Sii-Mobility

Supporto di Interoperabilità Integrato per i Servizi al Cittadino e alla Pubblica Amministrazione

Trasporti e Mobilità Terrestre, SCN_00112

Deliverable ID: DE3.16z
Title: Smart City API guidelines

Data corrente	06-04-2017
Versione (solo il responsabile puo' cambiare versione	1.6
Versione API	1
Stato (draft, final)	Current
Livello di accesso (solo consorzio, pubblico)	Pubblico
WP	
Natura (report, report e software, report e HW..)	
Data di consegna attesa	04-05-2017
Data di consegna effettiva	
Referente primario, coordinatore del documento	Paolo Nesi, Pierfrancesco Bellini
Contributor	
Coordinatore responsabile del progetto	Paolo Nesi, UNIFI, paolo.nesi@unifi.it

Sommario

1 Executive Summary

This document is reporting the description of the Smart City API supported by Km4City and Sii-Mobility. In addition, other features have been added thanks to the projects: RESOLUTE and REPLICATE.
The data underlined of the Km4City is defined as Km4City Ontology. The ontology can be accessed from the Km4City documentation: http://www.km4city.org/?infoDocs
The document is presenting the API and the corresponding scenarios.

Licensing constraints:

- **The Smart City API are now almost open, while the massive download of data is not allowed and strictly forbidden, as well as the download of data to store them in server on your premise.**
- **The data obtained from the Smart City API cannot be used for commercial purpose, direct or indirect. So that, they cannot be used to sell them, neither to set up a service to make money in any way, as a service or renting.**
- **The Smart City API can be freely used to set up free of charge web and mobile app, without storing data on local storage.**

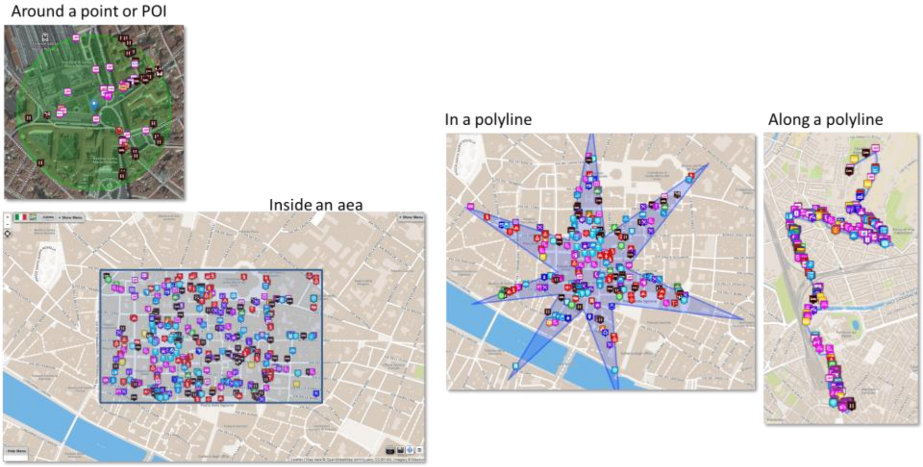
These conditions will be valid until a new communication will be performed and are retroactively valid since the creation of the service in the 2015.

2 Scenarios

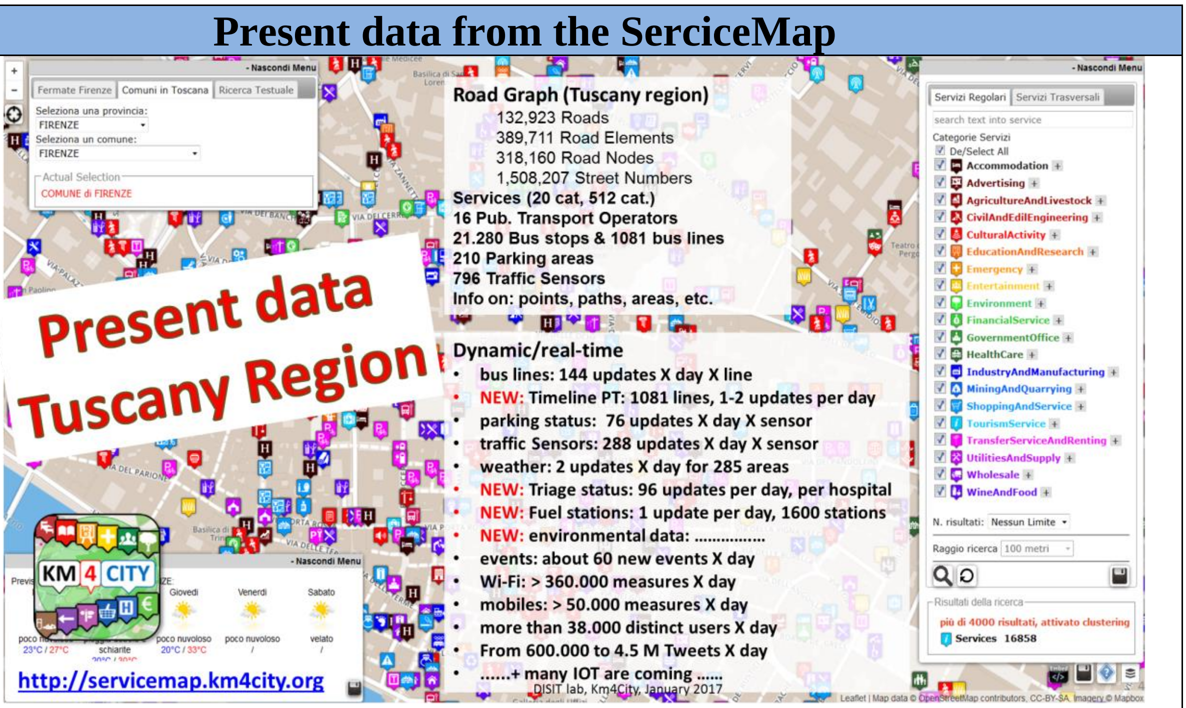
The main scabrous are:

- Empowering the city users
- Access to event information
- Supporting City Users in using Public Mobility
- Supporting City Users in using Private Mobility
- New Experience to access at Cultural and Touristic info
- New way to access at health services
- Access at Environmental information
- Profiled Suggestions to City Users
- Personal Assistant

See Slides http://www.disit.org/6995 for a summary on scenarios.

Search by shape or distance	
Each request or search in the Km4City model can be referred to a point and a ray, to an area, to a polyline	 Around a point or POI Inside an aea In a polyline Along a polyline

Present data from the SerciceMap



Road Graph (Tuscany region)
132,923 Roads
389,711 Road Elements
318,160 Road Nodes
1,508,207 Street Numbers

Services (20 cat, 512 cat.)
16 Pub. Transport Operators
21.280 Bus stops & 1081 bus lines
210 Parking areas
796 Traffic Sensors
Info on: points, paths, areas, etc.

Dynamic/real-time

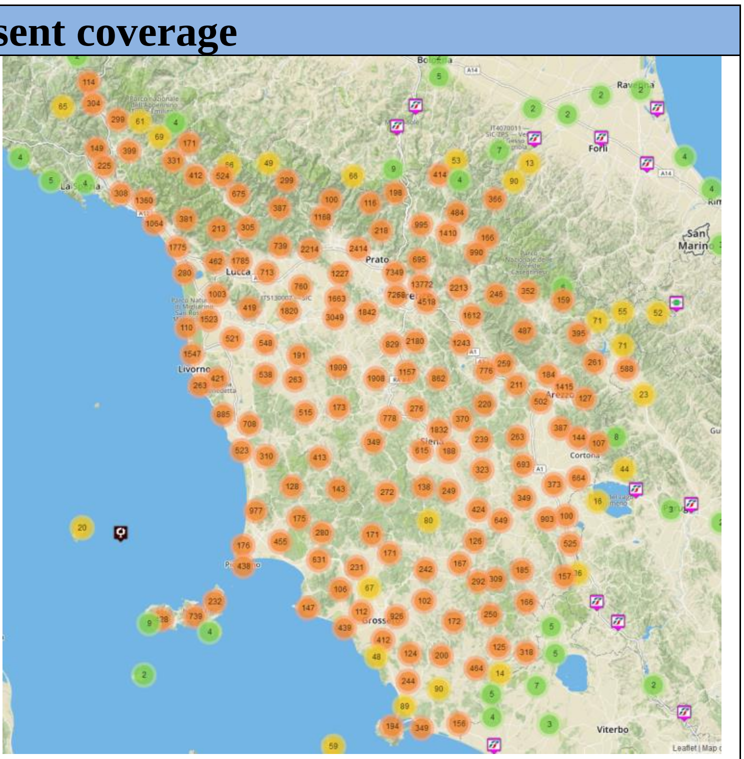
- **bus lines: 144 updates X day X line**
- **NEW: Timeline PT: 1081 lines, 1-2 updates per day**
- **parking status: 76 updates X day X sensor**
- **traffic Sensors: 288 updates X day X sensor**
- **weather: 2 updates X day for 285 areas**
- **NEW: Triage status: 96 updates per day, per hospital**
- **NEW: Fuel stations: 1 update per day, 1600 stations**
- **NEW: environmental data: ....**
- **events: about 60 new events X day**
- **Wi-Fi: > 360.000 measures X day**
- **mobiles: > 50.000 measures X day**
- **more than 38.000 distinct users X day**
- **From 600.000 to 4.5 M Tweets X day**
- **......+ many IOT are coming ......**

DISIT lab, Km4City, January 2017

http://servicemap.km4city.org

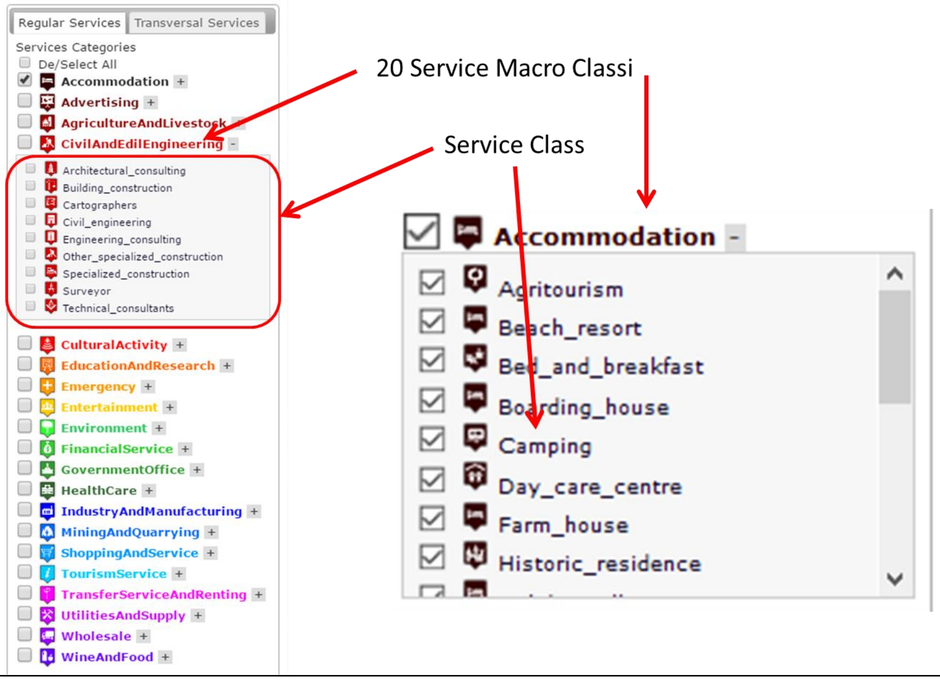
Present coverage

- **Street and geoinformation of the territory and details for routing, navigation, …**
- **Mobility and Transport**: public and private,
  public transport, parking status, fuel stations prices, traffic sensors, etc.
- **Culture and Tourism**: POI, churches,
  museum, schools, university, theatres,
  events in Florence
- **Environmental**: pollution real time, weather forecast, etc.
- **Social Media**: twitter data
- **Health**: hospital, pharmacies, status of the first aid triage in major hospitals, …
- **Alarms**: civil protection alerts, hot areas, …



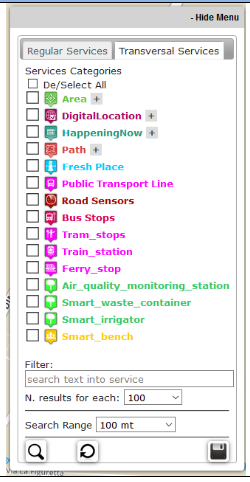
Present coverage

Concepts of Services: Macro and sub category



Access to Point of Interest information, POI

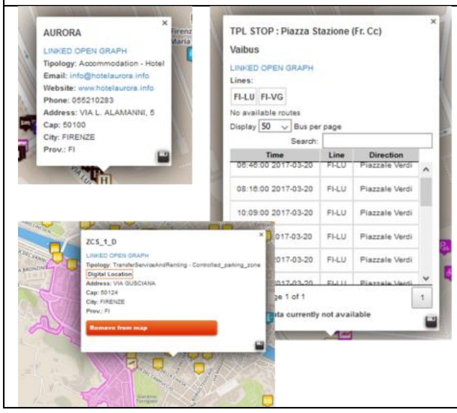
- **POI**: point of interest:
- **type**: macro and categoria
- **Position**: GPS, addres, telephone, fax, email, URL, ..
- **Description**: textual, multilingua, with images, …
- **Link** to dbPedia, Linked Open Data
- **Links to other services**
- **Real time data if any:** sensors data, timeline, events, prices, opening time, rules of access, status of services, status of queue, etc..
- *See transversal services on ServiceMap*
- *Regular and in test platform*



Filter: search text into service

N. results for each: 100

Search Range 100 mt



AURORA

LINKED OPEN GRAPH

Tipology: Accommodation - Hotel

Email: info@hotelaurora.info

Website: www.hotelaurora.info

Phone: 055210283

Address: VIA L. ALAMANNI, 5

Cap: 50100

City: FIRENZE

Prov.: FI

TPL STOP : Piazza Stazione (Fr. Cc)

Vaibus

LINKED OPEN GRAPH

Lines:

FI-LU FI-VG

No available routes

Display 50 Bus per page

Search:

Time	Line	Direction
06:46:00 2017-03-20	FI-LU	Piazzale Verdi
08:16:00 2017-03-20	FI-LU	Piazzale Verdi
10:09:00 2017-03-20	FI-LU	Piazzale Verdi
017-03-20	FI-LU	Piazzale Verdi
017-03-20	FI-LU	Piazzale Verdi
017-03-20	FI-LU	Piazzale Verdi

ge 1 of 1

ta currently not available

ZCS_1_D

LINKED OPEN GRAPH

Tipology: TransferServiceAndRenting - Controlled_parking_zone

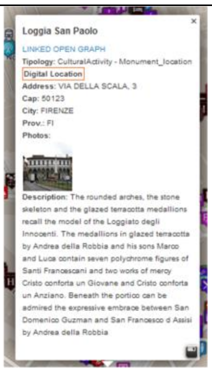
Digital Location

Address: VIA GUSCIANA

Cap: 50124

City: FIRENZE

Prov.: FI



Loggia San Paolo

LINKED OPEN GRAPH

Tipology: CulturalActivity - Monument_location

Digital Location

Address: VIA DELLA SCALA, 3

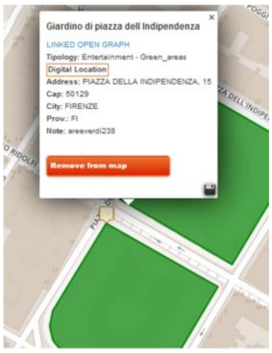
Cap: 50123

City: FIRENZE

Prov.: FI

Photos:

Description: The rounded arches, the stone skeleton and the glazed terracotta medallions recall the model of the Loggiato degli Innocenti. The medallions in glazed terracotta by Andrea della Robbia and his sons Marco and Luca contain seven polychrome figures of Santi Francescani and two works of mercy Cristo conforta un Giovane and Cristo conforta un Anziano. Beneath the portico can be admired the expressive embrace between San Domenico Guzman and San Francesco d Assisi by Andrea della Robbia



Giardino di piazza dell Indipendenza

LINKED OPEN GRAPH

Tipology: Entertainment - Green_areas

Digital Location

Address: PIAZZA DELLA INDIPENDENZA, 15

Cap: 50129

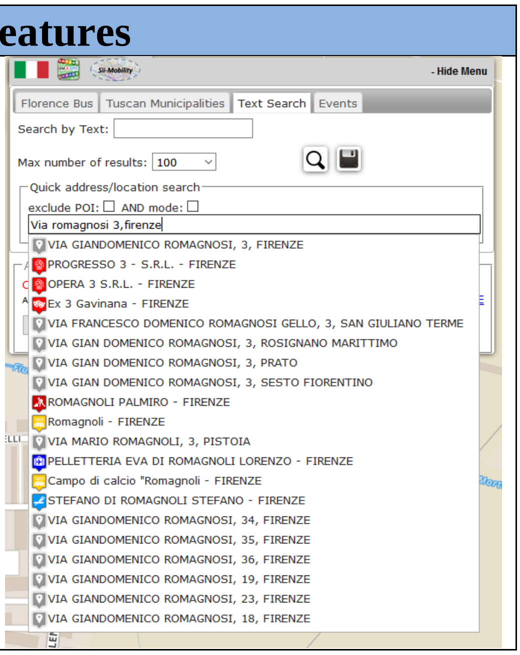
City: FIRENZE

Prov.: FI

Note: areeverdi238

General Text Search Features

- **Search by text for POIs via**
  - Full text: description, title, macro and category name
  - Filtering by macro and category
  - Filtering on distance and geometric shape
- **Search by text with assisted suggestion to get:**
  - Streets and civic numbers, or POI, locations



Search by Text:

Max number of results: 100

Quick address/location search

exclude POI: □ AND mode: □

Via romagnosi 3,firenze

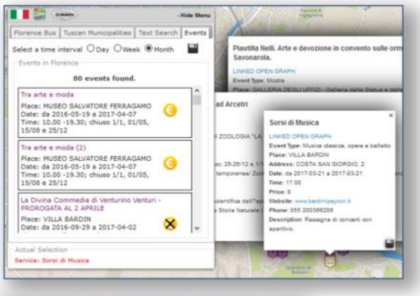
VIA GIANDOMENICO ROMAGNOSI, 3, FIRENZE
PROGRESSO 3 - S.R.L. - FIRENZE
OPERA 3 S.R.L. - FIRENZE
Ex 3 Gavinana - FIRENZE
VIA FRANCESCO DOMENICO ROMAGNOSI GELLO, 3, SAN GIULIANO TERME
VIA GIAN DOMENICO ROMAGNOSI, 3, ROSIGNANO MARITTIMO
VIA GIAN DOMENICO ROMAGNOSI, 3, PRATO
VIA GIAN DOMENICO ROMAGNOSI, 3, SESTO FIORENTINO
ROMAGNOLI PALMIRO - FIRENZE
Romagnoli - FIRENZE
VIA MARIO ROMAGNOLI, 3, PISTOIA
PELLETTERIA EVA DI ROMAGNOLI LORENZO - FIRENZE
Campo di calcio "Romagnoli - FIRENZE
STEFANO DI ROMAGNOLI STEFANO - FIRENZE
VIA GIANDOMENICO ROMAGNOSI, 34, FIRENZE
VIA GIANDOMENICO ROMAGNOSI, 35, FIRENZE
VIA GIANDOMENICO ROMAGNOSI, 36, FIRENZE
VIA GIANDOMENICO ROMAGNOSI, 19, FIRENZE
VIA GIANDOMENICO ROMAGNOSI, 23, FIRENZE
VIA GIANDOMENICO ROMAGNOSI, 18, FIRENZE

Empowering City Users

- Allow city users to
  - provide comments, images and scores associated with a certain Service (or place, via GPS)
  - Get list of last contributions of the same kind provided by other users
- They can be:
  - used as feedbacks
  - moderated by a back office personnel
  - ..
- In the future (→) connection with a more powerful server based on 311 standard would be possible

Access to Event information

- Getting Events in the city
  - Theater, museum, show, sport, etc.
- Getting Event details
  - Event kind, and thus ordering
  - in the day, week, and month
  - Location, and thus ordering, or selecting events per area, per residence
  - General information
  - Opening and cost (if any)
  - Etc.
- In the future → getting RSS Feeds and general news from agencies



Select a time interval ○ Day ○ Week ● Month

Events in Florence

80 events found.

