

Training on the Job - Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
 - o Tipi di dato
 - o Chiavi primarie ed esterne
 - o Vincoli di integrità



Training on the Job - Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
o Tipi di dato



Altra Windows
Per la installazione per Windows

Training on the Job - Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto

e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)

PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
 - o Tipi di dato
 - o Chiavi primarie ed esterne

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
 - o Tipi di dato
 - o Chiavi primarie ed esterne
 - o Vincoli di integrità

- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
 - o Tipi di dato
 - o Chiavi primarie ed esterne
 - o Vincoli di integrità

2. Installazione di [Postgres](#) e [PostGIS](#), Setup dell'Ambiente di Lavoro e Creazione Tabelle

Obiettivi:

- Installare [PostgreSQL](#) e [PostGIS](#)
- Configurare un ambiente di lavoro

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
 - o Tipi di dato
 - o Chiavi primarie ed esterne
 - o Vincoli di integrità

2. Installazione di [Postgres](#) e [PostGIS](#), Setup dell'Ambiente di Lavoro e Creazione Tabelle

Obiettivi:

- Installare [PostgreSQL](#) e [PostGIS](#)
- Configurare un ambiente di lavoro
- Creare lo schema di tabelle secondo specifiche
- Eseguire un test funzionale

2. Installazione di Postgres e PostGIS, Setup dell'Ambiente di Lavoro e Creazione Tabelle

Obiettivi:

- Installare PostgreSQL e PostGIS
- Configurare un ambiente di lavoro
- Creare lo schema di tabelle secondo specifiche
- Eseguire un test funzionale

Fasi Operative:

2.1 Installazione PostgreSQL + PostGIS:

- Scaricare da <https://www.postgresql.org>
- Installazione guidata (con StackBuilder per estensioni)

Fasi Operative:

2.1 Installazione PostgreSQL + PostGIS:

- Scaricare da <https://www.postgresql.org>
- Installazione guidata (con [StackBuilder](#) per estensioni)
- Mostrare l'utilizzo dell'applicazione [pgAdmin](#) per l'interfaccia grafica del db
- Installare l'estensione [PostGIS](#) tramite [StackBuilder](#) (oppure tramite terminale [Postgres](#) con `CREATE EXTENSION PostGIS`)

2.2 Configurazione Ambiente di Lavoro:

- Creazione database [progetto ISPRA](#) sia tramite [pgAdmin](#) sia tramite terminale [Postgres](#) (`CREATE DATABASE progetto ISPRA`)
- Attivazione di [PostGIS](#) sul nuovo database

2.3 Creazione Tabelle:

- Creazione tabelle tramite [pgAdmin](#) esempio
- Creazione tabelle tramite script SQL derivati dallo schema relazionale:

```
CREATE TABLE poligono popolazione (  
  Id\_poligono SERIAL PRIMARY KEY,  
  geom GEOMETRY(Polygon, 4326),
```

- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto
2. Identificazione delle entità principali (es. aree verdi, aree blu, poligono per popolazione, ingressi per aree)
3. Definizione delle relazioni (es. un'area può avere molti ingressi)
4. Rappresentazione E/R con strumenti visuali (es. draw.io)
5. Produzione di uno schema relazionale
6. Redazione delle specifiche tecniche con:
 - o Tipi di dato
 - o Chiavi primarie ed esterne
 - o Vincoli di integrità

2. Installazione di Postgres e PostGIS, Setup dell'Ambiente di Lavoro e Creazione Tabelle

Obiettivi:

- Installare PostgreSQL e PostGIS
- Configurare un ambiente di lavoro
- Creare lo schema di tabelle secondo specifiche
- Eseguire un test funzionale



o. redazione delle specifiche tecniche con:

- o Tipi di dato
- o Chiavi primarie ed esterne
- o Vincoli di integrità

2. Installazione di Postgres e PostGIS, Setup dell'Ambiente di Lavoro e Creazione Tabelle

Obiettivi:

- Installare PostgreSQL e PostGIS
- Configurare un ambiente di lavoro
- Creare lo schema di tabelle secondo specifiche
- Eseguire un test funzionale





File Explorer window titled "QGIS 3.42.2" showing the contents of the QGIS 3.42.2 folder. The left sidebar shows the navigation pane with "Questo PC" and "Rete" expanded. The main pane displays a list of files and folders.

Nome	Ultima modifica	Tip	Dimensione
GRASS GIS 6.4.1	14/05/2023 15:55	Collegamento	2 KB
OSGeo4W Setup	14/05/2023 15:54	Collegamento	2 KB
OSGeo4W Shell	14/05/2023 15:55	Collegamento	2 KB
QGIS Desktop 3.42.2	14/05/2023 15:54	Collegamento	1 KB
Qt Designer with QGIS 3.42.2 system wid...	14/05/2023 15:54	Collegamento	2 KB
SAGA GIS 5.7.1	14/05/2023 15:54	Collegamento	2 KB

pgAdmin
Management Tools for PostgreSQL

Waiting for pgadmin 4 to start...

Attiva Windows
Recupera le impostazioni per attivare Windows.

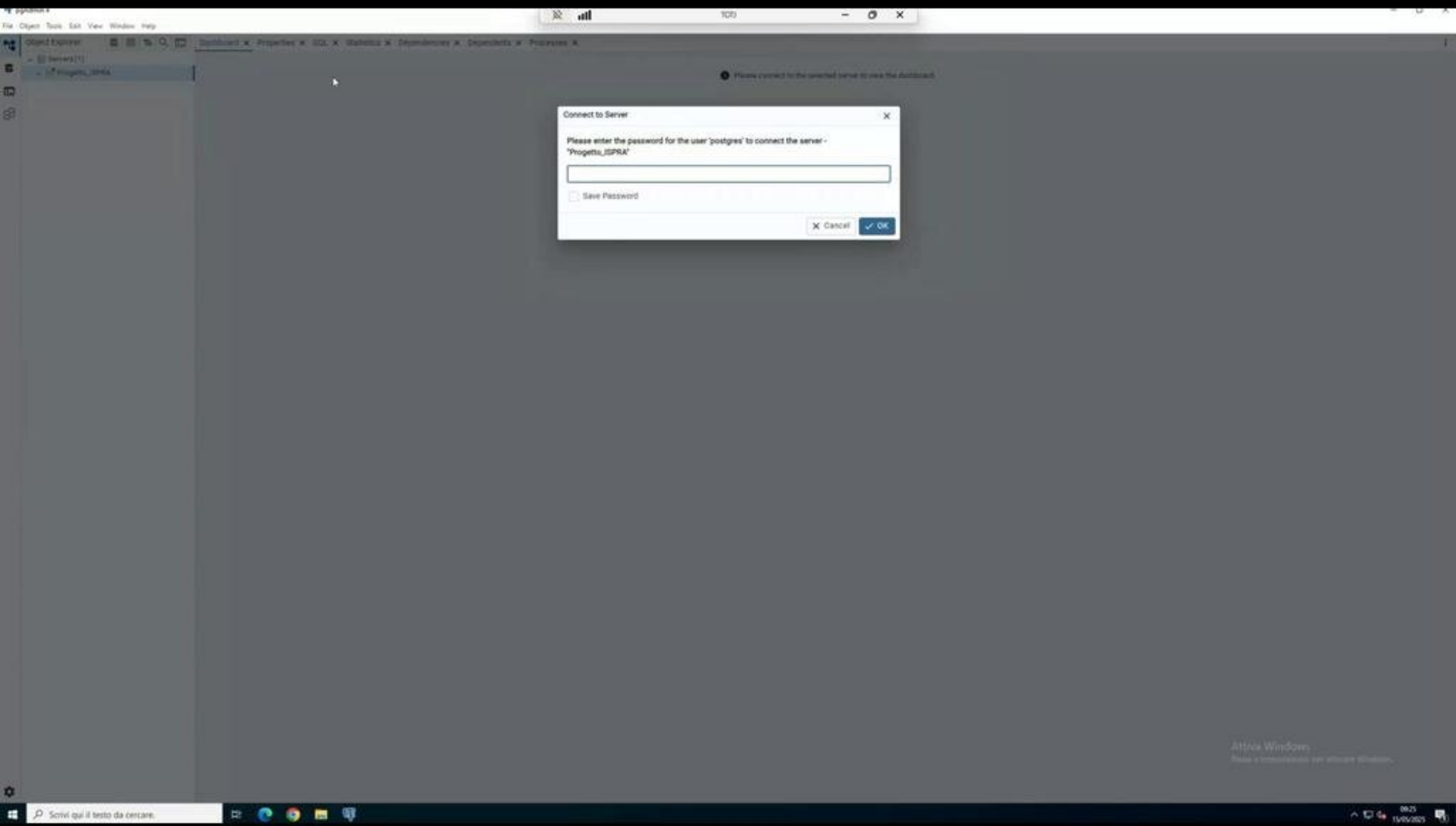


Attiva Windows
Vedi le impostazioni per attivare Windows.

Attiva Windows
Vai a Impostazioni per attivare Windows



Loading pgAdmin 4 v9.0...



- Attivazione di PostGIS sul nuovo database

2.3 Creazione Tabelle:

- Creazione tabelle tramite pgAdmin esempio
- Creazione tabelle tramite script SQL derivati dallo schema relazionale:

```
CREATE TABLE poligono_popolazione (  
  Id_poligono SERIAL PRIMARY KEY,  
  geom GEOMETRY(Polygon, 4326),  
  N_popolazione INT,  
  Data_valutazione DATE  
);
```

2.4 Test Funzionale:

- Inserimento dati di test tramite script SQL:

```
INSERT INTO poligono_popolazione (geom, n_popolazione, data_valutazione)  
VALUES (  
  ST_SetSRID(  
    ST_GeomFromText('POLYGON(( 1250000 5700000, 1251000 5700000, 1251000  
5701000, 1250000 5701000, 1250000 5700000 ))'), 3857),  
  1250,  
  '2025-05-07'
```

3. Installazione di Tomcat e GeoServer

Obiettivi:

- Installare Java, Tomcat e Geoserver
- Verificarne il funzionamento
- Connettere Geoserver a PostGIS
- Creare un layer su Geoserver

3.1 Scarica Apache Tomcat:

- Vai al sito ufficiale di Apache Tomcat: <https://tomcat.apache.org/download-90.cgi>
- Seleziona la versione di Tomcat 9.x, che è compatibile con GeoServer.
- Scarica il file ZIP per Windows o tar.gz per Linux/macOS.

3.2 Estrai Tomcat:

- Windows: Estrai il file ZIP in una cartella, ad esempio C:\tomcat.

3.3 Verifica il funzionamento di Tomcat:

- Windows: Esegui startup.bat in C:\tomcat\bin.
- Linux/macOS: Esegui ./startup.sh nella cartella bin.
- Visita <http://localhost:8080> nel browser.

3.4 Installazione di GeoServer su Tomcat:

- Seleziona la versione di Tomcat 9.x, che è compatibile con GeoServer.
- Scarica il file ZIP per Windows o tar.gz per Linux/macOS.

3.2 Estrai Tomcat:

- Windows: Estrai il file ZIP in una cartella, ad esempio C:\tomcat.

3.3 Verifica il funzionamento di Tomcat:

- Windows: Esegui startup.bat in C:\tomcat\bin.
- Linux/macOS: Esegui ./startup.sh nella cartella bin.
- Visita <http://localhost:8080> nel browser.

3.4 Installazione di GeoServer su Tomcat:

1. Scarica GeoServer da <https://geoserver.org/download/>
2. Copia il file geoserver.war nella cartella webapps di Tomcat.
3. Avvia Tomcat e accedi a <http://localhost:8080/geoserver>

3.5 Connessione di GeoServer a PostGIS:

- Login: admin / geoserver
- Menu Data > Stores > Add new Store > PostGIS
- Configura parametri (host, porta, database, username, password)
- Salva e pubblica layer

- Windows: Estrai il file ZIP in una cartella, ad esempio C:\tomcat.

3.3 Verifica il funzionamento di Tomcat:

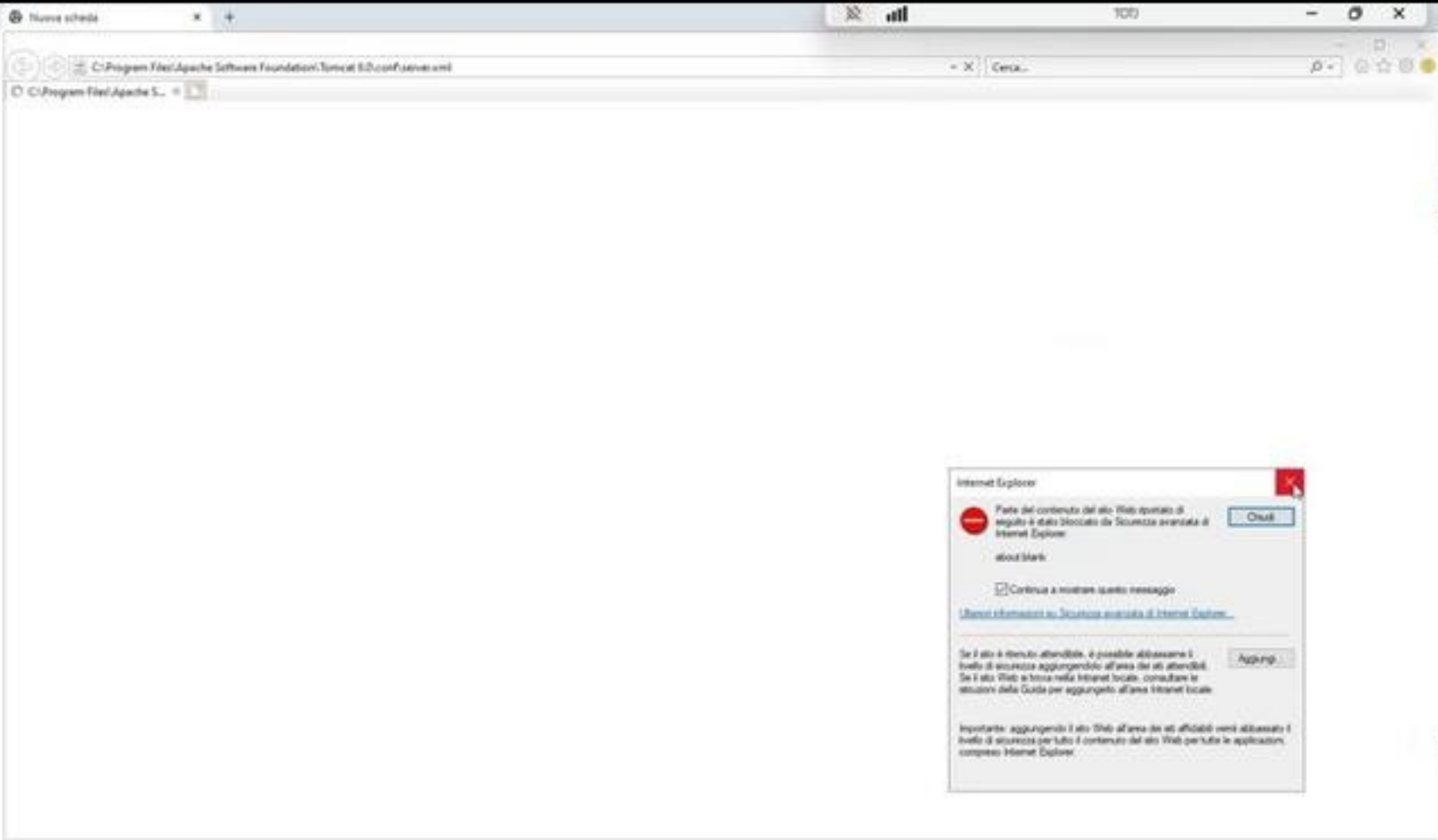
- Windows: Esegui startup.bat in C:\tomcat\bin.
- Linux/macOS: Esegui ./startup.sh nella cartella bin.
- Visita <http://localhost:8080> nel browser.

3.4 Installazione di GeoServer su Tomcat:

1. Scarica GeoServer da <http://www.geo-server.org/download/>
2. Copia il file geoserver.war nella cartella webapps di Tomcat.
3. Avvia Tomcat e accedi a <http://localhost:8080/geoserver>

3.5 Connessione di GeoServer a PostGIS:

- Login: admin / geoserver
- Menu Data > Stores > Add new Store > PostGIS
- Configura parametri (host, porta, database, username, password)
- Salva e pubblica layer



Attiva Windows
Ricerca e impostazioni per attivare Windows

Personalizza Chrome

3.2 Estrai Tomcat:

- Windows: Estrai il file ZIP in una cartella, ad esempio C:\tomcat.

3.3 Verifica il funzionamento di Tomcat:

- Windows: Esegui startup.bat in C:\tomcat\bin.
- Linux/macOS: Esegui ./startup.sh nella cartella bin.
- Visita <http://localhost:8080> nel browser.

3.4 Installazione di GeoServer su Tomcat:

1. Scarica GeoServer da <https://geoserver.org/download/>
2. Copia il file geoserver.war nella cartella webapps di Tomcat.
3. Avvia Tomcat e accedi a <http://localhost:8080/geoserver>

3.5 Connessione di GeoServer a PostGIS:

- Login: admin / geoserver
- Menu Data > Stores > Add new Store > PostGIS
- Configura parametri (host, porta, database, username, password)
- Salva e pubblica layer



Scale = 1 : 273K
Click on the map to get feature info
1249720.71233, 5698262.23133

Attiva Windows
Ricerca e impostazioni per attivare Windows

- Creazione tabelle tramite script SQL derivati dallo schema relazionale:

```
CREATE TABLE poligono_popolazione (  
  Id_poligono SERIAL PRIMARY KEY,  
  geom GEOMETRY(Polygon, 4326),  
  N_popolazione INT,  
  Data_valutazione DATE  
);
```

2.4 Test Funzionale:

- Inserimento dati di test tramite script SQL:

```
INSERT INTO poligono_popolazione (geom, n_popolazione, data_valutazione)  
VALUES (  
  ST_SetSRID(  
    ST_GeomFromText('POLYGON(( 1250000 5700000, 1251000 5700000, 1251000  
5701000, 1250000 5701000, 1250000 5700000 ))'), 3857),  
  1250,  
  '2025-05-07'  
);
```

- Creazione tabelle tramite script SQL derivati dallo schema relazionale:

```
CREATE TABLE poligono_popolazione (  
  Id_poligono SERIAL PRIMARY KEY,  
  geom GEOMETRY(Polygon, 4326),  
  N_popolazione INT,  
  Data_valutazione DATE  
);
```

2.4 Test Funzionale:

- Inserimento dati di test tramite script SQL:

```
INSERT INTO poligono_popolazione (geom, n_popolazione, data_valutazione)  
VALUES (  
  ST_SetSRID(  
    ST_GeomFromText('POLYGON(( 1250000 5700000, 1251000 5700000, 1251000  
5701000, 1250000 5701000, 1250000 5700000 ))'), 3857),  
  1250,  
  '2025-05-07'  
);
```


3.6 Definizione Layer in GeoServer:

- Layers > Add a new Layer
- Seleziona tabella (es. areeverdi), configura SRS EPSG:3857, stile, bounding box
- Pubblica

3.7 Verifica la visualizzazione tramite WMS/WFS:

- Apertura con OpenLayer da Geoserver
- Verifica corretta visualizzazione

4 Verifica competenze acquisite:

<https://docs.google.com/forms/d/e/1FAIpQLSe6fSlzeIrA-DNYivR2j94-xks6hCCRQlgBtQRa6Zk3uBNThw/viewform?usp=dialog>

- **ST_NumPoints(geometry)** returns the number of dimensions in the geometry
- **ST_SRID(geometry)** returns the spatial reference identifier number of the geometry

```
SELECT name, ST_GeometryType(geom), ST_NDims(geom), ST_SRID(geom)
FROM geometries;
```

name	ST_GeometryType	ST_NDims	ST_SRID
Point	ST_Point	2	0
Polygon	ST_Polygon	2	0
PolygonWithHole	ST_Polygon	2	0
Collection	ST_GeometryCollection	2	0
LineString	ST_LineString	2	0

9.3.1. Points



A spatial **point** represents a single location on the Earth. This point is represented by a single coordinate (including either 2-, 3- or 4- dimensions). Points are used to represent objects when the exact details, such as shape and size, are not important at the target scale. For example, cities on a map of the world can be described as points, while a map of a single state might represent cities as polygons.

```
SELECT ST_AsText(geom)
FROM geometries
WHERE name = 'Point';
```

```
POINT(0 0)
```

Some of the specific spatial functions for working with points are:

- **ST_X(geometry)** returns the X ordinate
- **ST_Y(geometry)** returns the Y ordinate

So, we can read the ordinates from a point like this:

```
SELECT ST_X(geom), ST_Y(geom)
FROM geometries
WHERE name = 'Point';
```

The New York City subway stations (`nyc_subway_stations`) table is a data set represented as points. The following SQL query will return the geometry associated with one point (in the **ST_AsText** column).

```
SELECT name, ST_AsText(geom)
FROM nyc_subway_stations
LIMIT 1;
```

9.3.2. LineStrings



A **linestring** is a path between locations. It takes the form of an ordered series of two or more points. Roads and rivers are typically represented as linestrings. A linestring is said to be **closed** if it starts and ends on the same point. It is said to be **simple** if it does not cross or touch itself (except at its endpoints if it is closed). A linestring can be both **closed** and **simple**.

The street network for New York (`nyc_streets`) was loaded earlier in the workshop. This dataset contains details such as name,

return the geometry associated with one point (in the **ST_AsText** column)

```
SELECT name, ST_AsText(geom)
FROM nyc_subway_stations
LIMIT 1;
```

9.3.2. Linestrings



A **linestring** is a path between locations. It takes the form of an ordered series of two or more points. Roads and rivers are typically represented as linestrings. A linestring is said to be **closed** if it starts and ends on the same point. It is said to be **simple** if it does not cross or touch itself (except at its endpoints if it is closed). A linestring can be both **closed** and **simple**.

The street network for New York (**nyc_streets**) was loaded earlier in the workshop. This dataset contains details such as name, and type. A single real world street may consist of many linestrings, each representing a segment of road with different attributes.

The following SQL query will return the geometry associated with one linestring (in the **ST_AsText** column).

```
SELECT ST_AsText(geom)
FROM geometries
WHERE name = 'Linestring';
```

```
LINESTRING(0 0, 1 1, 2 3, 2 2)
```

Some of the specific spatial functions for working with linestrings are:

- **ST_Length(geometry)** returns the length of the linestring
- **ST_StartPoint(geometry)** returns the first coordinate as a point
- **ST_EndPoint(geometry)** returns the last coordinate as a point
- **ST_NPoints(geometry)** returns the number of coordinates in the linestring

So, the length of our linestring is:

```
SELECT ST_Length(geom)
FROM geometries
WHERE name = 'Linestring';
```

```
3.41421356237309
```

9.3.3. Polygons



A **polygon** is a representation of an area. The outer boundary of the polygon is represented by a **ring**. This ring is a linestring that is both **closed** and **simple** as defined above. Holes within the polygon are also represented by rings.

Polygons are used to represent objects whose size and shape are important. City limits, parks, building footprints or bodies of water are all commonly represented as polygons when the scale is sufficiently high to see their area. Roads and rivers can sometimes be represented as polygons.

The following SQL query will return the geometry associated with one polygon (in the **ST_AsText** column).

The following SQL query will return the geometry associated with one linestring (in the **ST_AsText** column):

```
SELECT ST_AsText(geom)
FROM geometries
WHERE name = 'Linestring';
```

```
LINESTRING(0 0, 1 1, 2 1, 2 2)
```

Some of the specific spatial functions for working with linestrings are:

- **ST_Length(geometry)** returns the length of the linestring
- **ST_StartPoint(geometry)** returns the first coordinate as a point
- **ST_EndPoint(geometry)** returns the last coordinate as a point
- **ST_NPoints(geometry)** returns the number of coordinates in the linestring

So, the length of our linestring is:

```
SELECT ST_Length(geom)
FROM geometries
WHERE name = 'Linestring';
```

```
3.41421356237469
```

9.3.3. Polygons



A polygon is a representation of an area. The outer boundary of the polygon is represented by a ring. This ring is a linestring that is both closed and simple as defined above. Holes within the polygon are also represented by rings.

Polygons are used to represent objects whose size and shape are important. City limits, parks, building footprints or bodies of water are all commonly represented as polygons when the scale is sufficiently high to see their area. Roads and rivers can sometimes be represented as polygons.

The following SQL query will return the geometry associated with one polygon (in the **ST_AsText** column):

```
SELECT ST_AsText(geom)
FROM geometries
WHERE name LIKE 'Polygon';
```

Rather than using an `~` sign in our `WHERE` clause, we are using the `LIKE` operator to carry out a string matching operation. **You may be used to the `~` symbol as a "glob" for pattern matching, but in SQL the `~` symbol is used, along with the `LIKE` operator to tell the system to do globbing.**

```
POLYGON((0 0, 1 0, 1 1, 0 1, 0 0))
POLYGON((0 0, 10 0, 10 10, 0 10, 0 0),(1 1, 1 2, 2 2, 2 1, 1 1))
```

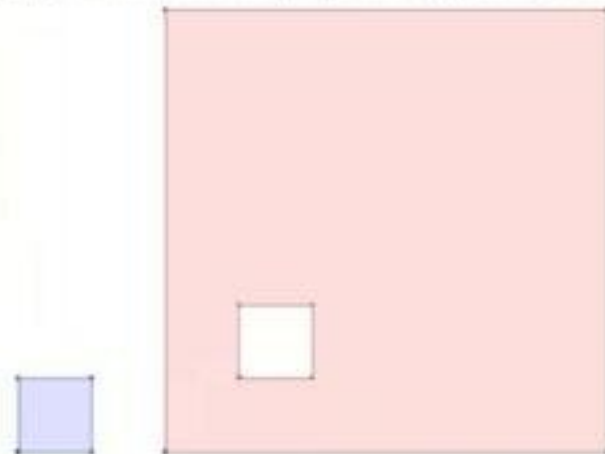
The first polygon has only one ring. The second one has an interior "hole". Most graphics systems include the concept of a "polygon", but GIS systems are relatively unique in allowing polygons to explicitly have holes.



Remember when writing SQL queries for spatial data, the same rules apply as when writing SQL queries for other data. When writing SQL queries for spatial data, the same rules apply as when writing SQL queries for other data. When writing SQL queries for spatial data, the same rules apply as when writing SQL queries for other data.

```
POLYGON((0 0, 10 0, 10 10, 0 10, 0 0))
POLYGON((0 0, 10 0, 10 10, 0 10, 0 0),(1 1, 1 2, 2 2, 2 1, 1 1))
```

The first polygon has only one ring. The second one has an interior "hole". Most graphics systems include the concept of a "polygon", but GIS systems are relatively unique in allowing polygons to explicitly have holes.



Some of the specific spatial functions for working with polygons are:

- **ST_Area(geometry)** returns the area of the polygons
- **ST_Nrings(geometry)** returns the number of rings (usually 1, more if there are holes)
- **ST_ExteriorRing(geometry)** returns the outer ring as a linestring
- **ST_InteriorRingN(geometry,n)** returns a specified interior ring as a linestring
- **ST_Perimeter(geometry)** returns the length of all the rings

We can calculate the area of our polygons using the area function:

```
SELECT name, ST_Area(geom)
FROM geometries
WHERE name LIKE 'Polygon%';
```

```
Polygon          1
PolygonWithHole  99
```

Note that the polygon with a hole has an area that is the area of the outer shell (a 10x10 square) minus the area of the hole (a 1x1 square).

9.3.4. Collections

There are four collection types, which group multiple simple geometries into sets.

- **MultiPoint**, a collection of points
- **MultiLineString**, a collection of linestrings
- **MultiPolygon**, a collection of polygons
- **GeometryCollection**, a heterogeneous collection of any geometry (including other collections)

Collections are another concept that shows up in GIS software more than in generic graphics software. They are useful for directly

- **ST_Nrings(geometry)** returns the number of rings (usually 1, more if there are holes)
- **ST_ExteriorRing(geometry)** returns the outer ring as a linestring
- **ST_InteriorRingN(geometry,n)** returns a specified interior ring as a linestring
- **ST_Perimeter(geometry)** returns the length of all the rings

We can calculate the area of our polygons using the area function:

```
SELECT name, ST_Area(geom)
FROM geometries
WHERE name LIKE 'Polygon';
```

Polygon	1
PolygonWithHole	99

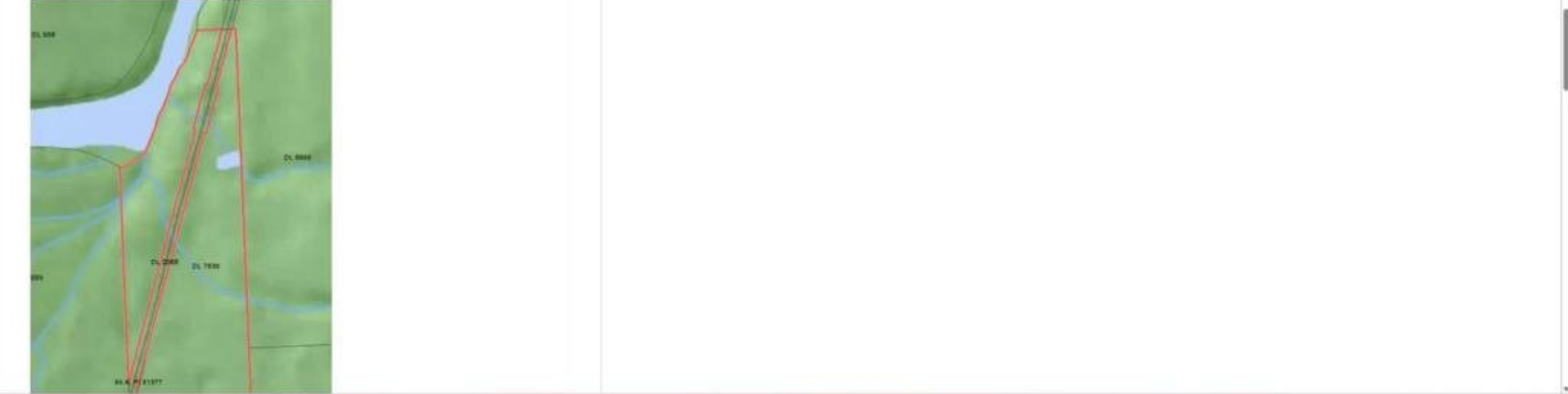
Note that the polygon with a hole has an area that is the area of the outer shell (a 10x10 square) minus the area of the hole (a 1x1 square).

9.3.4. Collections

There are four collection types, which group multiple simple geometries into sets.

