Platform Requirement Specification (PRS v. 0.0)

Content

PRS in Sweden

PRS in Romania

PRS in Germany

1. General UC description in Sweden

The end-user will fetch information about the sources of energy production, as well as the energy prices and consumption. Therefore, it is essential to use consumption and price forecasts, which should stretch for at least 24 hours into the future, but longer is also desirable. The resolution of the forecasts should be one hour for the upcoming 24 hours and may be sparser further ahead. The forecasts are used to plan the future consumption, based on the end-user's preferences. They are used as inputs of the smart energy platform.

Due to the complexity of energy forecasts (large amount of data and difficult to understand and interpret, particularly for ordinary consumers), it has either be processed by the user's devices, such as an electric vehicle, or if read by the end-users directly, visualised in any easily digestible format.

The desired forecasts are as follows:

- Electricity pricing;
- Energy consumption forecast;
- Carbon footprint based on the expected energy production;
- The energy mix; the means of production divided into rough categories such as: Fossil, Hydro, Nuclear, Wind, Solar…;
- Energy production from the distributed energy resources (DER) at the distribution network, such as solar PV panels and wind turbines.

Access to real-time data

The end-user will fetch close to real-time information about different aspects of the energy production. The information fetched should be up to date, preferably less than 10 minutes old.

The desired real-time information are as follows:

- Electricity price;
- The current production capacity and the current utilisation, as an aggregate and as a break-down with as fine granularity as is feasible;
- Carbon footprint;
- The energy mix;
- Local energy production.

Information use cases

- Requests/recommendations from the energy supplier to the end-user to shift consumption by deferring intensive energy consuming loads to specific time intervals.
- Information about power outages and reduced capacity is sent to the end-user by the energy supplier.
- The end-user sends information about scheduled energy consumption to the supplier to enable better energy resources management. This information may also be sent by the user's smart devices, such as washing machine or the heating system.
- .
- The end-user informs the producers of preferred energy sources/energy production sites.

Direct or coming through energy provider?
We prefer having as few end points as possible and hence we prefer if the information flows through one infrastructure that all energy providers use. If that is not possible, we prefer that the interface across providers are standardized.

Live or delayed?
As close to real-time data as possible.

However, a delay up to 10 minutes is fine.

Granularity (building or larger entity)?
As fine granularity as possible because we want to display data that is relevant to the end-user. Down to single apartment.

The consumption data is required for each appliance of each end-users. End-users refers to the residents of a single apartment.

What particular problems do the actors have?
The active player in this scheme is the energy end-user. We assume that providing them with knowledge can enable them to make informed decisions about what to consume, how much and when to consume.

We assume that the end user is interested in:

- Economic savings, cost reduction
- Environmental Consciousness, lower carbon footprint
- Environmental Consciousness, using local energy sources such as micro grids
- Environmental Consciousness, choosing preferred energy sources
- Increased comfort, enable automation in power consuming devices
- Increased comfort, a higher degree of control/intervention
- Increased comfort, an opportunity to intervene when necessary

The smart energy platform can adapt to different objectives of the end-users, having the possibility of considering more than one objective.

1.1 System Architecture
Our main concern is to have a clean and unified interface for all energy suppliers.

2. Functional Requirements

- Most likely SSL encryption to protect data transfer. User authentication for more advanced functions.
- Integration of external information sources like weather and spot price of energy.
- Aggregate information would be useful to us. Provisioning of data may be interesting.
- An interface to planner (user) with visualisation.
- Data from devices comes on a scale of a minute.

2.1 (De) Registration of Field Devices

2.2 Support for Device Protocols
OpenADR, BacNet, LightweightM2M seems like perfectly OK choices.

2.3 Platform-Internal Communication
Not applicable to our needs.

2.4 Service Interface to Smart Energy Platform Users

2.5 Data Aggregation

2.6 Data Analysis

- Data analysis are supporting measures to reduce energy usage and increase efficiency but it is not the only goal.
- Development of additional services (e.g. factory energy optimisation) based on the available data would be Good.
- Maybe some generic data analysis functionality will be presented in LCP, it is too early to answer what needs we have.
- Orion has internal MongoDB. Database will probably be PostgreSQL.
- Big Data COSMOS GE is known to be slow for analysis of data.

2.7 Data Gathering

Source	Type	Handling
Energy price	Updated at least once per day. Preferable continuous updates with 0 – 10-minute delay.	Preferably through an endpoint provided by the grid operator. REST API with JSON output. Data is pulled from the endpoint.
Energy price forecast	Updated at least once per day. Preferable continuous updates.	As above.
Carbon footprint	Updated at least once per day. Preferable continuous updates with 0 – 10-minute delay.	As above.
Energy mix; a breakdown of current production in categories	Updated at least once per day. Preferable continuous updates with 0 – 10-minute delay.	As above.
Energy consumption (nodal demand and aggregated demand)	Preferable continuous updates with 0 – 10-minute delay.	As above.

