

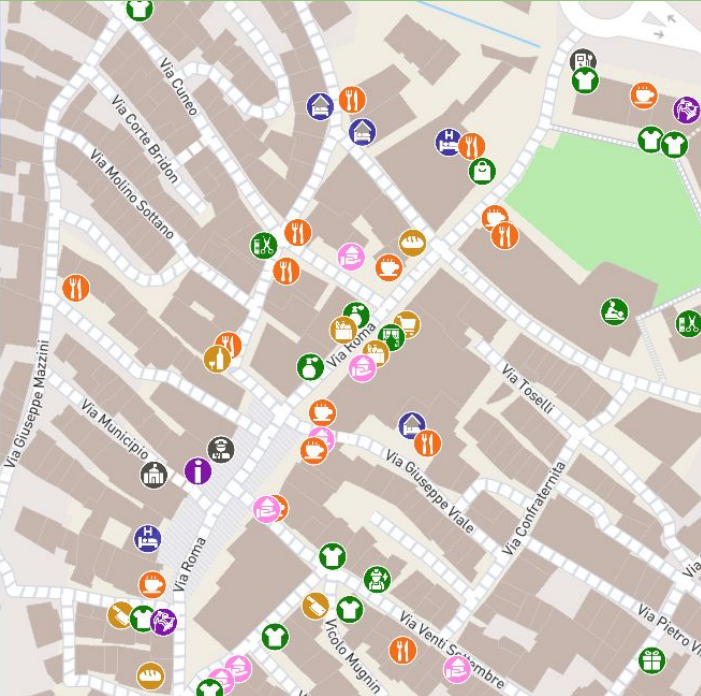
I dati di OpenStreetMap utilizzati a scopo turistico con la collaborazione dell'Ufficio Turistico per l'aggiornamento

Limone P.te - UTILITY

Mappa delle UTILITY - qui sotto puoi filtrare le categorie

Show Filters Info

Chalet Le Marmotte
restaurant
+39 348 330 1992;+39 329 311 5514
http://www.chaletlemarmotte.com
Baita Tienni
restaurant
http://www.baitatienni.it
Bagatto
restaurant
+39 0171 927543
http://www.releven11.it/
Severino Bottero
ticket
Luis Matlas
camp_site
+39 0171 927565
http://www.luismatlas.com
Chalet Sanremo
restaurant
+39 0171 92382





Realizzazione di una mappa delle utility
(area geografica del Comune di Limone P.te)

Google Sheet + Mapbox GL JS
controllo dati con Qgis

<https://www.mapbox.com/mapbox-gl-js/>

<https://docs.google.com/spreadsheets/>

<https://www.qgis.org/it/site/>

- i dati derivano dal database di openstreetmap e sono aggiornati giornalmente con una query overpass che importa i dati in un foglio di google:
- sul foglio di google, con la funzione =IMPORTDATA(vengono importati i dati con query csv di overpass turbo modificata grazie a [questo tutorial](#); di seguito esempio query

```
http://overpass-api.de/api/interpreter?
```

```
data=out:csv(,:id,:lat,:lon,:timestamp,:type,"name","amenity","phone","contact:phone","contact:mobile","mobile:phone",
,"website","facebook","contact:website","contact:facebook","email","contact:email";true;,"")
[timeout:20];area[admin_level=8][boundary=administrative][ref:ISTAT="004110"]->.boundaryarea; ( node(area.boundaryarea)
[amenity=cafe]; way(area.boundaryarea)[amenity=cafe]; node(area.boundaryarea)[amenity=bar]; way(area.boundaryarea)[amenity=bar]; ); out
meta center;"
```

- su questo foglio, al termine dai campi importati riportanti i tag di openstreetmap, sono presenti i campi "category", "name_web" e "search" dove con formule dedicate nel campo "category" si dà importanza ed univocità ad alcuni tag es. (se amenity=restaurant e tourism=hotel metti hotel), nel campo "name_web" si riporta il tag name e dove non presente la formula restituisce il primo valore non nullo degli altri tag e con il campo "name_web" si attribuisce una categoria dei punti di interesse. (Vedi [google sheet](#) per visualizzare le formule complete)
- i valori contenuti nel campo "category" sono utilizzati per creare le categorie riportate nel campo "search", che costituisce il parametro della funzione filtro presente nella mappa web
- con apps script è stata creata una funzione per aggiornare la query overpass turbo, che viene eseguita giornalmente mediante attivatore di funzioni di apps script (n.b.: è possibile creare la funzione anche registrando le operazioni manuali andando su estensioni>macro>registra macro)

```
function aggiornaImportData() { var foglioQuery = SpreadsheetApp.getActiveSpreadsheet().getSheetByName("query"); var cellaA1 =
foglioQuery.getRange("A1"); var formulaOriginale = cellaA1.getFormula(); cellaA1.clearContent(); cellaA1.setFormula(formulaOriginale); }
```