- **MultiPoint**, a collection of points
- **MultiLineString**, a collection of linestrings
- **MultiPolygon**, a collection of polygons
- **GeometryCollection**, a heterogeneous collection of any geometry (including other collections)

Collections are another concept that shows up in GIS software more than in generic graphics software. They are useful for directly modeling real world objects as spatial objects. For example, how to model a lot that is split by a right-of-way? As a **MultiPolygon**, with a part on either side of the right-of-way.



with a part on either side of the right-of-way



Our example collection contains a polygon and a point.

```
SELECT name, ST_AsText(geom)
FROM geometries
WHERE name = 'Collection';
```

```
GEOMETRYCOLLECTION(POINT(1 0),POLYGON((0 0, 1 0, 1 1, 0 1, 0 0)))
```



Some of the specific spatial functions for working with collections are:

- **ST_NumGeometries(geometry)** returns the number of parts in the collection
- **ST_GeometryN(geometry,n)** returns the specified part
- **ST_Area(geometry)** returns the total area of all polygonal parts
- **ST_Length(geometry)** returns the total length of all linear parts

9.4. Geometry Input and Output

Within the database, geometries are stored on disk in a format only used by the PostGIS program. In order for external programs to insert and retrieve spatial geometries, they need to be converted into a format that other applications can understand. Fortunately,

Training on the Job - Modulo 1

Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

te:

lisi dei requisiti informativi del progetto

• Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSSG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box

Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

Training on the Job - Modulo 2

Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

• Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database

Training on the Job - Modulo 2 Impo

Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPG) e metadata
- Comprendere quali aree verdi si importeranno

Espr operative

Training on the Job - Modulo 2

Importazione de

Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata

Training on the Job - Modulo 1

Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto

Training on the Job - Modulo 2

Importazione dei dati da |

Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSS) e metadata

Training on the Job - Modulo 2

Importazione dei dati da file Shapefile o altri DB georeferenziati

• Modulo 2 – Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database

Training on the Job - Modulo 2

Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni

Training on the Job - Modulo 2

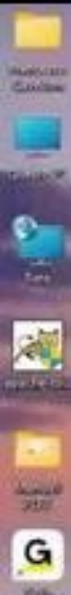
Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)

1. Analisi dei requisiti informativi del progetto



Area (Map) MMS2021 Update Sim Sublog sim

Training_on_the_job

Training_on_the_job

Cerca in Training_on_the_job

Nome

Ultima modifica

Tipi

Dimensione

DATA	10/05/2023 09:17	Cartella di file	
area_vendita_AC_10_04_2023 - Filippo_vend...	10/05/2023 11:09	Presentazione di...	5.002 KB
DAORAMMA_ER_10PFA.pdf	08/05/2023 11:08	Microsoft Edge H...	20 KB
Modulo_1.docx	15/05/2023 09:00	Documento di MS...	31 KB
Modulo_2.docx	10/05/2023 10:01	Documento di MS...	32 KB
Modulo_3.docx	10/05/2023 09:04	Documento di MS...	31 KB
Modulo_4.docx	11/05/2023 09:00	Documento di MS...	30 KB
Piano_attivita.docx	10/04/2023 11:05	Documento di file...	37 KB

0 elementi 1 elemento selezionato 32,5 KB

1 Fase (8 giornate)

1. Definizione del diagramma E/R e delle specifiche relative per la creazione dell'infrastruttura DB (1 giornata) – in sede ISPRA (già realizzata)
2. Installazione di PostGIS, definizione di un ambiente di lavoro, creazione delle tabelle ritenute necessarie per il progetto. Test funzionale. (1/2 giornata)
3. Installazione di Geoserver, definizione dei layer e integrazione delle tabelle. Test funzionale. (1/2 giornata)
4. Importazione dei dati da file Shape o altri db georeferenziati per la costruzione degli elementi verdi e blu (1 e ½ giornate)
5. Importazione dei dati della popolazione su raster fornito da ISPRA (1 e ½ giornate)
6. Installazione ambiente di lavoro usando Eclipse, definizione classi per accesso DB (2 giornate)
7. Confronto su sviluppo del geodatabase - (1 giornata) - in sede ISPRA

2 fase (24 giornate)

1. Calcolo dei punti di ingresso delle aree verdi e blu (4 giornate)
2. Calcolo della distanza minima fra i centroidi dei poligoni rappresentanti la popolazione e i punti di ingresso delle aree verdi e blu (2 giornate)
3. Stesura del codice per la visualizzazione degli elementi su mappa georeferenziata interattiva (10 giornate)
4. Confronto su attività di sviluppo parte client – (2 giornate) – in sede ISPRA



The screenshot shows a Windows File Explorer window with the address bar indicating the path: Questo PC > Disco locale (C:) > Programmi > Apache Software Foundation > Tomcat 8.5 > conf. The left sidebar shows the navigation pane with 'Questo PC' and 'Disco locale (C:)' selected. The main area displays a list of files and folders:

Nome	Ultima modifica	Tip	Dimensione
Catalina	29/05/2025 14:34	Catella di File	
catalina.policy	24/04/2025 14:32	File POLICY	3.2 KB
catalina.properties	24/04/2025 14:32	File PROPERTIES	8.8 KB
context.xml	24/04/2025 14:32	Documenti XML	2.4 KB
jeepic-providers.xml	24/04/2025 14:32	Documenti XML	2.4 KB
jeepic-providers.xml	24/04/2025 14:32	File XML	2.4 KB
logging.properties	24/04/2025 14:32	File PROPERTIES	4.4 KB
server.xml	29/05/2025 14:34	Documenti XML	9.8 KB
tomcat-users.xml	29/05/2025 14:34	Documenti XML	2.4 KB
tomcat-users.xml	24/04/2025 14:32	File XML	2.4 KB
web.xml	24/04/2025 14:32	Documenti XML	174 KB

The 'server.xml' file is highlighted in blue. At the bottom of the window, it shows '11 elementi' and '1 elemento selezionato: 9.00 KB'.



QGIS 3.42.2

QGIS 3.42.2

	Nome	Ultima modifica	Tipi	Dimensione
Accesso rapido				
Desktop	GRASS GIS 8.4.1	14/05/2025 15:53	Collegamento	2 KB
Download	OSGeo4W Setup	14/05/2025 15:54	Collegamento	2 KB
Documenti	OSGeo4W Shell	14/05/2025 15:53	Collegamento	2 KB
Immagini	QGIS Desktop 3.42.2	14/05/2025 15:54	Collegamento	1 KB
Questo PC	Qt Designer with QGIS 3.42.2 custom end...	14/05/2025 15:54	Collegamento	2 KB
Desktop	SAGA GIS 5.7.1	14/05/2025 15:54	Collegamento	2 KB
Documenti				
Download				
Immagini				
Musica				
Oggetti 3D				
Video				
Disco locale (C:)				
Rete				

8 elementi 1 elemento selezionato 375 byte

Attiva Windows
Fai la impostazione per attivare Windows.

1 Fase (8 giornate)

1. Definizione dell'infrastruttura
2. Installazione delle tabelle ritenute funzionali.
3. Installazione funzionale.
4. Importazione degli elementi verdi e blu (1 e ½ giornate)
5. Importazione dei dati della popolazione su raster fornito da ISPRA (1 e ½ giornate)
6. Installazione ambiente di lavoro usando Eclipse, definizione classi per accesso DB (2 giornate)
7. Confronto su sviluppo del geodatabase - (1 giornata) - in sede ISPRA

2 fase (24 giornate)

1. Calcolo dei punti di ingresso delle aree verdi e blu (4 giornate)
2. Calcolo della distanza minima fra i centroidi dei poligoni rappresentanti la popolazione e i punti di ingresso delle aree verdi e blu (2 giornate)
3. Stesura del codice per la visualizzazione degli elementi su mappa georeferenziata interattiva (10 giornate)
4. Confronto su attività di sviluppo parte client – (2 giornate) – in sede ISPRA



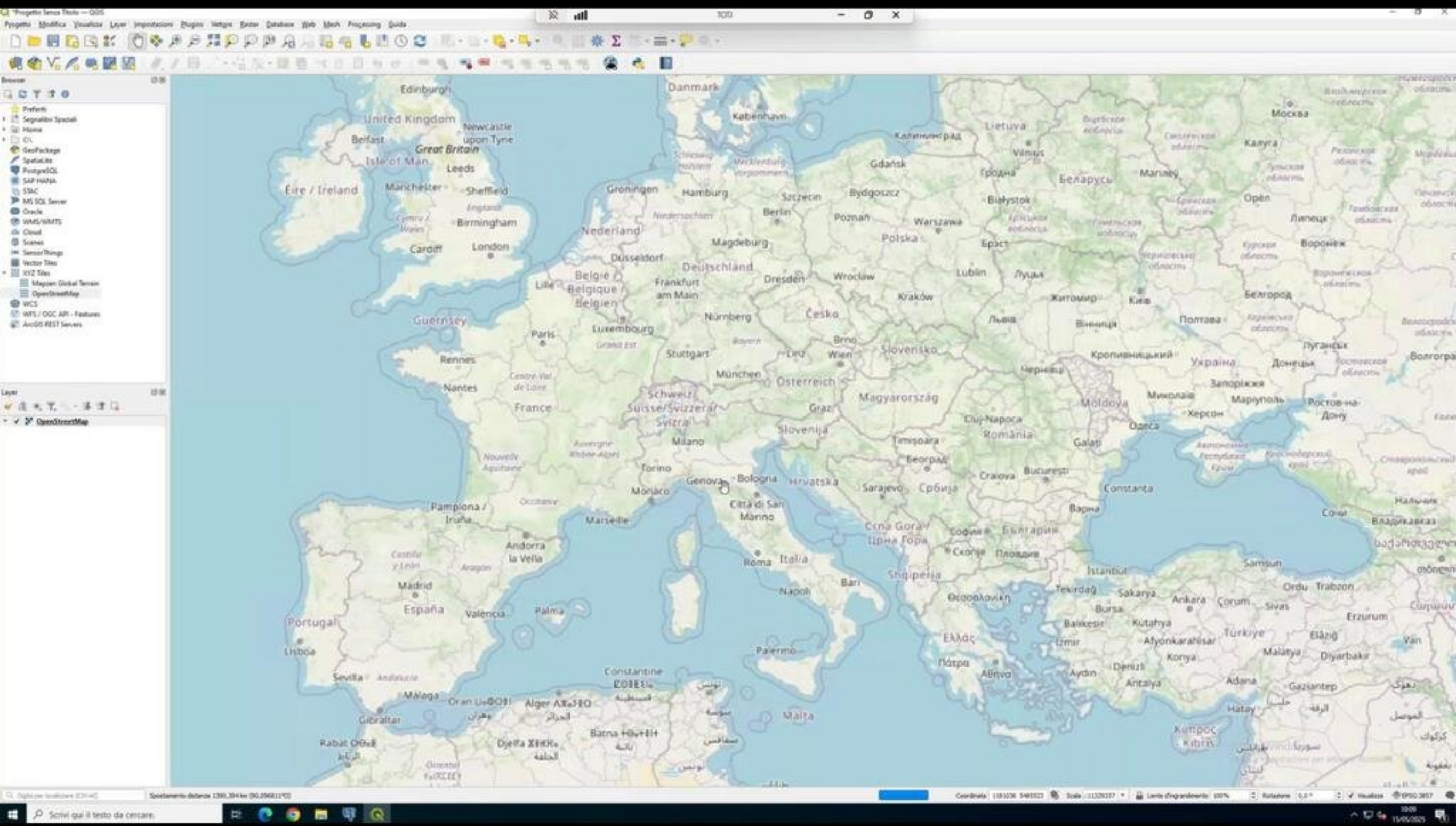
QGIS 3.42.2

QGIS 3.42.2

Nome	Ultima modifica	Tip	Dimensione
GRASS GIS 6.4.1	14/05/2025 15:53	Collegamento	2 KB
OSGeo4W Setup	14/05/2025 15:54	Collegamento	2 KB
OSGeo4W Shell	14/05/2025 15:53	Collegamento	2 KB
QGIS Desktop 3.42.2	14/05/2025 15:54	Collegamento	1 KB
Qt Designer with QGIS 3.42.2 custom end...	14/05/2025 15:54	Collegamento	2 KB
SAGA GIS 6.7.1	14/05/2025 15:54	Collegamento	2 KB

6 elementi 1 elemento selezionato 376 byte

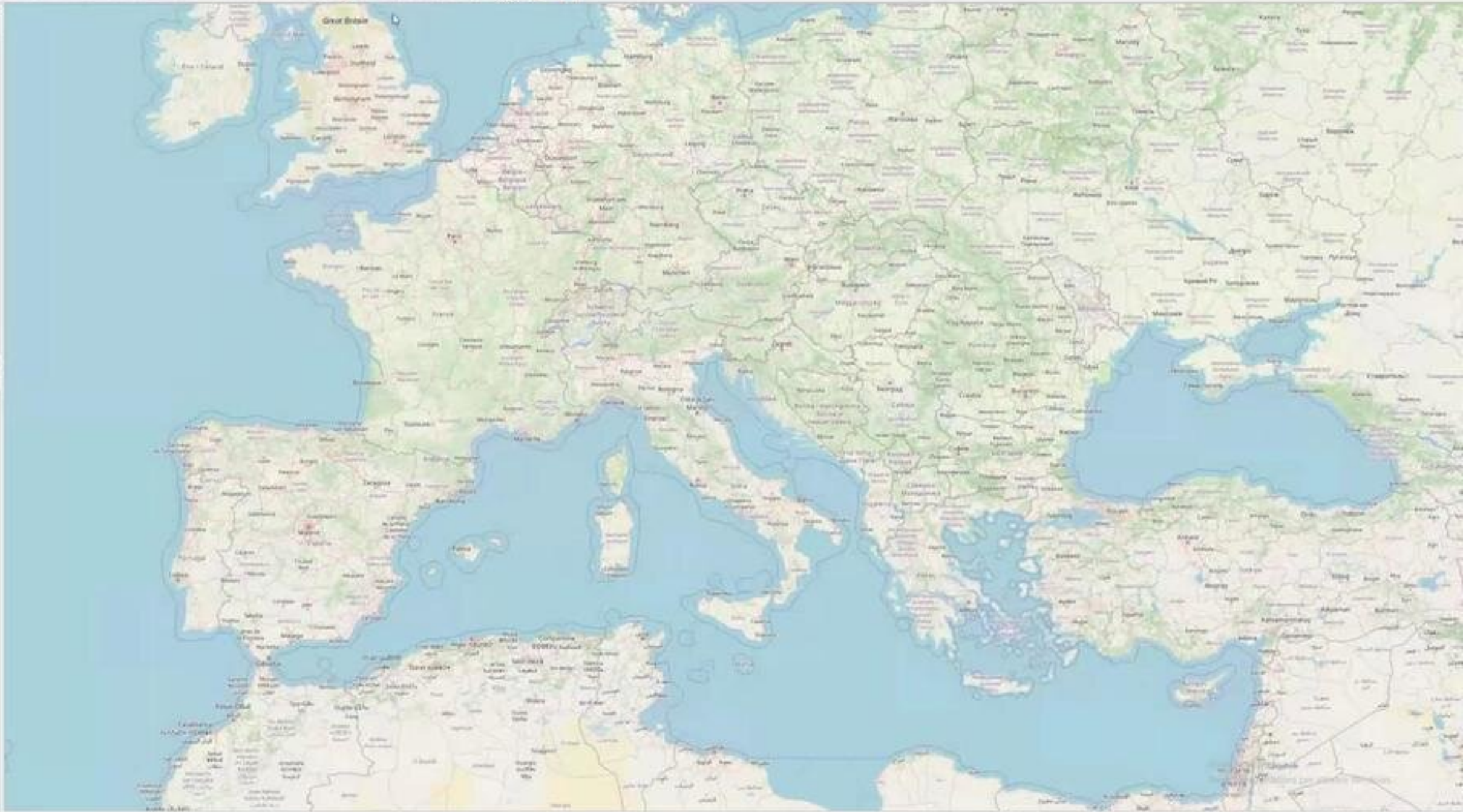
Attiva Windows
Recupera le impostazioni per attivare Windows.





- Browser
- Profilo
 - Segnalibri Spaziali
 - Home
 - OL
 - GeoPackage
 - SpazioLibre
 - PostgreSQL
 - SAP HANA
 - ESRI
 - MS SQL Server
 - Oracle
 - WMS/WMTS
 - Cloud
 - Scenes
 - SensorThings
 - Vector Tiles
 - XYZ Tiles
 - Mapbox Global Terrain
 - OpenStreetMap
 - WCS
 - WFS / OGC API - Features
 - ArcGIS REST Servers

- Layer
- ✓ OpenStreetMap



1 Fase (8 giornate)

1. Definizione del diagramma E/R e delle specifiche relative per la creazione dell'infrastruttura DB (1 giornata) – in sede ISPRA (già realizzata)
2. Installazione di PostGIS, definizione di un ambiente di lavoro, creazione delle tabelle ritenute necessarie per il progetto. Test funzionale. (1/2 giornata)
3. Installazione di Geoserver, definizione dei layer e integrazione delle tabelle. Test funzionale. (1/2 giornata)
4. Importazione dei dati da file Shape o altri db georeferenziati per la costruzione degli elementi verdi e blu (1 e ½ giornate)
5. Importazione dei dati della popolazione su raster fornito da ISPRA (1 e ½ giornate)
6. Installazione ambiente di lavoro usando Eclipse, definizione classi per accesso DB (2 giornate)
7. Confronto su sviluppo del geodatabase - (1 giornata) - in sede ISPRA

2 fase (24 giornate)

1. Calcolo dei punti di ingresso delle aree verdi e blu (4 giornate)
2. Calcolo della distanza minima fra i centroidi dei poligoni rappresentanti la popolazione e i punti di ingresso delle aree verdi e blu (2 giornate)
3. Stesura del codice per la visualizzazione degli elementi su mappa georeferenziata interattiva (10 giornate)
4. Confronto su attività di sviluppo parte client – (2 giornate) – in sede ISPRA
5. Sviluppo maschere per esportazione dati su formati definiti nella fase di analisi (2 giornate)
6. Sviluppo di una applicazione per la visualizzazione dei dati (2 giornate)

Training on the Job - Modulo 2

Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali)

Importazione dei dati da file Shapefile o altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

altri DB georeferenziati

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

Importazione dati

- Uso di shp2pgsql per esportare .shp in script SQL

Durata prevista: 1,5 giornate

Obiettivi

- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

Importazione dati

- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

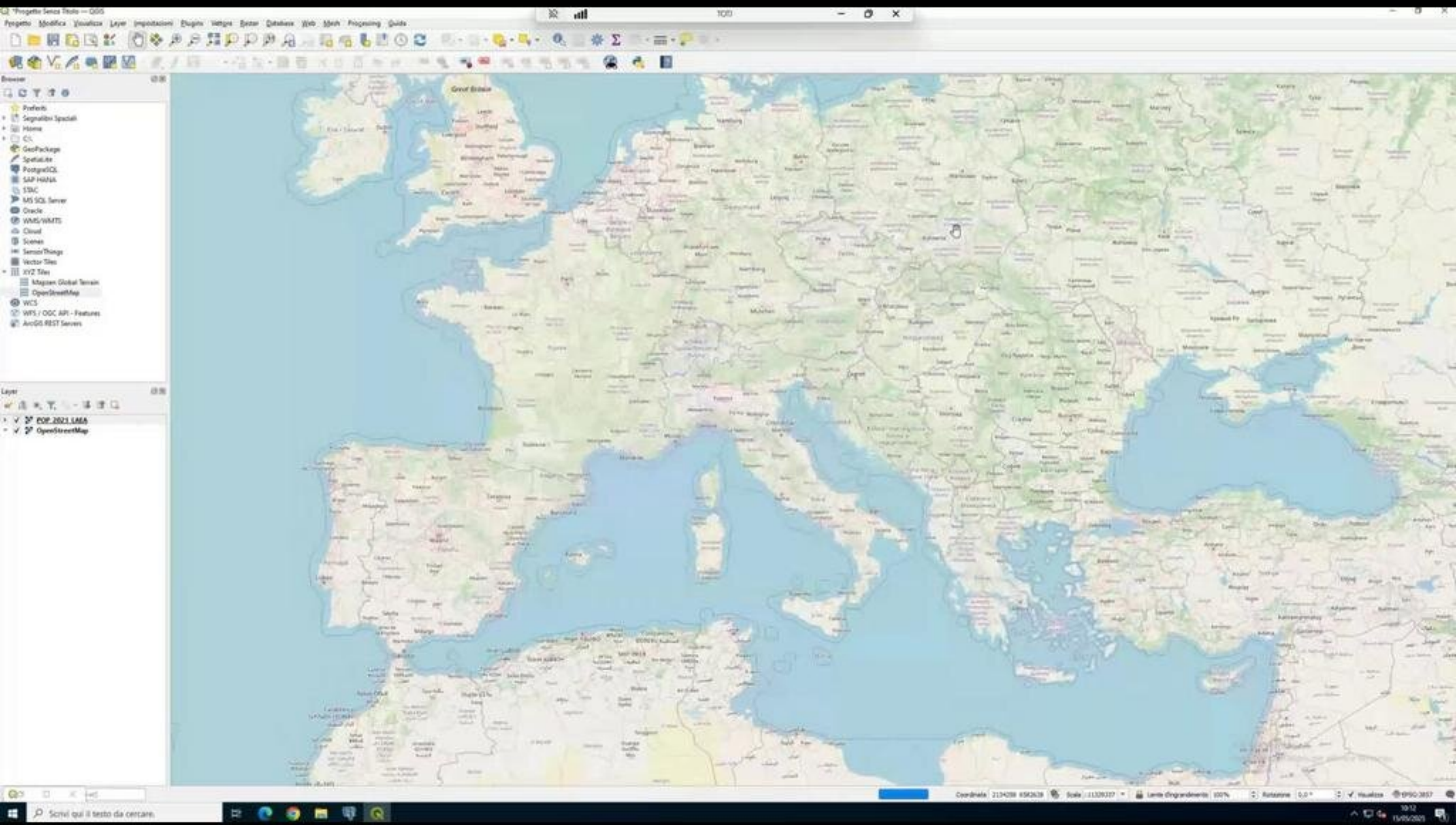
- Comprendere formati e strutture dei dati vettoriali georeferenziati (Shapefile, GeoPackage, PostGIS, ecc.)
- Importare ed integrare layer "verdi" (es. aree verdi) e "blu" (es. aree blu) nel database spaziale
- Garantire corretto sistema di riferimento (SRS/EPSG) e metadata
- Comprendere quali aree verdi si importeranno

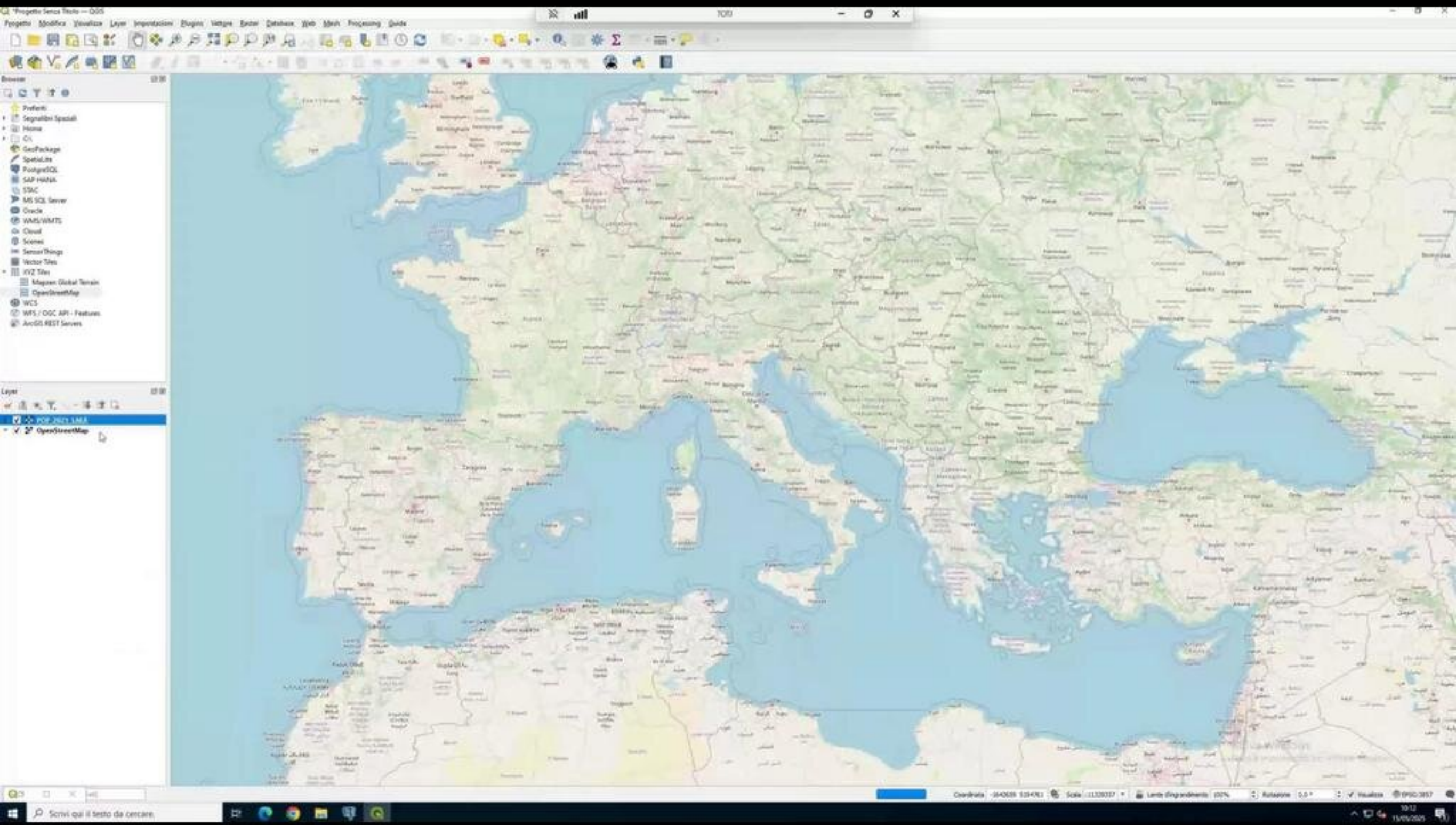
Analisi preliminare dei dati

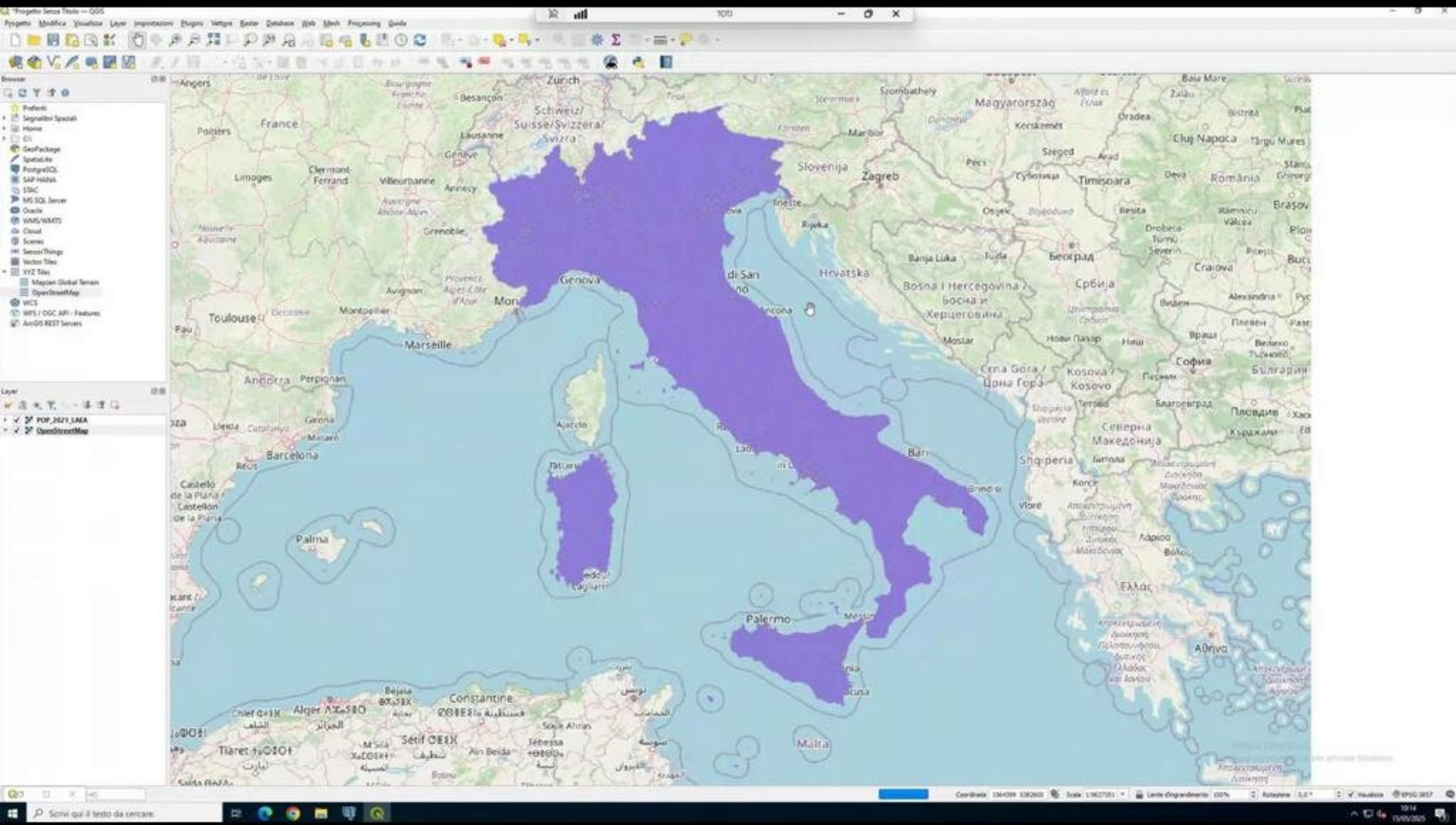
- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