Expected generation dispatch; breakdown on production sites, including local production from microgrids	Preferable continuous updates with 0 – 10-minute delay.	As above. It would make things easier if we could interact with microgrids through a centralized mechanism. The second best is to have a unified API for microgrids.
Demand response signals; if the provider intends to offer price-based or incentive-based demand response programs.		
Historical data	Not a strict requirement, but could simplify some applications. Three months of detailed data, one year of more sparse data.	As above.
User's planned consumption feedback	Instant	Preferably through an endpoint provided by the grid operator. REST API with JSON output. Data is pushed from the endpoint. Authenticated communication. Smart devices will provide most of the feedback.
User's energy preferences feedback	Instant	As above, although more through user input and less from smart devices.
Grid operator outage and reduced capacity information	Instant	Push to an endpoint of our design?
Grid operator request/ requirement/ recommendation of power consumption	Instant	As above.

2.8 Decision-making

It is important to consider the possible decisions that the smart energy platform can make. The approach that will be used to achieve this objective is also different. It can be a rule-based or an optimization-based approach. It also can take advantage from the learning algorithms.

3. Non-functional Requirements

3.1 Scalability

- first version of the Cloud Platform Infrastructure support:
    - VMs.
    - Server with 2 CPU, 16 GB memory, 100 GB disk.
    - GB storage with 0 – 1 TB of disk.
    - External IP Addresses.
- SEP Users probably be able to define and configure their own VMs and configure and run their own software systems on said VM.

3.2 Implementation Requirements

It is preferred a free and open attitude to solutions and code. We need a breakdown of the differences between different license models to answer this in detail. We trust you to take a good decision. Assuming that we want to include a device with energy storage capability, such as a robot vacuum cleaner, mobile phone, laptop, etc. There is a relationship between the energy entered to the device and the energy stored in the battery, which relates to the charging efficiency of the battery.

3.3 Data

3.3.1 Data request list

3.3.2 Data sources

Data coming from end user would be short message contains user status/ user preferences. It is desired to involve appliance in the future, which would send data like status of the appliance in the sense of on/off and energy consumption.

Our concept is to provide a clean and secure way to transfer and expose data between the grid operator and the end user. Additionally we will focus on the open information about the grid to the user.

3.4 Data Model

We have no opinion on this.

3.5 Security

- Data such as aggregate energy consumption and price could probably be open. Many of the functions need authenticated/authorised access.
- Authentication would be a per-user level
- Authentication and authorisation would be on a per-resource level as well.
- Access to resources are **an** open class that even external users can access.
- Encryption of the data payload carried by protocols between SEP and the field devices would be implemented

3.6 Storage

3.7 Availability

- SEP be accessible and available at all times (at acceptable service levels).

3.8 Reliability

3.8.1 Maintenance

3.8.2 Maximum Time to Repair

3.9 Usability

- The system provides generic user interface to support functionality.
- This functionality be based on using graphical elements instead of using text-based ones and can be used by individual SEP to provide their particular user interfaces.
- The system shall support access by multiple users at the same time (multi-tenant).
- SEP can decide whether individual services shall be multi-tenant in their instantiation.

3.10 Service Registry

- Services shall offered by SEP be registered to external users.
- Service performance shall be defined (For each service. => Platform shall deliver this performance.) Start with performance of basic services and develop details where and when needed.

3.11 Performance Requirements

- Data from devices come on a scale of Per-minute or offline.
- About 5 – 10 devices per end-user.

3.11.1 Response time

3.11.2 Capacity

- Response time aims for 2 s. As fast as technically feasible.
- The infrastructure have limits on parallel jobs but it should be able to handle many parallel requests.

3.12 Cloud Infrastructure Capacity

3.13 Online User Documentation and Help System Requirements

3.14 Purchased Components

- The entire Cloud Platform open source not a requirement**.**

3.15 Portability

- The system versions be packaged and released and packaging the system to run on PCs is required.

3.16 Interfaces

End-users through web or endpoints, end user's devices through endpoints wold inte5ract with the system. The system administrator will also interact. The interface do receive the preferences of the end-users and the status of the devices.

3.16.1 User Interfaces

- End-user. The end-user will interact with the platform through a set of defined services. The interaction might be through a webpage or a mobile app that is communicating with the platform.
- Devices of the end-user. Devices will interact with the platform through a set of defined services.

