everest

# Technical Note on Digital Information and E-Collaboration Services - Intermediate Version

| | | |
|---|---|---|
| **Workpackage** | **5** | **VRE Infrastructure and Service Design and Development** |
| **Task** | 5.3 | Digital Information and E-Collaboration Services |
| **Author (s)** | Sergio Ferraresi | MEEO |
| | Simone Mantovani | MEEO |
| **Reviewer (s)** | Hazel J. Napier | BGS |
| | Raul Palma | PSNC |
| **Approver (s)** | Cristiano Silvagni | ESA |
| | Pedro Gonçalves | Terradue |
| **Authorizer** | Mirko Albani | ESA |
| **Document Identifier** | EVER-EST DEL WP5-D5.3 | |
| **Dissemination Level** | Public | |
| **Status** | Approved by the EC | |
| **Version** | 1.1 | |
| **Date of issue** | 14 December 2018 | |

# Document Log

| Date | Author | Changes | Version | Status |
|---|---|---|---|---|
| 27/6/2016 | Fulvio Marelli | TOC | 0.1 | Draft |
| 07/10/2016 | Simone Mantovani | TOC | 0.2 | Draft |
| 19/10/2016 | Sergio Ferraresi | Updates | 0.3 | Draft |
| 10/11/2016 | Sergio Ferraresi | Reviewed NHP/PSNC comments. Added: Virtual Globe functions;. | 0.4 | Draft |
| 15/11/2016 | Sergio Ferraresi | Updated Sea Monitoring VRC.<br>Added: RO Search Visual Component; Natural Hazards, Land Monitoring and Supersites VRC descriptions. | 0.4 | Draft |
| 21/11/2016 | Sergio Ferraresi | Review. Moved installation guides to Annex A. | 0.5 | Draft |
| 05/12/2016 | Simone Mantovani | Final release | 1.0 | Draft to be approved by the EC |
| 14/12/2018 | Cristiano Silvagni | Document status | 1.1 | Approved by the EC |

# Table of contents

**1 Introduction ..... 10**
**1.1 Purpose and scope ..... 10**
**1.2 Who shall read this document ..... 10**
**1.3 System context ..... 10**
**1.4 Services list ..... 10**
**2 EVER-EST Portal ..... 11**
**2.1 Service overview ..... 11**
**2.2 VRC portals ..... 12**
2.2.1 Sea monitoring ..... 12
2.2.2 Natural Hazards ..... 14
2.2.3 Land monitoring ..... 15
2.2.4 Supersites ..... 16
**3 Digital Information Services ..... 17**
**3.1 Service overview ..... 17**
**3.2 Service visual components ..... 17**
3.2.1 EO data discovery visual component ..... 17
*3.2.1.1 Advanced search panel ..... 18*
*3.2.1.2 Results Panel ..... 18*
*3.2.1.3 Adding the visual component ..... 19*
3.2.2 RO manager visual component ..... 20
*3.2.2.1 RO manager bar ..... 20*
*3.2.2.2 RO search ..... 22*
*3.2.2.3 RO creator ..... 24*
*3.2.2.4 Add To RO button ..... 25*
*3.2.2.5 Remove from RO button ..... 26*
3.2.3 Virtual globe visual component ..... 27
*3.2.3.1 Adding the visual component ..... 29*
3.2.4 Upload to Seafile visual component ..... 30
**3.3 Using the service – quick overview ..... 30**
3.3.1 Performing discovery search (one-step search) on virtual globe ..... 30
3.3.2 Performing discovery search (two-step search) on virtual globe ..... 31
3.3.3 Performing research object creation ..... 32
3.3.4 Performing research object visualization ..... 32
3.3.5 Performing upload to Seafile ..... 33
**3.4 Service public API's ..... 33**
3.4.1 RO manager VC APIs ..... 33
*3.4.1.1 Create RO ..... 33*
*3.4.1.2 Get List of ROs ..... 35*
3.4.2 Virtual globe VC APIs ..... 39

**4 E-Collaboration Services ..... 40**

**4.1 Service overview ..... 40**

4.1.1 Forum overview ..... 40

4.1.2 Instant messaging overview ..... 40

4.1.3 Internal messaging and notification overview ..... 42

4.1.4 Calendar/scheduling overview ..... 42

**4.2 Service visual components ..... 43**

4.2.1 Forum ..... 43

4.2.2 Instant messaging ..... 44

4.2.3 Internal mailing and notification ..... 45

4.2.4 Calendar/scheduling ..... 46

*4.2.4.1 Task Lists ..... 47*

*4.2.4.2 My Tasks ..... 47*

**Annex A. Installation Guide ..... 49**

# List of Figures

Figure 2-1 EVER-EST Portal Home Page ..... 11
Figure 2-2 Sea Monitoring VRC Portal ..... 13
Figure 2-3 Natural Hazards VRC Portal ..... 14
Figure 2-4 Land Monitoring VRC Portal ..... 15
Figure 2-5 Supersites VRC Portal (a) and detailed view of Iceland Volcanoes (b) ..... 16
Figure 3-1 Visual Component ..... 17
Figure 3-2 EO Data Discovery Visual Component ..... 18
Figure 3-3 Advanced search panel ..... 18
Figure 3-4 Results panel ..... 19
Figure 3-5 RO Manager Visual Component ..... 21
Figure 3-6 RO manager tabs ..... 21
Figure 3-7 RO search icon ..... 22
Figure 3-8 RO search filters ..... 23
Figure 3-9 RO creator steps ..... 24
Figure 3-10 Add To RO visual component ..... 25
Figure 3-11 Remove From RO visual component ..... 26
Figure 3-12 Virtual Globe visual component ..... 28
Figure 3-13 One-step EO data discovery ..... 31
Figure 3-14 Two-steps EO data discovery ..... 32
Figure 4-1 Forum icon ..... 43
Figure 4-2 Forum home page ..... 43
Figure 4-3 Topics list ..... 44
Figure 4-4 Topic ..... 44
Figure 4-5 Instant messaging icon ..... 44
Figure 4-6 Instant messaging bar ..... 45
Figure 4-7 Internal mailing icon ..... 45
Figure 4-8 Inbox ..... 46
Figure 4-9 Calendar/scheduling icon and menu ..... 46
Figure 4-10 Task list ..... 47
Figure 4-11 My tasks ..... 48

# List of Tables

Table 3-1 Create RO API – Request body parameter description ..... 34
Table 3-2 Create RO API – Response HTTP codes and descriptions ..... 34
Table 3-3 Create RO API – Response Body Parameter Description ..... 35
Table 3-4 Get list of ROs API –Request body parameter description ..... 36
Table 3-5 Get list of ROs API - Response HTTP codes and descriptions ..... 37
Table 3-6 Get list of ROs API – Responsebody parameter description ..... 38
Table 4-1 EVER-EST portal system configuration ..... 49
Table 4-2 EVER-EST portal virtual environment configuration ..... 49
Table 4-3 EVER-EST portal source folder configuration ..... 50
Table 4-4 EVER-EST portal source folder configuration ..... 50
Table 4-5 EVER-EST portal main configuration ..... 51
Table 4-6 eJabberd server installation ..... 52
Table 4-7 eJabberd mod_webpresence module. ..... 53

# Definitions and Acronyms

| Acronym | Description |
|---|---|
| **ABAC** | Attribute Based Access Control |
| **ACL** | Access Control List |
| **AJAX** | Asynchronous JavaScript and XML |
| **AMQP** | Advanced Message Queuing Protocol |
| **API** | Application Programming Interface |
| **BAPI** | Business Application Programming Interface |
| **CMS** | Content Management System |
| **CRL** | Certificate Revocation List |
| **CRUD** | Create read Update Delete |
| **CSV** | Comma-Separated Values |
| **DOI** | Digital Object Identifier |
| **E-PDSC** | Extended-Preservation Dataset Content |
| **EAI** | Enterprise Application Integration |
| **EDA** | Event Driven Architecture |
| **EDI** | Electronic Data Interchange |
| **EO** | Earth Observation |
| **ERP** | Enterprise Resource Planning |
| **ES** | Earth Science |
| **ESA** | European Space Agency |
| **EVAT** | Expert-user Visual Analysis Tool |
| **EVER-EST** | European Virtual Environment for Research - Earth Science Themes |
| **FTP** | File Transfer Protocol |
| **FTPS** | FTP over SSL |
| **GIF** | Graphics Interchange Format |
| **GUI** | Graphical User Interface |
| **HTML** | Hypertext Mark-up Language |
| **HTTP** | Hypertext Transfer Protocol |
| **HTTPS** | HTTP over TLS, HTTP over SSL, and HTTP Secure |
| **IDE** | Integrated Development Environment |
| **IIOP** | Internet Inter-ORB Protocol |
| **IMAP** | Internet Message Access Protocol |
| **IO** | Input Output |
| **IPR** | Intellectual Property Rights |
| **IS** | Identity Server |

| | |
|---|---|
| **ISO** | International Organization for Standardization |
| **IT** | Information Technology |
| **IWA** | Integrated Windows Authentication |
| **JIT** | Just In Time |
| **JMX** | Java Management Extension |
| **JPEG** | Joint Photographic Experts Group |
| **JSON** | JS Object Notation |
| **LDAP** | Lightweight Directory Access Protocol |
| **LGPL** | Lesser General Public License |
| **MEA** | Multi-sensor Evolution Analysis |
| **MLLP** | Minimum Lower Layer Protocol |
| **MOM** | Message Oriented Middleware |
| **MQTT** | MQ Telemetry Transport |
| **OAGIS** | Open Applications Group Integration Specification |
| **OASIS** | Organization for the Advancement of Structured Information Standards |
| **OCSP** | Online Certificate Status Protocol |
| **OGC** | Open Geospatial Consortium |
| **OPI** | Operator Panel Interface |
| **PBAC** | Policy Based Access Control |
| **PDSC** | Preservation Dataset Content |
| **PNG** | Portable Network Graphics |
| **POP** | Point of presence |
| **RBAC** | Role Based Access Control |
| **RDF** | Resource Description Framework |
| **REST** | Representational State Transfer |
| **RO** | Research Objects |
| **RODL** | Research Objects Digital Library |
| **RSS** | Rich Site Summary |
| **SFTP** | Secure File Transfer Protocol |
| **SLA** | Service Level Agreement |
| **SMTP** | Simple Mail Transfer Protocol |
| **SPARQL** | Protocol and RDF Query Language |
| **SQL** | Structured Query Language |
| **SSO** | Single Sign-On |
| **SVG** | Scalable Vector Graphics |
| **TCP** | Transmission Control Protocol |
| **UDP** | User Datagram Protocol |

| | |
|---|---|
| **URI** | Uniform Resource Identifier |
| **URL** | Uniform Resource Locator |
| **VAC** | Visual Analysis Client |
| **VC** | Visual Component |
| **VRC** | Virtual Research Community |
| **VRE** | Virtual Research Environment |
| **WCF** | Windows Communication Foundation |
| **WCPS** | Web Coverage Processing Service |
| **WCS** | Web Catalogue Service |
| **WebCGM** | Web Computer Graphics Metafile |
| **WFMS** | Workflow Management Systems |
| **WFS** | Web Feature Service |
| **WMS** | Web Map Service |
| **XML** | EXtensible Mark-up Language |

## Applicable Documents

| Document ID | Document Title |
|---|---|
| **EVER-EST DEL WP5-D5.1** | VRE Architecture and Interfaces Definition |

## Reference Documents

| Document ID | Document Title |
|---|---|
| **EVER-EST DEL WP4-D4.2** | Workflows and Research Objects models in Earth Science |
| **EVER-EST DEL WP4-D4.3** | Design, implementation and deployment of Research Objects components for Earth Science |
| **EVER-EST DEL WP5-D5.1** | VRE Architecture and Interfaces Definition |
| **EVER-EST DEL WP5-D5.2** | Technical Note on Common Services |
| **EVER-EST DEL WP5-D5.4** | Technical Note on e-Research application services |
| **EVER-EST Visual Components List** | http://vm1.everest.psnc.pl/components/ |

# 1 Introduction

This Technical Note is the intermediate version of the deliverable of Task 5.3, whose objective is to design and implement the Digital Information and E-Collaboration Services that will be used within the EVER-EST Virtual Research Environment.

This document takes into account the overall design sketched in the DoA and refined during the first phase of the project as reported in deliverable D5.1, [EVER-EST DEL WP5-D5.1].

## 1.1 Purpose and scope

The purpose of this document is to describe the design of the Digital Information and E-Collaboration Services that will be integrated and used within the first version of the EVER-EST system.

## 1.2 Who shall read this document

This document has been thought mainly to provide implementation and configuration information about the Digital Information and E-Collaboration services to the technical partners in the consortium. It contains also some information about using of the common services that can be useful for the Virtual Research Community members.

## 1.3 System context

Digital Information and E-Collaboration Services provide the tools necessary to assist the users in solving the challenges of sharing the necessary data and resources within the Virtual Research Community. The Django Content Management System (CMS)[1]supports the easy integration of other existing web applications and it has a very intuitive drag and drop interface that facilitates all the needs for multi-lingual publishing and editing of websites pages and content.

The CMS offers the core collaboration tools as well as the easy integration of third party components via the support of Application Programming Interface's(API's) tools for (e.g.) video conferencing, instant messaging and notification. Additional information shall be provided about specific e-collaboration services that will focus on research objects and that shall be implemented during the project.

## 1.4 Services list

The Digital Information and E-Collaboration services offer:

- EVER-EST and Virtual Research Community portals;
- EO Data Discovery Visual Component;
- RO Manager Visual Component;
- Virtual Globe Visual Component;
- Upload to Seafile[2] Visual Component;
- Forum capabilities;
- Instant Messaging capabilities;
- Internal Messaging and Notification capabilities;
- Calendar/Scheduling capabilities.

[1]https://www.django-cms.org/en/

[2] https://www.seafile.com/en/home/

# 2 EVER-EST Portal

## 2.1 Service overview

The EVER-EST Virtual Research Environment (VRE) portal provides innovative e-research services to Earth Science user communities for communication, cross-validation and the sharing of knowledge and science outputs:

- Discover, access and process both existing and new heterogeneous Earth Science datasets including the associated information and preserved knowledge held by distributed data centres;
- Share data, models, algorithms, scientific results (including traceability of workflows and processes that would facilitate reproducibility of modelling and simulations) and their own experiences within a community or across communities (including those in other domains beyond Earth Science);
- Capture, annotate and store the workflows, processes and results from their research activities;
- Work together in a real-time environment that facilitates the sharing of expertise, information and data resources overcoming traditional working practices limitations related e.g. to physical meetings or the transfer of large datasets between users;
- Ensure the long-term sustainability and preservation of data, models, workflows, tools and services developed by existing communities of practice that can potentially be re-used in the future by other users either for validation of existing research or for new applications.

