D10.5 Requirements and gap analysis report v1

Lead Partner:	INFN
Version:	1
Status:	FINAL
Dissemination Level:	Public
Document Link:	https://documents.egi.eu/document/3463

Deliverable Abstract

One of the main goals of the Technical Coordination Work Package (WP10) in the EOSC-hub project is to play a supportive role to the participants (e.g. Thematic Services, Competence Centres and Business pilots) and external user communities (not directly involved in the project but engaged through other channels, e.g. the EOSC Portal) in the integration of their services into the Hub. In order to better structure and consolidate this support, WP10 has defined and is currently operating procedures to elicit, assess and track the technical requirements of such communities and, consequently, provide adequate assistance to the user communities to adopt the service of the Hub. The procedures described in this document cover also the cases when gaps (e.g. lacking one or more features) have been identified in the EOSC-hub service offer and, eventually, triggering other project activities that will take care of extending the offer accordingly. From a technical standpoint, the technical support activity is operating since the first months of the project. This allowed to extract metrics, suitable to depict the advancement achieved by each use case in the EOSC-hub integration path. These metrics are reported in this document (Section 6). Finally, details are provided about the outcome of the technical support activities for three exemplars use cases: DODAS and OPENCoastS thematic services and the ELIXIR competence centre.

COPYRIGHT NOTICE

DELIVERY SLIP

Date	Name	Partner/Activity	Date
From:	Giacinto Donvito Diego Scardaci	INFN/WP10 EGI Foundation/WP1	1/05/2019
Moderated by:	Malgorzata Krakowian	EGI Foundation/WP1	
Reviewed by:	Kostas Koumantaros Giuseppe Lar Rocca	GRNET/WP5 EGI Foundation/WP11	1/06/2019
Approved by:	AMB		25/06/2019

DOCUMENT LOG

Issue	Date	Comment	Author
v.1	01/03/2019	First draft ready for internal review	Pablo Orviz (CSIC), Diego Scardaci (EGI.eu) Giacinto Donvito (INFN)
v.2	01/04/2019	Feedback from T10.3 and other WP10 Members	Pablo Orviz (CSIC), Diego Scardaci (EGI.eu) Giacinto Donvito (INFN)
v.3	23/04/2019	Comments from internal review incorporated. Document ready for external review	Pablo Orviz (CSIC), Diego Scardaci (EGI.eu) Giacinto Donvito (INFN)
v.4	03/06/2019	Implemented changes suggested by reviewers	Diego Scardaci (EGI.eu)
v.5	04/06/2019	Minor changes to satisfy further comments from reviewers	Diego Scardaci (EGI.eu)
v. 6	05/06/2019	Minor changes	Diego Scardaci (EGI.eu)

TERMINOLOGY

https://wiki.eosc-hub.eu/display/EOSC/EOSC-hub+Glossary

Contents

Executive summary

One of the main goals of the Technical Coordination Work Package (WP10) in the EOSC-hub project is to play a supportive role to the participants (thematic services, competence centres and business pilots) and external user communities (not directly involved in the project but engaged through other channels, e.g. the EOSC Portal) in the integration of their services into the Hub. In order to better structure and consolidate this support, WP10 has defined and is currently operating procedures to elicit, assess and track the technical requirements of such communities and, consequently, provide adequate assistance to the user communities to adopt the service of the Hub. The procedure described in this document cover also the cases when gaps (e.g. lacking one or more features) have been identified in the EOSC-hub service offer and, eventually, after the analysis of a use case, triggering other project activities that will take care of extending the offer accordingly.

The T10.3 procedures to support customers have well-defined interfaces towards other project activities and processes of the EOSC Service Management System (SMS). These interfaces are shortly described below:

- Interface towards the *Service Order and Customer Relationship Management* (SOCRM) process (T4.2) and WP7 (Thematic services), WP8 (Competence Centres), WP9 (Business pilots) to receive new use cases and technical requirements.
- Interface towards *Service Portfolio Management* (SPM) process (T2.2), T10.1 *Technical Roadmap* and T10.2 *Service Catalogue Technical Evolution* to extend the service offer of the Hub when gaps have been identified after the analysis of a use case. This extension can be achieved either enhancing services already available in the service catalogue or enriching the catalogue with new services.
- Interface towards the *Change Management* (CHM) and *Release and Deploy Management* (RDM) processes (T4.6) to implement changes on the services to satisfy customer requirements.

When a new request from a user community is identified, through one of the above mentioned interfaces, T10.3 starts a process to gather technical requirements inspired in an agile software development top-down approach where, starting from the high-level *user stories*, *use cases* are defined and linked to the *technical requirements* that derive from them. This information is stored in the Community Requirements DB[1] and permanently reviewed by the technical support team in collaboration with the community stakeholders. Once the requirements are eligible for a deeper technical assessment, T10.3 creates an associated, one-to-one, issue in the Activity dashboard in JIRA[2].

---

[1] https://wiki.eosc-hub.eu/display/EOSC/Community+requirements+DB
[2] https://jira.eosc-hub.eu/projects/EOSCWP10/summary

All the defined steps to provide technical support, and the related interfaces towards processes and project activities, have been embedded in a procedure (SOCRM-04[3]) of the SOCRM process[4] of the EOSC SMS with the aim to:

- create a well-defined communication channel between the technical support team and people on the first line of support, working with requests from user communities received through the EOSC Portal or other channels.,
- make homogeneous the communication with the different customers, and
- simplify the work of the technical supporters that have clear guidelines to follow.

The technical support activity is operating since the first months of the project and the Community Requirements DB already provides detailed information about a relevant number of use cases analysed by WP10. Likewise, the JIRA issue tracking system contains the discussions and technical information (knowledge base) gathered throughout the lifetime of each technical requirement. This allowed to extract metrics, suitable to depict the advancement achieved by each use case in the EOSC-hub integration path. These metrics are reported in this document (Section 6). Finally, more details are provided about the outcome of the technical support activities for three exemplars use cases: DODAS and OPENCoastS thematic services and the ELIXIR competence centre.

---

[3] https://wiki.eosc-hub.eu/display/EOSC/SOCRM-04+Provide+technical+support
[4] https://wiki.eosc-hub.eu/display/EOSC/Service+Order+and+Customer+Relationship+Management+-+SOCRM

1 Introduction

The present document contains the first series of the Requirements and Gap Analysis report. Overall, this report focuses on describing the initial procedures elaborated and followed by WP10 in order to operate the whole requirement analysis workflow. The interfaces with the connecting processes - both internal and external to the WP - are described, as well as the shared tools required to implement them. Finally, an up-to-date report about the progress in the EOSC-hub integration path of each use case is provided with three exemplars use cases described in detail.

