

Project acronym: TRITON

Project title: Development of management tools and directives for immediate protection of biodiversity in coastal areas affected by sea erosion and establishment of appropriate environmental control systems

Deliverable No. 5.1

Development of web-based GIS and Plan of training

Delivery date: 14-09-20

Lead Partner

Technical support

Project Partners

PROGRAMME

Interreg V-A Greece-Italy Programme 2014-2020

AXIS

Axis 2 (i.e. Integrated Environmental Management)

THEMATIC OBJECTIVES

06 – Preserving and protecting the environment and promoting resource efficiency

PROJECT ACRONYM

TRITON

PROJECT WEBSITE URL

www.interregtriton.eu

DELIVERABLE NUMBER

No. 5.1

TITLE OF DELIVERABLE

5.1 Development of web-based GIS and Plan of training

WORK PACKAGE/TASK N°

WP5; Task 5.1

NAME OF ACTIVITY

Development of web-based GIS and Plan of training

PARTNER IN CHARGE (AUTHOR)

PB4 - EGTC Efxini Poli

PARTNERS INVOLVED

LB

STATUS

Final Version

DUE DATE

ADDRESSEE OF THE DOCUMENT¹

TRITON PROJECT PARTNERS; INTERREG V-A GREECE-ITALY PROGRAMME

DISTRIBUTION²

PP

Document Revision History

¹WPL (Work Package Leaders); PB (Project Beneficiaries); AP (Associates); Stakeholders; Decision Makers; Other (Specify)

²PU (Public); PP (Restricted to other program participants); CO (Confidential, only for members of the consortium)

Version	Date	Author/Reviewer	Changes
1.0 - Final	14/09/20	PB4 and LB	Integration of plan of training with Development of web-based Public Participatory GIS (PPGIS) operated through the PB web-site.
0.4 - Draft	15/12/2019	LB – Puglia Region	Plan on training
0.3 - Draft	24/09/2019	LB – Puglia Region	Version revised by LB1
0.2 - Draft	27/05/2019	LB – Puglia Region	Version revised by LB1
0.1 - Table of contents	14/05/2019	PB4 - EGTC Efxini Poli	First draft

Abstract

This Deliverable describe the preparation activities carried out before the realization of project training activities that consist, primarily, in the setting up of the programme of the lessons and the selection of the students of the Summer School, the planning of the training days and the web educational campaign.

Moreover, in this deliverable the development of a web-based Public Participatory GIS (PPGIS) and its components is described. The web-based GIS was implemented for the TRITON Greek pilot study site, covering the broader area of the Municipality Ieras Poleos Messolonghiou, Western Greece. The system collects environmental, meteorological, oceanographic, hydrographic, hydrological data, combined with land-use data, erosion/accretion rates, etc. from external platforms and data providers like NOAA, Copernicus, Corine, etc. and processes and visualizes these data in real-time mode. Data represent historical, present and future conditions; therefore, the user is capable to create smart maps of any parameter moving back and forth in time and zooming in at any part of the study area. A well-organized and user-friendly menu aids the user in operating the system. Through the web-GIS system the Municipality Ieras Poleos Messolonghiou will be able to inform its citizens on extreme weather events (storms, floods, waves, etc.), improve spatial management and visualize land use changes and coastal management emerging issues, as coastal erosion hotspots and areas vulnerable to sea level rise and climate change.

Table of Contents

1. Plan of training	6
The Call for the selection of the students	6
The programme of the International Summer School "Sustainable management of coastal heritage and actions to contrast erosion"	7
The subjects of study	7
The programme of the Training Days	9
The web based Educational campaign based on video clips and an handbook	9
2. Web GIS creation methodology	10
Web Mapping	10
WebGIS page creation	11
System architecture for web mapping	13
Leaflet.js	15
TRITON's webGIS page	19
TRITON's webGIS structure	19
TRITON's webGIS System architecture	19
TRITON's web mapping application (Client-side)	20
TRITON'S webGIS page elements	22
TRITON's webGIS Data integration	29
Bathymetric Thematic Layers	30
Topographic Thematic Layers	31
Shoreline Evolution Thematic Layers	31
Land Cover Thematic Layers	33
Seagrass Thematic Layers	34
Hydrography Thematic Layers	36
Oceanographic Thematic Layers	38
Weather Thematic Layers	42
References	45



Lead Partner



Technical support



Project Partners



1. Plan of training

This deliverable consist in the description of the planning and preparation activities for the training carried out in the Triton WP5.

The training is divided into the following main patterns:

- International Training Course (Summer School) on sustainable management of coastal heritage and on actions to combat erosion, including the use of performance and assessment indicators to be held in Puglia and in the Greek Western Region.
- Training course for experts of local public bodies and the technicians involved in the preparation of municipal plans of the coasts, in n. 5 days to be held in Puglia;
- A web based Educational campaign based on video clips and an handbook;

LB selected the teachers and the external experts for the Summer School, Training days and the educational video-clips and defined the related programmes with the support of all PPs in the first half of 2019.

It is important to underline the strict connection of those activities with the studies produced in the WP3, that will be presented during the lessons, and the analysis of the WP4 Pilot case: during the summer school pilot areas will be analysed and visited by the students.

The Call for the selection of the students

The call to attend the Triton project summer school on “Sustainable management of coastal heritage and actions to mitigate coastal erosion” was published on the Triton website on Jul 11, 2019 at the following link: <https://www.interregtriton.eu/call-to-attend-the-triton-project-summer-school-on-sustainable-management-of-coastal-heritage-and-actions-to-mitigate-coastal-erosion/>

The call aimed at selecting the 16 students to be involved in the Summer School. The school was targeted tor public officials and professionals and aimed at improving their skills in coastal management planning and actions against coastal erosion.

Specifically, the students involved in the call were Greeks and Italians residing in the INTERREG V-A GREECE-ITALY PROGRAMME area; the participation in the Summer School was for free and the students have benefited of travel and accommodation costs reimbursement up to 700 euros. The course was held in English and delivered by LB in collaboration with ARTI and all the Triton project partners and included lectures, visits and project works.

To participate candidates had to complete the form attached linked below (Annex 1), complete with Annex 2 to be sent exclusively at the PEC address: arti@pec.rupar.puglia.it by 12:00 of 12 September 2019.



Triton - Call for students of summer school



Annexes 1 - 2



Triton international Summer School Programme

In the Annex 2 the complete text of the call is also available.

The programme of the International Summer School "Sustainable management of coastal heritage and actions to contrast erosion"

The Triton project Summer School entitled "Sustainable management of coastal heritage and actions to contrast erosion" was a high-level cross-border course, aimed at improving the skills of public administration staff and technicians involved in coastal management and contrast actions to erosive phenomena.

The Summer School was divided into six training days, divided into two modules: the first one will be held in Western Greece and the second one in Puglia

a) the Western Greece module includes 2 days of lessons at the University of Patras and a study visit to the Patraikos gulf (which is affected by relevant erosion phenomena), respectively on 24, 25 and 26 September 2019.

b) the Puglia Module includes a study visit at a pilot case of Bari and 2 days of lessons in Bari on days of 1, 2 and 3 October 2019.

The subjects of study

The contents of the Summer School focussed on the following topics:

- Technical and geological aspects of coastal erosion: impact analysis;
- Outline of the territorial marketing of the coast and socio-economic-environmental impacts;

Lead Partner

Technical support

Project Partners

- Integrated management of the coast between GIS and SID (Maritime State Property Information System) and ICT planning and management tools;
- Use of software for integrated WEB GIS coastal planning;
- The EUSAIR strategy (Ionian Adriatic Euroregion) focused to the sustainable use of the coast;
- Key performance indicators (KPI) for coastal management planning in order to prevent the corrosive phenomenon;
- Analysis of the Bolkstein Directive and application effects between Puglia and Western Greece;
- Erosion and coastal submersion forecast scenario models;
- Use of GIS spatial planning tools to support decision-making systems and administrative policy guidelines (DSS).
- Good practices for contrasting coastal erosion, with study visits.

Professor Nicolò Carnimeo (University of Bari) was the scientific manager of the course on behalf of Regione Puglia and ARTI. Among the teachers there were also the representatives of the others Triton project partners (University of Patras, Puglia Region, Euro-Mediterranean Center on Climate Change-CMCC, Region of Western Greece, Network of European cities for sustainable development).

The teachers are listed below:

1. Giuseppe Roberto Tomasicchio (Wave action and response of beaches, with some applications for the Apulian coastline)
2. Simona Bramato (Morphological evolution, management tools and GIS)
3. Nikos Depountis (Good geotechnical practices for contrasting coastal erosion)
4. K. Nikolakopoulos (Good remote sensing practices for contrasting coastal erosion)
5. D. Christodoulou (Good marine practices for contrasting coastal erosion)
6. E. Fakiris (Technical and geological aspects and monitoring of coastal erosion)
7. G. Papatheodorou (Good marine practices for contrasting coastal erosion)
8. Elisa Furlan (Technical and geological aspects)
9. Silvia Torresan (Technical and geological aspects)
10. Lara Marchetta (State property management, Eusair strategy, position paper for the Adriatic Ionian strategy)

11. Giuseppe Delle Foglie (Comparison between Greece and Italy and a proposal based on the role of coastal zone management)

The Annex 1 "Triton Summer School brochure" contains additional information on the school programme.

The programme of the Training Days

The training days took place in Bari in October and November 2019.

They were finalized to overcome the gap at regional level for technicians to support the policy makers in actions and intervention connected to coastal planning and the mitigation of coastal erosion.

The pattern was composed by five modules to give an adequate training on:

- Legal terms: Principle and Legislation
- Planning and environmental and landscape protection
- The coastal planning in Puglia Region and Adriatic - Ionian area
- Maritime Domain and Business Development for enterprises in state concession
- State Domain process, sanction and tenders

In the Annex 3 the official programme of the training days (in Italian) is also available.

The web based Educational campaign based on video clips and an handbook

Finally, the project training activities comprise a set of educational videoclips and an handbook targeted to a wider public interested to the themes of coastal planning and erosion.

The video clips were designed based on a re-elaboration of the project lessons, interviews with participants and teachers of the summer school, an overview of the pilot cases.

The handbook was designed to collect the materials produced by the training activities, with the contribution of the teachers but also of the students, in a project work perspective. The handbook will also contain: 1) an articulated proposal on the role of coastal zone management in the new

legislation for the cross-border cooperation Greece-Italy 2021-2026; 2) a position paper for the Adriatic-Ionian Strategy (EUSAIR) and 3) a framework for local authorities and regional stakeholders for access to EIB / EIF / EASME funds for interventions on the mitigation of the effects of erosion.

2. Web GIS creation methodology

Web Mapping

Web mapping clients play a significant role in Geoportals of Spatial Data Infrastructures (SDI) allowing the visualization of spatial data from several sources. Likewise, these clients may be part of Web-based Geographic Information Systems (GIS) applications, in which users can directly interact with SDI services, visualize, query and integrate them with local data and GIS tools. Web mapping clients are pieces of software (applications, viewers, libraries and frameworks, among others) that either provide or extend a Web-based mapping component to view and interact with maps from remote sources on the Internet. Some of the projects that provide such a mapping component use pure client-side technology whereas the vast majority rely on server-side features allowing advanced tasks to be performed, such as security, user and group administration, advanced printing capabilities, spatial analysis support and customization of graphic user interface controls and functionalities, among others. The process of developing a web map can be broken down into three stages. The first stage involves gathering and geoprocessing of original data. Second stage includes the uploading of geospatial data to a server and the creation of a mapping service which serves the map data to user's web browser and final stage includes the web design process of mapping interface using a web mapping client library (Smith, 2016).



Figure Errore. Nel documento non esiste testo dello stile specificato..1 Stages of online interactive mapping design (Smith, 2016).

In order to publish spatial data in a Web map, Application Programming Interfaces (APIs) frameworks are often used as the intermediate to translate low level code to objects on map. In fact, APIs are a set of building blocks that you invoke using a programming language. They provide a set of classes and functions that help to avoid writing all the low-level code to perform specific actions or to display an interactive map image and drawing a new layer on it. Instead, you can just create a new map

object, create a new layer object, and methods such as `layer.addTo(map)`. The API abstracts the complexity of the task and makes it easy to focus on the mapping aspects of your application, rather than spending time on the low-level logistics.

There are many APIs designed specifically for the purpose of making web maps that make the creation and configuration of Web mapping clients easier. Below is a graph showing the relation between free and open source Web mapping clients.

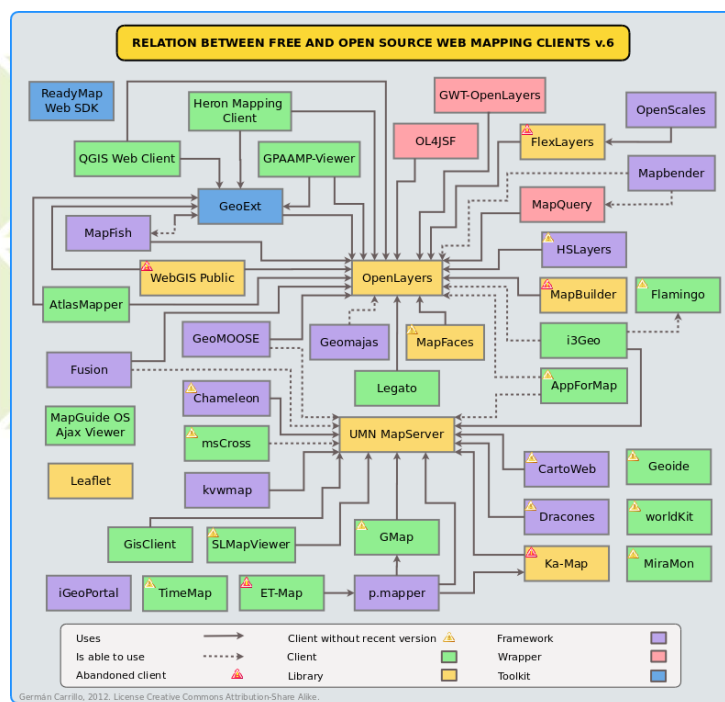


Figure Errore. Nel documento non esiste testo dello stile specificato..2 Relation between free and open source web mapping clients (Carrillo, 2012).

WebGIS page creation

In order to add cartographic objects to a map project, a HyperText Markup Language (HTML) index page is required to get a suitable structure. The map is placed in an element within the HTML file. The TRITON web GIS page is based on the combination of an HTML5 page with Cascading Style Sheets (CSS) styling components which give the control of the style and visual presentation of the web page, JavaScript client-side scripts which give the ability to add interactivity to the web page and python scripts for the manipulation of retrieved data.

The TRITON webpage follows the common HTML5 structure. Specifically, it begins with the HTML5 markup `<!DOCTYPE html>` declaration, which is an instruction to the web browser about the HTML version which the page is based on. The DOCTYPE is a small piece of code that is placed at the start of a web page to indicate which standard the respective page complies with. When a web page is accessed in a browser, the browser looks for a DOCTYPE declaration and examines it to decide whether the page is using modern web standards, such as HTML5, or if it was designed to work with older browsers. Afterwards, the browser uses this information to determine how to interpret the code and display it on the screen (West, 2012). The HTML5 page structure is based on several “elements” with particular purposes. The primary part of an HTML document consists of the following elements: `<html>`, `<head>`, `<style>` and `<body>` which are known as Document Structure Tags.