The EVER-EST portal and the VRC portals are developed using Django[3] and the Django-CMS[4]Content Management System technologies. The use of CMS simplifies portal development and allows the integration of third-party plugins to expand portal capabilities.

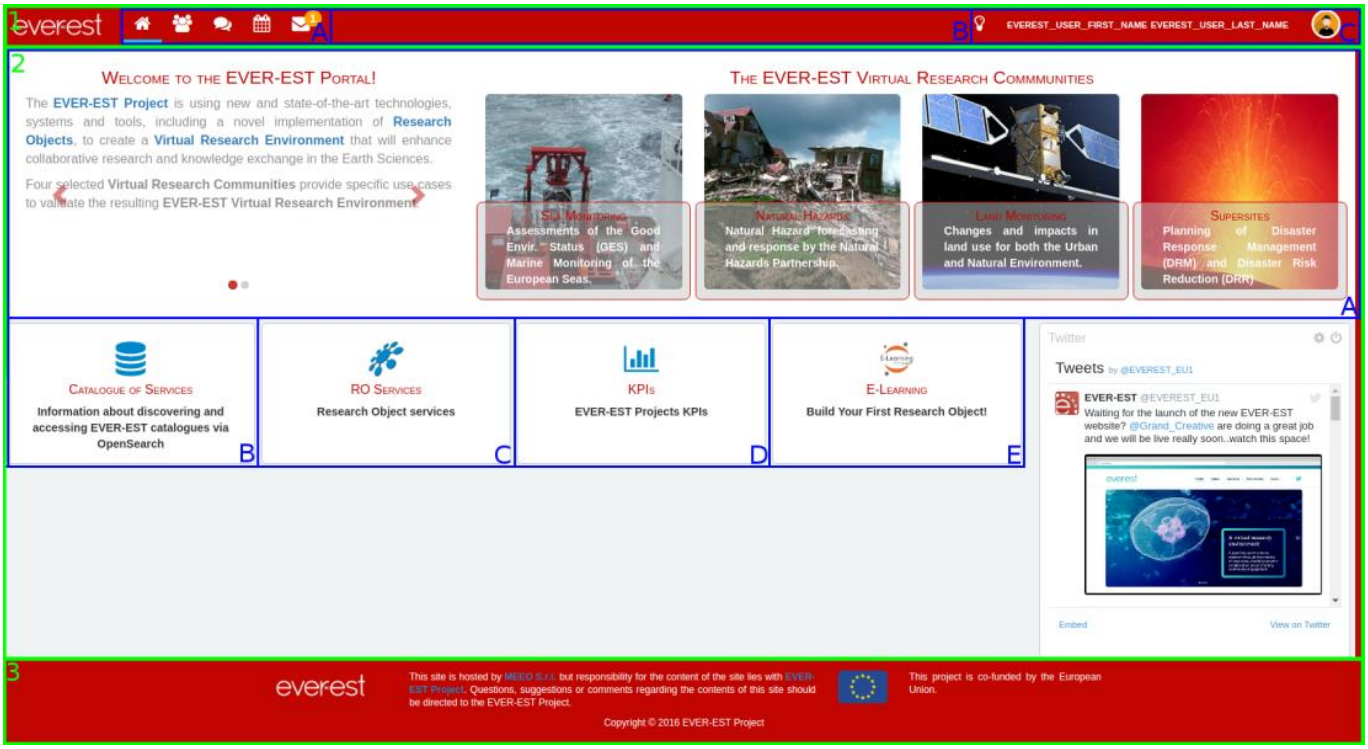



**Figure 2-1 EVER-EST Portal Home Page**

[3]https://www.djangoproject.com/

[4] https://www.django-cms.org/en/

Three main sections compose the EVER-EST Portal:

- The *Header Bar*(section 1 in Figure 2-1) is composed of three areas: the "Core Functions Area"(section 1A) that contains the functionalities shared among the all users belonging to the different Virtual Research Communities; the "VRC Functions Area" (optional, section 1B) that contains VRC related functionalities; and the "Authentication, Authorization and Accounting (AAA) Area" (section 1C) used for user authentication and management.

  The "Core Functions Area" contains mainly the E-Collaboration functionalities (detailed in chapter4, such as forum, internal messaging, calendar and chat capabilities), the "VRC Functions Area" is optional and it has been enabled and customized based on VRC requirements. The "AAA Area" allows the user to login, logout and manage user settings. The authentication components are described in detail in deliverable D5.2 [EVER-EST DEL WP5-D5.2];

- The *Main Area*(section 2 in Figure 2-1) contains:
  - The VRC portals section (section 2A)with the VRC portals entry points;
  - The Catalogue of Services section (section 2B)with information about EVER-EST services. This section shows an overview of the EVER-EST services used to discover and access EVER-EST Catalogues. It allows data discovering using the EO Data Discovery (see chapter 3.2.1) and the Virtual Globe (see chapter 3.2.3);
    More services could be added in the next releases of this document;
  - The RO Services (section 2C) that links to the ROHUB portal[5];
  - The KPIs section (section 2D) that contains the information on the project KPIs;
  - The E-Learning Services (section 2E) that links to the E-Learning modules;
  - The *Footer Area*(section 3 in Figure 2-1) contains the EVER-EST portal's copyright information and the EVER-EST and European Union logos.

The EVER-EST portal has been developed with modularization in mind. Django enables modularization, easily allowing component integration (e.g. forum capabilities). To better support the modularization and facilitate the integration of "modules", the concept of Visual Component (VC)has been conceived and followed by the different service providers. The VCs are independent web elements written using open web technologies (HTML, CSS, and JavaScript/jQuery) that can be added to any web page using a few lines of code. VCs make it possible to centralize a feature, add it on multiple web pages keeping the same behaviour and, at the same time, customize it with some page-related options.

The list of EVER-EST visual components is described in chapter3.

## 2.2 VRC portals

The modularization of the EVER-EST components, allow each VRC to have a dedicated portal that can be customized to fit the individual community's needs. The VRC portals for Sea Monitoring, Natural Hazards, Land monitoring and Supersites are presented in the following sections.

All the portals are under development. Functionalities and user interfaces may change in future releases of this document.

### 2.2.1 Sea monitoring

"The aim of the European Union's ambitious Marine Strategy Framework Directive (MSFD) is to protect more effectively the marine environment across Europe achieving the Good Environmental Status (GES) of the EU's

[5] http://www.rohub.org/portal/home

marine waters by 2020 and to protect the resource base upon which marine-related economic and social activities depend. It is the first EU legislative instrument related to the protection of marine biodiversity, as it contains the explicit regulatory objective that biodiversity is maintained by 2020, as the cornerstone for achieving GES".

According to the MSFD, the environmental status of marine waters is defined by 11 descriptors, and forms a proposed set of 29 associated criteria and 56 indicators. The objective of the Sea Monitoring Virtual Research Community (VRC) is to provide useful and applicable contributions to the evaluation of the variables that define Good Environment Status (GES) according to the EU's Marine Strategy Framework Directive (MSFD), visit website.

Biodiversity is the first descriptor of GES, the second is the presence of non indigenous species and the remaining nine (taking into consideration physical, chemical or geological variables) require proper functioning of the ecosystem, linked to a good state of biodiversity.

The Sea Monitoring portal provides the virtual research community users with tools and services aimed at enhancing their ability to interoperate and share knowledge, experience and methods for GES assessment and monitoring, such as:

- Digital information services for data management, exploitation and preservation (accessibility of heterogeneous data sources including associated documentation);
- e-collaboration services to communicate and share knowledge, ideas, protocols and workflows;
- e-learning services to facilitate the use of common workflows for assessing GES indicators;
- e-research services for workflow management, validation and verification, as well as visualization and interactive services.

The portal interface contains three main sections:

- The Virtual Globe (with EO Data Discovery and RO Manager VCs) to help researchers search and view EO Data and to facilitate RO creation and management;
- The Right Bar to allow researchers to search ROs (using RO Search VC, described in chapter 3.2.2.2);
- The Bottom Bar placed under the Virtual Globe and shown when required (e.g. to view RO list).

The "VRC Functions Area" of the Header Bar contains two main buttons:

- Seafile Uploader (described in chapter 3.2.4), used to upload files to the Seafile area to be later used;
- RO Search (described in chapter 3.2.2.2), used to show/hide the right bar that contains the Search filters.

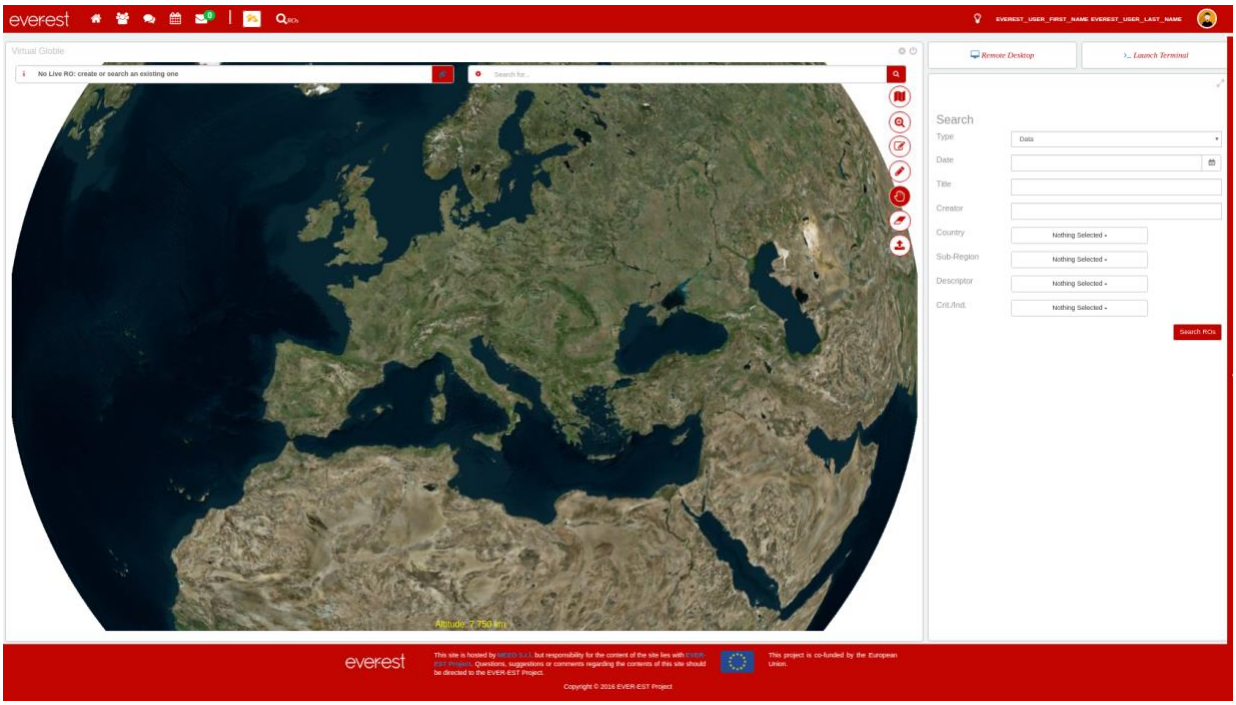



**Figure 2-2 Sea Monitoring VRC Portal**

### 2.2.2 Natural Hazards

The portal for the Natural Hazards community is focused on providing earth data discovery, analysis and visualization tools to support the Natural Hazards Partnership (NHP), a group of 17 collaborating UK public sector organisations comprising government departments, agencies and research organisations.

The NHP provides:

- Mechanism for delivering co-ordinated advice to government and agencies responsible for civil contingency and emergency response during natural hazard events;
- Forum for the exchange of knowledge, ideas, expertise, intelligence and best practice in relation to natural hazards;
- Environment for the development of new services to assist in disaster response.

The NHP has set-up the Hazard Impact Model (HIM) group that is modelling the impact of a range of natural hazards using a common framework and providing operational delivery of the Hazard Impact Model outputs.

Initially HIM is concentrating on modelling the impact of 3 key natural hazards: surface water flooding, land instability and high winds – and their impacts on people, their communities and key assets such as road, rail and utility networks. The partners share scientific information and expertise in hydrological modelling, meteorology, engineering geology, GIS and data delivery and modelling of socio-economic impacts via three integrated work packages.

The partnership is currently UK-based with locations across the UK. A steering group comprising representatives from all partner organisations monitors the work of the partnership.

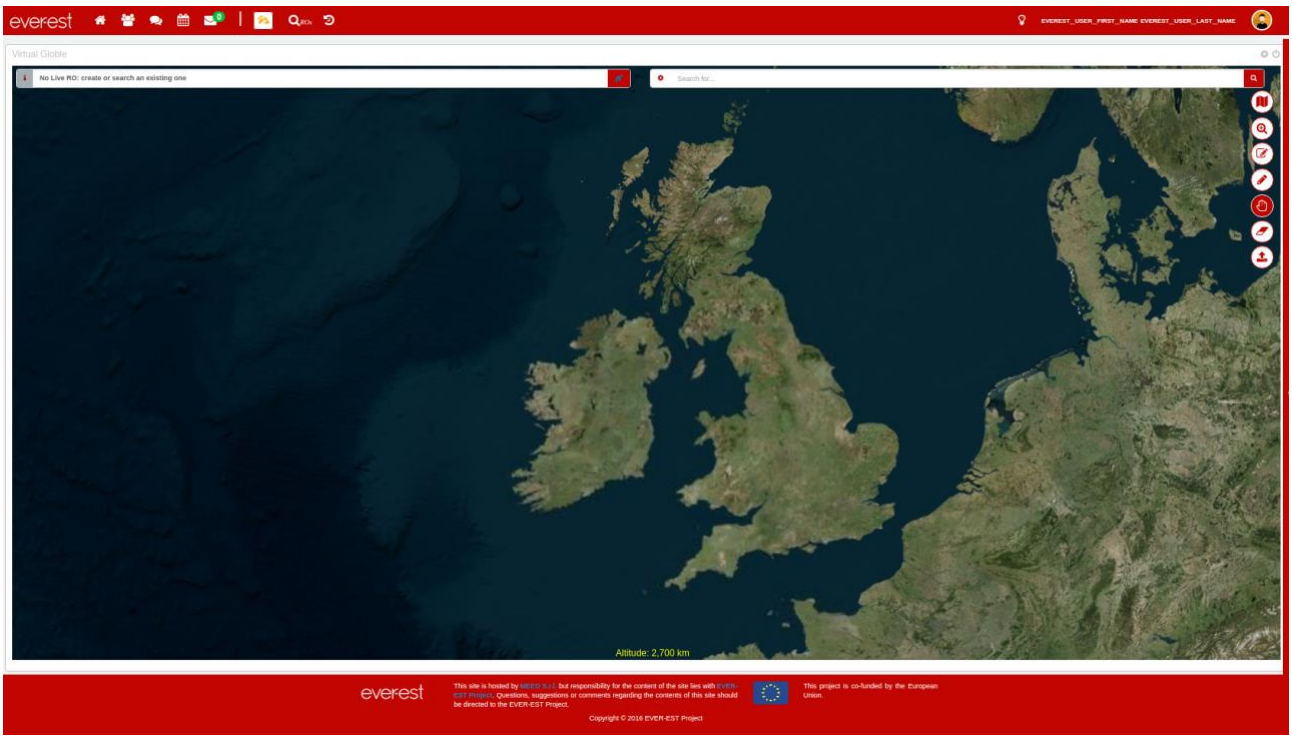



**Figure 2-3 Natural Hazards VRC Portal**

The portal main interface is the Virtual Globe (with EO Data Discovery and RO Manager VCs) to help researchers search and view EO Data and to facilitate RO creation and management.