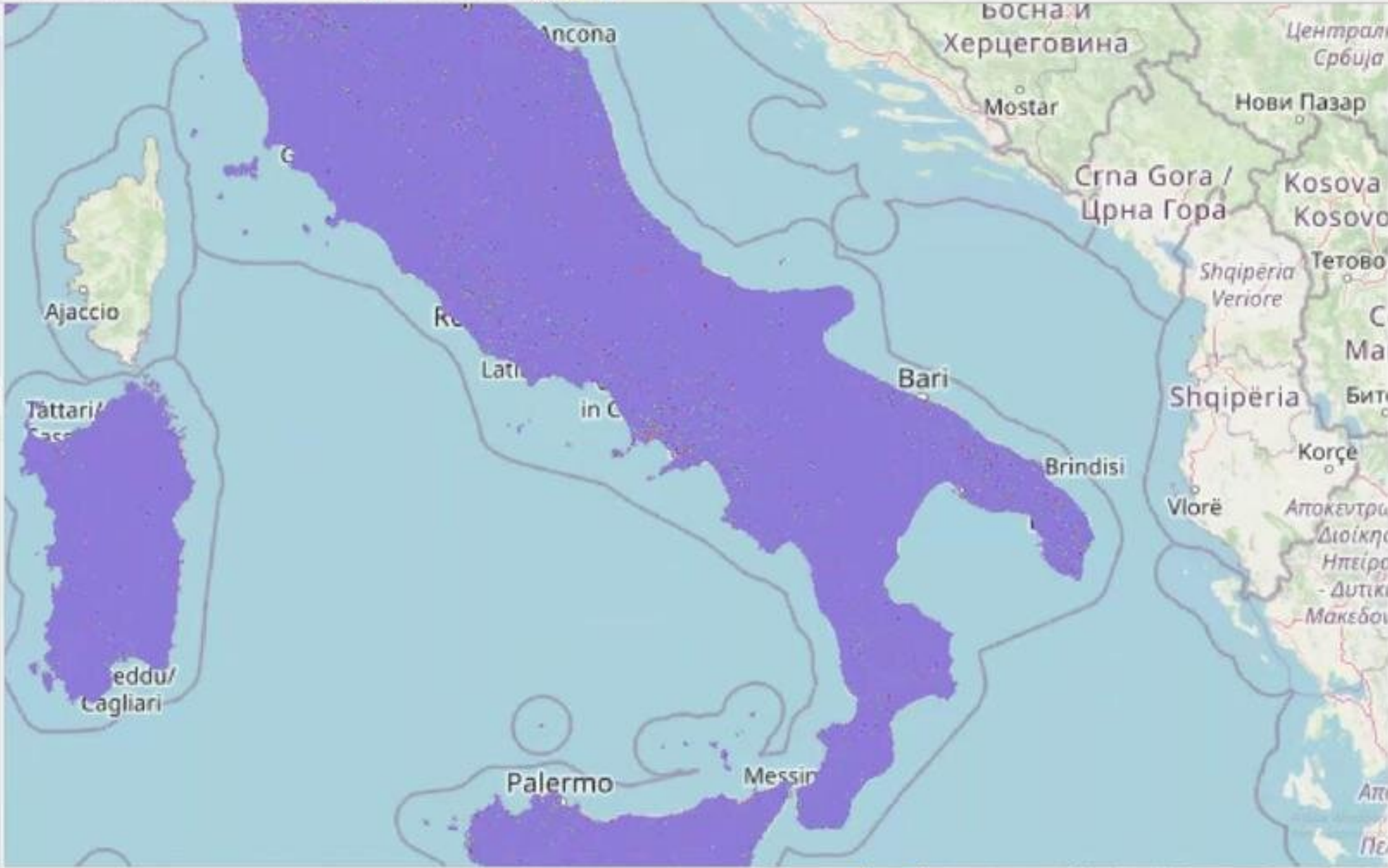
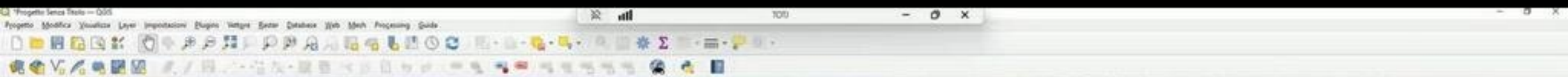
- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

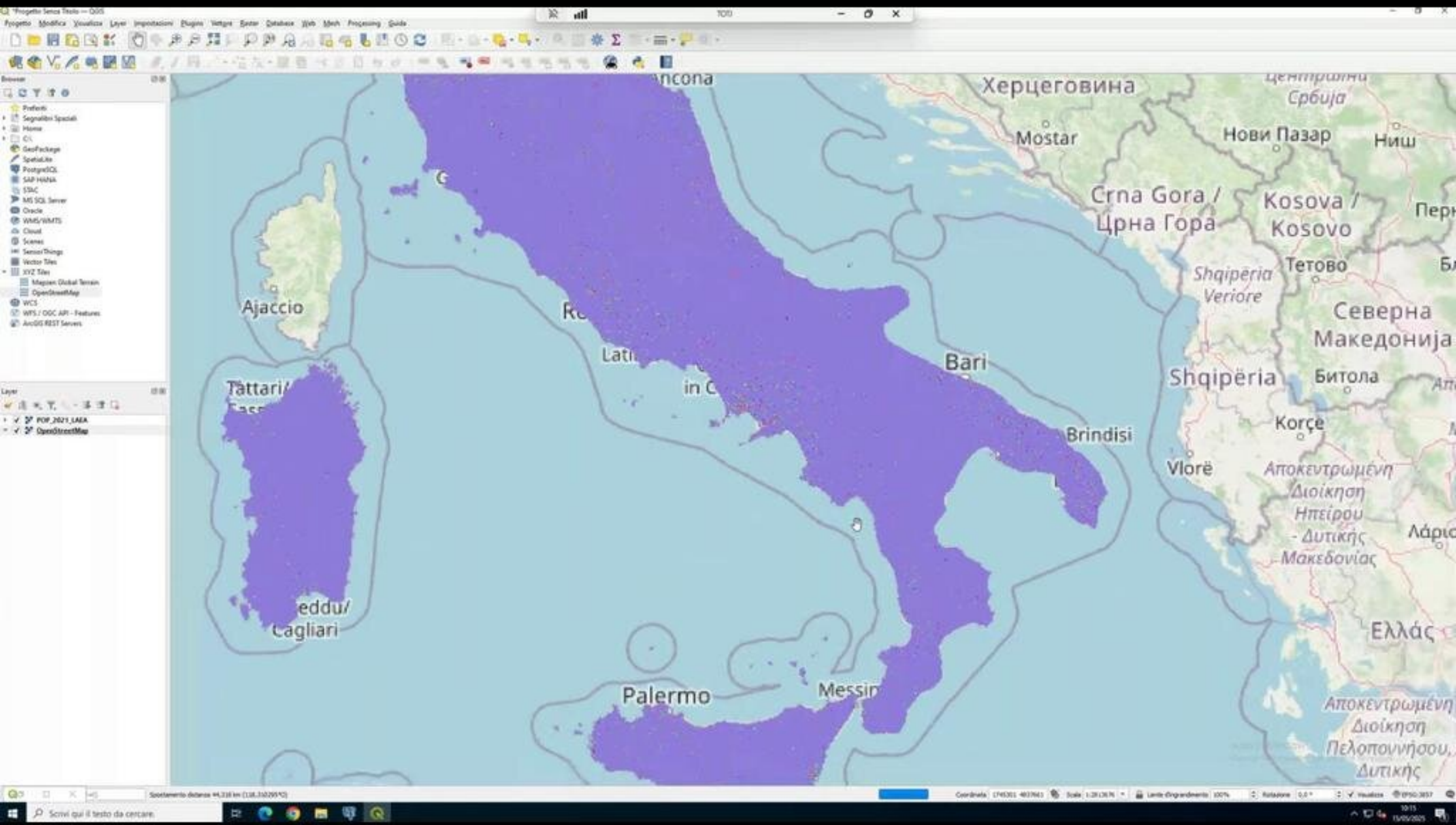
- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