The `<html>` element is the root of an HTML document. This means that all the above elements, except the DOCTYPE declaration are nested within an `<html>` tag and a subsequent end tag `</html>` which shape a frame allowing specific contents and structures. The `<html>` element is often used to define the language of the content within the document. This is done by adding a “*lang*” attribute to the start tag that has a two-character language code as its value. (This applies only to the actual page content and not to the tag names). The following example defines the content language as Greek by adding the “*lang*” attribute with the value “*gr*” (West, 2012).

```
<html lang="gr">
```

The `<head>` element contains some overall information about the page, such as a title (`<title>` element) and a specification of the character encoding (e.g., Unicode UTF-8) for the document (`<meta>` charset attribute). A typical mobile-optimized website and the one that is used in TRITON’s website contains something like the following:

```
<meta name="viewport" content="width=device-width, initial-scale=1"/>
```

The width property controls the size of the viewport. It can be set to a specific number of pixels like or to the special value device-width value which is the width of the screen in CSS pixels at a scale of 100%. The initial-scale property controls the zoom level when the page is first loaded. The maximum-scale, minimum-scale, and user-scalable properties control how users are allowed to zoom the page in or out.

An important part inside the head element, is the linking of a wide variety of elements like fonts, styles, scripts etc. This means that when the page is rendered by the browser, it will take into consideration to link these files and show the content of the page in the right order and style and with the right animations.

The head element also manages the embedding of larger scripts like the Leaflet.js library and external assets like the Leaflet style sheet CSS file which will be described below. This is first based on a

<script> "src" attribute, including a relative path to the source location of the external library file and second on a <link> "rel" attribute, connecting the index page to the external CSS file. Every Leaflet.js plugin is embedded in the website within the head element and with the above way.

```
<script src="plugins/leaflet.js"> </script>

<link rel="stylesheet" href="css/leaflet.css"/>
```

The benefits of using a separate JavaScript file include:

- Separating the HTML markup and JavaScript code to make both more straightforward
- Separate files make maintenance easier
- When JavaScript files are cached, pages load more quickly

The <body> element usually contains the content that should be presented in the browser window. The body contains the most relevant content of an html document, and where the main elements of the page reside. It contains all the contents of an HTML document, such as text, hyperlinks, images, tables, lists, etc. If a JavaScript library is involved, the body element usually contains a <script> element which is used to define client-side scripting statements in JavaScript. It is the fundamental element to create animated and interactive map content and to integrate multimedia.

The common structure of a HTML5 website is presented below:

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Title of HTML page</title>
5     <style>
6       h1 {
7         color:#009900;
8         font-size:46px;
9       }
10      p {
11        font-size:17px;
12        margin-top:-25px;
13        margin-left:15px;
14      }
15    </style>
16  </head>
17
18  <body>
19    <h1>HTML headline tag</h1>
20    <section class="drawer"></section>
21    <script>var osmLink = ...;</script>
22  </body>
23 </html>
24
```

Defines version of HTML
 HTML root element
 Contains page HTML metadata
 Title of HTML page
 Defines styles for specific elements of HTML page (css elements)
 Hold content of HTML
 HTML headline tag
 Global or Event Attributes in HTML.
 Client-side scripts

Figure Errore. Nel documento non esiste testo dello stile specificato..3 HTML5 webpage structure

System architecture for web mapping

The HTML5 structure presented before is a base for the inclusion of cartographic contents. Nevertheless, these HTMLs file and data have to be stored on physical machines with an Internet Protocol address (IP address). It can take several different physical machines to create, serve, and use a web map. These are often depicted in diagrams as separate levels, or tiers of architecture. Usually web mapping requires the levels below:

- desktop workstations that are used by administrators and internal client applications. These machines will also be used to prepare data and develop maps.
- a database and/or file server holding all of your GIS data. This machine might be equipped with redundant storage mechanisms and regular backup scripts that prevent the loss of data.
- a geospatial web services server that has specialized software and processing power for drawing maps, responding to feature queries, and performing GIS analysis operations.
- a web server that acts as a web entry point into an organization's network. This is also called a proxy server. It is protected by firewalls that shield malicious traffic into your internal network. It's also a place where you can put web application code (such as HTML and JavaScript files) for a web map.
- many client applications that use the web map. These can be apps that run on a desktop workstation or they could be mobile apps. All clients must be able to make web requests through HTTP, and any client coming from outside the network must have an Internet connection.

When a web map is developed and tested, a single physical machine can be used to play all these roles. This is a common practice that keeps things simple and more economic. When a web map is developed depicting parts of the "real world", then individual systems for each of the roles described above are essential. A diagram of how a mapping web site interacts with the end user and the back-end programs running on the servers is presented in the Figure **Errore. Nel documento non esiste testo dello stile specificato.** below.

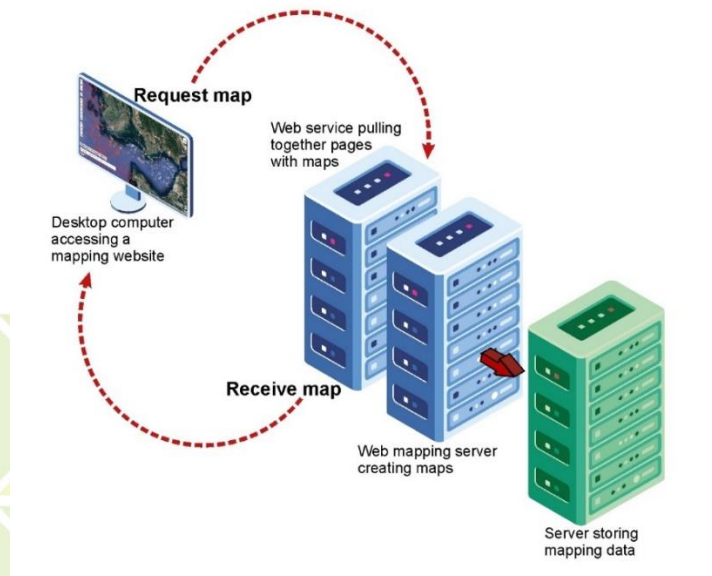


Figure Errore. Nel documento non esiste testo dello stile specificato..4 A diagram of how a mapping web site interacts with the end user and the back-end programs running on the servers.

Leaflet.js

At present, one of the most popular open-source web mapping APIs is Leaflet.js, that is designed to be lightweight (only 38kb of code at the time of this writing), mobile-friendly, and easy to get started with (Farkas, 2017a, 2017b; Muehlenhaus, 2013; Peterson, 2014). More specific, it is an open source JavaScript library originally developed by Vladimir Agafonkin in 2010 as "Web Maps API", for the CloudMade mapping provider, where Agafonkin worked at the time. In May 2011, CloudMade announced the first release of Leaflet, built from scratch but using parts of the old API code (Agafonkin, 2011). The aim of this project was to create an open source JavaScript API for users interested in building interactive and multimedia web mapping applications for common desktop and mobile browsers. Therefore, Leaflet is designed with simplicity, performance and usability in mind. It works efficiently across all major desktop and mobile platforms out of the box, taking advantage of HTML5 and CSS3 on modern browsers while being accessible on older ones too. A couple of workflow steps, an organizing HTML5 structure, a dynamic and accurate base map, a few JavaScript code blocks, and some geolocated data (based on leaflet elements), linked to style parameters in CSS3, are required to create simple, but professional interactive web maps with high performance (Edler & Vetter, 2019). Therefore, **simplicity and the open source characteristics of Leaflet is the reason that it was selected as the main mapping library of TRITON's web GIS page** and the reason that it has become extremely popular over the last years and quite a few companies, such as Mapbox, use it as a basis for their own APIs (MacWright, 2013). In addition, through its license, Leaflet.js explicitly encourages adaptation, study and distribution. What makes this library exceptional in its qualities is that it is especially concerned with providing compatibility

and interactivity for web maps across various browsers, including “web browsers on mobile devices” (Agafonkin, 2011). This fact suggests that using Leaflet.js is efficient in building professional web apps and pages which enable users to mash up data of various file formats with geographically referenced imagery (Mapbox, 2016; Reißig, 2016).

Basically, Leaflet is based on the creation of “Slippy Maps” with tiled base layers, allowing panning and zooming, and feature layers that user supplies. It handles various basic tasks like converting data to map layers and mouse interactions, and it's easy to extend with plugins.

Slippy Map is an AJAX component which defines a loose standard for how tiles should be requested based on 2 concepts, Zoom Levels and Tile Coordinates, in the browser. More specific, when user opens a web map in the browser a JavaScript component embedded in a web site runs and dynamically requests maps from a server in the background (without reloading the whole HTML page) to give a smooth “slippy zoomy” map browsing experience. The map image is built up of many little square images called “tiles” which are rendered and served from a “tile server”.

The structure of a typical “Slippy Map” tile request looks like this:

```
{url}/{zoom-level}/{x-coordinate}/{y-coordinate}.png
```

The {url} indicates possible server instances from which the map can draw tiles. For each loaded tile, the {zoom-level} indicates its zoom level, the {x-coordinate} indicates its horizontal coordinate, and {y-coordinate} indicates its vertical coordinate. Tiles are loaded “on the fly” and give the impression of a large seamless image. In addition, Zoom Levels define the scale of the map. At zoom level 0, an entire Mercator projection of the earth is contained in one 256px by 256px tile. At each further zoom level, the number of tiles increases by a factor of four and the spatial resolution (ground meters per pixel) of each tile roughly doubles. At zoom level 15, there are over a billion tiles each with a spatial resolution of 4.77 meters per pixel. As is illustrated in Figure Errore. **Nel documento non esiste testo dello stile specificato.**5, at any given zoom level, a specific tile can be identified by cartesian coordinates with (0, 0) starting at the top left of the map.

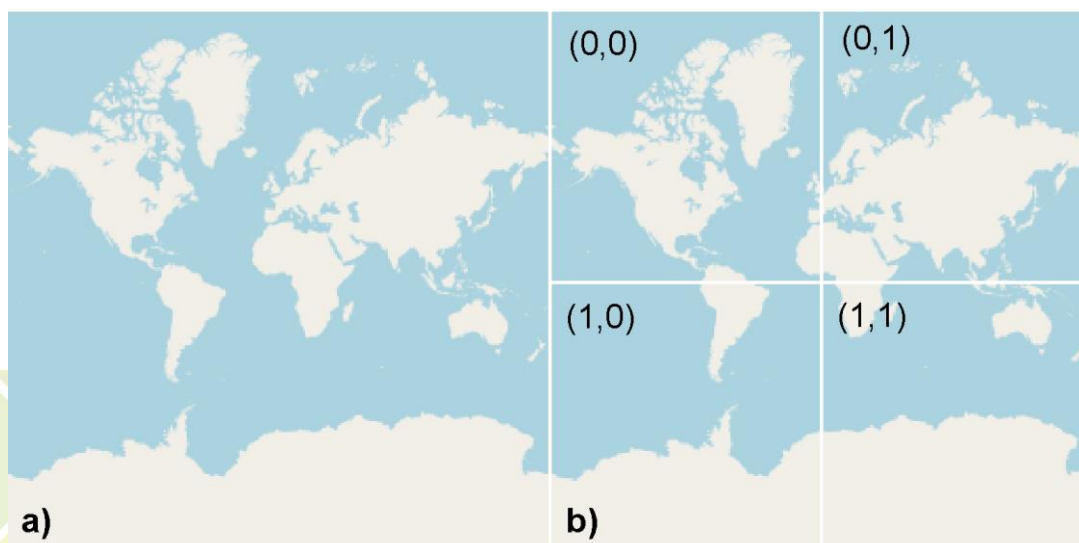


Figure Errore. Nel documento non esiste testo dello stile specificato..5 a) Zoom level 0 tile of a web map and b) cartesian coordinates of Zoom level 1 tiles of a web map.

The implementation of this "slippy maps" is provided by a JavaScript library, like Leaflet and it is the most popular way to display and navigate maps, replacing other methods such as Web Map Services (WMS) which typically display a single large image. Each time the user pans, most of the tiles are still relevant, and can be kept displayed, while new tiles are fetched. This improves the user experience, compared to fetching a single map image for the whole viewport. It also allows individual tiles to be pre-computed, a task easy to parallelize. Also, displaying rendered images served from a web server is less computationally demanding than rendering images in the browser.

As described above, Leaflet is characterized through its simplicity. In the Table 1 below the most important features that are built-in Leaflet JavaScript API are illustrated. As can be seen, one of the most important features Leaflet has, is the ability to support almost every browser in Desktop and Mobile devices.

Table 1. Leaflet.js built-in features (Agafonkin, 2011).

Layers Out of the Box	Customization Features	Browser Support
<ul style="list-style-type: none"> • Tile layers, WMS • Markers, Popups • Vector layers: polylines, polygons, circles, rectangles • Image overlays 	<ul style="list-style-type: none"> • Pure CSS3 popups and controls for easy restyling • Image- and HTML-based markers • A simple interface for custom map layers and controls 	<p>Desktop</p> <ul style="list-style-type: none"> • Chrome and any browser based on chromium technology

<ul style="list-style-type: none"> GeoJSON 	<ul style="list-style-type: none"> Custom map projections (with EPSG:3857/4326/3395 out of the box) Powerful OOP facilities for extending existing classes 	<ul style="list-style-type: none"> Firefox Safari 5+ Opera 12+ IE 7–11 Edge
Interaction Features <ul style="list-style-type: none"> Drag panning with inertia Scroll wheel zoom Pinch-zoom on mobile Double click zoom Zoom to area (shift-drag) Keyboard navigation Events: click, mouseover, etc. Marker dragging 	Performance Features <ul style="list-style-type: none"> Hardware acceleration on mobile makes it feel as smooth as native apps Utilizing CSS3 features to make panning and zooming really smooth Smart polyline/polygon rendering with dynamic clipping and simplification makes it very fast Modular build system for leaving out features you don't need Tap delay elimination on mobile 	Mobile <ul style="list-style-type: none"> Safari for iOS 7+ Android browser 2.2+, 3.1+, 4+ Chrome for mobile Firefox for mobile IE10+ for Win8 devices
Visual Features <ul style="list-style-type: none"> Zoom and pan animation Tile and popup fade animation Very nice default design for markers, popups and map controls 	Map Controls <ul style="list-style-type: none"> Zoom buttons Attribution Layer switcher Scale 	Misc <ul style="list-style-type: none"> Extremely lightweight No external dependencies

While Leaflet is designed to be as lightweight as possible, its functionality can be extended by using third-party plugins. The most amazing thing about Leaflet is its powerful plugin ecosystem as anyone with some coding skills can create a Leaflet plugin with some standards that have to follow. The Leaflet.js plugins provides functionalities in various areas of mapping, like tile and image layers, basemap providers, vector files and overlay map extends.

TRITON's webGIS page

TRITON's webGIS structure

The TRITON's WebGIS is a website for interactive visualization of the spatial data collected in the Project and organized in a common spatial infrastructure. The website is based in the Leaflet.js library as described above and combines Geographic Information Science (GIS) principles and tools to harmonize the relatively large and multi-dimensional datasets, including several themes like elevation, land uses, shoreline evolution, hydrography, weather, monitoring and oceanography. The tool was created using cutting edge technologies and combines intelligent web maps together with charts and text to unlock, make accessible and re-usable geospatial data to the broad public in a coordinated manner. More specifically, the tool is trying to find a way for the local and regional authorities to achieve transparent Coastal Zone Management governance and collaboration of all stakeholders and provide data and information to citizens and scientists.

TRITON's webGIS System architecture

As is illustrated in Figure Errore. Nel documento non esiste testo dello stile specificato..6, TRITON's webGIS system architecture is based on three parts that cooperate in harmony. Starting from the last part (Client side), user is interacting with a web mapping application that is created with Leaflet.js library and sends queries of screen visualization to a server. **This server is established at the Laboratory of Ecological Engineering and Technology, Department of Environmental Engineering, Democritus University of Thrace** and consists of two different Servers: a Web Server that hosts the TRITON's website and handles Python Server scripts and a Map server that pushes the user's queries to external Data Servers. These external Data Servers after receiving the queries respond to the TRITON's Map Server with the requested data. These data are transformed in the appropriate georeferenced format. Afterwards, the Web Server pushes the requested data back to the Web mapping application and data is visualized in a user-friendly and understandable map.

Lead Partner

Technical support

Project Partners

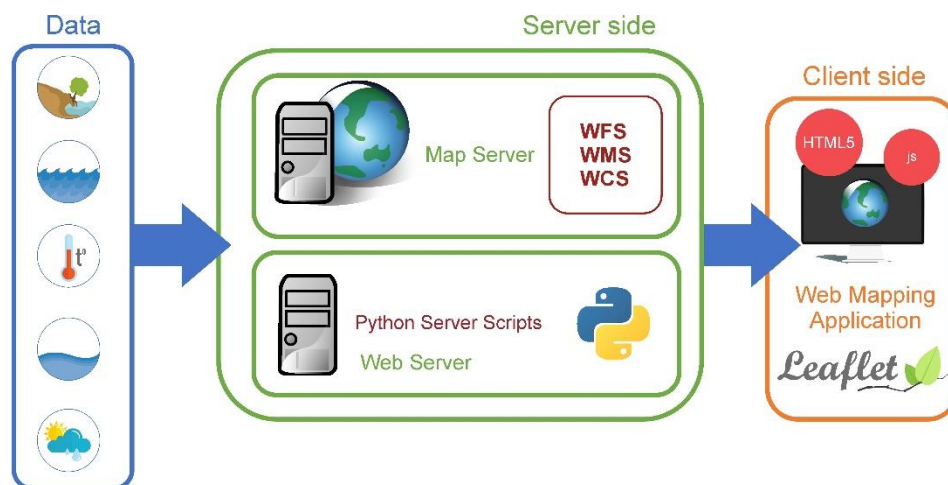


Figure Errore. Nel documento non esiste testo dello stile specificato..6 TRITON's webGIS system architecture.

TRITON's web mapping application (Client-side)

As described before the TRITON's website is based in the Leaflet.js library with the addition of many leaflet plugins for different purposes. In order to increase the functionalities of the website, many plugins were created and included in the TRITON website and all plugins were translated in Greek language to achieve users' interaction.

Table 2 presents the herein included plugins and describes the purpose of their use.

Table 2. TRITON's webGIS Leaflet plugins	
Plugins	Purpose
Leaflet Library (leaflet.js)	Main library of Leaflet
JQuery (jquery.min.js)	JavaScript library for DOM operations CSS animation, and Ajax requests
Leaflet Responsive Popup (leaflet.responsive.popup.js)	This plugin removes the need to move the map to be able to see the content of the popup
Zoom Home Button (leaflet.zoomhome.min.js)	This plugin provides a zoom control with a "Home" button to reset the view.
