TOWARDS SMART ZERO CO_2 CITIES ACROSS EUROPE

VITORIA-GASTEIZ + TARTU + SØNDERBORG

Deliverable 3.8 ICT Infrastructures Deployed and Commissioned. City Information Open Platform in Use

WP3, Task 3.7

Date of document

15/02/2022

Deliverable Version:	D3.8 V2.0
Dissemination Level:	PU[1]
Author(s):	Patxi Sáez de Viteri (MON), Felix Larrinaga (MON/MU), Sofía Mulero (CAR), José Luis Hernández (CAR), Julia Vicente (CAR), Susana Gutiérrez (CAR), Jose Antonio Sanchez (GIS), Josu Rollón (LKI), Asel Villanueva (TEC)

[1] PU = Public
PP = Restricted to other programme participants (including the Commission Services)
RE = Restricted to a group specified by the consortium (including the Commission Services)
CO = Confidential, only for members of the consortium (including the Commission Services)

This project has received funding from the European Union' s Horizon 2020 research and innovation programme under grant agreement No 691883

Document History

Project Acronym	SmartEnCity
Project Title	Towards Smart Zero CO_2 Cities across Europe
Project Coordinator	Francisco Rodríguez Tecnalia francisco.rodriguez@tecnalia.com
Project Duration	1st February 2016 - 31st July 2021 (66 months)

Deliverable No.	Deliverable 3.8 ICT Infrastructures Deployed and Commissioned. City Information Open Platform in Use	
Diss. Level	Public	
Deliverable Lead	MON	
Status	X	Working
	X	Verified by other WPs
	X	Final version
Due date of deliverable	31/07/2021	
Actual submission date	15/02/2022	
Work Package	WP3. Vitoria-Gasteiz Lighthouse Deployment	
WP Leader	VIS	
Contributing beneficiaries	MON, MON/MU, CAR, TEC, GIS, LKI	

Date	Version	Person/Partner	Comments
06/05/2021	REV 00	Patxi Sáez de Viteri (MON)	Document skeleton and ToC definition. Work contribution proposal.
03/06/2021	REV 01	Patxi Sáez de Viteri (MON), Felix Larrinaga (MON/MU)	New ToC structure revised and formulated
11/06/2021	REV 02	Patxi Sáez de Viteri (MON), Jose Hernández (CAR)	Doc combination with CAR contributions
28/06/2021	REV 03	Patxi Sáez de Viteri (MON), Felix Larrinaga (MON/MU), Jose Antonio Sanchez (GIS)	Added contributions by MON, GIS and MON/MU
02/07/2021	REV 04	Patxi Sáez de Viteri (MON)	Contributions and relationships with other WPs in the project
08/07/2021	Rev 05	Patxi Sáez de Viteri (MON), Josu Rollón (LKI), Asel Villanueva (TEC)	Contributions from LKI and TEC added to the main draft
16/07/2021	Rev 06	Patxi Sáez de Viteri (MON)	Final contributions, comments, and recommendations.
29/07/2021	V1.0	Patxi Sáez de Viteri (MON)	Contributions and corrections from reviewers (AGV & VIS)
29/07/2021	V1.0	Silvia Urra (TEC)	Submission to the EC
01/02/2022	V2.0	Patxi Sáez de Viteri (MON)	Modifications and clarifications as requested by the Commission

Table of content:

Table of Tables

Table of Figures

Abbreviations and Acronyms

Table 1: Abbreviations and Acronyms

Abbreviation/ Acronym	Description
AENOR	Asociación Española de Normalización y Certificación
API	Application Programming Interface
ASHRAE	American Society of Heating, Refrigerating and Air-Conditioning Engineers
AVS	Added Value Service
BBDD / DDBB	Database
BEI	Bus Eléctrico Inteligente (Smart Electric Bus)
CAN	Controller Area Network
CICD	Continuous Integration Continuous Development
CIOP	City Information Open Platform
CTN	Comités Técnicos de Normalización
CUPS	Código Universal de Punto de Suministro (Supply Point Universal Code)
DH	District Heating
DoA	Description of Action
DPC/CPD	Data Processing Centre
ESCO	Energy Service Company
EU	European Union
eV	Electric Vehicle
GDPR	General Data Protection Regulation
GFS	Grandfather-Father-Son (long term retention policy)
GIS	Geographic Information System
HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
ICT	Information & Communication Technologies
IP	Internet Protocol
IT	Information Technology
ITU	International Communications Union
JWT	JSON Web Token
KPI	Key Performance Indicator
LCA	Life Cycle Analysis
LPG	Liquefied Petroleum Gas
OGC	Open Geospatial Consortium
PCA	Principal Components Analysis

REST	Representational State Transfer
RSS	Really Simple Syndication
SN	Serial Number
SSD	Solid State Disk
SSH	Secure Shell
SSL	Secure Sockets Layer
UNE	Una Norma Española
URL	Uniform Resources Locator
WPx	Work Package X

0 Publishable Summary

The City of Vitoria-Gasteiz strategy for sustainability and environment protection is well recognised and was awarded European Green Capital in 2010. The involvement of the municipality in SmartEnCity Project as a lighthouse city is an example of its commitment towards energy efficiency, resources management, energy transition, and the improvement of the living conditions of its inhabitants.

A natural move towards a better and more efficient management of resources and services in all aspects of the city brings the municipality closer to a greener and more intelligent city, where the concept of a smart city platform becomes relevant.

During this project, the basic infrastructure to provide the city with the general backbone services and functionality of a city platform was designed, deployed, and commissioned. This backbone or CIOP (City Information Open Platform) serves as the global framework over which any city area will host the necessary intelligent IT infrastructure (or service) to assist intelligent data management and decision support to the city officials and service operators.

This document provides a thorough insight on the architecture deployed, intended as a guide for future replication, deployment, and further developments by advanced ICT skilled staff. A general grasp of the objectives, functionality and way of use can be inferred from the explanations and details contained.

In addition, SmartEnCity has provided the city of Vitoria-Gasteiz with an initial set of Added Value Services to manage and operate those pilot actions developed during the Project and set in place additional tools and services to enable their extension to other districts in the city and other city/area services. Description of these services for experts and ICT architects are included.

This document contains reference information about the Vitoria-Gasteiz CIOP for replication, service, and further development of its functionality.

1 Introduction

1.1 Purpose of this Document

In SmartEnCity, the city of Vitoria-Gasteiz has managed to successfully implement several demonstration actions showing the way to the transformation of the City towards a net zero CO2 city. Among these, a key demonstration action is the construction and commissioning of the CIOP, or Smart City Platform for Vitoria-Gasteiz. This major achievement is the result of hard effort and time on the partners' engineering side to set up the core system to provide IT support for data acquisition, storage, secure access and communications among others, and the ability to host advanced added value services for specific areas of the City.

This is, in essence, the CIOP or Smart City Platform for Vitoria-Gasteiz, commissioned in due time as the main result for Task 3.7 and in this way fulfilling the deliverable D3.8 as specified below:

D3.8	ICT Infrastructure deployed and commissioned. City Information Open Platform in use	5 - MON	Demonstrator	Public	M 66

Therefore, it must be stressed that the actual deliverable D3.8 is the CIOP itself, as defined in the DoA (latest amendment). Thus, this report cannot be held as the deliverable, but a support document for D3.8 so that the platform deployed for Vitoria-Gasteiz in WP3 can be adequately evaluated and reviewed.

This document contains detailed technical information about the architecture, data space and functionality of the core platform services as well as the Added Value Services implemented for the specific applications covered by the project actions. It is intended to be a guide to developers and ICT experts (as declared in 1.2) for the implementation and replication of this experience in other cities or environments where needed, and a supporting document for the Vitoria-Gasteiz ICT department for the maintenance of the CIOP services.

For each Added Value Service, this document contains implementation instructions and technical “users” guidelines which will assist the eventual aforementioned replication, deployment and training for an optimal user experience.

1.2 Target Group

Due to the technical nature of this document, although being public, it will be difficult to understand for the general public, especially for those collectives not well versed in ICT matters. A certain degree of specialisation is necessary for the proper interpretation and use of the information herein contained.

Typically, the profile of the target reader and user for this text, although not excluding, will be the following;

- ICT Engineers
- Software architects & developers
- Skilled programmers
- Communication technicians
- System architects and managers
- ICT systems integrators
- Municipality ICT Technicians
- Etc

1.3 Contributions from partners

The following **¡Error! No se encuentra el origen de la referencia.** depicts the main contributions from participant partners in the development of this deliverable.

Table 2: Contribution of partners

Participant short name	Contributions
MON, MON(MU)	MONDRAGON Corporation described the compliance of CIOP with current national and international standards, as well as the final deliverable recommendations. Mondragon Unibertsitatea (MON/MU) has contributed to the development and deployment of the CIOP the structure of the deliverable. MU has also completed sections 5.2, 6.1, 6.2.1, 6.2.2, 6.2.3 and 6.2.4
GIS	Estudios GIS describes in this document the structure and functionality of the AVSs deployed for public building energy monitoring and the Intelligent bus (BEI) monitoring in sections 6.2.5 & 6.2.6.
LKI	LKS Infraestructuras IT describes in entry 5.1 the IT infrastructure deployed to support the adequate running of services, data storage and system interoperability
CAR	Fundación CARTIF has contributed to the Vitoria-Gasteiz CIOP with the KPI and energy resource planning added value services. The first one linked to the WP7 in order to calculate the KPIs for the city of Vitoria-Gasteiz, while the second more focused on forecasting energy demand for better decision-making. CARTIF has completed sections 6.2.7 & 6.2.8 with de description of the functionality added to the platform.
TEC	The contributions from TECNALIA to the deliverable and this support document are on section 6.2.9 in reference to the GIS3D viewer added value service for the platform.

1.4 Relation to other activities in the project

The CIOP for Vitoria-Gasteiz is built upon the work developed on the first stages of SmartEnCity in WP6, there grounds for the city platforms were established. As defined in that WP and the related deliverables D6.1-D6.7 the generic structure and platform architecture is instantiated in this task T3.7 and comes to reality for the City of Vitoria-Gasteiz in the final deliverable D3.8, to which this document serves as reference and support.

The general systems and services implemented, as defined in WP6, give support to the Added Value Services deployed in this task (T3.7) for the actions carried out in tasks T3.2 to T3.6. Through the CIOP, SmartEnCity provides services to the city like citizen engagement and acceptance (T3.2), measuring the comfort in the dwellings (T3.3), energy consumption monitoring (T3.3), electric bus function overlooking (T3.6), energy demand and consumption forecasting (T3.6), and others. All these services are related to actions executed during the project.

Task T3.5 provides the infrastructure necessary to deploy sensors and systems producing data in the district. Those data are collected in the platform presented in this deliverable. Data are finally used by the different AVS presented in section 6.2. Monitoring is one of the main objectives for those AVS. Data are also used for KPI calculation in WP7.

In addition, the deployment of the district heating in the intervention area (Coronación District) relates also to the monitoring and forecast of the energy demand for the dwellings connected to the DH network. The readiness of the system to uptake data from this DH enables the calculation of short- and medium-term energy demand so that the ESCO providing the service can adequately manage the supply of biomass fuel to the boiler rooms.

Furthermore, these added value services are strongly related to the WP7, where the KPIs were identified (T7.1) and monitoring programmes defined (T7.2). Primarily, the KPI-related added value service provides a digital service where the KPIs are not only automatically calculated, but also visualized to support the evaluation (T7.5) based on collected data (T7.3). As well, other added value services, such as the 3D GIS, rely on the selected KPIs for Vitoria-Gasteiz.

2 Objectives and expected impact

2.1 Objective

The objective of this document is to shed light over the massive work developed during the deployment of the CIOP which is usually not visible to the user/evaluator, but it is necessary to support all the functionality deployed.

As a floating iceberg, the amount of coding, services, systems, databases, communication links, protocols and other elements conforms the backbone of a highly complex system defined to provide functionality today, and enough flexibility to accommodate future systems and services, as well as future IT technologies, yet to be designed.

To serve as reference, the description of the added value services and functionality is illustrated with screen captures and diagrams to help the user anticipate what the CIOP and a particular service will look like and will offer.

2.2 Expected impact

The deployment of the CIOP in Vitoria-Gasteiz is expected to have a manifold impact in several areas. Although not limited to the ones mentioned below, it is considered to be the catalyst to improved advanced services built on top of the smart platform.

One of the most interesting aspects of this deployment is the additional support offered to the city for the design, development, and deployment of its smart city strategy, which is currently under consideration in the municipality. The CIOP should be considered as a powerful tool upon which build this strategy.

In line with this, the platform offers the possibility to deploy the municipality open-data policies to help build additional data-intensive services for the benefit of all citizens.

Additional economic development may be expected for IT companies offering these added value services where solutions appear as a response to city and citizen demands. Subscription services based on real time data usually emerge when value is offered to the customer.

For the city, the smart platform or CIOP means the opportunity to have a central intelligence and data storage system to help build new services, generate additional cross sectorial decision support systems, real time data availability for better city management and the opportunity to bring the citizen updated information and communication channels, to improve their quality of life.

3 Overall Approach

3.1 Dependencies with other deliverables

The following Table 3 depicts the main relationship of this deliverable to other deliverables developed within the SmartEnCity project and that should be considered along with this document for further understanding of its contents.

Table 3: Relation to other activities in the project

Deliverable Number	Contributions
D3.1, D3.2, D3.3	Building retrofitting interventions completed
D6.1	General functional and non-functional specifications for a smart city platform
D6.2	CIOP Reference Architecture
D6.3	Data Model Architecture Implementation
D6.4	Interoperability Mechanisms Implementation
D6.5	Design Guide and Tool Catalogue
D6.6	Strategy for Value Added Services
D6.7	Integration and Validation Report (CIOP)
D7.2	Definition of the KPIs for each of the intervention pillars
D7.3	Evaluation procedures and methodologies with selected KPIs for each of the cities
D7.6	Monitoring plan for data collection in building related
D7.7	Monitoring plan for data collection in mobility related interventions.
D7.8	Monitoring plan for data collection in integrated infrastructures (i.e., District Heating) related interventions.
D7.9	Data collection approach as support for KPI calculation methods.

4 CIOP Reference Architecture

4.1 Approach

4.1.1 Compliance with UNE 178104

The Smart City Platform for Vitoria-Gasteiz is based on the extraordinary work developed by the normalisation association in Spain *Asociación Española de Normalización y Certificación* - formerly AENOR (now UNE).

The UNE committee CTN 178 is dedicated to de development and revision of a series of norms for Smart Cities, which account for 31 in total;

- Infrastructures and city platforms (subcommittee 1 – SC1)
- Indicators and Semantics (SC2)
- Mobility and Transport Platforms (SC3)
- Sustainability (SC4)
- Tourist Destinies (SC5)
- Lan Planning and Public Services (SC6)

Additionally, CNT 178 has strong international relations with other normalisation bodies, working together in several fora;

- CEN/TC 465 - Intelligent and sustainable intelligent cities
- CEN/CLC/ETSI/SF-SSCC - Sectorial Forum. Intelligent and sustainable intelligent cities
- IEC/SYCSMART_CITIES SyC - Electrotechnical aspects of smart cities
- ISO/TC 268 - Sustainable development of smart cities
- ISO/IEC/JTC 1/WG 11 - Smart Cities.

UNE's CTN 178 has brought to international consensus at the International Telecommunications Union (ITU) for its work on interoperability requirements:

- ITU-T Y.4201 "High-level requirements and reference framework of Smart city platform
- ITU-T Y.4200 "Requirements for interoperability of Smart city platforms[2]

[2] https://www.itu.int/itu-t/aap/AAPRecDetails.aspx?AAPSeqNo=7969

Based on this work, and the international recognition gained by CTN 178's normalisation work[3], the architecture for the CIOP was based on UNE 178104:2015. At the time of the drafting of this report, the current version of this norm is *UNE 178104:2017: Integrated Management Systems for SmartCities. Interoperability Requirements for a Smart City Platform*[4].

This is so from the initial stages of the definition of the platform architecture develop during WP6 and agreed for the implementation of the CIOPs in all three light house cities in SmartEnCity Project. Further in-depth information can be found on WP6 deliverable set, and in particular in D6.2. CIOP Architecture Generic Implementation.

The following picture summarises the general high-level architecture proposed by the CTN 178 and the UNE 178104 standard, which is developed in the above-mentioned deliverable, section 4.2 Architecture Design.

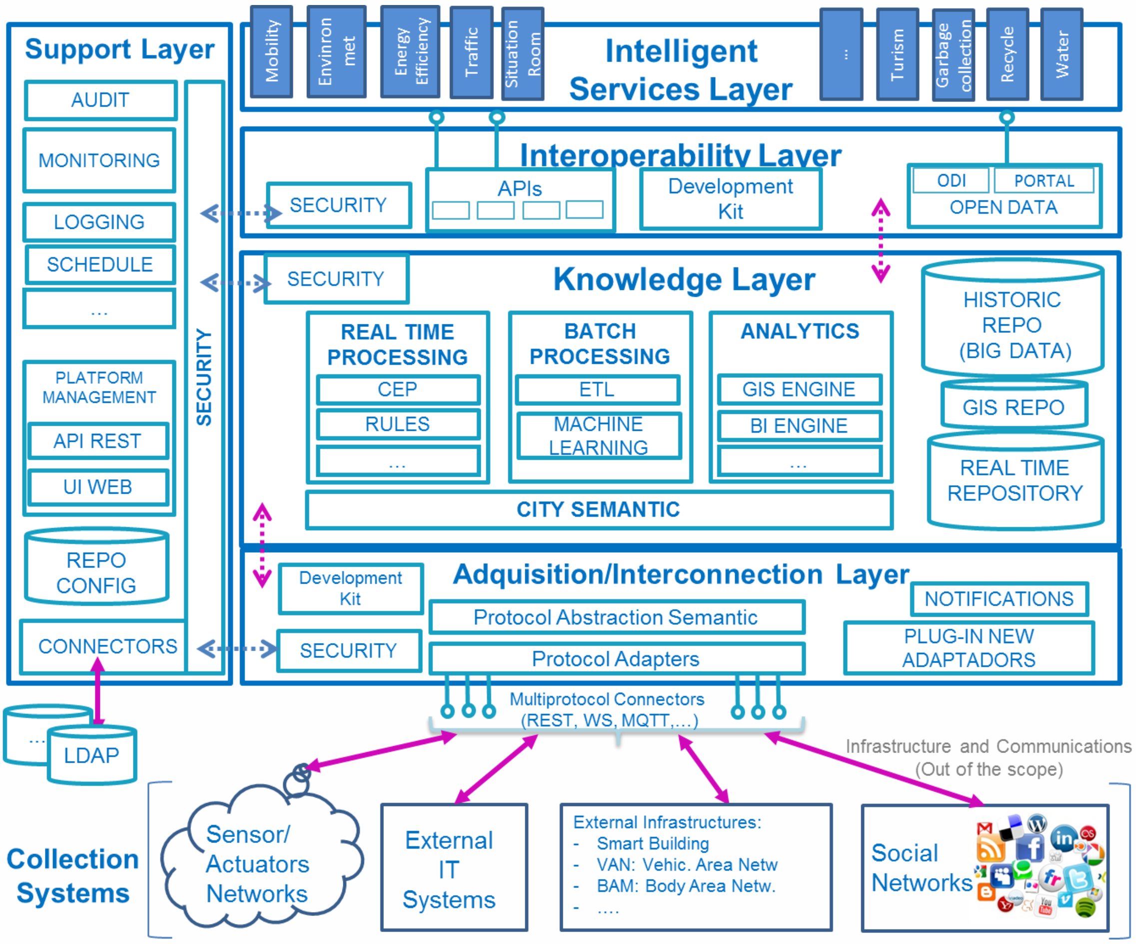


Figure 1: Smart City Platform General Structure (source UNE)

---

[3] https://www.red.es/redes/es/actualidad/magazin-en-red/espa%C3%B1a-logra-el-consenso-internacional-en-la-estandarizaci%C3%B3n-de-la

[4] https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma/?c=N0059471

5 CIOP Infrastructure & General Platform Services

5.1 *Servers' infrastructure*

This section outlines the hardware infrastructure deployed in the CIOP system that s the AVSs. We will describe the physical characteristics of the datacentre and the hardware as well as some technical details of the virtualised systems.

5.1.1 Datacentre

This hardware infrastructure is located in a multioperator TIER 4 compliant datacentre located in building 809 of the Bizkaia Technology Park, in the town of Derio (Bizkaia). It is strategically located to have absolute accessibility of media and people as well as infrastructures and telecommunications connection. 5 minutes from Bilbao airport (Derio) and 15 minutes from Bilbao, access to motorways and ring roads is about 400 meters away.

The Data Centre consists of approximately 20,000 m2 built, divided into four floors, with the following uses:

Figure 2: Datacentre at Bizkaia Technology Park

- **Basement:** dedicated mostly to parking staff vehicles, it also has the technical rooms of the building (general picture, UPS room, battery room, firefighting room,).
- **Floor -1:** Personnel offices, Data Processing Centre (DPC) or Data Centre.
- **Floor** 0: Personnel offices, technical rooms and telecommunications laboratories and assembly hall.
- **Floor** 1: Staff offices

The rooftop is used to locate the refrigeration groups, air conditioners and generator sets.

The DPC is located on the -1 floor; plant that has semi-basement characteristics. It consists of a total of 850 m2, divided into three independent rooms of 250 m2, 350 m2 and 260 m2. The proposed location is taking advantage of part of the first room although for future expansions there is availability in the remaining two rooms.

Possibilities of natural disasters

In 2002 a study was carried out to see the probability that this building, then under construction, suffered damage from natural disasters.

Although there is a water stream in the vicinity, already in project the building was located outside the influence of possible floods. There is a study carried out by the Basque Government Water Service in which it can be seen that the building is outside the line that indicates the probability of flooding in 500 years.

As for seismic disasters, Bizkaia is one of the areas with the lowest seismic probability in the entire State, as indicated in the Spanish earthquake-resistant Standard NCSE-02. The construction project of the building considers the guidelines indicated by this standard.

With regards to other types of natural disasters, such as high-speed winds, heavy snowfall, etc..., the building has been designed in accordance with the Spanish standard of actions in the AE-88 building.

Electromagnetic interference

There are no major sources of electromagnetic interference generation in the vicinity or within the building itself such as induction furnaces, arc furnaces, electrical substations, ...

Experience, after 9 years of continuous operation of the building, indicates that no event caused by this type of phenomenon has been recorded, which supports the absence of such interference.

Activities generating industrial pollution and pollution.

The building is located in the Technological Park of Bizkaia. This is an area where most of the services companies reside and with a significant high-tech component. There are no companies in the vicinity whose activity generates pollution and/or pollution worthy of taking into account.

Vibration sources

The building is away from major heavy traffic roads or railway lines.

It should be noted that despite the proximity of Bilbao airport, in particular the building is located 2 km from the head of runway 30 of the airport.

The acoustic levels that are perceived inside the building by this proximity are invaluable, not generating discomfort to employees in the offices. No aircraft vibrations have been detected.

Upcoming risk activities

As described above, the building is located in the Bizkaia Technology Park. It is an area of technological companies, hence there are no activities with industrial, political, mass concentration risk ...

Insulation of facilities with respect to nearby buildings

The building, being located within the Technology Park, shares the facilities that the Park supplies. In our case, only the hydraulic networks (supplies, rainwater, fecal and firefighting) and part of the telecommunications pipelines are shared (only the pipeline, not the optical fiber). The rest of the facilities are owned by Euskaltel, except of course for the electricity supply that is provided by Iberdrola.

Structural quality of installations

The building containing the DPC is of recent construction (2003). It is a building made of reinforced concrete, and prepared, in the case of the DPC, for an overload of use of 800 Kg / m^2 on average (in specific areas this load can be exceeded).

Accessibility and response time of emergency services

Given the location of the Technology Park next to the Txorierri corridor, which is part of the Bilbao ring road, it can be considered that the accesses are fast and well connected.

The main center of the Ertzaintza (Basque Autonomous Police) is located about 8 km along the same corridor of the Txorierri, with an estimated arrival time of 10 minutes.

There is also a Fire Station located directly in front of the building. The arrival time is less than 5 minutes from the service request, as could be verified in the last drill performed.

Availability of suppliers

The electricity supply is made through Iberdrola in medium voltage (30 Kv) from the Asúa substation, about 10 km away. The supply is in ring. The maximum capacity for which the building is prepared is 3,000 Kw, which includes the consumption of the DPC and the offices. Today, the maximum peaks detected are at 800 Kw, so the availability for growth is very high.

Seniority of public services

The Technology Park was constituted as such in 1985. All the infrastructures located there are therefore less than 35 years old.

Citizen security

Crime in the Park is practically zero, being located outside the urban core or residential area. In addition, the Technology Park itself has its own 24-hour security and surveillance service.

Some technical characteristics

soil	• Typical soil load value: 800 kg/m2 • Tile dimension: 60x60 cm • Room height: 2.75 m • Technical ground height: 0.45 m
air conditioning	• Ambient temperature and humidity control systems. • Average temperature: 21º (17º - 25º). • Relative humidity: 50% ± 15%
energy	• Electric ring connection to 4.7 MVA buildings. • Redundant UPS system of 2x400 KVA, expandable to 3x400 KVA, working in parallel. • Two emergency generator sets, with a minimum of 40 hours of autonomy. • Two independent electrical energy distribution tables. • Grounding and protection against power surges.
safety	• Access: 24 hours a day, 7 days a week. • Door opening system using proximity cards. • Restricted access to registered MCC TELECOM personnel. • The building is equipped with the BMS system for continuous monitoring of general alarms. • Access control, surveillance and remote visualization particularized for the ECOLEC cage.
Firefighting system	• VESDA (Very Early Smoke Detection Alarm) fire detection system in building 809. • Argon gas firefighting system.

5.1.2 Computing, storage, and backup

A high availability hardware architecture has been deployed for this project. A computer farm has been created with the following resources:

- 512 GB RAM
- 8 CPUS of 12 CORES
- 12 GB Dual Fibre Card
- 8 GB Ethernet ports
- 10 GB Ethernet ports
- Dual power supply

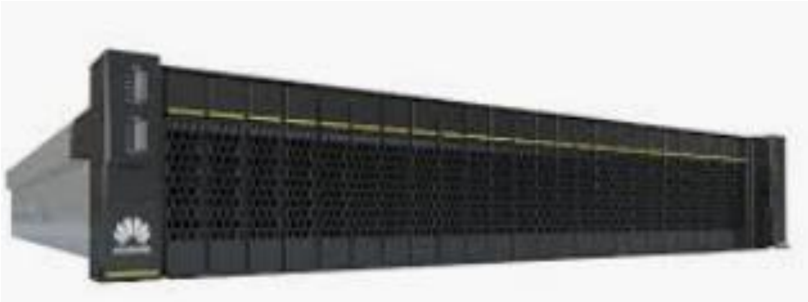


Figure 3: Computing farm detail

The storage needs that we plan to cover throughout the project have led us to look for a system with different types of disks and high growth capacity. A Dual controller solution has been implemented, with SAS, SSD disks, which is capable of supporting the non-relational databases involved in the system.

Figure 4: Storage Systems

The main storage is distributed under a 64 TB system, with SSD, SAS, SATA disks being applied on the quality-of-service system (Disk Tiering)

Mapped against this system we have the Huawei HOST that support the main architecture of the system and the SmartEnCity project.

