



Macaronesian Maritime Spatial Planning

D.6.10 WEB VISOR REPORT

MarSP Deliverable:
D.6.10 Web Visor – MarSP Atlas

December 2019



Coordinated by



Funded by



Partners



WP name	WP6. Macaronesian Cross-border Cooperation
Task name	Task 6.3. Cross-border cooperation challenges
Deliverable Name	D.6.10 Web Visor – MarSP Atlas report
Due Date of deliverable	31.10.2019
Actual submission Date	31.12.2019
Citation	Tello Antón O, Agudo Bravo LM, Bruque Carmona G, González Gil S, Suarez de Vivero JL. 2019. Web Visor – MarSP Atlas report. Deliverable - D.6.10., under the WP6 of <i>MarSP: Macaronesian Maritime Spatial Planning</i> project (GA nº EASME/EMFF/2016/1.2.1.6/03SI2.763106).

Document Information	
Document Name	Web Visor – MarSP Atlas report
Document ID	D.6.10
Version	1
Version Date	31.12.2019
Author(s)	Olvido Tello Antón, Luis Miguel Agudo Bravo, Gerardo Bruque Carmona, Sonsoles González Gil, Juan Luis Suarez de Vivero
Dissemination Level:	Public

Summary (English)

This report aims to inform about the deliverable D.6.10. Web Visor – MarSP Atlas. Its objectives, functionality and technology.

Resumen (Español)

El objetivo de este informe es dar a conocer el deliverable D.6.10 Web Visor – MarSP Atlas. Sus objetivos, funcionalidades y tecnología en la que se ha basado su desarrollo.

Sumário (Português)

O objetivo deste relatório é divulgar o D.6.10 Web Viewer - MarSP Atlas. Objetivos, funcionalidades e tecnologias nas quais se baseou para o seu desenvolvimento.

Contenidos

1. Introduction	2
2. Objectives of Web Visor – MarSP Atlas	2
3. Functionality	2
4. Technology.....	3
5. Development methodology.....	5

1. Introduction

The deliverable D.6.10 Web Visor report is a document about the Web Visor MarSP Atlas. This Web Visor is a deliverable of “WP6. Macaronesian Cross-border Cooperation.” “Task. 6.3 Cross-border cooperation challenges”. This task involves activities that cover important issues for cross-border cooperation in Macaronesia. An important deliverable of this task is the MarSP Atlas of the Macaronesia (D.6.9), its aim is to afford users an information platform with maps, statistical data and other complementary information that encourages users to learn about maritime scenarios, to view them and form opinions about them.

With the aim to facilitate the consult of maps and data included in the Atlas, all information has been integrated into a Web Visor (D.6.10), so it can be consulted for everyone everywhere.

2. Objectives of Web Visor – MarSP Atlas

The main objective of Web Visor is to visualize interactively through the internet all the geographical information integrated on the maps contained in the Atlas of Macaronesia, which could be necessary to advice in Maritime Spatial Planning.

One major advantage of this Visor is that it can be consulted by anyone interested in any place (the Visor itself as well as the data included in the Visor).

The Web viewer will allow making interactive queries to the data represented in the Atlas maps, obtaining a better performance of it.

The URL to connect with the Web Visor MarSP Atlas is:

<http://www.ideo-atlasmarsp.ideo.es/Home>

3. Functionality

The Web viewer will allow visualizing the geographical layers which represent the data included in the Atlas.

The main functionalities included in the Viewer are:

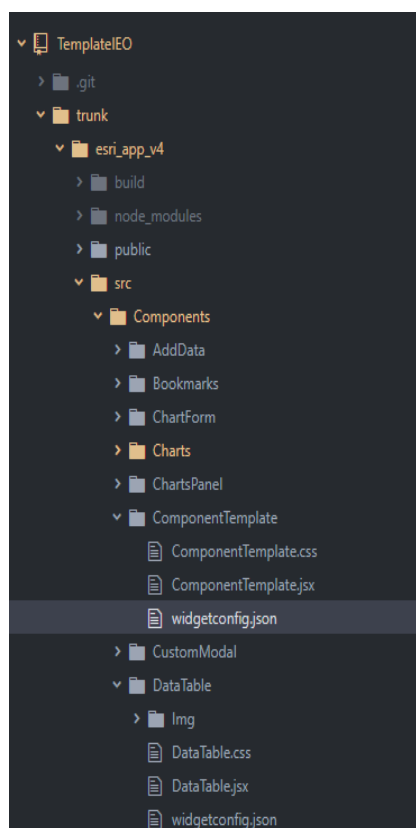
- To visualize the layers, turn on or turn off.
- To consult the attributes associated with each feature in each layer.
- To consult the attribute tables for all layers.
- To make measures
- To add new layers with new dataset from excel files containing geographical coordinates.
- To print a map.
- To visualize data in 3D.