A Bottom Bar, placed under the Virtual Globe, is shown when required (e.g. to view RO list).

A Right Bar, that contains RO Search filters, can be show using the RO Search (described in chapter 3.2.2.2) button in the “VRC Functions Area” of the Header Bar.

The “VRC Functions Area” of the Header Bar contains other two main buttons:

- "Seafile Uploader" (described in chapter 3.2.4), used to upload files to the Seafile area to be later used;
- "Recent Activity", used to show user’s last actions on ROs.

### 2.2.3 Land monitoring

The portal for the Land Monitoring community is focused on providing earth data discovery, analysis and visualization tools for the virtual research community leaded by the SatCen - European Union Satellite Centre.

Land Monitoring can refer to the monitoring of urban, built-up and natural environments to identify certain features or changes over areas of interest as well as to the assessment of natural resources. Land Monitoring is a transversal issue common to a number of user communities; the main aim is to provide useful information to those entities that have to make informed decisions to address environmental, scientific, humanitarian, health, political and security issues as well as to adopt sustainable management practices.

In this scenario, Earth Observation (EO) data play an important role and European programmes such as Copernicus currently coordinate activities related to their exploitation.

The Land Monitoring portal main interface is the Virtual Globe (with EO Data Discovery and RO Manager VCs) to help researchers search and view EO Data and to facilitate RO creation and management.

A Bottom Bar, placed under the Virtual Globe, is shown when required (e.g. to view RO list).

A Right Bar, that contains RO Search filters, can be show using the RO Search (described in chapter 3.2.2.2) button in the “VRC Functions Area” of the Header Bar.

The “VRC Functions Area” of the Header Bar contains also the Seafile Uploader (described in chapter 3.2.4), used to upload files to the Seafile area to be later used.

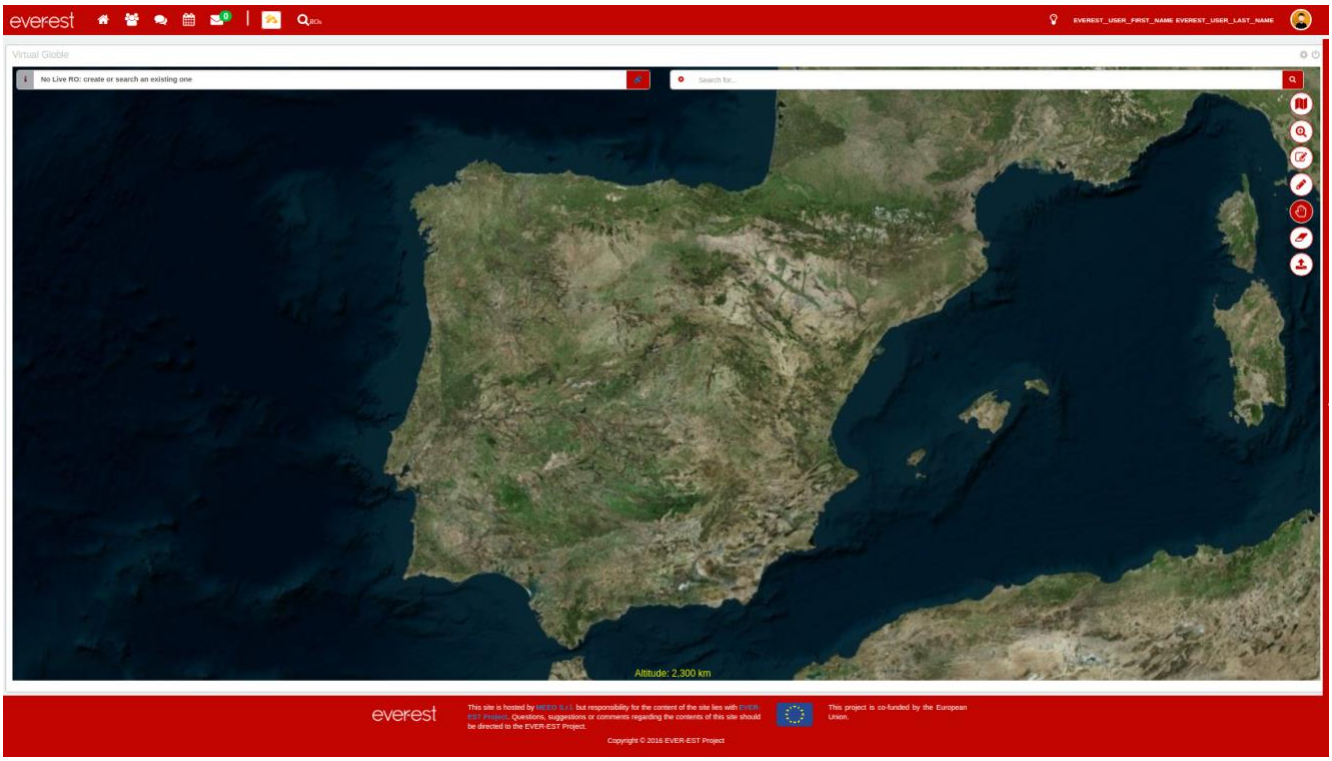



**Figure 2-4 Land Monitoring VRC Portal**

### 2.2.4 Supersites

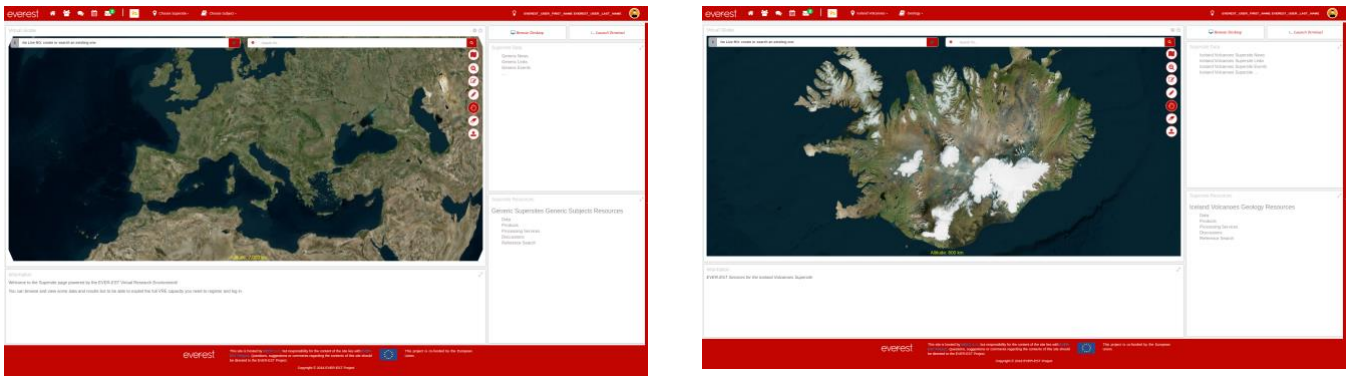
The portal for the Supersites community is tailored to support the scientific community involved in the Geohazard Supersites and Natural Laboratory (GSNL) initiative of GEO. GSNL is a voluntary international partnership aiming to improve, through an Open Science approach, geophysical scientific research and geohazard assessment, promoting rapid and effective uptake of the new scientific results for enhanced societal benefits in Disaster Risk Reduction (DRR).

GSNL is presently focused on seismic and volcanic hazards, and its main objectives are:

- To enable the global scientific community open access to space- and ground-based data, focusing over selected, high risk areas of the world: the Permanent Supersites and the Natural Laboratories;
- To promote advancements in geohazard science over the selected sites;
- To ensure that scientific results are timely used for informed decision-making in DRR activities;
- To innovate technologies, processes, and communication models, enhancing data sharing, global scientific collaboration, and capacity building in geohazard science.

The Supersite community is composed of globally dispersed scientists studying these high-risk volcanic or seismic regions (the Supersites and Natural Laboratories). New scientific results are promoted thorough free access to a large quantity of satellite and ground-based data, contributed by the CEOS space agencies and by local monitoring agencies. The local scientific institutions have a lead role at each Supersite, and provide the most efficient communication channel towards the decision-makers.

Supersite communities directly involved in the development of the EVER-EST VRE includes the governing body of the GSNL initiative and the scientific communities at Mt. Etna, Campi Flegrei/Vesuvius, and the Iceland volcanoes. Further Supersite communities will also be involved in the later stages of the project to provide an independent assessment of the EVER-EST virtual research environment.



(a) (b)

**Figure 2-5 Supersites VRC Portal (a) and detailed view of Iceland Volcanoes (b)**

The Supersite portal contains four main sections:

- The Virtual Globe (with EO Data Discovery and RO Manager VCs) to help researchers search and view EO Data and to facilitate RO creation and management;
- The Top-Right Bar to show data of the Supersite;
- The Bottom-Right Bar to show resources of the Supersite;
- The Bottom Bar to show information about the Supersite.

The “VRC Functions Area” of the Header Bar contains several buttons:

- Seafile Uploader (described in chapter 3.2.4), used to upload files to the Seafile area to be later used;
- “Choose Supersite” button, used to focus on a selected Supersite (e.g.: Iceland Volcanoes)
- “Choose Subject” button, used to focus on a selected Subject (e.g.: Geology)

Based on the Supersite and the Subject, the portal interface changes (see example in Figure 2-5 (b)).

# 3 Digital Information Services

## 3.1 Service overview

The EVER-EST Visual Components constitute the core of the EVER-EST Portal and VRC Portals. The VCs are developed as independent objects that could be added to a web page by adding just a few lines of code.

The VCs currently under development are:

- Discovery Visual Component: allows datasets and data discovery via OpenSearch queries on OpenSearch enabled repositories;
- RO Manager Visual Component: allows RO creation, visualization and editing;
- Virtual Globe Visual Component: allows EO data visualization;
- Upload to Seafile Visual Component: allows files upload to Seafile within the VRC.

The above list is available at[EVER-EST Visual Components List]. A detailed description of the VCs is presented in section 3.2.

## 3.2 Service visual components

To allow VCs integration in web pages, the client side is developed using open web technologies such as:

- HTML5 standard;
- Javascript;
- jQuery;
- Bootstrap 3 Framework;
- CSS 3.

Depending on the VC, the number of elements to include may vary, but generally the JavaScript and CSS dependencies and some HTML lines are required.

For each VC listed in [EVER-EST Visual Components List],the lines of code required to add the visual component to the web page is shown. For example:

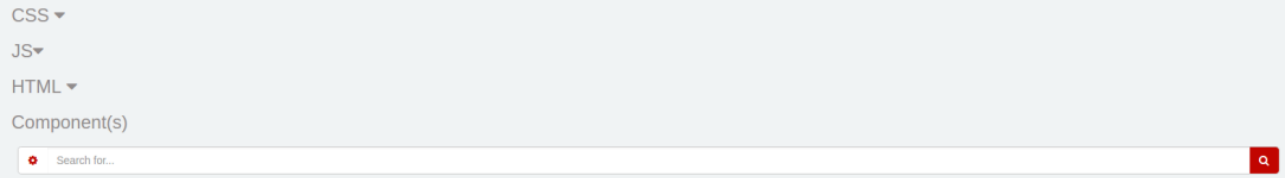



**Figure 3-1 Visual Component**

The visual components are going to be described in the following sections.

### 3.2.1 EO data discovery visual component

The EO Data Discovery VC allows data discovery on OpenSearch repositories. The EO data search may be divided in into two distinct parts:

- Step 1: a client will first identify datasets of interest (collection);
- Step 2: a client search for granules associated with those datasets (results).

This is called the “two-step solution”. The VC allows two-step (collection/results) search, but it also allows the user to do a one-step (results) search on pre-selected datasets/collections.

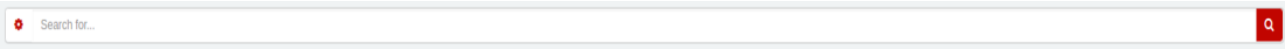



**Figure 3-2 EO Data Discovery Visual Component**

The main interface VC is composed of:

- A Search Text that allows the user to search by key words (depicted in Figure 3-2);
- A Search Button that allows the user to discover data (depicted in Figure 3-2);
- An Advanced Search button that allows the user to customize the search query through the Advanced Search Panel (described in section 3.2.1.1);
- A Show Results button that allows the user to show/hide the *Results Panel* (described in section 3.2.1.2). The button is shown only after a Search request.

To discover data, the user can enter keywords into the Search text, choose filters in the Advanced Search panel and then select the Search button. When the data is retrieved, the Results Button and the Results Panel are shown.

#### 3.2.1.1 Advanced search panel

The Advanced Search Panel allows user customization of the search query.

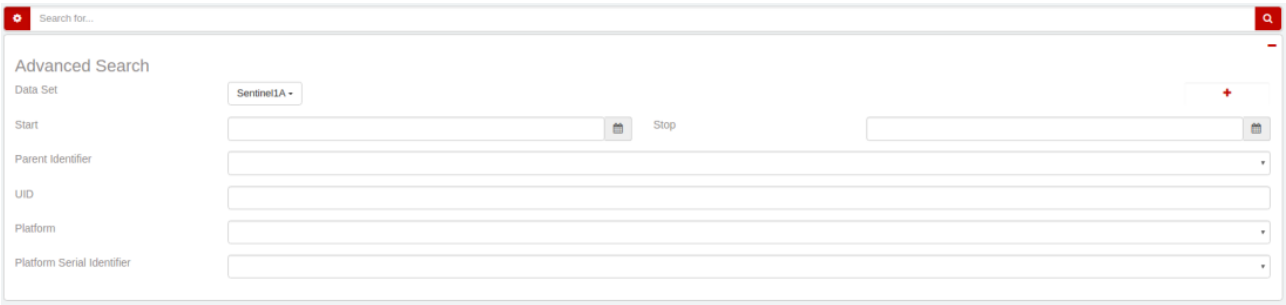



**Figure 3-3 Advanced search panel**

The panel fields are dynamically generated based on the metadata information that is gathered from the OpenSearch repository (chosen via the Data Set field), however there is a group of common filters:

- Data Set: used for data/catalogue search;
- Start and End Dates: used for temporal search;
- Parent Identifier: a string identifying the parent of the entry in a hierarchy of resources;
- UID: the local identifier of the product in the repository context;
- Bounding Box: used for spatial search.