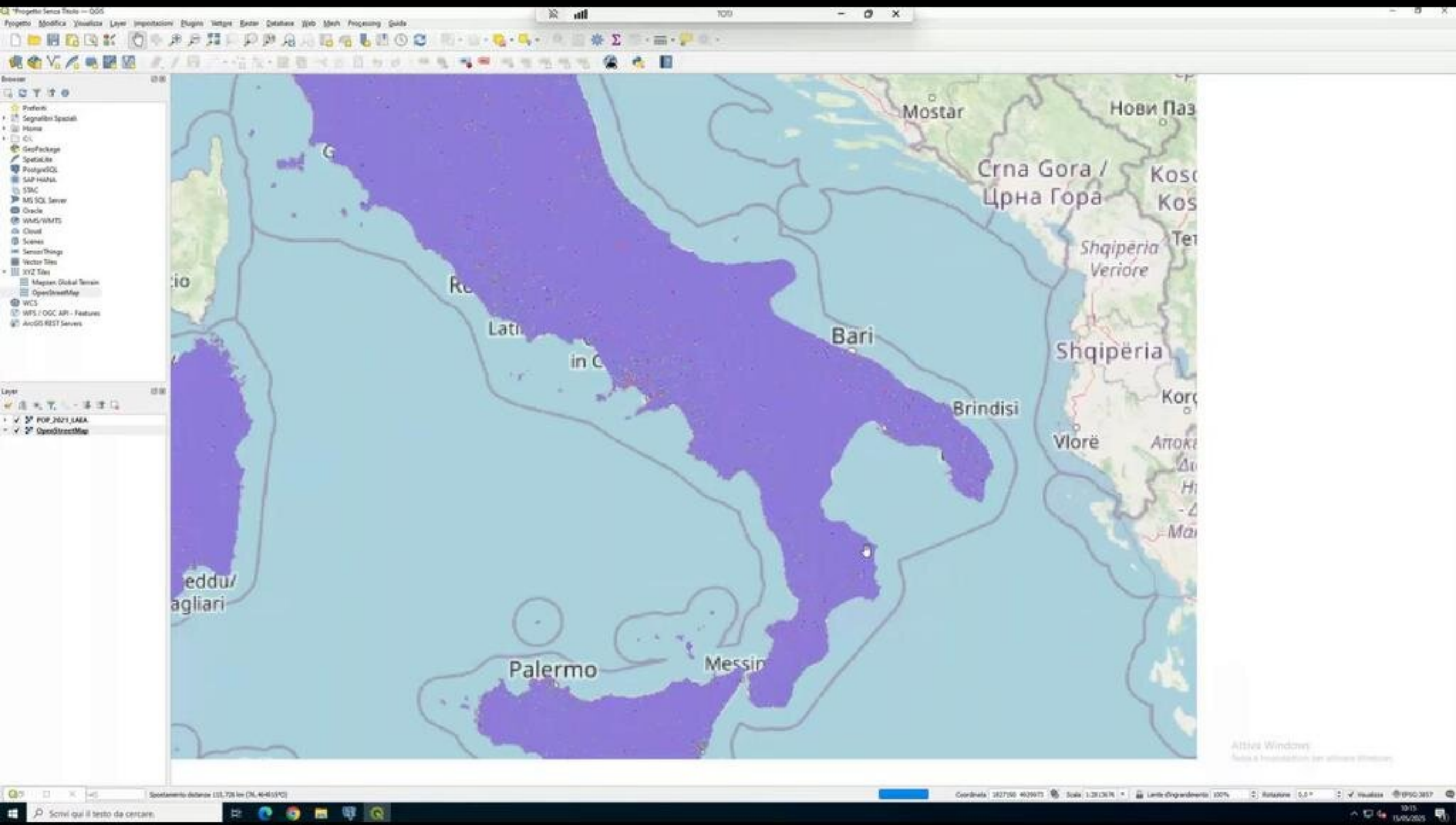


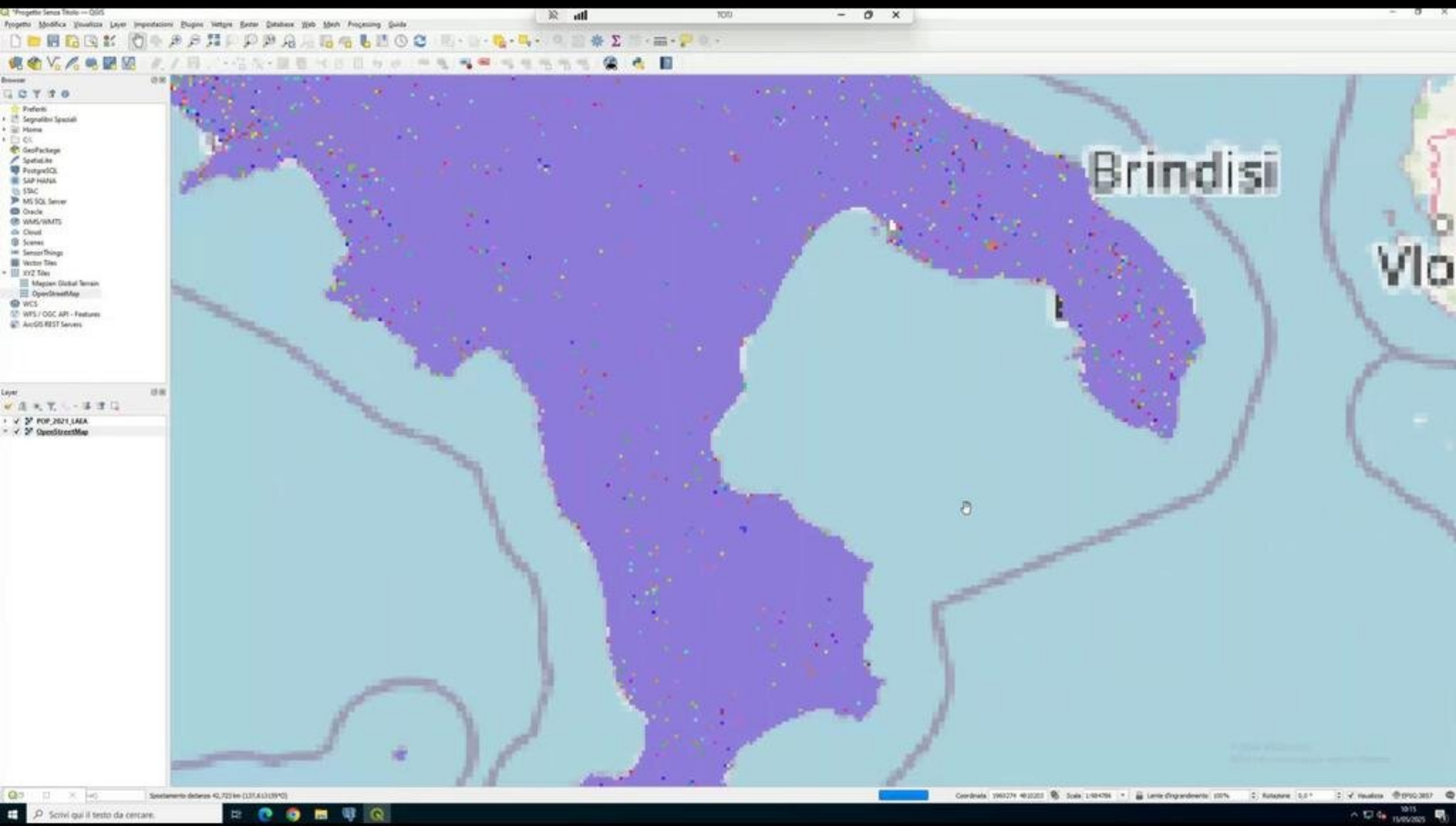


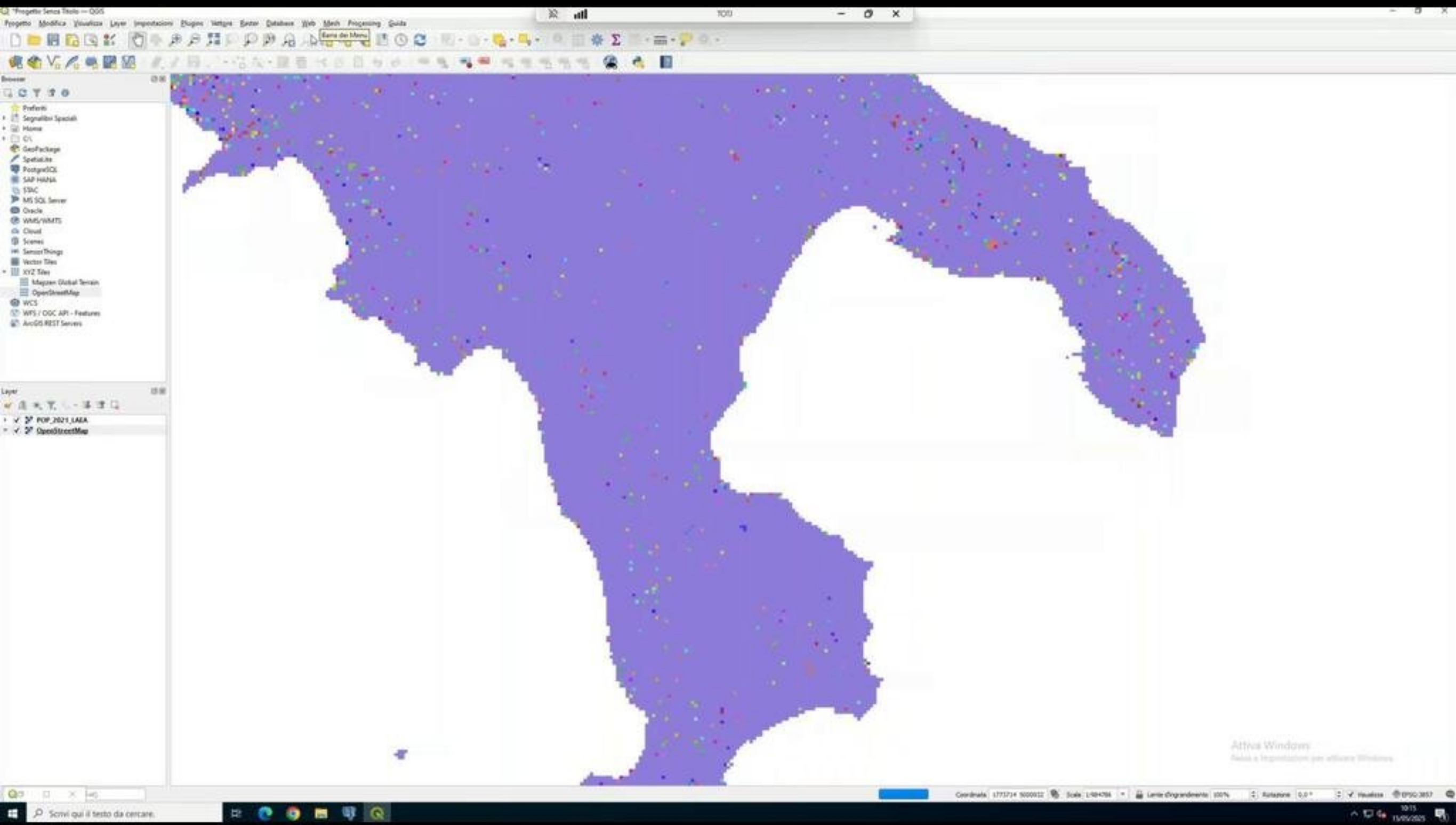


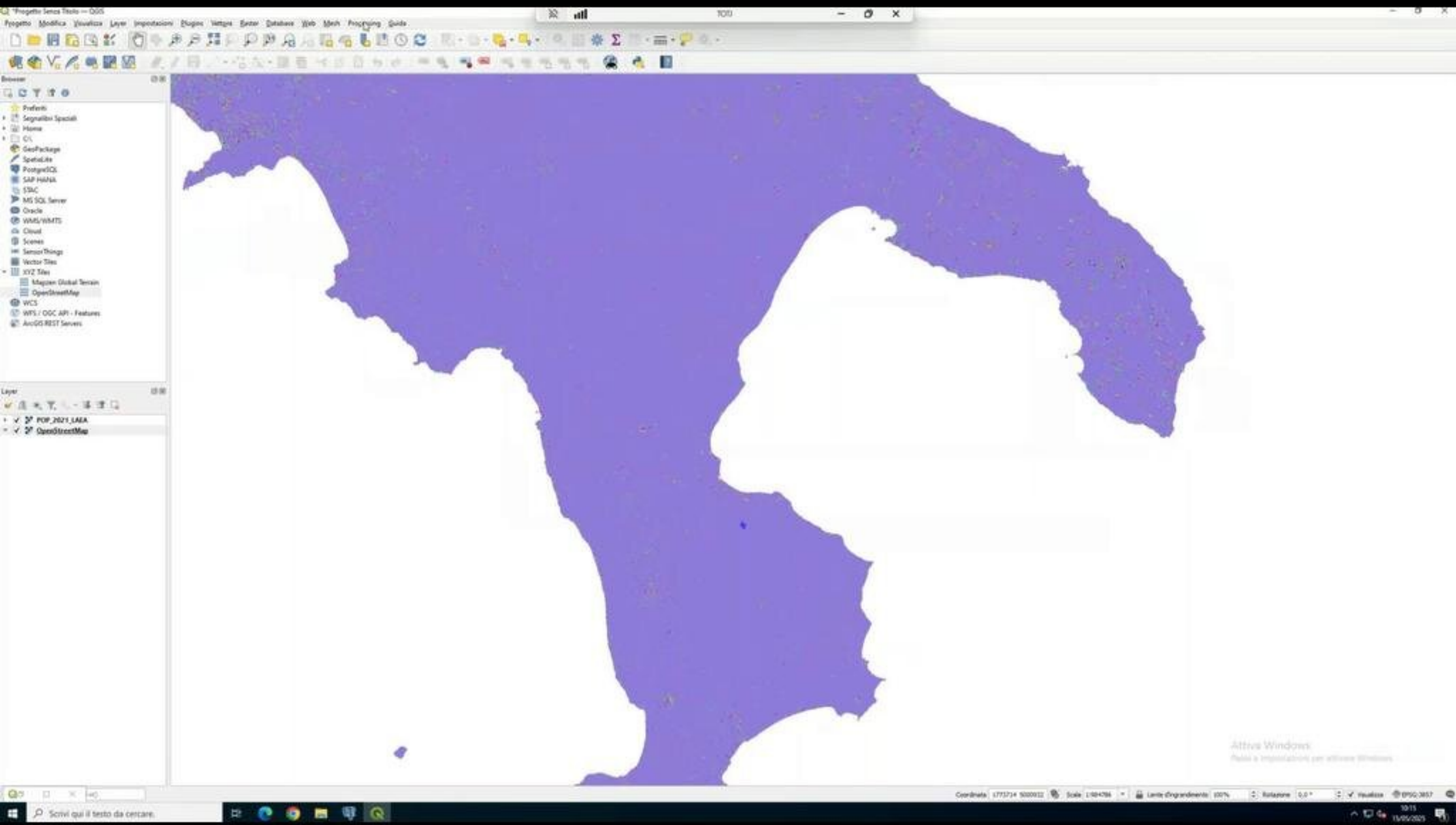


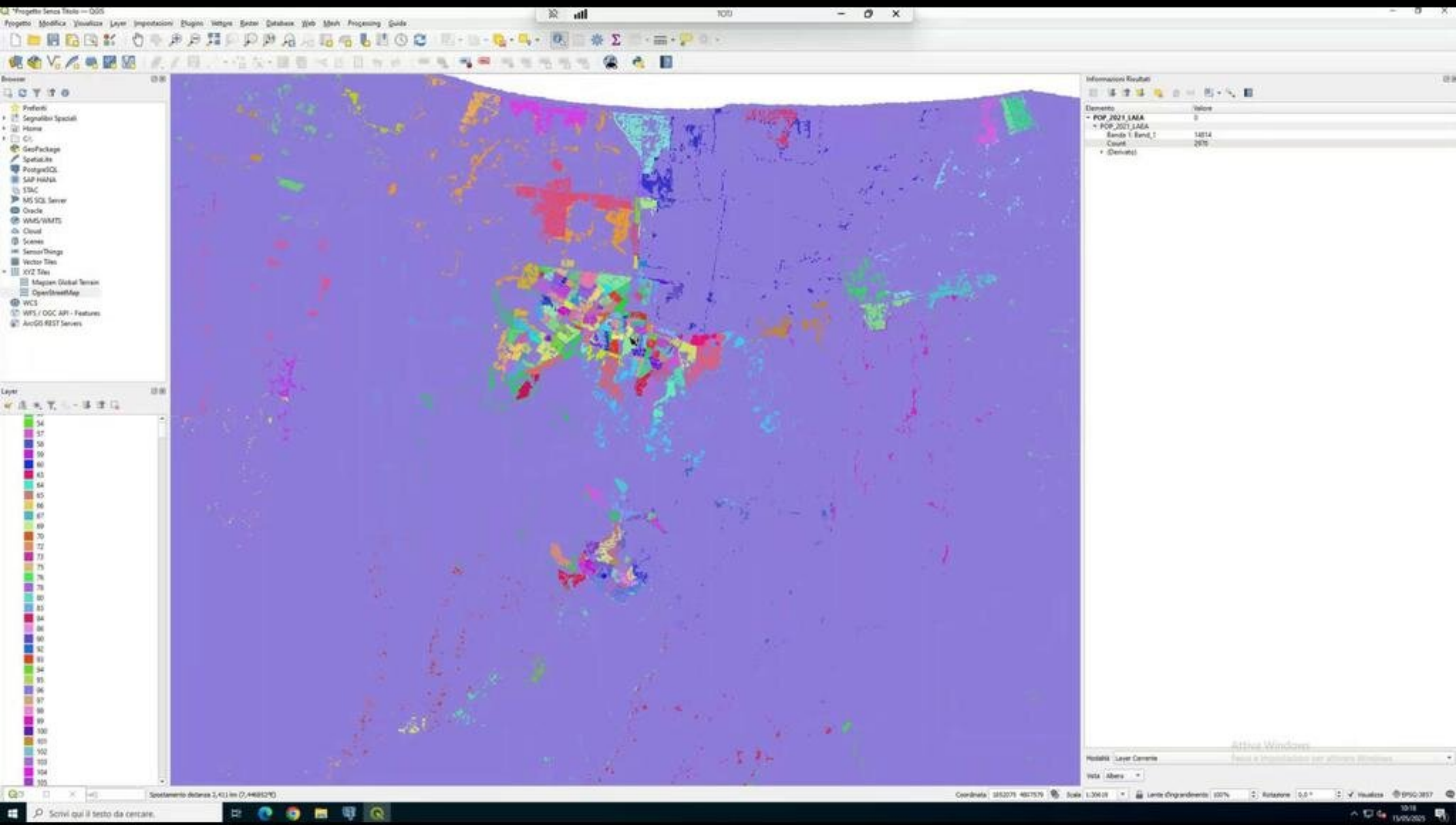


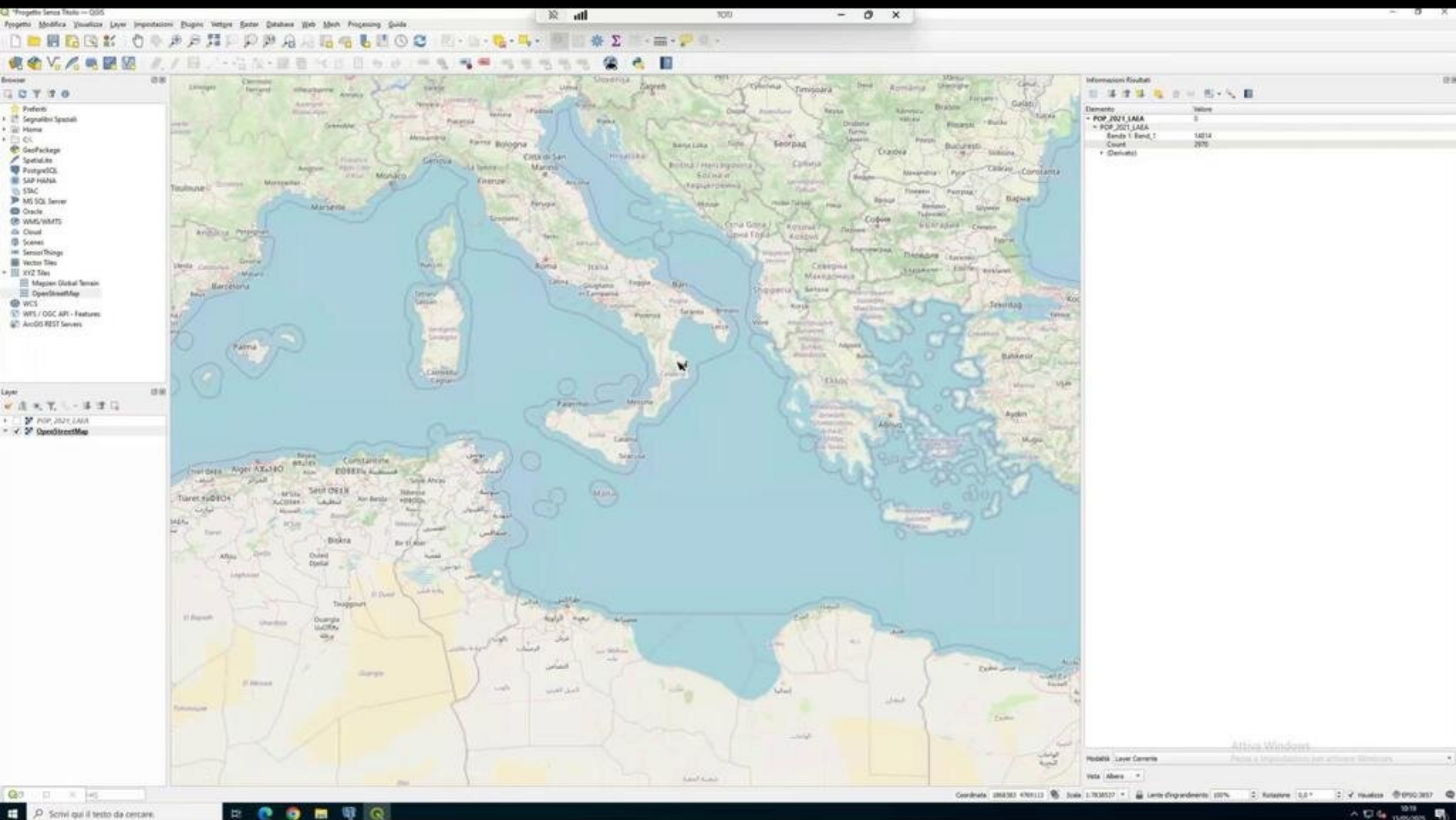


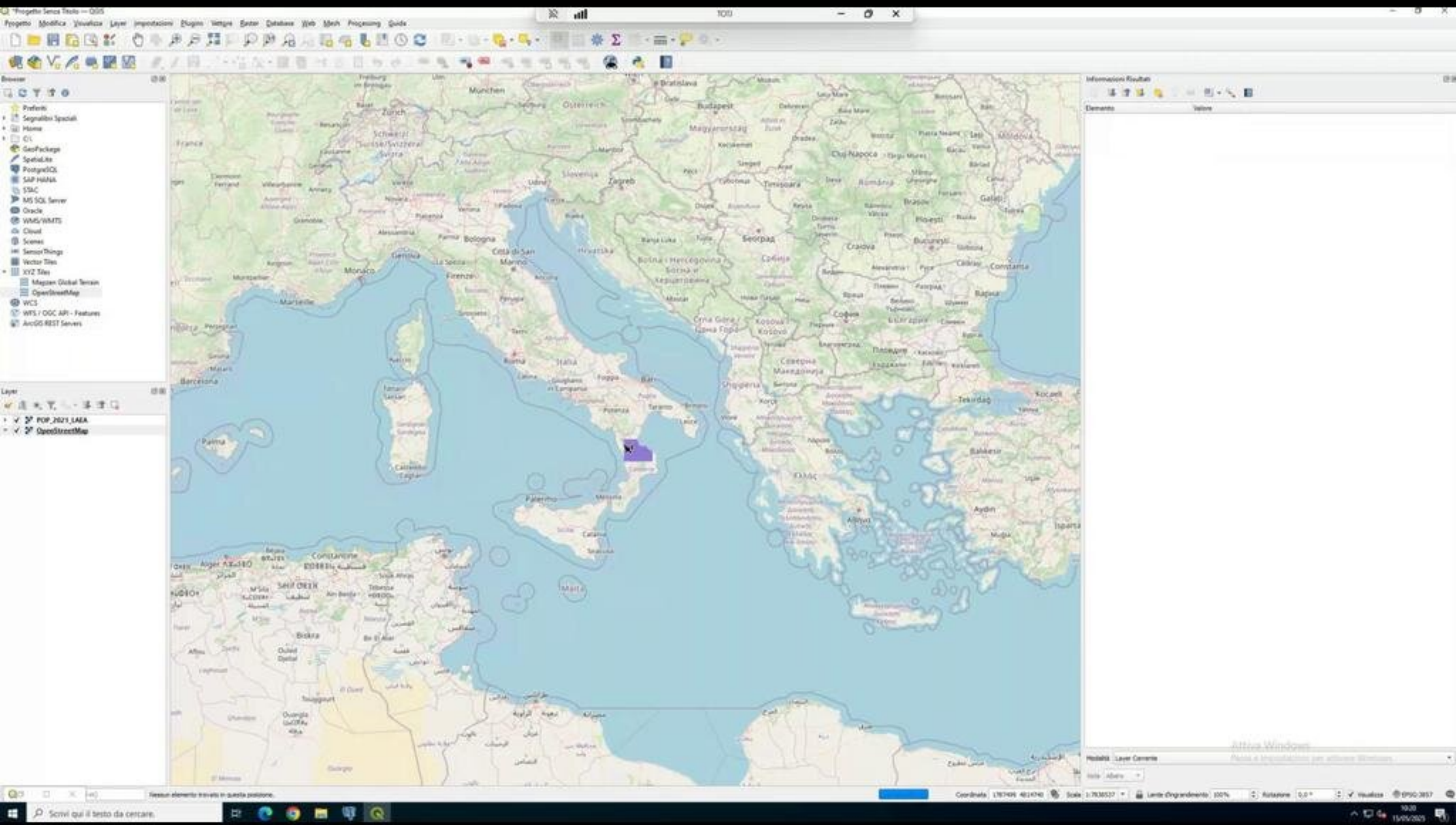












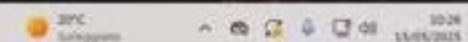
- ### Fasi operative

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

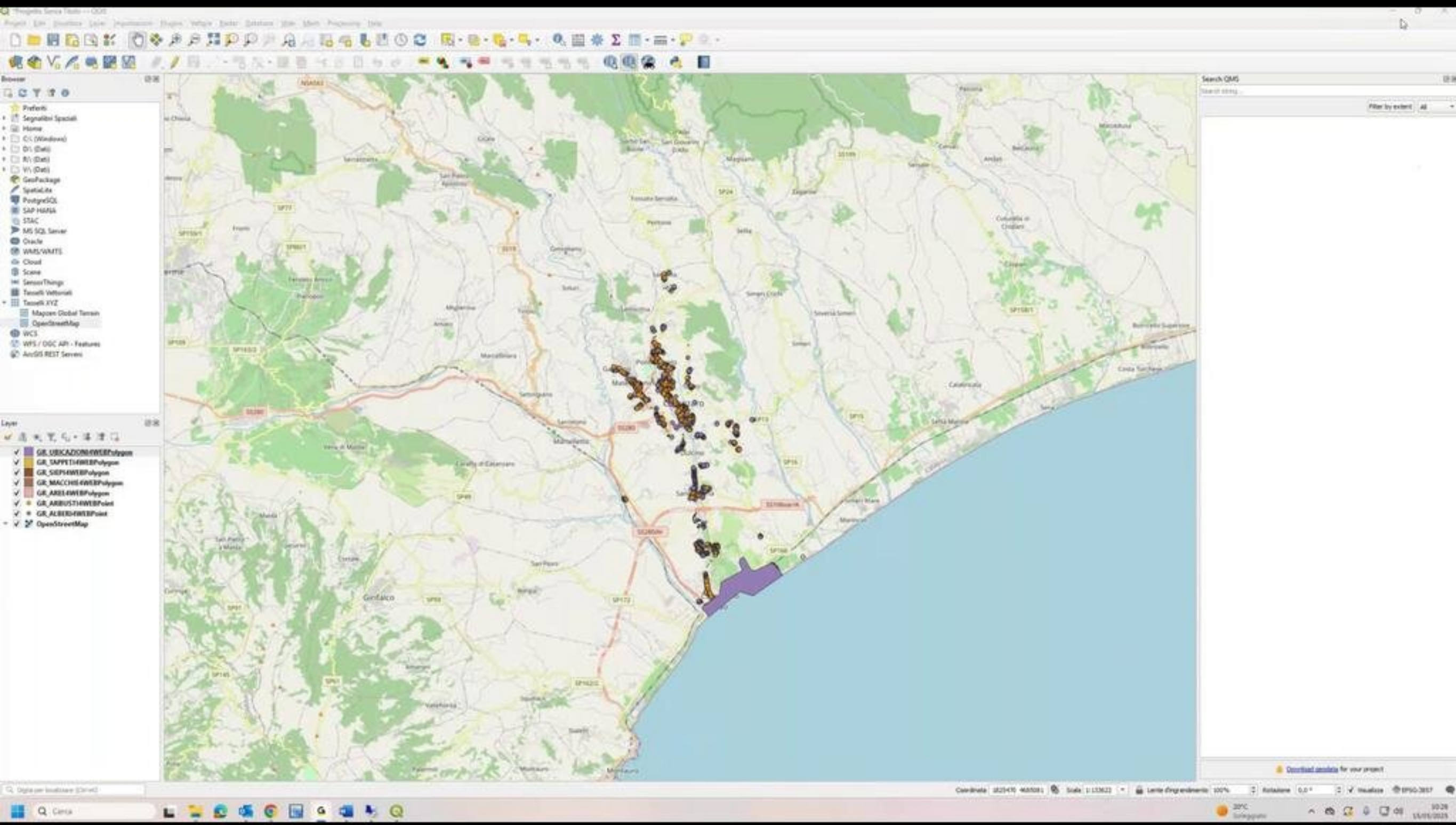
- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

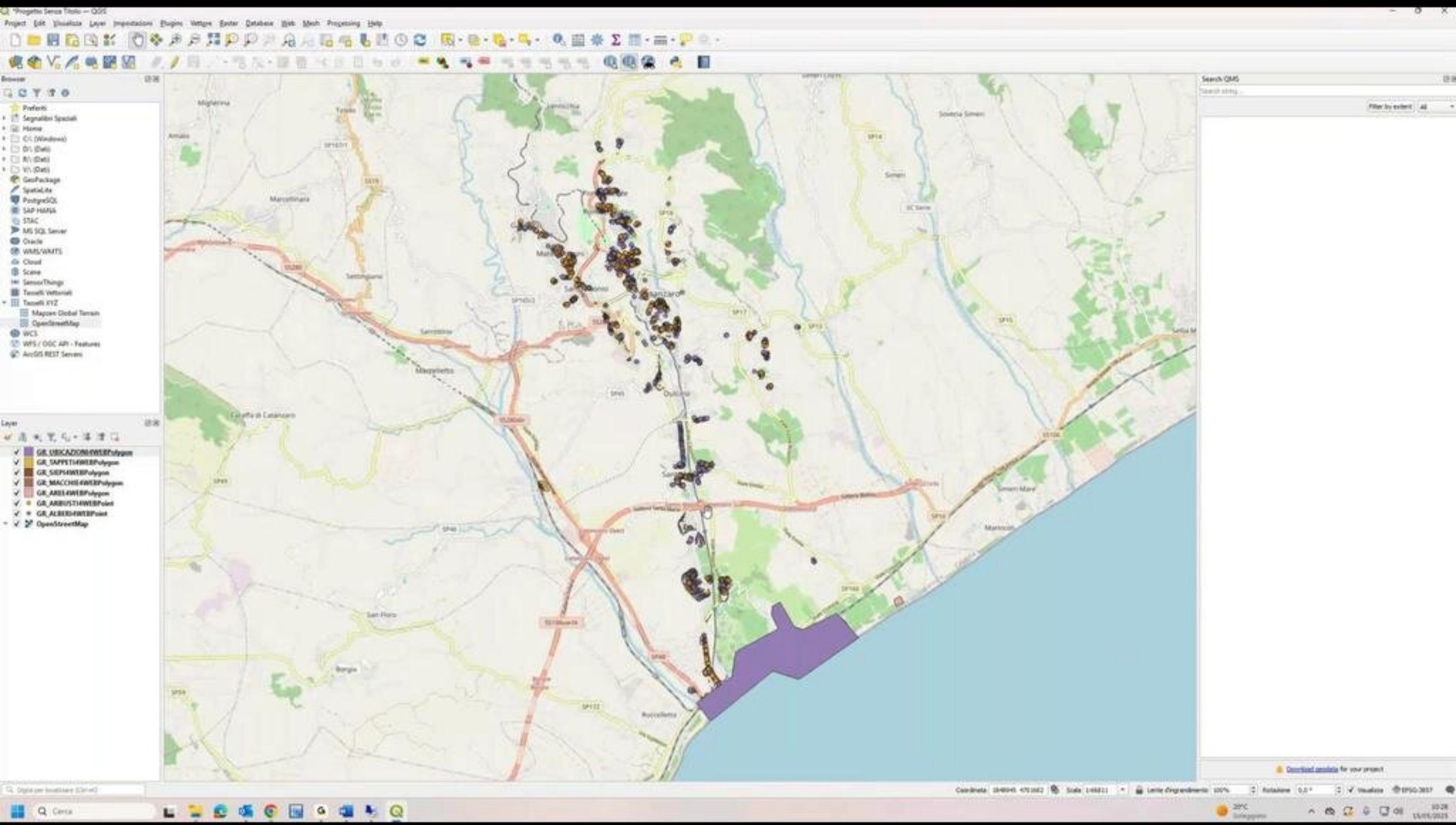
- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

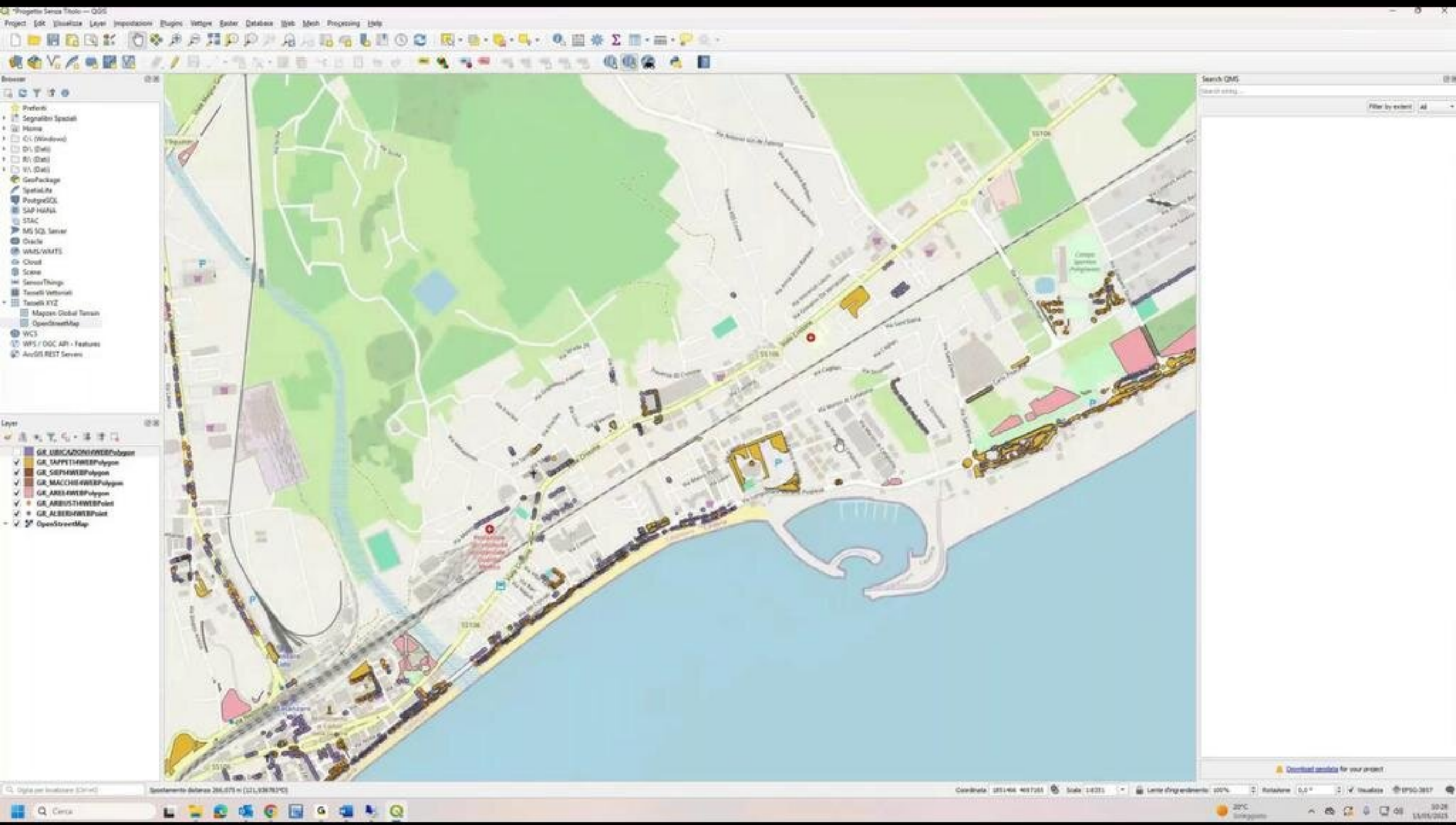
Fai clic destro sul layer che vuoi importare (es. uno shapefile)