2 Interfaces with other project activities and EOSC SMS processes

The technical support activity (T10.3) receives requests for support through different channels including:

- SOCRM process behind the EOSC Portal,
- WP7 Thematic Services,
- WP8 Competence Centres,
- WP9 Digital Innovation Hub, and
- WP3 Stakeholder Engagement).

After the cases are analysed T10.3 take cares of providing technical assistance to the user communities to adopt the service of the Hub leveraging, when needed, on the support given by the other project activities.

Furthermore, when a gap has been identified, T10.3, in conjunction with the gap analysis activity (T10.4), collaborates with other project technical activities (WP5, WP6 and WP7) to extend the service offer of the Hub. This extension could be achieved either enhancing services already available in the service catalogue or enriching the catalogue with new services that satisfy users' needs.

For these reasons, a successful coordination is requested between the involved work packages, as described hereunder:

- Strategy and Business Development (WP2)
  - o In particular, the task *Service Roadmap, Service Portfolio and Service Catalogue activity* (T2.2) manages the service roadmap and both internal and external service catalogues, in coordination with the Technology Committee (TCOM), designed by the *Technical Roadmap* activity (T10.1).
- Federated Service Management (WP4)
  - o The *Service Order and Customer Relationship Management* (SOCRM, T4.2) process of the EOSC SMS handles orders initiated by customers from user communities from the EOSC Portal. Orders can be requests for the usage of specific services or more sophisticated demands that involve multiple services (integrations and/or customisations) that may exist or not in the service catalogue.
  - o The *Configuration Management, Change Management, Release and Deployment Management* process of the EOSC SMS (T4.6) defines and implements ITSM best practices for the proper maintenance and update of the services and service components.
- Technical Coordination (WP10), where:
  - o T10.1 provides the technical roadmap leveraging the expertise of the TCOM to scout emerging technologies, analyse and prioritise requirements to be proposed for AMB/PMB approval. Major changes in the service catalogue are agreed with T2.2.
  - o T10.2 defines and assesses conformance of services to the Rules of Participation.

o T10.3 gathers, in a first phase, technical requirements from use cases, both internal (e.g.: Thematic Services, Competence Centres, Business Pilots) and external to the project (e.g.: identified by the Stakeholder Engagement activity or through an order received from the EOSC Portal). Further requirement analysis, tracking and technical support is provided in order to allow customers to fully exploit the services in the catalogue.

o T10.4 performs a deeper analysis of the requirements not satisfied by the current service catalogue. These requirements are first identified by T10.3, flagged as technical gaps and handed over to this task for further evaluation. Solutions that imply major changes in the service catalogue are evaluated in collaboration with T2.2.

Figure 1 describes interfaces between T10.3 and T10.4 with other project activities and EOSC-hub SMS processes, highlighting the impact of the requirement and gap analysis on the service catalogue and on its evolution. All the interactions are described later in this section.
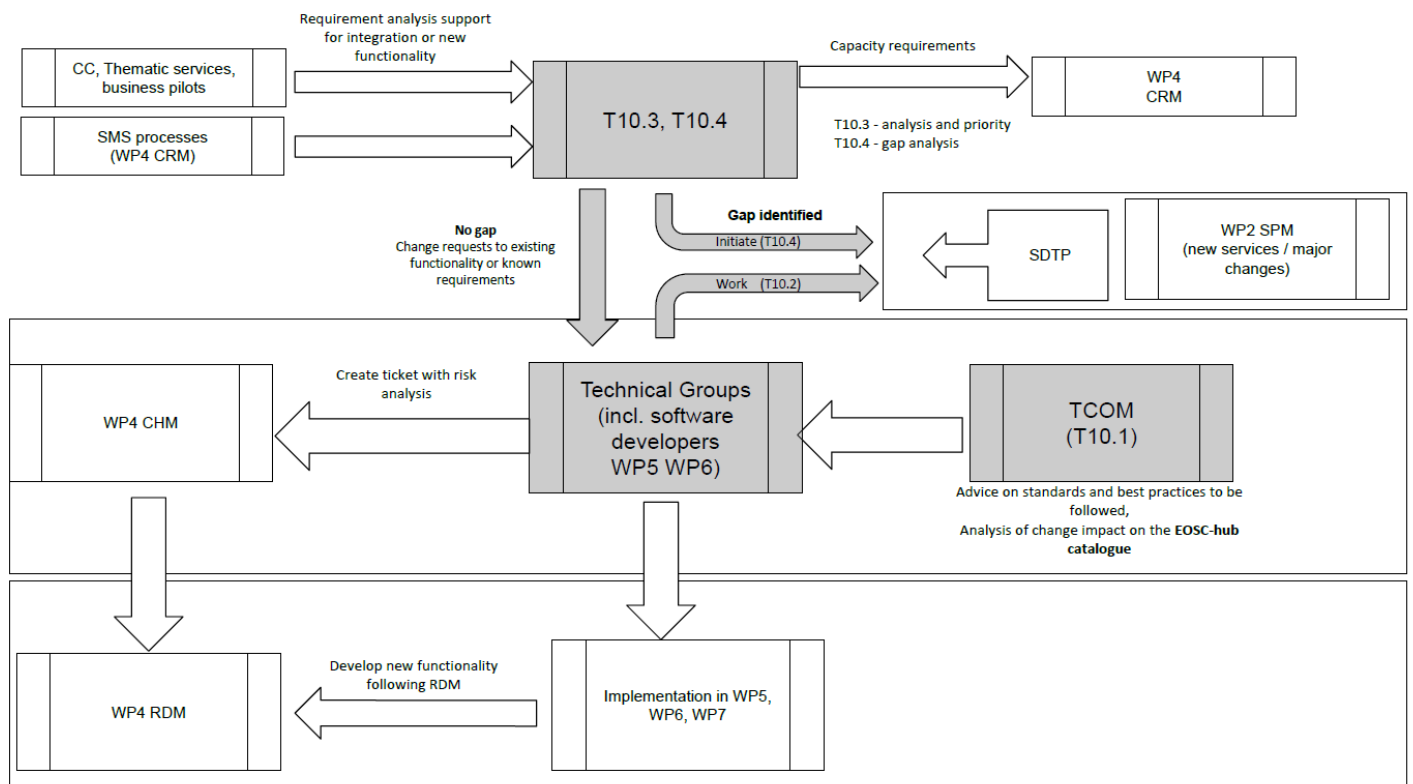

Figure 1. Evolution of the service catalogue driven by the requirements and gap analysis.