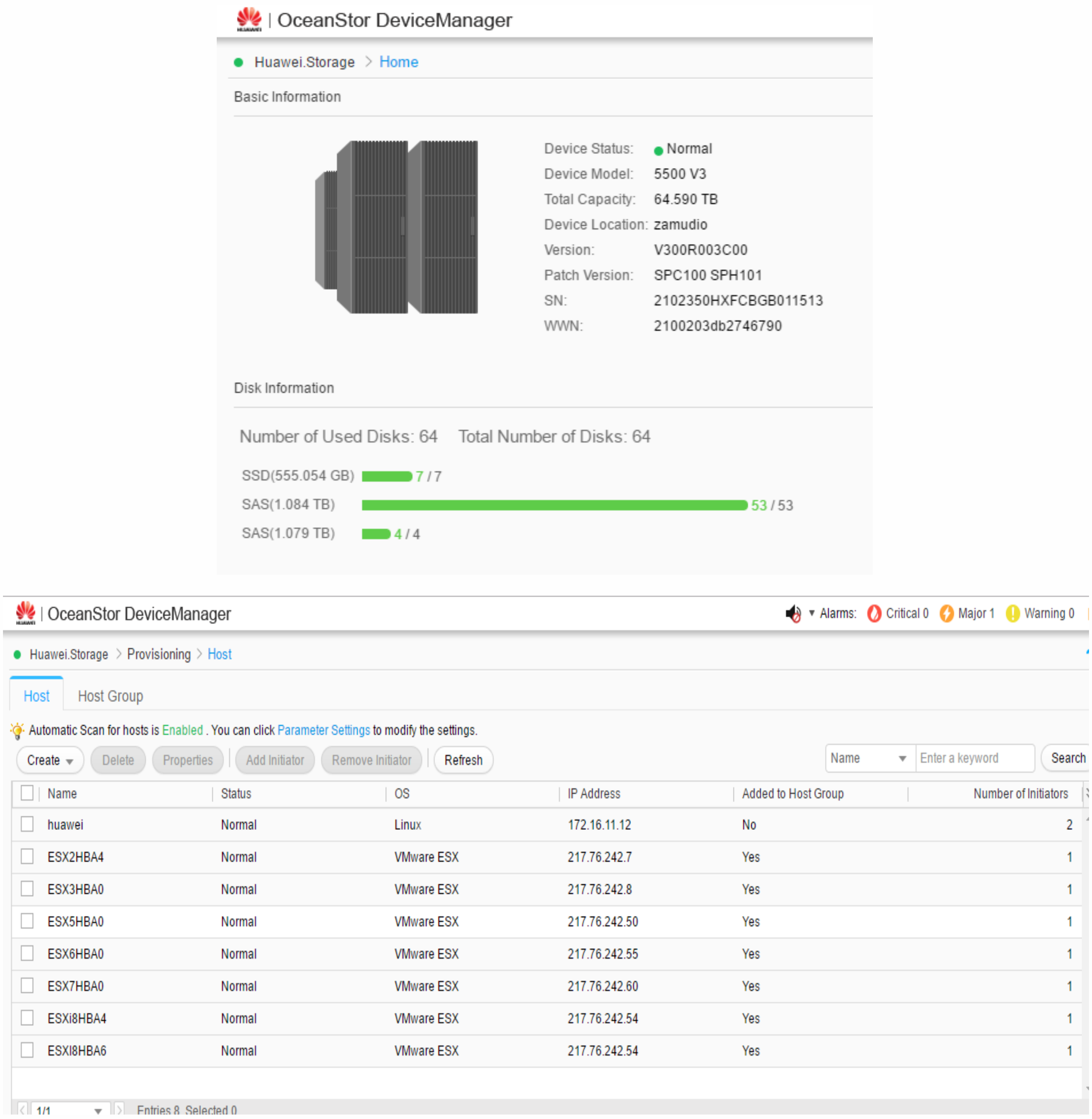


Figure 5: Device Manager software

The backup system is based on a **3-2-1 policy**, so it backs up to three formats: Fast Disk, Slow Disk, Tape

- Fast disk backup is performed on a 10 TB storage unit based on an IBM V7000 on SAS disk.
- Slow disk backup is performed on a 60 TB storage unit based on an IBM V7000 on SATA disk
- Tape backup is performed against a Quantum I80 system, whose image is as follows.

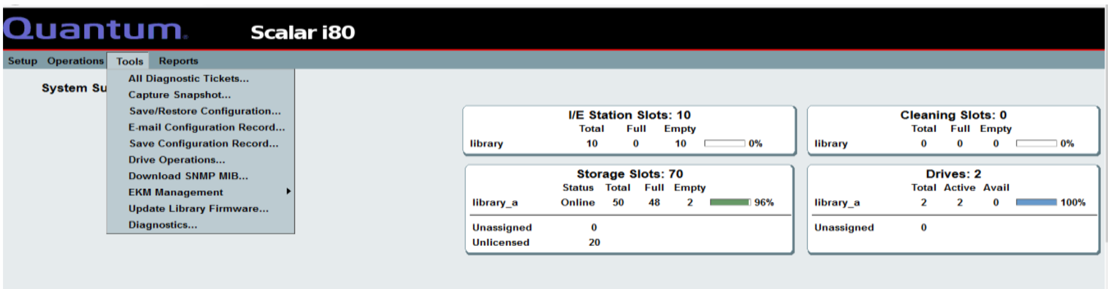


Figure 6: Backup configuration

The backup system is managed under the **Veritas Backup Exec** software. This software implements a GFS backup policy with daily, weekly, and monthly backups on all three types of storage. Storing copies with greater annual retention.

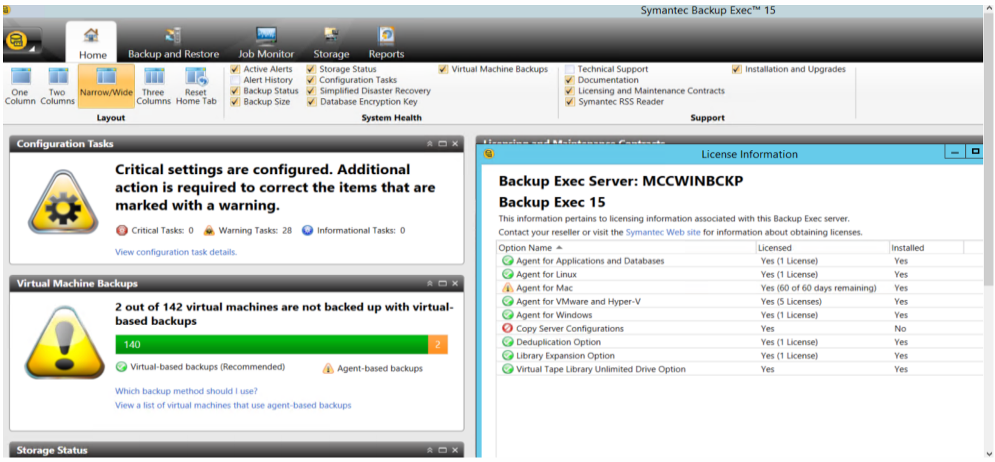


Figure 7: Backup management system

5.1.3 Networking

A security perimeter has been implemented with a Firewall solution based on a **Fortinet FortiGate 500D cluster**, which operates in several layers and also has an isolated network section for the solution.

Figure 8: Network hardware detail

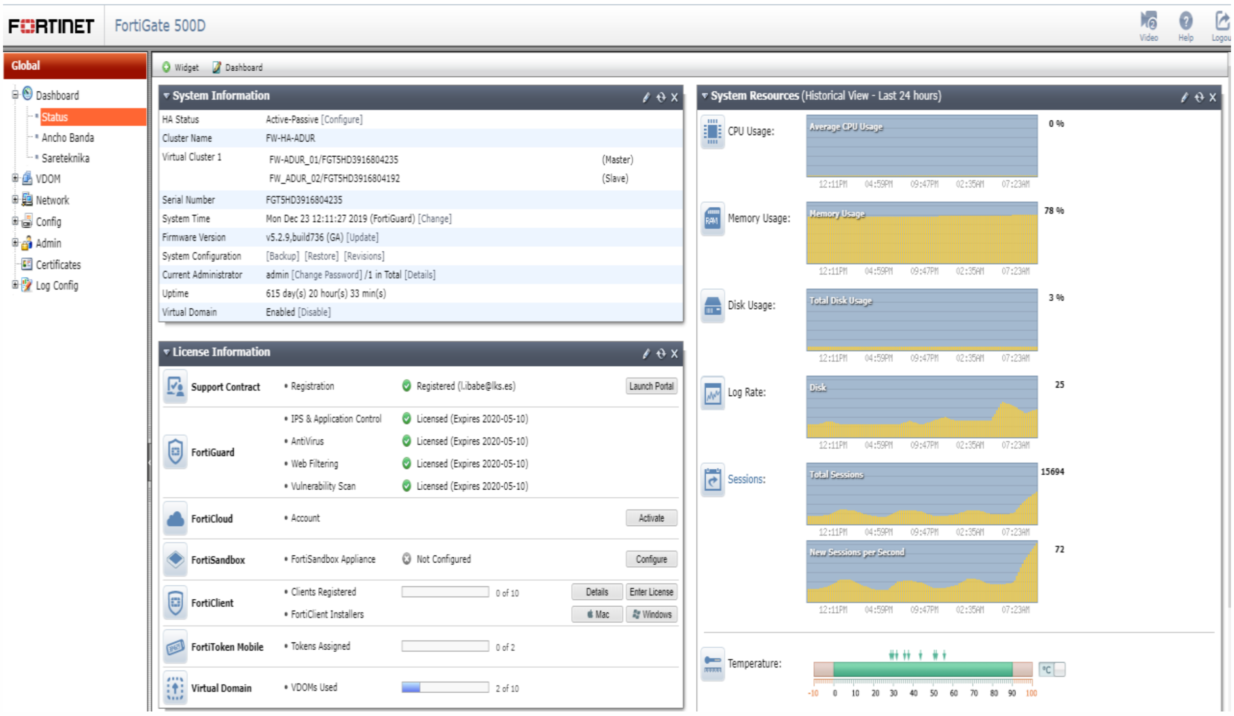


Figure 9: Fortinet management system

5.1.4 Hardware Summary

COMPUTING	STORAGE	BACKUP	NETWORK
2 Huawei 2288 V5* 3 IBM X3755 1 HP Proliant DL385 2TB MEM/398 GHZ	OCEANSTORE 5500 V3 64TB * STORWIZE V7000 64TB 128 TB	QUANTUM DXI 5700 10 TB	4 SWITCH CORE HUAWEI 5730 2 FORTINET FORTIGATE 500D * 196 PTOS

* Hardware cofounded by SmartEnCity project.

5.1.5 Virtual Machines

Table 4:Virtual Machines specifications[5]

Name	SmartEnCity_WSO	SmartEnCity_APPS	SmartEnCity_BBDD	SmartEncity_GIS	SmartEnCity_KPIplan	SmartenCity_TEST	pmccesxlnxcoronacion	pmcces xbdelk	pmcces xbdhw	pmccesx bdhw1	pmccesx bdhw2	pmccesx bdhw3
IP Private	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]					
IP Public		[redacted]		[redacted]	[redacted]		[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
Network mask	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]						
Gateway	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]						
Ports		[redacted]		[redacted]	[redacted]		[redacted]					
OS	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]
Memory (Gb)	8	4	4	4	8	8	4	8	20	10	10	20
Storage (Gb)	128	128	256	50 100	200	128	61	30	100	100	100	100
vCPU	2	2	2	2	4	2	1	2	4	4	4	4
Notes	Server managed by MGEP Alain Perez <aperez@mondragon.edu>	Server managed by MGEP Alain Perez <aperez@mondragon.edu>	Server managed by MGEP Alain Perez <aperez@mondragon.edu>	Server managed by TECNALIA José Luis Izkara <joseluis.izkara@tecnalia.com>	S Server managed by CARTIF José L. Hernández <josher@cartif.es>	Server managed by MGEP Alain Perez <aperez@mondragon.edu>	cluster Hortonworks	cluster Hortonworks	cluster Hortonworks	cluster Hortonworks	cluster Hortonworks	cluster Hortonworks

[5] Being this a Public Deliverable, some operational data like IP addresses and access ports must be kept confidential for security reasons.

5.2 General Platform Services

This section outlines services deployed in the CIOP platform that support AVS. These services provide security, persistency (storage), software development capabilities or functionality consumed by AVS. Stakeholders do not directly consume these services, but they are essential for the correct functioning of the CIOP platform. The section first presents the architecture and, after that, a short description for each service is given. Services are presented according to a common template where a description is given first, followed by the architecture of the service, the data structure, the tools employed and finally a short guide indicating the main functions is presented.

5.2.1 Architecture (service map)

The architecture for these services is shown in Figure 10. These support services are installed in 4 servers. The first one (APP (23)) provides services related to access and security. The second one (BBDD) offers storage capabilities. WSO (20) offers Node-RED which is a programming tool used in the project to build the back end for many of the AVS tools. Finally, the Test (22) server includes several services to support AVS and to test the solution. The detail for these services is given next.

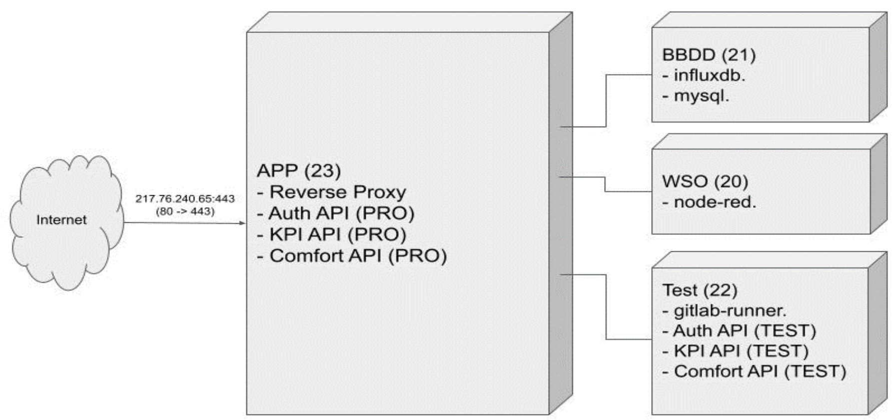


Figure 10: General platform service distribution

5.2.2 Reverse Proxy (NGINX)

5.2.2.1 Description

This service is the entry point for all the applications deployed on internal servers. In this way, we have a single access point with a single public IP. This allows us to redirect a URL to a specific service (e.g. https://vitoria-gasteiz.smartencity.eu/nodered redirects to the NodeRED service deployed on another server). All services whose URL begins with https://vitoria-gasteiz.smartencity.eu go through this reverse proxy.

In addition, this service also encrypts all the HTTP calls with SSL, so all the external connections are encrypted. In this way, we ensure that if any malicious attack intercepts a packet, it is unreadable to the attacker.

5.2.2.2 Architecture definition & Infrastructure

It is a simple service. No architecture definition is required.

5.2.2.3 Data Structure

This service has a configuration file where all redirections are configured. No additional data structures are required for its operation. No data storage is performed for this service.

5.2.2.4 Tools

This service is deployed using docker. It deploys a NGINX server, and the SSL certificate is automatically generated by an external service called Let's encrypt.

5.2.2.5 User guide (Location, functions, etc.)

The container with this service is deployed on the APP server. The container is called **sec_letsencrypt** and you can check if it is running by running the **docker ps** command. The configuration file is mounted using a docker volume, so it can be modified to add/edit/delete new redirects. Once the file has been modified, the container must be restarted to apply the changes (**docker restart sec_letsencrypt**).

5.2.3 Auth API

5.2.3.1 Description

Some of the applications in this project use JWT authentication to authenticate the users and for permission management. This API enables multiple services to use the same authentication mechanism.

5.2.3.2 Architecture definition & Infrastructure

It is a simple service. No architecture definition is required.

5.2.3.3 Data Structure

Figure 11 presents the entity-relationship diagram used to manage the tokens, permits, roles and user information for this service

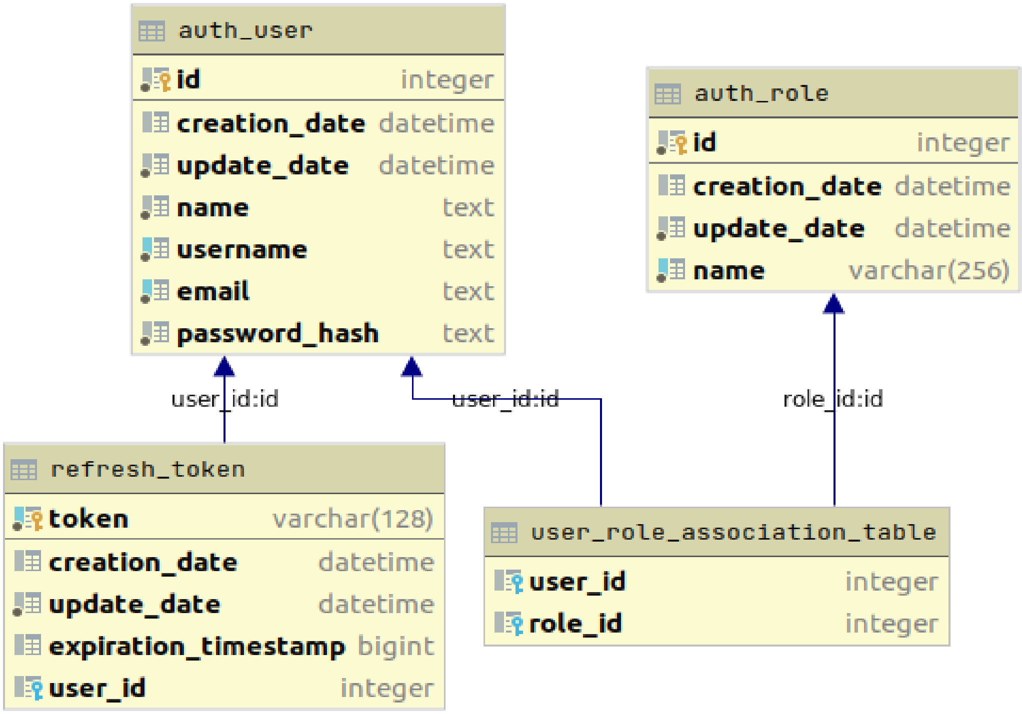


Figure 11: Auth database entity-relationship diagram

The database of the Auth API service contains 3 main tables:

1. Auth_user: This table stores the list of the users of the application.
2. Auth_role: This table stores the different roles of the applications, which determine if a user has permissions to perform an action. For example, if a user does not have the INST role, they will not have access to the Installations application.
3. Refresh_token: JWT should have a short lifespan, therefore refresh_tokens are created to maintain a "session" opened for a long period of time. This enables us to invalidate old JWTs quickly and therefore disallowing the access to any attacker that has stolen a JWT token.

The additional table is an association table that stores the N-to-N relationship between the roles and the users of the applications.

5.2.3.4 Tools

This service is developed using the python programming language along with the Flask and SQLAlchemy frameworks. No passwords are stored in plain text, they are all encrypted using the BCrypt algorithm.

The service is deployed using docker containers and uses an SQLite Database deployed inside the container.

It has 2 replicas, one for production and the second one for the test environment. This allows to have different usernames/passwords for each environment and separates the security risks.

5.2.3.5 User guide (Location, functions, etc.)

The container with this service is deployed in 2 servers:

1. Production Auth API: App server.
2. Testing Auth API: TEST server.

The container with this service is deployed on the APP server. The container is called **sec_auth** and you can check if it is running by running the **docker ps** command. The database is mounted using a docker volume, so the container can be redeployed with no problems.

This application is developed using continuous integration and continuous development methods (see GitLab Runner service for more information).

5.2.4 GitLab Runner

5.2.4.1 Description

Some services and applications developed in this project are developed using Continuous Integration Continuous Development (CICD) methods. Their code is stored in Git repositories so their modifications can be tracked, analysed, and restored if needed. GitLab Runner takes the modifications of these projects and performs different actions with them (depending on a file in the repository named gitlab-ci.yml):

- Build the application.
- Generate a docker container with the built application.
- Deploy de application in a server.

5.2.4.2 Architecture definition & Infrastructure

It is a simple service. No architecture definition is required.

5.2.4.3 Data Structure

No data storage is performed for this service.

5.2.4.4 Tools

No additional tools are used.

5.2.4.5 User guide (Location, functions, etc.)

In order to use the runner, it must be configured in the repository and gitlab-ci.yml file must exist there. To deploy the application, SSH keys must be stored also as CICD variables.

The code pushed in "test" branch will be deployed in "TEST" server and the code pushed to the "master" branch will be deployed in "APP" server.

5.2.5 Node-RED

5.2.5.1 Description

Node-RED is a programming tool for wiring together different functionalities, such as database access, API calls... We use it for different calculations and functionalities among the project:

- Offers an API that collects real time data and stores it in a real time repository (InfluxDB).
- Offers database access to different applications (such as HMI or Installations).
- Triggers repetitive calculations:
  - Aggregating measurement data hourly, extracting it from real time repository and storing it in structural repository.
  - Updating energy and comfort KPI data, extracting it from structural repository and storing in the KPI database.
  - Checking if any household has outdated data and notifying it creating an incidence.
- When a new household is registered, a subscription is created to receive the streaming data of its sensors.

5.2.5.2 Architecture definition & Infrastructure

It is a service to create AVS. No architecture definition is required at this point. See AVS.

5.2.5.3 Data Structure

Data structures are presented for each of the AVS where Node-RED is used.

5.2.5.4 Tools

This service uses docker to deploy a container with Node-RED in it.

5.2.5.5 User guide (Location, functions, etc.)

The container is deployed in WSO server. Docker ps command can be used to check if it is working correctly. https://vitoria-gasteiz.smartencity.eu/nodered URL can be used to access Node-RED interface. It has different flow tabs for the different functionalities. All the flows and the configuration are deployed using docker volumes so the container can be removed and updated without losing the data.

5.2.6 InfluxDB

5.2.6.1 Description

InfluxDB is a Time Series database. It is used to store the real time data of the sensors installed in the households. This NoSQL database is able to store a large amount of data and querying in a fast way.

5.2.6.2 Data Structure

Influx DB stores a single Measurement "table", called forwarder_data. The structure of the table is the following:

- Time: the timestamp when the measurement was done.
- Tags:
  - SN: The serial number of the gateway that is sending the measurement.
  - Channel: The channel where the sensor is configured in the gateway.
  - Device: the id of the device that has measured the measurement.
  - Medium: If the measurement is temperature, humidity, CO_2 or power.
- Value: The value of the measurement.

5.2.6.3 User guide (Location, functions, etc.)

The container is deployed in DDBB server. Docker ps command can be used to check if it is working correctly. The data is stored in a volume so it can be restarted without losing the data. The database cannot be accessed directly from outside the internal network.

5.2.7 MySQL

5.2.7.1 Description

MySQL is a relational database, and it is used to store aggregated and structural data for this project. It gives content to the household monitoring app and installations app. It is also used as the source data to calculate comfort and energy KPIs.

5.2.7.2 Data Structure

Its data structure can be seen in section 6.2.1.3 ("Monitoring of consumption and comfort conditions at home App" data structure. The same structure is also used in other AVS.

5.2.7.3 User guide (Location, functions, etc.)

The container is deployed in a DDBB server (IP 10.230.0.21). Docker ps command can be used to check if it is working correctly. The data is stored in a volume so it can be restarted without losing the data. The database cannot be accessed directly from outside the internal network.

5.2.8 KPI API

5.2.8.1 Description

This API is used to store data related to KPIs. The flow of energy and comfort data is the following.

- Node-RED is used to get real time sensor measurements and store it in the Real Time repository (InfluxDB).
- Every hour, Node-RED queries the real time repository and aggregates data to store it in the structural repository (MySQL).
- Node-RED queries the structural repository (when it is repeated depends on the KPI), aggregates the data and sends it to this API so it can be accessed from the KPI HMI for its visualization.

5.2.8.2 Architecture definition & Infrastructure

It is a simple service. No architecture definition is required.

5.2.8.3 Data Structure

The main tables for this database are the following (see Figure 12):

1. KPI: It stores the definition of a KPI and it is used to group KPI measures.
2. Kpi_type: It is used to group KPIs in groups.
3. Measure: It stores the value of a KPI in a timestamp.
4. Unit: It defines the unit of a KPI.

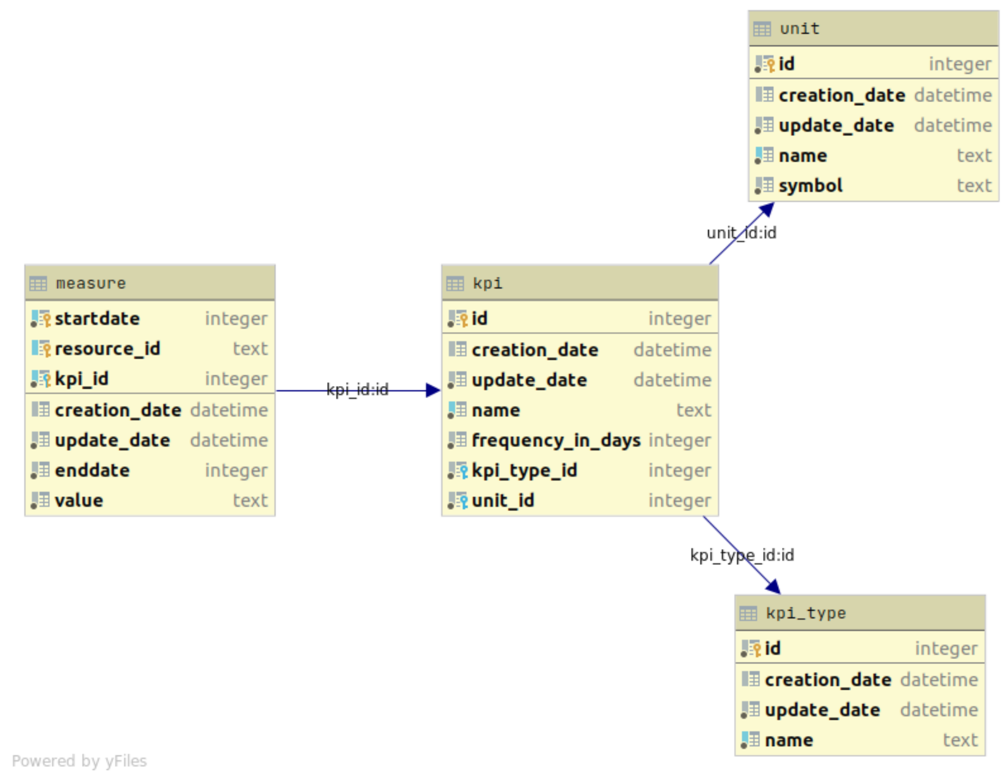


Figure 12: KPI API entity-relationship diagram

5.2.8.4 Tools

The API is developed using the python programming language along with the Flask and SQLAlchemy frameworks. It uses SQLite database, and it is deployed in a docker container.

5.2.8.5 User guide (Location, functions, etc.)

Access to the data storage is offered by means of a REST API where resources can be consumed using web services. This API uses the CICD techniques explained in the GitLab Runner service.

5.2.9 Comfort API

5.2.9.1 Description

This API receives a comfort measurement (temperature and humidity) and returns 2 comfort values data regarding to the ASHRAE standard:

- Expected Average Vote: what would be the average comfort data rating for the given temperature and humidity, being –3 very hot, and 3 very could (0 is the optimal value).
- Percentage of people in discomfort: a percentage value that represents the amount of people that would be in discomfort in the given temperature and humidity values (0 would be the optimal value).