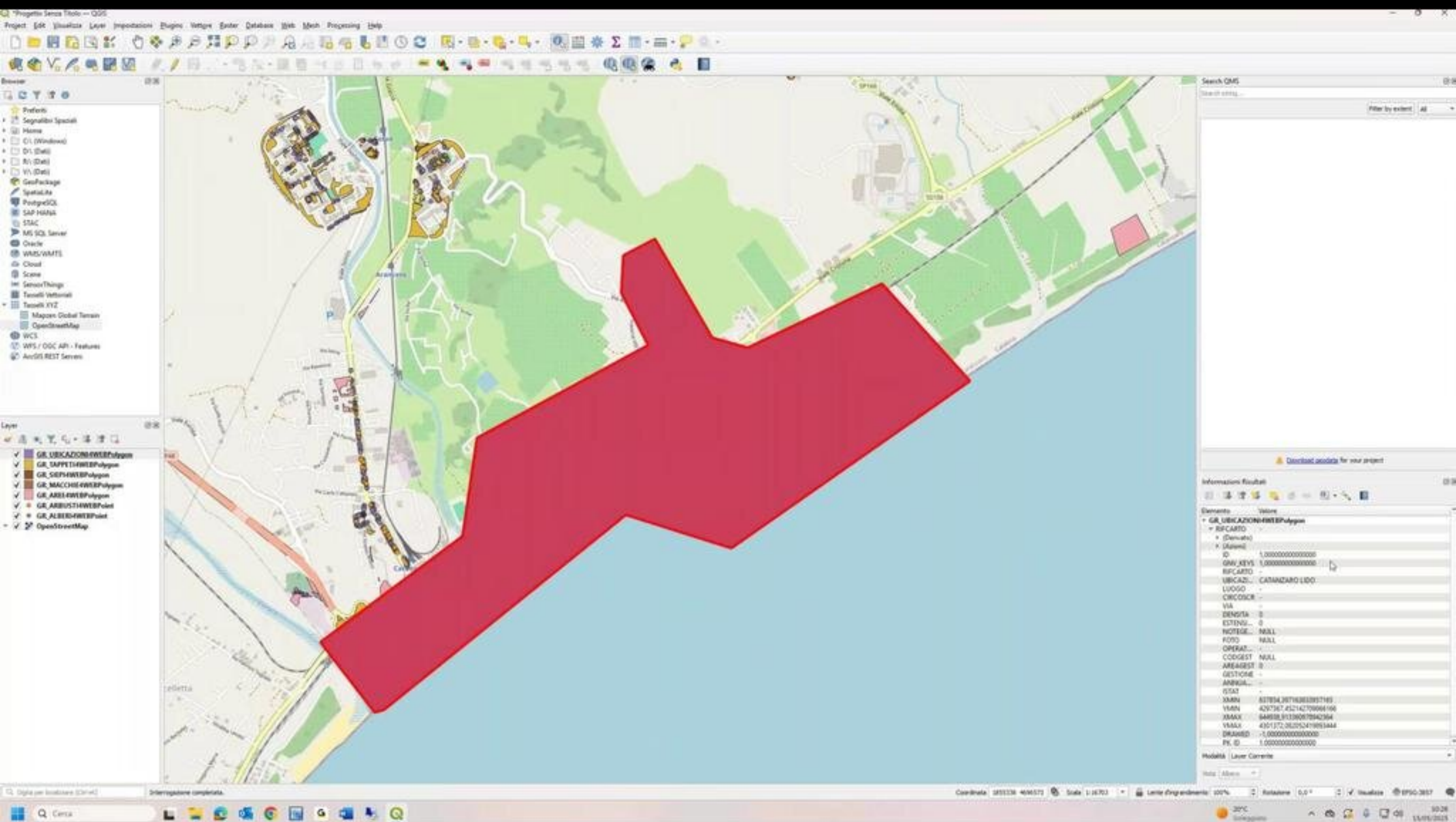


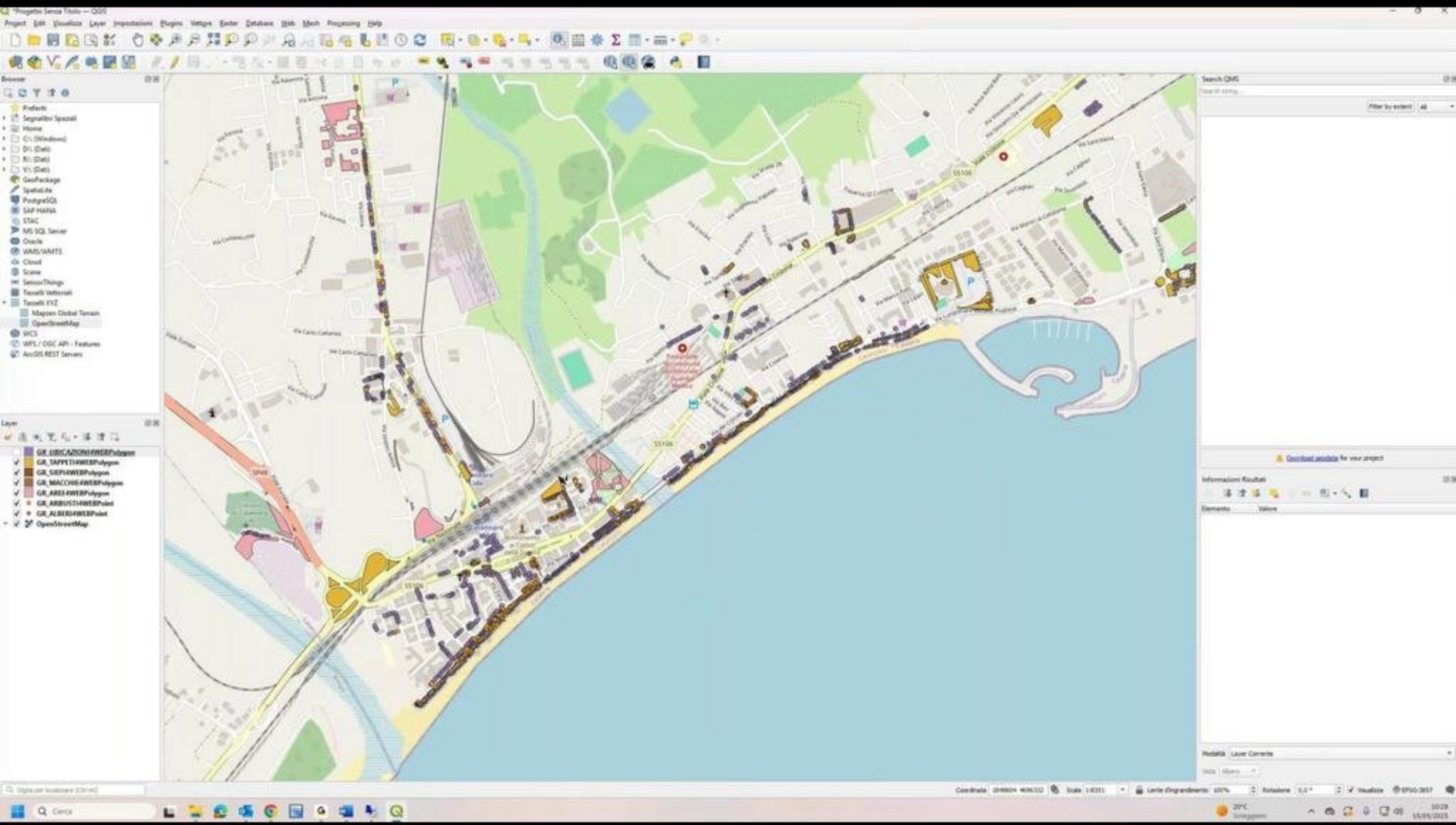


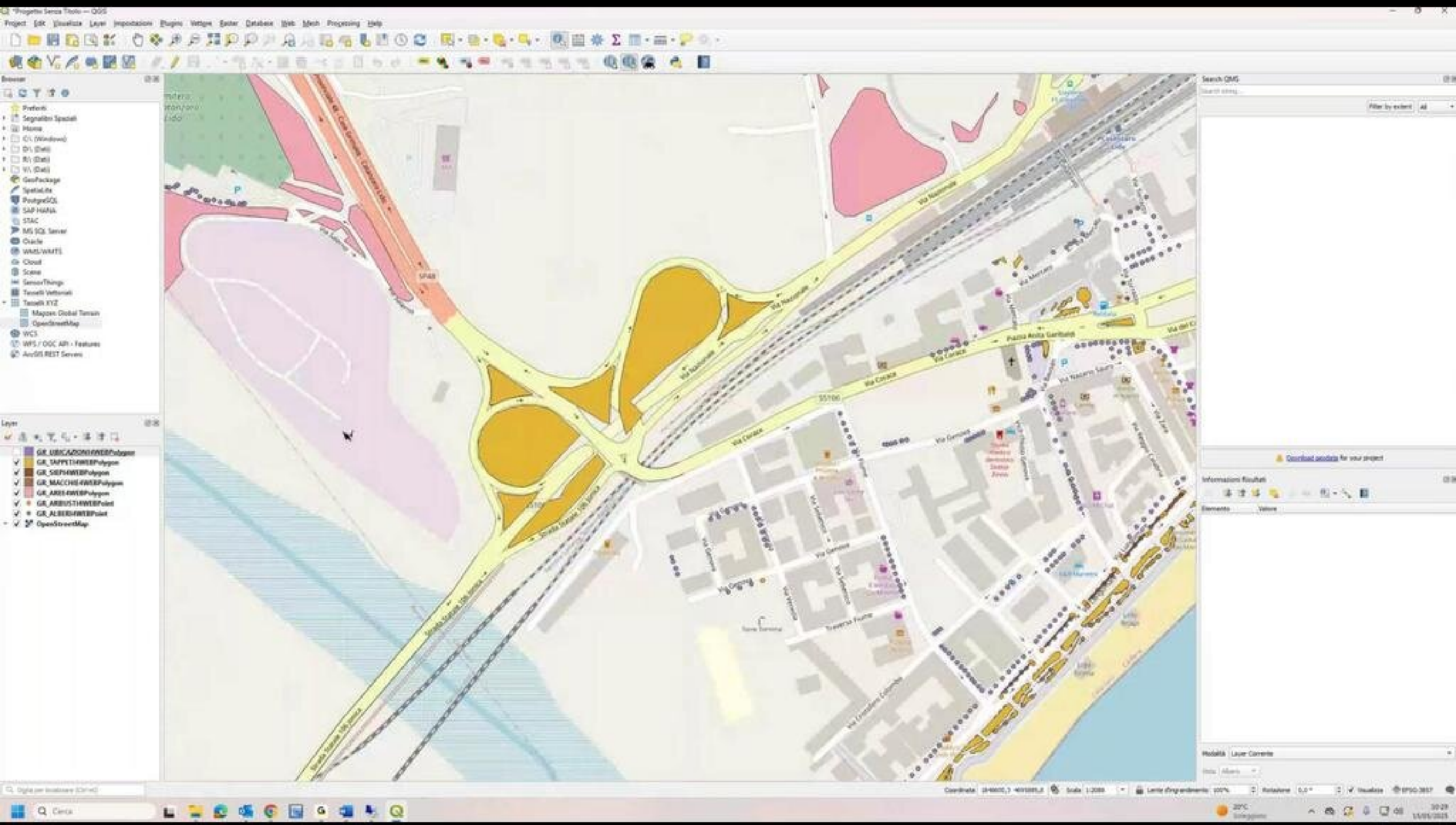


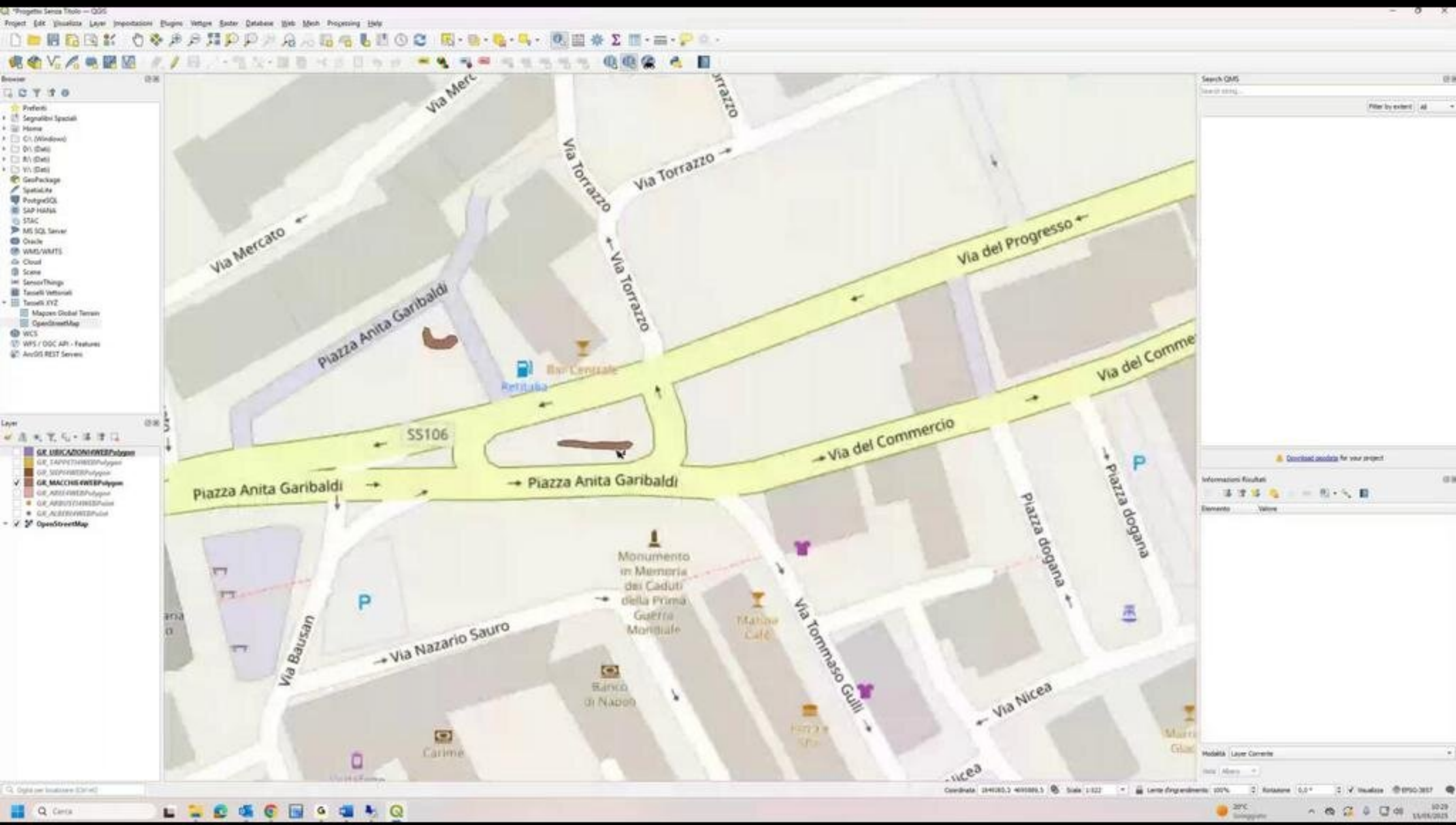


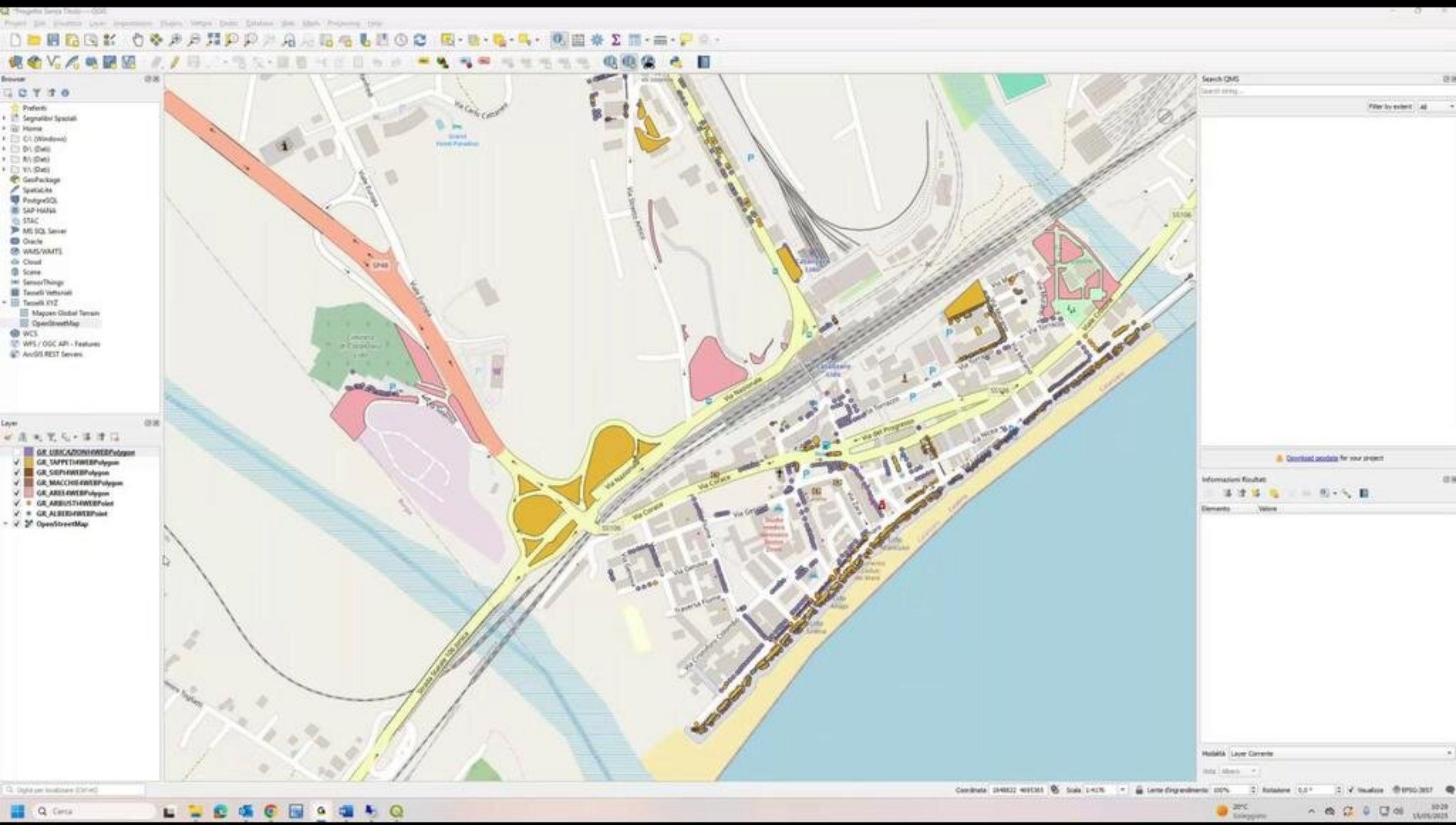


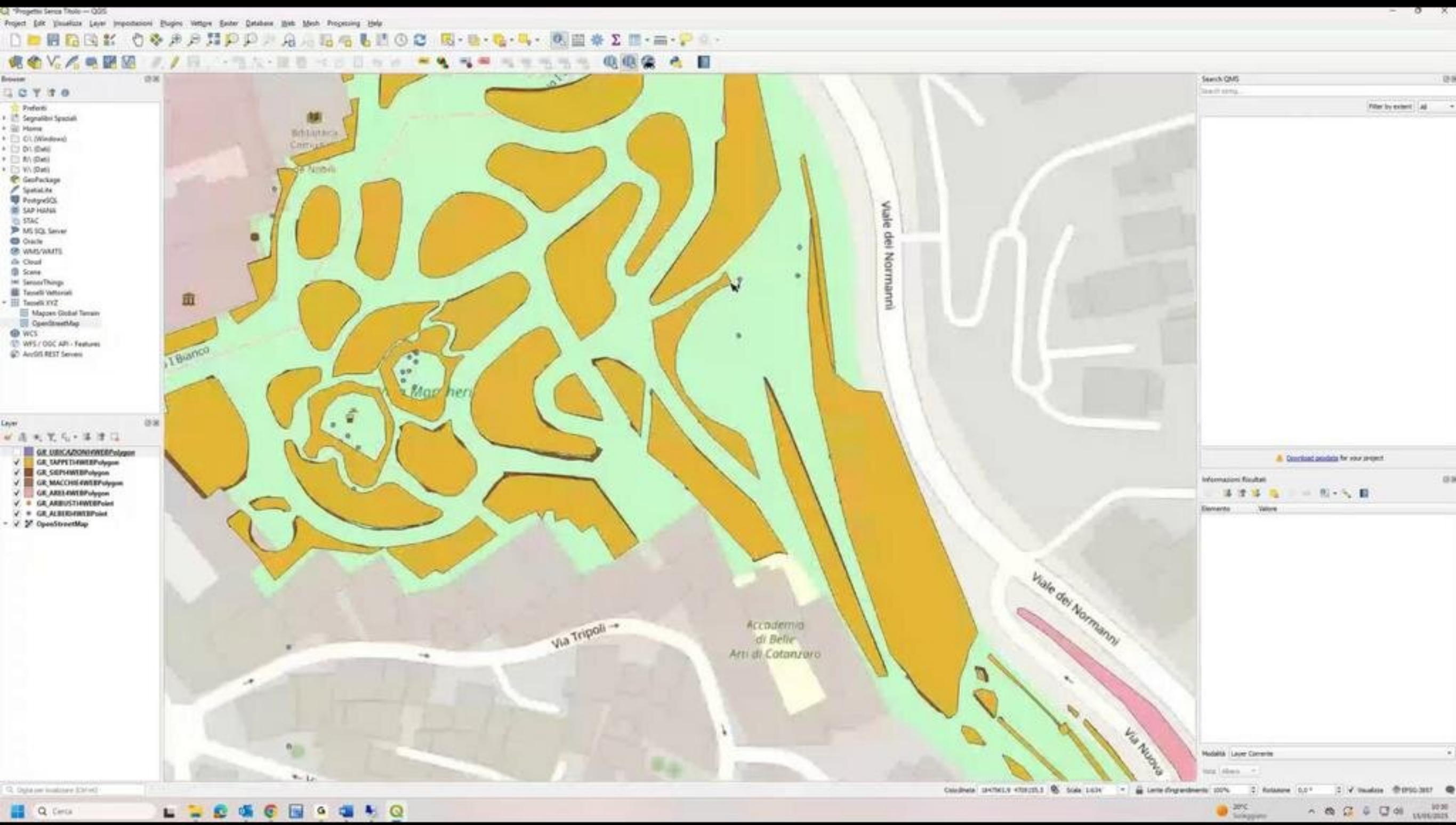


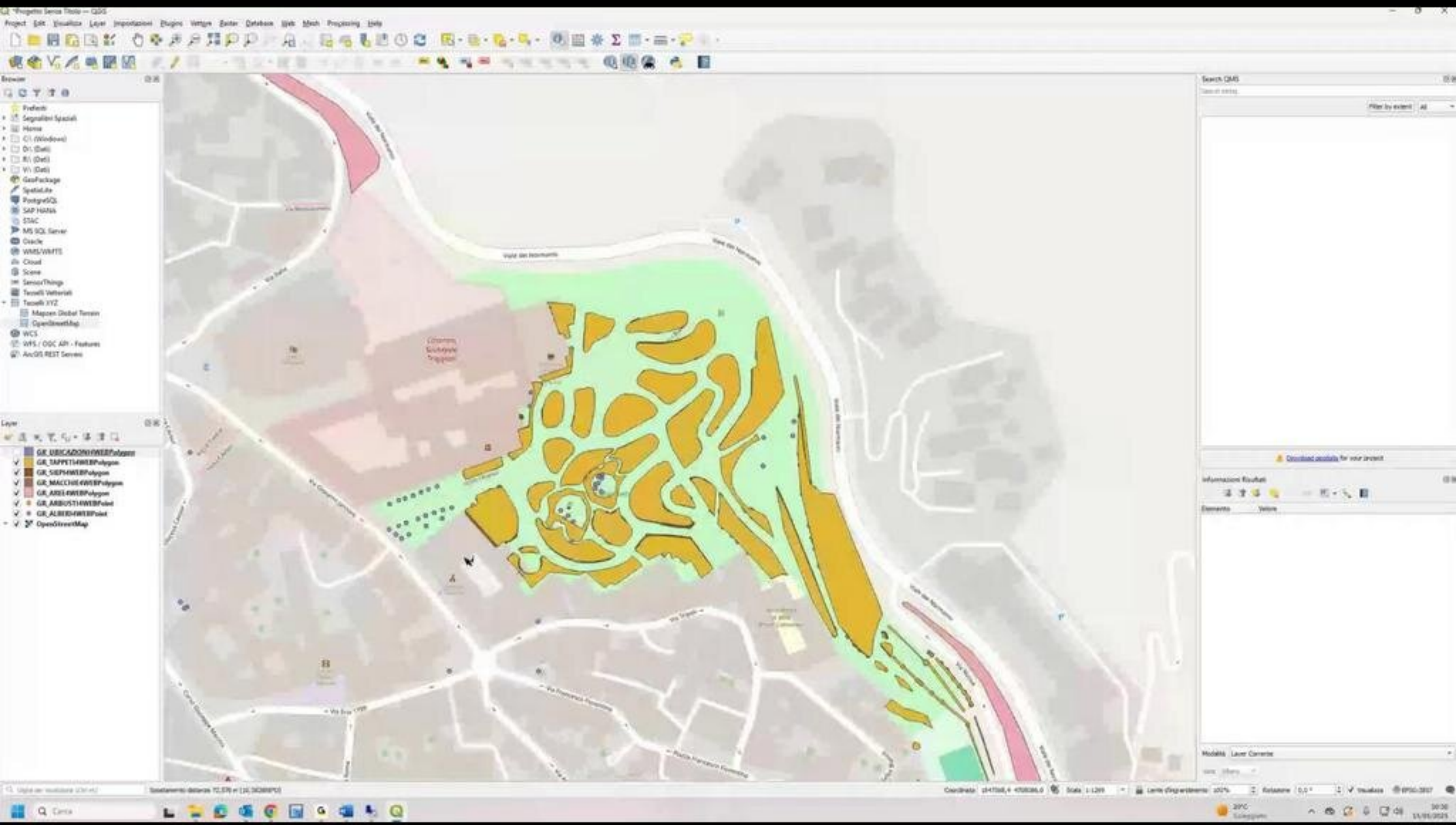


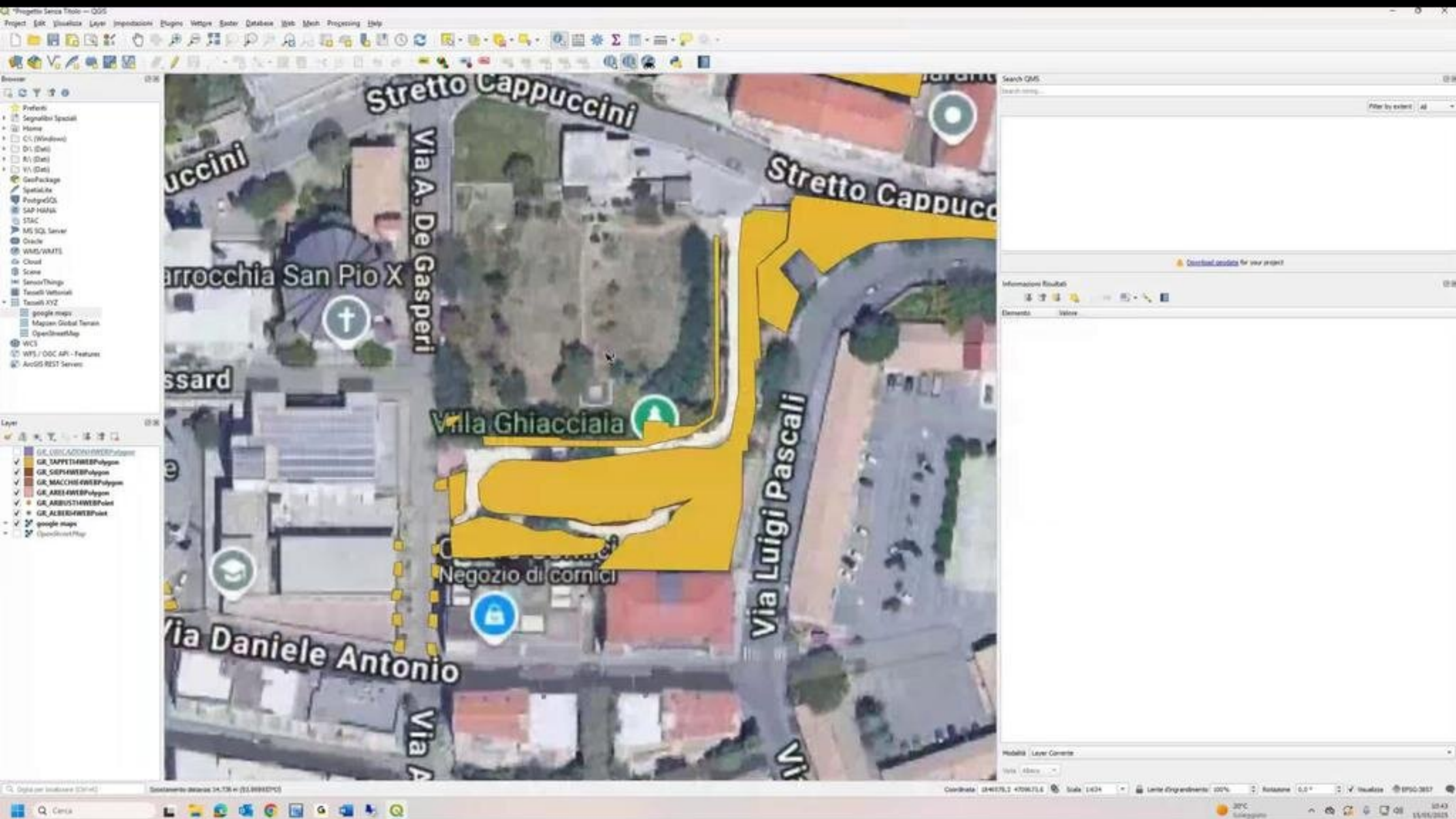


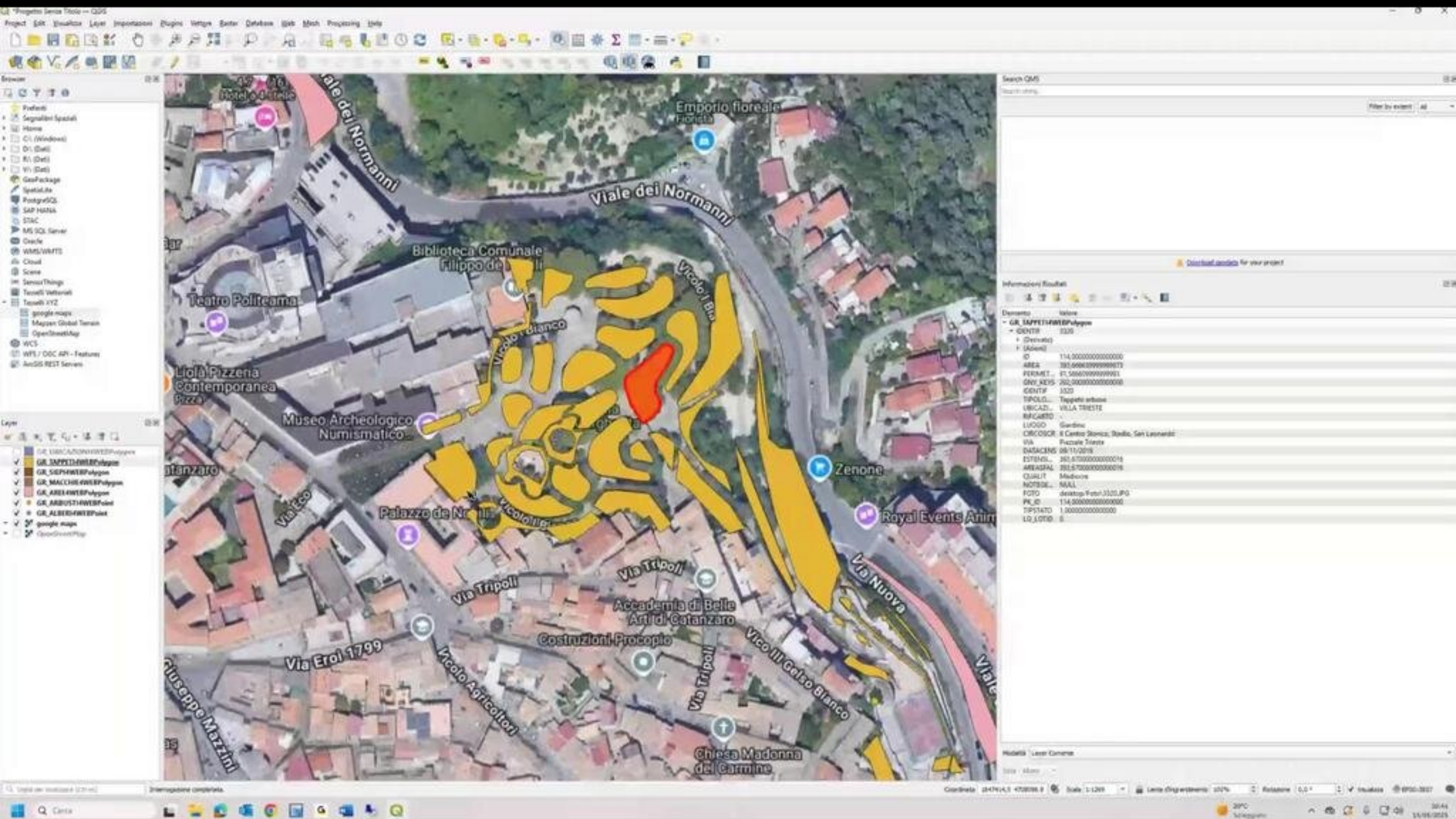


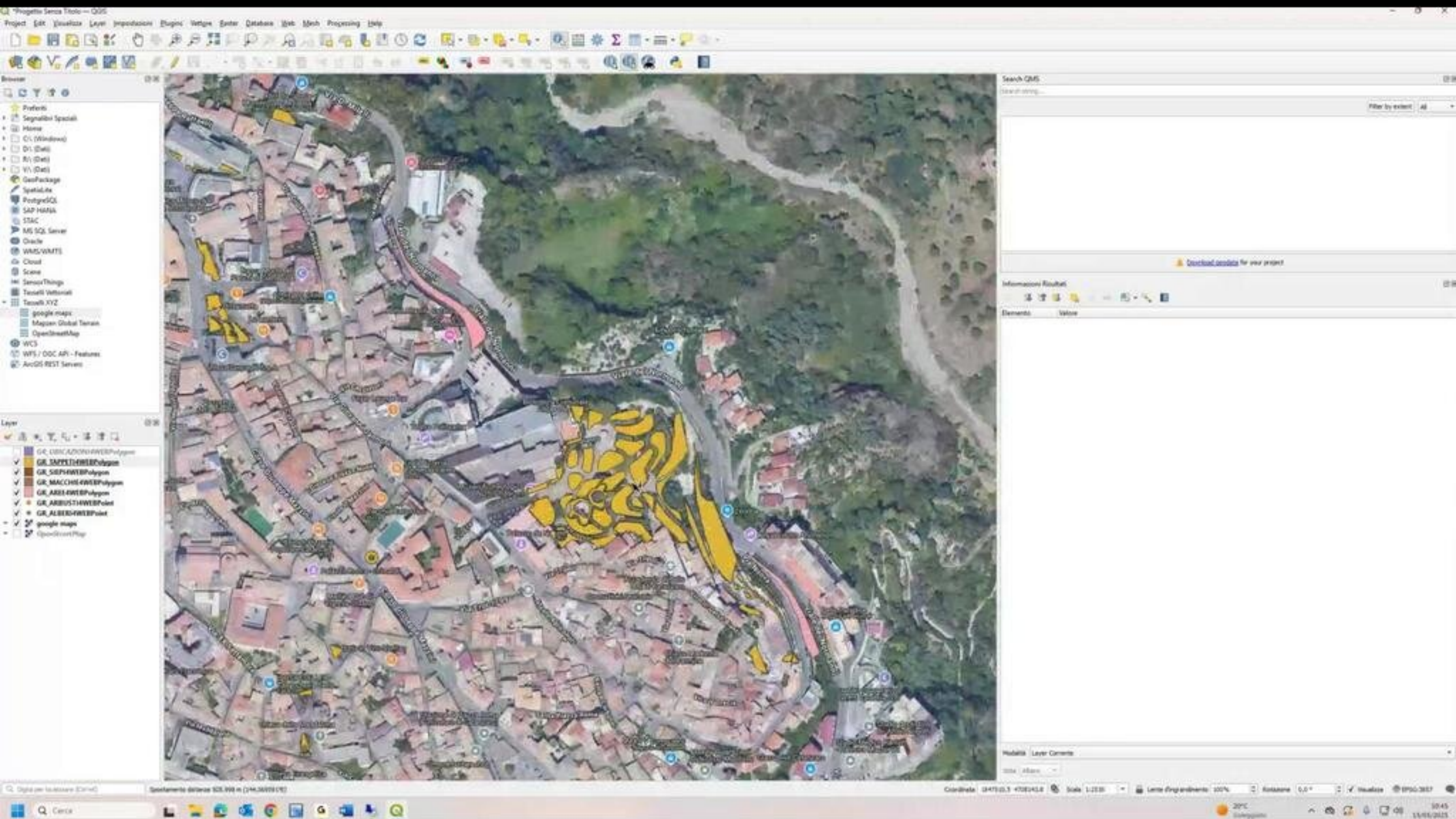


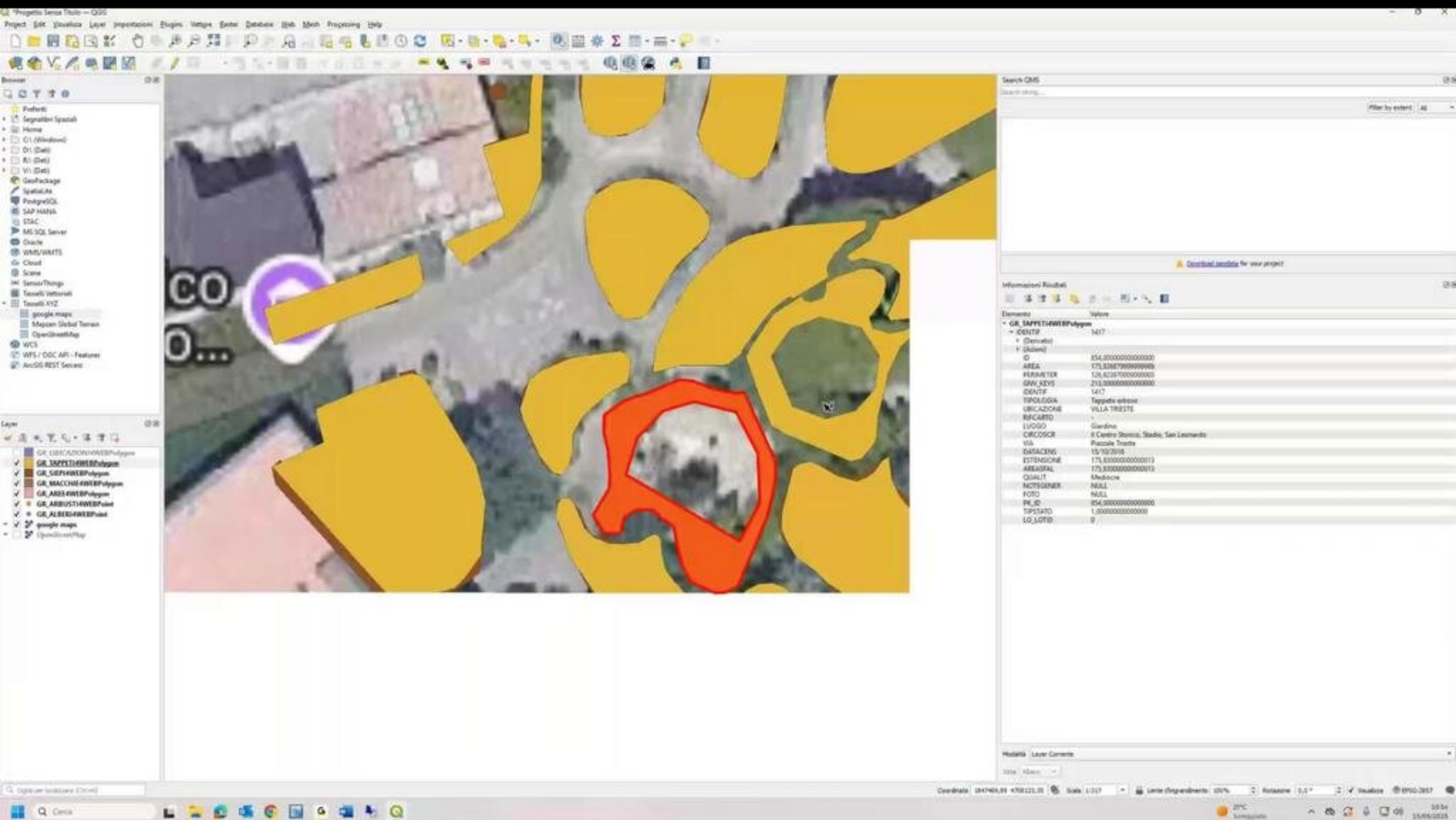


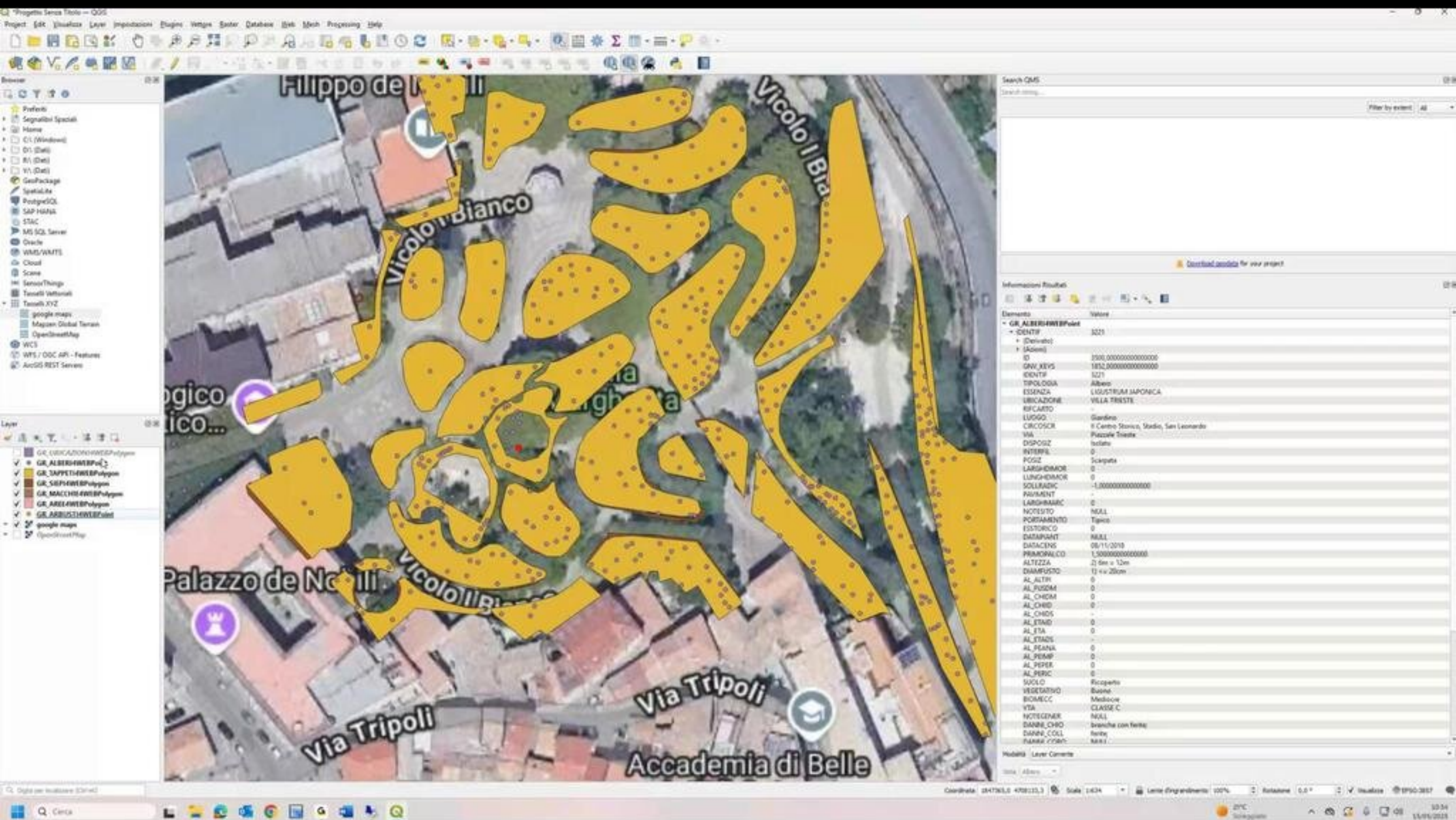


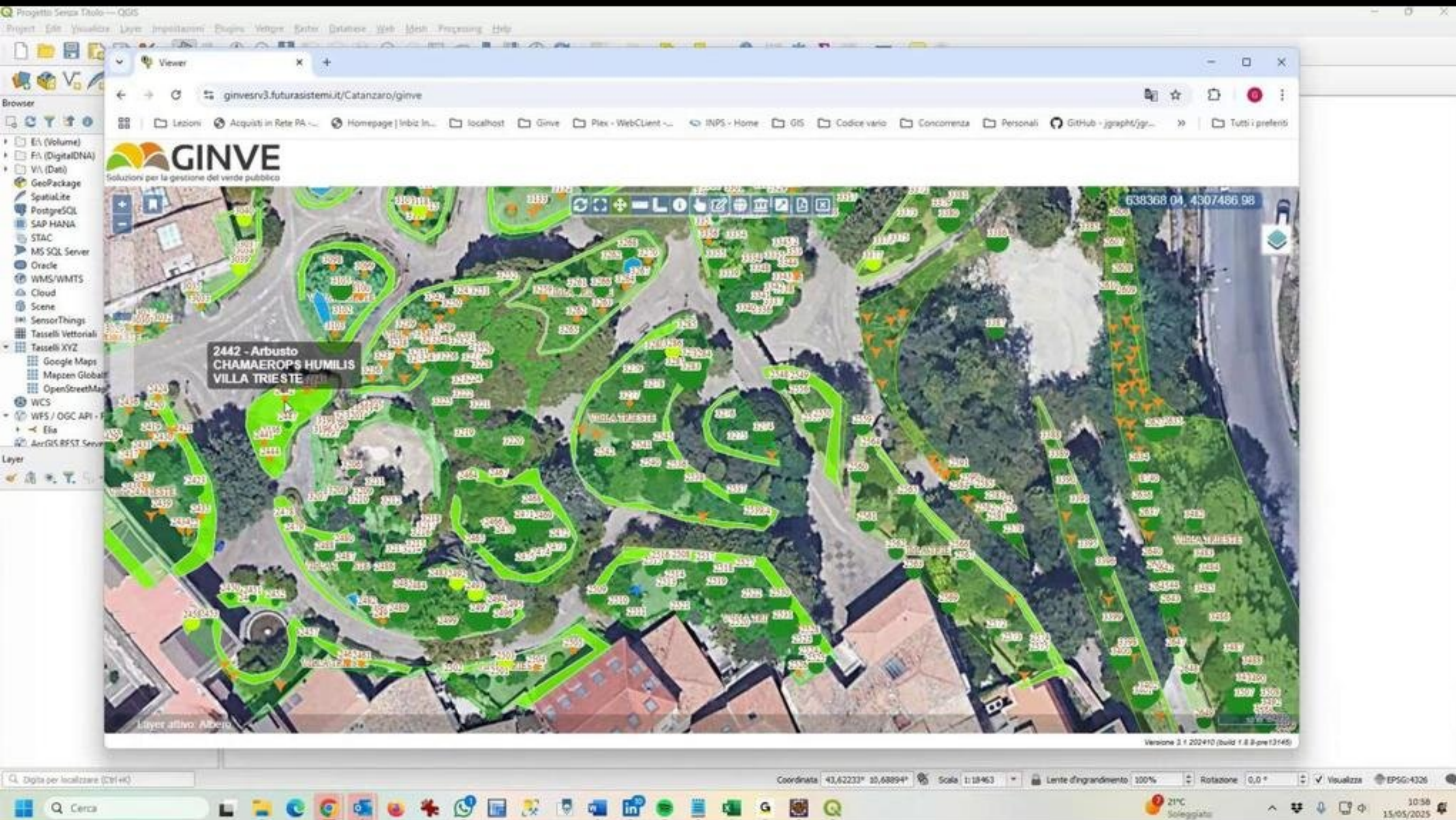


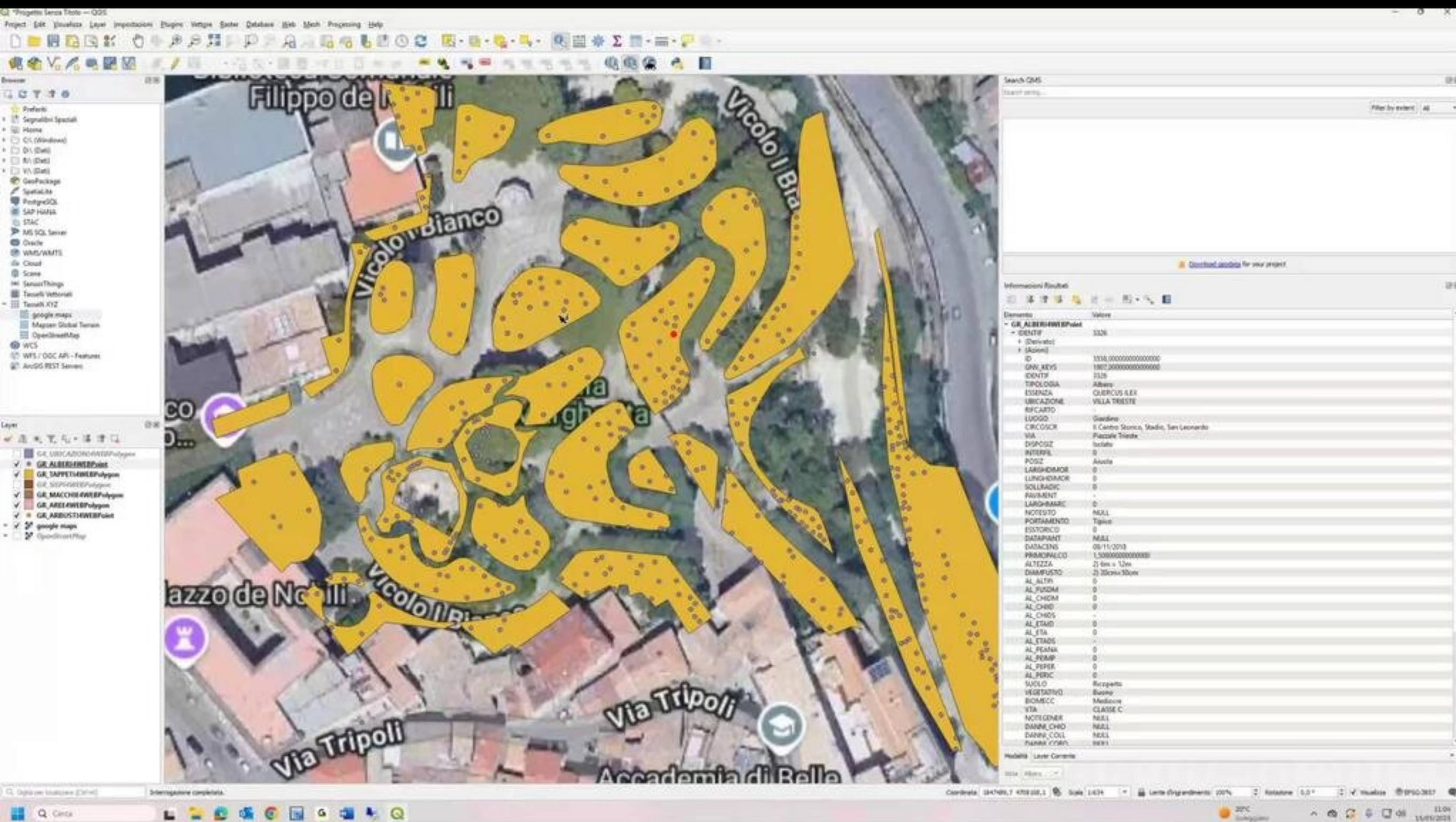














Project Browser

- Project
- Signatures Spatial
- Home
- C:\ (Windows)
- D:\ (Data)
- R\ (Data)
- V\ (Data)
- GeoPackage
- Spatialite
- PostgreSQL
- SAP HANA
- STAC
- MS SQL Server
- Oracle
- WMS/WMF
- Cloud
- Scene
- SceneThings
- Tanuki Vectors
- Tanuki XYZ
- google maps
- Mapbox Global Terrain
- OpenStreetMap
- WCS
- WFS / OGC API - Features
- ArcGIS REST Services

Layers

- GR_EDICAZIONEWEBPolygon
- GR_ALBERIWEBPoint
- GR_TAPPETIWEBPolygon
- GR_SOPRILEVATIWEBPolygon
- GR_MACCHIEWEBPolygon
- GR_AREIWEBPolygon
- GR_ARBUSTIWEBPoint
- google maps
- OpenStreetMap

Information Results

Elemento	Valore
GR_ALBERIWEBPoint	3326
ID	1336.0000000000000
GRV_KEYS	1807.0000000000000
IDENTIF	3326
TIPOLOGIA	Albero
ESSENZA	QUERCUS ILEX
UBICAZIONE	VILLA TRISTE
RFCARTO	-
LUGGO	Gardino
CIRCOSCR	Il Centro Storico, Medio, San Leonardo
VIA	Piazza Trieste
DISPOSIZ	Isolato
INTERF.	0
POSIZ	Albero
LARGHERIA	0
LUNGHERIA	0
SOLLEGGIO	0
PAYMENT	-
LARGHERIA	0
NOTESTO	MALL
PORTAMENTO	Tighe
ISTORICO	0
DATAPUNT	MALL
DATACENS	06/11/2018
PRIMORALCO	1.5000000000000
ALTEZZA	21.6m x 12m
DIAMETRO	21.20m x 10m
AL_ALTR	0
AL_FUSMA	0
AL_CHDM	0
AL_CHDO	0
AL_CHDS	-
AL_STAID	0
AL_STA	0
AL_STAID	-
AL_PANA	0
AL_PAMP	0
AL_PEPER	0
AL_PERIC	0
SUCLO	Ricoperto
VEGETAZIO	Buono
BIORECC	Medio
VITA	CLASSE C
NOTEGENER	MALL
DANNE_CHO	MALL
DANNE_COLL	MALL
DANNE_CORN	MALL



Browser

- Preferiti
- Segnalibri Spaziali
- Home
- CL (Windows)
- D1 (Data)
- R1 (Data)
- V1 (Data)
- GenPackage
- Spatialite
- PostgreSQL
- SAP HANA
- STAC
- MS SQL Server
- Oracle
- WMS/WMTS
- Cloud
- Scene
- SensorThings
- Tessuti Vettoriali
- Tessuti XYZ
- google maps
- Mapzen Global Terrain
- OpenStreetMap
- WCS
- WFS / OGC API - Features
- ArcGIS REST Services

Layer

- GR_UbicazioneWEBPolygon
- GR_ALBERIWEBPoint
- GR_TAPPETIWEBPolygon
- GR_SITIPWEBPolygon
- GR_MACCHIEWEBPolygon
- GR_AREIWEBPolygon
- GR_ARBUSTIWEBPoint
- google maps
- OpenStreetMap



Search QGIS

Search string...

Filter by extent: All

Download metadata for your project

Information Results

Elemento	Valore
GR_TAPPETIWEBPolygon	
IDENTIF	5315
IDENTIF (Derivato)	
IDENTIF (Asse)	450,00000000000000
AREA	201,29278899999997
PERIMETER	76,021100000000004
GNV_KEYS	994,00000000000000
IDENTIF	5315
TIPOLOGIA	Prato naturale
UBICAZIONE	SCUOLA DON MILANI VIA MARRICOLA PISTOIA
RIFCARTE	
LUOGO	Scuola
CIRCOSCR	III Sala, Sesto, Santa Maria, Germaneto
VIA	Via Domenico Mannicola Pistoia
DATACENS	08/01/2019
ESTENSIONE	190,00000000000000
AREASAL	0
QUALIT	Buono
NOTEGENER	NULL
FOTO	desktop-Foto 5315.JPG
PK_ID	450,00000000000000
TIPSTATO	1,00000000000000
LO_LOTID	0

Modello Layer Commenti

Stile: Abbruci



Browser

- ✓ Prefere...
- ✓ Segnalati Spatiali
- ✓ Home
- ✓ C:\ (Windows)
- ✓ D:\ (Data)
- ✓ R\ (Data)
- ✓ V\ (Data)
- ✓ GeoPackage
- ✓ Spatialite
- ✓ PostgreSQL
- ✓ SAP HANA
- ✓ STAC
- ✓ MS SQL Server
- ✓ Oracle
- ✓ WMS/WMTS
- ✓ Cloud
- ✓ Scene
- ✓ SensorThings
- ✓ Tasse... Vettoriale
- ✓ Tasse... XYZ
- ✓ google maps
- ✓ Mapbox Global Terrain
- ✓ OpenStreetMap
- ✓ WCS
- ✓ WFS / OGC API - Features
- ✓ ArcGIS REST Services

Layer

- ✓ GR_VORCAUTIONWEBPolygon
- ✓ GR_ALBERIWEBPoint
- ✓ GR_TAPPETIWEBPolygon
- ✓ GR_SOPRILEVATIWEBPolygon
- ✓ GR_MACCHIEWEBPolygon
- ✓ GR_AREIWEBPolygon
- ✓ GR_ARBUSTIWEBPoint
- ✓ google maps
- ✓ OpenStreetMap



Search QGIS

Search string:

Filter by extent: All

Download results for your project

Informazioni Risultati

Elemento Valore

GR_TAPPETIWEBPolygon

IDENTIF 9000

IDENTIF (Derivati)

IDENTIF (Identi)

ID 801,0000000000000000

AREA NULL

PERIMETER NULL

SNV_KEYS 802,0000000000000000

IDENTIF 9000

TIPOLOGIA Prato naturale

UBICAZIONE CATANZARO

RIFCARO

LUOGO

CIRCOSCR Il Centro Storico, Stadio, San Leonardo

VIA Viale Biontini

DATACENS 28/10/2020

ESTENSIONE 224,519999999999992

AREASAL 224,519999999999992

QUALIT

NOTEGENRE NULL

FCO NULL

PK_ID 801,0000000000000000

TIPSTATO 1,0000000000000000

LO_LOTTO 0

Modifica Layer Corrente

Stile: Modifica

Copio per incollare (Ctrl+V)

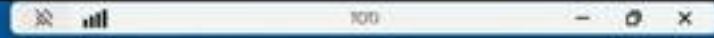
Spazio libero: 90,427 m (1A,1122170)

Coordinate: 284786, 1 407218,2 Scale: 1:1289 Lente d'ingrandimento: 200% Rotazione: 0,0° Invisibile EPSG:3857



Attiva Windows
Passa a Impostazioni per attivare Windows.





Sto eseguendo l'arresto







QGIS 3.42.2

File Home Contatti Visualizza Strumenti collegamenti Strumenti applicazioni

QGIS 3.42.2

Nome	Ultima modifica	Tipi	Dimensione
GRASS GIS 6.4.1	14/05/2025 15:33	Collegamento	248
QGIS2W Setup	14/05/2025 15:34	Collegamento	248
QGIS2W Shell	14/05/2025 15:33	Collegamento	248
QGIS Desktop 3.42.2	14/05/2025 15:34	Collegamento	198
Qt Designer with QGIS 3.42.2 custom wid...	14/05/2025 15:34	Collegamento	248
SAGA GIS 5.7.1	14/05/2025 15:34	Collegamento	248

Accesso rapido

- Desktop
- Download
- Documenti
- Immagini
- Questo PC
 - Desktop
 - Documenti
 - Download
 - Immagini
 - Musica
 - Oggetti 3D
 - Video
- Disco locale (C:)
- Rete

0 elementi 1 elemento selezionato: 175 byte



QGIS 3.4.2

QGIS 3.4.2

Nome	Ultima modifica	Tip	Dimensione
GRASS GIS 6.4.1	14/05/2025 15:53	Collegamento	2 KB
QGIS2W Setup	14/05/2025 15:54	Collegamento	2 KB
QGIS2W Shell	14/05/2025 15:55	Collegamento	2 KB
QGIS Desktop 3.4.2	14/05/2025 15:54	Collegamento	1 KB
Qt Designer with QGIS 3.4.2.2 custom wid...	14/05/2025 15:54	Collegamento	2 KB
SAGA GIS 5.7.1	14/05/2025 15:54	Collegamento	2 KB

QGIS 3.4.2 Münster



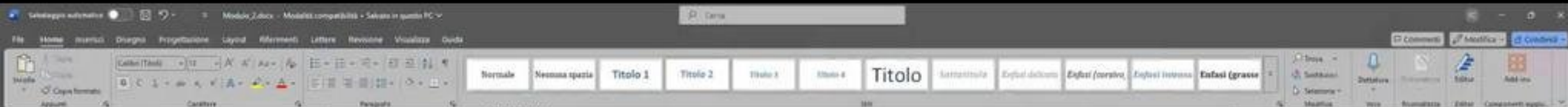
- Browser
- Profilo
 - Segnalibri Spatiali
 - Home Progetto
 - Home
 - CL
 - GeoPackage
 - Spatialite
 - PostgreSQL
 - SAP HANA
 - STAC
 - MS SQL Server
 - Oracle
 - WMF/WMTS
 - Cloud
 - Scenes
