envision

environmental services infrastructure with ontologies

Deliverable D1.2:

Environmental Services and Models
Scenarios and Pilots
Requirements specification

Date:	September 2010
Author(s):	Joël Langlois(BRGM), François Tertre(BRGM), Florian Husson(BRGM)
Dissemination level:	PU
WP:	WP1: Environmental Services and Models – Scenarios and pilots
Version:	0.1
Keywords:	ENVISION, Pilots, Requirements, Specification, UML
Description:	Report presenting requirement specifications

ICT for Environmental Services and
Climate Change Adaption

Small or Medium-scale Focused Research Project
ENVISION (Environmental Services Infrastructure with Ontologies)
Project No: 249120
Project Runtime: 01/2010 – 12/2012

Document metadata

Quality assurors and contributors

Quality assuror(s)	Maja Skrjanc(IJS), Roy Mellum(NMA), Patrick Maué(UoM)
Contributor(s)	BRGM, UoM, SINTEF

Version history

Version	Date	Description
0.1	October 2010	Edition for internal review

Executive Summary

The ENVISION project develops an infrastructure for deploying semantic web services and environmental models to given communities. The objective of this document is to present the requirements specification for the pilot cases of the project and to describe relevant data sources.

This document is therefore the basis for the validation of the ENVISION Portal capabilities from the user point of view.

We start by introducing the context of the document (section 1) and then explain the methodology used to identify requirements in the perspective of the evaluation of the results (section 2).

The requirements are described (section 3) focusing on the different viewpoints of the methodology. A resulting table is then presented (section 4) in order to formally summarize the requirements.

At the end, section 5 concludes the document.

Table of contents

Table of figures

1 Introduction

The ENVISION project [1] provides an ENVIronmental Services Infrastructure with Ontologies which aims to support non ICT-skilled users in the process of semantic discovery and adaptive service chaining of environmental services.

From the DOW [2], the goal of WP1 (Environmental Services and Models; Scenarios and pilots) is to define a set of industrial use cases and generate requirements. The use cases will also be used to check the applicability and interoperability of the technical results across different domains.

Within WP1, the goal of task 1.2 (Requirements specification) is to specify stakeholder and ICT requirements for the correct fulfilment of the pilot cases. These requirements are defined in sufficient detail so that the technical work packages are able to deliver the components needed for ENVISION and so that the relevant data sources and geo-treatments for the pilot cases can be made available by BRGM (Landslide Pilot) and SINTEF (Oil Spill Pilot).

This document (D1.2) is the output of task 1.2 and is relying on the previous works made in the work-package 1, and also by all the partners in the other work packages.

Some elements from the previous deliverables may have been updated within this document when changes occurred.

Complementary information and more details about previous activities may be found in the other work package related documents:

- D1.1 [3]: Definition of pilot cases
- D2.1 [4]: Architecture specification
- D3.1 [5]: MaaS Composition Portal – Architecture specification.
- D4.1 [6]: Ontology requirements analysis.
- D5.1 [7]: Deployment of the OGC Catalogue.
- D6.1 [8]: ENVISION Adaptative Execution Infrastructure – Architecture specification.

2 Methodology for ENVISION Pilots requirements specification

This section presents the methodology used in the ENVISION project to specify pilot requirements. The baseline for it is Reference Model of Open Distributed Processing (RM-ODP).

2.1 Description of RM-ODP Framework

From Wikipedia, the free encyclopaedia, RM-ODP is a reference model in computer science, which provides a co-ordinating framework for the standardization of open distributed processing (ODP). The RM-ODP view model provides five generic and complementary viewpoints on the system and its environment.

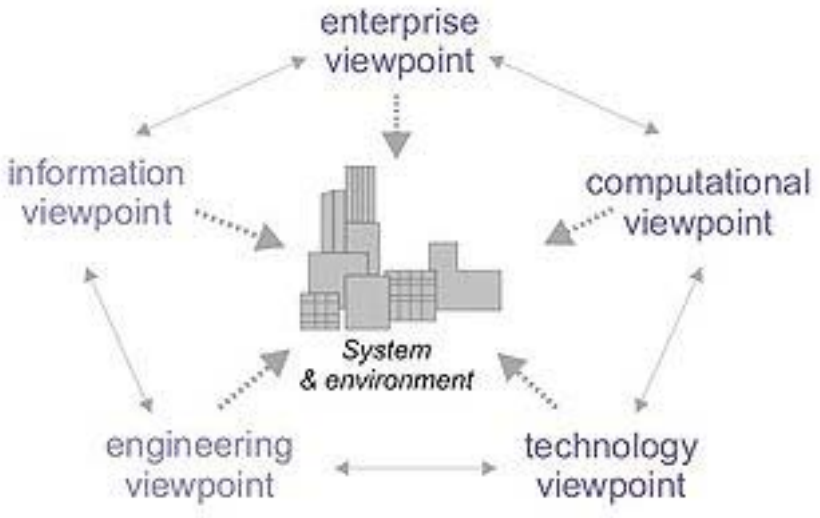

Figure 1 – RM-ODP different viewpoints (source Wikipedia)

These viewpoints are:

The enterprise viewpoint, which focuses on the purpose, scope and policies for the system. It describes the business requirements and how to meet them.

The information viewpoint, which focuses on the semantics of the information and the information processing, performed. It describes the information managed by the system and the structure and content type of the supporting data.

The computational viewpoint, which enables distribution through functional decomposition on the system into objects which interact at interfaces. It describes the functionality provided by the system and its functional decomposition.

The engineering viewpoint, which focuses on the mechanisms and functions required to support distributed interactions between objects in the system. It describes the distribution of processing performed by the system to manage the information and provide the functionality.

The technology viewpoint, which focuses on the choice of technology of the system. It describes the technologies chosen to provide the processing, functionality and presentation of information.

2.2 How we use RM_ODP in the ENVISION project

The RM_ODP framework allows specifying the system in terms of separate but interrelated viewpoint specifications as described above. In this document, we will go through the 5 viewpoints.

Because recent work from ORCHESTRA [9] and SANY [10] projects has renamed the “computational viewpoint” to the “service viewpoint”, we will use the last terminology.

To graphically support the expression of the specification, we make use of concepts maps and UML diagrams.

The UML case diagrams are used to overview the usage of the system from the different actors perspectives. It gives the possible actions.

UML sequence diagrams extend the graphical notations and allow describing the flows of logic within the system and between the modules.

This approach may be a first step toward the SERVUS methodology described in “Services Oriented Design of Environmental Information Systems” [11] by Thomas Usländer.

3 Requirements specification

This section goes through the different RM_ODP viewpoints in order to identify the requirements.

3.1 Enterprise viewpoint

The enterprise viewpoint focuses on the purpose, scope and policies for the system. It describes the business requirements and how to meet them. This chapter presents actors and roles of the system, and then, lists the actors' activities required to fulfil the project objectives.

3.1.1 Actors and Roles

Generally speaking, actors involved in the system are environmental data services providers (government agencies, academics, research Institutes …) and scientists providing Environmental Models. General public may also be interested by the resulting information delivered on the web.

In the ENVISION Portal, the following primary user roles are considered. One user can take up multiple roles.

End-User: uses Scenario Websites.

Designer: uses the Envision Portal to design a service composition and a dedicated website to be published for a given community.

Manager: uses the Envision Portal to manage resources involved in the Envision Infrastructure.

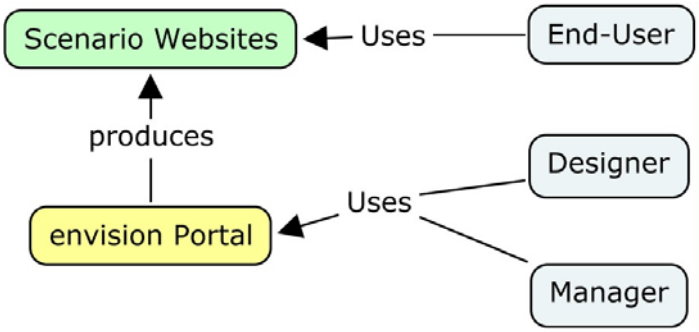

Figure 2 – User Roles

These primary user roles may be specialized as below when needed:

The designer may represent either:

- **scenario website designer**
- **service composition designer**

The manager may be either:

- **semantic catalogue manager**
- **annotations manager**

3.1.2 User activities

This chapter presents the key activities of the users of the system.

3.1.2.1 User activities

Envision Portal deals with authenticated user and non-authenticated users. Visitors can create an ENVISION Portal account. The activities which may involve any user regardless of its role are:

- Authenticate
- Manage personal account
- Join a community

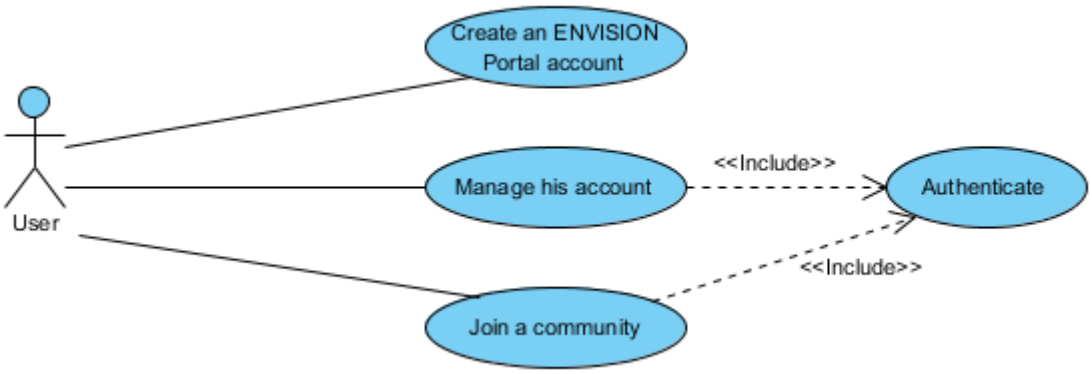

Figure 3 – User use case diagram

3.1.2.2 Scenario Website Designer

The Scenario Website Designer produces Websites for a given community. The produced websites are based on service compositions that are made available online.

A website is composed of modules (portlets). Primary modules are modules which are essential to do service compositions and subscribe to the model. Secondary modules are optional modules like Image gallery.