5.2.9.2 Architecture definition & Infrastructure

It is a simple service to support other AVS. No architecture definition is required.

5.2.9.3 Data Structure

No data storage is performed for this service.

5.2.9.4 Tools

The API is developed using the python programming language along with the Flask and framework. It is deployed in a docker container.

5.2.9.5 User guide (Location, functions, etc.)

As for the KPI API this service offers web services using a RESTful architecture where resources can be consumed. This API uses the CICD techniques explained in the GitLab Runner service.

6 CIOP Portal & AVS Implementation

6.1 CIOP Landing application and access page

6.1.1.1 Description

This Web Application is the landing page for the CIOP portal. From this application the rest of Added Values Services can be accessed. Its main function is to provide general information about the project and connect to the utilities the portal offers. The objectives of this web application are:

- Present general information about the project expressed in KPIs.
- Provide general information about the AVS applications available.
- Provide access to those AVS applications.
- Present the information in three languages (English, Spanish, and Basque)

6.1.1.2 Architecture definition & Infrastructure

The web application is built with Vue.js. Vue.js is a JavaScript based framework used to build friendly interfaces for responsive web solutions. This means that the solution adapts to the resolution of the device consuming the page. Thus, we can consume the application from multiple devices.

The requirements for data stored in the platform are little for this application since only collects some KPI information to display in the main page. That information is provided by a simple backend conformed by Node-RED as programming tool and MySQL as the database.

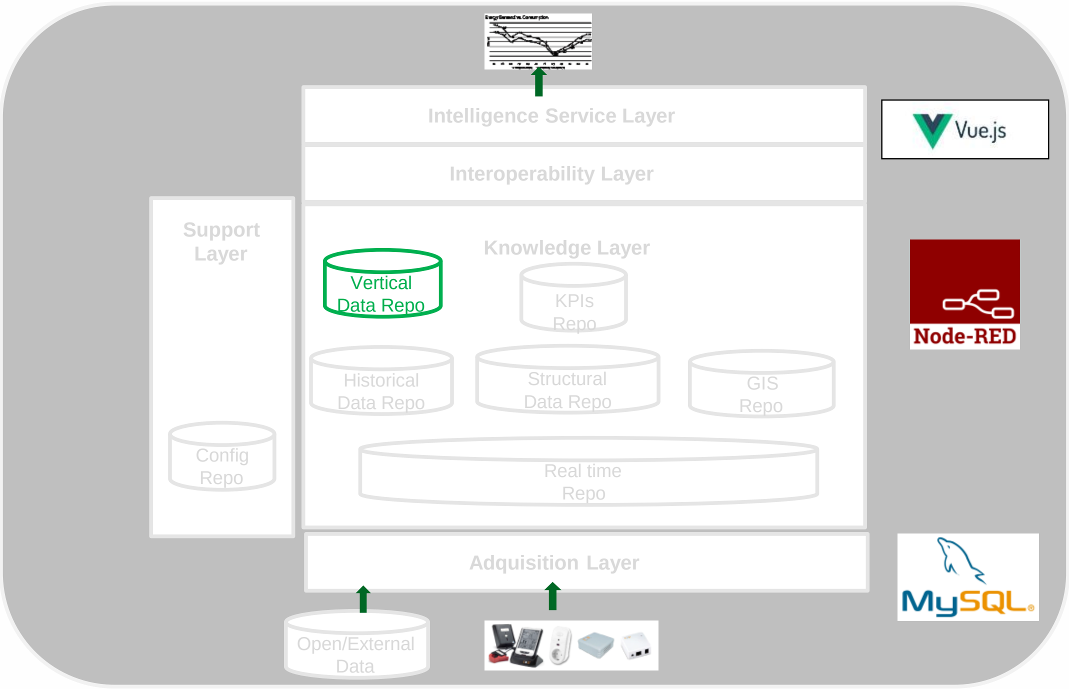


Figure 13: Software Architecture for CIOP Landing Application

6.1.1.3 Data Structure

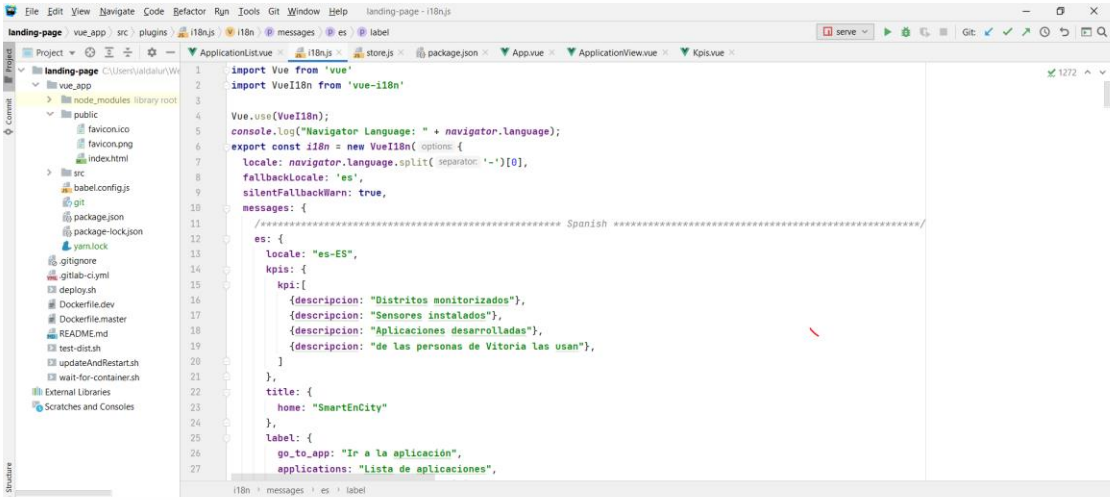
MySQL: is an open-source relational database management system that for this application is used to collect general KPI information. This application uses the KPI API service presented in section 5.2.8. Details for the KPI data structure and the entity-relationship diagram are given in section 5.2.8.3.

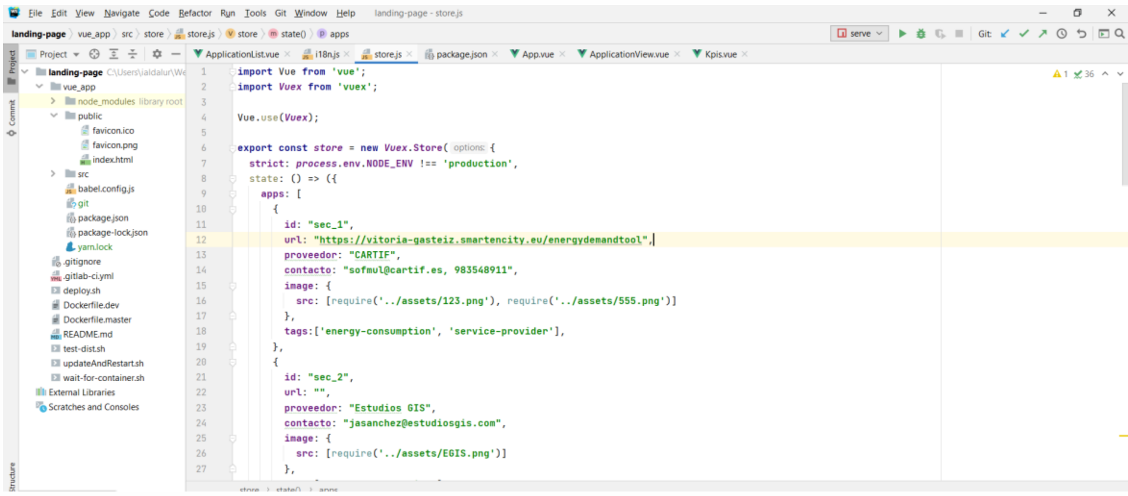
6.1.1.4 Tools

Two programming tools are used to develop this AVS.

Node-RED This programming framework is used to develop the backend for this application. Details for the tool are given in section 5.2.5.

Vue.js: This JavaScript based framework is used to build the front-end for the application. The tool improves the presentation of content enabling friendly HMI. Figure 14 shows different fragments of VUE.js code used to develop the CIOP landing application.





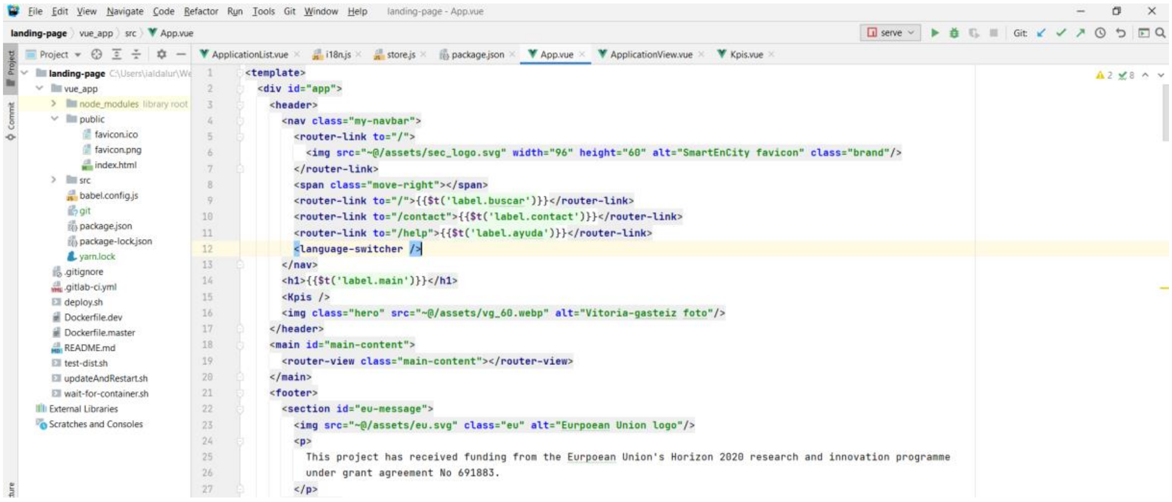


Figure 14: Code fragments in VUE.js

6.1.1.5 User guide (Location, functions, etc.)

The tool can be accessed using the following link https://vitoria-gasteiz.smartencity.eu

The tool is open and does not require credentials to access the information.

The landing page presents general information about the project in the form of KPIs (see Figure 15) and offers links to search for information, contact page managers, get support, and change language.

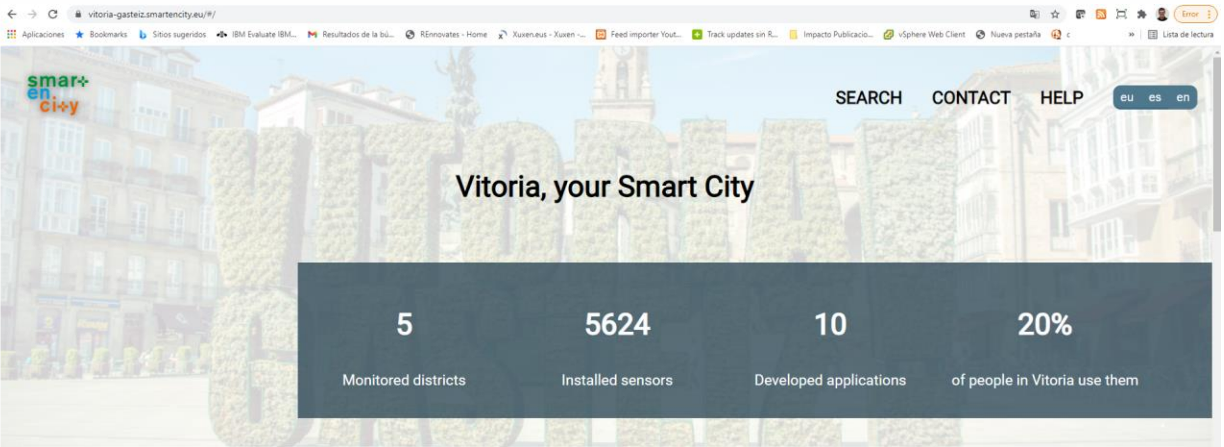


Figure 15: Portal Landing Page (development version)

Scrolling down the page information about all the AVSs available in the portal can be found. Figure 16 shows the list of apps available and a summary of their description. Each AVS is categorised by means of tags that enable accessing similar content or applications (same category).

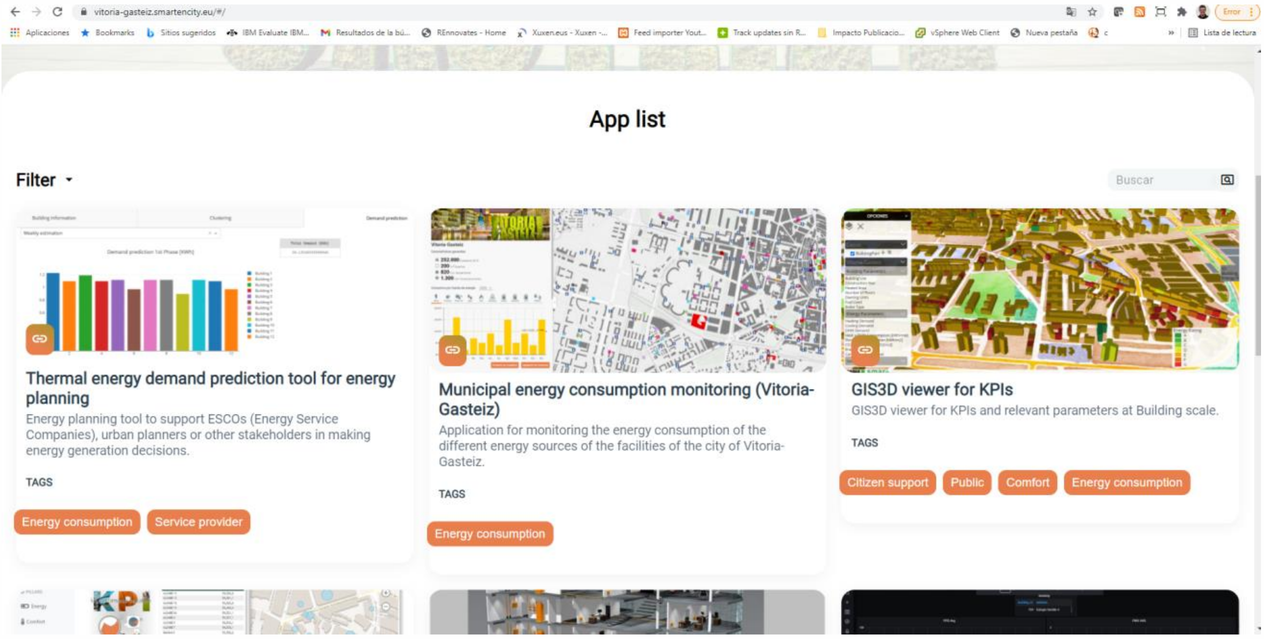


Figure 16: AVS list

By selecting one of the blocks, a template or detail description of the AVS is presented. A more extensive explanation of the tool is given, and pictures of the results presented. A button to access the application is provided with each AVS. Some AVS can be accessed without credentials but for other permissions are required.

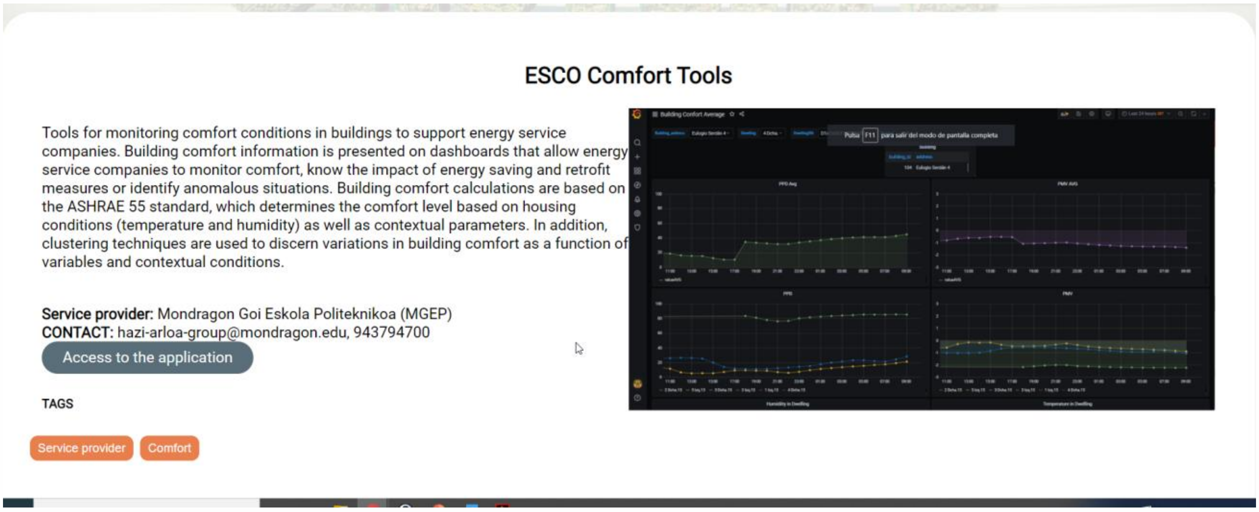


Figure 17: AVS Detail

6.2 CIOP Added Value Services AVSs

6.2.1 Monitoring of consumption and comfort conditions at home App

6.2.1.1 Description

This AVS tool enables residents in monitoring consumption and comfort conditions in the dwelling. The application monitors energy consumption and comfort conditions in the dwelling gathering information from sensors installed in the home. The main objective is to empower residents in the knowledge of comfort conditions and energy consumption. To this end, the application/solution objectives are:

- Collects electricity consumption and comfort data of the dwellings.
- Displays the data and recommendations to the residents (visualization tool).
- Allows comparisons (before and after the interventions).
- Calculates relevant indicators for the validation of interventions (KPIs).

6.2.1.2 Architecture definition & Infrastructure

The architecture of the solution is based in two main infrastructure blocks. On one hand, data are collected in the dwellings using the sensors installed in there. Figure 18 shows the infrastructure provision to collect information at dwelling level. The TV distributed system is used to direct that information to the CIOP platform without requiring the Internet connection from each of the dwellings residents. Internet connectivity is provided at building level to route comfort and energy data from each of the dwellings towards the sensor manufacturer that relays that information to the CIOP platform.

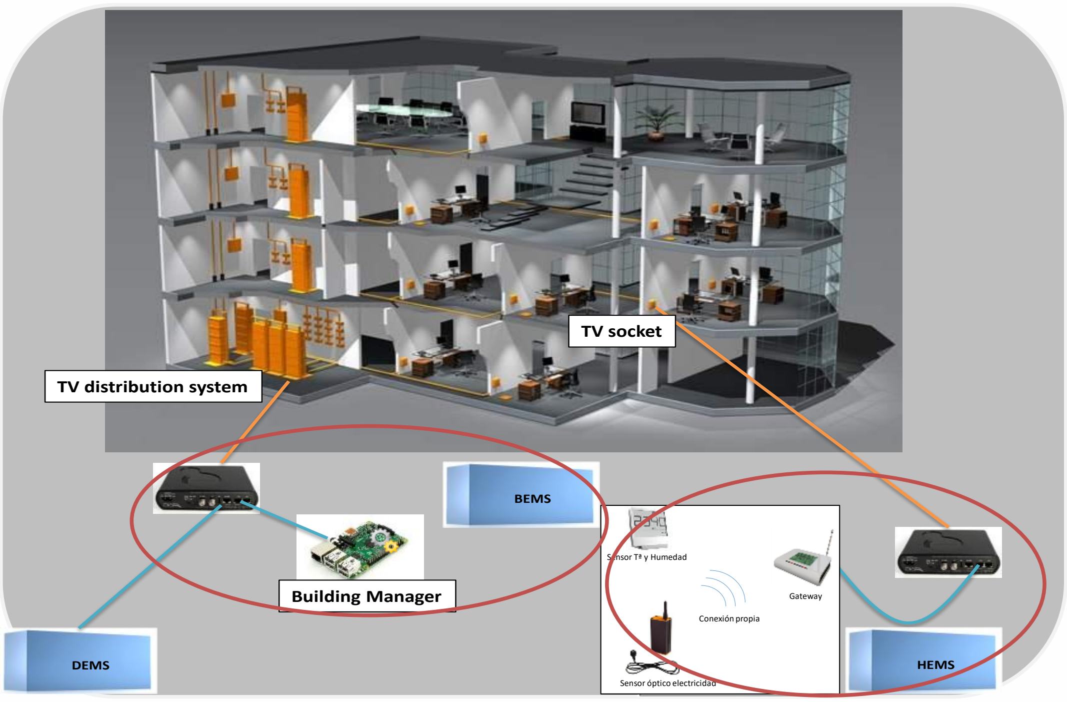


Figure 18: Sensor and Device Infrastructure at dwelling level

On the other hand, the CIOP platform provisions an acquisition and interoperability layer artifact build in Node-Red that collects the information, coming from the dwellings, and stores it in an InfluxDB database. Additionally, the CIOP platform holds a database where information about the district, buildings and dwellings is stored along aggregated comfort and energy data from the dwellings. This aggregated data are obtained and calculated from the InfluxDB repository using Node-RED as the programming tool. The information prepared in this second repository is offered by means of a REST API to be consumed by the front-end or any application that needs that information. The front-end or visualization application provision, in this case, has been developed with Vue.js which is a JavaScript framework to build Responsive Web Design applications. See Figure 19 for the technologies and repositories used in the CIOP platform for this AVS tool mapped to the Reference Architecture Layer Model.

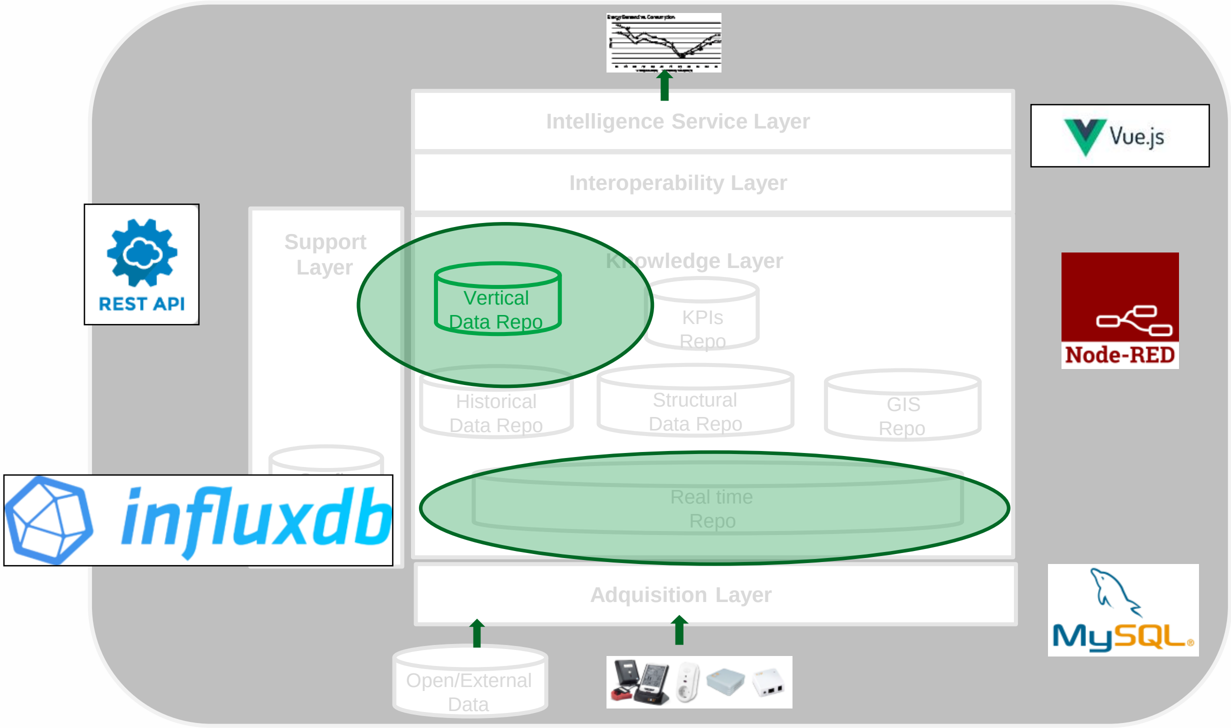


Figure 19: Technology and Repository Infrastructure at CIOP level

6.2.1.3 Data Structure

Two different types of repositories are used:

- InfluxDB: is an open-source time series database appropriate to monitor sensors in IoT deployments. For this AVS, it is used to collect data from the dwellings. See section 5.2.6 for more detail on the information stored in this repository.
- MySQL: is an open-source relational database management system that for this AVS is used to store structural information (district, building, dwelling, gateway…) and aggregated data obtained from the InfluxDB repository. Figure 20 presents the entity-relationship diagram for structural and aggregated data.

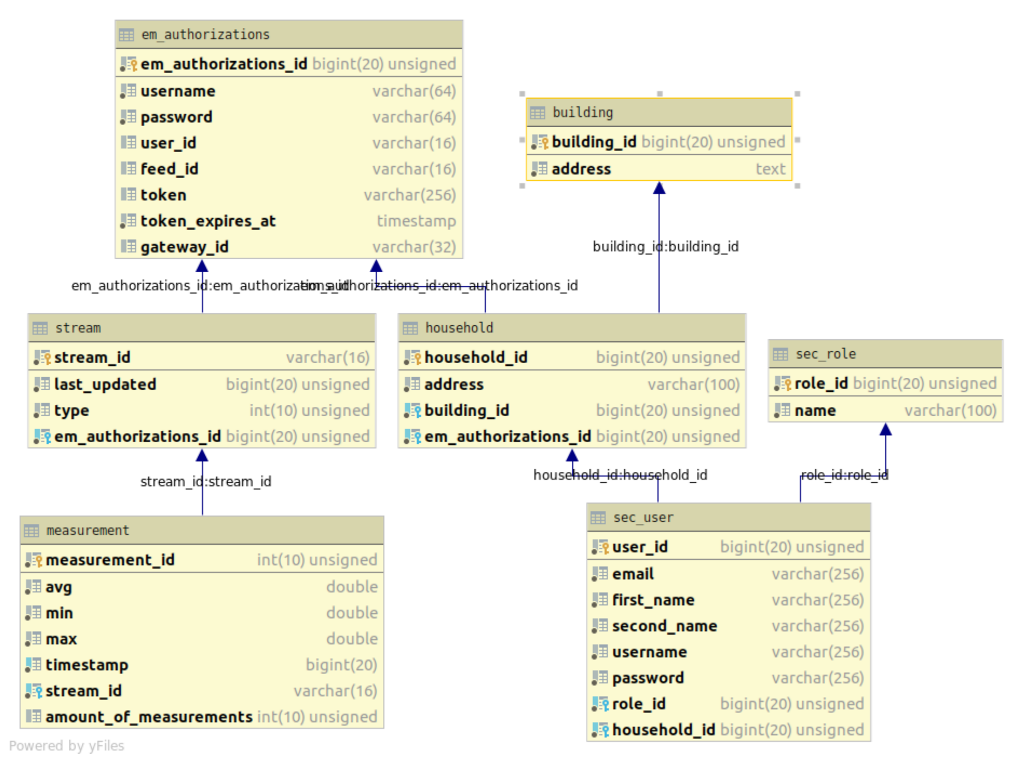


Figure 20: Structural and aggregated measurement DB

6.2.1.4 Tools

Node-RED: This programming framework is used to develop the backend for this application. Consist of several data flows that access databases and offer endpoints for the front-end to consume. There are dataflows to collect information from the gateways in the acquisition layer. Figure 21 presents some flows provision for the back end.