Tra arte e moda
Place: MUSEO SALVATORE FERRAGAMO
Date: da 2016-05-19 a 2017-04-07
Time: 10.00 -19.30; chiuso 1/1, 01/05, 15/08 e 25/12

Tra arte e moda (2)
Place: MUSEO SALVATORE FERRAGAMO
Date: da 2016-05-19 a 2017-04-07
Time: 10.00 -19.30; chiuso 1/1, 01/05, 15/08 e 25/12

La Divina Commedia di Venturino Venturi - PROROGATA AL 2 APRILE
Place: VILLA BARDIN
Date: da 2016-09-29 a 2017-04-02

Actual Selection
Service: Sorsi di Musica

Plautilla Nelli. Arte e devozione in convento sulle orm
Savonarola.
LINKED OPEN GRAPH
Event Type: Mostre
Place: GALLERIA DEGLI UFFIZI - Galleria delle Statue e delle

Sorsi di Musica
LINKED OPEN GRAPH
Event Type: Musica classica, opera e balletto
Place: VILLA BARDIN
Address: COSTA SAN GIORGIO, 2
Date: da 2017-03-21 a 2017-03-21
Time: 17.00
Price: 8
Website: www.bardinipeyron.it
Phone: 055 200366206
Description: Rassegna di concerti con aperitivo.

Supporting City Users in using Public Mobility

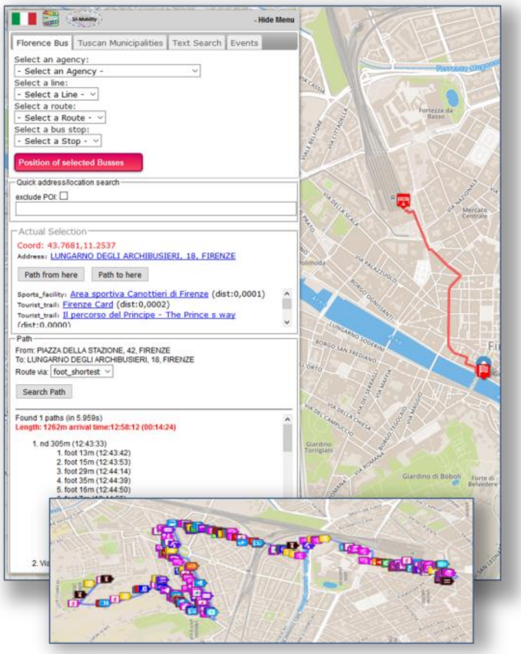
- **Public Transport, PT**
- Getting tickets
- Getting bus stops, lines, and timelines for bus, train and tramline
- Searching Services along a PT line or closer to a stop
- Searching the closest bus stops
- searching for BUS stops via name
- In the future → multimodal routing for Pub. Transport
- In the future → real time delays of busses

Supporting City Users in using Private Mobility

- **Private Transport**
- Getting closer parking
- Getting closer free space on parking and in the future predictions of the free places
- Getting fuel stations location and prices according to the different products of gasoline
- Searching Services along a path or closer to a point or Service as Hotel, Restaurants, square, etc.
- Getting closer cycling paths
- Saving the position of the private car, getting the path to ricover the position of the car/
- In the future → Recharging stations: location and status
- In the future → Getting traffic information

Private Mobility: routing and navigation paths

- To get the path from two points/POIs:
  - Shortest for pedestrian
  - Quietest for pedestrian
  - Shortest for private vehicles
- A text search can be used to find the place when one is interested to go.
- Search for POIs along the identified Path!
- http://www.disit.org/ServiceMap



Select an agency:
- Select an Agency -
Select a line:
- Select a Line -
Select a route:
- Select a Route -
Select a bus stop:
- Select a Stop -

Quick address/location search
exclude POI: □

Actual Selection
Coord: 43.7681,11.2537
Address: LUNGARNO DEGLI ARCHIBUSIERI, 18, FIRENZE

Sports_facility: Area sportiva Canottieri di Firenze (dist:0,0001)
Tourist_trail: Firenze Card (dist:0,0002)
Tourist_trail: Il percorso del Principe - The Prince s way (dist:0,0000)

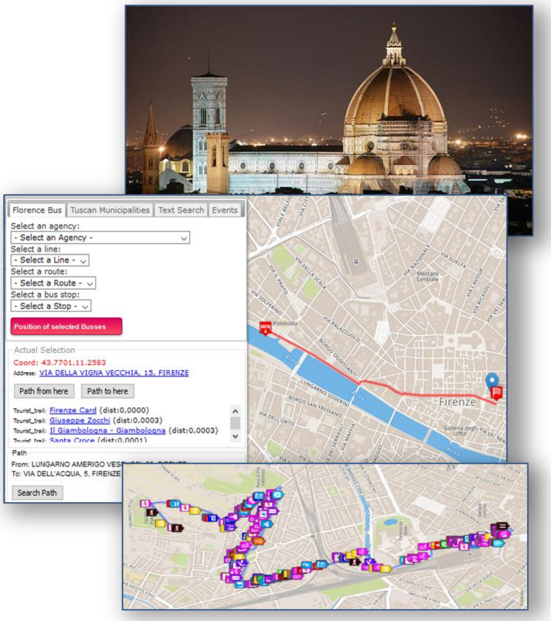
Path
From: PIAZZA DELLA STAZIONE, 42, FIRENZE
To: LUNGARNO DEGLI ARCHIBUSIERI, 18, FIRENZE
Route via: foot_shortest

Found 1 paths (in 5.959s)
Length: 1262m arrival time:12:58:12 (00:14:24)

```
1. nd 305m (12:43:33)
        1. foot 13m (12:43:42)
        2. foot 15m (12:43:53)
        3. foot 29m (12:44:14)
        4. foot 35m (12:44:39)
        5. foot 16m (12:44:50)
2. Via
```

New Experience to access at Cultural and Touristic info

- Getting location and description of Point of Interests, POIs: culture and tourism first
  - Location, images, phone, URL, etc.
- Search for POIs in areas, closer to a point, along a line/path.
- Get routing to reach location or POI by walking downtown
  - searching Services along the path
- Search for location, full text assisted



Select an agency:
- Select an Agency -
Select a line:
- Select a Line -
Select a route:
- Select a Route -
Select a bus stop:
- Select a Stop -

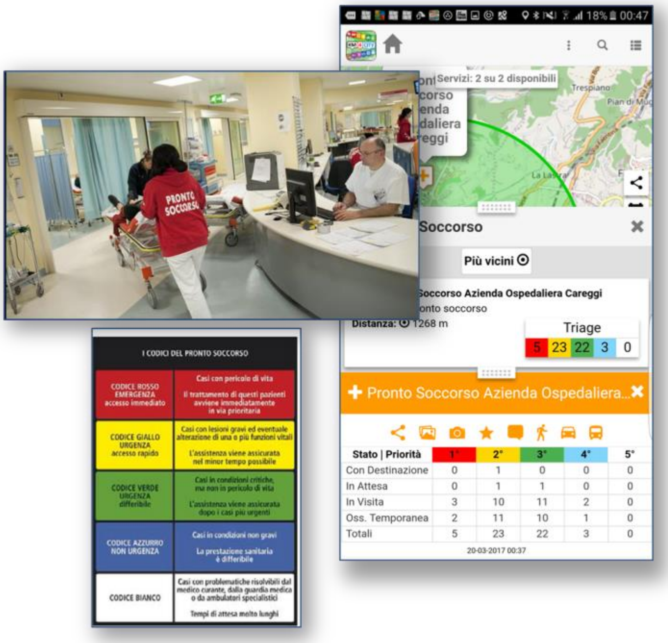
Actual Selection
Coord: 43.7701,11.2583
Address: VIA DELLA VIGNA VECCHIA, 15, FIRENZE

Tourist_trail: Firenze Card (dist:0,0000)
Tourist_trail: Giuseppe Zocchi (dist:0,0003)
Tourist_trail: Il Giambologna - Giambologna (dist:0,0003)
Tourist_trail: Santa Croce (dist:0,0001)

Path
From: LUNGARNO AMERIGO VES
To: VIA DELL'ACQUA, 5, FIRENZE

New way to access at health services

- Searching for pharmacies and hospitals
- Getting the closest hospital first aid locations and status
- Getting real time updated information about the first aid status of major hospitals (triage)
- http://www.disit.org/ServiceMap



Servizi: 2 su 2 disponibili

Soccorso

Soccorso Azienda Ospedaliera Careggi
onto soccorso
Distanza: 1268 m

Triage				
5	23	22	3	0

Pronto Soccorso Azienda Ospedaliera...

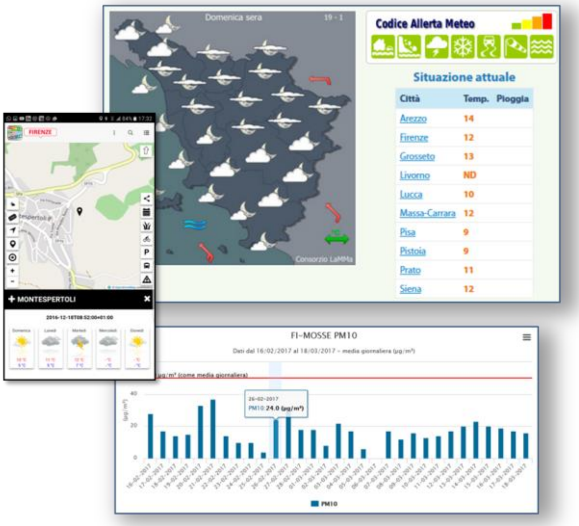
Stato | Priorità	1°	2°	3°	4°	5°
Con Destinazione	0	1	0	0	0
In Attesa	0	1	1	0	0
In Visita	3	10	11	2	0
Oss. Temporanea	2	11	10	1	0
Totali	5	23	22	3	0

20-03-2017 00:37

I CODICI DEL PRONTO SOCCORSO	
CODICE ROSSO EMERGENZA accesso immediato	Casi con pericolo di vita Il trattamento di questi pazienti avviene immediatamente in via prioritaria
CODICE GIALLO URGENZA accesso rapido	Casi con lesioni gravi ed eventuale alterazione di una o più funzioni vitali L'assistenza viene assicurata nel minor tempo possibile
CODICE VERDE URGENZA differibile	Casi in condizioni critiche, ma non in pericolo di vita L'assistenza viene assicurata dopo i casi più urgenti
CODICE AZZURRO NON URGENZA	Casi in condizioni non gravi La prestazione sanitaria è differibile
CODICE BIANCO	Casi con problematiche risolvibili dal medico curante, dalla guardia medica o da ambulatori specialistici Tempi di attesa molto lunghi

Access at Environmental information

- Getting weather forecast for the next days
- Getting alert information from Civil protection
- Getting air quality status: pollution, PM2, PM10, etc.
- In the future → getting actual weather status: temperature, humidity, pressure, rain level, etc.

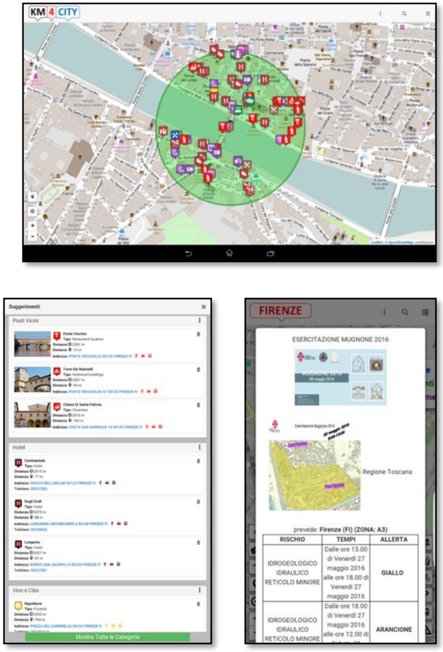


Situazione attuale

Città	Temp.	Pioggia
Arezzo	14	
Firenze	12	
Grosseto	13	
Livorno	ND	
Lucca	10	
Massa-Carrara	12	
Pisa	9	
Pistoia	9	
Prato	11	
Siena	12	

Profiled Suggestions to City Users

- Personalized suggestions
- The server provides suggestions in the user context (location and time) arranged in a number of categories
  - Culture, mobility, food and drink, etc.
  - Alerts: civil protection, city council, twitter data, etc.
- The city user may reject some of them, thus the suggestion engine learns about preferred topics and category



prevede: Firenze (FI) (ZONA: A3)

RISCHIO	TEMPI	ALLERTA
IDROGEOLOGICO IDRAULICO RETICOLO MINORE	Dalle ore 13.00 di Venerdì 27 maggio 2016 alle ore 18.00 di Venerdì 27 maggio 2016	GIALLO
IDROGEOLOGICO IDRAULICO RETICOLO MINORE	Dalle ore 18.00 di Venerdì 27 maggio 2016 alle ore 12.00 di	ARANCIONE

Profiled Engagements to City Users

- The users are profiled to learn habits:
  - Personal POI and paths
  - Mobility habits
- Information and engagements sent to the city users are programmed according to the user evolution to:
  - Stimulate virtuous habits
  - More sustainable habits
  - More healthy habits, etc.
  - Get feedbacks
  - Provide bonus and prices, …..
  - Send alerts, ….

3 Service MAP vs Smart City API

ServiceMap is a service and tool to pose geographic queries and see the knowledge base produced by the harvesting process based on **Km4City** model that includes: Street Graph from Tuscany region, Open Data from Florence Municipality, traffic monitoring, geo and weather forecast information from LAMMA, traffic sensors, services, events, parkings, real time of busses, from Florence Municipality, etc. Some of these data provide real time information as the status of the busses on the bus lines in Florence, parking sensors in Florence and Empoli, traffic sensors in Florence and Empoli, events, and the weather forecast in Tuscany. The ServiceMap accesses to an RDF Store based on **Km4City ontology and model**.

Service map API can be very useful for shortening the time for developers that want to realize Web or Mobile Apps, exploiting the km4city data. As a service, different kinds of queries can be saved from the ServiceMap when you find the icon disk . They are located on Services, Weather panel, selection panel and on the general web page. Them, click them and fill the form, thus the servicemap will perform two issues (1) save the query performed in a database for your further reuse, (2) send to you an email with a set of links, which are the specific REST calls that you can use to invoke the ServiceMap API from Web and Mobile applications to obtain the service you requested.
In addition, it is possible to take the service map and put in a third party web page by using the embedded functionality that can be activated by the icon on the lower right corner. See an example of Embedding for the **services close to DISIT Lab from this link.**

The call produced by ServiceMap and received by email will be different according to the icon you have selected, requesting: Services, Weather panel, selection panel and on the general web page. You can copy paste these links into your application shortening the programming time, since the REST call or query are visually programmed. You can avoid learning SPARQL query language. You can develop applications that can contain queries that can be directly updated by you on ServiceMap without redeploying the application on the mobile market. The provided mobile app source code can be used on iOS, Android, Windows Phone, BlackBerry, etc. You can access to the Sii-Mobility Km4City based RDF Store model via the browser **http://log.disit.org**.

APIs have been realized in order to cope with new possibilities and emerging needs of contextualize content, re-organizing information about services, keep real time data, and last but not least Re-use Open Data from Service Map to re-contextualize them in Mobile or Web Apps. These API are for those developers who want to exploit Open Data to create their own application.

The usage of the Smart City API is regulated by the so-called Affiliation Agreement. The Agreement has to be signed and give you access to more technical information and allows you to use the Smart City API, test and trial them without need of understanding fully the technical details and reinstalling the km4City platform in your premise.
The **Affiliation Agreement** is available in **ITALIAN** and ENGLISH languages. On the contrary, if you would like to use the tools starting from the Open Source Version, installing them and using them, you do not need to contact us. Just do it! We are happy to help you also in that case up to a reasonable amount of effort. Please note that you can access for free at our tutorial and training days.

In Section 2 there is a description of realization, semantic and uses of the REST APIs.

In alternative you can access directly to the **km4city RDF store** by using this link and interface for developers (see Section 3 for some details).

4 Smart City: API v1

This section provides a description of the API at version 1, an older version is available at http://servicemap.disit.org/WebAppGrafo/api/ that is used by old applications and it no more maintained please don't use them.
WARNING: These APIs are still under development and may change in the future (for bugs solving and improvements) however we will try to keep them backward compatible and introduce new parameters and new properties in the JSON objects but not change parameters names or property names. When it will not possible to be backward compatible we will switch to version 2.

4.1 Basics

The APIs are accessible mainly via HTTP GET requests at specific URLs with specific parameters provided in the query string. Query parameters are case sensitive (e.g. use maxDists and not maxdists). The "format" parameter in many cases can be equal to html or json (and json is assumed if it is not provided) to provide the result as machine readable JSON or as a human readable web page. Most APIs accept an optional user identifier (uid) that should be provided to identify the device (and indirectly the user) making the requests. The uid should be a unique identifier, currently the uid is generated as a SHA256 hash of the device uuid generated by cordova device plugin (see https://cordova.apache.org/docs/en/latest/reference/cordova-plugin-device/). The history of user requests is used to produce suggestions and user engagements.
The *multimedia* property provided by some APIs contain a URL to a multimedia file that is in many cases no more available, a caching service for images was setup because the images were too large for mobiles and now using this cache is currently the only way to retrieve these images. See the multimedia caching API to see how to use these images. Unfortunately the cache was realized only for images and thus pdf files and audio files are no more available.

The requests to the API are CORS enabled thus APIs can be used cross domain from other sites. Currently no API key or authentication is needed but this may change in the future.

The following table reports a list of the APIs currently available and reports:

- if the API call can be visually generated from the ServiceMap user interface using a Save button,
- if the API can be used to embed in a HTML iframe the results and
- where the API is currently available, if in the *Production* site (and also Test site) (http://servicemap.disit.org/WebAppGrafo) or only on the *Test* site (http://www.disit.org/ServiceMap).