The key activities for the Scenario Website Designer are:

- Create a community
- Create a dedicated website
- Select the template
- Select and configure the composition
- Select, add and configure secondary modules
- Create content (add pictures, write text…)
- Export/Import a website (deployment)
- Edit existing content/website

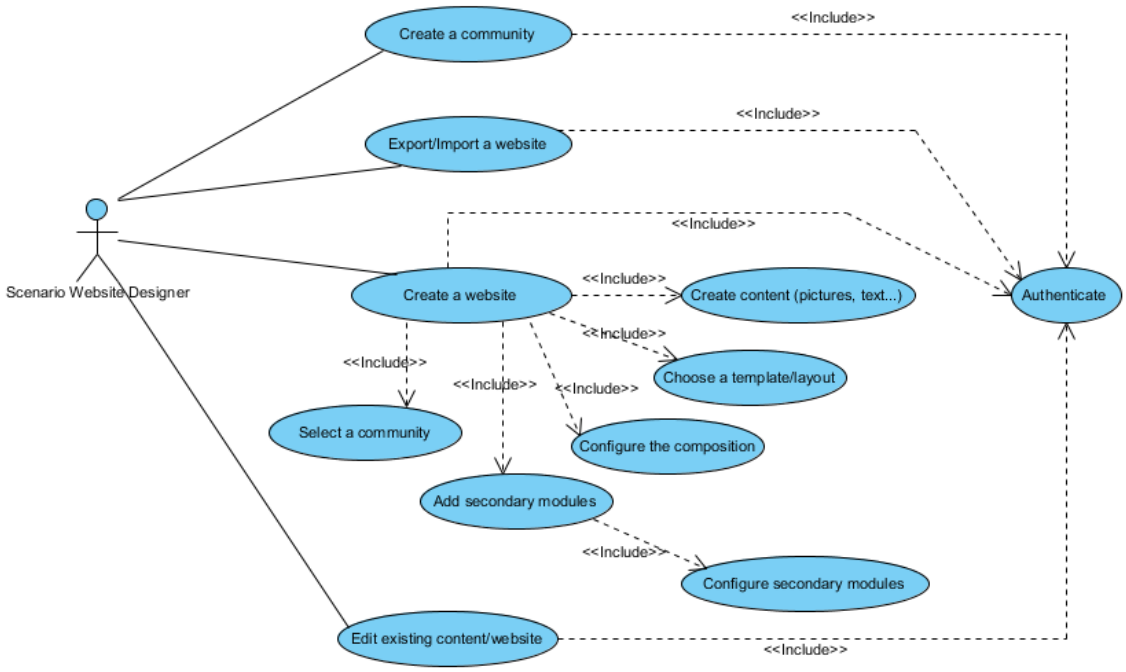

Figure 4 – Scenario Website Designer use case diagram

3.1.2.3 Service Composition Designer

The Service Composition Designer is the domain expert who is able to compose new Web services from existing data and processing Web services. He can use the Semantic Catalogue to discover existing Web services and add them to his current collection.

The key activities for the Service Composition Designer are:

- Create a new composition
- Create mediation rules
- Deploy workflow
- Edit existing composition
- Store composition

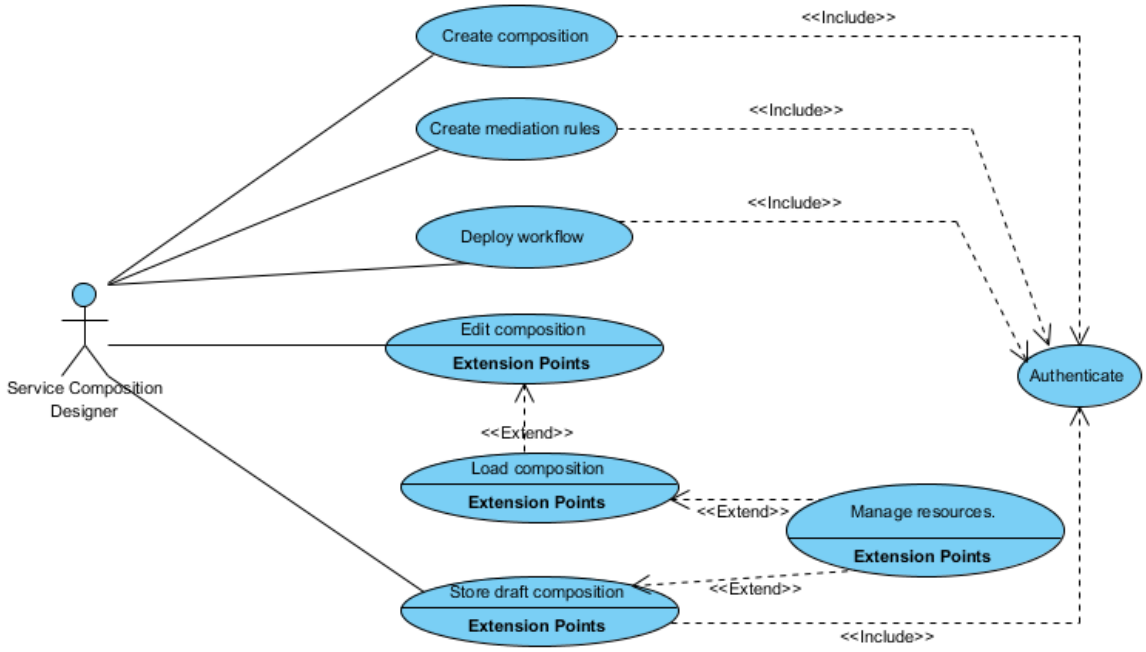

Figure 5 – Service Composition Designer use case diagram

3.1.2.4 Semantic Catalogue Manager

The Semantic Catalogue Manager is able to search and manage resources which may be useful for designing a new portal, or creating a new composition.

The key activities for the Semantic catalogue manager are:

- Search for resources
- Delete existing resources
- Publish resources
- Unpublish resources

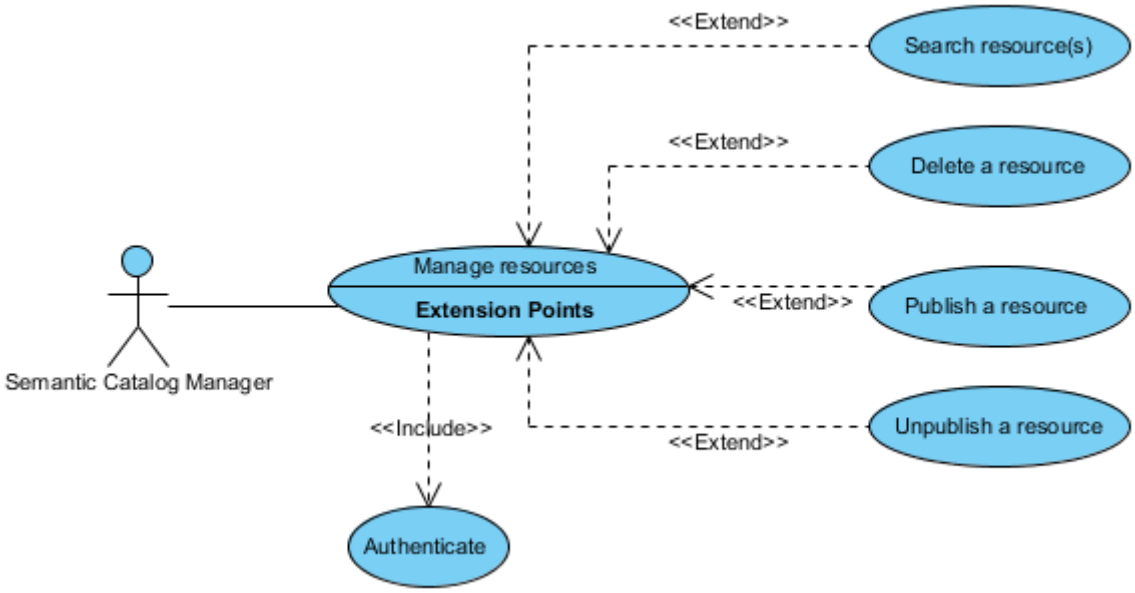

Figure 6 – Semantic Catalog Manager use case diagram

3.1.2.5 Annotations Manager

The Annotations Manager is able to semantically annotate the resources in the workspace which are not yet semantically described.

The key activities for the Semantic catalogue manager are:

- Load Ontologies
- Annotate a resource (draft or not)
- Update an annotation

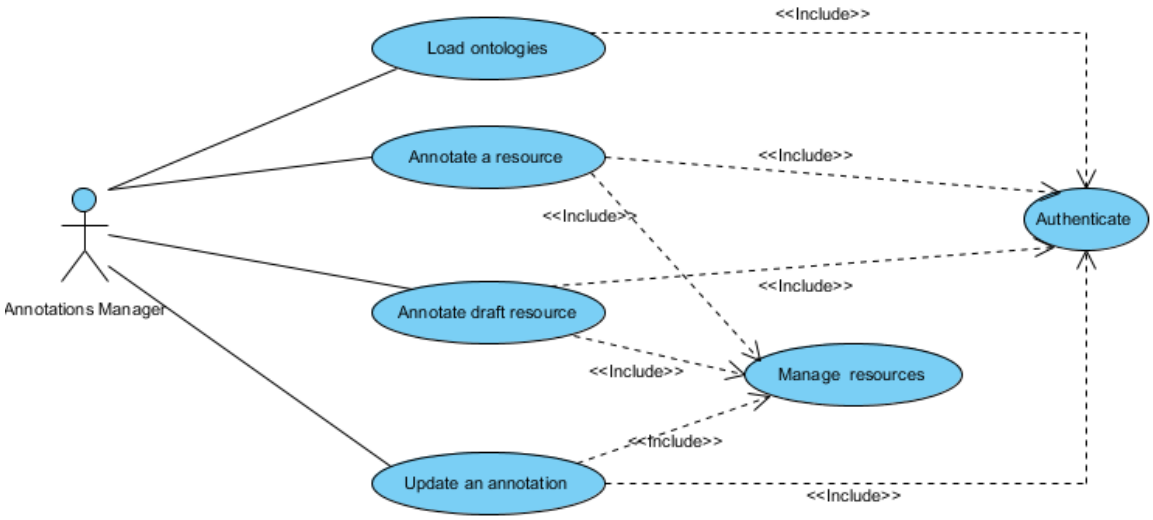

Figure 7 – Annotations Manager use case diagram

3.1.2.6 Websites End User

End users use resulting scenario websites made available by the website designer for a given community. The scenario website functionalities depend on what is made available by the website designer. However, in the context of the 2 pilots, the more relevant activities are:

- Interact with the map,
- Update model parameters,
- Visualize the executed composition (which produces the map)
- Subscribe to the model (alerts or informational messages produced by the model).