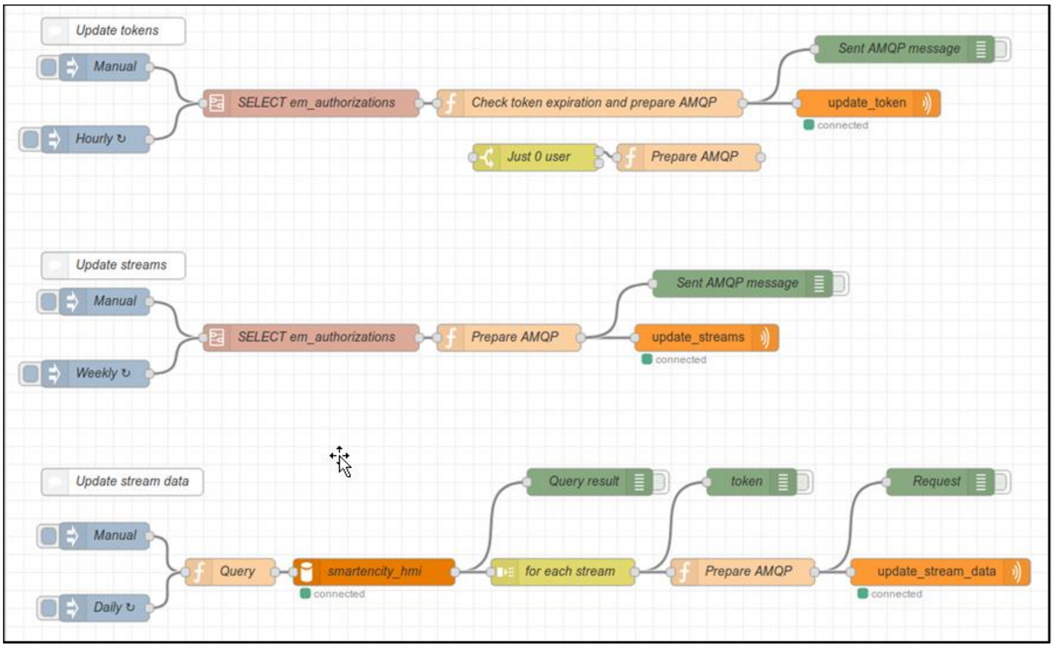


Figure 21: Backend code in Node-RED

Vue.js: This JavaScript based framework is used to build the front-end for the application. The tool improves the presentation of content enabling friendly HMI. The results obtained are shown in section 6.2.1.5.

6.2.1.5 User guide (Location, functions, etc.)

The tool can be accessed from the portal landing application or directly using the following link https://vitoria-gasteiz.smartencity.eu/hmi/.

The tool requires credentials to access the information.

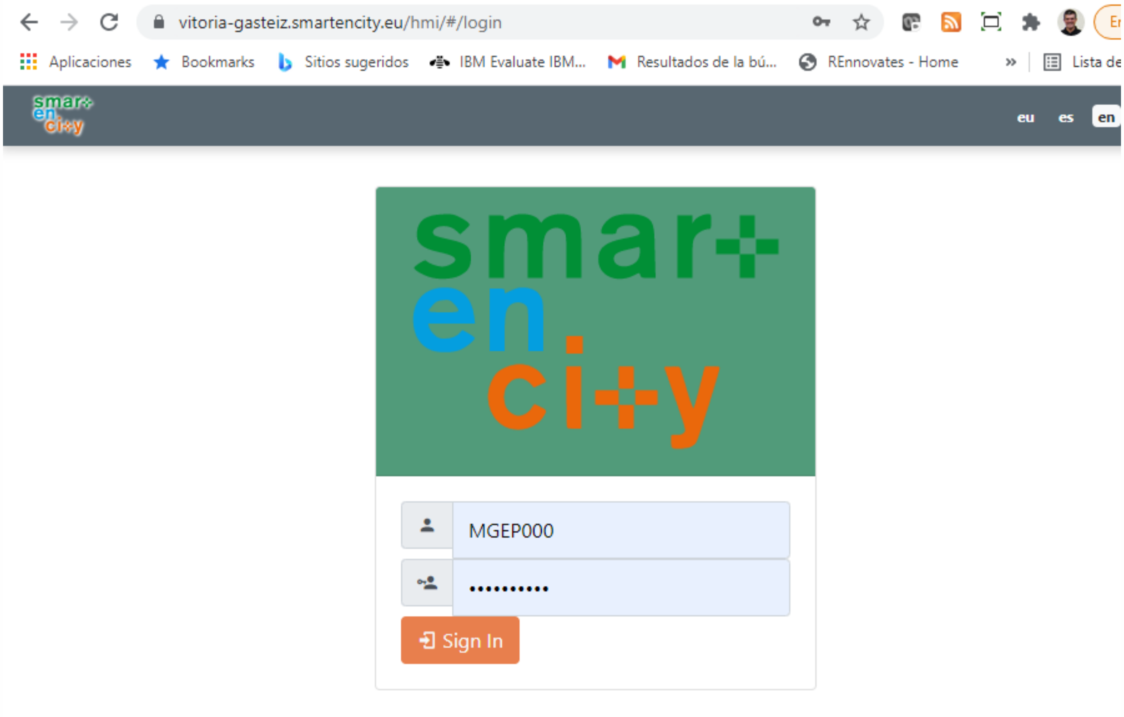


Figure 22: Access to the Home Monitoring APP

Once connected, the resident is presented with a comparison of his/her dwelling data with the average for the building and the district. There is an option to select the values for comparison (Power, Temperature, Humidity, CO_2 and Comfort). This last value is calculated from comfort measures applying the standard ASHRAE 55-2013. Figure 23 shows the Compare option of the tool. At the bottom of the page, news from the news channel are presented in a carrousel. The news channel collects news from Vitoria-Gasteiz provided by a third party (TENTU[6]) by means of web services. This news channel can be seen in any of the pages of the application.

[6] https://www.tentu.eus/

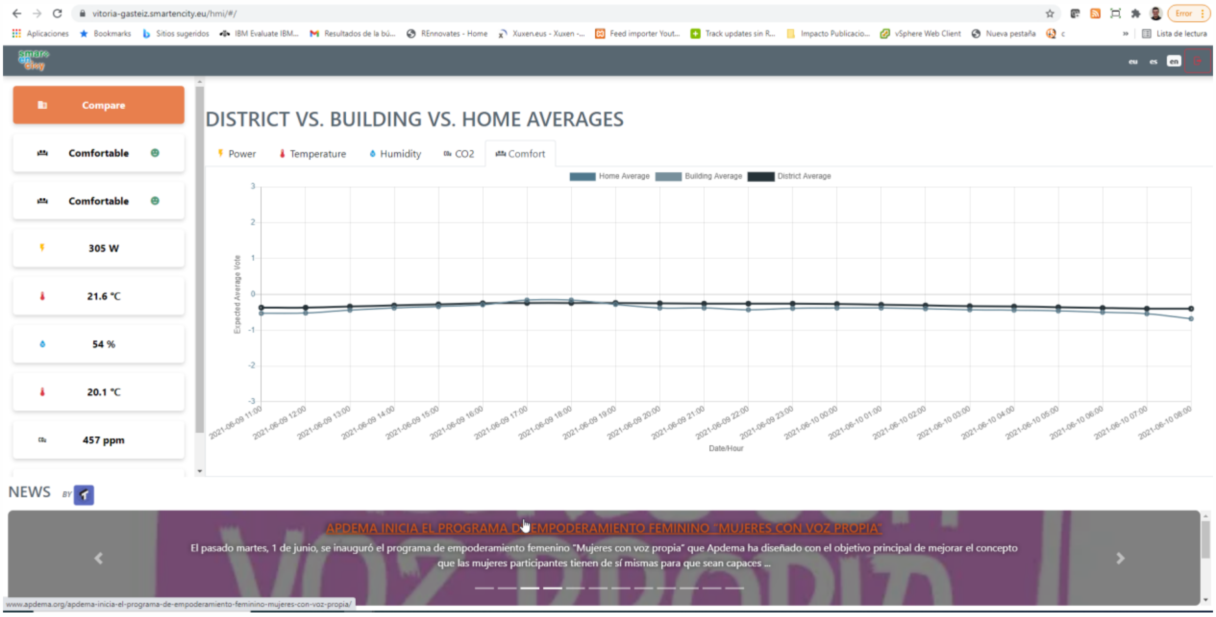


Figure 23: Home Monitoring APP (Compare)

The left-hand bar presents different selection options. Thus, in addition to the compare page we can select Comfort, Power, Temperature, Humidity or CO_2 options to see detail historical information. Some dwellings might present several comfortable, temperature and humidity options. This means the dwelling has several sensors installed. When selecting the Comfortable option, the values calculated with ASHRAE 55 are presented (see Figure 24). Two values are presented in this case: the ASHRAE comfort level and the percentage of people in discomfort under those conditions. For the first value, the closest to 0 the line is, the better the comfort level in the dwelling. For the second value, the lower the value the better. A table with historical data is presented in the right-hand side. Icons are provided to outline the status. The icons are coloured to show the status of the comfort levels. Green for comfortable, yellow for borderline and red for bad conditions.

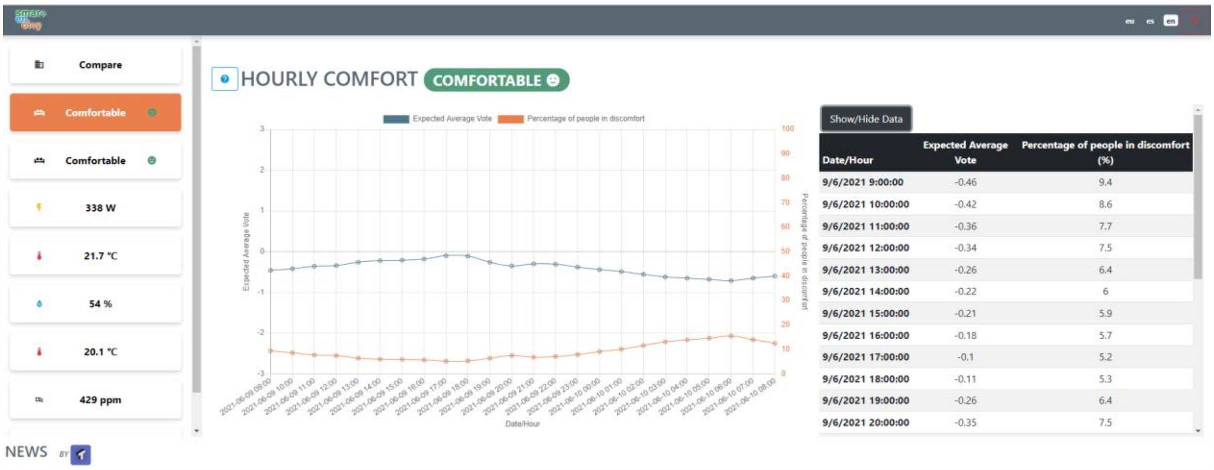


Figure 24: Home Monitoring APP (Comfortable)

For the rest of options, the presentation of the information is similar. Figure 25 presents the electric energy consumption in the dwelling for the last 24 hours. The maximum, minimum and average values for each hour are presented. The last measurement is also shown in a button.

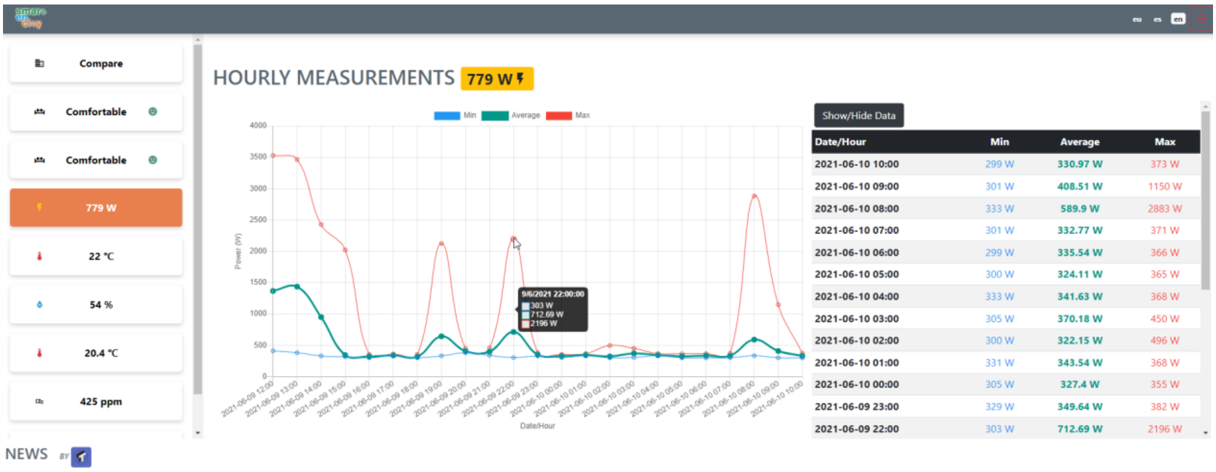


Figure 25: Home Monitoring APP (Power)

6.2.2 Installation and Management Support Tools

6.2.2.1 Description

This AVS tool supports installers and platform managers on providing service over sensor installations and monitoring its behaviour. On one hand, enables data commissioning for installations showing the status of each installation and its sensors and assuring that data are being acquired. On the other hand, it provides an alarm system for malfunctioning devices. Thus, when a device or sensor stops sending information detects this situation and informs platform managers about it. The tools also make possible to obtain and download data. The objectives for the tool are:

- Offer a form to support automatic registering of dwellings/gateways in the database during installations.
- Enable data commissioning. Correctness of data, frequency, completeness of data …
- Offer a system for installation sensor status monitoring.
- Offer an alarm system for malfunctioning devices/sensors.
- Offer a tool that facilitates the collection of missing information.
- Offer a tool that facilitates data consumption (download).

6.2.2.2 Architecture definition & Infrastructure

The architecture of the solution is the same as the one presented in section 6.2.1.2

6.2.2.3 Data Structure

The same data structure as in section 6.2.1.3 is used.

6.2.2.4 Tools

The same tools used in section 6.2.1.4 are used in this AVS.

6.2.2.5 User guide (Location, functions, etc.)

The tool can be accessed from the portal landing application or directly using the following link https://vitoria-gasteiz.smartencity.eu/installations/

The alternatives offered by the AVS are:

Installation Registry Tool: this application permits the registry of new installations on the system. The tool associates a dwelling with the gateway installed in the house and the sensors synchronized with it. The information collected in this form is stored in databases. From the moment of registration onwards the measurements collected from a given gateway are associated to the correspondent dwelling. The form is shown in Figure 26.

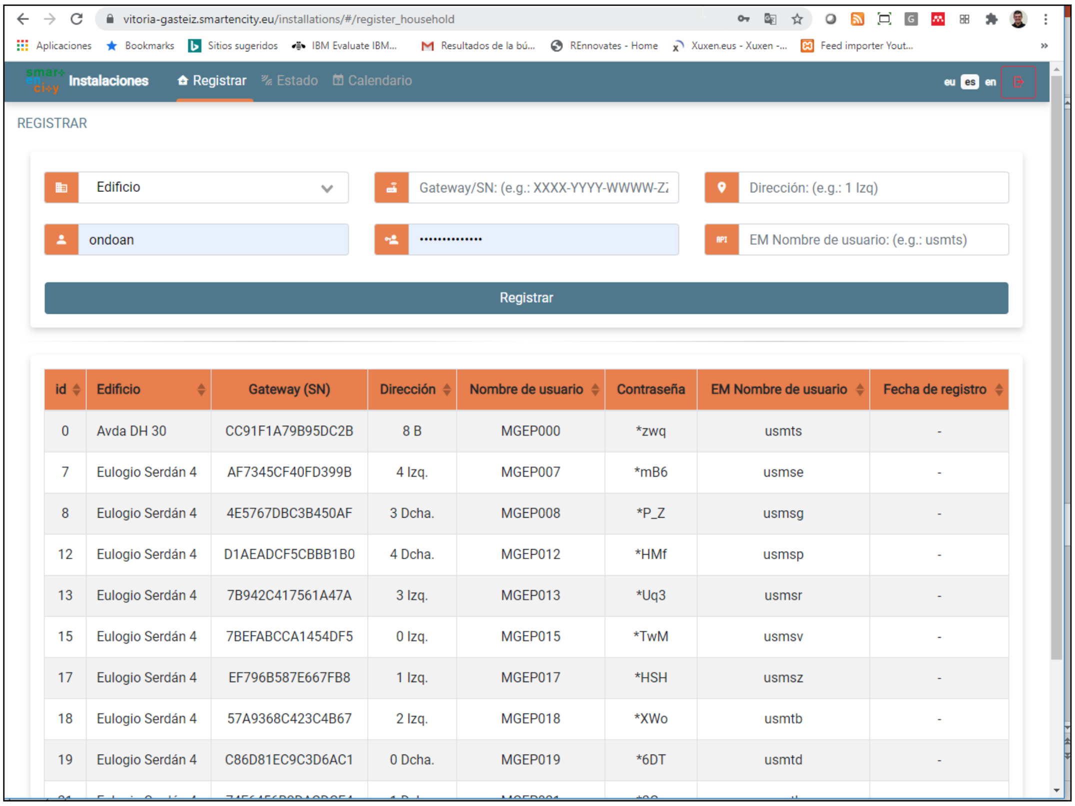


Figure 26: Installation Registration Tool

Installation Status Monitoring tool: This application shows the status of each dwelling in relation to connectivity and data provision. The web application serves as a commissioning tool where sensors not providing information can be detected. Figure 27 shows 5 dwellings from one of the buildings. Dwellings in green indicate that the gateway and sensors are working correctly. Dwellings in red mean that the information from the gateway is not reaching the CIOP platform. Dates are provided to indicate when was the last time data were collected. Dwellings in black indicate that the platform has never received information from a gateway or sensors. There are situations where the gateway is green, and a given sensor is red. This means that the sensor is no longer sending data to the gateway probably due to synchronization problem or power failure.

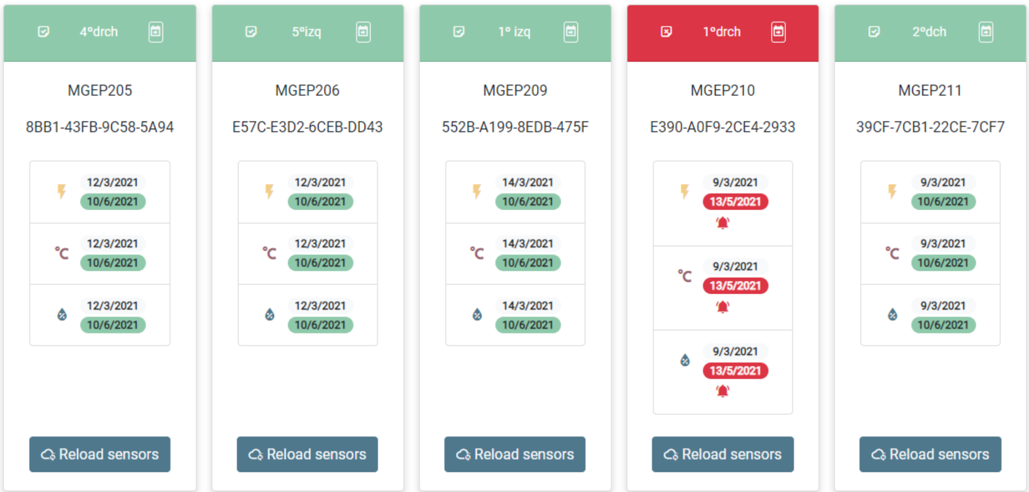


Figure 27: Installation Status Monitoring Tool

Data Download Utility: This utility enables the downloading of a given data set. The service operator or data commission stakeholder can select a dwelling and a parameter and download data in csv form to analyse it or calculate a KPI. The tool offers a calendar to select data for a specific period. The utility also presents a heat map that shows periods of time with missing information. If the information is still available, it can be pulled back to complete the data grid. In Figure 28 the data download utility interface is presented.

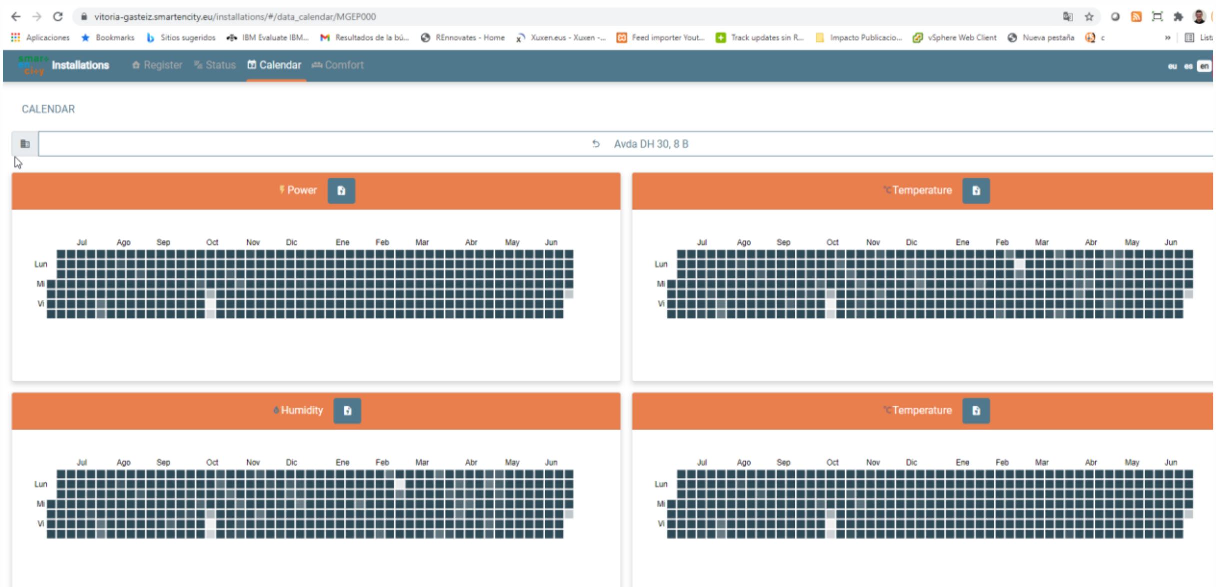


Figure 28: Data Download Utility

Alarm System: This application has been developed in the backend. The application detects if a sensor or gateway stops sending information and sends a message to the platform maintenance responsible generating a ticket. A period of 2 weeks is set as the threshold for detecting data collection interruptions.

In the installation application (described above), it is visible if the notification has been sent. In Figure 27 the sensors with a red bell are the ones already notified as failures. Figure 29 presents the alarm detection system developed in Node-RED.

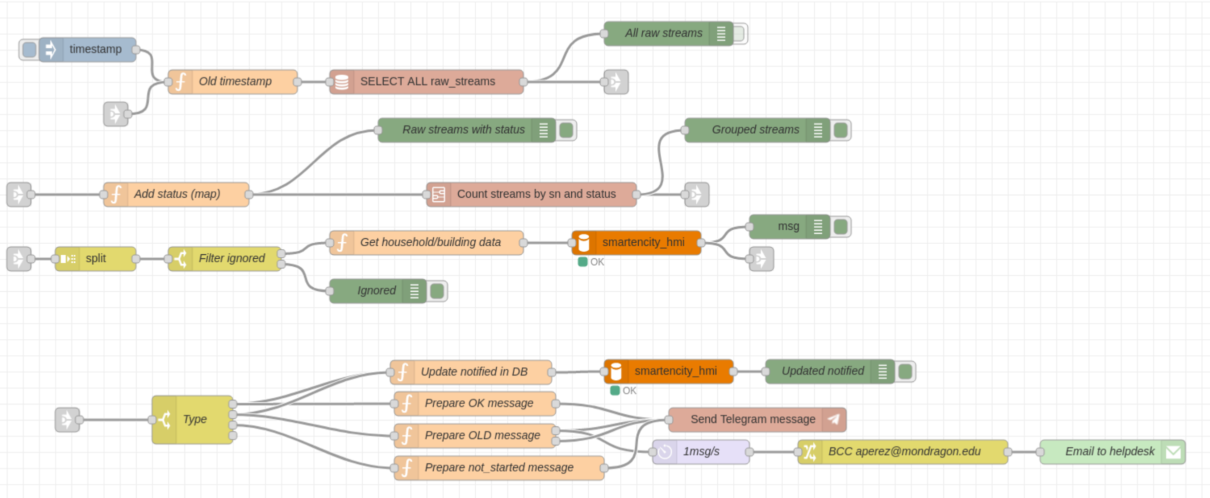


Figure 29: Alarm Detection System in Node-RED

6.2.3 Data analysis of comfort conditions for ESCO support App

6.2.3.1 Description

This block considers tools for monitoring comfort conditions in buildings to support energy service companies. Building comfort information is presented on dashboards that allow energy service companies to monitor comfort, know the impact of energy saving and retrofit measures or identify anomalous situations. Building comfort calculations are based on the ASHRAE 55 standard, which determines the comfort level based on housing conditions (temperature and humidity) as well as contextual parameters (occupancy, outside weather conditions, isolation level…). In addition, clustering techniques are used to discern variations in building comfort as a function of variables and contextual conditions (this part will be presented in WP7 in relation to KPIs). This AVS tool permits building dashboards from the data collected in the different data repositories. The aim is to offer ESCO providers with dashboards relevant to comfort and energy measurements. The objectives for the tool are:

- Offer a web tool where an ESCO can build their own dashboards.
- Provide a dashboard that presents comfort information to support condition validation. The ESCO will use this tool to monitor building and dwelling conditions.

6.2.3.2 Architecture definition & Infrastructure

The same infrastructure used in section 6.2.1.2 for collecting data and those data are used for this AVS.

To prepare and consume data a back-end build in Node-RED has been constructed and an API build in Python has been provided. See section 5.2.9 for more detail. This application calculates from the indoor comfort conditions and other parameters (weather outside, dwelling occupancy…) the comfort levels of the dwelling according to ASHRAE 55. The front-end or visual interface accommodated in the Intelligence Service layer is developed with Grafana. Figure 30 shows the technology architecture mapped to the reference architecture.