#### 3.2.1.2 Results Panel

The Results Panel contains the search results. The results are shown in a list and the panel provides the following functionalities:

- Expand/Reduce Panel button (section 1 in Figure 3-4): used to enlarge or reduce the results area;
- Minimize Panel button (section 2 in Figure 3-4): used to temporally hide the Results Panel;
- Pagination row (section 3 in Figure 3-4): contains information such as current page number, total page number, set of results shown, total results number and allows the user to move between pages;

- Show Cart button (section 4 in Figure 3-4): downloads the selected products to the VRC. It is enabled when the user selects one or more products.

**Figure 3-4 Results panel**

The Data Set field of the Advanced Search Panel embeds information about the type of search (one-step or two-steps search) and it dynamically generates the Advanced Search Panel fields. The Data Set field also controls the visualization of some functionalities of the Results Panel. All the following buttons are shown only in the results step:

- Show on Map (section A in Figure 3-4): displays the location of the product as a polygon on the Virtual Globe Visual Component (see chapter 3.2.3);
- View Details (section B in Figure 3-4): show the details of the product;
- Add to Cart (section C in Figure 3-4): adds the product to the Cart for future download to the VRC portal;
- Add to RO (section D in Figure 3-4): add the products to the RO that the user is currently working on (using the “Add To RO button” of the RO Manager Visual Component, see Section 3.2.2.4);
- Download to VRC (section E in Figure 3-4): downloads the product to the VRC portal.

Some of these functionalities are under development and could change in the next releases of this document.

#### 3.2.1.3 Adding the visual component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link  rel="stylesheet"  type="text/css"  href="/static/css/lib/bootstrap-datetimepicker.min.css"
/>
<link rel="stylesheet" type="text/css" href="/static/css/lib/bootstrap-multiselect.css" />
<link rel="stylesheet" type="text/css" href="/static/css/lib/clusterize.min.css" />
<link rel="stylesheet" type="text/css" href="/static/css/discovery.css" />
<!-- Add CSS style to adapt to your page (eg.: <style>#d-parent { margin-top: 5em; margin-left:
1em; }</style>) -->
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<script type="text/javascript" src="/static/js/lib/moment.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap-datetimepicker.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap-multiselect.js"></script>
<script type="text/javascript" src="/static/js/lib/clusterize.min.js"></script>
<script type="text/javascript" src="/static/js/discovery.js"></script>
<script type="text/javascript">
$(function () {
        $('#d-parent').discovery();
    });
</script>
```

- HTML:

```
<!-- component-related html -->
<div id="d-parent"></div>
```

### 3.2.2 RO manager visual component

The RO Manager visual component allows the user to search for ROs, view RO details, create a new RO, and edit an existing RO. The main interface of the VC is depicted in Figure 3-5, but the RO Manager is composed of a set of five independent VCs:

- RO Manager Bar;
- RO Search;
- RO Creator;
- Add To RO button;
- Remove From RO button.

#### 3.2.2.1 RO manager bar

The RO Manager Bar, depicted in Figure 3-5, is an enhanced status bar that shows information about the RO in use, and allows RO visualization (see Figure 3-6 RO manager tabs), editing (through Add To RO– see chapter 3.2.2.4 –and Remove From RO – see Figure 3-11 – buttons) and creation (through the RO Creator, see chapter 3.2.2.3).

**Figure 3-5 RO Manager Visual Component**

When a RO is being used, the status bar shows the title of the RO and the "Show RO Details" button is enabled allowing RO visualization and editing via the "Details Panel", which is composed of three tabs:

- Information: shows the main RO information, such as Identifier, Title, Template, and Description;
- Content: shows the content of the RO and allows the user to edit it;
- Progress: based on the RO checklist, allows the user to see the quality of the RO.

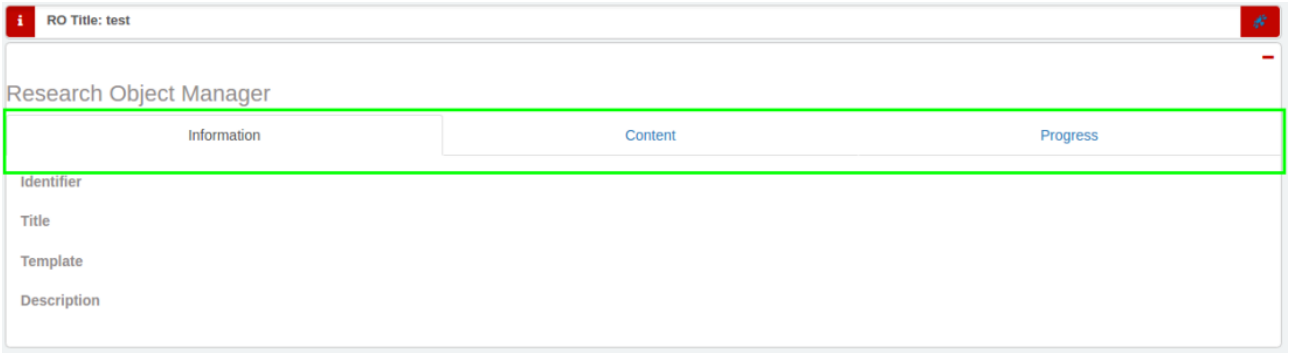



**Figure 3-6 RO manager tabs**

When no RO is selected, the Show RO Details button is not enabled and the status bar shows a default message that informs the user that no RO is selected.

#### 3.2.2.1.1 Adding the Visual Component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/vc/romanager.css">
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
```

```
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<script type="text/javascript" src="/static/js/vc/rocreator.js"></script>
<script type="text/javascript" src="/static/js/vc/addtoro.js"></script>
<script type="text/javascript" src="/static/js/vc/removefromro.js"></script>
<script type="text/javascript" src="/static/js/vc/romanager.js"></script>
<script type="text/javascript">
$(function () {
        $( '#r-parent' ).romanager();
    });
</script>
```

- HTML:

```
<!-- component-related html -->
<div id="r-parent"></div>
```

#### 3.2.2.2 RO search

The RO Search allows the user to search for and to show inquired ROs. The VC contains a default series of search parameters and adding VRC related parameters could extend it. The RO Search is identified by the icon depicted in Figure 3-7:

**Figure 3-7 RO search icon**

When the button is selected the following default search parameters are shown (see Figure 3-8):

- Type: to search for ROs filtering on type (e.g.: workflow);
- Date: to search for ROs filtering on creation date;
- Title: to search for ROs filtering on title;
- Creator: to search for ROs filtering on creator's name.

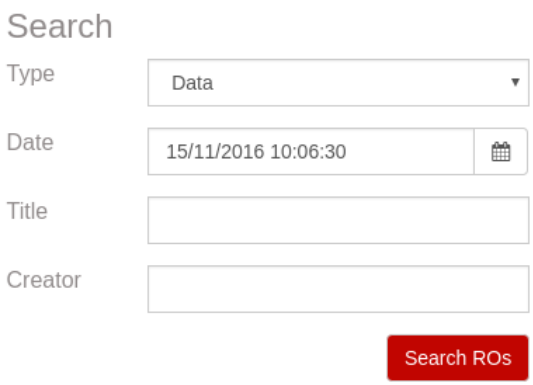



**Figure 3-8 RO search filters**

#### 3.2.2.2.1 Adding the Visual Component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/vc/romanager.css">
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<script type="text/javascript" src="/static/js/vc/rosearch.js"></script>
<script type="text/javascript">
$(function () {
        $( '#r-parent' ).rosearch( {
            searchParent: '#r-filters',
```

```
                listParent: '#r-list',
            } );
        });
    </script>
```

- HTML:

```
<!-- component-related html -->
<div id="r-parent"></div>
<div id="r-filters"></div>
<div id="r-list"></div>
```

#### 3.2.2.3 RO creator

The RO Creator allows the user to create a RO through a three-steps-wizard (see Figure 3-9):

1. RO Data: the first step requests general information about the RO, such as Identifier, Title, Template and Description;
2. RO Initial Content: the second step allows data upload. This is an optional step because the RO could be created without initial content;
3. RO Metadata: the third and last step allows the user to insert RO annotations (see [EVER-EST DEL WP4-D4.2] and [EVER-EST DEL WP4-D4.3]) that will allow RO identification during search queries.

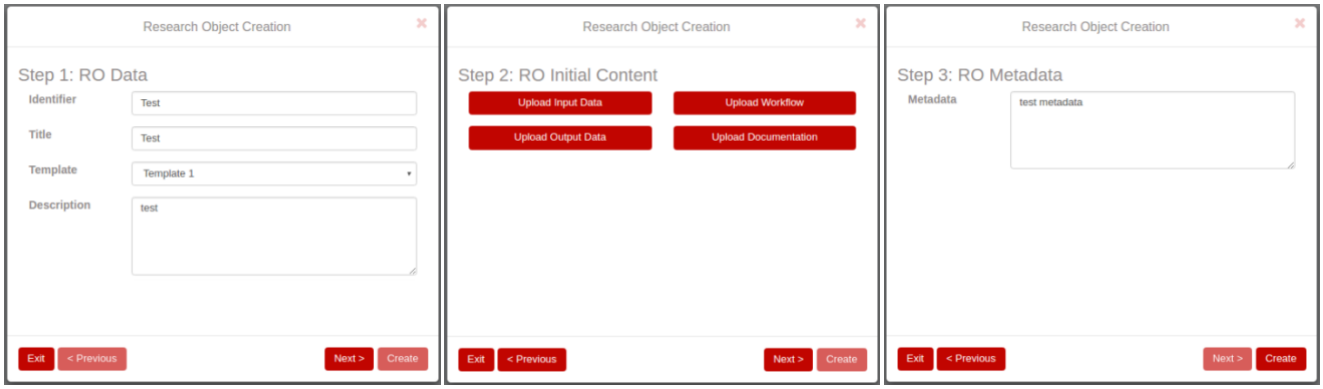



**Figure 3-9 RO creator steps**

##### 3.2.2.3.1 Adding the Visual Component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/vc/romanager.css">
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
```

```
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<script type="text/javascript" src="/static/js/vc/rocreator.js"></script>
<script type="text/javascript">
$(function () {
        $( '#r-cparent' ).rocreator();
    });
</script>
```

- HTML:

```
<!-- component-related html -->
<div id="r-cparent"></div>
```

#### 3.2.2.4 Add To RO button

The Add To RO button allows a resource to be added to a RO. The button, depicted in Figure 3-10, can be integrated into any interface to give to the user the opportunity to add elements to a RO (e.g.: data discovery results, see section D in Figure 3-4).

**Figure 3-10 Add To RO visual component**

##### 3.2.2.4.1 Adding the Visual Component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/vc/romanager.css">
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<script type="text/javascript" src="/static/js/vc/addtoro.js"></script>
<script type="text/javascript">
    $(function () {
        $( '#r-aparent' ).addtoro( {
            btnClass: 'bg-white fg-ee-primary',
        } );
    });
</script>
```

- HTML:

```
<!-- component-related html -->
<div id="r-aparent"></div>
```

#### 3.2.2.5 Remove from RO button

The Remove From RO button allows a resource to be removed from a RO. The button, depicted in Figure 3-11, can be integrated into any interface to give to the user the opportunity to add elements to a RO.

**Figure 3-11 Remove From RO visual component**

##### 3.2.2.5.1 Adding the Visual Component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
```

```
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/vc/romanager.css">
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<script type="text/javascript" src="/static/js/vc/removefromro.js"></script>
<script type="text/javascript">
$(function () {
        $( '#r-rparent' ).addtoro( {
            btnClass: 'bg-white fg-ee-primary',
        } );
    });
</script>
```

- HTML:

```
<!-- component-related html -->
<div id="r-rparent"></div>
```

### 3.2.3 Virtual globe visual component

The Virtual Globe VC is based on Web World Wind[6] (WWW), a free, open-source 3D virtual globe using web technologies such as JavaScript. It was developed by NASA and ESA and allows interactive visualizations of geographic information on an interactive 3D globe or 2D map. It provides an API that enables JavaScript programs to control every detail of visualization and interaction, and runs on all major operating systems, desktop and mobile devices, and web browsers.

[6] https://webworldwind.org/

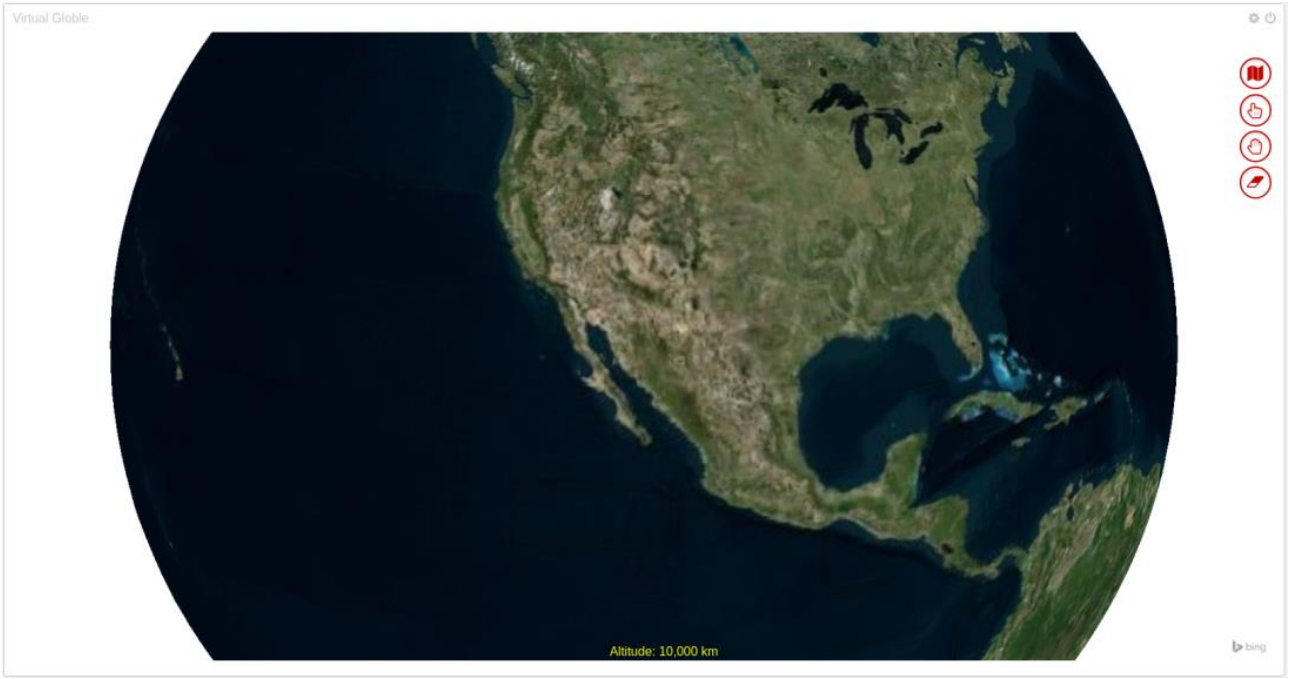



**Figure 3-12 Virtual Globe visual component**

The main features of the Virtual Globe are:

- Open-source, high-performance, 3D virtual globe for web pages and web applications and 2D Map mode with selectable and extensible map projections;
- JavaScript API for automating all aspects of interaction and visualization;
- Large collection of built-in high-resolution imagery and terrain;
- Displays high-resolution imagery, terrain and geographic data from any public or private source;
- Supports REST, WMS and Bing;
- Large collection of geometric and geographic shapes for representing information;
- Navigation and Viewing, Picking;
- Display multiple globes and maps on the same page;
- Simple to use, extend and modify;
- Graphics Capabilities: Placemark, Path and Curtain, Polygon and Extruded Polygon, Text, Terrain conforming shapes (Path, Polygon, Ellipse, Circle, Quadrilateral, Square), Imagery (JPEG, PNG), Graticules, and Shapefiles.