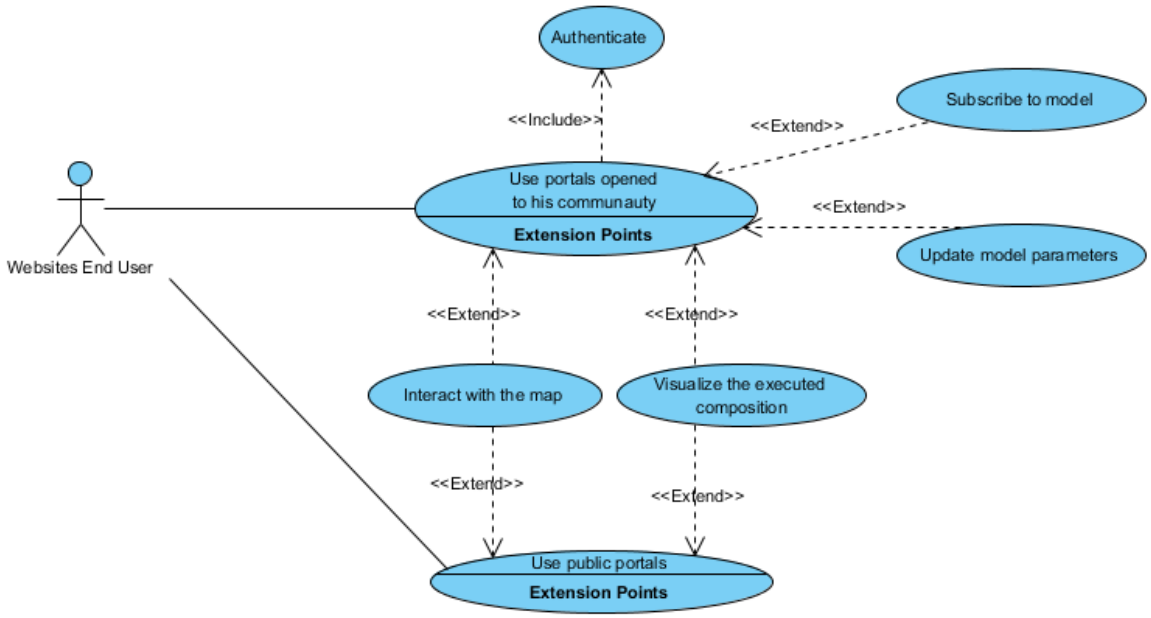

Figure 8 – Websites end user use case diagram

3.2 Information viewpoint

The Information viewpoint focuses on the information and the processing performed. It describes the information managed by the system and the structure and content type of the supporting data. This chapter summarizes the pilot oriented processes and information and then express generic requirements.

3.2.1 Landslides pilot

The following table summarizes the landslide pilot required processing services and their main inputs and outputs. More details about what are the inputs and treatments may be found in D1.1.

Input data			Model as a service		Output data		
Designation		Format			Designation		Format
In1	DEM	Grid	Ms1	MakeGeologicalModel	Out1	Geological / geotechnical model	Grid
In2	Geological map	Vector					
In3	Borehole data	Points					
In4	Geotechnical data	Grid					
In5	Precipitation data series	Series	Ms2	PredictWaterTable	Out2	Water table map	Grid
In6	Geotechnical model	Grid	Ms3	ComputeLandslideProbability	Out3	Map of safety factors (SF) or probabilities of landsliding	Grid
In7	Water table map	Grid					
In1	DEM	Grid	Ms4	ComputeLandslideRunout	Out4	Landslide hazard map	Grid
In8	Map of probabilities of landsliding	Grid					
In9	Landslide hazard map	Grid	Ms5	ComputeDamage	Out5	Risk map	Vector
In10	Map of elements at risk	Vector					

Table 1: The landslide pilot models as a service

These web services are made available online by BRGM at http://envision.brgm-rec.fr

3.2.2 Oil spill pilot

The following table summarizes the oil spill pilot required processing services and their main inputs and outputs. More details may be found in D1.1.

Input data			Model service	Output data		
Name		Data type		Name		Data type
In1	User input (location, time, amount, oil type)	Simple data types	Ms1 *PredictOilDrift*	Out1	Oil drift predictions: - Oil concentration in water column (3D) - Oil slick position (2D) - Mass balance (1D)	Grids time series
In 2	Wind forecast	2D grid time series				
In 3	Current forecast	3Dgrid time series				
In4	Sea depth data	2D grid				
In 5	Coast line data	2D grid				
In6	Cod species and population data	Simple data types; 2D grid	Ms2 *PredictCodEffects*	Out2	Predictions of the effect on cod: - Lethality	2D time series
Out1	Oil drift predictions	Grids time series				

Table 2 The oil spill pilot models as a service

These web services are made available online by SINTEF.

3.2.3 Generic requirements

When generalizing the requirement, the Envision Portal has to handle web services providing data (data source services), processes (processing services) or sensors. These web services are exposed on Internet trough OGC standards.

The main targeted specifications are:

- **WMS** (http://www.opengeospatial.org/standards/wms)

Specification for a web service interface serving map images generated on the fly.

- **WFS (**http://www.opengeospatial.org/standards/wfs)

Specification for a web service interface serving spatial features encoded in the Geography Markup Language (GML).

- **WCS (**http://www.opengeospatial.org/standards/wcs)

Specification for a web service interface serving coverages (raster data).

- **WPS (**http://www.opengeospatial.org/standards/wps)

Specification for a web service interface serving geoprocessing capabilities handled by a processing server.

- **WCPS (**http://www.opengeospatial.org/standards/wcps)

Specification for a web service interface serving coverage geoprocessing capabilities.

- **CS/W (**http://www.opengeospatial.org/standards/specifications/catalog)

Specification for a web service interface serving repository over the web.

- **SOS (**http://www.opengeospatial.org/standards/sos)

Sensor Observation Service Interface Standard (SOS) provides an API for managing deployed sensors and retrieving sensor data and specifically “observation” data. Whether from in-situ sensors (e.g., water monitoring) or dynamic sensors (e.g., satellite imaging), measurements made from sensor systems contribute most of the geospatial data by volume used in geospatial systems today. This is one of the OGC Sensor Web Enablement (SWE) [http://www.opengeospatial.org/ogc/markets-technologies/swe] suite of standards.

- **ISO19115 (Reference)**

For resources cataloging activities, the metadata ISO 19115 standards will be used.
It is a component of the ISO 191xx standards. ISO 19115 defines how to describe geographical information and associated services, including contents, spatial-temporal purchases, data quality, access and rights to use. The standard defines more than 400 meta data elements, 20 core elements.

3.3 Service viewpoint

The service viewpoint enables distribution through functional decomposition on the system into objects which interact at interfaces. This chapter summarizes the modules of the system.

3.3.1 Modules of the system

The following table summarizes the modules identified to fulfil the requirements. More details about this organisation may be found in D2.1.

The table shows the current status and the list may have to evolve in the next steps.

Module Name	Author	Key requirements description
User	BRGM	• Allows a visitor to log in, register. • Allows a user to manage his account. • Allows the manager to assign communities to user, to create new communities.
Website Designer	BRGM	• Is used to make the composition/model available to an end user community. • Allows the Website Designer to create a new website based on a composition. • Allows the Website Designer to set up the composition with some data specific to its domain. • Allows the Website Designer to add some pages (e.g. with a wiki or a forum) with some useful information for understanding the computer model and its results.
Map-Viewer	BRGM	• Is a map with standard interaction techniques like panning, zooming • Has map actions with a set of extended tools for changing the content of the map module, e.g. to change the displayed variables, or adjust the time. • Has a legend that displays information depending of the current content of the map
Chart-Viewer	BRGM	• Enables to view features displayed as charts
Profile-Viewer	BRGM	• Enables to view profiles displayed as 2D profile maps (Specific to the oil spill pilot)
Notification	UoM	• Allows for subscribing to the available information sources.
Resource	UoM	• Allows to browse/search resources • Allows context-dependent actions such as publishing/unpublishing.
Timeline	BRGM	• Allows to navigate within scenario and/or time steps
Property-Viewer	SINTEF	• Allows to editing the properties of the currently selected item in the workflow view.

Service Composition Editor	SINTEF	• Allows a designer to graphically create & test a composition using a modeling notation
Semantic Catalogue	UIBK	• Supports searching for OGC Web services using thematic, spatial, and temporal search criteria • Allows for browsing through registered services • Allows formulation of semantic queries
Semantic Annotation	JSI	• Is used for the semantic annotations of resources; supports ontology maturing through user feedback. • Allows viewing and graphical navigation into Ontologies
Ontology Querying	UOM	• Is used to search for Domain Ontologies, related or not to given resources

Table 3 – List of Modules

3.4 Engineering viewpoint

The engineering viewpoint focuses on the mechanisms required to support distributed interactions between modules in the system. Then, for each key activity, we give a narrative description of the activity and an UML sequence diagram that illustrates interactions.

3.4.1 Authenticate

The authentication is related to account management through the portal. This allows the portal to propose different Website with restricted access depending of communities.

3.4.1.1 Narrative description

Sequence of actions:

1. The user opens the portal.
2. He uses the menu to select the 'Sign In' option.
3. He enters its login and password and then clicks on 'Sign In' button.
4. Two cases can be reached:
    1. The couple login/password is not valid for the portal. So the authentication failed and an error message is displayed explaining what is the problem.
    2. The couple login/password is valid for the portal. The portal returns to the home page for the current user.