Bing JavaScript (Bing.js)	This plugin adds Bing Maps Layers
SlideDrawer (jquery.slidedrawer.js)	This plugin adds a sliding bottom bar on map
Google Maps Plugin (Leaflet.GoogleMutant.js)	This plugin adds Google Maps Layers
Leaflet Fullscreen (Leaflet.fullscreen.min.js)	This plugin adds a HTML5 Fullscreen control
Leaflet Measure (leaflet-measure.js)	This plugin adds the ability to measure distances and areas on map

Overview mini map Plugin (Control.MiniMap.min.js)	This plugin adds a MiniMap on map
Leaflet Drawing Plugin (leaflet.pm.min.js)	This plugin adds a drawing toolkit on map
Leaflet Providers (leaflet-providers.js)	This plugin gives the ability to add more basemap providers
ISO-8601 formatters (iso8601.min.js)	This plugin adds the ability of parsing ISO 8601 durations
Leaflet TimeDimension (leaflet.timedimension.min.js)	This plugin adds time control on map
Leaflet coordinates (Leaflet.Coordinates.min.js)	This plugin provides coordinates of a point on map
Leaflet Control Layers Tree (L.Control.Layers.Tree.js)	This plugin creates and controls grouped layers
Leaflet Velocity (leaflet-velocity.js)	This plugin creates animated effect of currents
Leaflet Waves (leaflet-waves.js")	This plugin creates animated effect of waves
Leaflet Wind (leaflet-wind.js)	This plugin creates animated effect of winds
Leaflet Openweathermap (leaflet-openweathermap.js)	This plugin adds Openweathermap weather parameters on map
Translation Control (gr.js, meteo_gr.js)	These plugins are used for the translation of objects in Greek language
BetterWMS (L.TileLayer.BetterWMS.js)	This plugin adds WMS service on map and enables getFeatureInfo information
Helper Toolkit (helper_fcns.min.js)	This plugin does conversion of parameters units
Highcharts Toolkit (highcharts.js, data.js, exporting.js, export-data.js, accessibility.js)	This plugin creates interactive charts
Visual Click (L.VisualClick.js)	This plugin adds the ability of pointing on-click action
Leaflet Condensed Attribution(leaflet-control-condensed-attribution.js)	This plugin concatenate Attribution in a beautiful way
GeometryUtil.js	This plugin does measurement of distances and areas between newly created features

More technically, the HTML5 structure presented before is a base for the inclusion of the TRITON's cartographic contents. The HTML generated by Leaflet.js library makes clear that all geographic data are marked up in distinct and nested <div> elements. As recommended by the HTML Standard, class attributes are used to classify these elements in sensible terms. These terms match up the terms in the Leaflet.js JavaScript API, and by doing this, each DOM Element can be related to the corresponding Leaflet JavaScript class responsible for it (Reiðig, 2016).

This structure provides a "virtual container" which receives visual details, such as JavaScript tasks or styles in CSS. A division <div> element was created with a given ID, and the size of both dimensions

(width and height) in a 2D extent was defined. The defined <div> was further linked with a class of the leaflet library: L.map. This core element of the API is used to load and manipulate basemaps or other (thematic) layers as tile layers with the ability to apply “on the fly” the CRS (Coordinate Reference System) transformation of the layers to the selected one, at all zoom levels.

In order to specify the area of interest, a set of coordinates was given to define the location where the initial map screen is centered (latitude: 38.30°N, longitude: 21.37°E). Moreover, an initial zoom level was defined matching the best perspective for the user of the area of interest (zoom level: 11). In the following Figure Error. Nel documento non esiste testo dello stile specificato..7 the initial screen of TRITON’s webGIS with some of the implemented tools is illustrated.



Figure Error. Nel documento non esiste testo dello stile specificato..7 Initial display of TRITON’s webGIS page.

TRITON’S webGIS page elements

The initial screen of TRITON’s webGIS can be divided into 5 parts:

- Moving clockwise over the screen, at the upper left part, the user may find the “Zoom Control” elements which consist of three buttons from where the user either controls the zoom level of the screen (plus and minus buttons) or returns to the initial screen (Home button). Each time, the user interacts with these buttons all tiles and other elements loaded on the map respond to this action and smoothly change to the requested zoom level. Additionally, zoom levels can change using the mouse scroll wheel on desktop PCs or by pitching on mobile devices (touch interaction). This shows the great compatibility of Leaflet.js with various types of devices.

- Just below the Zoom Control, two additional tools are implemented: the Fullscreen Control and the Draw Toolkit. With the Fullscreen Control button the user may force the page to be displayed on the entire device screen or disable it. This button helps the user to interact with the page, hiding all Operating System's elements and to create clear screenshots (only the map and its elements). The Fullscreen Control plugin supports localization, therefore as mentioned before plugin's captions were translated to the Greek language.
- The Draw Toolkit (Figure Error. **Nel documento non esiste testo dello stile specificato..8**) provides the user with the ability to create many types of features on a map. The user may mark points of interest on the map and receive the respective coordinates of these markers; draw polylines and receive their length; draw rectangles and polygons and receive their area and perimeter; draw circles and receive their radius and area. All these features are editable, can be moved or deleted after creation using the toolkit that is placed on the main creation bar. The tool was fully translated to the Greek language.

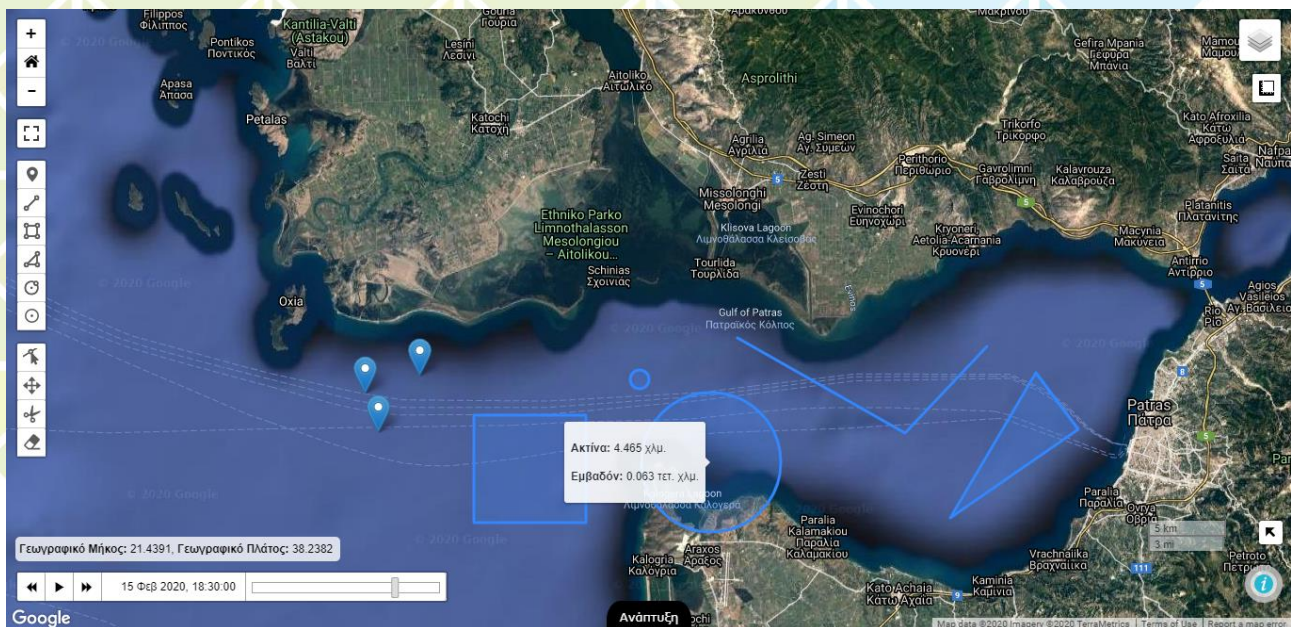


Figure Error. Nel documento non esiste testo dello stile specificato..8 Illustration of added features through the Leaflet Draw Tool.

- The Layer Manager, located at upper right corner of the screen of TRITON's webGIS. To create this Manager over ten plugins were combined with CSS styling files. This manager gives the ability to user to select the tiled basemap that will be displayed on the map and the thematic maps that will be placed on top of it.

The basemap is the foundation of a map which provides geographic context to any other map developed through the web-GIS. In other words, it would be difficult to interpret any map developed

through the web-GIS without the basemap. Most of the time, the basemap is a specific layer, which appears in tiled images at different zoom levels. These tiled maps consist of often thousands or millions of pre-drawn images that are saved on a server and passed out to web browsers as people pan around the map. The basemap may consist of many sublayers (such as roads, lakes, buildings) which often are fused together into a rasterized set of tiled images and treated as a single layer in the web map environment. In addition, some servers send a basemap as "chunks of vector coordinates", sometimes known as "vector tiles". This technique adds more flexibility in visualization, allowing the user to rotate the map while the labels stay right-side-up.

In the TRITON's webGIS, five basemaps were implemented: Google Hybrid, BING Satellite, Open Street Map Mapnik, Stamen Terrain and Stamen Watercolor. Each one of the basemaps was selected for a different reason. Google Hybrid was selected as the default basemap in order to be easier for the user to understand what is displayed on the map due to its intimacy with the most popular web map platform (Google Earth). BING Satellite was selected to provide to the user a clear cloudless satellite view of the area. Open Street Map Mapnik was selected to provide to the user Roads and Places' Names of the area. Stamen Terrain was selected to provide to the user an illusion of the land elevation as the hill shade of the area is displayed. Stamen Watercolor was selected to provides to teachers a tool for the kids easy to understand. A typical presentation of these basemaps is shown in Figure Errore. **Nel documento non esiste testo dello stile specificato..9.**

