that should be supported by the platform, depending on the entities involved and the current interaction between these entities. As there are various modes of interaction between entities, there are various visibility scopes, to adhere to the required level of privacy and isolation.

The last part of Task 3.3, which is summarized in this deliverable, describes the final architecture and capabilities provided by the blockchain platform in the context of the project. This was exemplified in D3.3 by the RFQ and the reputation mechanism support, and further described in this deliverable focusing on the support for the idea management tool for collaborative authoring. In the context of the collaborative ideas development the blockchain based support ensures the maintenance of full traceability and non-repudiation of incoming data and order, such that it can be made clear which entity is the origin of each contribution to the overall idea.

2.2 Blockchain – in a nutshell

A blockchain revolves around the concept of a shared ledger, representing the system of record and a single source of truth for business interactions. The shared ledger is maintained by a cluster of peer processes, potentially belonging to different organizations, providing an append only transactions log, while guaranteeing the immutability and finality of inserted and validated transactions. It enables a network of business partners to perform transactions across organizations without resorting to a single unified trusted authority. A blockchain transaction represents a state change or asset transfer in the ledger. Transactions are governed by smart contracts, which contain the rules for transactions to be invoked and the agreed upon resulting behaviour.

A Blockchain provides a shared, replicated, permissioned ledger ensuring provenance, immutability and finality, thus establishing the cornerstone of an elevated level of trust to replace inefficient, expensive, and vulnerable processes. Provenance provides the capability to trace the source and all the subsequent changes that were applied on an entity. Immutability refers to the property that ensures that a transaction that was recorded in the blockchain cannot be altered in any way (or at least without parties being able to identify an altered piece of information). Finally, the finality property ensures that once a transaction was recorded in the blockchain it cannot be removed.

These properties together provide a level of trust among partners which is difficult to achieve otherwise in an inherently trust-less and distributed environment. Most importantly, the trust is not due to a single actor within the network, but rather it is an outcome of the collective nature and properties of the underlying technology. Transactions through the platform are recorded in a final and immutable manner by the blockchain, providing all network members with an identical and trustworthy real-time view of the state. Validated transactions in a block in a ledger cannot be modified or deleted without leaving a noticeable trail.

As can be seen in Figure 2 the shared ledger provides a real-time common and replicated view of the state of the transactions among all members of a blockchain network. This reality stands in contrast to the pre-blockchain era in which each organization held its own ledger, opening the door to inconsistencies and disputes.

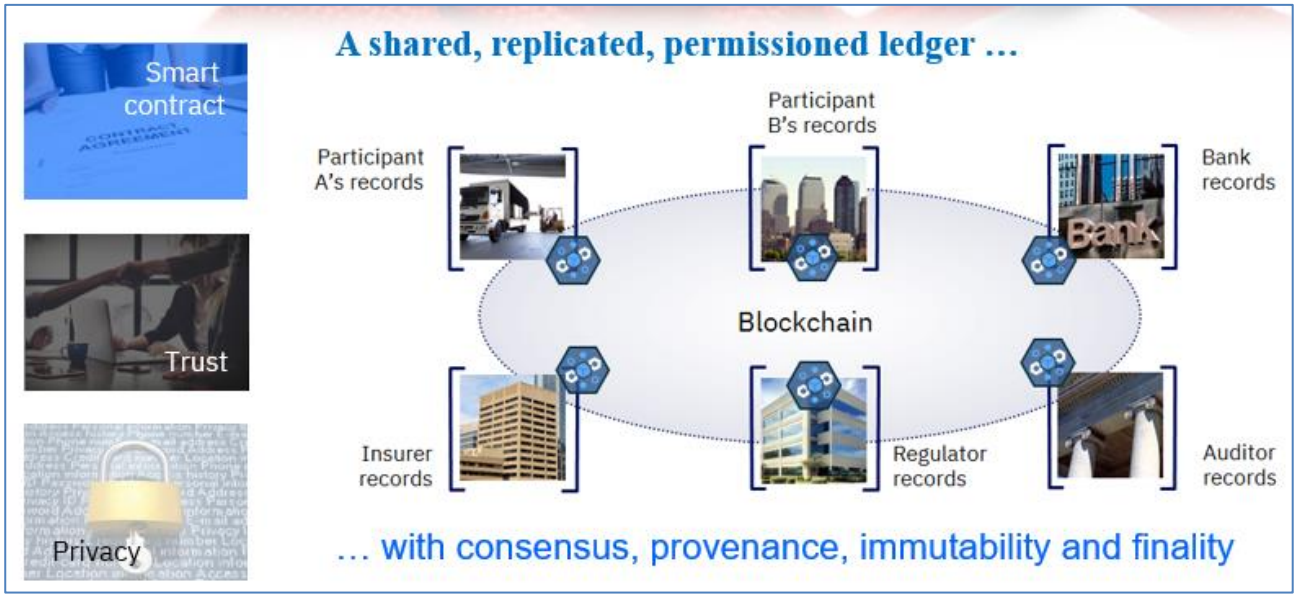


Figure 2: Blockchain core - the shared ledger

The four cornerstones comprising the blockchain structure are a shared ledger, transaction verification by network members (consensus), smart contacts, and security & privacy measures. All these building blocks combined together provide assurance for consensus, provenance, immutability, and finality. These capabilities lay the foundation for a blockchain platform for enterprises as can be seen in Figure 3. There exists at the center a shared ledger which replicates the information that was destined to be shared among all the identified collaborating partners. As a permissioned ledger, the identity of participants is known and each change can verifiably be associated with a certain identity. The different

partners in question can determine which transaction is eligible to be recorded in the blockchain. Finally, different levels of privacy can ensure that information is exposed only to the intended entities.

At a high level, the system is comprised of peer servers, potentially belonging to different organizations, which replicate and validate the blocks engulfing the transactions comprising the ledger; an ordering service which determines the total order of the transactions and publishes the corresponding blocks to be picked up by the peer processes; and a client that interacts with the system programmatically for invoking transactions or queries. A configurable sub-set of the peers is involved also in endorsing transactions submitted to the system; supporting consensus for inserted transactions. All entities hold verifiable security certificates issued by a Certification Authority.

Blockchain technology usage is relatively new but interest in it is growing in many fields. The first such field is the financial services arena, but more areas are exploring the usage of this technology, supply chain being in the forefront. In various analysis reports it can be seen that Banking / Financial Services and Supply Chain remain top industries for blockchain activity[1]. A lot of attention and funds are being devoted to exploring blockchain contribution to supply chain scenarios[2], both by industrial partners, as well as large IT providers, such as IBM, Oracle, and Microsoft.

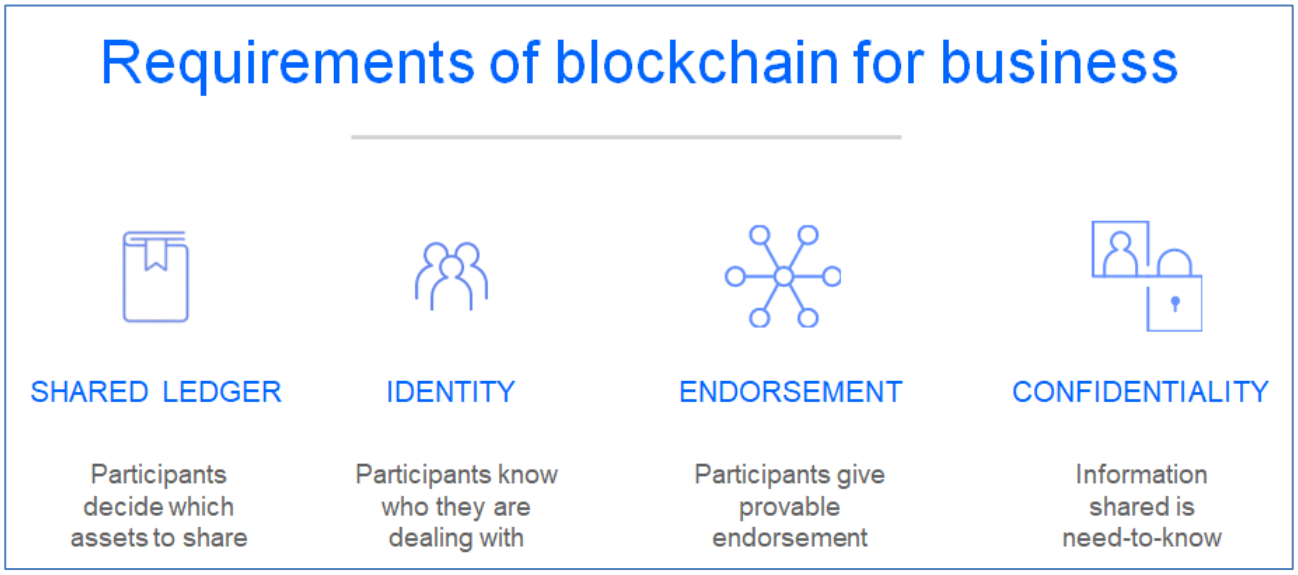


Figure 3: Blockchain essentials

2.2.1 Blockchain for supply chain

Blockchain solutions are prominent in business relationships which require data to be shared, or assets to be transferred, among different entities. Such data may be needed in real-time or close to it, or as a verifiable trace of past transactions to be used in the future. Companies involved do not , however, necessarily have trust in each other. Such relationships are prevalent in supply chain networks. The use of a blockchain based infrastructure enables parties which are a part of a supply chain relationship to leverage the technology to gain tangible benefits in important areas such as reduction in time, money, and risk. The blockchain serves as the single source of truth, which is shared among all participants, and is not controlled by a single entity.

[1] https://www2.deloitte.com/content/dam/Deloitte/cz/Documents/financial-services/cz-2018-deloitte-global-blockchain-survey.pdf
[2] https://newsroom.ibm.com/2018-08-09-Maersk-and-IBM-Introduce-TradeLens-Blockchain-Shipping-Solution
https://www.coindesk.com/pwc-australia-port-of-brisbane-unveil-blockchain-supply-chain-pilot
https://www.zdnet.com/article/alibaba-pilots-blockchain-supply-chain-initiative-down-under/
https://www.forbes.com/sites/bernardmarr/2018/03/23/how-blockchain-will-transform-the-supply-chain-and-logistics-industry/#748fab1e5fec

Some prominent examples of the global use of blockchain in large and elaborated supply chain scenarios include TradLens[3] and IBM Blockchain transparent Supply (IBM BTS), former Food Trust[4]. TradeLens is about simplifying and unifying container logistics worldwide, with an emphasis on the shipping industry. IBM BTS established a provenance audit trail of the food purchased and subsequently eaten; providing information such as the origin of the item, day it was picked (or produced), and its journey until the point of sale.

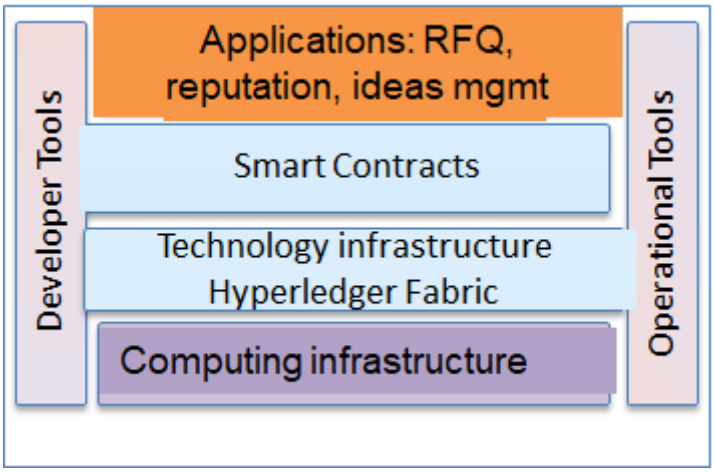


Figure 4: Technology Layers

In Figure 4, a high-level view of the technology layers involved is depicted.

At the bottom resides the computing infrastructure itself along with supporting services. This layer usually resides on a cloud infrastructure, but may be on a standalone VM or server as well. Above the computing infrastructure resides the blockchain infrastructure itself, which consists for example of consensus mechanism, cryptographic validation, mechanisms for replication of blocks, and certificate authorities (CA). At a higher layer there resides the elements which enable developers to insert specific logic which shall be tightly coupled to a specific deployment of a blockchain. This layer encompasses the specific rules governing the interactions supported for a specific network of participants. This layer is mostly associated with and implemented by Smart Contracts (chaincodes in Fabric). At a higher layer resides the solutions or applications, which serve as a mean to connect the underlying blockchain with the business processes and systems of companies, be it via interaction with end-users, or digital processes and devices operating on their behalf.

As a part of the vision to incorporate a blockchain based supply chain platform, the first step is to incorporate partners' data of different kinds, be it machine generated data or a part of a business process involving a human in the loop. The data serves as the driving element to the agreed upon logic which resides at the blockchain level as well in the form of smart contracts. Examples as to the kinds of data include tracking the contribution to a forming idea, in the MANU-SQUARE case this capability supports the Idea Management tool.

In the collaborative Idea Management tool, these capabilities enable a coherent and updated view of the status of an idea in formation, including an account of who is responsible for each entry in the overall forming idea.

The blockchain infrastructure can be used to reduce the rate of disputes and errors in logistics and to enable real-time tracking of transactions in the supply chain providing elevated accuracy, security and speed; while ensuring that data and interactions are not made visible to unauthorized partners. Moreover, full traceability and provenance of business processes execution is supported by the blockchain infrastructure. This is exemplified in the collaborative ideas generation too, by making the division of contributions attributed to specific users very clear. This underlying capability contributes to enhancing trust in an otherwise trust less environment such as in collaborative idea generation in which individuals working towards the same goal do not necessarily know each other. Trust is established by providing end-to-end provenance and

[3] https://www.tradelens.com/
[4] https://www.ibm.com/blockchain/solutions/food-trust

a full audit trail of the exact evolution of the developed ideas towards completing a challenge. Having a single source of truth, verifiable and auditable, can lead to a reduced number of disputes, and a shorter time to resolution of existing disputes. Finally, differential visibility and data privacy ensure that the information is shared only among the intended partners, thus ensuring the adequate flow of information.

3 BLOCKCHAIN USE IN THE MANU-SQUARE PROJECT

3.1 Blockchain roles in the overall platform architecture

The blockchain platform plays a role both in the underlying data layer as well as in the tools layer. As can be seen in Figure 5 and Figure 6 it is located at the lower infrastructure layer of the platform, exposing interfaces to services that the different higher-level components of the platform can use.

At the data layer it does serve as a unique kind of data store in the form of a shared ledger exhibiting the capabilities detailed in § 2.2.1. At the same time, it does play a role at the tools layer as well, since a part of the logic does reside in the blockchain internals in the form of a smart contract[5]. . A smart contract is a programmatic manner to declare and enforce the rules that govern specific interactions via the blockchain. Thus, different platform components use interfaces exposed by the blockchain component in order to take advantage of the capabilities and promises of a shared ledger. The main component that interacts with the blockchain layer on behalf of other components is the Ecosystem Data Manager (EDM). The blockchain component exposes REST interfaces that can be used by the EDM and all other platform components. The interfaces are grouped according to functionality provided. At the current deliverable; we mainly focus on the blockchain support for the idea management; in a previous deliverable we elaborated on the interfaces with RFQ and reputation related interfaces. There are two main kinds of actions supported by the interfaces:

1. **Invoke smart contact transactions** – intended for MANU-SQUARE modules to invoke transactions residing in smart contracts, and record the resulting state in the blockchain. These transactions are invoked to change the state of an entity and record that for posterity. For example, while supporting the idea management tool transactions can be invoked for publishing a new call for idea generation.
2. **Query** - expose query capabilities to retrieve data stored in the blockchain. For example, while supporting the idea management process such queries can be used for retrieving information on the complete lifecycle of an idea.

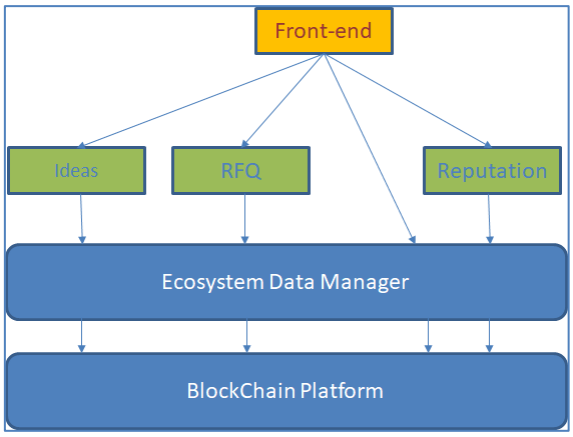


Figure 5: Components interactions

In Figure 5 a broad sketch of the interaction of the Blockchain component within MANU-SQUARE is depicted. In the diagram there is a blockchain platform, or blockchain network, depicted at the bottom. The blockchain layer includes an internal blockchain client embedding an application that interacts directly with the blockchain network while exposing a REST interface to the rest of the MANU-SQUARE components. The Ecosystem Data Manager acts as a mediator between the rest of the platform components and the data layer, including the blockchain.

[5] In Hyperledger Fabric smart contracts and implemented in the form of chaincode (https://hyperledger-fabric.readthedocs.io/en/release-1.4/chaincode.html)

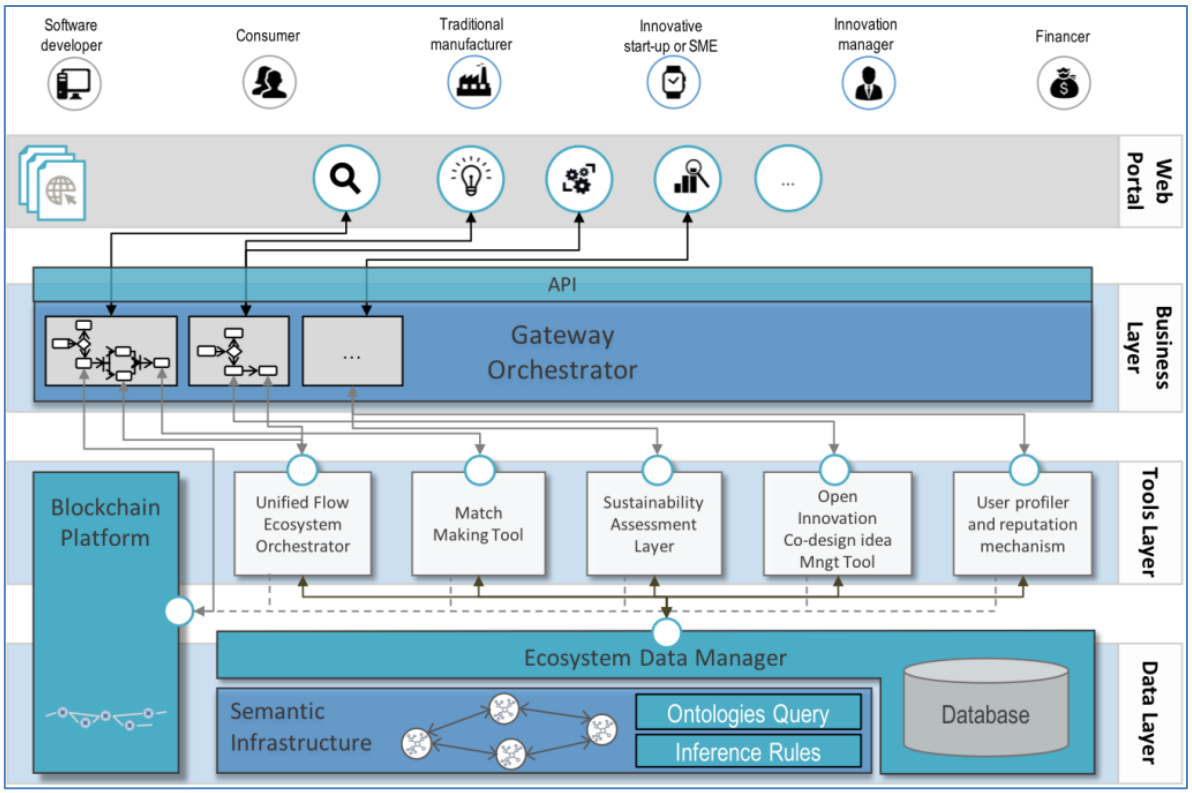


Figure 6: MANU-SQUARE high level architecture

Figure 6 provides a higher-level architecture view including the location of the Blockchain component within the platform, and the interactions with additional components and tools. The Blockchain component sits at the bottom part of the architecture, along with additional data processing and storage components, but also has a part in the tools layer, as the smart contracts (chaincodes) that are deployed in the system hold the logic which governs the interaction with the underlying blockchain storage.