3.4.1.2 UML description

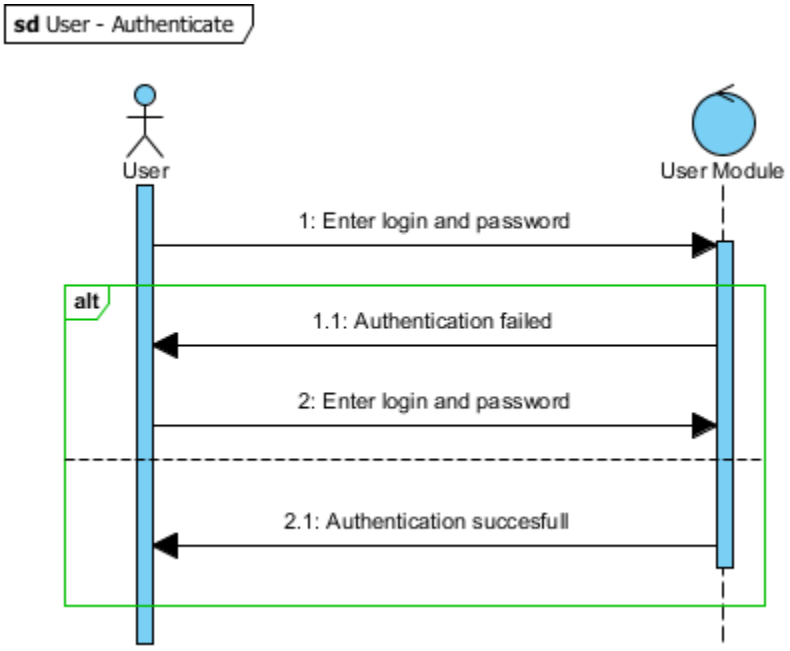

Figure 9 – Authenticate sequence diagram

3.4.2 Create a Website

The functionality for creating the End User Portal ships with the software. The only task is the configuration of the individual modules.

3.4.2.1 Narrative description

Sequence of actions:

1. Website Designer authenticates as authorized user to the system.
2. In the Portal Configuration, he creates a new page.
3. In the configuration, he chooses a community.
4. In the configuration, he chooses one of the preconfigured themes.
5. Extended Configuration: this includes issues like changing the logo in the header, setting a virtual host, and more)
6. Depending on the requirements of the End User Portal, the Website Designer chooses a selection of portlets which are needed for explaining the model output. (see also Appendix B: *designer screenshot* to see a Liferay example) This could include the model visualizer module, the provenance visualizer module, and more. A complete list of possible module can be found in Appendix A.
7. If needed, the website designer can add more pages.

3.4.2.2 UML description

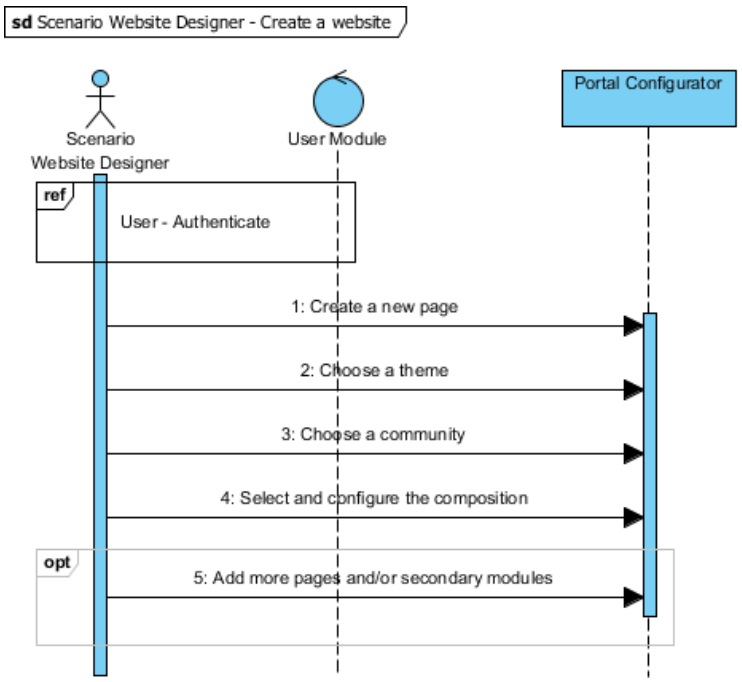

Figure 10 – Create a Website sequence diagram

3.4.3 Export/Import Website

Since the website is simply a configuration of pages (and accordingly modules), it can be simply exported and later on imported again on another portal installation. The ENVISION portal may be used to design the website. Any other portal based on the same technology can be used to deploy it.

3.4.3.1 Narrative description

Sequence of Activities:

1. In Community Configuration, choose export to download the designed portal as .lar-file.
2. Before import, make sure to load all required portlets from the ENVISION portlet repository
3. Import the exported file in another setup.

3.4.3.2 UML description

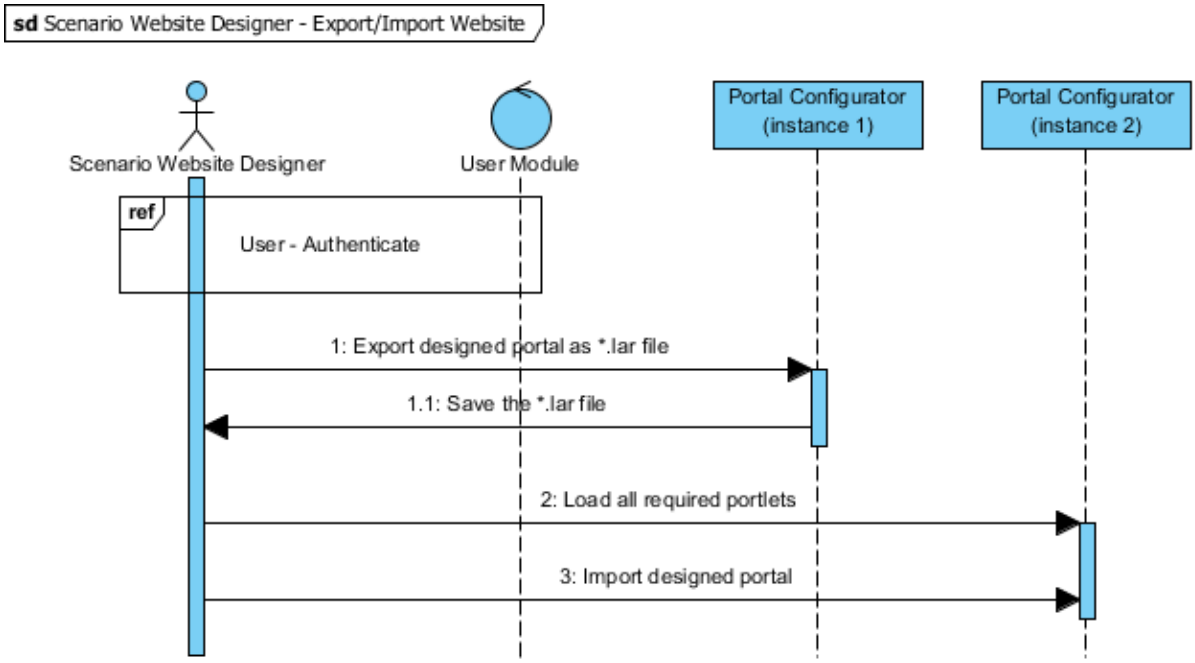

Figure 11 – Export/Import a Website sequence diagram

3.4.4 Create composition

The functionality for creating composition is a key feature of the portal. It allows to graphically compose together existing web services to produce a more sophisticated one.

3.4.4.1 Narrative description

Sequence of actions:

1. If no composition is loaded, the composition view will show two buttons (create new composition, load existing composition). The domain expert clicks the button "Create New Composition".
2. In a dialog, he configures the name of the composition (and other properties if needed).
3. He selects services (resources) to add to the composition.
4. He saves the composition as an unpublished resource.

3.4.4.2 UML description

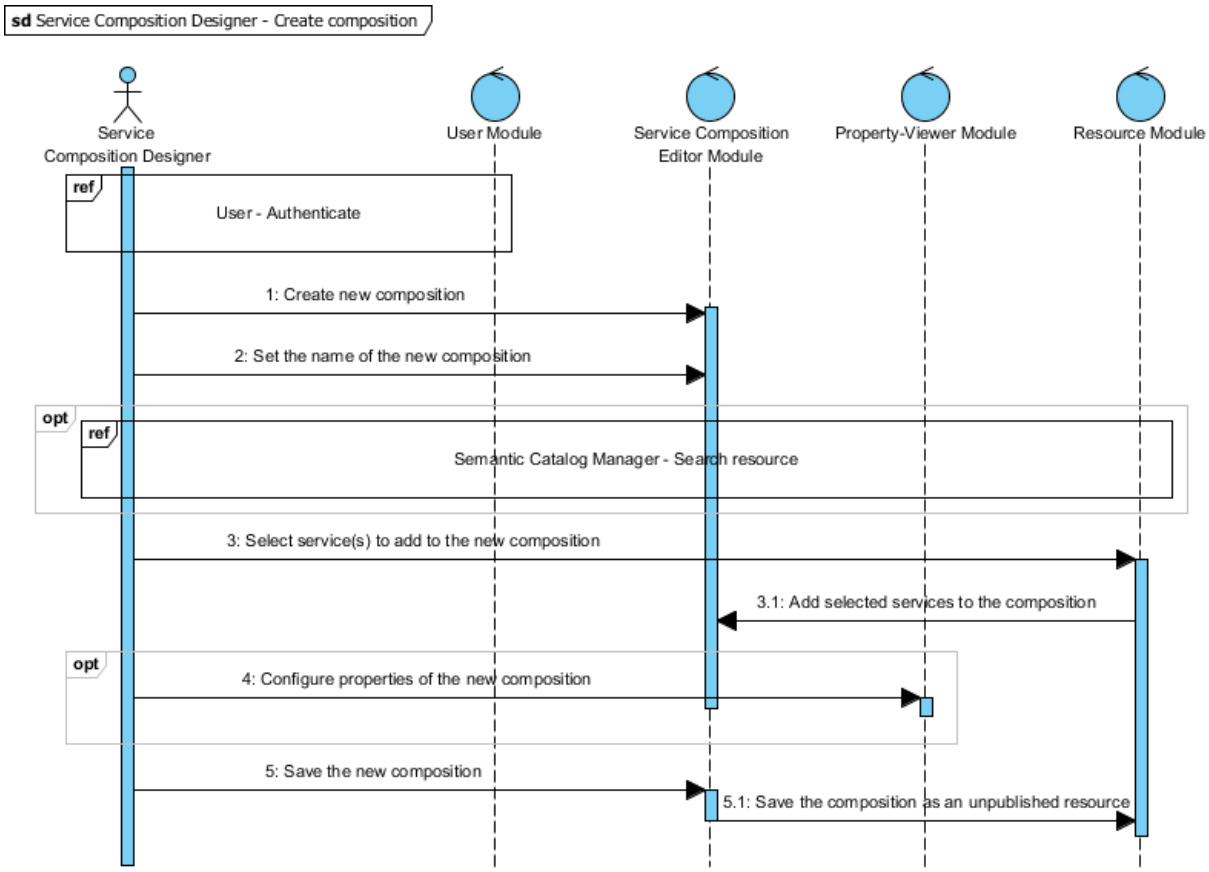

Figure 12 – Create composition sequence diagram

3.4.5 Edit composition

Sequence of actions:

1. Button "Load Existing Composition" was pressed,
2. Dialog with existing compositions in the workspace opens,
3. User selects one existing composition and clicks "Edit Composition",
4. User can add resource to the composition or change resource's properties.