The VC encapsulates WWW and provides the following functionalities:

- Projection button: choose the Virtual Globe projection;
- Zoom Box button: used to zoom over an Area Of Interest (AOI);
- Select Area button: used to draw a rectangular AOI;
- Draw Area button: used to draw a polygon of interest;
- Pan button: allows moving on the Virtual Globe;
- Clean button: removes all the drew polygons from the Virtual Globe;
- Upload Shapefile button: allows Shapefile upload and visualization on the Virtual Globe;
- Data analytics (time series, subset, etc.);
- GIS toolbar (zoom, select area of interest, pan, etc.).

### 3.2.3.1 Adding the visual component

To integrate the visual component, add the following dependencies:

- CSS:

```
<!-- bootstrap -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/bootstrap.min.css">
<!-- plugins -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/font-awesome.min.css">
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/animate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/iconate.min.css"/>
<link rel="stylesheet" type="text/css" media="all" href="/static/css/lib/jquery-ui.min.css" />
<!-- project css -->
<link rel="stylesheet" type="text/css" media="all" href="/static/css/component-base.css">
<!-- component-related css -->
<link      rel="stylesheet"      type="text/css"      href="/static/libs/malihu-custom-scrollbar-plugin/jquery.mCustomScrollbar.css"/>
<link rel="stylesheet" type="text/css" href="/static/css/vglobe.css"/>
```

- JavaScript:

```
<!-- jquery & bootstrap -->
<script type="text/javascript" src="/static/js/lib/jquery.min.js"></script>
<script type="text/javascript" src="/static/js/lib/jquery-ui.min.js"></script>
<script type="text/javascript" src="/static/js/lib/bootstrap.min.js"></script>
<!-- plugins -->
<script src="/static/js/lib/jquery.nicescroll.min.js"></script>
<script src="/static/js/lib/jquery.slimscroll.min.js"></script>
<script src="/static/js/lib/jquery.fullscreen-min.js"></script>
<script src="/static/js/lib/iconate.min.js"></script>
<script src="/static/js/lib/bootstrap-notify.min.js"></script>
<!-- project javascript -->
<script src="/static/js/app/notify.js"></script>
<script src="/static/js/component-base.js"></script>
<!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
<!-- component-related javascript -->
<!-- Scrollbar effects -->
<script             type="text/javascript"             src="/static/libs/malihu-custom-scrollbar-plugin/jquery.mousewheel.js"></script>
<script             type="text/javascript"             src="/static/libs/malihu-custom-scrollbar-plugin/jquery.mCustomScrollbar.js"></script>
<!-- moduel js -->
<script type="text/javascript" src="/static/js/vglobe-main.js"></script>
<script type="text/javascript" src="/static/js/selection-names.js"></script>
<script type="text/javascript" src="/static/js/dock-bar.js"></script>
<script type="text/javascript" src="/static/js/main-dock.js"></script>
<script type="text/javascript" src="/static/js/dock-panel.js"></script>
<script type="text/javascript" src="/static/js/select-panel.js"></script>
<script type="text/javascript" src="/static/js/gis-toolbar.js"></script>
<script type="text/javascript" src="/static/js/coordinate-overlay.js"></script>
<!-- WWW -->
<script type="text/javascript" src="/static/www/worldwindlib.js"></script>
<script type="text/javascript" src="/static/www/LayerManager.js"></script>
<script type="text/javascript" src="/static/www/coordinateController.js"></script>
<script type="text/javascript" src="/static/www/www.js"></script>
```

```
<script type="text/javascript">
$( document ).ready(function() {
        init();
    });
</script>
```

- HTML:

```
<!-- component-related html -->
<div class="col-md- col-sm-12 col-xs-12 padding-0">
<div class="col-md-12 col-sm-12 col-xs-12 padding-0">
<div class="col-md-12 col-sm-12 col-xs-12">
<div class="panel  ee-box">
<div class="panel-heading bg-white border-none">
<div class="col-md-6 col-sm-6 col-xs-6 text-left padding-0">
<h4 class="text-left">Virtual Globle</h4>
</div>
<div class="col-md-6 col-sm-6 col-xs-6 text-right padding-0">
<h4>
<span class="fa fa-gear box-title-icon-func text-right"></span>
<span class="fa fa-power-off box-title-icon-func text-right"></span>
</h4>
</div>
</div>
<div class="panel-body ">
<div id="service-container">
<canvas id="canvasOne"></canvas>
</div>
</div>
</div>
</div>
</div>
</div>
```

### 3.2.4 Upload to Seafile visual component

The Seafile visual component provides the EVER-EST users with the functionality to upload resources in their private space. The visual component is described in the Technical Note on Common Services document [EVER-EST DEL WP5-D5.2].

## 3.3 Using the service – quick overview

The following paragraphs describe some example of interactions with the Visual Components previously introduced. It will be assumed that the user is logged into the EVER-EST system.

### 3.3.1 Performing discovery search (one-step search) on virtual globe

This paragraph explains how to do a one-step EO Data Discovery search with results' polygons visualization on the Virtual Globe and data download on the VRC portal. To perform a one-step Discovery Search on Virtual Globe:

- Select the "Advanced Search" button, then choose the "Sentinel 1A" data set in the Advanced Search Panel;
- Select the "Search" button. The Results Panel is shown and the polygons of the products are drawn on the Virtual Globe;

- Choose the first element from the Results Panel and select the “Download to VRC” button. A message that informs the user that the download to the VRC portal has started is shown;
- The user will be notified by email when the download is finished.

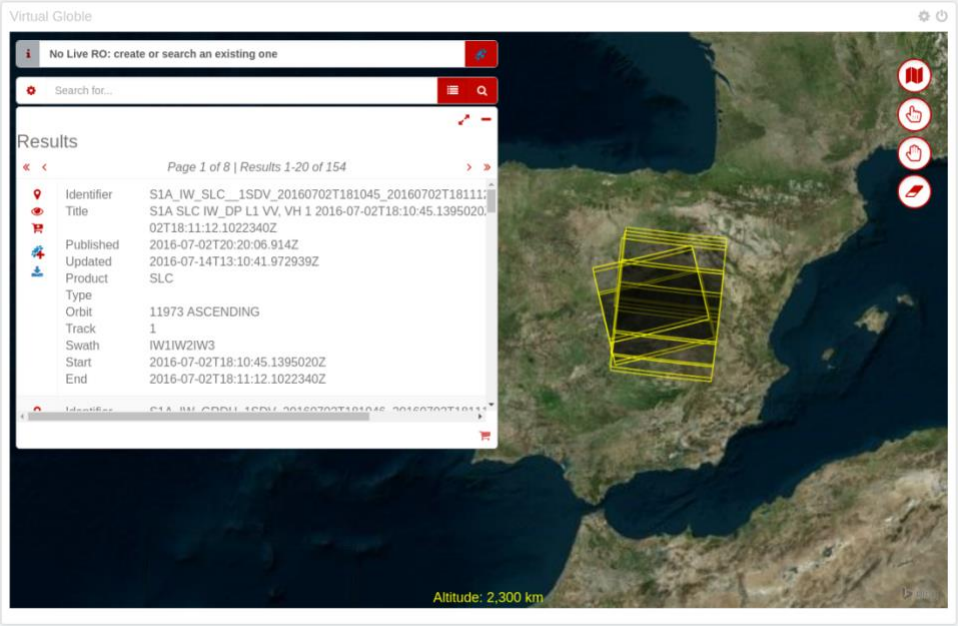



**Figure 3-13 One-step EO data discovery**

### 3.3.2 Performing discovery search (two-step search) on virtual globe

This paragraph explains how to do a two-step EO Data Discovery search with results’ polygons visualization on the Virtual Globe and data download on the VRC portal.

To perform a two-step Discovery Search on Virtual Globe:

- Select the “Advanced Search” button, then choose the “FedEO” data set and the “ESA:SMOS” Parent Identifier in the Advanced Search Panel;
- Select the “Search” button. The Results Panel is shown and the polygons of the collections (world polygons) are drawn on the Virtual Globe;
- To better view the world polygons, change the projection to “Equirectangular” using the first button on the right;
- Choose the first collection from the Results Panel. The product list is shown in the Results Panel and the polygons are drawn on the Virtual Globe;
- Use the “Add To Cart” button on the first two elements, then select the “Show Cart” button and confirm. A message informing the user that the download has started is shown;
- The user will be notified by email when the download is finished.

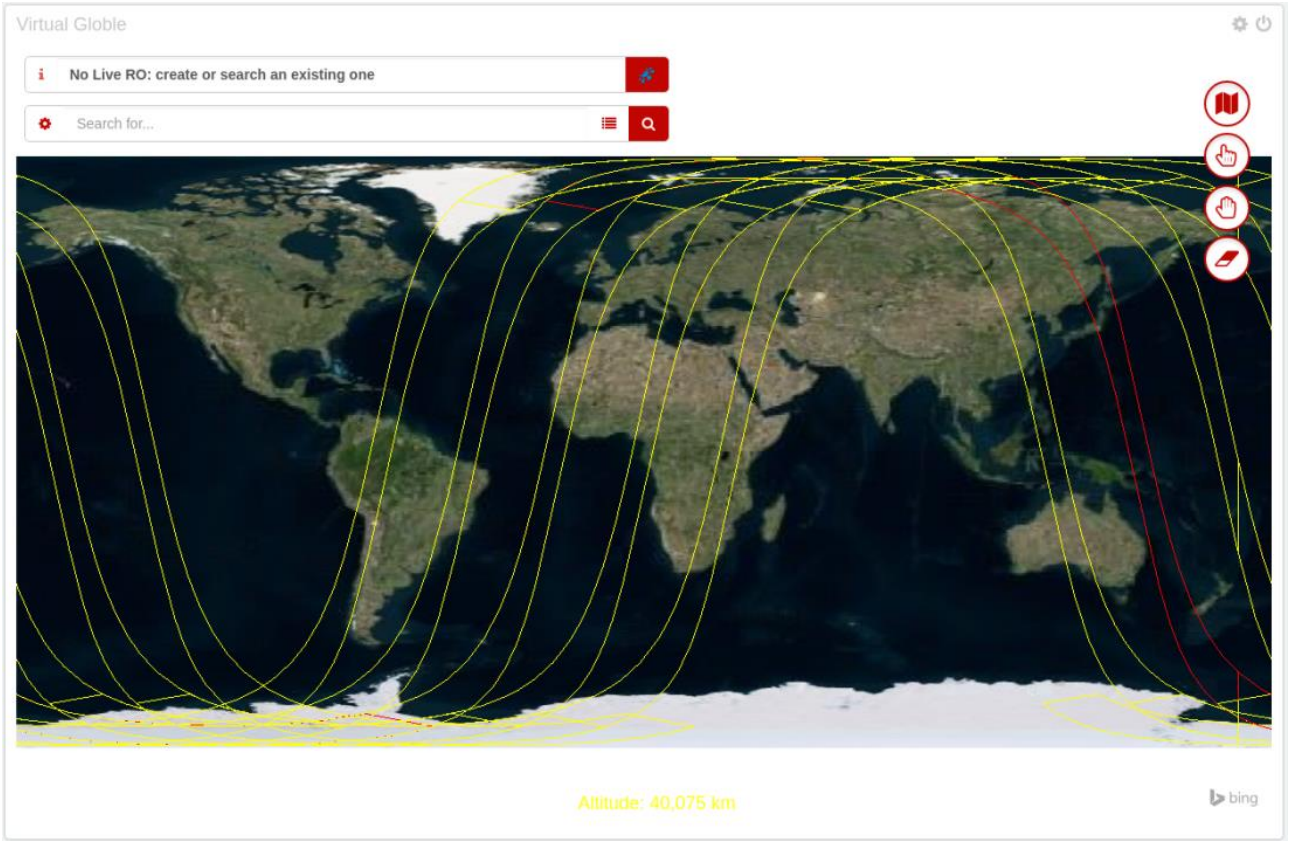



**Figure 3-14 Two-steps EO data discovery**

### 3.3.3 Performing research object creation

This paragraph explains how to create a research object within the VRC environment.

Select the “Create RO” button and follow the wizard filling the fields in the following way:

- Step 1: RO Data
  - Identifier: New RO;
  - Title: My New Test RO;
  - Template: A workflow-centric RO;
  - Description: Test Research Object;
- Step 2: RO Initial Content
  - Select “Upload Documentation” and upload an empty text file;
- Step3: select the “Create” button. The RO is created and the RO Manager Status Bar is updated with the title of the Research Object.

### 3.3.4 Performing research object visualization

This paragraph explains how to view and modify the content of a RO within the VRC environment.

Select a RO or create a new one:

- Check that the RO Manager status bar contains the title of the research object;
- Select the “RO Details” button and check the information contained into the Information, Content, and Progress tabs;
- Go to the Content tab, select the Add To RO button and upload an empty text file.

The visual component is under development and will be better described in next release of this document.

### 3.3.5 Performing upload to Seafile

The Visual Component provides the EVER-EST users with the functionality to upload resources in their private space. The VC wraps the Seafile APIs described in [EVER-EST DEL WP5-D5.2].

## 3.4 Service public API's

### 3.4.1 RO manager VC APIs

To communicate with the RO backend services, the RO Manager uses the RO Middleware APIs developed by ACS. The RO Middleware is an abstraction layer built on top of the RO APIs (exposed in ROHub) that merges multiple operations of the RO APIs into a single RO Middleware API tailored to the VRCs needs. This set of APIs is under development and they will be better described in the final release of this document.

#### 3.4.1.1 Create RO

The API allows creating a RO with a given name, title, and description. Optionally internal and/or external resources can be added and public/private visibility mode can be set. If no visibility mode is specified, READ-ONLY will be used as default value.