Description	API	where	App	AppKit	Save on ServiceMap	embed
Service search near GPS position	Y	Prod	Y	Y	Y	Y
Service search near a service	Y	Prod	?	?	Y	Y
Service search within a GPS area	Y	Prod	N	N	Y	Y
Service search within a WKT described area	Y	Prod	N	N	N	Y
Service search within a stored WKT described area	Y	Prod	N	N	Y	Y
Service search by municipality	Y	Prod	N	N	Y	Y
Service search by query id	Y	Prod	N	N	Y	Y
Full text search	Y	Prod	Y	N	Y	Y
Address/POI search by text	N	Prod	Not yet	N	Not yet	N
Event search	Y	Prod	Y	Y	Y	Not yet
Address and geometry search by GPS	Y	Prod	Y	Y	Not yet	Not yet
Service info	Y	Prod	Y	Y	Y	Y
Generic Service	Y	Prod	Y	Y	Y	Y
Event	Y	Prod	Y	Y	Y	Y
Parking service	Y	Prod	Y	Y	Y	Y
Traffic sensor	Y	Prod	Y	Y	Y	Y
Weather Forecast	Y	Prod	Y	Y	Y	N
Bus Stop	Y	Prod	Y	Y	Y	Y
Fuel Station	Y	Prod	Y	N	Y	Y
First aid	Y	Prod	Y	N	Y	Y
Air quality monitoring station	Y	Prod	Y	N	Y	Y
Smart waste container	Y	Test	Not yet	N	Y	Y
Smart bench	Y	Test	Not yet	N	Y	Y
Smart irrigator	Y	Test	Not yet	N	Y	Y
Energy meter	Not yet	-	Not yet	N	Not yet	Not yet
Recharge station	Not yet	-	Not yet	N	Not yet	Not yet
Smart street light	Not yet	-	Not yet	N	Not yet	Not yet
(Bus) Agency list	Y	Prod	N	N	N	N
(Bus) Lines list	Y	Prod	N	N	N	N
(Bus) Routes list	Y	Prod	Y	Y	Not yet	Not yet
(Bus) Stop list	Y	Prod	Y	Y	Not yet	Not yet
Search (Bus) Routes in a geographic area	Y	Prod	N	N	Not yet	Not yet
Estimated Bus position	Y	Prod	N	N	Not yet	Y
Rating and comment API	Y	Prod	Y	Y	N	N
Service Photo API	Y	Prod	Y	Y	N	N
Last contributions API	Y	Prod	Y	N	N	N
Geographic Annotation API	Not yet	-	-	-	-	-
Recommender API	Y	Prod	Y	N	N	N
Shortest path finder API	Y	Prod	Not yet	Not yet	Not yet	Y
User information API	Not yet	-	-	-	-	-
Image caching API	Y	Prod	Y	Y	N	Y

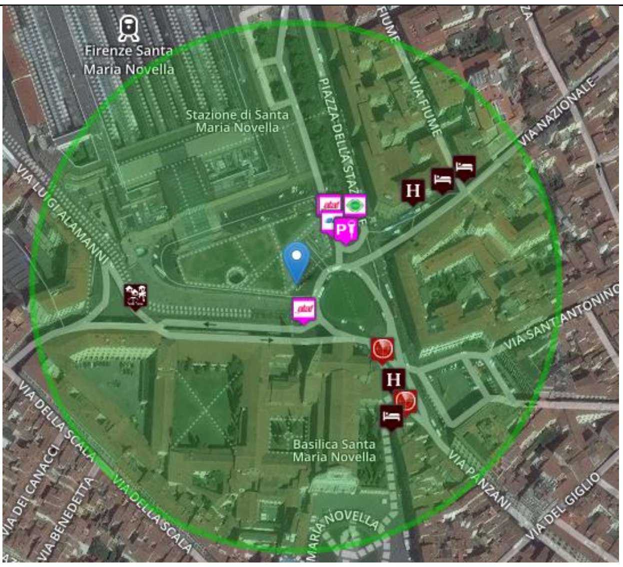
Note: For APIs supporting format “html” the following additional optional parameters may be used:

- *map*: to set the type of map to be used (“satellite”, “streets” or “grayscale”);

- *controls*: to control the appearance of the controls on the left and right of the page, it can be "hidden" or "false" to be not visible or "collapsed" do be collapsed;
- *info*: to control the appearance of the info tab on the lower left of the page, it can be "hidden" or "false" to be not visible or "collapsed" to be collapsed;

4.2 Service search near GPS position

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services that are near a given GPS position. The services can be filtered as belonging to specific categories (e.g. Accommodation, Hotel, Restaurant etc), or having specific words in any textual field. It can also be used to find services that have a WKT spatial description that contains a specific GPS position.	
Parameters:	
selection	<latitude>;<longitude> of the GPS position
categories	the list of categories of the services to be retrieved separated with semicolon, if omitted all kinds of services are returned. It can contain macro categories or categories, if a macro category is specified all categories in the macro category are used. The complete list of categories and macro categories can be retrieved on servicemap.disit.org.
text	words in this parameter are used to retrieve services that contain all these words in any textual description associated with the service.
maxDists	maximum distance from the GPS position of the services to be retrieved, expressed in Km (0.1 is used if parameter is missing) if it is equal to "inside" it searches for services with a WKT geometry that contains the specified GPS position (e.g a park)
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. "it", "en", "fr", "de", "es") to be used for returned descriptions if available in multiple languages. Currently for languages other than "it" and "en" it returns "en" descriptions. (if parameter is missing "en" is assumed)
geometry	true/false, if true it returns a "hasGeometry" property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing "false" is assumed)
uid	optional user identifier
format	html or json
Results:	
when format = "html" it produces a web page showing the results of the query, like the following:	



when format = “json” it returns the services split in three sections (BusStops , SensorSites, Services). Each section is provided as GeoJSON “FeatureCollection”, the results are sorted by distance, additionally in each section the “fullCount” property reports the full number of results available matching the query, for example:

```
{
    "BusStops": {
        "fullCount": 26,
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                    "type": "Point",
                    "coordinates": [11.249078, 43.775326]
            },
            "type": "Feature",
            "properties": {
                    "name": "Stazione Abside S.M.N.",
                    "typeLabel": "Fermata",
                    "tipo": "fermata",
                    "serviceType": "TransferServiceAndRenting_BusStop",
                    "busLines": "13 - 36 - 37",
                    "serviceUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Stop_FM0328_5",
                    "agency": "Ataf&Linea",
                    "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                    "photoThumbs": []
            },
            "id": 1
        }, … ]
    },
    "SensorSites": {
        "fullCount": 3,
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                    "type": "Point",
                    "coordinates": [11.24982, 43.77505]
            },
            "type": "Feature",
            "properties": {
                    "name": "FI055ZTL00101",
                    "tipo": "sensore",
                    "typeLabel": "Sensore",
                    "serviceType": "TransferServiceAndRenting_SensorSite",
                    "serviceUri": "http://www.disit.org/km4city/resource/FI055ZTL00101",
                    "photoThumbs": []
            },
            "id": 1
        }, … ]
    },
    "Services": {
          "fullCount": 84,
          "type": "FeatureCollection",
          "features": [{
               "geometry": {
                         "type": "Point",
                         "coordinates": [11.249473, 43.775867]
               },
               "type": "Feature",
               "properties": {
                         "name": "Parcheggio Stazione Firenze S.M.N.",
                         "tipo": "Parcheggio_auto",
                         "typeLabel": "Parcheggio auto",
                         "serviceType": "TransferServiceAndRenting_Car_park",
                         "serviceUri": "http://www.disit.org/km4city/resource/CarParkStazioneFirenzeS.M.N.",
                         "multimedia": ""
               },
               "id": 1
          }, … ]
     }
}
```

Examples:

- **Search for Accommodation, bus stop, sensor site or car park within 200m**
  http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.7756;11.2490&categories=Accommodation;BusStop;SensorSite;Car_park&maxResults=10&maxDists=0.2&lang=it&format=json
- **Any entertainment service within 200m**
  http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.7756;11.2490&categories=Entertainment&maxResults=20&maxDists=0.2&lang=it&format=json&geometry=true

```
{
  "Services": {
       "fullCount": 8,
       "type": "FeatureCollection",
       "features": [{
            "geometry": {
                 "type": "Point",
                 "coordinates": [11.24851, 43.77566]
            },
            "type": "Feature",
            "properties": {
                 "name": "Giardino di piazza della Stazione",
                 "tipo": "Aree_verdi",
                 "typeLabel": "Aree verdi",
                 "serviceType": "Entertainment_Green_areas",
                 "hasGeometry": true,
                 "serviceUri": "http://www.disit.org/km4city/resource/e62bc5f14bd412db00fcdcd6f9506857",
                 "multimedia": ""
            },
            "id": 1
       }, {
            "geometry": {
                 "type": "Point",
                 "coordinates": [11.249722, 43.77561]
            },
            "type": "Feature",
            "properties": {
                 "name": "Spartitraffico di piazza della Stazione",
                 "tipo": "Aree_verdi",
                 "typeLabel": "Aree verdi",
                 "serviceType": "Entertainment_Green_areas",
                 "hasGeometry": true,
                 "serviceUri": "http://www.disit.org/km4city/resource/37a2cdb39f7c8e86c55990b4f3125256",
                 "multimedia": ""
            },
            "id": 2
       }, {
            "geometry": {
                 "type": "Point",
                 "coordinates": [11.249624, 43.77658]
            },
                    "type": "Feature",
                    "properties": {
                        "name": "SCUDERIA DEL BEJ DI SIVORI GIOVAN BATTISTA E C. - S.A.S.",
                        "tipo": "Societa_sportive",
                        "typeLabel": "Societa' sportive",
                        "serviceType": "Entertainment_Sports_clubs",
                        "hasGeometry": false,
                        "serviceUri": "http://www.disit.org/km4city/resource/0a98b2ea221ba49356c20bed3c7b8f38",
                        "multimedia": ""
                    },
                    "id": 3
                }]
            }
        }
```

- **Any service whose geometry contains GPS position**
  http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.7754;11.2494&categories=Service&maxResults=20&maxDists=inside&lang=it&format=json&geometry=true

```
{
    "Services": {
        "fullCount": 6,
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                        "type": "Point",
                        "coordinates": [11.249722, 43.77561]
            },
            "type": "Feature",
            "properties": {
                        "name": "Spartitraffico di piazza della Stazione",
                        "tipo": "Aree_verdi",
                        "typeLabel": "Aree verdi",
                        "serviceType": "Entertainment_Green_areas",
                        "hasGeometry": true,
                        "serviceUri": "http://www.disit.org/km4city/resource/37a2cdb39f7c8e86c55990b4f3125256",
                        "multimedia": ""
            },
            "id": 1
        }]
    }
}
```

- **Accomodation within 1Km with “casa di dante” in a textual description**
  http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.7754;11.2494&categories=Accommodation&maxResults=2&maxDists=1&lang=it&format=json&text=casa%20di%20dante

```
{
    "Services": {
        "fullCount": 2,
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                        "type": "Point",
                        "coordinates": [11.256365, 43.771023]
            },
            "type": "Feature",
            "properties": {
                        "name": "CASA_DI_DANTE",
                        "tipo": "Affittacamere",
                        "typeLabel": "Affittacamere",
                        "serviceType": "Accommodation_Boarding_house",
                        "serviceUri": "http://www.disit.org/km4city/resource/c1cd4b12fabce2d9b3a1527fd5a7be79",
                        "multimedia": ""
            },
            "id": 1
        }, {
            "geometry": {
                        "type": "Point",
                        "coordinates": [11.256365, 43.771023]
            },
                    "type": "Feature",
                    "properties": {
                            "name": "CASA_DI_DANTE",
                            "tipo": "Affittacamere",
                            "typeLabel": "Affittacamere",
                            "serviceType": "Accommodation_Boarding_house",
                            "serviceUri": "http://www.disit.org/km4city/resource/8cb399e95b39475a9838eefa8ff5e683",
                            "multimedia": ""
                    },
                    "id": 2
                }]
            }
        }
```

Notes:

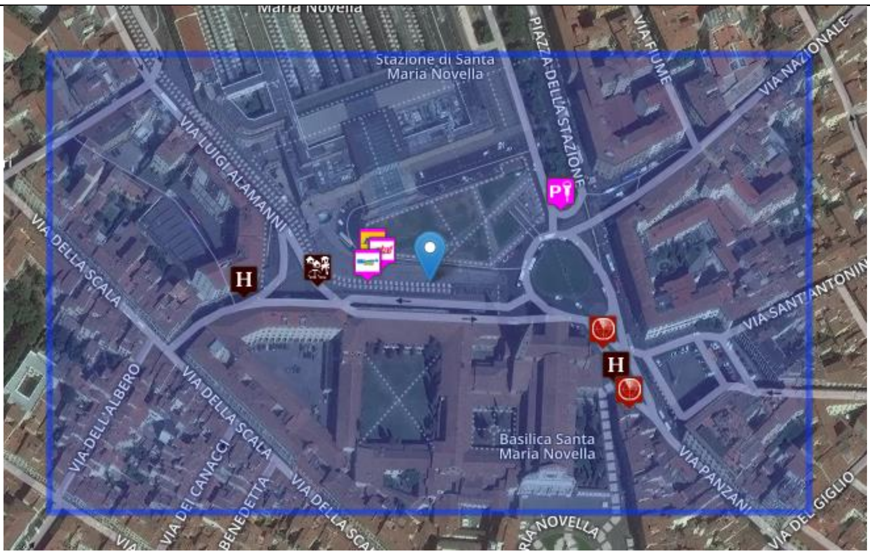
4.3 Service search near a service

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services that are near a given service identified by its serviceUri. The services can be filtered as belonging to specific categories (e.g. Accomodation, Hotel, Restaurant etc), or having specific words in any textual field. It can also be used to find services that have a WKT spatial description that contains a specific GPS position.	
Parameters:	
selection	serviceUri (http://...) of the service
categories	the list of categories of the services to be retrieved, if omitted all kinds of services are returned. It can contain macro categories or categories, if a macro category is specified all categories in the macro category are used. The complete list of categories and macro categories can be retrieved on servicemap.disit.org.
text	words in this parameter are used to retrieve services that contain all these words in any textual description associated with the service.
maxDists	maximum distance from the GPS position of the services to be retrieved, expressed in Km (0.1 is used if parameter is missing) if it is equal to “inside” it searches for services with a WKT geometry that contains the specified GPS position (e.g a park)
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
geometry	true/false, if true it returns a “hasGeometry” property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing “false” is assumed)
uid	optional user identifier
format	html or json
Results: The same format as Near a GPS position	
Examples: **Wine and food in 100m from Palazzo Vecchio** http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=http://www.disit.org/km4city/resource/7ad6d2d3be461b1f0514956279c00eab&categories=WineAndFood&maxResults=10&lang=it&format=json	

Notes:

4.4 Service search within a GPS area

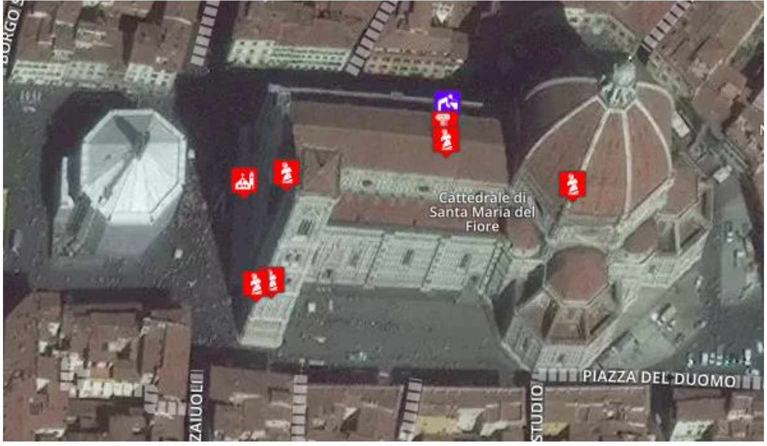
URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services that are inside a rectangular area. The services can be filtered as belonging to specific categories (e.g. Accomodation, Hotel, Restaurant etc), or having specific words in any textual field.	
Parameters:	
selection	<lat1>;<lng1>;<lat2>;<lng2> are two GPS coordinates describing a rectangle where (lat1,lng1) is a south west point and (lat2, lng2) is a north east point.
categories	the list of categories of the services to be retrieved, if omitted all kinds of services are returned. It can contain macro categories or categories, if a macro category is specified all categories in the macro category are used. The complete list of categories and macro categories can be retrieved on servicemap.disit.org.
text	words in this parameter are used to retrieve services that contain all these words in any textual description associated with the service.
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
geometry	true/false, if true it returns a “hasGeometry” property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing “false” is assumed)
uid	optional user identifier
format	html or json
Results:	
the results format is the same as the previous API, reresults are sorted by distance from the center of the rectangle	
Examples:	
• **Search for an accommodation, bus stop, sensor site or car park in a GPS area** http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=43.7741;11.2453;43.7768;11.2515&categories=Accommodation;BusStop;SensorSite;Car_park&maxResults=10&lang=it&format=json	

 Maria Novella Stazione di Santa Maria Novella PIAZZA DELLA STAZIONE VIA FIUME VIA NAZIONALE VIA LUIGI ALAMANNI VIA DELLA SCALA VIA SANT'ANTONINO VIA DELL'ALBERO VIA DEI CANACCI VIA DELLA SCALA BENEDETTA VIA DELLA SCALA Basilica Santa Maria Novella VIA PANZANI VIA DEL GIGLIO RIA NOVELLA
Notes:

4.5 Service search within a WKT described area

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services that are inside a geographic region described using the Well Known Text (WKT) format. The services can be filtered as belonging to specific categories (e.g. Accomodation, Hotel, Restaurant etc), or having specific words in any textual field.	
Parameters:	
selection	wkt:<WKT string> describes the geographic region as WKT string.
categories	the list of categories of the services to be retrieved, if omitted all kinds of services are returned. It can contain macro categories or categories, if a macro category is specified all categories in the macro category are used. The complete list of categories and macro categories can be retrieved on servicemap.disit.org.
text	words in this parameter are used to retrieve services that contain all these words in any textual description associated with the service.
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
geometry	true/false, if true it returns a “hasGeometry” property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing “false” is assumed)
uid	optional user identifier
format	html or json
Results:	
the results format is the same as the previous API, in this case the sort order of results is undefined.	
Examples:	
• to write a WKT string the following service can be used https://arthur-e.github.io/Wicket/sandbox-gmaps3.html	

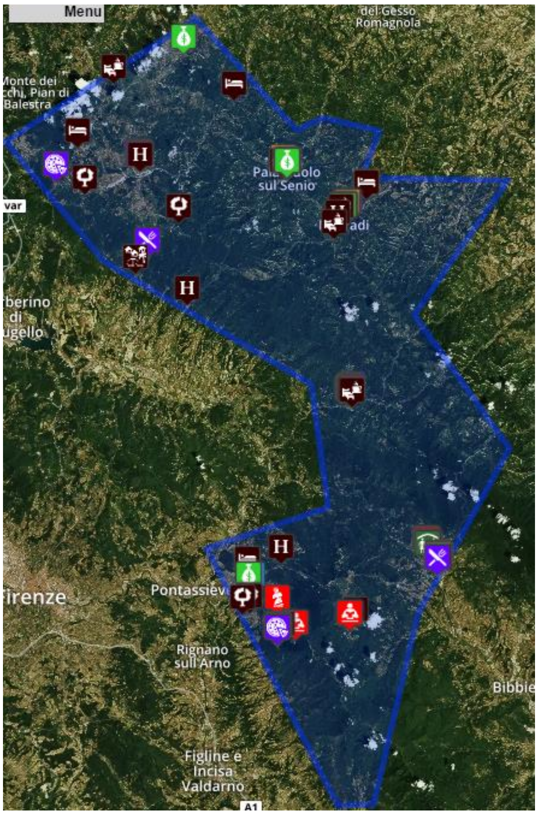
- **Search for any service in a WKT area**
  POLYGON((11.25539 43.77339,11.25608 43.77348,11.25706 43.77362,11.25759 43.77328,11.25755 43.77291,11.25675 43.77260,11.25536 43.77270,11.25539 43.77339))
  http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=wkt:POLYGON((11.25539%2043.77339,11.25608%2043.77348,11.25706%2043.77362,11.25759%2043.77328,11.25755%2043.77291,11.25675%2043.77260,11.25536%2043.77270,11.25539%2043.77339))&categories=Service&maxResults=0&lang=it&format=html



Notes:
the html version may not consider all the parameters

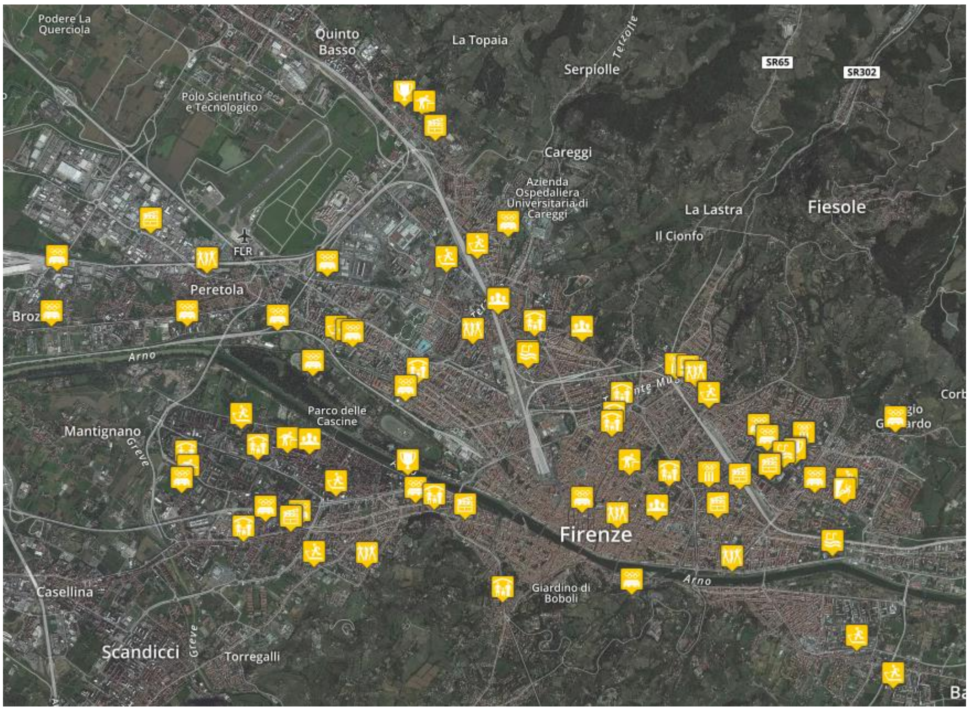
4.6 Service search within a stored WKT described area

