LIFE+10 ENV/IT/000389

INTEGREEN

Action 4: Implementation & Integration

P.4.1.4

Operators centre web interface prototype

Project Coordinating Beneficiary	Municipality of Bolzano
Project Associated Beneficiary n.2	TIS innovation park (TIS)
Project Associated Beneficiary n.3	Austrian Institute of Technology (AIT)

Document history

Date	Document Author(s)	Document Contribution
31/03/14	Roberto Cavaliere, Paolo Valleri, Stefano Seppi (TIS), Reinhard Kloibhofer, Wolfgang Ponweiser (AIT)	First document version submitted to the EC
31/05/15	Roberto Cavaliere, Paolo Valleri, Patrick Bertolla (TIS)	Second and final document version submitted to the EC. External contributions have been received by the external assistance companies Ethical Software and Madeincima.

Dissemination level: CO[1]
Delivery month: M45
Status: submitted to the EC

[1] **PU** = Public; **CO** = Confidential (for members of the consortium only, including the European Commission); **RE** = Restricted, i.e. only a selected target of stakeholders defined by project consortium and validated by European Commission can access to the document. The final dissemination level will be defined in cooperation between the INTEGREEN consortium and the European Commission.

Table of Contents

Table of Figures

1 Introduction

1.1 Purpose of the document

The purpose of this document is to present the first version of the operators centre web interface prototype, by considering the high-level architecture of the overall INTEGREEN Supervisor Centre, defined in [1] (Figure 1).

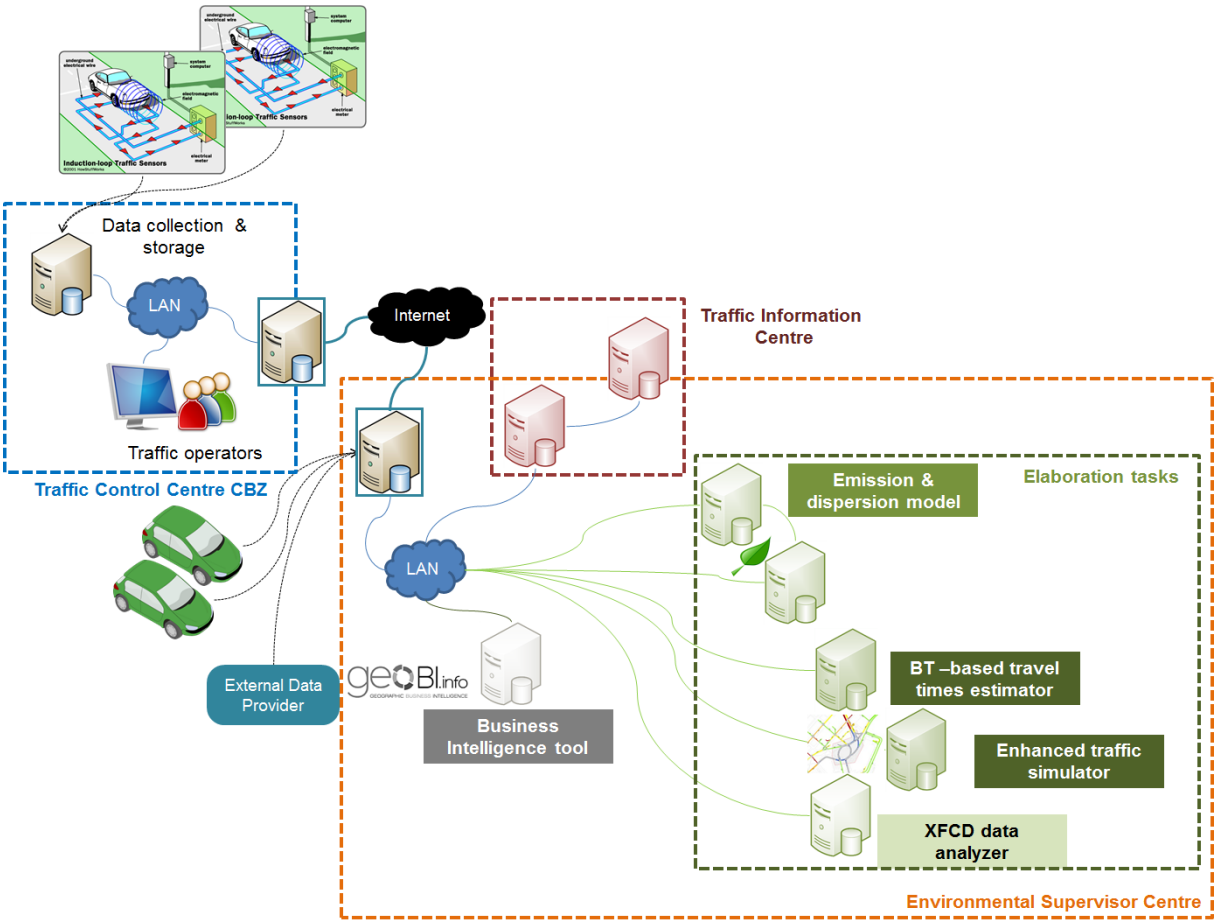

Figure 1: The high-level design architecture of INTEGREEN.

1.2 Document structure

The document is structured in one single chapter presenting not only the main functionalities of the user interface implementation but also the set of APIs made at disposal at the front-end layer in order to have a simplified and standardized access to the measured and elaborated data managed by the Supervisor Centre.

2 Prototype description

The operators web interface is not an internal component of the Supervisor Centre, as originally planned, but an external web application that the traffic operators can access remotely directly at the Traffic Control Centre. This choice has been a direct consequence of the high-level integration design choice carried out in Action n.3. Moreover, this application, similarly to all other ones that will be destined to external third parties (i.e. other service providers), has to access the measurements and the elaborations of the Supervisor Centre through the standardized front-end layer and not through a direct connection to the database (Figure 2).

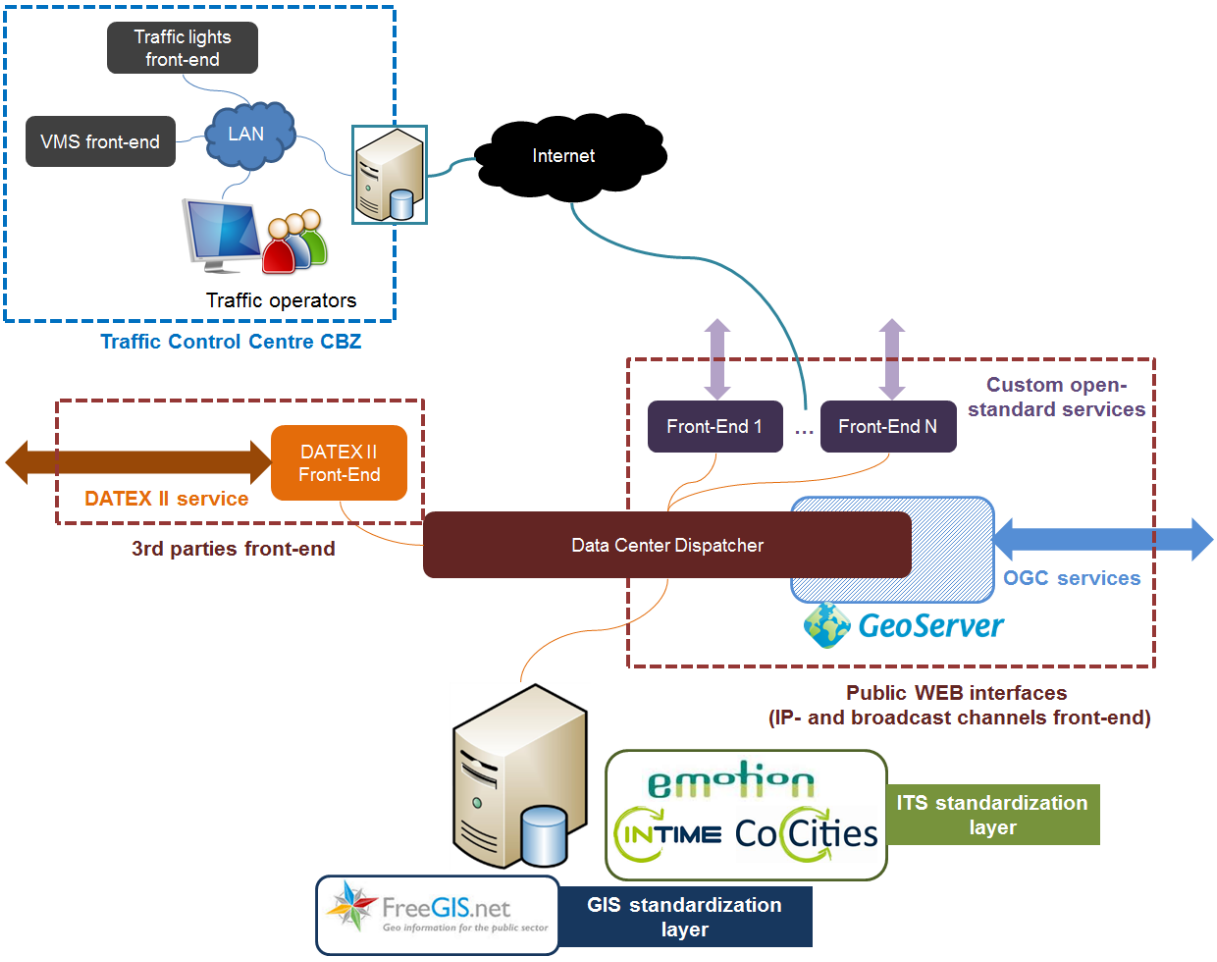

Figure 2: The interaction between the operators centre web interface and the Supervisor Centre.