3.4.6 Add resource to composition

Only Web services which are in the user's resource collection can be added in the composition. If, at design-time, no appropriate resource is available, he can switch to the catalog perspective to find an appropriate Web service (see Resource Module description)

Sequence of actions:

1. User selects resources from his collection (shown in Resource Module).
2. User adds selected resources to the composition.

3.4.7 Store draft composition

At this stage, the composition is simply a diagram stored in an internal format based on JSON. Storing it has draft means to export it as BPEL document, and register it to the workflow engine. The composition itself is then a web service described through a WSDL file, which is then again retrieved from the workflow engine (and stored in the user collection).

Sequence of actions:

1. In the Service Composition Editor Module, user clicks "Save draft composition".
2. The user clicks "Mark composition complete" (otherwise he won't be able to "Deploy composition").
3. Composition appears as new resource (WSDL service) in the user collection (and is marked as draft, since it is neither published nor annotated).

3.4.8 Search resource

Depending on the Query Module interface used by the user (see Query Module description) we have the following version of this activity.

3.4.8.1 Narrative description

Sequence of actions: (without semantic features)

1. In the Semantic Catalog Module, the user enters a keyword, set of keywords or selects a bounding box in the map
2. The user can specify additional options (e.g. the number of results he wants to be displayed, if the results should be ranked depending on chronological order of their creation/modification, etc.)
3. The user hits the search button
4. The relevant resources are displayed
5. For each relevant record, the user can choose to add it to his collection (displayed in the Resource Module)

Sequence of actions: (without semantic, advanced)

1. In the Semantic Catalog Module the user enters a keyword, set of keywords or selects a bounding box in the map the user can specify the type of resources she/he is interested in (checkbox for each type)
2. The user can specify additional options (e.g. the number of results she/he want to be displayed, if the results should be ranked depending on chronological order of their creation/modification, etc.)
3. The user hits the search button
4. The relevant resources are displayed
5. For each relevant record, the user can choose to add it to his collection (displayed in the Resource Module)

Sequence of actions: (with semantic)

1. In the Semantic Catalog Module query space the user specifies a set of concepts / annotations or a service template
2. The user can specify additional options (e.g. the number of results she/he want to be displayed, if the results should be ranked depending on chronological order of their creation/modification, etc.)
3. The user presses the "Search" button
4. The relevant resources are displayed
5. For each relevant record, the user can choose to add it to his collection (displayed in the Resource Module)

3.4.8.2 UML description

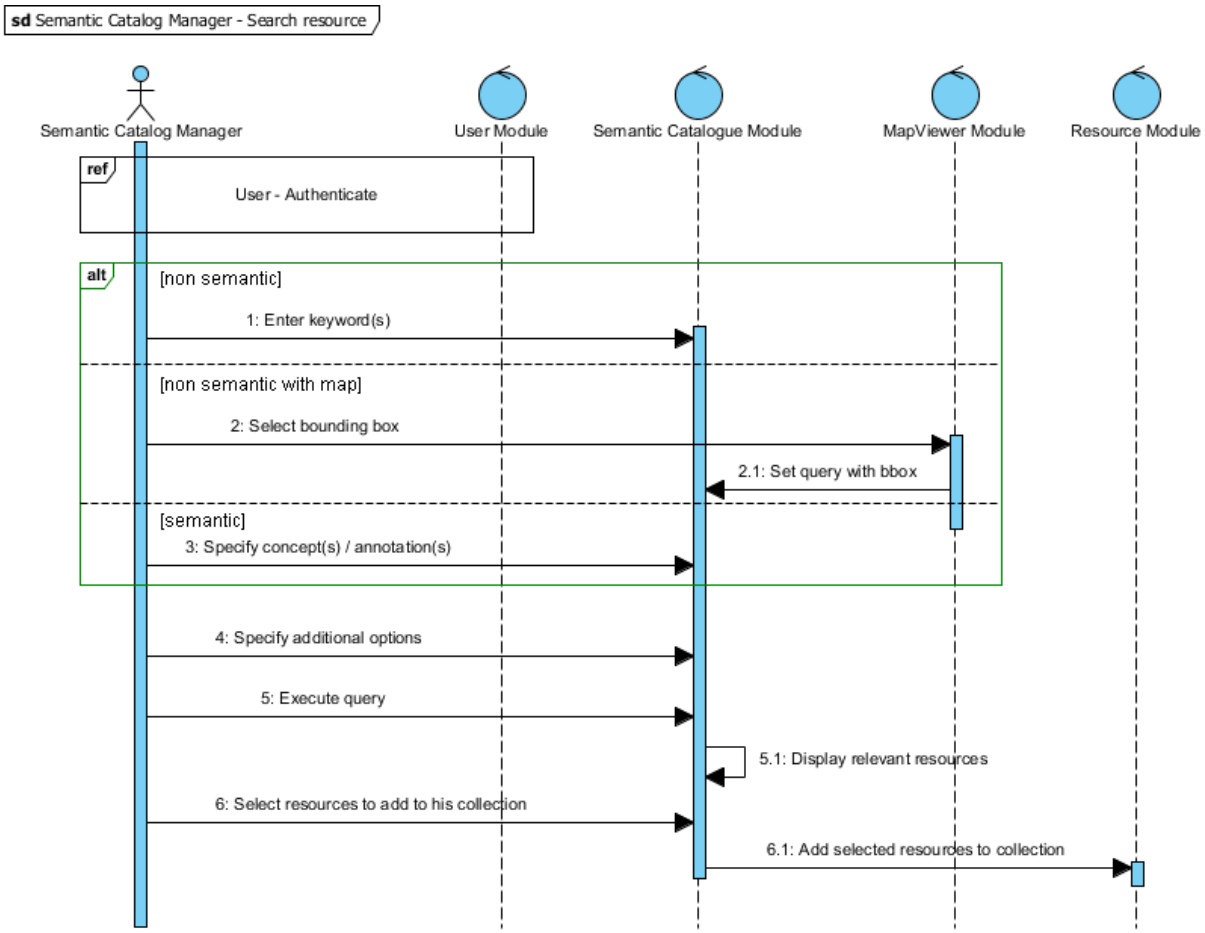

Figure 13 – Search Resource sequence diagram

3.4.9 Unpublish resource

Sequence of actions:

- In the Resource Module, the user searches/selects the resource he wants to unpublish.
- If he is marked as the creator of the resource (or is Semantic Catalog Manager), the button "Unpublish" appears.
- Pressing the button removes the resource from the Semantic Catalog, and changes the status of the resource to "unpublished"

3.4.9.1 UML description

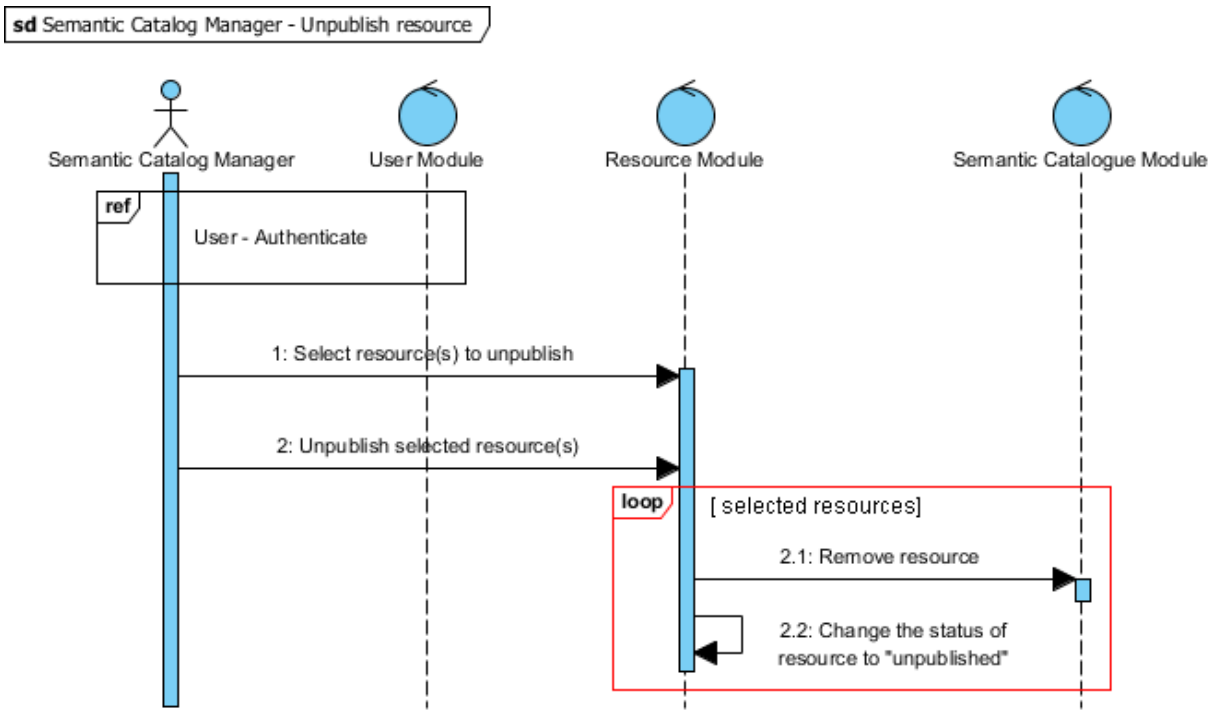

Figure 14 – Unpublish resource sequence diagram

3.4.10 Publish resource

3.4.10.1 Narrative description

Sequence of actions:

- In the resource module, the user selects a resource with the status "unpublished"
- In presses the button "Publish resource" in the resource actions
- Since all relevant information has been added before, no further user input is required here. The status of the resource changes to "published" (default).