 - SensorThings
 - Vector Tiles
 - XYZ Tiles
 - Mapzen Global Terrain
 - OpenStreetMap
 - WCS
 - WFS / OGC API - Features
 - AnyGIS REST Servers

- Layer
- GR_MISCAZIOINI4WEBPolygon
 - GR_ALBERI4WEBPoint
 - GR_TAPPE4WEBPolygon
 - GR_SIEPI4WEBPolygon
 - GR_MACCHIE4WEBPolygon
 - GR_AREI4WEBPolygon
 - GR_AREI514WEBPoint
 - OpenStreetMap



Information Results

Elemento	Valore
GR_TAPPE4WEBPolygon	
IDENT	2383
Identificatore	
ID	115,00000000000000
AREA	524,12364999999999
PERIMETER	143,14154999999999
SHA256	208,00000000000000
IDENT	2383
TROLOGIA	Prato naturale
UBICAZIONE	VILLA TRESTE
RIFCARTO	
LUOGO	Gardino
CIRCOLO	Il Centro Storico, Stadio, San Leonardo
VIA	Piazza Trento
SARACINI	15/10/2018
ESTENSIONE	524,12364999999999
AREA514	524,12364999999999
QUALIT	Medio
NOTEGENER	NAL
FOFO	desktop/Foto/2383.JPG
FILE	115,00000000000000
TIPIVIA	1,00000000000000
LOJOTO	0



spaziare

- Garantire corretto sistema di riferimento (SRS/EPSS) e metadata
- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

Importazione dati

- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

Metodo A - tramite "Esporta"

Fai clic destro sul layer che vuoi importare (es. uno shapefile)

- Comprendere quali aree verdi si importeranno

Fasi operative

Analisi preliminare dei dati

- Verifica formati disponibili: .shp/.dbf/.shx, GeoPackage, connessioni OGR/GeoJSON
- Controllo SRS e bounding box
- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

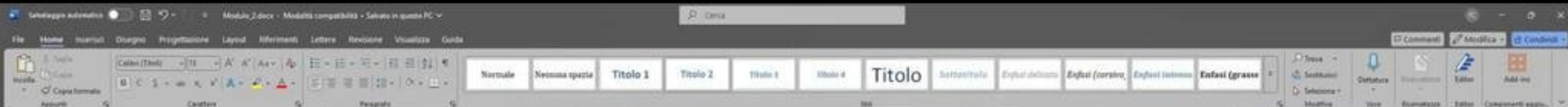
- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

Importazione dati

- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

Metodo A - tramite "Esporta"

Fai clic destro sul layer che vuoi importare (es. uno shapefile)



Importazione dati

- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

Metodo A - tramite "Esporta"

Fai clic destro sul layer che vuoi importare (es. uno shapefile)

Seleziona "Esporta" > "Salva elementi come..."

Nella finestra che si apre:

Formato: scegli PostgreSQL

Database: seleziona la connessione PostGIS creata prima

Nome tabella: scegli il nome da dare alla tabella nel DB

SRID: assicurati che sia corretto (es. 4326 o 3857)

Premi **OK**

QGIS esporterà tutti i campi e le geometrie, e creerà automaticamente la tabella in PostGIS.

Metodo B - tramite DB Manager

Vai su **Database > DB Manager**

Nella sezione di sinistra, trova il tuo database PostGIS.



- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

Metodo A - tramite "Esporta"

Fai clic destro sul layer che vuoi importare (es. uno shapefile)

I

Seleziona "Esporta" > "Salva elementi come..."

Nella finestra che si apre:

Formato: scegli PostgreSQL

Database: seleziona la connessione PostGIS creata prima

Nome tabella: scegli il nome da dare alla tabella nel DB

SRID: assicurati che sia corretto (es. 4326 o 3857)

Premi **OK**

QGIS esporterà tutti i campi e le geometrie, e creerà automaticamente la tabella in PostGIS.

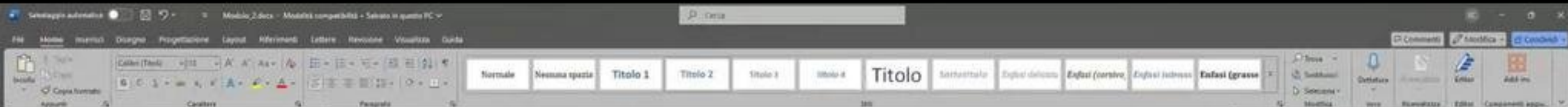
Metodo B - tramite DB Manager

Vai su **Database > DB Manager**

Nella sezione di sinistra, trova il tuo database PostGIS

Clicca su **"Importa Layer"**

Scegli il layer QGIS da importare



Seleziona "Esporta" > "Salva elementi come..."

Nella finestra che si apre:

Formato: scegli PostgreSQL

Database: seleziona la connessione PostGIS creata prima

Nome tabella: scegli il nome da dare alla tabella nel DB

SRID: assicurati che sia corretto (es. 4326 o 3857)

Premi **OK**

QGIS esporterà tutti i campi e le geometrie, e creerà automaticamente la tabella in PostGIS.

Metodo B – tramite DB Manager

Vai su **Database > DB Manager**

Nella sezione di sinistra, trova il tuo database PostGIS

Clicca su **"Importa Layer"**

Scegli il layer QGIS da importare

Configura:

Nome tabella

SRID

Se creare indice spaziale

Se sovrascrivere tabelle esistenti

Premi **OK**

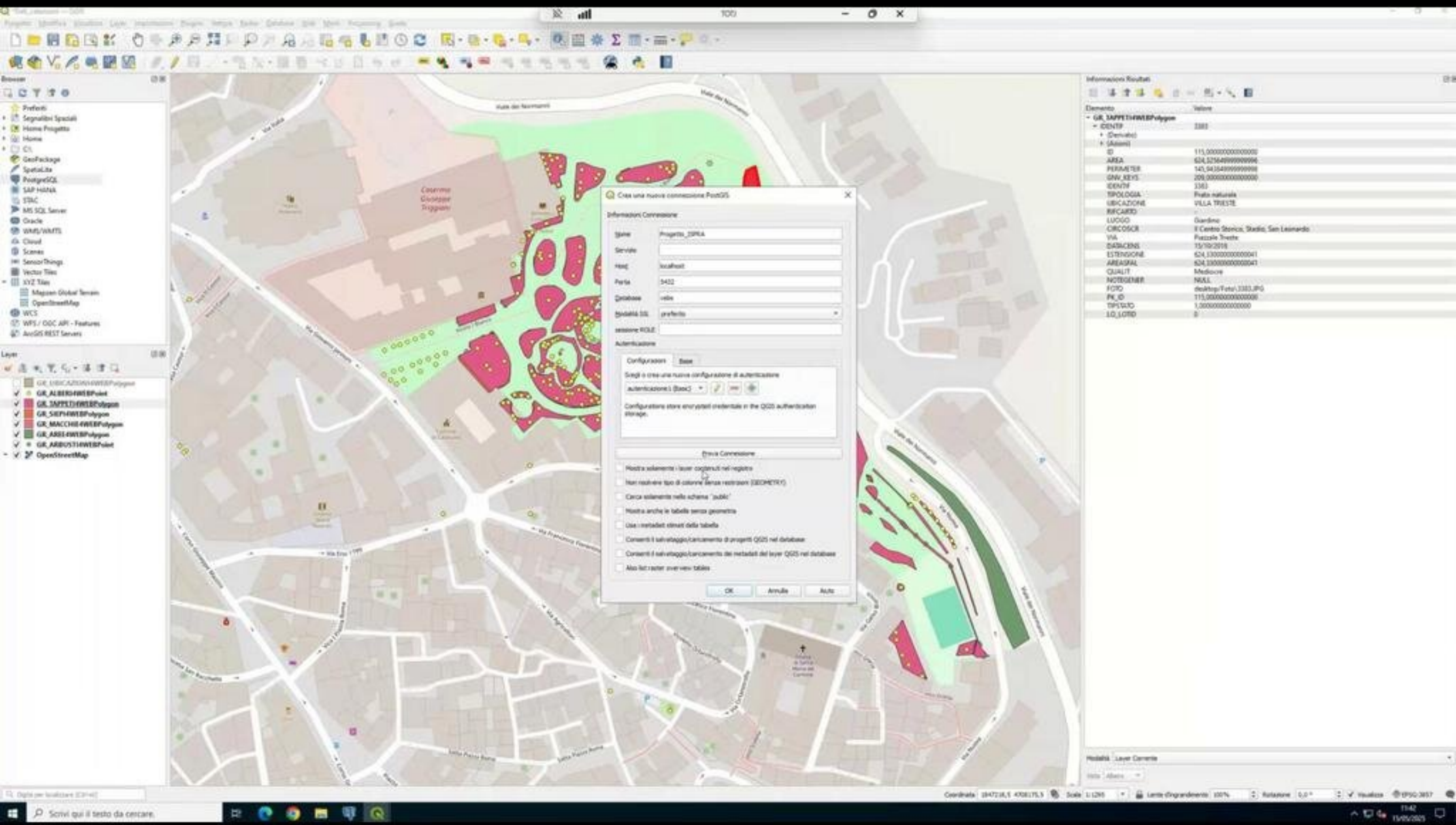
- Successivamente verificare l'importazione:

```
SELECT * FROM public.ree_verdi;
```

- Nel caso si dovesse calcolare gli attributi come densità_alberi:

Importazione e degli shapefile ALBERO e TAPPETI.

Tramite attributo (comune alle due tabelle) ubicazione andiamo a prendere i poligoni e punti.





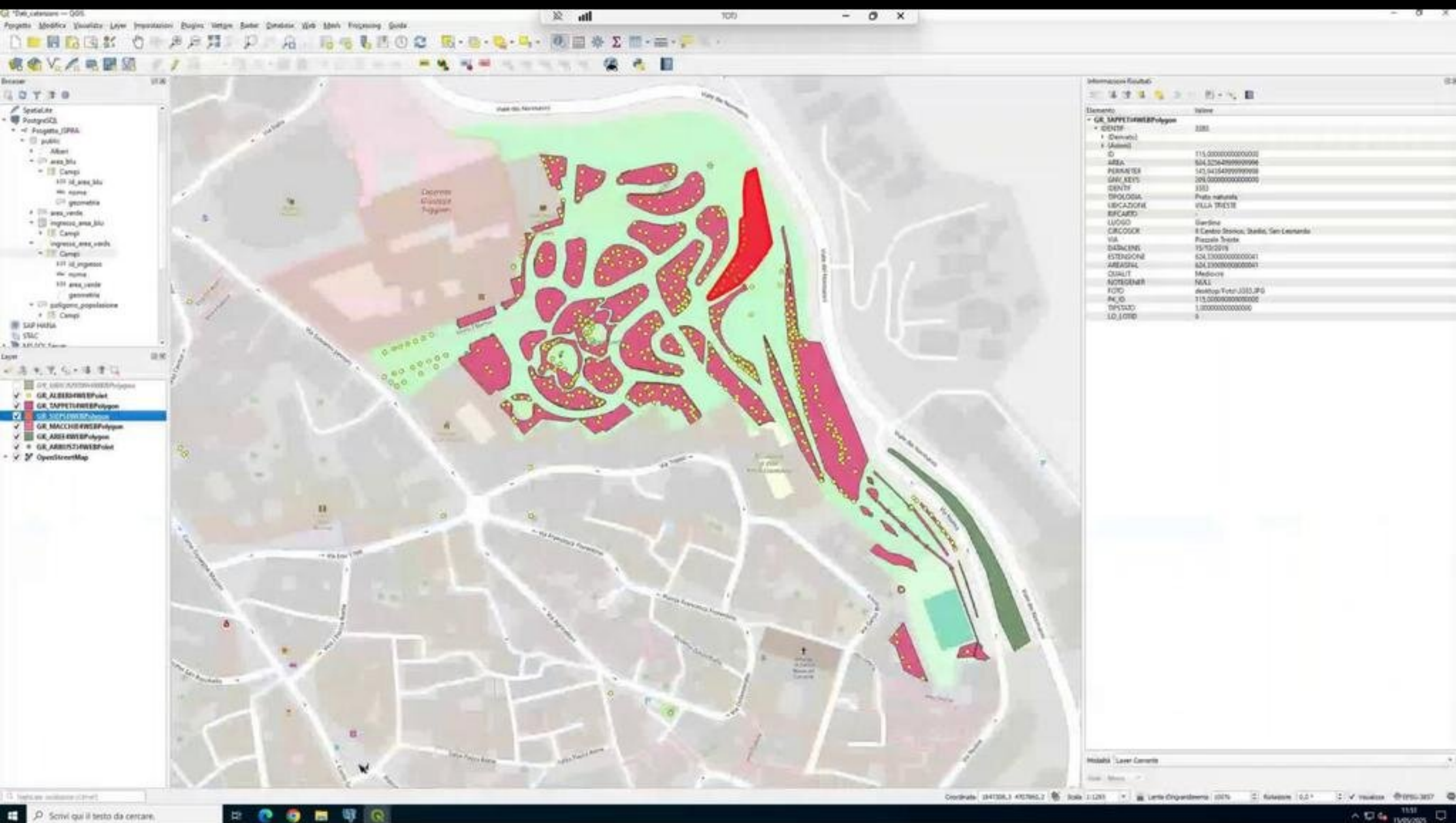
- Browser
- Progetti
 - Segnalibri Spaziali
 - Home Progetto
 - Home
 - CL
 - GeoPackage
 - Spatialite
 - PostgreSQL
 - Progetto_QGIS
 - pubblici
 - area_blu
 - 131_131_area_blu
 - no_nome
 - geometria
 - area_verde
 - ingresso_area_blu
 - ingresso_area_verde
 - poligoni_popolazione
 - SAP_H2018
 - STAT
 - MS SQL Server
 - Oracle
 - QGIS_Layers

- Layer
- GR_LIBERAZIONEWEBPolygon
 - GR_ALBERIWEBPoint
 - GR_SAPPIWEBPolygon
 - GR_SIEPIWEBPolygon
 - GR_MACCHIEWEBPolygon
 - GR_AREIWEBPolygon
 - GR_ARBUSTIWEBPoint
 - OpenStreetMap



Informazioni Risultati

Elemento	Valore
GR_SAPPIWEBPolygon	
CONF	3383
Area	115,00000000000000
PERIMETER	624,52640000000000
SHV_KEYS	143,94250000000000
SHV_KEYS	208,00000000000000
SHV_KEYS	3383
TPOLOGIA	Prato naturale
USCAGIONE	VILLA TRESTE
RICCARDO	
LUOGO	Gardino
CIRCOSCR	8 Centro Storico, Stadio, San Leonardo
VIA	Piazza Trento
SARCIEN	15/10/2018
ESTENSIONE	624,52640000000000
AREASAL	624,52640000000000
QUALIT	Medio
NOVENN	NAL
FOTO	desktop/Foto/3383.JPG
FILE	115,00000000000000
TIPOLOGIA	1,00000000000000
LO,LOTO	8



ppdmmx

File Object Tools Edit View Window Help

Welcome | vbs/postgres@Progetto_ISPRA X

vbs/postgres@Progetto_ISPRA

Query Query History

```
1: SELECT *  
2: FROM public.Tappeti;
```

Scratch Pad

Data Output Messages Geometry Viewer X Notifications

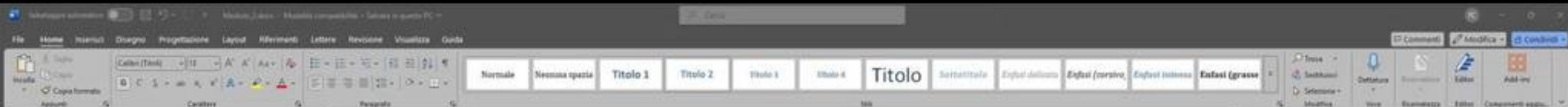
ERROR: la relazione "public.tappeti" non esiste
LINE 2: FROM public.Tappeti;
 ^

ERROR: la relazione "public.tappeti" non esiste
SQL state: 42P01
Character: 13

Total rows: 5916 Query complete 00:00:00.270

Scrivi qui il testo da cercare.

11/18
15/05/2025



Seleziona "Esporta" > "Salva elementi come..."

Nella finestra che si apre:

Formato: scegli PostgreSQL

Database: seleziona la connessione PostGIS creata prima

Nome tabella: scegli il nome da dare alla tabella nel DB

SRID: assicurati che sia corretto (es. 4326 o 3857)

Premi **OK**

QGIS esporterà tutti i campi e le geometrie, e creerà automaticamente la tabella in PostGIS.