2.1 Operators web interface APIs

The web application destined to the operators is based on a set of custom open-standard services, that are presented in detail in this paragraph. All relevant front-ends have been created with a scaffolding tool for **spring framework** called **spring roo**, an open source software tool that uses convention-over-configuration principles to provide rapid application

development of Java-based enterprise software [2]- [3].

Each front-end implements the **XML-RPC client interface** which handles the connection with the data center dispatcher. Three generic methods are implemented:

- **getStations**: this method returns all information that a front-end owns about each specific monitoring station; when required it returns a list containing several data such as the name of the station or the position expressed by the two parameters called latitude and longitude.

- **getDataTypes**: returns a list of data type information; each data type has a name, the measurement unit and the description. For example the meteo front-end returns for the station 8320 the following JSON string:

  ```
  [["ND","s",null],["LT","°C","air
  temperature"],["N","mm","precipitation"],["GS","W/m2","global
  radiation"],["SD","s","global radiation"],["WG","m/s","wind
  velocity"],["WG.BOE","m/s","wind velocity
  squall"],["LF","%","relative humidity of the
  air"],["LD","hPa","barometric pressure"],["WR","°","wind
  direction"]]
  ```

- **getRecords**(Object... objects): returns data by passing parameter specific to each front-end. The default parameters are:

  - "*station*": a string containing the identifier of the station;
  - "*name*": the data type represented as string;
  - "*seconds*": which is used to define the range of the historic data.

  From a more technical point of view the method returns a list of values regarding the data type requested that has the given unit and that they have been gathered by the selected station. The output is composed of a list of pair, and each pair contains a value and the its relative timestamp. For interoperability reasons the timestamp is expressed as a UNIX time which is defined as the number of seconds that have elapsed since 00:00:00 (UTC), Thursday, January 1st 1970.

Any data provided by the API is correlated to its specific timestamp which defines the time when the measurement data has been physically sampled by the monitoring instrument. An important and fundamental key in the visualization and correlation analysis of different data series is related to the **possibility to navigate the series along the time axis without any constraint**. Given that, the API has been developed in order to provide data not only about a fixed time frame (i.e., the last 24 [h]) but also to provide all the data that belongs to a given time frame requested by the operator. In other words, the API has been designed to allow an operator to visualize data series regarding any time period. In this view a period is defined by two different dates passed as input parameters to the API presented above.

All methods presented above receive as input two additional parameters which specify respectively the starting and the ending date. Due to this feature, there is the possibility to use the API in order to return data within any given time frame. While the starting date is a mandatory parameter, the ending date parameter is optional: in this case the method returns all available data available up to the last one stored.

All front-ends are entities providing the data collected by their correspondent data sources. They provide a **REST** interface which delivers data in JSON format to the client. The three generic methods have been implemented and retrieve the data by requesting it from the dispatcher. If the connection to the dispatcher fails, the front end returns a HTTP status SERVICE_UNAVAILABLE to the client. The reference end-points, are respectively:

Environmental Data: http://ipchannels.integreen-life.bz.it/EnvironmentFrontEnd

Bluetooth Detections Data (O/D): http://ipchannels.integreen-life.bz.it/BluetoothFrontEnd/

Travel Times Data: http://ipchannels.integreen-life.bz.it/LinkFrontEnd/

Meteorological Data: http://ipchannels.integreen-life.bz.it/MeteoFrontEnd

Mobile Data: http://ipchannels.integreen-life.bz.it/VehicleFrontEnd

The main starting page of this API is http://ipchannels.integreen-life.bz.it/doc/, where interested third parties can find detailed documentation and examples in how properly accessing all the data put at disposal by the Supervisor Centre.

2.2 Operators web interface prototype: the “BZAnalytics” application

The Operators centre web interface prototype is meant to be an accessible web interface to allow well defined and identified operators to **control the current traffic and environmental conditions in the urban road infrastructure situation** in real time. Thanks to this interface, operators will be in the conditions to use available traffic control systems (i.e. the VMS front-end and the traffic lights front-end) to dynamically change any available parameters in order to improve the current traffic flows, in particular from an environmental point of view; for example, an operator could change the traffic light phases in correspondence of a particular intersection when specific traffic / air pollution patterns are noticed.

The operator centre web interface is a powerful **monitoring console** that is able to plot in the same chart one or more data series, as illustrated in the plenty of **plots** presented in the other prototypes deliverables. By doing that, an operator is not only able to visualize the data trend of a given time series but also to correlate two or more data series together. The advantages of comparing data-series is mainly related to the fact that particular situations of traffic jams and high pollution can be detected and described by several information gathered by different sensors, and somehow motivated on top of specific boundary conditions.

In order to have a quicker view of the real-time situation, a presentation of data and information available on a **map overlay** has been moreover added

The interface is available at the link http://analytics.mobility.bz.it

The access is protected through credentials, which are shared among project staff and other authorized parties (in particular traffic operators of the Municipality of Bolzano) only.

The development of the HTML5 application (i.e. properly visualizable on any device with Internet connection) has been carried out by TIS in cooperation with an external company expert in software development. The Graphical User Interface (GUI) of the application, including its logo, has been designed and developed by another local company MadeInCima.

Figure 3: The logo of the operators web interface "BZAnalytics".

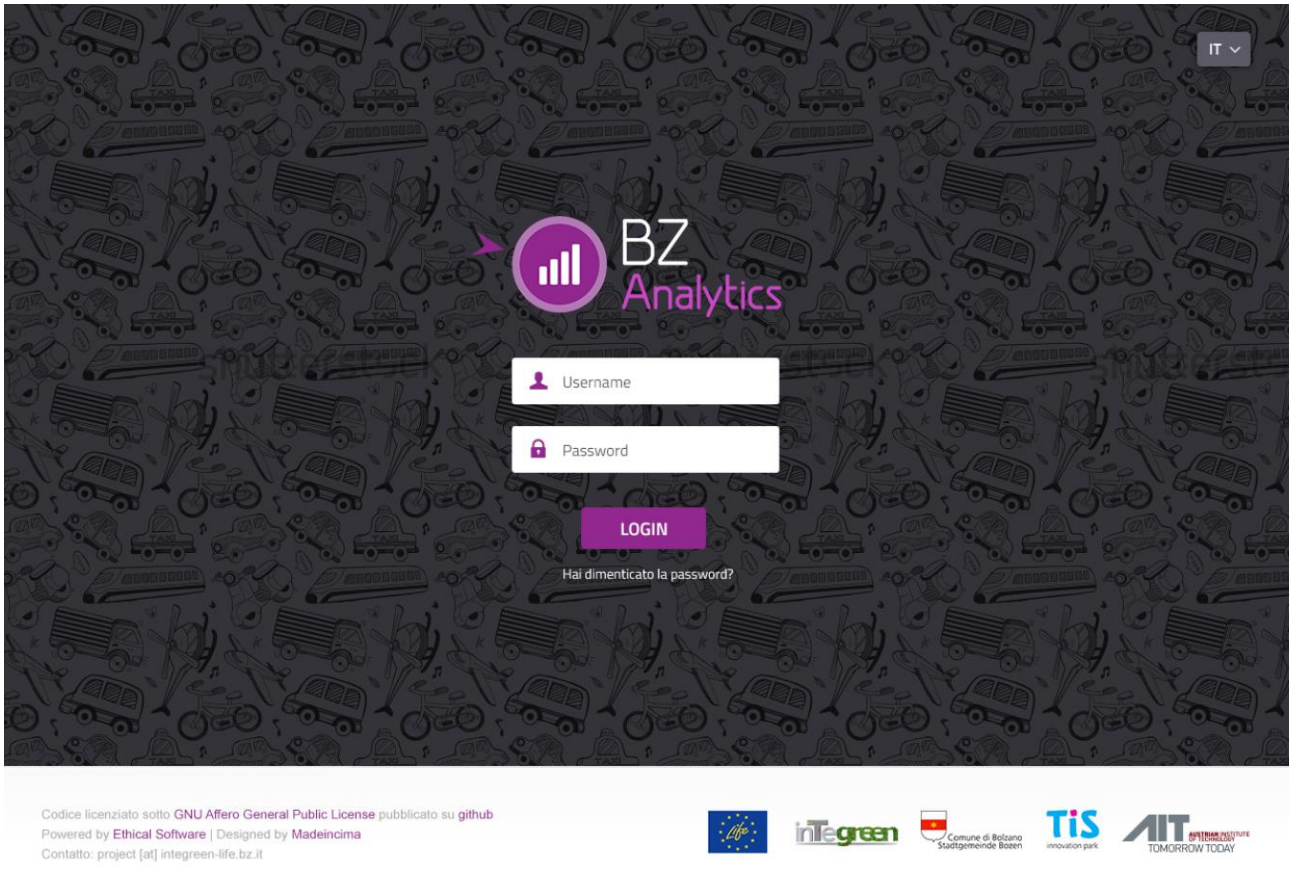

Figure 4: The home page of the operators web interface "BZAnalytics".

