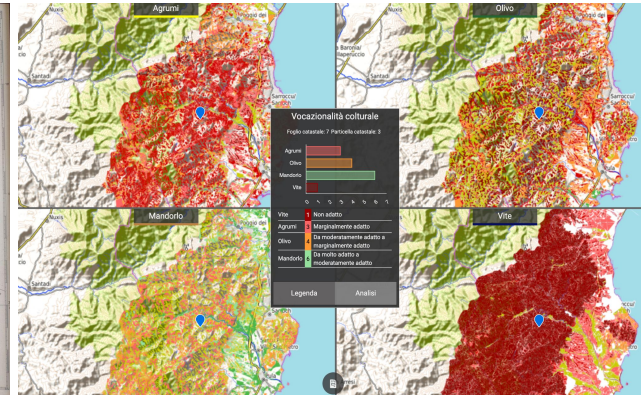




*MapWall*

## MapWall è un framework client-side Javascript per la realizzazione rapida di dashboard WebGIS

- Sviluppato nell'ambito del progetto *MUR Cagliari2020*
- UX uniforme e personalizzabile
- Ottimizzato per desktop e grandi superfici touch (control room)
- Basato su *OpenLayers* e *InteractJS*



# MapWall

- Integra Openlayers sottostante con:
  - Widget grafici mobili sulla mappa
  - Classi helper
  - API
- **L'utente organizza i componenti** dell'applicazione sopra la mappa per comporre la propria dashboard
- Modello interazione basato su **Watcher** ovvero popup associati a feature geografiche
- **Non è un CMS** per il GIS: i contenuti e la logica applicativa rimangono a carico dello sviluppatore
- **Template** con layout di base facilitano lo sviluppo delle applicazioni

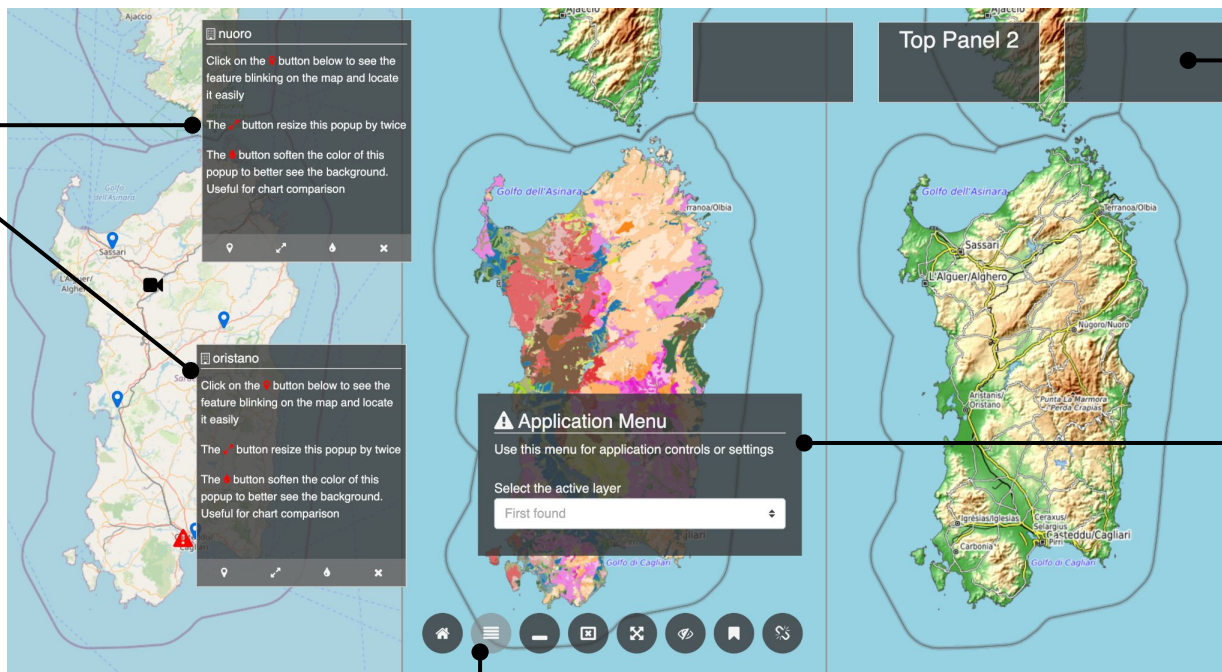


<http://mapwall.crs4.it/>

# MapWall

**Watcher**  
Visualizzati al click su una feature, il comportamento e il layout sono definiti attraverso callback. Aggiornabili attraverso apposite API

**MultiMap Layout**  
Si possono definire più mappe in un layout personalizzato. L'interazione con le singole mappe può essere condivisa o indipendente



**Panel**  
Configurati in *config.json*, se presenti sono sempre visibili, utili per informazioni application-wide

**Application Menu**  
Widget opzionale, utile per impostazioni e filtri. Mostrato sopra la toolbar al click sul relativo pulsante. Layout definito sull'html; emette evento *appmenuShow* quando viene visualizzato

**Toolbar**  
Sempre visibile e interamente configurabile da *config.json* permette di definire pulsanti custom o di utilizzare funzioni standard di mapwall come chiusura o hide dei watcher aperti, mostra application menu, restore layout, ecc.





# *Demo Frontend*

<http://webgis.crs4.it/>

|        |  |      |        |           |   |
|--------|--|------|--------|-----------|---|
| Demo 1 | ● <b>Frontend:</b> Dashboard MapWall con dati esposti in WFS e WMS da <b>Geoserver</b> | Live | Editor | Geoserver | ↓ |
|--------|--|------|--------|-----------|---|


## Obiettivo: Esposizione Layer vettoriale e raster con servizi OGC standard

1. Panoramica struttura template minimal di MapWall
2. Caricamento layer WFS esposto da Geoserver
3. Interrogazione feature vettoriale
4. Caricamento layer raster WMS esposto da Geoserver



<http://mapwall.crs4.it/>

<http://webgis.crs4.it/>

|        |   |      |        |        |   |
|--------|---|------|--------|--------|---|
| Demo 2 | <p>● <b>Frontend:</b> Aggiornamento <i>Demo 1</i> con dati esposti da backend custom in <b>NodeJS</b> attraverso API Rest</p> | Live | Editor | NodeJS |  |
|--------|---|------|--------|--------|---|

## Obiettivo: Upgrade applicazione *Demo 1* con nuovo layer da API custom

1. GET features in JSON da API Rest backend NodeJS
2. Dichiarazione *layer* vettoriale con array di feature
3. Popup MapWall personalizzato per nuovo layer



<http://mapwall.crs4.it/>

<http://webgis.crs4.it/>

|        |   |  |  |  |      |        |                |   |
|--------|---|--|--|--|------|--------|----------------|---|
| Demo 3 | <p>● <b>Frontend:</b> Dashboard MapWall <u>multi mappa</u> con dati esposti come Tiled Map da backend <b>mod_tile</b></p> |  |  |  | Live | Editor | Map 1<br>Map 2 | ↓ |
|--------|---|--|--|--|------|--------|----------------|---|

## Obiettivo: Visualizzazione multi-map di immagini Raster con layer Tile

1. Setup modalità multi-map di MapWall (config e layout)
2. Dichiarazione layer *Tile* con dati esposti da backend *mod\_tile*
3. Interrogazione simultanea multi mappa



<http://mapwall.crs4.it/>