3.16.2 Hardware Interfaces

3.16.3 Communications Interfaces

1. General UC description in Romania

Within the Field Test Romania, UPB will carry out monitoring activities of distribution grids in several installations operated by regional DSO and equipped by the energy division of EnergoBit. Here, the aim is to demonstrate the FISMEP platform's ability to integrate information on energy distribution and voltage quality. This information is extracted from real-time streaming via phasor measurement units (PMUs), i.e. a network of special devices that measure both complex amplitude and voltage at multiple locations synchronized in time ("synchrophasor").

The trial sites in Romania consist of the following:

- LV grid node on UPB campus, Bucharest
- LV grid node with PV installation
- 20/0.4 kV substation on LV side
- Point of common coupling (PCC) of wind park, 20 kV and 110 kV

Five mobile PMUs will be installed as configurable monitoring nodes on 110 kV and 20 kV bus bars of substations with intermittent energy input (PV- and wind-based). A novelty in the application of PMU data is the voltage quality monitoring. The acquired results of this field test are pivotal for the implementation of an electric energy market that can operate at different voltage levels (transmission vs. distribution).

Data files (*.csv) contain the following information:

- timestamp (in UTC format),
- frequency [Hz],
- rate of change of frequency (ROCOF),
- phase voltages rms vlues [V],
- phase voltages angles [degrees],
- phase currents rms vaues [A],
- phase currents angles [degrees],
- zero, positive and negative sequence of voltage magnitude (3 rms values, in V),
- zero, positive and negative sequence of voltage angle (3 values, in degrees),
- zero, positive and negative sequence of current magnitude (3 rms values, in V),
- zero, positive and negative sequence of current angle (3 values, in degrees),.

Data coming through the PDC is delayed due to the internal processes of reading archived data and putting them together in one unique *.csv file. Delay is in the order of hundres of ms, or 1s (conservative); delay is measured from the process data acquisition moment (timestamp)

In space: area of 500 km from the PDC to the actual devices; however, communication will come from PDC to Fiware, i.e. 1 point of data distribution

Power can be available in the files in terms of active power [W], reactive power [var], apparent power [VA], power factor, although at this time it is not setup to save this type of data in the archive.

Depends on schema used for PMUinstalation; normally one can now how much energy is transferred on the supervised line, but this is not a useful information

Particular problems: data is not available in a cloud storage yet, and the process of obtaining data for a specific period of time is currently restricted via IP (data files are available for download only if accessed from the PDC local network).

Another problem is the amount of data we need to read related with the communication speed of internet and GPRS network.

1.1 System Architecture

Trial sites in Romania implement two PDCs which will collect data from up to 10 PMUs each. The PDCs use openPDC, which in terms of communication supports the following protocols:
Input protocols:

- IEEE C37.118-2005
- IEEE C37.118-2011
- IEC 61850-90-5
- SEL Fast Messaging
- Macrodyne N and G
- IEEE 1344-1995
- BPA PDC Stream
- UTK FNET
- DNP3
- Gateway Exchange Protocol (GEP)

Output protocols:

- IEEE C37.118-2005
- BPA PDC Stream
- Gateway Exchange Protocol (GEP)
- Inter-Site Data (ISD) – Alstom Grid

Based on the official description of openPDC, it is a software that is able to gather time-series data from a live stream of several PMUs, and sort data by time in order to provide users with the requested timeframe of data, as well as to archive it for further purposes.

In WP3, the application runs on a server which features a utility called Historian View. The utility enables a user to select data timeframe to download in a **.csv* format file, and also browse through the available archives (file type of archives is **.d*). Communication input protocol used between the PMUs and the PDC in field test Romania is C37.118.2005. Also, openPDC Manager is used to setup or list the PMUs connected to the PDC. Usually it is able to display the PMU's Acronym, and also in the Name field it should display the PMU's location. The Acronyms were set as such they contain the producer of the PMU equipment (Arbiter or SEL for example), and the subnet of the network in which the PMU is connected (for example **ARB14** means the device is an *Arbiter* PMU which is located in subnet *192.168.**14**.x* and is connected to the local UPB network via a router with IP *192.168.14.1,* it can be accessed via OpenVPN with certificate name *branch**14***, available on *10.20.30.**14***).