##### 3.4.1.1.1 Request

**HTTP Request Header:**

```
POST /everest/ROs HTTP/1.1
Host: vm2.everest.psnc.pl:8280
Authorization: Bearer <access_token>
Content-Type: application/json
Slug: <ro_name>
Content-Length: ….

<request body>
```

<access_token>is the access token returned by the EVER-EST Identity Server (IS)[7].

<ro_name> is the name of the RO to be created. The name can be changed by the RO backend service while the RO is being created. The updated RO name must be read from the ro_name field of the response JSONbody.

**HTTP Request Body:**

```
{
    "ro_title": "title of the ro to be created",
    "ro_desc": "description of the ro to be created",
    "ro_vrc": "Virtual Research Environment of the RO creator",
    "internal": [
```

[7] The details about the EVER-EST Idetity Server component can be found in D5.2, EVER-EST DEL WP5-D5.2.

```
            {"filename of an internal resource": "base64-encoded file content"},
            …
        ],
        "external": [
            "uri of an external resource",
            …
        ],
        "mode": "visibility"
    }
```

**Body parameter description:**

| Parameter | Required | Min Cardinality | Max Cardinality |
|---|---|---|---|
| ro_title | Required | 1 | 1 |
| ro_desc | Required | 1 | 1 |
| ro_vrc | Required | 1 | 1 |
| internal | Optional | 0 | 1 |
| external | Optional | 0 | 1 |
| Mode | Optional | 0 | 1 |

**Table 3-1 Create RO API – Request body parameter description**

#### 3.4.1.1.2 Response

**HTTP Response Header:**

```
HTTP/1.1 <http_status_code><http_status_message>
Content-Type: application/json; charset=UTF-8
Date: <date>
Content-Length: …

<response body>
```

`<http_status_code>` and `<http_status_message>`can have the following values:

| Status | Response | Description |
|---|---|---|
| 201 | Created | RO Successfully Created |
| 4xx | | Error during RO creation |

**Table 3-2 Create RO API – Response HTTP codes and descriptions**

**HTTP Response Body:**

```
{
    "ro_name": "absolute uri of the created RO"
}
```

**Body parameter description:**

| Parameter | Required | Min Cardinality | Max Cardinality |
|---|---|---|---|
| ro_name | Required | 1 | 1 |

**Table 3-3 Create RO API – Response Body Parameter Description**

#### 3.4.1.1.3 Example

**Request:**

```
POST /everest/ROs HTTP/1.1
Host: vm2.everest.psnc.pl:8280
User-Agent: curl/7.49.1
Accept: */*
Authorization: Bearer 7d6df5f8-6b5c-4d88-95e1-da101dcee6f8
Content-Type: application/json
Slug: cde
Content-Length: 243

{"ro_title": "The Title", "ro_desc": "The Description", "ro_vrc": "CNR",
"internal": [{"myfolder/file1.txt": "MTIz"}, {"myfolder/file2.txt":
"NDU2"}],"external": ["http://sdeg.com/resource1.html", "https://sdeg.it/res2.do"],
"mode": "PRIVATE"}
```

**Response**

```
HTTP/1.1 201 Created
Content-Type: application/json; charset=UTF-8
Date: Wed, 16 Nov 2016 11:14:57 GMT
Content-Length: 64

{"ro_name":"http://sandbox.wf4ever-project.org/rodl/ROs/cde-9/"}
```

### 3.4.1.2 Get List of ROs

The API allows the retrieval of ROs matching optional criteria, such as RO name, title, description, author's VRC, author's username, creation date and RO Status. If not specified otherwise, results are returned sorted by title, creation date, author's username, RO name, description, author's VRC and RO status.

#### 3.4.1.2.1 Request

**HTTP Request Header:**

```
POST /everest/sparql HTTP/1.1
Host: vm2.everest.psnc.pl:8280
Accept: application/sparql-results+json
Content-Type: application/json
```

```
Content-Length: ….

<request body>
```

**HTTP Request Body:**

```
{
"entityType": "RO",
"vrc": "excerpt of the Virtual Research Environment of the RO creator",
"since_YYYY-MM-DD": "lower creation date",
"till_YYYY-MM-DD": " upper creation date ",
"title": "excerpt of the RO title",
"description": "excerpt of the RO description",
"creator": "excerpt of the RO-creator username",
"status": "RO Status",
"ro": "excerpt of the RO name",
"orderBy": "list of fields"
}
```

**Body parameter description:**

| Parameter | Required | Min Cardinality | Max Cardinality | Format | Values |
|---|---|---|---|---|---|
| entityType | Required | 0 | 1 | Case sensitive | |
| vrc | Optional | 0 | 1 | Case insensitive | |
| since_YYYY-MM-DD | Optional | 0 | 1 | YYYY-MM-DD (inclusive) | |
| till_YYYY-MM-DD | Optional | 0 | 1 | YYYY-MM-DD (inclusive) | |
| title | Optional | 0 | 1 | Case insensitive | |
| Description | Optional | 0 | 1 | Case insensitive | |
| creator | Optional | 0 | 1 | Case insensitive | |
| status | Optional | 0 | 1 | Case sensitive | LiveRO<br>SnapshotRO<br>ArchivedRO |
| ro | Optional | 0 | 1 | Case insensitive | |
| orderBy | Optional | 0 | 1 | Case sensitive | |

**Table 3-4 Get list of ROs API –Request body parameter description**

#### 3.4.1.2.2 Response

**HTTP Response Header:**

```
HTTP/1.1 <http_status_code><http_status_message>
Content-disposition: attachment; filename="result.json"
```

```
Content-Type: application/sparql-results+json
Date: <date>
Transfer-Encoding: chunked

<response body>
```

<http_status_code> and <http_status_message> can have the following values:

| Status | Response | Description |
|---|---|---|
| 200 | OK | RO list retrieved |
| 4xx | | Error during RO search |

**Table 3-5 Get list of ROs API - Response HTTP codes and descriptions**

**HTTP Response Body:**

```
{
"head": {
"vars": [ "ro" , "created" , "title" , "description" , "vrc" , "creatorName" ,
"creator" , "status" ]
} ,
  "results": {
  "bindings": [
{
"ro": { "type": "uri" , "value": "RO name" } ,
"created": { "datatype": "http://www.w3.org/2001/XMLSchema#dateTime" , "type":
"typed-literal" , "value": "RO creation datetime" } ,
"title": { "datatype": "http://www.w3.org/2001/XMLSchema#string" , "type":
"typed-literal" , "value": "RO title" } ,
"description": { "datatype": "http://www.w3.org/2001/XMLSchema#string" ,
"type": "typed-literal" , "value": "RO description" } ,
"vrc": { "datatype": "http://www.w3.org/2001/XMLSchema#string" , "type":
"typed-literal" , "value": "Virtual Research Environment the RO creator belongs
to" } ,
"creator": { "type": "uri" , "value": "RO creator" } ,
"status": { "type": "literal" , "value": "RO status" }
}
]
}
}
```

The response body will contain as many “bindings” elements as there are results found.

**Body parameter description:**

| Parameter | Required | Min Cardinality | Max Cardinality |
|---|---|---|---|
| ro | Optional | 0 | 1 |
| created | Optional | 0 | 1 |
| title | Optional | 0 | 1 |
| description | Optional | 0 | 1 |
| vrc | Optional | 0 | 1 |
| creator | Optional | 0 | 1 |
| status | Optional | 0 | 1 |

**Table 3-6 Get list of ROs API – Responsebody parameter description**

#### 3.4.1.2.3 Example

**Request:**

```
POST /everest/sparql HTTP/1.1
Host: vm2.everest.psnc.pl:8280
User-Agent: curl/7.49.1
Accept: application/sparql-results+json
Content-Type: application/json
Content-Length: 198

{"entityType": "RO", "vrc": "CNR", "since_YYYY-MM-DD": "2016-10-16",
"till_YYYY-MM-DD": "2016-11-17", "title": "The ", "description": "This",
"creator": "everest", "status": "LiveRO", "ro": "first"}
```

**Response**

```
HTTP/1.1 200 OK
Content-disposition: attachment; filename="result.json"
Content-Type: application/sparql-results+json
Date: Fri, 18 Nov 2016 14:39:59 GMT
Transfer-Encoding: chunked

{
"head": {
"vars": [ "ro" , "created" , "title" , "description" , "vrc" , "creatorName" ,
"creator" , "status" ]
} ,
"results": {
"bindings": [
{
"ro": { "type": "uri" , "value": "http://sandbox.wf4ever-
project.org/rodl/ROs/The%20First%20RO-2/" } ,
```

```
      "created": { "datatype": "http://www.w3.org/2001/XMLSchema#dateTime" , "type":
"typed-literal" , "value": "2016-11-17T16:10:59.609Z" } ,
      "title": { "datatype": "http://www.w3.org/2001/XMLSchema#string" , "type":
"typed-literal" , "value": "The First RO from the VRC" } ,
      "description": { "datatype": "http://www.w3.org/2001/XMLSchema#string" ,
"type": "typed-literal" , "value": "This is a TEST RO" } ,
      "vrc": { "datatype": "http://www.w3.org/2001/XMLSchema#string" , "type":
"typed-literal" , "value": "CNR" } ,
      "creator": { "type": "uri" , "value":
"http://sandbox.wf4ever.project.com/users/everest_user" } ,
      "status": { "type": "literal" , "value": "LiveRO" }
      }
      ]
      }
      }
```


### 3.4.2 Virtual globe VC APIs

Web World Wind provides an API that enables JavaScript programs to control every detail of visualization and interaction. Web World Wind runs on all major operating systems, desktop and mobile devices, and web browsers.

Web World Wind provides a rich set of features for displaying and interacting with geographic information. The API and functionality are described in detail on WWW's site[8].

[8]http://worldwindserver.net/webworldwind/api-doc/index.html

# 4 E-Collaboration Services

## 4.1 Service overview

The E-Collaboration Services component facilitates the interaction, communication and collaboration between VRC users. Django, the Content Management System (CMS) on which the EVER-EST Portal and the VRC Portals are based, allows the integration of most of the e-collaboration capabilities through third party plugins. Other e-collaboration services have been integrated as JavaScript extensions.

Based on Users Stories, an initial set of E-Collaboration functionalities has been integrated into EVER-EST and VRC Portals:

- Forum;
- 1-to1 and 1-to-N Instant Messaging (IM) through the Extensible Messaging and Presence Protocol (XMPP)[9] standard;
- Internal Messaging and Notification;
- Calendar/Scheduling.

### 4.1.1 Forum overview

The Forum capabilities are provided via the django-machina plugin. Django-machina[10] is a forum framework for Django providing a way to build community-driven websites. It offers a full-featured, yet very extensible forum solution, providing:

- Topic and post editing;
- Forums tree management;
- Per-forum permissions;
- Polls and attachments;
- Moderation and pre-moderation;
- Forum conversations search.

Only VRC administrators can perform forum creation, while all VRC members can manage topics and discussions.

### 4.1.2 Instant messaging overview

The Instant Messaging (IM) service is based upon the Extensible Messaging and Presence Protocol (XMPP)[11] standard, a set of open technologies for instant messaging, presence, multi-party chat, voice and video calls, collaboration, lightweight middleware, content syndication, and generalized routing of XML data. XMPP was originally developed in the Jabber open-source community to provide an open, decentralized alternative to the closed instant messaging services at that time. XMPP offers several key advantages over other such services:

- Open: the XMPP protocols are free, open, public and easily understandable; in addition, multiple implementations exist in the form clients, servers, server components and code libraries;
- Standard: the Internet Engineering Task Force (IETF) has formalized the core XML streaming protocols as an approved instant messaging and presence technology. The XMPP specifications were published as RFC 3920 and RFC 3921 in 2004, and the XMPP Standards Foundation continues to publish many XMPP Extension Protocols. In 2011 the core RFCs were revised, resulting in the most up-to-date specifications (RFC 6120, RFC 6121, and RFC 6122);

[9] http://xmpp.org/

[10] https://django-machina.readthedocs.io/en/stable/

[11] http://xmpp.org/

- Proven: the first Jabber/XMPP technologies were developed by Jeremie Miller in 1998 and are now quite stable; hundreds of developers are working on these technologies, there are tens of thousands of XMPP servers running on the Internet today, and millions of people use XMPP for instant messaging through public services and XMPP deployments at organizations worldwide;
- Decentralized: the architecture of the XMPP network is similar to email; as a result, anyone can run their own XMPP server, enabling individuals and organizations to take control of their communications experience;
- Secure: any XMPP server may be isolated from the public network (e.g., on a company intranet) and robust security using SASL and TLS has been built into the core XMPP specifications. In addition, the XMPP developer community is actively working on end-to-end encryption to raise the security bar even further;
- Extensible: using the power of XML, anyone can build custom functionality on top of the core protocols; to maintain interoperability, common extensions are published in the XEP series, but such publication is not required and organizations can maintain their own private extensions if so desired;
- Flexible: XMPP applications, beyond IM, include network management, content syndication, collaboration tools, file sharing, gaming, remote systems monitoring, web services, lightweight middleware, cloud computing, and much more;
- Diverse: a wide range of companies and open-source projects use XMPP to build and deploy real-time applications and services; you will never get “locked in” when you use XMPP technologies.

The client side of the Instant Messaging service is provided by the JSXC (JavaScript XMPP Client) project[12] while server side of the service is provided by the eJabberd[13] project.

JSXC adds real-time XMPP chat to any web application. The main features are:

- Full-featured XMPP client in the browser, no client installation required;
- Easy integration into existing web pages or applications;
- Interaction with web pages e.g. making Jabber IDs active;
- Uninterrupted browsing among JSXC-enhanced pages;
- One-to-one chats, optionally with end-to-end encryption (OTR) and user verification (SMP);
- End-to-end encrypted audio and video calls from Firefox and Chrome without plugin;
- Multi-user chat (MUC) rooms for text messages;
- A growing number of international languages supported;
- Event notifications with sound if browser/tab inactive (not yet supported in Internet Explorer).

The eJabberd Server is a solid, scalable and extensible XMPP open-source server. What follows is a list of the main important features:

- Modern: ejabberd supports the latest refinements of XMPP to build modern and up to date services on mobile and Web;
- Fault-Tolerant: ejabberd can run in a cluster out of the box along with the ability to upgrade the server while it is running;
- Scalable: ejabberd was designed from the outset to be scalable. Being clusterable, the code base is constantly optimized to handle more ‘load per node’;
- Manageable: ejabberd comes with a powerful command-line tool that allows controlling of most of the aspects of the server. Most common operations can be automated;
- Highly Versatile: ejabberd offers many customization hooks to adapt it easily to various use cases: mobile messaging, gaming, internet of things;

[12] https://www.jsxc.org/

[13] https://www.ejabberd.im/

- Modular: ejabberd code is extremely modular and can be extended through a powerful API.

### 4.1.3 Internal messaging and notification overview

The Internal Messaging and Notification capabilities are provided via the django-postman plugin. Django-postman[14]allows authenticated users of a site to exchange private messages within the site. Each user has access to a collection of messages stored in folders:

- Inbox for incoming messages;
- Sent for sent messages;
- Archives for archived messages;
- Trash for messages marked as deleted.

