CONSOLIDATED USER REQUIREMENTS

Deliverable D1.2.2

Circulation: PU: Public
Lead partner: FOMI
Contributing partners: IGN, Fraunhofer, CNR-IMATI-GE, Liguria
Authors: Bajjouk, Touria
Belényesi, Márta (co-editor)
Giachetta, Roberto
Képes, Attila
Klien, Eva
Kovács, Tamás
Kovács, Iván
Kristóf, Dániel (co-editor)
László, István
Lehoczki, Róbert
Millard, Kieran
Mikus, Gábor
Nguyen, Thai Binh
Olasz, Angéla
Palya, Tamás
Saur, Sébastien
Spagnuolo, Michela
Tancsik Ottó
Vass, Tamás

Quality controller: Ewald Quak
Version: 1.0
Date: 30.04.2013

DOCUMENT HISTORY

Version[1]	Issue Date	Stage	Content and Changes
1.0	30.04.2013		First release submitted to the Project Officer

[1]Integers correspond to submitted versions

EXECUTIVE SUMMARY

This deliverable documents the consolidation of the first set of requirements, carried out in the frame of Task 1.4. Requirements taken into consideration in this phase have been identified and documented in D1.1 “State of the Art Analysis” on one hand, and in D1.2.1 “Initial User Requirements” on the other hand. Nevertheless, as requirement consolidation is a continuous and iterative activity involving numerous project-internal and external actors, Task 1.4 is designed to run continuously during the first 30 months of the project. Further activities of requirement consolidation will also be documented in the online requirement management and documentation system, developed in the frame of Task 1.1.

The initial set of requirements contains overlapping, redundant and ambiguous elements; moreover, priorities have to be assigned to each requirement to make defining the development roadmap easier. Therefore, further specification and consolidation of requirements has to be carried out, in close collaboration with both the end-users and developers.

Altogether 139 User Stories (short natural language sentences of the intended functionality of a software system) have been analysed, of which 56 have arisen from D1.1 and 83 from D1.2.1. To set a clearer focus on the development directions and timing, filtering and prioritization of User Stories have been carried out based on several criteria (relevance to the project and to the test beds, importance to users, and availability of datasets).

Then, selected User Stories have been translated into Use Cases, following a formal structure (template) covering details such as preconditions, steps in a success scenario or input/output data, etc., to identify functional and non-functional requirements with the aim of presenting how the system would function from the users’ perspective. In this phase the two main aspects, namely “User” and “Developer” viewpoints, have been aligned based on discussions and iterations - “requirement consolidation” in its pure sense.

As the next step, the outcome of the above analysis was used to define “showcases”, i.e., basic sets of requirements to drive the early development of the infrastructure, prototypes and basic services in the first phase of the project. A Marine and a Land Showcase were elaborated, both containing a set of User Stories. Each of these showcases contains a synthesis of multiple original User Stories to simplify development work and clarify further tasks while keeping in mind specific user needs.

This document is part of the series of deliverables dealing with User Requirements (D1.2.1, D1.2.2, D1.2.3, and D1.2.4). It illustrates the process of gaining deeper mutual understanding of the two main areas through intensive dialogue, deepening and consolidating the requirements to reach a clear roadmap for the planning of user-driven developments within the IQmulus project.

TABLE OF CONTENTS

1 INTRODUCTION

This document is a working document, connected to the IQmulus Task 1.4 "Requirement Consolidation".

An initial set of requirements collected in previous stages of the IQmulus project were published in D1.1[2] and D1.2.1[3]. These requirements contain redundant and ambiguous elements. Moreover, priorities have to be assigned to each requirement to make defining the development roadmap easier. Therefore, specification and consolidation of requirements had to be carried out in close collaboration with both the end-users and developers. This process had to be complemented by the categorisation and prioritisation of requirements. The current document describes the dynamic procedure and the status of requirement consolidation, and introduces the results.

Identification of the requirements is being carried out by keeping in mind that the well-structured requirement specification must be usable for many Work Packages, especially for:

- WP2 "Infrastructure and design"
- WP3 "Data integration platform"
- WP4 "Processing services implementation"
- WP5 "Interactive visual decision support".

In order to achieve the best results, continuous consultations with the users and WP leaders/ members concerned were carried out in this phase of the project and a meeting in Darmstadt was organised (April 22&23, 2013), specifically to share and discuss the outcome of user story analysis/use case extraction among different (national and thematic) working groups and to carry out a consolidation and prioritization of use cases. During the preparation of the present deliverable, the following documents were taken into consideration (links are related to eRoom):

- Description of Work document (https://project.sintef.no/eRoom/math/IQmulus/0_2dadb) (in particular the IQmulus system architecture and design)
- D1.1 "State of the art analysis" (Annex 1: User Stories)
- IQmulus Use Case template (Annex 1of this document)
- Working papers, presentations and experiences of the Darmstadt meeting (https://project.sintef.no/eRoom/math/IQmulus/0_2f6e2)

As a continuation of this work, the still on-going collection and revision process of the requirements will result in D1.2.3 "Revised User Requirements" at the end of Project Month 18, and the requirement collection process will be concluded in Project Month 30, documented in D1.2.4 "Finalized User Requirements".

---

[2] D1.1 "State of the art analysis" Link in eRoom: https://project.sintef.no/eRoomReq/Files/math/IQmulus/0_2e93f/D1.1_State-of-the-art-analysis.pdf

[3] D1.2.1 "Initial User Requirements" Link in eRoom: https://project.sintef.no/eRoomReq/Files/math/IQmulus/0_2f5c2/D1.2.1_Initial_User_Requirements_Version_1.0.pdf

2 METHODOLOGY

The methodological description of Use Case and requirement identification in this chapter is organised around two main issues. First, the relation of the current task to the corresponding Work Packages and system architecture is pointed out. Next, the details of decomposing user stories to use cases are discussed further, following these steps:

1) Preliminary filtering and prioritization of user stories based on several aspects (further work is done on selected user stories).
2) Decomposing User Stories to Use Cases:
    a) Deriving Use Case diagrams from User Stories
    b) Elaborating atomic Use Cases (creating Use Case descriptions and activity diagrams).
3) Formulation of Showcases (Super User Stories) for the Land and Marine Scenario, in order to collect the most typical and most frequently occurring Use Cases and to illustrate the connections among them.

2.1 RELATION OF THE CURRENT TASK TO THE CORRESPONDING WORK PACKAGES AND SYSTEM ARCHITECTURE

As was referred to in the introduction of this document, the results of requirement consolidation form a key factor for WP2, WP3, WP4 and WP5 as an input, but WP6 is also involved indirectly. The relation of the current task to the corresponding work packages and the system architecture of IQmulus is shortly described in the next sections.

WP2 aspects

As described in the "Description of Work" (DoW), the WP2 goals, which are related to the WP1 work, are to design the components and interfaces of the IQmulus processing framework. One of the WP1 goals is to provide high-level requirements, clear component specifications and communication workflows to WP2 to drive the design of the infrastructure.

WP3 aspects

WP1 has the role of providing detailed user requirements to drive the development work carried out for the data integration platform in WP3. The WP3 goals related to WP1 can be summarised as follows:

1) Implementation of the infrastructure management tools (i.e., for workflow and deployment management) based on existing frameworks.
2) Implementation of the toolkits for the functional processing services.
3) Implementation of the required integration components for domain processing services.
4) Implementation of the components and facades.

WP4 and WP5 aspects

WP4 aims to develop the functional and domain processing services of IQmulus, while WP5 is responsible for a better support of users concerning visual decision making and interactive visual communication on large heterogeneous geo-spatial and temporal data sets. Therefore

developers must know exactly what the users want, which means that after collecting User Stories, Use Cases should be provided to the developers. The WP4 and WP5 tasks cannot be carried out properly without a detailed and simple use-case development, not to mention testing.

Aspects of System Architecture

The IQmulus system architecture as described in the "DoW" document (see page 79) has been divided into 3 layers: the Facade, Processing and Storage layers (see Figure 1).

- The Facade Layer is responsible for services published to users or application entry points.
- The Processing Layer is responsible for algorithms and processing libraries implemented by WP4 and WP5.
- The Storage Layer is responsible for storing and handling data.

In order to ease the implementation work of WP4 and WP5, WP1 intends to relate the received User Stories and corresponding Use Cases to the above-mentioned layers.

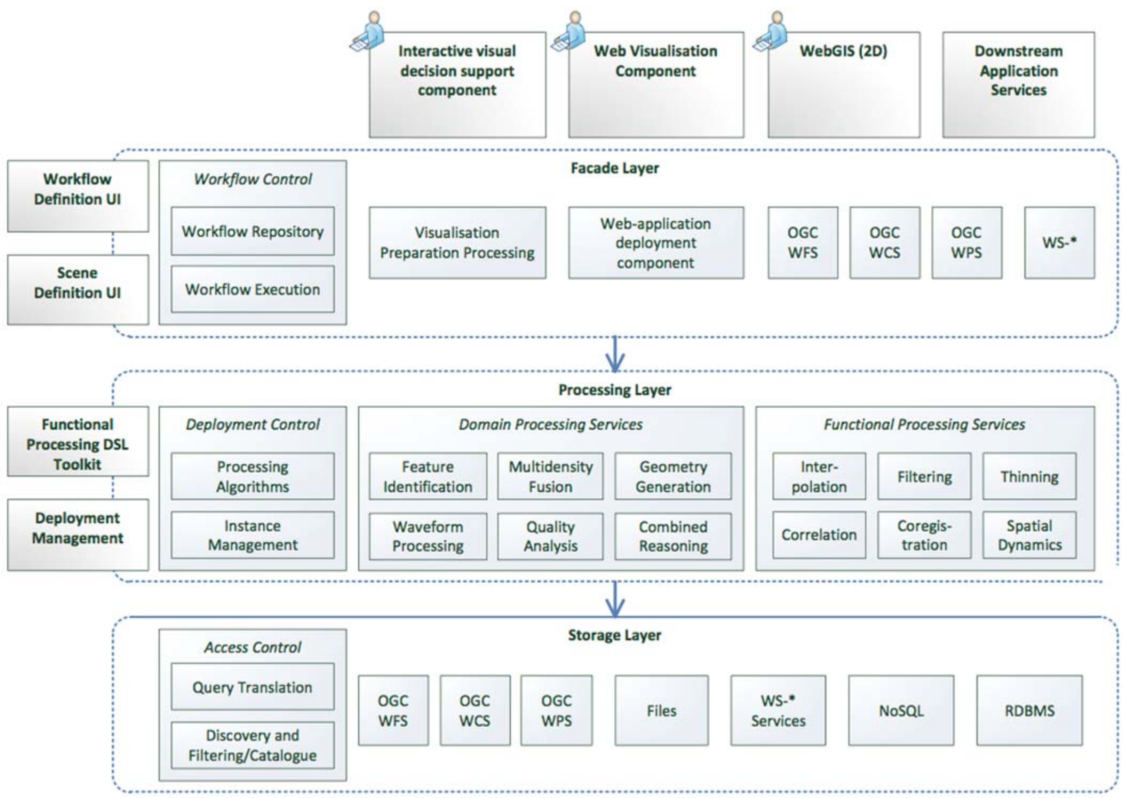


Figure 1. System architecture from Document of work

Facade Layer

This layer is related to WP5, for instance concerning interactive visual decision support techniques. Therefore, after grouping user stories, WP5 developers should:

- Be familiar with the relation of user stories to WP5;
- Prioritize implementation related to user stories;
- Know pre- and post conditions for user stories (as user stories may have dependencies).

Processing Layer

After relating Use Cases to the processing services, WP4 developers should know the:

- Communication pipeline between use cases;
- Data pipeline between use cases;
- Dependency among services;
- Communication interface for services;

Storage Layer

Before implementing a service, WP4 and WP5 developers must know

- which data format is used as input and output for a specific service;
- where data sets are stored;
- where to store intermediate result data.

Due to the complexity of the User Stories, one goal is to provide a specification for developers where they can identify which user story and adherent use cases are related to which service in the Processing and Facade Layer. Use Case consolidation and requirement specification has to proceed with caution because many requirements (and even Use Cases) will be similar in the Land and Marine Scenario, even if they are currently expressed in very different ways. During requirement consolidation it is also very important to place great emphasis on data flow between algorithms – this is particularly important for WP2 "Infrastructure and design" and WP3 "Data integration platform" developers.

2.2 FILTERING AND PRIORITIZATION OF USER STORIES

Preliminary filtering and prioritization of User Stories is necessary, and has to be carried out carefully, because the further work is only performed on the selected User Stories. Here, filtering was based on several criteria, partly based on WP1 aspects, and partly on WP4 aspects:

1) User Stories have to have relevance to the IQmulus project.
2) User Stories have to have relevance to the test beds foreseen in the validation scenarios (user stories not related to the (Land or Marine) scenarios and test beds should have lower priority; we can undertake finalization of such user stories later on).
3) User Stories have to have relevance to the developed Showcases (Super User Stories).
4) User Stories have to be of importance to the users.
5) Requested datasets have to be available.
6) User Stories have to provide scientific and technical challenges for the IQmulus project.
7) The development process has to be complex enough for the IQmulus project.
8) User stories, where the user only provides the necessary input (geodata and/or derived descriptive/statistical/etc. data), for high-level decision making should be filtered out. In this case the user will not carry out any further GIS data processing/analysis/service tasks to derive new data from the input. No use cases directly connected to GIS/RS tasks can be formulated from these User Stories.
9) User Stories formulated too generally or too complex should be filtered out. Extracting tasks from these is too complicated.
10) User stories with presumed permission problems (e.g., getting access to external proprietary databases) should be filtered out.

11) Tasks that may be carried out using other, easily accessible (commercial or open source) tools and do not require the sophisticated environment of IQmulus, should be filtered out.

12) Partially or fully overlapping User Stories – resulting in similar tasks – should be filtered out.

2.3 FORMULATION OF SHOWCASES (SUPER USER STORIES)

Resulting from extensive discussions among consortium partners, two Showcases (Super User Stories) have been created by grouping and aligning multiple User Stories synthesized from input received thus far. Each Showcase is intended to collect User Stories and their related Use Cases from the Land and Marine test bed, respectively. Preliminary Use Case diagrams and activity diagrams have been produced by project partners, as a result of the extraction process some of these diagrams are being integrated into the Super User Stories' specifications.

Super User Stories are regarded as a good starting point for the system architecture and development teams to analyse. Continuous refinement and clarification will be made when needed.

2.4 DECOMPOSING USER STORIES TO USE CASES

What is a use case?

A USE CASE IS A DESCRIPTION OF THE POSSIBLE SEQUENCES OF INTERACTIONS BETWEEN THE SYSTEM UNDER DISCUSSION AND ITS EXTERNAL ACTORS, RELATED TO A PARTICULAR GOAL.

A USE CASE CAPTURES A CONTRACT BETWEEN STAKEHOLDERS OF A SYSTEM ABOUT ITS BEHAVIOUR. THE USE CASE DESCRIBES THE SYSTEM'S BEHAVIOUR UNDER VARIOUS CONDITIONS AS THE SYSTEM RESPONDS TO A REQUEST FROM ONE OF THE STAKEHOLDERS, CALLED THE PRIMARY ACTOR.

From "Writing Effective Use Cases" by Alistair Cockburn (Addison Wesley 2001)

The process of defining and writing a use case can be divided into four stages:

1) ACTORS AND GOALS. Identify actors participating in current user story. Determine their goal against the software. Review this list.
2) MAIN SCENARIOS. Collect the major scenarios across specified user stories, those which appear in almost every user story. Main scenarios make sure that the system will deliver interests and values to partners participating in the project.
3) RAINY DAYS. Brainstorm all the failures that could occur. Review this list, without thinking on how to handle them.
4) FAILURE HANDLING. Specify how you would expect the system should behave to each failure. Be prepared that this task may take most of the time, due to different user opinions on handling failures.

Steps

1) PRODUCTIVITY. Be productive without perfection. Don't be afraid to make mistakes at first. Use cases will be refined gradually. You can always refine more later on. Note that use cases are made for humans.

2) ACTORS. Get to know your actors. Pick primary and non-primary actors for the use case.
3) SUNNY DAYS AND RAINY DAYS. Define your sunny day and rainy day scenarios. Hope for the best and prepare for the worst.
4) RECYCLE. Identify reuse opportunities for use cases. Make a noticeable sign on reusable use cases.
5) OVERVIEW. Create a use case overview table. An overview table may consist of (UC#, UC_name, primary actor, complexity)
6) DESCRIBE. Name and describe first. First step to identify your use case.
7) MAIN FLOW. Define the primary flow of use cases. The workflow represents the goal of your user story.
8) FAILURE FLOW. Define the rainy day workflow. Don't forget about failure handling.
9) USE CASE DIAGRAM. Make a use case diagram.

Use case levels

Depending on their level of complexity use cases can be categorized into levels as

1) CLOUD LEVEL – High level summary

   Very high level, involves multiple user goals.

2) KITE LEVEL – Summary

   This high level takes several hours to specify involving complex steps.

3) SEA LEVEL – User goal

   Something the actor is trying to get done, may involve several fish-level and clamp level use cases.

4) FISH LEVEL – Sub-function

   Usually a reusable or frequently used use case. For example: selecting the extent of interest area.

5) CLAMP LEVEL – Very low (programming language details)

   Not usually written up in a use case, due to its simplicity or triviality.

During use case specification, occasionally use cases are too complex or not obvious at first. Due to the nature of use cases, they are refined gradually; therefore at first set the level to kite or sea level, as specification evolves the level of detail will lower gradually.

For example, in User story #22 – Use case "Extract building contours", there is a sub-use-case "data fusion". It is very complex and not clarified at first; so set its level to sea-level to inform designers and developers that the specification is not final.

Use case attributes

Many attributes are available to specify a use case; hereafter the most important attributes will be defined.

By definition a use case is a list of atomic steps defining interactions between an actor and the system to achieve a goal. Notice that we are still working with a high-level specification, therefore do not bother with the following aspects:

- User interface
- Implementation level requirements
- Sometimes performance issues are neglected; only specify those if they are crucial.

To sum up the major steps of high-level specification:

1) Initialization: User stories have been collected.
2) Primary goal: Prioritization of user stories has been decided.
3) Sub-function: Decomposition of user stories into use cases.

Deriving use case diagrams

A use case diagram (UCD) can be used to describe system or sub-system functionality. A UCD shows all the available functionalities and their interactions. A UCD does not show the details of the use case: it only summarizes some of the relationships between use case, actors, and system. In particular, the diagram does not show the order in which steps are performed to achieve the goals of each use case; therefore it is important to note that UCDs are different from workflow and sequence diagrams. Present use cases only deal with functional requirements for a system. Other requirements like business rules, security, which are non-functional, should be described in the non-functional requirements section.

Major components

1) Actors who will interact with the system.
2) Use cases or services.
3) Lines representing relationships between use cases or services.

How to draw a use case diagram

After actors, use cases, services and relationships among them have been identified, the following steps are available:

- Name your UCD. It should identify the functionalities performed.
- Position your actors in an organized manner.
- Relationships: use only those which are needed, do not introduce unnecessary ones.
- Use notes when you want to clarify ambiguous points in the UCD.

Basic relationships in the use case diagram

1) COMMUNICATES. Participants are actor and use case. Notation: <<communicates>>
2) EXTENDS. One use case may add functionality to another use case under certain circumstances. The arrow should point to the main, extended use case.

Figure 2. Basic relationships in a use case diagram: EXTENDS

3) INCLUDES. Shows that one use case describes some of the details of another. Each of the included more detailed use cases is a step that the actor(s) might have to perform to achieve the goal. The arrow should point at the more detailed, included use case.

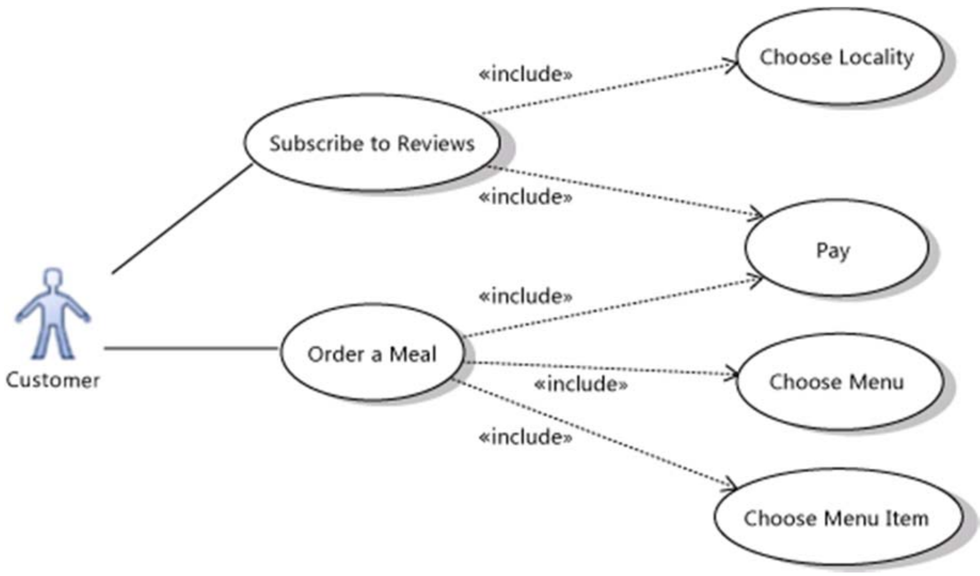


Figure 3. Basic relationships in a use case diagram: INCLUDE

4) GENERALIZATION. Shows that a specialized use case has been derived from a general use case to achieve the goal. The arrow should point at the more general use case.

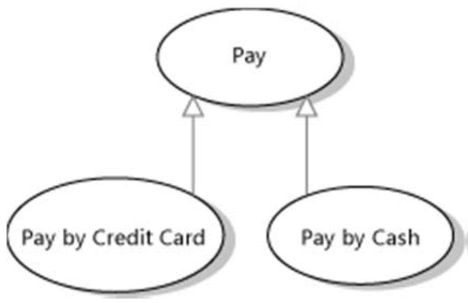


Figure 4. Basic relationships in a use case diagram: GENERALIZATION

Aspects of decomposition and elaboration of use cases

1) Establishing links to pre-defined components.
2) Elaboration of atomic use cases, where the following must be added:
   a) Identify source of input data, e.g.:
      i) From WMS (Web Map Services) or WFS (Web Feature Services),
      ii) From database, spatial database,
      iii) From file(s).
   b) Identify output data/result as done for input data.
   c) Specify if there is any temporary data needed or produced.
   d) Identify which component in the Processing Layer (see Figure 1) corresponds to the use case.
   e) Identify dependency of use cases, if there is any.
   f) Identify pre- and post-condition(s) for each use case.
3) Specify data travelling direction of each user story. A user story may involve more than one Processing Layer component, therefore WP2, WP3, and WP4 must know which component is the most relevant to develop first.
4) Deriving requirements.