- i dati in formato csv sono stati inseriti su mappa web usando la libreria [Mapbox Gl JS](#) partendo da [questo tutorial](#);

Controllo dei dati di OpenStreetMap appartenenti alle categorie della mappa di cui sopra con Qgis e google sheet.

Google Sheet:

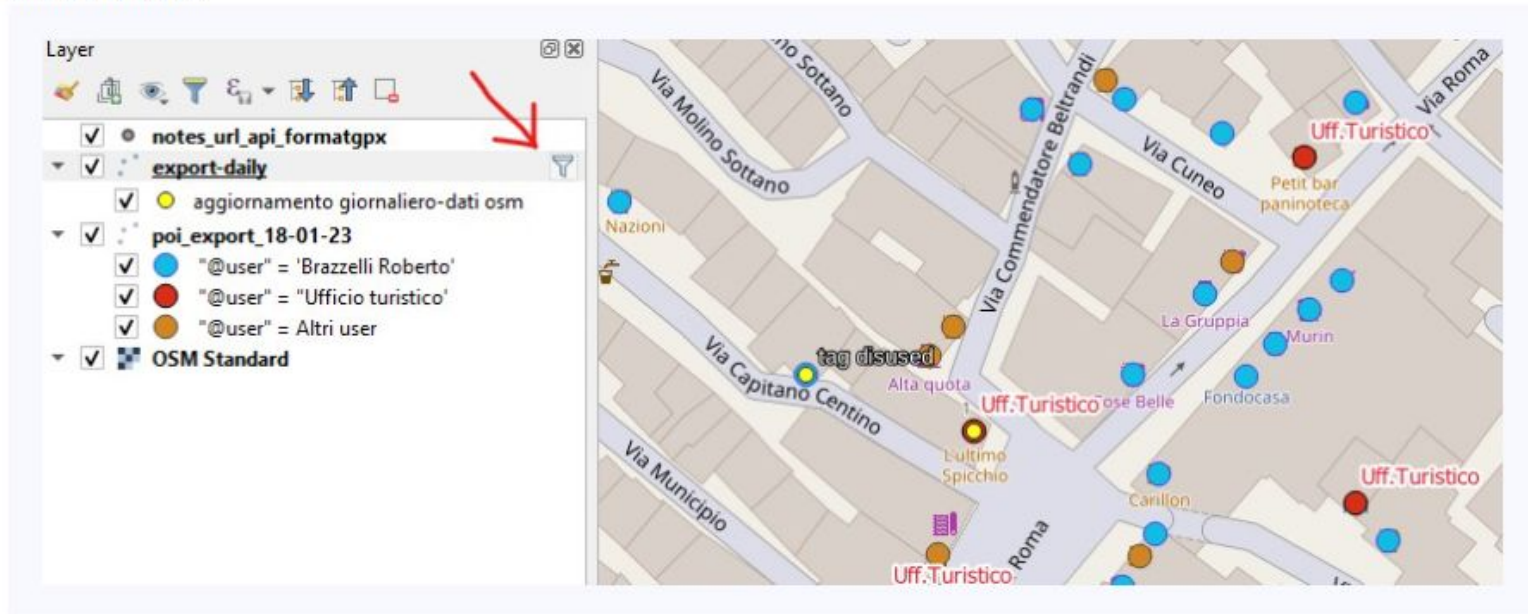
- partendo dai dati importanti su google sheet che si aggiornano giornalmente è stato creato uno script per esportare i dati in formato geojson in una cartella presente su google drive:

```
function convertRangeToGeoJSON() { var filename = "qgis_export.geojson"; // GeoJSON file name var folder = "yojr folder ID"; // Folder ID var spreadsheet = SpreadsheetApp.getActive(); spreadsheet.setActiveSheet(spreadsheet.getSheetByName('query'), true); var geoJSON = { type: "FeatureCollection", features: [] }; var rangeValues = SpreadsheetApp .getActiveSpreadsheet() .getActiveSheet() .getDataRange() .getValues(); var headers = rangeValues[0]; for (var i = 1; i < rangeValues.length; i++) { var properties = {}; for (var j = 0; j < headers.length; j++) { properties[headers[j]] = rangeValues[i][j]; } var coordinates = [rangeValues[i][2], rangeValues[i][1]]; // Latitudine: terza colonna, Longitudine: seconda colonna var feature = { type: "Feature", geometry: { type: "Point", coordinates: coordinates }, properties: properties }; geoJSON.features.push(feature); } var geoJSONString = JSON.stringify(geoJSON); // Verifica se il file esiste var files = DriveApp.getFolderById(folder).getFilesByName(filename); if (files.hasNext()) { // Il file esiste, sovrascrivilo files.next().setContent(geoJSONString); } else { // Il file non esiste, crealo DriveApp.getFolderById(folder).createFile(filename, geoJSONString, MIMEType.JSON); } var url = DriveApp.getFolderById(folder).getFilesByName(filename) .next() .getDownloadUrl() .replace("?e=download&gd=true", ""); return url; }
```

