Platone

PLATFORM FOR OPERATION
OF DISTRIBUTION NETWORKS

Platone

PLATform for Operation of distribution NEtworks

D2.12

PlatOne Blockchain Customer Access Layer (v2)

The project PLATform for Operation of distribution NEtworks (Platone) receives funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement no 864300.

Project name	Platone
Contractual delivery date:	31/10/2022
Actual delivery date:	28/10/2022
Main responsible:	Ferdinando Bosco, ENG
Work package:	WP2 – Platform Implementation and Data Handling
Security:	P
Nature:	DEM
Version:	V1.0
Total number of pages:	25

Abstract

The Platone Open Framework aims to create an open, flexible, and secure system that enables distribution grid flexibility/congestion management mechanisms, through innovative energy market models involving all the possible actors at many levels (DSOs, TSOs, customers, aggregators). The Platone Framework is an open-source framework based on blockchain technology that enables a secure and shared data management system, allows standard and flexible integration of external solutions (e.g., legacy solutions), and is open to integration of external services through standardized open application program interfaces (APIs).

This document accompanies the second software delivery of the Platone Blockchain Access Layer and reports all the updates respecting to the previous version of the software. It also includes technical specifications and guidelines for installation and deployment of the software.

The Platone Blockchain Access Layer is part of the Platone Open Framework and includes two main components: the Platone Blockchain Access Platform and the Platone Shared Customer Database.

The second version of the Platone Blockchain Access Layer will be integrated, tested and evaluated in the German and Greek demo sites architectures for the second evaluation phase in 2023.

Keyword list

Platone Blockchain Access Layer, Platone Blockchain Access Platform, Platone Shared Customer Database, Data Integration, Data Certification, Data Access Management, Smart Contracts

Executive Summary

*"*Innovation for the customers, innovation for the grid" is the vision of project Platone - Platform for Operation of distribution Networks. Within the H2020 programme "A single, smart European electricity grid", Platone addresses the topic "Flexibility and retail market options for the distribution grid". Modern power grids are moving away from centralised, infrastructure-heavy transmission system operators (TSOs) towards distribution system operators (DSOs) that are flexible and more capable of managing diverse renewable energy sources. DSOs require new ways of managing the increased number of producers, end users and more volatile power distribution systems of the future.

Platone is using blockchain technology to build the Platone Open Framework to meet the needs of modern DSO power systems, including data management. The Platone Open Framework aims to create an open, flexible and secure system that enables distribution grid flexibility/congestion management mechanisms, through innovative energy market models involving all the possible actors at many levels (DSOs, TSOs, customers, aggregators). It is an open-source framework based on blockchain technology that enables a secure and shared data management system, allows standard and flexible integration of external solutions (e.g. legacy solutions), and is open to integration of external services through standardized open application program interfaces (APIs). It is built with existing regulations in mind and will allow small power producers to be easily certified so that they can sell excess energy back to the grid. The Platone Open Framework will also incorporate an open-market system to link with traditional TSOs. The Platone Open Framework will be tested in three European demos and within the Canadian Distributed Energy Management Initiative (DEMI).

The **Platone Blockchain Access Layer** is one of the Key Exploitable Results (KERs) of the Platone Open Framework. It is a blockchain-based layer that includes two different components: the **Platone Blockchain Access Platform,** which allows the integration of the data coming from the physical infrastructure adding a level of security, transparency and trustworthiness thanks to the blockchain technology and smart contracts, and the **Platone Shared Customer Database, which** contains all the energy data (e.g., measurements, set points, etc.), providing access to the data to all the stakeholders involved, implementing data security, data privacy and data access policies mechanisms.

While the first prototype of the Platone Blockchain Access Layer has included all key features and has been successfully integrated into Platone demo trials, the second prototype focuses on the implementation of a complete data access management system, including a web dashboard for the users (data providers and data consumers) as well as the necessary updates in terms of architecture, data modelling and deployment procedures for the integration in the Platone Open Framework and the validation within the Platone demo sites.