3.4.10.2 UML description

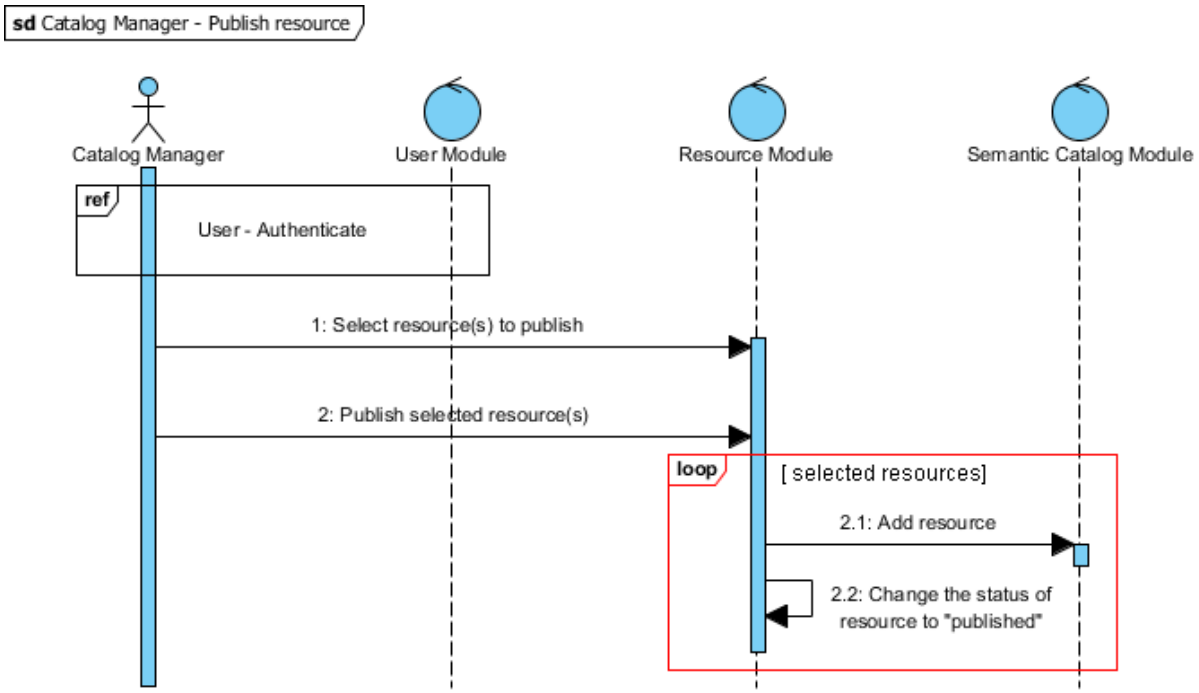

Figure 15 – Publish resource sequence diagram

3.4.11 Annotate draft resource in Resource Module

3.4.11.1 Narrative description

Sequence of actions:

- The user selects a resource from the Resource Module. He selects the EDIT ANNOTATION option from the Resource Actions menu. The resource is visualized in the Semantic Annotation Module.
- The user opens the Ontology Querying Module and double-clicks on a concept in the domain ontology to create its instance in the Semantic Annotation Module. The user creates as many instances as required for the annotation.
- To relate one instance to another, the user first selects an instance by clicking on it and then right-clicks on the other instance to access the context menu. In the context menu, the user selects the LINK option which provides a list of all possible relations between the two instances. The selected relation is established between the two instances and visualized as an arrow.
- To define a complete annotation, the instances of the domain ontology concepts need to be related to the instances of the resource. To establish these relations, the user selects a domain ontology instance and right-clicks on a resource instance (or vice versa). From the context menu, he selects the ANNOTATE option. This relates the domain ontology instance to the resource instance through the “annotate” relation (visualized as a red arrow).
- By selecting VIEW WSML from the Semantic Annotations Actions menu, the user can preview the annotation in the WSML format.
- When the user is done editing the annotation, he selects Save from the Semantic Annotations Actions menu. This saves the annotation in his resource module. The corresponding resource changes its status from “no annotations” to “unpublished” (i.e. it becomes ready to be published in the catalog). The annotation remains visible in the Semantic Annotation Module. The user is able to modify it.

3.4.11.2 UML description

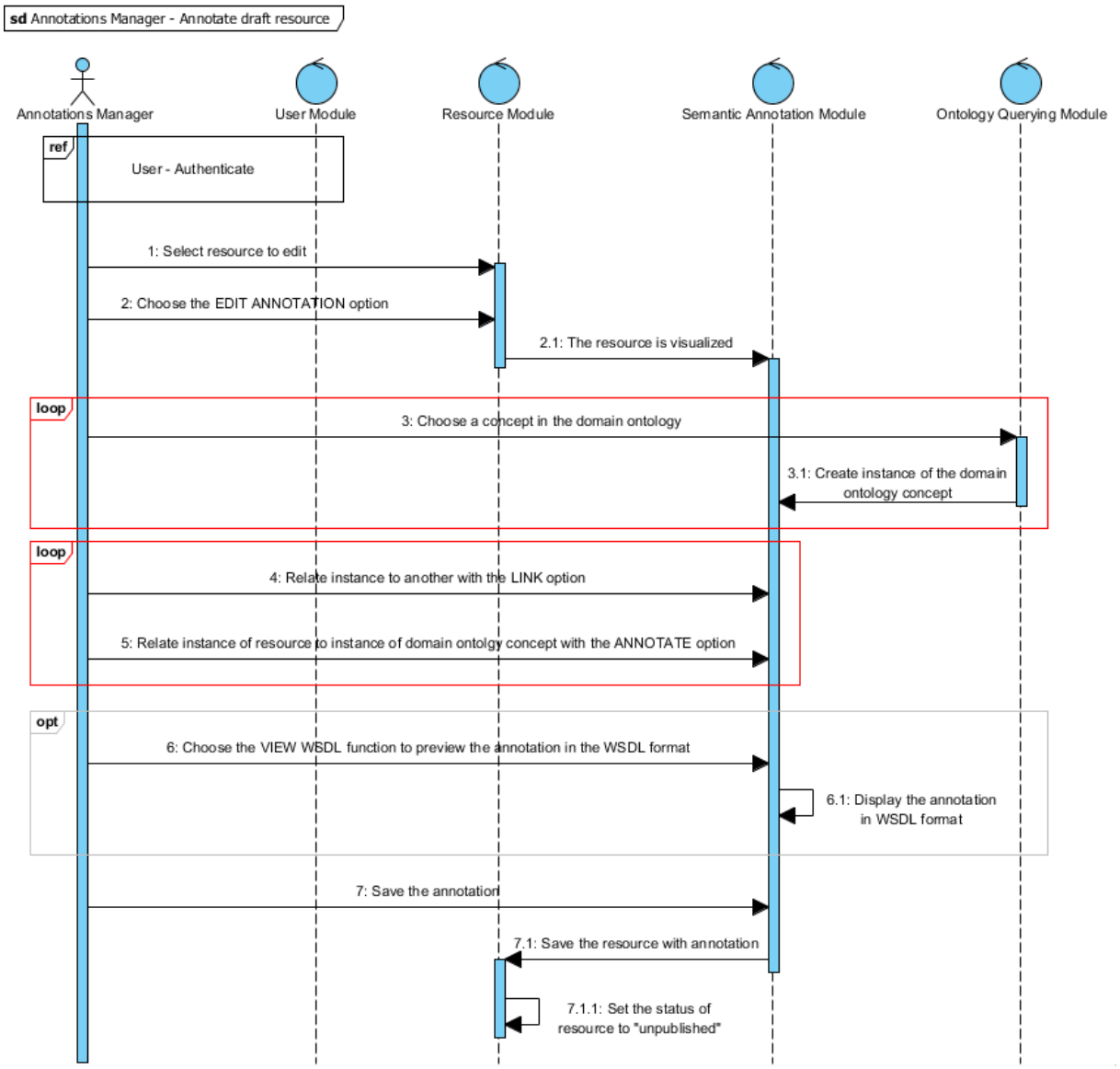

Figure 16 – Annotate draft resource sequence diagram

3.4.12 Annotate a resource

3.4.12.1 Narrative description

Sequence of actions:

1. The user selects a resource from the Resource Module. He selects the Edit Annotation option from the Resource Actions menu. The resource is visualized in the Semantic Annotation Module.
2. The user opens the Ontology Querying Module and enters a set of Google-like natural-language queries. He can add a query by clicking on the add button next to the last query. Similarly, he can remove a query by clicking on the remove button next to the query.
3. When the queries are set, the user clicks on the Search button. The two lists – the list of proposed concepts and the list of proposed triples – are populated with items.
4. The user inspects the list of proposed concepts, from top to bottom, and selects relevant concepts by checking the corresponding check-boxes. If the required concepts are not found at the top of the list, the user should consider reformulating the queries. Checking or unchecking a concept causes the triples to reorder: the triples containing at least one checked concept are bolded and pushed to the top of the list.
5. The user selects the required triples by checking the appropriate check-boxes in the list of triples. If all the required concepts have been checked, the user only needs to consider the bolded items.
6. When the user has selected all the required building blocks, the selected concepts and triples are transferred to the Semantic Annotation Module. In this process, each selected concept is instantiated exactly once.
7. The user has the ability to split the graph at a particular instance, join two instances into one single instance, or duplicate a particular instance. These operations are accessible from the context menu of an instance. Note that joining two instances requires precedingly selecting one of the two instances (by clicking on it); it is also necessary that the two instances belong to the same concept.
8. The user now performs steps 4 and 6 from the previous activity description (interrelating instances with the “annotate” relation and saving the annotation in the resource module).
9. The user changes status of the resource from “translated” to “annotated”.