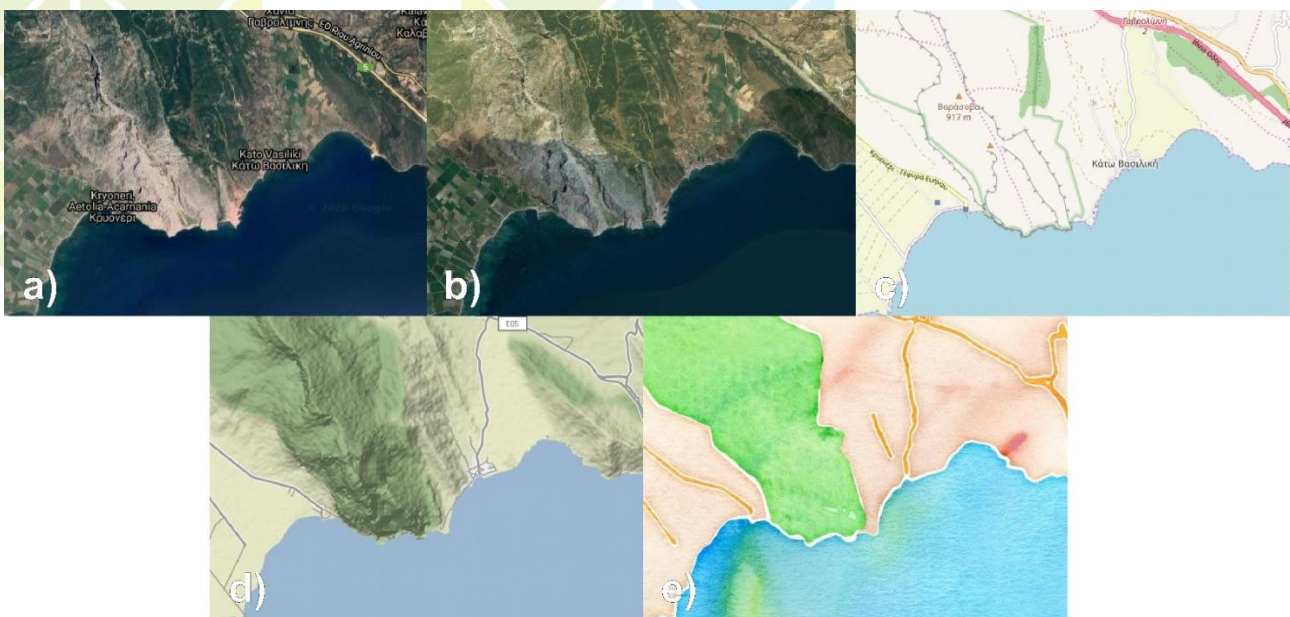


Figure Errore. Nel documento non esiste testo dello stile specificato..9 a) Google Hybrid, b) BING Satellite, c) Open Street Map Mapnik, d) Stamen Terrain and e) Stamen Watercolor basemaps for the same area as displayed on TRITON's webGIS page.

Thematic layers (or Operational layers) are placed on top of the Basemaps (or Background layer) and these are the reason people visit a webmap. If thematic layers are placed on a basemap, nobody will be interested in, but if they are placed on a focused web map, they are the main layer of interest. When everything is placed on the map it can be described as a “map sandwich” approach, where the basemap is the bottom layer and all thematic layers are placed one over the other (Figure Errore. Nel documento non esiste testo dello stile specificato..10). It should be mentioned that is not necessary the thematic layers to be placed on the top of basemaps, as a basemap can be placed on top of all thematic layers and used as a Reference layer providing users with relevant information (like roads or addresses).

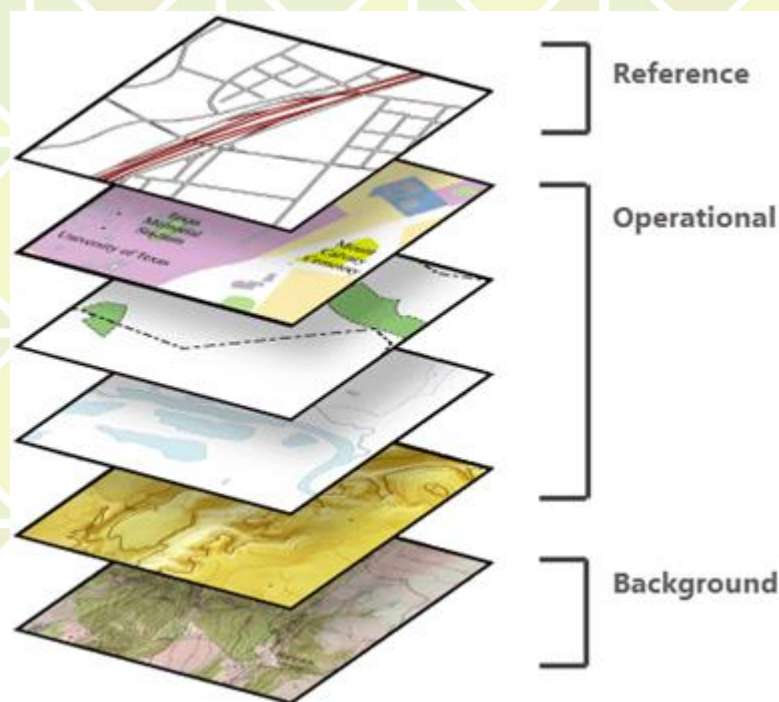


Figure Errore. Nel documento non esiste testo dello stile specificato..10 “Map sandwich” of basemaps and thematic layers (ESRI, 2019).

Usually, thematic layers are created locally by a Geographical Information System (GIS) software and uploaded in a Map server which can handle different types of geospatial data or can come directly from other servers via a server-to-server interaction. There are various web services that comply with the Open Geospatial Consortium (OGC) Standards, such as WMS, WCS, WMTS or WFS, etc., designed to draw maps “on the fly” by displaying thematic layers as tiles. In addition, using queries to these services the user might receive data for a specific location providing the ability to create interactive elements, such as popups.

Thematic layers can be several vectorized layers with sublayers or a single web service with multiple sublayers or multiple web services that each contains a single layer, i.e., the Layer Manager allows the web mapping developer to include all these types of thematic layers and display them using a CCS style in a user-friendly and easy-to-understand manner. Therefore, this plugin, allows the users to toggle the layer they prefer to visualize through an "on" and "off" button.

Figure Error. **Nel documento non esiste testo dello stile specificato..11** illustrates an indicative screen of the TRITON's webGIS page displaying the Layer Manager. The Layer Manager is structured with the Basemap appearing at its top and all Thematic Layers below following a "tree structure" where many layers have subsequent sublayers, allowing the user to select either each specific sublayer or all of them through the "Select All" tickbox. As described in previous sections, the existing data layers contain data like coastal topography, land uses, shoreline evolution, hydrography, historic and prognostic weather, waves and currents, physico-chemical oceanographic parameters and data from on-line monitoring stations, visualized either through the created Map Server or by direct queries on several Servers and WMS Servers.

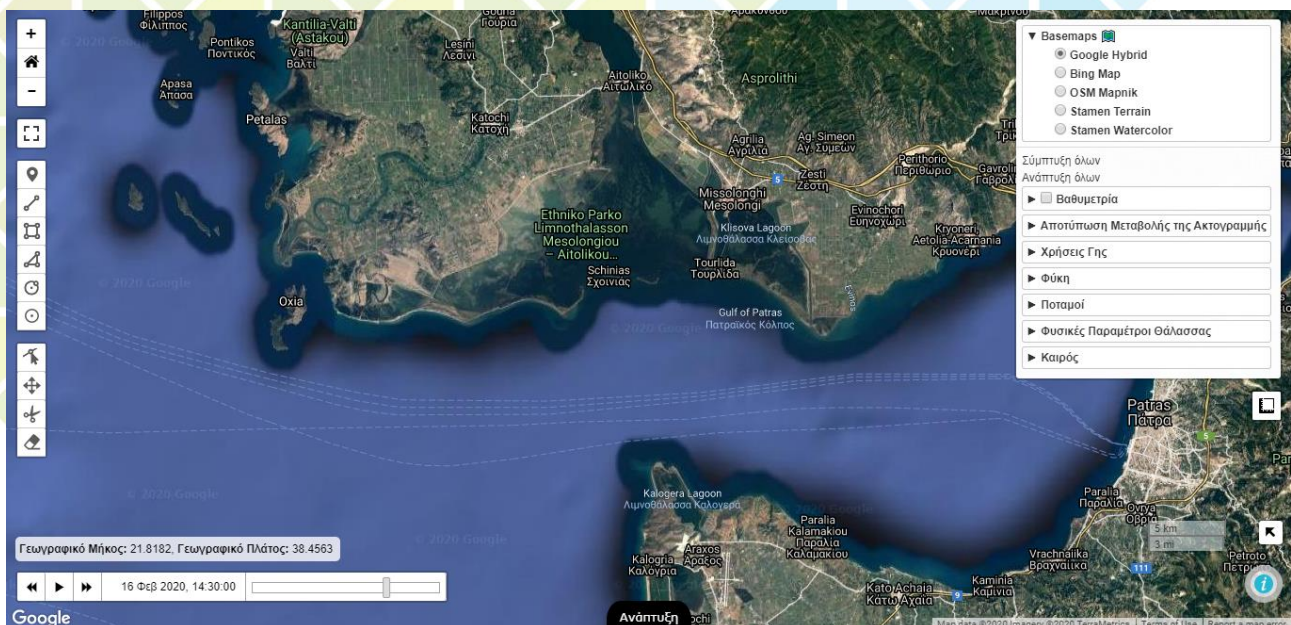


Figure Error. Nel documento non esiste testo dello stile specificato..11 TRITON's webGIS Layer Manager.

- The Measure Tool is shown at the upper right part of the web-page screen (Figure Error. **Nel documento non esiste testo dello stile specificato..12**). This plugin allows the user to draw lines or shapes and get their distances and areas in many different units. In addition, the user may modify the drawn area by moving the node points and receive the re-measured distance, area and perimeter of the drawn shape. Drawn shapes will be removed right after

the user completes the measurement. As in every tool implemented, the Measure Tool was also translated in the Greek language.

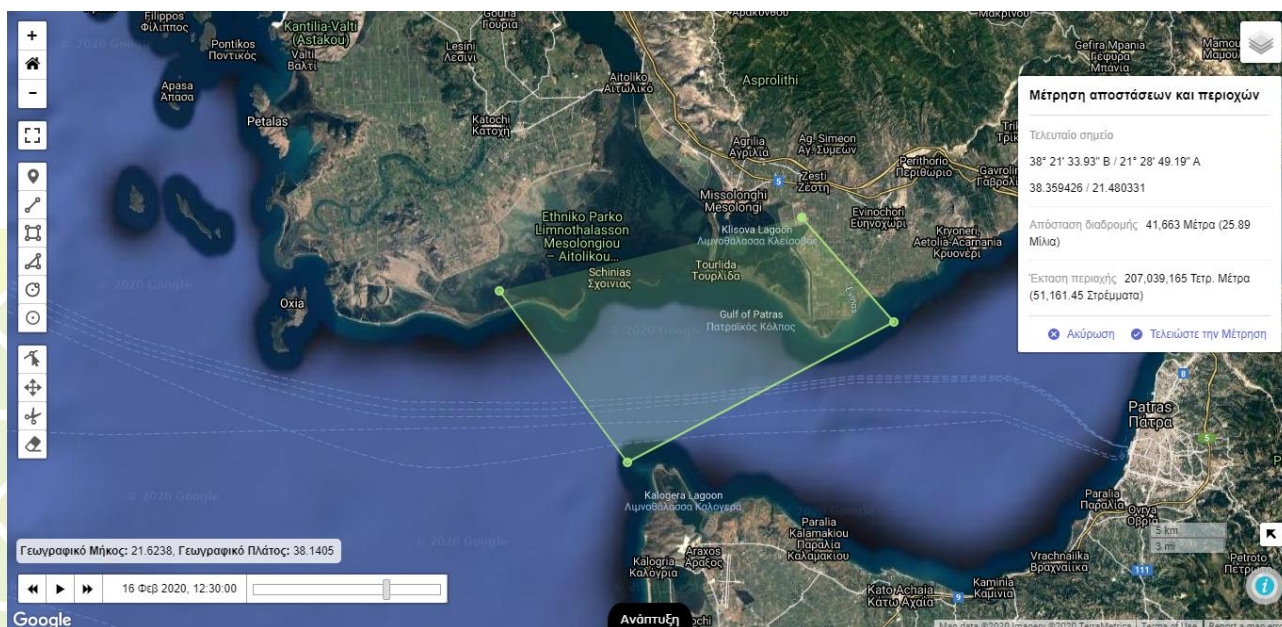


Figure Error. Nel documento non esiste testo dello stile specificato..12 TRITON's webGIS Measure Tool.

- At the lower right part of the screen, the user might find three different implemented elements. The first is the MiniMap tool, designed to show a bigger area than the area of interest (Figure Error. **Nel documento non esiste testo dello stile specificato..13a**). This is a useful tool because it provides the user with a better understanding of the area initially displayed on the map. The user may interactively open and close the MiniMap tool or move the map of the MiniMap to other parts of the screen. The next element that is implemented is the ScaleBar, through which the user may scale the map. When the user zooms in and out on the map this element instantly responds to the newly defined scale. The last element is the Information Bar through which the user may receive Information about the loaded thematic Layers (Figure Error. **Nel documento non esiste testo dello stile specificato..13b**). Each thematic layer has its own attributes and when loaded an interactive link will appear in the Information Bar providing metadata information about the layer.

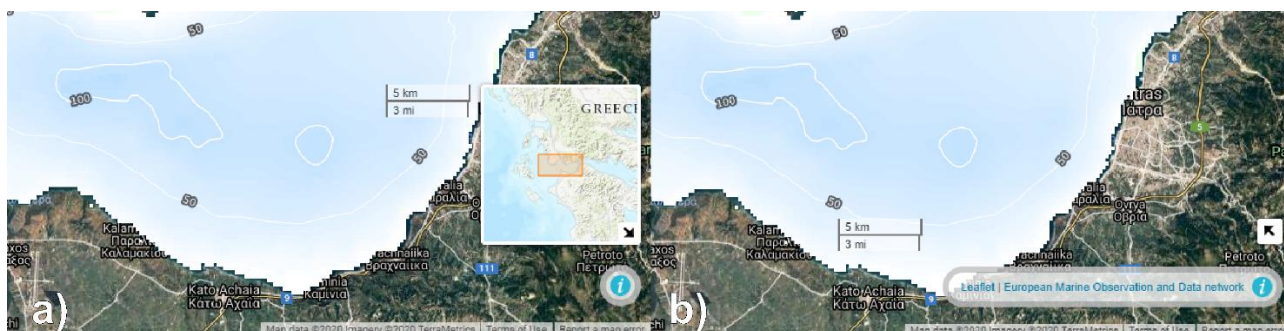


Figure Errore. Nel documento non esiste testo dello stile specificato..13 TRITON's webGIS a) MiniMap and ScaleBar and b) Information Bar.

- At the lower-central part of TRITON's webGIS page the user may find the Development Information button. When the user clicks on this button, a sliding bar appears immediately displaying the logos of the development partners. This tool is not really useful for the user but it is placed there to provide information on the contact organizations for additional information.



Figure Errore. Nel documento non esiste testo dello stile specificato..14 TRITON's webGIS Development Information Bar a) closed and b) opened.

- Finally, at the lower left part of the TRITON's webGIS page, two plugins are implemented at the initial screen of the web map (



- Figure **Errore. Nel documento non esiste testo dello stile specificato..15**). More specifically, the user may find the Coordinates Tool, providing the map coordinates corresponding to the cursor's location. In addition, when double clicking on the Coordinates bar the user may copy and paste these values to/from the clipboard. The last plugin that user can find is the Time Control Tool. With this tool the user controls the time of a thematic layer visualized on the web map. Time Control tool is divided into two sections: at the first the user may move backward or forward in time or click on the Play button and produce an animated view of all the available time-frames of the selected thematic layer. On the next section, the user may slide the mouse on a slider and select a specific time for the thematic layer visualization. Initially, the slider is fixed on the time the user may access in the webpage. Therefore, initially the thematic layer displays the data corresponding to the time the user has accessed the webpage. As the user interacts with the Time Control tool, immediately a query is sent either to the Map Server or to any external servers and they respond with the requested data. The TRITON's webGIS Time Control tool span is fixed: two days before and two days after the present day. Therefore, the webGIS displays data only within this 5-day time window. The only exception is the Wind thematic layer. Due to the fact that different datasets provide data corresponding to different time zones, the Time Control tool has automatically adjusted time to the local time zone and displays the relevant datasets.