Once connected to a PDC, a PMU will start sending data regardless of the PDC's further internet connection to a cloud storage. Therefore, data can also be lost as PMUs have no support for a *package received confirmation* (such as a computer can benefit from a ping request). In addition to that, control communication in this architecture is unidirectional, from PMUs to PDC and further to the user, with no possibility to send direct configuration requests from a user towards the PDC or any given PMU. The only bidirectional type of communication refers to the user request for a specific timeframe regarding data

There are 2 possible solution to design the system architecture:

Solution A. Communication PMU→PDC→Fiware platform

(detailed above; however, it is of no clear why should you create a *.csv file instead of creating a local database? Why should we use Fiware platform when everything can be solved more easily with a local SQL database on the same server where the PDC is installed [Energobit observation]

Solution B (agreed to be explored by EnergoBit):

Already some activities have been pursued:

- We installed in a lab 2 PMUs (SEL and Arbiter) to study their data acquisition model

- We used Fiware platform to create a virtual machine where we installed a couple necessary applications to write data on the platform and create a database

- We ran some tests to read, write, modfy, delete data in/from the database

Observations regarding our tests:

- After first tests, we cannot say that Fiware platform (cloud version) is the best solution to centralise PMU data as this platform uses Smart City concept, where information is gathered from a series of sensors, and data is updated once in a couple of minutes, once per hour, even once per day. In our case, we wish to read data from the PMUs, data which in order to be relevant and usable, should be updated every 20ms (or even every 10ms), which means that every second we should write 1300 pieces of information from 1 PMU (13000 pieces of information from 10 PMUs). Data writing, reading and deleting in Fiware is done through HTML commands, a rather slow process. From the tests we have run so far, we did not find a way to write "data packages", fastest speed we have obtained was 50 sequences of 26 information/second, which anyway is a bit faster than the internet provides.

- Due to the security of data and the read/write speed limit in the cloud, it is possible that the best solution is to use Fiware platform technologies, but on a local server without sending data to the cloud. (this was also the conclusion when we talked about our project with a FIWARE expert at the conference in Malaga from end of 2017)

- Due to the high amount of data which should be read from the PMUs, it is possible that we will not be able to use GSM modems for data transmission, due to slow transmission speed. Here, we are still studying C37.118 protocol to determine the best way to read data from the PMUs. There is a possibility that using a PDC connected directly to the PMUs through optic fiber or Ethernet network would be the only solution (an application to read centralised data from one PMU installed in each location, in real time, might not be possible).

2. Functional Requirements

If possible, a security authentication key should be implemented so that only someone who knows a PMU's exact location at a given time should have access to data registers from that given time. The integration of external information sources are needed. We also need to aggregate data, provision of data as a service. It is planned to have an interface with visualisation.

PMUs have a report rate of 50 frames/s. Romanian field test will implement two Phasor Data Concentrators (PDC) which will provide data from the five mobile PMUs (solution A) or every PMU talks with Fiware (hundreds of devices with 50 frames/s., solution B)

2.1 (De) Registration of Field Devices

Field devices (PMUs) are installed once in a substation usually and do not support the plug&switch system. At this moment we don't need device registration/deregistration, but still we need to define how to proceed when communication with one device is lost.

2.2 Support for Device Protocols

For reading information from field devices we use C37.118/2011 standard.

2.3 Platform-Internal Communication

- Interwork between functions shall be defined in terms of HTTP protocol
- Particular interfaces of particular functions shall be designable as a supporting real-time communication.
- It shall be possible to send real-time messages on such interfaces and latency.

- A non-real time message-based communication scheme shall be supported between functions.
- Data files should be available for downloading by means of a form input containing the selected PMU from which data was gathered, the desired timeframe (starting time – ending time) and which types of data should the end file contain (e.g.: all, frequency only, voltages only, voltages, currents, frequency and ROCOF). Therefore, implementing a Publish/Subscribe broker (such as MQTT) is not needed.
- There is GPRS and internet limitation.
- Latency from the PDC to the cloud depends on the input choices of the user regarding timeframe and types of electrical measures selected. It can be anywhere from seconds to minutes (longer timeframes take more time).
- …

2.4 Service Interface to Smart Energy Platform Users

- Apps using SEP shall do so through services offered over a REST interface.
- Iinterface's between functions inside of SEP shall be defined as a services offered over a REST interface in which the FIWARE platform uses REST API, so our application (ION Agent) must use the same API in order to send/read data to/from FIWARE platform

2.5 Data Aggregation

Functionality for aggregation is contrary to the functionality of PMUs.

2.6 Data Analysis

Additional services (e.g. factory energy optimisation) would be developed based on the available data for protection, forecast (of energy state), and estimation beside some generic data analysis functionality be present in LCP.

2.7 Data Gathering

Measure of electrical parameters data, which is almost real time.