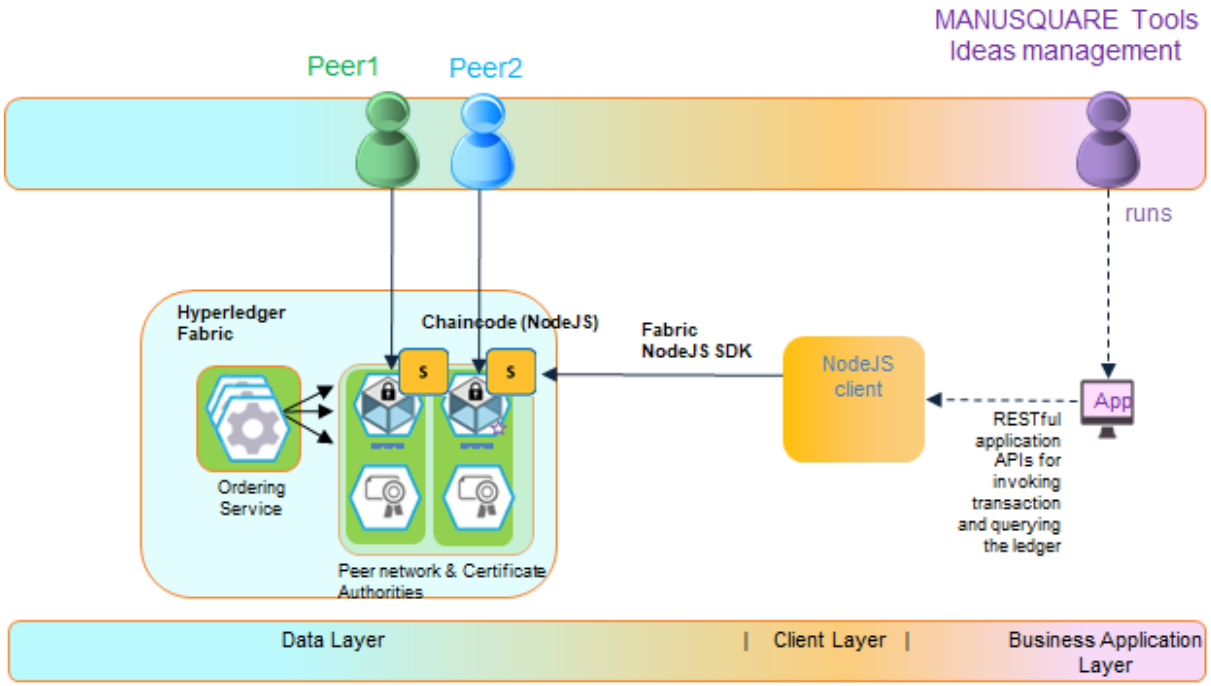


Figure 7: Embodiment within MANU-SQUARE

Furthermore, as it can be seen in Figure 7, the blockchain support architecture for the different MANU-SQUARE tools consists of several components. First, the blockchain network itself which currently involves two channels per service supported; one for development and a second one for production. The channel is the construct that hosts the ledger that is shared among the partners declared to eb participants in that channel. The aim is to be able to test and experiment freely with various processes in the development environment, including the use of fake or synthetic information, without that data having an effect on the production environment. The blockchain support for the other processes (RFQ and reputation) is deployed in a similar manner.

The channel hosts the chaincodes which drive the interaction with the component itself. The network is further comprised of peers that obtain and validate blocks of transactions and hold a replica of the shared distributed ledger. In addition, there is a chaincode implementing the blockchain smart contract, storing and updating the state of entities and allowing querying the ledger and the associated world state for information on those entities. The chaincode supporting the idea management tool is written in Golang. In addition, there exists a NodeJS client embedding internally a Hyperledger Fabric NodeJS SDK to invoke/query the appropriate chaincode methods. In addition, a Hyperledger Fabric CLI tool is used as well for interaction between the client application and the blockchain infrastructure. The MANU-SQUARE tools acting as blockchain clients invoke the blockchain client via a set of exposed RESTful APIs, passing JSON objects whose structure is agreed upon between the components, to invoke transactions that change internal state or query for stored information. An ordering service is associated with a channel (several channels can share an ordering service) with the mission to create a total order of incoming transactions, cut blocks, and make the blocks available to the designated peers.

The application acting as a proxy to connect MANU-SQUARE with the blockchain communicates as well with a certificate authority (CA) in order to resolve the cryptographic material and identification of entities.

3.2 Capabilities supported in the MANU-SQUARE platform

This section is intended to introduce the specific use cases and tools that make use of the blockchain technology within the MANU-SQUARE project. As aforementioned, the Blockchain integration currently focuses on three representative scenarios, namely RFQ management, reputation management, and traceability of innovative ideas (i.e. idea management).

At a high level, the use cases in MANU-SQUARE can be divided into two broad categories. First, matching between surplus and need (for example of production capacity or by-products) to support capacity sharing. Second, innovation management for collaborative design. For both categories the MANU-SQUARE platform takes advantage of a blockchain based infrastructure as the provider of a trusted (in a trust-less environment) single source of truth. RFQ-related blockchain support can be used in the first family of scenarios, idea management-related blockchain support can be used in the second family of scenarios, and the reputation management- related blockchain support can be used in both families of scenarios.

The functionalities that a blockchain infrastructure can support in these scenarios will be translated into plans for specific use cases to be deployed in the MANU-SQUARE platform. Hereinafter, a short description of each supported scenario is provided. RFQ management support and reputation management support have been detailed in a previous deliverable and are summarized here for completeness.

3.2.1 Traceability of innovative ideas

In this scenario, the blockchain is applied to the tracking of contributions of innovative ideas to challenges within the MANU-SQUARE ecosystem. Considering that the platform is intended to support the evolution of ideas form basic concepts to fully set-up projects in a cooperative and open manner, the blockchain is involved in keeping track of the contributions of each participant and registering the ownership of every single contribution. This capability supports the ability to reward respectively the participants of ideas creation at a later stage in the product development, by a higher level idea management tool that can track the evolution of the idea and participants via queries to the blockchain layer.

The blockchain layer exposes interfaces for adding information and perspectives to an idea along with the possibility to query the blockchain to obtain a validated history of the idea throughout its evolution.

3.2.2 RFQ management

RFQ is a structured and often complex process which may involve multiple hops and interactions between the entities involved (from the initial offer through a negotiation process that can culminate in a signed deal). The blockchain helps to structure the process and safeguard all the interactions and advancements of the process throughout its lifecycle. This capability further helps to centralize the process and related communication, to overcome current scattered RFQ related documentation.

The process is initiated by a prospective customer and is targeted towards a set of potential suppliers and may consist of various items to be agreed upon (such as price and delivery date). Several rounds of negotiation may be required for the positive (or negative) finalization of the process. All information exchanged digitally shall be part of a permanent record kept and made available by the blockchain.

In supporting such a process there needs to be awareness of the data that is distributed on the blockchain platform and the visibility scope associated with that data. Different stages of the process require the visibility to a different subset of stakeholders.

3.2.3 Reputation management

Reputation management is of crucial importance to the adoption of the platform. Thus, the trust that can be associated with this component is of great importance as well. The blockchain infrastructure shall support the traceability of the entire history related to the reputation of all involved entities at different points in time. As the platform is intended mostly to establish relationships among entities that have no prior engagement between them, the reputation management capability plays an important role in the process.

Without the MANU-SQUARE platform, companies often go through much pain over a prolonged amount of time before embarking on a business interaction with a new partner. The intention in this case is that with the added trust that can be associated to this component, companies will rely on reputation scores provided by the platform to make more informed and faster decisions for establishing new relationships. The blockchain layer exposes interfaces for company representatives to be able to enter and store reputation scores and descriptions and to query for the reputation history of an entity including the evolution of the score.

3.3 Contextualising blockchain use in the collaborative ideas scenario

This sub-section clarifies the way the blockchain based infrastructure is used within the innovation management and ideas tracking, which is one of the main business processes enables by the MANU-SQUARE platform. For innovation management and ideas tracking the blockchain infrastructure provides a basis for the management of the entire process from a design need all the way to the agreement on a solution, tracking along the way the contributions of each party to the final products.

Following the business process flow for a collaborative idea generation scenario (more details can be found in *D4.5 Open innovation idea management tool*), we observe the following steps:

1. Customer publishes a new challenge through the idea management tool. The new challenge creation is propagated to the blockchain through the Ecosystem Data Manager component. When publishing a new challenge, the details are added to the blockchain, and a new challenge unique identifier is generated in the blockchain server side and is returned as part of the response to the calling client.
2. At this stage the collaborative ideas creation for this specific challenge is open and can be accessed by additional platform users, be they individuals, companies, or facilitators and experts such as innovation managers.

3. The challenges list can be retrieved, and the details of a specific challenge can be retrieved as well. The list of challenges can include or exclude closed challenges. Moreover it can be filtered by the id of the creator of the challenge.
4. Similarly ideas to be incorporated in open challenges can be added, queried, or updated.
5. Finally, a challenge can be closed.

In the following paragraphs the main corresponding interactions of the blockchain component with additional MANU-SQUARE platform components can be seen (for a visual reference please refer to Figure 6). For all such interactions there are two broad categories of actions that are taken and supported by a RESTful API exposed by the blockchain platform, namely invoking a transaction on the blockchain, intended to record data, and querying the blockchain platform for data previously recorded (latest state or historical data).

1. Collaborative Ideas development – tracking the evolution of a challenge through related supplied ideas. A challenge is opened by an entity and is recorded in the blockchain. The open process enables multiple entities to contribute ideas to a specific challenge. Each such idea is in turn registered in the blockchain. The history of the evolution of ideas leading to the fulfilment of a challenge is fully trackable by using the blockchain platform. This component is further described later in § 5.
2. RFQ management – the blockchain component provides the underlying mechanism for keeping track and advancing the RFQ process among the customer and the potential suppliers. Naturally, the blockchain keeps track of the history of the interaction from beginning to end and can serve as the reference point for the agreed upon terms and the evolution of the process.
3. Reputation management – the reputation management process is handled by the MANU-SQUARE platform, for all involved entities, both suppliers and customers. This process is deemed an important one for attracting entities to use the platform, and for the long term sustainability of the platform. Having a blockchain based infrastructure to keep track of reputation management calculated scores is essential for the trust associated with the entire platform by current and prospective customers joining and using the platform.

4 SUPPORTING BLOCKCHAIN PLATFORM ARCHITECTURE

4.1 Network and peers

Underlying a Hyperledger Fabric network, as most permissioned networks is the notion of a consortium. A consortium consists of a group of organizations that agreed to set up a blockchain network between them, establishing the governance body and rules. As can be seen in Figure 8, most central organizations within a blockchain network will deploy (or use) a Certificate Authority (CA) on their behalf, and will contribute peer(s), which are server components that endorse, validate, and hold replicas of the shared ledger. In addition, an ordering service needs to be set up by the organisations, to order the transactions, cut blocks and make them available to the peers.

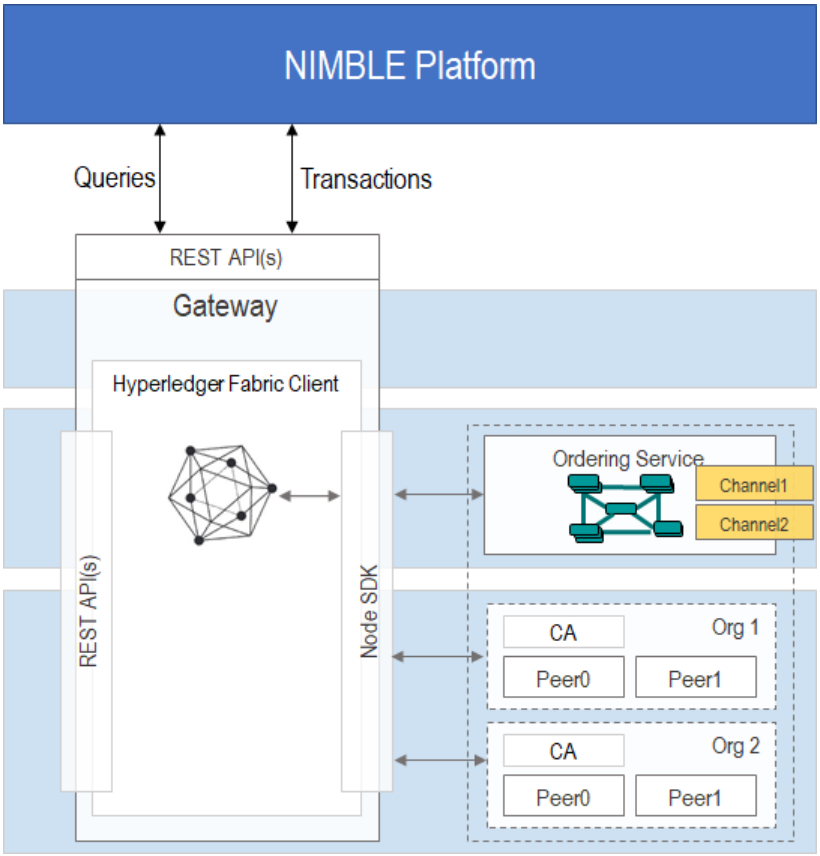


Figure 8: Blockchain network components

Transactions among members in a consortium are performed in the context of a channel. A channel creates a separate ledger visible only to the organizations included in the channel.

To become a network member, first, each participant needs to be registered and enrolled in the network via a Certificate Authority. A user with an appropriate role (such as admin) can register additional users from his organization. Using a secret (password) received during the registration process the new user can enroll, thus receiving the required credentials for participating in the blockchain network. Using these credentials a user may start invoking transactions on the blockchain.

To bootstrap a network there needs to be an ordering service, peer processes need to be established on behalf of organizations, corresponding channels need to be created, and the appropriate cryptographic material should be distributed, including the certificates required to participate in the network, to each entity. Once the backbone is in place, chaincodes can be installed and instantiated. A chaincode needs to be installed on each peer which may endorse transactions for that chaincode (endorsing peers are the only ones that actually execute the chaincode). The chaincode

needs to be instantiated on one of the peers, to create the bond between the chaincode and the channel, and run the initialization method specific to that chaincode.

Once the chaincode is in place users can start invoking transactions and queries on the blockchain channel. By using a client the user assembles a transaction and sends it to the endorsing peers. In the MANU-SQUARE case the application receiving REST requests from MANU-SQUARE tools performs this operation. Endorsing peers policy is determined per chaincode and establishes the identity of potential endorsers and the conditions that have to be satisfied for a transaction to be approved. Once the client has received responses from the endorsing peers, the client can evaluate whether the transaction abides by the endorsement policy and can thus go through and be sent to the ordering service to be included in a future block, or whether it needs to be dropped. Endorsing peers are the only ones that actually run the chaincode (in a simulating mode), and return the corresponding read and write sets of the transaction, namely the keys and versions of variables that were read or written by the simulated execution of the chaincode. Allowed tractions are then sent, along with the corresponding read and write set to the ordering service. The ordering service in turn orders incoming transactions, cuts blocks, and makes the blocks available to the peers. The Peers in turn obtain a new block, validate the transactions in it, and apply the write set for the transactions that have been determined to be valid.

In Figure 8 the components described above in play can be seen. On the right hand side the main components of the blockchain network itself namely the ordering service, peers, and the certificate authorities are presented. The channels which are declared in the system associated with an ordering service and a sub-set of the peers can be further seen. In the middle we can see an application which embeds a Fabric client to communicate with the blockchain while exposing a REST interface for other MANU-SQUARE components.

The blockchain system mostly consists of three layers. First, a physical layer of deployment which includes the establishment of the network consisting of organizations, their participating servers (peers in Fabric), channels, cryptographic material, and more. A second layer includes the establishment and distribution of smart contracts (chaincodes), which programmatically determine the rules and actions to be followed. These smart contracts control the state that is saved in the underlying blockchain DB. On top of these lie the business layer which connects between external entities and the underlying blockchain infrastructure. In our deployment, as in most cases, this layer consists of the programmatic core of the interactions to follow, which exposes, on the one hand, to the higher layers, consisting of external applications, a RESTful interface through which the interactions with the blockchain are mediated. On the other side it embeds a Blockchain client (such as the Fabric NodeSDK client) which is in charge of interacting directly with the blockchain in the form of invoking transactions, invoking queries, and establishing call-backs. These call-backs enable an asynchronous mode of operation in which a process is notified by the blockchain network on the occurrence of events which were declared as being of interest to the application or higher layers components.

4.2 Deployment

The deployment of the blockchain based support within MANU-SQUARE consists of 2 main stages, namely first the underlying blockchain network is created and deployed, and second the proxy gateways that connect the blockchain with the rest of the platform are deployed in the form of web responsive applications. For each supported tool (RFQ, reputation management, and collaborative idea generation) two environments and parallel applications are installed; one to support development, integration, and testing operations for the platform; and a second one to keep clean to be used in a production environment.

The deployment described herein is a one-time setup, which gets all components up and running from scratch.

As prerequisites the following technologies are required to be installed in the target VM:

- docker – as components of the solution run in docker containers
- docker-compose – deployment done through docker-compose
- node-js – as the proxy gateway application processes run as a node-js application.

- Moreover the target VM should have public internet access, such that interaction with it from the outside can be supported.

The first phase consists of creating and deploying the underlying blockchain network, along with all its components. The corresponding code and information is available through the project GitLab repository located in https://gitlab.com/manusquare/blockchain.git.

