smartCEM

Smart connected electro mobility

D2.4.4 (Platform integration for Reggio Emilia pilot site)

Version number	1.6
Main author(s)	Leandro Guidotti, Daniele Pinotti (UNIMORE), Pietro Mascolo (ICOOR)
Dissemination level	PU
Lead contractor	ICOOR/UNIMORE
Due date	31.05.2013
Delivery date	

CIP - Information and Communications Technologies Policy Support Programme (ICT PSP)
Information Society and Media Directorate-General
Grant agreement no.: 297328
Pilot type B

Revision and history sheet

Version history			
Version	**Date**	**Main author**	**Summary of changes**
0.1	22.04.2013	Guido Di Pasquale (Pluservice)	Document creation
0.2	25.04.2013	Thomas Kemmere (Xerox)	Suggestions to improve structure
0.3	06.07.2013	Leandro Guidotti (UNIMORE), Pietro Mascolo (ICOOR)	Contribution on content
0.4	25.07.2013	Sergio Damiani (CRF)	EV-efficient driving application description
0.5	30.07.2013	Guido Di Pasquale (Pluservice)	EV-sharing management
0.6	16.08.2013	Pietro Mascolo (ICOOR)	Reggio Emilia key management system description. Executive summary and Introduction added.
0.7	26.08.2013	Pietro Mascolo (ICOOR)	Preliminary conclusions
0.8	28.08.2013	Pietro Mascolo (ICOOR)	Minor modifications, acceptance of preliminary contribution from partners and integration
0.9	29.08.2013	Michael Hubschneider	PTV contribution
0.91	28.10.2013	Daniele Pinotti, Leandro Guidotti (UNIMORE), Roberto Tola (CRF)	Integration of the new contributions and review
1.0	29.10.2013	Dorin Palanciuc	TeamNet contribution
1.1	04.11.2013	Thomas Kemmere (Xerox)	Internal Peer Review

1.2	07.11.2013	Leandro Guidotti, Daniele Pinotti (Unimore), Roberto Tola (CRF)	Review after Peer Review comments
1.3	20.11.2013	Leandro Guidotti, Daniele Pinotti (Unimore)	Document finalization
1.4	04.02.2014	Leandro Guidotti (Unimore)	Minor changes, last review of contents concerning REG PS and EV-Sharing management
1.5	05.02.2014	Guido Di Pasquale (Pluservice)	Document overall review
1.6	14.02.2014	Leandro Guidotti (Unimore)	Review and general fixing
	Name		**Date**
Prepared	Leandro Guidotti, Pietro Mascolo, Daniele Pinotti		29.10.2013
Reviewed	Thomas Kemmere (Xerox)		04.11.2013
Authorised	Fernando Zubillaga		
Circulation			
Recipient		**Date of submission**	
European Commission			
Pilot consortium			

Authors (full list)

Guido Di Pasquale (Pluservice)
Leandro Guidotti, Daniele Pinotti (UNIMORE)
Pietro Mascolo (ICOOR)
Sergio Damiani, Roberto Tola (CRF)
Michael Hubschneider (PTV)

Project Coordinator

Fernando Zubillaga
MLC ITS Euskadi

Clúster de Movilidad y Logística
Centro de Negocios CTVi, Oficina 1, 3-A
Lermandabide, 8. Polígono Industrial Júndiz
01015 Vitoria-Gasteiz
Tel.: +34 945 10 80 88
Email: fzubillaga@mlcluster.com

Legal Disclaimer

The information in this document is provided "as is", and no guarantee or warranty is given that the information is fit for any particular purpose. The above referenced consortium members shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials subject to any liability which is mandatory due to applicable law.

Table of Contents

Abbreviations

Abbreviation	Definition
3G	3rd generation of mobile telecommunications technology
AD	Activity Diagram
API	Application programming interface
APP	Application (software)
BEV	Battery Electric Vehicle
CAN	Controller Area Network
CIP	Competitiveness and Innovation Framework Programme
CS	Charging Station
CSV	Comma Separated Values
DB	Data base
DOD	Degree of Discharge
EV	Electric Vehicle
FC	Functionality
FTP	File transfer protocol
FTS	Forensic Telecommunications Services
GPRS	General packet radio service
GPS	Global Positioning System
GTFS	General transit feed specification
GUI	Graphical user interface
HMI	Human Machine Interface
HTTP	Hypertext Transfer Protocol
ICE	Internal Combustion Engine
ICT	Information and Communication Technology

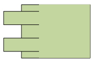
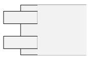
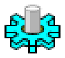
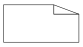
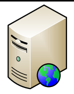
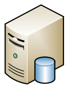
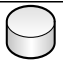
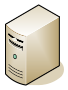
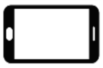
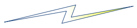
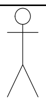
Abbreviation	Definition
IEEE	Institute of Electrical and Electronics Engineers
ITS	Intelligent Transport Systems
IVR	Interactive Voice Response
KPI	Key Performance Indicators
LTE	Long term evolution
MCB	Miniature Circuit Breaker
MSDU	MAC Service Data Unit
NC	Network classes
OBU	On-Board Unit
OCPP	Open charge point protocol
O-D	Origin-Destination
OSM	Open street map
PAYG	Pay as you go
PHEV	Plug-In Hybrid Electric Vehicles
POI	Point Of Interest
PT	Public transport
RCB	Residual Current Breaker
RCD	Residual Control Device
REST	Representational state transfer
RFID	Radio Frequency Identification
RQ	Requirement
SaaS	Software as a service
SC	Speed classes
SDK	Software development kit

Abbreviation	Definition
SH	Stakeholder
SIM	Subscriber Identity Module
SMMT	Society of Motor Manufacturers and Trader
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SOC	State Of Charge
TCP/IP	Transfer Control Protocol/Internet Protocol
UC	Use Case
UCAP	Ultra capacitors
UMTS	Universal Mobile Telecommunications System
UN	User Need
UNEW	Newcastle University
VPN	Virtual Private Network
WDM	Workflow and Demand Manager
WiFi	Commercial name of the wireless communication standard IEEE 802.11b
WP	Work Package
WSDL	Web Services Description Language
XML	eXtensible Mark-up Language

Table 0-1 Abbreviations

Graphical symbols

Table 0-2 describes the graphical symbols used for depicting the architectures.