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services that are inside a geographic region described using the Well Known Text (WKT) format, by referring to the WKT with and identifier provided when the WKT is stored. The services can be filtered as belonging to specific categories (e.g. Accomodation, Hotel, Restaurant etc), or having specific words in any textual field. The list of available geometries can be retrived from servicemap in the “Search Area” selection box (with Search Range “specific area”). New geometries can be provided using the http://www.km4city.org/wkt web service which can store a WKT from a shp file or providing directly the WKT string.	
Parameters:	
selection	*geo:<geo_id>* where <geo_id> identifies a WKT string stored on the server.
categories	the list of categories of the services to be retrieved, if omitted all kinds of services are returned. It can contain macro categories or categories, if a macro category is specified all categories in the macro category are used. The complete list of categories and macro categories can be retrieved on servicemap.disit.org.
text	words in this parameter are used to retrieve services that contain all these words in any textual description associated with the service.
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
geometry	true/false, if true it returns a “hasGeometry” property for each service stating if

	the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing “false” is assumed)
uid	optional user identifier
format	html or json
Results:	
the results format is the same as the previous API	
Examples:	
• **Search for any service in a WKT area** http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=geo:ritmi_01&categories=Service&maxResults=100&lang=it&format=html  Menu del Gesso Romagnola Monte dei chi, Pian di Balestra Pa olo sul Senio adi var berino di ugello irenze Pontassiev Rignano sull'Arno Bibbie Figline e Incisa Valdarno A1	
Notes: the html version may not consider all the parameters	

4.7 Service search by municipality

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services that are in a specific municipality. The services can be filtered as belonging to specific categories (e.g. Accomodation, Hotel, Restaurant etc), or having specific words in any textual field.	
Parameters:	
selection	name of the municipality like FIRENZE, EMPOLI, PISA possibly with prefix “COMUNE di “
categories	the list of categories of the services to be retrieved, if omitted all kinds of services are returned. It can contain macro categories or categories, if a macro category is specified all categories in the macro category are used. The complete list of categories and macro categories can be retrieved on servicemap.disit.org.

text	words in this parameter are used to retrieve services that contain all these words in any textual description associated with the service.
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
geometry	true/false, if true it returns a “hasGeometry” property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing “false” is assumed)
uid	optional user identifier
format	html or json
Results:	
the results format is the same as the previous API	
Examples:	
• **Search for any Entertainment service in the municipality of FIRENZE** http://servicemap.disit.org/WebAppGrafo/api/v1/?selection=COMUNE%20di%20FIRENZE&categories=Entertainment&maxResults=100&lang=it&format=html 	
Notes: the html version accepts only a selection with prefix “COMUNE di ”	

4.8 Service search by query id

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the set of services associated with a query stored using the servicemap user interface.	
Parameters:	
queryId	identifier of the query stored on servicemap
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned

	descriptions if available in multiple languages. Currently for languages other than "it" and "en" it returns "en" descriptions. (if parameter is missing "en" is assumed)
geometry	true/false, if true it returns a "hasGeometry" property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing "false" is assumed)
uid	optional user identifier
format	html or json
Results:	
the results format is the same as the previous API	
Examples:	
Search for any BusStop or CulturalActivity service in 100m near Santa Maria del Fiore http://servicemap.disit.org/WebAppGrafo/api/v1/?queryId=e02db54355fea40808300473c3537ff&format=json&lang=it	
Notes:	

4.9 Full text search

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve the geolocated entities (not only services) that match with a list of keywords. The results can be possibly filtered to be within a specified distance from a GPS position, or within a rectangular area or inside a WKT geolocated area.	
Parameters:	
search	the keywords separated with spaces that have to match with any textual description associated with an entity.
selection	optional "<lat>;<lng>" with a GPS position or "<lat1>;<lng1>;<lat2>;<lng2>" for a rectangular area or "wkt:<WKT_string>" or "geo:<geoid>" for a geographic area described as Well Known Text (see other APIs for more details)
maxDists	optional maximum distance from the GPS position of the entities to be retrieved, expressed in Km
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
lang	ISO 2 chars language code (e.g. "it", "en", "fr", "de", "es") to be used for returned descriptions if available in multiple languages. Currently for languages other than "it" and "en" it returns "en" descriptions. (if parameter is missing "en" is assumed)
geometry	true/false, if true it returns a "hasGeometry" property for each service stating if the service has a complex WKT geometries (linestring, polygon) associated with it (if parameter is missing "false" is assumed)
uid	optional user identifier
format	html or json
Results:	
the results format is a GeoJSON "FeatureCollection" with the matching entities, additionally the "fullCount" property provides the full count of results available matching the query. For each "Feature" a minimal set of properties are provided.	
Examples:	
Search for any geolocated entity matching "via nave" http://servicemap.disit.org/WebAppGrafo/api/v1/?search=via%20nave&maxResults=10&lang=	

en&format=json

```
{
   "fullCount": 558,
   "type": "FeatureCollection",
   "features": [{
         "geometry": {
            "type": "Point",
            "coordinates": [11.315443, 43.756367]
         },
         "type": "Feature",
         "properties": {
            "serviceUri": "http://www.disit.org/km4city/resource/e96076db6e4e2b8b43fb660579eb4de8",
            "name": "PICCIOLI DANIELE",
            "tipo": "servizio",
            "photoThumbs": [],
            "multimedia": "",
            "civic": "",
            "serviceType": "CulturalActivity_Theatre",
            "typeLabel": "Theatre"
         },
         "id": 1
   }, … {
         "geometry": {
            "type": "Point",
            "coordinates": [10.898357, 43.729973]
         },
         "type": "Feature",
         "properties": {
            "serviceUri": "http://www.disit.org/km4city/resource/RT04801406596TO",
            "name": "VIA NAVE DI VITIANA",
            "tipo": "servizio",
            "photoThumbs": [],
            "multimedia": "",
            "civic": "1",
            "serviceType": "",
            "typeLabel": "Road"
         },
         "id": 8
   }, … ]
}
```

Notes:

the html version may not consider all the parameters

4.10 Event search

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/events/
it allows to retrieve the geolocated events in a given temporal range (day, week or month). The results can be possibly filtered to be within a specified distance from a GPS position, or within a rectangular area or inside a WKT described geographic area.	
Parameters:	
range	time range for the events to be retrieved, it can be ‘day’ for the events of the day of the request, ‘week’ for the events in the next 7 days or ‘month’ for the events in the next 30 days (if omitted ‘day’ is assumed).
selection	optional “<lat>;<lng>” with a GPS position or “<lat1>;<lng1>;<lat2>;<lng2>” for a rectangular area or “wkt:<WKT_string>” or “geo:<geoid>” for a geographic area described as Well Known Text (see other APIs for more details).
maxDists	optional maximum distance from the GPS position of the events to be retrieved, expressed in Km.
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
uid	optional user identifier
format	only json
Results:	

the results format is a GeoJSON “FeatureCollection” with the matching events. For each “Feature” a set of properties is provided.

Examples:

Search for events of today

http://servicemap.disit.org/WebAppGrafo/api/v1/events/?range=day&format=json

```
{
   "Event": {
         "type": "FeatureCollection",
         "features": [{
              "geometry": {
                  "type": "Point",
                  "coordinates": [11.251058, 43.769848]
              },
              "type": "Feature",
              "properties": {
                  "serviceUri": "http://www.disit.org/km4city/resource/Event_18794_973b96efaf3f99f1b70af19cda4e3bf4",
                  "name": "Tra arte e moda",
                  "tipo": "event",
                  "place": "MUSEO SALVATORE FERRAGAMO ",
                  "startDate": "2016-05-19",
                  "startTime": "10.00 -19.30; chiuso 1/1, 01/05, 15/08 e 25/12",
                  "endDate": "2017-04-07",
                  "freeEvent": "NO",
                  "address": "PIAZZA DI SANTA TRINITA",
                  "civic": "2",
                  "categoryIT": "Mostre",
                  "price": "6 (incluso museo/including museum)",
                  "phone": "055 3562466",
                  "descriptionIT": "La mostra riflette il complesso rapporto fra arte e moda prendendo spunto dalla storia di Salvatore
Ferragamo che si ispirò alle avanguardie artistiche del '900 per realizzare le sue creazioni. ",
                  "website": "www.ferragamomuseo.com/museo",
                  "serviceType": "Event"
              },
              "id": 1
         }, ... ]
}}
```

Notes:

problems with duplicated events and with accented chars (solved for new events, still present for old events).

4.11 Address and geometry search by GPS

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/location

it allows to retrieve the complete address (municipality, street and civic number) given the GPS position. It may also provide a list of services or public transport lines intersecting with the provided GPS position.

Parameters:

position	“<lat>;<lng>” with a GPS position.
intersectGeom	true or false (assumed false if missing), if true it reports all the services and public transportation lines that have a geometry intersecting with the provided GPS position.
uid	optional user identifier
format	only json

Results:

A JSON object with properties:

- *address*: the street name.
- *number*: the civic number.
- *addressUri*: the URI identifying the civic number in the road graph.
- *municipality*: the estimated municipality (it may not work properly on the municipalities

borders)

- *municipalityUri*: the URI identifying the municipality in the road graph.
- *intersect*: array of objects with properties:
  - ○ *name*: name of the intersecting service or public transport line.
  - ○ *uri*: URI of the intersecting service or public transport line.
  - ○ *class*: URI representing the class
  - ○ *type*: type of geometry intersecting the GPS position, can be *lineString* or *Polygon*
  - ○ *routeType:* type of route can be Bus, LightRail, Ferry, Train
  - ○ *agency*: name of the agency providing the service
  - ○ *direction*: direction of the line
  - ○ *distance*: distance of the GPS position with the intersecting geometry

note: address, number and addressUri may be not present if the GPS position is outside a populated place.

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/location/?position=43.7741;11.2505&format=json

```
{
    "address": "VIA PANZANI",
    "municipality": "FIRENZE",
    "number": "17\/A",
    "addressUri": "http:\/\/www.disit.org\/km4city\/resource\/RT048017023351CV",
    "municipalityUri": "http:\/\/www.disit.org\/km4city\/resource\/048017"
}
```

http://servicemap.disit.org/WebAppGrafo/api/v1/location/?position=43.7741;11.2505&intersectGeom=true&format=json

```
{
    "address": "VIA PANZANI",
    "municipality": "FIRENZE",
    "number": "17\/A",
    "addressUri": "http:\/\/www.disit.org\/km4city\/resource\/RT048017023351CV",
    "municipalityUri": "http:\/\/www.disit.org\/km4city\/resource\/048017",
    "intersect": [{
        "distance": 1.2392468323025842E-4,
        "name": "Firenze Card",
        "class": "http:\/\/www.disit.org\/km4city\/schema#Tourist_trail",
        "type": "LineString",
        "uri": "http:\/\/www.disit.org\/km4city\/resource\/2a93692aa1eb7d680d9b4e0da668b408"
    }, {
        "distance": 3.1448272583131523E-4,
        "routeType": "Bus",
        "direction": "Salviatino",
        "name": "11",
        "agency": "Ataf&Linea",
        "class": "http:\/\/vocab.gtfs.org\/terms#Route",
        "type": "LineString",
        "uri": "http:\/\/www.disit.org\/km4city\/resource\/Bus_ataflinea_Trip_1923_3337883"
    }, ... ]
}
```

Notes:

4.12 Address/POI search by text

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/location

it allows to retrieve a list of street addresses and service names based on a text search. The search may be filtered excluding POIs and to be within a maximum distance from a GPS position.

Parameters:

search	a text with the words to be found in the names of the streets, civic number, municipality names and service names

searchMode	optional can be AND or ANDOR (default ANDOR), indicates if all or any word of the query need to match
position	optional "<lat>;<lng>" with a GPS position.
maxDists	optional maximum distance in km from *position* for searching the text (if omitted 5 km is assumed)
excludePOI	optional true or false (assumed false if missing), if true the search is performed only on street names, civic numbers and municipalities
maxResults	optional maximum number of results provided (default 10)
uid	optional user identifier
format	optional format of results, only json

Results:

A GeoJSON FeatureCollection object with the matching objects

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/location/?search=via%20calzaioli&format=json

```
{
     "type": "FeatureCollection",
     "count": 1263003,
     "features": [{
          "geometry": {
               "type": "Point",
               "coordinates": [11.255358, 43.77244]
          },
          "type": "Feature",
          "properties": {
               "serviceUri": "http://www.disit.org/km4city/resource/817ccce02f5aa0ef8c34744c4c25bcc6",
               "serviceType": "CulturalActivity Monument_location",
               "name": "Via dei Calzaiuoli",
               "city": "FIRENZE",
               "id": 1
          }
     }, {
          "geometry": {
               "type": "Point",
               "coordinates": [11.309124, 43.835896]
          },
          "type": "Feature",
          "properties": {
               "serviceUri": "http://www.disit.org/km4city/resource/f9fbe7453f63d3063cd4e33c43c1f5eb",
               "serviceType": "CulturalActivity Printing_and_services",
               "name": "FUTURE GRAPHIC DI CAMAIOLI GIUSTI VERONICA",
               "city": "FIESOLE - CALDINE",
               "id": 2
          }
     }, … {
          "geometry": {
               "type": "Point",
               "coordinates": [11.255217, 43.77035]
          },
          "type": "Feature",
          "properties": {
               "serviceUri": "http://www.disit.org/km4city/resource/RT048017002601CV",
               "serviceType": "StreetNumber",
               "address": "VIA DEI CALZAIOLI",
               "civic": "15 R",
               "city": "FIRENZE",
               "id": 5
          }
     }, …]
}
```

http://servicemap.disit.org/WebAppGrafo/api/v1/location/?search=via%20calzaioli&excludePOI=true&format=json

```
{
     "type": "FeatureCollection",
     "count": 1261873,
     "features": [{
          "geometry": {
               "type": "Point",
            "coordinates": [11.255217, 43.77035]
        },
        "type": "Feature",
        "properties": {
            "serviceUri": "http://www.disit.org/km4city/resource/RT048017002601CV",
            "serviceType": "StreetNumber",
            "address": "VIA DEI CALZAIOLI",
            "civic": "15 R",
            "city": "FIRENZE",
            "id": 1
        }
    }, …]
}
```

Notes:
Next version will consider also TPL stop names and municipalities

4.13 Service info

The Service info API allows getting information about a specific service or entity identified by a serviceURI property returned from the search APIs. Information can be get using the following REST API but also using the Linked Data paradigm using the serviceURI itself.

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/
it allows to retrieve information about a service using its serviceUri. It can return an html representation (format=”html”) or a machine readable representation (format=”json”)	
Parameters:	
serviceUri	the serviceUri of the service
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
realtime	true or false (if omitted true is implied) indicates if the last value of the time varying properties should be provided in the result or not.
uid	optional user identifier
format	html or json
Results:	

if format is “html” provides a visual representation of the service on the map. If format is json the API provides a GeoJSON description of the service with the main properties (name, address, city, type, etc.) and possibly some time varying properties for some kinds of services (traffic sensors, car park sensors, etc.).
The following is an example for a SensorSite (traffic sensor)

```
{
    "Sensor": {
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                "type": "Point",
                "coordinates": [11.2702, 43.77467]
            },
            "type": "Feature",
            "properties": {
                "name": "FI055ZTL02001",
                "typeLabel": "Sensor",
                "serviceType": "TransferServiceAndRenting_SensorSite",
                "serviceUri": "http://www.disit.org/km4city/resource/FI055ZTL02001",
                "municipality": "FIRENZE",
                "address": "VIA DELLA MATTONAIA",
                "photos": [],
                "photoThumbs": [],
                "photoOrigs": [],
                "avgStars": 0.0,
                "starsCount": 0,
                "comments": []
                },
                "id": 1
            }]
        },
        "realtime": {
            "head": {
                "sensor": ["FI055ZTL02001"],
                "vars": ["avgDistance", "avgTime", "occupancy", "concentration", "vehicleFlow", "averageSpeed", "thresholdPerc", "speedPercentile",
"instantTime"]
            },
            "results": {
                "bindings": [{
                    "avgDistance": {
                        "type": "literal",
                        "value": "Not Available"
                    },
                    "avgTime": {
                        "type": "literal",
                        "value": "Not Available"
                    },
                    "occupancy": {
                        "type": "literal",
                        "value": "Not Available"
                    },
                    "concentration": {
                        "type": "literal",
                        "value": "0.0"
                    },
                    "vehicleFlow": {
                        "type": "literal",
                        "value": "42.0"
                    },
                    "averageSpeed": {
                        "type": "literal",
                        "value": "0.0"
                    },
                    "thresholdPerc": {
                        "type": "literal",
                        "value": "Not Available"
                    },
                    "speedPercentile": {
                        "type": "literal",
                        "value": "Not Available"
                    },
                    "instantTime": {
                        "type": "literal",
                        "value": "2017-01-17T16:32:00+01:00"
                    }
                }]
            }
        }
}
```

Examples:

- see the following sections for details on the various kinds of services

Notes:

4.14 Generic service

For generic services (e.g. Accommodations, Restaurants, etc.) the following properties are provided in the GeoJSON properties:

- *serviceUri:* an URI identifying the service globally
- *name*: name of the service
- *typeLabel*: label associated with the type of service in the language provided with the lang parameter
- *serviceType:* a string containing “<MacroClass>_<ServiceType>”

- *city, address, civic*; municipality, address and civic number of the service
- *phone, fax, website, email:* phone, fax, website, email of the service
- *note:* notes associated with the service
- *description, description2:* two descriptions of the service, one in Italian and the other in English if available.
- *multimedia:* an url to a multimedia resource
- *linkDBpedia:* array of urls to dbpedia resources
- *photo, photoThumbs, photoOrigs:* array of urls to photos, thumbnails and original photos provided using the photo API.
- *wktGeometry:* a Well Known Text geometry associated with the service
- *avgStars:* average number of stars provided with the stars API
- *starsCount:* number of ratings provided by users.
- *comments:* array of comments on the service provided by users using the comments API

the following is an example:

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.361144, 44.00213]
                },
                "type": "Feature",
                "properties": {
                    "name": "IL_BRONCO",
                    "typeLabel": "Boarding house",
                    "serviceType": "Accommodation_Boarding_house",
                    "phone": "0558430207",
                    "fax": "",
                    "website": "www.ristoranteilbronco.it",
                    "province": "FI",
                    "city": "SCARPERIA",
                    "cap": "50038",
                    "email": "info@ristoranteilbronco.it",
                    "linkDBpedia": [],
                    "note": "",
                    "description": "",
                    "description2": "",
                    "multimedia": "",
                    "serviceUri": "http://www.disit.org/km4city/resource/9fc542b468509b922aeb833273dd40d0",
                    "address": "VIA DANTE",
                    "civic": "95",
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                },
                "id": 1
            }
        ]
    }
}
```

4.14.1 Event

http://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/Event_18794_973b96efaf3f99f1b70af19cda4e3bf4

```
{
    "Event": {
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                "type": "Point",
                "coordinates": [11.251058, 43.769848]
            },
            "type": "Feature",
            "properties": {
                "serviceUri": "http://www.disit.org/km4city/resource/Event_18794_973b96efaf3f99f1b70af19cda4e3bf4",
                "name": "Tra arte e moda",
                "name2": "Accross art and fashion",
                "website": "www.ferragamomuseo.com/museo",
                "address": "PIAZZA DI SANTA TRINITA",
                "number": "2",
                "province": "FI",
                "city": "Firenze",
                "note": "",
                "description": "La mostra riflette il complesso rapporto fra arte e moda prendendo spunto dalla storia di Salvatore Ferragamo che si ispirò alle avanguardie artistiche del '900 per realizzare le sue creazioni. ",
                "description2": "The exhibition reflects the complex relationship between art and fashion starting from the the story of Salvatore Ferragamo who realized his creations inspired by the avant-garde art of the ' 900.  ",
                "startDate": "2016-05-19T00:00:00+02:00",
                "startTime": "10.00 -19.30; chiuso 1/1, 01/05, 15/08 e 25/12",
                "endDate": "2017-04-07T00:00:00+02:00",
                "eventCategory": "Mostre",
                "eventCategory2": "Exhibitions",
                "photos": [],
                "photoThumbs": [],
                "photoOrigs": [],
                "avgStars": 0.0,
                "starsCount": 0,
                "comments": []
            },
            "id": 1
        }]
    }
}
```

4.14.2 Parking service

http://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/RT04801702315PO

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.24947, 43.77587]
                },
                "type": "Feature",
                "properties": {
                    "name": "Garage La Stazione Spa",
                    "typeLabel": "Car park",
                    "serviceType": "TransferServiceAndRenting_Car_park",
                    "phone": "055284784",
                    "fax": "",
                    "website": "",
                    "province": "FI",
                    "city": "FIRENZE",
                    "cap": "50123",
                    "email": "",
                    "linkDBpedia": [],
                    "note": "",
                    "description": "",
                    "description2": "",
                    "multimedia": "",
                    "serviceUri": "http://www.disit.org/km4city/resource/RT04801702315PO",
                    "address": "PIAZZA DELLA STAZIONE",
                    "civic": "3A",
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                },
                "id": 1
            }
        ]
    },
    "realtime": {
        "head": {
            "parkingArea": ["Garage La Stazione Spa"],
            "vars": ["capacity", "freeParkingLots", "occupiedParkingLots", "occupancy", "updating"]
        },
        "results": {
            "bindings": [{
                "capacity": {
                    "value": "617"
                },
                "freeParkingLots": {
                    "value": "322"
                },
                "occupiedParkingLots": {
                    "value": "579"
                },
                "occupancy": {
                    "value": "0.0"
                },
                "status": {
                    "value": "enoughSpacesAvailable"
                },
                "updating": {
                    "value": "2017-01-18T14:25:00+01:00"
                }
            }]
        }
    }
}
```