Figure Errore. Nel documento non esiste testo dello stile specificato..15 TRITON's webGIS Time Control and Coordinates.

The above described, are just the tools that appear on the initial screen of TRITON's webGIS page. During the following sections, more tools embedded on the system will be described.

TRITON's webGIS Data integration

In this study, data from several external databases were retrieved, processed and visualized through the TRITON web-GIS system. This procedure and data origin were explicitly described in the TRITON's Deliverable 3.4.1. More precisely, starting from above in the Layer Manager:

- Bathymetric data were retrieved from the European Marine Observation and Data Network (EMODnet) Bathymetry Portal;
- Topographic data were retrieved from the Shuttle Radar Topography Mission Database;
- Shoreline evolution data were retrieved from historic satellite images analyzed created according to the methodology developed by the Laboratory of Ecological Engineering and Technology, Democritus University of Thrace;
- Land Use data were retrieved from the Copernicus Land Monitoring Service (CLMS) and the Hellenic National Cadastral;
- Seagrass data were retrieved from EMODnet Seabed Habitats portal;
- Hydrologic data for each of the rivers in the area were retrieved from the Swedish Meteorological and Hydrological Institute (SMHI);
- Oceanographic physicochemical data, such as salinity, water temperature, sea level, chlorophyll-a concentration, currents and waves, were retrieved from Copernicus Marine Environmental Monitoring Service (CMEMS); and

- Meteorological data, such as wind speed and direction, precipitation, cloud cover, etc., were retrieved from Global Forecasting System (GFS) of NOAA.

Each of these datasets will be presented and more web map tools interacting with these data will be described in the following sections.

Bathymetric Thematic Layers

The bathymetric data of the study area was derived from the EMODnet Bathymetry portal (<http://www.emodnet-hydrography.eu>). EMODnet bathymetry is composed of a multitude of datasets from a multitude of data providers and the produced Digital Terrain Model (DTM) has a grid size of $1/16 \times 1/16$ arc minutes. The GIS layers of EMODNET Bathymetry portal are shared as OGC WMS and WCS services. The OGC services can be found at the following URLs:

WMS: <https://ows.emodnet-bathymetry.eu/wms>

WFS: <https://ows.emodnet-bathymetry.eu/wfs>

WMTS: <https://tiles.emodnet-bathymetry.eu>

WCS: <https://ows.emodnet-bathymetry.eu/wcs>

In TRITON's webGIS page an L.TileLayer was created to interact with the EMODnet WMS server. Each time user selects the Bathymetry layer on the Layer Manager, a direct query is served to EMODnet's WMS server requesting bathymetry tile for the selected zoom level. The result is visualized on the web map with a pre-defined color scale that comes from the same server which immediately appears on the upper left part of the page in a legend division (Figure Error. **Nel documento non esiste testo dello stile specificato.**..16). In addition, contour lines are created on the map to identify the isobaths of the area. When the user selects a location by clicking, another query is served to EMODnet's WMS server and a popup window appears identifying the coordinates and depth of the clicked location.

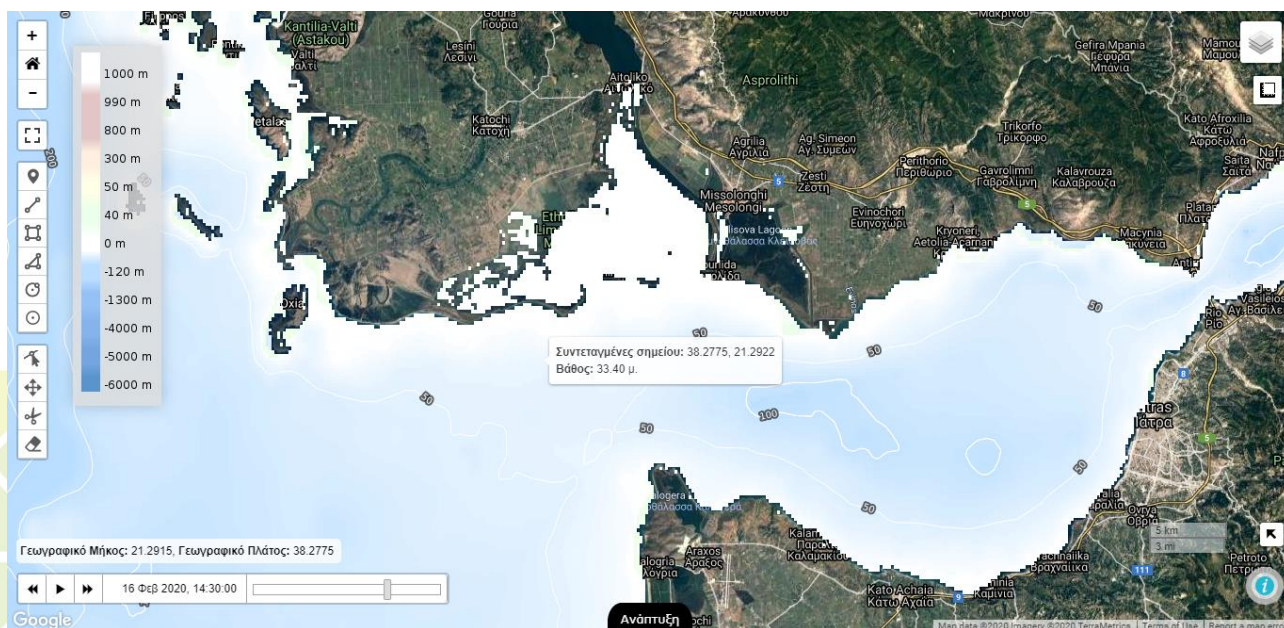


Figure Errore. Nel documento non esiste testo dello stile specificato..16 Bathymetric data as displayed on TRITON's webGIS page.

Topographic Thematic Layers

The topographic data of the study area was derived from the Digital Elevation Model (DEM) of "Shuttle Radar Topography Mission" (SRTM30). SRTM30 is a near-global digital elevation model (DEM) comprising a combination of data from SRTM, flown in February, 2000 and the U.S. Geological Survey's GTOPO30 data set. It can be considered to be either an SRTM data set enhanced with GTOPO30, or as an upgrade to GTOPO30. Spatial resolution of SRTM data is 1 arc-second in latitude and longitude (approximately 30 meters at the equator). In TRITON's webGIS page an L.TileLayer was created to interact with the SRTM WMS server. The result is visualized on the web map with a pre-defined color scale that comes from the same server.



Figure Errore. Nel documento non esiste testo dello stile specificato..17 Topographic data as displayed on TRITON's webGIS page.

Shoreline Evolution Thematic Layers

The methodology applied to assess and define the shoreline evolution and identify coastal erosion and accretion ‘hotspots’ along the coastal zone of Messolonghi Municipality was based on the shoreline evolution analysis by processing historical satellite images for the period 2009 to 2019, using remote sensing techniques and GIS software, as developed by the Laboratory of Ecological Engineering and Technology, Democritus University of Thrace. When the user selects the Shoreline Evolution in the Layer Manager (Figure Errore. **Nel documento non esiste testo dello stile specificato..18**) another menu, with three additional sublayers appears. These sublayers correspond to: a) the Historical Snapshots of the shoreline, b) the Shoreline Change for the period 2009-2019, and c) the Shoreline Change Rate for the period 2009-2019.

At the first layer the user can visualize historical snapshots of the shorelines for the years 2009, 2012, 2015, 2019. This data come from shapefiles created by the Laboratory of Ecological Engineering and Technology, being transformed into web GeoJSON (*.pjson) files in order to maintain all of their characteristics and subsequently being stored in the Map Server. When the user displays these layers, a legend appears to indicate the year corresponding to each extracted shoreline. Additionally, the user could identify the respective date by placing the mouse over the shoreline, as illustrated in Figure Errore. **Nel documento non esiste testo dello stile specificato..18**. Finally, the user may interact with the shorelines and edit them by using the Draw Tool but this action is not recommended as the system is unable to save any changes.

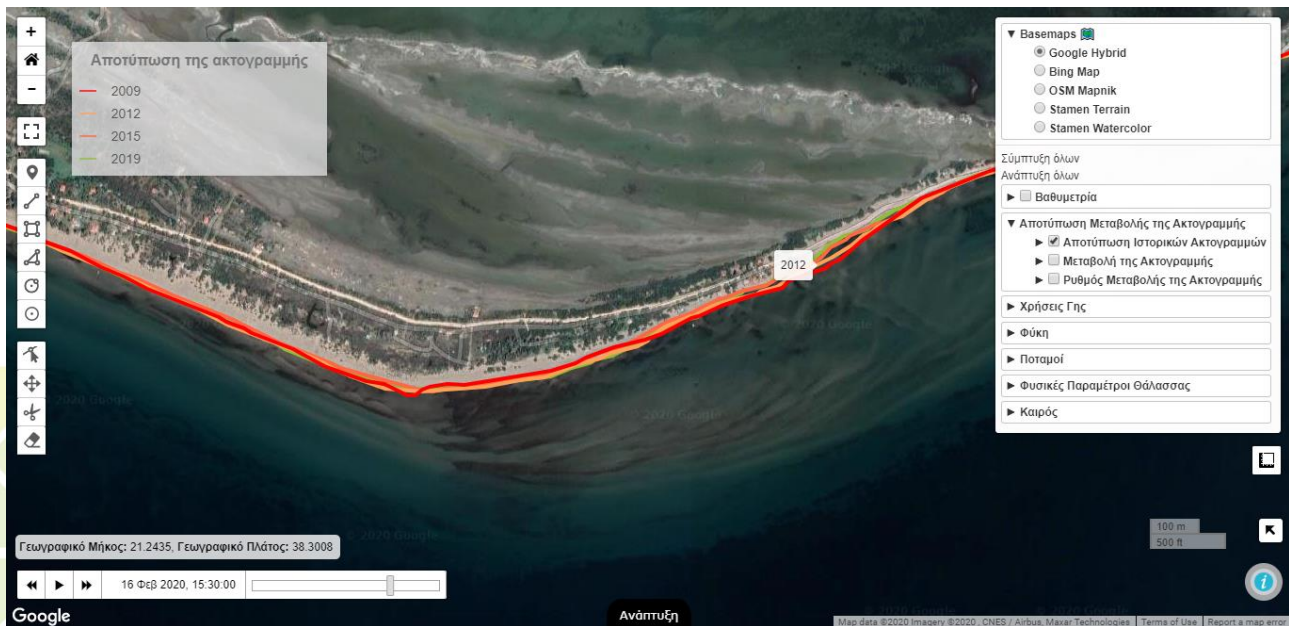


Figure Errore. Nel documento non esiste testo dello stile specificato..**18** Historical shorelines data as displayed on TRITON's webGIS page.

At the second and third layers the user is able to display the Total Shoreline Change from 2009 to 2019 (in meters) and the Shoreline Change Rate, as the change taken place per year, during the 2009-2019 period. In order to retrieve these data, the same procedure as at the previous layer was applied. More specifically, when the user selects these layers, transects perpendicular to the shoreline appear. These transects are colored using 7 color classes: Large Accretion (>20m), Medium Accretion (10 - 20m), Small Accretion (1 - 10m), Stable Shoreline (-1 - 1m), Small Erosion (-1 - -10m), Medium Erosion (-10 - -20m), Large Erosion (>-20m) (minus symbol corresponds to erosion). Each of these two sublayers has three additional sublayers corresponding to the shoreline status (Accretion, Stable Shoreline, Erosion). This way the user can filter the shorelines based on their overall characterization. In addition, a legend appears explaining these classes, but the user can always place the mouse over the shoreline and receive the status of the selected shoreline segment, the total shoreline change or change rate, as illustrated in Figure Errore. **Nel documento non esiste testo dello stile specificato..19.**

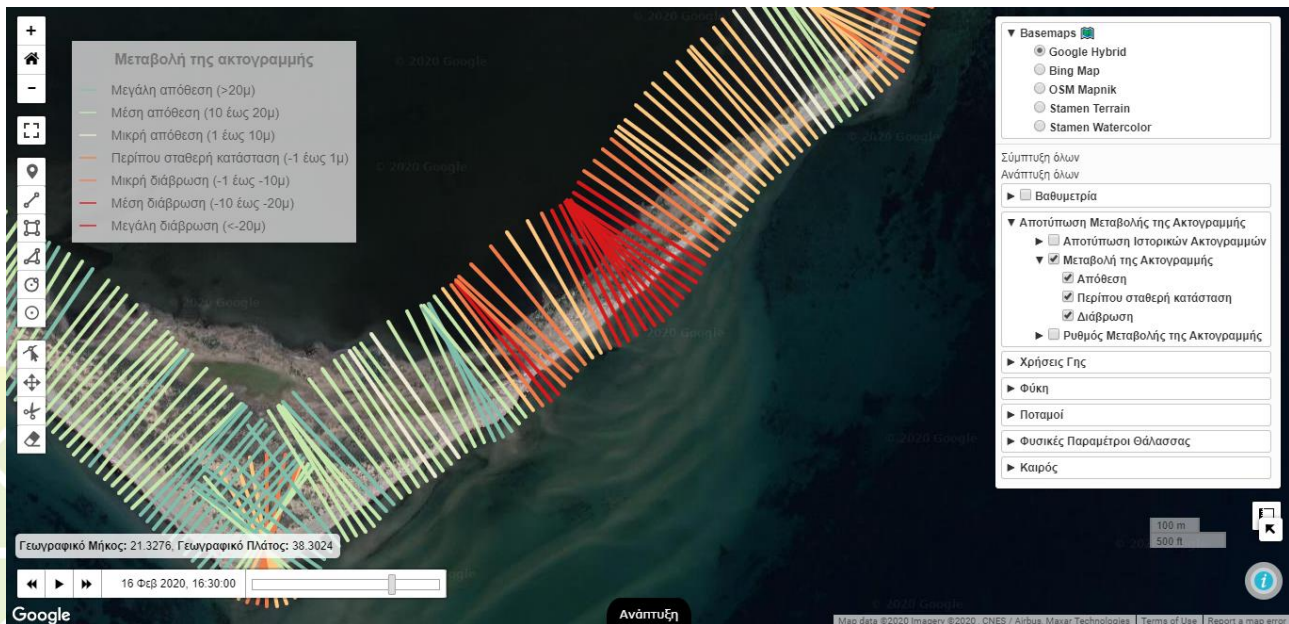


Figure Error. Nel documento non esiste testo dello stile specificato..19 Shoreline movement data as displayed on TRITON's webGIS page.