Graphic symbol	Description
	smartCEM ICT component
	Existing or external ICT component
	Software service (e.g. interface)
	Data message
	Web server
	Database server
	Database
	General server. It documents, from a high level point of view, an entity that hosts a service.
	Smartphone or Tablet (PAD)
	Wireless communication (WiFi, Bluetooth, 3G, LTE, GPRS)
	Actor: end-user, operator, administrator, EV-driver

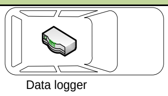
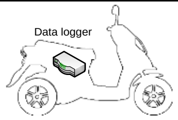
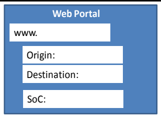
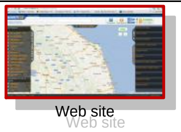
Graphic symbol	Description
	On-board unit, data logger
	On-board unit, data logger
	Generic web portal
	www.transportdirect.info web site
	Multimodal trip planning web site

Table 0-2 Graphical symbols

Executive Summary

Deliverable Platform integration for Reggio Emilia Pilot Site (REG PS) presents the adaptation and integration process for REG PS, following the guidelines provided by D2.3: *Guidelines and requirements local systems integration into smartCEM*. This deliverable is an executive project of the implementation of platforms for Electro Mobility services at Reggio Emilia Pilot Site.

smartCEM provides added value by setting up a common framework that allows combining the six electro-mobility services defined in the project (i.e. EV-charging station manager, EV-trip management, EV-sharing, EV-navigation, EV-efficient driving and EV-policy tool) into more complex services.

Services cooperate and exchange information in a harmonized way to provide users with the best user experience, contributing to raise the perception of being well assisted during all the travel experience with an electric vehicle and thus increasing acceptance of this kind of vehicles, which is still quite low in Italy.

Within the activities of Task 2.4, this deliverable describes how the smartCEM technical architecture is adopted by Reggio Emilia pilot site and how the local implementation is done.

smartCEM driving services will be installed onto an Android-based device mounted in-vehicle. Technical solutions were proposed to solve the encountered issues, allowing full data transfer with real time data collection and info visualization for end-users services (EV-sharing, EV-Efficient Driving and EV-Navigation) and also for back-office service (EV-sharing).

1 Introduction

This document is the outcome of Task 2.4: *Technical adaptation work for pilot sites: platform integration*, in case of Reggio Emilia pilot site. In this Task, the smartCEM partners work on the adaptation and integration of existing software platforms into one smartCEM platform. The work is focused also on the technical adaptation starting from the outputs of Task 2.1: *Requirements and specifications* and considering the design and integration work done in Tasks 2.2: *Develop reference architecture, standards and best practices* and 2.3: *Technical Architecture*. In Task 2.3, ICT providers have implemented software components and functionalities needed in the pilot sites (after the analysis conducted together with the local stakeholders) that meet the important requirement of being interoperable and re-usable between the different pilot sites. For this purpose a suite of software components (smartCEM deployment kit) has been implemented in view of the local integration.

Considering the encountered local deployment barriers, each pilot site will have a specific implementation of the smartCEM platform. Consequently, the objective of this document is to describe how the smartCEM services must be adapted and connected to existing local applications in case of Reggio Emilia pilot site. In addition, this document is strongly related to deliverable D2.3: *Guidelines and requirements local systems integration into smartCEM*, where the guidelines to implement the smartCEM services in each pilot site were explained.

The present document is structured in four main chapters:

- Chapter #2: Architecture. This section provides a graphical overview of the Reggio Emilia local architecture, how components are connected and interfaced locally. Moreover, the architecture updates and modifications are provided within this chapter.

- Chapter #3: Components implemented. Describes the instantiation of the smartCEM components in the local systems.

- Chapter #4: Interfaces and standards implemented. Describes the software interfaces implemented in Reggio Emilia pilot site.

- Chapter #5: Conclusion. This chapter provides a summary of the issues encountered during the implementation process.