4.14.3 Traffic sensor

http://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/METRO487

```
{
    "Sensor": {
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                "type": "Point",
                "coordinates": [11.25003, 43.7747]
            },
            "type": "Feature",
            "properties": {
                "name": "METRO487",
                "typeLabel": "Sensor",
                "serviceType": "TransferServiceAndRenting_SensorSite",
                "serviceUri": "http://www.disit.org/km4city/resource/METRO487",
                "municipality": "FIRENZE",
                "address": "ZTL02 - Preferenziale P.zza Unità-Panzani",
                "photos": [],
                "photoThumbs": [],
                "photoOrigs": [],
                "avgStars": 0.0,
                "starsCount": 0,
                "comments": []
            },
            "id": 1
        }]
    },
    "realtime": {
        "head": {
            "sensor": ["METRO487"],
            "vars": ["avgDistance", "avgTime", "occupancy", "concentration", "vehicleFlow", "averageSpeed", "thresholdPerc", "speedPercentile",
"instantTime"]
        },
        "results": {
            "bindings": [{
                "avgDistance": {
                    "type": "literal",
                    "value": "Not Available"
                },
                "avgTime": {
                    "type": "literal",
                    "value": "2.49806"
                },
                "occupancy": {
                    "type": "literal",
                    "value": "Not Available"
                },
                "concentration": {
                    "type": "literal",
                    "value": "3.522905"
                },
                "vehicleFlow": {
                    "type": "literal",
                    "value": "330.0"
                },
                "averageSpeed": {
                    "type": "literal",
                    "value": "93.6727"
                },
                "thresholdPerc": {
                    "type": "literal",
                    "value": "Not Available"
                },
                "speedPercentile": {
                    "type": "literal",
                    "value": "Not Available"
                },
                "instantTime": {
                    "type": "literal",
                    "value": "2017-01-18T09:41:00+01:00"
                }
            }]
        }
    }
}
```

4.14.4 Weather Forecast

http://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/048017

```
{
    "head": {
        "location": "FIRENZE",
        "vars": ["day", "description", "minTemp", "maxTemp", "instantDateTime"]
    },
    "results": {
        "bindings": [{
            "day": {
                "type": "literal",
                "value": "Mercoledi"
            },
            "description": {
                "type": "literal",
                "value": "nuvoloso"
            },
            "minTemp": {
                "type": "literal",
                "value": "4"
            },
            "maxTemp": {
                "type": "literal",
                "value": "6"
            },
            "instantDateTime": {
                "type": "literal",
                "value": "2017-01-18T09:39:00+01:00"
            }
        }, {
            "day": {
                "type": "literal",
                "value": "Giovedi"
            },
            "description": {
                "type": "literal",
                "value": "coperto"
            },
            "minTemp": {
                "type": "literal",
                "value": "3"
            },
            "maxTemp": {
                "type": "literal",
                "value": "7"
            },
            "instantDateTime": {
                "type": "literal",
                "value": "2017-01-18T09:39:00+01:00"
            }
        }, {
            "day": {
                "type": "literal",
                "value": "Venerdi"
            },
            "description": {
                "type": "literal",
                "value": "poco nuvoloso"
            },
            "minTemp": {
                "type": "literal",
                "value": "1"
            },
            "maxTemp": {
                "type": "literal",
                "value": "7"
            },
            "instantDateTime": {
                "type": "literal",
                "value": "2017-01-18T09:39:00+01:00"
            }
        }, {
            "day": {
                "type": "literal",
                "value": "Sabato"
            },
            "description": {
                "type": "literal",
                "value": "poco nuvoloso"
            },
            "minTemp": {
                "type": "literal",
                "value": ""
            },
            "maxTemp": {
                "type": "literal",
                "value": ""
            },
            "instantDateTime": {
                "type": "literal",
                "value": "2017-01-18T09:39:00+01:00"
            }
        }, {
            "day": {
                "type": "literal",
                "value": "Domenica"
            },
            "description": {
                "type": "literal",
                "value": "nuvoloso"
            },
            "minTemp": {
                "type": "literal",
                "value": ""
            },
            "maxTemp": {
                "type": "literal",
                "value": ""
            },
            "instantDateTime": {
                "type": "literal",
                "value": "2017-01-18T09:39:00+01:00"
            }
        }]
    }
}
```

4.14.5 Bus stop

http://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/Bus_ataflinea_Stop_FM0022_5

```
{
    "BusStop": {
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                "type": "Point",
                "coordinates": [11.249069, 43.776485]
            },
            "type": "Feature",
            "properties": {
                "name": "Stazione Pensilina",
                "serviceUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Stop_FM0022_5",
                "typeLabel": "BusStop",
                "address": "",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "serviceType": "TransferServiceAndRenting_BusStop",
                "photos": [],
                "photoThumbs": [],
                "photoOrigs": [],
                "avgStars": 0.0,
                "starsCount": 0,
                "comments": []
            },
            "id": 1
        }]
    },
    "busLines": {
        "head": {
            "busStop": "Stazione Pensilina",
            "vars": ["busLine", "lineUri", "lineDesc"]
        },
        "results": {
            "bindings": [{
                "busLine": {
                    "type": "literal",
                    "value": "1"
                },
                "lineUri": {
                    "type": "literal",
                    "value": "http://www.disit.org/km4city/resource/Bus_ataflinea_Route_122797549"
                },
                "lineDesc": {
                    "type": "literal",
                    "value": "Lapo\/Boccaccio - S.Maria Novella Fs"
                }
            }, {
                "busLine": {
                    "type": "literal",
                    "value": "11"
                },
                "lineUri": {
                    "type": "literal",
                    "value": "http://www.disit.org/km4city/resource/Bus_ataflinea_Route_1073492795"
                },
                "lineDesc": {
                    "type": "literal",
                    "value": "Salviatino-Le Gore"
                }
            }, {
                "busLine": {
                    "type": "literal",
                    "value": "17"
                },
                "lineUri": {
                    "type": "literal",
                    "value": "http://www.disit.org/km4city/resource/Bus_ataflinea_Route_1208385503"
                },
                "lineDesc": {
                    "type": "literal",
                    "value": "Viale Verga-Via Boito\/Cascine"
                }
            }, …
            ]
        }
    },
    "timetable": {
        "head": {
            "vars": ["date", "arrivalTime", "lineName", "lineDesc", "routeName", "trip"]
        },
        "results": {
            "bindings": [{
                "date": {
                    "type": "literal", "value": "2017-01-18"
                },
                "arrivalTime": {
                    "type": "literal", "value": "14:52:00"
                },
                "departureTime": {
                    "type": "literal", "value": "14:52:00"
                },
                "lineName": {
                    "type": "literal", "value": "6"
                },
                "lineDesc": {
                    "type": "literal", "value": "Novelli-Smn-Torregalli"
                },
                "routeName": {
                    "type": "literal", "value": "Ospedale Torre Galli"
                },
                "trip": {
                    "type": "uri", "value": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3364525"
                }
            }, {
                "date": {
                    "type": "literal", "value": "2017-01-18"
                },
                "arrivalTime": {
                    "type": "literal", "value": "14:56:00"
                },
                "departureTime": {
                    "type": "literal", "value": "14:56:00"
                },
                "lineName": {
                    "type": "literal", "value": "11"
                },
                "lineDesc": {
                    "type": "literal", "value": "Salviatino-Le Gore"
                },
                "routeName": {
                    "type": "literal", "value": "La Gora"
                },
                "trip": {
                    "type": "uri", "value": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3344062"
                }
            }, …
            ]
        }
    },
    "realtime": {
    }
```

```
}
```

4.14.6 Fuel Station

http://www.disit.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/Fuel_station_01a234db6235dd55448a5044d9d26a52

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.279211, 43.78041]
                },
                "type": "Feature",
                "properties": {
                    "serviceUri": "http://www.disit.org/km4city/resource/Fuel_station_01a234db6235dd55448a5044d9d26a52",
                    "serviceType": "TransferServiceAndRenting_Fuel_station",
                    "name": "PINI E SETTESOLDI SNC",
                    "typeLabel": "Fuel station",
                    "phone": "",
                    "fax": "",
                    "website": "",
                    "province": "FI",
                    "city": "FIRENZE",
                    "cap": "50131",
                    "email": "",
                    "note": "",
                    "description": "",
                    "description2": "",
                    "multimedia": "",
                    "address": "VIALE DEI MILLE",
                    "civic": "",
                    "brand": "AgipEni",
                    "linkDBpedia": [],
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                }
            }
        ]
    },
    "realtime": {
        "head": {
            "vars": ["measuredTime", "fuel", "price", "currency", "self"]
        },
        "results": {
            "bindings": [{
                "measuredTime": {
                    "value": "2017-01-13 16:01:52"
                },
                "fuel": {
                    "value": "Benzina"
                },
                "price": {
                    "value": "1.579"
                },
                "currency": {
                    "value": "EUR"
                },
                "self": {
                    "value": "true"
                }
            }, {
                "measuredTime": {
                    "value": "2017-01-13 16:01:52"
                },
                "fuel": {
                        "value": "Blue Diesel"
                    },
                    "price": {
                        "value": "1.539"
                    },
                    "currency": {
                        "value": "EUR"
                    },
                    "self": {
                        "value": "true"
                    }
                }, {
                    "measuredTime": {
                        "value": "2017-01-13 16:01:52"
                    },
                    "fuel": {
                        "value": "Blue Super"
                    },
                    "price": {
                        "value": "1.729"
                    },
                    "currency": {
                        "value": "EUR"
                    },
                    "self": {
                        "value": "true"
                    }
                }, {
                    "measuredTime": {
                        "value": "2017-01-13 16:01:52"
                    },
                    "fuel": {
                        "value": "Gasolio"
                    },
                    "price": {
                        "value": "1.439"
                    },
                    "currency": {
                        "value": "EUR"
                    },
                    "self": {
                        "value": "true"
                    }
                }]
            }
        }
}
```

4.14.7 First aid (added with RESOLUTE project)

http://www.disit.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/dde440c760ef578da41599feb2396631

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.260015, 43.773457]
                },
                "type": "Feature",
                "properties": {
                    "name": "PRONTO SOCCORSO OSPEDALE SANTA MARIA NUOVA",
                    "typeLabel": "First aid",
                    "serviceType": "Emergency_First_aid",
                    "phone": "0552758844",
                    "fax": "0552758844",
                    "website": "",
                    "province": "FI",
                    "city": "FIRENZE",
                    "cap": "50100",
                    "email": "",
                    "linkDBpedia": [],
                    "note": "",
                    "description": "",
                    "description2": "",
                    "multimedia": "",
                    "serviceUri": "http://www.disit.org/km4city/resource/dde440c760ef578da41599feb2396631",
                    "address": "PIAZZA SANTA MARIA NUOVA",
                    "civic": "1",
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                },
                "id": 1
            }
        ]
    },
    "realtime": {
        "head": {
            "vars": ["measuredTime", "state", "redCode", "yellowCode", "greenCode", "blueCode", "whiteCode"]
        },
        "results": {
            "bindings": [{
                "measuredTime": {
                    "value": "2017/01/19T15:25:00.000"
                },
                "state": {
                    "value": "Con Destinazione"
                },
                "redCode": {
                    "value": "0"
                },
                "yellowCode": {
                    "value": "5"
                },
                "greenCode": {
                    "value": "5"
                },
                "blueCode": {
                    "value": "1"
                },
                "whiteCode": {
                    "value": "0"
                }
            }, {
                "measuredTime": {
                    "value": "2017/01/19T15:25:00.000"
                },
                "state": {
                    "value": "In Attesa"
                },
                "redCode": {
                    "value": "0"
                },
                "yellowCode": {
                    "value": "2"
                },
                "greenCode": {
                    "value": "5"
                },
                "blueCode": {
                    "value": "1"
                },
                "whiteCode": {
                    "value": "0"
                }
            }, {
                "measuredTime": {
                    "value": "2017/01/19T15:25:00.000"
                },
                "state": {
                    "value": "In Visita"
                },
                "redCode": {
                    "value": "0"
                },
                "yellowCode": {
                    "value": "4"
                },
                "greenCode": {
                    "value": "5"
                },
                "blueCode": {
                    "value": "1"
                },
                "whiteCode": {
                    "value": "0"
                }
            }, {
                "measuredTime": {
                    "value": "2017/01/19T15:25:00.000"
                },
                "state": {
                    "value": "Oss. Temporanea"
                },
                "redCode": {
                    "value": "0"
                },
                "yellowCode": {
                    "value": "1"
                },
                "greenCode": {
                    "value": "2"
                },
                "blueCode": {
                    "value": "1"
                },
                "whiteCode": {
                    "value": "0"
                }
            }, {
                "measuredTime": {
                    "value": "2017/01/19T15:25:00.000"
                },
                "state": {
                    "value": "Totali"
                },
                "redCode": {
                    "value": "0"
                },
                "yellowCode": {
                    "value": "12"
                },
                "greenCode": {
                    "value": "17"
                },
                "blueCode": {
                    "value": "4"
                },
                "whiteCode": {
                    "value": "0"
                }
            }]
        }
    }
}
```

4.14.8 Smart waste container (added with REPLICATE project)

http://www.disit.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/cassonetto01

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.2557, 43.7745]
                },
                "type": "Feature",
                "properties": {
                    "serviceUri": "http://www.disit.org/km4city/resource/cassonetto01",
                    "serviceType": "Environment_Smart_waste_container",
                    "name": "Cassonetto via martelli",
                    "typeLabel": "Smart waste comtainer",
                    "phone": "055232323",
                    "province": "FI",
                    "city": "Firenze",
                    "cap": "",
                    "address": "via martelli",
                    "civic": "2",
                    "wasteType": "http://www.disit.org/km4city/schema#anyWaste",
                    "capacity": "200",
                    "collectionTime": "alle 13:00 tutti I giorni",
                    "physicalShape": "campana",
                    "linkDBpedia": [],
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                }
            }
        ]
    },
    "realtime": {
        "head": {
            "vars": ["measuredTime", "wasteLevel", "batteryLevel"]
        },
        "results": {
            "bindings": [{
                "measuredTime": {
                    "value": "2017-01-19T15:46:31+01:00"
                },
                "wasteLevel": {
                    "value": "0.53592324"
                },
                "batteryLevel": {
                    "value": "261.33566"
                }
            }]
        }
    }
}
```

4.14.9 Smart bench (added with REPLICATE project)

http://www.disit.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/bench001

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.2554, 43.7737]
                },
                "type": "Feature",
                "properties": {
                    "serviceUri": "http://www.disit.org/km4city/resource/bench001",
                    "serviceType": "Entertainment_Smart_bench",
                    "name": "Panchina via martelli",
                    "typeLabel": "Smart bench",
                    "phone": "055232323",
                    "fax": "",
                    "website": "",
                    "province": "FI",
                    "city": "Firenze",
                    "cap": "",
                    "email": "",
                    "note": "",
                    "description": "",
                    "description2": "",
                    "multimedia": "",
                    "address": "via martelli",
                    "civic": "2",
                    "seats": "4",
                    "withWifi": "true",
                    "withUsb": "true",
                    "withAudio": "true",
                    "linkDBpedia": [],
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                }
            }
        ]
    },
    "realtime": {
        "head": {
            "vars": ["measuredTime", "temperature", "humidity", "pressure", "airQualityCO2", "light", "sittingsInRefPeriod", "totalSittings",
"passagesInRefPeriod", "totalPassages"]
        },
        "results": {
            "bindings": [{
                "measuredTime": {
                    "value": "2017-01-19T15:42:52+01:00"
                },
                "temperature": {
                    "value": "36.675144"
                },
                "humidity": {
                    "value": "83.60987"
                },
                "pressure": {
                    "value": "24.017311"
                },
                "airQualityCO2": {
                    "value": "85.21111"
                },
                "light": {
                    "value": "0.51458716"
                },
                "sittingsInRefPeriod": {
                    "value": "0"
                },
                "totalSittings": {
                    "value": "656"
                },
                "passagesInRefPeriod": {
                    "value": "3"
                },
                "totalPassages": {
                    "value": "3313"
                }
            }]
        }
    }
}
```

4.14.10 Smart irrigator (added with REPLICATE project)

http://www.disit.org/ServiceMap/api/v1/?serviceUri=http://www.disit.org/km4city/resource/irrigatore01

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.2496, 43.7736]
                },
                "type": "Feature",
                "properties": {
                    "serviceUri": "http://www.disit.org/km4city/resource/irrigatore01",
                    "serviceType": "Environment_Smart_irrigator",
                    "name": "Irrigatore p.zza S. Maria Novella",
                    "typeLabel": "Smart irrigator",
                    "phone": "0552556677",
                    "province": "FI",
                    "city": "Firenze",
                    "note": "",
                    "description": "",
                    "description2": "",
                    "address": "p.zza Santa Maria Novella",
                    "civic": "23",
                    "linkDBpedia": [],
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                }
            }
        ]
    },
    "realtime": {
        "head": {
            "vars": ["measuredTime", "currentlyActive", "temperature", "internalTemperature", "humidity", "soilWaterPotential", "leafWetness"]
        },
        "results": {
            "bindings": [{
                "measuredTime": {
                    "value": "2017-01-19T15:46:31+01:00"
                },
                "currentlyActive": {
                    "value": "true"
                },
                "temperature": {
                    "value": "14.397217"
                },
                "internalTemperature": {
                    "value": "26.770363"
                },
                "humidity": {
                    "value": "26.808607"
                },
                "soilWaterPotential": {
                    "value": "329.1715"
                },
                "leafWetness": {
                    "value": "23.110199"
                }
            }]
        }
    }
}
```

4.14.11 Air quality monitoring station

http://servicemap.disit.org/WebAppGrafo/api/v1/?serviceUri=http://www.disit.org/km4city/resource/ARPAT_QA_FI-GRAMSCI

```
{
    "Service": {
        "type": "FeatureCollection",
        "features": [
            {
                "geometry": {
                    "type": "Point",
                    "coordinates": [11.271168, 43.772068]
                },
                "type": "Feature",
                "properties": {
                    "name": "FI-GRAMSCI",
                    "typeLabel": "Air quality monitoring station",
                    "serviceType": "Environment_Air_quality_monitoring_station",
                    "phone": "",
                    "fax": "",
                    "website": "",
                    "province": "FIRENZE",
                    "city": "FIRENZE",
                    "cap": "",
                    "email": "",
                    "linkDBpedia": [],
                    "note": "",
                    "description": "",
                    "description2": "",
                    "multimedia": "",
                    "serviceUri": "http://www.disit.org/km4city/resource/ARPAT_QA_FI-GRAMSCI",
                    "address": "",
                    "civic": "",
                    "wktGeometry": "",
                    "photos": [],
                    "photoThumbs": [],
                    "photoOrigs": [],
                    "avgStars": 0.0,
                    "starsCount": 0,
                    "comments": []
                },
                "id": 1
            }
        ]
    },
    "realtime": {
        "head": {
            "vars": ["NO2", "SO2", "H2S", "CO", "Benzene", "PM2_5", "PM10", "annualPM10ExceedCount", "measuredTime"]
        },
        "results": {
            "bindings": [{
                "NO2": {
                    "value": "108.0"
                },
                "SO2": {
                    "value": ""
                },
                "H2S": {
                    "value": ""
                },
                "CO": {
                    "value": "0.8"
                },
                "Benzene": {
                    "value": "2.1"
                },
                "PM2_5": {
                    "value": "21.0"
                },
                "PM10": {
                    "value": "35.0"
                },
                "annualPM10ExceedCount": {
                    "value": "10"
                },
                "measuredTime": {
                    "value": "2017-04-02T00:00:00+02:00"
                }
            }]
        }
    }
```

}

4.14.12 Energy meter (added with REPLICATE project)

Under development

4.14.13 Recharge station (added with REPLICATE project)

Under development

4.14.14 Smart street light (added with REPLICATE project)

Under development

4.15 Public transport API

In the following the API that are related with public transports are reported.
Note: The information regarding timetable is acquired in GTFS format. Due to different names used in the previous version of the API that was only for buses, the names used in the API are not aligned with GTFS nomenclature in particular bus lines are mapped to GTFS routes and bus routes are mapped to GTFS trips. In the next version of the API names used may change to be aligned with GTFS.

4.15.1 Agency list

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/agencies
the API provide a list of the public transport agencies available	
Parameters:	
uid	optional user identifier
format	only json
Results:	
the API provides an array of JSON objects of the agencies available, for each agency is provided the agency name and the agency URI used to identify the agency in other APIs	
Examples:	