Land Cover Thematic Layers

The Land Cover data of the study area was retrieved from the Copernicus Land Monitoring Service and Hellenic National Cadastral (HNC). When the user selects the specific Land Use layer in the Layer Manager two more sublayers drop down containing data originated from the CORINE Land Cover (CLC) database and the HNC.

Specifically, the first sublayer includes three more sublayers for CORINE data retrieved from the CLMS for the years 2006, 2012 and 2018. The retrieved data were retrieved in geo-package format and were clipped (sub-setting) to a smaller area to reduce the web page load. Retrieved data were transformed into the web GeoJSON format files. These files were uploaded to the Map Server and from there they are servicing the TRITON's webGIS page as interactive features. When the user places the mouse over a specific colored surface of the webGIS screen (colors are aligned with CORINE's colors for each land use type) the boundaries of the feature are highlighted and a popup window appears stating in Greek the land use type corresponding to this feature (Figure Error. **Nel documento non esiste testo dello stile specificato..20**). Applying this method for visualization purposes, the creation of long list of land use types in the legend window is avoided.

The second sublayer includes orthophotos maps originated from HNC's WMS Server. An L.TileLayer was created on TRITON's web mapping service querying this server during users' interaction and responding the required tiles for the specific zoom level (Figure Error. **Nel documento non esiste testo dello stile specificato..21**).



Figure Errore. Nel documento non esiste testo dello stile specificato..20. Corine land cover as displayed on TRITON's webGIS page.

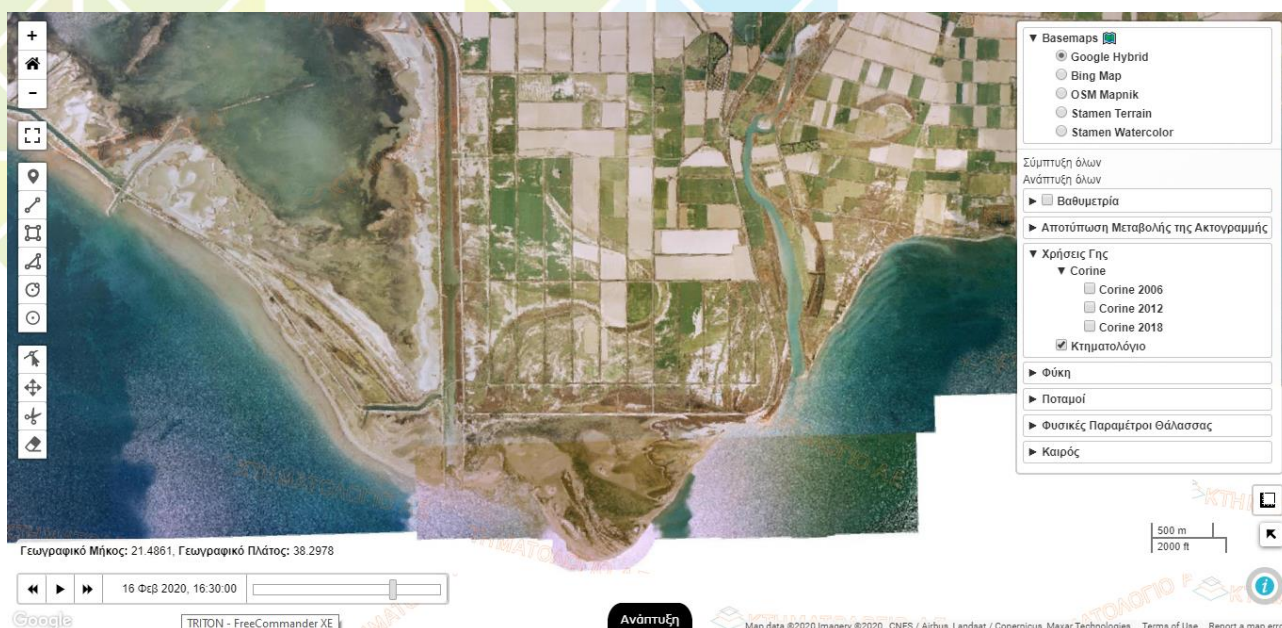


Figure Errore. Nel documento non esiste testo dello stile specificato..21. HNC orthophotos as displayed on TRITON's webGIS page.

Seagrass Thematic Layers

Seagrass data were retrieved from EMODnet Seabed Habitats portal. Seabed Habitats was one of seven themes of the European Marine Observation and Data Network (EMODnet) initiative, funded by the European Maritime and Fisheries Fund. When the user selects the Seagrass layer of the Layers Manager, two additional sub-layers appear representing the areal coverage of Seagrass Meadows and the probability of occurrence of *Posidonia Oceanica* species.

The first sublayer displays the currently-known coverage of Seagrass Meadows in European waters, collated by EMODnet Seabed Habitats. Seagrasses provide essential habitat and nursery areas for many marine fauna. There are approximately 72 seagrass species that belong to four major groups: Zosteraceae, Hydrocharitaceae, Posidoniaceae and Cymodoceaceae. Zostera beds and Cymodocea meadows are named on the OSPAR Threatened or Declining Habitats list. In TRITON's webGIS page an L.TileLayer was created to interact with the EMODnet WMS server and retrieve the data for the selected zoom level (Figure Error. **Nel documento non esiste testo dello stile specificato..22**).

The second sublayer presents the modelled spatial distributions of *Posidonia oceanica* at the seafloor. This dataset is an output of the "Mediterranean Sensitive Habitats" project (MEDISEH). More specifically, a Random Forest (i.e., a Machine Learning technique) was "trained" on data from regions where information was available and then used to predict the probability of occurrence of *P. oceanica*, with a spatial resolution of 0.004166 decimal degrees. When the user selects this layer from the Layer Manager tool, a tiled map of the modelled spatial distribution of *Posidonia oceanica* with values fluctuating in the range between [0,1] representing the probability of occurrence in this species (Figure Error. **Nel documento non esiste testo dello stile specificato..23**).

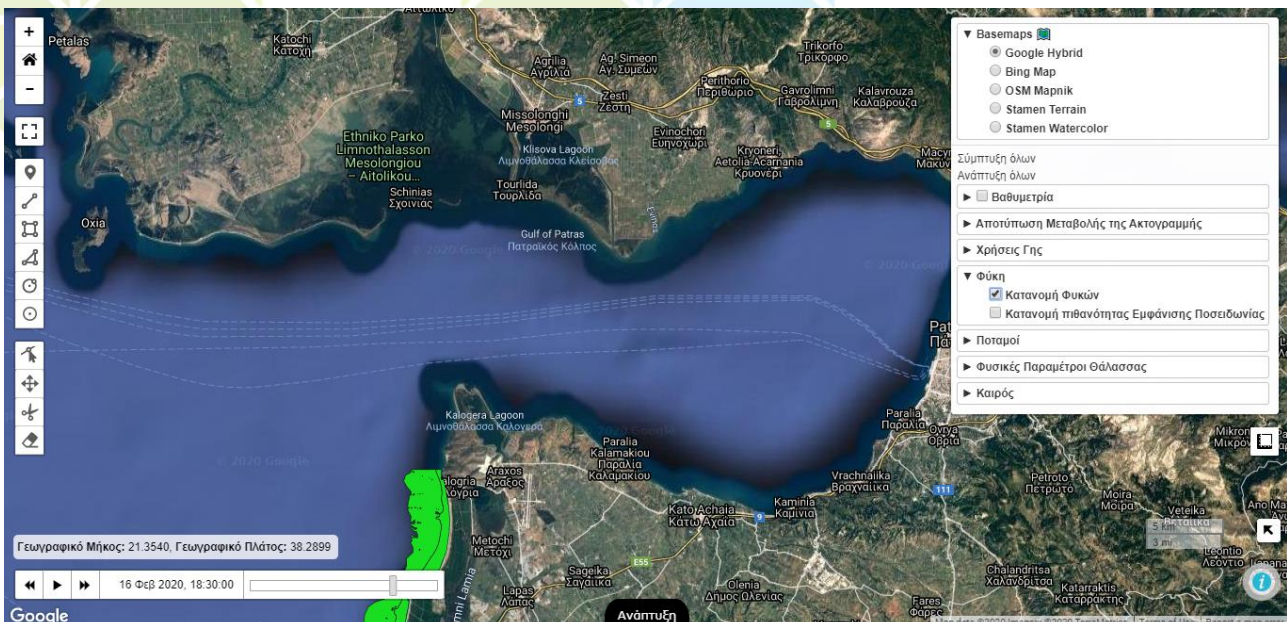


Figure Error. Nel documento non esiste testo dello stile specificato..22 Collated seagrass polygon data as displayed on TRITON's webGIS page.

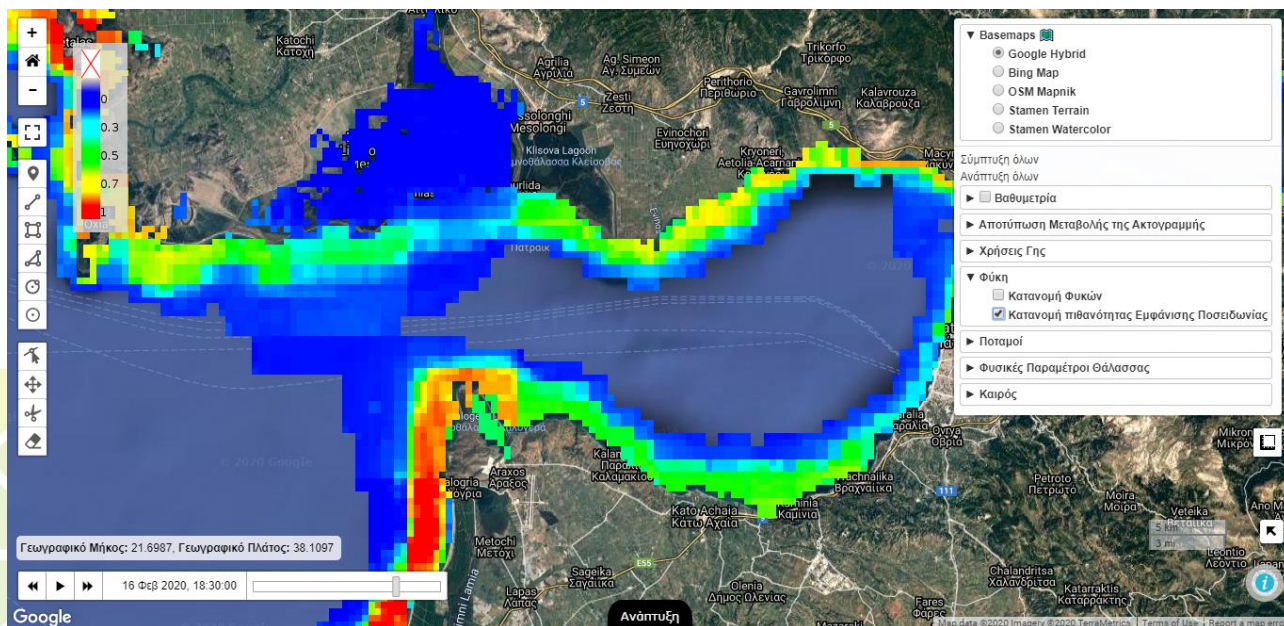


Figure Errore. Nel documento non esiste testo dello stile specificato..23 Spatial distribution of *Posidonia oceanica* as displayed on TRITON's webGIS page.

Hydrography Thematic Layers

Hydrologic data for each of the rivers in the area were retrieved from the Swedish Meteorological and Hydrological Institute (SMHI). When the user selects any of the rivers in the study area, a layer on the Layer Manager with two additional sublayers appears illustrating the historical river discharge data and the forecasted river discharge data (in m³/s).

To avoid the heavy loading of the web page, only the major rivers in the study area were selected. These are the River Acheloos and the River Evinos, flowing within the boundaries of the Municipality Ieras Poleos Messolonghiou, the Peiros River in the Municipality of West Achaia and the Glafkos River in the Municipality of Patras. Discharge data for these two rivers were retrieved from the SMHI (e-HYPE) database, developed by applying the Hydrological Predictions for the Environment Model (Lindström et al., 2010). In TRITON's webGIS page, for both sublayers, an AJAX request was created retrieving discharge data for each of the rivers, based on the sub-basin they belong to. When the user selects these sublayers, a set of polyline features related to the above four rivers appears on the map. When the user selects with the mouse a specific river, the SMHI server responds based on the selected sublayer presenting the historical discharge data covering the period from 1981 to 2010 or the forecasted river discharge data for the next 10 days (Figure Errore. Nel documento non esiste testo dello stile specificato..24). The visualization of this data is based on an interactive chart in which the user may zoom in/zoom out on the timeframe selected. Additionally, the chart offers the user the ability to download the raw data in csv or xls format, download the chart as raster image (png, jpeg or pdf format) or as a vectorized image (svg format) or print it (Figure Errore. Nel documento non esiste testo dello stile specificato..25).

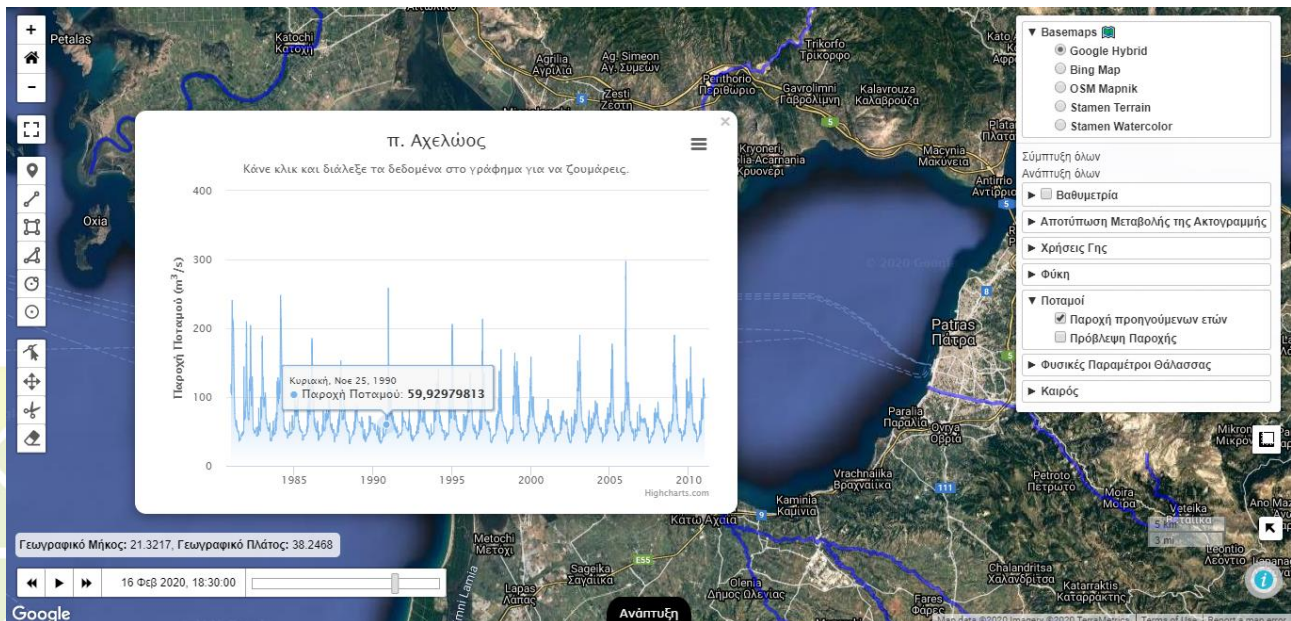


Figure Errore. Nel documento non esiste testo dello stile specificato..**24** Generated chart illustrating the historical discharged data for the Acheloos River, as displayed by the TRITON's webGIS page.

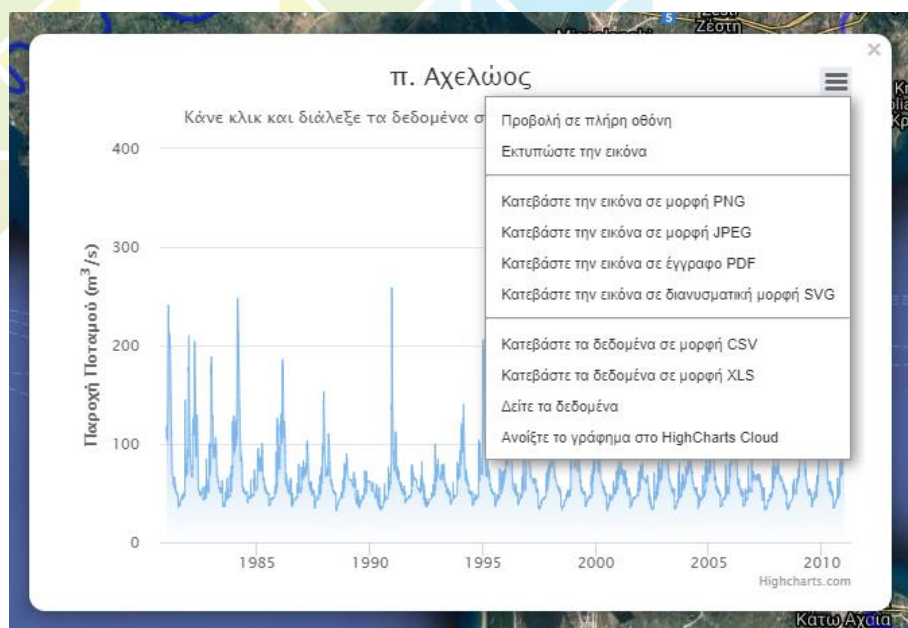


Figure Errore. Nel documento non esiste testo dello stile specificato..**25:** Export tool of the generated chart presenting the historical discharged data of the Acheloos River.

Oceanographic Thematic Layers

Physical and marine biogeochemical components are useful for coastal water quality monitoring and pollution control. Sea level rise is a key indicator of climate change and helps to assess coastal erosion. Currents and waves play a crucial role for the determination of the impact of the water circulation and the incident wave energy contributing to coastal erosion in the study area. Sea surface temperature elevation has direct consequences on marine ecosystems and appearance of tropical cyclones. Chlorophyll-a is one of the most significant datasets for monitoring the impact of the rivers in the study area. Therefore, the retrieval of these datasets appears to be necessary to understand the underlying processes in a cost-effective manner. The retrieval of most of these datasets is through the Copernicus Marine Environment Monitoring Service (CMEMS), part of the Copernicus Program, and is based on L.TileLayers that were created for each of the retrieved parameters to interact with the CMEMS WMS server. When the user selects the Sea Parameters layer from the Layer Manager, seven new sublayers appear. The first five sublayers correspond to five different parameters (currents, salinity, water temperature, sea level and chlorophyll-a concentration) and the next two sublayers are an animated visualization of currents and waves. Each time the user selects one of the first five sublayers, a tiled raster of the selected parameter at the time shown by the Time Control tool is displayed (Figure Error. **Nel documento non esiste testo dello stile specificato.**..26). This gives the ability to the user to select different time periods and receive the corresponding tiled raster. In addition, a legend appears having a pre-defined color scale and range of values. The following figures show each of the parameters as being displayed on the map.