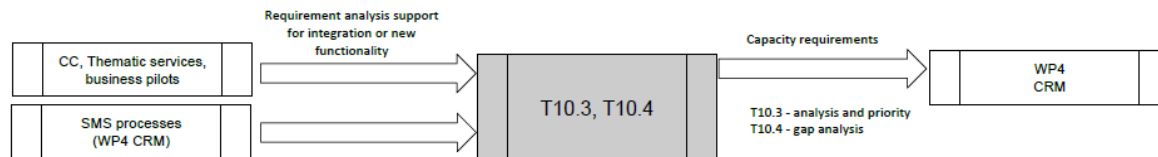## 2.1 WP10 link with use cases (WP7, WP8, WP9) and SOCRM process (T4.2)

Figure 2. WP10 analyses and supports requirements from both internal and external use cases.

As furtherly explained in the Section 4, WP10 manages the process of gathering, analysing and tracking technical requirements extracted from the feedback that is continuously obtained from the project's use cases, i.e. Thematic Services (WP7), Competence Centres (WP8) and Business Pilots (WP9), together with those identified by the SOCRM process and the WP3 Stakeholder Engagement activity. Figure 2 depicts the different sources from which the requests for requirement handling may come from.

In what regards to orders coming from external communities, they are received from a single point of access, represented by the EOSC marketplace (a key component of the EOSC Portal), and subsequently processed by the SOCRM process (T4.2). SOCRM uses the same ticketing system (see Section 4 – *Tools to record and handle requirements*) to tracks each order and handle the internal requirements. Consequently, whenever a technical assessment is required, T4.2 can assign the ticket to WP10.3 that will perform the requirements and gap analysis.

2.1.1 SOCRM process (T4.2) and Technical Support (T10.3) interaction

By the time of writing of this document, according to the number of services requested, we can distinguish three different types of orders:

1. Orders related to a unique service or, more in general, to a unique provider;
2. Orders that involve multiple services managed by more than a single provider;
3. Orders that do not or incorrectly identify services.

T10.3 intervention is required when the SOCRM receives a not well-defined order (e.g. the customer wants to achieve something without a clear idea of which services should use), or an order that requires the combined usage of services from different providers because this could raise the need to implement or enable integration between services.

When multiple services from different providers are required, T10.3 evaluates the suitability of the solution by analysing compatibility issues, identifying the potential technical gaps, and suggesting alternative options when needed. In those cases where orders do not detail the individual services, or they are wrongly formulated, T10.3 is involved by SOCRM to help on identifying the most suitable services. This is done with dedicated meetings with the communities, interviews, sharing of information about services, etc.
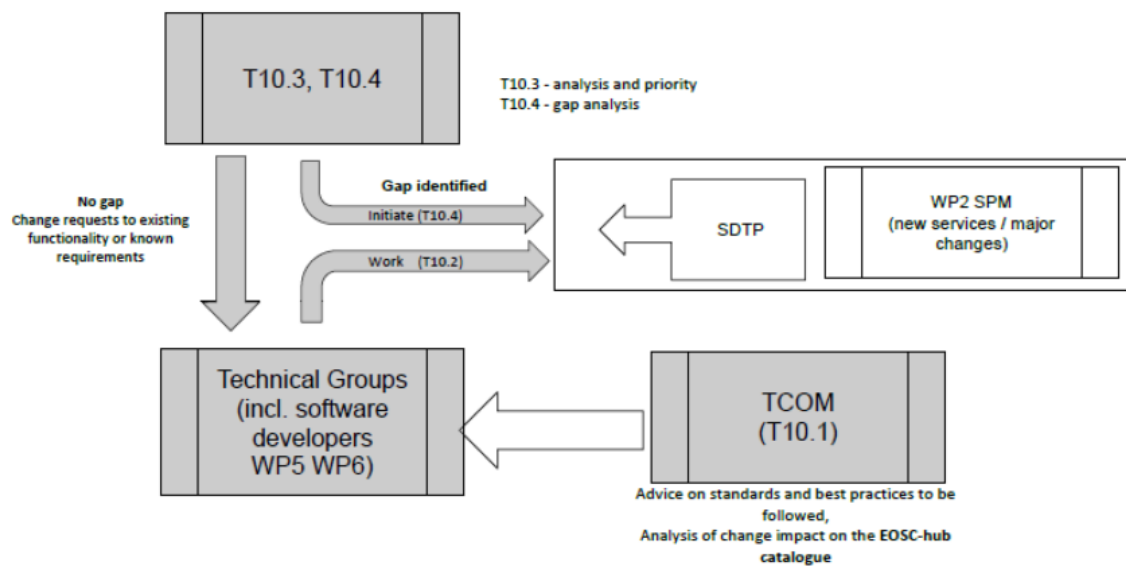
2.2 WP10 and SPM process interaction

Figure 3. Workflows initiated (gap and no gap) by WP10 to tackle the technical requirements implementation.

Technical requirements can be either satisfied by the set of services existing in the service portfolio or conversely, the required functionality cannot be addressed by extending the present services but only by the usage of off-the-shelf software. In the latter case, a technical gap is issued by the support groups in T10.3. Gap analysis is performed by T10.4 in partnership with T10.2, leveraging on the expertise of Technical Committee (TCOM) and technical groups (from WP5, WP6 and WP7) and taking into account advice on standards and best practices. The outcome of this gap analysis is the agreement on the solutions required revised from state-of-the-art open source technologies, once the analysis of the change impact in the EOSC-hub catalogue and the compliance with standards and best practices have been assessed. In case no solutions are identified during this analysis, WP5. WP6 and WP7 will be initially contacted to implement the missing features. Other solutions, outside EOSC-hub, will be investigated in case the WP5, WP6 and WP7 effort requested to implement the new solution is not enough. Solutions that imply major changes to the service portfolio are ultimately evaluated in collaboration with T2.2, as part of the portfolio management. In the latter case, a Service Description Transition Package (SDTP) is created in order to ensure proper evaluation, define the necessary pieces of information regarding the service design, delivery and transition planning according to the SPM process of the EOSC SMS.

When there are no technical gaps in one or several services from the catalogue, T10.3 assigns the requirement to the Technical Groups in WP5, WP6 and WP7 in order to be prioritised and, eventually, implemented.