The first stage consist of cloning the project repository from GitLab to have all the code and information available locally. Once the repository exists locally, there are two scripts that need to be invoked in sequence:

- *deployment/bootstrap.sh* - This script installs Hyperledger Fabric version in of the indicated version (tested for 1.4.3); Downloads the required sub-components for a new blockchain network including main services, such as cryptogen, ordering node, peer, and fabric-ca. The script further downloads the required Hyperledger Fabric docker images.
- *deployment/run_network_local.sh* up – orchestration and execution of all the building block of a basic blockchain network. At the successful completion of this step you should see the following successful output meesage: All GOOD, entire network is up and running. A corresponding screen shot can be seen in Figure 9, including main components that have been deployed and are up and running.

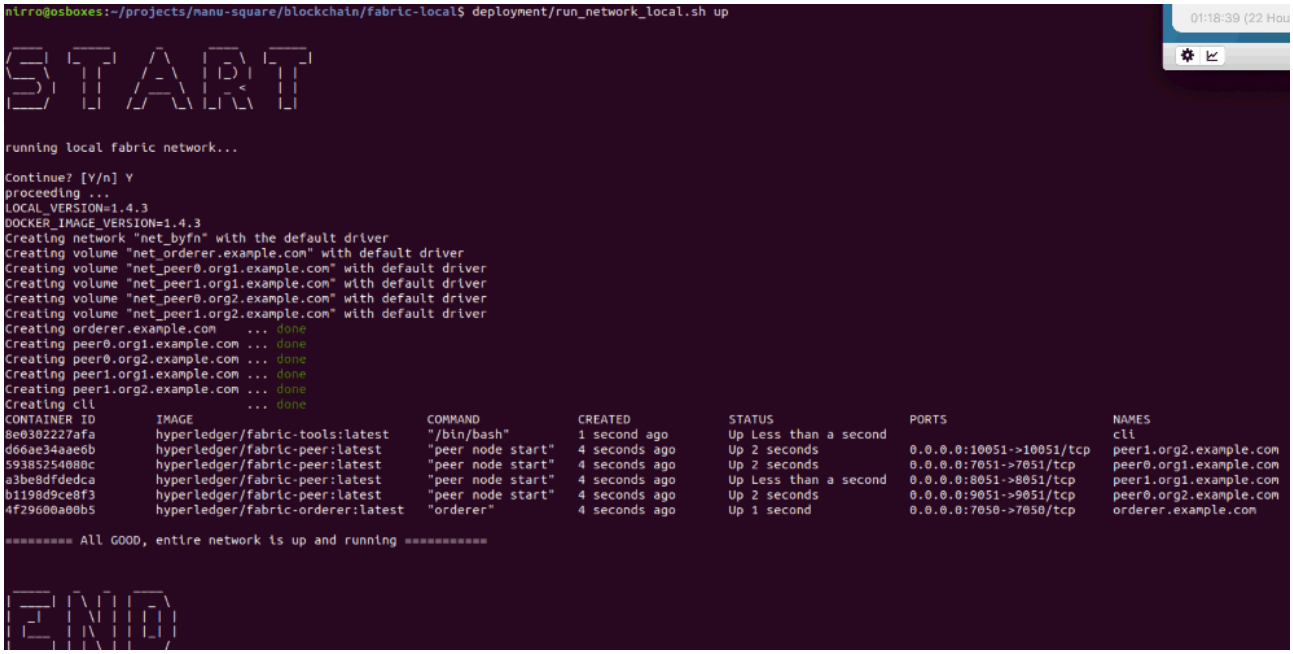


Figure 9: Successful deployment of a blockchain network

The second stage comprises of deploying the corresponding chaincodes and proxy gateway apps. In the fabric-apps project directory there are dedicated folders for each supported tool and target environment, for example *man-idea-management* which includes all necessary files for instantiating support for the development version of the support for collaborative evolution of ideas. Furthermore, each such folder contains a server/swagger.json file, which includes the Swagger[6] definition of the specific component interface.

There is one script in this repository which installs all components, namely the *deploy_all.sh* script. This includes an automation script to deploy *manu-rfq* and *manu-ideas-management* using two environments, namely development and production. Once these stages are performed successfully the entire blockchain support for RFQ and idea management should be in place. The script for installing the different components is as follows:

[6] https://swagger.io/

- *create_new_channel.sh -c manu-idea-management-prod-channel* – creates the corresponding channel that will be used by the specific component.
- *deploy_chaincode.sh -c manu-idea-management-prod-channel -n idea_management_prod -p github.com/chaincode/idea-management -v 1.0* creates and deploys the corresponding chaincode that will be used by the specific component.

As a helper and a sanity check it's possible to access the online swagger files for the different components in their specific URLs. Moreover. The swagger enables quick testing that the components are indeed operating as expected. The swagger definitions can be found in the following URLs:

- Http:// manusquaredev.holonix.biz:6891/api
- Http:// manusquaredev.holonix.biz:10891/api

All the general components comprising a blockchain network are in place (such as peers and orderers), and specific support for MANU-SQUARE tools are in place as well, such as a channel, chaincode, and application for each tool in each corresponding environment.

5 MANU-SQUARE'S BLOCKCHAIN API - DOCUMENTATION AND EXAMPLES

Once a generic blockchain based platform infrastructure (Figure 8) has been put in place, it's possible to develop specific blockchain constructs on top of the infrastructure to provide capabilities which are specific to the MANU-SQUARE platform hosting the blockchain network. That has been done and deployed for the three chosen scenarios.

5.1 Collaborative ideas evolution

The blockchain support for collaborative idea generation is comprised of two major entities, namely challenges and ideas. Challenges encompass the overall concept or question that needs to be resolved using external contributions. Ideas are possible components that can add value or move us closer in the direction of resolving a challenge.

Consequently, the interface provided by the idea management support tool is divided into the two major entities involved, as can be seen in Figure 10. The figure depicts the high level swagger definition of the capabilities and interfaces exposed by the blockchain based service.

For challenges there are interfaces for creating a new challenge, for listing existing challenges (with some variations based on parameters set for the call. Further users can obtain information about a specific challenge, and finally a challenge can be closed.

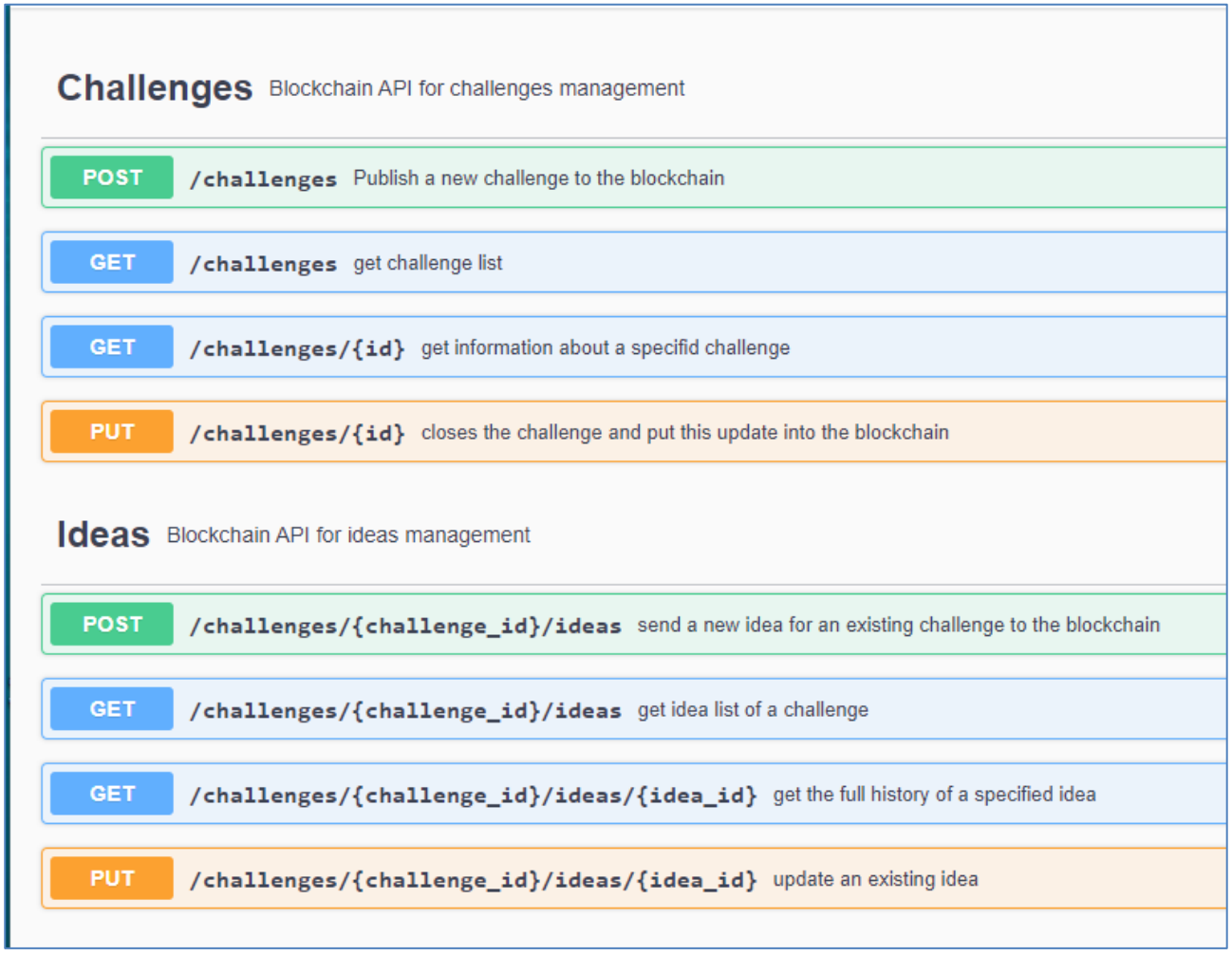


Figure 10: Swagger definition of the blockchain based idea management support

The first step in the process is to publish a challenge, thus opening up a call for contributions. This can be achieved by invoking a POST REST command to the challenge interface as can be seen in Figure 11. The call includes as parameters the title of the challenge, which comprises of a text defining what is the issue that needs help resolving.

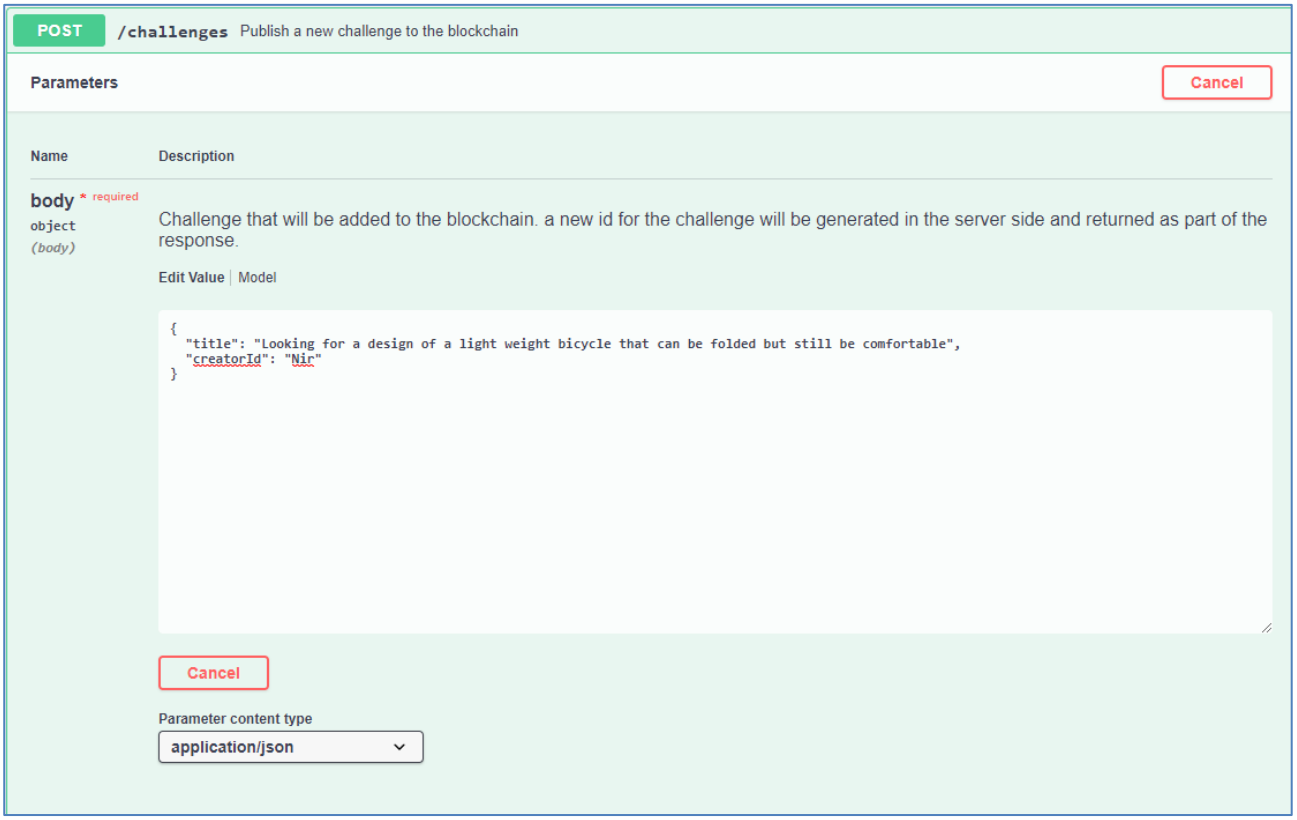


Figure 11 Publish a new challenge to the blockchain

The successful response to the request above to open a new challenge can be seen below in Figure 12. In the figure itself we can see the curl command that is used to access the required capability via the respective interface, the URL that is used and the actual response from the server. The response indicates a successful operation and the id assigned to it by the server is returned as well. The status of the challenge is ser to “opened” and an associated timestamp is returned.

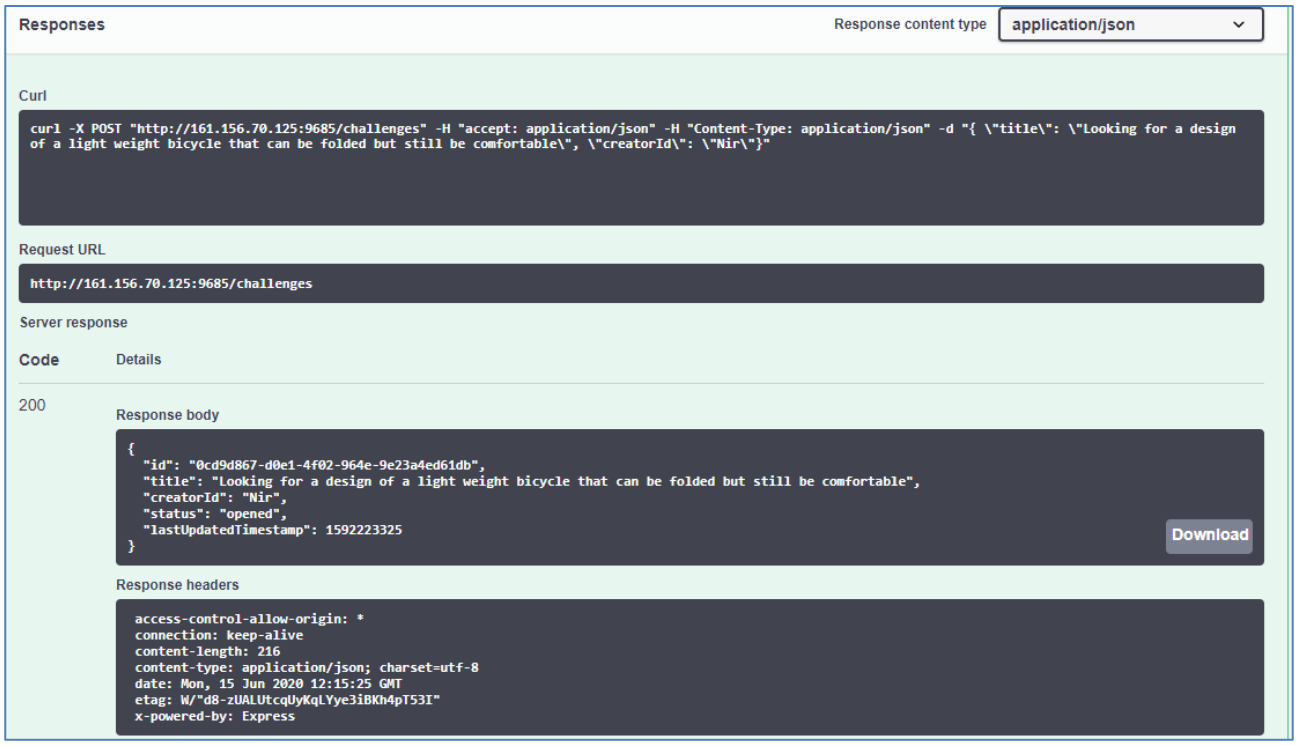


Figure 12: publish challenge response

The list of existing challenges can be retrieved, as seen in Figure 13. There's a possibility to retrieve only challenges in the "opened" state. Furthermore, there exists an option to retrieve all challenges created by a specific user id.

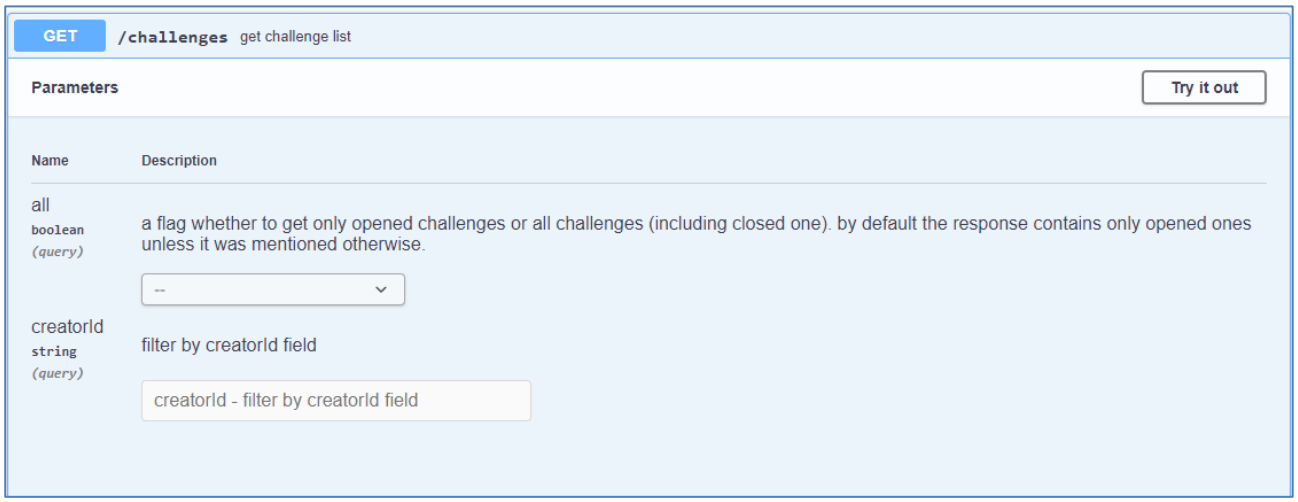


Figure 13: Retrieve the challenges list

The response can be seen below in Figure 14 . Once again the curl command being used, the target URL, and the actual response received from the server can be seen. The response consists of all the challenges that abide by the criteria set based on the parameters set in the corresponding GET request.