3.4.12.2 UML description

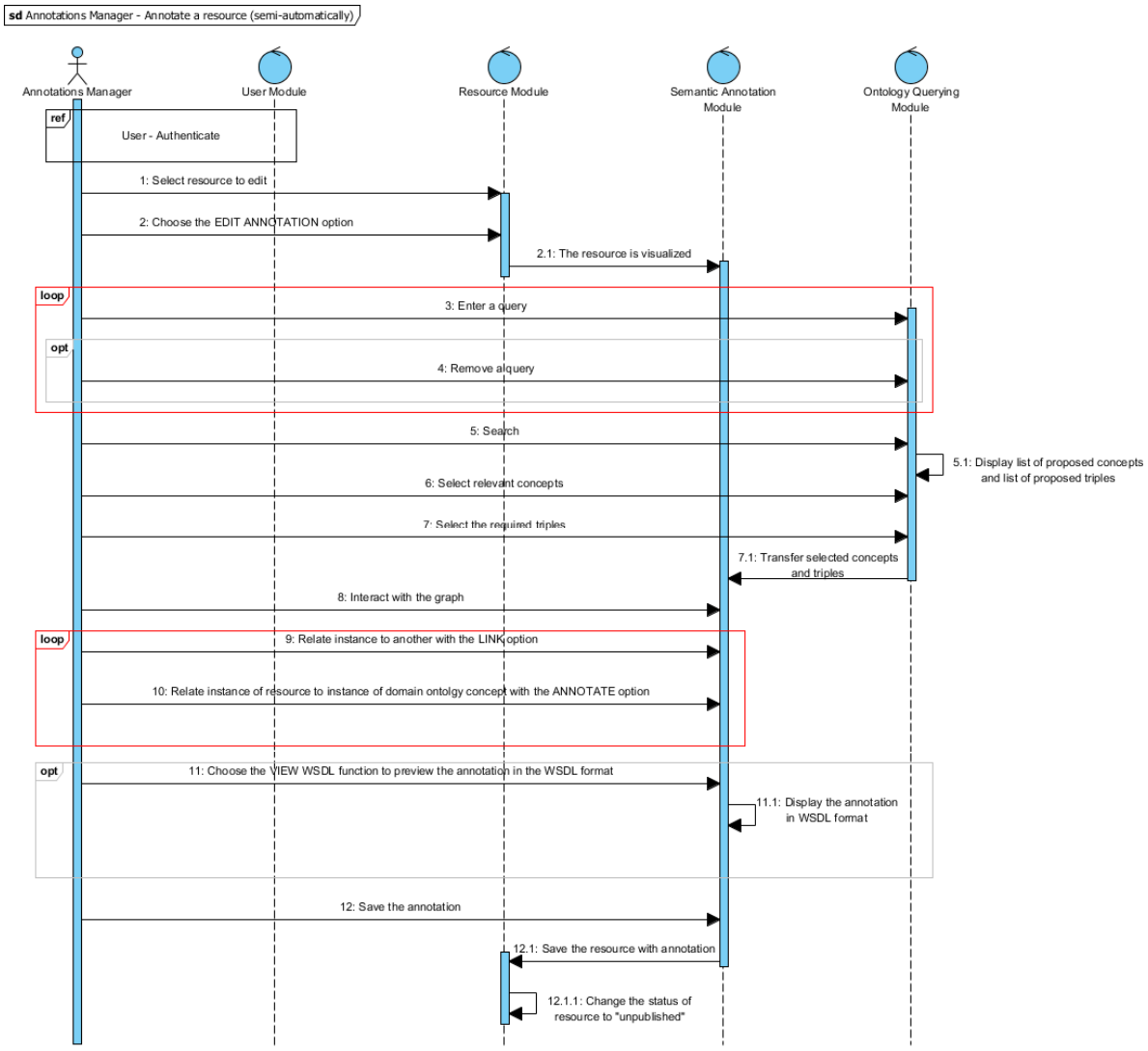

Figure 17 – Annotate a resource sequence diagram

3.4.13 Update an annotation

The user selects a resource from the Resource Module. He selects the EDIT ANNOTATION option from the Resource Actions menu. The resource and the corresponding semantic annotation are visualized in the Semantic Annotation Module.

The user can either modify the annotation manually by using the ontology's graph in the Semantic Annotation Module or semi-automatically by using the Ontology Querying Module. He is able to remove instances and links through the options in the context menus.

3.4.14 Interact with the map

The map is the way to give access to the information produced by the model.

They are multiple means to interact:

- Using standard interaction techniques to change the current view (pan, zoom in, zoom out…)
- Manage the layers (select, modify order, adjust transparency…)
- Click on displayed elements to obtain the underlying information (ex : visualise data from a localized piezo-meter)
- Make use of dedicated tools (ex: the profile viewer in the oil-spill scenario)

3.4.15 Update Model Parameters

When the designer made it available for the user, the user may update some parameters and re-run the underlying model.

3.4.15.1 Narrative description

Sequence of actions:

1. User presses "Configure Model" button
2. Dialog opens with configurable parameters
3. After adjusting the default values, the user presses "save parameters" button
4. User presses "Refresh" button to re-run underlying model and update the resulting map.

3.4.15.2 UML description

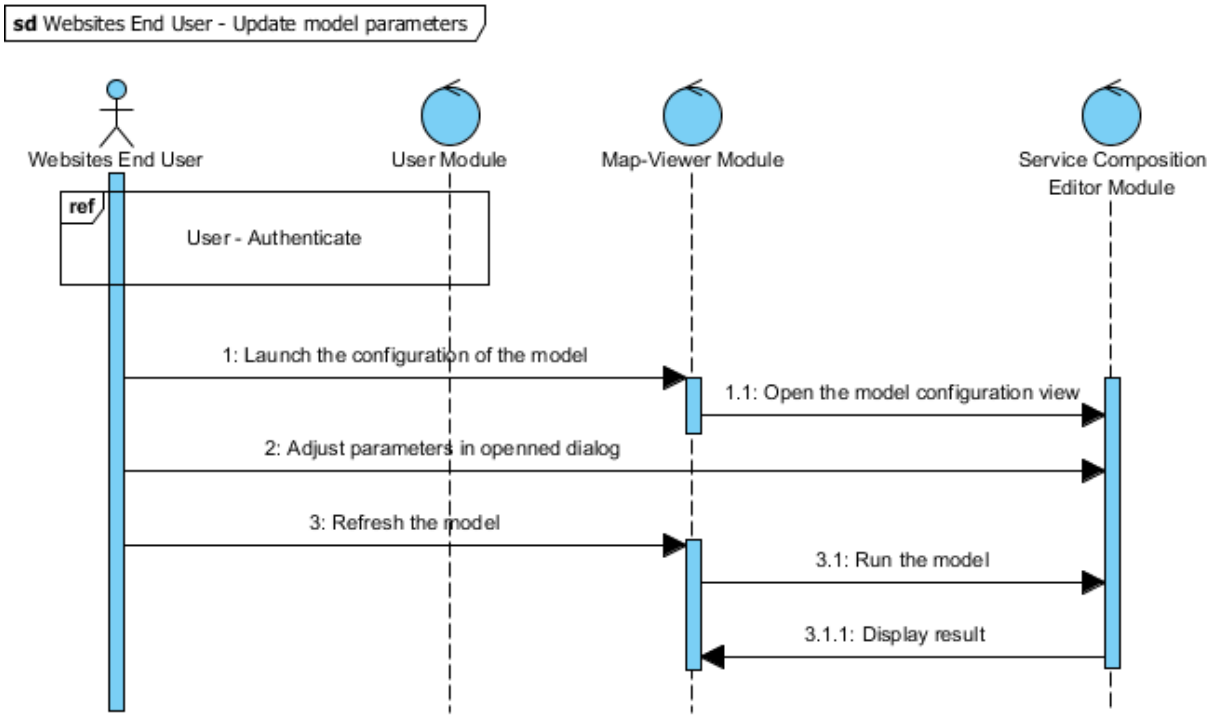

Figure 18 – Update model parameters sequence diagram

3.4.16 Visualize the executed composition

When the designer made it available for the user, the user may view the workflow representing the underlying model.

3.4.16.1 Narrative description

Sequence of actions:

1. User presses "View Model" button
2. Dialog opens with a view on the illustrating the model
3. Select items to display more information

3.4.16.2 UML description

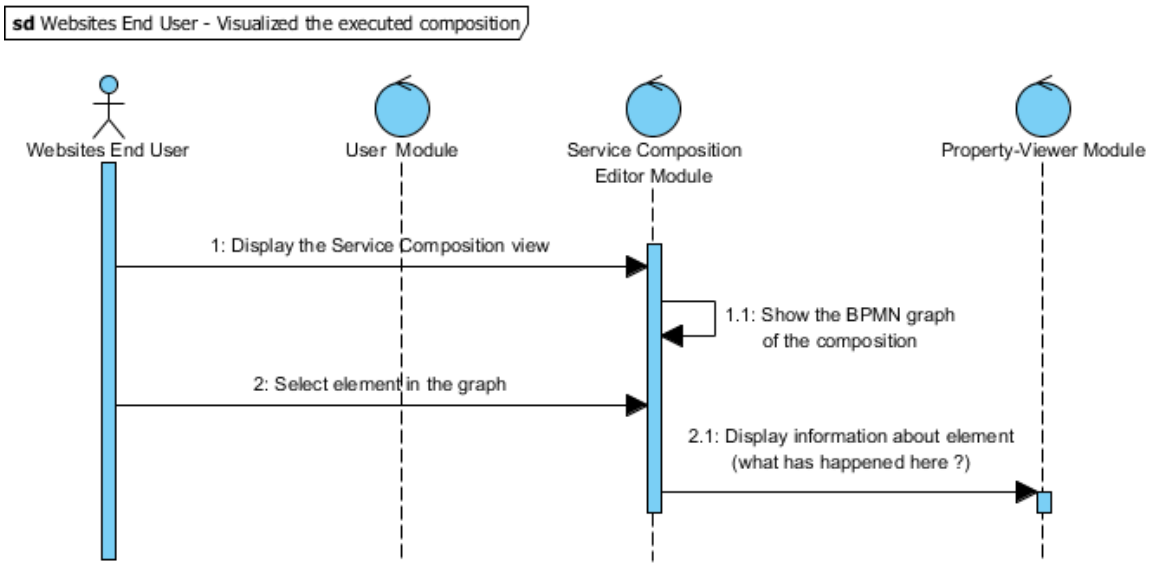

Figure 19 – Visualize the executed workflow sequence diagram

3.4.17 Subscribe to event notification

A model describes an event by defining certain thresholds for all the inputs, for example sensor observations. When an event, described by a model, is triggered the event notification notifies the subscriber. For example, if you subscribe to an notification for landslide events, a notification is sent out when an event is triggered within the selected region

3.4.17.1 Narrative description

A possible sequence of action is:

1. Select the region (BBox in the map)
2. Choose kind of notification (EMail or SMS provided by Radio Buttons)
3. Type in the contact (EMail address or phone number)
4. Select the events by clicking the button “Select Events” (if the button is clicked an extra window will appear)
5. The user clicks the “Subscribe” button

The sequence of 1 - 4 doesn’t matter. Only step 5 has to be always the last step, because in this step it is checked, whether a region is selected, a contact is given and a minimum of one event is selected.