The result is the following diagram:

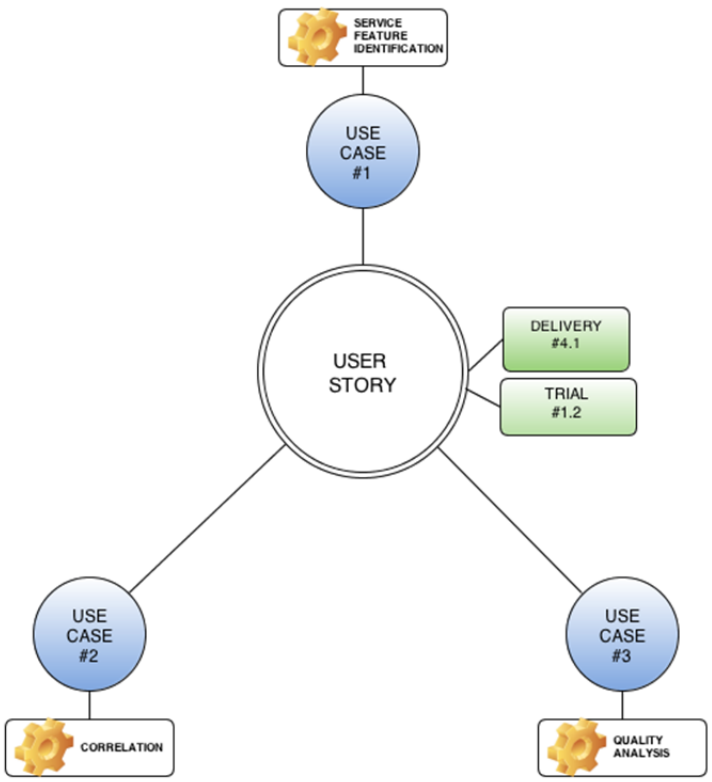


Figure 5. Decomposing user stories into use cases

Example of a use case diagram

The example presented here was provided by FOMI. The diagram is derived from the following User Story:

"As a RS expert I want to compare the automatically classified building contours based on orthophoto, Digital Elevation Model and Digital Surface Model with the building boundaries in the cadastral database, so that I can verify the adequacy of the cadastral records."

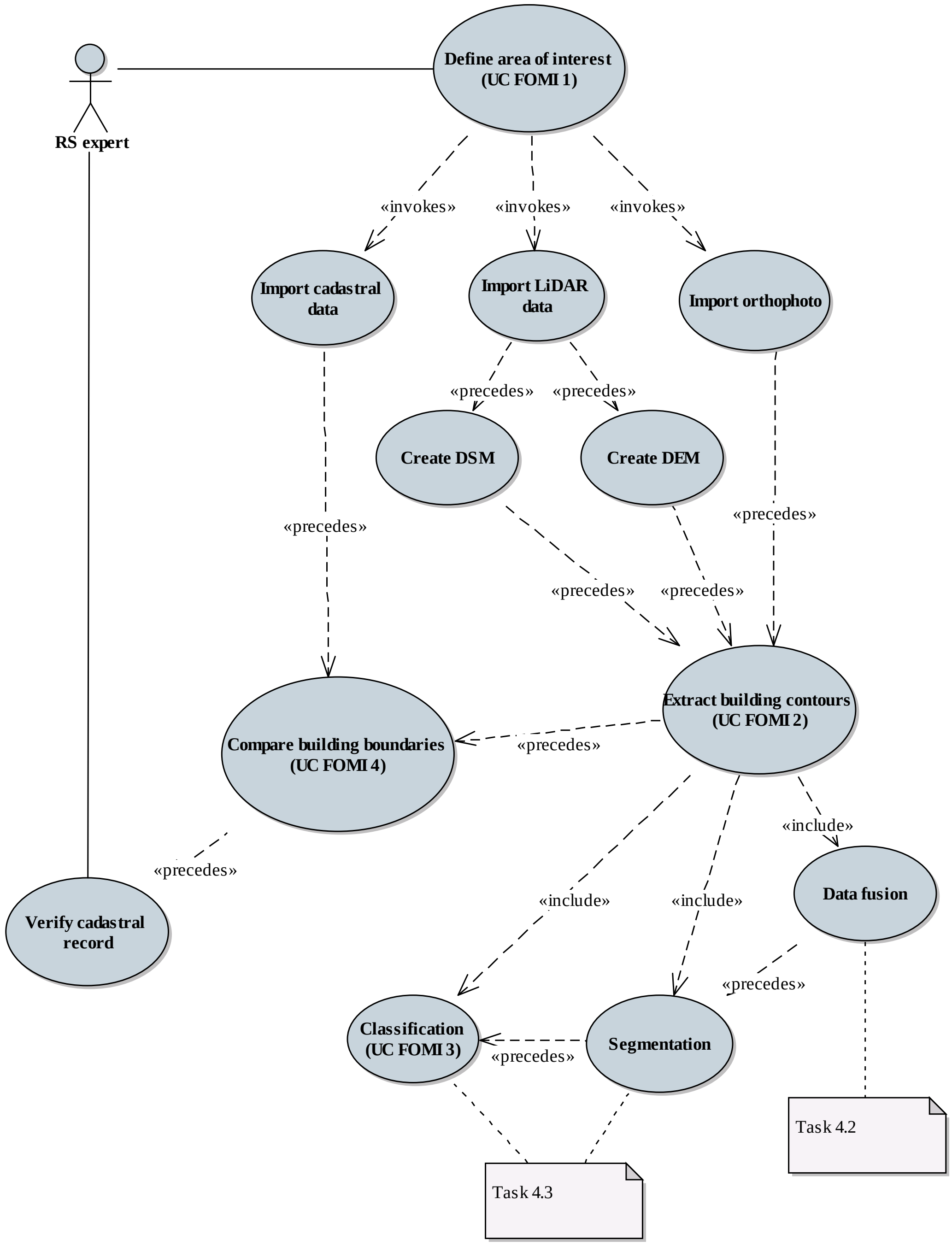


Figure 6. Use Case Diagram (example)

Elaborate atomic use cases (use case description and activity diagram)

What is an activity diagram?

An activity diagram shows a software process as a workflow through series of actions. Participants are actors, software components or computers. Activity diagrams are used to describe:

- Workflow between actors and system.
- Steps performed in a use case.
- Protocols describing sequences of interactions between software components.
- A specific algorithm.

Use case diagram elements

Simple workflows may consist of the following elements.

1) ACTION. A step in an activity flow, where software or user perform a task.
2) CONTROL FLOW. Shows the flow of control between actions.
3) INITIAL NODE. First action(s) in the activity.
4) FINAL NODE. End action(s) in the activity.
5) DECISION NODE. Conditional branch in the activity flow.
6) GUARD CONDITION.

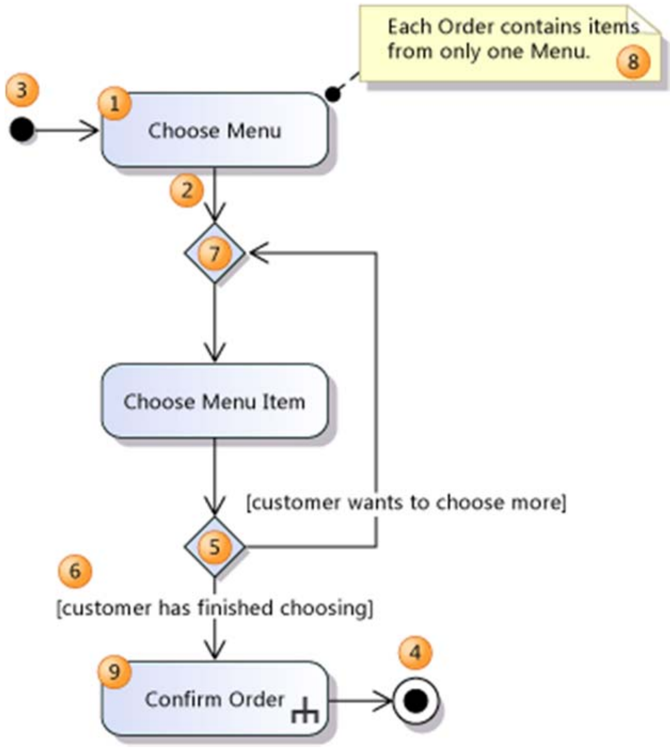


Figure 7. Simple workflow I.

For a concurrent workflow, the following elements are introduced in order to describe concurrent actions:

1) FORK NODE. Dividing a single flow into concurrent flows .

2) JOIN NODE. Combines concurrent flows into a single flow.

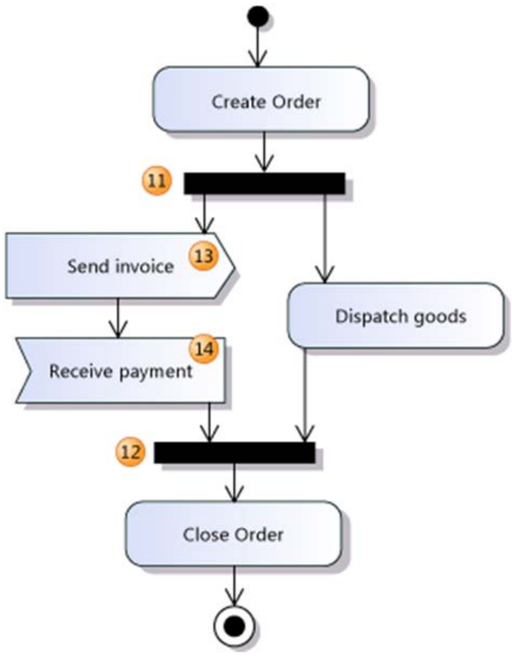


Figure 8. Simple workflow II.

For data flow describing input and output data, the following elements are introduced:

1. INPUT PIN. Represents data that an action can receive when executed.
2. OUTPUT IN. Represents data that an action produced after having been executed.

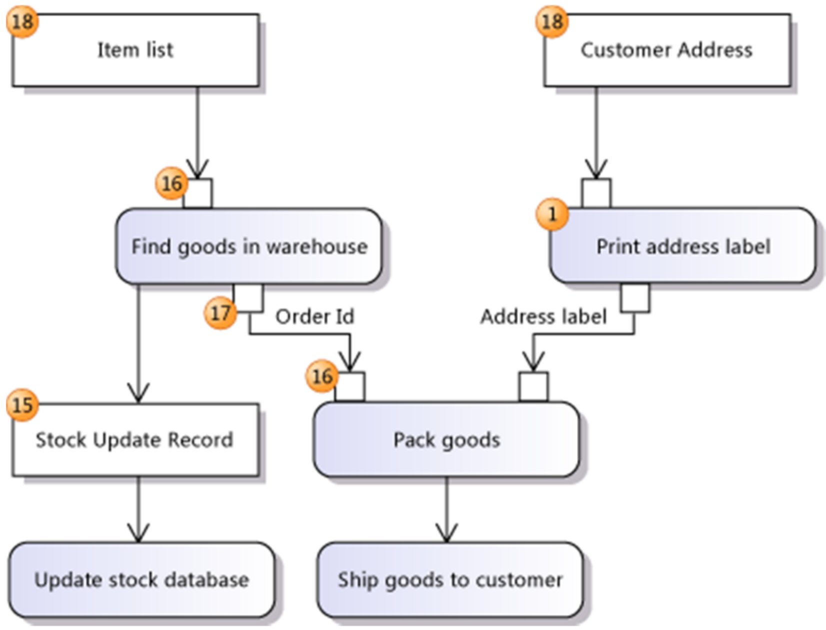


Figure 9. Simple workflow III.

Example for an Activity Diagram

The example presented here was provided by FOMI. The diagram was derived from the Use Case: Extract building contours (UC FOMI 7).

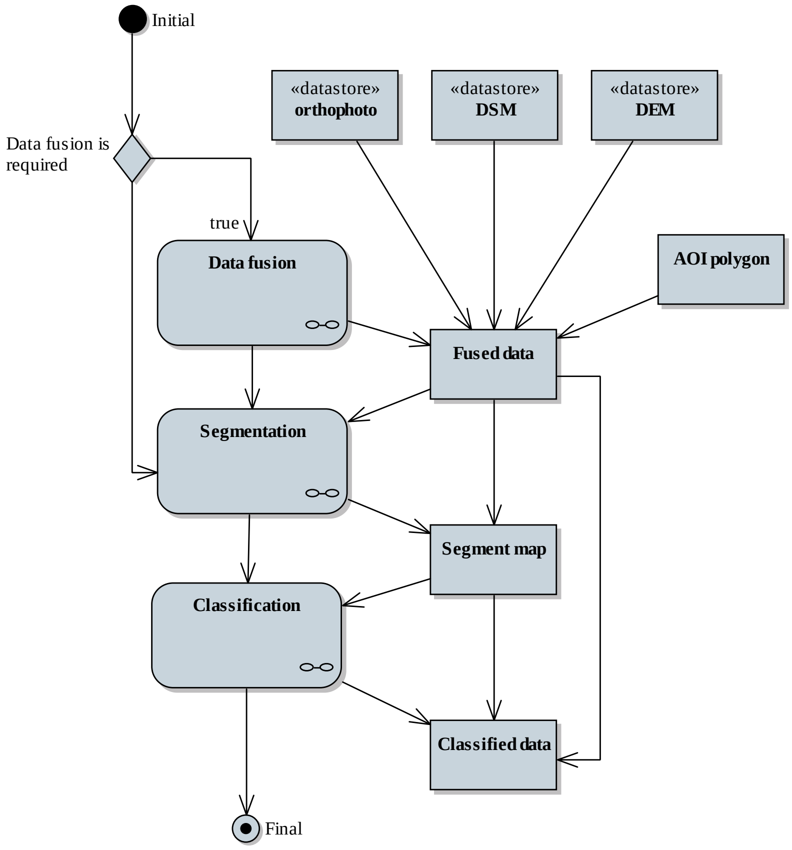


Figure 10. Activity Diagram (example).

2.5 DOCUMENTATION

The present Deliverable 1.2.2 contains the results arising from Task 1.4, namely:

- The results of the prioritization of User Stories collected in the previous and current phases of WP1 (Task 1.2, Task 1.3).
- The Showcases formulated related to the two main test beds (Land and Marine), with their description and components.
- Use Case Diagrams connected to different User Stories where available.
- Use Case descriptions, requirements and activity diagrams as provided by the IQmulus partners in relation to WP1. It is also important to mention that many Use Cases were also examined from a WP4 perspective. Results of the latter approach are included in Deliverable 4.1.1 of WP4.

Methodological choices have been described and shared by all IQmulus partners involved in WP1 through two main documents: the Guidance Document and the Questionnaire (documented in D1.2.1 "Initial User Requirements".

IQmulus Use Case template

Use Case descriptions and requirements were documented as it was pre-defined in the "Use Case and Requirements" template, which can be found in Annex 1 of the current document, and in eRoom:
https://project.sintef.no/eRoomReq/Files/math/IQmulus/0_2f6de/IQmulus%20USE%20CASE+REQ%20template.doc

2.6 ORGANIZATION

The four local boards previously created have been involved in the user requirement consolidation process. Each board is led by an IQmulus partner who is in charge of requirement consolidation, namely: prioritization of User Stories and extraction of Use Cases according to the methodology. Local board leaders have also profited from their direct links to external local/national users during this process.

The four local boards are the following:

- Hungary, leader: FOMI
- Italy, leader: Liguria (with CNR-IMATI)
- Marine applications, leader: UBO and IFREMER (with HRW)
- France, leader: IGN.

2.7 REQUIREMENT CONSOLIDATION PROCESS

During the requirement consolidation process, the online requirement management system was used extensively. Nevertheless, some steps in the work were easier carried out via documents, uploaded as files in eRoom, the central project documentation system.

This deliverable incorporates all relevant documents as annexes. The outcome of the requirement consolidation, once ready to be fed directly into the development process, is being uploaded to the Redmine requirement tracking system for task distribution and progress monitoring.

3 RESULTS

3.1 PRELIMINARY FILTERING AND PRIORITIZATION OF USER STORIES

During filtering, User Stories were put into the following categories:

- "Very High" priority was given to User Stories selected to be included in Showcases;
- "High" priority was given to User Stories selected as being relevant for use case extraction at this phase;
- "Related" priority was given to User Stories strongly related to those selected in one of the first categories;
- "Later" category was given to User Stories to be addressed in later stages of the project.

The table below shows statistics for the User Story selection at this stage of the project:

	Very High	High	Related	Later
1.1 State of the Art Analysis	**7**	**9**	**1**	**39**
1.2.1 Initial User Requirements	**13**	**13**	**2**	**56**
Total	***20***	***22***	***3***	***95***

3.2 SHOWCASES (SUPER USER STORIES)

IQmulus has adopted an agile type approach to its development. It has coined the phrase 'showcase' to represent the outcomes of a particular development iteration. The term 'showcase' was used because we wanted to make sure each development iteration had something we could show to users, particularly in early iterations where the users may not fully appreciate how developments are aligned with their needs.

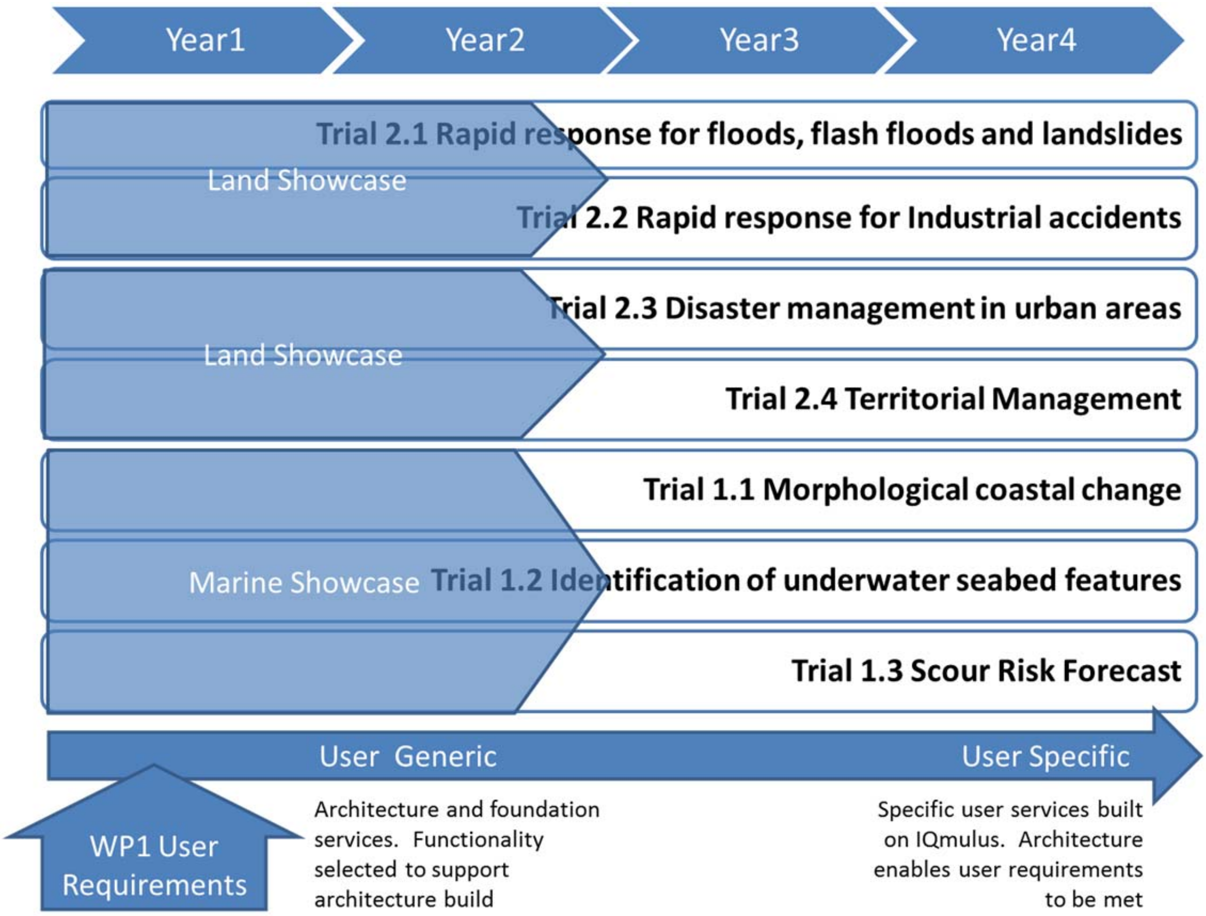
In the Description of Work it was stated: "Within the IQmulus project, two test beds are built up to evaluate the approach and resulting technology in real-world scenarios". These are based on 'as is' scenarios that could be improved by IQmulus and are user-driven based on the users in IQmulus. Importantly the consortium has the data and models for these test beds and to operate the 'as is' workflows for these test beds.

The two test beds and associate trials in the DoW are as follows:

- Land
  - Trial 2.1 Rapid response for floods, flash floods and landslides
  - Trial 2.2 Rapid response for Industrial accidents
  - Trial 2.3 Disaster management in urban areas
  - Trial 2.4 Territorial Management
- Marine
  - Trial 1.1 Morphological coastal change
  - Trial 1.2 Identification of underwater seabed features
  - Trial 1.3 Scour Risk Forecast

The test beds are labelled as 'marine' and 'land' but this is simply a label related to the user domain they refer to. There is clearly too much complexity to deliver an IQmulus solution for all of the trials all at the same time, so for this reason the concept of showcases was realised that would implement the key (generic) functionality applicable to each of the test beds in the early stages of the project. The three showcases could enable technical development to be focused for Years 1 and 2 with its

aim to deliver a fully working prototype for the Year 2 review. In subsequent years the functionality in IQmulus would be scaled to meet more specific functionality for each of the test beds identified in the proposal. The scope of the three showcases for Year 1 was informed by the outcome of WP1 (user requirements).



The relationship between the trials in each of the test beds and the associated showcases.

To demonstrate the most typical and most frequently appearing Use Cases, requirements and processes related to the two main test beds (Land and Marine), integrated (Super) User Stories were formulated from the original User Story database. These Showcases will be analyzed in detail by Use Case derivation and description.

Two ***Land Showcases*** were formulated. ***Land Showcase 1*** includes the first phases of disaster management in case of a flood – the preparation phase where the main task is monitoring the flood risk. The ***Land Showcase 2*** formulates the reaction phase and the assessment phase where the task is the assessment of flood damage. The ***Marine Showcase*** focuses on deriving an accurate elevation model from heterogeneous data sets to support further analysis and processing tasks. This showcase covers three main processes: (1) topographic and bathymetric LiDAR (or SONAR) point clouds co-registration, (2) integration of topographic and bathymetric point clouds and existing Digital Terrain Models (DTM) / Digital Surface Models (DSM) with sensor sonar data of the coast line and seafloor in a unique surface model, and finally (3) sea/land classification and shoreline detection based on combined datasets.

These three showcases, formulated in Project Month 6, represent a non-exhaustive synthesis of important issues that appeared repeatedly in User Stories, as documented by the list of relevant User Stories for each showcase. As a Month 12 update concerning these Showcases, additional tables have been provided concerning the relation of steps of these Showcases to services developed in Work Package 4[4] in the first year or foreseen for the second year as well as describing the involved data types and available sample data sets and their typical sizes.

Note, however, that a considerable number of services considered in Work Package 4 are multipurpose services, functional services as named in the DoW, which are necessary basic building blocks for composite tasks and which arise in several Showcases and also other User Stories. Consequently a list of first-year multi-purpose services is included and also another list of those foreseen for the second project year.

As mentioned above, the three Showcases, as indicative as they may be of the users' typical list of desiderata, can only be initial but not exclusive guidelines for the implementation efforts of IQmulus. To illustrate this, formulations of three complementary showcases have been included as well.

They form an example of considerations by the developers that may add to or subtract from the original user requirements. Not only may there be research results beyond the state of the art in the consortium that open up opportunities that users are simply not that aware of. On the other hand some issues that are typically perceived by a user as being hard may be tractable by standard, possibly commercial, software and should not be further addressed in this research project. It is also crucial to identify where the handling of "Big Geospatial Data" by IQmulus supposedly provides the biggest impact.

