FLEXCoop

Democratizing energy markets through the introduction of innovative flexibility-based demand response tools and novel business and market models for energy cooperatives

WP6 – Semantically Enhanced DER Registry and Open Marketplace for Flexibility Sharing

D6.3 –FLEXCoop Prosumer Application - Preliminary Version

Version 1

FLEXCOOP KEY FACTS

Topic:	LCE-01-2016-2017 - Next generation innovative technologies enabling smart grids, storage and energy system integration with increasing share of renewables: distribution network
Type of Action:	Research and Innovation Action
Project start:	01 October 2017
Duration:	36 months from **01.10.2017** to **30.09.2020** (Article 3 GA)
Project Coordinator:	Fraunhofer
Consortium:	13 organizations from nine EU member states

FLEXCOOP CONSORTIUM PARTNERS

Fraunhofer	Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V.
ETRa	ETRA INVESTIGACION Y DESARROLLO SA
HYPERTECH	HYPERTECH (CHAIPERTEK) ANONYMOS VIOMICHANIKI
DTU	DANMARKS TEKNISKE UNIVERSITET
GRINDROP	GRINDROP LIMITED
CIRCE	FUNDACION CIRCE CENTRO DE INVESTIGACION DE RECURSOS Y CONSUMOS ENERGETICOS
KONCAR	KONCAR - INZENJERING ZA ENERGETIKUI TRANSPORT DD
SUITE5	SUITE5 DATA INTELLIGENCE SOLUTIONS Limited
S5	SUITE5 DATA INTELLIGENCE SOLUTIONS Limited
CIMNE	CENTRE INTERNACIONAL DE METODES NUMERICS EN ENGINYERIA
RESCOOP.EU	RESCOOP EU ASBL
SomEnergia	SOM ENERGIA SCCL
ODE	ORGANISATIE VOOR HERNIEUWBARE ENERGIE DECENTRAAL
Escozon	ESCOZON COOPERATIE UA - affiliated or linked to ODE
MERIT	MERIT CONSULTING HOUSE SPRL

EXECUTIVE SUMMARY

The scope of this document is to develop dedicated interfaces and applications for prosumers to enable efficient monitoring of real-time demand data and Demand Response events triggered by aggregators.

The main features and functionalities as specified in the DoA [1]: energy demand monitoring and analytics, financial and economic management of contractual agreements, demand forecasts as extracted from analytics services, notifications of demand response events triggered by aggregators and verification of compliance with contracts, smart controls and automation overriding options, along with a rich analytics dashboard for increasing awareness around energy consumption and demand response (e.g. comparison with similar peers, previous periods analytics, DR convenience and strategy overriding metrics, etc.).

Overall, this document takes into account end user requirements as specified early in the project and the definition of the architecture as reported in WP2. In addition, the results of modelling work in WP3 and the definition of the common information model in WP4 are considered in order to proceed with the definition of the main functionalities and development principles of the application.

By taking into account these high-level requirements, the structure of this document is presented:

- Overview of the architecture and system requirements
- Design of the application following a modular approach
- Development of the application for end users
- User guide documentation for the prosumers of the project in the last section of the document

It is evident, that the analysis follows a typical software development process starting from the requirements definition, the design and specifications of the tool, the development and further testing of the application. We have to point out that this is the 1st version of the application; we may consider it as the almost final version. A 2nd version of the application will be available by M32 incorporating some updated functionalities to meet end users' objectives as defined during the validation process. The updates of the application will be documented in the final version of the deliverable, "D6.7 FLEXCoop Prosumer Application - Final Version".

Table of Contents

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

DB	DATABASE
MVC	Model View Controller
KPI	Key Performance Indicators
D	Deliverable
DR	Demand Response
DER	Distributed Energy Resources
UI	User Interface
CIM	Common Information Model
DPIA	Data Protection Impact Assessment
ORM	Object-Relational Mapping
WP	Work Package
PP	Prosumer Portal (Requirements)
DoW/DoA	Description of Work/Actions

1. INTRODUCTION

The objective of this document is to report the work performed in the associated task 6.3 towards providing the dedicated interfaces and applications for prosumers to enable efficient monitoring of real-time demand data and supporting their participation in Demand Response campaigns triggered by aggregators. Therefore, this is the tool to engage end customers towards their active participation in the different market schemas and business models within the context of the deregulated energy market environment as promoted in the FLEXCoop project.

The main features and functionalities to be supported as defined in the DoA [1]: energy demand monitoring and analytics, financial and economic management of contractual agreements, energy forecasts as extracted from analytics services, notifications of demand response events triggered by aggregators and verification of compliance with contracts, smart controls and automation overriding options, along with a rich analytics dashboard for increasing awareness around energy consumption and demand response (e.g. comparison with similar peers, previous periods analytics, DR convenience and strategy overriding metrics, etc.). By taking into account these high-level requirements we proceed with the design and the development of the prosumer portal app.

Overall, the document takes into account end user requirements and the definition of the architecture as reported in WP2. In addition, the results of modelling work in WP3 and the definition of the common information model in WP4 are main prerequisites in order to proceed with the definition of the main functionalities and features of the FLEXCoop prosumer application.

1.1. Structure of the Document

By taking into account the high-level requirements and objectives for this task, the structure of this document follows:

- Overview of the architecture and system requirements in Chapter 2
- Design of the application following a modular approach in Chapter 3
- Development of the application for end users in Chapter 4
- Content of the application in Chapter 5
- User Guide for the prosumers of the project in Chapter 6

The final outcomes and next steps of the work are reported in the final chapter with the focus to be on the updates to be delivered in 2nd and final version of the deliverable.

2. DESIGNATION OF THE SOFTWARE DEMONSTRATOR

The scope of this section is to present the high-level objectives and the list of requirements as defined for the **Prosumer Portal** application. For this analysis we take into account the list of end user requirements as defined in D2.1 [2] and the business models defined in D2.4 [3] but mainly the definition of system specifications in D2.6 FLEXCoop Framework Architecture including functional, technical and communication Specifications - Preliminary Version & D2.6 FLEXCoop Framework Architecture including functional, technical and communication Specifications [4].

The definition of requirements is based on the early definition of prosumer focused use cases and business models presented in WP2 along with the list of specifications about prosumer portal as defined at the design of the FLEXCoop Framework Architecture. More specifically, the high-level functionalities (expressed in terms of use cases) to be supported by the prosumer portal are presented:

- Prosumers **increase their awareness and knowledge** of their consumption patterns, generation forecasts and demand/storage capacity, allowing them get a personalized assessment of their consumption and to understand their demand flexibility → *Enriched visualization of energy related information as a means of engagement in energy efficiency and other business strategies*
- Prosumers to know the services delivered using their **flexibility** and the **remuneration** for their participation in **balancing and ancillary services markets**, or on the wholesale market as well as the amounts of saved CO2 → Enriched visualization of market related information & participation in open marketplace/contracts etc.
- Prosumers to address the **self-consumption** by leveraging cost-effective storage solutions. End users to get insights about their participation in **self-consumption** schemas → *Enriched visualization of self-consumption related information (focus on collaborative self-consumption schemas rather on individual self-consumption)*

Special remark about a key objective of the project: to establish a plug and play framework for enabling end user's participation in different contractual agreements with the minimum of intrusiveness. Therefore, the design of the application should also take into account this specific business requirement.

The list of Prosumer Portal related requirements is summarized in Table 1.We first present the list of functional requirements with a high interest about the information to be available for further processing or visualization. In addition, the non-functional requirements are reported as a wish list from the pilot users.

Req ID	Requirement Description	Type	Priority
PP01	End-users to get updates about environmental conditions (temperature, humidity, health conditions) in a personalized way.	Functional	High
PP02	End-users to get access on energy consumption information (and generation if available) in a personalized way	Functional	High
PP03	The end users get updates about energy cost taking into account the retailer tariff schemas applicable per pilot site	Functional	High

PP04	Simple analytics over historical data (consumption, cost etc..) will be provided by the App	Functional	High
PP05	Low resolution data is the requirement for visualization (hourly for real time, daily for historical data)	Functional	Medium
PP06	End-users to set updates on preferences about the environmental conditions, comfort levels	Functional	Medium
PP07	The end users will be able to configure the Prosumer Portal parameters (e.g. DER parameters, DR participation etc…)	Functional	Medium
PP08	The application should support day ahead production/consumption visualization; taking into account the availability of forecasting data from analytics services of the project	Functional	High
PP09	The application should implement technologies to allow customer energy awareness in order to enhance efficient behaviour (e.g. comparison with similar peers, previous periods analytics)	Functional	Medium
PP10	Advices and recommendations should be triggered to the end users; focused on the business domains examined in the project (energy efficiency, self-consumption)	Functional	Medium
PP11	Notifications about triggers of auto Demand Response Strategies should be available via the prosumer portal	Functional	High
PP12	The end users should get informed about performance and incentives (during DR participation)	Functional	High
PP13	Additional indicators about DR convenience and strategy overriding metrics as calculated by Aggregator app will be available for end users	Functional	High
PP14	The application will support communication with Aggregators; aggregators to provide day to day updates about DR campaigns, energy costs, DR rewards, DR compliance etc..	Technical	High
PP15	Bidirectional communication prosumer-aggregator will be established towards setting contractual agreements through an open marketplace	Technical	High
PP16	Bidirectional communication with Aggregators and participation in marketplace will be ensured via a central middleware and repository; responsible for information management and message routing among system components	Technical	High
PP17	End users should be able to get contract offers from Aggregators through Marketplace, taking into account the availability/registry of controllable DERs in the market	Functional	High
PP18	End users should be able to accept/reject a contract offer from the Aggregator	Functional	High
PP19	Once a contract is accepted by the user, the contract is automatically established with the Aggregator; the DER is not further available for renegotiation	Functional	High
PP20	Once a contract is rejected by the user, the Aggregator may provide an additional offer to the end users through the open marketplace platform	Functional	High

PP21	End users should get informed about the list of active DR contracts and contractual parameters in a personalized way	Functional	High
PP22	The application should retrieve static (registration related) parameters for the asset and the DERs through DER Registry	Functional	High
PP24	The app should be as user-friendly as possible	Look and feel	High
PP25	The app and the features supported should be as simple & intuitive as possible for non-experts	Look and feel	High
PP26	Data must be presented in an accessible, understandable and flexible format that enables users to take action	Look and feel	High
PP27	Data should be consistent and reliable	Reliability	High
PP28	The prosumer portal app will only provide access to authenticated users	Privacy	High
PP29	A fully anonymized application should be available for the users; access granted through the association with the unique ID/Property Credentials with no need to provide personal data	Privacy	High
PP30	The Prosumer Portal interfaces should be based in standards to guarantee interoperability (a core aspect of FLEXCoop)	Interoperability	High
PP31	Users will be able to access the Prosumer Portal App from a browser	Accessibility	High
PP32	The web application needs to be designed in a responsive to ensure access from different devices	Accessibility	High
PP33	The structure of the app should address the project requirement for development scalable applications	Scalability	Low

Table 1: List of Prosumer Portal Requirements

We presented above the list of requirements following the typical segmentation on functional / non-functional requirements and further alignment of functional requirements with the high-level overview of use cases:

- PP01 – PP07: Enriched visualization at individual /collaborative level
- PP08 – PP10: End users' engagement in self-consumption schemas
- PP11 – PP22: Contractual Agreements establishment and participation in DR campaigns

Following the definition of system requirements in the previous table, we report the detailed list of functional cases/ functionalities to be supported by the app in the 1st version. This is a task performed in cooperation with the pilot partners, to express in a narrative manner the different functionalities/features to be supported by the app.

Registration and User Settings management

This feature describes the preliminary actions required before a user start using the app. A registration to the platform following authorization is required. The requirements is the process to be as simple as possible (performed in an automated way) and to ensure the full anonymity of the end users; no need to provide personal data to the platform.

The user opens the URL Browser and moves through registration process. In this process, a

---

unique ID is available associated with the GW device to be installed in premises and specified in the project. Then, the user specifies configuration parameters, namely: pilot site, username and a secret password. In case username already exists, the user is prompted to select another username. The user also accepts the terms of usage (mandatory) of the app which once accepted may get access to his personal UI page.

Once entering the app, the profile (Settings) section provides some typical settings parameters and some extra information about specific market parameters: availability of generation/controllable/storage devices in DR campaigns, DER settings for market participation (availability for the market) etc... In addition, the user is prompted to provide additional information about comfort preference settings etc...

Enriched visualization of energy & business-related information

This feature describes the main function of the app about visualization of energy and business-related information.

Starting with the main view of the system, the end users should have access to near real time information about overall asset performance (energy consumption, energy cost, generation data etc...).

Along with real time information, the users will be able to get insights about the history for specific KPIs that are of interest for the business stakeholders. Specific time frames are specified for data visualization (day, month, year)

Time-series analysis is of interest for the end users with the special focus on information about self-consumption. Nevertheless, this is a feature to be supported in the 2nd version of the app.

Visualization of information in a comparative manner

As the focus of the project is at providing a tool to promote end users' participation in the FLEXCoop activities, a dedicated view of the system should focus on presenting information about cumulative performance in a way to engage end users in the project activities.

More specifically, it is of interest for the end users to get a quick overview of their performance by comparing the actual status with typical profiles. Therefore, a comparison with past/typical performance is supported by the app. Moreover, quick insights about user performance are extracted following comparison with similar peers. That way we promote the engagement of end users as indispensable part of the cooperative community.

Once again, the analysis is performed for specific KPIs of interest with the main focus to be about total energy consumption. The user may also select a time period (day, month, year) for this comparative view.