2 Architecture

2.1 Proposed architecture

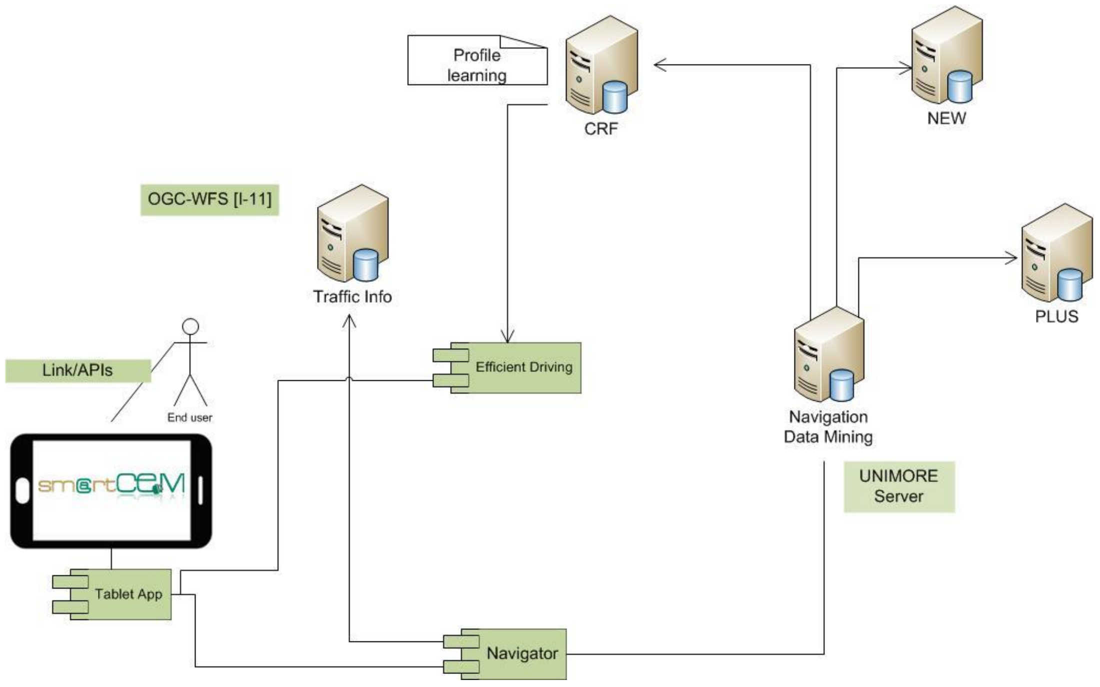


Figure 2-1 PS REG Architecture 1

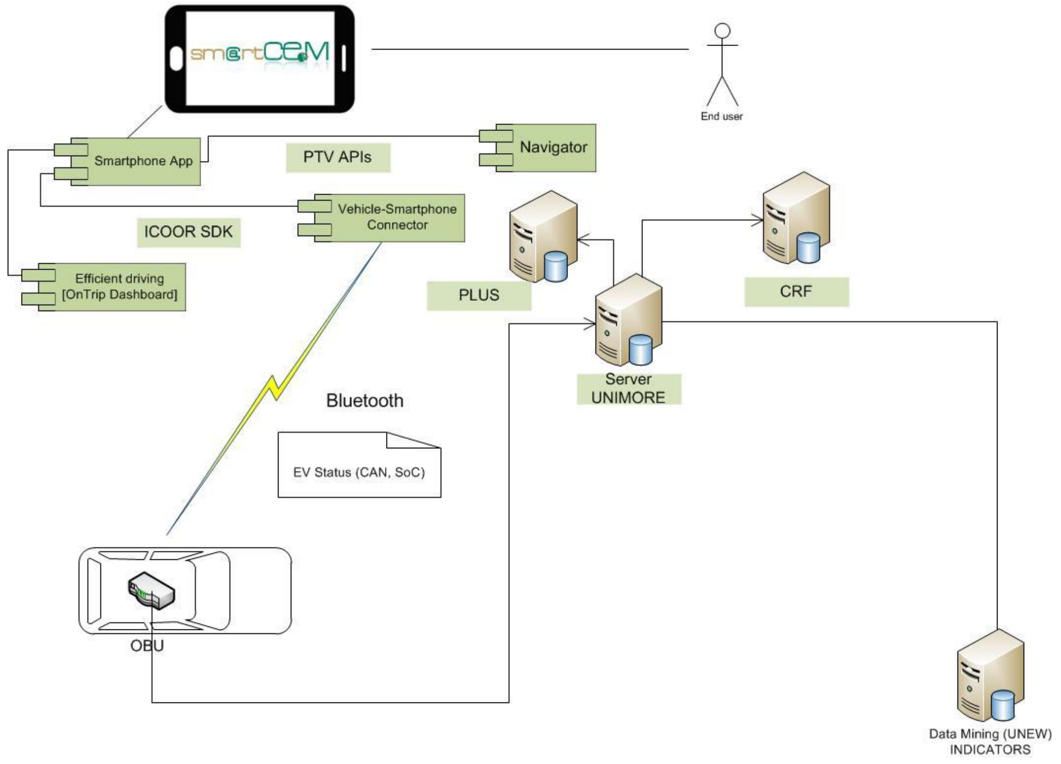


Figure 2-2 PS REG Architecture 2

2.2 Changes or adaptations to architecture

On-board units will equip EV-vehicles in order to collect usage data and transmit them to servers and Android tablet in order to be then used by Apps. First OBU will adapt analog signals to digital (i.e. to CAN bus) and another unit will add sensors (e.g. GPS) and also send data to server (via GPRS) and to tablet (via Bluetooth). A specific wiring has been designed in order to equip vehicles. UNIMORE server in figures Figure 2-1 PS REG Architecture 1 and Figure 2-2 PS REG Architecture 2 include a first server (owned and operated by DQuid) on which raw data sent by OBUs will be initially stored and a second server operated by UNIMORE on which calculations and communications to other entities (i.e. the central server in Newcastle) will be performed.

2.3 Architecture limitations

The lack of CAN bus on the vehicles limits the number of signals that can be collected. Only analogue signals from the vehicle electrical network can be collected thus limiting collection frequency and, consequently, data transmission to the server.

The necessity of not hindering the daily activities of the Municipality employees: usage of the equipped vehicles will not be exclusive to smartCEM users, though other users will be asked to favour other vehicles, when practical. Municipality vehicles are stored and charged within the Municipality facilities and are also usually parked at the Municipality premises.

3 Components implemented

3.1 Component #1 Vehicle sharing

The employees of the Municipality of Reggio Emilia have access to an EV sharing system, which they can use for work-related activities.

3.1.1 Reggio Emilia Key Management system

Every employee entitled to use one of the Municipality EVs, is provided a magnetic badge, which can be used to unlock the keys of one of the EVs. All of the EVs keys are stored at the Municipality premises and are managed by an electronic cabinet: a system called KMS88/88 by APICE®. The KMS consists in a cupboard containing the keys, which can be accessed by using either a magnetic badge or a different personal identification method (e.g. fingerprints, PIN, etc.). When a user is identified, the system releases a set of keys, based on the user’s privileges, from which he or she can choose one to take.