3.1 Scalability

First version of the Cloud Platform Infrastructure support:

- 0 VMs *(FIWARE should replace the idea of VMs)*
- 1 Servers (specify Server size)
- 36 TB storage *for 1 PDC x 10 PMUs – for 5 years (or 1 PDC x 5 PMUs – for 10 years)*
- 141.85.184.11 external IP Addresses *(not 100% sure yet)*

3.2 Implementation Requirements

SEP development shall be based on the use of available technologies .

3.3 Data

3.3.1 Data request list

Example of data collection can be like:

- Measurements of equipment, not just a construction model, etc.
- Storages real-time
- Dynamic data, then we also need information about how we get the data: protocol, etc.
- Sufficient data basis

- Address data static
- Year of construction (possibly last year of renovation) static
- Energy consumption data
- Monitoring / metering data real-time
- Network data el. Medium and low voltage network (ELE?) static
- Other system data (PV / heat pumps / CHP where available? Capacity? ...) static
- …

3.3.2 Data sources

Users should be able to input the following:

- Select PMU (based on location)
- Choose starting and ending point of the timeframe
- Choose one or more from the measurements available from the given PMU (frequency, ROCOF, voltages, currents etc.)

PMUs can only be configured remotely through tunnelling applications such as openVPN, but cannot be turned on/off. However, it is recommended to manually restart the PMU after changing the settings so that the new configuration (usually saved as a volatile information) is saved permanently.

3.4 Data Model

3.5 Security

No anonymous user should ever have access to cloud-stored data. For example, SEP administrators should be able to access all data in the idea that in case of a problem (e.g. field device malfunction) an administrator could fix it regardless of the type of resource. Users should only have access to resources, which they are explicitly authorised to access.

The user authentication shall be implemented in such a way as to enable single sign-on but we do not need to have encrypted database on the Cloud platform.

3.6 Storage

- Data should be stored as entities with a list of attributes.
- Administrators need to have direct access to database. Other users must have access to the data limited by user settings.
- Database shall support SQLite.
- A time series database needed

Data should be stored in cloud, and not accessible to anyone, but only to a user holding an authentication key in order to prove the user has knowledge of where the respective equipment was at the time he is trying to download the data.

3.7 Availability

- SEP shall be accessible and available at all times (at acceptable service levels).
- SEP shall support SLAs with clear policies and guidelines for maintenance and version management of the platform and policies for version compatibility for APIs between the platform and the application.

3.8 Reliability

3.8.1 Maintenance

We cannot do maintenance without a downtime on PMUs or even on the PDC, unless we consider maintenance a simple reconfiguration without restarting the equipment's . The system shall allow the removal/addition of functionalities/technologies, modules and version upgrades without loss of availability or synchronization .

3.8.2 Maximum Time to Repair

3.9 Usability

The system shall provide generic user interface to support functionality and this functionality be based on using graphical elements instead of using text-based ones. Shall the system support access by multiple users at the same time (multi-tenant). We should have possibility for access by multiple users in the same time

3.10 Service Registry

3.11 Performance Requirements

- Data from up to 10 PMUs is gathered by one PDC and sent towards the cloud platform. Each PMU sends data at a rate of 50 frames/s. Romanian field test implements two PDCs, each gathering data from up to 10 PMUs. (Version A)
- Data from 10 PMUs (version B)

3.11.1 Response time

For WP3, response time varies, and is strong dependant on user-chosen timeframe. At this time, 6 hours is the maximum capability of the timeframe so that the PDC is able to process the request, and a timeframe of 6 hours can take up to 15-20 minutes.

3.11.2 Capacity

3.12 Cloud Infrastructure Capacity

3.13 Online User Documentation and Help System Requirements

Could be a good feature, but it is not mandatory to implement.

3.14 Purchased Components

The entire Cloud Platform shall be open source.

3.15 Portability

3.16 Interfaces

Professional users, including applications (state estimators) would interact with the platform. This interface allows users to download specified timeframe data; allows visualisation (heat map) in real time.

3.16.1 User Interfaces

Maybe some user with limited access is needed too.

3.16.2 Hardware Interfaces

3.16.3 Communications Interfaces

1. General UC description in Germany

The trial test Germany (WP4) consists in performing the automation functionalities of a Medium Voltage DC and hybrid AC-DC grids in the cloud platform.
In particular, the processes that will be virtualized concern the fault management, the SCADA functionalities and the system control.