2.3 WP10 and CHM/RDM processes interaction
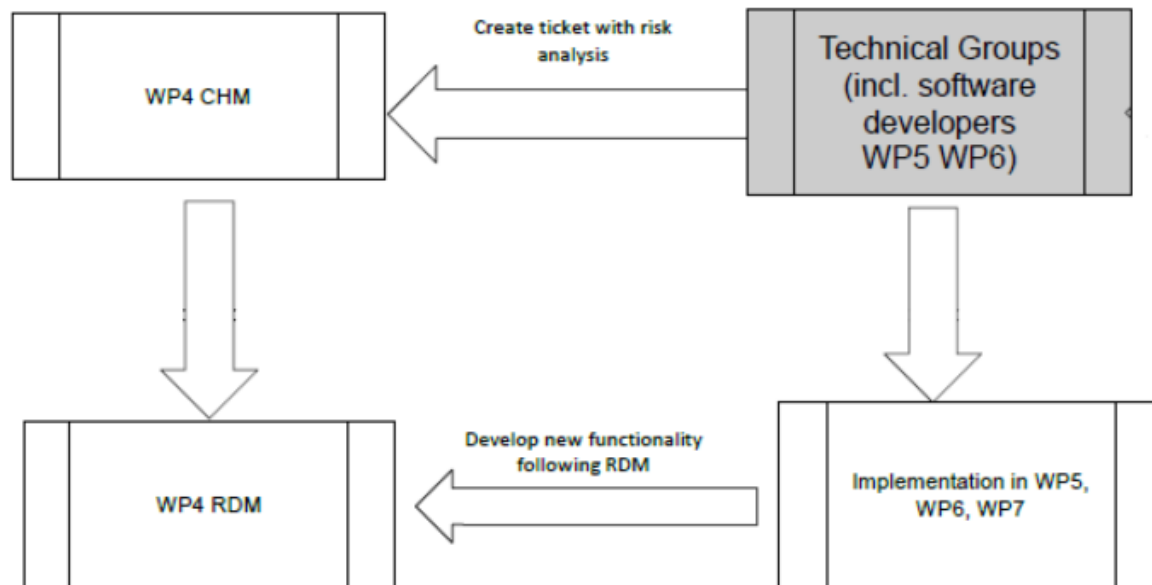
Figure 4. WP4 drives the implementation and release of each new change that may be derived from the assessment of the technical requirements.

In order to proceed with the implementation, Technical Groups first perform an initial risk assessment of the change before being registered as a change request, and prioritized, in the Change Management (CHM) process of the EOSC SMS.

The actual implementation of changes and new functionalities are organized and managed by software developers within WP5, WP6 and WP7. The development and delivery of new software changes will follow the Release and Deployment Management (RDM) guidelines, as the final step before making these new functionalities available to the communities that requested them.

3 Gathering technical requirements

When a new request from a user community is identified, as described in the previous section, T10.3 starts the process to gather technical requirements. It is inspired in an agile software development top-down approach where, starting from the high-level *user stories*, the *use cases* are defined and linked to the *technical requirements* that derive from them. This information is compiled in a document for each analysed use case and stored in the Community Requirements DB as described in Section 4.

User stories and use cases are periodically reviewed by the community stakeholders to keep them updated. Technical requirements will be updated accordingly.

For the sake of completeness, in the next session, we will briefly review the three aforementioned pieces of information that, once extracted, help to identify the expected objectives and outcomes of the different stakeholders involved (i.e. customers and development teams) and facilitate the communication among them.

3.1 User stories

User stories are informal, natural language description of one or more features of a software system. They can be characterized by:

- Description of the benefit and value to the user.
- Outline, in a non-technical language, of the desired outcome of a customer's or user's end goal.
- User stories might take the following forms:
  - "As a <role>, I want <capability> so that <receive benefit>"
  - "In order to <receive benefit> as a <role>, I want <goal/desire>"
  - "As <persona>, I want <what?> so that <why?>" where a persona is a fictional stakeholder (e.g. user). A persona may include a name, picture; characteristics, behaviours, attitudes, and a goal which the product should help them achieve.

The following is an example of user story: "*As provider of the Climate gateway I want to empower researchers from academia to interact with data sets stored in the Climate Catalogue and bring their own applications to analyse this data on remote cloud servers offered via EGI.*"

3.2 Use cases

Use cases are action lists or event steps typically defining the interactions between a role/actor and a system to achieve a goal. They:

- Describe the interaction between the actor and a system.
- Refer to the list of actions, derived from the user stories, between a role or actor and a system to achieve the goal (where 'actor' can be people or other systems).
- Since a system is involved, the dependencies with EOSC-hub services for each action are already identified at this stage.

- Use case descriptions do not follow a specific format as user stories do, they describe a single action which implies a subject-verb-object structure and the system that is involved in the action.

A simple example of a use case is the following: "*1) Data are selected; 2) Software selected/uploaded; 3) Data staging at HPC resources; 4) Processing of data; 5) Results presented*".

3.3 Technical requirements

Technical requirements describe the technical work that needs to be implemented on the EOSC-hub services to enable the use cases.

- Technical requirements are the small units of development or integration work to be done by the technical groups and product teams within WP5, WP6 and WP7.
- A JIRA issue must be created for each identified new technical requirement.
- Each requirement contains the following information:
    - Formal description.
    - Source Use case.
    - JIRA issue reference.
- Technical support people[5] oversee the implementation of the requirements identifying the best technical solution to satisfy each requirement and acting as a bridge between the customer and the product teams.

---

[5] Members of the technical support team was chosen according to their expertise with the aim to have enough knowledge of all the EOSC-hub services to properly oversee the requirement implementation.

4 Tools to record and handle requirements

EOSC-hub main tools for documentation and issue tracking are, respectively, Confluence and JIRA. WP10 relies on the tight integration among these two tools in order to connect requirement descriptions (Confluence) together with their associated integration and development activities (JIRA). Consequently, customers appear as involved actors within those activities, following their progress, and being able to provide and receive timely feedback.

Each community has assigned a unique WP10 contact. The main role of this responsible person is driving the integration of the service in the Hub by bridging users and development teams, leading the coordination of the technical activities. The main WP10 contact is backed by a team composed by 4/5 additional technical experts from WP10 that are selected, after the use case analysis, according to the specific needs of the customers.

4.1 Community Requirements DB

The primary source of information is the Community Requirements DB. Each entry in the database points to a document containing an analysis of the technical requirements of an internal (e.g. Thematic Service, Competence Centre Business Pilot) or external use case. The structure of this document has three main sections corresponding to the three core components of the agile process described in Section 3 (i.e. user stories, use cases and technical requirements).

Whenever a new use case comes in, WP10 initially holds an interview with the user community's main contacts in order to build a shared understanding that will yield to the elaboration of the user stories. In subsequent iterations, the WP10 responsible will break down the stories into use cases and use cases into technical requirements. The whole process will be transparent for the customers as WP10 works directly in the document --stored in the Community Requirements DB-- at this stage in the technical evaluation process.

4.2 Activity dashboard

Once the requirements are eligible for a deeper technical assessment, the WP10 contact creates the associated, one-to-one, issues in the Activity dashboard in JIRA, under a specific queue[6] (*project* in JIRA terminology) for WP10 (*EOSCWP10*).

Each issue contains additional relevant information, such as:

- Components or services involved from the catalogue.
- Project tasks involved.
- Gap assessment.
- Technical committee (TCOM) area.

---

[6] https://jira.eosc-hub.eu/projects/EOSCWP10/summary

Each JIRA issue is then mapped back to the associated requirement definition described in the Community Requirements DB.