The IQmulus position concerning the interplay of user and developer requirements will be addressed in the revision of the requirements due in Month 18 (April 2014) in the Deliverable *D1.2.3 Revised User Requirements,* which will provide the definitive setting for the first prototype implementation due in Month 24.

[4] Detailed service descriptions are included in the two toolkit deliverables already submitted (D4.2.1 and D4.3.1) or are forthcoming in the two that are due in Month 18 (D4.4.1 and D4.5.1)

The Land Showcase (1.2.2_SC2) and corresponding User Stories

User Story 1.2.2_SC2_1

I want to extract and visualise features relevant to flood prediction and monitoring (dams, break lines - ridges, channels, drainage network, etc.) and compare them to previous situations based on the „best" representation of the topographic surface (calculated from LIDAR point clouds, existing DTM, DSM) of an area of interest to produce high-quality input for hydrologic modelling and simulation, along with data related to the source of flooding (dam break, precipitation fields, etc.).

The User Stories strongly related to this Showcase are as follows:

- 1.1_52: As a hydrologist, I want to extract significant break lines from high resolution DTMs or point clouds, such as dikes, dams, banks, ditches so that I can improve my hydraulic model and make more accurate simulations.
- 1.2.1_62: I want to create an automatic dam recognition algorithm (automatic feature extraction, eCognition) especial for dam break lines, to be able to monitoring the water height and the dam height differences.
- 1.2.1_64: As hydrologist supporting the decision maker in civil protection I would like to use radar reflectivity field to interpolate observed rainfall data from traditional rain gauge network (punctual data)
- 1.2.1_65: As hydrologist supporting the decision maker in civil protection I would like to have a shape file of the Ligurian small basins (< 10 km2) derived from a high resolution DTM exploiting the available Lidar data of Regione Liguria to derive the DTM and a cut interpolated rainfall field over the shape file.
- 1.1_54: As a hydrologist, I want to combine flood simulation models with all kind of topographic data in order to know the flooded area and all the human and economic stakes impacted so that I can provided decision makers with accurate and reliable information.
- 1.1_33: As a GIS expert, I want to delineate the extent and the maximum possible extent of the area potentially flooded in case of an industrial disaster (e.g. dam burst of a waste reservoir) so that I can provide the polygons of the risk area to the disaster management authorities.
- 1.2.1_16: I want to get an overview map of areas affected potentially by flood, in order to make the best decision for minimize the potential impacts of the disaster.
- 1.2.1_31: I want to interactively modify a water level predicted height and see the affected land parcels on the cadastre so that I can expose maps of submersion areas
- 1.2.1_67: I would like to have easy tools to integrate up-to-date data about hydraulic, hydro-geological, seismic, industrial risk for the production of multi-risk maps
- 1.1_53: As a hydrologist, I want to combine flood simulation model with high resolution DTM so that I can publish a water height map on the flooded area.

RELATIONS TO IQMULUS COMPONENTS/SERVICES

Step	Related IQmulus components and/or services
Step 1: The user defines the area of interest	[generic UI functionality, Workflow Editor]
Step 2: The user identifies the input datasets to be used	[generic UI functionality, Workflow Editor]
Step 2A: The use selects a temporal interval for the rain fall data	generic system I/O functionality, IQmulus Hub]
Step 3: The system computes the 'best' representation of the topographic surface	[generic system I/O functionality, IQmulus Hub] If the data are not pre-processed: Service 10 (T4.3, Y1): Outlier classification in point clouds Service 34 (T4.3, Y2): Attribute-based segmentation of the point clouds Service 14 (T4.2, Y2): Coarse registration Service 6 (T4.2, Y1): ICP registration Service 2 (T4.3, Y1): Point cloud to regular grid Service 9 (T4.4, Y2), 55 (T4.4, Y2): Generate spline surface from parameterized point cloud Service 38 (T4.4, Y2): Local refinement of DEM with higher resolution elevation data Service 48 (T4.4, Y2): Multi-resolution triangulation Service 49 (T4.4, Y2): Constrained triangulation
Step 4: The system extracts features (dams, breaklines, ridges, drainage network...)	[generic system I/O functionality, IQmulus Hub] Service 4 (T4.3, Y2): Breakline extraction Service 42 (T4.3, Y1): Detection of flow lines and drainage basins Service 59 (T4.3, Y2): Multi-object classification of 3D point clouds
Step 5: the system computes the precipitation field	[generic system I/O functionality, IQmulus Hub] Service 40 (T4.4, Y2): Approximation of rainfall data Service 58 (T4.4, Y2): Spline representation of rainfall data or other associated data Service 67 (T4.4, Y2): Approximation of rainfall data with radial basis functions
Step 7: The system detect critical points and changes	[generic system I/O functionality, IQmulus Hub] Service 44 (T4.3, Y1): Extraction of critical points from grids or

	triangulations. Service 45 (T4.5, Y2): Topological analysis of distance maps for co-registered surfaces
Step 8: The system stores the resulting features along with quality maps	[generic system I/O functionality, IQmulus Hub]
Step 9: the user visualizes extracted features	[generic system I/O functionality, IQmulus Hub] [generic & IQmulus visualization functionality]

FIRST-YEAR SERVICES INVOLVED

#	Name	Input	Output	Toolkit
2	Point cloud to regular grid	point cloud (LAS)	gridded point cloud (GeoTiff)	4.3
6	ICP registration	point cloud (LAS)	array (.mat)	4.2
10	Outlier Clasification in Point Clouds	point cloud (LAS)	enriched point cloud (LAS)	4.3
42	Detection of flow lines and drainage basins	gridded point clouds (LAS, GeoTiff)	vector data (Shapefile) and gridded point clouds (GeoTiff)	4.3
44	Critical Points	triangle mest (PLY) or gridded point cloud (GeoTiff)	3 vector data (Shapefile), one for each dimension of criticalities	4.3

SECOND-YEAR SERVICES INVOLVED

#	Name	Input	Output	Toolkit
4	Breakline Extraction	point cloud (LAS)	polygons (ShapeFile)	4.3
9	Spline surface from parameterized point cloud	point cloud (LAS)	Spline surface. g2-format (internal SINTEF format) or iges (for B.spline surfaces)	4.4
14	Coarse registration	point cloud (LAS)	array (.mat)	4.2
34	Attribute-based Segmentation of point	point cloud (LAS)	enriched point cloud (LAS)	4.3

	clouds			
38	Local refinement of DEM with higher resolution data	enriched point cloud (LAS), DEM (GeoTiff)	Gridded point cloud (GeoTiff)	4.4
40	Approximation of rainfall data	enriched point cloud (LAS), DEM (GeoTiff)	Gridded point cloud (GeoTiff)	4.4
45	Topological Change Detection	Gridded point clouds (GeoTIFF) or triangulations (ply)	2D array and a scalar parameter; Vector data (Shapefile)	4.5
48	Multi-resolution triangulation	triangulation (PLY)	to be defined	4.4
49	Constrained triangulations	point cloud (LAS) lines, polygons (ShapeFile)	triangulation (ply), polygons (ShapeFile)	4.4
55	Generate spline surface from point cloud Modified to handle a gridded point cloud	gridded point cloud (GeoTIFF)	Spline surface (g2 or iges if possible)	4.4
58	Spline representation of rainfall data	enriched point cloud (LAS) and gridded point cloud (GeoTIFF)	Spline surface (g2 or iges)	4.4
59	Multi-object classification of 3D point clouds	point cloud (LAS)	enriched point cloud (LAS)	4.3
67	Approximation of rainfall data with radial basis functions	enriched point cloud (LAS), DEM (GeoTiff)	Gridded point cloud (GeoTiff)	4.4

DATA INVOLVED

Data involved	**Data size, complexity**
Punctual observed rainfall data	Liguria Region: * ARPAL dataset: historical rainfall data over Liguria Region [time-varying data set, overall size 10Tera, one-day measures account for about 2MB, .txt] * Genova Municipality dataset: rainfall data over the municipality of Genova [time-varying data set, overlall size 1GB, one-day measures account for about 1,5MB, .txt]
Lidar data	Liguria: 85% coverage of the Liguria region, raw Lidar data

	(500Gb, size depends on point cloud density. For the testbed, 60 Mb/sqkm @ 2 pts/sqm density, .LAS]
Low resolution Digital Surface Model	Liguria: 2.8 GB (5x5m resolution), Geotiff]
Surface Models of the Genova Municipality	Cartographic data from Genova Municipality (1:1000 for urban area, 1:2000 elsewhere, Shapefile)

User Story 1.2.2_SC2_2

I want to quickly delineate and categorize flooded areas (and landslides) based on satellite/aerial imagery (optical and/or radar) and DTM and combine it with existing spatial datasets (roads, population, cadastral map, economical stakes, etc.) so that I can provide decision makers with information and maps for damage assessment.

The User Stories strongly related to this Showcase are the follows:

- 1.1_30: As a GIS expert I want to generate a categorized waterlog map (with at least the following categories: open water surface, flooded vegetation, wet soil) in a particular area in order to provide data for a quicker and more reliable agricultural damage assessment and yield loss estimation.
- 1.2.1_18: I need to get the boundary-polygon of the affected area, and the location of the dam break on a quick and automated way, in order to provide the overview map and the basic statistics for the decision-maker in case of a disaster.
- 1.2.1_13: I have to generate a categorized inland inundation map in order to provide data for agricultural field experts and office experts.
- 1.1_28: As a field expert in disaster management, I want to perform real-time updating of road status attributes in an emergency situation (flooding or high-level inland excess water) on my mobile device so that I can make route planning by taking into account the flooded areas or other barriers.
- 1.1_32: As a Remote Sensing expert I want to receive answers about the damage of a land parcel (natural: water, drought, hail, disease; damage caused by game/ animals; agrotechnical errors), in order to be able to estimate the scope and the extent of the damage.

RELATIONS TO IQMULUS COMPONENTS/SERVICES

Step	Related IQmulus components and/or services
Step 1: The user defines the area of interest	[generic UI functionality, Workflow Editor]
Step 2: The user identifies the input datasets to be used	[generic UI functionality, Workflow Editor]
Step 3: The user identifies the reference (training and test) data containing geometries and (possibly hierarchical) thematic attributes about known flooded areas and landslides	[generic UI functionality, Workflow Editor] If the data are not preprocessed, Service 72 (T4.2, Y2): Image registration Service 73 (T4.2, Y2): Preprocessing of raster data are used.
Step 4: The user identifies ancillary spatial datasets (e.g. roads, land parcels, administrative boundaries, etc. - previously made available in the system) and determines features (e.g. spatial relations, statistics) to be calculated and visualized	[generic UI functionality, Workflow Editor]
Step 5: The system loads input and reference data	[generic system I/O functionality, IQmulus Hub]
Step 6: The system computes indices and metrics (NDWI, NDVI, NDSI)	Service 76 (T4.3, Y2): Computation of Spectral Indices

Step 7: The system determines thresholds and classification parameters by optimisation based on reference data	Service 77 (T4.3, Y2): Raster segmentation Service 78 (T4.3, Y2): Clustering of raster data Service 79 (T4.3, Y2): Thematic classification of raster data
Step 8: The system carries out classification	Service 77 (T4.3, Y2): Raster segmentation Service 78 (T4.3, Y2): Clustering of raster data Service 79 (T4.3, Y2): Thematic classification of raster data Service 80 (T4.3, Y2): Waterlog detection on raster data
Step 9: The system calculates desired features (spatial relations, statistics, etc.) by combining results with ancillary spatial datasets	Service 74 (T4.2, Y2): Topological analysis on 2D data sets
Step10: The system stores and visualizes the resulting features along with quality maps	[generic system I/O functionality, IQmulus Hub] [generic and IQmulus visualization functionality]

SECOND-YEAR SERVICES INVOLVED

#	Name	Input	Output	Toolkit
72	Image Registration	Raster image (GeoTiff) without spatial reference	Raster image (GeoTiff) with spatial reference	4.2
73	Preprocessing of raster data	Raster image (GeoTiff)	Preprocessed raster image (GeoTiff)	4.2
74	Topological analysis on 2D data sets	Raster images (GeoTiff) or vector data (Shapefile)	Processed raster and vector data, or topological relations.	4.2
76	Computation of Spectral Indices	Raster image (GeoTiff)	Index raster image (GeoTiff)	4.3
77	Raster segmentation	Raster image (GeoTiff)	Segmented raster image (GeoTiff) and/or vector segment data (Shapefile)	4.3
78	Clustering of raster data	Raster image (GeoTiff) and reference data	Clustered raster image (GeoTiff) and/or vector claster data (Shapefile)	4.3
79	Thematic classification of	Raster image (GeoTiff) and reference data (training field	Classified / thematic raster image (GeoTiff)	4.3

	raster data	and control area) with categories and cluster map of the image (may be Shapefile or GeoTiff)	or thematic vector map (Shapefile)	
80	Waterlog detection on raster data	Raster image (GeoTiff)	Classified image (GeoTiff) and/or vector data with water categories (Shapefile).	4.3

DATA INVOLVED

Data involved	**Data size, complexity**
Medium-resolution (10-30 m) multispectral imagery: Landsat TM/ETM+/OLI, SPOT HRVIR or HRG	E.g. Landsat 8 OLI: 1.7 Gb / image, 7 images covering Hungary: 11.9 Gb for a single coverage
Airborne AISA Eagle/Hawk hyperspectral scanning	~70 Gb / flight hour, ~200 km2 covered
Orthophotos containing NIR band	16 Mb / km2 with 0.5 m resolution, e.g. full territory of Hungary: 1.5 Tb for a single coverage
Surface Model	16 Mb / km2 with 0.5 m resolution, e.g. full territory of Hungary: 1.5 Tb
Digital Elevation Model	e.g. full territory of Hungary: 18.6 Gb
Cadastral data	e.g. full territory of Hungary: 7 Gb

The Marine Showcase (1.2.2_SC1) and corresponding User Stories

User Story 1.2.2_SC1_1

I want to create a seamless land/underwater elevation model by the integration of land and underwater data sources (topographic and bathymetric LIDAR and SONAR point clouds, existing digital elevation models and surface models in different formats - data models - resolutions), and want to extract the shoreline based on it, to obtain a seamless data product that can also be used in further analysis and processing tasks.

The related User Stories for the Marine test bed are listed below:

- 1.1_47: As a GIS expert in a data production institute, I want to co-register topographic LiDAR data and bathymetric LiDAR data acquired at different times so that I can perform sea/land classification.
- 1.1_48: As a GIS expert in a data production institute, I want to perform accurate sea/land classification of combined LiDAR datasets so that I can produce a continuous sea/land DTM.
- 1.2.1_ 42: I want to merge topographic and bathymetric data on the land-sea interface (data coming from different horizontal and vertical datums)
- 1.2.1_60: I want to make a data fusion in between SONAR, LiDAR and satellite images (boundary of the waterbed) and point dataset to get the best digital surface model (underwater, seamless DEM) to be able to measure in the section of the riverbed.
- 1.2.1_69: I would like to have tools for integrating data about the coast line and seafloor in a unique surface model which allow me to work with large data (Lidar), with historical data (validation, reliability of time series), with better integration results at the border of tiles.

RELATIONS TO IQMULUS COMPONENTS/SERVICES

Step	Related IQmulus components and/or services
Precursor	Database of multiple LIDAR and SONAR point-cloud datasets. Each dataset has metadata describing temporal and spatial location as well as attributes that determine their use (e.g. license conditions, price, access constraints [user based]).
Step 1: The user defines the area of interest	generic UI functionality, Workflow Editor]
Step 2: The user identifies the input datasets to be used	[generic UI functionality, Workflow Editor] User searches metadata to identify data to be used User selects/de-selects dataset to use
Step 3: The system computes the 'best' representation of the topographic surface	[generic system I/O functionality, IQmulus Hub] If the data are not pre-processed: Service 10 (T4.3, Y1): Outlier classification in point clouds Service 34 (T4.3, Y2): Attribute-based segmentation of the

	point clouds Service 14 (T4.2, Y2): Coarse registration Service 6 (T4.2, Y1): ICP registration Service 2 (T4.3, Y1): Point cloud to regular grid Service 9 (T4.4, Y2), 55 (T4.4, Y2): Generate spline surface from parameterized point cloud Service 38 (T4.4, Y2): Local refinement of DEM with higher resolution elevation data Service 48 (T4.4, Y2): Multi-resolution triangulation Service 49 (T4.4, Y2): Constrained triangulation
Step 4: The system stores the resulting surface along with quality maps	[generic system I/O functionality, IQmulus Hub]
Step 5: the user visualizes surface	[generic system I/O functionality, IQmulus Hub] [generic & IQmulus visualization functionality]

SERVICES INVOLVED

It is assumed that all surface generation algorithms will be the same regardless whether the LAND or MARINE test bed is considered (see 1.2.2_SC2_1 above). The key difference is where physical features are present such as a harbour wall. The survey boat has to go around this and therefore this needs to be masked out.

DATA INVOLVED

Data involved	**Data size, complexity**
Lidar data	Various test sites in Belgium, Denmark, Norway, UK, France, Italy. Each site roughly 500Gb
SONAR Data	Various and extensive coverage across European Waters. Most complete for UK waters. Roughly 30TB
Charted points	XYZ of depth soundings derived from charts. Various locations across Europe. A “simple” processing task (surface generation) but with many input data sets (10^3) and a large storage volume (10’s TB).

FUNCTIONAL MULTI-PURPOSE SERVICES TO BE USED AS BASIC BUILDING BLOCKS

FIRST-YEAR

#	Name	Input	Output	Toolkit
7	Static tiling of LAS file	point cloud (LAS)	multiple point clouds (LAS)	4.3
10	Outlier Clasification in Point Clouds	point cloud,(LAS)	enriched point cloud (LAS)	4.3
11	Spatial Extent	point cloud (LAS)	Vector Data (Shapefile)	4.2
12	Dataset intersection	Point Cloud (LAS) Gridded Point Cloud (GeoTIFF)	Vector Data (Shapefile)	4.2
16	Quick visualization	point cloud (LAS)	-	4.3
17	3D Local keypoint extraction from point clouds (low-level feature)	gridded point cloud (GeoTIFF)	matrix, (.mat)	4.3
18	Matching 3D keypoint descriptor vectors	2 shape files & 2 array (.mat)	array (.mat)	4.2
23	GeoTIFF to LAS	gridded point cloud, (GeoTIFF)	point cloud (LAS)	4.2
35	Resampling of Point Cloud	point cloud (LAS)	point cloud (LAS)	4.2
36	Sub-sampling (thinning) of Point Cloud	unstructured point cloud (LAS)	unstructured point cloud (LAS)	4.2
44	CriticalPoints	triangle mest (PLY) or gridded point cloud (GeoTIFF)	3 Vector Data (Shapefile)	4.3
46	Isosurface extraction	enriched point cloud (LAS) or voxel grid	triangulation (ply)	4.2
47	Isocontour extraction	gridded point cloud (GeoTIFF)	Vector data (ShapeFile)	4.2
54	Filter Point Cloud by Attribute & Coordinate	point cloud (LAS)	point cloud (LAS)	4.3

SECOND-YEAR

#	Name	Input	Output	Toolkit
9	Spline surface from parameterized point cloud	point cloud (LAS)	Spline surface. g2-format (internal SINTEF format) or iges (for B.spline surfaces)	4.4
14	Coarse registration	point cloud (LAS)	array (.mat)	4.2
37	Low-level feature Point	point cloud (LAS)	Vector Data (Shapefile) & array (.mat)	4.3
41	Spatial indexing		Basic data type (struct)	4.2
48	Multi-resolution triangulation	triangulation (PLY)	to be defined	4.4
49	Constrained triangulations	point cloud (LAS) lines, polygons (ShapeFile)	triangulation (ply), polygons (ShapeFile)	4.4
50	Tiling and stitching triangulated surfaces	multiple triangulation (PLY)	triangulation (ply)	4.4
51	Triangulation of gridded point cloud	gridded point cloud (GeoTIFF)	triangulation (ply)	4.4
55	Generate spline surface from point cloud Modified to handle a gridded point cloud	gridded point cloud (GeoTIFF)	Spline surface (g2 or iges if possible)	4.4
56	Parameterize triangulated point set	triangulation (ply)	Enriched point cloud	4.4
57	Update spline surface with additional points	Spline surface and point cloud (LAS)	Spline surface (g2 or iges)	4.4
62	Extracting point cloud surface normals using triangle meshes	Triangle mesh (PLY)	Vector Data (.mat)	4.4
63	Extracting surface normals from point clouds	Unstructured point cloud (LAS)	enriched point cloud (LAS)	4.4
64	PersistentCriticalPoints	Triangle mesh (PLY) Gridded point clouds	List of features (ShapeFile)	4.3

		(geoTIFF)		
75	Principal components analysis (PCA)	Point cloud (LAS)	enriched point cloud, format: LAS	4.2

DATA INVOLVED

The functional services will be tested and demonstrated on any of the datasets listed in the showcase tables, provided that their type and format correspond to the type accepted as input by the services.

SOME COMPLEMENTARY SHOWCASES

After Month 6 these have been derived from relevant user stories and identified by WP4 as important to be considered for the IQmulus Hub.

An Urban Showcase

I want to update my existing 3D catalogue of urban topographic objects given a new data set. For that purpose I want to be able to

- *remove non-static objects (like cars, rubbish bins, bikes, people,...),*
- *characterize changes (addition, removal, deformation) in static objects (like trees, bus-stops, facade elements, chimneys,...),*
- *possibly include new or exclude existing categories (e.g. remove category of roof-top antennas; add category of charge points for electric cars)*

DATA INVOLVED

Data involved	Data size, complexity
Mobile Mapping data Toulouse *Mobile mapping data representing 10 km2 of the city of Toulouse*	- 309 Gbytes for 3800 image nodes (*) with no compression - 25 Gbytes for the point cloud (**) - 36 Gbytes for the volumetric point cloud (***)
	(*) Information on the images node: Each image node comprises a bundle of images acquired by two stereo rigs (one looking frontward, one looking backward) with full HD cameras (24 Mbytes per node) and a panoramic head with five 2k*2k cameras (60 Mbytes per node). Thus in all for one image node 84 MBytes. Images will be delivered under RGB 8 bytes in uncompressed TIFF
	(**) Information in the point cloud files: time X Y Z Intensity id-pulse id-echo (***) Information in the volumetric point clouds: time X Y Z Intensity id-pulse id-echo Xo Yo Zo, where (Xo

	Yo Zo) is the position of the laser head when the pulse is sent.
Further data supporting this showcase (catalogue update) will be acquired on the spot using e.g. handheld cameras or static laser scanners that are available to the project	

Lidar full wave form attribute extractions

"I want to extract useful attributes (such as skewness; kurtosis; left-side length; left-side angle; right-side length; right-side angle; amplitude) from Lidar full wave forms (FWF) raw signal so that I can provide information for high accuracy classification and habitat mapping of seabed types (substrate and vegetation).