More specifically, the Platone Blockchain Access Layer implements now a complete data access management tool for allowing the data providers to manage in an easy way the access to their own data. This tool is strictly connected both with the integration layer (exploiting the MQTT broker functionalities) and the Shared Customer Database, ensuring a high level of security during the data collection and provisioning

Authors and Reviewers

Main responsible		
Partner	**Name**	**E-mail**
ENG		
	Ferdinando Bosco	Ferdinando.bosco@eng.it
Author(s)/contributor(s)		
Partner	**Name**	
ENG		
	Vincenzo Croce	Vincenzo.croce@eng.it
	Angelo Triveri	Angelo.Triveri@eng.it
Reviewer(s)		
Partner	**Name**	
Avacon		
	Benjamin Petters	Benjamin-georg.petters@avacon.de
BAUM		
	Andreas Corusa	a.corusa@baumgroup.de
	Kristin Petersen	k.petersen@baumgroup.de
Approver(s)		
Partner	**Name**	
RWTH		
	Amir Ahmadifar	aahmadifar@eonerc.rwth-aachen.de

Table of Contents

1 Introduction

The project “PLATform for Operation of distribution Networks – Platone” aims to develop an architecture for testing and implementing a data acquisition system based on a two-layer Blockchain approach: an “Access Layer” to connect customers to the Distribution System Operator (DSO) and a “Service Layer” to link customers and DSO to the Flexibility Market environment (Market Place, Aggregators, …). The two layers are linked by a Shared Customer Database, containing all the data certified by Blockchain and made available to all the relevant stakeholders of the two layers. This Platone Open Framework architecture allows a greater stakeholder involvement and enables an efficient and smart network management. The tools used for this purpose will be based on platforms able to receive data from different sources, such as weather forecasting systems or distributed smart devices spread all over the urban area. These platforms, by talking to each other and exchanging data, will allow collecting and elaborating information useful for DSOs, transmission system operators (TSOs), Market, customers and aggregators. In particular, the DSOs will invest in a standard, open, non-discriminatory, blockchain-based, economic dispute settlement infrastructure, to give to both the customers and to the aggregator the possibility to become flexibility market players more easily. This solution will allow the DSO to acquire a new role as a market enabler for end users and a smarter observer of the distribution network. By defining this innovative two-layer architecture, Platone strongly contributes to aims to removing technical and economic barriers to the achievement of a carbon-free society by 2050 [1], creating the ecosystem for new market mechanisms for a rapid roll out among DSOs and for a large involvement of customers in the active management of grids and in the flexibility markets. The Platone platform will be tested in three European demos (Greece, Germany and Italy) and within the Distributed Energy Management Initiative (DEMI) in Canada. The Platone consortium aims to go for a commercial exploitation of the results after the project is finished. Within the H2020 programme “A single, smart European electricity grid” Platone addresses the topic “Flexibility and retail market options for the distribution grid”.

The Platone solution consists of a two-layer blockchain architecture named Platone Open Framework that includes a series of core components, including the Platone Blockchain Access Layer.

The Platone Blockchain Access Layer’s main goal is to enable a standard, secure, and easy integration of energy data coming from the physical infrastructure and grant the access to this data to DSOs and other energy stakeholders.

The Platone Blockchain Access Layer was already tested and successfully integrated in the Platone demo sites, and the feedback collected with the requirements expected conducted to the implementation of the second prototype, fully functional and ready to be integrated in the intermediate version of the Platone Open Framework.

1.1 Task 2.5

This deliverable is related to the Task 2.5 [2] that aims at the implementation of the Platone Blockchain Customer Access Layer.

The Platone Blockchain Customer Access Layer was renamed **Platone Blockchain Access Layer**, to avoid misunderstandings on the word “Customer”, as customers are not the only stakeholders to which the component is addressed. In fact, the Blockchain Access Layer aims to integrate and to manage all the data coming from the physical infrastructure and offers its features to all the energy stakeholders.

The second version of the Platone Blockchain Access Layer follows the functional and non-functional requirements defined in D2.2 [3] as results of the update on Platone Architecture and Platform requirements.

1.2 Objectives of the Work Reported in this Deliverable

The objective of this deliverable is to present the second prototype of the Platone Blockchain Access Layer and its realization following the technical specification and requirements expected. The Platone Description of Action defines this deliverable as a demonstrator. This document accompanies the code repository with a more detailed architecture description as well as some extended deployment instructions for deploying, testing, and integrating the platform.