One of the main objectives of the project is to establish a framework for active participation of end users in energy markets (DR markets or other market types). Therefore, special focus about the detailed description of the associated features:

Participation in the markets and contractual agreements

This is the work flow about signing a contractual agreement; part of the definition of a framework for active participation of consumers/ prosumers in the energy open marketplace.

Once appropriately configuring the DER parameters in the registry, the end user makes these DERs available on the marketplace. Thus, the aggregator is able to select from the pool of DERs, the ones that cover its business needs.

The selection of DERs is associated with an offer triggered to the respective end user for signing a contractual agreement. Therefore, a pop-up notification should be available to the end users once a market offered is published by the Aggregator.

This contractual agreement is reported to the end customer through a dedicated view where the details of the offer are specified (data structure of the contractual agreement is reported in D2.4). The user is able to accept, negotiate or decline the offer.

Once the market offer is accepted by consumer, then the contractual agreement is established as a valid agreement and the associated contract (with the specific contractual parameters) is allocated to the list of active contracts (description follows) for evaluation.

Monitoring of contractual parameters and performance

This is the work flow following the establishment of a contractual agreement.

Through this view, the user is able to get an overview of the list of active contracts and take a detailed analysis of each contractual agreement parameters/settings.

In addition, the end user is able to get insights about her/his participation in the specific contract. Information about DR performance, DR remuneration/compensation, DR convenience, DR *rewards or penalties, self-consumption performance… will be available through the dashboard.*

Notification Service for DR participation

A notification service is established to inform end users about DR events to happen. Considering that the FLEXCoop framework is defined as fully automated framework, the role of the notifications is to provide an a-priori information about the implementation of control actions at the controllable devices in premises.

The analysis performed in terms of notifications is twofold:

- *Marketplace related notifications associated with the definition of the contractual agreements and further information exchange among the different business entities*
- *DR event notifications are associated with the insights about DR performance as reported in previous work flow.*

Access on data (history notifications) should be enabled in order to give access to the users over their history.

As stated above, a core part of the FLEXCoop framework is the definition of the contractual schema and the business models to set the basis for the establishment of the marketplace of the project. Subsequently, it is of high importance to specify the model parameters that clearly define the contractual template.

This is work performed in WP2, though a summary of the contractual templates' definition is presented in this section. Towards the definition of the contractual and business models as defined in the project, we have to take into account:

- *National regulators and TSOs should oversee the creation of streamlined, simple **contractual relationships** between incumbent players and the aggregator*

- *Establishment of **appropriate and fair measurement** and **communication** protocols*

- *Ensuring that **Demand Response services are compensated** at the **full market value** of the service provided*

A state-of-the-art analysis of regulation in E.U. and of the available market schemas is presented in WP2 (D2.4). It is clear that there is not a holistic model to handle the contractual agreements between aggregators and prosumers; thus we have to take into account the different principles of DR market operation, towards contractual templates definition:

- Definition of **business models** to be examined in the project. Specific business models (Aggregator as FSP for DSOs, BRPs, etc..) with the associated model parameters are considered (per pilot site).
- Definition of the **operation framework** for the role of Aggregator: e.g. Aggregator standard model, virtual transfer points model, Flex-BR model → *this decision does not affect the contractual template but the remuneration process*
- The DR Aggregator **business perspective**: Combined Aggregator-Supplier, Aggregator-BRP etc…→ *this decision does not affect the contractual template but the remuneration process*
- Definition of the **MV framework for DR services** → associated with the operational role of aggregator and the selection between drop-by and drop-to services → *this decision does not affect the contractual template but the remuneration process*

- The DR Aggregator **device characteristics**: Aggregator holistic management, thermostats, EVs, etc… and whether we set specific parameters for loads participation in DR schemas

Overall, the template definition process should take into account the project specific requirements considering the:

- Definition of **pilot specific business models** as presented above
- Evaluation of **different types of controllable loads** in the project

In D2.4, we blueprint the different business models with the associated model parameters to be examined in the project. Also, in WP6, we specify the specific DER model parameters as reported to the DER Registry in order to support the active participation of DERs in energy markets (via the FLEXCoop Marketplace). By taking into account the aforementioned specifications, the model for contracts management is presented in the following figure:
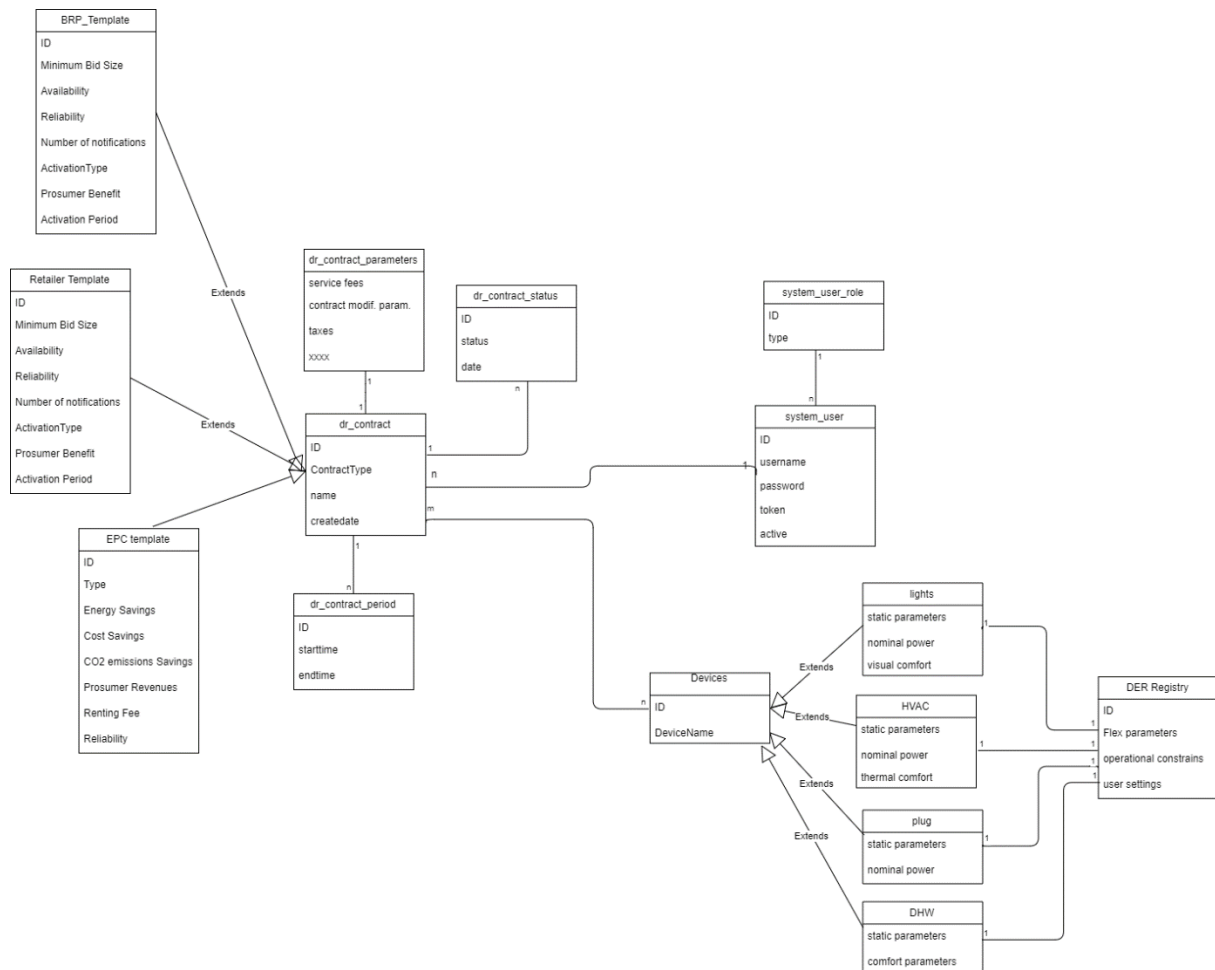

Figure 1: Contractual Parameters management- Data Model

Still further clarifications are required for the specific data attributes that consist of the contractual model as presented above. Nevertheless, the current data schema is providing the

skeleton for the contractual parameters modeling in FLEXCoop project and based on this the 1st version of the contractual management process is supported by the App.

The functional analysis as described above is in line with the definition of use cases and requirements and presents the hierarchy of the features to be considered at the development of Prosumer Portal App. By taking into account the aforementioned requirements analysis, the developments specs of the application are presented in the following section. First, some initial summary of the development specs are provided.

3. VERSIONING/VERSION OF THE SOFTWARE DEMONSTRATOR

Preliminary Release Version 0.5. The final version will be available at the 2nd version of the deliverable

4. RELEASE DATE

27.09.2019 as the 1st version of the app presented in the following.

5. RELEVANT LICENCES USED IN THE DEMONSTRATOR

Presented in section 7/

6. OVERVIEW OF THE PROTOTYPE/ABSTRACT

The scope of this section is to provide the design specifications of the Prosumer Portal App. This analysis takes into account the definition of technical use cases and user requirements as presented in previous section.

The aim is to develop dedicated interfaces for prosumers to enable efficient monitoring of real-time demand data and Demand Response events triggered by aggregators. The main features and functionalities as defined in previous section, specified in the DoA [1]: energy demand monitoring and analytics, financial and economic management of contractual agreements, demand forecasts presentation, notifications of demand response events triggered by aggregators and verification of compliance with contracts, smart controls and automation overriding options, along with a rich analytics dashboard for increasing awareness around energy consumption and demand response (e.g. comparison with similar peers, previous periods analytics, DR convenience etc.).

Taking into account the list of core principles and requirements, we proceed with the design of the Prosumer Portal App.

6.1. Prosumer Portal Architecture Principles

As a typical software application, the Prosumer Portal is designed as a 3-tier architecture with the data management, application & visualization to set the different layers of the application. In addition, the software architecture is based on the MVC principles, commonly used for developing user interfaces that clearly separates the view part from the modeling part, namely:

- The model is the central component of the pattern. It directly manages the data, logic and rules of the application; incorporating the data management & analytics services developed in the project.
- The view part is the output representation of information; the front end/visualization of the application.
- The third part or section, the controller, accepts input and converts it to commands for the model or view. The controller is responsible for responding to the user input and perform interactions on the data model objects. The controller receives the input, optionally validates the input and then passes the input to the model.

The reference architecture of the Prosumer Portal app is presented in the following figure.
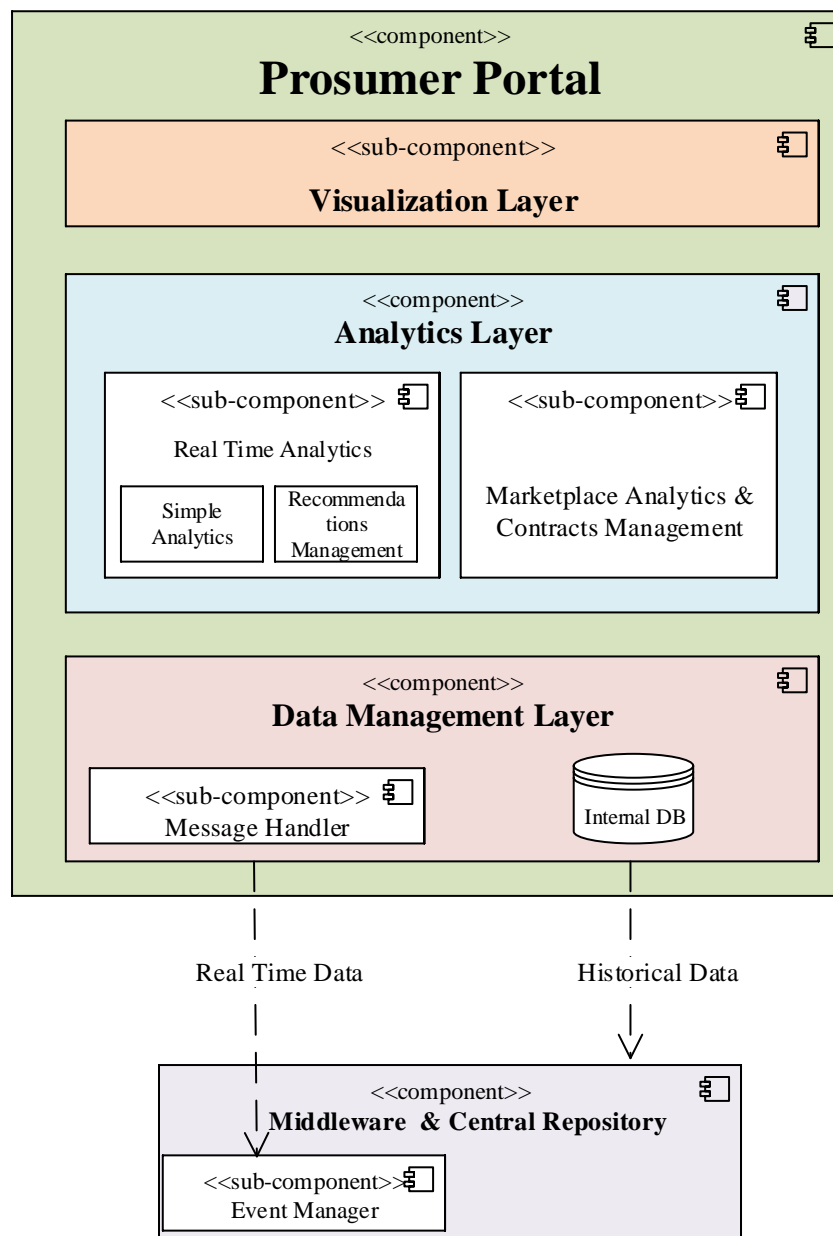
Figure 2: Prosumer Portal App Internal Architecture

As defined in the DoA, *the application is meant to be a tool that is offered by the Cooperative/Aggregator actors to the customers/ members so that the latter can have enhanced visibility on their energy-related activities, including energy consumption, generation, cost, etc.*

It is thus evident that the application is not defined as a standalone module, rather is encapsulates information available at the central repository. That said, it has been defined in WP2 – Reference Architecture that all dynamic information that will be visualized by the Prosumer Portal app will be retrieved from the Data Middleware which acts as the central data management entity of the project. This decision has important positive repercussions on security and privacy as it makes much easier for the FLEXCoop platform to handle any request from a user to provide or even delete any piece of information (owned by the user) that is stored within the system.