- per l'aggiornamento dei dati un attivatore di funzioni di apps script esegue la funzione ad orari prestabiliti (vedi sopra)

Qgis:

- è stato creato un progetto qgis contenente i dati aggiornati giornalmente inseriti con questo schema [https://drive.google.com/uc?id="your_id_file"&export?format=geojson](https://drive.google.com/uc?id=) che vengono rinominati come "export-daily" e quelli salvati in locale nel giorno del controllo "poi_export_datacontrollo"; eseguendo filtro temporale sui dati "export-daily" è possibile visualizzare solo quelli modificati dal giorno della data del precedente controllo; (dopo il controllo è sufficiente salvare una copia dei dati live "export-daily" sul proprio computer rinominando con la data del controllo)



- per il controllo delle modifiche avvenute sui dati, sia grafico che di contenuto, sono stati applicate espressioni, stili, filtri ed etichette (posizionando il cursore sui punti si apre subito un popup informativo...)

- per il controllo delle modifiche avvenute sui dati, sia grafico che di contenuto, sono stati applicate espressioni, stili, filtri ed etichette (posizionando il cursore sui punti si apre subito un popup informativo...)

The screenshot shows the OpenStreetMap interface. On the left, there is a legend with several layers checked, including 'notes_url_api_formatgpx', 'export-daily', 'aggiornamento giornaliero-dat', 'poi_export_18-01-23', and 'OSM Standard'. The map displays various colored points: blue, red, and orange. A yellow popup window is overlaid on the map, displaying the following information:

DATI OSM

USER OSM: Brazzelli Roberto

ID OSM-url: "id"

GEOMETRIA: node

DATA ULTIMA MODIFICA : 2023-01-09T15:08:48.000Z

Nel diario utente di OpenStreetMaps è possibile scaricare i file degli stili qml di qis utente *Brazzelli Roberto*

- per un controllo dei dati sono state inserite anche le note con url dinamico utilizzando le api di openstreetmap

Aggiornamento dei dati

Grazie alla collaborazione con Olivier Bottero dell'ufficio turistico del comune di Limone Piemonte, si è deciso di promuovere la mappatura in openstreetmap per l'aggiornamento dei dati, utilizzata ad oggi proprio dall'ufficio turistico, con successivo controllo da parte del sottoscritto tecnico Geom. Roberto Brazzelli incaricato dal Comune per aggiornamento della mappa.

Nelle informazioni della [mappa delle utility](#) si danno prime informazioni a cittadini o esercenti attività per segnalare errori o nuove attività

..credo che la cosa più difficile sia stata studiare e testare una soluzione semplice pensando che possa essere riutilizzata facilmente da altri uffici turistici, atl o consorzi turistici...

mappa delle utility

<https://www.bwebmap.it/maps/limoneUtility/index.html>

procedura

<https://www.openstreetmap.org/user/Brazzelli%20Roberto/diary/400822>

GRAZI PER L'ATTENZIONE

Roberto Brazzelli
geom.brazzelli@gmail.com
www.bwebmap.it