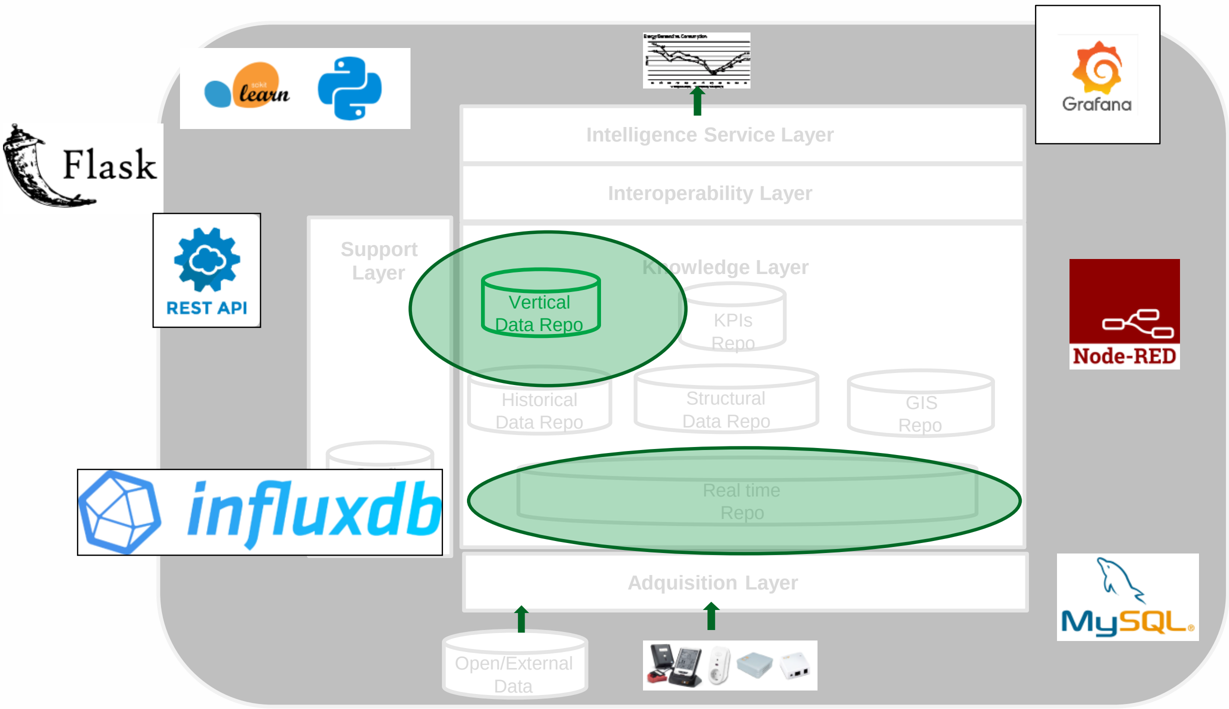


Figure 30: Technology & Repository Infrastructure at CIOP level for ESCO support

6.2.3.3 Data Structure

The same data structure as in section 6.2.1.3 is used in this case.

6.2.3.4 Tools

In addition to Node-RED also used in the previous AVS for this solution, Grafana has been employed. Grafana is a multi-platform open-source analytics and interactive visualization web application. Grafana is installed as a server and dashboards can be viewed and edited from the web interface using a browser (see Figure 31). In our case and as for most of the solutions installed, Docker technology has been employed.

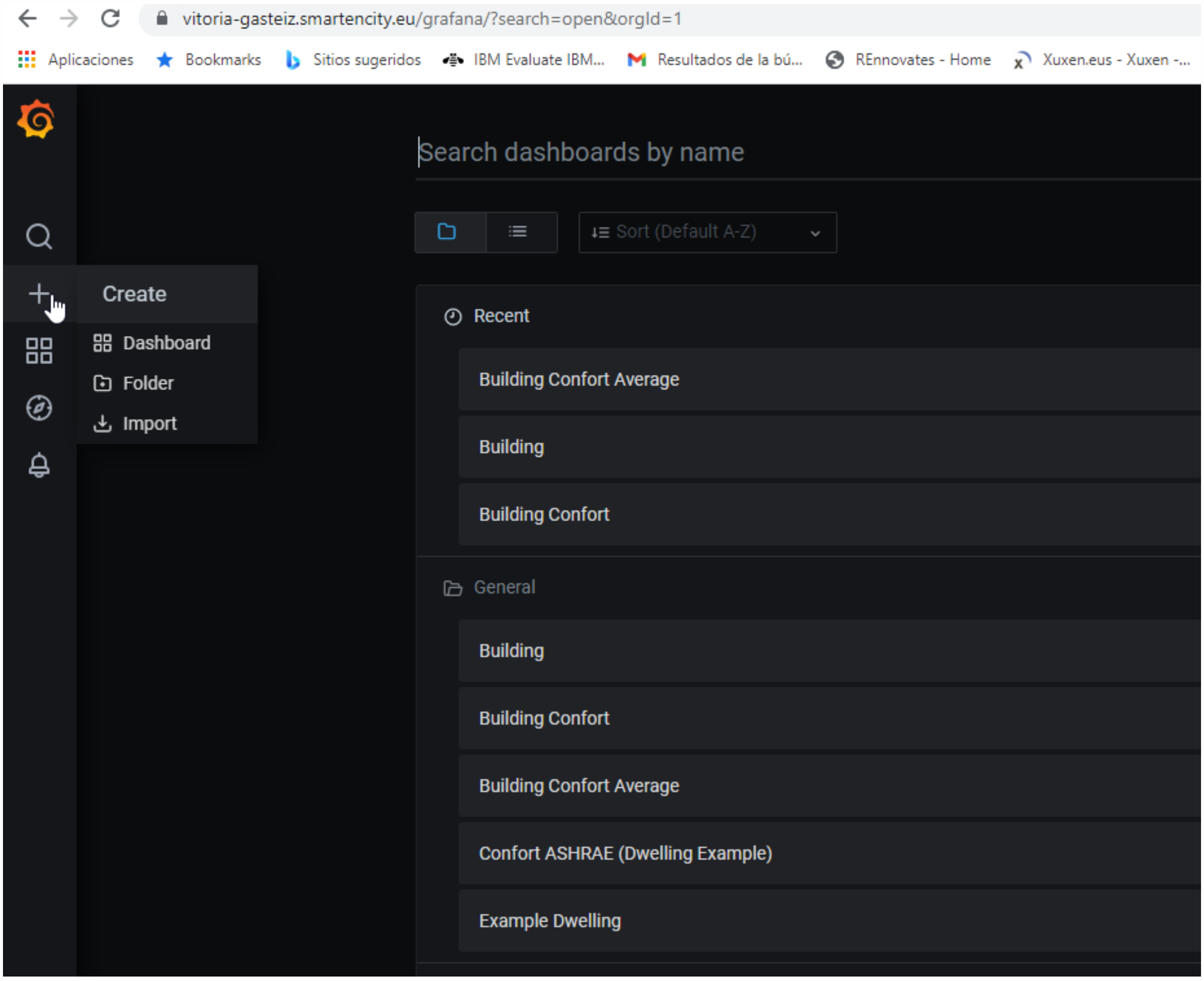


Figure 31: Grafana Home Page

6.2.3.5 User guide (Location, functions, etc.)

The tool can be accessed from the portal landing application or directly using the following link https://vitoria-gasteiz.smartencity.eu/grafana/d/poVdwwAMz/building-confort-average?orgId=1. As in the previous cases, a user and a password are necessary.

The AVS dashboard presents several panels (see Figure 32). At top building average values are presented. The average value for the ASHRAE 55 Comfort level and the percentage of people in discomfort are shown there. This shows the general comfort conditions of the building. The details for each dwelling in the building are presented in the second row of panels. This enables to compare dwellings among them.

Figure 32: Comfort Dashboard for ESCO

The third row presents the temperature and humidity of a given dwelling. Building and dwelling can be selected in the drop-down menu provided on top of the page.

In addition, Grafana enables the creation of any dashboard or panel required by the ESCO. As an example, Figure 33 presents a dashboard created with the measurements of one dwelling.

Figure 33: Customized Dashboard

6.2.4 Local news channel (TV)

6.2.4.1 Description

This AVS is a Web portal offering a local news channel based on RSS from websites about Vitoria-Gasteiz that can be consumed through building infrastructure (TV). The aim is to offer residents a news channel with relevant information through TV in a similar way as offered in airports, public buildings, or street panels. The objectives for the tool are:

- Collect data from websites with news about Vitoria.
- Present data in a web portal and consume it on TV through building infrastructure.
- Enable a local news channel.

6.2.4.2 Architecture definition & Infrastructure

The architecture supports the presentation of the content using the TV distribution system. To achieve that, adapters are placed at the TV distribution system header and each of the dwellings. Equipment for TV streaming is also placed in the header. Figure 34 and Figure 35 present the architecture and the equipment necessary for the solution deployed in a laboratory.

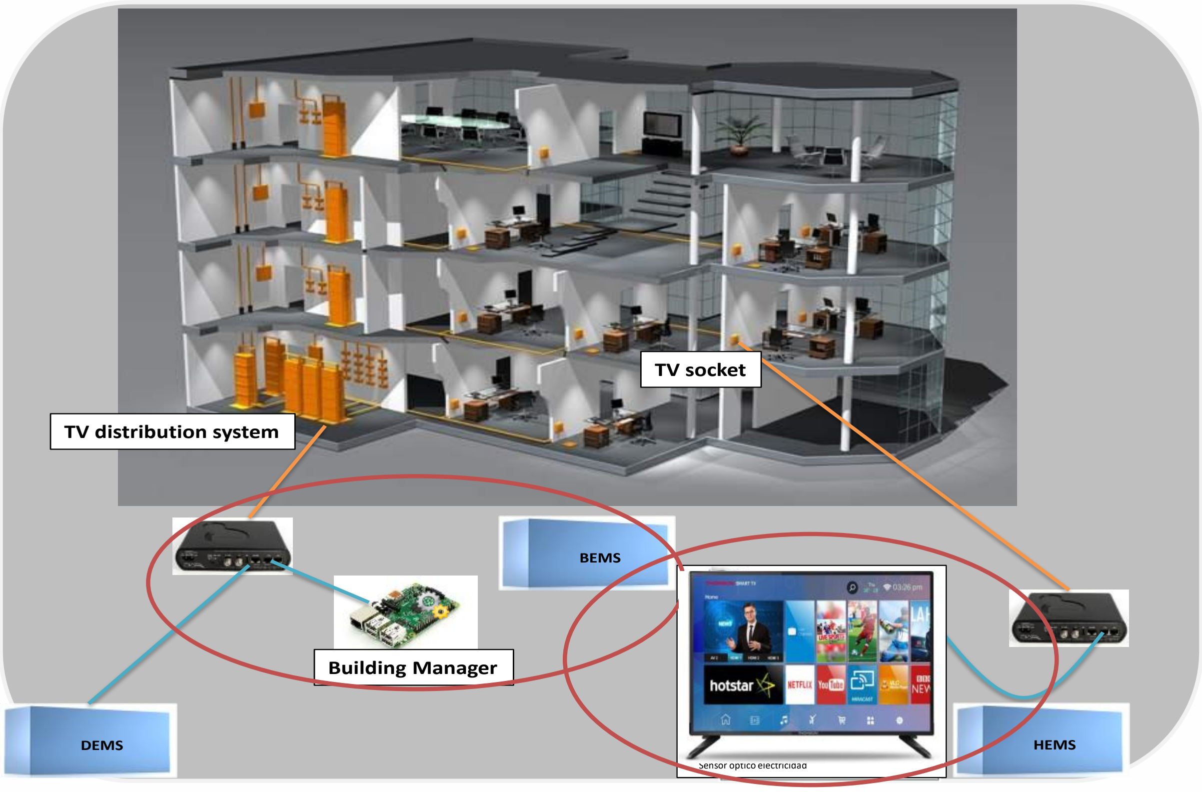


Figure 34: Local TV Channel Building Architecture

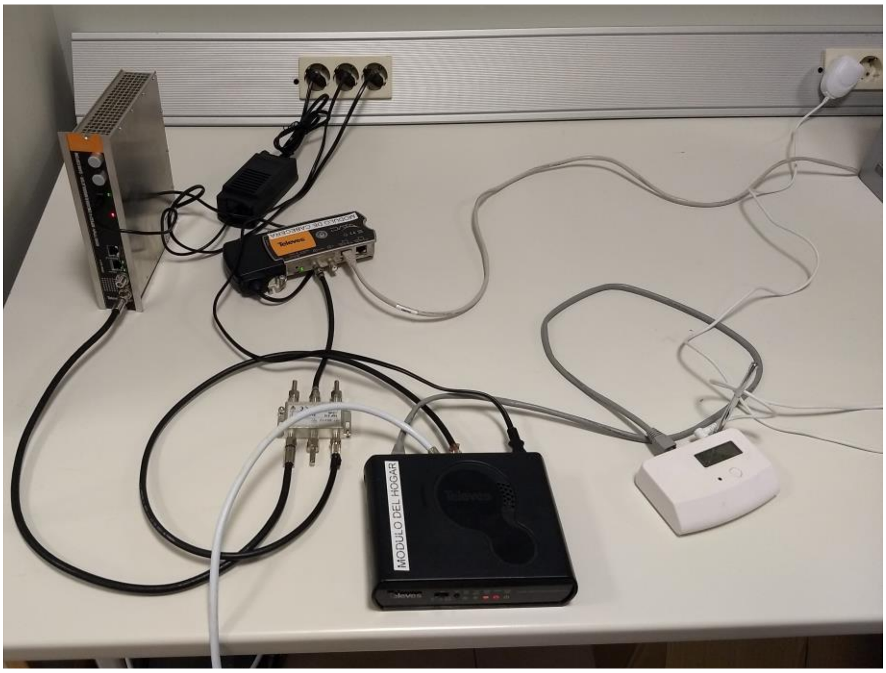


Figure 35: Equipment in laboratory

The software architecture is presented in Figure 36. In this case data is consumed by a third-party provider that offers news through web services. The third-party organization (TENTU) collects information from multiple sources using different techniques (RSS, scraping, etc.). They offer news about Vitoria-Gasteiz that we collect into the CIOP portal. Using similar techniques as those of sections 6.2.1 and 6.2.2 we built a web application that is streamed down the TV system and can be seen in a TV set.

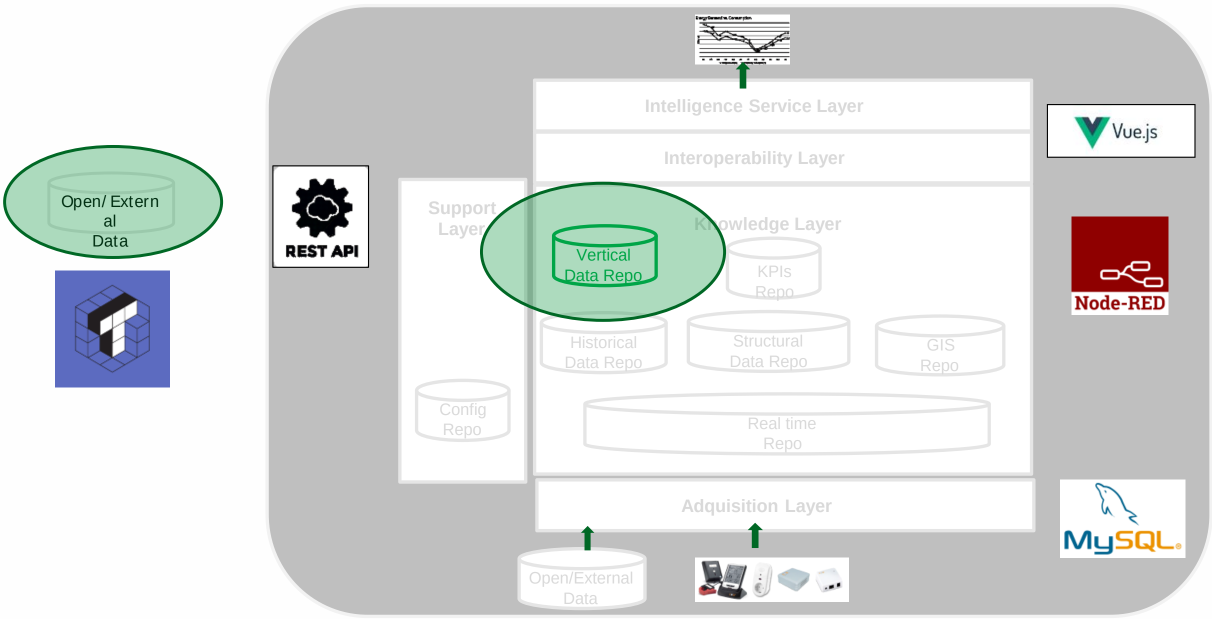


Figure 36: Software Architecture

6.2.4.3 Data Structure

Information coming from the third-party is not stored in the CIOP platform.

6.2.4.4 Tools

Vue.js: This JavaScript based framework is used to build the front-end for the application. The tool improves the presentation of content enabling friendly HMI. See section 6.2.4.5 presents the result obtained.

6.2.4.5 User guide (Location, functions, etc.)

The tool can be accessed from the portal landing application or directly using the following link https://vitoria-gasteiz.smartencity.eu/hmi/#/news.

The tool is open and does not require credentials to access the information. It can be programmed in a TV channel to be viewed or can be also consumed with a browser. The AVS offers a carrousel with the latest news offered by the third-party provider.

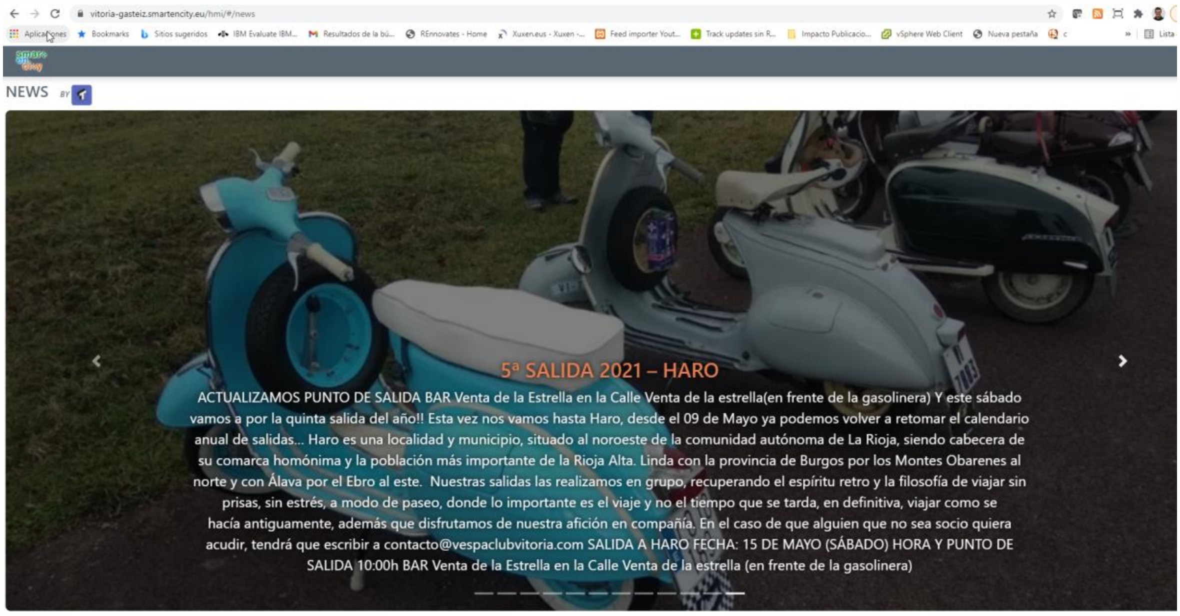


Figure 37: Local News Channel

6.2.5 Municipal Buildings Energy Usage Monitoring

6.2.5.1 Description

The municipal energy consumption application provides information on the different municipal consumptions of the city, including municipal buildings or facilities, water used in parks and gardens, public lighting and mobility of vehicles used by the city council.

It is specially oriented to analyse the different consumptions in the city. It also offers a carbon footprint indicator to observe the environmental impact. The City Council is currently carrying out a process of uploading historical consumption in order to perform information analytics.

The application feeds a database that records the readings and invoices of the different meters in the city. In this way, in addition to energy consumption, the economic costs of each energy source are obtained.

The information is supported by the georeferencing of the consumption points distributed throughout the city. This georeferencing helps to locate the different meters, municipal buildings…

The application is aimed especially at municipal technicians to analyse consumption, check the areas or points of greatest consumption and see the variations or evolution that occurs after refurbishment towards more sustainable energy sources.

6.2.5.2 Architecture definition & Infrastructure

The architecture is composed of several components shown in the following diagram:

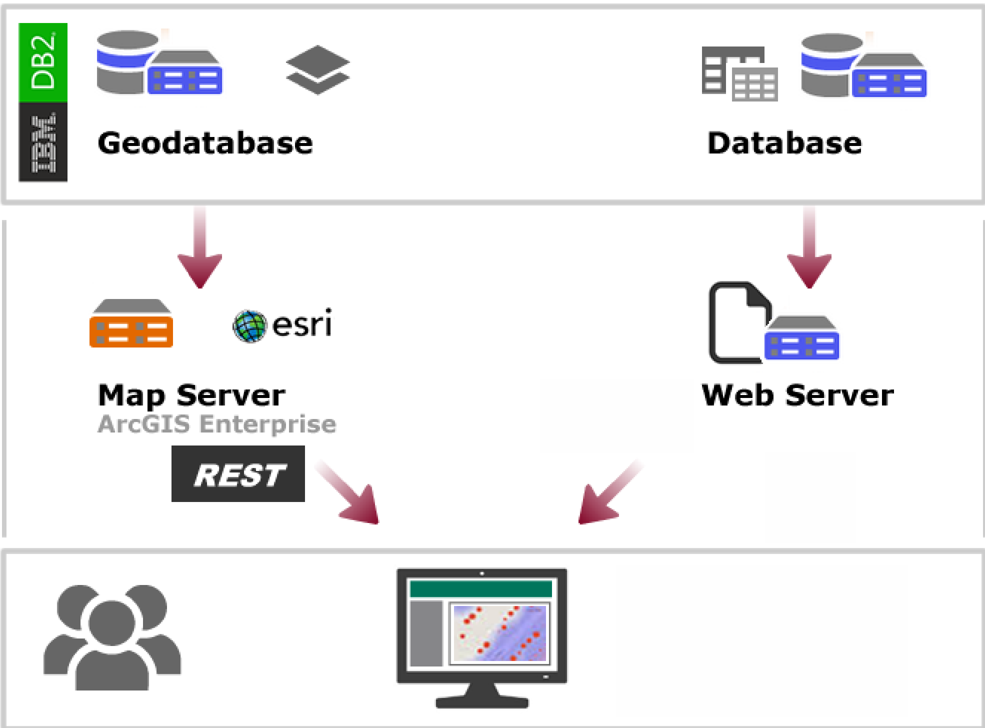


Figure 38: Architecture scheme

Data sources provide the information to show the different consumptions of municipal facilities.

Geo Data Source has the georeferenced data of the facilities and consumption points and their associated installations.

Map Server (ArcGIS Server + Portal for ArcGIS) provides the hosted information and makes the information on consumption and geographic data available through REST services.

Web Server hosts the web application in charge of interpreting, consulting and representing the information so that the user has the municipal information.

6.2.5.3 Data Structure

The project makes use of both alphanumeric and geographic data, which are supported by a database and a geodatabase.

Alphanumeric data:

The application makes use of a database that currently exists in the City Council of Vitoria-Gasteiz and that collects data on consumption points, meters, and the consumption of each meter for each period. The database is more extensive, including supplier data and other additional information that is not indicated in the following diagram.

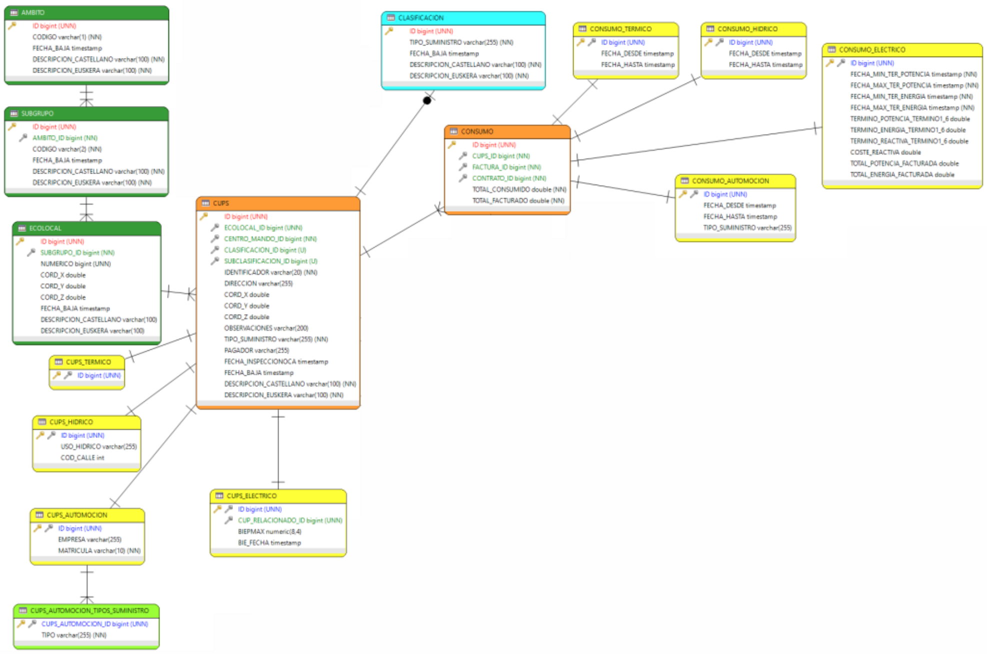


Figure 39: Relational Database Scheme

The following tables are highlighted to help to understand the information managed by the application:

- ✓ [**AMBITO + SUBGRUPO**] – This table allows the classification of consumption by type of building or installation.
- ✓ [**ECOLOCAL**] – It is the table of buildings or consumption facilities.
- ✓ [**CUPS**] – This table stores the data of the meters that are associated to the facilities. Each CUPS (Supply Point Universal Code) has a unique identifier and the type of energy (electricity, water, gas, ...)
- ✓ [**CONSUMO**] – This is the consumption recorded for each period, both the amount consumed, and the cost consumed. This is mainly data from invoices, but it can also be supplier records.

Geographical data (geodatabase):

The geographic database stores the consumption points and the municipal buildings, as well as other references layers such as buildings and neighbourhoods so that the user can contextualize the information.

Figure 40: Geodatabase content

6.2.5.4 Tools

The application has a geographic viewer and uses a series of tools to support the storage, processing and consultation of information, which is why the following components stand out:

- **ArcGIS Enterprise** and other ESRI products and services, to cover the entire data cycle, from data processing and georeferencing, storage and publication of GIS services.

It is noted in this use of the tool that the addresses for the meters were available, and an inverse georeferencing has been performed using processing services with HERE cartography from these addresses.

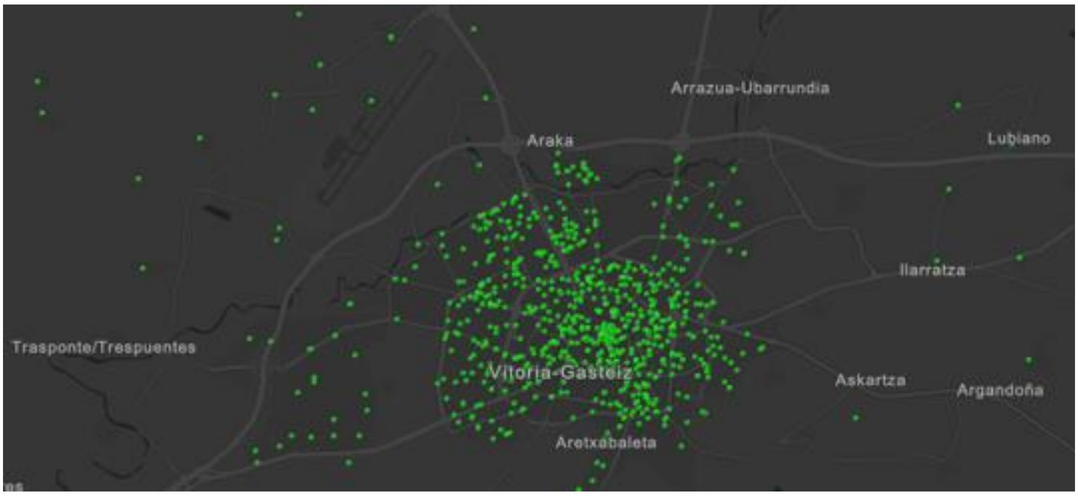


Figure 41: Inverse geocoding – Consumption points

- **Web Appbuilder**. This software has been used for the development of the user interface. This product is a framework that allows the creation and customization of applications with a geographic component. The application bases its composition on Widgets and templates that make it responsive. The development is implemented with JavaScript and Dojo, which is responsible for consuming the REST services to query the information.