After the authentication, the user enters in the following situation. On the top right, he/she can choose the preferred languages (available options are Italian / German / English). Moreover, he can decide to switch between the “plot” view (the current selection) and the “map” view. In the bar just below this selection, the user can decide to:

- activate the “**live update**” option, which means that the console will automatically display the new real-time data received;
- visualize plots as **histograms or lines**;
- select the **period** he wants to visualize on the plots (preconfigured selections are available, with the possibility to define a custom range).

On the left, the user has at disposal a menu with which he/she can start to visualize the data of interest between all ones that is handled by the INTEGREEN system. The operations are the following:

1. **Selection of the station type** (in the example, Bluetooth)

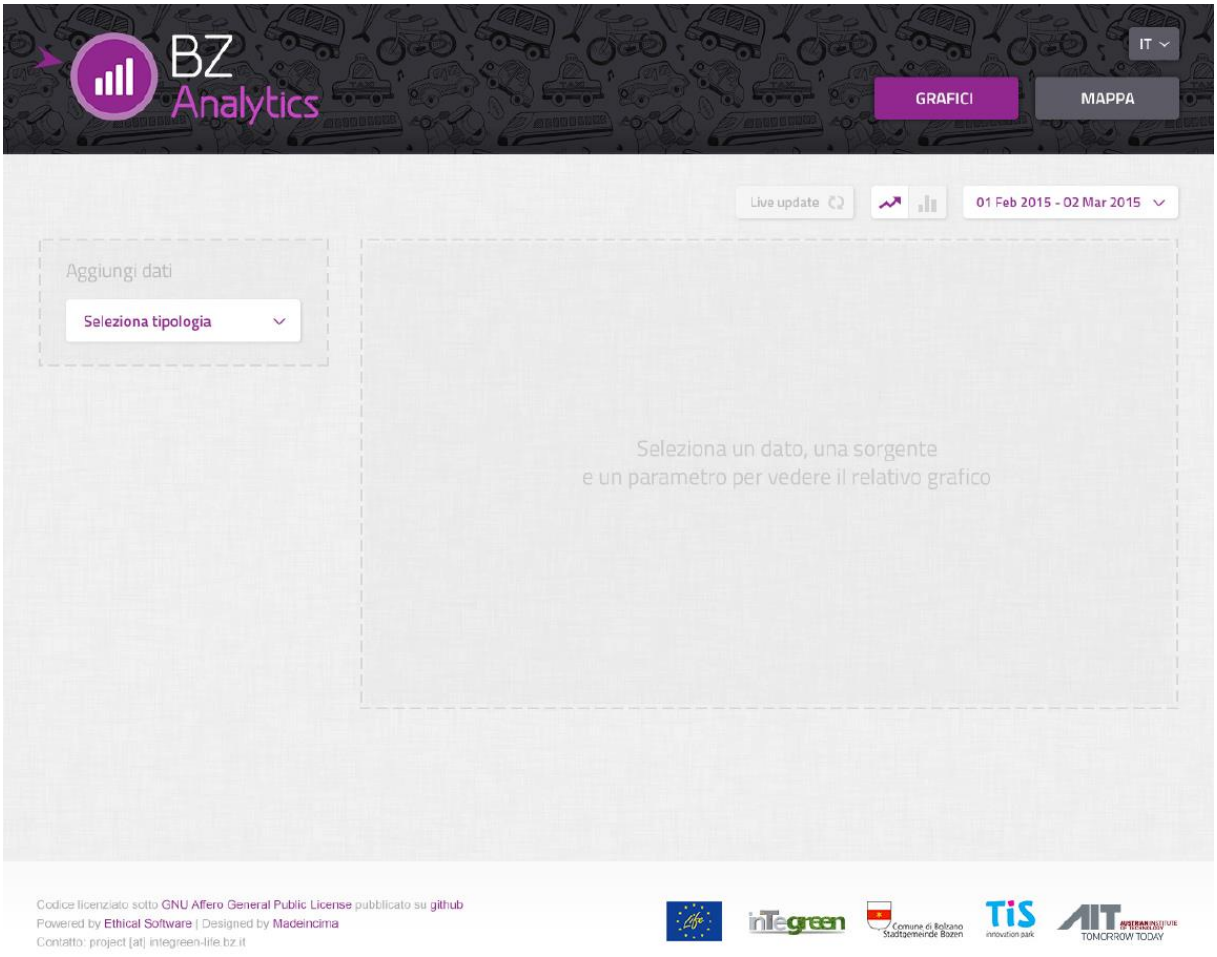

Figure 5: The main page of the operators web interface “BZAnalytics” (section plots).

2. **Selection of the station name** (in the example, Siemens Street)

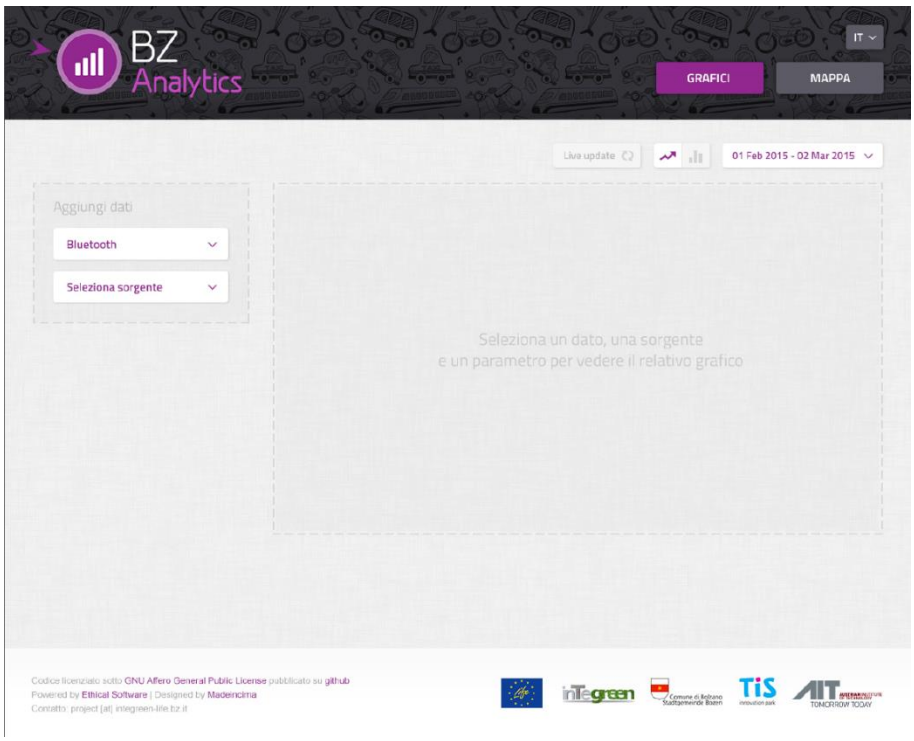

Figure 6: The selection of the station in the operators web interface "BZAnalytics" (section plots).

3. **Selection of the data type** (in the example, Bluetooth count record)

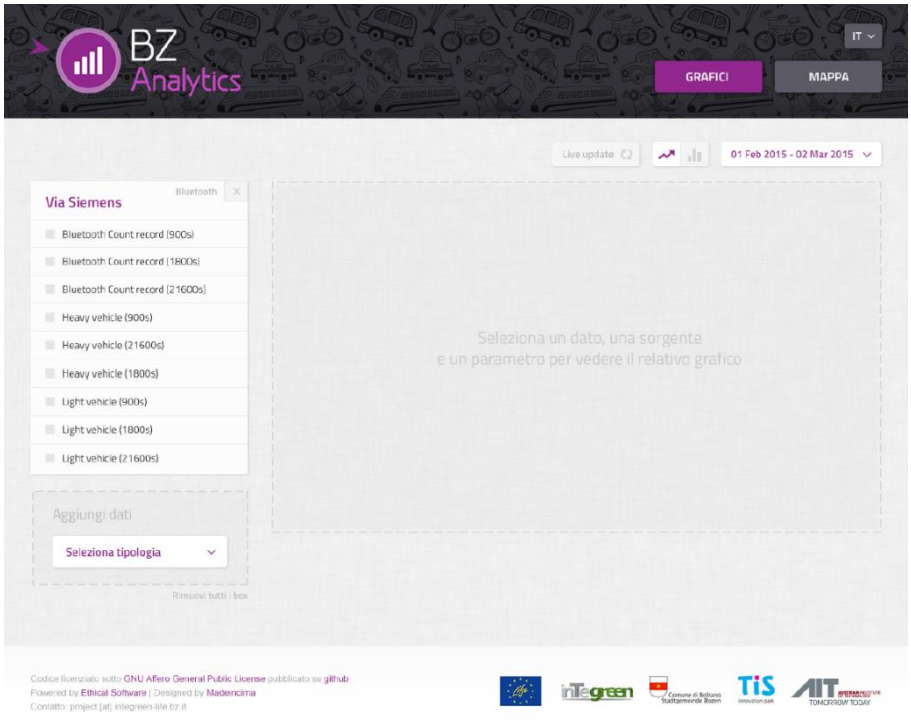

Figure 7: The selection of the data type in the operators web interface "BZAnalytics" (section plots).

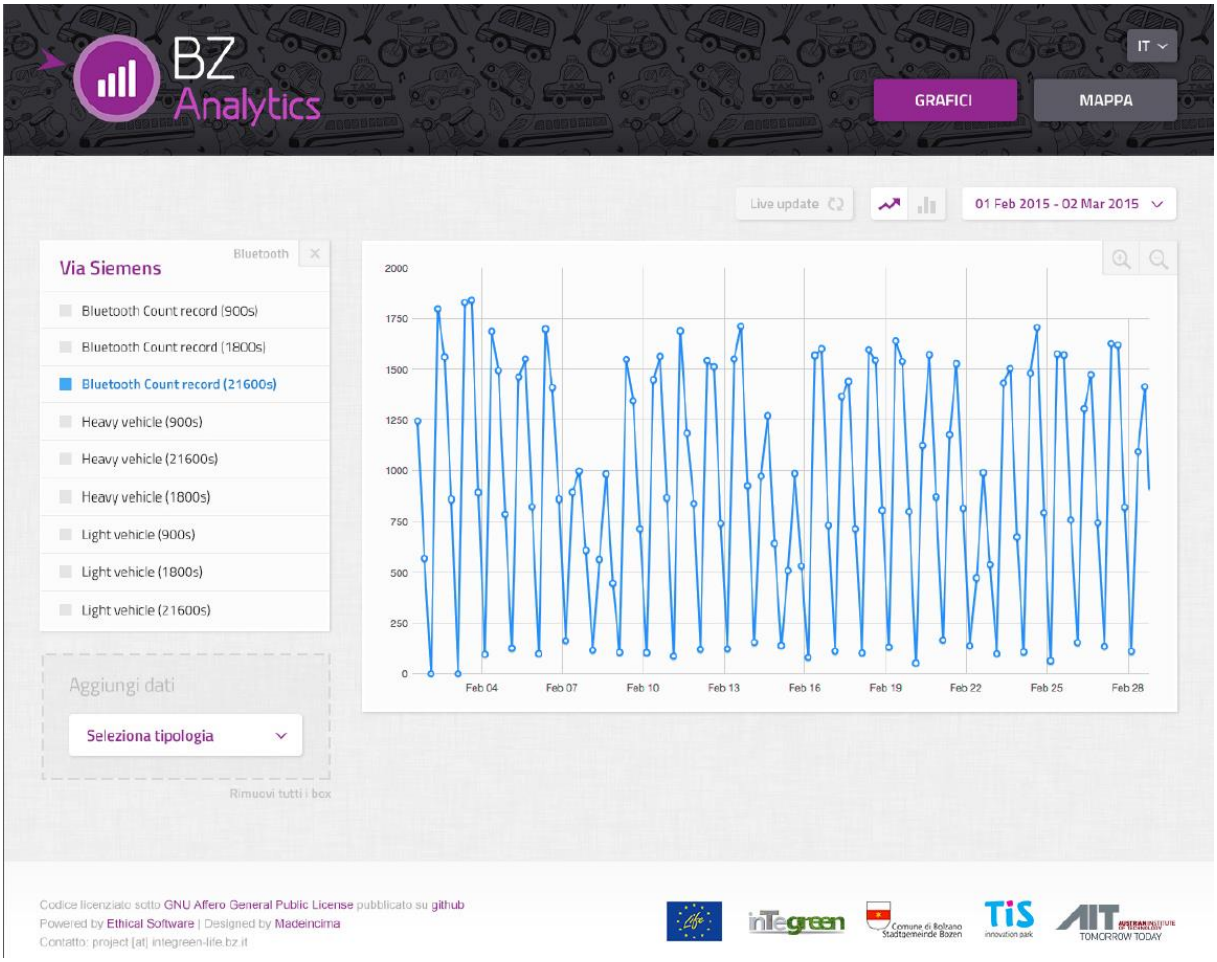

Figure 8: The visualization of time series in the operators web interface "BZAnalytics".

The user has moreover the possibility to repeat this operation and visualize several data series all together. In case of multiple data sets requested, the scale of the plot is automatically adjusted in order to optimize the presentation of the data.

By clicking on the map view, the user has at disposal on the right the empty map, and on the left the full list of data and elaboration he can visualize, which are:

- **Bluetooth stations;**
- **traffic stations;**
- **travel times;**
- **parking information;**
- **meteorological data;**
- **static air pollution;**

- **mobile air pollution data;**
- **NO_x / PM_{10} / CO_2 emissions;**
- **dispersion map**.

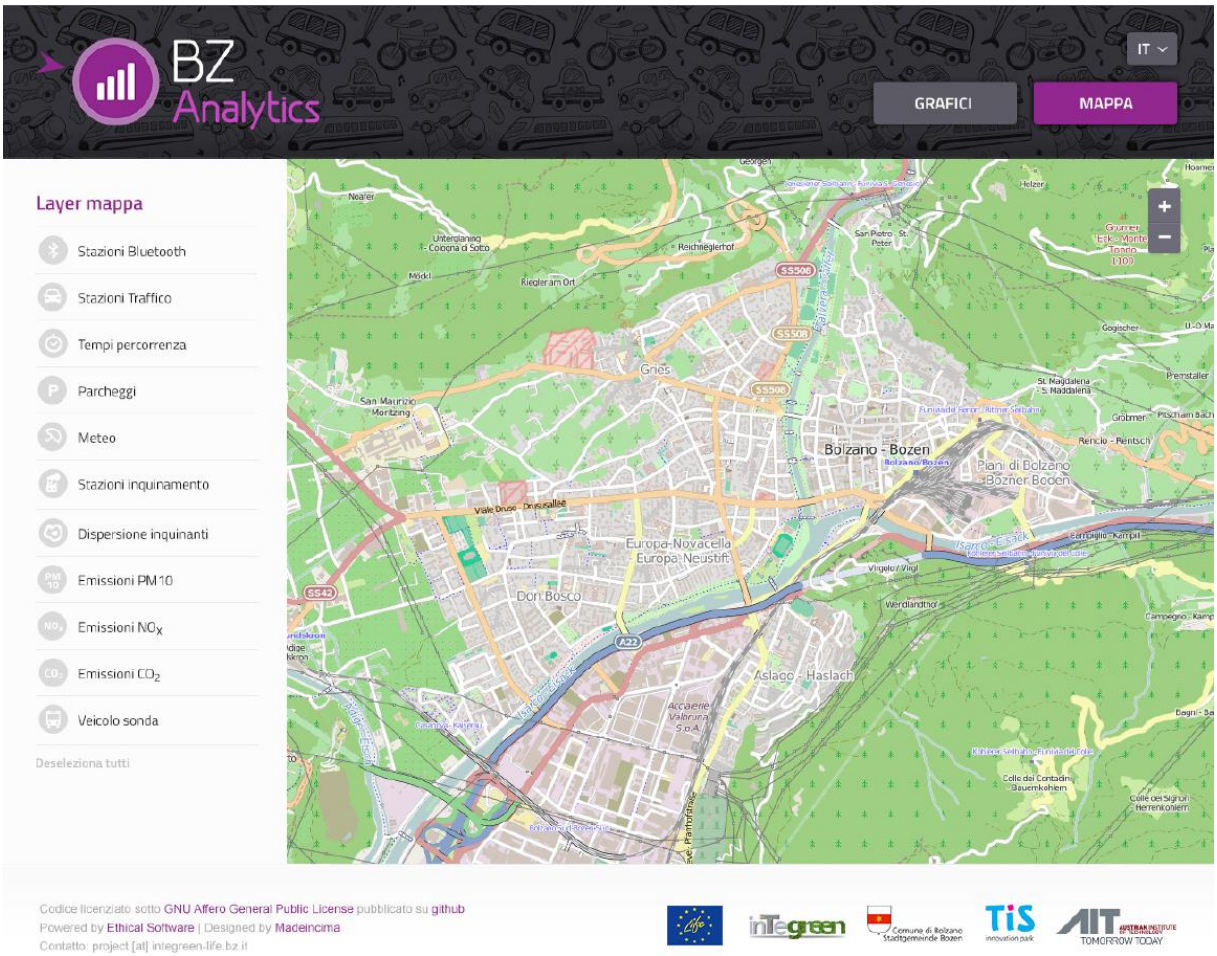

Figure 9: The main page of the operators web interface "BZAnalytics" (section maps).

In the following figures, it is shown what the operators can see if he/she decides to activate one of each options. Each station is associated by a color. The intensity of each icon is associated to an indicator associated to the "amount" of detections associated to each station (e.g. vehicular transits, air pollution levels, level of occupancy of parking areas, etc.) By clicking on an icon on a map, the operator can moreover see the most relevant real-time data associated to this station. Even in this case, it is possible to have multiple layers "on" in order to check for possible correlations (e.g. bad weather + parking areas full occupied).

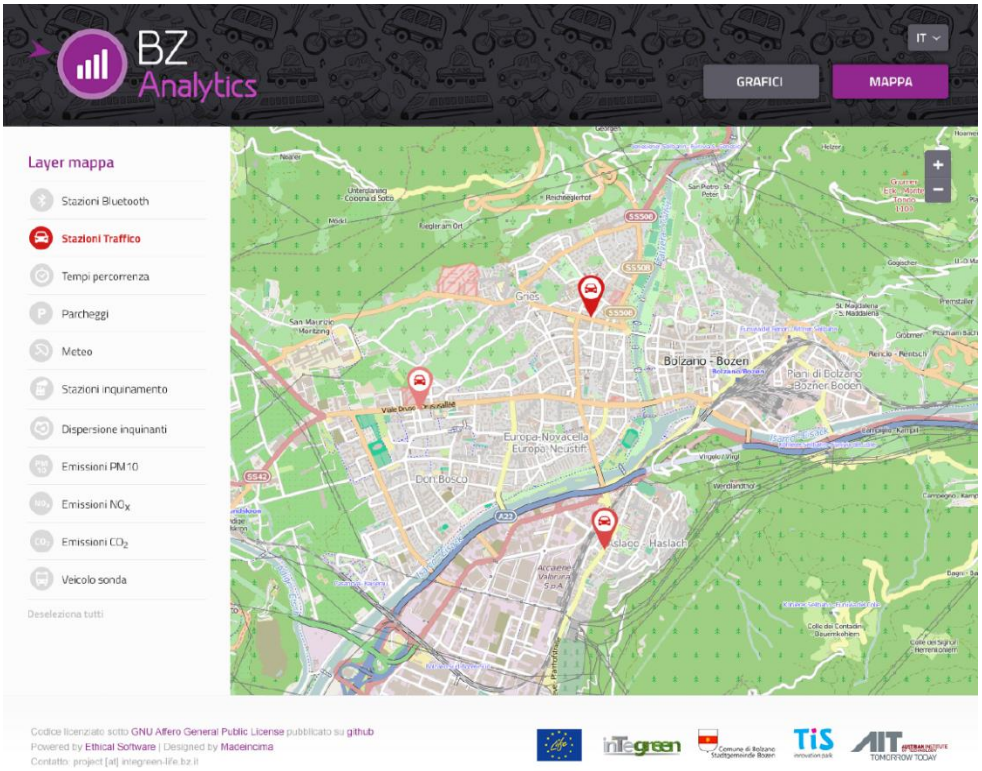

Figure 10: The map visualization of the traffic stations on the operators web interface "BZAnalytics".

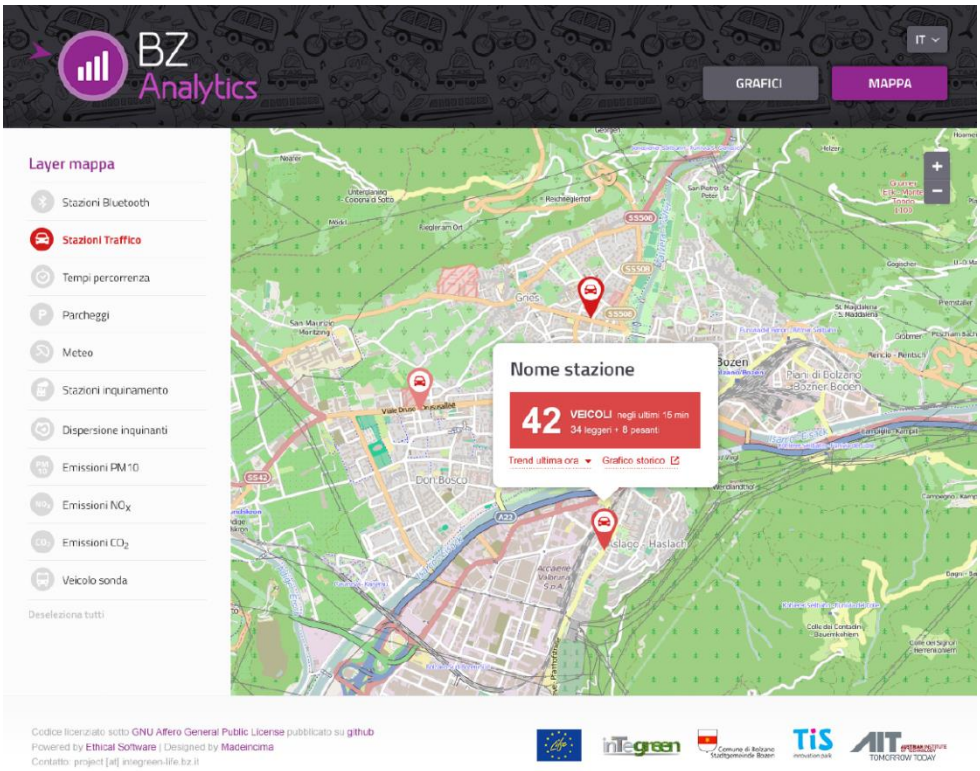

Figure 11: The map visualization of the traffic stations on the operators web interface "BZAnalytics" (tooltip detail).

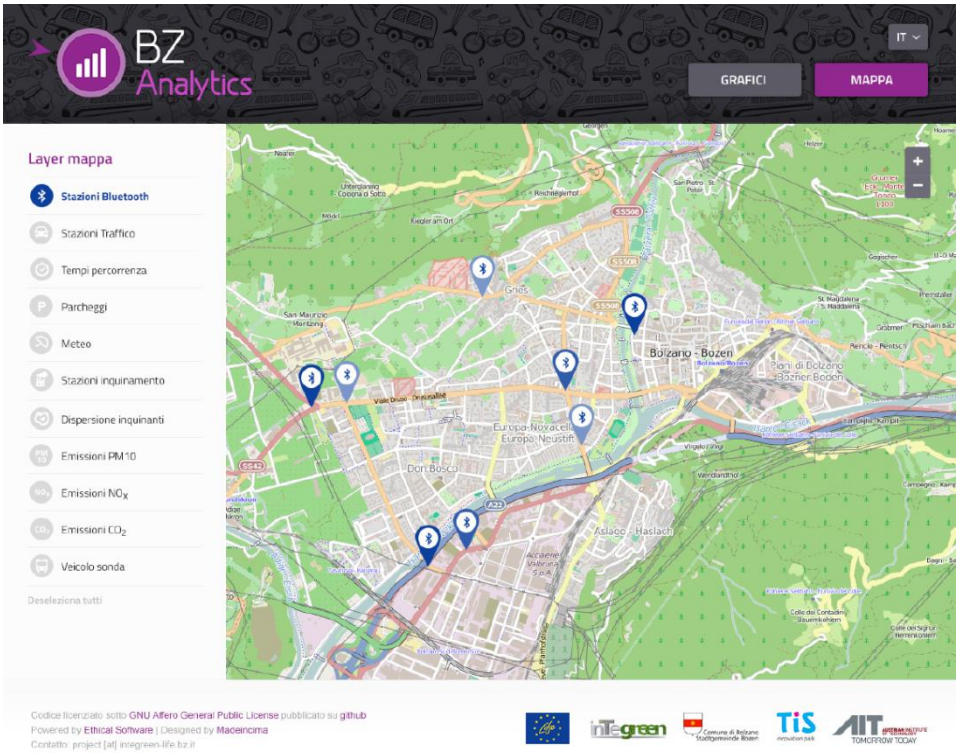

Figure 12: The map visualization of the Bluetooth stations on the operators web interface "BZAnalytics".

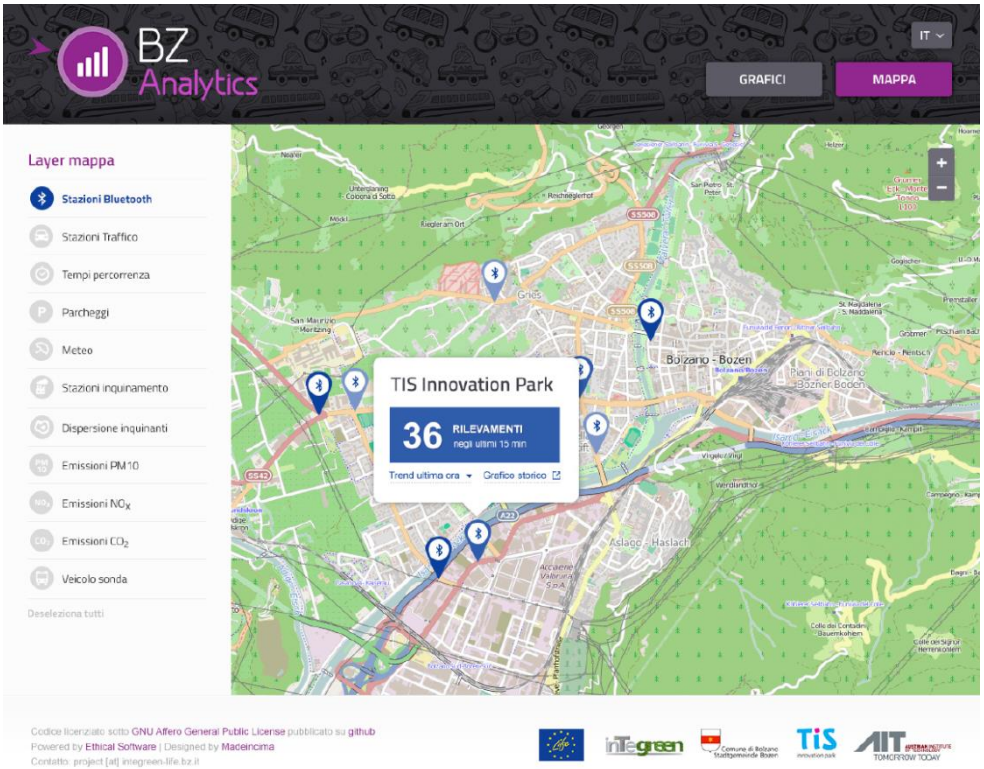

Figure 13: The map visualization of the Bluetooth stations on the operators web interface "BZAnalytics" (tooltip detail).

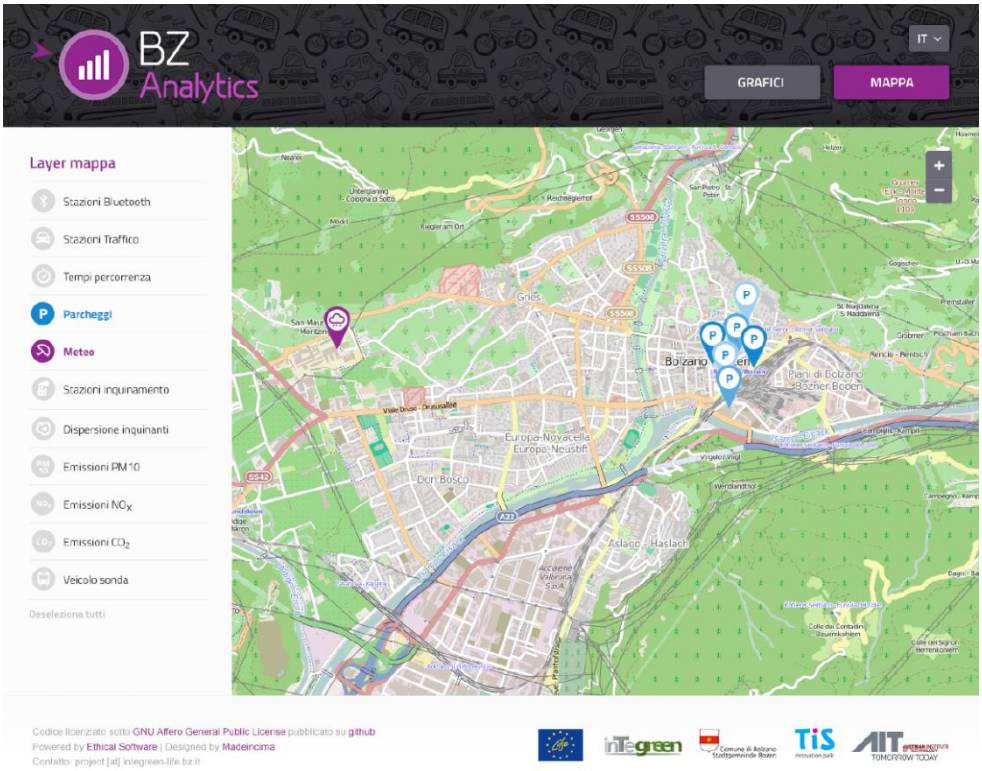

Figure 14: The map visualization of the parking areas and the meteorological stations on the operators web interface "BZAnalytics".

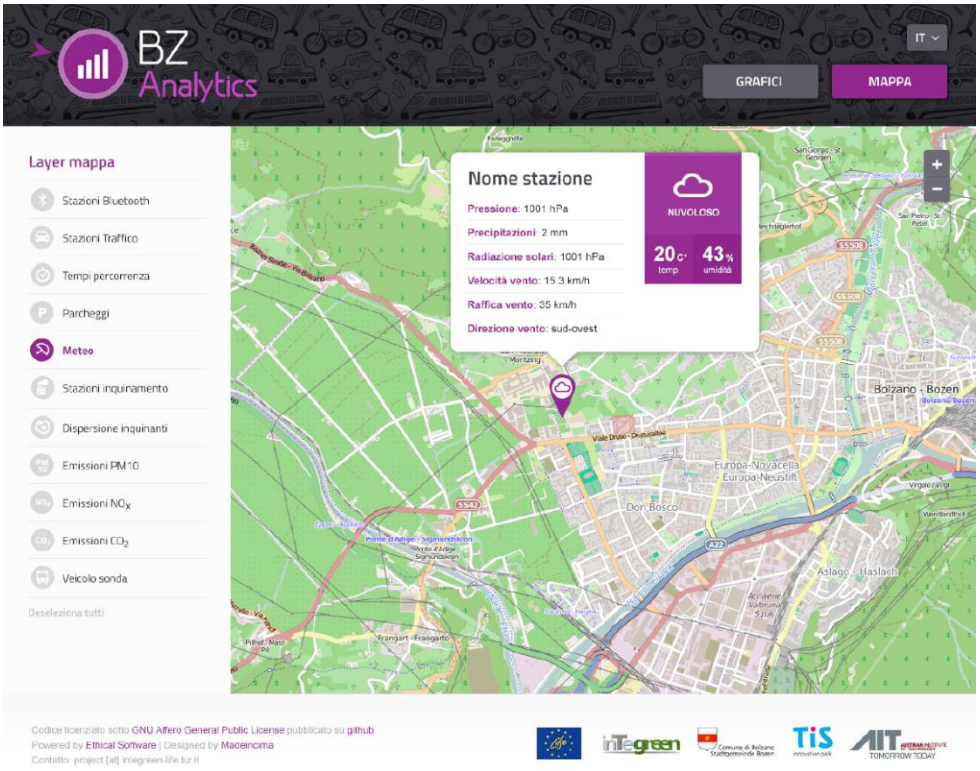

Figure 15: The map visualization of the meteorological stations on the operators web interface "BZAnalytics" (tooltip detail).

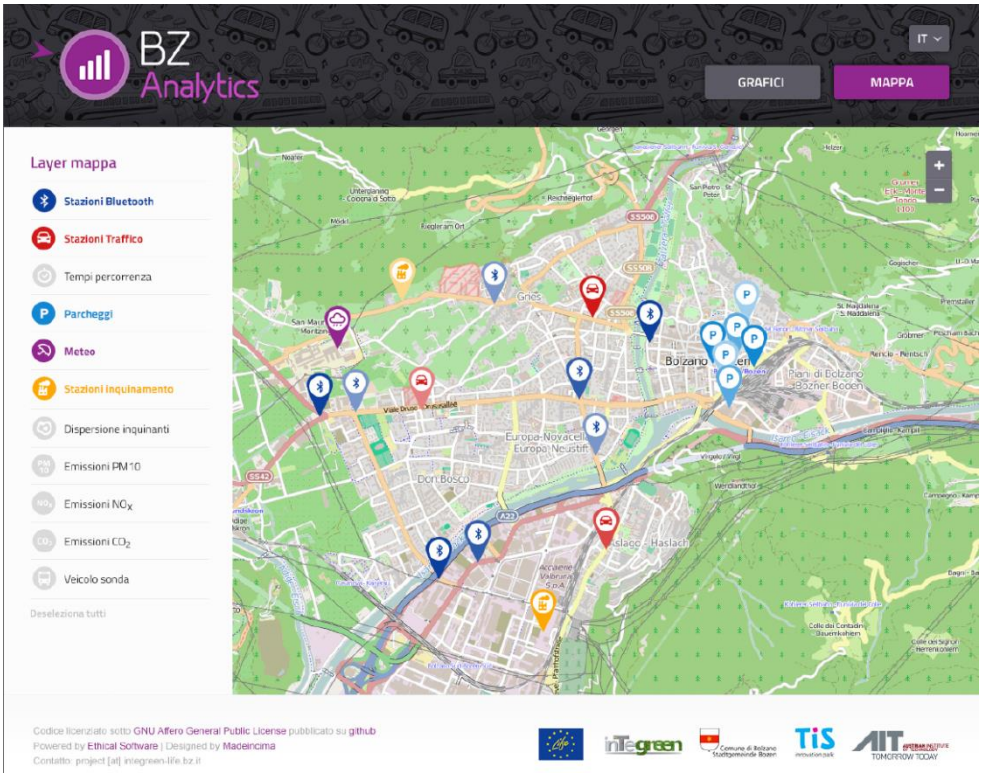

Figure 16: Overall view of fixed monitoring stations on the operators web interface "BZAnalytics".

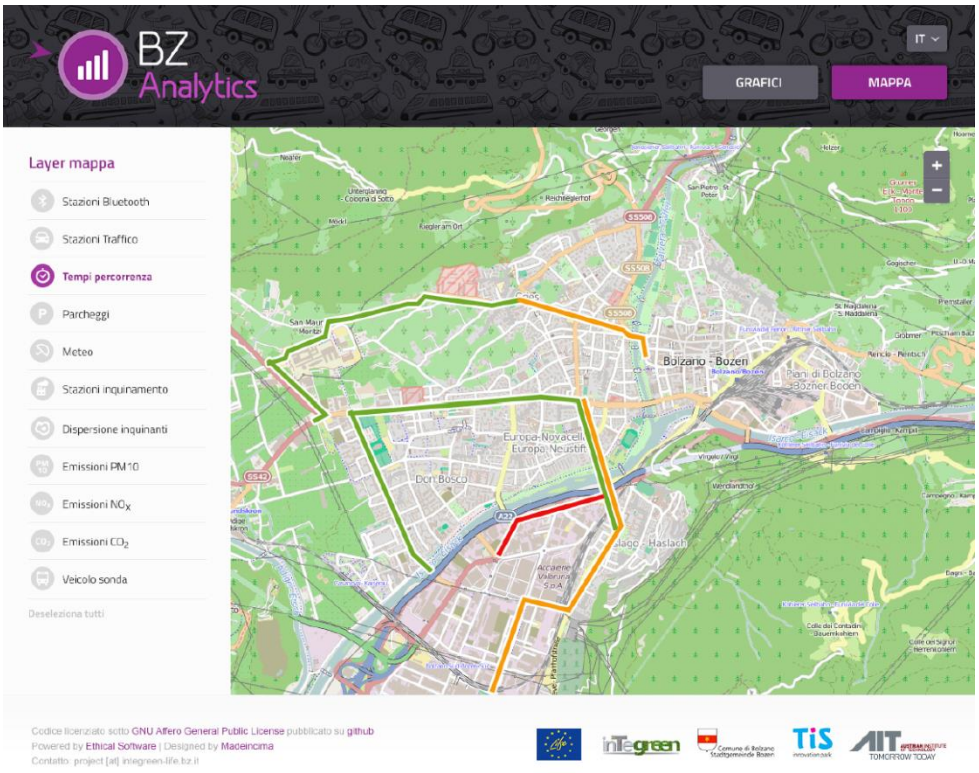

Figure 17: The map visualization of the vehicular travel times on the operators web interface "BZAnalytics".

Figure 18: The map visualization of the emissions on the operators web interface "BZAnalytics".

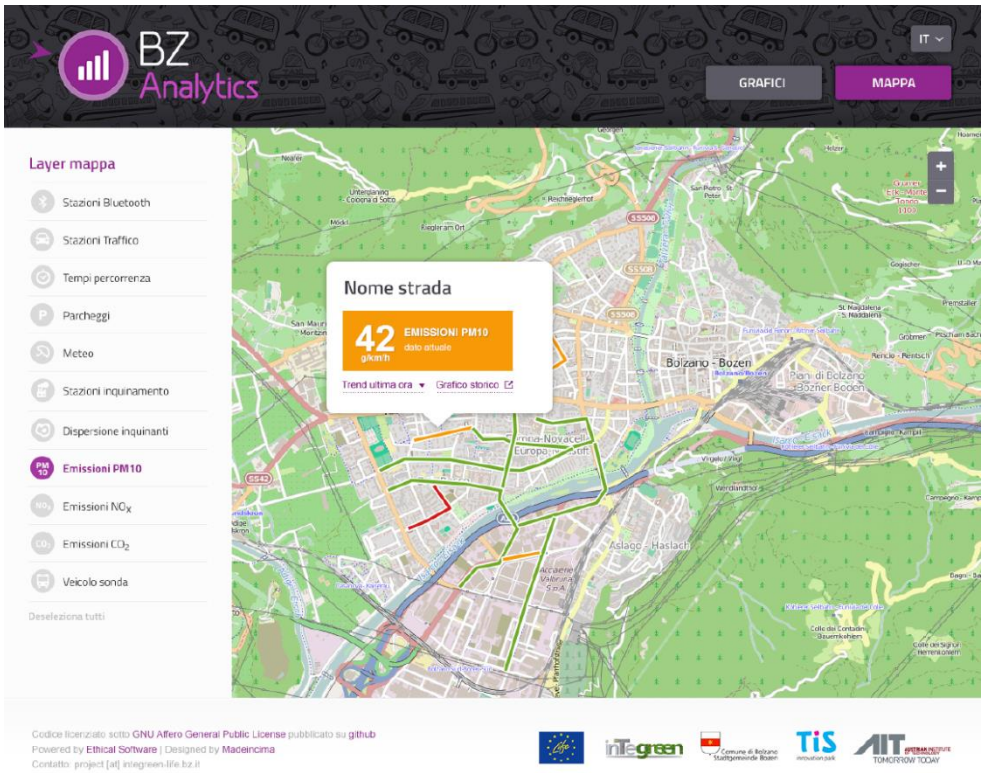

Figure 19: The map visualization of the emissions on the operators web interface "BZAnalytics" (tooltip detail).

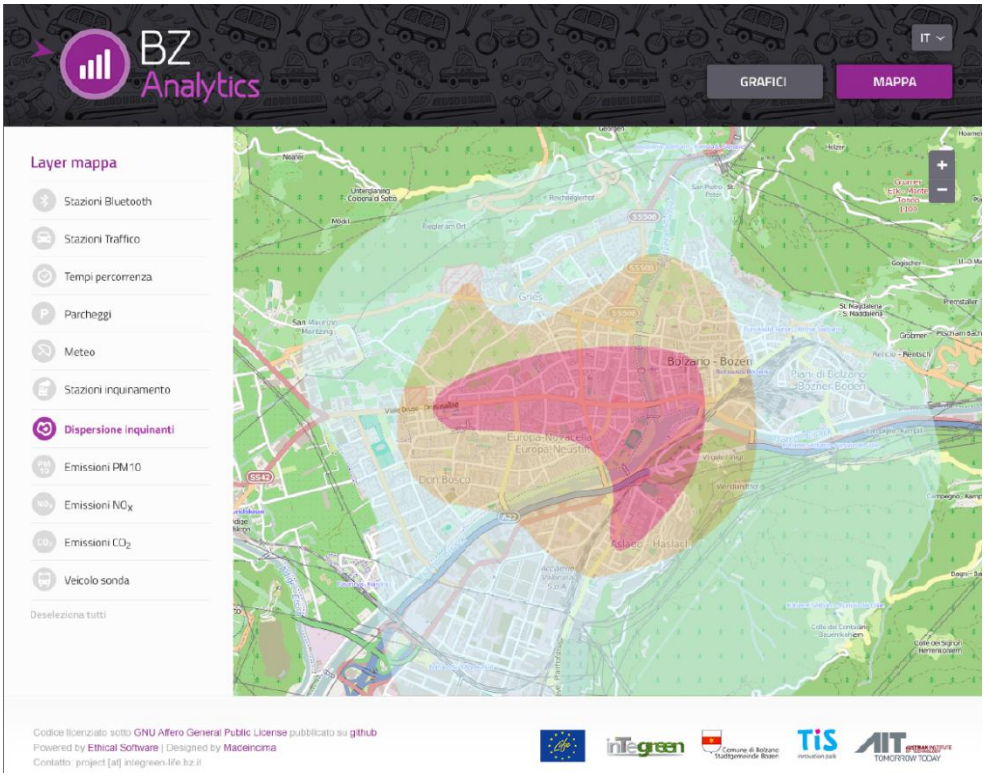

Figure 20: The map visualization of the dispersion model output on the operators web interface "BZAnalytics".