In folders, messages can be presented in two modes:

- by conversation, for a compact view: the original message and its replies are grouped in a set to constitute one sole entry. The latest message (based on the time) is the representative of the set;
- by message, for an expanded view: each message is considered by itself.

Here is a summary of features:

- Exchanges can be moderated (with auto-accept and auto-reject plug-ins);
- Optional recipient filter plug-ins;
- Optional exchange filtering plug-ins (blacklists);
- Multi-recipient writing is possible (can be disabled by configuration) with min/max constraints;
- Messages are managed by conversations;
- Messages in folders are sortable by sender, recipient, subject or date;
- A Quick-Reply form to ask for a response text only.

### 4.1.4 Calendar/scheduling overview

The Calendar/Scheduling capabilities are provided via the django-todo plugin. Django-todo[15]is a multi-user, multi-group task management and assignment application for Django. It includes drag and drop task prioritization and email task notification. It can serve as anything from a personal to-do system to a complete, working ticketing system for organizations/publications/companies.

The assumption is that the organization/publication/company has multiple groups of people, each with multiple users. Users may belong to multiple groups, and each group can have multiple to-do lists. Users can view and modify all to-do lists belonging to their group(s). Privileged users can delete lists. Identical list names can exist in different groups, but not in the same group. Emails are generated to the assigned-to person when new tasks are created. Comment threads can be added to tasks. Each participant in a thread receives email when new comments are added.

[14] http://django-postman.readthedocs.io/en/stable/

[15] https://github.com/shacker/django-todo

## 4.2 Service visual components

### 4.2.1 Forum

The Forum button, placed in the Core Functions Area in the Header Bar of the EVER-EST portals, allows the user to access the forum capabilities:

**Figure 4-1 Forum icon**

When a user selects the Forum button, the index page is shown:

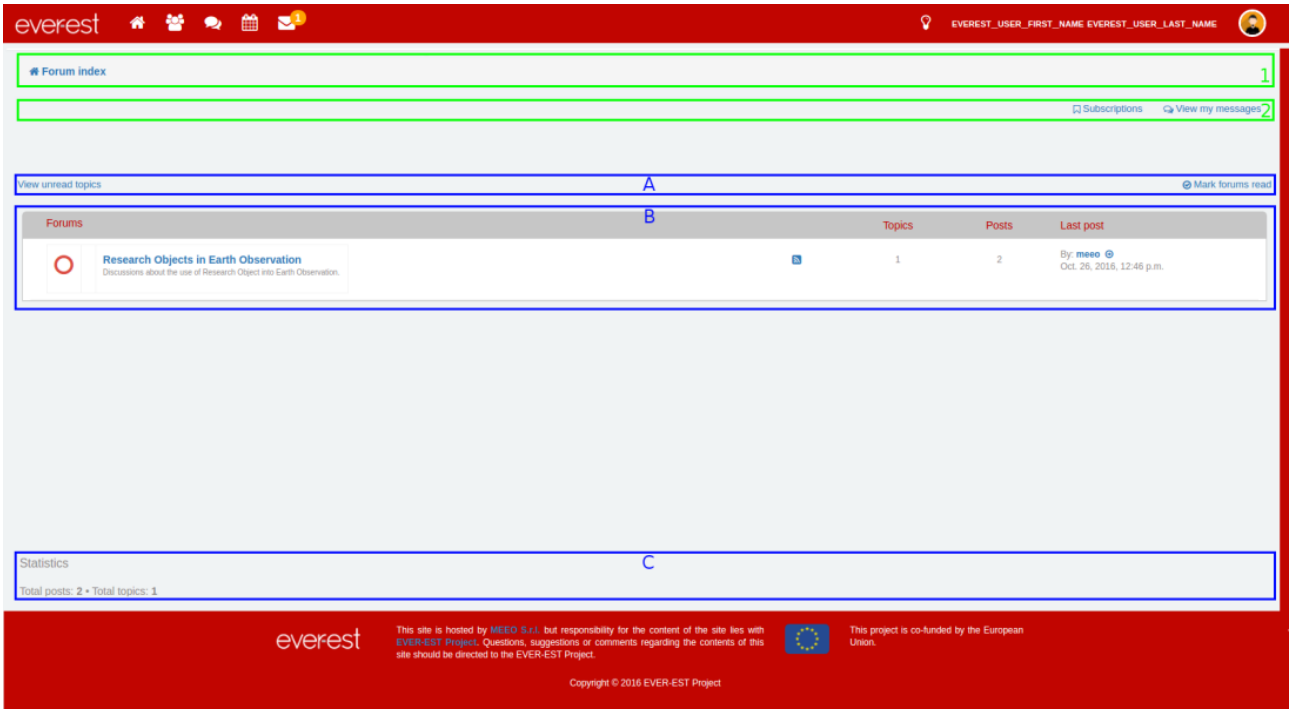



**Figure 4-2 Forum home page**

Each forum page is composed of a static section that is always visible and a variable section that changes based on user position inside the Forum.

The static section contains two main elements:

- Navigation Bar (section 1 in Figure 4-2): shows the user position inside the Forum;
- Common User Actions Bar: (section 2 in Figure 4-2) contains a few links to some user-related functions, such as show user Subscriptions, show user Messages, and show Moderation queue.

When the Forum button is selected, the variable section shows the list of available forums and contains the following elements:

- Forum Actions Bar (section A in Figure 4-2) containing the buttons to mark all the forums as ‘read’ and to view only unread topics;
- Forum List (section B in Figure 4-2)containing the list of forums;

- Forum Statistics Bar (section C in Figure 4-2) containing the number of Post and Topics.

When a forum is selected, the variable section shows the list of topics of the forum and contains the following elements (shown in Figure 4-3):

- Forum Title;
- Topic Action Bar containing the buttons to mark all the topics of the forum as 'read' and to create a new topic (with the possibility to attach files and to create a poll);
- Topic List containing the list of forum topics.

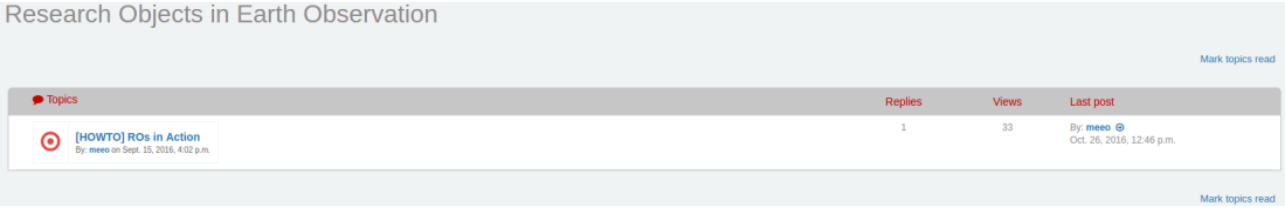



**Figure 4-3 Topics list**

When a topic is selected, the variable section displays the list of posts relevant to the topic and contains the following elements (see Figure 4-4):

- Topic Title;
- Post Action Bar containing the buttons to post a reply (with the possibility to attach files), subscribe to the topic and moderate the topic;
- Post List containing the list of posts.

Each topic can be edited by the owner or removed.

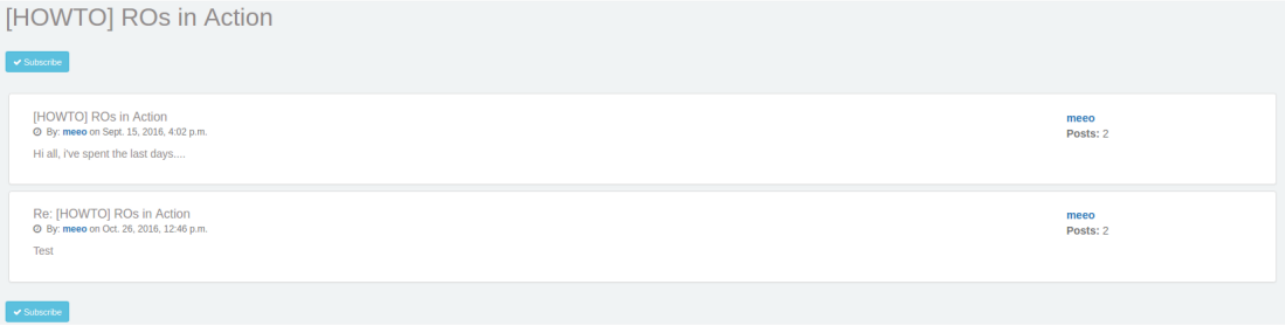



**Figure 4-4 Topic**

### 4.2.2 Instant messaging

The Instant Messaging button, placed in the Core Functions Area in the Header Bar of the EVER-EST portals, allows the user to access to the IM capabilities:

**Figure 4-5 Instant messaging icon**

When the IM button/bar is selected, the IM bar on the right expands.

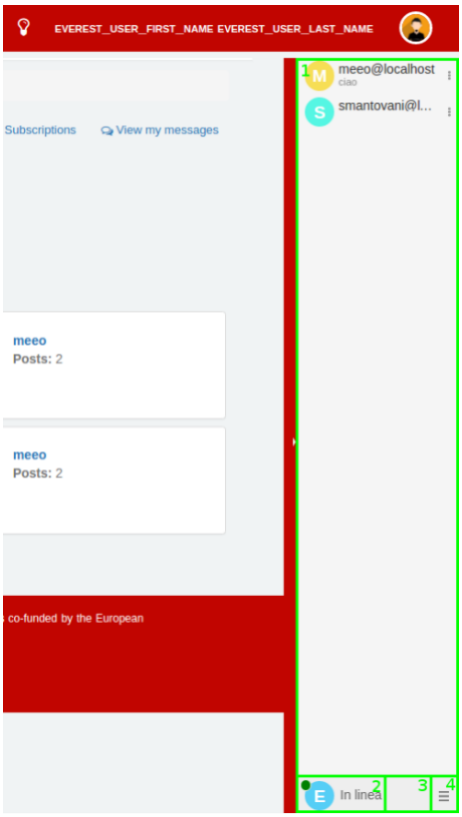



**Figure 4-6 Instant messaging bar**

The IM bar has four main sections:

- Contact List (section 1 in Figure 4-6);
- User Info (section 2 in Figure 4-6): allows the user to change the on-line status;
- Notification (section 3 in Figure 4-6): shows users' notifications;
- Settings (section 4 in Figure 4-6): allow the user to manage the chat settings.

To chat to another user, simply click on a user in the contact list.

### 4.2.3 Internal mailing and notification

The Internal Mailing button, placed in the Core Functions Area in the Header Bar of the EVER-EST portals, allows the user access to the mailing capabilities:

**Figure 4-7 Internal mailing icon**

The main page of the mailing service shows the Inbox, but allows the user to:

- Navigate through Sent Messages, Archives, and Trash folders;
- Select a message to Delete/Archive;
- Change the message view from Conversation (default) to Message;

- Write a new message.

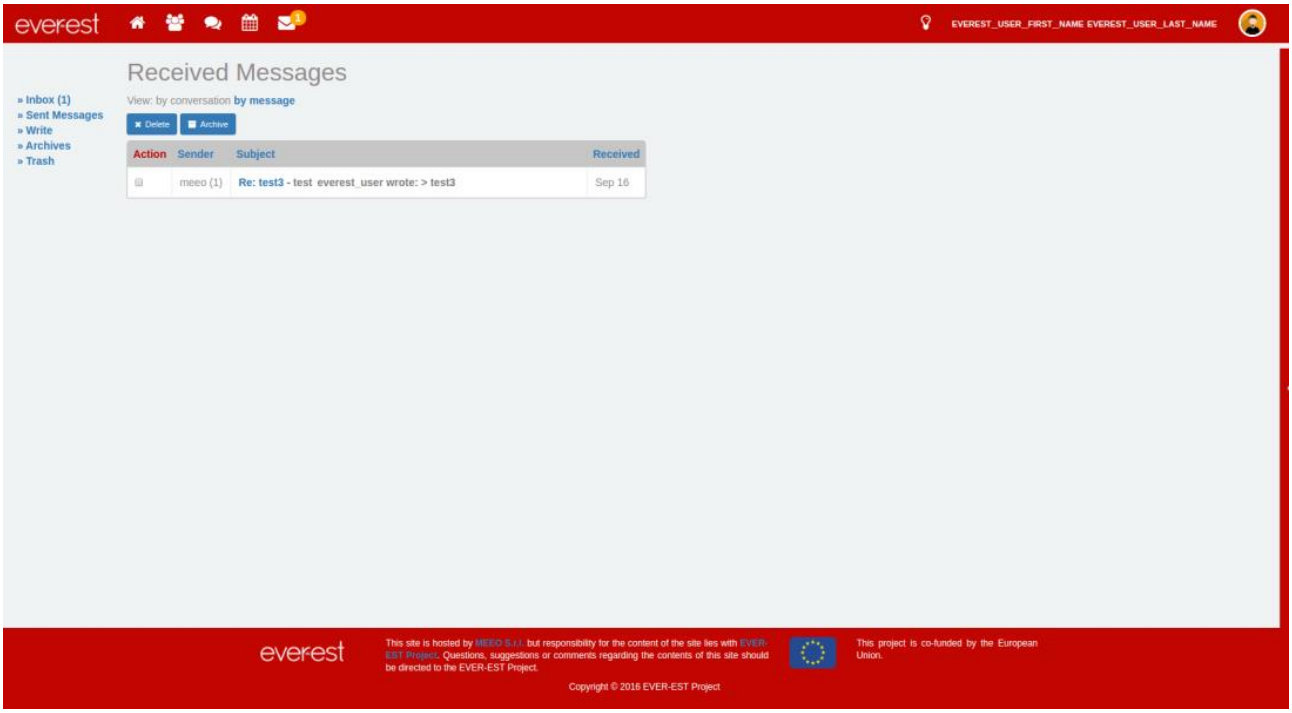



**Figure 4-8 Inbox**

When a message is selected, the interface shows the message and allows the user to Delete, Archive, or Reply to the message.

If one button between the Write and Reply buttons is selected, the interface is changed and allows the user to write a new e-mail to reply to another mail. The interface is very similar to the one used in all the other mailing services.

### 4.2.4 Calendar/scheduling

The Calendar/Scheduling button, placed in the Core Functions Area in the Header Bar of the EVER-EST portals, allows the user access to the Calendar/Scheduling capabilities:

- Task Lists: shows the task lists and allows task and group task creation;
- My Tasks: show the tasks assigned to the user.

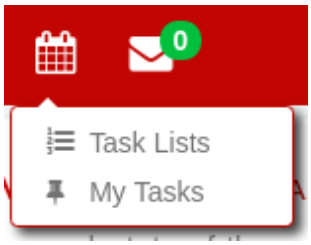



**Figure 4-9 Calendar/scheduling icon and menu**

#### 4.2.4.1 Task Lists

The Task Lists interface allows the user to see Task Lists and to create a new Task List.