Another key point is that the Prosumer Portal is a web app and thus should be accessible by the users through different communication means ensuring also the responsiveness and the usability of the application (one of the major requirements was about a responsive design, so that the interface renders well on any device).

Following the high-level overview of the Prosumer Portal architecture, we proceed with the detailed specification of the different layers of the architecture. Considering that a core functionality of the Prosumer Portal app is the visualization of information towards the engagement of end users, we start the analysis with the design principles of the presentation layer.

6.2. Visualization Layer

This is the front-end layer of the application, responsible for the visualization of enriched information. We present here the high-level principles and guidelines for the design of the application (taking into account the workflows and requirements as presented in previous section):

- The user interface should be simple, visually appealing and easily understandable: the typical user will not be familiar with the technologies and thus the focus should be on simple KPIs and on the domains of interest for a cooperative member e.g. **energy consumption, energy cost & CO2 intensity.**
- The business scope of the cooperatives is to establish a collaborative framework among the users of the portfolio. Therefore, the focus is not only about individual personalized information; rather at the visualization of indicators & metrics related to the collective performance of the cooperative members, e.g. **self-consumption related KPIs** or **CO2 intensity** indicator
- The **minimum of intrusiveness** of end users is a main prerequisite. The users are prompted to use the application on demand. On the other hand, notifications when a DR and contract event is triggered by the platform should be available to the users.
- **Time series graphs**: the only case scenario of interest is when there is a clear linkage of time series view with a specific business objective. This is the case of generation/ consumption/flex forecasting (day ahead) as a time-series view is an intuitive diagram towards promoting maximization of self-consumption. In any case, timeseries visualizations will be supported in the 2nd version of the application.
- **Participation in energy markets** should be decoupled from the day to day operation/ performance of the users. The management of contracts should be simple and intuitive for the end users (rewards and contract time-period are the parameters of interest for the end users)

The design principles for the front-end has been fine-tuned after several iterations with the pilot partners of the project. The detailed description of the supported widgets is provided on the basis of the work flows and requirements as presented in previous section (wish list from pilot users).

<u>Configuration/ Authorization</u>

This the configuration layer of the application as the intro screen a new user will encounter. The process is delivered following the steps:

1. The end user is selecting the cooperative with whom he/she has an active commercial engagement; This information is necessary so that the prosumer portal knows which instance to contact in order to obtain the prosumer/consumer related information

2. The end user is setting the unique key associated with the list of devices that belong to the personal account; main prerequisite is the user to have already configured the OSB in the platform in order to have the associated DER parameters reported in the FLEXCoop platform.

3. This unique token is used in all information-related transactions between the FLEXCoop to identify the specific asset/property. The configuration of each property (with the list of measured/controllable devices) pre-exists in the central management layer (Middleware) of the FLEXCoop platform; established in an automated way once the smart devices (FLEXCoop OSB, additional FLEXCoop Smart metering units etc...) commissioned in the network.

4. In addition, a username/password is defined by the user (linked with the unique token) to facilitate the log in process.

5. An early version of terms and conditions should be accepted by the user. Which will be developed in the refined version per demo site at the 2nd version of the app

6. The user is able to configure some additional parameters (comfort settings etc..) as specified in the associated view of the system. Developed in the 2nd version of the app.

It is evident that the configuration process is defined as a single step process; making the overall commissioning as simple as possible. During the registration, no personal information is requested from the users. After the user is registered in the portal, may provide the username/password credentials to log in to the Prosumer Portal application.

<u>(Near) real time information visualization</u>

Information about specific metrics is available for visualization in a personalized way. Following consultation with end users, we specify the list of (at hourly level) real time metrics for visualization: **Energy Consumption**, **Demand Flexibility Potential**, **Energy Generation** (if available), **Self- Generation level** (if available), **Self-Consumption Level** and **Energy Cost** are the requirements as expressed from the pilot users of the project for data visualization.

<u>Historical information visualization – Aggregate Information Visualization</u>

Information about specific metrics are available for visualization in a personalized way. Following consultation with end users, we specify the list of metrics over the past period (e.g. last month, last year) available for visualization: Energy Consumption, Demand Flexibility Potential, Energy Generation, Self-Consumption Level and Energy Cost towards enhanced visualization.

<u>End Users Engagement – Comparative View</u>

The end users of the platform are prompted to get insights about their typical behaviour and performance. There are two ways considered for this type of engagement in project activities.

- Comparison with the past where the actual performance (for a specific time period) is compared with a similar/typical period in the past.
- Comparison with similar peers where the overall actual user performance is compared with the typical/ low /high portfolio performance. These groups are defined after consultation with the partners; in the project with cooperative members

There are different ways to define comparisons for the users; the definition of comparative views was performed after consultation with the pilot partners and their interest for peer comparisons (special interest for the energy cooperatives).

<u>Contractual Agreements Management</u>

This is one of the main objectives of the project; to establish a mechanism that facilitates the enrollment of end users in a marketplace and further active participation in contractual agreements with the different marketplaces, in FLEXCoop project Cooperatives & Aggregators. The list of features supported are: activation/de-activation of controllable devices in DER Registry, contracts notification, contracts acceptance/rejection, list of active/past contracts.

<u>Demand Response/Other Campaigns Management</u>

One of the main objectives of the project is to evaluate the participation in different business campaigns. Towards the engagement of end users in the overall concept, a separate view of the system is available providing aggregated information about: DR rewards, DR convenience/reliability, level of compliance with the associated contractual agreements. This function is complemented with the notifications triggered in case of DR campaign and user setting about participation/non-participation in a DR campaign.

We presented above the list of main functionalities supported by the 1st version of the Prosumer Portal. Additional configurations should be available (but not of high priority) for the end users, e.g. account settings reset, settings management for DERs. These modifications are considered as part of the user settings management.

6.3. Data Management Layer

As mentioned above, the Prosumer Portal vision is to become a tool provided by the Aggregators/Cooperatives to their customers/members in order to raise their awareness and engagement with energy-related aspects. Therefore, the Prosumer Portal app cannot stand as a separate tool, rather it relies on the availability of data from Aggregators/Cooperatives databases.

Towards this direction, any data request messages triggered by the Prosumer Portal should be received by the FLEXCoop Middleware implementation who will interpret these to the associate messages to the FLEXCoop components that contain the respective consumer/ prosumer level information. This is actually the main role of the data management layer; to

establish a seamless communication between the Prosumer Portal and the FLEXCoop Middleware in order to request specific data attributes required for further analysis and visualization. The core data elements required for analysis are summarized in the following table, highlighting also the Prosumer Portal business and user perspective. The detailed definition of the different data attributes was specified in D4.2 as part of the FLEXCoop CIM definition.

Metric /KPI	Reference Period				Aggregation	Objective
	Real Time (Last period)	Day	Week	Month	Individual or Cooperative	
Energy Consumption (KWh) & Forecasting	+	+	+	+	Individual /Cooperative	Enriched Visualization / End Users Engagement
Energy Generation (KWh) & Forecasting	+	+	+	+	Individual /Cooperative	
Potential Demand Flexibility (KWh) & Forecasting	+	+	+	+	Individual /Cooperative	
Energy Cost (m.u.)	+	+	+	+	Individual	
Self-Consumption Level (%)	+	+	+	+	Individual /Cooperative	
CO2 Emissions (Kg)	+	+	+	+	Individual /Cooperative	
Grid Import/Export	+	+	+	+	Individual /Cooperative	
DR / other Rewards (m.u.)	+	+	+	+	Individual	DR Campaigns Monitoring
DR Convenience (%)				+	Individual	
DR Compliance (%)				+	Individual	

Table 2: Data requirements Prosumer Portal App

In addition to the typical KPI and metric values as reported above, additional data attributes are considered for visualization: configuration parameters for DERs (information available in DR Registry), contractual parameters (information available in marketplace), DR notifications (triggered by DR management layer). These static parameters are defined as part of the internal data model, further specified as part of the FLEXCoop Common Information Model [5].

Along with the definition of information attributes, the way to retrieve these data is specified in this document. Overall, we are considering two ways of communication (the details are specified in Annex B) following the principles defined in D4.3 [5]:

- Request-response paradigm, to publish requests for the time-series based information; real time or historical
- Publish-Subscribe paradigm, where the Prosumer Portal is registered in specific topics/queues of the FLEXCoop Middleware to a) receive notifications about DR events, updates in DER registry and contractual offers & b) to update information about i) availability of DERs in DR or other campaigns in Registry ii) acceptance/ rejection of contracts iii) comfort related parameters

The Prosumer Portal application will rely on different tools for retrieving the aforementioned information. Most of the information will be available through the FLEXCoop central repository, especially information about metering / timeseries data. In addition, FLEXCoop Forecasting Engine & FLEXCoop Demand Response Layer will provide additional

information required for visualization. Finally, interfaces of Middleware with DER Registry and Open Marketplace are considered in order to retrieve marketplace related model parameters. In all these cases, the information exchanged will comply with the message format as specified in Annex B.

Along with the definition of the wrapper with the FLEXCoop Middleware at the application, an internal database is defined to stand as the local repository for the credentials of the user – username & password, configuration & contractual parameters (for persistency reasons). The abstract data model of the local DB is presented in the following figure.
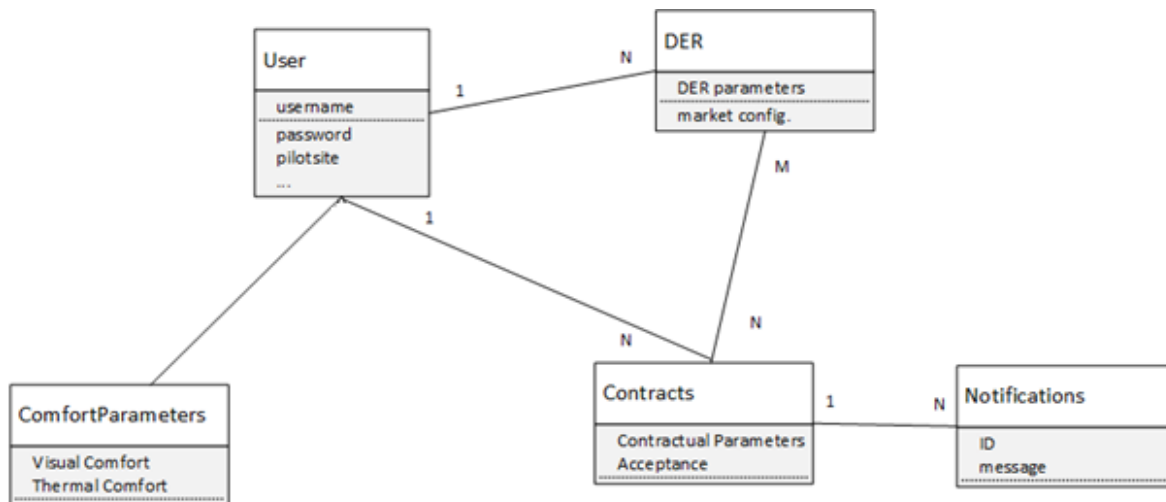
Figure 3: DB schema- Prosumer Portal

This is a conceptual view of the data model in order to incorporate (a) asset specific parameters, (b) business related parameters and (c) user related attributes that blueprint the overall design of the application. No metering data will be stored in the DB and thus the overall functionality is relying on the operation of the rest of the system components and the coordination of communications from FLEXCoop Middleware.

6.4. Prosumer Portal Application Layer

This is the application layer of the app, responsible to incorporate analytics processes to further support the different features of the Prosumer Portal App. Taking into account the list of requirements and use cases, we describe the different calculation processes at the analytics layer of the application.

6.4.1. Simple Analytics (Aggregation, Normalization)

One of the features supported by the app is about information to be presented in different ways. Towards this direction, aggregation and further normalization (based on baseline values) of energy consumption/generation/flexibility etc.… information is a main feature of the application. The two methods supported by this module are presented:

- **Data Aggregation**: Presentation of different profiles about **consumption**, **generation**, **energy cost**, **demand flexibility** by aggregating raw information available from central data repository. Given the seasonality of energy patterns as well as the requirements of end users, we define the different time periods for aggregation: daily, weekly, monthly.

- **Data Normalization**: Analysis following the definition of baseline (previous period data analysis) for comparative view. This enables information presentation in an intuitive way.

The latter incorporates a complex analytics process in the overall functionality. Baselines are not defined by default, rather these are periodically updated taking into account the history of data. The extraction of baseline values will set the basis for a more enhanced presentation of the associated metrics (colour coding of the different graphs taking into account actual vs. baseline values) and thus this calculation methodology is incorporated in the project.

6.4.2. End users' awareness and engagement

One of the main objectives of the application is to increase the level of awareness about energy consumption, self - consumption and demand response participation by providing enriched analytics about different KPIs: comparison with similar peers, previous period's analysis, etc... For that reason, a comparative analysis/view is supported as a means of insights for the end users. There are two layers of analytics:

- **Comparison with previous "similar" period** (consumption) as a means of motivation for energy efficiency

- **Comparison** with **portfolio/similar peers** (consumption) as a motivation towards energy efficient behaviour

The KPI calculation is further associated with intuitive visualization of information through the appropriate graphs. The end users are prompted to get insights about their typical performance at daily, monthly or annual level as a means of engagement in more efficient and sustainable behaviours. Moreover, alerts and recommendations may be supported through a colour coding visualization of the actual (vs. typical) energy profile.