This trial test is divided into three parts:

1. Scaled MVDC grid, at 380V: communication with a unique Intelligent Electronic Device (IED), connected to four power converters and primary controllers. SCADA functionalities and secondary control to be implemented in the cloud platform.
2. MVDC grid at 5kV: communication with three different IEDs, each connected to a power converter. SCADA functionalities and, eventually, fault location algorithm, to be implemented in the cloud platform.
3. Model of hybrid AC-DC grid: real time simulations as interface to the cloud platform. Fault Location, Isolation and Service Restoration (FLISR) and network topology reconfiguration to be implemented in the cloud platform.

The user will be able to access the status of the grid in real-time, be informed about faults or other emergency conditions, visualize the actual network topology and the power flow information and manage the set points of the system control activities.

Data to be collected:

The necessary data to perform these activities consist of measurements directly collected by transducers (current, voltage) installed in the grid, status of the network devices and all the possible information provided by Intelligent Electronic Device (IED) and data concentrators.

These measurements are directly received from transducers, via IEDs; they must be live data, in order to be processed as soon as possible and, in case of fault condition, start the protection algorithms.

The granularity is essential, in order to identify the status of each device and to determine an accurate situation of the grid. The fault management and topology reconfiguration algorithms need to identify each device present in the electrical grid, in order to send instructions and receive the confirmation of successful implementation/changes of status.

Additional computations will be carried out on the collected data in the cloud. They will be implemented in the specific component of trial test Germany, by accessing the cloud database (eventually, via “Big Data Analysis – Cosmos” FIWARE GE).

Due to protection issues, once particular data are received (as those indicating the tripping of a circuit breaker) a specific algorithm for fault management must be initiated and carried out in the shortest time possible.

Other processes, as visualization and secondary control, own lower priority level and must be suspended once the emergency condition (hence, the fault management algorithm) is activated.

Some signals, as instructions, are sent from the Smart Energy Platform to the specific IED and/or device installed in the network. It is necessary to verify that the instruction has been correctly implemented, before updating the stored data.

1.1 System Architecture

- The architecture of the platform is based on FIWARE Generic Enablers and Domain Specific Enablers, in addition to dedicated components that will carry out the automation algorithms, together with the FISMEP API as connection point with the users.
- The interface between the cloud platform and the measuring devices, or a data management device as IED, is constituted by the metering component.

The measurements are then stored into the cloud database, accessible by specific component developed for this trial test. These data constitutes the input for the specific algorithms for the management of the electrical grid. The output of these algorithms generate specific commands to devices in the grid; these commands must be delivered and their implementation must be verified.

2. Functional Requirements

A security functionality for the connected devices (from which electrical measurements and control signals are received) can be implemented but does not own a primary importance in this trial test.
In fact, the devices are installed in locked switchgears or in the substation control centers that can be accessed only by the owner (electricity distribution companies).
Anyway, a security level for the connected instruments can be useful if we consider the future expansion and new applications for the cloud platform.
The same considerations are applicable to the visualization interface: since the user is the employee of the distribution company; anyway, a basic security system is necessary.

The device registration is necessary. Each device that constitutes the electrical grid is defined and has its specific attributes in the database (e.g. for a circuit breaker: the status, the fault condition, the associated load,…). The addition of a new device would affect the network topology and, hence, all the implemented algorithms.

The measurements that will be provided from transducers and IED are based on specific standards and protocols. For instance, in the 380V grid the measurements will implement the UDP protocol; on the other hand, in the 5kV grid the MMS protocol of IEC-61850 will be used.

The initial stage does not need the collection of external information (e.g. irradiation, wind speed, electricity price, consumption forecast), but they could be added for additional application.

The data collected must be used as input to the automation algorithms and for the visualization purpose to the user (SCADA functionality), a database would be necessary to store and access all these data.

The number of devices providing the measurements depends on the dimension of the grid that will be tested. For the 380V and 5kV grids there will be one and three IEDs, respectively, providing 10 to 20 measurements each. The frequency of the measurements is in the range of few milliseconds.

2.1 (De) Registration of Field Devices

The autonomous registration or de-registration can be an additional feature that would improve the platform quality. Anyway, changes in the medium voltage network are not frequent.

In case of inclusion of this functionality, the new device must be instantiated in the data-base according to a dedicated registration process, in order to become visible to the other devices and to the algorithm.

The protocol of communication between the device and the cloud platform could support autonomous (de-)registration; it will be successively clarified.

2.2 Support for Device Protocols

The protocols to be used for this trial test is related to the measurements and signals that are exchanged between the devices and the Smart Energy Platform.

The IEDs used in the Medium Voltage network communicate according to defined protocols as IEC-61850 (MMS or GOOSE), UDP, DNP3. For this specific trial test, the IEC-61850 will be mainly used and it will be considered as reference.