Figure Error. Nel documento non esiste testo dello stile specificato..**26** Surface currents retrieved from CMEMS and displayed by the TRITON's webGIS page.

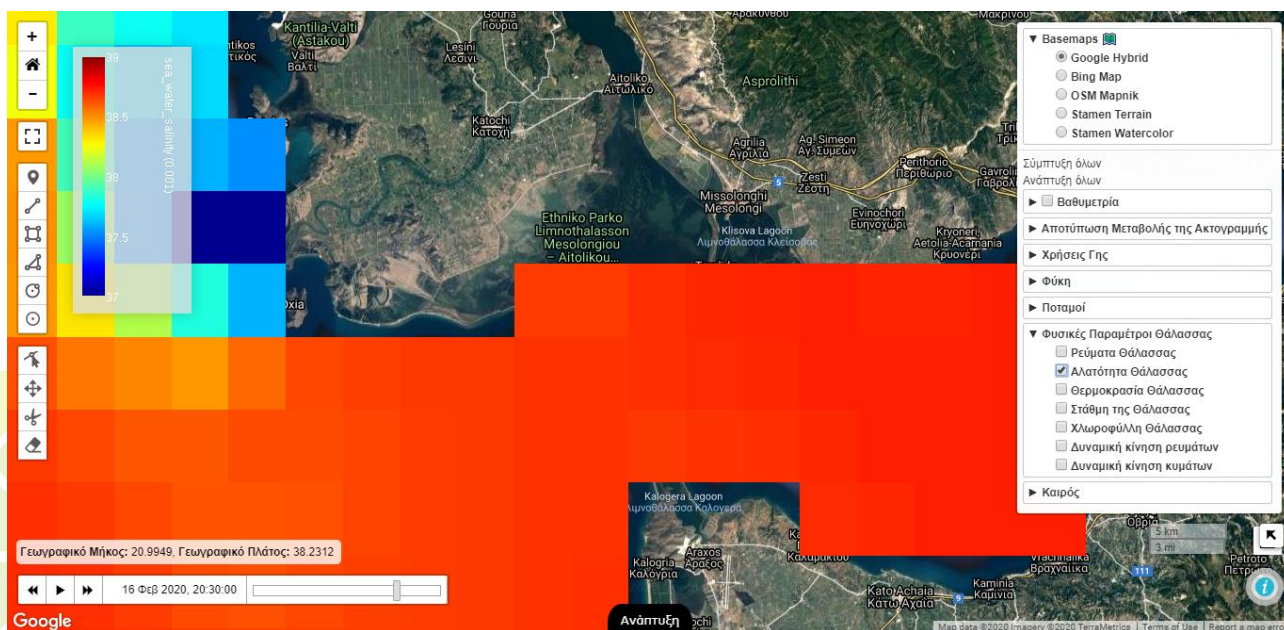


Figure Errore. Nel documento non esiste testo dello stile specificato..27 Surface salinity data retrieved from CMEMS and displayed by the TRITON's webGIS page.

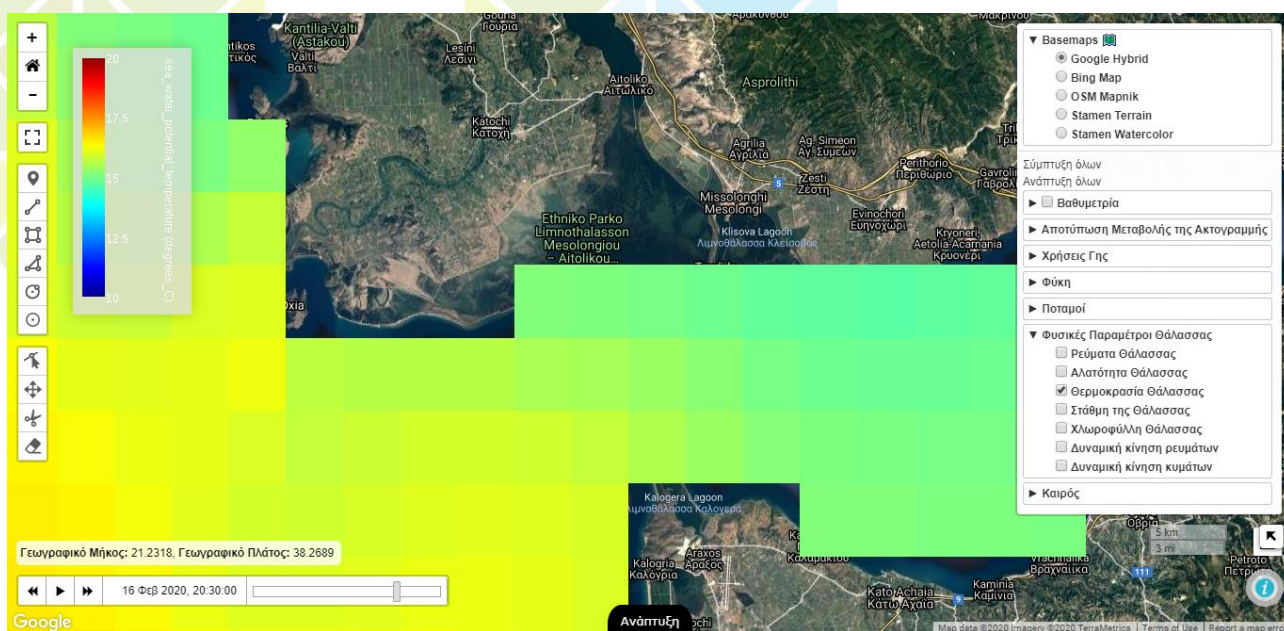


Figure Errore. Nel documento non esiste testo dello stile specificato..28 Surface water temperature distribution retrieved from CMEMS and displayed by the TRITON's webGIS page.

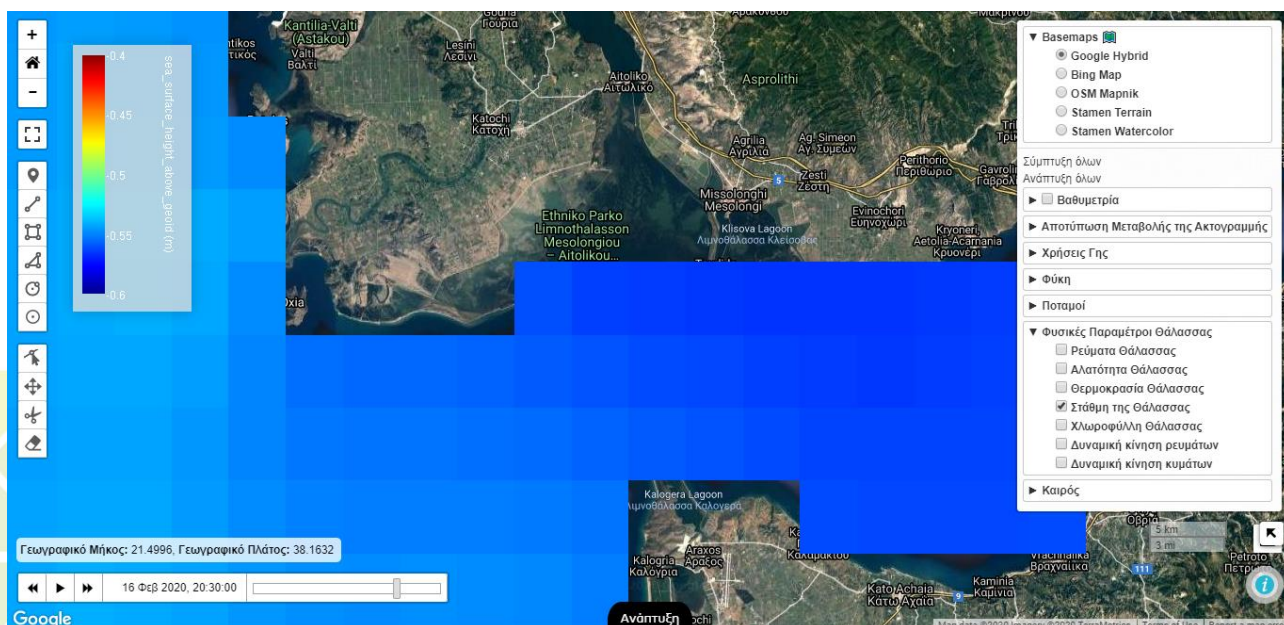


Figure Errore. Nel documento non esiste testo dello stile specificato..29 Sea level variability distribution retrieved from CMEMS and displayed by the TRITON's webGIS page.

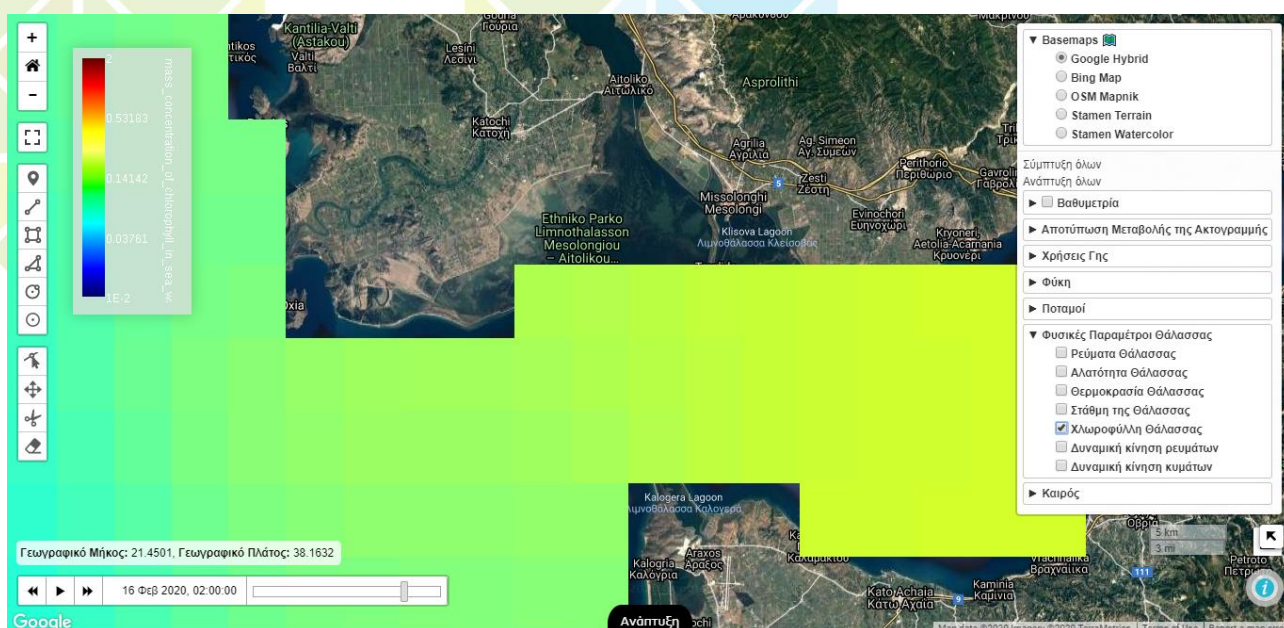


Figure Errore. Nel documento non esiste testo dello stile specificato..30 Surface concentration of Chlorophyll-a retrieved from CMEMS and displayed by the TRITON's webGIS page.

One of the most interesting features of TRITON's webGIS is the capacity to animate currents and waves over time. More specifically, these data as well as all other parameters are retrieved from the CMEMS servers. A python script was created for each of the parameters utilizing the Motu client that CMEMS offers. Data are downloaded as netCDF format files and are subsequently transformed into the web GeoJSON files having the necessary time indexing. This gives the capacity to animate the different frames displaying the spatial distribution of each parameter covering specific time intervals through the Time Control tool. As seen on the following figure, another bar appears above the Coordinates bar. When the user places his mouse over the displayed animated data, the distribution of currents speed and direction and the distribution of significant wave height and direction are displayed. The following Figures 2.26 and 2.27 show these two animations as displayed on TRITON webGIS page.

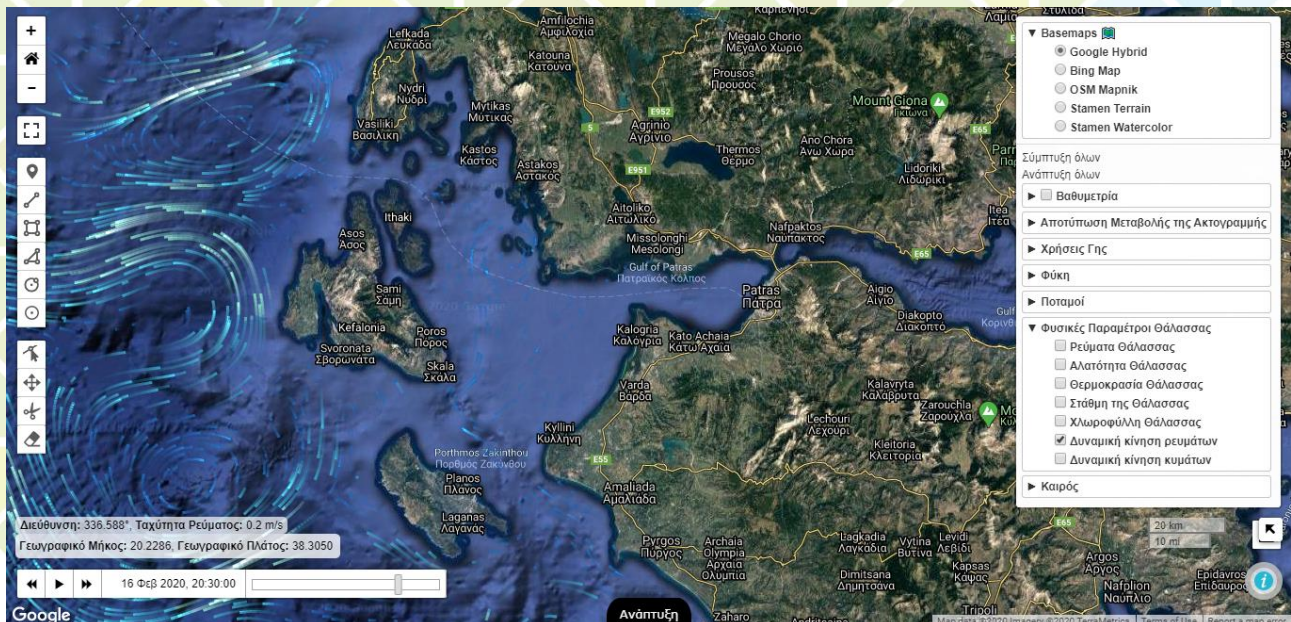


Figure Errore. Nel documento non esiste testo dello stile specificato..31 Animation of sea surface currents as displayed by the TRITON's webGIS page.

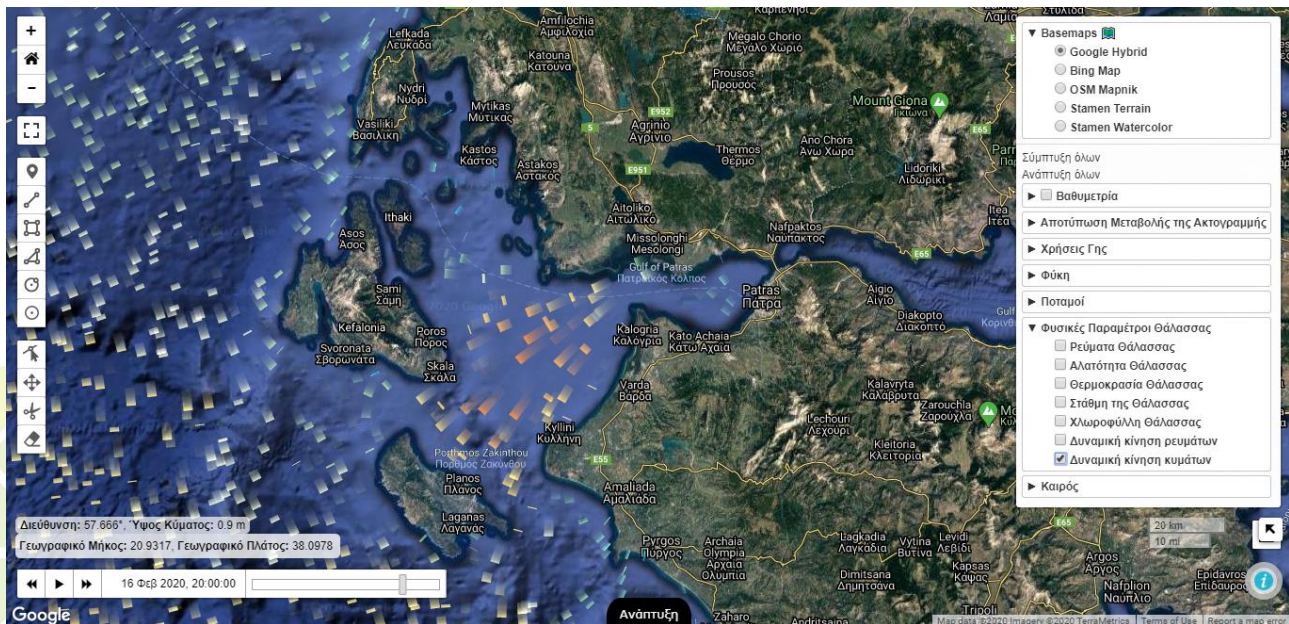


Figure Error. Nel documento non esiste testo dello stile specificato..32 Animation of waves as displayed by the TRITON's webGIS page.

Weather Thematic Layers

The weather data of the study area were retrieved from the Global Forecasting System (GFS) of NCEP and from the Open Weather Map Service. When the user selects the Weather layer from the Layers Manager three additional sublayers representing the distribution of wind, precipitation and cloud coverage is displayed. More specifically, when the wind sublayer is selected an AJAX request makes queries to an external server and retrieves the wind components data. Wind data are animated in the same manner as the currents. The user may receive the wind direction and speed at any location on the map screen (Figure Error. **Nel documento non esiste testo dello stile specificato..33**). When precipitation or cloud cover sublayers are selected an L.TileLayer for each of the parameters is created to interact with the WMS server of the Open Weather Map. Through this request data for each selected parameter are retrieved in a tiled manner.

Additionally, when the user selects a location on the map by positioning the mouse, a popup windows appears showing the weather data collected by the nearest meteorological station. The popup window is designed to be easily understandable from any user and provides information on the variability of the air temperature, air pressure, humidity, cloud coverage, wind direction and speed.

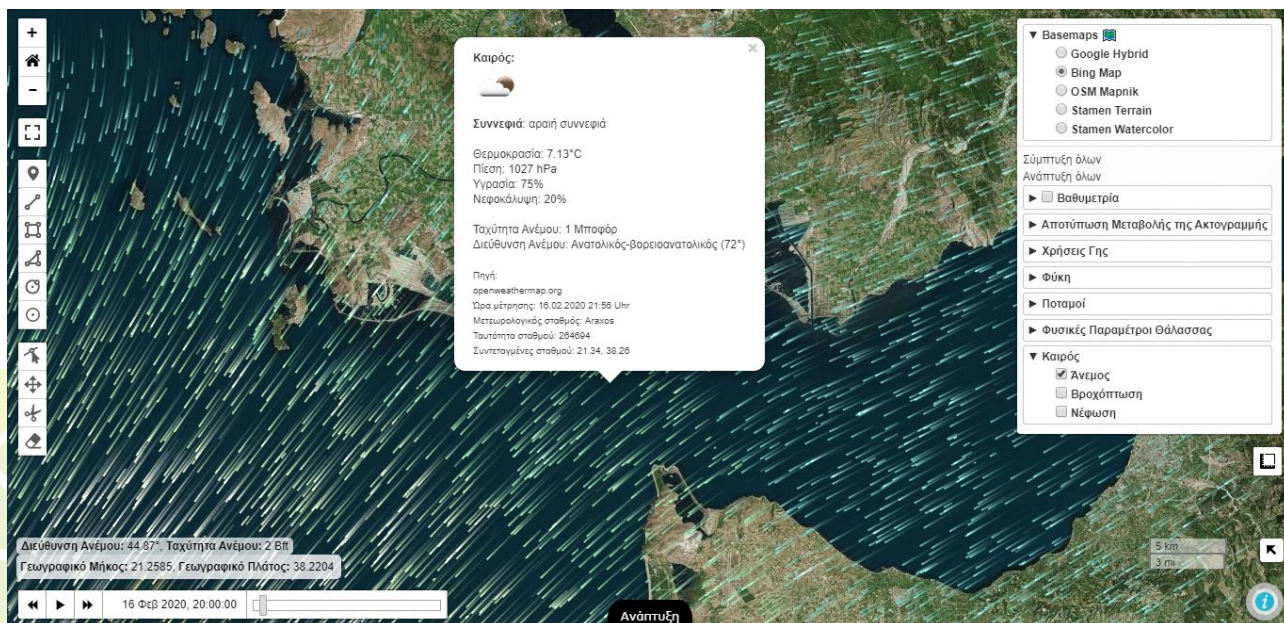


Figure Errore. Nel documento non esiste testo dello stile specificato..33 Animation of wind data as displayed by the TRITON's webGIS page.

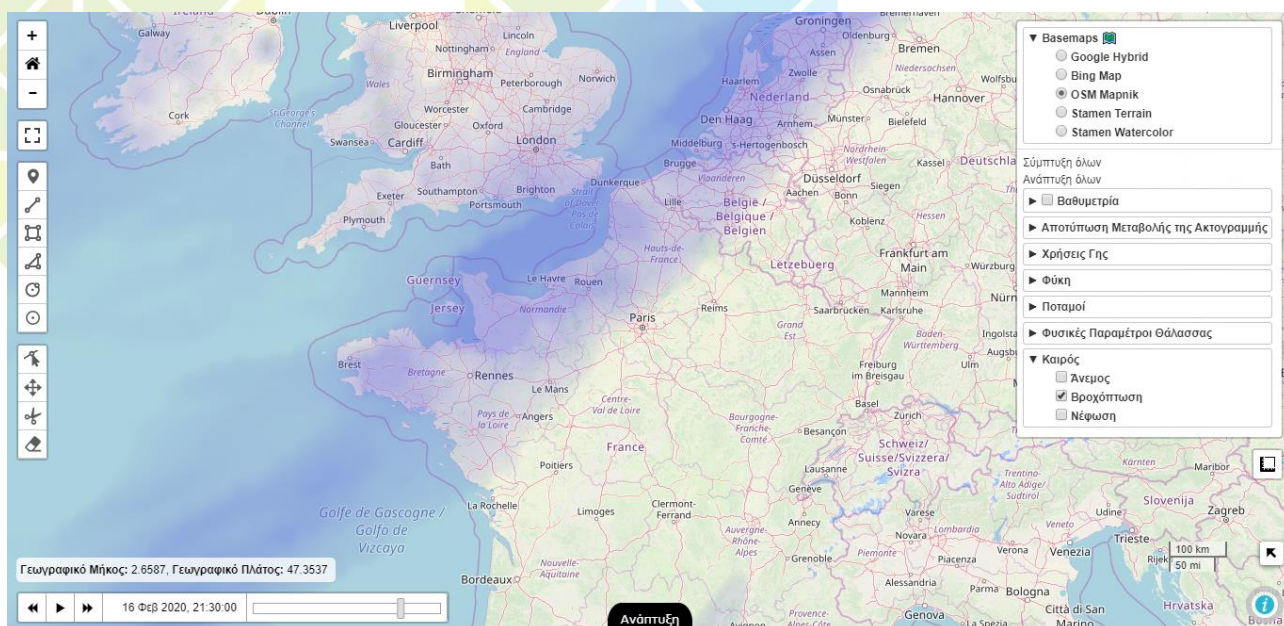


Figure Errore. Nel documento non esiste testo dello stile specificato..34 Precipitation data as displayed by the TRITON's webGIS page.

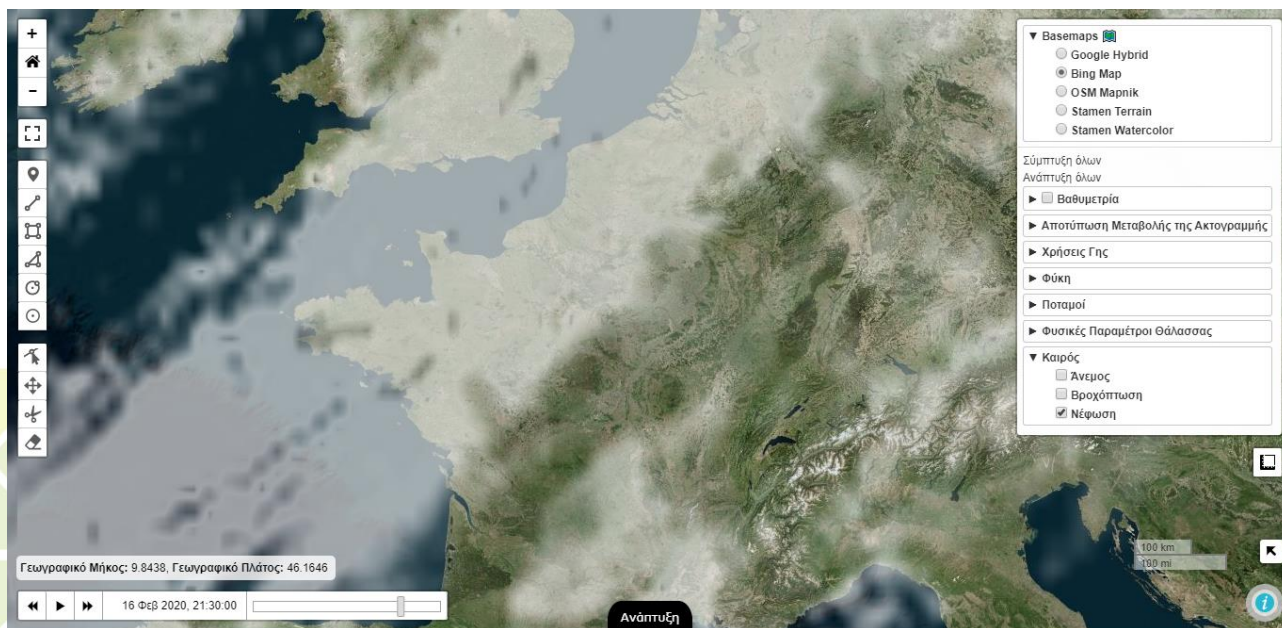


Figure Errore. Nel documento non esiste testo dello stile specificato..35 Cloud Cover data as displayed by the TRITON's webGIS page.

References

- Agafonkin, V. (2011). Leaflet - a javascript library for interactive maps. Retrieved from <https://leafletjs.com/>
- Carrillo, G. (2012). Web mapping client comparison v.6. Retrieved from <http://geotux.tuxfamily.org/>
- Edler, D., & Vetter, M. (2019). The Simplicity of Modern Audiovisual Web Cartography: An Example with the Open-Source JavaScript Library leaflet.js. *KN - Journal of Cartography and Geographic Information*, 69(1), 51-62. doi:10.1007/s42489-019-00006-2
- ESRI. (2019). Basemaps. *ArcGIS Pro*. Retrieved from <https://pro.arcgis.com/en/pro-app/help/mapping/map-authoring/author-a-basemap.htm>
- Farkas, G. (2017a). Applicability of open-source web mapping libraries for building massive Web GIS clients. *Journal of Geographical Systems*, 19(3), 273-295. doi:10.1007/s10109-017-0248-z
- Farkas, G. (2017b). *Practical GIS*: Packt Publishing Ltd.
- Lindström, G., Pers, C., Rosberg, J., Strömqvist, J., & Arheimer, B. (2010). Development and testing of the HYPE (Hydrological Predictions for the Environment) water quality model for different spatial scales. *Hydrology research*, 41(3-4), 295-319.
- MacWright, T. (2013). Announcing MapBox.js 1.0 with Leaflet. URL. Retrieved from <https://www.mapbox.com/blog/mapbox-js-with-leaflet/>
- Mapbox, I. (2016). Mapbox.js Examples. Retrieved from <https://www.mapbox.com/mapbox.js/example/v1.0.0/>
- Muehlenhaus, I. (2013). *Web Cartography: Map Design for Interactive and Mobile Devices*: Taylor & Francis.
- Peterson, G. N. (2014). *GIS Cartography: A Guide to Effective Map Design, Second Edition*: CRC Press.
- Putz, S. (1994). Interactive information services using World-Wide Web hypertext. *Computer Networks and ISDN Systems*, 27(2), 273-280.
- ReiBig, M. (2016, 2016). *Semantic Markup for Geographic Web Maps in HTML*.
- Smith, D. A. (2016). Online interactive thematic mapping: Applications and techniques for socio-economic research. *Computers, Environment and Urban Systems*, 57, 106-117. doi:<https://doi.org/10.1016/j.compenvurbsys.2016.01.002>
- West, M. (2012). *HTML5 Foundations*: Wiley.



Lead Partner



Technical support



Project Partners



www.greece-italy.eu



Interreg V-A Greece-Italy Programme is a European Territorial Cooperation Programme that aims to help public institutions and local stakeholders to develop cross-border projects and pilot actions and to create new policy, products and services, with the final goal to improve the citizens' quality of life. Strategically, the programme will enhance innovation in a number of fields such as blue growth, tourism and culture, agro food and cultural and creative industries. Interreg V-A Greece-Italy Programme aims to get maximum return from EUR 123 million financed per 85% by European Regional Development Fund (ERDF) and per 15% by the 2 member states through a national co-financing.

Lead Partner



Technical support



Project Partners