6.4.3. Financial and economic management of contractual agreements

The role of this module is twofold: a) to provide information about energy costs by taking into account the tariff schemas applicable at each pilot b) to present information about financial rewards & DR convenience following end user's participation in demand response campaigns.

In addition, the management of contractual agreements is the main feature of this module; KPIs about compliance with contracts, negotiation history etc… are supported by the app. More specifically, the different market-oriented features supported by the app are presented:

- **DER availability in DER Registry**: The end user is able to select which DER will be accessible in DER Registry for further engagement in market schemas of the Aggregator.

- **Contracts Management**: The end user participating in the energy marketplace is able to negotiate the contractual parameters with the Aggregators.

- **Contracts History Visualization**: The end user may have access to active or close contractual agreements with one (or more) Aggregators and get an overview of user performance under the different contractual agreements.

It is evident that the role of this module is to mainly handle the parameters about the contractual agreements in a persistent way and further transform this knowledge to information for further visualization. In addition, information about DR performance is handled by this component and is available for the end users of the app.

We presented above the design principles of the Prosumer Portal App as a core component of the FLEXCoop framework. The updated design is based on the list of requirements and user needs as presented in the previous section, along with the definition of FLEXCoop architecture in WP2 and the development of the definition of the data management layer of the project in WP4.

7. PROGRAMMING LANGUAGE & RELEVANT LICENCES

The implementation of the Prosumer Portal app involves a number of different frameworks. The paragraphs below indicate the implementation technologies are involved in each aspect of the implementation.

For the **data management layer**, REST/JSON services are defined together with CIMNE (middleware leader) for requests over historical data while a RabbitMQ client implementation is considered as the means of communication with the FLEXCoop middleware; FLEXCoop middleware message is based on RabbitMQ and thus the prosumer portal app includes appropriate wrappers to interface with it.

An internal DB structure is defined mainly for persistency reasons and thus a Postgres DB [6] is defined per pilot to store only the necessary information for persistency reasons. For storing historical time-series data, Influx DB[7] was selected as the framework for timeseries data management.

For the backend, the development as a Python application through the usage of Django[8] which natively support ORM integration. Django supports by default the MVC principles and thus the development was performed in accordance with the definition of the architecture.

What relates to the FLEXCoop notifications engine, a websocket based implementation is support to enable live updates about the status of notifications. At the back-end, the Firebase (https://firebase.google.com/docs/cloud-messaging/) platform is integrated to enable multimodal delivery of notifications to the different end users of the platform.

For the front-end view of the system and considering that Prosumer Portal is a web-based application, we are using typical web technologies (HTML/HTML5/CSS3) for the design and the most recent JavaScript frameworks for information rendering. Vue.JS [9] is the framework supported for information visualization, working seamlessly with the Django framework presented above.

To enable also dynamic characteristics on the front end, typical JavaScript libraries have been considered, namely charts.js [11] and apexcharts.js [12] which support a numerous types of visualization widgets. As the requirement is for a responsive visualization, the Vuetifyjs framework [10] is incorporated on top of the Vue.JS. Vuetifyjs is defined as a Material Design Component Framework and thus it supports the seamless rendering of the different design components in order to address the different visualization requirements (visualization through different communication means).

What related to the **deployment** of the Prosumer portal, as defined in WP2 all FLEXCoop components should be deployed as dockerized components in local repositories. Therefore, the dockerized image for the Prosumer Portal app contains an Uwsgi which acts as the application server / Web Server Gateway Interface in conjunction with a web server such as Apache and Nginx to support web visualization. The deployment as a Docker [13] app, supports any further plug and play deployment of the app to any server in the project (e.g. demonstration server at any pilot or test server for stress test of the technical implementation).

It is evident that the overall development is performed by exploiting open source and license free technologies and thus no limitations apply at any further commercialization or demonstration of the solution.

7.1. Prosumer Portal Security & Privacy Aspects

Data security and privacy is of high importance in the project. Towards this direction, the design and implementation of the Prosumer Portal app took into account from the beginning a number of end user requirements to ensure that data will not be compromised.

State of the art techniques and mechanisms are considered to achieve this goal, namely for data security any communications we are using the HTTPS protocol with information encryption under TLS1.2; for data privacy we adopt a **username & password** mechanism as specified above: The username can be an alias of choice of the users which does not contain personal information and the password will be hashed and stored in the local DB, to authenticate users when accessing the app.

The Prosumer Portal application specific authentication process is further enhanced with the adoption of the FLEXCoop holistic authentication process as reported in WP4. More specifically, to enable end users to get authenticated in the project, the system needs a way to create new users in the Middleware. As stated above the aim is about full anonymization by not "knowing" who the user is but providing a way to make sure that "somebody" has this data. OAuth 2.0 allows the use "**third party issued assertions**". This means that a third party can provide a secure channel for confirmation that a person / entity can create an account. The drawing below shows a simplified view on this process
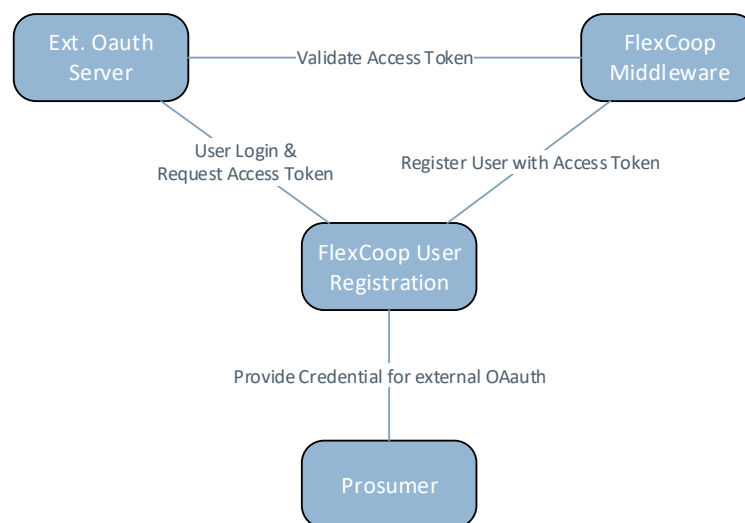
Figure 4: FLEXCoop holistic authentication process

Therefore, the sequence diagram for communication of Prosumer Portal App with FLEXCoop Middleware is specified.

1. User registration process at the Prosumer portal Client side (Web app) with (username, password, OSB ID) to be provided by the end user and info available at Prosumer Portal Server side.

2. Prosumer Portal Server communicates with Ext OAuth Server (providing username, OSB ID) and request a unique Access Token for this specific user
3. Unique Access Token sent back to Prosumer portal Server
4. Prosumer portal server request information from the Middleware by using this unique Token

As highlighted above, the Prosumer Portal App will store the minimum of information required for the demonstration. No streams of data will be stored locally; while information about the configuration parameters of each property, the list of notifications and the different contractual agreement (non-sensitive information) parameters will be handled in a fully anonymized way by the app. Complementary to the development and deployment view of the system, the manual documentation for the application (1st version) is reported in the following section.

8. CONTENTS OF THE CURRENT RELEASE

The development of the different features is in line with the design principles as reported in previous section along with the technical requirements specified in Section 3. The different views (developed as micro-widgets) of the Prosumer portal app are presented.

A demo account is available and accessible for demo reasons:

http://flexcoop.s5labs.eu/

user/pass: user/user

8.1. Prosumer Portal App Visualization Framework

The different views of the tool are presented; this is the first version of the application for the end users. The design of the app has been performed in a fully responsive manner to enable visibility from different device types. Therefore, screenshots from different screens are presented in this section.

Configuration & Authorization View

This is the view of the system about the configuration of an account. The process is delivered in a semi-automated way: user providing the unique asset credentials and the list of parameters is dynamically retrieved in the background. Username/Password registration process is supported:
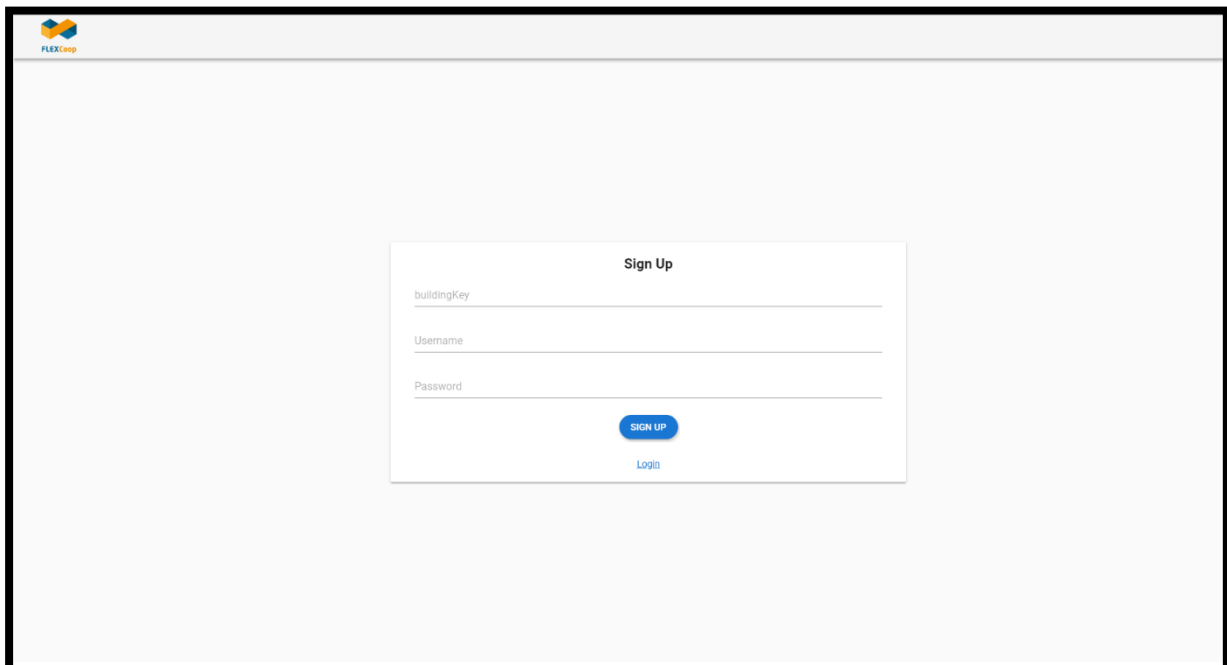
Figure 5: Screenshot from registration process - Web View
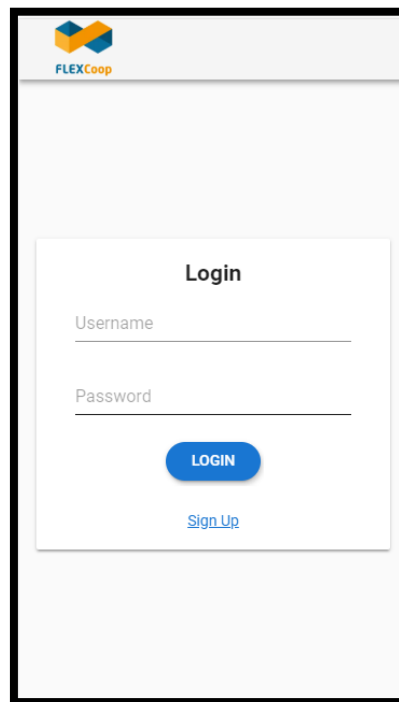
Figure 6: Screenshot from registration process - Smartphone View

(Near) real time information visualization

Information about specific metrics are available for visualization in a personalized way. This information is periodically updated covering consumption, cost and contextual conditions information.
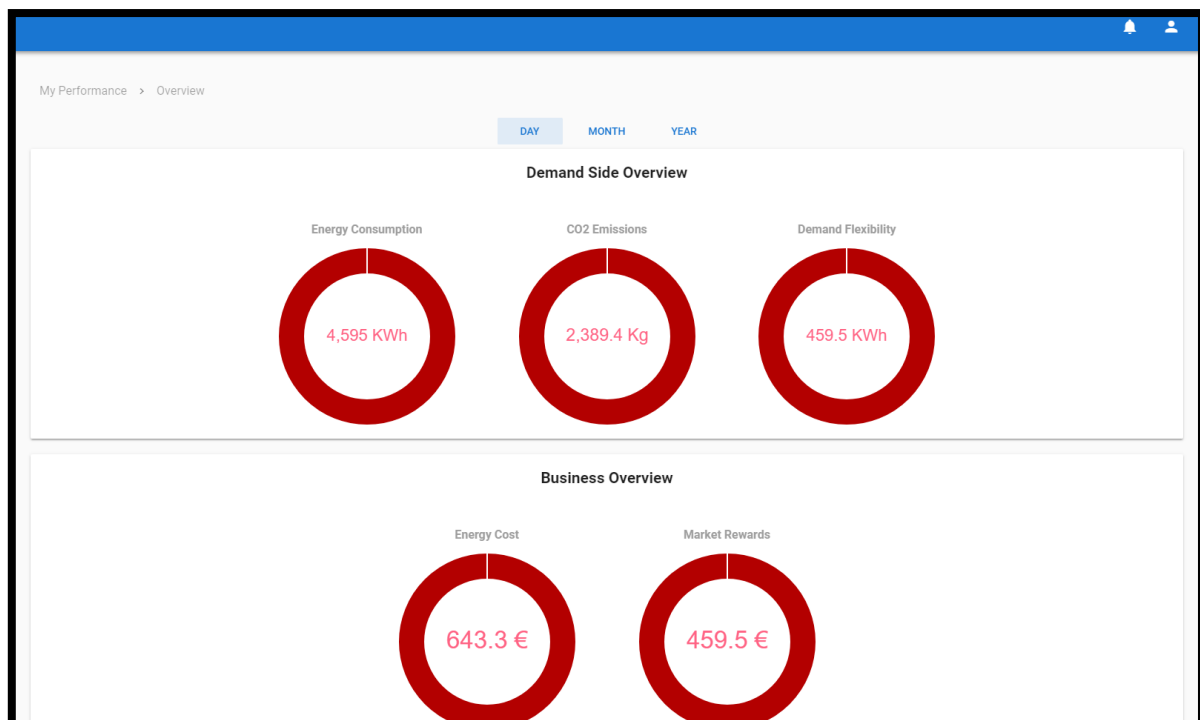
Figure 7: Screenshot from (Near) real time information visualization - Web View
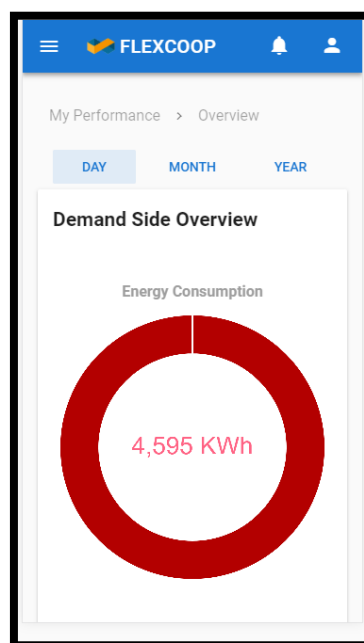
Figure 8: Screenshot from (Near) real time information visualization - Smartphone View