DATA INVOLVED

Data involved	Data size, complexity
Airborne HAWKEYE II Lidar system	~ 300 GO / 100 km^2 area

Land and seafloor morphological changes

"I want to monitor and visualize temporal displacement rates and surface changes of some landforms (landslides, shoreline, submarine dunes etc.) from multitemporal DEMs and optical images to complement in-situ measurements, infer relationships with environmental parameters (e.g. rainfall, ocean currents) and predict future evolution"

DATA INVOLVED

Data involved	Data size, complexity
Punctual observed rainfall data *Forecasted rainfall data* *Meteorological radar data* *Water levels and ocean waves/currents*	Liguria region: 2 Mb Liguria region: 5.6 Gb Liguria region: 1 Mb Brittany region : 1 Gb
Lidar data	Data size depends on point cloud density. E.g. ca. 60 Mb/sqkm @ 2 pts/sqm density Liguria: ca 42 Gb
Low resolution Digital Surface Model	5x5m resolution, Liguria: ca 800 Mb
High resolution Digital Surface Model	16 Mb / km2 with 0.5 m resolution, La Spezia 2 Gb France (Litto3D product at 1 m resolution); * Multi-temporal datasets of some small areas (Brittany beaches, Volcanic Islands, Submarine dunes etc.); size around 10-100 Mb each
Very-high resolution optical images	Pleiades satellite images, Barcelonnette area ~ 4 GB (orthophotos of lanslides are only ~ 100 MB)

•

3.3 USE CASE DIAGRAM – MARINE SHOWCASE

As a part of the requirement consolidation process, the use case diagram of the Marine Showcase was created according to the approved methodology. The diagram provides information not just about the result of Use Case derivation from the User Stories (e.g., Upload data, Select ROI, Create DTM), but also gives particulars about the connections and relations between the Use Cases and processes, Use Case relations to the system architecture's layers and the actors of the procedure (e.g., data provider, user, IQmulus Storage Cloud, IQmulus Visualization) and highlights additional important elements or circumstances connected to a given Use Case of the procedure (e.g., "Declaring what data is to be exposed for the DEM creating" – connected to the Use Case: Upload Data).

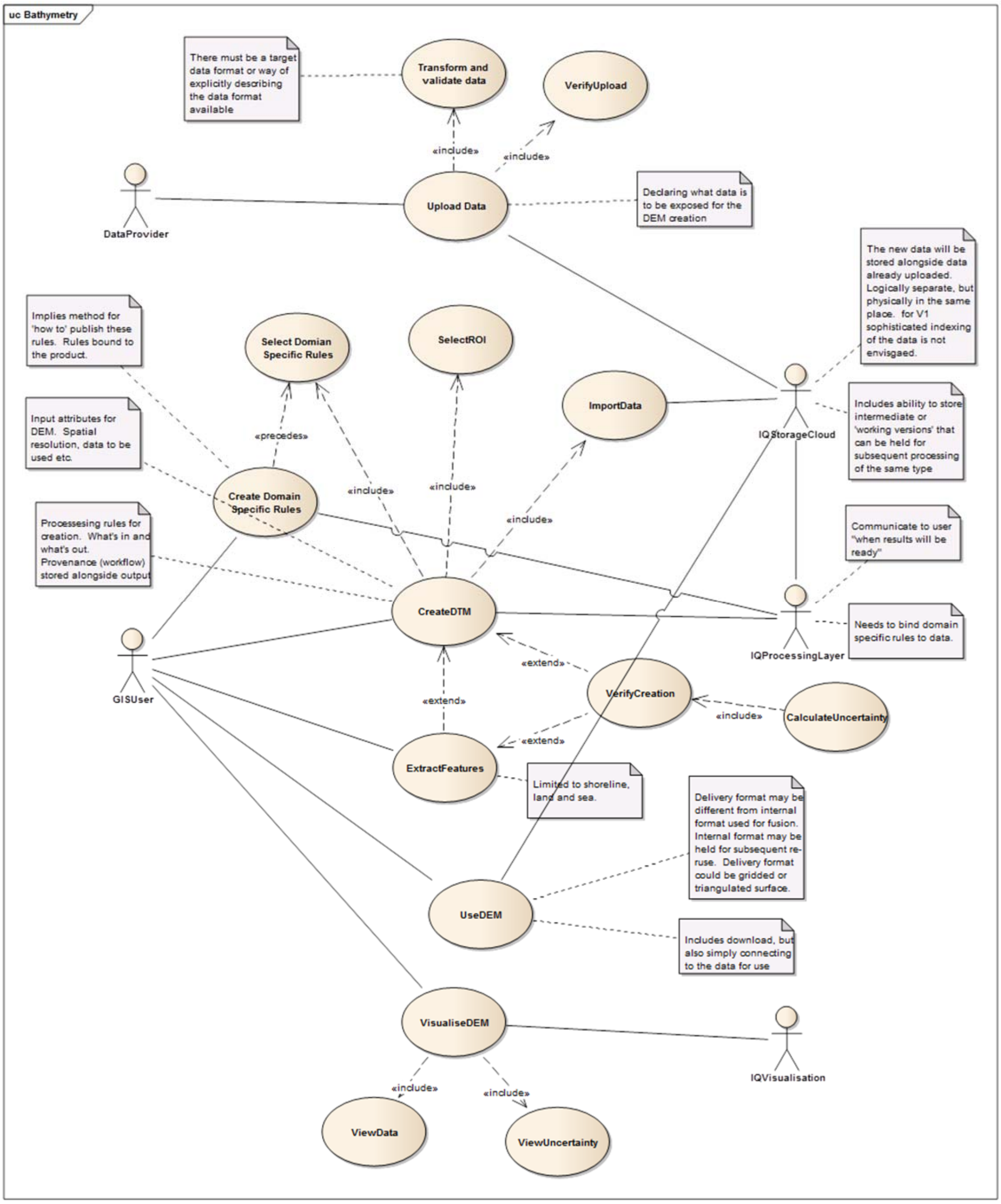


Figure 11. The Use Case Diagram derived from the Marine Showcase

3.4 ATOMIC USE CASES

Many atomic Use Cases were derived from different User Stories by the institutions involved in the consolidation process of the User Requirements. In this chapter Use Cases and requirements directly derived from the Marine and Land Showcases are introduced, others are documented in the annexes of the current deliverable.

Use Cases corresponding to Land Showcase User Story 1

Identifier	*1.2.2_SC2_1_UC1*
Description	*Production of the "best" available representation of the topographic surface by using all available altimetric datasets*
Actors	*GIS expert*
Initial conditions	*All datasets are in the final coordinate system (including vertical)*
Final results	*A terrain model adapted to the needs of flood simulation*
Main process	*Step 1: the expert defines data selection criteria (e.g., date, accuracy resolution, sensor…)* *Step 2: the system extracts and loads input data according to selection criteria* *Step 3: the system performs regularization (surface generation) and provides DTM along with quality maps*
Alternative processes	
Exceptional situations	
Processed data	*Altimetric data (point clouds, DTM, etc.)*
Generated data	*DTM integrated from various sources* *Quality maps*
Activity diagram	
Related User Stories	
Functional requirements	– *Use pre-defined parameters that can be modified by the user (output resolution, regularization method, data selection criteria : date, accuracy, resolution, sensor type) with the DSL* – *support input data that can be : raster DTM, TINs, points clouds* – *Output data is a DTM plus metadata / quality maps (file giving spatial quality indicators: estimated accuracy, date, sensor type if applicable)*
Non-functional requirements	– *Output data quality must be the highest possible according to input data quality*

Identifier	*1.2.2_SC2_1_UC2*
Description	*Extraction of dams, break lines, channels, drainage network from the „best" representation of the topographic surface*
Actors	*hydrologist, System*
Initial conditions	*Resolution high enough to detect relevant features, reference data (e.g. river map)*
Final results	*Break lines (3D polylines)* *Dams, channels (3D polygons)* *Drainage network (streams: linear, basins: area)*
Main process	*Step 1: the user defines the geographic area* *Step 2: the system extracts input data inside the ROI* *Step 3: user selects threshold parameters for feature detection* *Step 4: the system detects the break lines, dams, channels, drainage network based on user-defined parameters* *Step 5: User validates the extracted features by interactive visualization of results along with reference data* *if the result is not satisfactory, repeat from step 3* *Step 6: the system stores the detected features*
Alternative processes	
Exceptional situations	
Processed data	*[already available: LIDAR: dataset 22-23, map of the rivers (reference map): dataset 21, interpolated rainfall data: dataset 24]* *Altimetric data, reference data (thematic maps), user-defined parameters*
Generated data	*Features*
Activity diagram	
Related User Stories	
Functional requirements	*Support input data that can be : raster DTM, point clouds*
Non-functional requirements	

Use Cases corresponding to Land Showcase User Story 2

Identifier	*1.2.2_SC2_2_UC1*
Description	*Fast detection of flooded areas and landslides based on remote sensing imagery and reference data*
Actors	*RS expert*
Initial conditions	*Availability of previously aligned datasets*
Final results	*Thematic map representing flooded areas with different categories*
Main process	*Step 1: The user defines the area of interest* *Step 2: The user identifies the input datasets to be used* *Step 3: The user identifies the reference (training and test) data containing geometries and (possibly hierarchical) thematic attributes about known flooded areas and landslides* *Step 4: The system loads input and reference data* *Step 5: The system computes indices and metrics (NDWI, NDVI)* *Step 6: The system determines thresholds and classification parameters by optimisation based on reference data* *Step 7: The system carries out classification* *Step 8: The system stores the resulting features along with quality maps*
Alternative processes	
Exceptional situations	
Processed data	*Remote sensing images (containing SWIR band) possibly of different resolution* *Digital Terrain Model* *LiDAR point clouds / DSMs of different acquisition dates*
Generated data	*Thematic maps (flood categories – at least: open water surface, inundated vegetation, wet soils)*
Activity diagram	
Related User Stories	

Functional requirements	
Non-functional requirements	

Marine Showcase and corresponding Use Cases

Identifier	*1.2.2_SC1_1_UC1*
Description	*Topographic and bathymetric LiDAR (or SONAR) point clouds co-registration*
Actors	*GIS expert, System*
Initial conditions	*Estimated accuracy of each data set is known* *Data sets must overlap*
Final results	*Co-registered point clouds*
Main process	*Step1: compute the overlapping area* *Step2: load data on the overlapping area* *Step3: compute transformation parameters for each dataset to ensure good geometric co-registration, with respect to initial estimated accuracy* *Step4: transform each dataset using computed transformation parameters* *Step5: compute quality indicators for registration process, such as altimetric differences between datasets*
Alternative processes	
Exceptional situations	
Processed data	*topographic LiDAR data* *bathymetric LiDAR data and / or SONAR*
Generated data	*Co-registered topographic LiDAR data* *Co-registered bathymetric LiDAR data* *Quality indicators*
Activity diagram	
Related User Stories	
Functional requirements	– *Compute the overlapping area between two point clouds* – *Compute altimetric differences between two overlapping point clouds* – *Compute co-registration transformation parameters for co-registration of two point clouds, weighted by initial estimated accuracy of the point clouds* – *Transform a point cloud using transformation parameters*
Non-functional requirements	– *Use ISPRS LAS format for point clouds*

Identifier	*1.2.2_SC1_1_UC2*
Description	*Integration topographic and bathymetric point clouds and existing Digital Terrain Models (DTM) / Digital Surface Models (DSM) with sensor sonar data of the coast line and seafloor in a unique surface model*
Actors	*GIS expert; Field expert*
Initial conditions	*The LIDAR model is done* *The DTMs of the historical data is done* *The sensor sonar data are provided*
Final results	*Surface model of the coastal area*
Main process	*Step 1: the user selects the geographical area* *Step 2: the system regularizes the DTMs* *the system fuses the data into the LIDAR model* *Step 3: the system visualizes the model*
Alternative processes	
Exceptional situations	
Processed data	*LIDAR (datasets 22, 23 in x,y,z and LAS format)* *time series of data, multibeam points (dataset 17) or depth grid (dataset 18)* *raster images* *Sonar images (.tiff?)*
Generated data	*DTM of the coastal area*
Activity diagram	*An activity diagram showing the main process diagrammatically*
Related User Stories	
Functional requirements	*Alignment,* *fusion,* *regularization,* *interpolation* *visualization of the surface model*
Non-functional requirements	*quality is important*

Identifier	*1.2.2_SC1_1_UC3*
Description	*Sea/land classification and shoreline detection based on combined datasets*
Actors	*GIS expert, System*
Initial conditions	*At least one dataset has been acquired using a bathymetric lidar* *Lidar and sonar datasets must have been co-registered (if many)*
Final results	*Classified point clouds, shoreline delineation*
Main process	*Step 1: the system loads the point clouds* *Step 2: the system performs sea/land segmentation and / or classification of points (3 classes are populated: land, seabed, water surface), wrt the geometric/morphological characteristics and the sensor information* *Step 3: the system computes the shoreline (edge of the water surface) as a 3D polyline* *Step 4: The system visualizes classification results and the detected shoreline*
Alternative processes	*Allow the user to interactively check and edit the classification* *The process can be tiled to work on large areas*
Exceptional situations	
Processed data	*Lidar point clouds from different datasets*
Generated data	*Classified Lidar point clouds* *Shoreline representation*
Activity diagram	
Related User Stories	
Functional requirements	
Non-functional requirements	*Use ISPRS LAS format for point clouds*

In addition to the Use Cases and requirements listed above, other important and frequently arising ones are collected in the Annexes of the current document.

- Annex 2 presents the complete process of the formulation of Use Case Diagrams and Use Case descriptions with Activity Diagrams provided by the Foldmeresi és Taverzekelesi Intezet (FOMI).
- Annex 3 presents the same process on the example of Lidar FullWave Forms Processing Package by Institut Francais de Recherche pour L'exploitation de la Mer (Ifremer).
- Annex 4 presents Use Case descriptions provided by Institute Geographique National (IGN).
- Annex 5 documents the Use Case descriptions provided by Regione Liguria.
- Annex 6 lists the Prioritized User Stories from D1.1 and D1.2.1.

4 CONCLUSIONS

The process of (both internal and external) user requirements gathering provided IQmulus partners with a large set of User Stories addressing all application categories (or “Trials”) that have been identified in the Description of Work, and even more. These requirements are well documented in D1.1 “State of the Art Analysis” and D1.2.2 “Initial User Requirements”, and a dynamic online requirement documentation system.

To ensure that Use Cases and Functional Requirements derived from this material target the areas where IQmulus aims at generating significant progress for users, and to align this with the project development plan, Showcases (both Land and Marine) have been identified by thorough analysis and prioritization of user stories. Moreover, Use Cases have been derived from pre-selected User Stories.

This document contains the first results of the requirement consolidation process, and constitutes a solid basis for the development in the first project phase. Nevertheless, as requirement consolidation is a continuous process, these results will be further polished and extended with additional material to be submitted to the online Requirement Management and Tracking System.

5 ANNEXES

5.1 ANNEX 1: IQMULUS USE CASE AND REQUIREMENTS TEMPLATE

Link in eRoom:

https://project.sintef.no/eRoomReq/Files/math/IQmulus/0_2f6de/IQmulus%20USE%20CASE+REQ%20template.doc

Uploaded to eRoom: 08-04-2013

Identifier	*The unique identifier for the use case*
Description	*A short natural language description of the use case.*
Actors	*A reference to the actors as identified beforehand.*
Initial conditions	*A description of any relevant precondition that might be fulfilled before the use case can be executed.*
Final results	*A description of the final results of the use case.*
Main process	*The steps required for achieving the final result.*
Alternative processes	*Any alternatives that might occur in the main process.*
Exceptional situations	*Any exceptional situation that might occur during the main process.*
Processed data	*A description of the data types relevant / processes in this use case.*
Generated data	*A description of the data generated by executing this use case and, e.g., whether it is persisted or not.*
Activity diagram	*An activity diagram showing the main process diagrammatically.*
Related User Stories	*A reference to related user stories.*

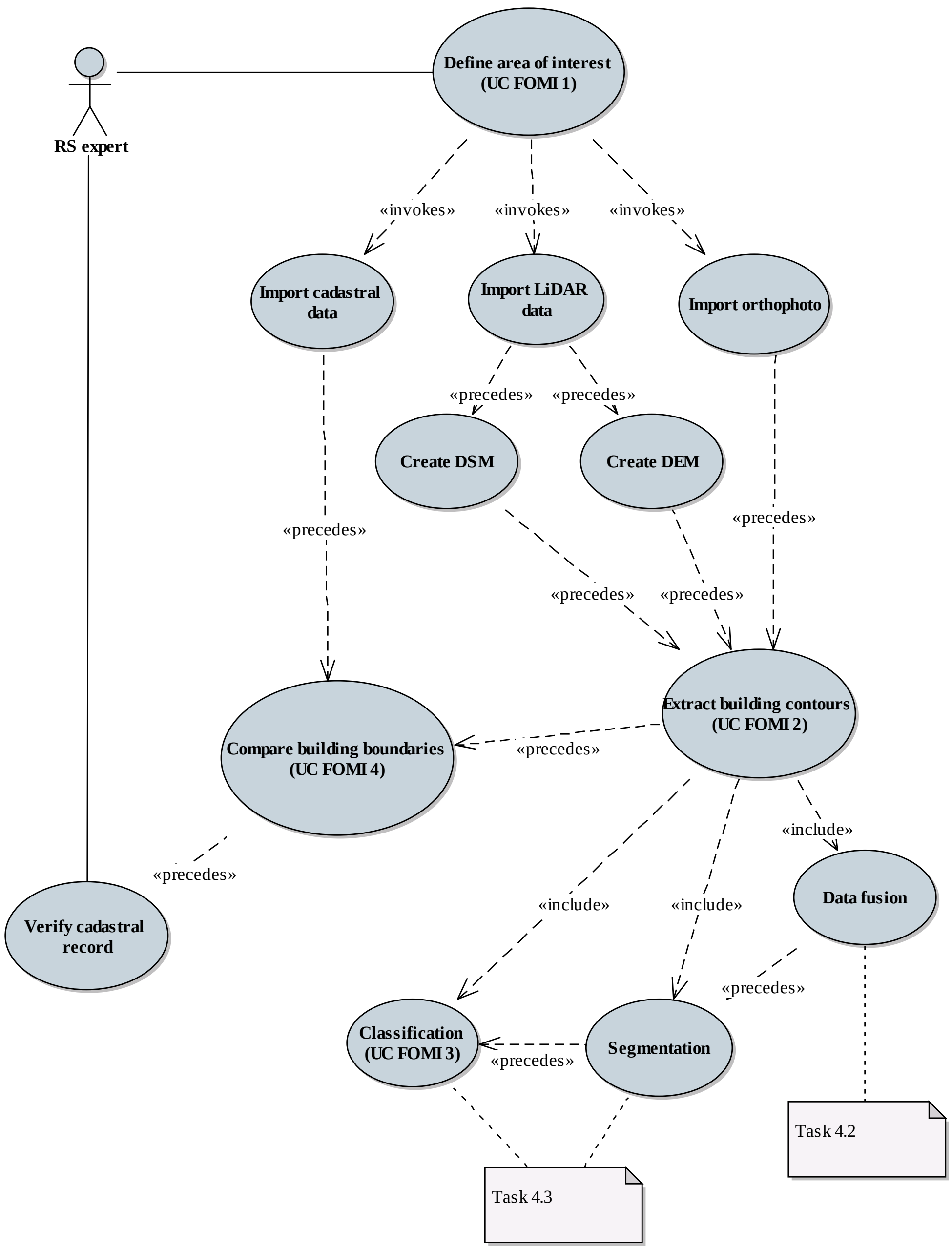
Functional requirements	*A list of functional requirements related to the use case*
Non-functional requirement	*A list of non-functional requirements related to the use case*

5.2 ANNEX 2: USE CASE DESCRIPTIONS PROVIDED BY THE FOLDMERESI ES TAVERZEKELESI INTEZET (FOMI)

User Story (1.1_22)

"As a RS expert I want to compare the automatically classified building contours based on orthophoto, Digital Elevation Model and Digital Surface Model with the building boundaries in the cadastral database, so that I can verify the adequacy of the cadastral records."

Use case diagram



Use Case descriptions and requirements

Select AOI

Identifier	1.1_22_UC1
Description	*Select the (polygonal) extent of area of interest.*
Actors	*GIS/RS expert, Decision maker*
Initial conditions	Web based UI with base map.
Final results	*Bounding polygon.*
Main process	*Step 1: Select all coordinates of the bounding polygon in clockwise order (at least 3 coordinates must be selected).* *Step 2: Verify and confirm the polygon.* *Step 3: Submit the selection.*
Alternative processes	*Step 1: Input all coordinates of the bounding polygon in clockwise order in free text mode using the base map's reference system (at least 3 coordinates must be given).* *Step 2: Verify and confirm the polygon.* *Step 3: Submit the selection.*
Exceptional situations	
Processed data	
Generated data	Bounding polygon (3…n coordinate pairs).
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	

Import cadastral data

Identifier	1.1_22_UC2
Description	*Load cadastral data (building footprint polygons, cadastral parcels) from a file format compatible with GDAL/OGR.*
Actors	*System, User*
Initial conditions	*Building polygons, cadastral parcels available in the user's local file system in a format compatible with GDAL/OGR. The data contains at least the following attributes:* *buildings: geometry (polygon), UID;* *cadastral parcels: geometry (polygon), UID*
Final results	*Cadastral data loaded in the storage layer*
Main process	*Step 1. User defines the path where the files are located (select files from local file system)* *Step 2. User identifies the file format to be used for each layer* *Step 3. User defines the attributes (geometry, UID) to be used for each layer.* *Step 4. Check conformance to predefined data structure and geometrical constraints* *Step 5. Transfer cadastral data to the storage layer into a predefined data structure.* *Step 6. Data is stored and accessible for the System in the storage layer.*
Alternative processes	
Exceptional situations	*Data is not present* *Data does not comply to predefined structure and constraints*
Processed data	*Building polygons (GDAL/OGR compatible file), Cadastral parcels (GDAL/OGR compatible file)*
Generated data	*Cadastral data in the storage layer*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	

Import point cloud

Identifier	1.1_22_UC3
Description	*Load a point cloud (LAS format) based on LIDAR or stereo-photogrammetric measurements.*
Actors	*System*
Initial conditions	*Point cloud present in the local file system in LAS format.*
Final results	*Point cloud data loaded in the storage layer*
Main process	*Step 1. User defines the path where the files are located (select files from local file system)* *Step 5. Transfer point cloud data to the storage layer into a predefined data structure.* *Step 6. Data is stored and accessible for the System in the storage layer.*
Alternative processes	
Exceptional situations	*Data is not present* *Data does not comply to predefined structure and constraints*
Processed data	*Point cloud in LAS format*
Generated data	*Point cloud data in the storage layer*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	

Import orthophoto

Identifier	1.1_22_UC4
Description	*Load multispectral orthophotos from a file format compatible with GDAL/OGR.*
Actors	*System*
Initial conditions	*Multispectral orthophotos available in the user's local file system in a format compatible with GDAL/OGR.*
Final results	*Orthophotos loaded in the storage layer*
Main process	*Step 1. User defines the path where the files are located (select files from local file system)* *Step 2. User identifies the file format to be used for each layer* *Step 3. User defines the bands to be imported.* *Step 4. Transfer orthophotos to the storage layer.* *Step 6. Data is stored and accessible for the System in the storage layer.*
Alternative processes	
Exceptional situations	*Data is not present* *Data does not comply to predefined structure and constraints*
Processed data	*Orthophotos (GDAL/OGR compatible raster files)*
Generated data	*Orthophotos in the storage layer*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	

Create DSM

Identifier	1.1_22_UC5
Description	*Create Digital Surface Model (regularly gridded height data) from point cloud*
Actors	*System*
Initial conditions	*Point cloud present in the Storage Layer,* *Orthophotos present in the Storage Layer*
Final results	*Digital Surface Model (DSM) in the storage layer*
Main process	*Step 1: Define regular grid according to the extent, resolution and position of the orthophoto raster.* *Step 2: Identify points positioned over each grid cell.* *Step 3: Assign a height value to each grid cell based on the average height of first-return points belonging to it over the base surface of the reference system.* *Step 4. Apply predefined "no data" values to grid cells with no point.* *Step 5. Store the resulting grid in the storage layer.*
Alternative processes	
Exceptional situations	*No overlap between point cloud and orthophotos.*
Processed data	*Point cloud, orthophotos*
Generated data	*Digital Surface Model (DSM, regular grid with height attribute)*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	Feedback on processing status: failed, success. Progress of processing status.