6.2.5.5 User guide (Location, functions, etc.)

The application displays the information through graphs that show the evolution and consumption of each energy, costs, and carbon footprint. The data are also presented aggregated by year to observe consumption trends. To make it more visual, each energy source has a representative colour (Electricity-Yellow, Water-Blue, Gas-Orange, Hydrocarbons - Gray and Dark green, ...)

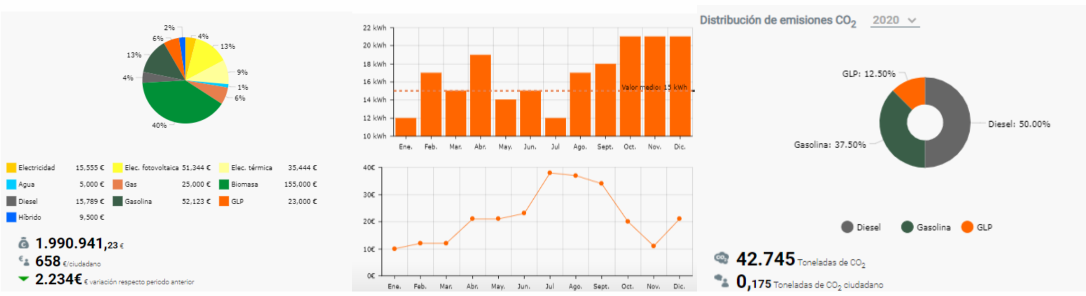


Figure 42: Graphics available in the presentation

The application has different levels of information aggregation:

Figure 43: Levels of information

- [Municipio] – City
- [Barrio] – Neighbourhood
- [Equipamientos] – Installations
- [Equipamiento] – Facility
- [CUPS] – Meters

Various types of energy consumption are shown:

- Electricity, solar energy, thermal energy, water, gas, biomass, diesel, gasoline, LPG, hybrid energy

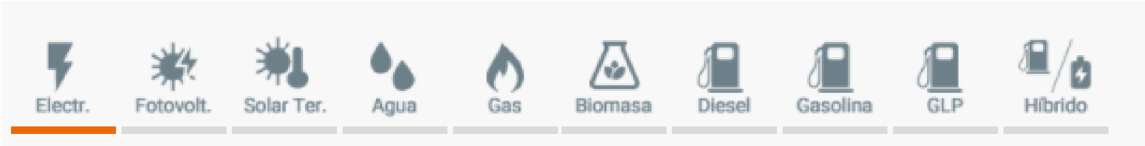


Figure 44: Types of energy

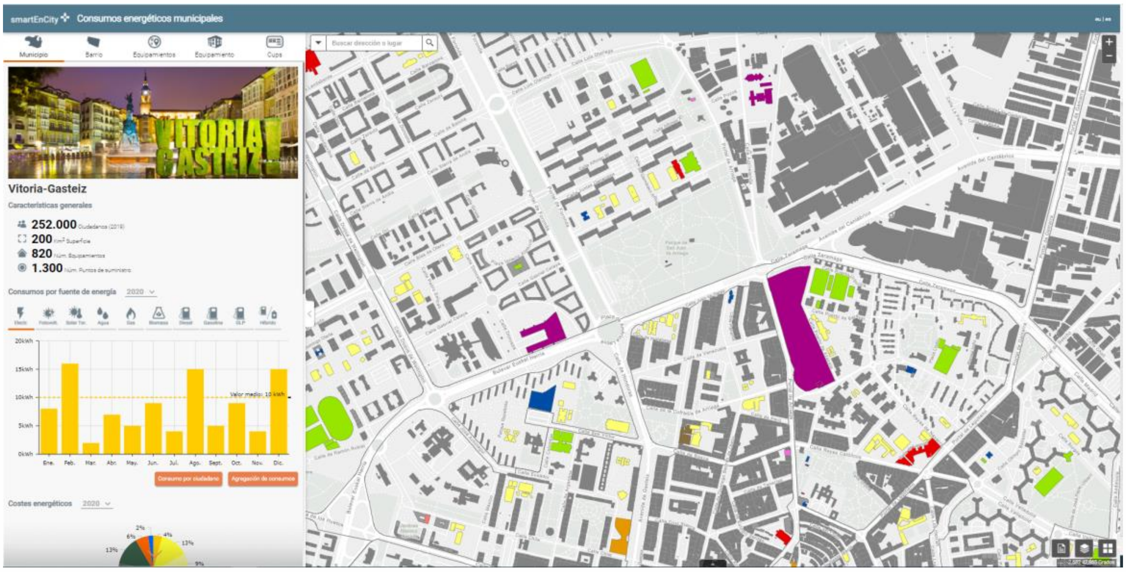


Figure 45: Municipal energy consumption application

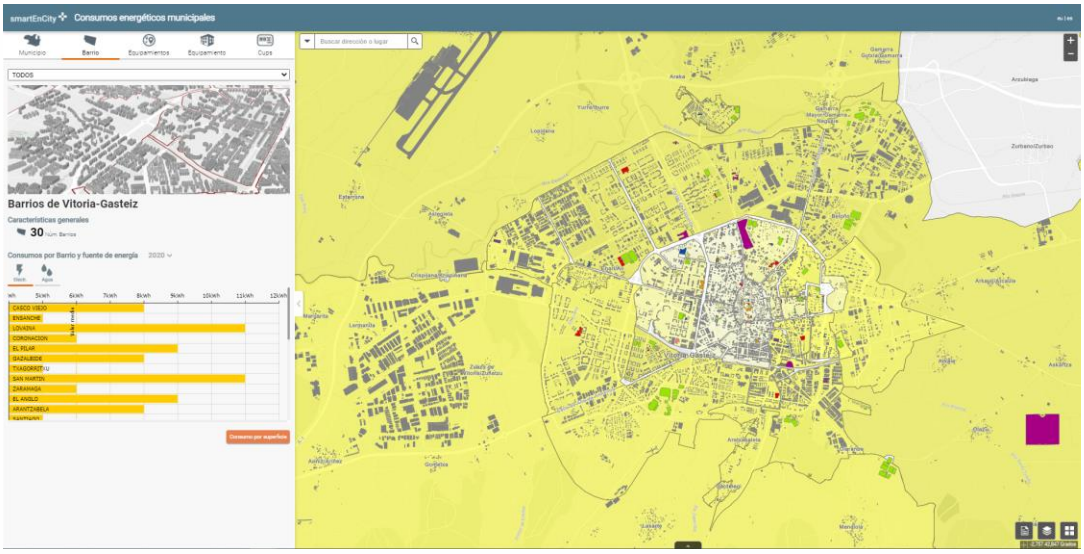


Figure 46: Electricity consumption associated with the neighbourhood (lighting)

The application is specially oriented for municipal technicians, but it is foreseen that in the future, when the information recorded on consumption is more complete, the data will be open to the public.

Temporary access URL: https://gissrv/smartencity/

6.2.6 Smart Electric Bus - BEI - Monitoring

6.2.6.1 Description

The Smart Electric Bus -BEI- application provides KPIs and the positioning of the future electric buses in Vitoria-Gasteiz. The BEI is a project for the evolution of urban transport towards a cleaner or greener energy source that impacts on the mobility, sustainability, and modernization of the city.

This project replaces one of the main urban bus lines of the city with a more modern and sustainable one, improving the urban mobility and the environmental impact.

Our web application offers information related to the data taken by the electric chargers and the vehicles themselves through the CAN bus. This new means of transport and its associated infrastructure (chargers, stops…) has advanced technology for monitoring, recording incidents, planning, ticketing, etc.

The application obtains data of the electric bus from the iPanel Platform that manages and centralises all the information on this new urban transport.

6.2.6.2 Architecture definition & Infrastructure

The architecture and applications are based on very similar technology to the Municipal Buildings Energy Usage Monitoring application since the GIS technology is the same and the user interface is similar. There is a new source of data in this case, the DATIK Platform, which offers web services for consulting the information.

The following diagram shows a basic scheme of the infrastructure that supports the IT solution.

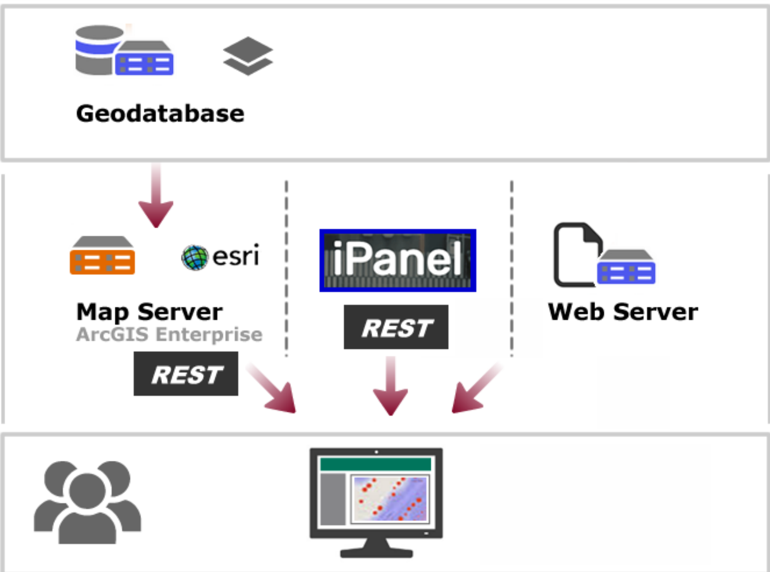


Figure 47: Architecture scheme

Geo Data Source has the georeferenced data of the BEI lines, such as the stops, the recharging points or the location of the different vehicles.

Map Server (ArcGIS Server + Portal for ArcGIS) provides the information hosted in GIS services mode to be consumed by the application.

API iPanel (DATIK) – This is the platform accessed by our application to consult vehicle and charger data. It offers a REST API for accessing the information.

https://datik.ipanel.cloud/help/es/api/

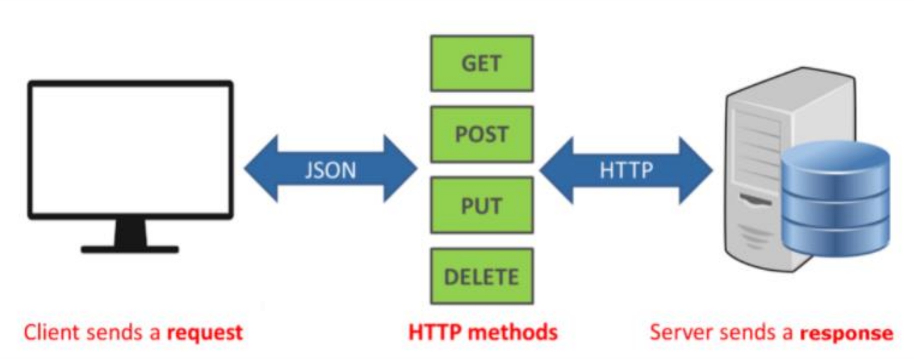


Figure 48: iPanel info access

Web Server – It hosts the web application in charge of interpreting, consulting, and representing the information so that the user can consult the KPIs and the location of the electric buses.

6.2.6.3 Data Structure

The data structure of the BEI is based on two parts, the geodatabase with the supporting geographic information for visualization in a GIS viewer and the information collected through the iPanel via WEB services.

Geographical data (geodatabase)

The geographic database stores the BEI lines, the stops, and the recharging points, as well as the current positions of the vehicles.

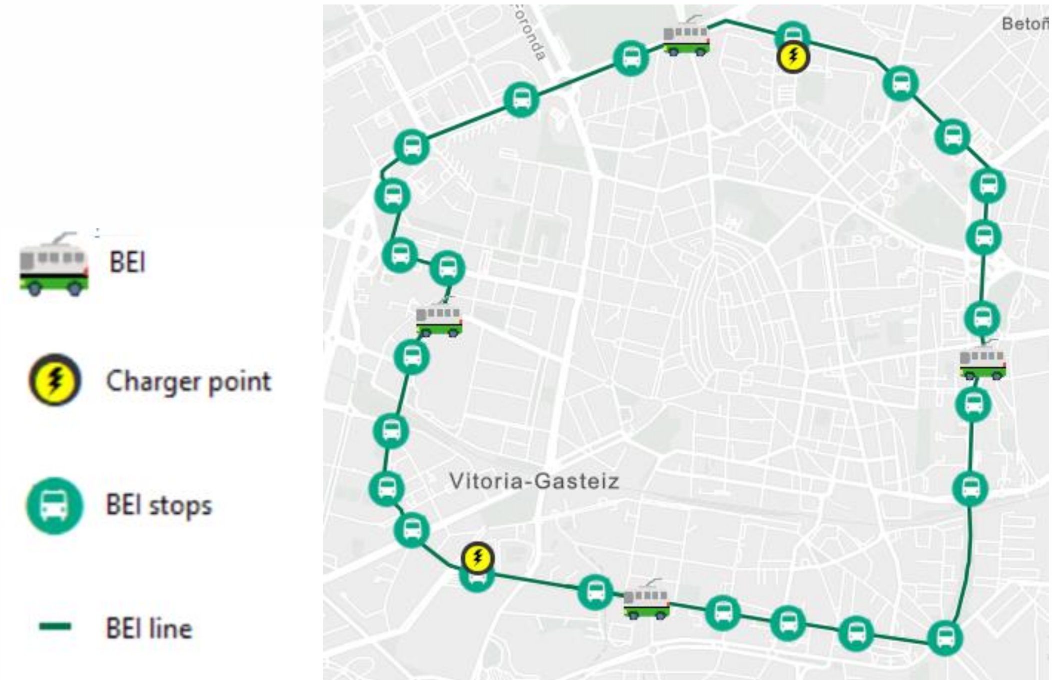


Figure 49: BEI application geographical data

Alphanumeric data (iPanel API Rest)

As mentioned above, we access the information through the REST API provided by iPanel. In our case, there are two sections or modules that are of interest to obtain the information for our KPIs.

Vehicle module: It offers services associated with vehicles, both in real time and historical queries.

https://datik.ipanel.cloud/help/es/api/chargers-api/

Chargers module: There are services associated with electric vehicle chargers and electric charging infrastructures in general.

https://datik.ipanel.cloud/help/es/api/chargers-api/

6.2.6.4 Tools

The application has a geographic viewer and uses a series of tools to support the storage, processing and consultation of information. The following components stand out:

- **ArcGIS Enterprise** and other ESRI products and services, to cover the entire data cycle, from data processing and georeferencing, storage and publication of GIS services.
- **Web Appbuilder**. This software has been used for the development of the user interface. This product is a framework that allows the creation and customization of applications with a geographic component. The application bases its composition on Widgets and templates that make it responsive. The development is implemented with JavaScript and Dojo, which is responsible for consuming the REST services to query the information.

6.2.6.5 User guide (Location, functions, etc.)

The Intelligent Electric Bus -BEI- Monitoring web application offers intuitive, user-friendly and responsive interfaces as shown below:

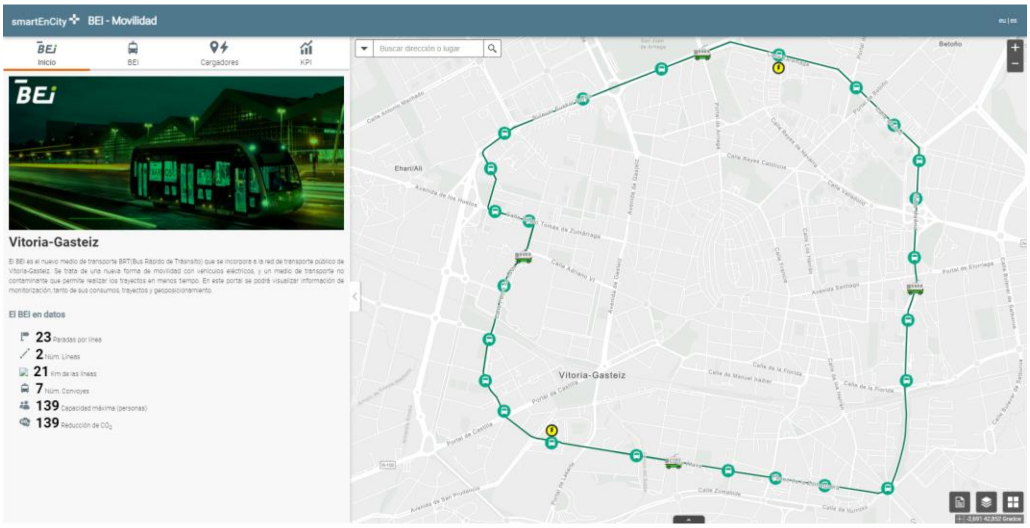
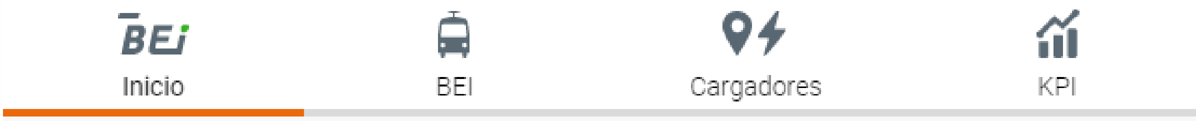


Figure 50: BEI user interface (Home page)

The application shows 4 different sections of information, in order to present the information in an organised and user-friendly way.



[**Inicio**] – This is the introductory part of the application and displays basic information about the BEI, the level of service, the infrastructure, and the characteristics of the lines.

[**BEI**] – It provides information on the vehicles, their condition, the number of kilometres, the number of loads, etc. as well as their location on the map.

[**Cargadores**] – It shows information related to the chargers, allowing access to their location and data on daily consumption and number of loads.

[KPI] - This last section collects information on other related indicators such as total consumption, estimated CO_2 levels and other related KPIs.

It shows the information to the user through graphs with month and year selectors to consult the different information periods.

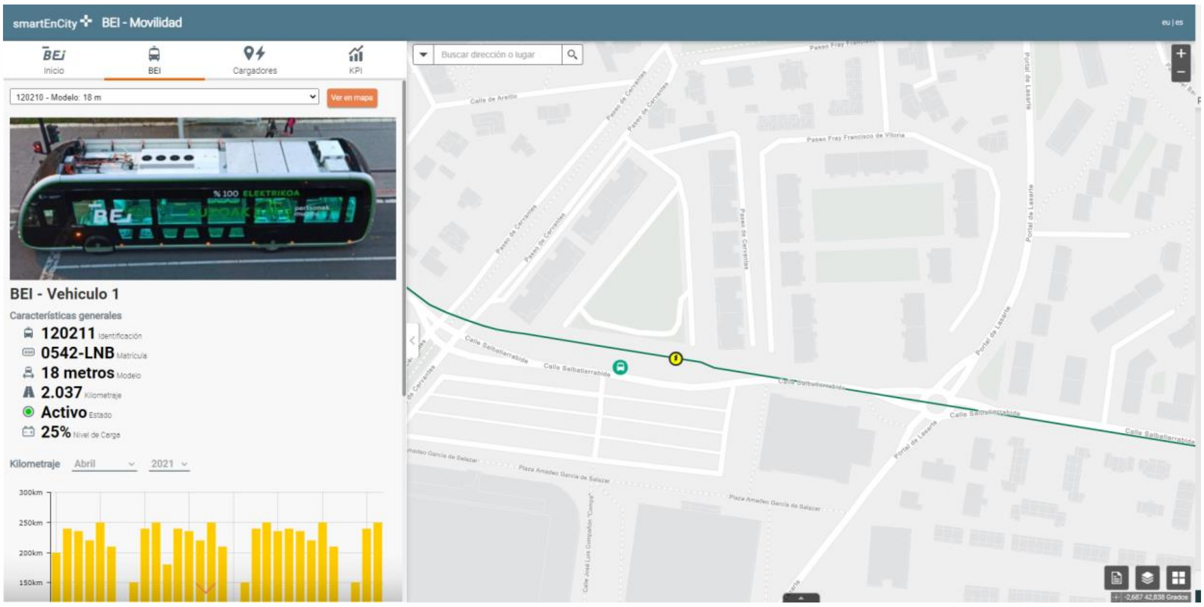


Figure 51: BEI user interface (Vehicles page)

The application is currently not accessible in a production environment due to the fact that the BEI is currently in testing and there is not enough information available to validate the application and make it available to end users.

Access URL: Not accessible (Development environment).

6.2.7 KPI Evaluation tool

6.2.7.1 Description

The KPI evaluation tool service is an application dedicated to the calculation, storage, and visualization of KPIs (not for all the pillars, where external calculators are used, e.g., comfort). The main aim of this tool is to provide a dashboard for decision-making and self-assessment of the interventions carried out in the city of Vitoria-Gasteiz, more particularly, in the Coronación district. In this sense, the tool calculates the KPIs defined within WP7 of the SmartEnCity project (refer to D7.3 Evaluation protocols v2.0). The service includes the main pillars of the project depicted as follows:

- Energy: this pillar includes the energy use per building, primary energy, and CO_2 emissions due to energy usage. Both thermal and electrical energy are considered within this pillar through diverse energy carriers.
- Comfort: although comfort is obtained at dwelling level within the application “Data analysis of comfort conditions for ESCO support App” (see section 6.2.3), this pillar analyses the comfort levels in an aggregated way, per building, thus, avoiding issues with the GDPR for a public evaluation.
- Mobility: the use of clean and green vehicles, as well as the associated chargers, contributes to the reduction of the greenhouse emissions, therefore, the impact of the project actions related to these elements is included within this pillar.

- ICT: the Vitoria-Gasteiz CIOP itself is also globally assessed in terms of performance and other metrics such as scalability and extensibility.
- Social & Citizen: within this pillar, the social acceptance and citizen engagement are evaluated to obtain the satisfaction, involvement, and degree of commitment of the citizens as core of the Smart City.
- Environmental: Life Cycle Analysis (LCA) is another important aspect to remark, which is included as environmental pillar for the evaluation of sustainability aspects focused on the reduction of environmental impacts due to the district intervention.
- Economic: finally, economic aspects to evaluate the cost effectiveness of the district renovation and mobility interventions and the citizen engagement actions are also included, such as investments, cost savings associated to energy savings, return of investments and other indicators to provide some figures in terms of economy.

6.2.7.2 Architecture definition & Infrastructure

The architecture followed for the implementation of the service is shown in Figure 52. It is a simplified view, but better for understanding. It is composed of four main layers:

- Data sources, which provide data for the calculation. In this sense, multiple and diverse data sources are identified. Among others:
  - Dwelling level meters, where comfort parameters, as well as electricity consumption are measured.
  - Building level measurements, where thermal energy consumption (either gas or district heating) is obtained.
  - eV (including their associated charging points).
  - Citizens, whose input data are mainly questionnaires.
- Business Intelligence, which is in charge of the scripts for automated calculation. This layer is responsible for connecting the data sources or making use of already collected data from those data sources so as to execute commands and calculation methods combining such a piece of information to obtain the different KPIs. These scripts also contain the static data that are necessary, such as conversion factors, buildings features (e.g., heated area) and other for the proper calculation.
  - It is important to highlight here that, even though the architecture below depicts a single element, there are multiple and distributed scripts. For instance, MON makes use of the comfort app for running the script in charge of the aggregation, GIS deploys its own script for mobility and CAR calculates the rest of pillars.
- KPI database, which persistently stores the KPIs calculated in the layer above. This layer is prepared by MON.
- Finally, the dashboards, which obtain filtered data from the KPI database in order to create the visualization and decision-making tool.

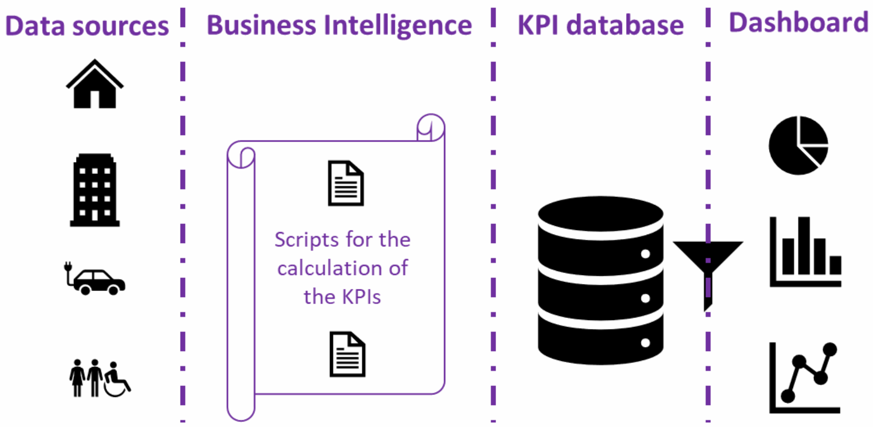


Figure 52: Architecture for the KPI evaluation tool

In terms of infrastructure, the tool takes advantage of the aforementioned LKI infrastructure (see section 5) and deploys the application within this infrastructure. Such as it is observed in Figure 53, raw data (represented by the binary code) are collected either using the scripts or other data collection mechanisms. But, in both cases, the storage is performed in the LKI databases servers and accessible via API.

Next, the scripts run in dedicated virtual machines (depending on the developer, i.e., MON for the comfort pillar, GIS about mobility, while CAR the others) hosted in the aforementioned servers. In the case, data are collected with external tools, the script queries the proper database to obtain data, otherwise the script gathers raw data, stores it in the proper database and makes use of these data to run the calculator of KPIs.

Once KPIs are calculated, these are stored in the KPIs database hosted in the databases server, whereas, PowerBI has been selected as tool for the dashboards and this is running in the cloud. Therefore, it connects the LKI servers via API to obtain the filtered KPI data and provide the dashboards to the virtual machine where the web server is deployed. Finally, via a Web browser (note the tool is prepared to be accessed via computer browser and not optimised for mobile devices) the tool is accessible.

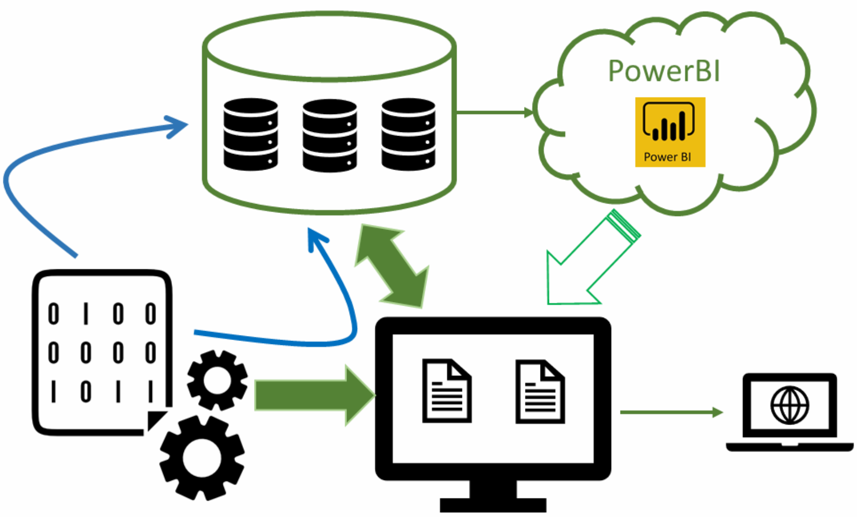


Figure 53: Infrastructure for the KPI evaluation tool

6.2.7.3 Data Structure

About data structure, it should be split into input and output data. Starting with input data, multiple data sources are connected as mentioned before. First of all, the building characterization, which, basically consists of a table with the fields that are explained in Table 5 and provided in Excel format as an argument of the “main” method.