Metodo B - tramite DB Manager

Vai su **Database > DB Manager**

Nella sezione di sinistra, trova il tuo database PostGIS

Clicca su "Importa Layer"

Scegli il layer QGIS da importare

Configura:

Nome tabella

SRID

Se creare indice spaziale

Se sovrascrivere tabelle esistenti

Premi **OK**

- Successivamente verificare l'importazione:

```
SELECT *
```

```
FROM public aree_verdi;
```

- Nel caso si dovesse calcolare gli attributi come densità_alberi:

Importazione e degli shapefile ALBERO e TAPPETI.

Tramite attributo (comune alle due tabelle) ubicazione andiamo a prendere i poligoni e i punti.



Schermata di Microsoft Word, documento "Modulo 2.docx".

Contenuto del documento:

Ogni layer può essere importato.

Configura:

- Nome tabella
- SRID
- Se creare indice spaziale
- Se sovrascrivere tabelle esistenti

Premi OK

- Successivamente verificare l'importazione:

```
SELECT *  
FROM public.ree_verdi;
```

- Nel caso si dovesse calcolare gli attributi come densità_alberi:

Importazione e degli shapefile ALBERO e TAPPETI.

Tramite attributo (comune alle due tabelle) ubicazione andiamo a prendere i poligoni e i punti.

Con la funzione ST_Within(punto, poligono) = TRUE

```
SELECT  
a.ubicazione, --campo ubicazione dell'area  
COUNT (t.*) AS numero_alberi, --Campo calcolo numero di alberi totale  
ST_Area(a.geom) AS area_mq, --Campo area del poligono area verde  
ROUND (COUNT (t.*) / ST_Area(a.geom), 2) AS densita_alberi_mq, --Campo  
calcolo densità alberi  
FROM aree a  
LEFT JOIN alberi t ON ST_Contains(a.geom, t.geom)  
GROUP BY a.id_area, a.geom;
```

Pagina 1 di 1 | 452 parole | Completamento del testo atteso | Accessibilità: buona

Barra di sistema: 32°C, 12:04, 11/05/2023.

FROM public aree_verdi;

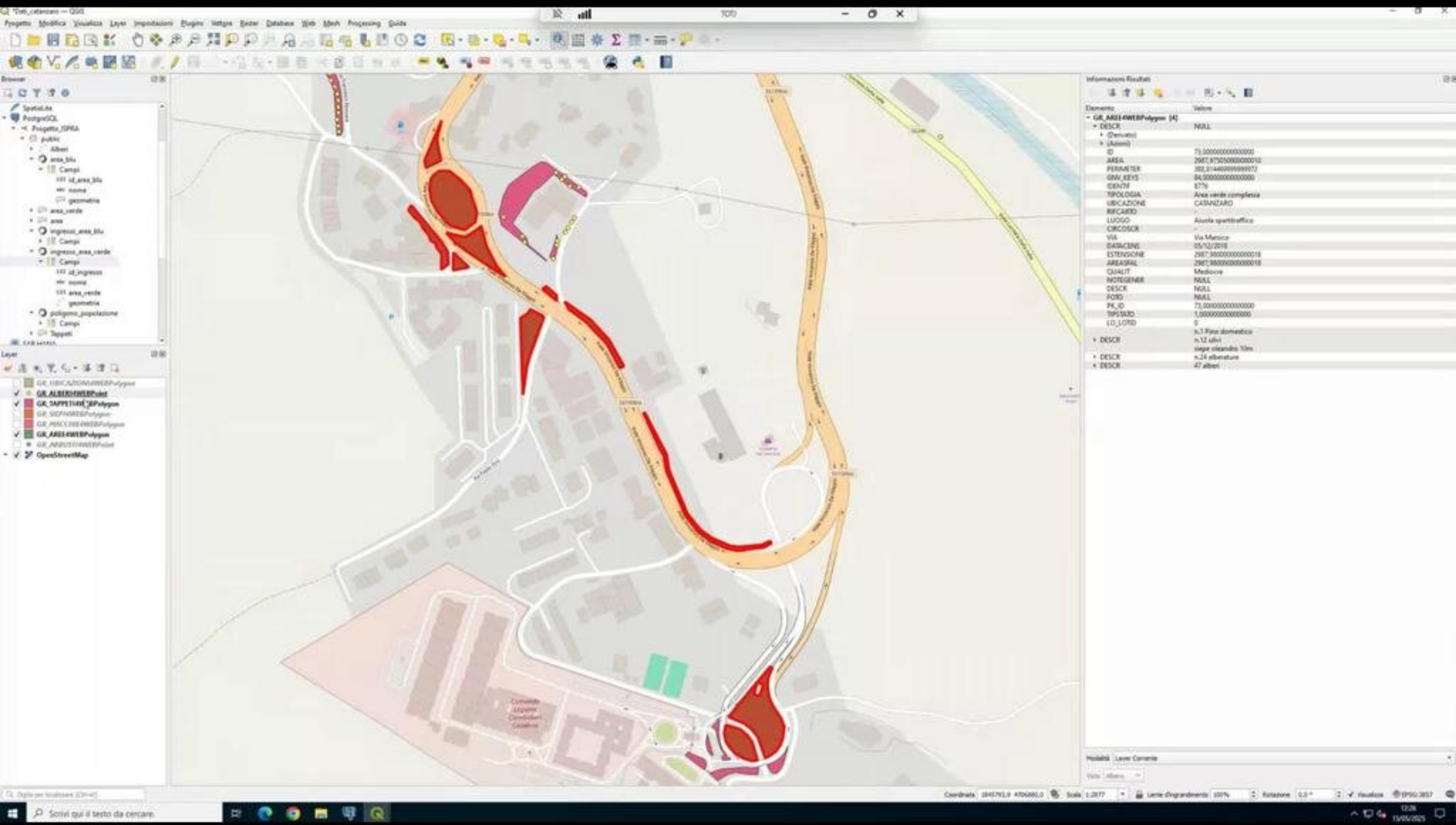
- Nel caso si dovesse calcolare gli attributi come densità_alberi:
Importazione e degli shapefile ALBERO e TAPPETI.
Tramite attributo (comune alle due tabelle) ubicazione andiamo a prendere i poligoni e i punti.
Con la funzione **ST_Within**(punto, poligono) = TRUE
SELECT
a.ubicazione, --campo ubicazione dell'area
COUNT (t.*) AS numero_alberi, --Campo calcolo numero di alberi totale
ST_Area(a.geom) AS area_mq, --Campo area del poligono area verde
ROUND (COUNT (t.*) / ST_Area(a.geom), 2) AS densita_alberi_mq, --Campo calcolo densità alberi
FROM aree a
LEFT JOIN alberi t ON ST_Contains(a.geom, t.geom)
GROUP BY a.id_area, a.geom;

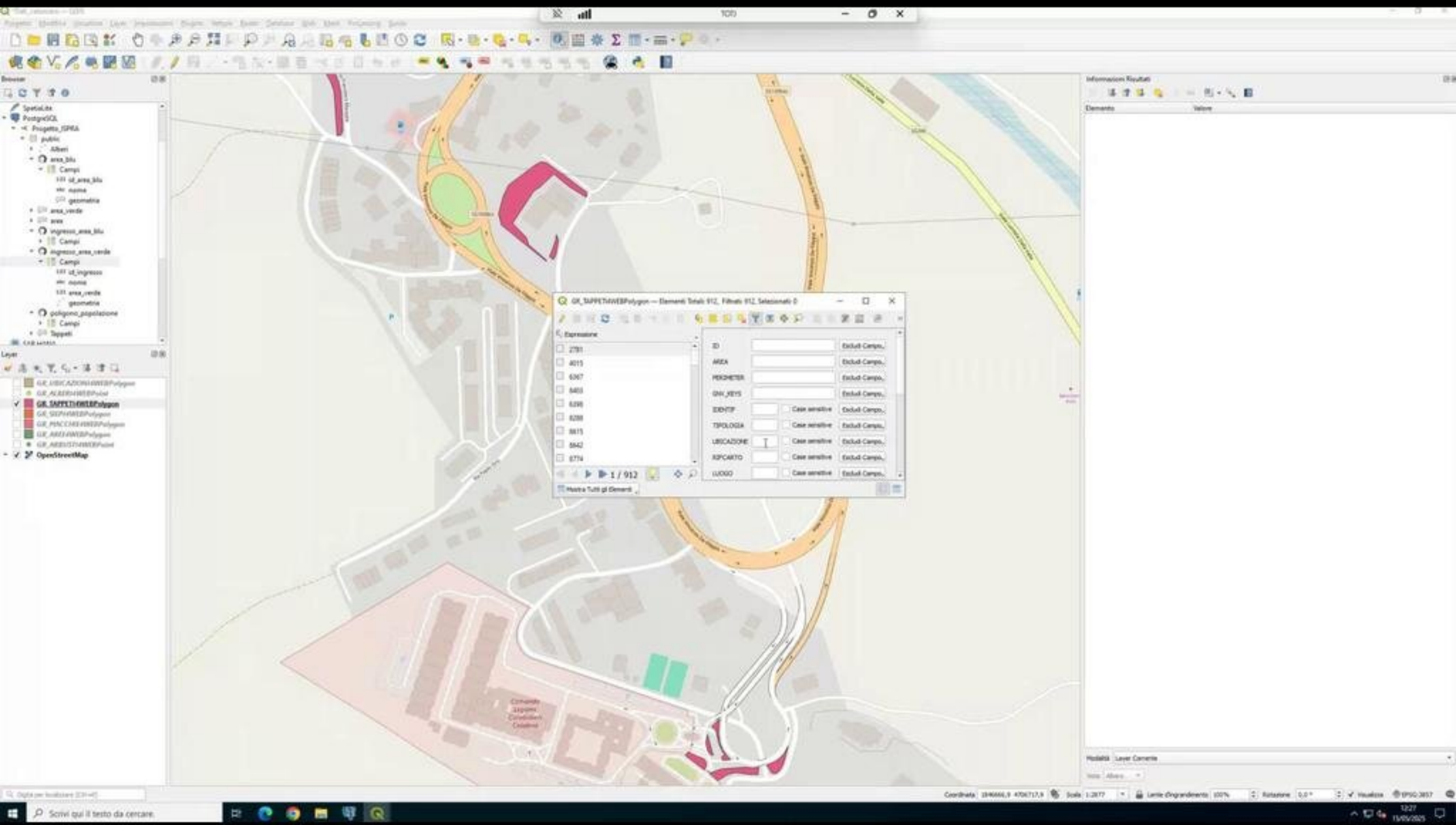
Aggiungere Densità = Numero di alberi / Superficie

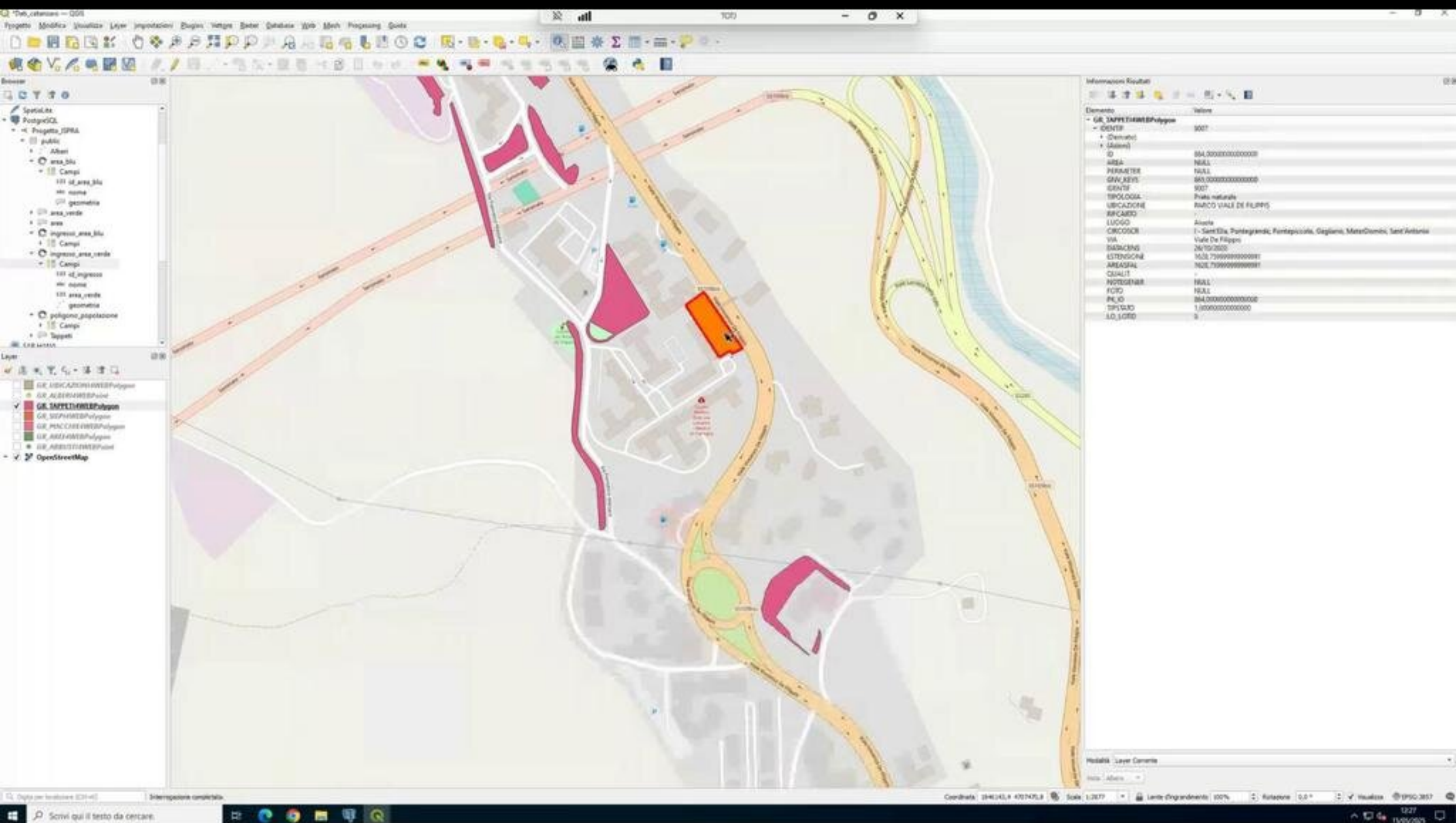
FROM public aree_verdi;

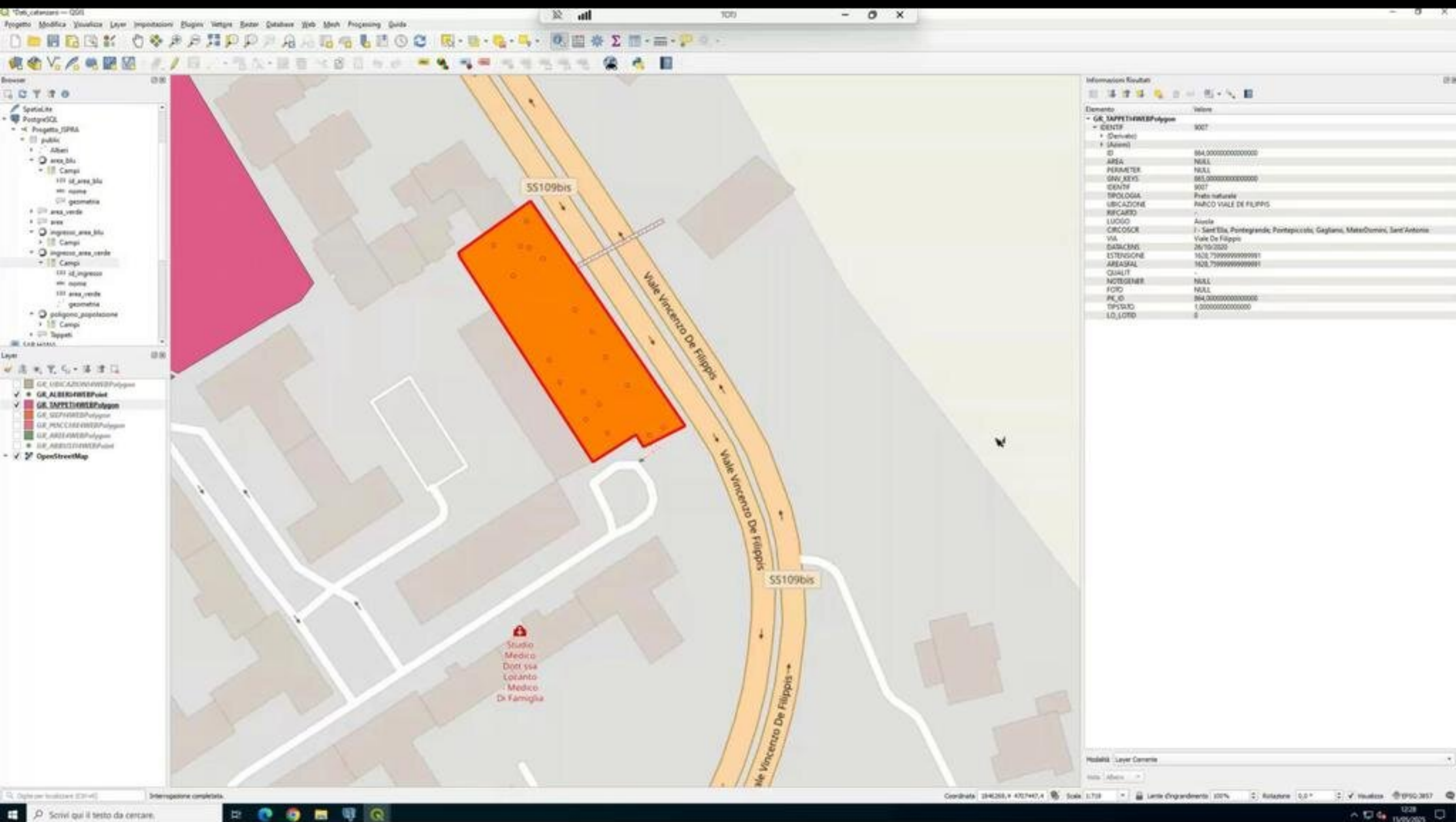
- Nel caso si dovesse calcolare gli attributi come densità_alberi:
Importazione e degli shapefile ALBERO e TAPPETI.
Tramite attributo (comune alle due tabelle) ubicazione andiamo a prendere i poligoni e i punti.
Con la funzione ST_Within(punto, poligono) = TRUE
SELECT
a.ubicazione, --campo ubicazione dell'area
COUNT (t.*) AS numero_alberi, --Campo calcolo numero di alberi totale
ST_Area(a.geom) AS area_mq, --Campo area del poligono area verde
ROUND (COUNT (t.*) / ST_Area(a.geom), 2) AS densita_alberi_mq, --Campo calcolo densità alberi
FROM aree a
LEFT JOIN alberi t ON ST_Contains(a.geom, t.geom)
GROUP BY a.id_area, a.geom;

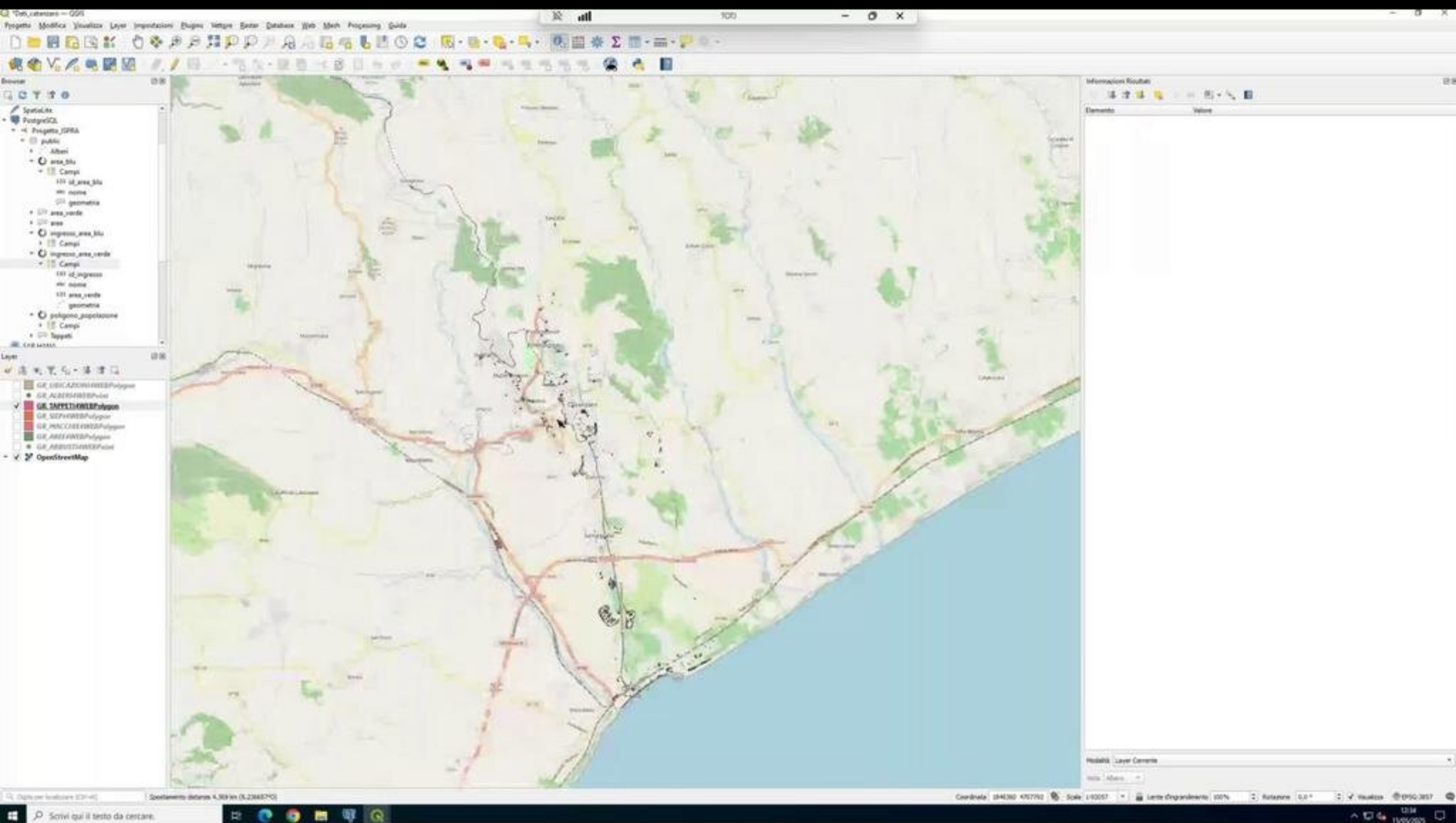
Aggiungere Densità = Numero di alberi / Superficie





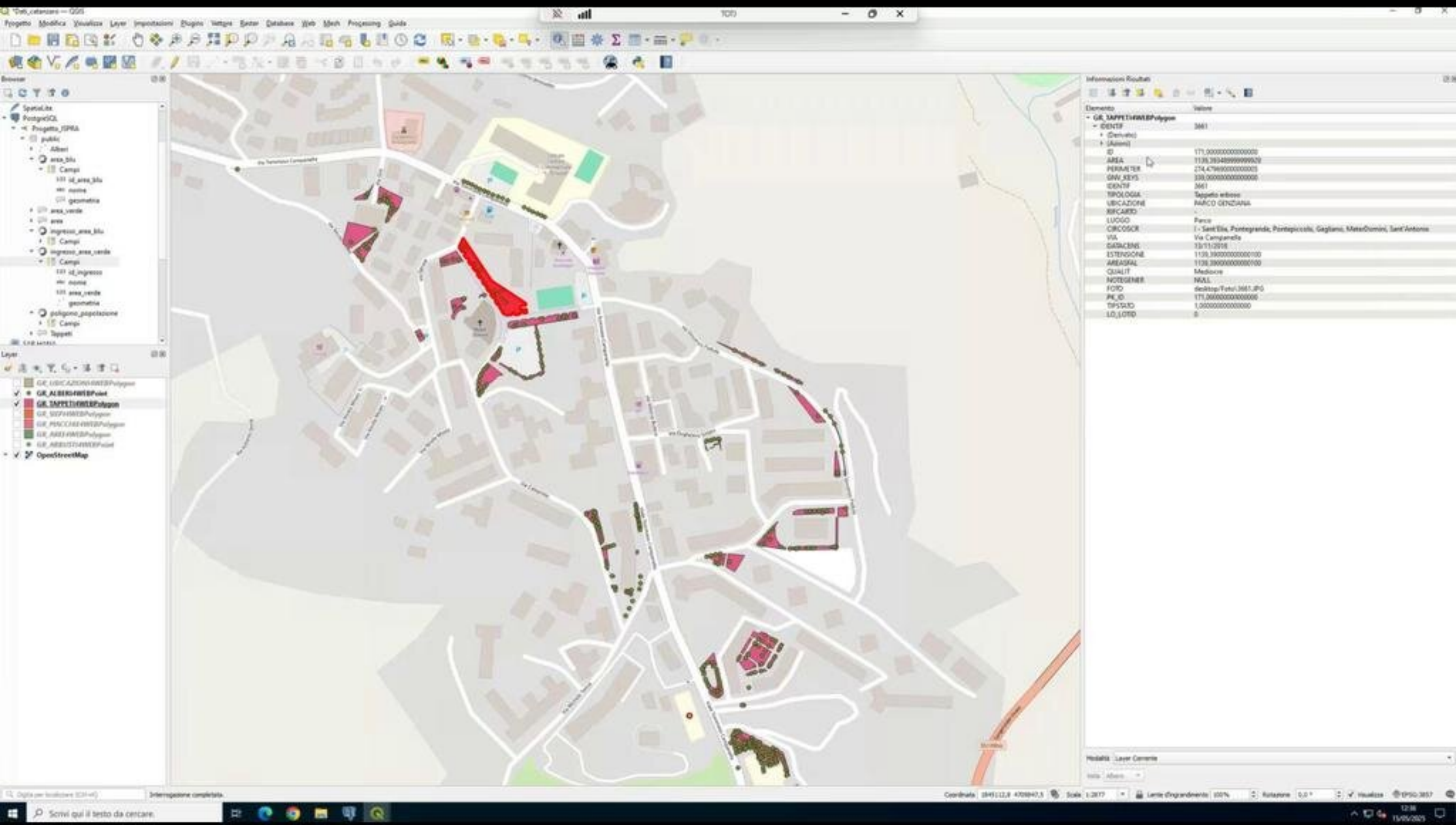


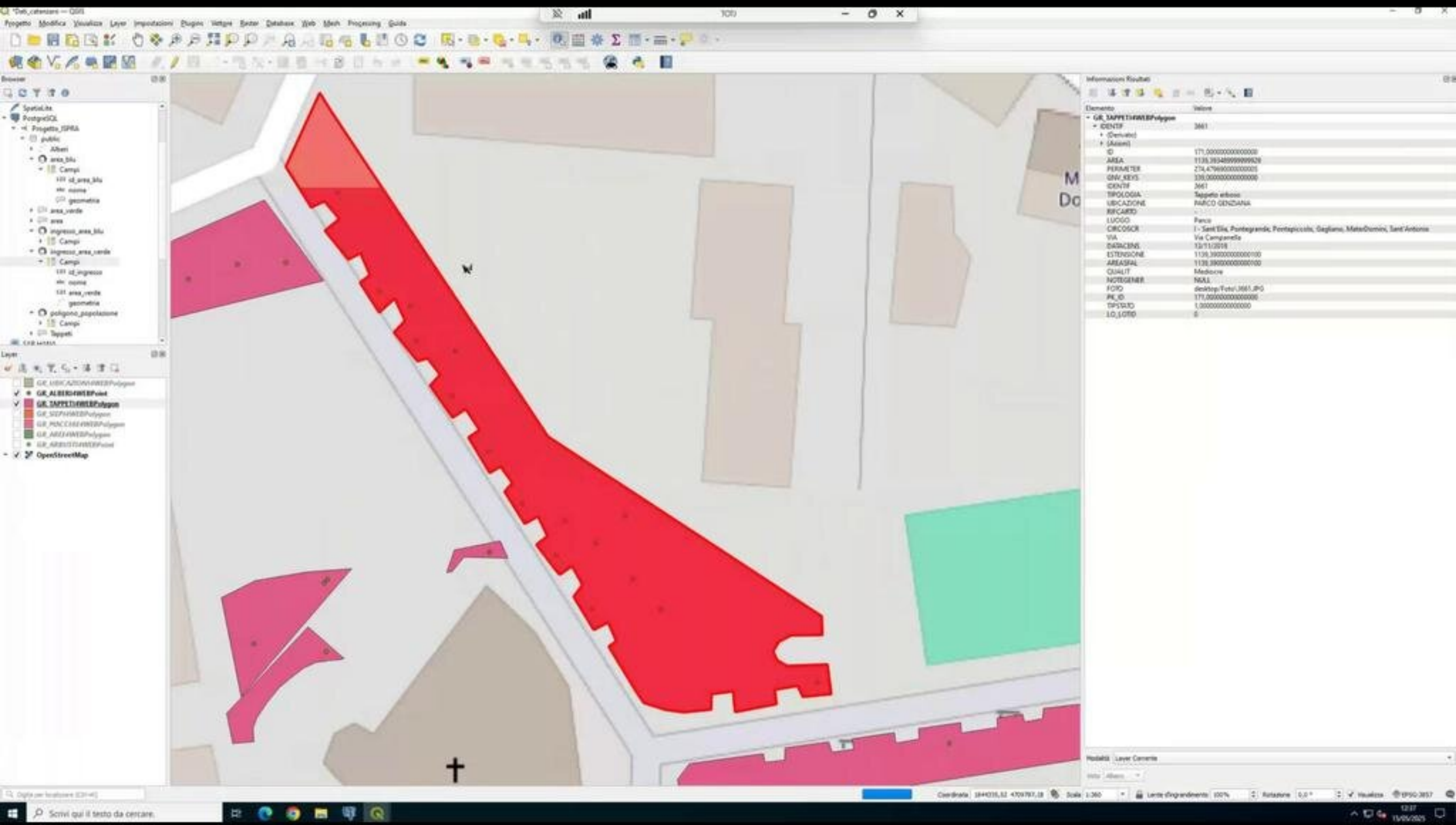


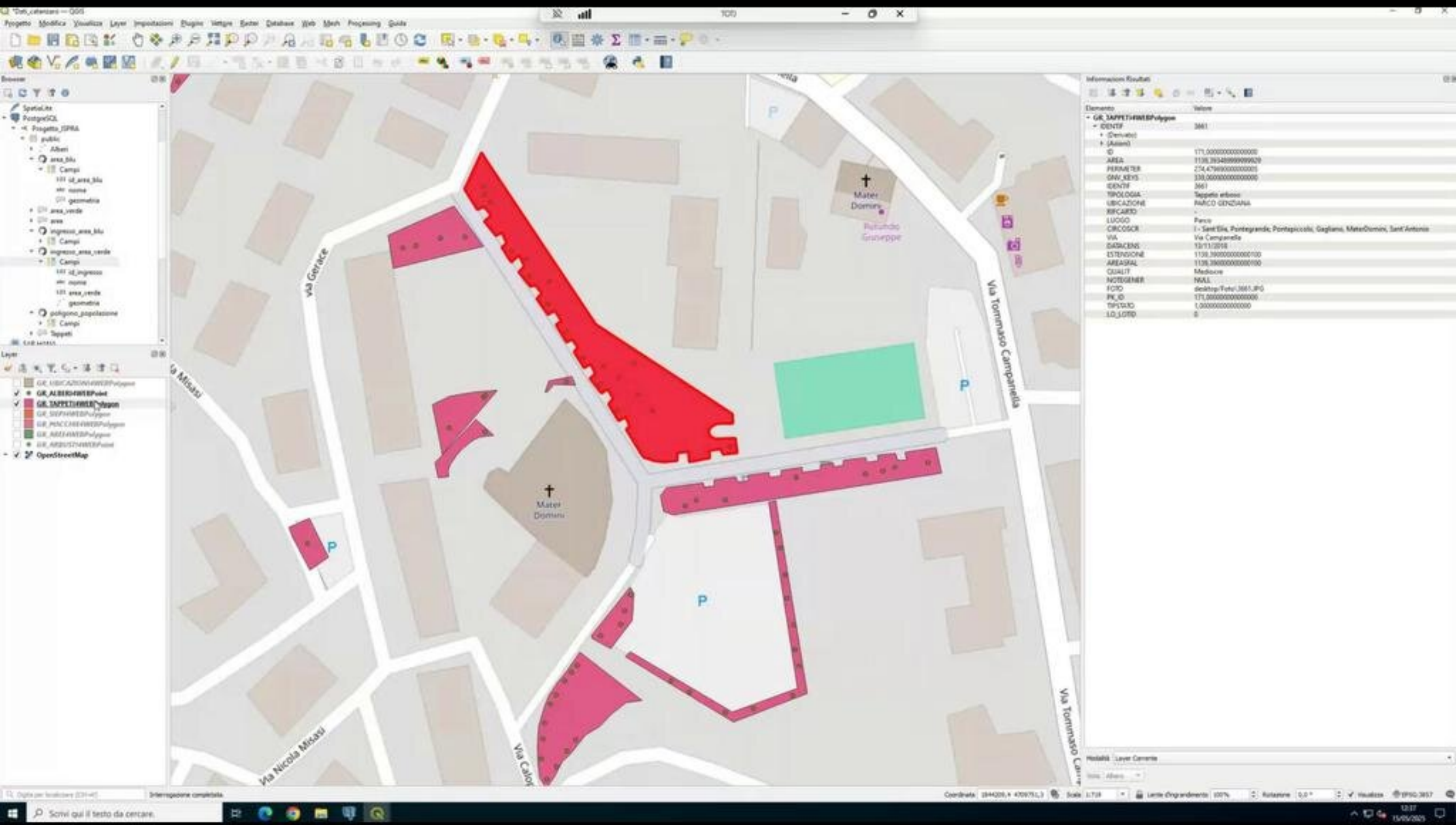


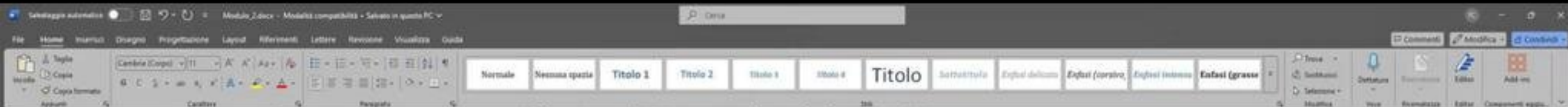
Verifica e pulizia post-importazione

- Query di controllo (SELECT COUNT, bounding box)
- Correzione geometrie invalidi con ST_MakeValid
- Aggiunta di metadati: data import, fonte









Nella sezione di sinistra, trova il tuo database PostGIS

Clicca su **"Importa Layer"**

Scegli il layer QGIS da importare

Configura:

Nome tabella

SRID

Se creare indice spaziale

Se sovrascrivere tabelle esistenti

Premi **OK**

- Successivamente verificare l'importazione:

```
SELECT *
```

```
FROM public.ree_verdi;
```

- Nel caso si dovesse calcolare gli attributi come densità_alberi:

Importazione e degli shapefile ALBERO e TAPPETI.