1.3 Outline of the Deliverable

The second Chapter of this document describes the second realization of the Platone Blockchain Access Laye, according to the specification provided in Deliverable D2.1 [4] and updated in D2.2 [3], and discusses the functionalities implemented in this second version more in detail. Chapter 3 provides a brief overview of Interfaces and Communication Mechanisms. Chapter 4 delivers a compilation of Languages, Technologies and External Tools used throughout the platform. Chapter 5 is closely linked to the software delivery and provides detailed installation, setup and configuration instructions. Finally, Chapter 6 concludes this deliverable.

1.4 How to Read this Document

The document aims to give an overview to the Platone Blockchain Access Layer second prototype release. A description of the foreseen functional and non-functional requirements expected can be found in D2.1 and D2.2. As this document presents an update with respect to the previous version, it is strongly recommended to refer to D2.11 [5] for an exhaustive description of the Platone Blockchain Access Layer. Anyway, for the convenience of the reader, some important information already reported in D2.11 is also reported in this document.

2 Platone Blockchain Access Layer Architecture (v2)

As described in D2.11, the Platone Blockchain Access layer includes two main components:

- **Platone Blockchain Access platform (BAP)**, that implements all the functionalities offered by the blockchain technology through smart contracts and provides an interface for the integration of the data coming from the physical infrastructure.
- **Platone Shared Customer Database (SCD)**: it contains all the measurements, set points and other needed data collected from customer physical infrastructure. It allows the other components and stakeholders of the Platone Open Framework to access data in an easy way and without compromising security and privacy.

It also includes:

- **Integration Layer**, that allows the integration of data coming from the physical infrastructure using standard communication protocols for IoT (e.g., MQTT) and REST services
- **Communication Layer**, that enables the communication among the different internal layers of the BAP, the SCD and other components (e.g., DSO Technical Platform). It provides standard communication mechanisms like REST APIs and Message Broker.

Blockchain infrastructure, that includes a private implementation of Ethereum Blockchain infrastructure including some Ethereum nodes. The Figure 1 below shows the updated version of the BAL architecture.

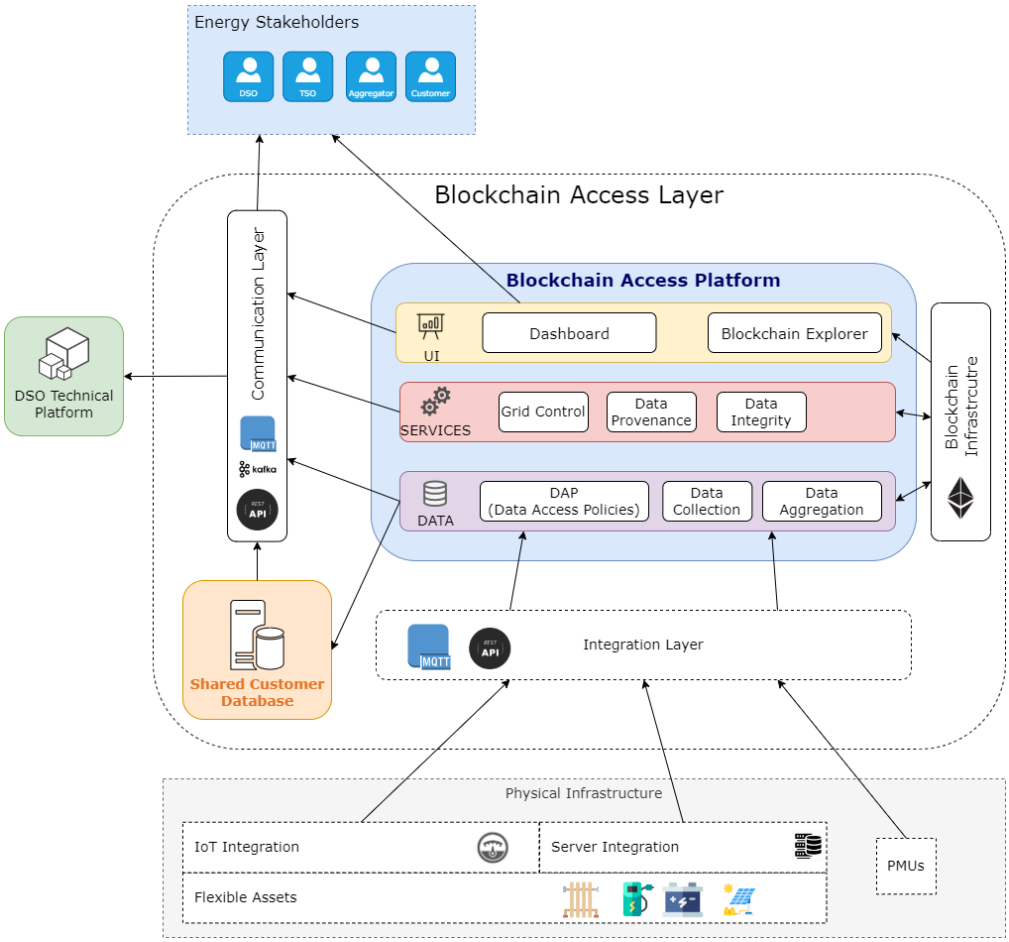

Figure 1: BAL Architecture (v2)

There are no relevant changes in the overall BAL architecture. The only minor change is the introduction of a new component in the UI Layer of the Blockchain Access Platform: BAL Web Dashboard.

More details about the Dashboard are provided in the Chapter 2.1 .

2.1 Functionalities

As already reported in D2.2, all the requirements expected for the BAL were already implemented. For this reason, the second version of the BAL focuses on:

- the implementation of the Data Management Tool (Data Access Policies, DAP)
- the availability of the deployment mechanisms based on Kubernetes [6]
- the consolidation of the features already provided, based on the first validation feedback

2.1.1 Data Management Tool (Data Access Policies, DAP)

The implementation of the Data Management Tool ensures energy stakeholders to handle the data provisioning and consumption in a secure and trusted way, defining specific Data Access Policies (DAP).

The schema below (Figure 2) shows how the data management flow is implemented within the BAL Architecture.

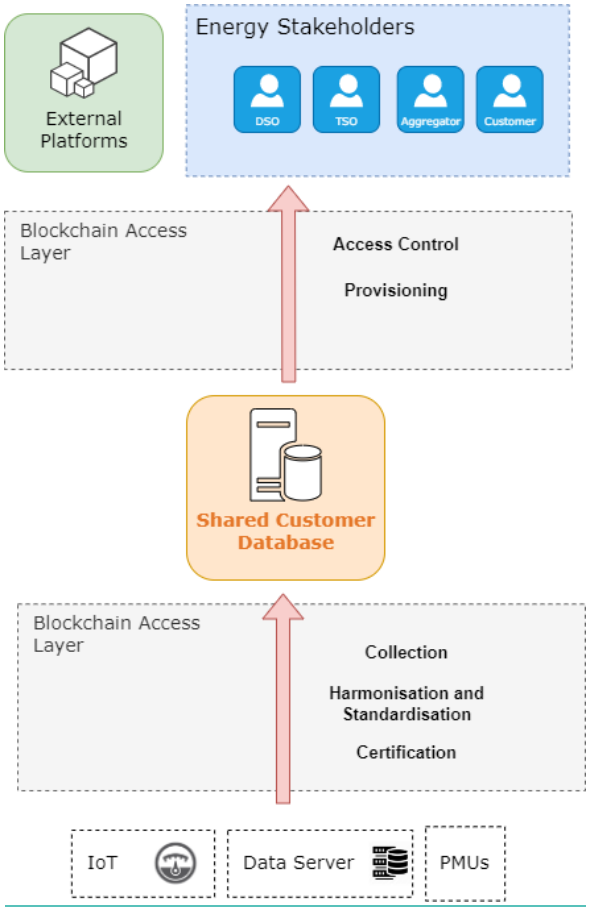

Figure 2: BAL Data Management Workflow

Data are collected in a secure way in the Blockchain Access Layer (through the integration layer), that is in charge of collecting, harmonising and certifying the data. The SCD stores all the data and makes them available through the BAL (and Communication Layer) that includes specific mechanisms for access control and the secure provision of the data to the external actors and platforms.

In order to implement a complete data management tool, methods for authentication and access control are needed.

The BAL implements a low layer access control, exploiting the Mosquitto Dynamic Security Plugin [7] that allows, thought standardized APIs, to interface with the Mosquitto Access Control List (ACL) [8] and define access control rules.

2.1.2 Data Management Tool - Web Dashboard

The BAL Web Dashboard for the Data Management Tool consist of a Graphical User Interface (GUI) that allows to manage the subscription to specific data sources.