2.3 Platform-Internal Communication

The internal communication can be based on flags as messages that determines which specific process has to be initiated.

Anyway, particular applications, as protection algorithms, strictly need real-time messages in order to be solve the emergency condition in the shortest time possible.

If using Publish/Subscribe broker mechanism, a very frequent subscribe action must be implemented for these processes.
An important role in this direction can be covered by the GE “Complex Event Processing - Proton”.

Other functionalities (as visualization in the SCADA system) does not require an immediate action and can support non-real time messages.

2.4 Service Interface to Smart Energy Platform Users

2.5 Data Aggregation

In this specific field test, the data are received with high frequency from few devices. Considering also the implementation in a real distribution power system (in which the number of devices is much higher), it is necessary to use a data concentrator. With respect to the FIWARE components, “Big Data Analysis – Cosmos” can be suitable for our applications.

The data have to be accessed from the specific automation algorithms, in order to detect emergency conditions and activate the specific processes.

The same data, filtered and re-arranged, must be visible and accessible from platform user (according to SCADA functionalities) together with outputs and messages from the automation processes.

Anyway, these data are addressed only to the electricity distribution company and cannot be considered as “open data”.

2.6 Data Analysis

A database should be necessary to fulfill the requirements of this trial test. There are no restrictions at this stage about the kind of DB; in any case, the speed of data analysis is an important parameter for some applications.
The database is accessed by automation algorithm, whose data are taken as input.
The main goal of this trial test is to carry out the automation operation of a distribution network in the cloud. The energy efficiency of the distribution grid is a main pillar, and it can be done only with data analysis.

2.7 Data Gathering

Source	Type	Handling
Voltage measurement, from transducer	Live, every few ms	Provided by IED, using IEC-61850 MMS or UDP
Current measurement, from transducer	Live, every few ms	Provided by IED, using IEC-61850 MMS or UDP
System Control signal	Live, every 10 ms (approximately)	Provided by IED or Primary control board
Circuit breaker status (open-close)	Live, binary value	Provided by protection relay (IED) – can be IEC 61850
Circuit breaker tripping condition	Live, binary value	Provided by protection relay (IED) – can be IEC 61850
Load connected/disconnected	Live, every few s	Provided by IED

3. Non-functional Requirements

3.1 Scalability

Openstack is taken as a solution.

3.2 Implementation Requirements

The SEP can be based on existing technologies. In fact, the FINESCE project and its results have to be considered as reference, using the components that have been developed.
Moreover, all the suitable FIWARE components, GEs and DSEs should be included in the cloud platform. For this reason, the SEP should be labelled as “FIWARE-based”.
Some new components of this trial test will be extremely specific and tailored on particular automation functionalities for MV DC and AC-DC grids; hence, they are not suitable for the FIWARE catalogue.

3.3 Data

3.3.1 Data request list

Collected data are mainly constituted by real measurements that are received from transducers installed in the grid:

- Current measurement
- Voltage measurement

In addition, information related to the system control (to perform the secondary level control) are imported. The precise types of data have to be defined; as example:

- Gate signals
- Droop constant
- Correction signals
- Power measurement

Live data regarding the devices that constitute the network (generally referred to IEC-61850 standard) can be:

- Status of the switch (open / close)
- Circuit breaker tripped due to a fault (fault condition)
- Electrical line: operating / out of operation
- Load: connected / disconnected
- Load: power (or current) absorbed
- Position of tap-changers
- Distributed Generation (DG): operating / out of operation
- DG: Power injected

Permanent data are related to the characteristic of the network:

- Parameters of electrical lines
- Parameters of transformers
- Parameters of loads
- Parameters of DG unit
- Parameters of switching devices

Moreover, other data that provide additional information to the network can be imported in SEP, but they are not directly used in the automation algorithms:

- Address (position) of devices
- Year of construction of devices
- Geometrical information of devices
- Software/Firmware version of IEDs
- Maintenance activities list of electrical devices
- Condition of components (alarm message in case of necessary maintenance)
- Buchholz relay for dielectric failures (oil-filled power transformer)

3.3.2 Data sources

For this trial test, there are not necessary data coming from the user. The interface with the user is mainly for visualization purposes.

Functionalities can be included, for which the user can define set-points in the system control (at the higher level).

Commands must be sent to the devices in the network and the implementation of these instructions must be verified: or by checking the live measurements or using dedicated messages of successful operation.
These commands are related to position of switches, position of tap-changers and control signals (current/voltage/gate signals that will impact on the converters operation).

The sources of data are listed in the previous paragraph.