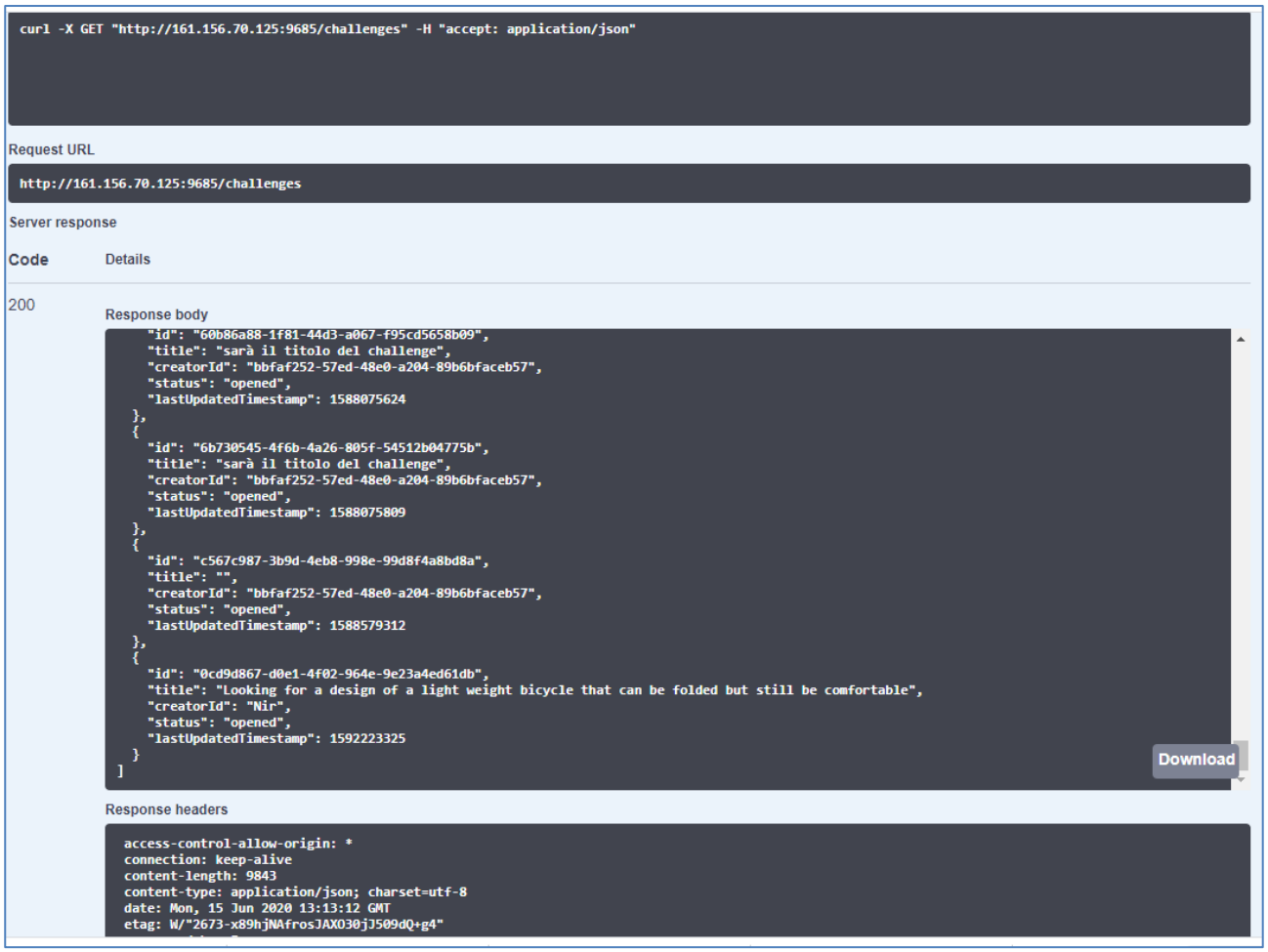


Figure 14: retrieve challenges response

There exists the possibility to obtain information on a specific challenge, specified by the corresponding challenge id, as can be seen in Figure 15.

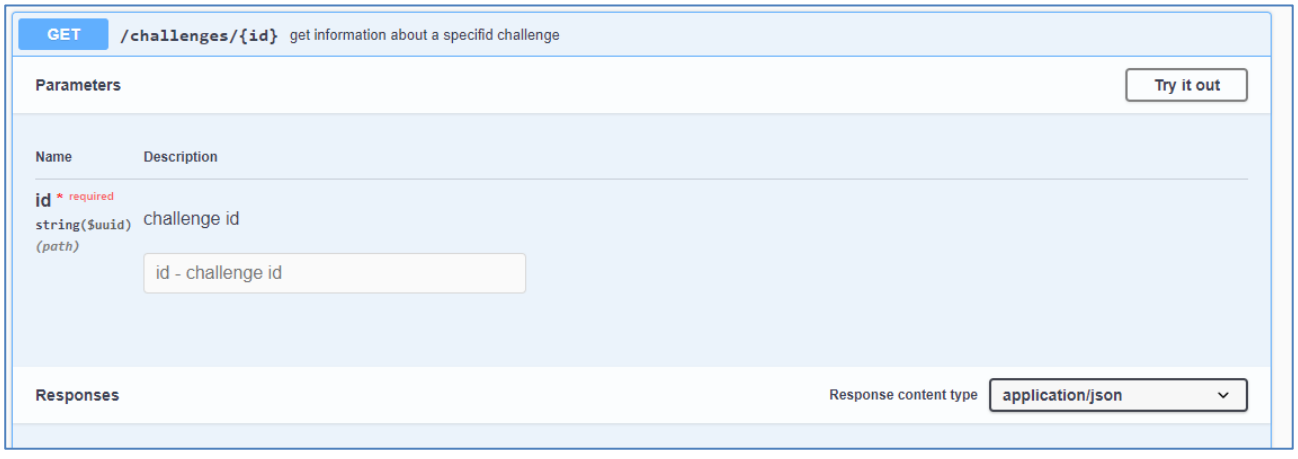


Figure 15: retrieve information on a specific challenge

The corresponding response can be seen in Figure 16.

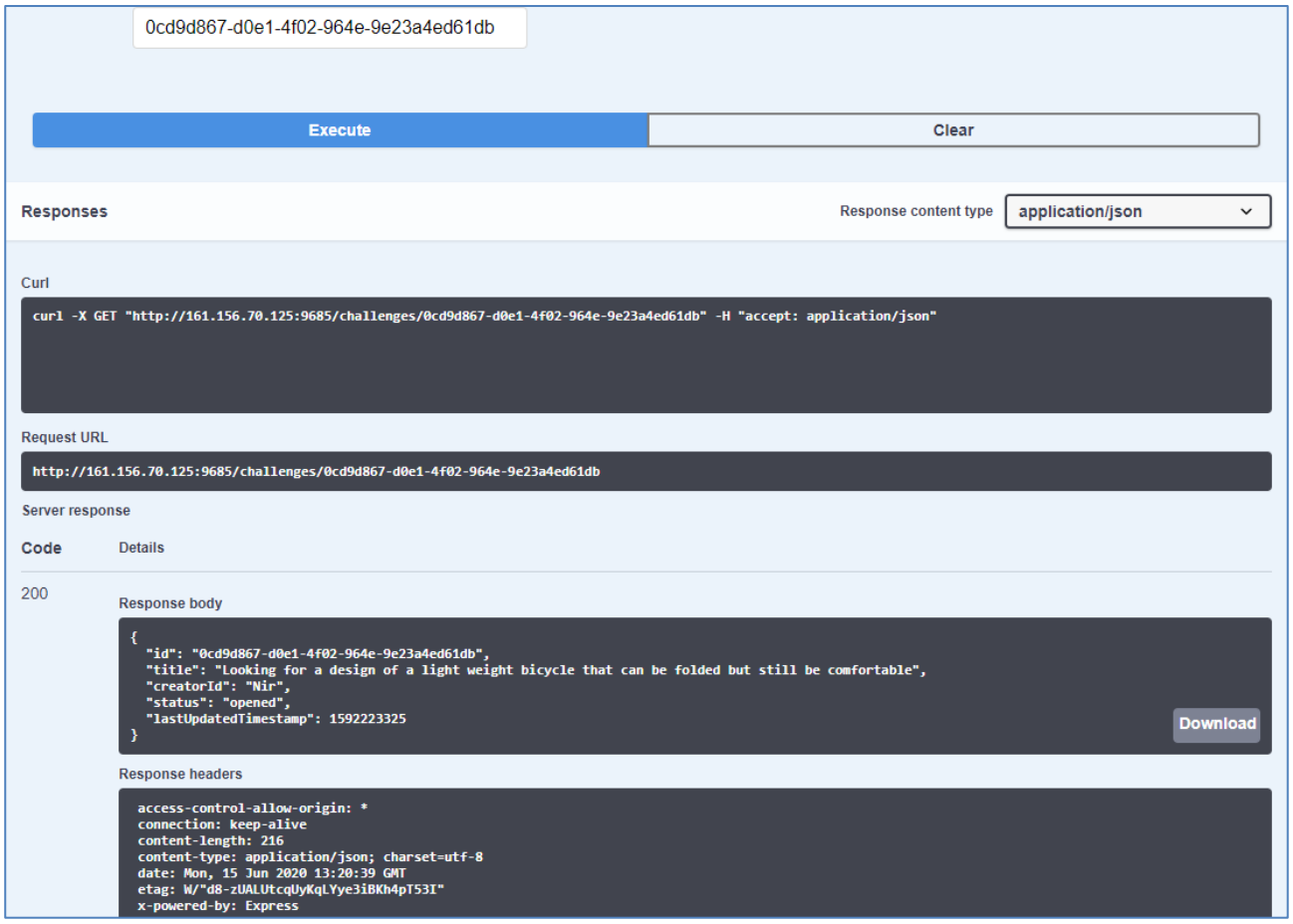


Figure 16: retrieve challenge response

Finally, a challenge can be closed (see Figure 17).

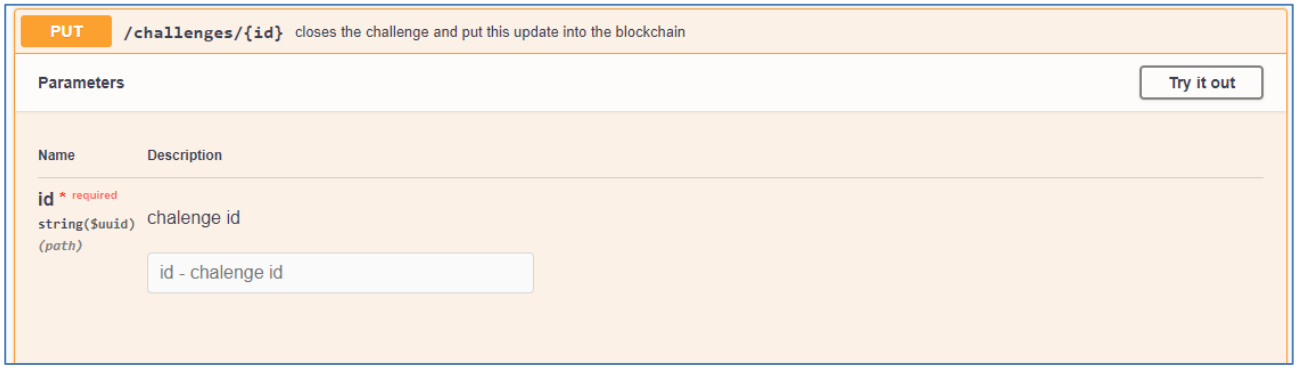


Figure 17: challenge close

The response can be seen in Figure 18, in which we can see that the corresponding challenge status was set to “closed”.

Figure 18: delete challenge response

Next, there exists a second family of operations dealing with idea management within existing challenges. First, the possibility to add an idea to an existing challenge (see Figure 19) is available via the following POST command. In addition users can obtain a list of ideas associated with a challenge. Users can further obtain the full history of a specified idea. Finally there is a possibility to update a specific idea.

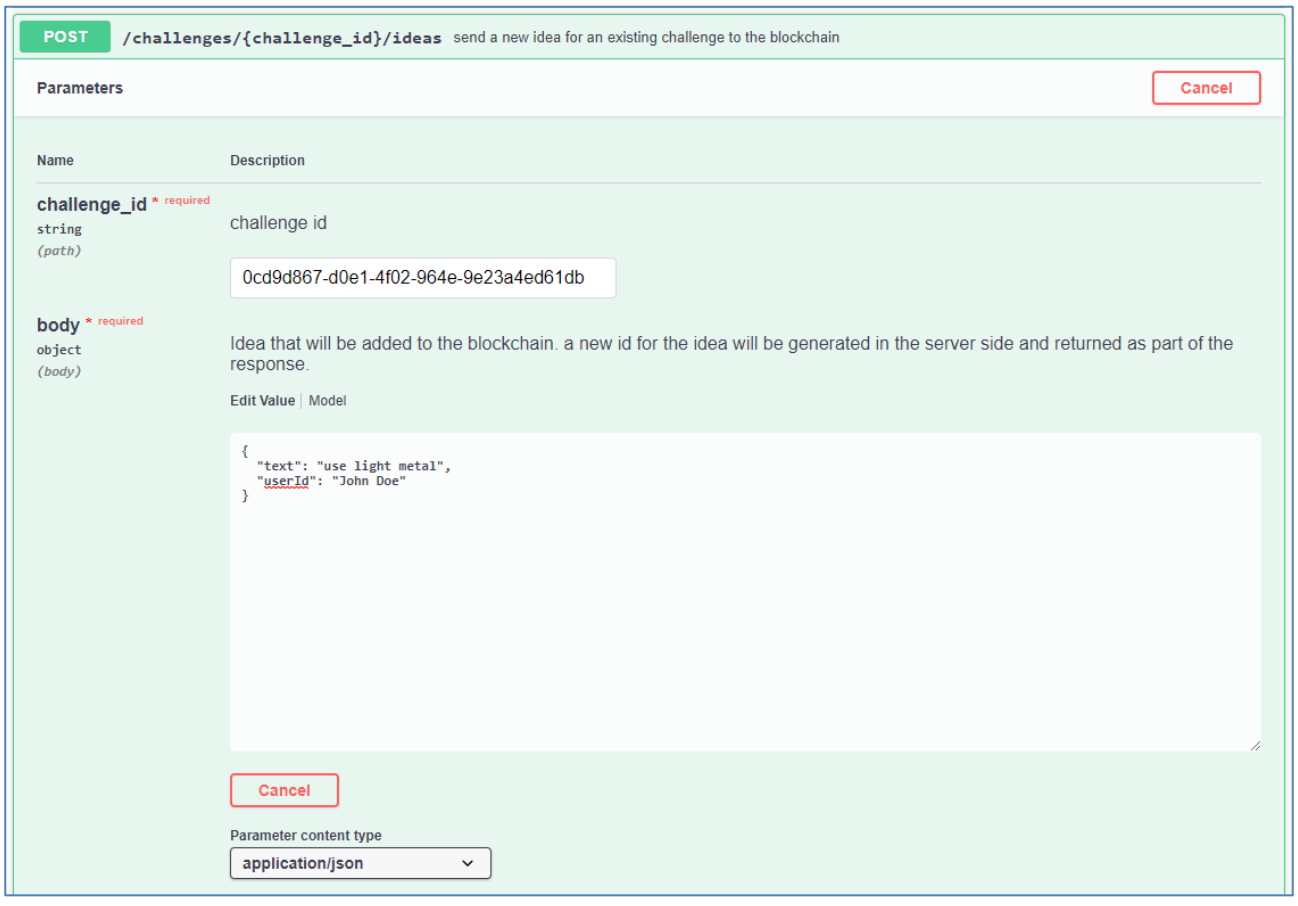


Figure 19: add an idea to a challenge

The corresponding response can be found in Figure 20.

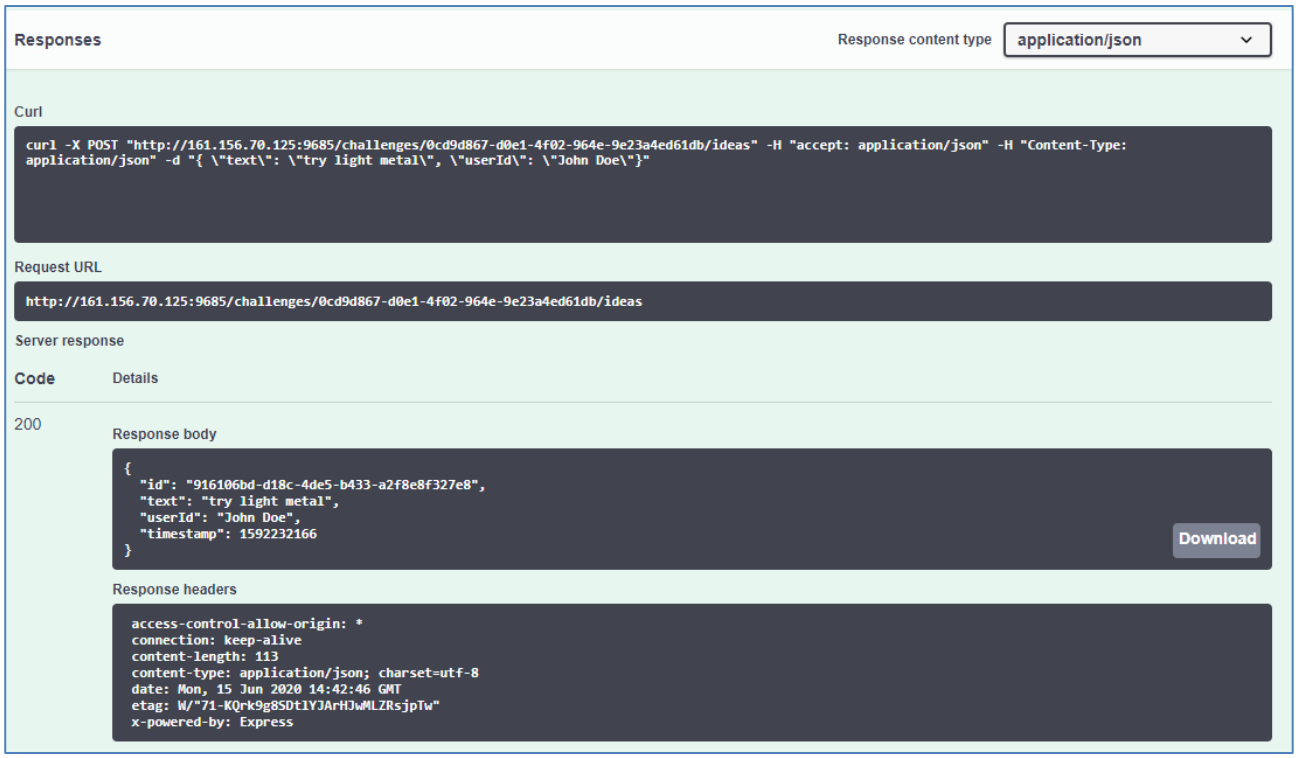


Figure 20: response to the addition of an idea to a challenge

Next we can obtain the list of ideas that were contributed and are associated with a specific challenge (See Figure 21).