<u>**Historical Analysis - Individual Reports**</u>

Information about historical performance is available for visualization. The analysis covers both KPIs (normalized) visualization and time-series analysis. Focus on consumption, generation, potential flexibility, costs & aggregate rewards data visualization.
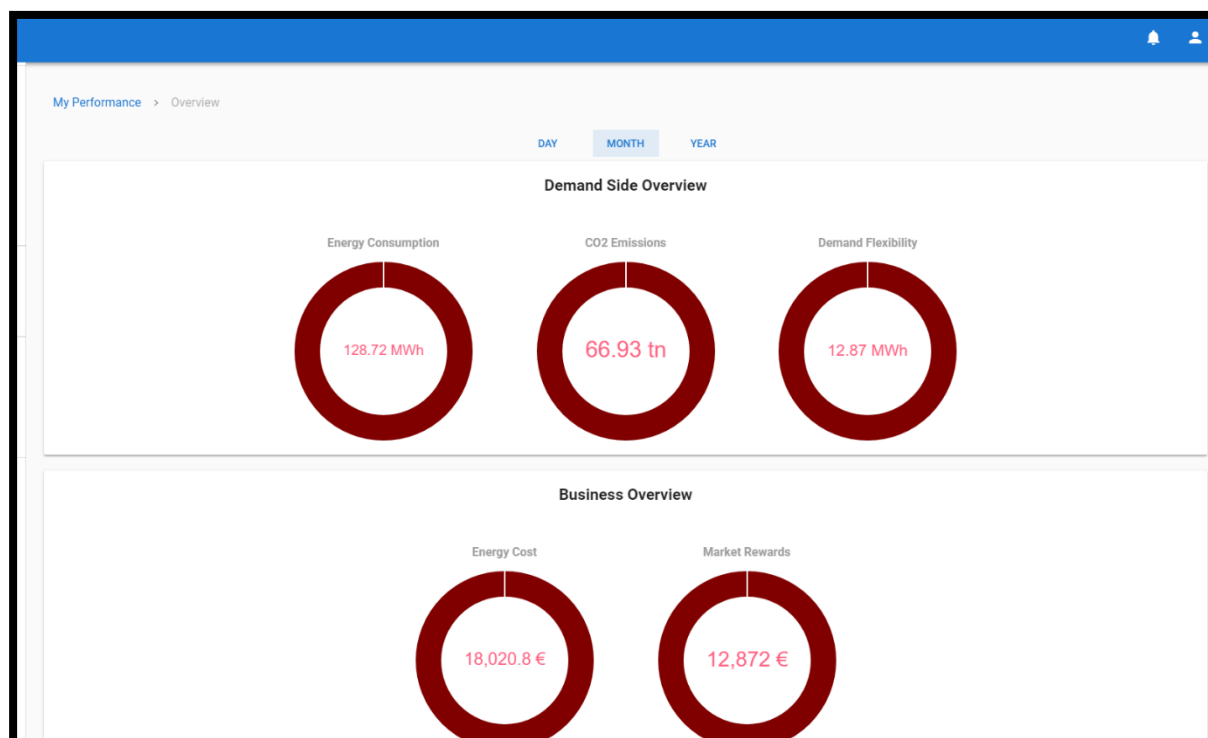
Figure 9: Screenshot from Historical Analysis - Individual Reports - Web View

End users Engagement – Comparative View

End users' engagement by providing meaningful insights. The user is prompted to participate in the project by getting quick messages related to his/her performance. Comparison with a previous period (typical performance) is an indicator to compare with the current performance. In addition, comparison with similar peers as a means to easily evaluate individual performance with this of a typical/ energy efficient/ energy wasteful user.
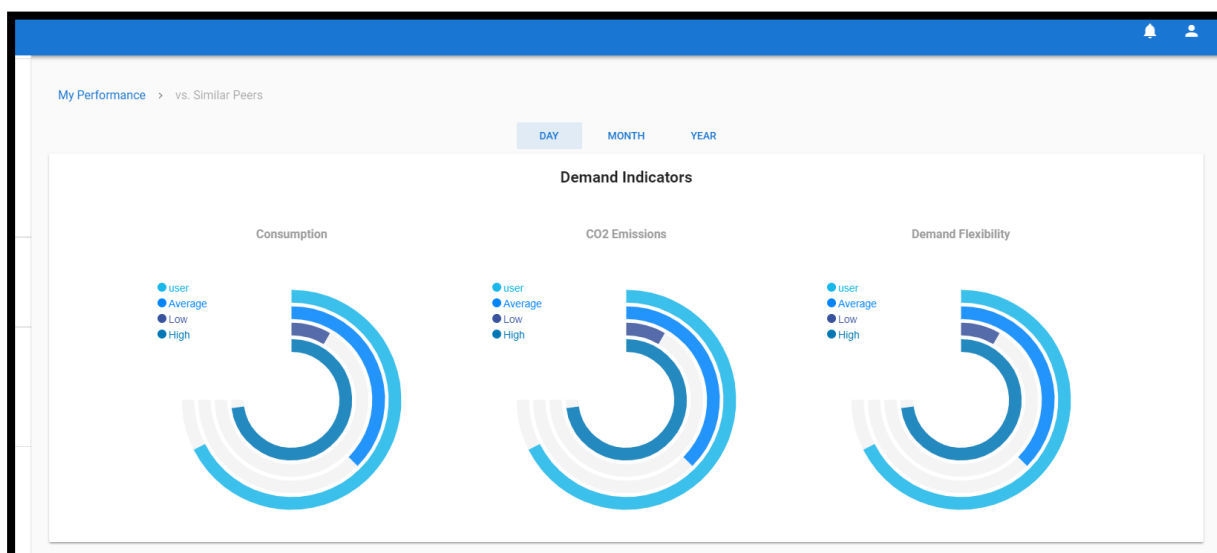
Figure 10: Screenshot from End users Engagement – Comparative View - Web View
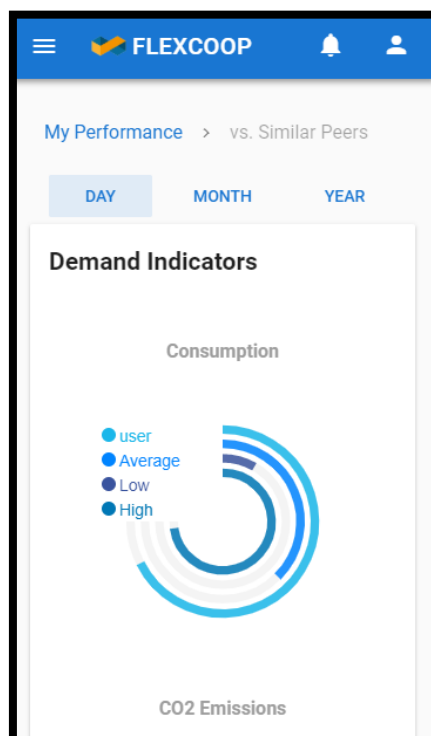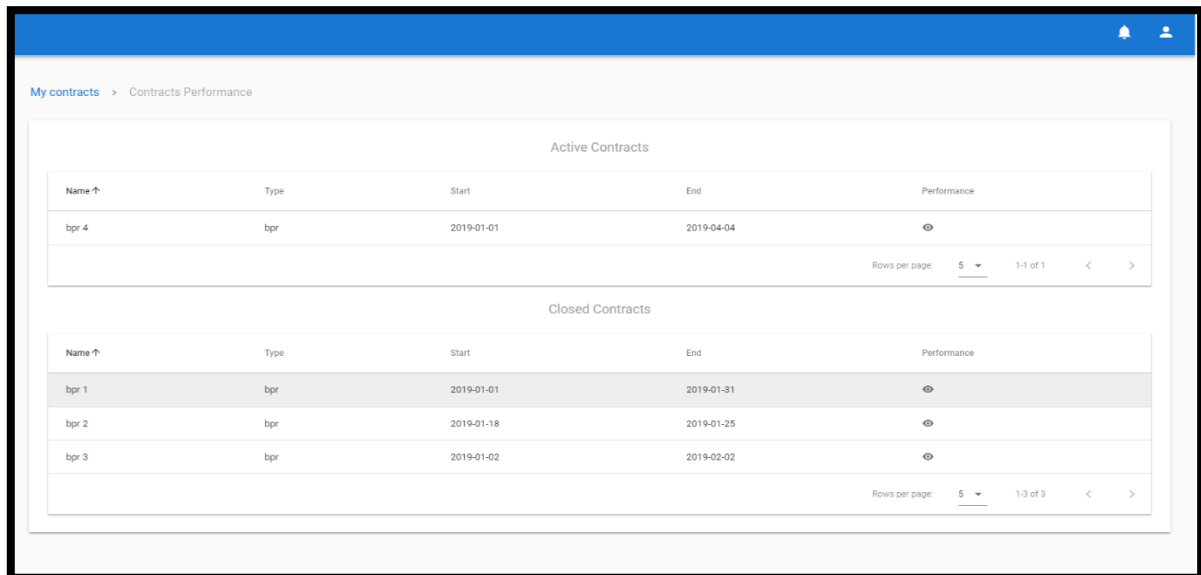
Figure 11: Screenshot from End users Engagement – Comparative - Smartphone View

<u>Demand Response Campaigns Management</u>

This is the view of the system focusing on Demand Response campaigns. DR performance KPIs are reported along with rewards for participation in these campaigns.

Figure 12: Screenshot from DR performance monitoring - Web View

In addition, notifications for DR activation are available through the dashboard.

Figure 13: Screenshot from DR notifications - Web View
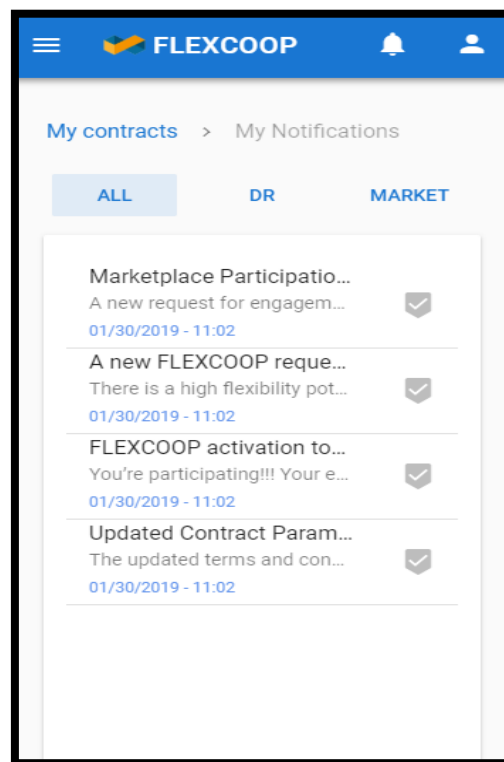
Figure 14: Screenshot from notifications - Smartphone View