3.4.17.2 UML description

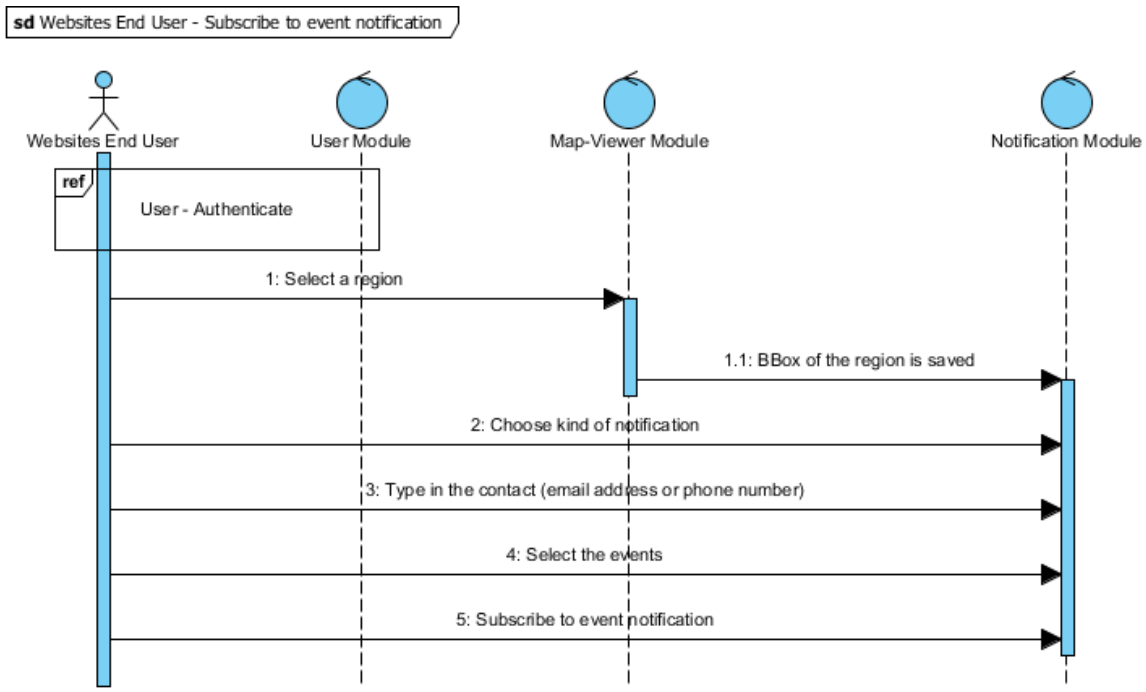

Figure 20 – Subscribe to event notification sequence diagram

3.5 Technology viewpoint

The technology viewpoint focuses on the choice of technology of the system. The main point to be considered at this stage of the project is that we target portals build on components compliant to the JSR 286 standard. All modules have then to be implemented using the portlets approach. Portlets have to be developed using JDK6 for JRE1.6 and have to be in compliance with JSR286 standard.

We have also to consider that all data, treatments have to be delivered online and have to be exposed on the web through OGC standards (WFS, WPS, SOS, …)

4 table of summarized requirements

In order to provide a synthetic requirements list as input for the next task of the work package, the following table summarized the main topics from section 3.

Id	Description
R_1	ENVISION Portal is a role-based system including users and roles management. Any ENVISION user can create an account and manage it.
R_2	ENVISION Portal is a community-based system allowing for creating communities. With regards to the pilots, a landslide community and an oil spill community are to be managed.
R_3	The website designer can create a new scenario website using available templates, compositions and portlets.
R_4	The website designer can export a scenario website to allow a deployment on another portlet container.
R_5	The composition designer can easily and graphically create new compositions, based on any existing WPS, WFS,WCS,SOS available online.
R_6	The composition designer can load/edit/store existing compositions
R_7	The catalogue manager can manage the CS-W catalogue.
R_8	The catalogue manager can publish/unpublish available resources.
R_9	The catalogue manager can search for resources in the catalogue, either by keywords, geographic, or semantic means.
R_10	The annotations manager can load ontologies related to the current domain.
R_11	The annotations manager can annotate any available web services
R_12	The annotations manager can load/edit/store existing annotations
R_13	The website end-user can interact with a map which is the basic interacting object. Standard interaction are provided; Layers management, Navigation tools (zoom, pan, ..), and querying management.
R_14	When required by the model, the website end-user can update the composition (model) parameters and inputs.
R_15	When allowed by the website designer, the website end-user can view the composition workflow and open separate viewers on intermediate outputs.
R_16	The website end-user can launch data series, and/or chart viewers from items on the map.

R_17	The website end-user can launch data series viewers, and/or chart viewers from items on the map.
R_18	The website end-user can subscribe to items that are graphically represented on the map (ex: sensors alert)
R_19	The website end-user can draw a section on the map and obtain the corresponding section view. (specific to oil spill scenario)
R_20	Any ENVISION Portal module (Portlets) conforms to JSR286
R_21	Any Data, Sensor, or Processing service is delivered online using OGC standards

Conclusion

This deliverable concludes the work made in task T1.2 in order to specify the user's requirements.

It describes the requirements of the system required to fulfil the scenario and pilots needs.

Using the RM_ODP framework, the different viewpoints are used to identify the requirements. This approach may be a first step toward the SERVUS methodology.

At the end, a table of requirements is provided as a baseline for the evaluation of the results of the project with regard to the ENVISION Portal.

In the next task T1.3 of the work package, the methods and plans for the validation of the results of ENVISION project will be prepared.

Appendix A: Project Terminology

The following table contains the description of the main current terms used in the project.

Term	Description
The Infrastructure ENVISION Infrastructure	Terms the hardware and software set up in order to fulfil the project objectives.
The Pilots ENVISION Pilots	Terms the demonstrations that will be set up during the project to illustrate the capabilities of the infrastructure.
The Portal ENVISION Portal Environmental semantic Web Portal	Describes the front-end application of the infrastructure
Website User Website Scenario Website	A Website produced by the portal for a dedicated purpose and a given community. During the project, we will produce dedicated Websites for the 2 pilots (landslide and oilspill).
Page	One particular view rendered in the browser is one page. One page consists of multiple modules.
Module	A user interface module is used to perform a certain task, or to convey a certain message. A module has a standard view, and can be configured through dialogs. Changing the content of module can (but does not have to) have an effect on the content of other components.
Service Web Service	Web services are typically a web API that are accessible via HTTP and executed on a remote system hosting the requested services
Dialogue	A Ddialogue is a new frame in the view (a pop up), which allows for additional interaction modes for the selected component.
Composition	The executable combination of existing web services. A Web service by itself.
Model Computer Model	Computer simulation of real world processes to make forecasts of a certain behaviour of natural phenomena.
Model as a Service (MaaS)	A model made available as a web service. A composition the user can interact with.

Appendix B: Project Acronyms

The following list contains the definition of the main current acronyms used in the project.

BPEL	Business Process Execution Language
BPMN	Business Process Modelling Notation
CAP	Common Alerting Protocol
CSW	Catalogue Service for the Web
EIP	Enterprise Information Portal
FTP	File Transfer Protocol
GCI	GEOSS Common Infrastructure
GeoDRM	Geospatial Digital Rights Management
GeoRSS	Geospatial RSS
GEOSS	Global Earth Observation System of Systems
GML	Geography Markup Language
HTTP	Hyper-Text Transfer Protocol
ICT	Information and Communication Technologies
INSPIRE	INfrastructure for SPatial InfoRmation in Europe
MasS	Model as a Service
OGC	Open Geospatial Consortium
Open-LDAP	Lightweight Directory Access Protocol
OWS	OGC Web Services
RM-ODP	Reference Model of the Open Distributed Processing
RSS	Really Simple Syndication
SEIS	Shared Environmental Information System
SISE	Single Information Space in Europe for the Environment
SoA	Service Oriented Architecture
SOS	Sensor Observation Service
SPS	Sensor Planning Service
UML	Unified Modelling Language
WCS	Web Coverage Service
WFS	Web Feature Service
WMS	Web Map Service
WPS	Web Processing Service
WSDL	Web Service Description Language

Appendix C: Document References

[1] ENVISION Project http://www.envision-project.eu/

[2] DOW Description of Work, cf FP7 Grant Agreement N° 249120, annex I

[3] D1.1 Definition of pilot cases

[4] D2.1 Environmental Semantic Web Portal Architecture specification

[5] D3.1 MaaS Composition Portal – Architecture specification

[6] D4.1 Ontology requirements analysis

[7] D5.1 Deployment of the OGC Catalogue

[8] D6.1 ENVISION Adaptative Execution Infrastructure – Architecture

[9] ORCHESTRA Project http://www.eu-orchestra.org/

[10] SANY Project http://www.sany-ip.eu/

[11] Services Oriented Design of Environmental Information Systems, by Thomas Uslander, KIT Scientific Publishing 2010.