A set of dashboards were created to facilitate the management of the JIRA issues per involved WPs, tasks and technical areas. The following pictures show parts of the dashboards that group JIRA issues per WP6 tasks and technical areas.

Filter Results: EOSC-hub T6.1			
T	Key	Summary	P
+	EOSCWP10-82	B2FIND service for LOFAR/RACC	↓
+	EOSCWP10-45	Support for PIDs by EGI DataHub/onedata and support for PID profiles	↑
+	EOSCWP10-44	Storage requests for WP8.2 Fusion CC	↑
+	EOSCWP10-32	WeNMR RQ4: Onedata should be integrated within West-Life Virtual Folder	↑
+	EOSCWP10-31	WeNMR RQ3: B2DROP should be integrated within West-Life Virtual Folder	↑
+	EOSCWP10-27	OPENCoastS need to expose historical catalogue as Open Data	↑
+	EOSCWP10-25	Search catalogue of high-quality/premium forecasts for OPENCoastS TS	↑
+	EOSCWP10-24	OPENCoastS need to store high-quality/premium forecast archives in permanent storage	↑
+	EOSCWP10-5	Catalogue of OGC WS endpoints	↑

Filter Results: EOSC-hub T6.2			
T	Key	Summary	P
+	EOSCWP10-86	Raman spectroscopy: provide jupyter notebooks environment	↓
+	EOSCWP10-85	CWL workflow support for LOFAR/RACC	↑
+	EOSCWP10-84	Singularity container support for LOFAR/RACC	↑
+	EOSCWP10-83	CVMFS service for LOFAR/RACC	↑
+	EOSCWP10-73	Support for Opportunistic Access to FedCloud nodes	↑
+	EOSCWP10-66	iRODS instance accessible from the Jupyter	↓
+	EOSCWP10-65	Customisable and permanent kernels in Jupyter (EGI Notebook)	↓
+	EOSCWP10-64	Data available in notebooks	↓
+	EOSCWP10-29	WeNMR RQ1: DIRAC4EGI usage should be integrated into the portals	↑
+	EOSCWP10-23	Support for Docker container execution in OPENCoastS service	↑

Figure 5. Dashboard grouping JIRA issues per WP6 tasks. This picture shows the part of the dashboard related to the T6.1 *Discovery and* access and T6.2 *Cloud Compute*.

Filter Results: TCOM Federation tools			
T	Key	Summary	P
+	EOSCWP10-93	Different find results in EOSC Portal and Marketplace	↓
+	EOSCWP10-92	Master ticket and subtickets in JIRA	↓
+	EOSCWP10-91	Transfer order attributes to JSON file	↓
+	EOSCWP10-90	Store information about country of the customers in SO JIRA issues	↑
+	EOSCWP10-89	Status values of services in projects	↑
+	EOSCWP10-79	Issue with 'Terms and conditions' when adding service to a project	↑
+	EOSCWP10-75	Extraction of JIRA adapter component and implementation of internal MP API	↑
+	EOSCWP10-74	Investigate schema.org as interface to pull service information from thirdy part catalogues	↑
+	EOSCWP10-72	Identy entities used in the MP through Permanent Identifiers	↓
+	EOSCWP10-71	EGI perstistent userID in JIRA	↑
1–10 of 36			**1** 2 3 4 ▶

Filter Results: TCOM Cloud Compute			
T	Key	Summary	P
+	EOSCWP10-86	Raman spectroscopy: provide jupyter notebooks environment	↓
+	EOSCWP10-85	CWL workflow support for LOFAR/RACC	↑
+	EOSCWP10-84	Singularity container support for LOFAR/RACC	↑
+	EOSCWP10-83	CVMFS service for LOFAR/RACC	↑
+	EOSCWP10-68	ECAS/IM Integration	↑
+	EOSCWP10-66	iRODS instance accessible from the Jupyter	↓
+	EOSCWP10-65	Customisable and permanent kernels in Jupyter (EGI Notebook)	↓
+	EOSCWP10-64	Data available in notebooks	↓
+	EOSCWP10-23	Support for Docker container execution in OPENCoastS service	↑
+	EOSCWP10-22	EOSC-hub to provide Kubernetes 'as a service'	↑
1–10 of 13			**1** 2 ▶

Figure 6. Dashboard grouping JIRA issues per technical areas. This picture shows the part of the dashboard related to the *Federation tools* and *Cloud Compute* technical areas.

4.3 Handle technical requirements

Technical support starts once the identified requirements have been registered in the WP10 queue as JIRA issues. At that point in time, WP10 support teams supervise and monitor the overall operation throughout the issue life cycle. This may imply further discussion and/or negotiation with the service developers, especially if multiple services are involved. From WP10 point of view, supervision of open issues covers the following tasks:

- Assign issues to a suitable technical contact that could support the case for each service the customer is going to test/use:
  - A suitable technical expert from T10.3.
  - Leader of the technical team that is maintaining and developing the service in case the requirement triggers technical development or integration activities or there is not enough expertise within T10.3.
- Act as an interface between customers and technical teams:
  - Timely reporting to user communities through the JIRA issues, online meetings, etc.
  - Setup testbed to allow customers to try and validate services.
  - Digest the technical feedback (documentation, etc.) --received by technical groups-- to extract the relevant parts that need to be understood by the supported communities.
- Report on the status of the technical activities in T10.3 meeting:
  - Report about achievements.
  - Describe open issues and showstoppers.

5 Procedure to provide technical support

A procedure to provide technical support (SOCRM-04) has been defined as part of the Service Order and Customer Relationship Management (SOCRM) process of the EOSC SMS. Aim of this procedure is creating a well-defined communication channel between the technical support team and first line of support working with requests from user communities received through the EOSC Portal, identified by the Stakeholder Engagement activity (T3.2) or by other channels, implementing the interfaces described in Section 2. Furthermore, the procedure defines formal steps to support use cases, implementing the concepts described in Section 4.3 , and provides standard materials (e.g. e-mail templates) to make homogeneous the communication with the different customers, simplifying the work of the technical supporters (the *technical support shifters*) that have clear guidelines to follow.

The SOCRM-04 procedure is described in the Confluence area of the EOSC SMS[7] and is summarised below:

1. When a new request for technical support arrives, a technical support shifter is identified in T10.3 by the task leader[8].
2. The shifter creates a new blank entry in the Community requirements DB.
3. The shifter contacts the customer by e-mail and invite him/her to a teleconference attaching materials/documentation related to services the customer is interested in.
4. The shifter defines an agenda and run the teleconference. The meeting is structured as follows:
    a. Customer present the technical case.
    b. Open discussion to identify the services that best fit with the customer's requirements.
    c. If requested by the customer: present certain EOSC-hub services.
    d. Agree on the next steps before closing the meeting.
5. After the teleconference, the shifter will:
    a. Complete/finalise the meeting notes.
    b. Identify the next steps.
    c. Fill in the customer database entry with the information collected during the meeting.
    d. Create technical requirements in the WP10 JIRA queue if needed.
    e. Identify a technical person that could support the case for each service the customer is going to test/use:
        i. The same shifter if with an adequate knowledge of the service.
        ii. A member of the T10.3 team or, if needed, on WP5, WP6 and WP7 teams.
    f. Send a follow-up e-mail where for each service requested by the customer are identified:
        i. A technical expert that support the communities.