Figure 3-1 Key cabinet

Since within smartCEM project ten vehicles will be used to test the services, a current hypothesis is that of reviewing users’ privileges concerning key access, so that smartCEM users may be directed towards the use of smartCEM vehicles. When a user takes a key, the system logs information such as: user ID, EV ID, time of usage (taking and returning of the key). This data will not be available in real time, but will be stored for post processing analysis.

3.1.2 Functionalities

The component implements the EV-sharing management service in Reggio Emilia. The service will be used as a monitoring system for the municipality vehicles, to track their usage. Since users not related to the project will still retain access privileges to the vehicles, smartCEM users will be asked to register to an additional service when using a equipped smartCEM vehicle.

3.1.3 Process

The OBU will send data from the vehicle to a first server which will initially store raw data and transfer them to a second server operated by UNIMORE. Data from key management system will be picked-up and transferred to UNIMORE servers manually with file transfer procedures. It will be done in post-processing.

3.2 Component #2 – smartCEM Mobile Portal

3.2.1 Functionalities

The application gives the users the opportunity to access applications and websites that provide electro-mobility IT services. The smartCEM Mobile Portal is therefore a gateway to the other EV services implemented within the project. It is delivered as an Android Application, and once installed, provides information on the EV-services available and offers the possibility to install and launch other applications on the Android platform. Managing of login functionality is under evaluation by UNIMORE, TeamNet and CRF: during the experimental phase users should be to log through app. A required login for users should represent a barrier for usage of smartCEM services, thus a post processing matching between users and vehicles according to data coming from key management cabinet is under evaluation. Post processing will merge data of vehicle usage in order to check and identify drivers in first project phase (i.e. basline).

The SmartCEM Mobile Portal is currently available at:

http://smartcem.teamnet.ro/SmartWebManager/Apk

It will become available on Google Play by the end of the baseline period. There are currently no plans to develop an iOS version of the Portal.

3.2.2 Process

The smartCEM Mobile Portal provides access to other applications and websites. For this, the following metadata is used:

- The description of smartCEM implementation sites. This includes basic data and the coordinates of the geographical area of the site

- The description of the available smartCEM services. A mapping of service-to-site is specified. Several service categories are defined. A service can be part of more than one category

- The description of the available smartCEM applications and their interfaces. A mapping of service-to-application is specified

All installations of smartCEM Mobile portal connect to a central database in order to retrieve the current version of the metadata. A separate web application, the smartCEM Management Console is used to manage the metadata used by all Mobile Portal installations. Once installed on the user's smartphone/tablet, the smartCEM Mobile Portal connects to the central database, downloads the metadata and presents the user with the UI that allows him to:

- Browse available sites and their information.

- Select the current site (the current site can be computed automatically, if the mobile device that runs the portal has GPS capabilities).

- Browse available services.

- Select any service and launch its corresponding application/website.

3.2.3 Issues

No issues emerged regarding this component. Being aimed at easy installation and use, and in order to avoid deployment barriers, the smartCEM Mobile Portal does not require any registration of the user. If any smartCEM Service is supplied by an application launched by the Portal, it must provide its own registration, authentication and authorization mechanisms.

3.2.3.1 *Applicability*

The installation of the smartCEM Mobile portal (as well as the auxiliary management web console) has no site-specific implementation details pertaining to the Reggio Emilia pilot site. All the specifics are being handled by customization of the site metadata, within the portal's central database. If provided, the user sign-on functionality must be encapsulated in a separate Android application.

3.3 Component #3 – Navigator

3.3.1 Functionalities

The EV-Navigator is a common component in the smartCEM architecture, designed to run standalone, connected to the vehicle or connected to other applications: at the Reggio Emilia pilot site, the EV-Navigator will be installed on an Android tablet installed onto the Municipality's EVs. The navigator component implements the EV-navigation service; the Platform will send trip details (origin / destination) to the Navigation for setting O-D automatically. Users will be asked to use it as much as possible when travelling through the town.

The EV-Navigator provides classical Navigation functionality such as:

- Maps display and route visualization.
- Input of destinations.
- Turn-by-turn route guidance.
- Various setting-possibilities.
- Programming interface.

These are combined with additional functionalities to specifically support the eMobility-user, such as:

- Display reachable range as a polygon, taking into account many vehicle and situation-dependent parameters (see section 3.3.2c for additional details).
- EV-Assisant: display current status of electric consumption and other current values.
- EV-Assisant-Simulator: lets the user simulate potential situations (e.g. maximum range in case the battery was only half full).
- EV-Charging point Assistant.
- EV-ecodriving routes.
- EV-ecodriving hints.
- EV-Key-Value store interface.

Drivers can use the Navigator as integrated in a tablet, which would be installed in the car or given to car users, but there is also the possibility to use the application on any android smartphone, so a multitude of screen resolutions and devices can be supported.

Since the user will often switch between the EV-efficient driving CRF App and the Navigator app, the Navigator application contains a direct Link to the CRF app on the main Navigation / Map Screen to switch easily between both apps.

3.3.2 Process

3.3.2.1 Learning process

As a base for most EV specific functionality, a vehicle profile with specific EV-relevant values is needed. This vehicle profile is derived from learning processes. A rough estimation profile will initially be used and it will later be refined by learning processes (based on log files and frame parameter calculations), as well as field test results.

3.3.2.2 Typical usage

Before starting the navigation, the Navigator lets the user enter his own

destinations: there are various options such as addresses, zip codes, last destinations, favourites, home and office, POI or selection positions from the map. During the navigation, the user gets turn-by-turn navigation hints, route and manoeuvres are shown visually and automatic zooming always provides a situation dependent optimal view of the situation. Sign posts and junction views provide additional orientation.
The Navigator runs with 20 languages and 16 Text-to-Speech languages are provided; it can also be customized in many aspects, like specific vehicle attributes, map designs, map types, international units. Logistics vehicles are strongly supported by taking into account their different routing needs (like avoidance of U-turns, avoiding residential areas, accessing restricted areas, avoiding smaller streets or avoiding toll roads as well as restrictions (tolls, specific measures, dangerous goods, type optimized routes).