The dashboard is accessible by three different types of users (administrator, consumer, and producer) and each of these can take advantage of different services dedicated to them within the platform itself.

The various sections of the application and the features offered are described below.

Login

As shown in Figure 3, Login section allows Administrator, Consumer and Producer users to access the platform.

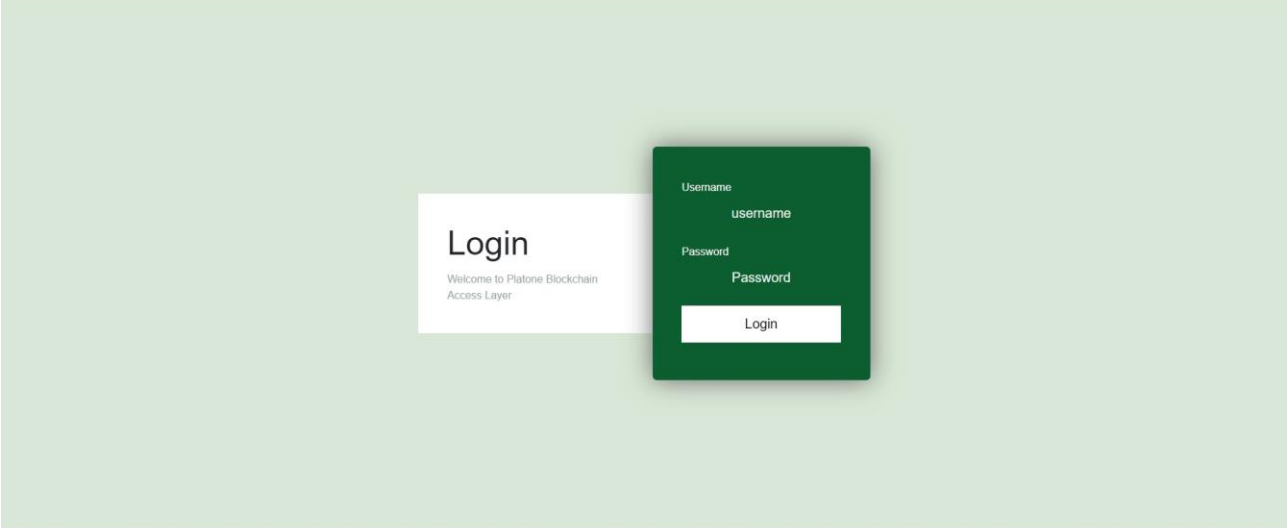

Figure 3: BAL Web Dashboard - Login section

Users are created and initialized, both on the database and on the MQTT broker in order to manage a centralized user database, which includes credentials, roles and permissions. In the case of the Producer user, during the initialization phase a unique topic is created and assigned, together with the related roles and permissions on the MQTT broker.

Dashboard Consumer

Once logged in, the Consumer will be able to view a list of Producers with their associated topic. If the Producers have already been subscribed, The Consumer can view the status of such subscriptions. The Figure 4 shows the Consumer section.