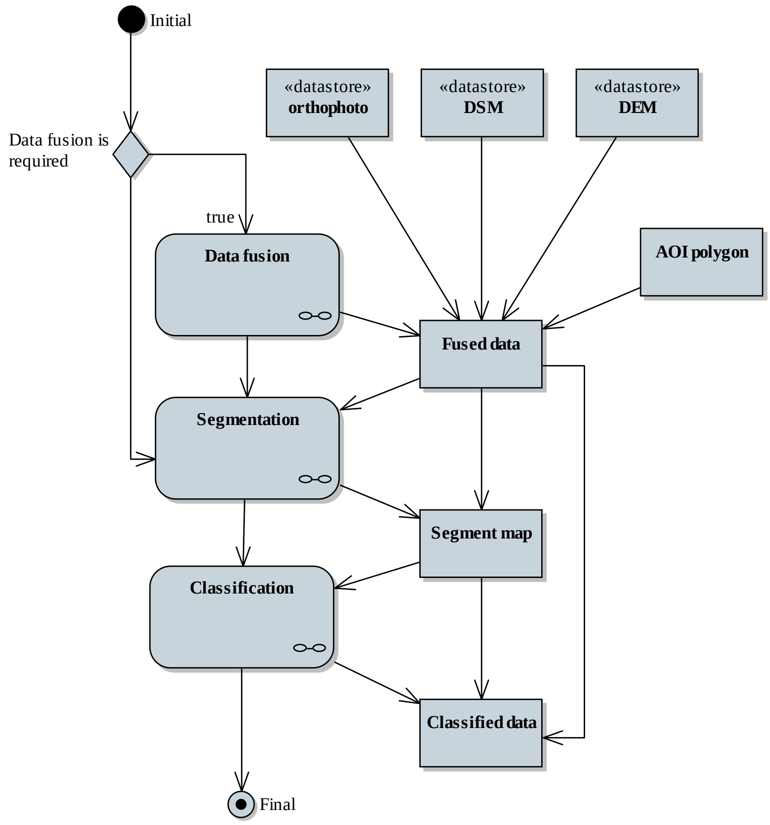
Create DEM

Identifier	1.1_22_UC6
Description	Create Digital Elevation Model (regularly gridded height data) from point cloud
Actors	*System*
Initial conditions	Point cloud present in the Storage Layer, Orthophotos present in the Storage Layer
Final results	*Digital Elevation Model (DEM) in the storage layer*
Main process	Step 1: Define regular grid according to the extent, resolution and position of the orthophoto raster. Step 2: Identify points positioned over each grid cell. Step 3: Assign a height value to each grid cell based on the average height of last-return points belonging to it over the base surface of the reference system. Step 4: Remove terrain objects and patch their place with data interpolated from the surroundings to obtain a topographical surface. Step 5. Store the resulting grid in the storage layer.
Alternative processes	
Exceptional situations	No overlap between point cloud and orthophotos.
Processed data	Point cloud, orthophotos
Generated data	Digital Elevation Model (DEM, regular grid with height attribute)
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	Feedback on processing status: failed, success. Progress of processing status.

Extract building contours

Identifier	1.1_22_UC7
Description	*Obtain a polygonal representation of the building contours based on DSM, DEM and orthophotos.*
Actors	*System*
Initial conditions	*DSM, DEM and orthophoto of the selected area is available and loaded. For classification reference data is available for building identification.*
Final results	*Polygonal representation of building contours.*
Main process	*Step 1: Data fusion of DSM, DEM and orthophoto (UC FOMI 10).* *Step 2: Segmentation (UC FOMI 11).* *Step 3: Automated classification based on ruleset derived by using reference data (UC FOMI 12).*
Alternative processes	*If data fusion is not needed (because feature extraction can run on diverse data sets):* *Step 1: Segmentation.* *Step 2: Automated classification based on ruleset derived by using reference data.*
Exceptional situations	
Processed data	*DSM, DEM, orthophoto, reference data, AOI rectangle.*
Generated data	*Object boundaries (vector data) and object metrics (descriptive data).* *Building contour polygons (vector data).*

Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	Feedback on processing status: failed, success. Progress of processing status.

Compare building boundaries

Identifier	1.1_22_UC8
Description	*Compare the shape (classification result) to the cadastral data and denote differences.*
Actors	*System*
Initial conditions	*Cadastral data is available and loaded. Classification result is available.*
Final results	*New field in the cadastral map's attribute table including total area of buildings/parcel*
Main process	*Step 1: Calculate the total area of buildings / parcel from the cadastral map.* *Step 2: Overlap cadastral map and the classification result.* *Step 3: Calculate the total area of buildings / parcel from the classification result maps.* *Step 4: Compare the results/parcel. Calculate the difference.* *Step 5: Write the value of difference to a new field in the cadastral map's attribute table.*
Alternative processes	
Exceptional situations	
Processed data	*Cadastral data, building shapes (from classification)*
Generated data	*New field in the cadastral map's attribute table with the value of difference (given in square meters)*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	*Feedback on processing status: failed, success.* *Progress of processing status.*

Verify cadastral record

Identifier	1.1_22_UC9
Description	*Verify cadastral record based on the result of building boundary comparison with a given threshold.*
Actors	*GIS/RS expert, Decision maker, System*
Initial conditions	*New field in the cadastral map's attribute table with the value of difference (given in square meters) is generated.*
Final results	*Point map (centre of parcels) with all original and calculated attributes presented in the cadastre map and calculated during the process.*
Main process	*Step 1: Display map of building boundary differences and descriptive data table* *Step 2: Define a threshold for acceptance criteria interactively* *Step 3: Write the decision result to the cadastre map's attribute table. (Change/No change)* *Step 4: Convert polygon cadastre map into point map (centre of parcels). Keep all attributes.*
Alternative processes	-
Exceptional situations	
Processed data	*Point map* *Descriptive data in table format*
Generated data	*New field in the cadastral map's attribute table with the value of difference (given in square meters)* *Point map*
Activity diagram	
Related User Stories	*1.1_22*

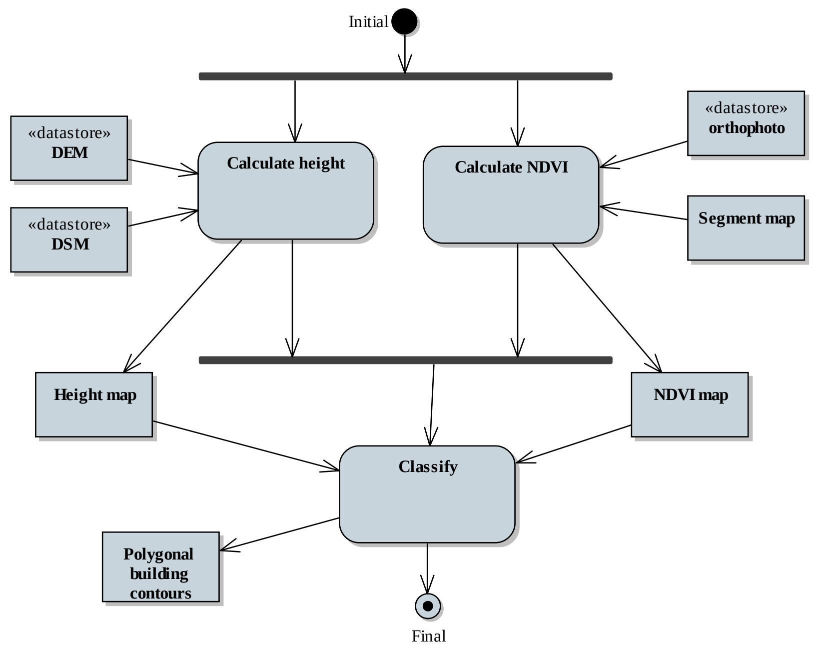
Functional requirements	
Non-functional requirements	*Slider for setting a threshold of acceptance criteria is available for the user.* *User should be able to modify the attribute table. (Rights to write)*

Segmentation

Identifier	1.1_22_UC11
Description	*Generate image objects as a basis for classification*
Actors	*RS/GIS expert, System*
Initial conditions	*Data fusion is ready. Orthophoto is available.*
Final results	*Polygonal representation of segments*
Main process	*Step 1: Segmentation with parameters defined interactively* *Step 2: Result is stored in the storage layer*
Alternative processes	
Exceptional situations	
Processed data	*Orthophoto of the selected area*
Generated data	*Segment map (image objects)*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	*Sliders for setting segmentation parameters are available for user.* *Feedback on processing status: failed, success.* *Progress of processing status.*

Classification

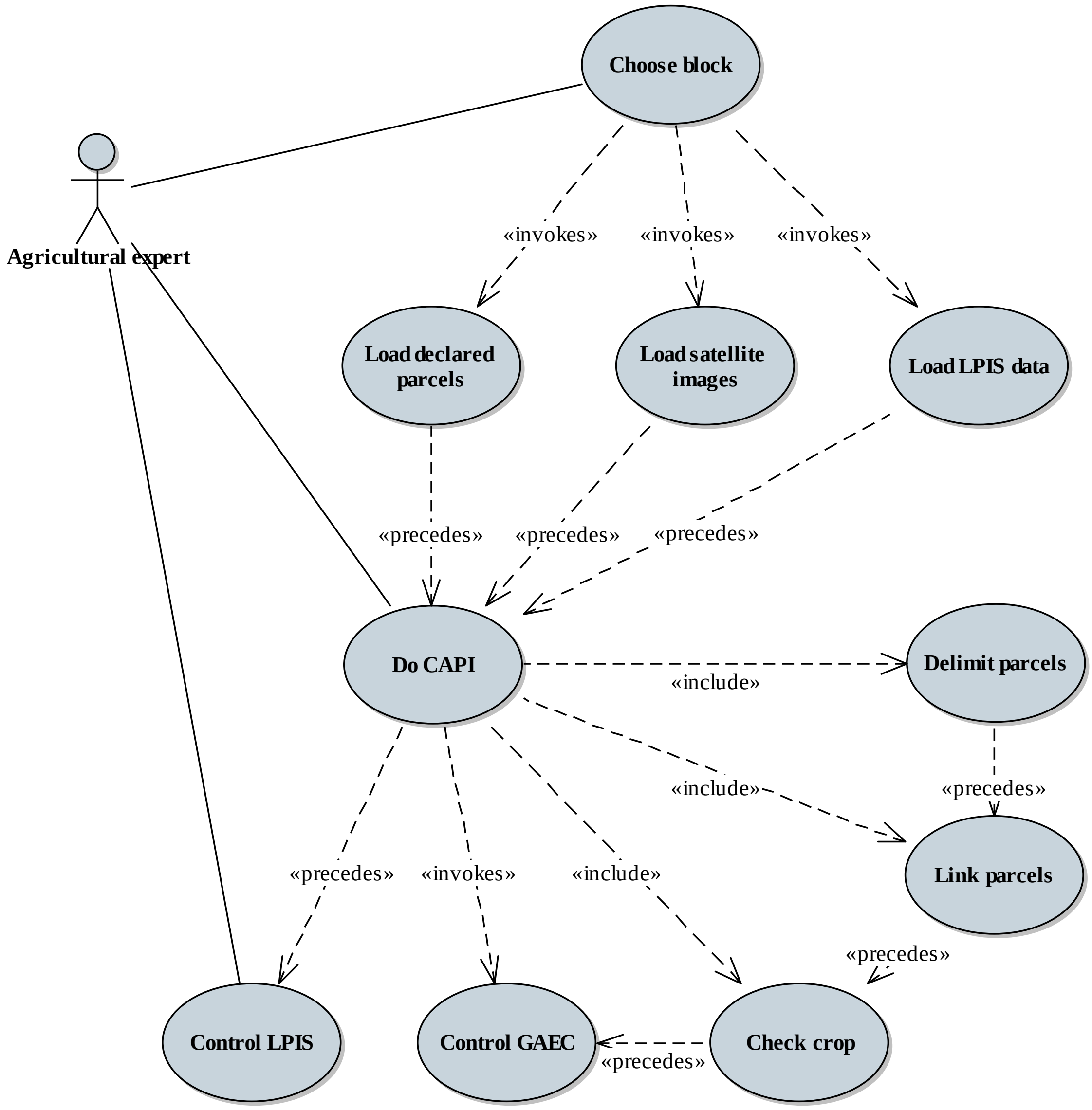
Identifier	1.1_22_UC12
Description	*Classify every image object derived from segmentation.*
Actors	*System*
Initial conditions	*DSM, DEM and orthophoto of the selected area is available and loaded. Orthophoto is segmented. Segmentation result is available from the storage layer.*
Final results	*Polygonal representation of building contours.*
Main process	*Step 1: Calculate NDVI from orthophoto. (In parallel with Step 2.)* *Step 2: Calculate height from DEM and DSM. (In parallel with Step 1.)* *Step 3: Classification using mean NDVI and height of image objects*
Alternative processes	
Exceptional situations	
Processed data	*DSM, DEM, segment map (image objects) of the orthophoto*
Generated data	*Object boundaries (vector data) and object metrics (descriptive data).* *Building contour polygons (vector data).* *Object height map*
Activity diagram	
Related User Stories	*1.1_22*

Functional requirements	
Non-functional requirements	*Feedback on processing status: failed, success.* *Progress of processing status.*

User Story (1.1_35)

"As an agricultural expert, I want to determine crop cover for crop parcels of a given area using color composites of multispectral satellite images taken at different dates for documenting the status at given dates and to consider the justness of agricultural subsidies for this area."

Use case diagram



Use Case descriptions and requirements

Import reference data

Identifier	*1.1_35_UC13*
Description	*Load polygonal vector reference data for classification process*
Actors	*System, RS expert*
Initial conditions	*Polygons are available in the user's local file system in a format compatible with GDAL/OGR.*
Final results	*Reference data is loaded in the storage layer*
Main process	*Step 1. User defines the path where the files are located (select files from local file system)* *Step 2. Load the data*
Alternative processes	
Exceptional situations	*Data is not present*
Processed data	*Reference polygons (GDAL/OGR compatible)*
Generated data	*Reference polygons in the storage layer*
Activity diagram	
Related User Stories	*1.1_35*

Functional requirements	
Non-functional requirements	*Feedback on processing status: failed, success.* *Progress of processing status.*

Check crop

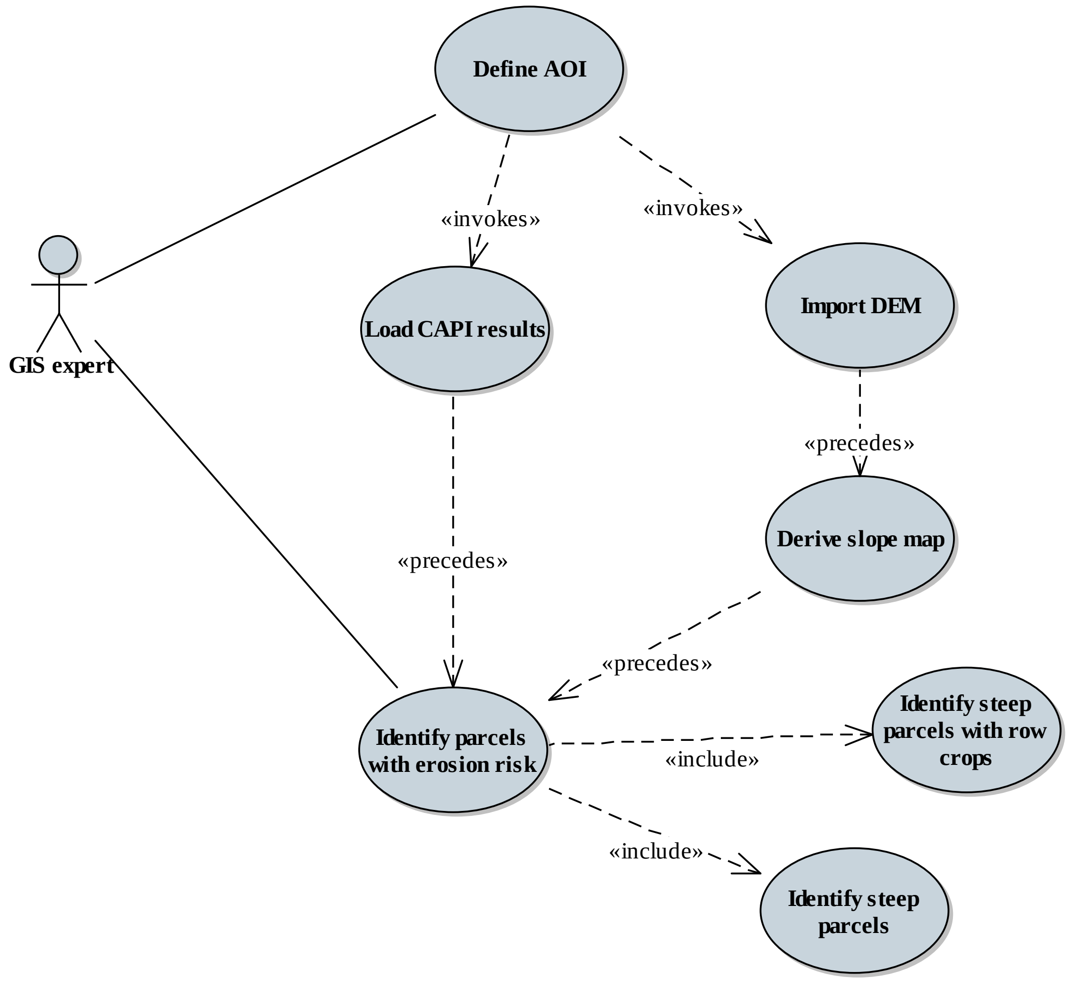
Identifier	*1.1_35_UC8*
Description	*Establish diagnostics regarding declared and observed crops*
Actors	*Agricultural expert*
Initial conditions	*The client software of Computer-aided Photo-interpretation* *In Alternative process, a crop map derived from satellite data is also available.* *A hierarchical system of crops and interpretation crop groups must be defined in advance.*
Final results	*Diagnostic code regarding declared and observed crops* *In case of mismatch, the code of observed crop*
Main process	*Step 1: If image time series shows the characteristics of declared crop, ACCEPT* *Step 2: If image time series does not show clearly the characteristics of declared crop, but one cannot unambiguously state that actual crop belongs to another interpretation crop group, ACCEPT* *Step 3: If image time series show the characteristics of another crop or interpretation crop group, REJECT, and provide actual crop/interpretation crop group.* *Step 4: In other cases observed crop is not interpretable by remote sensing.*
Alternative processes	*Alternative process takes place if crop map is available.* *Step 1: If the majority (using a pre-defined threshold) of pixels of crop map within the parcel equals to the declared crop, ACCEPT* *Step 2: If the majority of pixels of crop map within the parcel belongs to the interpretation crop group of declared crop, ACCEPT* *Step 3: If the majority of pixels designate an interpretation crop group differing from the one of declared crop, REJECT, and provide crop/interpretation crop group derived from crop map.* *Step 4: In other cases decision cannot be made using the crop map. Use the methodology of Main process.*
Exceptional situations	*Decision cannot be made by remote sensing.*
Processed data	*Subsidy claim database (polygons+attributes of declared agricultural parcels) of a selected physical block* *Land Parcel Identification System data* *Satellite image time series* *If applicable: crop map derived from remote sensing data*
Generated data	*Diagnostic codes regarding the correctness of declared crop for each declared agricultural parcel in the physical block* *The observed crop/interpretation crop group for parcels where the declared crop was not found to be correct*
Activity diagram	
Related User Stories	*1.1_35 (parent), 1.1_27*

Functional requirements	
Non-functional requirements	

User Story (1.1_24)

"As a GIS expert I want to delineate slopes steeper than a given threshold (with DDM, without LiDAR, with LiDAR, etc.) so that I can support the definition of erosion risk areas."