Interactive assistants support in various situations like a speed assistant or a danger warning assistant. If the Navigator should be integrated into processes (like automatically entering targets and start navigation) an extensive API can be used.

3.3.2.3 EV-use cases

Additionally, the user of an electric vehicle has various options beyond that. If he starts driving (with or without navigation) the EV-Assistant shows how much energy he is consuming currently, both due to the engine and to the other consumers in the car. These values come preferably from current fed values, entered by the EV-efficient driving CRF-APP via the programming interface. If no current data is available, the Navigator automatically uses a simulation based on default vehicle/driver values to calculate an EV scenario, which is needed for range calculations.
With a simulation mode, also the user can define values like Battery usage (SoC), usage of climate control (AC/heating) and temperature to work on a “what if” status. If there are no current values fed from the car, the system can work in a completely simulation-based mode, using user input or predefined values: thus, the system can also be used on any smartphone outside of a car.

The reachable range is as well important: it is calculated on an external server and transferred fluently into the navigator, showing exactly how far the EV car could go by means of a polygon defined range. If the user realizes that he may have difficulties to reach his destination, he can switch to an eco-mode that calculates a very energy-efficient route and shows hints (like given speeds on different roads) to the user. Included in the range calculations are a lot of vehicle/driver specific parameters like:

- Road type dependent energy consumption.
- Road type dependent driven speeds.
- Battery degrading over time or distance.
- Battery capacity and usable energy portion.
- Current SOC.

- Temperature influence on battery.
- Heating/cooling influences.
- Current consumers.
- Battery temperature.
- Outside temperature.
- Height/slope influence factors on energy consumption.
- Vehicle weight and additional loads/people.
- Current Traffic situation.
- Vehicle restrictions like truck restrictions or toll preferences.

A charging point assistant helps to find charging points along the routes by showing them on the map and by easily integrating them into the route without losing the original target.

3.3.3 Issues

The vehicle-specific learning process may consume a lot of effort, if there are not many vehicles where it can be used.

The same occurs with feeding online data to the Navigator: this may not be applicable if there are not many vehicles or if there are legal restrictions, so the Navigator has to be usable without real time data acquired from the car. Then the required data can be entered manually by the user.

If data are available, Navigator app can integrate all the functionalities. Advanced functionalities as slope dependant ranges, eco-driving, eco-hints and real time traffic will be included during the test execution as more.

Possible issues may arise from the update of the firmware (to be determined).

3.3.3.1 *Lessons learnt*

A lean learning process for EV-vehicles is highly appreciated, as a too complex learning process may not be possible.

3.3.3.2 *Applicability*

Not all the Navigator’s features will be used, especially detailed learning processes and interface connections for feeding real time data or entering destinations automatically, but these possibilities are offered in order to open perspectives if the system would be expanded.

3.4 Component #4 – Efficient driving analyser component

3.4.1 Functionalities

The app (called CRF-App hereafter) will operate in conjunction with PTV navigator, implementing POIs and specific indication to achieve a better efficiency. CRF-App is a complex app that absolves to the following tasks:

- Acquisition, via Bluetooth channel, of vehicle data by means of a dedicated HW developed by DQuid (http://www.dquid.com) and called "Bluedash". This HW is directly interfaced to the vehicle itself.
- Data collected is stored locally into the device hosting the application. Afterwards, the same data can be uploaded automatically (only in the case that 3G connection is available) or manually via USB from the device to the UNIMORE server.
- Transfer relevant and statistically arranged information to the navigation system (i.e. Current SOC, speed, vehicle state, battery voltage and current etc.).
- EV-Assistant: displays current status of electric consumption and other current values.
- EV-ecodriving hints: displays the correspondent values acquired by the EV-Navigator.
- EV-Efficient driving: it provides real-time on trip advice on how to improve driving style to attain a better efficiency. The warning messages are obtained by integrating the EV-ecodriving current data with the current vehicle signals (speed, SoC) etc.

3.4.2 Process

3.4.2.1 Typical usage

The CRF-App is started like any other Android app. At the beginning, after some seconds while the splash screen is visible, the login page appears. The user is requested to select his/her own id then to confirm.

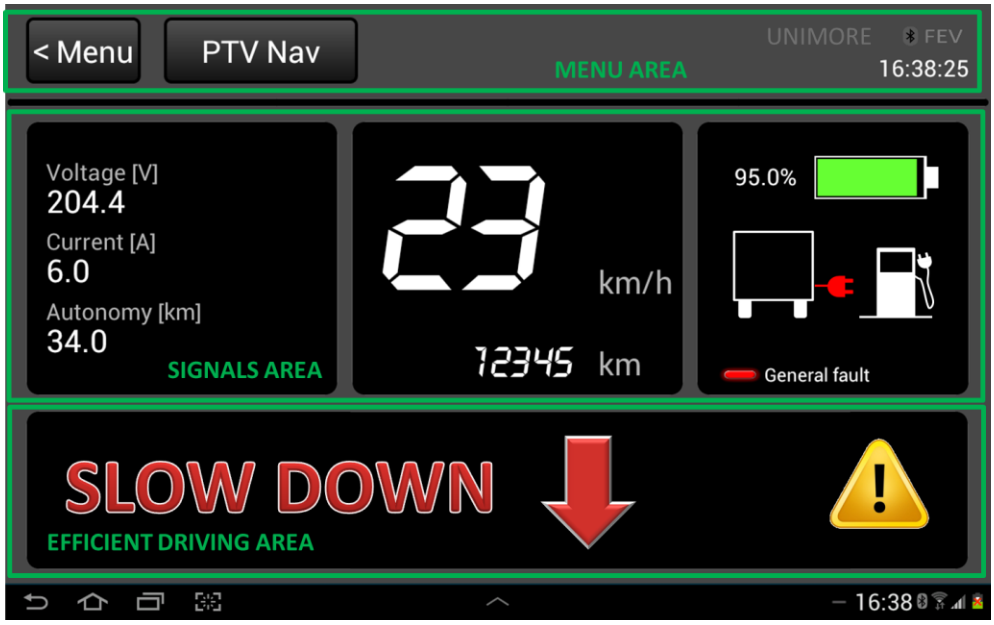
After that the Main page is shown on the screen (see figure below). Some info can be seen located in different parts of the screen. In details:

1. Menu area: on the left some contextual buttons are available, usually the Menu back button is present and causes the visualization of the main menu, The PTV Nav button activates the PTV navigator also causing the CRF-App to continue running in the background. On the right the state of the connection with the vehicle (FEV) via Bluetooth and with the UNIMORE server via 3G are visible.

2. Signals area: on the left both the voltage and current acquired by the FEV and the autonomy acquired by the PTV Navigator are visible and refreshed as soon as a new value is acquired. In the centre, the speed and (if available) the total covered distance (kms) is displayed in real-time as soon as acquired by the DQuid Bluedash device via Bluetooth. On the right, the current SOC, the charging state and the general fault alarm are available.

3. Efficient driving area: the whole area is reserved for the visualization of both graphic and textual warning message with the aim to bring to the driver's attention.

The current version of the HMI is still under evaluation by CRF experts with the aim to improve and, especially after the baseline tests, to meet users' needs by inserting new info.



Voltage [V]
204.4
Current [A]
6.0
Autonomy [km]
34.0

23 km/h
12345 km

95.0%

General fault

SLOW DOWN

Figure 3-2 CRF-App main page

3.4.2.2 *Admin usage*

People with admin rights are allowed to upload the data acquired during the day trip. This offline functionality is assumed to be available as default solution for the transfer of the whole history of the vehicle signals to the UNIMORE server. The real-time transfer (during the trip) of such big amounts of data is still under evaluation.

Besides, it is still in progress the evaluation and development of some admin pages for the tuning and integration of the info required for the efficient driving. The objective is to make as easiest as possible the modification of some parameters (i.e. maximum acceleration related with a determined SoC allowed for an efficient driving style) without the direct intervention of a SW expert. The access to these pages should be granted to admin right people.

3.4.2.3 *EV use-cases*

The typical usage described above represents also the most common use case. The driver brings a tablet at the beginning of the day, start the CRF-App which in turn starts the PTV Navigator, selects the target(s) and then starts driving. During the trip the driver can switch between the CRF-App and the PTV navigator depending on which info (s)he needs.

The second use case is the manual upload of the real-time data to the UNIMORE server as explained above. In this case the actor should be an expert having admin rights and at least a basic knowledge of Android devices.

The third and final use case is the tuning of the efficient driving parameters that should be done by project experts with the support of CRF.

3.4.3 Lessons learnt

The most important feedbacks for such an innovative HMI are expected to come from the real users therefore after the first months of the final experimentation.

3.4.4 Applicability

Some innovative features like HMI notifications obtained by the aggregation of info acquired by different sources could be taken into account both internally for the development of new standard and electric vehicles and also by the whole consortium as an important project outcome. A definitive evaluation of such applicability can be done only with all the final experimentation feedbacks available.

3.5 Component #5 – Android apps

3.5.1 Functionalities

Group of Android applications developed to provide on trip EV-Services for the end users. Their functionality is based on data coming from the servers of each of the service's development company (PTV and UNIMORE servers).

The services are provided behind a unique application that includes the access or link to each of the service's applications. The main three services are divided in two groups:

ON-TRIP SERVICES

- Navigator: it works as a conventional Navigation system, with some additional information, like CS location information or efficient driving instructions.

- Efficient driving: it provides real-time advice on how to improve driving style to attain a better efficiency.

3.5.2 Process

According to interests of smartCEM project, the selected development platform for the smartphone/tablet apps has been Android. In fact, this Operating Systems allows more flexibility in development phase than others (i.e. Apple iOS, Windows Mobile, RIM Blackberry) and is suitable to implement services and third parties applications to several devices coming from manufacturers with not comparable features such as screen sizes. Moreover Android devices are usually more affordable than others.

3.6 Component #6 – CS Manager

3.6.1 Functionalities

This component implements the EV-Charging Station management service.

3.6.2 Process

The smartCEM common component interacts with the EV-navigation, in order to provide charging station details (location, IDs, name).

CS data are made available in CSV format by the local charging station provider. The EV-Navigator takes these data into account to plan the most suitable route for the EV-vehicle.

3.6.3 Issues

None.

3.6.3.1 *Lessons learnt*

None.

3.6.3.2 *Applicability*

The installation of the smartCEM CS Manager mobile application has no site-specific implementation details. The integration solution employed between CS Manager App and Navigator app is unique, used in the same configuration for all the sites.

4 Interfaces and Standards implemented

4.1 Interface #1 Android App and BlueDash™

4.1.1 Functionalities

BlueDash™ is an electronic component[1] that can be easily installed on an EV and allows gathering data from the vehicles: this is achieved by establishing a Bluetooth connection between server, on-board tablet and the BlueDash™ unit. The BlueDash™ unit is able to read vehicle data via CAN bus and is equipped with GPS for position data and GPRS for data communications with the DQuid server and, consequently, the UNIMORE server. The BlueDash unit that will be used in Reggio Emilia pilot site is type called "DQuid Xee".