3.4 Data Model

The scope of the Cloud Platform is a domain where different data models exist and are continually changing and developing, a solution for handling these different data models inside one system is needed. Unifying data models is a job for Standards bodies. Harmonising data models (i.e. making them work together) is an approach which can be applied to two or three models taken together, e.g. CIM and IEC 61850's model of a transformer, but will not scale to handling the many models in the Smart Energy domain.

3.5 Security

Cloud security using Open ID and OpenAuth is key because it removes the need to store credential information on the mobile device. This is important because it really is not possible to "firewall" or otherwise isolate a smart mobile device. Open ID and Open Auth both enable a user to authenticate to a web resource provider or service provider without having to compromise their authentication credentials.

The cybersecurity level that will be implemented can be independently set, since it does not affect the operations of the automation algorithms.

All the access to the cloud platform must be authenticated. The SEP administrator can, obviously, access the whole platform. On the other hand, the SEP user can only access the visualization platform (SCADA system); he cannot access the automation algorithms and the database.

As an possible solution we can take :

- KeyRock, Wilma, AuthZForce
- KeyRock + OAuthZ protocol = role-based
- FIWARE Security, OpenADR

3.6 Storage

The data can be stored in a database; the measurements should be time-tagged and stored for a limited period of time; live data (as the status of a switch) can be overwritten in the DB (after being compared with the previous value).

Database can be accessed only by SEP administrator and can support SQL and non-SQL queries.

The amount of data is not defined yet.

3.7 Availability

The SEP shall always be accessible.

Anyway, due to protection issue its activity cannot be interrupted. In case of maintenance, special measures must be implemented in order to reduce the outage time as much as possible.

For security and reliability reasons, the database or other components can be replicated outside the cloud platform.

3.8 Reliability

3.8.1 Maintenance

The implementation of maintenance activities without downtime would be an important feature.

Any change in the platform shall not interrupt or compromise the protection/control of the network. In doing it, the comparison with the previous status of electrical devices must be carried out.

3.8.2 Maximum Time to Repair

Every maintenance activity should be parallelized with special back-up functionalities of manual operations in order to continue the monitoring of the network.

The maintenance should be sub-divided (if possible), in order to reduce the continuous downtime.

The automation algorithms are directly based on the database, hence its downtime will impede the main platform functionalities.

3.9 Usability

A graphic interface is necessary for SEP user, in order to implement SCADA functionalities. Moreover, a presentation interface can be useful for system/VM administrators.

The simultaneous accesses of multiple users is an important feature, in particular in case of large networks.

3.10 Service Registry

The activities and operations of SEP can be registered for external use.

Since there are different functionalities carried out by the cloud platform, the performance can be evaluated independently.

3.11 Performance Requirements

The frequency of incoming data depends on the kind of device.

The measurements from IEDs and transducers come on a scale of few ms. In this field test, the number of devices is between 20 and 30.Anyway, in the real application, this number is much higher.

The live data coming from electrical devices provide information or on a higher time scale or when changes occur. The number of these components is higher, around several tens.

3.11.1 Response time

The response time depends on the different functionalities. The fault isolation and service restoration must be carried out in the shortest time possible, in few seconds.

Other processes, as secondary level control or network reconfiguration, can allow a higher response time to complete the activity.

3.11.2 Capacity

Multiple activities could be carried out in parallel (reducing the single computation time) until an emergency situation (fault condition) occurs. In this case, all the secondary activities must be suspended and the fault management algorithms must be initiated.

To be defined how to determine the fault occurrence: or a message from switching device in the grid, or a comparison of switches status in the DB with the previous value.

3.12 Cloud Infrastructure Capacity

Sufficient servers shall be available so that the limits of the OpenStack infrastructure shall not be a constraint to the system functionality.

3.13 Online User Documentation and Help System Requirements

3.14 Purchased Components

The entire cloud platform will be, most likely, completely open source relying on FIWARE components and algorithms in Python code.

3.15 Portability

3.16 Interfaces

The interface will access the relevant information that need to be shown (coming from DB or as output of automation algorithms) and all the users interact with it.
Moreover, the set points of the system control could be changed and defined by the user.

The interface should be graphical, including plots of desired electrical quantities that can be directly selected from electrical circuit drawing (showing the actual configuration of the network).

3.16.1 User Interfaces

Example of user profiles:

- **SEP user**: the platform user shall interact with the visualization of selected data, as per SCADA functionalities, with the possibilities to define some set-points of system control.

- **SEP administrator:** the administrator shall interact with the whole system through the Linux command prompt, or through the interface provided by the SEP management system.

3.16.2 Hardware Interfaces

3.16.3 Communications Interfaces