Use case diagram



Use Case descriptions and requirements

Derive slope map

Identifier	*1.1_24_UC4*
Description	*Derive slope category map from DEM*
Actors	*GIS expert*
Initial conditions	*GIS software with Digital Elevation Map (DEM)*
Final results	*Raster or vector map delineating areas steeper than a pre-defined threshold*
Main process	*(Assuming raster output)* *Step 1: Calculate the slope steepness for each parcel, using local gradients in given directions.* *Step 2: Mark pixels having steeper slope than the pre-defined threshold as steep.*
Alternative processes	*(Assuming vector output)* *Step 1: Calculate the slope steepness for each parcel, using the magnitude of local gradient. A continuous raster layer is obtained.* *Step 2: Mark pixels having steeper slope than the pre-defined threshold as steep. A thematic raster layer is obtained (0 – not steep, 1 – steep).* *Step 3: Convert this raster layer to vector using a decent generalization so as to obtain smooth output.*
Exceptional situations	
Processed data	*Digital Elevation Map in raster format* *Threshold of slope (precentage or degrees)*
Generated data	*Raster output: bi-level thematic layer designating not steep (0) and steep (1) areas.* *Vector output: polygon layer, with polygons delimiting steep areas.*
Activity diagram	
Related User Stories	*1.1_24 (parent), 1.1_35*

Functional requirements	
Non-functional requirements	

Identify steep parcels

Identifier	*1.1_24_UC6*
Description	*Overlay the boundaries of observed parcels with slope category map, and select observed parcels that have high elevation angle*
Actors	*Agricultural expert*
Initial conditions	*GIS software with slope category map and the result of Computer-aided Photo-interpretation*
Final results	
Main process	*(Assuming slope category map is available in raster format)* *Step 1: Overlay the boundaries of observed parcels with raster slope category map.* *Step 2: Determine the area of steep pixels of slope category map within each parcel.* *Step 3: Apply the pre-defined rule for each parcel to determine whether it is steep.*
Alternative processes	*(Assuming slope category map is available in vector format)* *Step 1: Overlay the boundaries of observed parcels with vector slope category map.* *Step 2: Determine the area of intersection with polygons delimiting steep areas for each parcel.* *Step 3: Apply the pre-defined rule for each parcel to determine whether it is steep.*
Exceptional situations	
Processed data	*Slope category map in raster or vector format* *Polygons of observed agricultural parcels* *Rule for the distinction between steep and flat parcels (for example, threshold for area ratio of steep area within the parcel and total area of parcel)*
Generated data	*An attribute for each parcel (steep/not steep)*
Activity diagram	
Related User Stories	*1.1_24 (parent), 1.1_35*

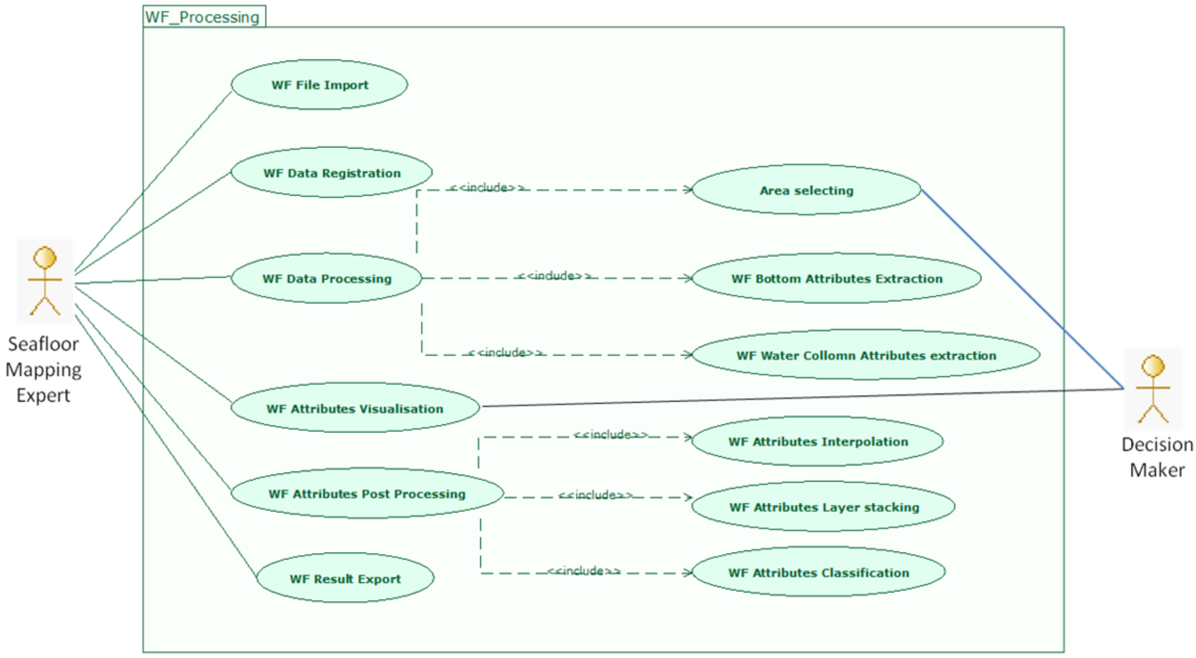
Functional requirements	
Non-functional requirements	

5.3 ANNEX 3: USE CASE DESCRIPTIONS PROVIDED BY THE INSTITUT FRANCAIS DE RECHERCHE POUR L'EXPLOITATION DE LA MER (IFREMER) - LIDAR FULLWAVE FORMS PROCESSING PACKAGE

Objectives

The final aim of Lidar fullwave forms processing is the production of a seafloor 2D map with labels corresponding to the identified sea bottom types. The use case diagram bellow summarizes all use cases needed to meet this objective. **It is related to eRoom User stories: 1.2.1_39 to 1.2.1_43**

Use case diagram



The main Use Cases are the next:

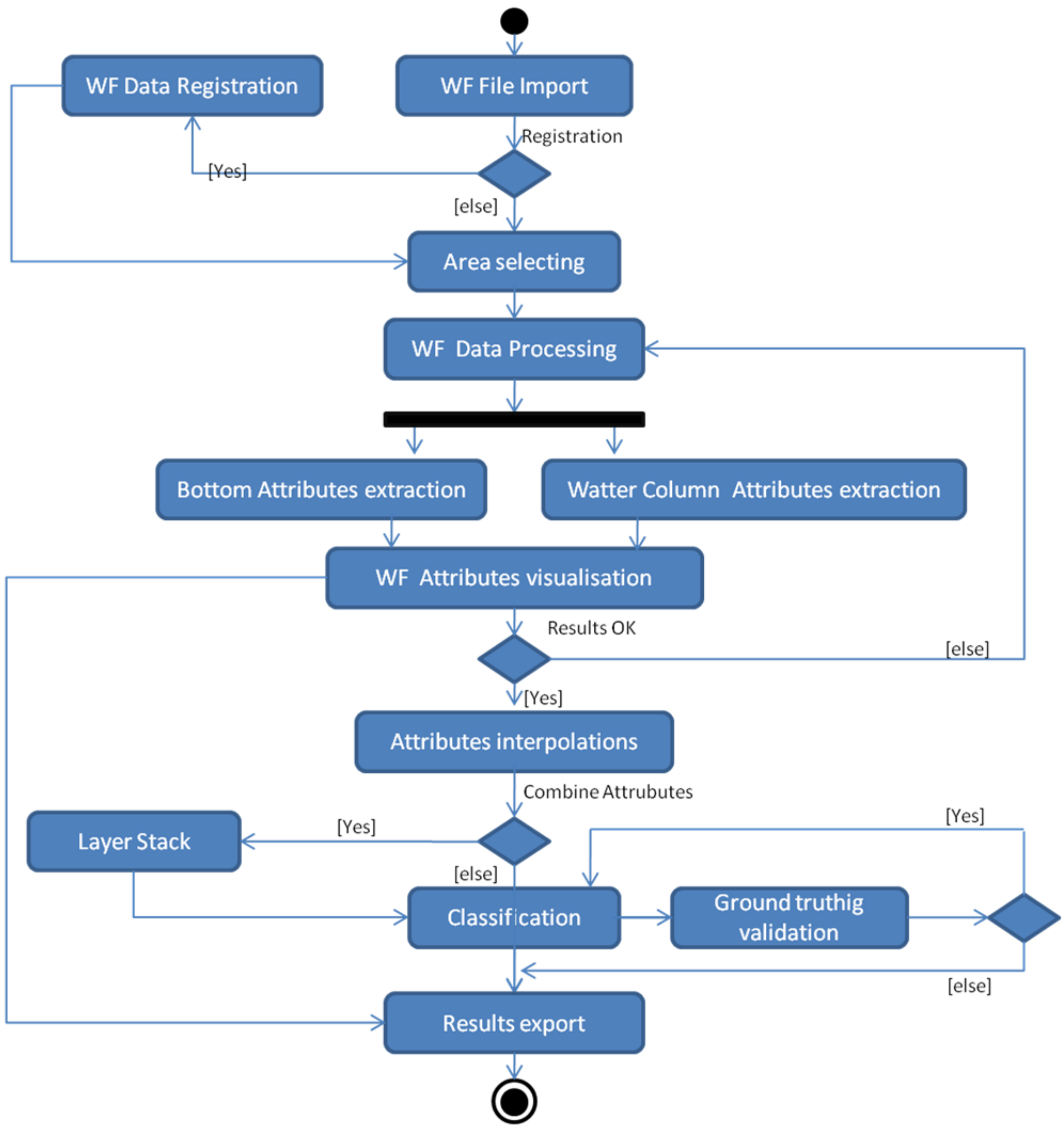
1) WF File import: for Lidar data import in IQmulus system
2) WF Data Processing :for processing the full wave form signal
   a) WF Bottom Attributes Extraction: for processing
   b) WF Water Column Attributes Extraction: for processing
3) Area Selecting: to define the geographical area
4) WF Data Registration: to make all the data available in the same reference frame
5) WF Attribute visualization: to visualize data and explore the results
6) WF Attribute Post-Processing
   a) WF Attribute Interpolation: to interpolate WF attribute cloud data
   b) WF Attribute Layer stacking: to create a WF attribute multilayer raster data
   c) WF Attribute Classification: to classify maybe refines the results
7) WF Results Export: to export a fixed product for the decision maker or an interactive product for the expert.

[ES] in the use case description means Exceptional Situation (followed by a number for the exception)

WF: Waveforms

Activity diagram

The following activity diagram summarizes different steps to perform Lidar fullwave forms data processing.



Use Case: WF File import

Identifier	**1.2.1_39-43_UC1**
Description	*Allows the user to import a Lidar Wave Forms data file*
Actors	*habitat mapping expert, System*
Initial conditions	*The system is functional* *The user is authenticated* *The import file is in the* correct *format (ascii or Las)*
Final results	*Wave Forms Lidar Data are imported into the system and can be manipulated*
Main process	*The scenario begins when the user wants to import a file containing Full WF Lidar data.* *1. The system opens a browser window and asks the user to choose a file to import among the available files he can access* *2. The user selects a file* *3. The system reads the data [ES1]* *4. The system uploads the data and displays their location by zooming in on their geographical extend* *End of scenario*
Alternative processes	
Exceptional situations	*[ES1] A problem is encountered by the system to read the file, an error message is displayed.*
Processed data	*Raw Wave Forms Lidar data file*
Generated data	*Wave Forms Lidar data are imported into the system with for each point an associated series of values which represents the signal amplitude versus time.*
Activity diagram	*See General diagram*
Related Use cases	*WF Data registration* *Area selection*
Functional requirements	*Capability of the system to read Lidar WF file format*
Non-functional requirements	*Rapid WF Lidar data integration*

WF Data registration

Identifier	**1.2.1_39-43_UC2**
Description	*Allows the user to put the Wave Forms Lidar datasets in the same projection system.*
Actors	*Habitat mapping expert, System*
Initial conditions	*The system is functional and the user is authenticated* *The data remapping is already loaded in the system* *Projection system references and their characteristics are known by the system.*
Final results	*The Wave Forms Lidar datasets specified by users are in the same projection system and can be handled together*
Main process	*The scenario begins when the user wants to put WF Lidar data in a defined projection system.* *1. The user tells the system which data sets to transform* *2. The system asks the user to specify the initial data reference system* *3. The user selects the reference from a drop-down menu in the system* – *WGS 84* – *Lambert 93* – *Others to be defined ...* *4. The system asks the user the desired output projection* *5. The user chooses the new system by selecting it from a drop-down menu in the system* *6. The system applies the adequate transformation to put the data into the new projection system [ES1]* *7. The system displays the location of data in the new projection system by zooming in on their geographical extend*
Alternative processes	
Exceptional situations	*[ES1] A problem is encountered by the system durring the reprojection data, an error message is displayed.*
Processed data	*Lidar data file Wave Forms uploaded into the system*
Generated data	*Wave Forms Lidar cloud data (with for each point an associated series of values representing the signal amplitude versus time) are reprojected*
Activity diagram	*See General diagram*
Related Use cases	*WF File import* *Area selection*
Functional requirements	*User should be able to set a Region of Interest, by setting geographic coordinates and by drawing geometric primitives* *System should provide a data planimetric registration quality*
Non-functional requirements	*Methods should be fast* *System should inform the user about the required processing time*

Area Selection

Identifier	**1.2.1_39-43_UC3**
Description	*Allows the user to interactively select displayed data in the current view*
Actors	*habitat mapping expert, End-user, System*
Initial conditions	*The system is functional and the user is authenticated* *WF Lidar cloud Data are uploaded into the system*
Final results	*Une selection de données est effectuée* *A data selection is performed*
Main process	*The scenario begins when the user wants to make a selection of WF cloud data.* *1. The user chooses a selection tool available in the GUI (eg box / polygon)* *2. The user outlines the area of interest (box / polygon) in the view.* *3. The system selects all the entities present within the specified area of interest* *4. The system highlights all the selected objects. 5. The user has the ability to export selected data (UC Export results)* *End of Scenario*
Alternative processes	*Alternative A: The user wishes to complete the current selection* *Continuing on from point 4 of the nominal scenario* *4.a.1 The user indicates he wishes to add items to the selection* *4.a.2 The user draws the area of interest (box / polygon) in the view* *4.a.3 The system maintains the initial selection and adds the elements of the current selection* *The Use Case returns to step 5.* *Alternative B: The user wishes to delete items from the current selection* *Continuing on from point 4 of the nominal scenario* *4.a.1 The user indicates that he wishes to remove items from the selection* *4.a.2 The user draws the area of interest (box / polygon) in the view* *4.a.3 The system deletes the elements of the current selection from the original selection* *The Use Case returns to step 5.*
Exceptional situations	
Processed data	*WF Lidar data cloud*
Generated data	*No new data is generated, only a graphic selection is performed*
Activity diagram	*See General diagram*
Related Use Cases	*WF Data Export*

Functional requirements	*Visual interface for spatial data* *Tools for interactive selection* *Symbolizing data / features*
Non-functional requirements	*Rapid data display*

WF Data Processing (includes 2 use cases)

- *WF Bottom attribute extraction (nominal Scenario)*
- *WF Water column attribute extraction (alternative Scenario)*

Identifier	**1.2.1_39-43_UC4**
Description	*Allows the user to extract data from the WF Lidar data useful parameters related to: Seabed bottom types (rocks, vegetation, soft bottom, etc..) for habitat mapping. Water column properties to qualify Water bodies*
Actors	*habitat mapping expert, System*
Initial conditions	*The system is functional and the user is authenticated Input data is already uploaded into the system. All the necessary pre-processing has already been performed (e, g, registration) The list of parameters of the background signal are known by the system The list of parameters of the water column signal are known by the system*
Final results	*WF bottom attributes or water column characteristics are extracted from LIDAR data set.*
Main process	*The scenario begins when the user wants to process WF Lidar data and extract parameterst that are characteristics of bottom types. 1. The system opens a window and asks the user to choose input files required for processing either from the system files or from personal files stored locally / on the network: – Lidar data full WF – Data on the state of the sea surface (wind stress) – Kd data: estimated attenuation coefficient (satellite data) – Sensor acquisition configuration – Environmental conditions (eg. sun position, atmosphéric properties) 2. The user selects for each input data the corresponding file 3. The system loads the relevant data [ES1] 4. The system asks the user to specify which part of the signal to process giving him the choice between: – Bottom signal (UC-WF Bottom attribute extraction) – Water column signal to (UC-WF Water column attribute extraction) 5. The user selects bottom signal processing 6. The system asks the user to specify which attributes to extract, giving him the choice between All Parameters or a selection from the following list: – skewness; – kurtosis; – left-side length; – left-side corner; – right-side length; – right-side corner; – amplitude – Waveform Peak SNR – Reflectivity 7. The system applies the following appropriate processes for each data point – 7a.WF signal noise removal – 7b.WF signal mathematical approximation – 7c. WF parameters extraction – 7d. WF correct parameters (eg slope, roughness influence ...) 8. The system displays the results of attribute extraction (UC WF Attribute visualization) 9. The user wishes to save the result (UC Result export) 10. The user wishes to refine the process>> go back to step 2 End of scenario*

Alternative processes	*Starts at point 4 of the nominal scenario* *4. The user selects the water column signal processing* *5. The system asks the user to specify which parameters to calculate, giving him the choice between All Parameters or a selection from the following list:* – *MY;* – *Chla;* – *Yellow Substances;* – *Event Detection (e.g. phytoplankton bloom, thermocline)* *6. The system asks the user to choose from a list the type of water body:* – *Homogeneous water mass* – *Stratified water mass* *7. The system applies the appropriate processing method to the selected water body type for parameter extraction Return to point 8 of the nominal scenario*
Exceptional situations	*[ES1] A problem* is *encountered by the system when opening or reading the input files*, the system *informs the user with a message*.
Processed data	– *FullWave Forms Lidar system with for each point an associated set of values representing the signal amplitude versus time* – *Data on the state of the sea surface (eg. wind stress)* – *Kd attenuation coefficient estimated from satellite data* – *Sensor acquisition configuration* – *Environmental conditions (eg. sun position, atmosphéric properties, …)*
Generated data	*A geolocated cloud file* with *for each point an associated set of values that correspond to the waveforme attributes selected by the user for their extraction (see nominal scenario for* bottom *attributes and the alternative scenario for water column attributes)*
Activity diagram	*See general activity diagram*
Related Use Cases	*WF Attribute Visualisation* *WF Attribute Interpolation* *WF result export*

Functional requirements	*Combine different sea floor and sea surface data sets* *IQmulus should analyze the amount and type of data* *IQmulus should display all available data*
Non-functional requirements	*At present this type of processing requires parallel calculations on multiple processors* – *Methods should be fast* – *System should inform the user about the required processing time* – *System should store intermediary results* – *System should be almost always available*

WF Attribute visualization

Identifier	**1.2.1_39-43_UC5**
Description	*Allows the user to visualize on a graphic interface geolocated data resulting from processing of WF Lidar data*
Actors	*habitat mapping expert, End-user, System*
Initial conditions	*The system is functional and the user is authenticated* *The visualization environment is active*
Final results	*Data are presented to the user and have a special display symbology according to the type of display requested by the user*
Main process	*The scenario begins when the user wishes to view a Lidar WF attribute extraction results file that could be:* – *Cloud data* – *Mono or multilayer raster data* *1. The system opens a window and asks the user to choose input files required for processing either from the system files or from personal files stored locally / on the network:* *2. The user selects a file* *3. The system loads the relevant layer and zooms in on the layer extend [ES1]* *4. The system offers the user the following options:* *4a. for a point layer and for each attribute* – *Read the values associated with each point* – *Calculate statistics (min, max, mean, mode, median, standard deviation) for each attribute* – *Assign a color code for different classes whose number and boundaries are indicated by the user [ES2]* *4b. for a raster layer* – *Read the values associated with each pixel* *4c. for each band of the raster layer* – *Calculate statistics (min, max, mean, mode, median, standard deviation)* – *Assign a color code for different classes whose number and boundaries are set by the user* *End of scenario*
Alternative processes	
Exceptional situations	*[ES1] A network problem prevents the user from having access to the requested file, the system cancels the upload and displays an error message* *[ES2] An inconsistency in the class boundary values, the system notifies the user and offers to redefine them*
Processed data	*Lidar WF attributes files:* – *Cloud data* – *Mono or multilayer raster generated by interpolation*
Generated data	*Data visualisation only*
Activity diagram	
Related Use Cases	*WF Data Processing* *WF Results export*

Functional requirements	*IQmulus should display all available data* *2D + 3D visualization* *Visualizing time series, stacks of data* *Symbolizing data/features*
Non-functional requirements	*Rapid data display* *Should support many data types (vector and raster data)*

WF Attribute Post-Processing

a. WF Attribute interpolation

Identifier	**1.2.1_39-43_UC6**
Description	*Allows the user to interpolate geolocalised data related to Lidar WF attributes*
Actors	*Habitat mapping expert, System*
Initial conditions	*The system is functional* *the user is authenticated*
Final results	*Data are presented to the user and have a special display symbology according to the type of display requested by the user*
Main process	*The scenario begins when the user wishes to interpolate a set of Lidar WF extracted attributes data:* *The system opens a window and asks the user to choose input files required for processing either from the system files or from personal files stored locally / on the network:* *2. The user selects a file* *3. The system uploads the relevant layer [ES1]* *4. The system asks the user to choose an interpolation method selected from a list:* – *Inverse Distance Weighted* – *Spline* – *Nearest Neighbour* – *Kriging* *5. The system asks the user to indicate depending on the method used:* – *The attribute to interpolate* – *The size of the output cell* – *The point search radius* – *The optimal number of points* – *The type of interpolation model* – *Others to be defined* *6. The system generates a raster file in the format specified by the user* *7. The system displays the result of interpolation by zooming in on the layer extend* *End of scenario*
Alternative processes	
Exceptional situations	*[ES1] A network problem prevents the user from having access to the requested file, the system cancels the upload and displays an error message*
Processed data	*Lidar WF extracted attributes cloud files*
Generated data	*2D mono layer Raster WF attribute data*
Activity diagram	
Related Use Cases	*WF Data Processing* *WF Data visualisation* *WF Results export*

Functional requirements	*System should analyze the amount and type of data* *System should display all available data* *System should provide information on interpolation quality*
Non-functional requirements	*Methods should be fast* *System should inform the user about the required processing time*

WF Attribute Post-Processing

b. WF Attribute Layer stacking

Identifier	**1.2.1_39-43_UC7**
Description	*Allows the user to combine multiple monolayer files relative to Lidar WF attributes into a single multilayer file*
Actors	*habitat mapping expert, System*
Initial conditions	*The system is functional and the user is authenticated*
Final results	*Raster WF Lidar attribute monolayers regrouped into a single multilayer raster file*
Main process	*The scenario begins when the user wishes to combine several Lidar WF attributes monolayer files into a single multilayer file:* *1. The system opens a window and asks the user to choose input files required for processing either from the system files or from personal files stored locally / on the network 2. The user selects the files* *3. The system uploads the relevant layers [ES1]* *4. The system asks the user for the name and format of the output file* *5. The user indicates the home folder and file name and specifies the format [ES2]* *6. The system generates a raster file with user specified format* *7. The system displays the result by zooming in on the layer extend* *End of scenario*
Alternative processes	
Exceptional situations	*[ES1] A network problem prevents the user from having access to the requested file, the system cancels the upload and displays an error message* *[ES2] file already exists, the system informs the user and asks user to enter a new file name*
Processed data	*Mono or multiband raster files resulting from the interpolation of results extracted from Lidar attributes data files*
Generated data	*2D multilayer Raster data*
Activity diagram	
Related Use Cases	*WF Data Processing* *WF Results export*

Functional requirements	*System should analyze the amount and type of data* *System should display all available data* *Process should be fast*
Non-functional requirements	

WF Attribute Post-Processing

c. *WF Attribute Classification*

Identifier	**1.2.1_39-43_UC8**
Description	*Allows the user to generate a map of bottom types from Lidar data WF attributes*
Actors	*Habitat mapping expert, System*
Initial conditions	*The system is functional and the user is authenticated*
Final results	*Production of 2D maps with labels corresponding to identified sea bottom types*
Main process	*The scenario begins when the user wishes to apply a classification to a single or multi-layer file:* *1. The system opens a window and asks the user to choose input files required for processing either from the system files or from personal files stored locally / on the network T* *2. The user selects the file* *3. The system uploads the relevant layer [E1]* *4. The system asks the user to select a classification method* – *Supervised Classification* – *Unsupervised Classification* *5. The user chooses to apply a supervised classification,* *6. The system asks the user to indicate the layer containing the training areas and their description, or allows him to define the areas of interest interactively* *7. The system performs the supervised classification and displays the results* *8. The user can validate the generated data by comparing it to the field data* *9. The user has the option to save the classification results (UC Result export)* *10. The user has the option to refine the classification if the results are not sufficient >> return to step 4* *End of scenario*
Alternative processes	*Starts at point 4* *5. The user chooses to apply an unsupervised classification* *6. The system asks the user to indicate the number of output classes* *7. The system performs an unsupervised classification and displays the results* *Return to step 8 of the nominal scenario*
Exceptional situations	*[ES1] A network problem prevents the user from having access to the requested file, the system cancels the upload and displays an error message*
Processed data	*Mono or multiband raster files resulting from the interpolation of results extracted from Lidar attributes data files*
Generated data	*2D mono layer Raster data*
Activity diagram	
Related Use Cases	*WF attribute layer stacking* *WF Results export*

Functional requirements	*System should provide classification accuracy* *Graphic interface for geolocalised data visualization* *Interactive data input tools (polygon and attribute entities)*
Non-functional requirements	*System should store intermediary data*