Contractual Agreements Management

This is a main innovation of the project, enabling the functionality of end users participation in energy markets. The way to register DERs in the registry and operate in the market is defined through the following screenshots.
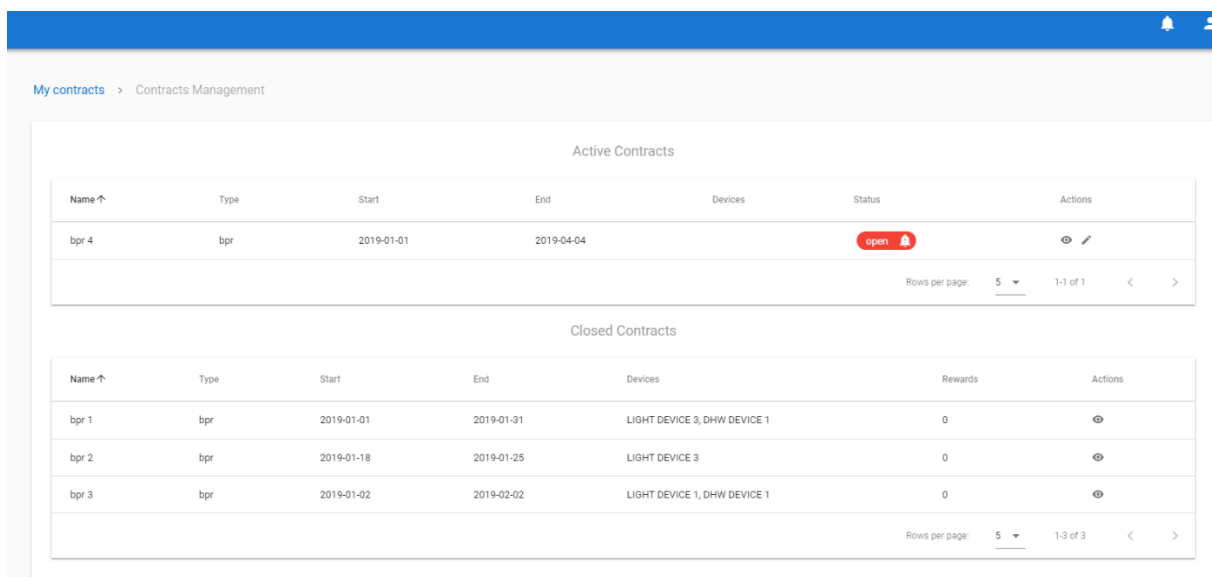
Figure 15: Screenshot from Contractual Agreements Management - Web View

The end users are prompted to accept/ reject of even further negotiate (e.g. duration, remuneration etc..) a contractual offer.
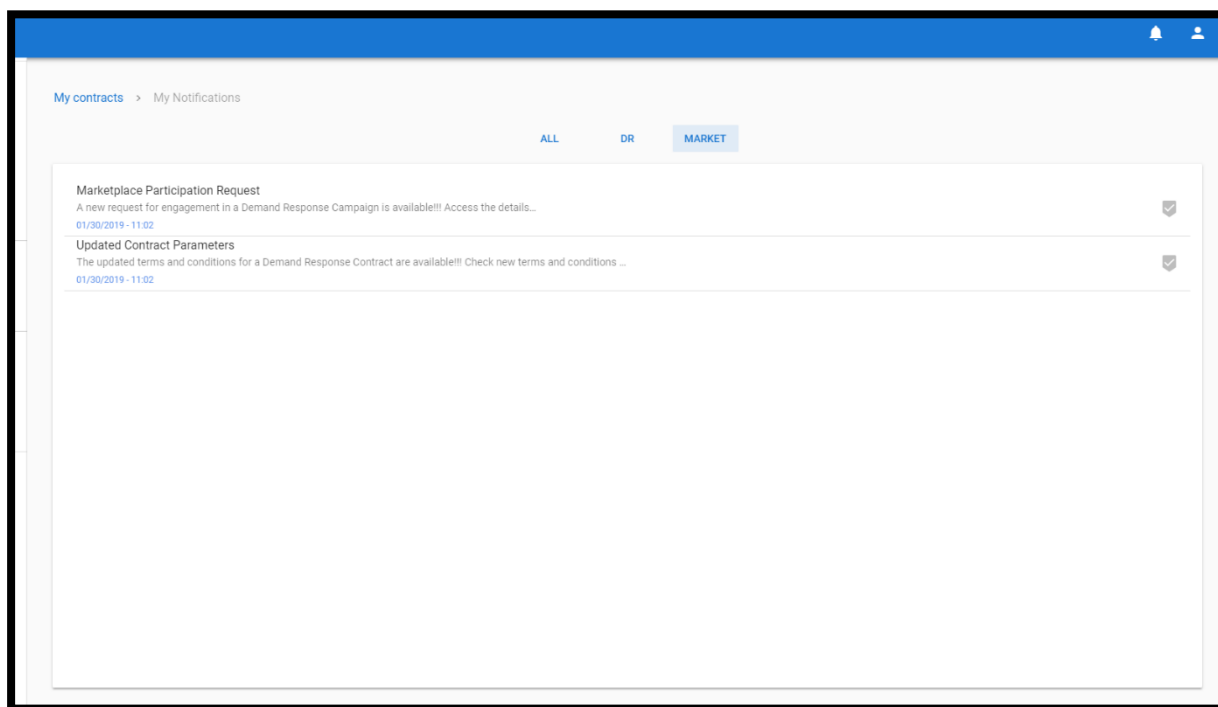
Figure 16: Screenshot from Contractual Agreements Notifications - Web View
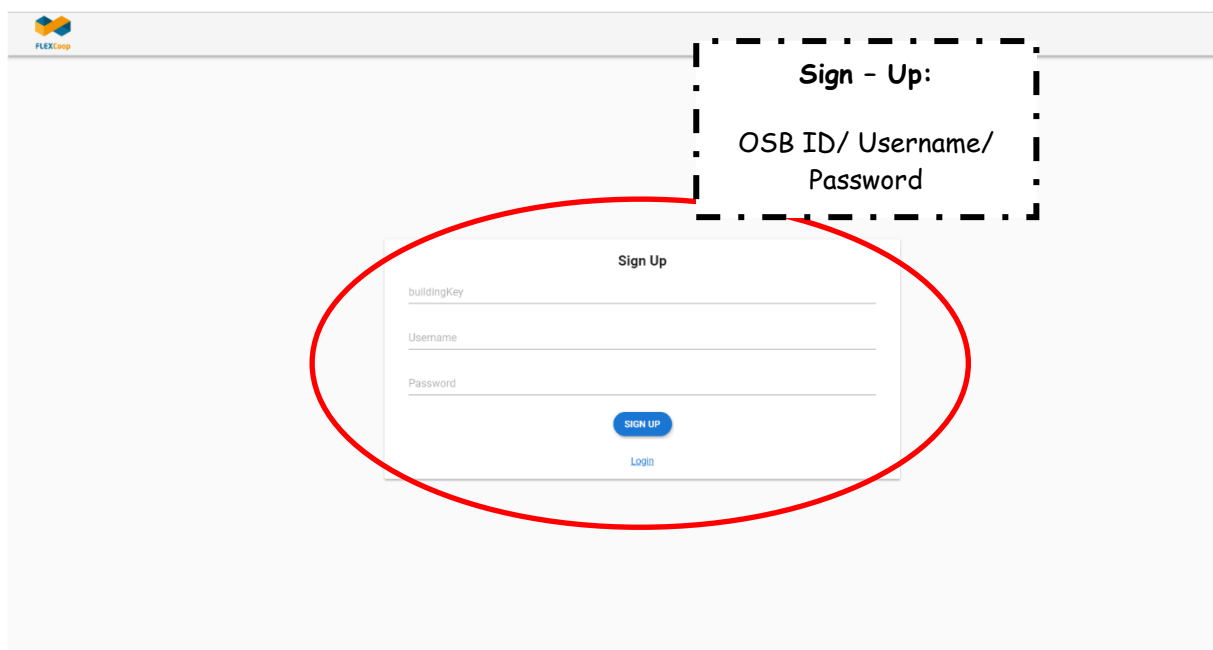
We presented the different views of the Prosumer Portal application as an outcome of the development process. The technical details about the development of the application were presented in previous section.

9. USER GUIDE

As this is a tool for end user's non-experts in the domain a user guide is required. This is the section about the manual documentation for the end users/prosumers. The overall presentation of the manual is in accordance with the presentation of the different views of the application as reported in previous section. We have to point out that a 2nd version of this document will follow in the 2nd version of the deliverable, presenting the full list of features supported by the app.

Configuration & Authorization View

The users set the parameters at the configuration process. Unique ID/Key for the configuration and username/password for account settings

Once registered in the system, the user provides the credentials for logging into the portal
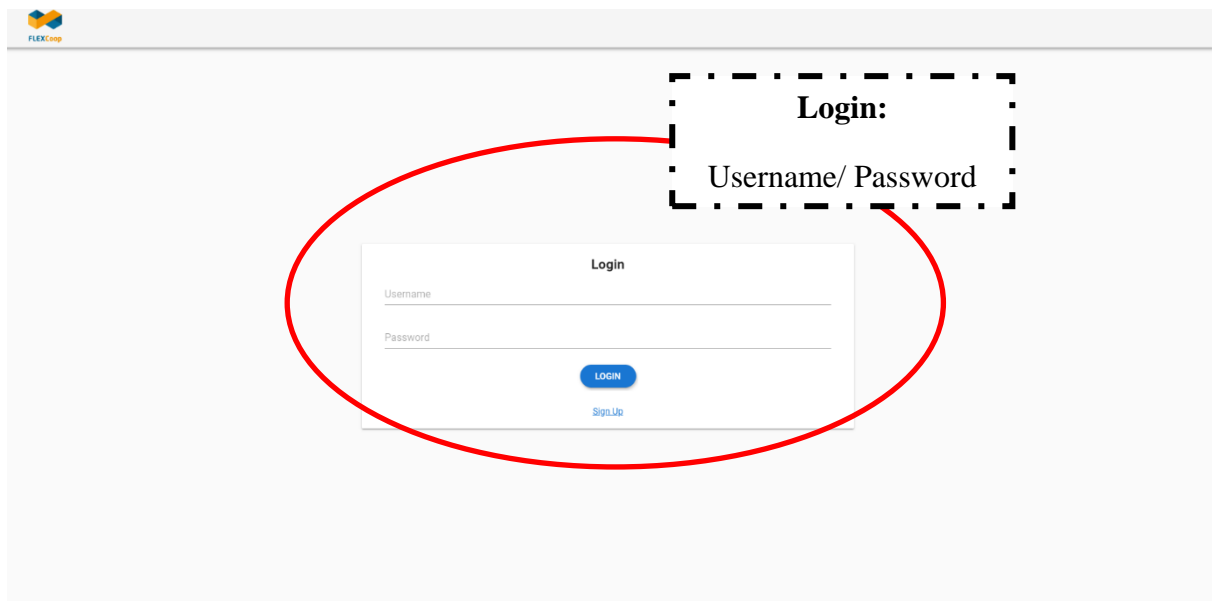
Main Features

From the left side drop-down menu the user can select from the list of features

- Real Time Dashboard/ Intro
- Historical Analysis
- Insights (Comparative view)
- Contracts monitoring and management
- User Settings

Real Time Analysis: this dashboard displays information about the performance of the user's home in terms of energy in (near) real time

Historical Analysis: this dashboard displays information about the performance of the user's home in terms of energy in the past (aggregated information)

Consumer Insights: Comparison with the past and comparison with similar peers as a means of user engagement towards energy efficiency (participation in energy markets)

Contracts monitoring and management: This dashboard visualizes information about contractual agreements on progress with Aggregators. User is able to accept/reject a contract and get a quick overview of active/close contracts. In addition, this view of the system presents information about participation in DR campaigns: DR rewards/ penalties, DR level of compliance and the list of DR events triggered

User Settings: The role of this dashboard is twofold: account management (reset password) and device settings (real time control of devices and update of DER registry).

A more detailed presentation of the different views of the application:

(Near) real time information visualization

Information about specific metrics are available for visualization in a personalized way. This information is periodically updated covering consumption, cost and contextual conditions information.
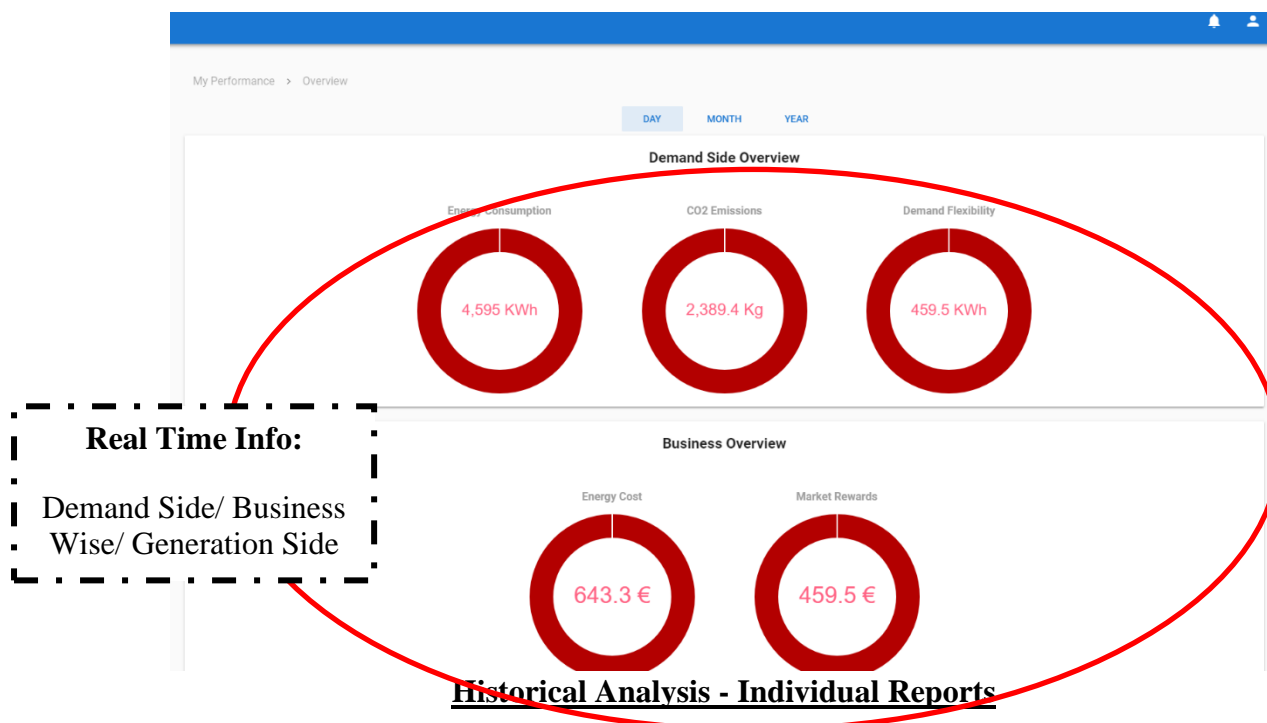