Figure 21: The map visualization of the mobile system data on the operators web interface "BZAnalytics".

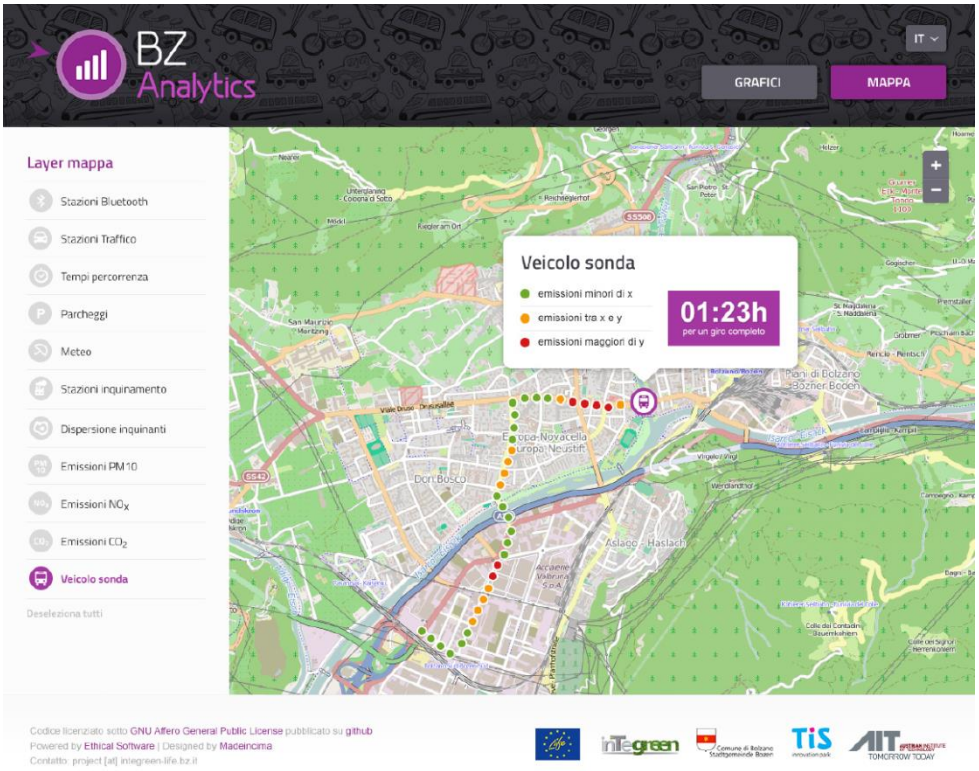

Figure 22: The map visualization of the mobile system data on the operators web interface "BZAnalytics" (mobile system tooltip detail).

Figure 23: The map visualization of the mobile system data on the operators web interface "BZAnalytics" (mobile measurement tooltip detail).

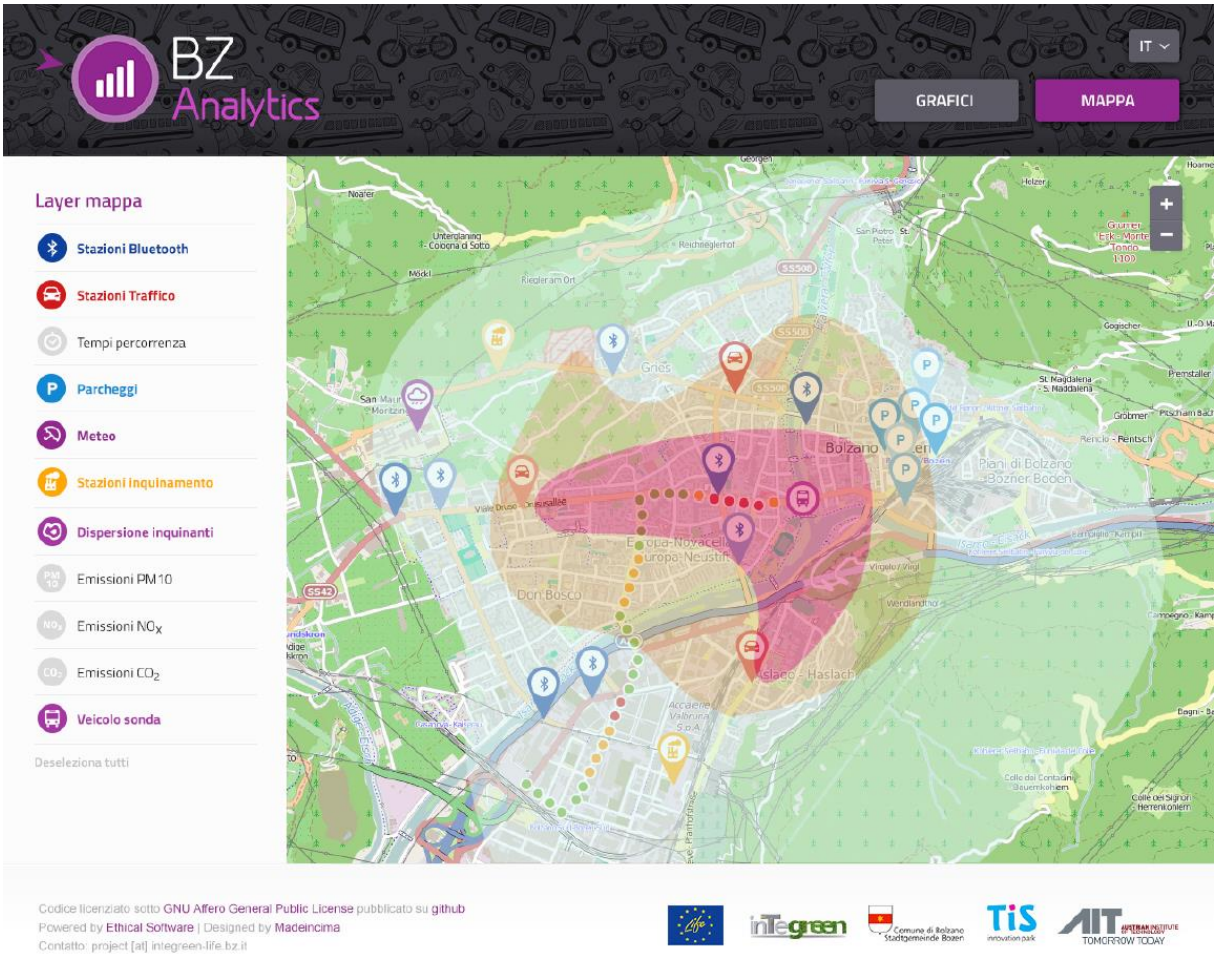

Figure 24: The complete map visualization of the INTEGREEN data and elaborations on the operators web interface "BZAnalytics".

Conclusions

The report has presented the final implementation status of the operators centre web interface prototype, called "**BZAnalytics**". This HTML5 application, that traffic operators can access from any device with Internet connection, is reachable at the link http://analytics.mobility.bz.it, allows to control both in real-time mode as well as in offline conditions the time series of the measurements of the heterogeneous monitoring systems covered by INTEGREEN. Through an additional map overview, specialized users of this application can immediately check the presence of traffic jams of air pollution peaks.

This application relies on a custom **open-standard front-end API** that will be easily scalable once new data types will be available.

Bibliography

[1] INTEGREEN consortium, "D.3.1.1 "Data management unit and environmental stations front-end"," 2014.

[2] R. Johnson, J. Hoeller, A. Arendsen, T. Risberg and C. Sampaleanu, Professional Java Development with the Spring Framework, Wiley Publishing, 2005.

[3] J. C. Castrejon, R. Lopez-Landa and R. Lozano, "Model2Roo: A Model Driven Approach for Web Application Development based on the Eclipse Modeling Framwork and Spring Roo," in International Conference on Electric Communications and Computers, Choula, Puebla (Mexico), 2011.