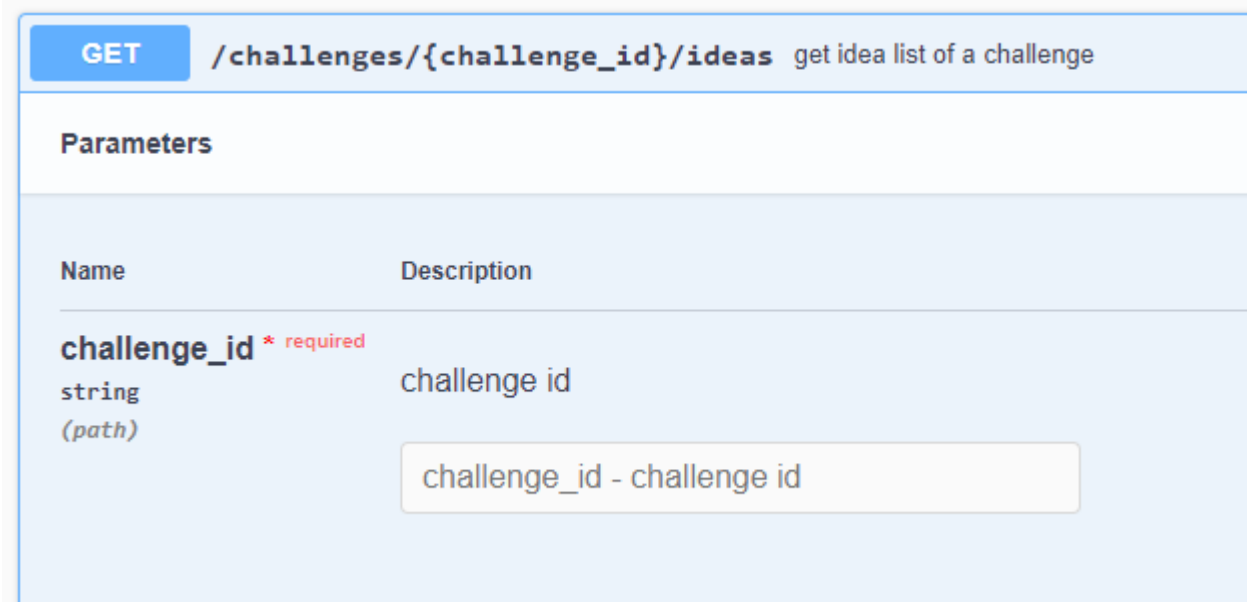


Figure 21: Obtain a list of ideas per challenge

The corresponding response can be found in Figure 22.

Curl

```
curl -X GET "http://161.156.70.125:9685/challenges/0cd9d867-d0e1-4f02-964e-9e23a4ed61db/ideas" -H "accept: application/json"
```

Request URL

```
http://161.156.70.125:9685/challenges/0cd9d867-d0e1-4f02-964e-9e23a4ed61db/ideas
```

Server response

Code	Details
200	Response body


```
[
  {
    "ideaHistory": [
      {
        "id": "916106bd-d18c-4de5-b433-a2f8e8f327e8",
        "text": "try light metal",
        "userId": "John Doe",
        "timestamp": 1592232166
      }
    ]
  }
]
```


Download

Response headers

```
access-control-allow-origin: *
connection: keep-alive
content-length: 133
content-type: application/json; charset=utf-8
date: Tue, 16 Jun 2020 06:38:37 GMT
etag: W/"85-pb+8syqRdyU5v1BJY2oeHN3yxp4"
x-powered-by: Express
```

Figure 22: response for obtaining ideas per challenge

There also exists the possibility to obtain the history of a specific idea (Figure 23).

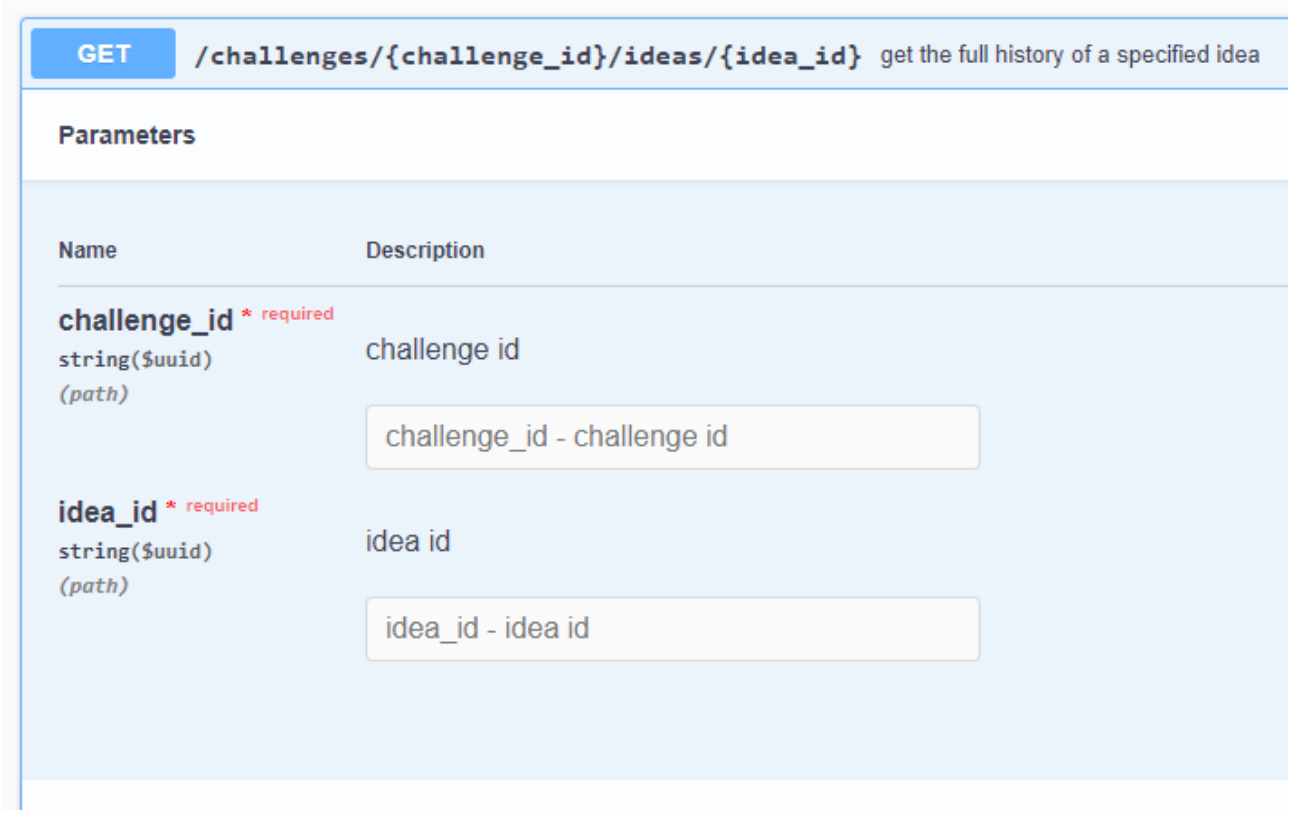


Figure 23: obtain the history of an idea

The response can be seen in Figure 24.

Figure 24: response for obtaining the history of an idea

We expose the possibility to update an existing idea (Figure 28).

Figure 25: update an idea

The response can be seen in Figure 26. In the response we can see the construction and evolution of an idea based on multiple contributions.

Figure 26: response for idea update

Finally, several models are used for the creation of these capabilities, as can be seen in Figure 27.

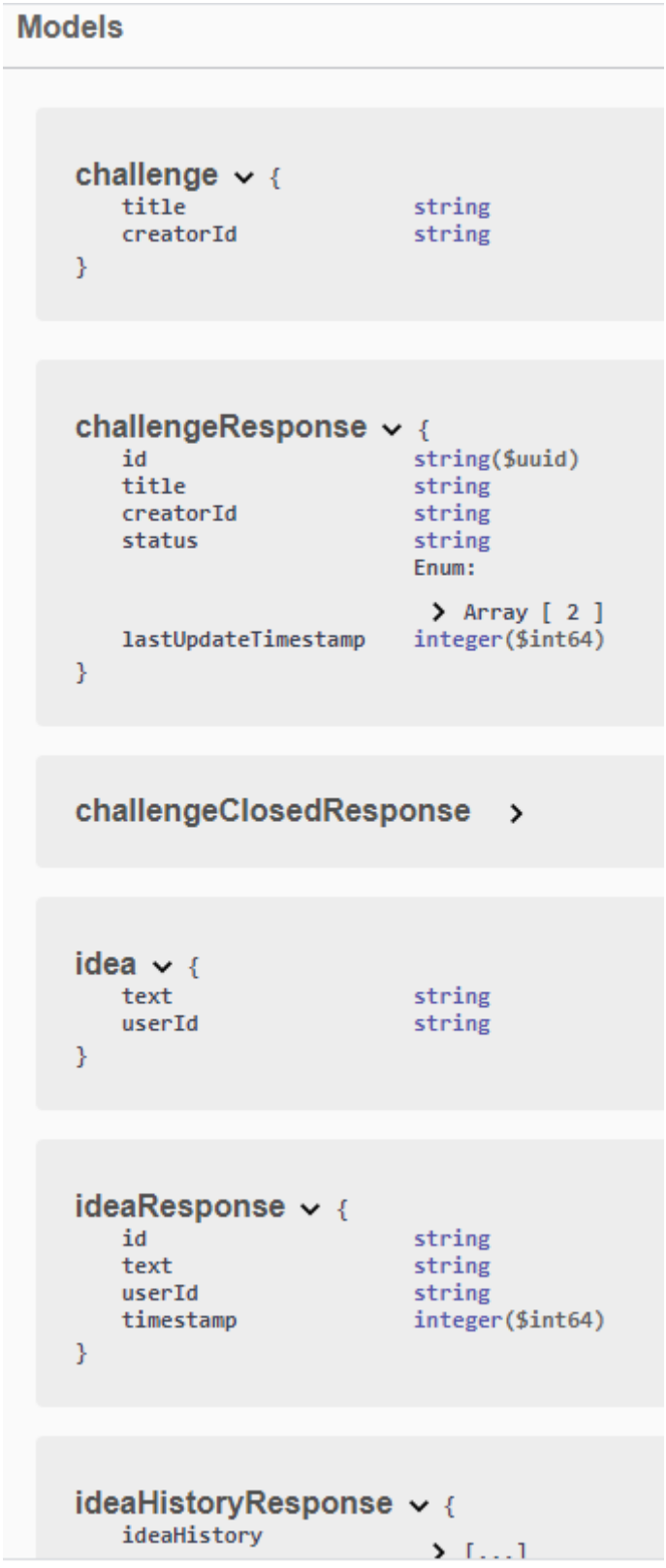


Figure 27: models used for capabilities introduced

5.2 RFQ

One of the prime examples of supply chain constructs that is supported by a blockchain backbone is the RFQ process, which is manifested in MANU-SQUARE as well. We have created a simple template which demonstrates the utility of the blockchain as the underlying RFQ process back-end. The blockchain serves as the single source of trust and truth between

participating partners. Naturally, variations of this RFQ process can be constructed, based on the foundations we detail below, supporting both structured and less structured manners of interaction between the counter-parts.

The proposed and demonstrated RFQ process consists of three main primitives, namely: Publish RFQ (offer), negotiate / communicate (counter-offer), and finish (accept or reject). The initial published RFQ corresponds to an action that is advertised using a single channel to which the entity publishing the RFQ participates along with all relevant entities that potentially are able and willing to respond to the proposed RFQ. Thus, the process is initiated by an entity producing an RFQ and submitting it to the blockchain as a transaction invoking the "publish_rfq" function of the RFQ smart contract.

In Figure 28 we can see the swagger declaration of the blockchain based RFQ support in MANU-SQUARE. The definitions are divided into three categories, namely RFQ related actions, quotation related actions, and data models. The swagger definition can be accessed and tested online at the following address: http://161.156.70.125:6891/api/, corresponding to a VM on the IBM Cloud on which the blockchain infrastructure is deployed. For the RFQ section we can observe that there are established calls available for the creation and publishing of a new RFQ, for querying the existing RFQs, for extracting information on a particular RFQ, and for declaring the selected entity for its accepted offer.

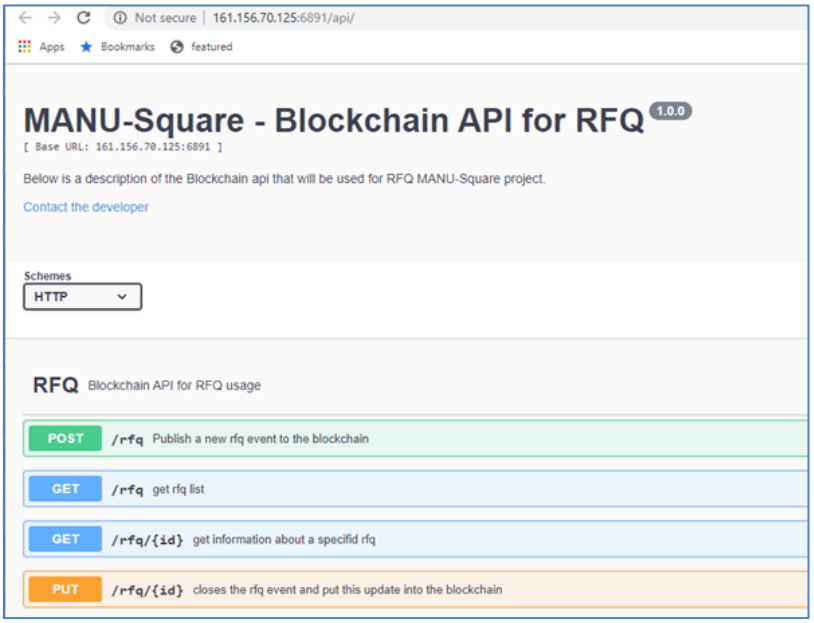


Figure 28: Swagger - RFQ calls

Figure 29 depicts the quotation related calls available for the RFQ process, which include the creation of a new quotation, querying for the latest information of a quotation, responding to a quotation, sending a counter-offer, and obtaining the historic evolution of a specific quotation.

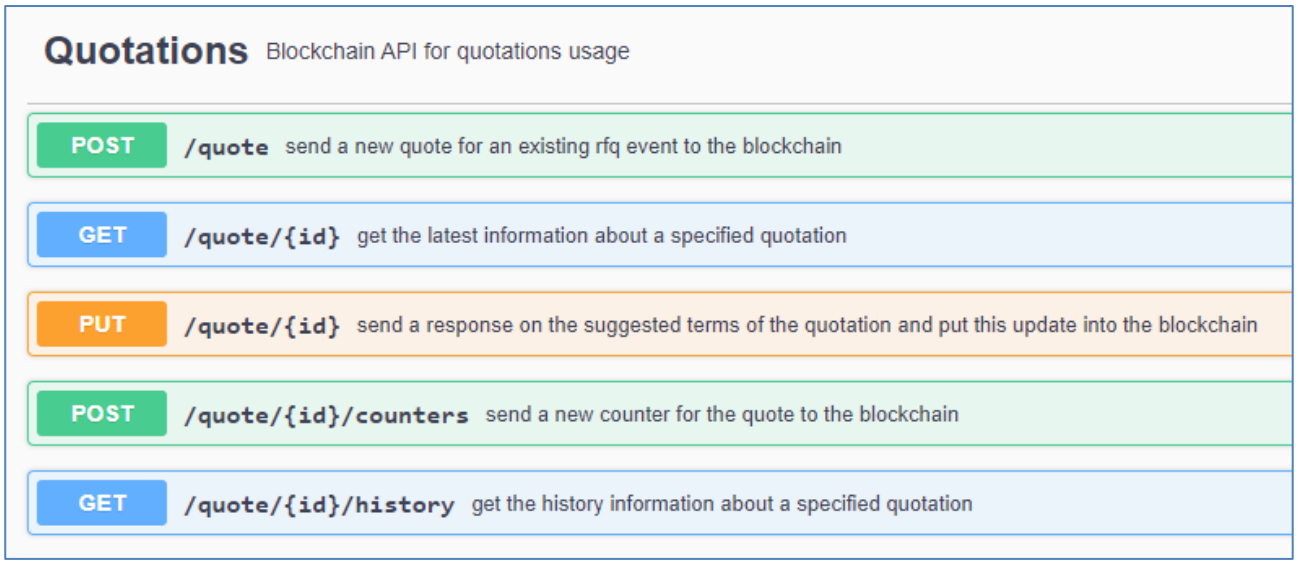


Figure 29: Swagger - quotation calls

Figure 30 depicts some of the data models that are used for the interaction between the MANU-SQUARE tools (especially the Ecosystem Data Manager), and the blockchain infrastructure supporting the RFQ process.