Information about historical performance is available for visualization. This view is about aggregate information visualization. Focus on consumption, generation, flexibility, costs & reward data visualization.
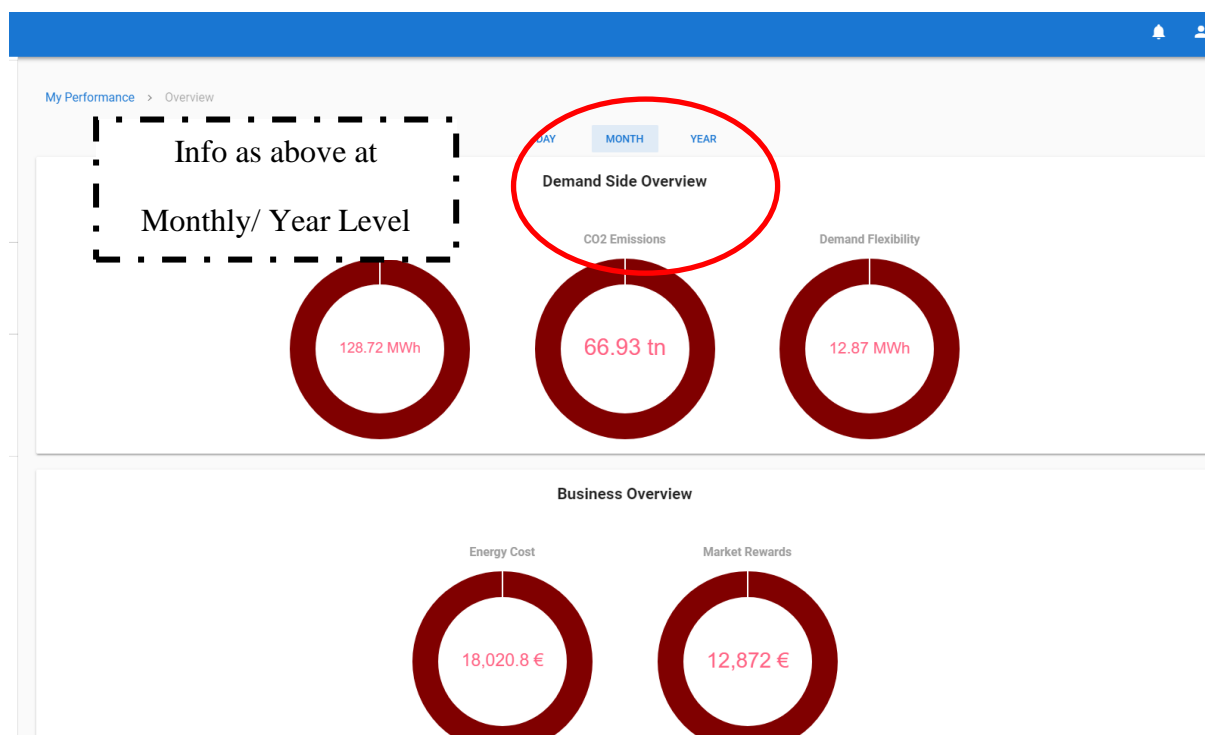
End users Engagement – Insights about performance

In addition to the collaborative level engagement, the tool should support engagement of end users at individual level. Towards this direction, the insights view of the system provides useful information about the actual performance compared with a past/ typical period or compared with other users.
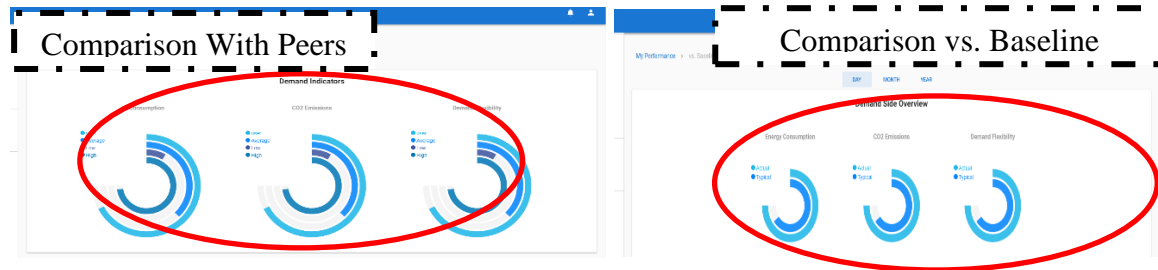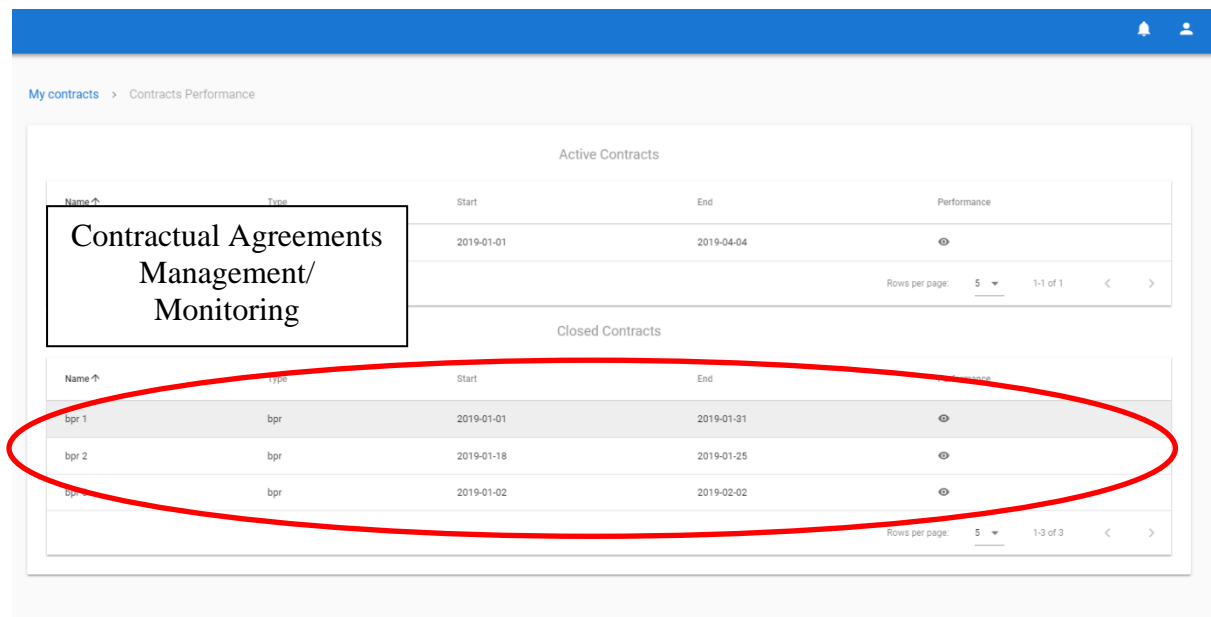

In addition to steer presentation of metrics/KPIs, recommendations are indirectly extracted by providing a color coding to the different KPI values.
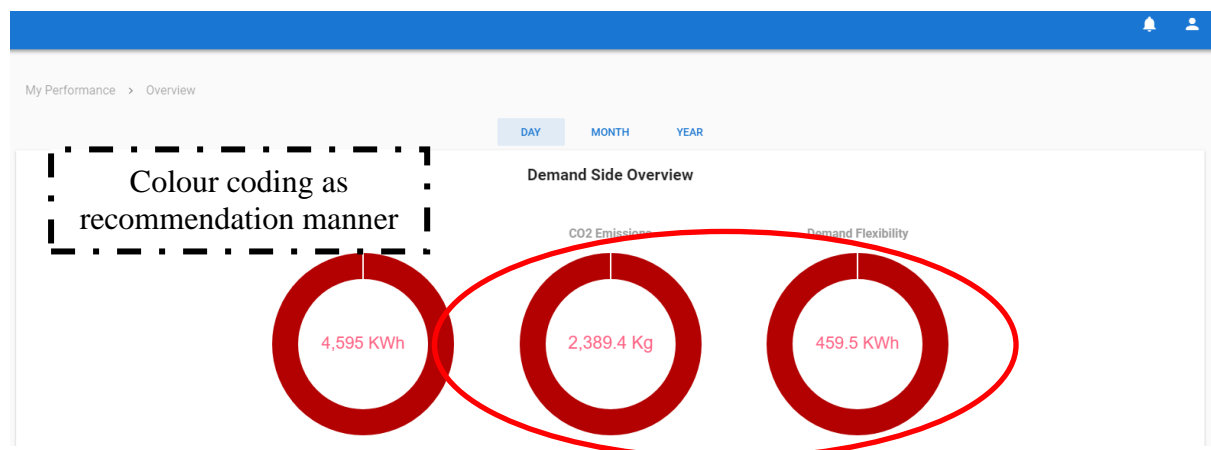
Contractual Agreement Monitoring

This is the view of the system focusing on Demand Response campaigns. DR performance KPIs are reported along with the rewards for participation in these campaigns.
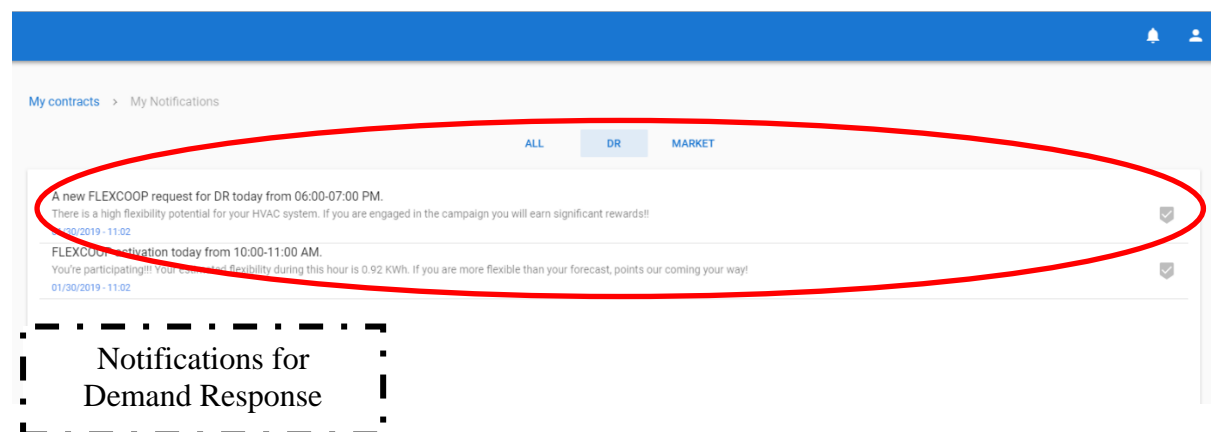
In addition, notifications about DR activation are available through the dashboard.

Contractual Agreements Management

This is a main innovation of the project, as the end users are prompted to participate in energy markets. The users are receiving notifications about market participation
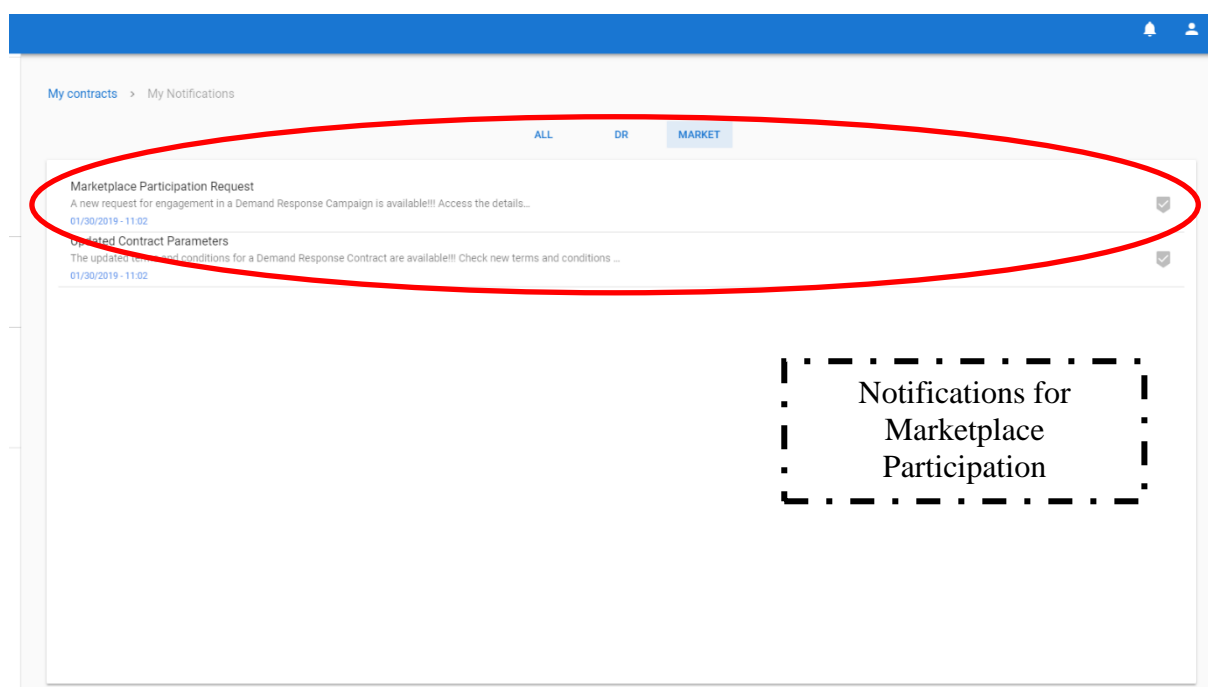
The end users may accept/ reject or negotiate the contractual parameters as presented in the following:
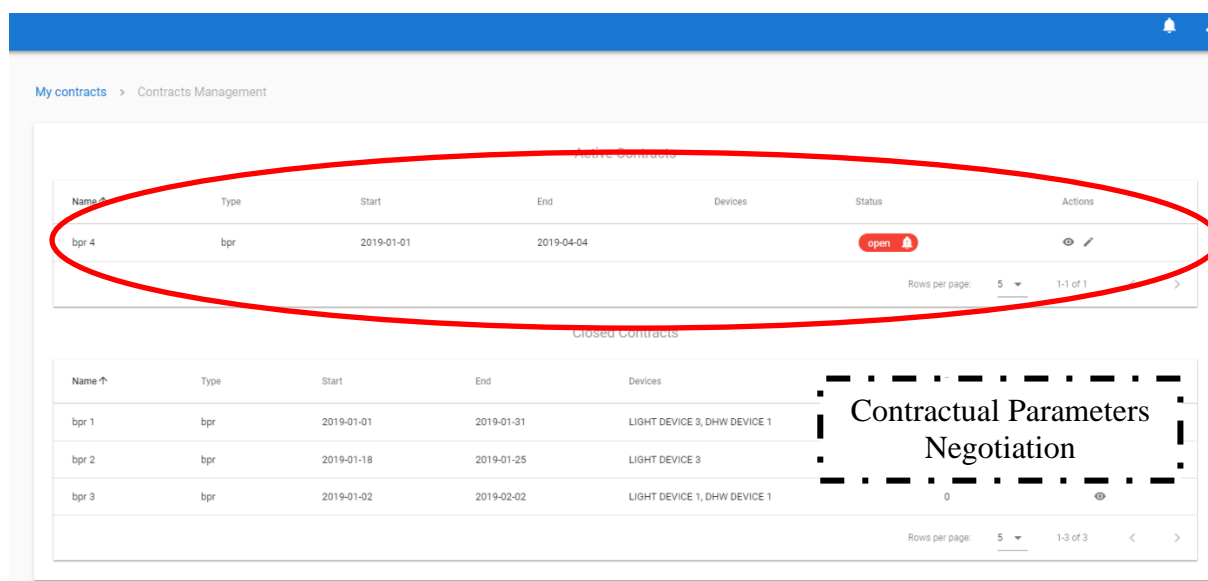
User Settings Management