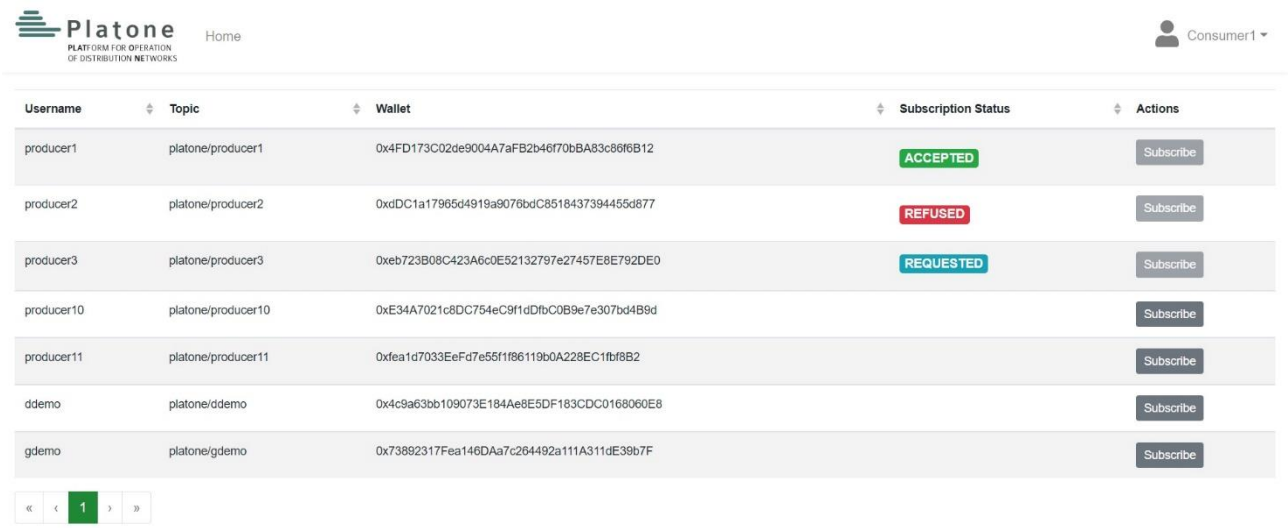

Figure 4: BAL Web Dashboard - Consumer Section

Dashboard Producer

Once logged in, the Producer will be able to view a list of Subscription requests sent by Consumers who ask to be enabled to follow the topic assigned to that specific Producer.

The Producer will have the option to Accept or Reject such request.

Once the subscription is accepted, its roles and permissions are registered both on the database and on the MQTT broker. Figure 5 shows the Producer section.

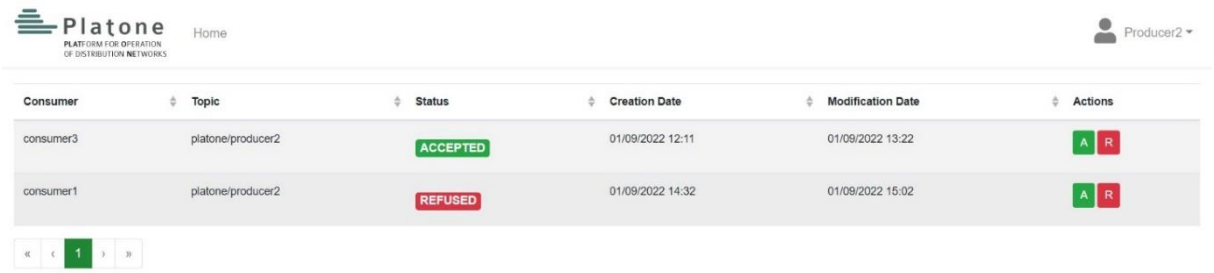

Figure 5: BAL Web Dashboard - Producer Section

2.1.3 Deployment updates - Kubernetes

During the first integration phase within the German demo site, it was necessary to change the BAL deployment environment, for satisfying the needs and requirements of the already existing private cloud infrastructure.

For this reason, another deployment and integration approach was made available, based on Kubernetes [6] that facilitates the deployment and continuous integration of the software.

Kubernetes is open-source orchestration software that provides a set of tools and APIs to manage the orchestration and running of the containers. It allows to run the Docker containers and workloads and helps to tackle some of the operating complexities when moving to scale multiple containers, in particular for deploying across multiple servers. Kubernetes offers a series of benefits for deploying and managing the Docker containers, out of which scalability, flexibility and portability of the solution are the most important ones.

2.2 Data Models

In addition to the data models implemented for the first prototype of the Platone Blockchain Access Layer, two new data models were developed for ensuring the new functionalities of the BAL:

- Users, for modelling the different users able to participate in the data exchange as data providers and/or data consumers.
- Subscriptions, for modelling the request and the status of the subscriptions to specific dataset.

User

Table 1: User model

Field	Type	Description
username	String	**Required, Unique**, the username of the user
password	String	**Required,** the crypted password of the user
role	String	**Required,** the role of the user [admin, producer, consumer]
topic	String	**Unique**, in case of a data producer, the topic for which the user has the permission of publishing
wallet	String	**Required, Unique,** Address of the blockchain wallet
account	Object	**Required,** addition information of the blockchain account

Subscription

Table 2: Subscription model

Field	Type	Description
id_consumer	String	**Required,** the id of the consumer that requests for the subscription
topic	String	**Required,** the topic for which the consumer requests the subscription
status	String	**Required,** the status of the subscription [requested, accepted, refused]
creation_date	Date	the creation date of the subscription request
modificatin_date	Date	the modification date that reports the status change

2.3 Mapping with Platform Requirements

Table 3 reports the list of functional and non-functional requirements expected for the integration of the BAL in the Platone Open Framework and in the demo sites. As already reported in D2.2, all the requirements expected for the BAL were already implemented.