```
{
    "Agencies": [{
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_ataflinea_Agency_172",
        "name": "Ataf&Linea"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_acvbus_Agency_173",
        "name": "Autolinee Chianti Valdarno"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_amvbus_Agency_171",
        "name": "Autolinee Mugello Valdisieve"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_blubus_Agency_175",
        "name": "BluBus"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_cap_Agency_169",
        "name": "C.A.P. Consorzio Autolinee Pratesi"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_ctt_Agency_500",
        "name": "CTT NORD"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_cpt_Agency_176",
        "name": "Consorzio Pisano Trasporti"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_etruriamobilita_Agency_168",
        "name": "Etruria Mobilità"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Tram_gest_Agency_303",
        "name": "GEST S.p.A."
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_piubus_Agency_170",
        "name": "Piùbus"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_sienamobilita_Agency_167",
        "name": "Siena Mobilità"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Train_tft_Agency_196",
        "name": "T.F.T. S.p.A."
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_tiemme_Agency_400",
        "name": "TIEMME SPA"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Train_trenitalia_Agency_163",
        "name": "TRENITALIA S.p.A."
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Ferry_toremar_Agency_205",
        "name": "Toremar Toscana Regionale Marittima Spa"
    }, {
        "agency": "http:\/\/www.disit.org\/km4city\/resource\/Bus_vaibus_Agency_174",
        "name": "Vaibus"
    }]
```

Notes:

4.15.2 (Bus) Lines list

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-lines
the API provide a list of the public transport lines available for a given agency.	
Parameters:	
agency	URI of the agency whose lines are to be retrieved
uid	optional user identifier
format	only json

Results:

the API provides an array of JSON objects of the lines available, for each line is provided the line long and short name, the uri identifying the line.

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-lines/?agency=http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172

```
{
    "BusLines": [{
            "agency": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
            "shortName": "C1",
            "longName": "Parterre-Ponte Alle Grazie",
            "uri": "http:\/\/www.disit.org\/km4city\/resource\/Bus_ataflinea_Route_1380827827"
        }, {
            "agency": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
            "shortName": "S3",
            "longName": "Scuola Marconi-L'Olmo",
            "uri": "http:\/\/www.disit.org\/km4city\/resource\/Bus_ataflinea_Route_1858266107"
        }, … ]
}
```

Note:

The API can be used on any kind of public transport (Tram, Train, etc.) not only Bus.

4.15.3 (Bus) Routes list

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-routes

API provide a list of the public transport routes available for a given agency, line or passing by a specific stop.

Parameters:

agency	URI of the agency whose lines are to be retrieved
line	URI or shortName of a line (if URI is provided the agency is not needed)
busStopName	URI or name of a stop (if URI is provided the agency is not needed)
geometry	if true the WKT geometry of the route is returned (false is assumed if not provided)
uid	optional user identifier
format	only json

Results:

the API provides an array of JSON objects of the routes available, for each route is provided:

- *line*: line shot name
- *route*: the route URI
- *routeName*: optional route name
- *wktGeometry*: the WKT geometry of the route
- *firstBusStop*: name of the first bus stop
- *lastBusStop*: name of the last bus stop

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-routes/?agency=http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172&line=11&geometry=true

```
{
    "BusRoutes": [{
        "line": "11",
        "route": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3337883",
        "routeName": "",
        "wktGeometry": "LINESTRING(11.2172537345524 43.7326316393217, 11.2173853491045 43.7325390476232, …)",
        "firstBusStop": "La Gora",
        "lastBusStop": "Salviatino"
    }, {
        "line": "11",
        "route": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3338595",
        "routeName": "",
        "wktGeometry": "LINESTRING(11.2939833018846 43.7848045962375, 11.2939931338599 43.7848236867993,…)",
        "firstBusStop": "Salviatino",
        "lastBusStop": "La Gora"
    }]
}
```

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-routes/?agency=http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172&busStopName=Stazione%20Pensilina

```
{
    "BusRoutes": [{
        "line": "1",
        "route": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3337874",
        "routeName": "",
        "firstBusStop": "Boccaccio",
        "lastBusStop": "Stazione Palazzo Congressi"
    }, {
        "line": "2",
        "route": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3322861",
        "routeName": "",
        "firstBusStop": "Calenzano",
        "lastBusStop": "Stazione Palazzo Congressi"
    }, {
        "line": "4",
        "route": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3323029",
        "routeName": "",
        "firstBusStop": "Cappuccini",
        "lastBusStop": "Stazione Mercato Centrale"
    }, … ]
}
```

Note:

The API can be used on any kind of public transport (Tram, Train, etc.) not only Bus.

4.15.4 (Bus) Stop list

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-stops

API provide a list of the public transport stops available for a given route.

Parameters:

route	URI of the route whose bus stops are to be retrieved
geometry	if true the WKT geometry of the route is returned
uid	optional user identifier
format	only json

Results:

the API provides an JSON Object with line number (aka line short name) and line name (aka line long name) and a GeoJSON FeatureCollection with the stops. The stops are provided in stop order, from the first to the last.

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-stops/?route=http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3337883&geometry=true

```
{
    "Route": {
        "lineNumber": "11",
        "lineName": "Salviatino-Le Gore",
        "wktGeometry": "LINESTRING(11.2172537345524 43.7326316393217, 11.2173853491045 43.7325390476232, …)"
    },
    "BusStops": {
        "type": "FeatureCollection",
        "features": [{
            "geometry": {
                "type": "Point",
                "coordinates": [11.217254, 43.73263]
            },
            "type": "Feature",
            "properties": {
                "popupContent": "La Gora",
                "name": "La Gora",
                "serviceUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Stop_FM1208_5",
                "tipo": "fermata",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "serviceType": "TransferServiceAndRenting_BusStop"
            },
            "id": 1
        }, {
            "geometry": {
                "type": "Point",
                "coordinates": [11.220704, 43.73418]
            },
            "type": "Feature",
            "properties": {
                "popupContent": "Volterrana 02",
                "name": "Volterrana 02",
                "serviceUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Stop_FM1209_5",
                "tipo": "fermata",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "serviceType": "TransferServiceAndRenting_BusStop"
            },
            "id": 2
        }, … ]
    }
}
```

Note:
The API can be used on any kind of public transport (Tram, Train, etc.) not only Bus.

4.15.5 Search (Bus) Routes in a geographic area

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/

API provides a list of the public transport routes that have a stop in a specified area.

Parameters:

selection	optional "<lat>;<lng>" with a GPS position or "<lat1>;<lng1>;<lat2>;<lng2>" for a rectangular area or "wkt:<WKT_string>" or "geo:<geoid>" for a geographic area described as Well Known Text (see other APIs for more details)
maxDists	optional maximum distance from the GPS position of the entities to be retrieved, expressed in Km (0.1 is assumed if not present)
maxResults	maximum number of results to be returned (if parameter is missing 100 is assumed), if it is 0 all results are returned.
agency	optional URI of an agency to restrict the search to a specified agency
geometry	if true the WKT geometry of each route is returned (considered false if not provided)
uid	optional user identifier
format	only json

Results:

the API provides a JSON Object with all the routes that have stops on the specified area. For each route the following properties are provided:

- lineNumber: the line short name
- lineName: the line long name
- route: the route name
- routeUri: an URI identifying the route (it can be used to retrieve all the stops of the route)
- direction: with first and last stop
- agency: with agency name
- agencyUri: with agency URI
- polyline: with the WKT geometry of the route

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/?selection=43.7755;11.2495&maxDists=0.1&maxResults=5&geometry=true

```
{
    "PublicTransportLine": {
        "type": "FeatureCollection",
        "features": [{
            "type": "Feature",
            "properties": {
                "lineNumber": "12",
                "lineName": "",
                "route": "",
                "routeUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3137547",
                "direction": "Campo Marte Fs ➔ Stazione Parcheggio",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "polyline": "LINESTRING(11.2762059770919 43.7774442270155, 11.2761623454295 43.777427353435, …)",
                "serviceType": "PublicTransportLine"
            },
            "id": 1
        }, {
            "type": "Feature",
            "properties": {
                "lineNumber": "36",
                "lineName": "",
                "route": "",
                "routeUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3317289",
                "direction": "Cascine Del Riccio ➔ Stazione Abside S.M.N.",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "polyline": "LINESTRING(11.2551477298522 43.7339067055819, 11.2550069037315 43.7335043206344, …)",
                "serviceType": "PublicTransportLine"
            },
            "id": 2
        }, {
            "type": "Feature",
            "properties": {
                "lineNumber": "13",
                "lineName": "",
                "route": "",
                "routeUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3300218",
                "direction": "Il David ➔ Stazione Palazzo Congressi",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "polyline": "LINESTRING(11.2648824363224 43.7625434190618, 11.2648878248007 43.7625306663665, …)",
                "serviceType": "PublicTransportLine"
            },
            "id": 3
        }, {
            "type": "Feature",
            "properties": {
                "lineNumber": "11",
                "lineName": "",
                "route": "",
                "routeUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3337883",
                "direction": "La Gora ➔ Salviatino",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "polyline": "LINESTRING(11.2172537345524 43.7326316393217, 11.2173853491045 43.7325390476232, …)",
                "serviceType": "PublicTransportLine"
            },
            "id": 4
        }, {
            "type": "Feature",
            "properties": {
                "lineNumber": "C2",
                "lineName": "",
                "route": "",
                "routeUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Trip_1923_3365643",
                "direction": "Leopolda ➔ Piazza Beccaria",
                "agency": "Ataf&Linea",
                "agencyUri": "http://www.disit.org/km4city/resource/Bus_ataflinea_Agency_172",
                "polyline": "LINESTRING(11.2389601794313 43.7773069544217, 11.2389099511421 43.777364365556, …)",
                "serviceType": "PublicTransportLine"
            },
            "id": 5
        }]
    }
}
```

Note:

The API can be used on any kind of public transport (Tram, Train, etc.) not only Bus.

4.15.6 Estimated Bus position

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-position
API provides the estimated current position of buses	
Parameters:	

uid	optional user identifier
format	json or html

Results:

when format is html the API provides web visualization of the current bus positions while if format is json it provides a GeoJSON “FeatureCollection” with the data of each bus that is currently active. For each bus the following properties are provided:

- vehicleNum: the number of vehicle
- line: the line short name
- direction: with first and last stop
- detectionTime: the delay in minutes from the current time and the time the position was acquired.

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-position/?format=html

+
−
Torrente Mugnone
Torrente Mugnone
Arno
Arno
Firenze
Arno
Giardino di Boboli
Leaflet | Map data © OpenStreetMap contributors, CC-BY-SA, Imagery © Mapbox

http://servicemap.disit.org/WebAppGrafo/api/v1/tpl/bus-position/?format=json

```
{
    "type": "FeatureCollection",
    "features": [{
        "geometry": {
            "type": "Point",
            "coordinates": [11.340633, 43.735943]
        },
        "type": "Feature",
        "properties": {
            "vehicleNum": "3133579",
            "line": "24",
            "direction": "Sorgane Piazza Rodolico &#10132; Grassina",
            "tipo": "RealTimeInfo",
            "serviceUri": "busCode3133579",
            "detectionTime": "0",
            "serviceType": "bus_real_time"
        },
        "id": 1
    }, {
        "geometry": {
            "type": "Point",
            "coordinates": [11.272773, 43.774574]
        },
        "type": "Feature",
        "properties": {
            "vehicleNum": "3134531",
            "line": "12",
            "direction": "Piazzale Michelangelo &#10132; Stazione Parcheggio",
                "tipo": "RealTimeInfo",
                "serviceUri": "busCode3134531",
                "detectionTime": "0",
                "serviceType": "bus_real_time"
            },
            "id": 2
        }, {
            "geometry": {
                "type": "Point",
                "coordinates": [11.253791, 43.78007]
            },
            "type": "Feature",
            "properties": {
                "vehicleNum": "3137538",
                "line": "12",
                "direction": "Piazzale Michelangelo &#10132; Stazione Parcheggio",
                "tipo": "RealTimeInfo",
                "serviceUri": "busCode3137538",
                "detectionTime": "2",
                "serviceType": "bus_real_time"
            },
            "id": 3
        }, … ]
}
```

Note:
Currently it provides the position of ATAF&Linea buses based on the timetable.

4.16 Feedback API

These APIs are used from applications to provide some kind of feedback on services from real users like photos of the services, comments on the services, ratings of the services.

4.16.1 Rating and comment API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/feedback
API accepts a star rating (1-5) and/or a comment on a specific service. Comments are not automatically associated with the service, a moderator has to validate the comment provided.	
Parameters:	
serviceUri	URI identifying a service
stars	value 1 to 5 (if omitted no ratings is provided)
comment	comment provided by the user
lang	the language used in the comment
uid	a user identifier associated with the user providing the data
Results:	
the API fails using HTTP error code 404 if the serviceURI is not valid, stars or comment is not provided or user id is not provided.	
Examples:	
http://servicemap.disit.org/WebAppGrafo/api/v1/feedback?service=...&stars=2&comment=a%20comment&uid=...	
Notes:	

4.16.2 Service Photo API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/photo/
API accepts in POST as a multipart form the serviceUri, the user id and a photo in jpeg or png format. The photo provided is not automatically associated with the serviceUri a moderator will check it and decide.	
Parameters:	
serviceUri	URI identifying a service

uid	a user identifier associated with the user providing the data
file	a part named “file” with the photo to be uploaded, the part should contain the mimetype or the filename
Results:	
the API fails using HTTP error code 404 if the serviceURI is not valid, user id is not provided or a part named “file” is not present and the mimetype of this file cannot be found or if it’s not valid.	
Examples:	
NA	
Notes:	

4.16.3 Last contributions API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/feedback/last
API reports a list of the last photos, comments and starred services from the users.	
Parameters:	
uid	a user identifier
lang	ISO 2 chars language code (e.g. “it”, “en”, “fr”, “de”, “es”) to be used for returned descriptions if available in multiple languages. Currently for languages other than “it” and “en” it returns “en” descriptions. (if parameter is missing “en” is assumed)
format	only json
Results:	
the API reports a JSON object with informations on the last contributions..	
Examples:	

http://servicemap.disit.org/WebAppGrafo/api/v1/feedback/last

```
{
     "LastPhotos": [{
          "serviceUri": "http://www.disit.org/km4city/resource/af388d64a33b2624456a9a268ab01b54",
          "typeLabel": "Free WiFi point",
          "serviceType": "TourismService_Wifi",
          "long": "11.25355",
          "lat": "43.77682",
          "serviceName": "Firenze WIFI",
          "photo": "http://servicemap.disit.org/WebAppGrafo/api/v1/photo/file-5690474034488739316.jpg",
          "photoThumb": "http://servicemap.disit.org/WebAppGrafo/api/v1/photo/thumbs/file-5690474034488739316.jpg",
          "photoOrig": "http://servicemap.disit.org/WebAppGrafo/api/v1/photo/originals/file-5690474034488739316.jpg",
          "timestamp": "2017-01-22 16:38:20.0"
     }, ...],
     "LastComments": [{
          "serviceUri": "http://www.disit.org/km4city/resource/cd9fa722072d84aa47d5bc6a74932c46",
          "typeLabel": "Museum",
          "serviceType": "CulturalActivity_Museum",
          "long": "11.263607",
          "lat": "43.769848",
          "serviceName": "MUSEO DI CASA BUONARROTI",
          "comment": "Palazzo del seicento comprato da Michelangelo nel quale si trovano diverse sculture e disegni di Michelangelo",
          "timestamp": "2016-12-17 09:04:14.0"
     }, ...],
     "LastStars": [{
          "serviceUri": "http://www.disit.org/km4city/resource/20950a98d5fc0d1d69115d2b531b7793",
          "typeLabel": "Museum",
          "serviceType": "CulturalActivity_Museum",
          "long": "11.263603",
          "lat": "43.769836",
          "serviceName": "CASA_BUONARROTI",
          "stars": 5,
          "timestamp": "2016-12-17 10:14:37.0"
     }, ...]
}
```

Notes:

4.17 Annotation API

The annotation APIs allows to accociate with a geographic position or a serviceURI some private or public information of any kind.

4.17.1 Submit annotation API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/annotation/

API accepts in POST a JSON object with information on an annotation.
{
"uid": "...",
"id" : "<unique id identifier>,
"position" : "<lat>;<long>", # ma anche con un serviceUri
"type" : "<tipo definito dall'applicazione>",
"visibility" : "private| public", # private=visibile solo a utente in una app, public= visibile a tutti gli utenti che usano una app
["timestamp" : "2017-02-13T12:34:15",] #se omesso prende data ora sottomissione
"properties" : { ... un qualsiasi oggetto json, è l'applicazione che sa come interpretare i dati... }
}	

JSON object properties:	
uid	a user identifier associated with the user providing the data
id	unique identifier of the annotation, if already present the annotation is updated, we suggest to use your application name as a prefix when generating the id.
position	GPS position as “<lat>;<long>” or a serviceURI of the annotation
type	type of the annotation, expressed as names separted by dots, it implies a hierarchy “myapp.parking” or “myapp.sensor.traffic” to avoid name clashes the root type should identify the application providing the annotation.
visibility	should be *private* or *public*, private means that the annotation is visible only to the user identified by the uid and public it can be retrived by all users
timestamp	optional timestamp associated with the annotation, if omitted the current time is used.
properties	a JSON object with any properties as needed by the application
Results:	
the API fails using HTTP error code 400 if uid, position, type visibility are not correct	
Examples:	
NA	
Notes: Under developemnt	

4.17.2 Delete annotation API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/annotation/
API accepts in HTPP DELETE a request to delete an annotation given the uid and the annotation id. Only the creator can delete an annotation.	
Paramters:	
uid	a user identifier associated with the user providing the data
id	unique identifier of the annotation, if already present the annotation is updated
Results:	
the API fails using HTTP error code 400 if uid, or id are not valid or the user has no rights to delete the annotation	

Examples:
Notes: Under developemnt

4.17.3 Retrieve annotations API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/annotation/
API accepts in GET a request for a list of annotations, around a point, of a specific type or super type, with a specific visibility	
Paramters:	
uid	a user identifier (optional if requesting public annotations)
type	type or super type of the annotations to be retrieved, if the type ends with ".*" all the sub types are considered (e.g. myapp.*, myapp.sensor.*)
visibility	optional visibility of the annotations to be retrieved, it can be private, public or any (default any)
position	optional position expressed with <lat>;<long> or via serviceUri
maxDists	optional maximum distance in Km from the position specified (default 0.1)
maxResults	optional maimum number of results (default 100)
minTimestamp	optional filter on the minimum timestamp
order	optional order of results it can be timestamp_desc, timestamp_asc, distance_asc, distance_desc
Results:	
the API provides a GeoJSON FeatureCollection object with information of the annotations matching the requested data.	
Examples:	
http://servicemap.disit.org/WebAppGrafo/api/v1/annotation/?type=myapp.*&visibility=public	
Notes: Under developemnt	

4.18 Recommender API

URL	http://screcommender.km4city.org/SmartCityRecommender/
The API suggests a set of services near a user, the services are grouped depending on the user profile and it can suggest services on the basis of past user behavior (svd=true) or only by position.	
Parameters:	
action	have to be "recommend"
user	user identifier
profile	profile of the user (one of tourist, student, commuter, citizen, all)
language	user language (one of en, it, fr, es, de)
latitude	latitude in decimal format
longitude	longitude in decimal format
distance	the search range from GPS position in km
mode	optional, it can be "gps" or "manual", states if the position provided is real user position acquired by a device or manually identified from the user on a map.
version	optional, version of the application, if provided the API suggests also tweets from some channels, depending on user profile, on Twitter Vigilance (e.g. PAAlert, PAMeteoNews, PAProtCivile)
aroundme	optional, it can be true or false (default false), if true imply svd false

	and it provides recommendation of the nearest services regardless if they have been already suggested.
svd	optional, it can be true or false (default true), if true and *aroundme* is false or missing the API chooses the category of services to suggest on the basis of past user of behavior (searches made and viewed services) and it can suggest services that are quite far away from user position.
alreadyRecommended	optional, it can be true or false (default false), if false and *aroundme* is false or missing the API does not suggest services already suggested in the last seven days.

Results:

the API provides a JSON array of groups of suggested services (the groups names and types depend on the user profile), for each group are provided an array of suggestions with at most 3 services, the label of the group in the language provided (default English), a priority (used to order the groups) and a group identifier. For the tweets are provided some information as the message, the twitter user, the date, etc. The following is an example:

```
[{
      "suggestions": [ ... ],
      "label": "Things to do",
      "priority": 1,
      "group": "Things to do"
}, {
      "suggestions": [ ... ],
      "label": "Events",
      "priority": 2,
      "group": "Events"
}, {
      "suggestions": [ ... ],
      "label": "Wine and Food",
      "priority": 3,
      "group": "Wine and Food"
}, {
      "suggestions": [ ... ],
      "label": "Places Nearby",
      "priority": 4,
      "group": "Places Nearby"
}, {
      "suggestions": [ ... ],
      "label": "Services and Utilities",
      "priority": 5,
      "group": "Services and Utilities"
}, {
      "suggestions": [ ... ],
      "label": "Transfer Services",
      "priority": 6,
      "group": "Transfer Services"
}, {
      "suggestions": [...],
      "label": "Education",
      "priority": 8,
      "group": "Education"
}, {
      "suggestions": [ ... ],
      "label": "Bus",
      "priority": 9,
      "group": "Bus"
}, {
      "suggestions": [ ... ],
      "label": "Financial Services",
      "priority": 10,
      "group": "Financial Services"
}, {
      "suggestions": { ... },
      "label": "Weather",
      "priority": 11,
      "group": "Weather"
}, {
      "suggestions": [{
            "Tweet": {
                "hashtagsOnTwitter": "#Toscana",
                "geo_lat": "0.00000",
                "publicationTime": "2017-01-20 09:45:00",
                "twitterUser": "arpatoscana",
                "links": "https:\/\/t.co\/bpej68mTli http:\/\/bit.ly\/2ixn46s https:\/\/t.co\/u6maxPUtjz
https:\/\/twitter.com\/i\/web\/status\/822364292355739650",
                "message": "Come ha lavorato ARPAT nel 2016: il parere dei cittadini della #Toscana https:\/\/t.co\/bpej68mTli\u2026
https:\/\/t.co\/u6maxPUtjz",
                "lang": "it",
                "twitterId": "822364292355739650",
                "retweetCount": "0",
                "favoriteCount": "0",
                "geo_long": "0.00000"
            }
        }, …],
        "label": "Twitter Environment",
        "priority": 12,
        "group": "Twitter3"
}, {
        "suggestions": [ … ],
        "label": "Twitter News",
        "priority": 14,
        "group": "Twitter1"
}, {
        "suggestions": [ … ],
        "label": "Twitter Alert",
        "priority": 15,
        "group": "Twitter2"
}]
```

Examples:

http://screcommender.km4city.org/SmartCityRecommender/?action=recommend&user=3043b85d23d6f4879e1765c2c2e431cbc71d393065af06b03486ba4a04642b5b&profile=student&language=en&latitude=43.7727&longitude=11.2532&distance=1&version=1

Notes:

4.19 Shortest path finder API

Shortest path finder API

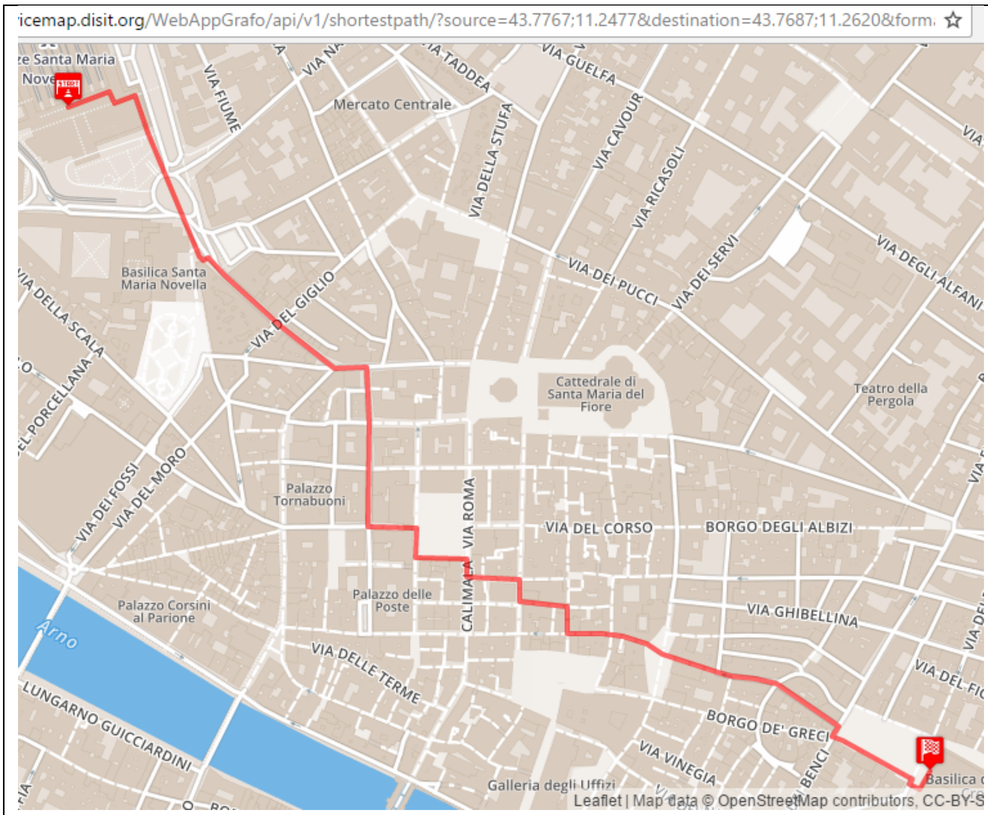
URL	http://servicemap.disit.org/WebAppGrafo/api/v1/shortestpath
This API allows to get a path from a source point to a destination point. The points can be specified as latitude;longitude coordinates or using the serviceUri of a service. The path is provided as WKT geometry and as a sequence of arcs between nodes (the service uses the OpenStreetMap road graph). The type of route can be specified as using public transport, feet, car or bike (using cycle paths whenever possible). The start datetime is used to select the options for public_transport and to evaluate the time needed to make the path.	
Parameters:	
source	“<lat>;<lng>” or service URI of the starting point
destination	“<lat>;<lng>” or service URI of the destination
routeType	can be “public_transport”, “foot_shortest”, “foot_quiet”, “car”, “bike_security” (foot_shortest is assumed if missing).
maxFeetKM	maximum distance in km by feet for routeType=public_transport (default 0.1)
startDatetime	datetime of start (current datetime if omitted) (e.g. “2017-01-13T12:34:00”)
format	json or html
uid	optional user identifier
Results:	
the API provides a JSON object with the path from the source to the destination.The following is an	

example:

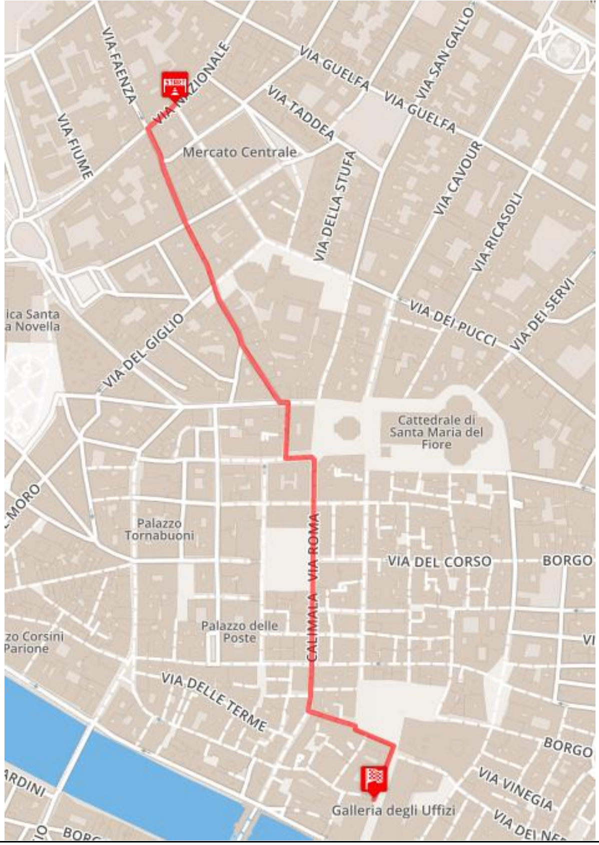
```
{
    "journey": {
        "source_node": {
            "node": "2531656503",
            "lon": 11.2477,
            "lat": 43.7767
        },
        "destination_node": {
            "node": "4006368396",
            "lon": 11.262,
            "lat": 43.7687
        },
        "search_route_type": "shortest_foot_optimization"
        "search_max_feet_km": 0.1,
        "start_datetime": "2017-01-12T12:34:00",
        "routes": [{
            "wkt": "LINESTRING(11.247773299999992 43.7765506,11.24788979999999 43.7765815,...)",
            "eta": "12:40:00",
            "time": "00:07:00",
            "distance": 1.1609571842375381,
            "arc": [{
                "distance": 0.009997643869982628,
                "start_datetime": "12:34:00",
                "end_datetime": "12:34:07",
                "destination_node": {
                    "lon": 11.24788979999999,
                    "lat": 43.7765815,
                    "node_id": "2531656509"
                },
                "source_node": {
                    "lon": 11.247773299999992,
                    "lat": 43.7765506,
                    "node_id": "2531656503"
                },
                "transport_provider": "",
                "transport": "foot",
                "transport_service_type": "",
                "desc": "nd"
            }, ... {
                "distance": 0.013927524955311556,
                "start_datetime": "12:38:05",
                "end_datetime": "12:38:15",
                "destination_node": {
                    "lon": 11.250239099999982,
                    "lat": 43.77465860000007,
                    "node_id": "271149487"
                },
                "source_node": {
                    "lon": 11.250108900000013,
                    "lat": 43.77474099999993,
                    "node_id": "1754184405"
                },
                "transport_provider": "",
                "transport": "foot",
                "transport_service_type": "",
                "desc": "Via Panzani"
            }, ... ]
    },
    "response": {
        "error_message": "successful",
        "current_operation": "route optimization",
        "error_code": "0"
    },
    "elapsed_ms": 3943,
    "message_version": "1.0"
}
```

Examples:

http://servicemap.disit.org/WebAppGrafo/api/v1/shortestpath/?source=43.7767;11.2477&destination=43.7687;11.2620&format=html


icemap.disit.org/WebAppGrafo/api/v1/shortestpath/?source=43.7767;11.2477&destination=43.7687;11.2620&form.

http://servicemap.disit.org/WebAppGrafo/api/v1/shortestpath/?source=43.7772;11.2522&destination=http://www.disit.org/km4city/resource/e76655ae0ae0a956df3a60500b2861dd&format=html



Notes:

This API is under development and currently supports only foot_shortest, foot_quiet and car

routes.

4.20 User information API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/userinfo
API reports a information about a user identified by a uid	
Parameters:	
uid	a user identifier
format	optional format of results (only json)
Results:	
the API reports a JSON object with informations about a user: kind of profile selected (tourist, citizen, student, commuter), home and work inferred positions, date of first use, comments provided, starred services and uploaded phots	
Examples:	

http://servicemap.disit.org/WebAppGrafo/api/v1/userinfo?uid=....
{
"uid": "abce123…",
"profile": "tourist",
"homePosition": "43.123;10.123",
"workPosition": "",
"firstUseDate": "2017-01-12 08:12:01",
"comments": […],
"stars": […],
"photos": […],
}	

Notes: under development	

4.21 User mobility information API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/userinfo/mobility
API reports mobility information about a user identified by a uid, the data is provided on a dates range, it provides an estimation of the types of vehicles used for mobility: foot, car, bus, train with the distance and time.It can optionally include an estimation of the car driving style in the date range provided. The driving style information include an evaluation from 1 to 5 (1 poor, 5 excellent) and the distance travelled evaluated in the urban area and in the extra urban area.	
Parameters:	
uid	a user identifier
fromDate	optional a date to start from (included) e.g. “2017-02-12”, today is assumed if omitted
toDate	optional a date to arrive to (included) e.g “2017-02-16”, today is assumed if omitted
includeDrivingStyle	optional true/false (assumed false if omitted) states if details on the driving style of the user should be provided
includeTransports	optional true/false (assumed true if omitted)
format	optional format of results (only json)
Results:	
the API reports a JSON object with informations on mobility of user for each day in the range of dates provided (some dates can be skipped if no data is available for a date) and	
Examples:	
http://servicemap.disit.org/WebAppGrafo/api/v1/userinfo/mobility?uid=....&includeDrivingStyle=tr	

ue