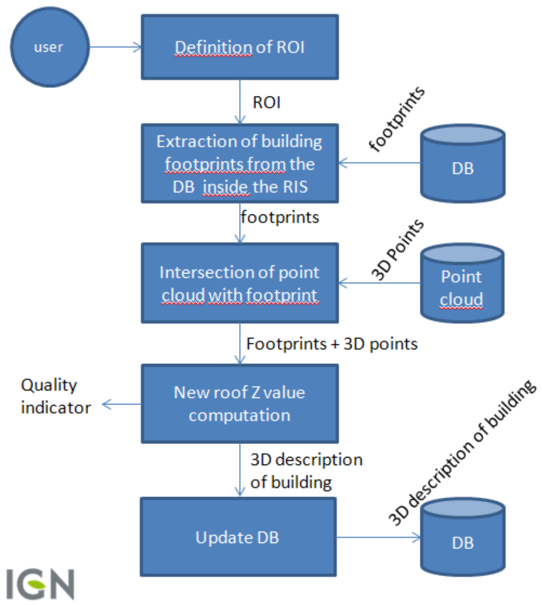
WF Result export

Identifier	**1.2.1_39-43_UC9**
Description	*Allows the user to save the results of WF data processing in a file*
Actors	*habitat mapping expert, End-user, System*
Initial conditions	– *The system is functional and the user is authenticated* – *The results of WF Lidar data processing are already uploaded on the system interface*
Final results	*The result file is available to the user separately from the IQmulus system*
Main process	*The scenario begins when the user requests to export results data* *1. The system compares the user's profile to the uploaded data in order to verify data rights [ES1]* *2. The system provides an interface to specify the storage folder [ES2]* *and to enter the export folder name [ES3]* *3. The system asks the user to specify the export format by choosing from a list of available formats :* – *For vector data (ascii, shpape file ...)* – *For raster data (GeoTIFF, ENVI, ...)* *4. The system exports the data:* – *The layer is a vector and a selection exists, the system exports only the data associated with the selected records* – *The layer is a vector and no selection exists, the entire layer is exported by the system* – *The layer is a multiband raster, the system allows the user to select which bands are exported.* *5. The system generates the file physically [ES4]* *End of scenario*
Alternative processes	
Exceptional situations	*ES1] The system tells the user that he is not authorised to export data* *[ES2] The storage directory is not available, the system notifies the user with a message* *[ES3] A file with the same name already exists. The system asks the user to enter a new name or to overwrite the existing file* *[ES4] Disk space is insufficient, the system informs the user and cancels the export*
Processed data	*Results of Lidar WF processing*
Generated data	*WF Lidar processed data File (vector or raster)*
Activity diagram	*See general activity diagram*
Related Use Cases	*WF Data visualization* *WF data processing* *Area selection*

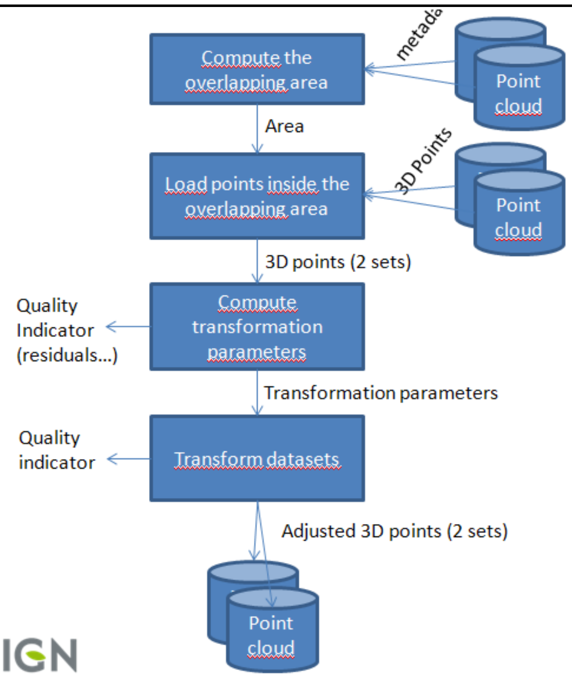
Functional requirements	*Should support many data types*
Non-functional requirements	*System should inform the user about the disk space needed for data storage*

5.4 ANNEX 4: USE CASE DESCRIPTIONS PROVIDED BY INSTITUT GEOGRAPHIQUE NATIONAL (IGN)

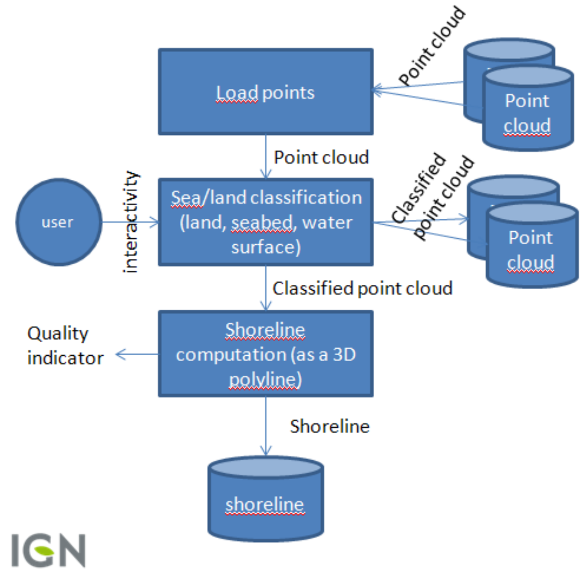
Identifier	***1.1_46_UC1***
Description	*Enhancement of polygonal building description*
Actors	*GIS expert, System*
Initial conditions	*Registration done* *Accuracy of the data is sufficient?*
Final results	*Enhanced polygonal building description in the DB*
Main process	*Step 1: The expert defines the geographical area* *Step 2: The System extracts the footprint of the building from the DB* *Step 3: the system intersects the point cloud with the foot print* *Step 4: the systems computes new Z values for the building* *Step 5: The system updates the object in the DB*
Alternative processes	*Enhancement of 2D position and 2D description of building* *Full building detection from point cloud for change detection:* *Step 1: The expert defines the geographical area* *Step 2: The System extracts the footprint of the building from the DB* *Step 3: the system detects the building footprint from the point cloud* *Step 4: comparison of footprints, quality indicators* *if indicator is OK* *Step 5: the system intersects the point cloud with the detected foot print,* *Step 6: the systems computes new Z values for the building* *Step 7: The system updates the object in the DB* *if indicator is not OK* *Step 5: flag the building*
Exceptional situations	*Building has changed/disappeared*
Processed data	*Buildings from DB* *Lidar Point cloud*
Generated data	*Buildings from DB*

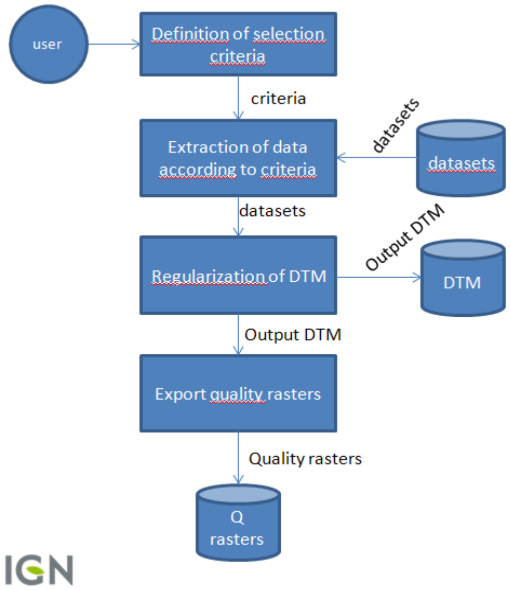
Activity diagram	
Related User Stories	*1.1_46: As a GIS expert in a data production institute, I want to combine 2D/3D building polygons from a topographic DB with LiDAR data in order to extract building elevation so that I can enhance geometric description of buildings in my DB..*

Functional requirements	*Use a predefined geographical area that have been set by the user with the DSL as a region of interest*
Non-functional requirements	*New Z value computation may use several functions (arithmetic mean, mediane, percentile..)*

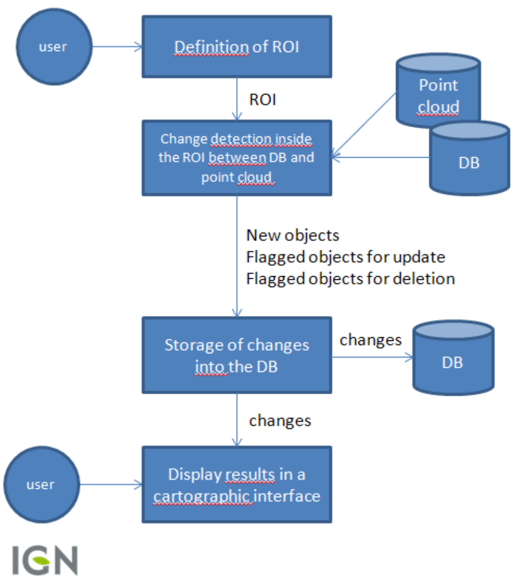
Identifier	***1.1_47_UC1***
Description	*Topographic and bathymetric LiDAR point clouds co-registration*
Actors	*GIS expert, System*
Initial conditions	*Estimated accuracy of each data set is known* *Data sets must overlap*
Final results	*Co-registered point clouds*
Main process	*Step1: compute the overlapping area* *Step2: load data on the overlapping area* *Step3: compute transformation parameters for each dataset to ensure good geometric co-registration, with respect to initial estimated accuracy* *Step4: transform each dataset using computed transformation parameters* *Step5: compute quality indicators for registration process, such as altimetric differences between datasets*
Alternative processes	
Exceptional situations	
Processed data	*Topographic LiDAR data* *Bathymetric LiDAR data*
Generated data	*Co-registered topographic LiDAR data* *Co-registered bathymetric LiDAR data* *Quality indicators*
Activity diagram	
Related User Stories	*1.1_47: As a GIS expert in a data production institute, I want to co-register topographic LiDAR data and bathymetric LiDAR data acquired at different times so that I can perform sea/land classification.*

Functional requirements	*Transformation parameters lead to physical model refinement, not to 3D transformation of point cloud* *Transformation parameters computation must be weighted by initial estimated accuracy of each point cloud*
Non-functional requirements	*Use ISPRS LAS format for point clouds*

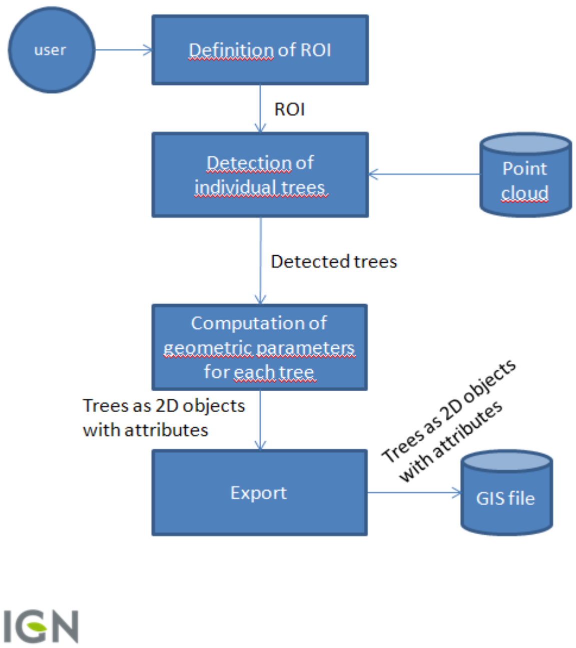
Identifier	***1.1_48_UC1***
Description	*Sea/land classification of combined LiDAR datasets*
Actors	*GIS expert, System*
Initial conditions	*At least one dataset has been acquired using a bathymetric lidar* *Lidar datasets must have been co-registered (if many)*
Final results	*Classified point clouds*
Main process	*Step 1: the system load the point clouds* *Step 2: the system performs sea/land classification of points (3 classes are populated: land, seabed, water surface)* *Step 3: the system computes the shoreline (edge of the water surface)as a 3D polyline*
Alternative processes	*Allow the user to interactively check and edit the classification* *The process can be tiled to work on large areas*
Exceptional situations	
Processed data	*Lidar point clouds from different datasets*
Generated data	*Classified Lidar point clouds* *Shoreline description*
Activity diagram	
Related User Stories	*1.1_48: As a GIS expert in a data production institute, I want to perform accurate sea/land classification of combined LiDAR datasets so that I can produce a continuous sea/land DTM.*
Functional requirements	
Non-functional requirements	*Use ISPRS LAS format for point clouds*

Identifier	***1.1_49_UC1***
Description	*Altimetric data fusion*
Actors	*System, GIS expert*
Initial conditions	*All datasets are in the final coordinate system (including vertical)* *All datasets are correctly filtered (bare earth)*
Final results	*Countrywide DTM*
Main process	*Step 1: the expert defines data selection criteria (egdate, accuracy resolution, sensor…)* *Step 2: the system extracts input data according to selection criteria* *Step 3a: the system performs regularization of DTM* *Step 3b: and quality rasters (eg covariance)*
Alternative processes	
Exceptional situations	
Processed data	*Raster DTM from various sources*
Generated data	*Raster DTM* *Quality rasters*
Activity diagram	
Related User Stories	*1.1_49: As a GIS expert in a data production institute, I want to merge altimetric datasets from various sensors at different resolution and acquired at different times so that I can produce a continuous DTM at an optimized resolution and accuracy.*

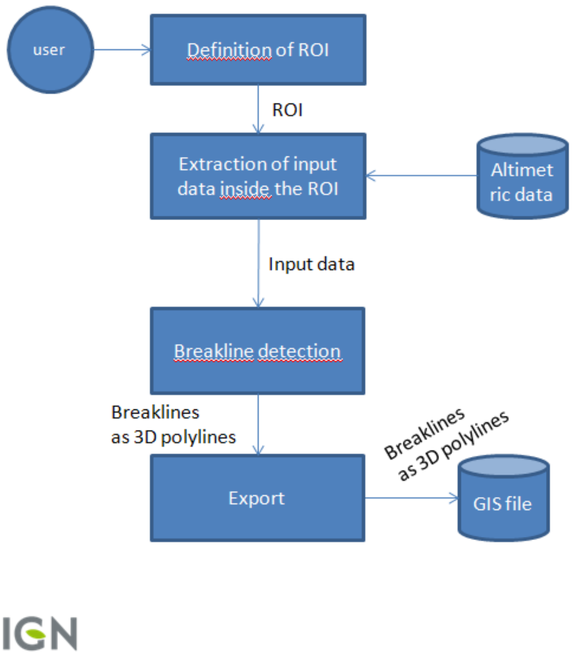
Functional requirements	*Use pre-defined parameters that can be modified by the user (output resolution, regularization method, data selection criteria : date, accuracy, resolution, sensor type) with the DSL* *Support input data that can be : raster DTM, TINs, points clouds* *Output data is a raster DTM plus metadata (multilayer raster file giving quality indicators for each DTM cell: estimated accuracy, date, sensor type if applicable).*
Non-functional requirements	*Output data quality must be the highest possible according to input data quality*

Identifier	***1.1_50_UC1***
Description	*Change detection between a topographic DB and a new lidar point cloud*
Actors	*System, GIS expert*
Initial conditions	*All datasets are in the same final coordinate system (including vertical)* *LiDAR point cloud must be significantly more recent than DB*
Final results	*Detection of probable changes in the DB*
Main process	*Step 1: the user define a region of interest* *Step 2: the system detects changes in the database by comparison with the point cloud. A change can be: a new object that has to be created, an existing (in the DB) object that must be removed, an existing (in the DB) object that has evolved and must be updated* *Step 3: the system stores the results in the database:* – *Creation of new objects indicating where new features were detected* – *Flag objects to be deleted* – *Flag objects to be modified* *Step 4: the user checks every detected change, validate/unvalidate object deletion and prepare DB updating phase*
Alternative processes	
Exceptional situations	
Processed data	*Topographic database* *LiDAR point cloud*
Generated data	*New objects in the database corresponding to new features detected* *Flagged existing objects corresponding to features that have disappeared* *Flagged existing objects corresponding to features that have changed*
Activity diagram	
Related User Stories	*1.1_50: As a GIS expert in a data production institute, I want to detect evolutions of a landscape by combining topographic database and a newly acquired LiDAR dataset so that I can improve DB updating process.*

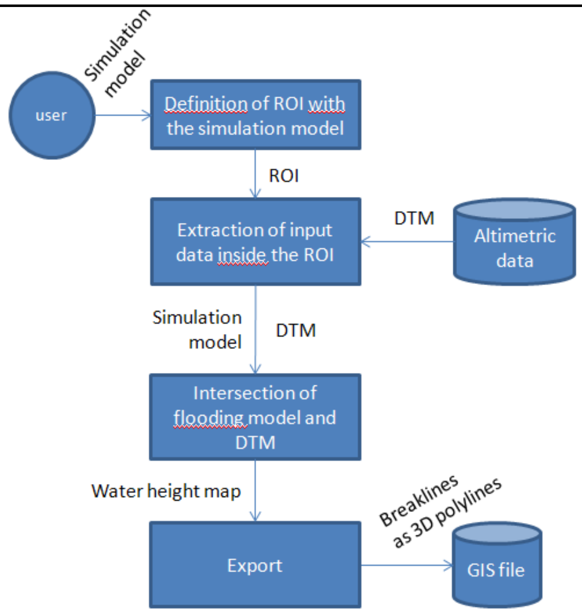
Functional requirements	*Use a predefined geographical area that have been set by the user with the DSL as a region of interest* *Display new and flagged objects to the user in a cartographic user interface*
Non-functional requirement	

Identifier	***1.1_51_UC1 (1.2.1_21_UC1)***
Description	*Extraction of tree parameters from LiDAR point cloud*
Actors	*System, forest expert*
Initial conditions	*LiDAR point cloud must be classified (at least ground class)* *LiDAR point cloud must have sufficient density*
Final results	*Geometric description of detected trees*
Main process	*Step 1: the user define a region of interest* *Step 2: the system detects individual trees* *Step 3: the system compute geometric parameters for each detected tree: height above ground (provided by ground class), crown diameter…* *Step 4: the system export resulting data under the shape of GIS file (2D objects with attributes)*
Alternative processes	
Exceptional situations	
Processed data	*LiDAR point cloud*
Generated data	*3D objects representing trees, with attributes (height, size or canopy diameter…)*
Activity diagram	
Related User Stories	***1.1_51****: As a forest expert, I want to extract information from LiDAR datasets, such as vertical distribution of wood/leaves, tree position, height, size or canopy diameter so that I can populate a forest inventory DB.* ***1.2.1_21****: As a GIS expert (geodata provider/integrator) working at local level, I want to automatically detect individual trees from a LiDAR point cloud in a urban area, so that I can monitor growth and foresee pruning works.*

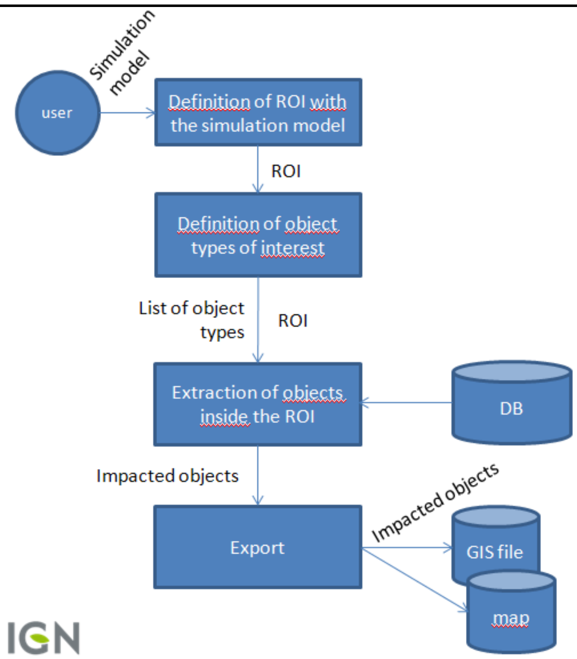
Functional requirements	*Use a predefined geographical area set by the user with the DSL as a ROI* *Load LiDAR points inside the ROI*
Non-functional requirements	

Identifier	***1.1_52_UC1***
Description	*Breakline extraction from altimetric data*
Actors	*hydrologist, System*
Initial conditions	*Resolution high enough to detect relevant features*
Final results	*Breaklines (3D polylines)*
Main process	*Step 1: the user defines the geographic area* *Step 2: the system extracts input data inside the ROI* *Step 3: the system detects the breaklines* *Step 4: the system exports the breaklines as 3D polylines*
Alternative processes	*Add an identification/validation step by the user*
Exceptional situations	
Processed data	*Altimetric data*
Generated data	*Breaklines*
Activity diagram	
Related User Stories	*1.1_52 As a hydrologist, I want to extract significant breaklines from high resolution DTMs or point clouds, such as dikes, dams, banks, ditches so that I can improve my hydraulic model and make more accurate simulations.*

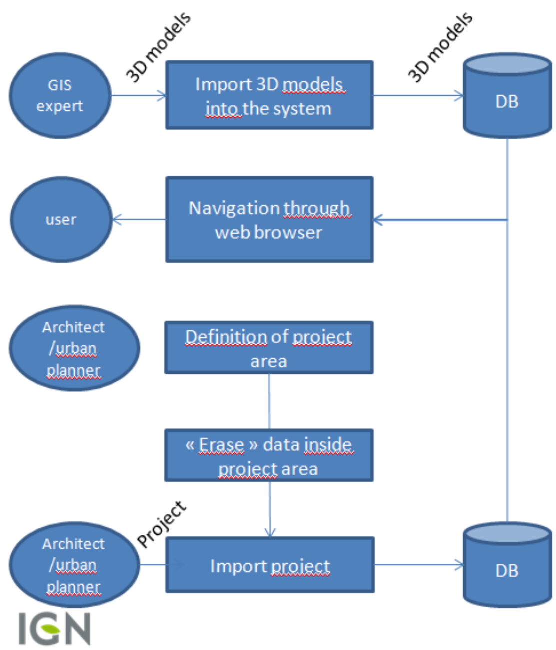
Functional requirements	*Use a predefined geographical area set by the user with the DSL as a ROI* *Support input data that can be : raster DTM, point clouds* *Store extracted breaklines into a GIS file or into a DB*
Non-functional requirements	

Identifier	***1.1_53_UC1***
Description	*Publication of water height map on flooded areas*
Actors	*Hydrologist, System*
Initial conditions	*Data sets must overlap*
Final results	*Water height map*
Main process	*Step 1: The hydrologist defines the geographical area with the flood simulation model* *Step 2: The system extracts DTM onside the ROI* *Step 3: The system intersects the flooded area with the DTM* *Step 4: the system generates a water height map on the flooded area*
Alternative processes	
Exceptional situations	
Processed data	*Flood simulation model* *High resolution DTM*
Generated data	*Water height map*
Activity diagram	 IGN
Related User Stories	*1.1_53 As a hydrologist, I want to combine flood simulation model with high resolution DTM so that I can publish a water height map on the flooded area.*

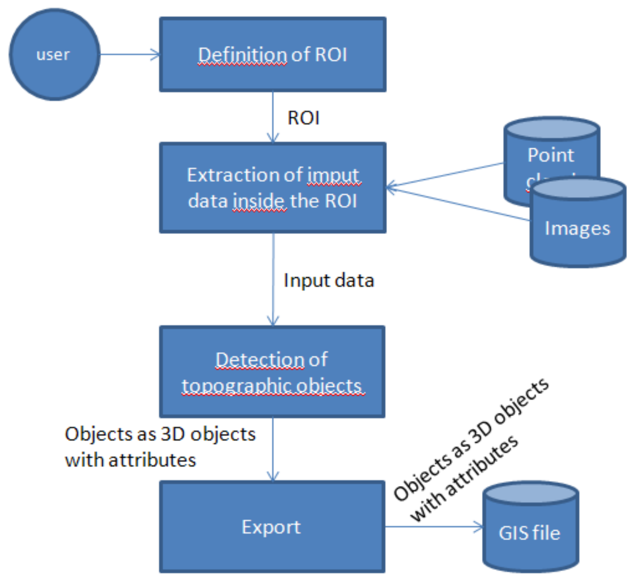
Functional requirements	*Use a predefined geographical area that have been set by the user with the DSL as a ROI* *Supports formats of input data that can be: ESRI Grid, Geo TIFF* *Use pre-defined parameters that can be modified by the user (output resolution) with the DSL*
Non-functional requirements	*Output data quality must be the highest possible according to input data quality*