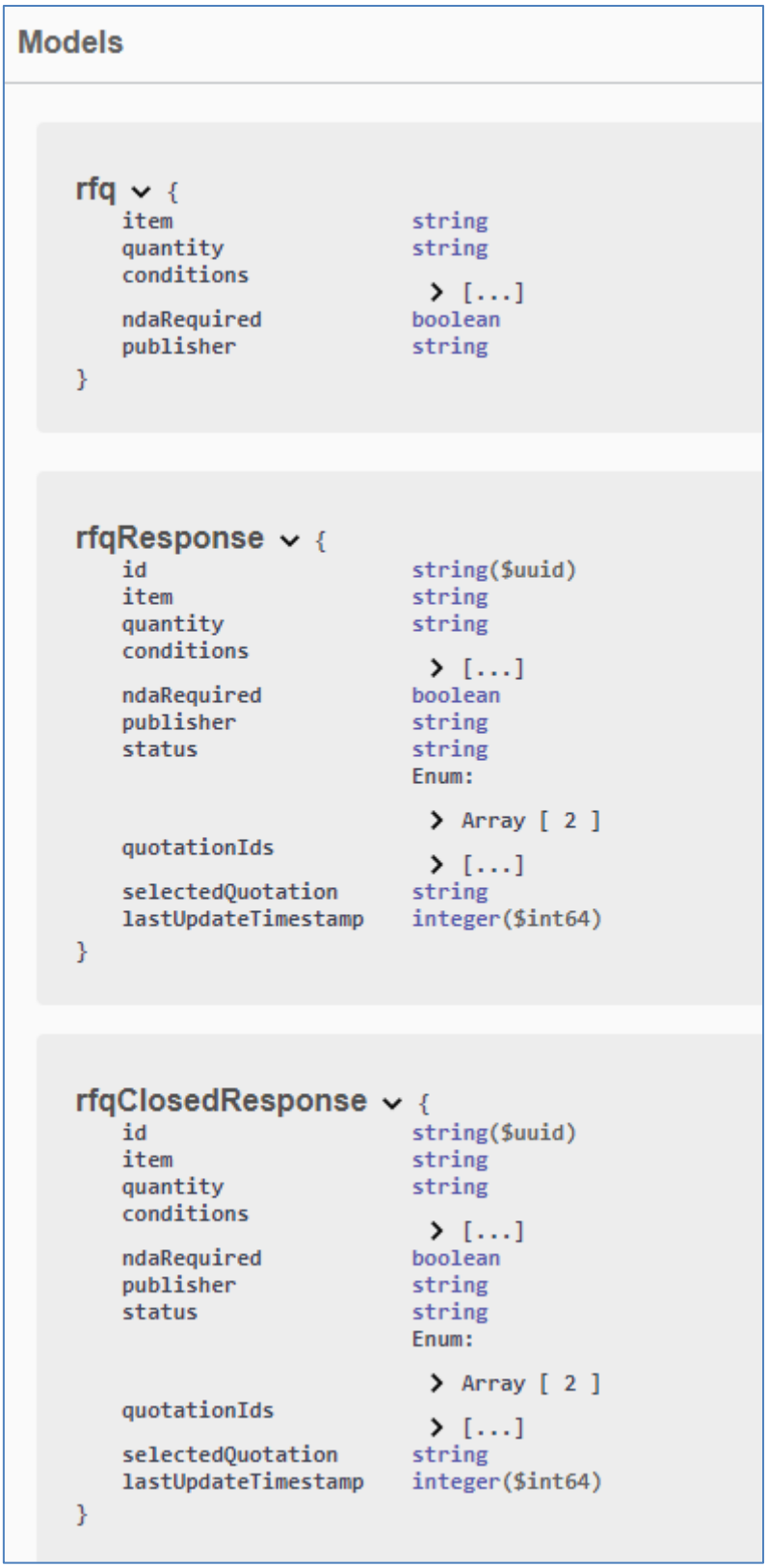


Figure 30: RFQ related data models

The RFQ process can work in a pull or push mode. In a push mode, in the background entities which are interested in receiving new RFQs, register themselves as interested in receiving such events on relevant RFQ related transactions performed on the blockchain. Upon the inclusion of such a transaction in a block, the corresponding call-back is invoked. A complementary pull mode requires the deployment of smart contracts that support various kinds of queries to respond to an entity looking for open and available RFQs or looking for the history of a particular RFQ. Once again, such operations

should be supported by a single channel for total visibility. For the integration of the blockchain as an infrastructure for MANU-SQUARE we make use of the pull mode in which entities query the blockchain for available relevant information.

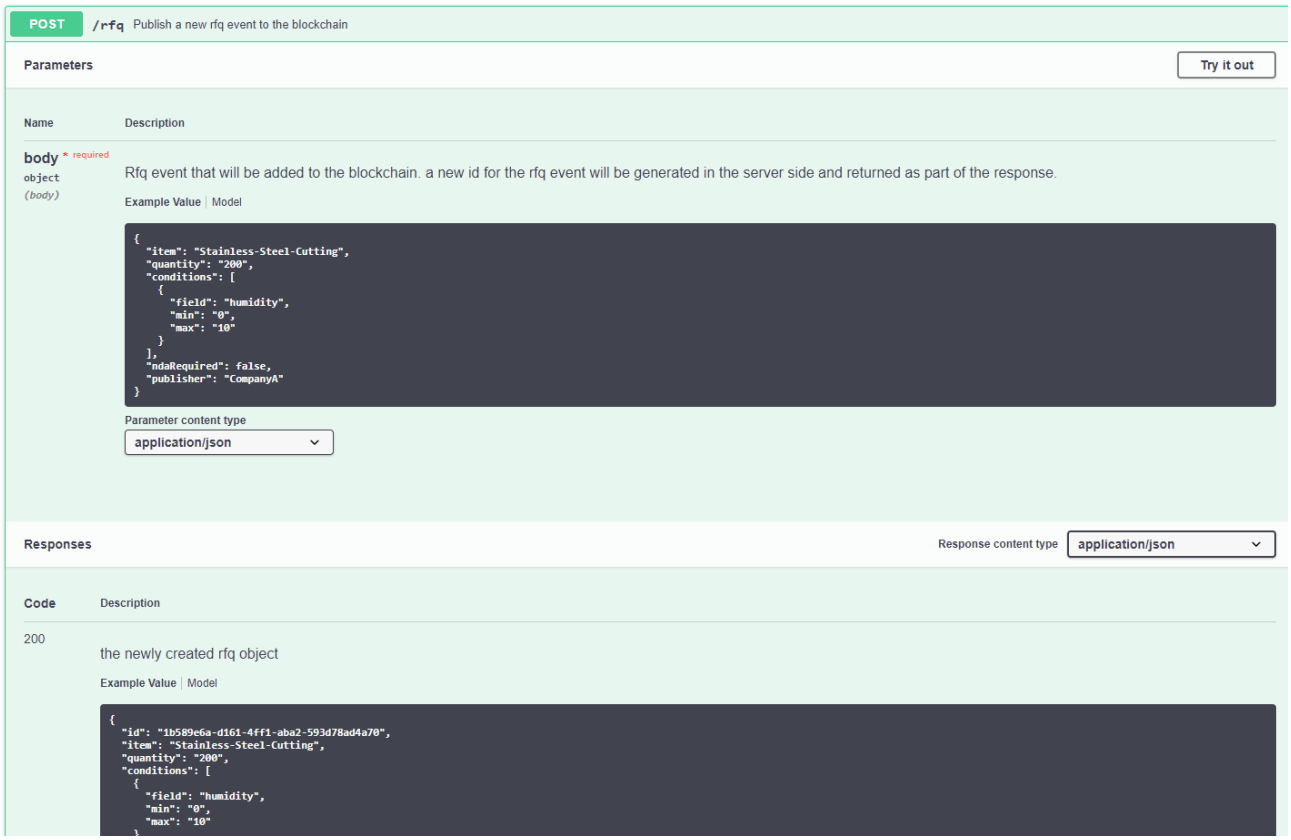


Figure 31: Publish a new RFQ

In the running example we use in this section, a company is interested to buy a capacity of a service for stainless steel cutting, and the conditions dictate the humidity in which the produced artefact should be kept throughout the process. Figure 31 depicts the API for the creation of a new such RFQ.

Figure 32 depicts the API for querying for RFQs; only ones in the state “open” are retrieved by default. All RFQs regardless of state can be retrieved as well by including a parameter in the call.

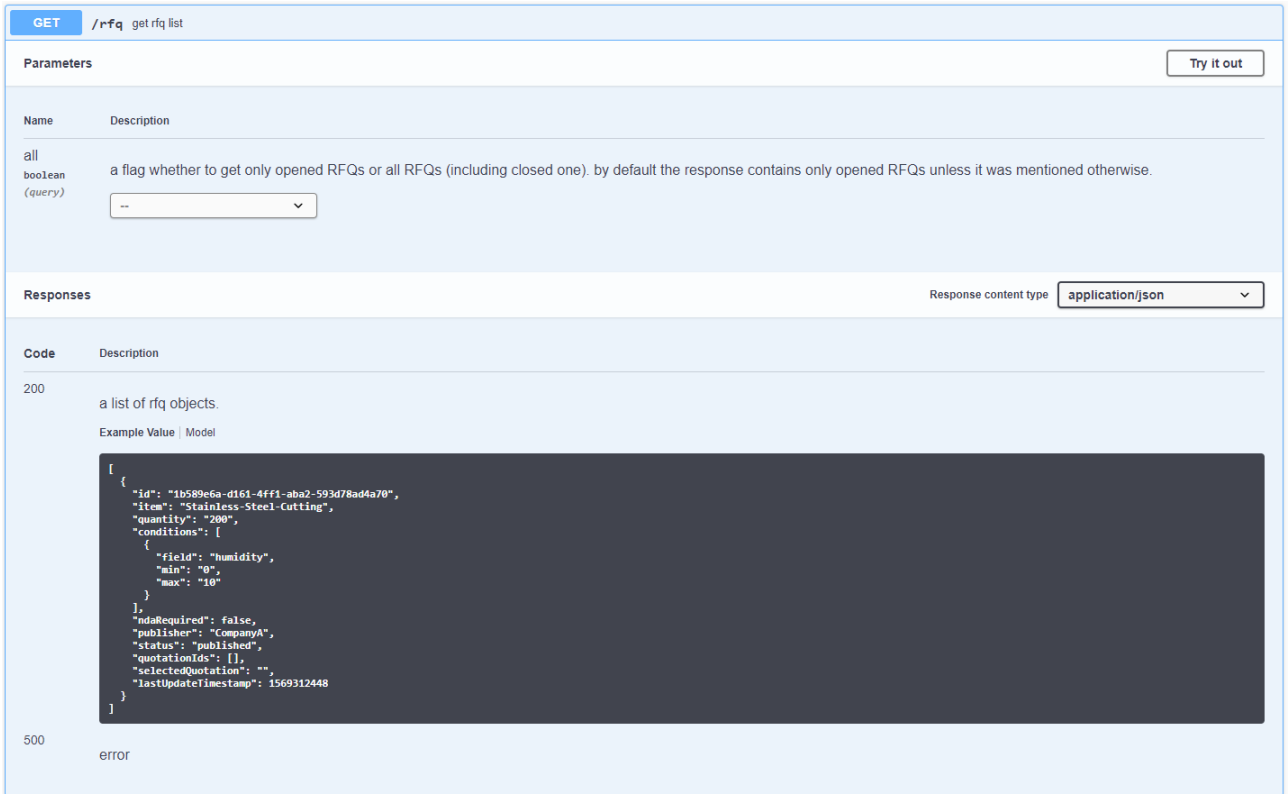


Figure 32: Query for the RFQ list

Figure 33 depicts the call necessary for obtaining the current state of a specific RFQ instance.

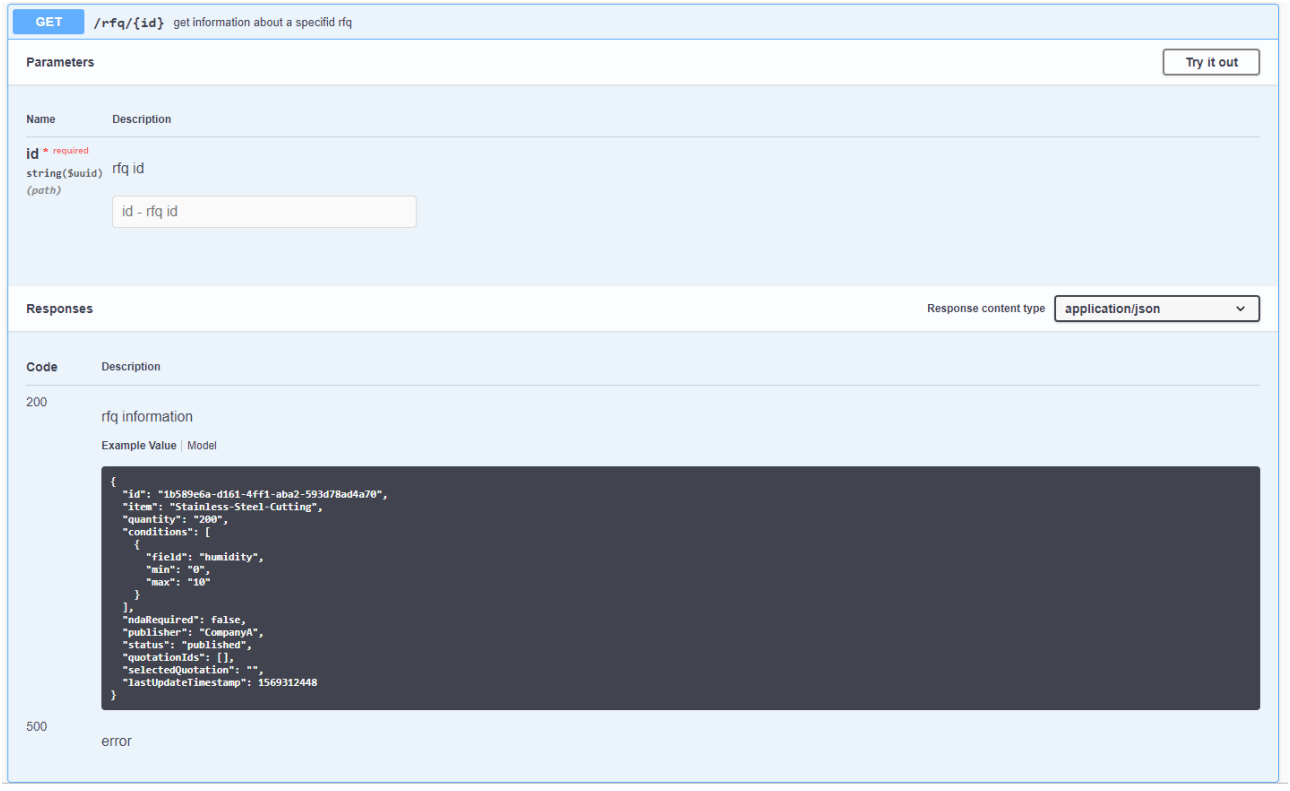


Figure 33: Obtain the current state of an FRQ

Entities receiving the notification of the publication of a new RFQ evaluate it internally and if interested can take one of two kinds of actions. First, they can ask for clarifications and negotiate, and second they can respond to the RFQ. Note that

these operations may be repeated multiple times until both sides are satisfied. This corresponds to a phase of clarifications, and potentially negotiations between the initiator of the RFQ and potential responders.

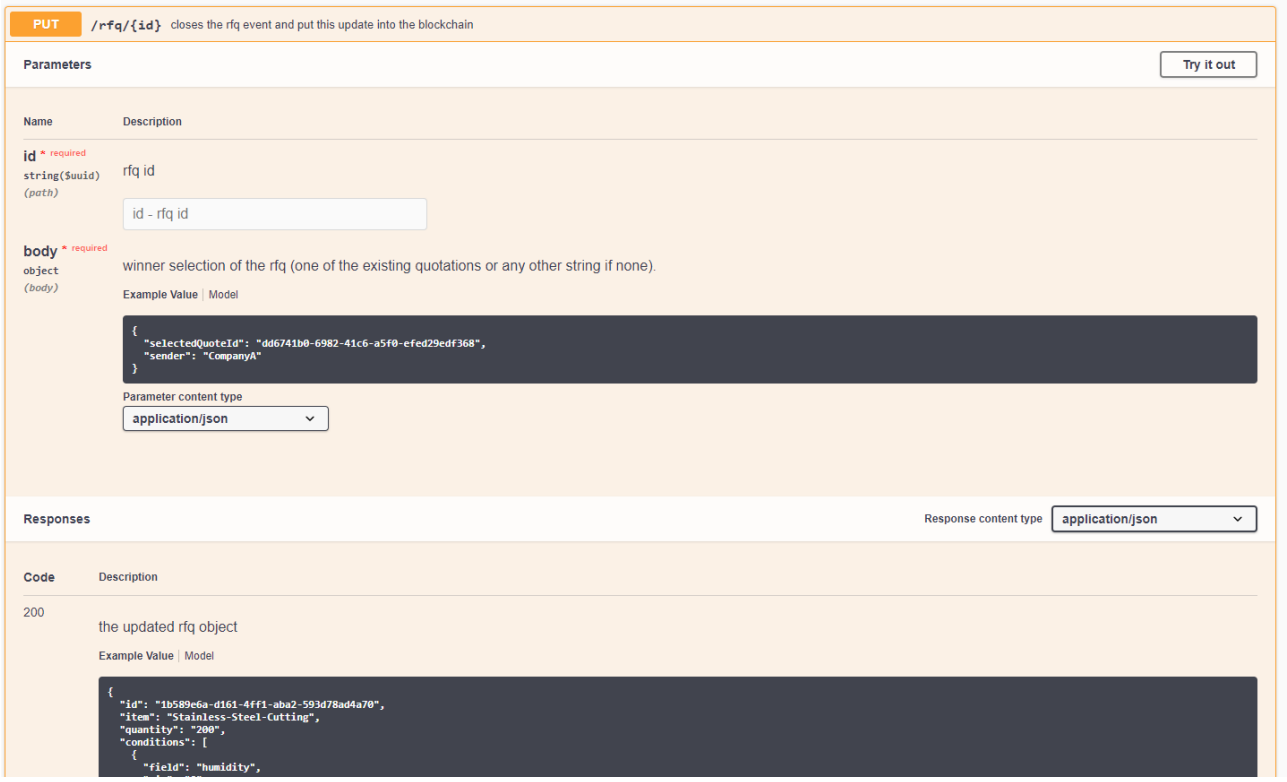


Figure 34: Accepting a quotation

The negotiation process consists of transactions being submitted to the blockchain, which are in turn recorded and committed into the ledger. Both parties of the negotiation process can query and receive the updated new relevant information by querying the blockchain passing through the Ecosystem Data Manager. Note that all access to the blockchain is performed though MANUSQUARE tools which are served by a dedicated application embedding a blockchain client.

Finally, the last step consists of the RFQ issuer choosing to accept an offer thus rejecting all other proposals. The API used for this operation can be seen in Figure 34. An acceptance message is sent to the selected company and all the other companies that participated in the RFQ process receive a rejection message indicating that the RFQ is now closed, which means that they were not chosen.

5.2.1 Quotations - Blockchain API for quotations usage

This sub-section provides additional information on the API interface calls used to handle quotations within the RFQ framework. Figure 35 depicts the API call to be used for sending a quotation for a specific RFQ.