Table 5: Building characterization input data for KPI evaluation tool

Field	Description	Example
BuildingID	Code for identifying the building. It is taken from the cadastre number but simplified.	59_473_7
Address	Address of the building.	EULOGIO SERDAN 8
Latitude	Latitude of the building location.	42.85223102
Longitude	Longitude of the building location.	-2.675011765
Carrier1	Type of main carrier for thermal energy, which is the one covering most of the dwellings in a building block.	Gas
m2_1	Square meters heated by the main carrier.	600
Carrier2	Carrier 2, if available. Within a building block, some dwellings could be based on a carrier, while other could use a different carrier.	Electricity
m2_2	Square meters heated by the secondary carrier.	150

Regarding dynamic data, the thermal energy measurements are taken from one of the databases within the CIOP server, accessible through an API provided by MON. Table 6 summarises the structure of such a database table. The format of the data is JSON.

Table 6: Thermal measurements data structure of the KPI evaluation tool

Field	Type	Description
thermal_measurement_id	integer	Unique ID for the thermal measurement.
timestamp	integer	Date in Unix timestamp format for the thermal measurement.
value	float	Value for the thermal measurement.
resource_id	string	Resource (i.e., building code) to relate the thermal measurement to the building.
carrier	string	Carrier for the thermal measurement (e.g. gas, biomass…)

In terms of mobility, iPanel from Datik (https://datik.ipanel.cloud/#/signIn) is used as data source. Basically, it contains the eBuses data, as well as the charging points associated to these eBuses. The data are obtained in JSON format from the API offered by iPanel. Table 7,

Table 8 and

Table 9 summarises the fields obtained for the eV, chargers, and charges, respectively.

Table 7: eV data structure for the KPI evaluation tool

Field	Description
vehicleId	Identifier of the eV.
vehicleCode	Code for the eV (i.e. license plate of the eV).
date	Date when the register was obtained.
odometryDistance	Travelled distance in km.
duration	Duration of the trip.
speedAverage	Average speed for such a trip in km/h.
averageConsumption	Average consumption during the trip in kWh/km.
consumption	Total consumption in kWh.
regeneration	Energy that has been regenerated in kWh.
auxConsumption	Consumption of the auxiliary systems.
tractionConsumption	Consumption due to traction engine.
insideTemperatureAverage	Average of the indoor temperature during the trip.
outsideTemperatureAverage	Average of the outdoor temperature during the trop.
dieselCO2Emissions	CO_2 emissions due to the diesel engine.
electricCO2Emissions	CO_2 emissions due to the electric engine.

Table 8: Chargers data structure for the KPI evaluation tool

Field	Description
chargerId	Identifier for the charger.
chargerCode	Code to namely identify the charger.
acRequired	If alternative current is required (True or False).
vehicle	Status of the charger, if a vehicle is charging, this field identifies such a vehicle.

lastUpdate	Last update of the charger status.
globalSoc	State of Charge.
power	Power of the charger.
geolocationLatitude	Latitude of the location of the charger.
geolocationLongitude	Longitude of the location of the charger.

Table 9: Chargers data structure for the KPI evaluation tool

Field	Description
chargerId	Identifier for the charger.
chargerCode	Code to namely identify the charger.
date	Data of the charge.
chargeNumber	Number of the counted (cycles) for the total charges.
duration	Duration of the charge.
durationAverage	Average of the charges' duration.
suppliedEnergy	Supplied energy during the charge in kWh.

The other vehicles to be included in the KPI evaluation tool will provide the needed data in Excel format due to the manual collection requirement (not automated mechanisms). Moreover, ICT, LCA, social and economic pillars are also Excel sheets with the values of the KPIs. Note these pillars either do not have complex calculations and the values are directly obtained from the source or calculated on dedicated excel sheets (as the economic protocol) or specific software packages (as the LCA one).

Regarding the output data, as mentioned before, the information is stored in a KPI database, whose structure was shown in section 5.2.8. As observed, there are several tables to store the information related to the KPIs. Within the tables “kpi”, “unit” and “kpi_type”, the contextual information of the KPIs is stored. That is to say, the list of defined KPIs according to WP7, their measurement units and the types (e.g., energy, comfort…). The table with the calculations is “measure”, which persistently stores the values for the KPIs. The fields are self-explanatory, but it is worth mentioning the field “resource_id”, which is used for the link between KPIs and buildings (through the cadastral codes), KPIs and vehicles (through license plate) or any other useful relationship.

6.2.7.4 Tools

The KPI evaluation app relies on some tools for the implementation of the service. In short:

- PowerBI has been the main software tool used for the creation of the dashboards, which allows the integration of data, creation of analytics and embedding the results into a Web page.
- Python, with the numpy and pandas libraries, has been the main tool for the scripts in terms of communication with APIs for data gathering, KPI calculator implementation and data management (i.e. cleaning, preparation…).
- Django is the Web framework used for the creation of the Web interface and the integration of dashboards.

6.2.7.5 User guide (Location, functions, etc.)

The KPI evaluation tool is already available on the Vitoria-Gasteiz CIOP, and its direct link is https://vitoria-gasteiz.smartencity.eu/kpiservice/. It is important to remark that the tool requires log-in in order to access, as illustrated in Figure 54. The main reason of this authentication process is to maintain access control. In this sense, if a user does not have credentials, just clicking "Create an Account", an automatic mail is prepared to be sent to the tool administrator who will accept or reject the request. In case of acceptance, credentials will be provided.

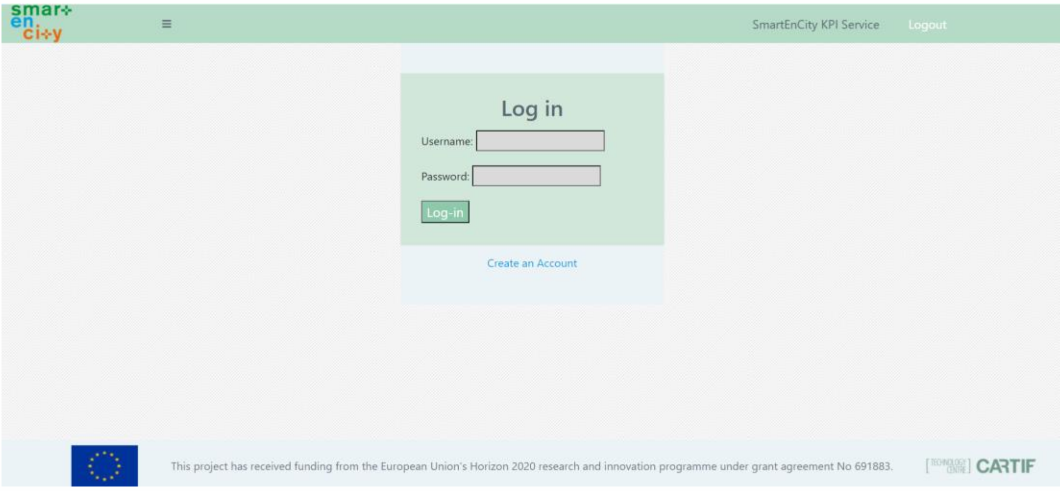


Figure 54: Access page for the KPI evaluation tool

Once the access is granted, the entrance is made through the main page depicted in Figure 55.This main page shows the buildings that are part of the KPI evaluation tool and is accompanied by a summary of some key general figures as follows:

- **Total primary energy** provides the entire primary energy of the district aggregated along the years with data. In the example, it can be observed 8,960 kWh/m^2. Disaggregating this value, and considering the current 4 years of data, the primary energy of the district would be 2,240 kWh/m^2. Given that there are 25 buildings (2 out of the 27 are electricity and gasoil based), an average of89.6 kWh/m^2 is obtained per building. At the moment of writing this deliverable, only thermal energy is available for baseline, while electricity will be included in next releases with the new installed electricity meters.

- **Primary energy per carrier** provides an overview of the different sources. At the moment of writing this deliverable, only gas data were available, and therefore 100% of the thermal energy comes from gas. In the next releases, when district heating is available, the percentage of sources will change.
- **Thermal comfort per building** will show the summary of comfort parameters in average per building. At the moment of writing the deliverable, comfort KPIs are not yet aggregately calculated.
- Finally, **CO_2 emissions avoided by the eV** in order to provide an overview of the mobility results. Similar to some cases before, the availability of the mobility data is still pending, and therefore, this figure will be available on next releases.

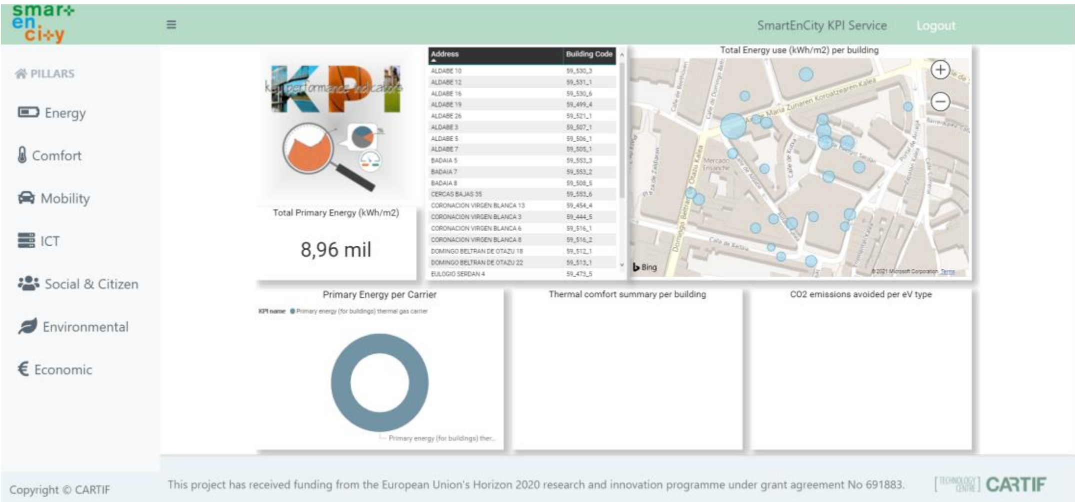


Figure 55: Main page of the KPI evaluation tool with the summary of KPIs

Entering into the details per pillar, energy is the first one (Figure 56). It provides 4 possibilities accessible via the buttons located on the top right corner: energy use (marked in orange), primary energy, CO_2 emissions and district heating (this is so far disabled until district heating data become available). Within this dashboard, by default, the information for all the buildings during the whole period is presented, but it allows filtering by data periods and/or building(s). Then, about the rest of graphs:

- The load curve for the thermal energy is drawn (note that electricity is still not available and will be in next releases), being capable of clearly identifying the profiles and consumption patterns along the years and/or seasons.
- The aggregated energy use analytic provides an overview about the distribution of this energy use along the years of data, making it simple to compare the year-by-year energy consumption.
- Distribution of energy, just as before, presents the percentage of contribution of each carrier, being currently only gas, as gas consumption is only available.
- Finally, numeric summary of the total consumption, whose value is modified according to the specific filters applied for building and dates.

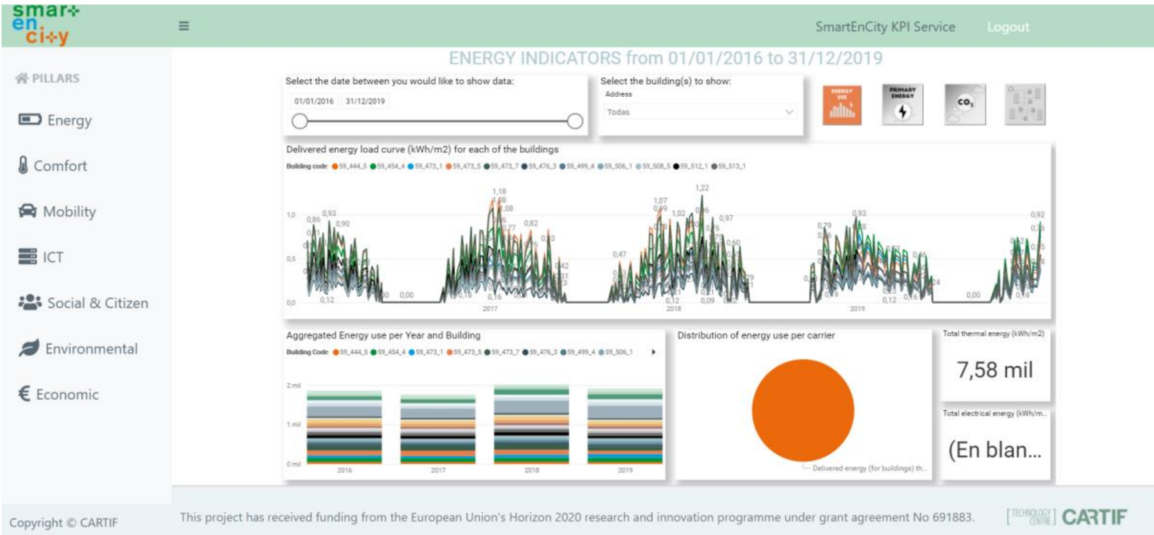


Figure 56: Energy dashboard of the KPI evaluation tool

Figure 57 below depicts the dashboard for the ICT pillar. It consists of two parts. Firstly, the graph with the KPIs, which are normalized by its own maximum value, and therefore, always between 0 and 1 so as to keep the same scale in the visualization (e.g., data size could be much higher than APIs number). Despite this, with the aim of giving the size difference, the blue dots indicate the real size of the KPI value. For instance, the highest values (thus, bigger blue dots) correspond to storage capability and amount of generated data. By browsing the graph values, statistical parameters can be obtained like median, quartiles… from the values for each of the services. On the other hand, browsing the blue dots, the details for the service and its values are gathered (see Figure 58 left side for statistical values, and right side for service's details).

The second part of the dashboard provides two tables. The upper one summarises the KPI values for the performance of the Vitoria-Gasteiz CIOP, such as response time, scalability, extensibility and so on. The bottom one translates the numeric values for those KPIs that are expressed in text format, such as the types of measurements and services. Meanwhile, graphically speaking, these values are translated into numeric count of distinct measurements of services, this table provides the details.

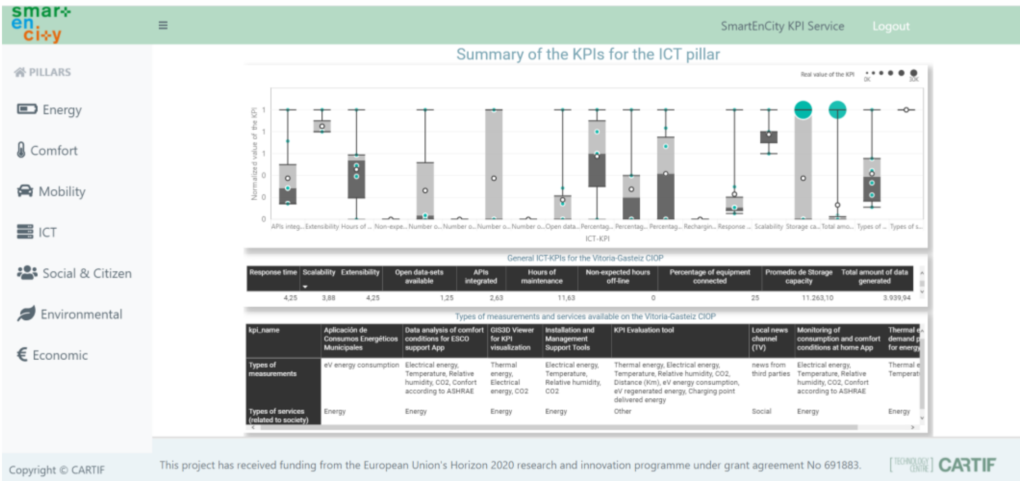


Figure 57: ICT dashboard of the KPI evaluation tool

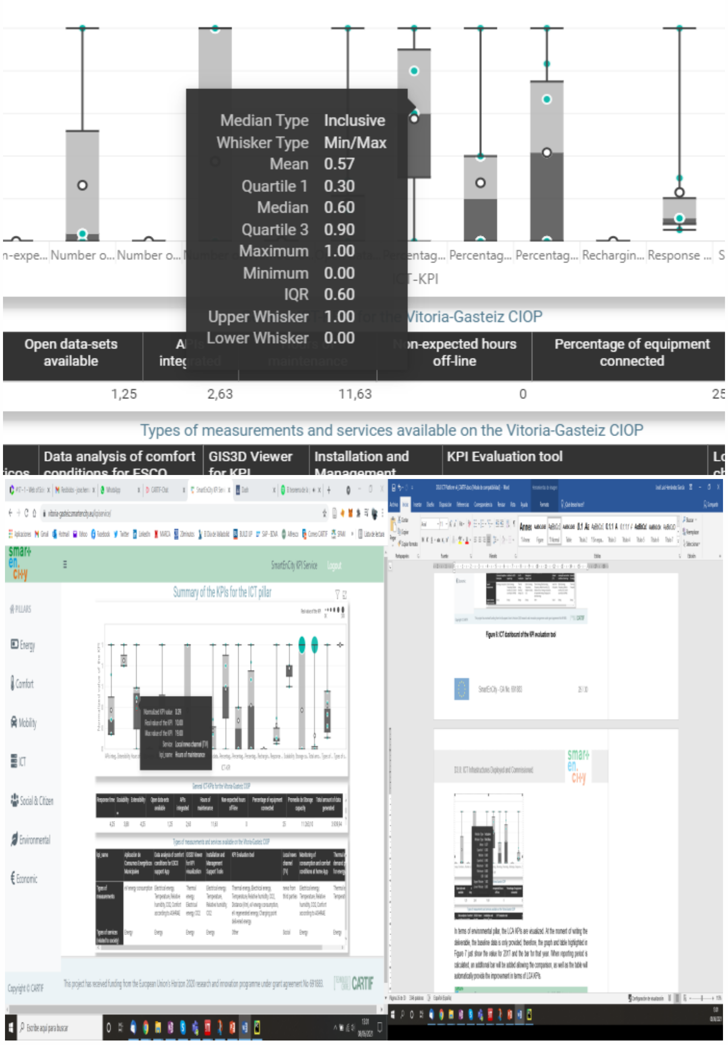


Figure 58: Details on the ICT KPIs

For the case of the environmental pillar, the LCA KPIs are visualized. At the moment of writing the deliverable, the baseline data is only provided as the final numbers will be later on calculated. Therefore, the graph and table highlighted in Figure 59 just show the value for 2017 and the bars for that year. Once the reporting period for LCA is over and the KPIs final values are calculated, an additional bar for each one will be added allowing the comparison, as well as the table will automatically provide the improvement in terms of LCA KPIs.

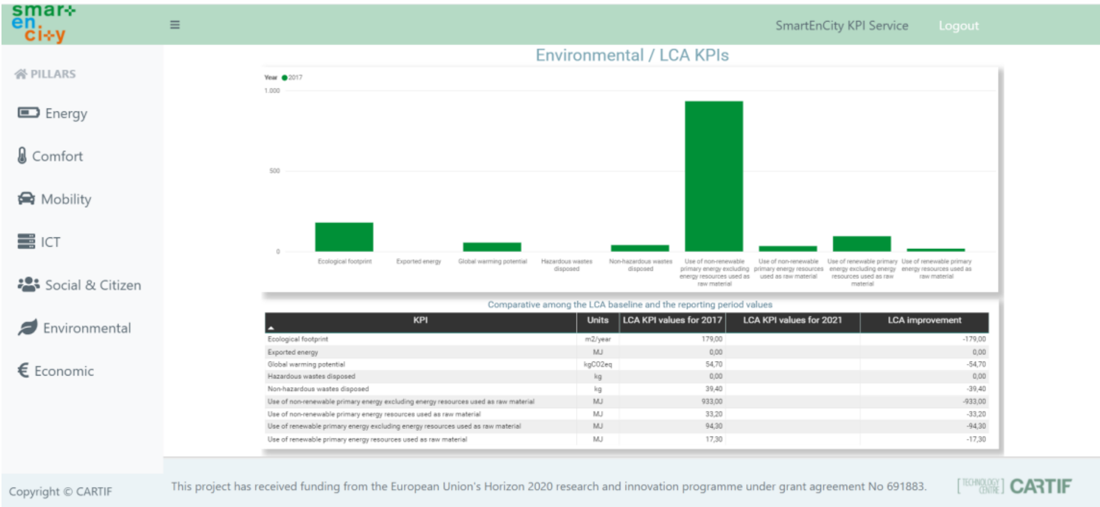


Figure 59: LCA dashboard of the KPI evaluation tool

The rest of the pillars, at the moment of writing this deliverable, do not contain data. Mobility is still in test phase; therefore, it is expected to start collecting data fluently in the next months. Comfort pillar will be upgraded in the upcoming days/weeks. The social, citizen and economic pillars need values collected at the end of the project, once interventions have been completed and the data gathering period has finished or nearly to be so. Thus, next releases will be including additional pillars and data. In the meantime, when accessing one of those pillars, a message informing about it is printed as illustrated in Figure 60.

Figure 60: Screen for the pillars with no data within the KPI evaluation tool

6.2.8 Thermal energy demand prediction tool for energy planning

6.2.8.1 Description

The thermal energy demand prediction tool is a service whose aim is to provide an energy demand forecasting at multiple time spans in order to plan the energy generation and distribution according to this expected demand. The tool makes use of machine learning techniques to learn about the energy consumption patterns from historical data in order to predict the behaviour of the buildings. For that end, two techniques are applied:

- Clustering in charge of classifying the buildings according to their features. This technique groups data sets in contrast to similarities in data features. In this case, building characteristics that affect the energy demand have been selected. The explanation of these features is included below.
  - Biomass DH: it responses (Yes/No) to the question “is the building connected to the DH?”
  - Height: this feature provides the number of floors that the building has.
  - Heated Area: total area that is heated within the building.
  - Carrier: when the building is not connected to the biomass district heating, it will need a different heating system, such as gas boiler. If the Biomass DH answer was “Yes”, this feature is neglected and assigned to “biomass” automatically by the algorithm.
  - Orientation: solar gains depend on the orientation. The 'N', 'NE', 'NW', 'S', 'SE': 0.75, 'SW': 0.75, 'E': 0.5 and 'W' orientations are possible.
  - # Adj. Block: number of adjacent blocks to the building. In other words, the façades that are not affect by climate conditions like winds or with less solar gains.
  - # Shadowed Façades: number of façades with shadows, that is to say, limited solar gains.
  - Retrof. Status: status of retrofitting (insulation), with three possibilities (‘Non-renovated', 'In-progress' and 'Renovated').
  - % Window Façade: percentage of windows that are part of the façades.
  - % New Windows: percentage of windows that are new (i.e. better insulation).
  - % Double Windows: percentage of double-glazing windows.
- Regression, whose objective is to obtain an equation that relates the dependent variables (i.e. energy demand) with independent variables (i.e. climate conditions in form of Heating Degree Days). At the moment of writing this deliverable, only linear regression is available, while next releases will provide additional models with their respective accuracies so that scalability and extensibility could be ensured (i.e., depending on the available data, some models fit better than others).

By combining both techniques, the tool does not need to train the energy demand for all the buildings, but just one representative building per cluster. The major advantage is the response time as the number of models is highly reduced.

6.2.8.2 Architecture definition & Infrastructure

The architecture shows a very simple structure. It is based on three layers: data, machine-learning algorithms layer and visualization, in form of a dashboard. Data layer includes two sets, one containing information for the building characterization (static data), the other with the historical data of heating energy per building (dynamic data, which should be updated on a regular basis). These datasets supply the input for the aforementioned machine learning techniques (i.e., clustering plus regression) and, finally, the results are shown in a dashboard.

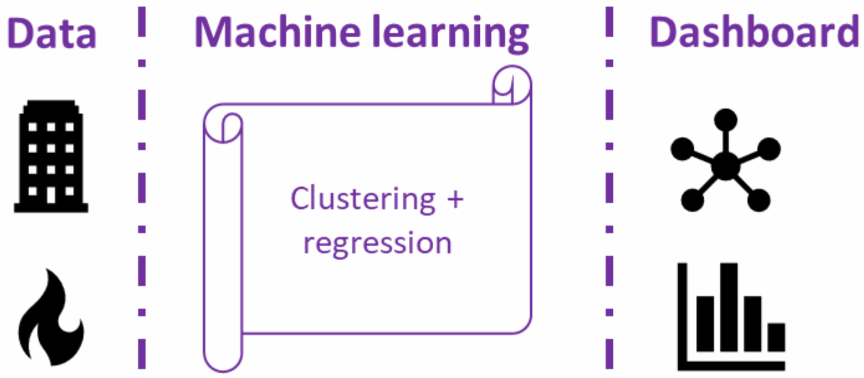


Figure 61: Energy planning tool architecture

The infrastructure is the use of the databases in the Vitoria-Gasteiz CIOP (as mentioned before) and a dedicated virtual machine where the algorithms are executed.

6.2.8.3 Data Structure

Regarding the data structure, the thermal energy measurements follow the same format that was explained in the KPI evaluation tool (see Table 6). Moreover, the building characterization data is provided within an Excel sheet that contains the aforementioned building features in columns, and where each of the buildings to be analysed are represented by a different row. In addition to the previous characteristics, the number assigned to each building, the building code, address, latitude, and longitude are also included. This Excel sheet is a requirement for the first execution, while, afterwards, the tool manages internally the data structures as data frames that can be exported as csv files.

6.2.8.4 Tools

In this case, the tools that have been used for the implementation are completely based on Python language. In short:

- Scikit-learn libraries have been used as machine-learning tools.
- Plotly-Dash framework provides the tool for the visualization.
- Numpy-Pandas libraries have been used for the data management, cleansing, internal storage…

6.2.8.5 User guide (Location, functions, etc.)

The tool is already embedded into the Vitoria-Gasteiz CIOP and, therefore, accessible from the main portal. Nevertheless, the direct link to the tool is https://vitoria-gasteiz.smartencity.eu/energyplanningtool. This tool does not require log-in of users and it has an open, free license, use. Then, when accessing, the front page shows the buildings that are part of the analysis, which are also geolocated as illustrated in Figure 62. When tapping on one of the buildings, its detailed information (building code, estimated demand and coordinates) is shown (Figure 63).

In the first execution of the tool, an initial clustering is rendered, selecting the number of clusters under the best score condition. Under this clustering result, the estimated energy demand is forecasted for the next 7 days by default, therefore, it should be kept in mind these aspects when reading the value in this main screen.