The only requirements not implemented and no longer planned are those related to “network control” (FR-BAP-NC-01, FR-BAP-NC-02, FR-BAP-NC-03). Indeed, after an internal technical evaluation conducted with the German demo partners, was decided to avoid sending the Setpoints through the BAL since is too complex to be implemented during the project phase. The DSO has its own module for controlling the network devices and changing this process it would risk compromising the security and stability of the process itself without an adequate test and verification plan, which would have required too much time and effort outside the project.

Table 3: BAL Requirements

Requirement ID	Requirement name	Requirement description	Use Cases	Status	Notes
P-BAP-01	Communication protocols	The BAP is able to receive data from sensors via MQTT protocol	UC-GR-2 UC-GR-3 UC-GR-4 UC-GR-5 UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	Completed	
FR-BAP-DM-01	Blockchain Data Management	The BAP is able to acquire Measurements from network	UC-GR-1 UC-GR-2 UC-GR-3 UC-GR-4 UC-GR-5 UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	Completed	
FR-BAP-DM-02	Blockchain Data Management	The BAP certifies Measurements via Smart Contracts	UC-GR-1 UC-GR-2 UC-GR-3 UC-GR-4 UC-GR-5 UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	Completed	
FR-BAP-DM-03	Blockchain Data Management	The BAP provides certified measurement in a secure way to DSOTP	UC-GR-1 UC-GR-2	Completed	

			UC-GR-3 UC-GR-4 UC-GR-5 UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4		
FR-BAP-NC-01	Network Control	The BAP is able to receive set points from DSOTP	UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	**Cancelled**	Not Implemented
FR-BAP-NC-02	Network Control	The BAP certifies set points via Smart Contracts	UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	**Cancelled**	Not Implemented
FR-BAP-NC-03	Network Control	The BAP is able to send certified set points to Data Management Backend	UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	**Cancelled**	Not Implemented
P-BAP-02	Communication protocols	The BAP is able to integrate data coming from external server via TCP/IP protocol	UC-GR-1 UC-GR-2 UC-GR-3 UC-GR-4 UC-GR-5 UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	**Completed**	
T-BAP-01	Timing	BAP is able to receive measurement every 10 seconds from sensors	UC-GE-1 UC-GE-2	**Completed**	

			UC-GE-3 UC-GE-4		
T-BAP-02	Timing	BAP is able to receive measurement every 15 minutes from Data Management Backend	UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	**Completed**	
S-BAP-01	Security	All the external interfaces of the BAP must be under TLS connection	UC-GR-2 UC-GR-3 UC-GR-4 UC-GR-5 UC-GE-1 UC-GE-2 UC-GE-3 UC-GE-4	**Completed**	

3 Interfaces and Communication Mechanisms

As described in D2.11, the BAL implements two types of interfaces with different communication mechanisms: MQTT for the data provisioning and consuming, REST APIs for backend services and synchronous requests.

In addition to the first version and in order to validate the data certification within the SCD, a new API was implemented allowing the users to check the validity of a specific transaction within the blockchain and retrieving the certified data. The new API is reported in Table 4.

Similar to the first release, a complete list of the APIs is provided together with the source code in the Open API [9] standard format [10].

Table 4: New API

Name	Url	Method	Parameters	Responses
Check Validation	/api/certification/getByTransaction	GET	In Params: transaction: String – Hash of the transaction saved into the blockchain	Success (200) Certified Data Error (500) Error Message - *String*

4 Languages, Technologies and External Tools

There are no updates respect on the first version of the BAL. For a complete list of used languages, technologies and tools please refer to D2.11.

5 Packaging and Deployment

This chapter reposts a technical step-by-step guideline for the deployment and installation of the software. Specifications in terms of hardware and software are also provided for a correct installation.

Hardware

Operating System: Linux Host

Ram: > 4GB

Disk: > 100GB

Software

Docker > 18.06.1-ce

The steps below report the complete installation of the BAL. Major information can be found in the READ.ME file within the source code repository [10].