---

[7] SOCRM-04 procedure: https://wiki.esc-hub.eu/display/EOSC/SOCRM-04+Provide+technical+support
[8] A rota of technical shifters will be setup when the number of requests will increase.

      ii. The main phases of the technical support activity needed to allow the customer to exploit the service.

      iii. Timeline to have a testbed ready and steps to access.

      iv. Next actions.

6. After that, the shifter will start to monitor the use case. He/she periodically checks the progresses contacting the customer. He can also organise a new teleconference, as is needed, as described in step 3 or involve more technical experts.

7. The shifter will close the use case when:

    a. The user is happy with the services and would like to move to production. In this case the SOCRM process will be notified to take care of the next steps to finalise the order.

    b. The user considered completed the test phase for now and no immediate follow-up is planned. The use case can be considered closed and SOCRM process is notified. In this case, the shifter will invite the customer to submit an order through the EOSC marketplace whether he/she is interested to use the tested services in the future.

6 Use case analysis

The Community Requirements DB already provides detailed information about each of the use cases being analysed by WP10 in this first period. Likewise, JIRA issue tracking system contains the discussions and technical information (knowledge base) gathered throughout the lifetime of each technical requirement. To sum up, the following tables provide the suitable metrics that depict the advancement achieved by each use case in the EOSC-hub integration path.

Thematic services

Table 1. Summary of progress achieved by the Thematic Services in EOSC-hub.

Community	Number of user stories (US)	Number of use cases (UC)	Technical requirements (RQ)[9]			
			Total	Tracked (JIRA)	Gap	Solved
CLARIN	13	7	7	5	3	1
DODAS	8	9	9	4	2	3
ECAS	13	12	7	3	2	2
GEOSS	NA	NA	6	0	3	1
OPENCoastS	11	14	14	11	5	3
WeNMR	3	4	5	5	3	2
EO Pillar	11	6	7	3	6	1
DARIAH	5	1	11	2	8	1

Competence Centers

Table 2. Summary of progress achieved by the Competence Centers in EOSC-hub.

Community	Number of user stories (US)	Number of use cases (UC)	Technical requirements (RQ)[10]			
			Total	Tracked (JIRA)	Gap	Solved

---

[9] The number of technical requirements tracked in JIRA can be different from the total because some requirements could be not fully defined yet and, then, not ready to be handled by the technical support team.
[10] The number of technical requirements tracked in JIRA can be different from the total because some requirements could be not fully defined yet and, then, not ready to be handled by the technical support team.

ELIXIR	2	4	2	2	2	0
Fusion	2	5	3	3	2	0
Marine	7	7	2	2	1	0
EISCAT_3D	3	3	5	NA	5	NA
EPOS-ORFEUS	4	5	5	1	1	0
Radio astronomy	3	14	12	6	4	0
ICOS	NA	NA	NA	NA	NA	NA
Disaster Mitigation	3	3	NA	NA	NA	NA

Business Pilots

During this first period, EOSC-hub business pilots have been directly supported by technical experts involved on WP9 and have not being assessed by the technical support teams in WP10.

External communities

Technical support is not limited to communities participating in the project. WP10 might receive requests to support additional communities interested in the EOSC-hub service catalogue, both for usage and integration purposes. As described in Section 3 *Interfaces with other processes,* new communities' requests are coming through SOCRM activities but they are also proactively reached by the Stakeholder Engagement Programme (from task WP3.2) or even by task WP10.3 itself.

During this period, the following external communities: MaX Centre of Excellence[11] and Raman Spectroscopy Data[12], have been incorporated in the requirement and gap analysis processes. While initial contacts have only being done with the former community, Raman Spectroscopy Data use case is in a more advanced state. The following metrics correspond to this latter community:

---

[11] Max Centre of Excellence community db entry:
https://wiki.eosc-hub.eu/display/EOSC/MaX+Centre+of+Excellence
[12] Raman Spectroscopy Data community db entry:
https://wiki.eosc-hub.eu/display/EOSC/Raman+spectroscopy+data+-+CNRS+Saint-Gobain

Table 3. Technical support metrics for Raman Spectroscopy Data use case.

Number of user stories (US)	Number of use cases (UC)	Technical requirements (RQ)			
		Total	Tracked (JIRA)	Gap	Solved
1	4	3	2	0	0

7 Sample use cases

This section describes, in detail, the technical support activities for three exemplars use cases, DODAS and OPENCoastS Thematic Services and the ELIXIR Competence Center, giving information about identified user stories, use cases and technical requirements with the related status.

7.1 Dynamic On Demand Analysis Service (DODAS) Thematic Service

The Compact Muon Solenoid (CMS) is one of the two general-purpose experiments at the Large Hadron Collider (LHC) at CERN in Geneva. CMS relies on the distributed computing capacities of the Worldwide LHC Computing Grid in order to process and analyse the collision data taken during LHC live time. Solutions having the potential to provide additional computing capacity to the LHC experiments, and hence to CMS, are of extreme interest to address the future data and computing challenges of the next generation High Luminosity LHC.

DODAS provides the end-user with an automated system that simplifies the process of provisioning, creating, managing and accessing a pool of heterogeneous (possibly opportunistic) computing resources. At the state of the art, DODAS provides multi-virtual platforms support for a wide plethora of cloud systems such as: the EGI Federated Cloud, OpenStack, OpenNebula, Amazon AWS and Microsoft Azure.

Technical support activity

As described in Section 3, these technical requirements result from a former identification of the user stories and, subsequently, use cases. Detailed data about user stories and use cases collected for DODAS service are reported in the following tables, while Table 4 showcases the links between these and the resultant technical requirements, gathered by the T10.3 support team in the initial round of contacts with the DODAS service owners.

Table 4. User stories for DODAS Thematic Service.