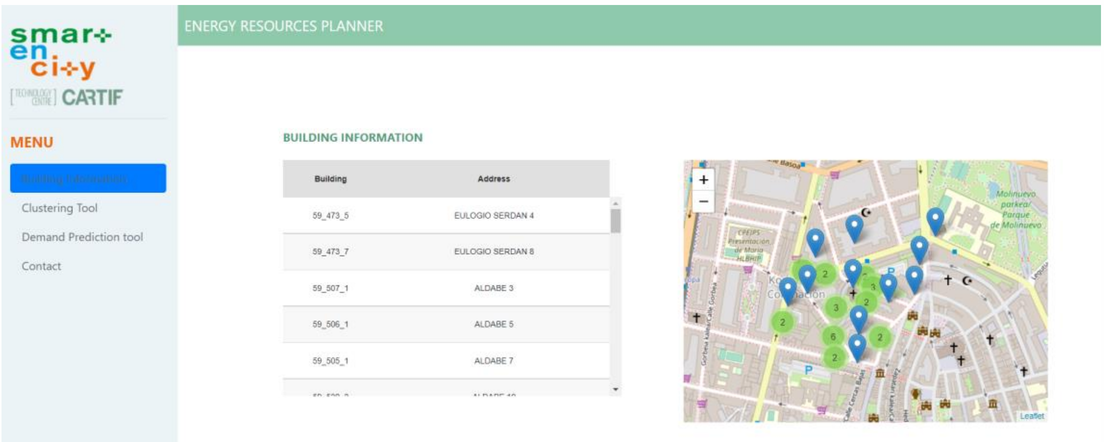


Figure 62: Front page of the energy planning tool

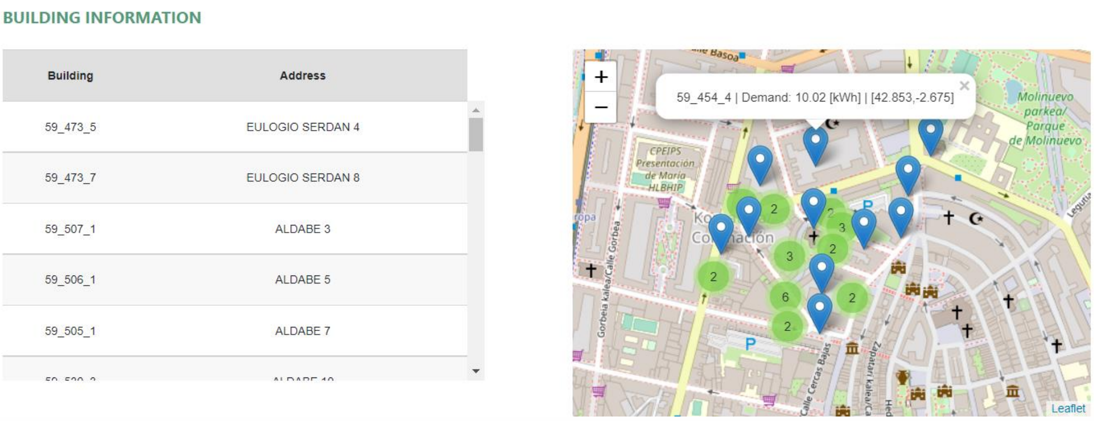


Figure 63: Data visualization of one building of the energy planning tool

As stated above, an initial number of clusters is selected, but the user has the capability of selecting the number between 3 and 7. The selection has been made based on accuracy vs complexity. Note that clustering is made to reduce the complexity, therefore, although the higher number of clusters, the higher accuracy, the complexity also increases. When changing the number of clusters, also the regression is re-trained. The reason is due to the selection of one representative building per cluster, therefore, if the clustering changes, the regression should follow the new grouping.

Secondly, there is an editable table. This table shows data according to the current building features. When these have not been changed from the first execution, its values will be the same than the Excel sheet. However, after any change, the data frame will be replaced. If the user wants to keep record, it is possible to download the data by clicking "Export data" button. Then, when the user wants to change a value due to changes in the building or mistakes, he simply must edit the field of fields to be modified and click on "Process new data". The tool re-calculates the clustering and regression models for the selected number of clusters. It is also possible to add buildings ("Add row") or remove buildings (clicking on the "x" on the left of each row). Also, it is possible to add new features (although, in the current release, it is disabled) by adding a new column. Anyway, after any change, the new data need to be processed and models be re-trained.

Figure 64 depicts how the clustering screen looks like. As observed, each of the clusters is represented in a different colour to identify the different elements being part of them. When pointing the mouse over the dots, the clustering value according to the result of a Principal Component Analysis (PCA) technique is shown. This technique allows reducing the dimensions of the data to be printed, in this case, two-dimension data. The PCA "coordinates" are complemented with the number of building in order to univocally identify the element and the cluster to which it belongs.

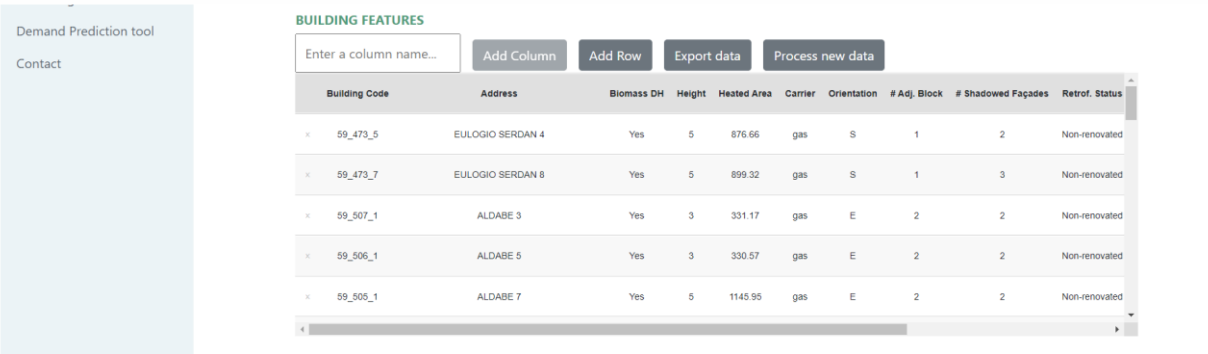


Figure 64: Clustering screen for the energy planning tool

The energy forecasting is provided on the following screen, shown in Figure 65. As mentioned before, 7 days option is the default selected value, but the user has the ability to select from one day to one-week estimation. In the snapshot, 6 days are selected. Then, the total aggregated demand is obtained. In the example, 386 kWh are estimated. Below the summary, there is a graph with the details per building. Each of the buildings is identified by a number. Additionally, information is included about the cluster to which it belongs (each one of them, represented by a different colour), and the address. By hovering the mouse over the bars of the graph, the estimated energy demand for the selected days is also shown.

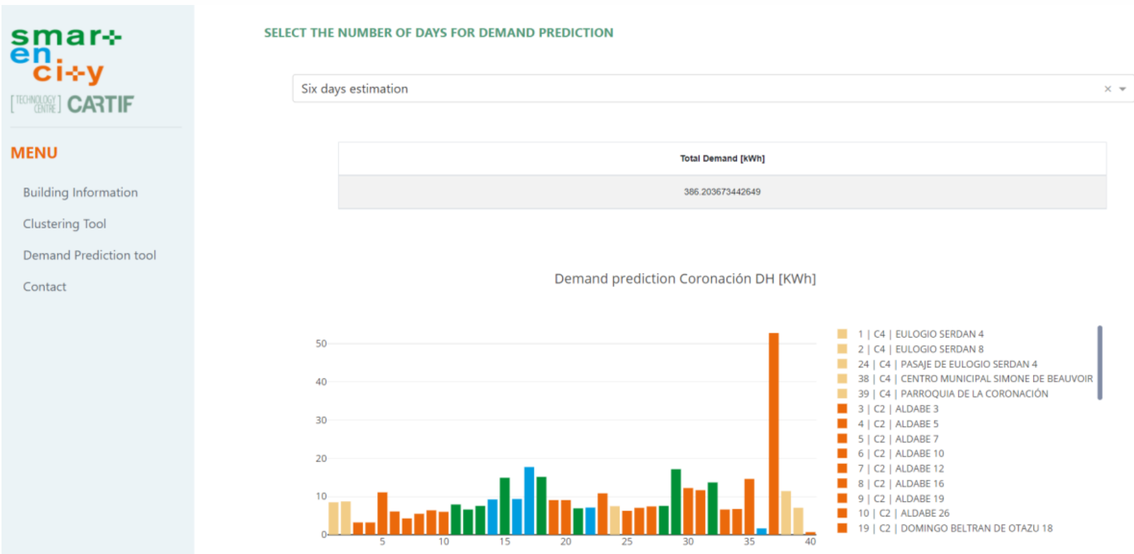


Figure 65: Energy demand forecast screen for the energy planning tool.

Finally, there is a last screen with the contact details and the EU project funding information, such as depicted in Figure 66.

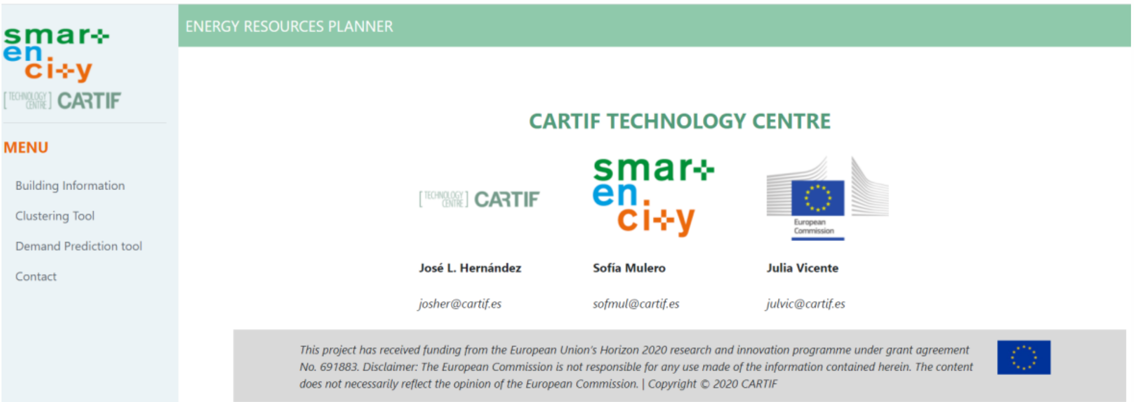


Figure 66: Contact and EU flag for the energy planning tool

6.2.9 GIS3D Viewer for KPI visualization

6.2.9.1 Description

The GIS3D Viewer for KPI visualization is a web application over Cesium.js, it is developed using Tecnalia CITYMIRROR VIEWER. This application starts with a login. Once we are logged, the application loads all the buildings in Vitoria-Gasteiz city (+40,000) using OGC 3DTiles, which allows a faster and non-blocking loading of the application. The 3D model of Vitoria-Gasteiz follows the OGC CityGML standard. Three types of indicators have been added to the model, building, energy, and solar indicators, these last two were calculated and added to the 3D model database, all these indicators are loaded into the 3D Model and can be displayed with building information. The Viewer also allows us to launch graphical queries on these indicators applying different styles to the buildings. Finally, the Viewer has a connection to the tool "6.2.7 KPIs Evaluation Tool", the connection has been made through Swagger API, of all the KPIs, only the most relevant at the city level are shown in the Viewer, Selected KPIs were time series, this time series has been converted to monthly KPIs for visualization purposes and loaded onto a chart.

Figure 67: GIS3D viewer preview

6.2.9.2 Architecture definition & Infrastructure

The architecture implemented in the GIS3D Viewer is shown in Figure 68: GIS3D viewer architecture. In the figure we can see the four main layers and the connected external services:

- Data Sources, shown in yellow at the bottom of the figure, there are mainly three types of data.
  - Building information, retrieved from cadastre (https://catastroalava.tracasa.es/?lang=en) and public building information.
  - Energy consumption data at building level provided by Vitoria-Gasteiz.
  - Solar Potential, calculated.

- In the persistence layer, in the centre, coloured in green, the data is stored in the Postgres DB, explained in "Data Structure" section.
- The business layer appears in the top side coloured in blue, this layer is implemented using Java, the connection with the viewer is through REST Services. The most important service is the login service and the building KPI service.
- Finally, in the top left side appears the presentation layer, here is the 3D viewer web application, developed over Cesium JavaScript library. The functionality is explained in user guide. The 3D model is loaded using 3D Tiles technology and connection to the business layer is throw REST API, the connection to "6.2.7 KPIs Evaluation Tool" is also throw REST API, using Swagger.

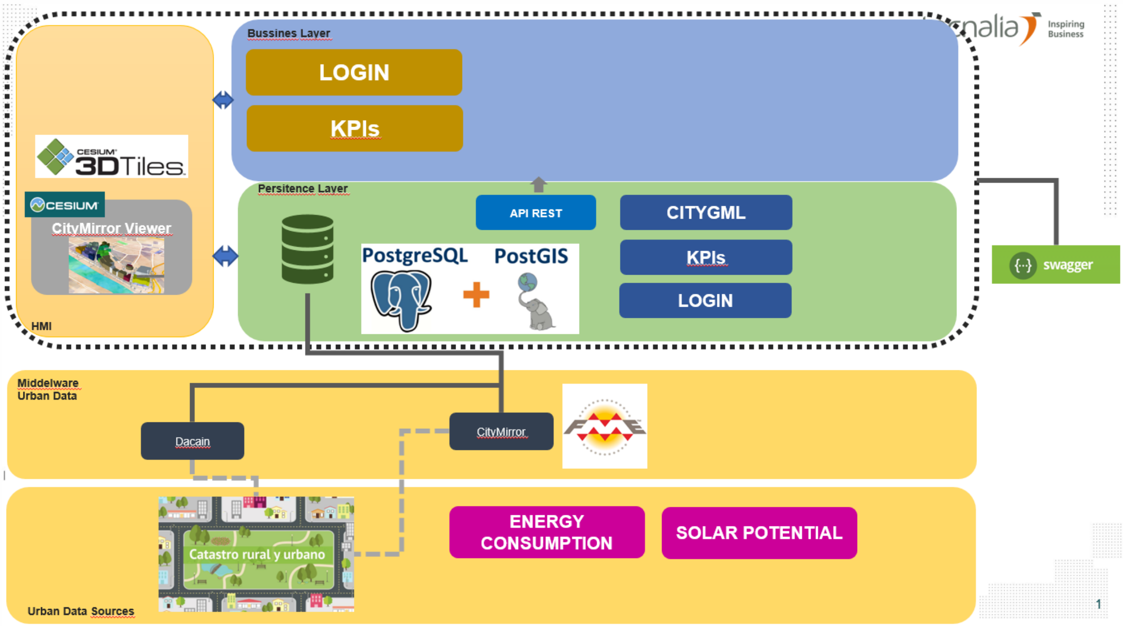


Figure 68: GIS3D viewer architecture

6.2.9.3 Data Structure

Postgres: an open-source object-relational database, is used for storing three types of data:

- Web application data, such as user accounts and internal data.
- City model in CityGML format.
- Energy consumption and solar potential KPIs, assigned to each building.

in the viewer for storing user accounts and for storing the building level KPIs and the monthly KPIs.

CityGML: The city model has been generated using OGC standard CityGML. CityGML is an open data model and XML-based format for the storage and exchange of virtual 3d city Models. As the model is very large, for visualization propose the model is exported to 3DTiles, OGC standard, 3DTiles can be loaded by parts, allowing faster model load, and not blocking the viewer.

6.2.9.4 Tools

The GIS3D viewer web application relies on some tools, listed below.

- CesiumJS: CesiumJS is an open-source JavaScript library for creating world-class 3D globes and maps.
- ChartJS: is an open-source JavaScript library for data visualization, making easy adding interactive graphs.
- FME Server: FME Server is a networked data transformation application that uses the same underlying technology as FME Desktop. It can operate on a local-area network or over the Internet. Allowing the transformation of the CityGML to 3D Tile format for visualization.

6.2.9.5 User guide (Location, functions, etc.)

The tools is accessible from the main portal https://vitoria-gasteiz.smartencity.eu/#/, the final URL will be https://vitoria-gasteiz.smartencity.eu/3dviewer/ during development and testing will be also accessible through http://projects.hei-tecnalia.com/SMARTENCITY/. The first window that appears is the login Figure 69: GIS3D login window.

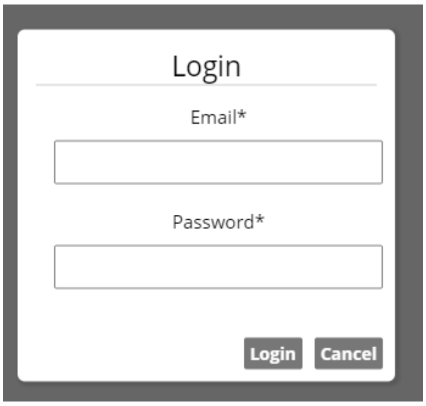


Figure 69: GIS3D login window

Once we are logged, we enter in the viewer main window Figure 70: GIS3D main window, the earth globe and Vitoria-Gasteiz 3D model are loaded. All buildings are geolocated and have the information of the project.

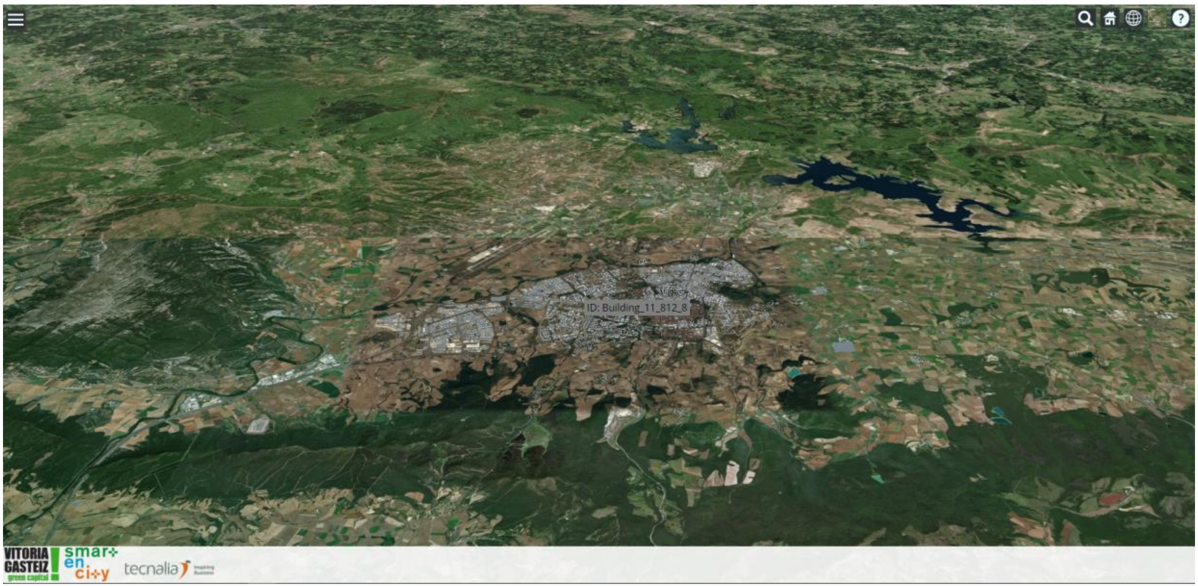


Figure 70: GIS3D main window

In top right part of the application, we can see cesium buttons, allowing us to fly to a street, returning the view to home or change map options like photometry or terrain.

In top left part of the application there is a hamburger menu button, if we click it the viewer menu appears, Figure 71: GIS3D menu.

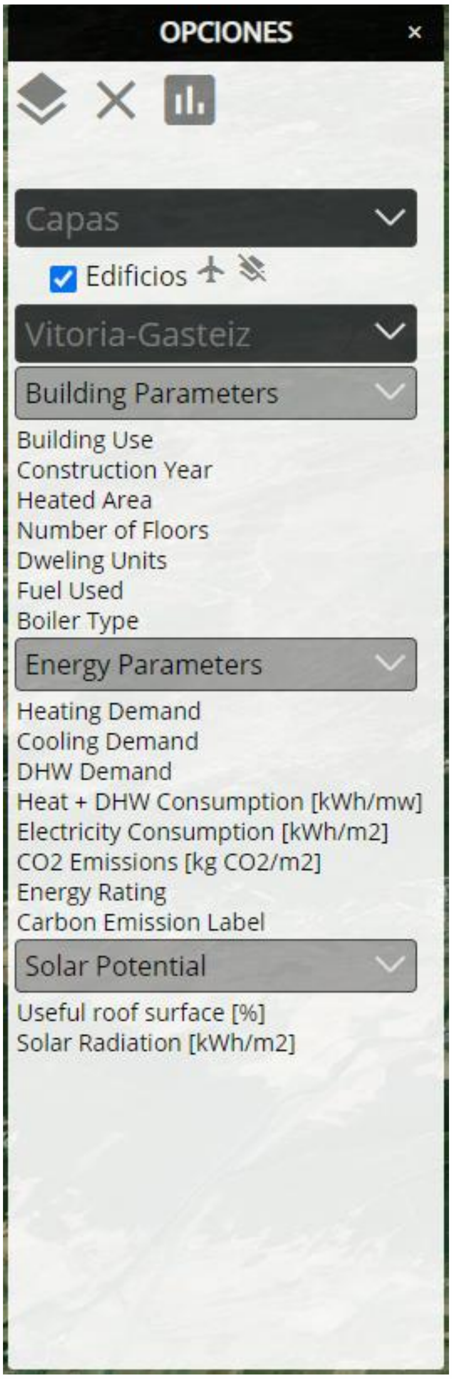


Figure 71: GIS3D menu

In this menu we can apply different styles or disable it from 3D Model, we can also enable/disable layers or put it transparent. In the final part of the menu, we can see the queries, these queries apply a different style to the buildings according to their values, which are loaded from de database. There are three differenced types of queries, "Building Parameters" for information that is in already in the model, "Energy Parameters" for energy type queries and "Solar Potential". For launching the queries, we just need to click in the name. Here we can see an example of solar potential, useful roof surface %, Figure 72: GIS3D query example.

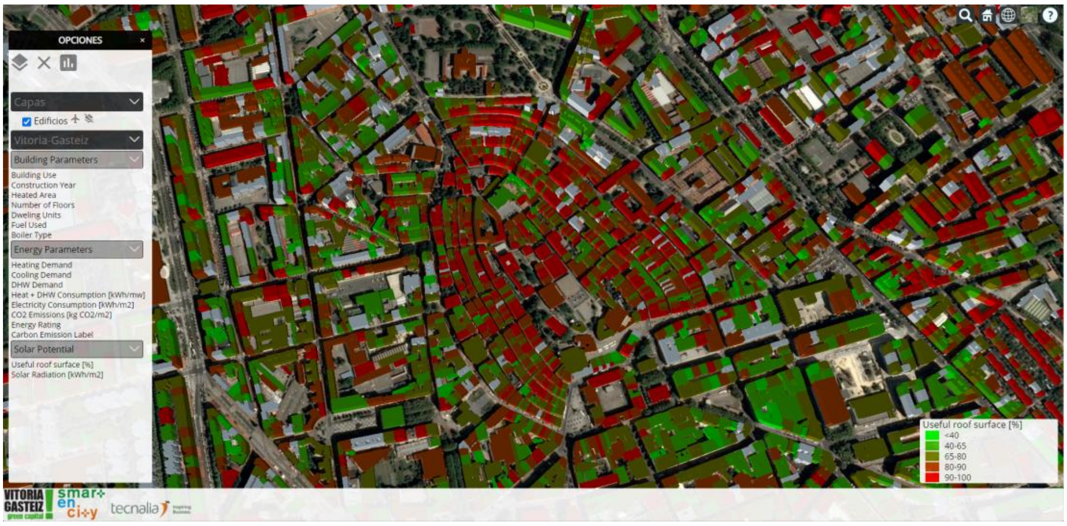


Figure 72: GIS3D query example

If the query is launched and we put the mouse on a building, the value for that query will be shown. We can also click the close button that appears in the menu for disabling the query and take the 3D model back to white again.

The next option is to click on a building, in that case, a side right menu appears, showing us all the information of the building, energy and solar potential included. Some of the parameters show up a help text if we pass the mouse over them, Figure 73: GIS3D building .

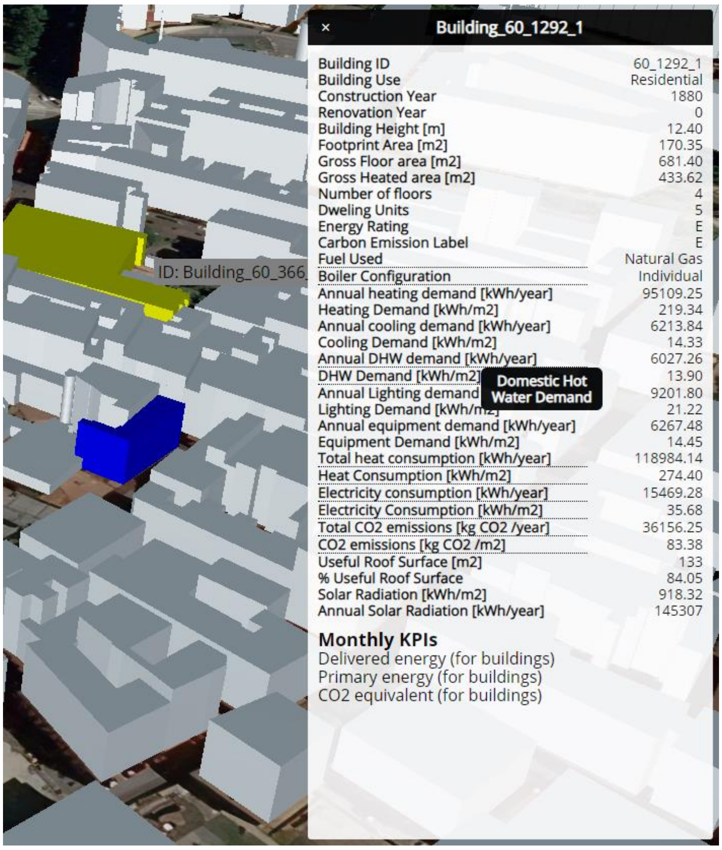


Figure 73: GIS3D building information.

Finally, inside building information menu we got the option of showing monthly KPIs calculated by "6.2.7 KPIs Evaluation Tool", this is made through a call to Swagger API and is only available for the case of study neighbourhood “barrio de Coronación”, Figure 74: GIS3D monthly KPI.

Figure 74: GIS3D monthly KPI

7 RECOMMENDATIONS

From the work previously described, a high degree of complexity can be inferred from both the backbone services and common framework, and the specific functionality deployed to support the actions executed in SmartEnCity project. This complexity could be a burden when the time comes to service, upgrade or migrate the CIOP components to a new location or IT service company.

With this in mind, and following the recommendations by the standardisation bodies, the architecture was developed to comply with the existing standards and to facilitate future maintenance, upgrading and migration operations when necessary. Therefore, elements like the data communication pathways for the dwelling monitoring services, or the interoperability with external systems like the BEI IT system were developed to provide the maximum flexibility and openness for future maintenance and upgrading operation.

These characteristics were provided to the CIOP by design, so scalability, replicability and upgradeability are inherent attributes to the system with the objective of easing as much as possible the evolution of the smart platform and the inclusion of additional city services.

The microservice architecture followed in the construction of the CIOP platform offers a scalable and flexible solution that eases the deployment of the platform in other environments, and it is able to grow along with the demand and usage of the services. Additionally, most components are available through Docker technology and code is stored in Github repositories. Thus, CIOP can be replicated in other cities or districts.

Currently, all components of the CIOP for Vitoria-Gasteiz are running at the facilities provided by LKI, the partner hosting the platform, data storage and running the dedicated services. Should it be required, the smart city platform may be seamlessly migrated to other facilities or hosting services upon request.