4. Technology

All the services used in the viewer were generated using ArcGIS Server from ESRI and are hosted in the IEO servers. ArcGIS Server is a back-end server software that makes the geographic information available to other users. This is accomplished through web services, which allow a powerful server computer to receive and process requests for information sent by other services. Once web services are running, it's possible to use them in any app, device or API than can communicate through HTTP protocols.

The Web viewer was developed using a customizable template. The Spanish Institute of Oceanography (IEO) has developed a new customizable web application template, using REACT.JS library in the client side, the API for Javascript 4.6 from ESRI as map library, and NPM/Node.JS as dependency manager. This application template is open source. It is available in a GITHUB repository (<https://github.com/IEOGit/IEOTemplate>).

The application template is customizable, at the style level and at the functionality level. In the appconf.JSON file, the user can select the widgets availables, the URL of the proxy file (to handle the requests from the final users), and the languages availables (the user can define new languages in the components). Each widget/component has a folder with three files.



The css file has the info about the styles exclusives of this component. JSX file has the logic in the Javascript/REACT code language. Finally in the JSON file the user has available all the configurable parameters for this widget. For example, in the widgetconfig.JSON file stored in the EsriMap folder is possible to configure the operational layers in the map, in addition to many other style options map.

The user can create new widgets/components with new functionality and reuse these components in new applications. The architecture of one component is a folder with three files (CSS, JSX and JSON) described in the “customizing template” section. The name of the CSS and JSX files is the same that the folder’s name. The json file is called “widgetconfig.json”.

In the JSON file the user can define if the component is loaded in view or if is loaded in the left toolbar or if is necessary that has access to the layers, and all requirements in order to deploy in the application. Also it is possible to define the configurable parameters for this new component.

All the new necessary dependencies in this new component are added in the package.json file, and used by NPM/Node.JS in order to install them. The logic of the component is developed in the JSX file (is a preprocessor step that adds XML syntax to JavaScript. JSX makes React more elegant).

```
[
  "definition":{
    "layers": true,
    "onviewloaded": true,
    "id":"widget_compTemplate",
    "custom":{
      "inPanel": true,
      "icon": "glyphicon glyphicon-pushpin",
      "mountingnode":"",
      "title":""
    },
    "comments": ""
  },
  "labels":[{
    "id": 0,
    "es": "Etiqueta",
    "en": "Label"
  }]
]
```

Once downloaded the user has a large number of configurable widgets available (Measure, Info, Graphs, Data Tables, Legend, 3D, Symbology, Print, Add Data, Heat Maps, etc.). It is also possible to develop new components/widgets. In the MarSP Atlas viewer are included the following widgets:

- Layer manager: Modify or visualize several paramaters for each layer.
- Bookmarks: Access and create new bookmarks for interested areas
- Legend: Legend simbology.
- Measure: Tool to draw poligons and lines in order to get its measure.
- Info: Tool to get information for each feature.
- Add data: Tool to add external information.
- 3D: Tool in order to visualize layer in 3D.

In order to manage users, roles, projects, access permits and widgets/components in the server side, another web application was developed using .NET framework. This application is also available in the GITHub repository (<https://github.com/IEOGit/IEOTemplate>).

To deploy this app only a server with Windows as OS and with the .NET libraries available is required. Also a SQL Server database is necessary in order to store the information about users,

roles, projects and widgets. All the information to deploy and to configure this application is described in a document in the GitHub repository.

5. Development methodology

The methodology has been based mainly on the potential and strength of the tools offered by the Geographic Information Systems (GIS). The tools of these programs are essential for the management of cartographic information, its agile visualization and its analysis.

On this occasion, as SIG, ArcMap software (ArcGIS, ESRI) was used to create and develop a GIS project with all the relevant information for MarSP, both the one used for the analysis and the generated one.

The different data sets were loaded and visualized in the project.

After that, all the spatial information was analyzed and grouped according to the subject of the information contained in each one. In this way, the following groups of thematic layers were created. As an example:

- General overview
- Portugal: basic mapping
- Spain: basic mapping
- European Union Jurisdiction Cartography
- Exclusive Economic Zones and Jurisdiction Cartography
- Macaronesia Third Countries
- Planning Scope
- Major Territorial Divisions
- Regulations and Spatial Planning
- Uses and Activities
- Nature and Conservation

Then, when all thematic cartographic layers were available from the same GIS project, it was stored in a single data warehouse (geodatabase format) but internally organized into subsets (Feature Datasets) in the same way as the thematic groups mentioned above.

Next, the information was carefully studied by comparing with that shown in the MarSP Atlas in order to approximate the symbology as closely as possible to that represented in the MarSP Atlas.

Some layers or datasets in which information was duplicated were also removed.

Finally this GIS project is read from ArcGIS Server to be able to serve it through the web and to be able to visualize the information collected in the project and contained in the MarSP Atlas. This web service is used in the MarSP Atlas Viewer.