The end user of the platform may:

- **Reset password** for user to modify password credentials

- Provide **settings parameters** about the availability of the DERs in the market/ DR campaign

- Provide settings parameters about **comfort conditions**, to support the establishment of a user centric energy management framework as defined in the project.

The final version of the manual will be reported in the 2nd version of the deliverable. It will incorporate the updated features to be considered in the final version of the app, taking into account end users' feedback during the demonstration period.

9.1. Alpha Phase Demonstration Activities

The next step is the early demonstration of the project application at the different demo sites; thus, we blueprint the activities to be performed in order to ensure the prompt evaluation that will provide the feedback required for the 2nd version of the app (to be reported in D6.7 and the 2nd version of this document).

The different activities to be performed are specified:

- An early selection of friendly users has already been performed; to act as the alpha testers for the application
- In addition, the detailed evaluation activities have been reported and presented in the 1st version of the document in D7.2 (M22)
- The details about the deployment of the Prosumer Portal App will be reported in D7.3 in M26 presenting also the profile of the alpha testers
- Finally, the preliminary results from this early evaluation process will be reported in D7.5 in M30 along with the key remarks that will lead the work for the updated and final version of the Prosumer Portal App.

Considering the definition of the validation framework and the evaluation of the app at the different demo sites the focus is about:

- End users' evaluation, providing a short questionnaire version in order to get quick insights about the current status of the application
- Functional evaluation of the app, by quantifying end user's participation / interaction with the app. Indicative KPIs defined for the functional evaluation of the app: New/ Returning Visitors, Participation in marketplace, Participation in DR campaigns etc.

The detailed evaluation plan is reported in WP7. Therefore, a short overview was provided to screen the landscape for the next steps of the work in the project.

10. INTERFACES WITH OTHER COMPONENTS AND THEIR INTEROPERABILITY

As stated above, integration is performed via FLEXCoop Middleware, though access on data is ensured through via Middleware communication with other system components. The detailed analysis of data attributes required and interface layers are presented in the CIM document of the project in D4.3- FLEXCoop Common Information Model - Preliminary Version. The actual implementation of these interfaces with Middleware will be reported in D6.8- FLEXCoop Integrated DR Optimization Framework and Pre-validation results - Final Version as part of the integration activities performed in the project.

Nevertheless, a summary version of these interfaces is presented in the following highlighting the interconnection with the different functionalities supported by the Prosumer Portal App:

- Interfaces with Middleware SEAC layer to handle authorization/ authentication process as defined in section 4.1
- Interfaces with OSB for real time access of the information about consumption and indoor environmental conditions in building premises
- Interfaces with Aggregator Portal for accessing cost and CO2 emissions related information along with notifications about auto DR events triggered by the FLEXCoop platform
- Interfaces with DER Registry for setting availability of the DERs available for monitoring and control
- Interfaces with Open Marketplace for actively participating in the marketplace (setting contracts and agreements)
- Interfaces with DR settlement and remuneration for the campaign's performance monitoring and remuneration for marketplace participation.

It is evident that the Prosumer Portal App interfaces with almost all software modules defined in the project, ensuring that way the active engagement and participation of the prosumers/ consumers in all processes of the FLEXCoop energy management value chain. From a technical viewpoint and while the details of interfacing are presented in T6.4, a conceptual overview of interfaces definition for the prosumer portal is presented in Annex B.

11. CONCLUSION

The conclusions are summarized in this section reporting the main outcomes from the design process for the Prosumer Portal application. Starting from the end users' requirements analysis and business objectives definition, we aim to deliver a fully-fledged application that includes features beyond of what was reported in WP2.

As stated in the DoA, the Prosumer Portal Application main objective is to develop the web application for the domestic prosumers involved in FLEXCoop project to be able to monitor and manage their energy profiles and parameters with a high objective to support the modification of their energy behaviour for reducing consumption and emissions and promoting energy efficient and sustainable behaviour. Moreover, the Prosumer Portal goal is to actively involve prosumers at the electricity markets through demand response schemes in order to further support the reduction of stability problems of the distribution grid network and smoothly promote the integration of renewable energy in the grid.

In accordance with the main objectives as presented above, the work performed during the reporting period has been focusing at the **design and the specifications definition** of the Prosumer Portal App visualization environment. More specifically, the starting point is the analysis of the requirements and use cases as presented in WP2 in order to define the user stories that prescribe the different features to be supported by this software component.

The next and main step of the work is the **development** of the Prosumer Portal App. The Prosumer Portal App consists of the front-end and the back-end system. The front-end is the visualization and user interaction layer in order to (i) monitor, (and partially manage) consumers consumption; (ii) engage end users in Demand Response campaigns and energy efficient behaviors; and (iii) enable end users to actively participate in energy marketplaces offering their potential of flexibility. The front-end was constructed using mature and standardized web-based technologies. To support the development an enriched and intuitive information visualization, state of the art (.js based) visualization techniques and frameworks were considered at the development process.

The back end consists of the model that incorporates the functional implementation of the Prosumer Portal app. The data management layer has been defined including the non-exhaustive list of the static and dynamic data attributes required by the app. More specifically (a) user profiling parameters (b) asset related parameters and (c) business (contracts and DR) related parameters are defined as part of the internal model.

In addition, a non-exhaustive list of features is supported going beyond the requirements and the expectations of the pilot users. Along with the sample analytics (data aggregation and normalization), more enhanced analytics are incorporated to support the engagement of the users and further their active participation in the different market and contractual schemes defined in the project.

Considering the technical implementation, the development was delivered in mature and standardized frameworks (Python based) that support the easy scalability and the expandability of the main functionalities, while the deployment as a container app is in line with the requirement for a plug and play and fast deployment in different working environments. Out of the list of requirements, several key requirements related to privacy and

security concerns were also considered at the development of the Prosumer Portal Application.

Overall, this 1st version of the app may be defined as a mature version for alpha testing at the different pilot sites. Additional enhancements and updates of the app will be available at the 2nd version of this software component, incorporating also in the analysis the additional requirements and user needs to be defined during the alpha testing period.

APPENDIX A: LITERATURE

[1]. FLEXCoop Description of Actions
[2]. FLEXCoop D2.1: End-user & Business Requirements
[3]. FLEXCoop D2.4: Emerging Business Models, Associated DR Strategies and Standard Contract Templates
[4]. FLEXCoop D2.6: FLEXCoop Framework Architecture including functional, technical and communication Specifications - Preliminary Version
[5]. D4.3 FLEXCoop Common Information Model - Preliminary Version
[6]. https://www.postgresql.org/
[7]. https://www.influxdata.com/
[8]. https://www.djangoproject.com/
[9]. https://vuejs.org/
[10].   https://vuetifyjs.com/en/
[11].   https://www.chartjs.org/
[12].   https://apexcharts.com/
[13].   https://www.docker.com/

APPENDIX B: PROSUMER PORTAL API DEFINITION

In this section, we specify the details of the message structure between Prosumer Portal and FLEXCoop Middlware towards retrieving information for further analytics and visualization. As part of the header of the message we specify a set of metadata, as the minimum required information so that the different tools can associate messages to each other (e.g. responses with requests):

- **senderID**: is the unique ID of the tool that sends the message. Prosumer Portal will send the request and FLEXCoop Middleware will send the response. The most important aspect in this attribute is that it specifies exactly which instance of the tools is the sender, in case multiple instances/deployments
- **recipientID**: is the unique ID of the target message recipient, same things apply as above.
- **conversationID**: is a unique identifier that is common to a pair of request and response messages; acting as an acknowledgement to keep track of message pairs that comprise a "conversation" .
- **messageType**: is the field that specifies the type of the message, whether is a request or a response.

The details of the Message structure that contain the content of the request from Prosumer Portal in order to retrieve the data required for visualization are further presented:

assetKey: The unique ID specifying the asset/ prosumer associated with the specific account. This is the token associated with the OSB ID as the unique identifier for an asset. An extra token (the reference ID of the Cooperative) is defined for retrieving aggregate / cooperative level information.

metricType: This is an enum type specifying the metric type for data retrieval; selection from the list of metric types: ENERGY_CONSUMPTION, ENERGY_COST, ENERGY_GENERATION, DEMAND_FLEXIBILITY, SELF_CONSUMPTION, etc…

dataType: This is an enum type specifying the type of data format to be included in the response (in case of time-series or cumulative data the respective time duration will be specified by the startTime, endTime and Interval parameters); selection from the list of data types: REALTIME, CUMULATIVE, TIMESERIES.

Note 1: More specifically, the optional fields specify a time duration and a sampling interval; apply when there is a request about cumulative or time-series analysis.

startTime: Timestamp specifying the start time in case a cumulative amount or a time-series is requested

endTime: Timestamp specifying the end time in case a cumulative amount or a time-series is requested

Note 2: We have to point out that due to the specific requirements from the end users, we may replace these two fields with a specific enum. types for the predefined time periods: day, week, month

In addition, in case of TIMESERIES values, we need to specify the interval period for the analysis.

interval: Enumeration specifying the time interval required when a time-series is requested

Indicative values are: HOUR, DAY, and MONTH in accordance with the functional requirements and the selection of day, month, and year period for cumulative analysis.

Note 3: We have to point out that time-series information is expected to include only the respective values within the specified interval, not the cumulative progression.

The following message is an example of a specific message request from the Prosumer Portal App to the FLEXCoop Middlware asking for the following information:

•   The time-series of consumption of the assetKey"ES001" between XX:XX and YY:YY on XX/YY/ZZZZ in intervals of 15 minutes; and

•   The cumulative energy generation for the same asset for the same start date until now

```
{
  "header": {
    "Sender": "ESProsumerApp ",
    "Recipient": " ESMiddleware",
    "conversationID": "ECPDK1004001",
    "messageType": "REQUEST"
  },
  "body": [
    {
      "assetKey": " ES001",
      "dataType": "TIMESERIES",
      "metricType": "ENERGY_CONSUMPTION",
      "startTime": "2018-01-01T00:00:00.000 Z",
      "endTime": "2018-01-01T03:00:00.000 Z",
      "interval": "HOUR"
    },
    {
      "assetKey": " ES001",
      "dataType": "CUMULATIVE ",
      "metricType": " ENERGY_GENERATION ",
      "startTime": "2018-01-01T00:00:00.000 Z",
    }
  ]
}
```

```
{
  "header": {
    "Sender": " ESMiddleware ",
    "Recipient": " ESProsumerApp ",
    "conversationID": "ECPDK1004",
    "messageType": "RESPONSE"
  },
  "body":[
    {
      "assetKey": " ES001",
      "dataType": "TIMESERIES",
      "metricType": "ENERGY_CONSUMPTION",
      "metricTimeseries": [
        {
          "value": 1.20,
          "timestamp": "2018-01-01T01:00:00.000 Z",
        },
        {
          "value": 1.44,
          "timestamp": "2018-01-01T02:00:00.000 Z",
        },
        {
          "value": 1.07,
          "timestamp": "2018-01-01T03:00:00.000 Z",
        }
      ]
    },
    {
      "assetKey": " ES001",
      "dataType": "CUMULATIVE ",
      "metricType": "ENERGY_GENERATION",
      "value": 250.00
    }
  ]
}
```

The response message from ES Middleware to Prosumer App is presented.