Identifier	Description
US1	As a DODAS user I want a service that simplifies the process of provisioning, creating, managing and accessing a pool of heterogeneous computing resources, including private and public clouds.
US2	As a DODAS administrator, I want to be able to monitor DODAS services (DODAS Core services)
US3	As a DODAS administrator I want to be able to monitor processes and services running on the DODAS clusters (check status resources consumption etc.)
US4	As a DODAS end user (e.g. physicist) I want to transparently access my remote data
US5	As a DODAS end user (e.g. physicist) I want to temporary cache input and output data from DODAS cluster

US6	As community adopting DODAS, I want to be able to globally share libraries and software (e.g. runtime environment) more in general.
US7	As user of DODAS I want to be able to move my sandboxes (input/output) through Dropbox like solution
US8	As a DODAS user, I want to be able to access all my portals using the same credentials, including authentication through the EGI CheckIn when desired.

Table 5. Use cases for DODAS Thematic Service.

Identifier	Description
UC1	The DODAS admin requests the automated deployment of a DODAS cluster
UC2	The DODAS admin checks the status of the DODAS core services
UC3	DODAS users access remote stored data from the DODAS cluster
UC4	User jobs running on DODAS clusters store temporary data on local cluster
UC5	User jobs running on DODAS clusters access software from specific file system path
UC6	User jobs require to move input and output sandbox without the needs of moving data through storages (manually)
UC7	DODAS admin and users authenticate against a single AAI system
UC8	Resource providers authenticate and authorize DODAS requests with INDIGO-IAM tokens
UC9	Accounting data are gathered from the resource providers

Relationship between user stories, use cases and technical requirements are shown in Table 5. Once the requirement is well defined, the support team identifies the services from the EOSC-Hub catalogue that could satisfy it and creates a JIRA issue. In some cases, either the functionality of the service is not yet present, so a technical gap is signalled. As previously discussed, an identified gap may not be solved by the designated service, thus a further analysis is required.

Table 6. Relationship between user stories, use cases and technical requirements for DODAS Thematic Service, according to their IDs. Note that UC9/RQ9 was added as a general requirement for all the EOSC-hub services.

User stories (US)	Use cases (UC)	Technical requirements
US1	UC1	RQ1

US2	UC2	RQ2
US3		
US4	UC3	RQ3
US5	UC4	RQ4
US6	UC5	RQ5
US7	UC6	RQ6
US8	UC7	RQ7
	UC8	RQ8
	UC9	RQ9

Table 7. List of technical requirements for DODAS Thematic Service.

ID	Description	EOSC-hub service	Gap	JIRA ticket	Status[13]
RQ1	Automated deployment of the DODAS cluster on top of heterogeneous cloud environments through TOSCA orchestration	INDIGO PaaS	No	EOSCWP10-37	done
RQ2	Monitoring information gathering	ARGO	Yes	EOSCWP10-34	on hold
RQ3	User data remote access	OneData XRootD	No	N.A.[14]	on hold
RQ4	Store temporary data locally	OneData XRootD	No	N.A.	on hold
RQ5	Usage of specific global file system for software distribution	CVMFS	No	N.A.	on hold
RQ6	Job input and output sandbox data movement	OneData	No	N.A.	on hold

[13] Requirements can be 'on hold' for different reasons, e.g. the implementation is on-going or the community is not ready yet to test the new features.

[14] A JIRA issue will be created when the requirement will be better defined.

RQ7	Provide authN/authZ for DODAS users	INDIGO-IAM	No	N.A.	done
RQ8	Resource providers integration with INDIGO-IAM	ESACO AAI EOSC-hub	Yes	EOSCWP10-36	todo
RQ9	Accounting data gathering	APEL	Yes	EOSCWP10-35	in progress

7.2 ELIXIR Competence Center

The ELIXIR Competence Center will contribute to establish an ELIXIR Compute Platform (ECP) which will allow ELIXIR cloud and data providers to share cloud compute and storage capacity to replicate and share reference data sets with each other and with their users. The platform aims to enable researchers to combine technical components of the ELIXIR Compute Platform services into a seamless ecosystem, thereby creating a science ready, standardized interface to the key resources and technological capabilities that are available for Life Sciences community. The ECP aims to leverage the EOSC Service Catalogue to enable two related yet distinct activities for ELIXIR.

Technical support activity

As outcome of the technical support activity performed by the T10.3 in collaboration with the competence center, two user stories and four use cases were identified. They are detailed in the following tables.

Table 8. User stories for the ELIXIR Competence Center.

Identifier	Description
US1	ELIXIR wants to establish a federation of cloud sites, each providing storage and compute capacity for researchers. The federated clouds should be connected to a data replication service (Reference Data Set Distribution Service with the ELIXIR terminology - RDSDS) that enables ELIXIR to stage 'ELIXIR Core Data Resources' to the cloud sites on-demand. As a result, the cloud sites become data hosting nodes which are equipped with CPUs/GPUs and are suited for large-scale data analysis and analytics. Centrally provided and curated data sets can ensure high-quality research in any of the partner states/regions. Researchers can go to their 'local' ELIXIR cloud provider, choose an already pre-staged ELIXIR data set or request the staging of an ELIXIR data set, choose an application of their choice (from a VM catalogue or container catalogue), maybe upload some additional data and then perform data analysis/analytics. Different conditions of access may apply at the different cloud sites, but it is expected that the cloud compute resources would be free at the point of use for

	national/local researchers, while pay-for-use or other special conditions apply for foreigners. The replication of community assets to national cloud providers maximizes the utilization of national funding and lowers the total cost of access for researchers. The services in the setup should recognize users via their ELIXIR identity, therefore ELIXIR AAI (Life science AAI) should be integrated with the RDSDS as well as with the national clouds.
US2	The cloud federation can be also equipped with a 'container replication and orchestration service' that enables application providers to deploy containerized community/reference applications to any of the federated cloud sites, and users to instantiate and use the applications on those sites. Having a centrally managed, or self-managed, container orchestration service will allow users, who do not have access to cloud resources of their own, to deploy containerized workflows co-located with existing data sets in cloud locations. The services in the setup should recognise users via their ELIXIR identity, therefore ELIXIR AAI (Life Sciences AAI) should be integrated with the RDSDS as well as with the national clouds.

Table 9. Use cases for ELIXIR Competence Center.

Identifier	Description

UC1	Joining the cloud federation with a cloud site (cloud provider perspective):
	
	   1. AAI integration between ELIXIR and EOSC.
	   2. Connect ELIXIR cloud compute or data storage location with EOSC.
	   3. Policy compliance with policies (ELIXIR, EGI cloud, and EOSC-hub policies).

UC2	Making reference/core data sets available for replication to the federated cloud providers (data provider perspective):
	
	   1. Data provider publishes reference data set to RDSDS using ELIXIR AAI.
	   2. Based on a defined data set replication policy RDSDS triggers one or more data transfer/synchronisation process via a central EOSC FTS service.
	   3. EOSC FTS manages the transfer process between source and location and notifies RDSDS on completion.
	   4. RDSDS notifies Data Provider that the data has been synchronised and/or any errors.