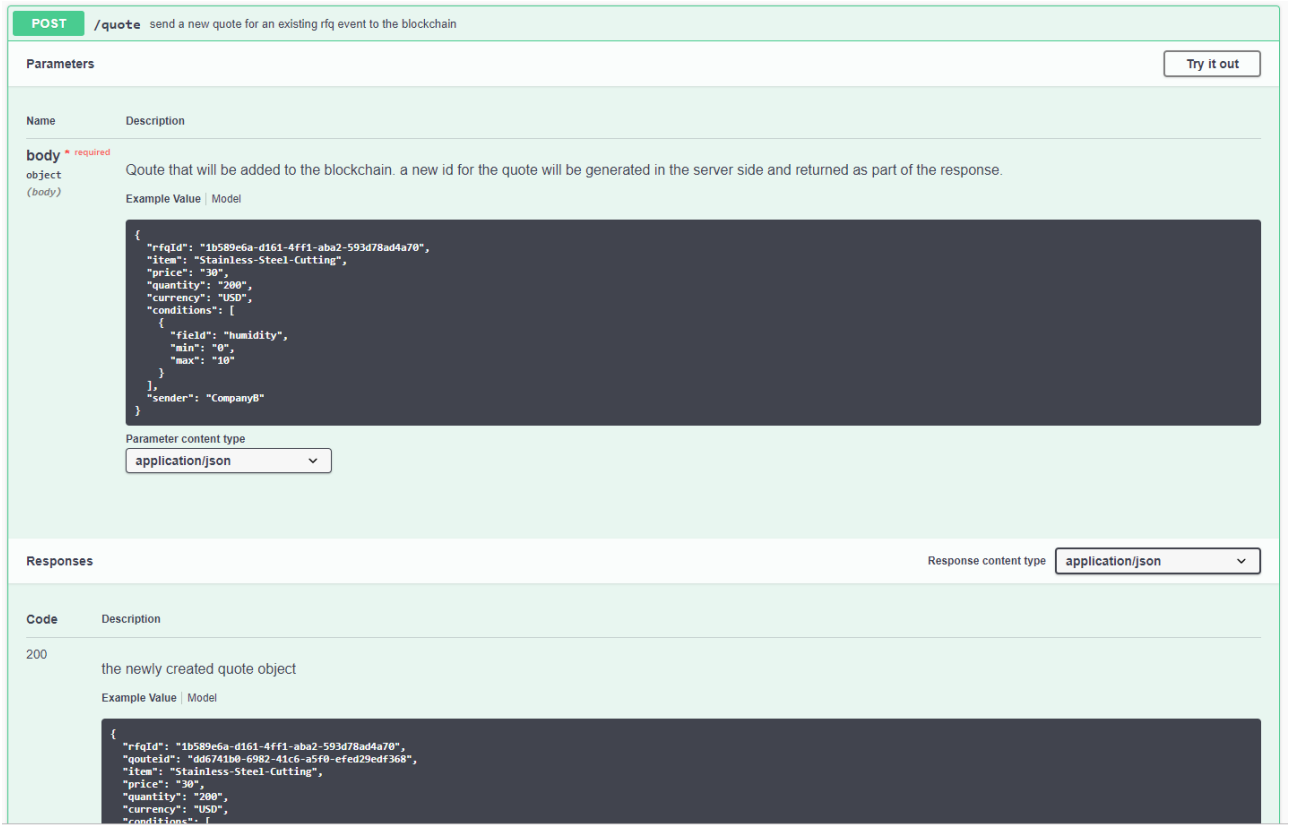


Figure 35: send a quotation for an RFQ

Figure 36 depicts the API call used to retrieve information on a specific quotation.

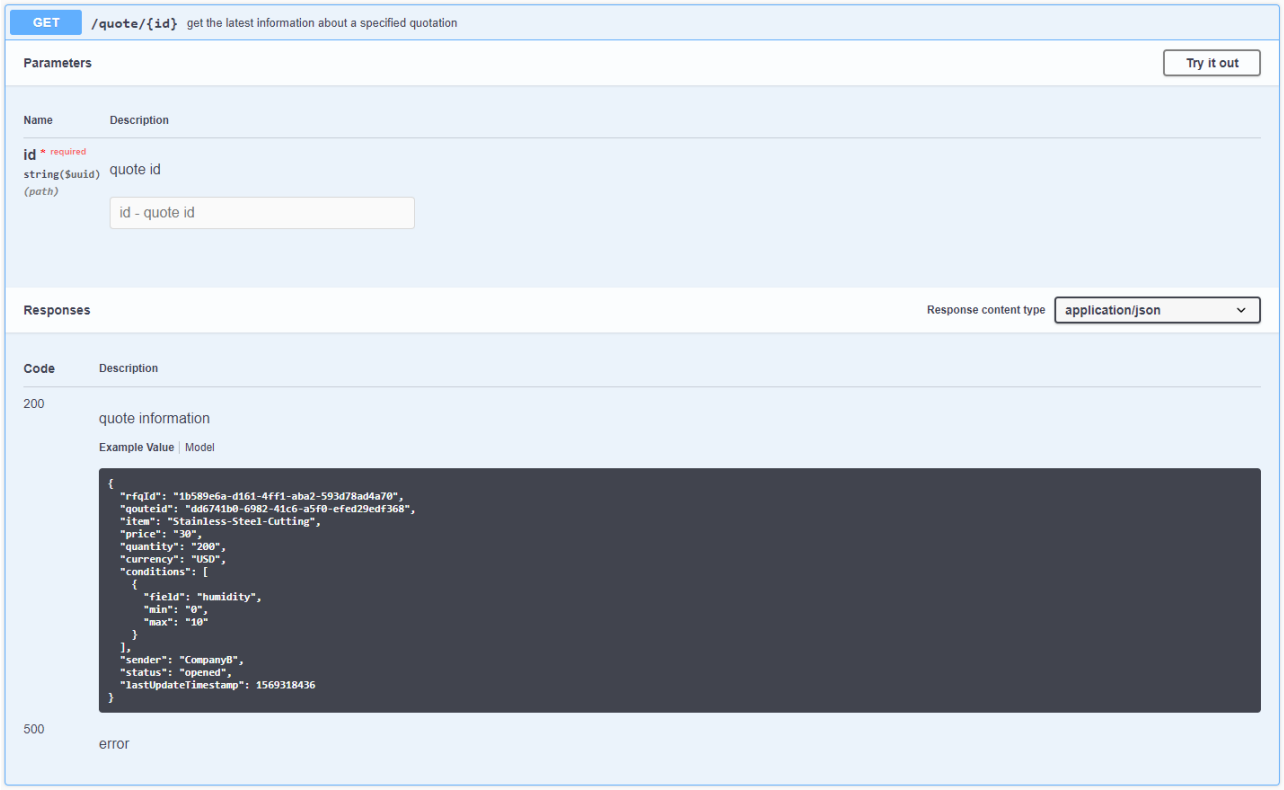


Figure 36: Obtain information on a given quotation

Figure 37 depicts the API call for a responding (accepting or rejecting) to the suggested terms of a quotation.

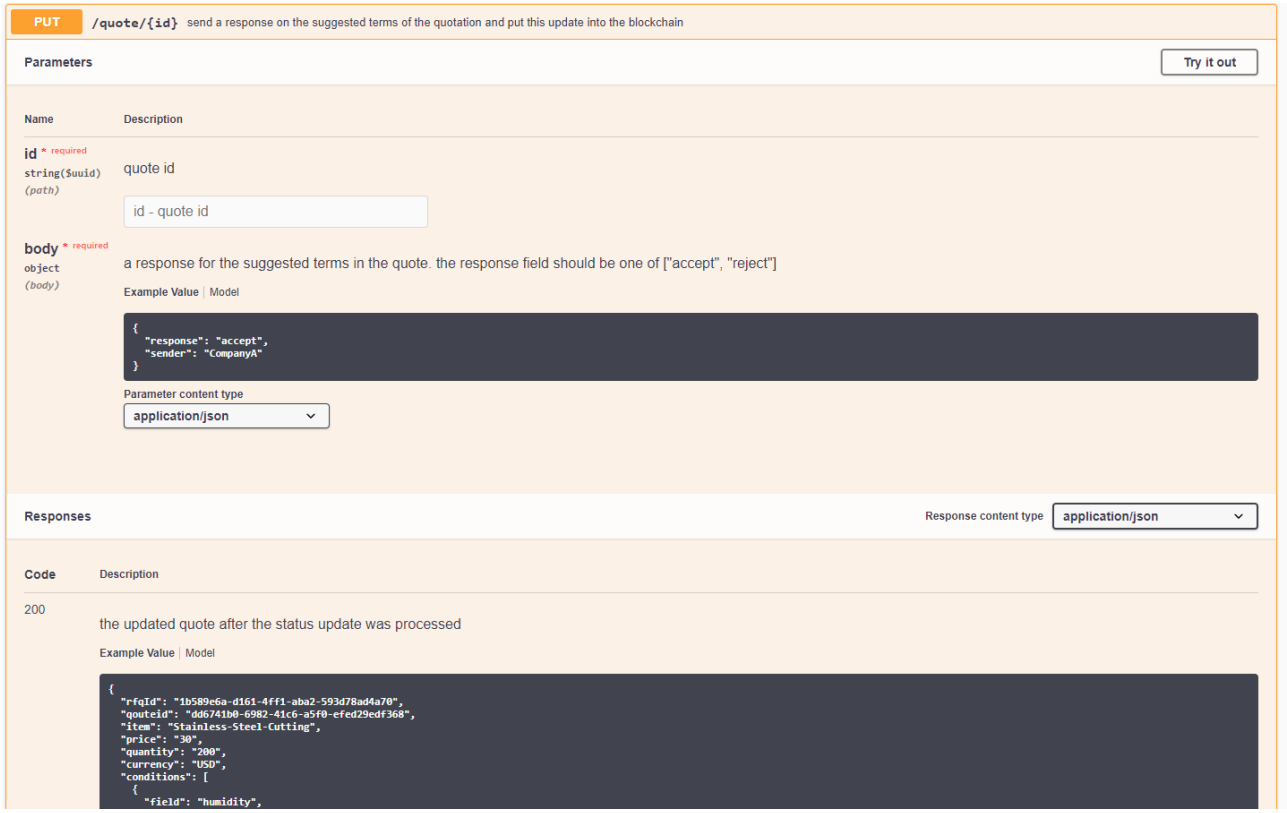


Figure 37: quotation response

Figure 38 depicts the interface exposed to enable negotiation among entities by providing counter offers in which they can negotiate both on the product, its characteristics and the environmental condition of the goods while in production or in transit.

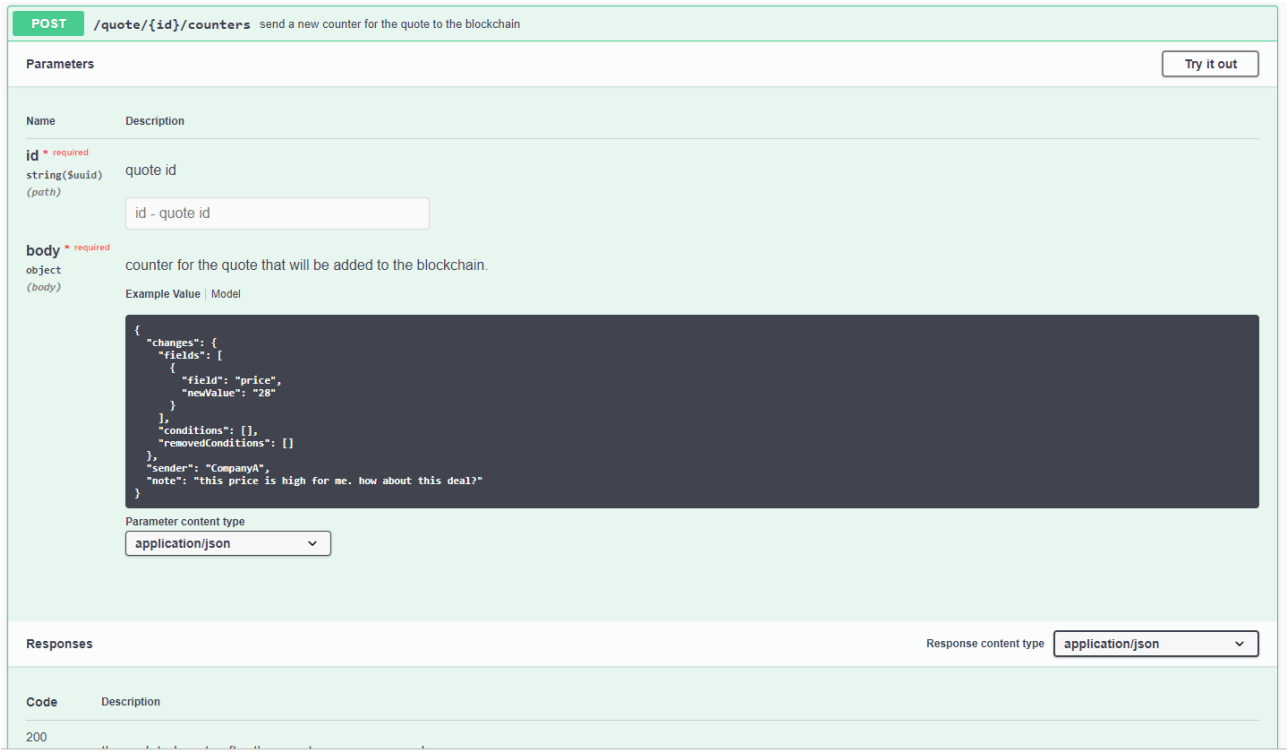


Figure 38: Counter offer

Figure 39 depicts quote history information including all the negotiation that took place between both parties. This interface provides the possibility to follow the path taken by both parties concerning a quotation and its evolution through time.

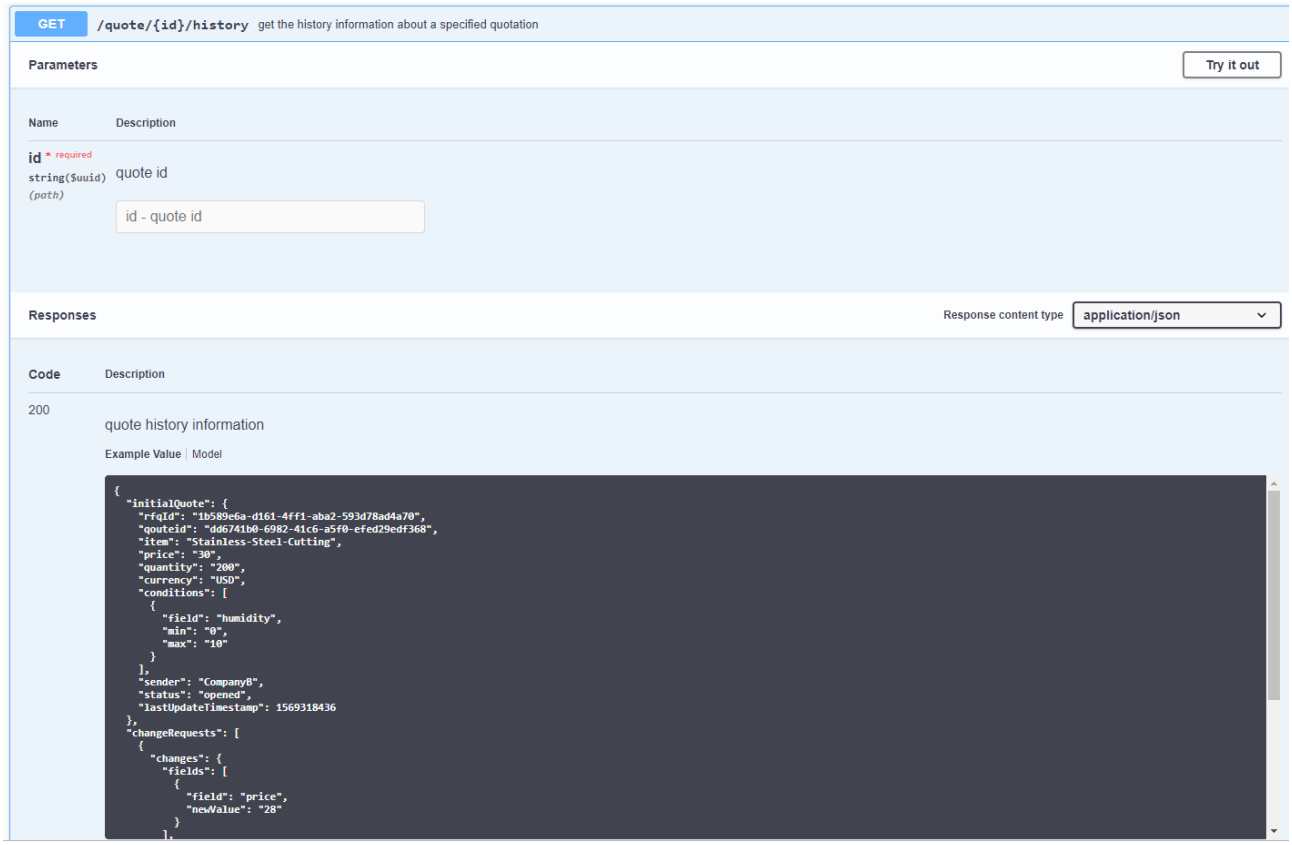


Figure 39: RFQ history

5.3 Reputation Management

Reputation management is an important service within a platform such as MANU-SQUARE, by the virtue of it being able to ease the decision of a company to conduct business processes with another company with which it had no business relations in the past. In addition, it may be interesting for companies to see what other companies are saying about an already known company. Without the use of a platform such as MANU-SQUARE companies could spend a lot of time, effort, and resources to establish business relationship with new companies. One of the changes required by such a digital transformation is to be more agile and open to the establishment of new relationships. The reputation management component comes into play for companies to reduce the risk they take upon themselves by being exposed to the opinion of other entities which have collaborated with the company being considered as a new business partner and take that information into consideration while ranking the potential companies to collaborate with. For companies to be able to trust the information that is shown by the reputation management tool MANU-SQUARE uses a blockchain back-end to ensure full provenance and traceability of the information gathered by the reputation manager. Thus, MANU-SQUARE provides an extra level of trust in an otherwise thrustless environment, by ensuring that information in the reputation manager cannot be altered or deleted in any way, without being detected.

The interfaces exposed mainly operate using a reputation entity structure depicted in Figure 40. The reputation entity is anchored by an entity id and a specific service provided by the entity in question. Thus, support for different reputation entities for different services provided by the same entity can be provided. In addition, there is support for several reputation dimensions, divided into two main categories, namely subjective and objective dimensions (more details can be found in D4.2 Reputation tool). The entity as a whole has an overall reputation score associated to it, and each dimension type has an overall score for that dimension type. All scores are calculated by the reputation management tool and provided to the

blockchain backend. Thus, the blockchain acts as trusted storage and provenance of the scores, but the "logic" determining the manner in which scores are calculated resides in the reputation management tool itself.