```
{
      "uid": "abce123…",
      "mobility": [ {
         "date": "2017-02-12",
         "transports" : [ {transport: "foot",  distance: 300, time: "00:15:00" },
             {transport: "car",  distance: 25000, time: "01:15:00" },
             {transport: "bus",  distance: 5000, time: "01:15:00" }],
         "drivingStyle" : { "urban" : { points: 4, distance: 11000}, "extraurban" : { points: 2, distance :14000 } }
      }]
}
```

Notes: under development

4.22 Image caching API

URL	http://servicemap.disit.org/WebAppGrafo/api/v1/imgcache
The API provides a cache for the given image url, it downloads the image, scales it to the thumbnail or medium size depending on the size requested and save it for future requests.	
Parameters:	
imageUrl	url to the image
size	the size of the image to be produced, it can be equal to “thumb”, “medium” or a number between 1 and 2000 pixels.
Results:	
It provides the scaled image produced in the same format as the original to be fit in a square of *size* x *size*, if the url is not an image it redirects to the original url.	
Examples:	
http://servicemap.disit.org/WebAppGrafo/api/v1/imgcache?imageUrl=http://www.florenceheritage.it/mobileApp/immagini/zocchi/148.jpg&size=thumb	
Notes:	

5 Linked data and SPARQL access

The data is currently available also using the standard W3C linked data protocol. It allows getting a machine readable representation of a resource like http://www.disit.org/km4city/resource/048017 as RDF/XML format. In the HTTP request protocol the header parameters Accept with “application/rdf+xml” should be specified while if the resource url is open in a web browser a (quite) human readable html version is generated.
Details on the RDF/XML format can be found at https://www.w3.org/TR/rdf-syntax-grammar/

Data can be also accessed using the standard W3C SPARQL 1.1 language and SPARQL query protocol at http://servicemap.disit.org/WebAppGrafo/sparql
Details on SPARQL 1.1 can be found at https://www.w3.org/TR/sparql11-overview/
At http://log.disit.org/sparql_query_frontend/ a query user interface can be found to play with SPARQL queries with some examples. Moreover the knowledge graph can be navigated using the Linked Open Graph viewer available at http://log.disit.org

6 Appendix

6.1 Macro classes

Macro class
Accommodation
Advertising
AgricultureAndLivestock
CivilAndEdilEngineering
CulturalActivity
EducationAndResearch
Emergency
Entertainment
Environment
FinancialService
GovernmentOffice
HealthCare
IndustryAndManufacturing
MiningAndQuarrying
ShoppingAndService
TourismService
TransferServiceAndRenting
UtilitiesAndSupply
Wholesale
WineAndFood

6.2 Service classes

Macro class	Service class
Accommodation	Agritourism
Accommodation	Beach_resort
Accommodation	Bed_and_breakfast
Accommodation	Boarding_house
Accommodation	Camping
Accommodation	Day_care_centre
Accommodation	Farm_house
Accommodation	Historic_residence
Accommodation	Holiday_village
Accommodation	Hostel
Accommodation	Hotel
Accommodation	Mountain_shelter
Accommodation	Other_accommodation
Accommodation	Religiuos_guest_house
Accommodation	Rest_home
Accommodation	Summer_camp
Accommodation	Summer_residence
Accommodation	Vacation_resort
Advertising	Advertising_and_promotion
Advertising	Market_polling
AgricultureAndLivestock	Animal_production
AgricultureAndLivestock	Crop_animal_production_hunting

AgricultureAndLivestock	Crop_production
AgricultureAndLivestock	Fishing_and_aquaculture
AgricultureAndLivestock	Hunting_trapping_and_services
AgricultureAndLivestock	Support_animal_production
AgricultureAndLivestock	Veterinary
CivilAndEdilEngineering	Architectural_consulting
CivilAndEdilEngineering	Building_construction
CivilAndEdilEngineering	Cartographers
CivilAndEdilEngineering	Civil_engineering
CivilAndEdilEngineering	Engineering_consulting
CivilAndEdilEngineering	Other_specialized_construction
CivilAndEdilEngineering	Specialized_construction
CivilAndEdilEngineering	Surveyor
CivilAndEdilEngineering	Technical_consultants
CulturalActivity	Archaeological_site
CulturalActivity	Auditorium
CulturalActivity	Botanical_and_zoological_gardens
CulturalActivity	Churches
CulturalActivity	Cultural_centre
CulturalActivity	Cultural_sites
CulturalActivity	Historical_buildings
CulturalActivity	Journalist
CulturalActivity	Leasing_of_intellectual_property
CulturalActivity	Library
CulturalActivity	Monument_location
CulturalActivity	Motion_picture_and_television_programme_activities
CulturalActivity	Museum
CulturalActivity	News_agency
CulturalActivity	Other_broadcasting
CulturalActivity	Photographic_activities
CulturalActivity	Printing_and_recorded_media
CulturalActivity	Printing_and_services
CulturalActivity	Publishing_activities
CulturalActivity	Radio_broadcasting
CulturalActivity	Reproduction_recorded_media
CulturalActivity	Sound_recording_and_music_publishing
CulturalActivity	Squares
CulturalActivity	Television_broadcasting
CulturalActivity	Theatre
CulturalActivity	Translation_and_interpreting
EducationAndResearch	Automobile_driving_and_flying_schools
EducationAndResearch	Conservatory
EducationAndResearch	Cultural_education
EducationAndResearch	Dance_schools
EducationAndResearch	Diving_school
EducationAndResearch	Educational_support_activities
EducationAndResearch	Higher_education
EducationAndResearch	Language_courses
EducationAndResearch	Performing_arts_schools
EducationAndResearch	Post_secondary_education

EducationAndResearch	Pre_primary_education
EducationAndResearch	Primary_education
EducationAndResearch	Private_high_school
EducationAndResearch	Private_infant_school
EducationAndResearch	Private_junior_high_school
EducationAndResearch	Private_junior_school
EducationAndResearch	Private_polytechnic_school
EducationAndResearch	Private_preschool
EducationAndResearch	Private_professional_institute
EducationAndResearch	Public_high_school
EducationAndResearch	Public_infant_school
EducationAndResearch	Public_junior_high_school
EducationAndResearch	Public_junior_school
EducationAndResearch	Public_polytechnic_school
EducationAndResearch	Public_professional_institute
EducationAndResearch	Public_university
EducationAndResearch	Research_and_development
EducationAndResearch	Sailing_school
EducationAndResearch	Secondary_education
EducationAndResearch	Ski_school
EducationAndResearch	Sports_and_recreation_education
EducationAndResearch	Training_school
EducationAndResearch	Training_school_for_teachers
Emergency	Carabinieri
Emergency	Civil_protection
Emergency	Coast_guard_harbormaster
Emergency	Commissariat_of_public_safety
Emergency	Corps_of_forest_rangers
Emergency	Emergency_medical_care
Emergency	Emergency_services
Emergency	Fire_brigade
Emergency	First_aid
Emergency	Italian_finance_police
Emergency	Local_police
Emergency	Towing_and_roadside_assistance
Emergency	Traffic_corps
Emergency	Useful_numbers
Entertainment	Amusement_activities
Entertainment	Amusement_and_theme_parks
Entertainment	Aquarium
Entertainment	Betting_shops
Entertainment	Boxoffice
Entertainment	Cinema
Entertainment	Climbing
Entertainment	Discotheque
Entertainment	Fishing_reserve
Entertainment	Gambling_and_betting
Entertainment	Game_reserve
Entertainment	Game_room
Entertainment	Gardens

Entertainment	Golf
Entertainment	Green_areas
Entertainment	Gym_fitness
Entertainment	Hippodrome
Entertainment	Operation_of_casinos
Entertainment	Pool
Entertainment	Rafting_kayak
Entertainment	Recreation_room
Entertainment	Riding_stables
Entertainment	Skiing_facility
Entertainment	Social_centre
Entertainment	Sports_clubs
Entertainment	Sports_facility
Entertainment	Sport_event_promoters
Environment	Building_and_industrial_cleaning_activities
Environment	Cleaning_activities
Environment	Disinfecting_and_exterminating_activities
Environment	Forestry
Environment	Geologists
Environment	Landscape_care
Environment	Materials_recovery
Environment	Photovoltaic_system
Environment	Sewerage
Environment	Street_sweeping
Environment	Waste_collection_and_treatment
Environment	Weather_sensor
FinancialService	Accountants
FinancialService	Auditing_activities
FinancialService	Bank
FinancialService	Financial_institute
FinancialService	Insurance
FinancialService	Insurance_and_financial
FinancialService	Labour_consultant
FinancialService	Legal_office
FinancialService	Tax_advice
GovernmentOffice	Airport_lost_property_office
GovernmentOffice	Civil_registry
GovernmentOffice	Consulate
GovernmentOffice	Department_of_motor_vehicles
GovernmentOffice	District
GovernmentOffice	Employment_exchange
GovernmentOffice	Income_revenue_authority
GovernmentOffice	Other_office
GovernmentOffice	Police_headquarters
GovernmentOffice	Postal_office
GovernmentOffice	Prefecture
GovernmentOffice	Social_security_service_office
GovernmentOffice	Train_lost_property_office
GovernmentOffice	Welfare_worker_office
GovernmentOffice	Youth_information_centre

HealthCare	Addiction_recovery_centre
HealthCare	Community_centre
HealthCare	Dentist
HealthCare	Doctor_office
HealthCare	Family_counselling
HealthCare	Group_practice
HealthCare	Haircare_centres
HealthCare	Healthcare_centre
HealthCare	Health_district
HealthCare	Health_reservations_centre
HealthCare	Human_health_activities
HealthCare	Local_health_authority
HealthCare	Medical_analysis_laboratories
HealthCare	Mental_health_centre
HealthCare	Paramedical_activities
HealthCare	Physical_therapy_centre
HealthCare	Poison_control_centre
HealthCare	Private_clinic
HealthCare	Psychologists
HealthCare	Public_hospital
HealthCare	Red_cross
HealthCare	Residential_care_activities
HealthCare	Senior_centre
HealthCare	Social_work
HealthCare	Youth_assistance
IndustryAndManufacturing	Animal_feeds_manufacture
IndustryAndManufacturing	Beverage_manufacture
IndustryAndManufacturing	Building_materials_manufacture
IndustryAndManufacturing	Coke_and_petroleum_derivatives
IndustryAndManufacturing	Computer_data_processing
IndustryAndManufacturing	Computer_programming_and_consultancy
IndustryAndManufacturing	Food_manufacture
IndustryAndManufacturing	Footwear_manufacture
IndustryAndManufacturing	Ict_service
IndustryAndManufacturing	Installation_of_industrial_machinery
IndustryAndManufacturing	Knitted_manufacture
IndustryAndManufacturing	Leather_manufacture
IndustryAndManufacturing	Machinery_repair_and_installation
IndustryAndManufacturing	Manufacture_of_basic_metals
IndustryAndManufacturing	Manufacture_of_chemicals_products
IndustryAndManufacturing	Manufacture_of_clay_and_ceramic
IndustryAndManufacturing	Manufacture_of_electrical_equipment
IndustryAndManufacturing	Manufacture_of_electronic_products
IndustryAndManufacturing	Manufacture_of_furniture
IndustryAndManufacturing	Manufacture_of_glass
IndustryAndManufacturing	Manufacture_of_jewellery_bijouterie
IndustryAndManufacturing	Manufacture_of_machinery_and_equipment
IndustryAndManufacturing	Manufacture_of_motor_vehicles
IndustryAndManufacturing	Manufacture_of_musical_instruments
IndustryAndManufacturing	Manufacture_of_non_metallic_mineral_products

IndustryAndManufacturing	Manufacture_of_paper
IndustryAndManufacturing	Manufacture_of_paper_products
IndustryAndManufacturing	Manufacture_of_pharmaceutical_products
IndustryAndManufacturing	Manufacture_of_plastics_products
IndustryAndManufacturing	Manufacture_of_refined_petroleum_products
IndustryAndManufacturing	Manufacture_of_refractory_products
IndustryAndManufacturing	Manufacture_of_rubber_and_plastics_products
IndustryAndManufacturing	Manufacture_of_rubber_products
IndustryAndManufacturing	Manufacture_of_sports_goods
IndustryAndManufacturing	Manufacture_of_structural_metal_products
IndustryAndManufacturing	Manufacture_of_textiles
IndustryAndManufacturing	Manufacture_of_toys_and_game
IndustryAndManufacturing	Manufacture_of_transport_equipment
IndustryAndManufacturing	Manufacture_of_travel_articles
IndustryAndManufacturing	Manufacture_of_wearing_apparel
IndustryAndManufacturing	Manufacture_of_wood
IndustryAndManufacturing	Manufacture_of_wood_products
IndustryAndManufacturing	Mining_support_services
IndustryAndManufacturing	Other_manufacturing
IndustryAndManufacturing	Quality_control_and_certification
IndustryAndManufacturing	Sawmilling
IndustryAndManufacturing	Software_publishing
IndustryAndManufacturing	Specialized_design
IndustryAndManufacturing	Stone_processing
IndustryAndManufacturing	Tannery
IndustryAndManufacturing	Technical_testing
IndustryAndManufacturing	Textile_manufacturing
IndustryAndManufacturing	Tobacco_industry
IndustryAndManufacturing	Web_and_internet_provider
MiningAndQuarrying	Extraction_of_salt
MiningAndQuarrying	Mining_of_metal_ores
MiningAndQuarrying	Other_mining_and_quarrying
MiningAndQuarrying	Petroleum_and_natural_gas_extraction
MiningAndQuarrying	Quarrying_of_stone_sand_and_clay
ShoppingAndService	Adult_clothing
ShoppingAndService	Antiques
ShoppingAndService	Artisan_shop
ShoppingAndService	Art_galleries
ShoppingAndService	Auctioning_houses
ShoppingAndService	Audio_and_video
ShoppingAndService	Beauty_centre
ShoppingAndService	Boat_equipment
ShoppingAndService	Bookshop
ShoppingAndService	Building_material
ShoppingAndService	Carpentry
ShoppingAndService	Carpets
ShoppingAndService	Carpets_and_curtains
ShoppingAndService	Car_washing
ShoppingAndService	Cleaning_materials
ShoppingAndService	Clothing

ShoppingAndService	Clothing_accessories
ShoppingAndService	Clothing_and_linen
ShoppingAndService	Clothing_children_and_infants
ShoppingAndService	Clothing_factory_outlet
ShoppingAndService	Coffee_rosters
ShoppingAndService	Computer_systems
ShoppingAndService	Computer_technician
ShoppingAndService	Cultural_and_recreation_goods
ShoppingAndService	Curtains_and_net_curtains
ShoppingAndService	Dairy_products
ShoppingAndService	Dating_service
ShoppingAndService	Diet_products
ShoppingAndService	Discount
ShoppingAndService	Door_to_door
ShoppingAndService	Estate_activities
ShoppingAndService	Fine_arts_articles
ShoppingAndService	Fish_and_seafood
ShoppingAndService	Flower_shop
ShoppingAndService	Food_and_tobacconist
ShoppingAndService	Footwear_and_accessories
ShoppingAndService	Footwear_and_leather_goods
ShoppingAndService	Footwear_factory_outlet
ShoppingAndService	Frozen_food
ShoppingAndService	Fruit_and_vegetables
ShoppingAndService	Funeral
ShoppingAndService	Funeral_and_cemetery_articles
ShoppingAndService	Fur_and_leather_clothing
ShoppingAndService	Games_and_toys
ShoppingAndService	Garden_and_agriculture
ShoppingAndService	Gifts_and_smoking_articles
ShoppingAndService	Haberdashery
ShoppingAndService	Hairdressing
ShoppingAndService	Hairdressing_and_beauty_treatment
ShoppingAndService	Hardware_electrical_plumbing_and_heating
ShoppingAndService	Hardware_paints_and_glass
ShoppingAndService	Herbalists_shop
ShoppingAndService	Household_appliances_shop
ShoppingAndService	Household_articles
ShoppingAndService	Household_fuel
ShoppingAndService	Household_furniture
ShoppingAndService	Household_products
ShoppingAndService	Household_utensils
ShoppingAndService	Hypermarket
ShoppingAndService	Industrial_laundries
ShoppingAndService	Jeweller
ShoppingAndService	Jewellery
ShoppingAndService	Laundries_and_dry_cleaners
ShoppingAndService	Lighting
ShoppingAndService	Maintenance_repair_of_motorcycles
ShoppingAndService	Maintenance_repair_of_motor_vehicles

ShoppingAndService	Manicure_and_pedicure
ShoppingAndService	Meat_and_poultry
ShoppingAndService	Mechanic_workshop
ShoppingAndService	Medical_and_orthopaedic_goods
ShoppingAndService	Minimarket
ShoppingAndService	Motorcycles_parts_wholesale_and_retail
ShoppingAndService	Motorcycles_wholesale_and_retail
ShoppingAndService	Motor_vehicles_wholesale_and_retail
ShoppingAndService	Musical_instruments_and_scores
ShoppingAndService	Music_and_video_recordings
ShoppingAndService	Newspapers_and_stationery
ShoppingAndService	Non_food_large_retailers
ShoppingAndService	Non_food_products
ShoppingAndService	Office_furniture
ShoppingAndService	Optics_and_photography
ShoppingAndService	Other_goods
ShoppingAndService	Other_retail_sale
ShoppingAndService	Parties_and_ceremonies
ShoppingAndService	Perfumery_and_cosmetic_articles
ShoppingAndService	Personal_service_activities
ShoppingAndService	Pet_care_services
ShoppingAndService	Pet_shop
ShoppingAndService	Pharmaceuticals
ShoppingAndService	Pharmacy
ShoppingAndService	Repair
ShoppingAndService	Repair_musical_instruments
ShoppingAndService	Repair_of_communication_equipment
ShoppingAndService	Repair_of_consumer_electronics
ShoppingAndService	Repair_of_footwear_and_leather_goods
ShoppingAndService	Repair_of_garden_equipment
ShoppingAndService	Repair_of_home_equipment
ShoppingAndService	Repair_of_household_appliances
ShoppingAndService	Repair_of_sporting_goods
ShoppingAndService	Restorers
ShoppingAndService	Retail_motor_vehicles_parts
ShoppingAndService	Retail_sale_non_specialized_stores
ShoppingAndService	Retail_trade
ShoppingAndService	Rope_cord_and_twine
ShoppingAndService	Sale_motor_vehicles_parts
ShoppingAndService	Sale_of_motorcycles
ShoppingAndService	Sale_of_motor_vehicles
ShoppingAndService	Sale_of_motor_vehicles_and_motorcycles
ShoppingAndService	Sale_via_mail_order_houses_or_via_internet
ShoppingAndService	Sanitary_equipment
ShoppingAndService	Second_hand_books
ShoppingAndService	Second_hand_goods
ShoppingAndService	Security_systems
ShoppingAndService	Sexy_shop
ShoppingAndService	Shopping_centre
ShoppingAndService	Single_brand_store

ShoppingAndService	Small_household_appliances
ShoppingAndService	Souvenirs_craftwork_and_religious_articles
ShoppingAndService	Sporting_equipment
ShoppingAndService	Stalls_and_markets
ShoppingAndService	Stalls_and_markets_of_clothing_and_footwear
ShoppingAndService	Stalls_and_markets_of_food
ShoppingAndService	Stalls_and_markets_other_goods
ShoppingAndService	Stamps_and_coins
ShoppingAndService	Supermarket
ShoppingAndService	Tattoo_and_piercing
ShoppingAndService	Telecommunications
ShoppingAndService	Textiles_products
ShoppingAndService	Tobacco_shop
ShoppingAndService	Travel_goods
ShoppingAndService	Trinkets
ShoppingAndService	Underwear_knitwear_and_shirts
ShoppingAndService	Upholsterer
ShoppingAndService	Vacating_service
ShoppingAndService	Vehicle_trade
ShoppingAndService	Vending_machines
ShoppingAndService	Wallpaper_and_floor_coverings
ShoppingAndService	Weapons_and_ammunition
ShoppingAndService	Wedding_favors
ShoppingAndService	Wellness_centre
TourismService	Beacon
TourismService	Camper_service
TourismService	Fresh_place
TourismService	Pedestrian_zone
TourismService	Ticket_sale
TourismService	Toilet
TourismService	Tourist_complaints_office
TourismService	Tourist_guides
TourismService	Tourist_information_office
TourismService	Tourist_trail
TourismService	Tour_operator
TourismService	Travel_agency
TourismService	Travel_bureau
TourismService	Travel_information
TourismService	Wifi
TransferServiceAndRenting	Airfields
TransferServiceAndRenting	Airplanes_rental
TransferServiceAndRenting	Bike_rack
TransferServiceAndRenting	Bike_rental
TransferServiceAndRenting	Boats_and_ships_rental
TransferServiceAndRenting	BusStop
TransferServiceAndRenting	Bus_tickets_retail
TransferServiceAndRenting	Cargo_handling
TransferServiceAndRenting	Car_park
TransferServiceAndRenting	Car_rental_with_driver
TransferServiceAndRenting	Charging_stations

TransferServiceAndRenting	Charter_airlines
TransferServiceAndRenting	Civil_airport
TransferServiceAndRenting	Controlled_parking_zone
TransferServiceAndRenting	Courier
TransferServiceAndRenting	Cycle_paths
TransferServiceAndRenting	Flight_companies
TransferServiceAndRenting	Freight_transport_and_furniture_removal
TransferServiceAndRenting	Fuel_station
TransferServiceAndRenting	Helipads
TransferServiceAndRenting	Land_transport
TransferServiceAndRenting	Land_transport_rental
TransferServiceAndRenting	Lifting_and_handling_equipment_rental
TransferServiceAndRenting	Logistics_activities
TransferServiceAndRenting	Passenger_air_transport
TransferServiceAndRenting	Postal_and_courier_activities
TransferServiceAndRenting	RTZgate
TransferServiceAndRenting	SensorSite
TransferServiceAndRenting	Support_activities_for_transportation
TransferServiceAndRenting	Taxi_company
TransferServiceAndRenting	Taxi_park
TransferServiceAndRenting	Train_station
TransferServiceAndRenting	Tramline
TransferServiceAndRenting	Tram_stops
TransferServiceAndRenting	Urban_bus
TransferServiceAndRenting	Vehicle_rental
TransferServiceAndRenting	Warehousing_and_storage
TransferServiceAndRenting	Water_transport
UtilitiesAndSupply	Accommodation_or_office_containers_rental
UtilitiesAndSupply	Agents
UtilitiesAndSupply	Associations
UtilitiesAndSupply	Business_support
UtilitiesAndSupply	Call_center
UtilitiesAndSupply	Combined_facilities_support_activities
UtilitiesAndSupply	Consulting_services
UtilitiesAndSupply	Credit_collection_agencies
UtilitiesAndSupply	Energy_supply
UtilitiesAndSupply	Equipment_for_events_and_shows_rental
UtilitiesAndSupply	Extraction_of_natural_gas
UtilitiesAndSupply	Internet_point_and_public_telephone
UtilitiesAndSupply	Internet_service_provider
UtilitiesAndSupply	Investigation_activities
UtilitiesAndSupply	Machinery_and_equipment_rental
UtilitiesAndSupply	Management_consultancy
UtilitiesAndSupply	Office_administrative_and_support_activities
UtilitiesAndSupply	Organization_of_conventions_and_trade_shows
UtilitiesAndSupply	Other_telecommunications_activities
UtilitiesAndSupply	Packaging_activities
UtilitiesAndSupply	Personal_and_household_goods_rental
UtilitiesAndSupply	Private_security
UtilitiesAndSupply	Recreational_and_sports_goods_rental

UtilitiesAndSupply	Recruitment
UtilitiesAndSupply	Reporting_agencies
UtilitiesAndSupply	Secretarial_support_services
UtilitiesAndSupply	Security_systems_service
UtilitiesAndSupply	Temp_agency
UtilitiesAndSupply	Video_tapes_disks_rental
UtilitiesAndSupply	Water_collection_treatment_and_supply
Wholesale	Non_specialized_wholesale_trade
Wholesale	Other_specialized_wholesale
Wholesale	Wholesale_agricultural_raw_materials_live_animals
Wholesale	Wholesale_commission_trade
Wholesale	Wholesale_food_beverages_tobacco
Wholesale	Wholesale_household_goods
Wholesale	Wholesale_ict_equipment
Wholesale	Wholesale_machinery_equipmentent_supplies
Wholesale	Wholesale_motor_vehicles_parts
Wholesale	Wholesale_trade
WineAndFood	Bakery
WineAndFood	Bar
WineAndFood	Canteens_and_food_service
WineAndFood	Catering
WineAndFood	Dining_hall
WineAndFood	Drinking_fountain
WineAndFood	Food_and_ice_cream_truck
WineAndFood	Food_trade
WineAndFood	Grill
WineAndFood	Highway_stop
WineAndFood	Ice_cream_parlour
WineAndFood	Literary_cafe
WineAndFood	Pastry_shop
WineAndFood	Pizzeria
WineAndFood	Restaurant
WineAndFood	Sandwich_shop_pub
WineAndFood	Small_shop
WineAndFood	Sushi_bar
WineAndFood	Take_away
WineAndFood	Trattoria
WineAndFood	Wine_shop_and_wine_bar

7 Acronimi

- API: Application Program Interface
- AVL: Automatic vehicle location
- AVM: Automatic Vehicle Monitoring
- BDaaS: Big Data as a Service
- CAP principle: Consistency Availability Partition Tolerance principle
- CBB: Content Based Billing
- CBB: Content Based Billing
- CEN: European Committee for Standardization
- DBMS: database management system
- FCD: Floating Cellular Data

- GPRS: General packet radio service
- GPS: Global positioning System
- GSM: Global System for Mobile
- ICT: Information and Communication Technologies
- ITS: Intelligent Transport Systems
- LCD: liquid-crystal display
- LOD: linked open data
- MC: Mobile Collector
- MMS: Multimedia Messaging Service
- NLP: Natural Language Processing
- NoSQL: no SQL database
- OD: open data
- OD: Open Data
- OGC: Open Geospatial Consortium
- OWL: Web Ontology Language
- PA: Pubblica Amministrazione
- PMI: Piccola e Media Impresa
- PMS: Private Mobile Systems
- POS: part-of-speech
- RDF: Resource Description Framework
- RFID: Radio Frequency IDentification o Identificazione a radio frequenza
- RTTI: Real-time Travel & Traffic Information
- SDI: Spatial Data Infrastructures
- SII: sistema di interoperabilità integrato
- SIMONE: progetto Simone
- SMS: Short Message Service
- SN: social networking, oppure sensor network
- SOA: Service Oriente Architecture
- SOAP: Simple Object Access Protocol
- SSAMM: Agenzia per la Mobilità Metropolitana strumenti di supporto, TOSCANA
- TPEG: Transport Protocol Experts Group
- TPL: gestore trasporto pubblico locale
- UML: Unified Modeling Language
- UMTS: Universal Mobile Telecommunications System
- UTC: Urban Traffic Control
- UUDI: Universal Description Discovery and Integration
- V2I: Vehicle-to-Infrastructure
- V2V: vehicle-to-vehicle
- VMS: Variable Message Sign
- VWSN: Vehicular Wireless Sensor Networks
- W3C: World Wide Web Consortium
- WSD: Word Sense Disambiguation
- WSDL: Web Services Description Language
- WSN: Wireless Sensor Networks
- XMI: XML Metadata Interchange standard di OMG
- XML: Extensible Markup Language
- ZTL: Zona a Traffico Limitato