UC3	Requesting the replication of a reference/core data set to my local cloud (researcher perspective): 1. User searches and finds for a data set with the RDSDS catalogue. 2. User initiates a transfer of the data set to their local cloud resource using the EOSC central FTS. 3. EOSC FTS notifies the completion of the data transfer and/or errors.
UC4	Making virtualised, reference/core applications available for replication and orchestration on the federated cloud providers (data provider perspective): 1. User searches the EOSC Service Catalogue to identify a container orchestration service. 2. The user uses their ELIXIR credentials to instantiate a container orchestration service. 3. User deploys a containerised application or workload to the container orchestration service.

Two technical requirements were already identified from the use cases. They are described in the following tables, the first showing the relationship between the requirements, the use cases and the user stories, the latter with a short description of the requirements and their current state.

Table 10. Relationship between user stories, use cases and technical requirements for ELIXIR Competence Centres, according to their IDs.

User stories (US)	Use cases (UC)	Technical requirements
US1	UC1, UC2, UCS3	RQ1
US2	UC4	RQ2

Table 11. List of technical requirements for the ELIXIR Competence Center.

ID	Description	EOSC-hub service	Gap	JIRA ticket	Status
RQ1	EOSC-hub to provide an FTS data transfer service	EGI FTS	Yes	EOSCWP10-21	In progress
RQ2	EOSC-hub to provide Kubernetes as a service	EGI Cloud Container Compute	Yes	EOSCWP10-34	In progress

7.3 OPENCoastS Thematic Service

The OPENCoastS (On-demand Operational Coastal Circulation Forecast Service) service builds on-demand circulation forecast systems for user-selected sections of the North Atlantic coast and maintain them running operationally for the timeframe defined by the user. This daily service generates forecasts of water levels, 2D velocities and wave parameters over the spatial region of interest for periods of 72 hours, based on numerical simulations of all relevant physical processes. Currently, the service is deployed at a single computing site (NCG-INGRID-PT). Enlarging the service to international access requires additional computing capacity for scaling and resiliency (service continuity). Furthermore, it requires access to distributed data. Consequently, integration with more sophisticated distributed data management services is necessary.

Technical support activity

As outcome of the technical support activity performed by the T10.3 in collaboration with the thematic service, eleven user stories and fourteen use cases were identified. They are detailed in the following tables.

Table 12. User stories for OPENCoastS Thematic Service.

Identifier	Description
US1	As a service user, I want to log into the OPENCoastS web portal through my home institution credentials.
US2	As a service user, I want to download data from the simulation of the model.
US3	As a service owner, I want to preserve high-quality/premium forecasts to be offered for re-analysis.
US4	As a service owner, I want to publish the catalogue of high-quality/premium forecasts to be offered to the service users.
US5	As a service user, I want to be able to search the forecast catalogue based on a given set of characteristics.
US6	As a service owner, I want to promote the simulation data as Open Data.
US7	As a service owner, I estimate 40TB per month of storage consumption.
US8	As a service user, I want to run my simulations up to 72 hours.
US9	As a service owner, I want to run jobs in HTC, Grid and Cloud environments.
US10	As a service owner, I want to be able to deploy the service in an automated way using the Cloud.
US11	As a service owner, I want to obtain monitoring and accounting information of my running service.

Table 13. Use cases for OPENCoastS Thematic Service.

Identifier	Description
UC1	User logs in the OPENCoastS service using eduGAIN.
UC2	OPENCoastS service manages the authorization/attribute provision.
UC3	OPENCoastS service obtains a x509 certificate for Grid submission
UC4	User constructs a broadcast simulation.
UC5	User submits jobs with the broadcast simulation.
UC6	OPENCoastS service may support docker container execution.
UC7	User obtains outputs from the simulation for the next (at most) 72h.
UC8	OPENCoastS service automatically performs quality checks to identify high-quality/premium forecasts.
UC9	OPENCoastS service stores high-quality/premium forecasts.
UC10	User searches the catalogue of high-quality/premium forecasts for re-analysis.
UC11	OPENCoastS service is deployed automatically.
UC12	OPENCoastS service is deployed in the Cloud as a long-running service.
UC13	OPENCoastS service is monitored.
UC14	User resource consumption (compute, data) is tracked and accessible.

Fourteen technical requirements were already identified from the use cases. They are described in the following tables, the first showing the relationship between the requirements, the use cases and the user stories, the latter with a short description of the requirements and their current state.

Table 14. Relationship between user stories, use cases and technical requirements for OPENCoastS Thematic Service, according to their IDs.

User stories (US)	Use cases (UC)	Technical requirements
US1	UC1	RQ1
	UC2	RQ2
	UC3	RQ4

US2	UC7	RQ7
US3	UC9	RQ8
US4	UC9	RQ8
US5	UC10	RQ9, RQ10
US6	UC8	N.A.
	UC9	RQ8
US7	UC9	RQ8
	UC14	RQ14
US8	UC4	RQ3
	UC5	RQ5
	UC7	RQ7
US9	UC6	RQ6
	UC12	RQ12
US10	UC11	RQ11
US11	UC13	RQ13
	UC14	RQ14

Table 15. List of technical requirements for OPENCoastS Thematic Service.

ID	Description	EOSC-hub service	Gap	JIRA ticket	Status
RQ1	eduGAIN support for EOSC-hub AAI	EGI Check-in	No	EOSCWP10-19	TODO
RQ2	Attribute provision for OPENCoastS	OPENCoastS	No	EOSCWP10-19	TODO

RQ3	Forecast simulation composition	OPENCoastS	No	N.A.	
RQ4	OpenID Connect token translation to x509 certificate for Grid submission	WaTTS/MasterPortal	No	EOSCWP10-26	TODO
RQ5	Multi-site job submission	DIRAC4EGI	No	EOSCWP10-18	TODO
RQ6	User-space Docker container execution	udocker	No	EOSCWP10-23	DONE
RQ7	Job output management	DIRAC4EGI	No	EOSCWP10-18	TODO
RQ8	OPENCoastS need to store high-quality/premium forecast archives in permanent storage	B2SAFE	No	EOSCWP10-24	TODO
RQ9	OPENCoastS need to handle metadata for user access to historical data	EGI DataHub B2FIND B2NOTE	No	EOSCWP10-27	TODO
RQ10	OPENCoastS need to expose historial catalogue as Open Data	B2HANDLE B2DROP B2SHARE	No	EOSCWP10-27	TODO
RQ11	Automated deployment using Ansible of OPENCoastS service	N.A.	No	N.A.	
RQ12	OPENCoastS service deployment in the Cloud	PaaS Orchestrator	No	N.A.	
RQ13	Monitoring as a service	ARGO	No	EOSCWP10-43	ON HOLD
RQ14	Tracking compute and storage consumption	Accounting	No	EOSCWP10-42	TODO