Network Creation

```
$ docker network create bap-net
```

SCD Container

```
$ docker run --network bap-net -d --name db -v <your-volume-path>:/data/db -p 27017:27017
mongo:latest
```

Mosquitto Container

```
$ cd app/mosquitto #location of docker-compose.yml
$ docker-compose up -d
```

BAP Container

```
$ cd app #location of DockerFile
$ docker build -t platone-bap:1.0 .
$ docker run --name bap -p 8082:8082 --network bap-net -e DATABASE_URL=db:27017 -e
MQTT_HOST=mqtt://mosquitto_container:1883 -d platone-bap:1.0
$ docker exec -it bap bash
$ npm run deploy-local
$ exit
$ docker restart bap
```

Web Dashboard

```
$ cd client #location of DockerFile
$ export API_URL=<your-api-url> #URL of BAP
$ docker build -t platone-bal-ui:1.0
$ docker run -p 80:80 -p 443:443 -d platone-bal-ui:1.0
```

5.1 Availability

The source code and the Docker Files necessary for the deployment are available in the RWTH GIT repository. ENG also provides a demo version, hosted in its cloud environment located at Pont-Saint-Martin (Italy).

Software REPO

GitLab-> https://git.rwth-aachen.de/acs/public/deliverables/platone/platone-blockchain-access-layer

Demo Version

Web Dashboard-> http://platone.eng.it:8080

API Gateway -> http://platone.eng.it:8082/api

MQTT Broker -> mqtts://platone.eng.it:8883

6 Conclusion

The work done at this stage contributed to the implementation of the second prototype of the Platone Blockchain Access Layer.

Since the first version of the Platone BAL was implemented as fully functional, satisfying all the functional and non-functional requirements expected, the second prototype focused on increasing the functionalities of the platform itself.

This second version includes a Data Management Tool for the definition of data access control on the data provisioning and consuming.

This data access control is implemented in two different layers of the architectural stack of the BAL: at low layer, in the MQTT broker and at high level, within the Blockchain Access Platform.

In addition, a new web dashboard tailored for data provider and consumer was enabled for an easy management of the data access control.

As result of this new implementation, all data collected into the BAL and shared through the Shared Customer Database are strongly secured using the Data Management Tool.

An updated detailed description for installation and configuration of platform components is provided to ensure the usability. A demonstrative version of the entire Platone Blockchain Access Layer is available within ENG cloud infrastructure, too.

Finally, the second release of the BAL will be integrated and tested in the Platone Open Framework and in particular in the physical infrastructure of the German and Greek demo sites.

7 List of Tables

8 List of Figures

9 List of References

[1] European Commission, “2050 long-term strategy”, [Online]. Available: https://ec.europa.eu/clima/policies/strategies/2050_en

[2] Grant Agreement No. 864300 – Platone.

[3] “D2.2 PlatOne Platform requirements and reference architecture (v2)”, Platone EU Horizon 2020 Project, 2022.

[4] “D2.1 PlatOne Platform requirements and reference architecture (v1)”, Platone EU Horizon 2020 Project, 2020.

[5] “D2.11 PlatOne Market Platform (v1)”, Platone EU Horizon 2020 Project, 2020.

[6] «Kubernetes,» [Online]. Available: https://kubernetes.io/.

[7] «Mosquitto - Dynamic Security Plugin,» [Online]. Available: https://mosquitto.org/documentation/dynamic-security/.

[8] «MQTT authentication & authorization, » [Online]. Available: https://www.chirpstack.io/project/guides/mqtt-authentication/.

[9] OpenAPI Initiative, «Open API Specification,» [Online]. Available: https://spec.openapis.org/oas/latest.html.

[10] “Platone GIT Repository,” [Online]. Available: https://git.rwth-aachen.de/acs/public/deliverables/platone/platone-blockchain-access-layer.

10 List of Abbreviations

Abbreviation	Term
ACL	Access Control List
API	Application Programming Interface
BAL	Blockchain Access Layer
BAP	Blockchain Access Platform
DAP	Data Access Policies
DSO	Distribution System Operator
DSOTP	DSO Technical Platform
GUI	Graphical User Interface
KER	Key Exploitable Result
MQTT	Message Queue Telemetry Transport
OS	Operating System
REST	REpresentational State Transfer
SCD	Shared Customer Database
TSO	Transmission System Operator