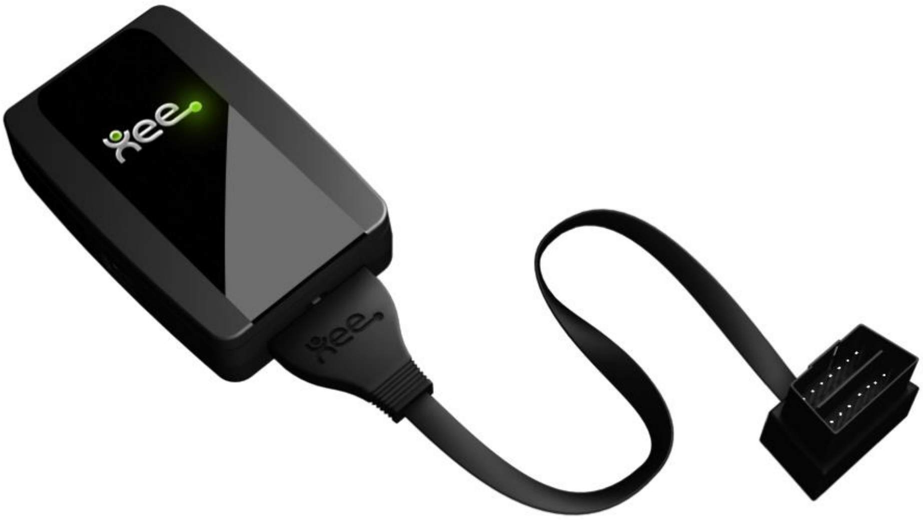


Figure 4-1 BlueDash "DQuid Xee" on-board unit for Reggio Emilia pilot site

4.1.2 Choice of standard

Standard used are: Bluetooth, GPRS, GPS, CAN bus.

4.1.3 Process

The BlueDash™ unit will send the collected data from the vehicle to the DQuid and UNIMORE servers for elaboration. Results will then be fed into the smartCEM application installed on the on-board device. Since there is no direct interfacing of

[1] www.dquid.com

BlueDash™ and the application, it is of critical importance that the data collected from the vehicle and later processed retain maximum consistency.
The user will interface directly with the applications running on the Android-based device mounted on board and will have no contact whatsoever with the Bluedash™ unit.

4.1.4 Issues

As explained above, data consistency should be retained throughout the chain vehicle- BlueDash™-servers-apps.

4.1.4.1 Applicability

This interface is not site-specific in any way, so it can be as well implemented in other pilot sites (i.e. Newcastle).

4.2 Interface #2 Vehicles and data logging server

4.2.1 Functionalities

Communication between both entities in order to log data from vehicles.

4.2.2 Choice of standard

A GPRS connection has been chosen since it is already present within the BlueDash™ unit.

4.2.3 Process

An A-D converter unit, called signal machine, will be connected to the vehicle electrical network (i.e. CAN bus) from which it will extract data on the usage of the vehicle. Data collected by the signal machine will be then sent to the DQuid Bluedash™ unit which will combine it with the GPS and other sensors embedded in it and send it via GPRS to the DQuid server for the initial storage. Type of this signal-machine and manufacturer are under evaluation.

Data will then be sent to UNIMORE server for calculations, storage and access by CRF and UNIMORE. The server will then send the refined data to the applications installed on the Android device installed on vehicle. Update of data will be daily sent by UNIMORE server to the Central Database in Newcastle.

Collection and transmission frequency was set to 0.5Hz (every 2 seconds). The whole chain can be seen in figure.

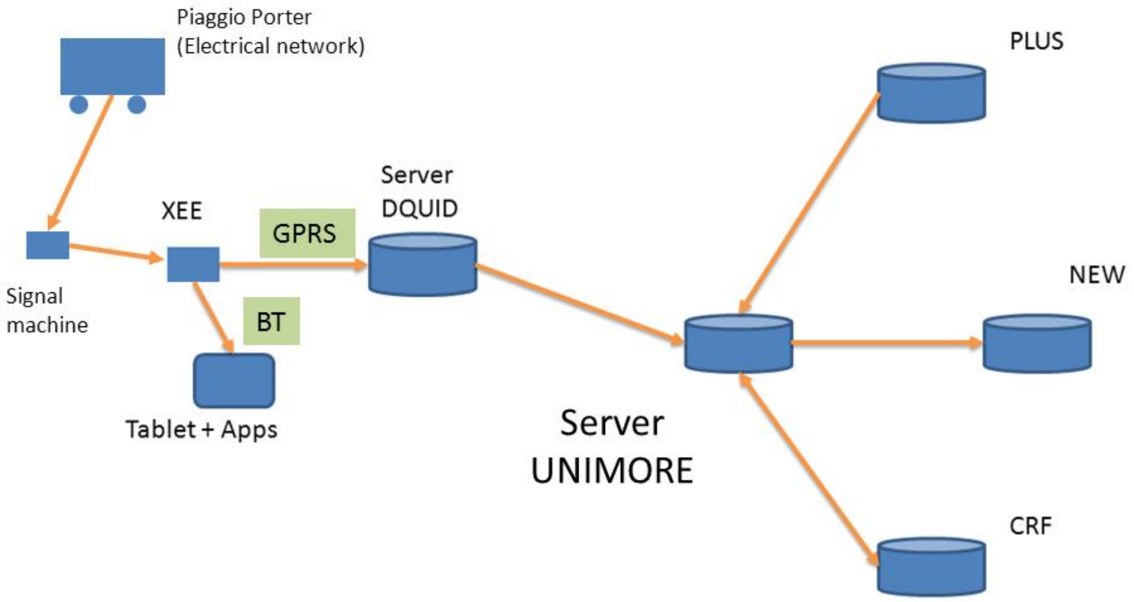


Figure 4-2 Communication structure in Reggio Emilia PS

4.2.4 Issues

Connection vehicle-OBU is rendered difficult by absence of CAN bus connector.

4.2.4.1 Lessons learnt

A-D converter units (signal machines) can be a solution when vehicles have no CAN Bus.