Identifier	***1.1_54_UC1***
Description	*Information about human and economic stakes on flooded area*
Actors	*hydrologist, System*
Initial conditions	
Final results	*Human and economic stakes maps on flooded area*
Main process	*Step 1: The hydrologist defines the geographical area with the flood simulation model* *Step 1b: the user defines the list of object types of interest (human and economic stakes; e.g. hospital, factories, schools...)* *Step 2: The system intersects the flooded area with the topographic database* *Step 3: The system exports objects impacted by flood into GIS file* *Step 4: The system generates a map with topographic data on the flooded area*
Alternative processes	
Exceptional situations	
Processed data	*Flood simulation model* *Topographic database*
Generated data	*GIS file containing objects impacted by flood* *Human and economic stakes map on flooded area*
Activity diagram	
Related User Stories	*1.1_54: As a hydrologist, I want to combine flood simulation models with all kind of topographic data in order to know the flooded area and all the human and economic stakes impacted so that I can provide decision makers with accurate and reliable information.*

Functional requirements	*Use a predefined geographical area that have been set by the user with the DSL as a region of interest*
Non-functional requirements	

Identifier	***1.2.1_19-20_UC1***
Description	*Online access to 3D city model with different profiles access*
Actors	*GIS expert, citizens, architects, urban planners, System*
Initial conditions	
Final results	*3D light web client*
Main process	***Public city 3D model*** *Step 1: the GIS Expert imports 3D city models in the system* *Step 2: the user have access and navigation capabilities through his web browser* ***Architects/urban planners*** *Step 1b: the GIS Expert imports 3D city models in the system* *Step 2: the architect/urban planner defines the geographic extent of the project* *Step3: the system "erases" all existing objects inside this area* *Step 4: the architect/urban planner imports his project into the system* *Step 5: the user have access and navigation capabilities through his web browser*
Alternative processes	*Provide functions for project evaluation/assessment (user vote, comments, forum…)*
Exceptional situations	
Processed data	*DTM, 3D models, project data*
Generated data	*Visualization, real time navigation*
Activity diagram	
Related User Stories	***1.2.1_19****:As a GIS expert (geodata provider/integrator) working at local level, I want to provide citizens with access to virtual navigation tool through 3D city model using a simple light web client embedded in any kind of web browser.* ***1.2.1_20****: As a GIS expert (geodata provider/integrator) working at local level, I want to provide architects and urban planners online access to city 3D model so that they are able to integrate their projects into the model and share it with administration and citizens for communication and project assessment purposes.*

Functional requirements	*Use a predefined geographical area that have been set by the user with the DSL as a ROI* *Define different levels of access: only visualization (citizens), system management (GIS Expert), ability to add projects (architects/urban planners)* *Allow user (according to his access level) to make all objects (building, vegetation…) inside his project area invisible* *Provide real time navigation with zoom, geographical request, layers manager* *Allow the user to define a geographical area* *Allow the user to use any kind of browser*
Non-functional requirements	*Time response to user request must be as fast as possible.* *Data security must be ensured according to access rights defined*

Identifier	***1.2.1_22_UC1***
Description	*Topographic objects extraction from mobile mapping systems*
Actors	*GIS Expert, System*
Initial conditions	*Resolution high enough to detect relevant features*
Final results	*Topographic objects database*
Main process	*Step 1: the user defines the geographic area* *Step 2: the system detects the topographic objects* *Step 3: the system export detected objects into GIS files*
Alternative processes	*Add an interactive validation/edition phase by the user*
Exceptional situations	
Processed data	*LIDAR point cloud, images*
Generated data	*GIS files containing topographic object data with attributes*
Activity diagram	
Related User Stories	*1.2.1_22: As a GIS expert (geodata provider/integrator) working at local level, I want to capture topographic objects (such as cable networks, street edges, urban furniture, traffic lights...) from data acquired by mobile mapping systems (lidar point clouds, images) so that I can create or update topographic city maps.*

Functional requirements	*Use a predefined geographical area that have been set by the user with the DSL as a ROI* *Support input data that can be : images, points clouds* *Allow user to define which kind of topographic data and attributes to extract*
Non-functional requirements	

5.5 ANNEX 5: USE CASE DESCRIPTIONS PROVIDED BY REGIONE LIGURIA

Identifier	***1.2.1_64_UC1***
Description	***As hydrologist supporting the decision maker in civil protection** I would like to use radar reflectivity field to interpolate observed rainfall data from traditional rain gauge network (punctual data)*
Actors	*Field expert, Regione Liguria - ARPAL*
Initial conditions	*availability of: reflectivity field (dataset 26)* *observed rainfall data from gauge network (dataset 27)*
Final results	*Interpolated observed Precipitation field*
Main process	*Step 1: Load rainfall data:* *(rain gauge data : dataset 27 and radar: dataset 26)* *Step 2: the system interpolates the rainfall measured by gauge net by using the reflectivity radar data (and visualizes the result). Could be performed a comparison with interpolated rainfall data: dataset 24* *Step 3: the system visualizes the results*
Alternative processes	*Only one of the fields could be available*
Exceptional situations	*The data are partially missing or affected by noise (outliers)*
Processed data	*rainfall field:radar dataset 26,dataset 27 and dataset 24*
Generated data	*Rainfall measurements (a raster image)* *Binary file containing the rainfall field over the points of the grid*
Activity diagram	*An activity diagram showing the main process diagrammatically.*
Related User Stories	*1.2.1_66: I would like to integrate rainfall measures at the Regional level, as provided by ARPAL in Liguria, with rainfall measures gathered from semi-professional gauge network – (Genova Municipality)* *1.2.1_74: As hydrologist supporting the decision maker in civil protection I would like to have a 3D (volumetric) visualization of the rainfall field derived from radar overlapped with DTM model*

Functional requirements	Data alignment (gauge network and radar sources with the same DTM grid Field interpolation Visualization of the field over a DTM model
Non-functional requirements	Interaction with the output (non capisco cosa voglia dire) Time constrains: a few seconds

Identifier	*1.2.1_65_UC1*
Description	***As hydrologist supporting the decision maker in civil protectionI would like to have a shape file of the ligurian small basins (< 10 km2) derived from a high resolution DTM exploiting the available Lidar data of Regione Liguria to derive the DTM and a cutted interpolated rainfall field over the shape file***
Actors	*Field expert, Regione Liguria - ARPAL*
Initial conditions	*Availability of :Lidar, high resolution DTM, map of the rivers, observed interpolated rainfall*
Final results	*Small basins identification (shape file) + cutted rainfall field over the same shape file*
Main process	*Step 1: data load LIDAR: dataset 22-23, map of the rivers (reference map): dataset interpolated rainfall data: dataset 24* *Step 2: the system fuses the data (DTM and LIDAR) to obtain high resolution DTM* *Step 3: features detection based on geometric properties (gradient flow, etc..) to obtain small basins identification* *Step 4: verification-overlapping on a reference map (river map)*
Alternative processes	
Exceptional situations	
Processed data	*LIDAR: datasets 22 and 23, Rivers dataset* *observed interpolated rainfall(dataset24)*
Generated data	*small basin shape file*
Activity diagram	*An activity diagram showing the main process diagrammatically.*
Related User Stories	

Functional requirements	*LIDAR segmentation* *Data fusion and interpolation* *Basins identification* *Rainfall identification at basin scale*
Non-functional requirements	*Interaction/selection of the size of the basins* *quality of the results*

Identifier	***1.2.1_74 _UC1***
Description	***As hydrologist supporting the decision maker in civil protection I would like to have a 3D (volumetric) visualization of the rainfall field derived from radar overlapped with DTM model***
Actors	*Decision maker, Field expert, Regione Liguria - ARPAL*
Initial conditions	*Availability of different level reflectivity data(dataset 26)* *and DTM (dataset….)*
Final results	*Visualization of 3D volumetric data over a DTM and basin divide.*
Main process	*Step 1: load reflectivity data (dataset 26) and DTM (dataset..)* *Step 3: 3D volumetric data generation* *Step 4: visualization of the final 3D volumetric data*
Alternative processes	
Exceptional situations	
Processed data	*radar reflectivity data (dataset 26)* *- regione Liguria DTM*
Generated data	*3D (volumetric) data of the terrain and the rainfall field volumetric data*
Activity diagram	
Related User Stories	***1.2.1_64****: As hydrologist supporting the decision maker in civil protection I would like to use radar reflectivity field to interpolate observed rainfall data from traditional rain gauge network (punctual data)* ***1.2.1_65****: As hydrologist supporting the decision maker in civil protection I would like to have a shape file of the ligurian small basins (< 10 km2) derived from a high resolution DTM exploiting the available Lidar data of Regione Liguria to derive the DTM and a cutted interpolated rainfall field over the small basin shape file*

Functional requirements	– *Interactive data visualization/navigation* – *real time exploration of data*
Non-functional requirements	*possibly, rainfall field updated on-the-fly*

Identifier	***1.2.1_85_UC1***
Description	***As meteorologist supporting the decision maker in civil protection I would like to have an operational tool to identify similar shape and features of two rainfall fields: observed and forecasted fields***
Actors	*Field expert, Regione Liguria - ARPAL*
Initial conditions	*Availability of observed interpolated data (US64 and dataset 24) and forecasted rainfall field (dataset 25 and 28)*
Final results	*Comparison of forecasted rainfall with the observed rainfall fieldand identification of an index of the forecasted rainfall precision*
Main process	*Load:* *forecasted rainfall field (dataset 25-28)* *interpolated rainfall field (dataset 24) + US64* *comparison of the two datasets and precision evalueting*
Alternative processes	
Exceptional situations	
Processed data	*- integrated rainfall measurements (dataset 24)* *- forecasted rainfall field (dataset 25-28)*
Generated data	*Image of the differences between the two fields at different time step (hourly data)an image of the differences between the two maps, and an image of the index*
Activity diagram	
Related User Stories	***1.2.1_64***

Functional requirements	
Non-functional requirements	

Identifier	**1.2.1_74 _UC1**
Description	Integrated visualization of rainfall field, DTM and drainage basins **As a decision maker in water management, I would like to have a 3D (volumetric) visualization of the rainfall field integrated with a visualization of the DTM and drainage basins**
Actors	Decision maker, Field expert
Initial conditions	- integration of rainfall measurements done(US 64) - drainage basins extraction done (US 65)
Final results	Visualization of 3D volumetric data
Main process	Step 1: the decision maker imports data in the system Step 2: rainfall field data are integrated with DTM Step 3: 3D volumetric data generation Step 4: visualization of the final 3D volumetric data
Alternative processes	
Exceptional situations	
Processed data	- integrated rainfall measurements (US 64) - segmented Lidar data (US 65) - regione Liguria DTM
Generated data	3D (volumetric) data of the terrain and the rainfall field
Activity diagram	
Related User Stories	**1.2.1_64**: As a decision maker in water management, I would like to integrate rainfall radar measurements in the precipitation field computed from the rainfall measured by the gauge network **1.2.1_65**: As a decision maker in water management, <u>I would like to exploit available Lidar data of Regione Liguria to extract small scale drainage basins</u>

Functional requirements	– *Multivariate data generation (Task 4.4)* – *Interactive data visualization/navigation* – *real time exploration of data*
Non-functional requirements	*possibly, rainfall field updated on-the-fly*

Identifier	*1.2.1_67_UC1*
Description	*Preparation of risk maps to support crisis management and development of civil protection plans* ***I would like to have easy tools to integrate up-to-date data about hydraulic, hydro-geological, seismic, industrial risk for the production of multi-risk maps***
Actors	*Decision maker; Administrator; Field expert*
Initial conditions	*The DTM is done* *The hydraulic risk map is done* *The earthquake map is done* *The map of the sensitive and strategic buildings are provided* *The map of streets and secure areas is provided (datasets 29, 30)* *The administrative limits map is done* *The geological risk map is provided (dataset 31)*
Final results	*A multi-risk map*
Main process	*Step 1: the user selects the geographical area of interest* *Step 2: The system aligns the data (at different scales)* *the system regularizes the data* *the system fuses the data* *the system interpolates the different maps with the DTM* *Step 3: the system visualizes the interpolation* *Step 4: The RS Expert refines the results (optional)*
Alternative processes	
Exceptional situations	*The data are not updated*
Processed data	*DTM* *raster images* *text data* *various layers of information, at Regional and City scale, datasets 29, 30, 31: format Access*
Generated data	*Multi-risk map* *classification*
Activity diagram	*An activity diagram showing the main process diagrammatically.*
Related User Stories	*Drainage basin map in the 1.2.1_65: I would like to exploit available Lidar date of Regione Liguria to extract small scale drainage basins (< 15 km2) which are essential for evaluating the potential risks of flooding* *and also the risk US74: I would like to have a 3D (volumetric) visualization of the rainfall field integrated with a visualization of the DTM and drainage basins*

Functional requirements	classification multi-layer classification visualization of a single map and multiple maps visualization of secure areas with a visualization of the multi risk map
Non-functional requirements	Friendly change of the visualization layer

Identifier	*1.2.1_69_UC1*
Description	*Integration of LIDAR and historical data (DTM) with sensor sonar data of the coast line and seafloor in a unique surface model* ***I would like to have tools for integrating data about the coast line and seafloor in a unique surface model which allow me to work with large data (Lidar), with historical data (validation, reliability of time series), with better integration results at the border of tiles***
Actors	*GIS expert; Field expert*
Initial conditions	*The LIDAR model is done* *The DTMs of the historical data is done* *The sensor sonar data are provided*
Final results	*Surface model of the coastal area*
Main process	*Step 1: the user selects the geographical area* *Step 2: the system regularizes the DTMs* *the system aligns the DTMs and the LIDAR model* *the system fuses the data into the LIDAR model* *the system classifies/segments the data wrt the geometric/morphological characteristics and the sensor information* *Step 3: the system detects the coastal line* *Step 4: the system visualizes the model*
Alternative processes	
Exceptional situations	
Processed data	*LIDAR (datasets 22, 23 in x,y,z and LAS format)* *time series of data, multibeam points (dataset 17) or depth grid (dataset 18)* *raster images (old maps of the Genoa gulf)* *Sonar images (.tiff?)*
Generated data	*DTM of the coastal area*
Activity diagram	*An activity diagram showing the main process diagrammatically*
Related User Stories	*1.2.1_71: I would like to have effective 3D visualization and navigation tools for coastal area exploration* *1.2.1_72: I would like to have multi-resolution surface models of the coastal zone to see more details only where important and improve interactivity* *1.2.1_73: I would like to have tools for the analysis of the geomorphology of the seafloor* *1.2.1_79: I would like to have tools to study the dynamics of sediments in the seafloor (volumes of sediments moved in temporal series)* *1.2.1_80: I would like to be able to handle Lidar data more easily and integrate them in DTM: now the size is really a limiting factor*

Functional requirements	Alignment, fusion, regularization, interpolation visualization of the surface model
Non-functional requirements	Slider to move from an historical view to another quality is important

Identifier	*1.2.1_70_UC1*
Description	*Data format conversion* ***I would like to easily convert the surface model of the seafloor (regular grid, xyz) into the format required by other groups of the Regione Liguria (eg, Civil Protection) or Municipaities***
Actors	*GIS expert; Field expert*
Initial conditions	*The x, y, z regular grid is available*
Final results	*Other data formats*
Main process	*Step 1: the system loads the DTM* *Step 2: the user selects the new data format* *Step 3: the system stores the data in the new format*
Alternative processes	
Exceptional situations	*The data formats wished aren't considered/supported or are incompatible*
Processed data	*x, y, z regular grid (dataset 17)*
Generated data	*other formats (DTM, raster, etc)*
Activity diagram	*An activity diagram showing the main process diagrammatically.*
Related User Stories	

Functional requirements	*Data conversion* *Choice of the data type in output*
Non-functional requirements	*Friend visualization?* *Quality is important*

Identifier	***1.2.1_71_UC1***
Description	*Effective 3D data visualization and navigation* **As a GIS expert for coastal monitoring, I would like to have effective 3D visualization and navigation tools for coastal area exploration**
Actors	*GIS expert, Field expert*
Initial conditions	*The surface model of the coastal area is done (1.2.1_69)*
Final results	*3D data visualization and navigation tool*
Main process	*Step 1: the GIS expert imports data (the surface model) in the system* *Step 2: the GIS expert has visualization and navigation capabilities through his/her workstation* *Step 3: the GIS expert classifies data according to his/her needs*
Alternative processes	
Exceptional situations	
Processed data	*Surface model (1.2.1_69)*
Generated data	*Surface model of the coastal area with classification of any relevant parameters/attributes*
Activity diagram	
Related User Stories	***1.2.1_69**: As a GIS expert for coastal monitoring, I would like to have tools for integrating data about the coast line and seafloor in a unique surface model…* ***1.2.1_72**:As a GIS expert for coastal monitoring, I would like to have multi-resolution surface models of the coastal zone…*

Functional requirements	*Feature extraction and classification (Task 4.3)* *Real time visualization*
Non-functional requirements	*Allow user to visualize data according to a selection of parameters/attributes*

Identifier	*1.2.1_72 _UC1*
Description	*Multi-resolution data analysis* **As a GIS expert for coastal monitoring, I would like to have multi-resolution surface models of the coastal zone to see more details only where important and improve interactivity**
Actors	*GIS expert, Field expert*
Initial conditions	*The surface model of the coastal area is done (1.2.1_71)*
Final results	*multi-resolution surface model of the coastal area*
Main process	*Step 1: the GIS expert imports data in the system* *Step 2: part of data is simplified according to relevant parameters/attributes* *Step 3: A multi-resolution surface is generated*
Alternative processes	
Exceptional situations	*Natural emergencies may result in the need of updated models*
Processed data	*Surface model of the coastal area with classification of relevant parameters/attributes (1.2.1_71)*
Generated data	*Multi-resolution surface model of the coastal area*
Activity diagram	
Related User Stories	***1.2.1_71**: As a GIS expert for coastal monitoring, I would like to have effective 3D visualization and navigation tools for coastal area exploration* ***1.2.1_73**:As a GIS expert for coastal monitoring, I would like to have tools for the analysis of the geomorphology of the seafloor* ***1.2.1_75**:As a GIS expert for coastal monitoring, I would like to have tools to study the dynamics of sediments in the seafloor (volumes of sediments moved in temporal series)*

Functional requirements	– *Data simplification, surface generation (Task 4.3);*
Non-functional requirements	– *no time criticality* – *high data quality is required* – *Relevant parameters/attributes modifiable by the user*

Identifier	*1.2.1_73 _UC1*
Description	*Analysis of the geomorphology of the seafloor* **As a GIS expert for coastal monitoring, I would like to have tools for the analysis of the geomorphology of the seafloor.**
Actors	*GIS expert, Field expert*
Initial conditions	*The surface model of the coastal area is done (1.2.1_69, 1.2.1_71)* *seafloor morphology acquired*
Final results	*Seafloor geomorphology analysis*
Main process	*Step 1: the GIS expert imports data in the system* *Step 2: the surfaces model is integrated with the seafloor morphology and the sonar data* *Step 3: the GIS expert has analysis capabilities through his/her workstation* *Step 4: enriched surface model generation*
Alternative processes	
Exceptional situations	
Processed data	*surface model of the coastal area (1.2.1_69, 1.2.1_71)* *seafloor morphology(multibeam points,,grd data, see dataset #17)* *sonar data (back scattering, .geotiff)*
Generated data	*Surface model of the coastal area, with geomorphology*
Activity diagram	
Related User Stories	***1.2.1_69**: As a GIS expert for coastal monitoring,I would like to have tools for integrating data about the coast line and seafloor in a unique surface model…* ***1.2.1_71**: As a GIS expert for coastal monitoring, I would like to have effective 3D visualization and navigation tools for coastal area exploration* ***1.2.1_72:** As a GIS expert for coastal monitoring, I would like to have multi-resolution surface models of the coastal zone to see more details only where important and improve interactivity*

Functional requirements	– *Feature extraction, classification and correlation for, e.g., seafloor typology (rocks, vegetation…) recognition (Task 4.3)* – *Multivariate surface generation (Task 4.4)*
Non-functional requirements	– *No time criticality;* – *High data quality is required*

Identifier	*1.2.1_75_UC1*
Description	*Study and analysis of sediments dynamics in the seafloor* **As a GIS expert for coastal monitoring, I would like to have tools to study the dynamics of sediments in the seafloor (volumes of sediments moved in temporal series)**
Actors	*GIS expert, Field expert*
Initial conditions	*- Surface model of the coastal area (with geomorphology) is done (1.2.1_73)* *- Granulometric data acquired*
Final results	*Tools for the study and analysis of sediments dynamics in the seafloor*
Main process	*Step 1: the GIS expert imports data in the system* *Step 2: surface model is integrated with granulometric data;* *Step 2: the GIS expert has analysis capabilities (see functional requirements) through his/her workstation;* *Step 3: enriched surface model generation*
Alternative processes	
Exceptional situations	
Processed data	*Surface model of the coastal area (with geomorphology)(1.2.1_73)* *Granulometric analysis (Excel XYZ points)*
Generated data	*Surface model of the coastal area, with quantitative characterization of the changes in the sediment location*
Activity diagram	
Related User Stories	***1.2.1_72****: As a GIS expert for coastal monitoring, I would like to have multi-resolution surface models of the coastal zone to see more details only where important and improve interactivity* ***1.2.1_73****: As a GIS expert for coastal monitoring, I would like to have tools for the analysis of the geomorphology of the seafloor*

Functional requirements	– *analysis and detection of the dynamic of sediments (Change detection and dynamics, task 4.5)* – *bathymetric comparison for volume and/or sediment movements evaluation (temporal series) (Task 4.5)* – *Multivariate surface generation (Task 4.4)*
Non-functional requirements	– *no time criticality* – *high data quality is required*

Identifier	*1.2.1_80_UC1*
Description	*Easy use of Lidar data and integration with DTM* ***I would like to be able to handle Lidar data more easily and integrate them in DTM: now the size is really a limiting factor***
Actors	*Geodata provider/integrator; land cover/ land use monitoring*
Initial conditions	*Availability of Lidar and DTM models of the same area*
Final results	*Integration of Lidar and DTM*
Main process	1. *Collection of Lidar and/or DTM data related to the overlapped areas [Data sets: 17, 18, 22, 23]* 2. *Identification of a common area where DTM and Lidar data are available* 3. *Selection of methods for fusing DTM and Lidar data* 4. *Integration of DTM and Lidar data* 5. *Validation of the accuracy and reliability of the final result*
Alternative processes	
Exceptional situations	*No areas where different data types, representation, or versions are available*
Processed data	– *Lidar data* – *DTM* – *Information on reliability and accuracy is available as input or computed (output)*
Generated data	– *Fusion of Lidar data DTM* – *Reliability, accuracy maps*
Activity diagram	
Related User Stories	*1.2.1_78 CNR-IMATI: I would like to be able to assess the validity/reliability of various types and versions of data about the environment (e.g., DTM, sensor data, time series).*

Functional requirements	*Tools for region selection, data fusion, validation of data accuracy and reliability*
Non-functional requirements	– *Response time* – *Robustness* – *Scalability* – *Stability* – *Usability by target user community*

5.6 ANNEX 6: PRIORITIZED USER STORIES FROM D1.1AND D1.2.1