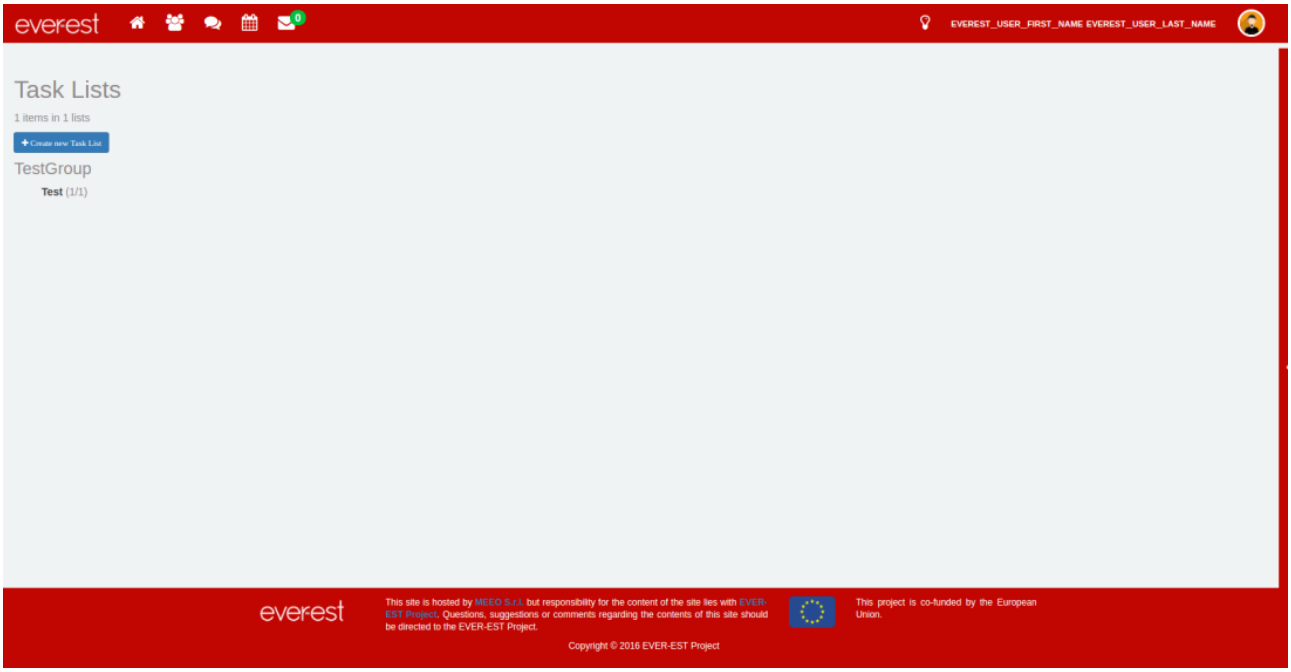



**Figure 4-10 Task list**

The “Create Task List” button allows the user to create a new List.

When a Task List is selected, the list of tasks is shown and it is possible to:

- Delete the List;
- Add a new Task;
- View Tasks Details;
- Mark Tasks as Done;
- See the Completed Task List.

#### 4.2.4.2 My Tasks

The Tasks interface allows the user to navigate, view and edit the Tasks assigned to the user.

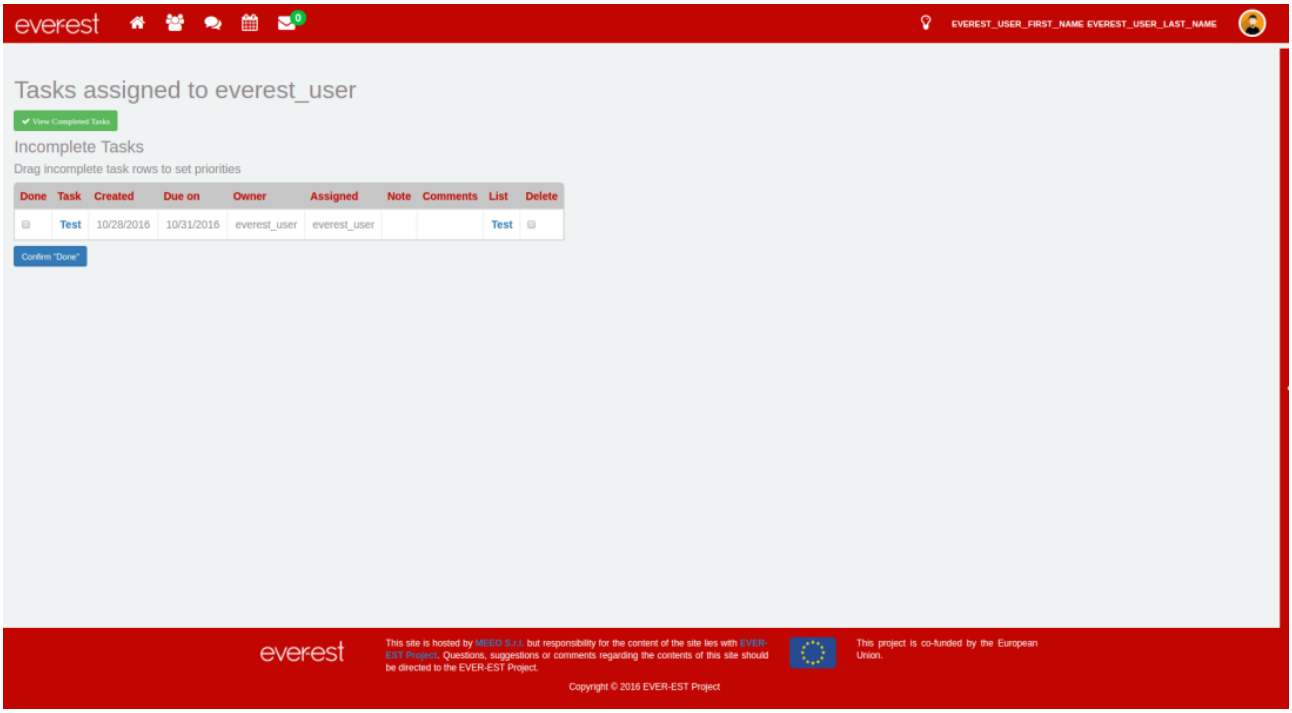



**Figure 4-11 My tasks**

# Annex A. Installation Guide

## A.1. EVER-EST Portal installation guide

### A.1.1. Software prerequisites

Minimum software requirements for the EVER-EST Portal:

- OS: CentOS Linux 7.2-1511 (Core);
- Apache 2.4.6 with mod_wsgi;
- PostgreSQL 9.2.15;
- Python 2.7.5 with Python Virtualenv (v. 1.10.1) and PIP (v. 8.1.2).

### A.1.2. Hardware prerequisites

Minimum hardware requirements for the EVER-EST portal:

- CPU: 2 core CPU x86_64;
- RAM: 8 GB;
- Disk Storage: 20 GB.

### A.1.3. Service installation

#### A.1.3.1. System Configuration

The following configuration steps should be executed as “root” user.

| Step | Command |
|---|---|
| **Create “everestadmin” user** | useradd –p ev3R@dm1nE$t everestadmin |
| **Add “everestadmin” user to sudoers group** | visudo |
| **Update Repository** | yum update |
| **Install Required Software** | yum install screen python-virtualenv epel-release httpd mod_wsgi svn net-tools postgresql.x86_64 postgresql-devel.x86_64 postgresql-server.x86_64 postgresql-libs.x86_64 gcc wget unzip |
| **Install Python PIP** | yum install python-pip |
| **Deactivate SElinux: edit the selinux config file and set the “SELINUX” property to“disabled”** | vim /etc/selinux/config |
| **Deactivate SElinux: reboot the system** | reboot |

**Table 4-1 EVER-EST portal system configuration**

#### A.1.3.2. Virtual Environment Configuration

The following configuration steps should be executed as “root” user.

| Step | Command |
|---|---|
| **Create Virtual Environment directory** | mkdir /venv/ |
| **Change directory ownership** | chown everestadmin: -R /venv/ |

**Table 4-2 EVER-EST portal virtual environment configuration**

### A.1.3.3. Source Folder Configuration

The following configuration steps should be executed as “everestadmin” user.

| Step | Command |
|---|---|
| **Sourcesfolder creation** | mkdir ~/svn && cd ~/svn/ |
| **Update Sources** | svn co --username everestadmin https://svn.services.meeo.it/EVER-EST/trunk/project/software/ . |

**Table 4-3 EVER-EST portal source folder configuration**

The following configuration steps should be executed as “root” user.

| Step | Command |
|---|---|
| **Go to OPT directory** | cd /opt/ |
| **Create soft link to sources folder** | ln -s /home/everestadmin/svn/ever-est/ |
| **Go to Apache Configuration folder** | cd /etc/httpd/conf.d/ |
| **Create soft link to EVER-EST Apache Configuration file** | ln -s /opt/ever-est/ever-est_apache.conf |
| **Change configuration file permissions** | chmod 775 /opt/ever-est/ever-est_apache.conf |
| **Edit the Apache Main Configuration File** | vim /etc/httpd/conf/httpd.conf |
| | Change “user apache” to “user everestadmin” |
| | Change “user group” to “user everestadmin” |
| | Change “ServerAdmin root@localhost” to “helpdesk@meeo.it” |
| **Restart Apache** | systemctl restart httpd.service |

**Table 4-4 EVER-EST portal source folder configuration**

### A.1.3.4. EVER-EST portal configuration

The following configuration steps should be executed as “everestadmin” user.

| | Step | Command |
|---|---|---|
| **Database Initialization** | Initialize PostgreSQL | sudo postgresql-setup initdb |
| | Change all ‘ident’ in the methods column to ‘md5’ | sudo vim /var/lib/pgsql/data/pg_hba.conf |
| | Restart PostgreSQL | sudo systemctl restart postgresql |
| | Start PostgreSQL client | sudo -u postgres psql |
| | Create “ever-est” database | CREATE DATABASE “ever-est” OWNER postgres; |
| | Create “everestadmin” user with “ev3R@dm1nE$t” password | CREATE USER everestadmin WITH SUPERUSER INHERIT CREATEROLE CREATEDB PASSWORD 'ev3R@dm1nE$t'; |
| | Exit PostgreSQL client | \q |
| **Virtual Environment** | Virtual Environment creation | cd /venv/&&virtualenv ever-est |

| | | |
|---|---|---|
| **Initialization** | Virtual Environment activartion | source /venv/ever-est/bin/activate |
| | Upgrade PIP | pip install --upgrade pip |
| | PIP: install EVER-EST required packages | pip install -r /home/everestadmin/svn/ever-est/requirements.txt |
| **Application Initialization** | Create Django "everestadmin" superuser with "ev3R@dm1nE$t" password | python2.7 /opt/ever-est/manage.py createsuperuser --username everestadmin |
| | Create Django Tables | python2.7 /opt/ever-est/manage.py migrate |
| | Fill Django Tables | python2.7 /opt/ever-est/manage.py loaddata 0001_initial_data.json |
| | Create Apache static files | python2.7 /opt/ever-est/manage.py collectstatic |
| | Restart Apache | sudo systemctl restart httpd.service |

**Table 4-5 EVER-EST portal main configuration**

### A.1.4. Uninstallation procedure

The uninstallation procedure will be better described in next release of this document.

## A.2. E-Collaboration Services installation guide

The client part of the E-Collaboration services has been installed during initial EVER-EST portal configuration (explained in section A.1). Installation of the server part is addressed in this chapter. In particular, it explains how to install and configure the eJabberd[16] XMPP server used to add chat capabilities to the portals. Details about the eJabberd server are presented in section 4.2.2.

The eJabberd server could be installed and configured on the same machine that hosts the EVER-EST Portal.

### A.2.1. Software prerequisites

Minimum software requirements for the eJabberd server:

- OS: CentOS Linux 7.2-1511 (Core);
- Erlang (v. 19.0).

### A.2.2. Hardware prerequisites

Minimum hardware requirements for the eJabberd server:

- CPU: 2 core CPU x86_64;
- RAM: 8 GB;
- Disk Storage: 20 GB.

### A.2.3. Service installation

#### A.2.3.1. Server installation

The following installation steps should be executed as "root" user.

[16] https://www.ejabberd.im/

| Step | Command |
|---|---|
| **Install dependencies** | yum groupinstall 'Development Tools' && yum install gcc glibc-devel make ncurses-devel openssl-devel autoconf libyaml-devel expat-devel |
| **Clone last version of the eJabberd Git Repository** | mkdir -p repository/git && cd repository/git/ && git clone https://github.com/processone/ejabberd.git |
| **Install ERLANG** | cd ~ && wget http://erlang.org/download/otp_src_19.0.tar.gz && tar -xzvf otp_src_19.0.tar.gz&& cd otp_src_19.0/ && ./configure && make && make install |
| **Prepare eJabberd Installation** | cd ~/repository/git/ejabberd/ && ./autogen.sh && ./configure&& make && make install |
| **Start eJabberd Server** | ejabberdctl start |
| **Register an XMPP Admin account** | ejabberdctl register admin localhost ev3R@dm1nE$t |
| **Update Module List** | ejabberdctl modules_update_specs |
| **Show the list of the available modules** | ejabberdctl modules_available |
| **Start eJabberd Server** | ejabberdctl start |
| **Test the Web Admin Console** | http://localhost:5280/admin/ |

**Table 4-6 eJabberd server installation**

#### A.2.3.2. Server configuration

Edit the "/etc/ejabberd/ejabberd.yml" configuration file as "root" user in the following way:

- Check the admin account configuration in the "acl" section:

```
acl:
    admin:
        user:
            - "admin": "localhost"
access:
    configure:
        admin: allow
```

- Add the listener on the eJabber XMLRPC port:

```
Listen:
-
    port: 4560
    module: ejabberd_xmlrpc
    access_commands: {}
```

- Configure Modules: see section A.2.3.3
- Restart the eJabberd server

```
ejabberdctl restart
```

#### A.2.3.3. Modules configuration

##### A.2.3.3.1. mod_webpresence

| | |
|---|---|
| **Module Name** | mod_webpresence |
| **Description** | Publish user presence information in the web |
| **Link** | https://github.com/processone/ejabberd-contrib/tree/master/mod_webpresence |

**Table 4-7 eJabberd mod_webpresence module.**

Installation:

ejabberdctl module_install mod_webpresence

Configuration:

- Edit the "/etc/ejabberd/ejabberd.yml" configuration file and add the module configurations:

```
listen:
    -
        port: 5280
        module: ejabberd_http
        [...]
        request_handlers:
        "presence": mod_webpresence
...
modules:
    mod_webpresence:
        pixmaps_path: "/path/to/pixmaps"
```

### A.2.4. Uninstallation procedure

The uninstallation procedure shall be described in detail within the next release of this document.