4.3 Interface #4 Data extraction for EV Efficient Driving and WP4 indicators

4.3.1 Functionalities

This concerns traffic data elaboration for Efficient Driving and communication to the PTV navigator installed on the in-vehicle tablet.

The tablet will also make use of some parameters deduced from these data sets to provide each driver with a functionality to support him in driving in the most way .

4.3.2 Process

From the data logged on the UNIMORE server a set of synthetic parameters that allows the tablet (installed on the vehicles) to transmit corrected SOC values to the PTV navigator application will be extracted.

The supporting software is planned to comprise two parts, one working at runtime, another that can be recalled by the user at the end of the day to understand where are the possible margins to increase the efficiency (speed behaviour, acceleration and deceleration modalities).

4.4 Interface #5: Android App and EV Services

4.4.1 Functionalities

SmartCEM Mobile Portal has a very light interface towards the applications that provide EV services. Available functionalities are:

- Install an Android application (if the .apk indicated by the portal metadata is not yet installed on the user’s mobile device).
- Launch an Android application (if already installed).
- Launch an internet browser and point it to the website indicated by the portal metadata.

There are no prerequisites that need to be fulfilled by the applications that provide EV services. They need not be “portal aware” and may be launched independently.

Applications will be installed on a 7 inches tablet. Tablet type will be selected according to requirements:

- Android Operating System.
- 7 inches and high resolution, capacitive touchscreen display.
- Lightweight to be handled.
- GPRS/3G connectivity;
- Bluetooth connectivity;
- Affordable price for according to budget,

Currently the device-candidate for equipping EVs is ASUS Fonepad 7", model K004 (ME371MG)

Figure 4-3 Tablet 7" to be used in Reggio Emilia PS

4.4.2 Choice of standard

A Bluetooth connection will be established between the Android device and the Bluedash™ unit.

4.4.3 Process

Three different applications will be provided: EV Navigator, EV Efficient Driving and EV-Sharing services.

- **EV Navigator:** It provides an on trip service and if the app needs to communicate with the server, then 3G communication may be used
- **EV Efficient Driving:** User profiles for this app will be established during baseline and logged in the Android device. When using the service, the users will log into the system and the correct profile will be used by the service
- **EV-Sharing:** The communication for the sharing service, from the Bluedash™ unit to the servers, will be established through a GPRS connection

4.5 Interface #6 CS Manager and Navigator

4.5.1 Functionalities

The CS Manager Service maintains a CS database. Each CS entity has a set of attributes, out of which the following are relevant to the Navigator:

Attribute name	Description
Name	The name of the POI (Point of Interest) that is represented by the CS
Street	The street name of the address of the CS
HouseNo	The house number of the address of the CS
PostCode	The postal code of the address of the CS
Town	The town name of the address of the CS
Latitude	The latitude coordinate for the CS
Longitude	The longitude coordinate for the CS
PhoneNo	The phone number associated to the CS (it is usually the phone number of the CS operator)
FaxNo	The fax number associated to the CS – usually the fax number of the CS operator

Table 4-1 Attributes related to the Navigator

These attributes are being made available to the Navigator via a REST web-service, in two formats:

- csv text
- proprietary binary format

The csv text is available at:

<server_url>/CSManagerService/cs?scope=atf&name=<cs_name>

The binary stream is available at:

<server_url>/CSManagerService/cs?scope=dat

4.5.2 Choice of standard

Both outputs obey the format specifications as requested by the Navigator. They are made available online at the URL-s indicated above, in HTTP REST services

(GET requests).

- The csv format is “text/plain” content
- The binary format is “application/octet-stream”

The current conversion between the internal database representation of the CS and the binary format is being performed by a specialised tool supplied by Navigator provider. Later modifications and extensions of the attributes set is therefore possible with minimal effort.

4.5.3 Process

Whenever the Navigator is making a service request for the binary form of the CS list, the CS Manager Service is performing the extraction of the needed attributes and the generation of the corresponding binary stream, by calling the dedicated converter tool supplied by the creators of the Navigator. An exact and up-to-date binary representation of CS data is therefore available at any moment to the Navigator

5 Conclusion

This document has described the local platform integration and processes in place at Reggio Emilia pilot sites for integrating and deploying smartCEM services. The Reggio Emilia PS has undergone several changes from the initial proposal due to different issues that arose during the planning and development process. The Municipality of Reggio Emilia is not part of the smartCEM consortium, so any change must respect the current operating services.

smartCEM driving services (EV-Navigation, developed by PTV and EV-Efficient Driving, developed by CRF) will be installed onto an Android-based device mounted in-vehicle. A signal machine coupled with a Bluedash™ unit (called also "DQuid Xee") will enable the communication between the vehicle and the servers (DQuid and UNIMORE) which will then feed the necessary information to the in-vehicle device.

The main issues encountered in Reggio Emilia test site are:

- Lack of CAN bus on the vehicles that are going to be used for the project. This initial problem will be tackled by the proposed coupling of the signal machine with the Bluedash™ unit.
- Necessity for the users to be registered when using a vehicle. Since neither smartCEM mobile portal, nor any of the services to be deployed log the users, a system to provide user registration will be developed within the Reggio Emilia test site.

According to above issues, technical solutions were evaluated and finally issues were solved allowing full data transfer with real time data collection and info visualization for end-users services (EV-Efficient Driving and EV-Navigation).

6 References

[1] smartCEM consortium, D2.1 Reference Architecture, 2013

[2] smartCEM consortium, D2.2 Platform Architecture, 2013

[3] smartCEM consortium, D2.3 Guidelines and requirements for the integration of local systems into the smartCEM architecture, 2013

[4] smartCEM consortium, D3.2 Common data exchange protocol for smartCEM, 2013

[5] smartCEM consortium, D4.3 smartCEM experimental design, 2013

[6] smartCEM consortium, Description of Work, 2013

List of Figures

List of Tables