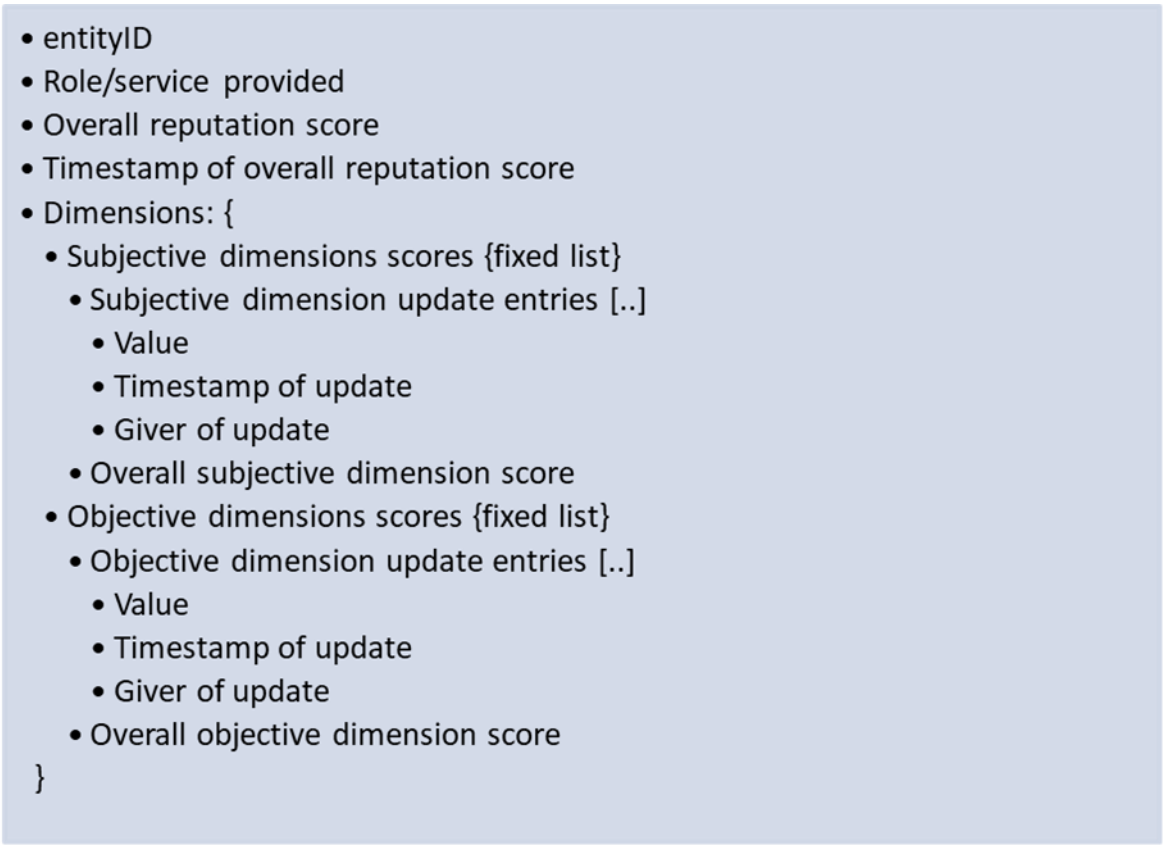


Figure 40: Reputation entity structure/model

To provide the role envisioned for the blockchain based support there are two main types of operations exposed, namely transactions and queries. The transactions supported enable the creation of a new reputation entity and to update an existing reputation entity. The latter updates an existing object with the given key with a new object provided by the caller of the interface. In addition, there are several query flavours supported. The interface supports the extraction of a reputation entity object of a given entity id, or a combination of id and service. In addition, the scores for all dimensions or for a particular dimension can be queried. The reputation history per entityID and service, as well as the overall score for all dimensions, and for a particular dimension can be retrieved. Figure 41 depicts the swagger representation of the blockchain based support for reputation management in MANU-SQUARE.

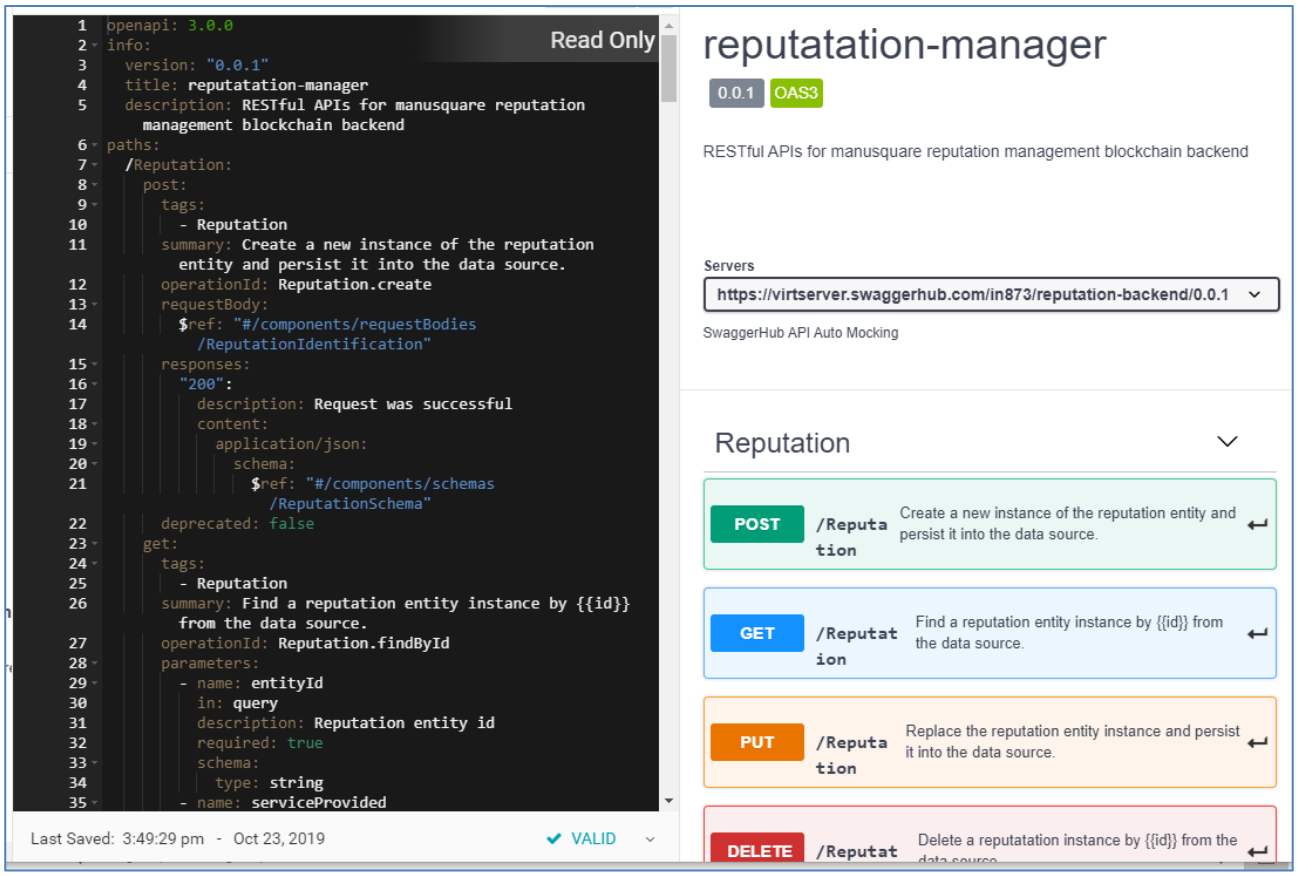


Figure 41: Reputation manager swagger

The interface provided for the addition of a new reputation entity is depicted in Figure 42. Figure 43 shows the interface for obtaining back the reputation object associated with a specific id and service provided by that entity. Figure 44 shows the interface called for updating the reputation entity of a specific entity and a service it provides. Figure 45 depicts the interface to be used to delete a reputation information of a specific instance. A note to keep in mind is that since the underlying mechanism is a blockchain, the result of the deletion is the removal of the entry from the world state, which is a DB hosting the latest version of inserted values, but past transaction information still remains stored and available in the blocks of the shared ledger. Figure 46 shows the manner in which it is possible to query the overall reputation score of a particular entity. Figure 47 shows the interface for querying to obtain the overall score of all the dimensions of a particular entity. Figure 48 shows the interface for obtaining the overall score for a given dimension of the reputation entity.

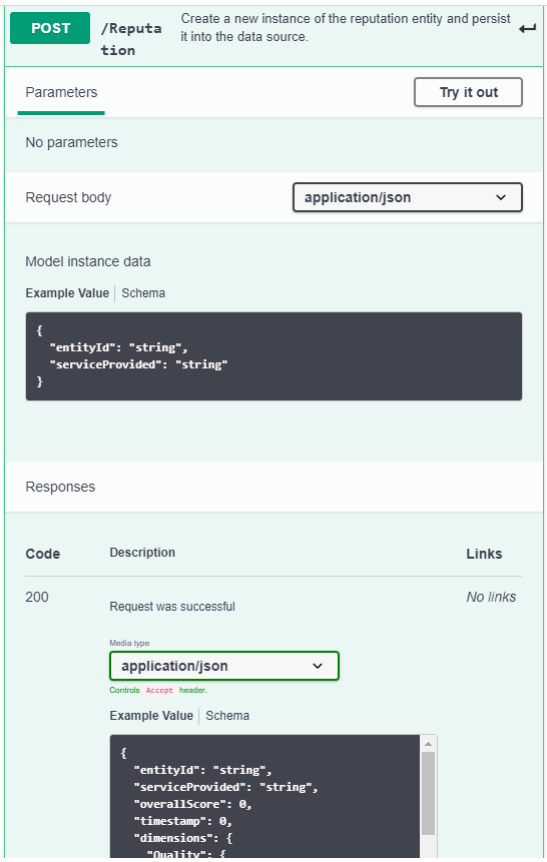


Figure 42: add a new reputation entity

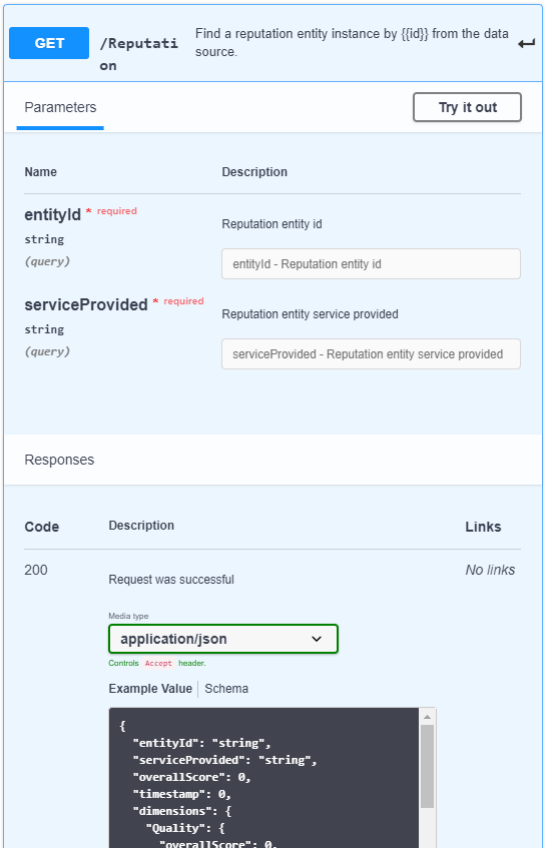


Figure 43: retrieve a reputation entity

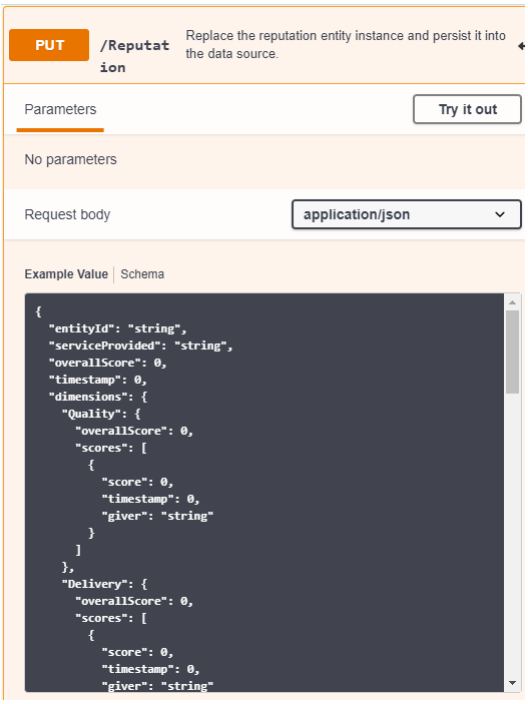


Figure 44: update a reputation entity

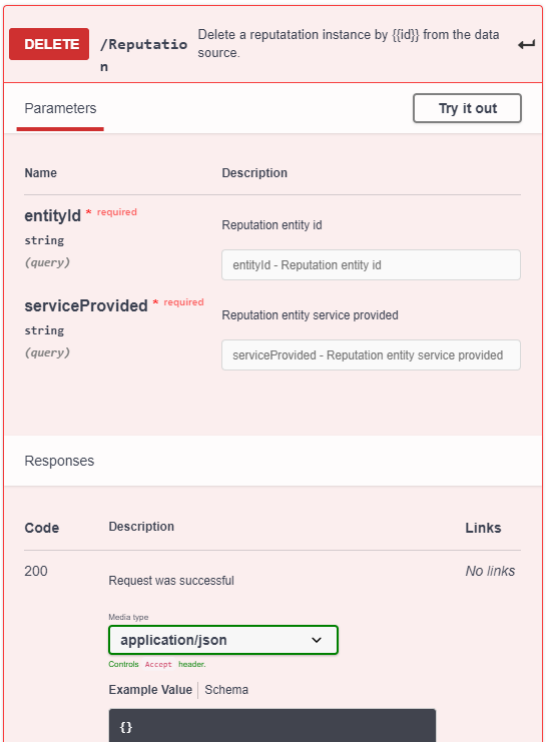


Figure 45: Delete a reputation entity

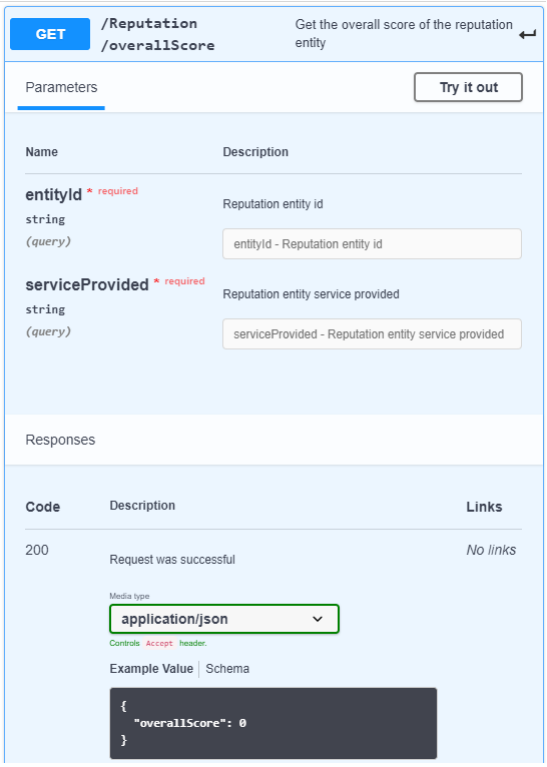


Figure 46: Query for the overall score

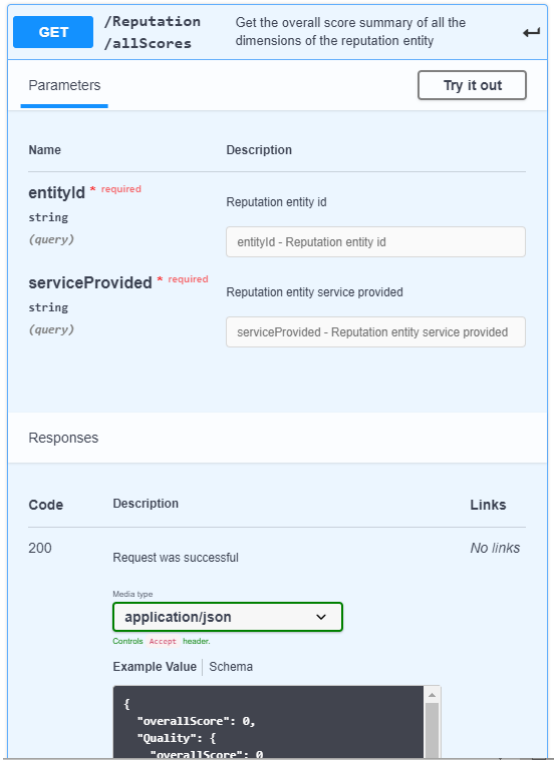


Figure 47: Get overall score summary of all dimensions

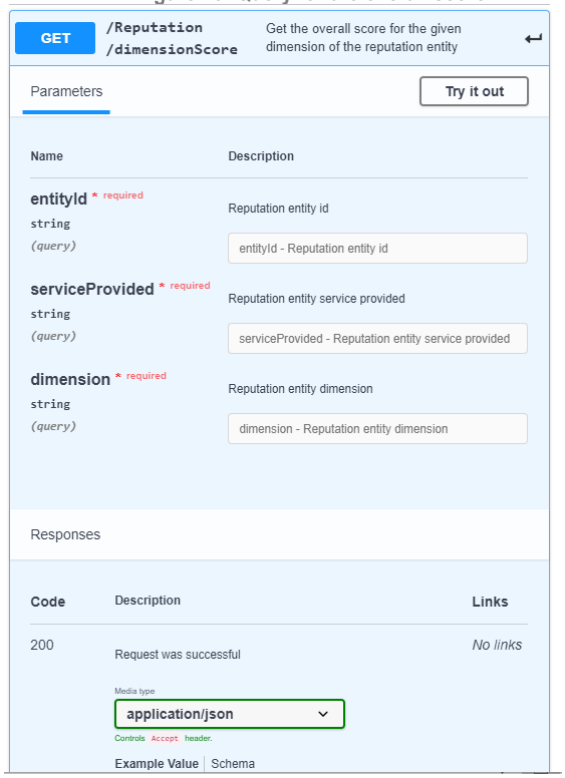


Figure 48: Get the overall score for a given dimension

6 CONCLUSIONS

This document delved into the benefits of using a blockchain network, as the back-end for MANU-SQUARE related scenarios, with specific emphasis on supply chain related constructs that can be generalized. Being a shared ledger, all the invoked transactions and data inserted to the blockchain are clearly stored and can be made available to the parties involved, while ensuring that each entity has access only to the information that is within the scope of visibility of that entity. This document elaborated on the underlying infrastructure which ensures trust in processes in an inherently thrustless environment, by establishing a blockchain based support for supply chain related processes and constructs within a multi-sided platform. Due to the technical characteristics of the blockchain, the shared ledger can serve as the single source of truth in which all parties can access their data and act upon it, without revealing information to non-intended audience.

This deliverable has enhanced D3.3 (Blockchain based supply chain platform – first version), which provided an overall description of the blockchain platform which is used as a cornerstone for the supply chain related activities within the MANU-SQUARE project, and D3.2 (Enhancing trust through blockchain-based security and privacy) which established differential visibility constructs within the extended blockchain network. The primary focus of the current deliverable is to describe and demonstrate, based on the building blocks established in previous deliverables, the manner in which a blockchain can serve as a corner stone for supply chain related business processes and establish additional constructs that can be used in such platforms. It has provided detailed information on the blockchain based support for collaborative ideas generation, which enables people to share and contribute to forming ideas knowing that the underlying blockchain based infrastructure keeps track of the contributions of each individual to the forming idea and can later provide an accurate and indisputable trail of the path that led from the idea inception to its current or final state. That information enables higher layers to determine the relative importance of each contribution to the final idea.

The supply chain constructs described in this deliverable are integrated with the rest of the platform and are part of the deployed MANU-SQUARE software stack. Subsequently, continuous integration with the project use cases shall be pursued. The development of the MANU-SQUARE blockchain component is finished. Some changes and bug fixing are to be expected based on the advances in use case deployment and integration.