Tramite attributo (comune alle due tabelle) ubicazione andiamo a prendere i poligoni e i punti.

Con la funzione `ST_Within(punto, poligono) = TRUE`

```
SELECT
```

```
  a.ubicazione, --campo ubicazione dell'area
```

```
  COUNT (t.*) AS numero_alberi, --Campo calcolo numero di alberi totale
```

```
  ST_Area(a.geom) AS area_mq, --Campo area del poligono area verde
```

```
  ROUND (COUNT (t.*) / ST_Area(a.geom), 2) AS densita_alberi_mq, --Campo  
calcolo densità alberi
```

```
FROM aree a
```

```
LEFT JOIN alberi t ON ST_Contains(a.geom, t.geom)
```

```
GROUP BY a.id_area, a.geom;
```


Modulo_2.docx - Modulo compatibilità - Salvato in questo PC

File Home Inserto Disegno Progettazione Layout Riferimenti Lettere Revisione Visualizza Guida

Carattere Paragrafo

Normale Nessuna spaziatura Titolo 1 Titolo 2 Titolo 3 Titolo 4 Titolo 5 Titolo 6 Sottotitolo Enfasi del tutto Enfasi corsiva Enfasi intensa Enfasi (grasse)

- Controllo dei poligoni in particolare dell'attributo area, ubicazione e tipologia delle AREE COMPLESSE (aree verdi con all'interno alberi) e dei TAPPETI ERBOSI (prati con eventualmente alberi recensiti)
- Identificazione attributi rilevanti (es. Tipologia, area, codice ID, ubicazione)

Eventuale modifica dello schema DB di destinazione

- Definizione schema "aree_verdi" e "aree_blu" in PostGIS
- Aggiunta di indici spaziali (GIST che sta per *Generalized Search Tree*. È un tipo di indice flessibile che PostgreSQL usa per diversi tipi di dati, inclusi i dati spaziali (geometrie come punti, linee, poligoni, ecc.))
<https://postgis.net/workshops/postgis-intro/indexing.html>

Importazione dati

- Uso di shp2pgsql per esportare .shp in script SQL
- Verifica inserimento e proiezione
- Ripetizione per layer "blu"
- Mostrare entrambe le modalità possibili di importazione:

Metodo A - tramite "Esporta"

Fai clic destro sul layer che vuoi importare (es. uno shapefile)

Pagina 1 di 3 34 di 432 parole Completamento del testo: atteso Accessibilità: verifica

12:36 15/05/2023

Training on the Job - Modulo 1

Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database



requisiti informativi del progetto

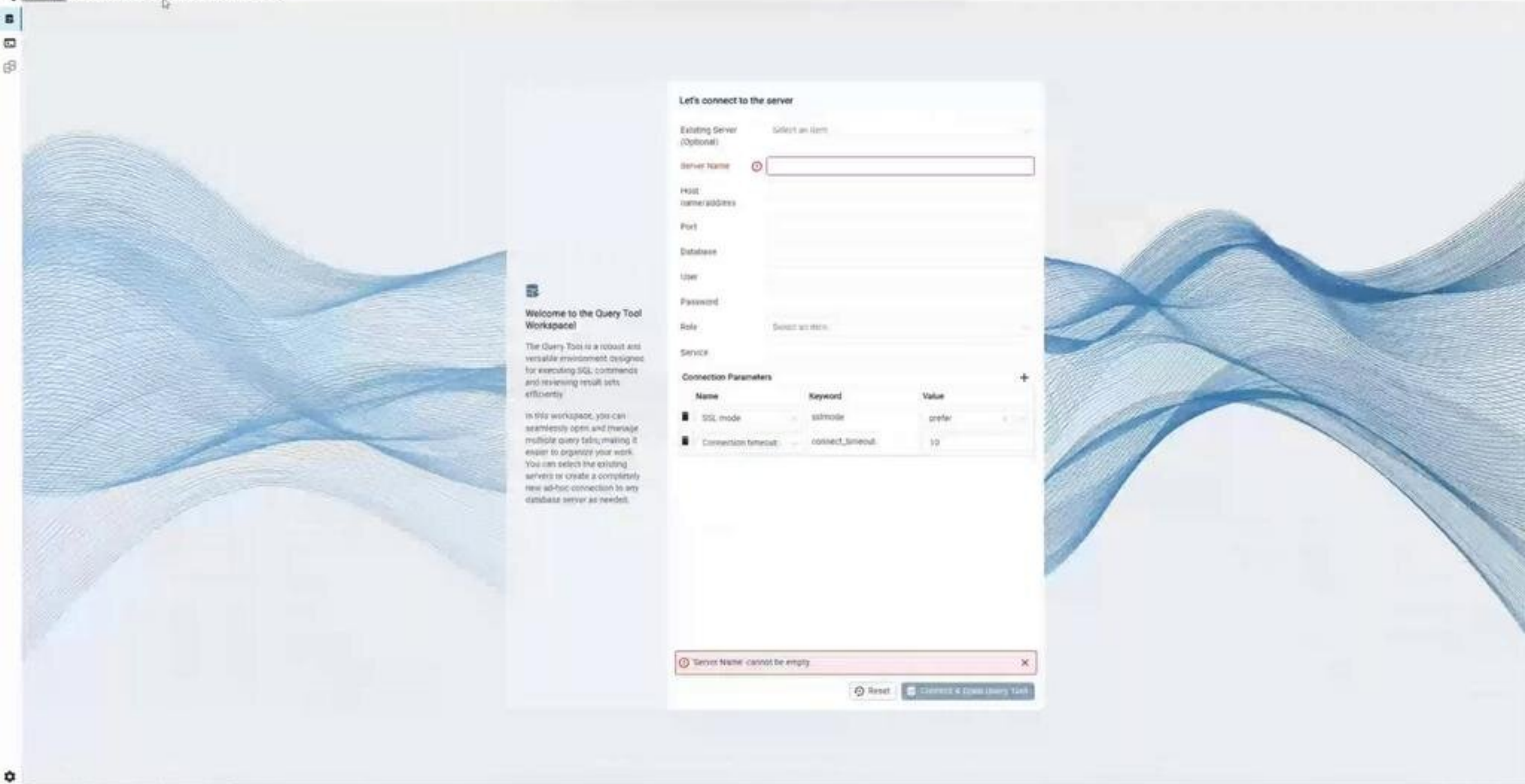
1 Fase (8 giornate)

1. Definizione del diagramma E/R e delle specifiche relative per la creazione dell'infrastruttura DB (1 giornata) – in sede ISPRA (già realizzata)
2. Installazione di PostGIS, definizione di un ambiente di lavoro, creazione delle tabelle ritenute necessarie per il progetto. Test funzionale. (1/2 giornata)
3. Installazione di Geoserver, definizione dei layer e integrazione delle tabelle. Test funzionale. (1/2 giornata)
4. Importazione dei dati da file Shape o altri db georeferenziati per la costruzione degli elementi verdi e blu (1 e ½ giornate)
5. Importazione dei dati della popolazione su raster fornito da ISPRA (1 e ½ giornate)
6. Installazione ambiente di lavoro usando Eclipse, definizione classi per accesso DB (2 giornate)
7. Confronto su sviluppo del geodatabase - (1 giornata) - in sede ISPRA

2 fase (24 giornate)

I

1. Calcolo dei punti di ingresso delle aree verdi e blu (4 giornate)
2. Calcolo della distanza minima fra i centroidi dei poligoni rappresentanti la popolazione e i punti di ingresso delle aree verdi e blu (2 giornate)
3. Stesura del codice per la visualizzazione degli elementi su mappa georeferenziata interattiva (10 giornate)
4. Confronto su attività di sviluppo parte client – (2 giornate) – in sede ISPRA
5. Sviluppo maschere per esportazione dati su formati definiti nella fase di analisi (2 giornate)
6. Sviluppo codice per la creazione di grafici/tabelle (2 giornate)
7. Verifica funzionale del progetto di formazione (1 giornata) – in sede ISPRA
8. Test delle competenze acquisite (1 giornata) – eventualmente in sede ISPRA



Welcome to the Query Tool Workspace!

The Query Tool is a robust and versatile environment designed for executing SQL commands and reviewing result sets efficiently.

In this workspace, you can seamlessly open and manage multiple query tabs, making it easier to organize your work. You can select the existing servers or create a completely new ad-hoc connection to any database server as needed.

Let's connect to the server

Existing Server (Optional)

Server Name

Host name/address

Port

Database

User

Password

Role

Service

Name	Keyword	Value
SSL mode	sslmode	prefer
Connection timeout	connect_timeout	10

Server Name cannot be empty



File Explorer window showing the 'Downloads' folder. The window title is 'Download'. The address bar shows 'Questo PC > Download'. The left sidebar shows the 'Downloads' folder selected under 'Accesso rapido'. The main pane displays a list of files with columns for 'Nome', 'Ultima modifica', 'Tipo', and 'Dimensione'. A context menu is open over the file 'eclipse-jee-2021-09-R-win32-x86_64.exe', showing options like 'Apri', 'Apri in una nuova finestra', 'Salva tutto...', 'Aggiungi a Start', 'Condividi', 'Apri con...', 'Dare accesso a', 'Ripristina versioni precedenti', 'Invi a', 'Taglia', 'Copia', 'Crea collegamento', 'Elimina', 'Rinomina', and 'Proprietà'.

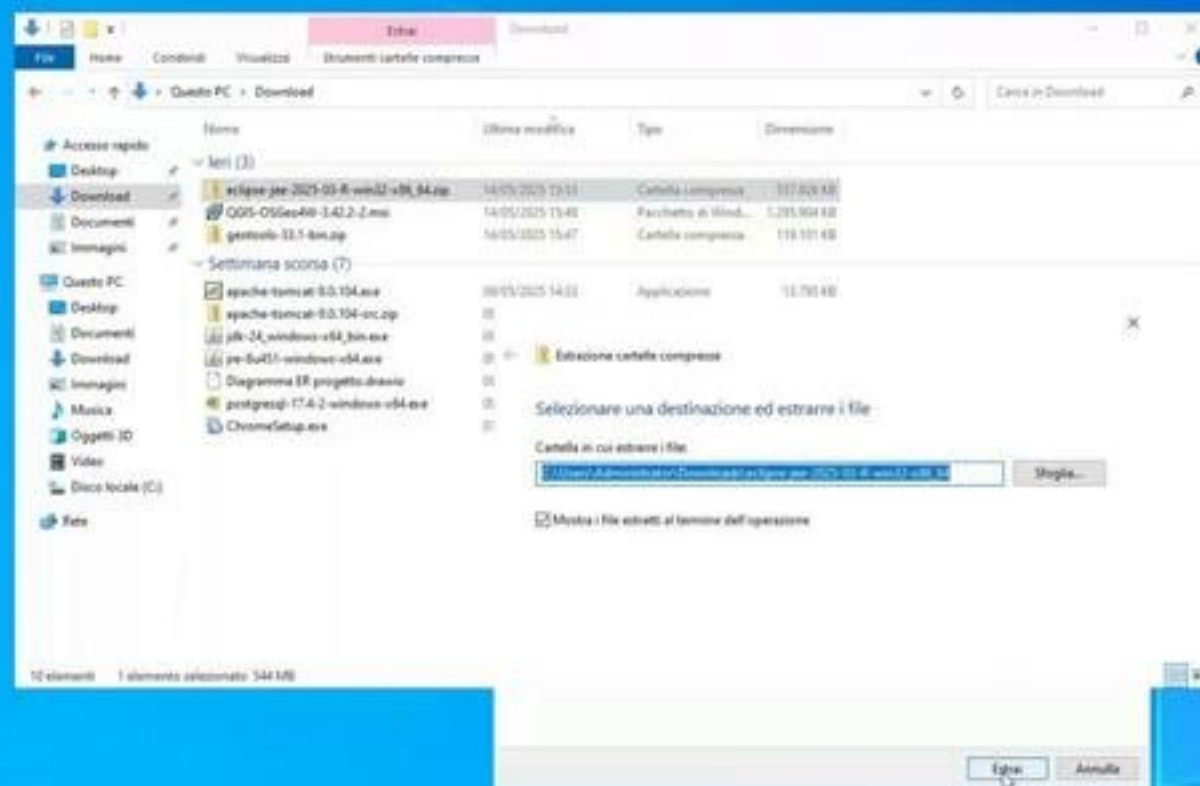
Nome	Ultima modifica	Tipo	Dimensione
Aprile (3)			
eclipse-jee-2021-09-R-win32-x86_64.exe	14/05/2021 13:12	Cartella compressa	107.824 KB
QGIS-OSGeo4W-1.42.2-Linux		File di testo	1.285.564 KB
gestools-32.1-bin.zip		File compresso	178.101 KB
Settimana scorsa (7)			
apache-tomcat-9.0.104.exe		Esecutore	13.791 KB
apache-tomcat-9.0.104-enc		File compresso	11.126 KB
jdk-24-windows-x64_bin.exe		Esecutore	210.796 KB
pre-5x431-windows-x64.exe		Esecutore	88.504 KB
Diagramma ER progetto di		File di testo	28 KB
postgres-17.4-2-windows		Esecutore	345.553 KB
ChromeSetup.exe		Esecutore	11.109 KB

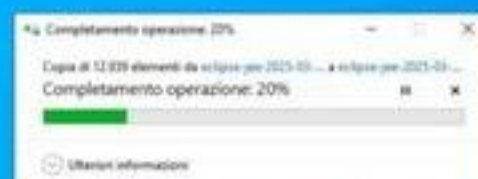
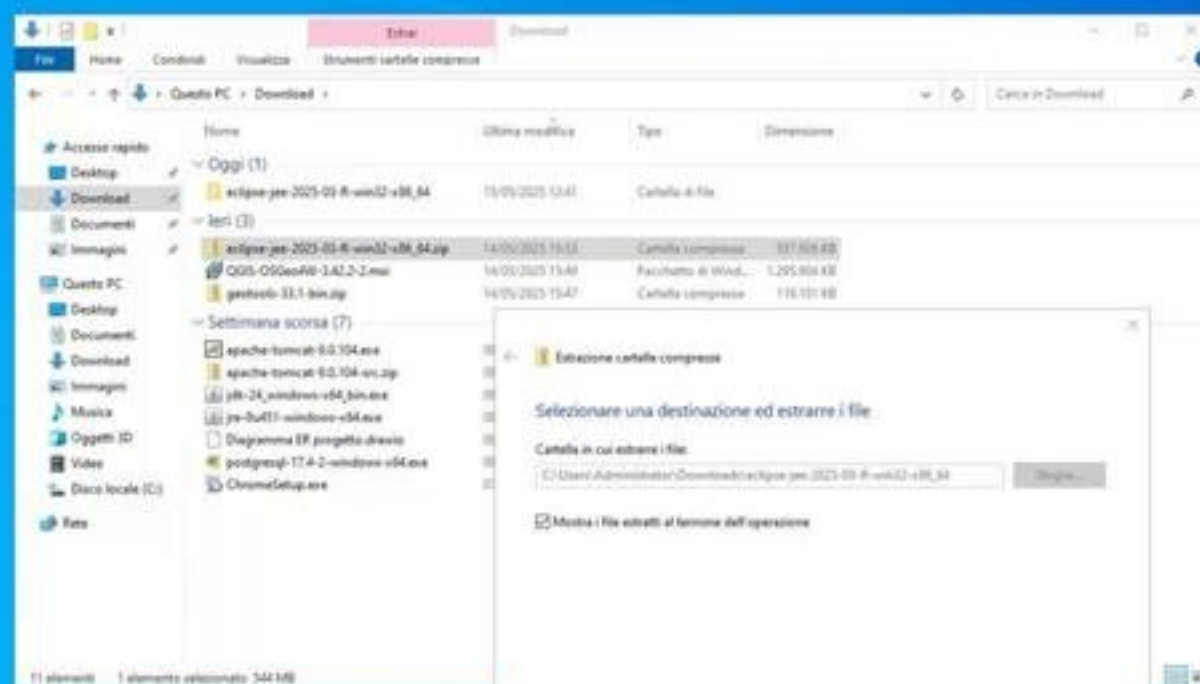


Scrivi qui il testo da cercare.



1248
15/05/2025





Modulo 4 – Installazione ambiente di lavoro con Eclipse + Jersey + Tomcat + Postgresql

Durata prevista: 2 giornate

Obiettivi

- Configurare un progetto Java Web in Eclipse usando Jersey come framework REST
- Gestire manualmente le librerie necessarie (Jersey, PostgreSQL, GeoTools)
- Sviluppare le classi di accesso al database e i servizi REST
- Esporre servizi RESTful per interagire con dati spaziali

Fasi operative

Installazione e configurazione Eclipse

- Scaricare e installare Eclipse IDE for Enterprise Java and Web Developers
- Impostare Java JDK (11 consigliato, compatibile con Jersey 3.x)
 - Vai su: `Window > Preferences > Java > Installed JREs`
 - Clicca su **Add...**, seleziona **Standard VM** e clicca **Next**
 - Inserisci il percorso della JDK (es. `C:/Program Files/Java/jdk-17`)
 - Dai un nome (es. `Java 17`) e clicca **Finish**
 - Seleziona il JDK appena aggiunto come predefinito
- Creare un nuovo Dynamic Web Project in Eclipse:

Modulo 4 – Installazione ambiente di lavoro con Eclipse + Jersey + Tomcat + Postgresql

Durata prevista: 2 giornate

Obiettivi

- Configurare un progetto Java Web in Eclipse usando Jersey come framework REST
- Gestire manualmente le librerie necessarie (Jersey, PostgreSQL, GeoTools)
- Sviluppare le classi di accesso al database e i servizi REST
- Esporre servizi RESTful per interagire con dati spaziali

Fasi operative

• Installazione e configurazione Eclipse

- Scaricare e installare Eclipse IDE for Enterprise Java and Web Developers
- Impostare Java JDK (11 consigliato, compatibile con Jersey 3.x)
 - Vai su: `Window > Preferences > Java > Installed JREs`
 - Clicca su **Add...**, seleziona **Standard VM** e clicca **Next**
 - Inserisci il percorso della JDK (es. `C:/Program Files/Java/jdk-17`)
 - Dai un nome (es. `Java 17`) e clicca **Finish**
 - Seleziona il JDK appena aggiunto come predefinito
- Creare un nuovo Dynamic Web Project in Eclipse:

Un Dynamic Web Project in Eclipse è una struttura progettuale pensata per lo sviluppo di applicazioni web Java (servlet, JSP, API REST). Consente di generare contenuti dinamici gestiti da un server applicativo come Tomcat.



- Crea Server:

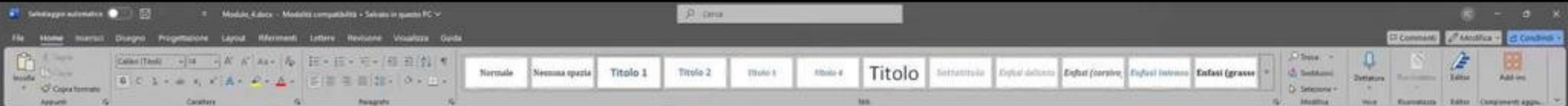
- Click destro nello spazio bianco > **New > Server**
- Seleziona Tomcat v9.0 Server e specifica la cartella di installazione di Tomcat
- Premi **Finish**
 - Trascina il tuo progetto nel server o clic destro su Tomcat > **Add and Remove...**
- Seleziona il tuo progetto
- Clic su **Add > Finish**

Aggiunta manuale delle librerie

- Scaricare i JAR di Jersey, JDBC PostgreSQL e GeoTools
- <https://eclipse-ee4j.github.io/jersey.github.io/download.html>
- <https://jdbc.postgresql.org/download/>
- <https://sourceforge.net/projects/geotools/>
- Inserire i JAR in WebContent/WEB-INF/lib/
- Verificare che i JAR siano referenziati nel build path del progetto

Configurazione web.xml

- Definizione servlet JerseyREST
- Associazione URL-pattern /api/*



Creazione dei package e classi

- Ad esempio:
 - com.progetto.ispra.db: gestione connessione DB PostgreSQL/PostGIS
 - com.progetto.ispra.dao: classi dao
 - com.progetto.ispra.model: classi dto e entity
 - com.progetto.ispra.rest: classi REST
 - com.progetto.ispra.service: classi service
- Esempio di connessione al DB
- Classe DBConnectionManager con JDBC DriverManager

Esempio REST API con Jersey

- Implementazione classe AreaResource con metodi GET e POST
- Uso di MediaType.APPLICATION_JSON e Response

Deploy e test

- Deploy del progetto su Tomcat
- Test con Postman o browser

Training on the Job - Modulo 1

Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto

Training on the Job - Modulo 1

Gestione Database e Pubblicazione Geospaziale con PostGIS e GeoServer

1. Definizione del Diagramma E/R e Specifiche per la Creazione dell'Infrastruttura DB

Obiettivi:

- Comprendere la struttura logica dei dati
- Definire le entità, relazioni e attributi
- Produrre un diagramma E/R coerente con i requisiti del progetto
- Redigere specifiche tecniche per la realizzazione fisica del database

Attività previste:

1. Analisi dei requisiti informativi del progetto



Sicurezza di Windows



Le credenziali specificate non funzionano

L'amministratore del sistema non consente l'utilizzo di credenziali salvate per l'accesso al computer remoto TOTJ perché l'identità di tale computer non è stata completamente verificata. Immettere le nuove credenziali.

Administrator

Password

.....



IMSET\administrator

Tentativo di accesso non riuscito

[Altre opzioni](#)

OK

Annulla



FUTURA



Scopri di più
su questa...



Francesco
Cavedon



MXGRAPH



Questo PC



Hydros_Dom...



Rete



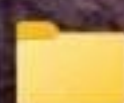
Training_on...



apache-to...



Manuali PLEX



Buste Paga



GoTo



Cestino

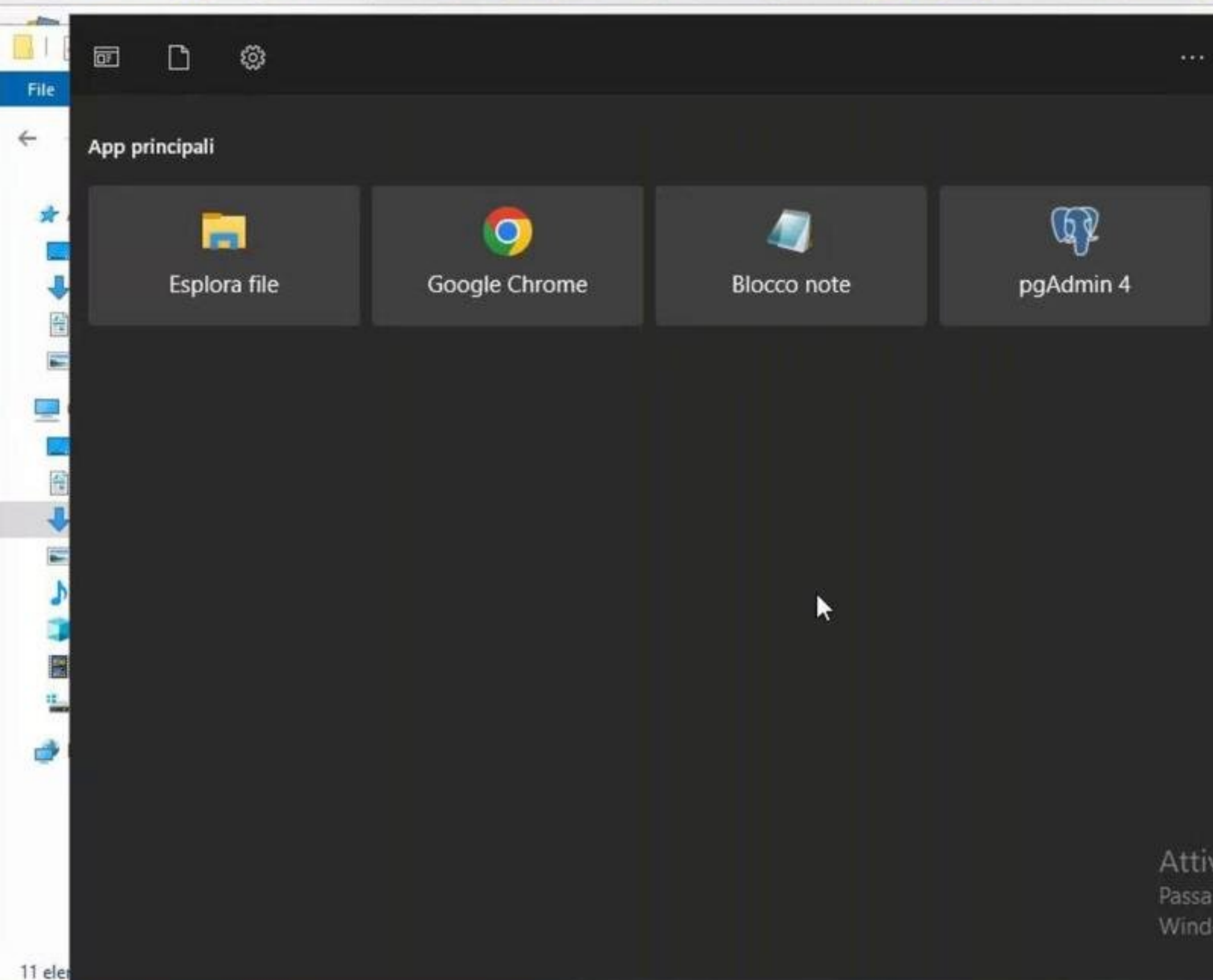


23°



14:04
15/05/2025





Attiva Windows
Passa a Impostazioni per attivare Windows.

- Sviluppare le classi di accesso al database e i servizi REST
- Esporre servizi RESTful per interagire con dati spaziali

Fasi operative

Installazione e configurazione Eclipse

- Scaricare e installare Eclipse IDE for Enterprise Java and Web Developers
- Impostare Java JDK (11 consigliato, compatibile con Jersey 3.x)
 - Vai su: Window > Preferences > Java > Installed JREs
 - Clicca su **Add...**, seleziona **Standard VM** e clicca **Next**
 - Inserisci il percorso della JDK (es. C:/Program Files/Java/jdk-17)
 - Dai un nome (es. Java 17) e clicca **Finish**
 - Seleziona il JDK appena aggiunto come predefinito
- Creare un nuovo Dynamic Web Project in Eclipse:

Un Dynamic Web Project in Eclipse è una struttura progettuale pensata per lo sviluppo di applicazioni web Java (servlet, JSP, API REST). Consente di generare contenuti dinamici gestiti da un server applicativo come Tomcat.

Ecco come funziona e come si crea: