D26.1 – Innovation Ecosystem platform specifications and architecture – M9 issue

Document Owner:	Martino Maggio (ENG), Davide Storelli (ENG), Calogero Crapanzano (ENG)
Contributors:	H. Boyé (Hardis), Miguel Loitxate Cid (Tecnalia), Iñigo Martinez Criado (Tecnalia)
Dissemination:	Public
Contributing to:	WP 26
Date:	31 07 2012
Revision:	Version 1.0

VERSION HISTORY

VERSION	DATE	NOTES AND COMMENTS
0.1	15.06.2012	INITIAL TABLE OF CONTENTS
0.2	22.06.2012	INITIAL CONTRIBUTION BY ENGINEERING
0.3	29.06.2012	CONTRIBUTIONS FROM TECNALIA AND HARDIS
0.4	06.07.2012	REDEFINITION OF THE TABLE OF CONTENTS
0.5	20.07.2012	INTEGRATION OF COMMENTS AND CONTRIBUTIONS AFTER THE MILANO MEETING
0.6	30.07.2012	INTEGRATION OF REVISIONS FROM TECNALIA AND HARDIS AND GENERAL REVISION BY ENGINEERING
1.0	31.07.2012	VERSION 1.0 FOR PEER REVIEW

DELIVERABLE PEER REVIEW SUMMARY

ID	**Comments**	**Addressed (✓) Answered (A)**
1	In general the document requires some revision in order to complete some of the missing contents, that would establish necessary textual interconnections for a smooth and continuous reading and understanding (please consider the following remarks).	
2	The text has some few typos, e.g., Page 7: "Particular" should be "particular"	Fixed
3	To be coherent with the contents of the deliverable, the title should be: Innovation Ecosystem Platform specifications and functional architecture (however, if this is the contractual title, this should be clarified in the introduction, otherwise the reader would expect to get the complete IEP architecture, and not just the functional one.	Added clarification in section 1.1
4	Section 1.1: The objectives address the use cases. These should be presented in a "AS-IS" and "TO-BE" scenarios, indicating the relevance and advantages of the proposed specifications and functional architecture from the move from the AS-IS to the TO-BE.	Addressed in section 2.1
5	Section 1.4: Relevant projects. This section is incomplete. There are many other projects (EC and others) that are of	Changed title in "Reference project", and added an introduction to the

	relevance for this deliverable and should be listed here, e.g., iSurf project (providing the ISU).	section. The scope of the section was not to have state of art or a survey of possible interesting projects, but only to have a brief introduction about COIN and Fiware, the two projects referenced in the DoW and strictly connected with D26.1
6	Figure 2. The message depicted by figure 2 is not clear. As it is, it seems to be incomplete, and irrelevant to complement the text. The meaning of the arrows, and of AS-IS and TO-BE is not evident. Legend is missing. Figure 2 needs to be enriched.	Updated figure and description
7	Page 11/12. The concepts "organizational-human, manufacturing-physical, information-IT" need to be defined.	Updated description
8	Page 12, Paragraph 2. "This deliverable, while taking in to account the WHOLE Service Life Cycle and ALL the TECHNOLOGICAL platforms and modules that support it…". This sentence is not coherent with the contents of the deliverable, as it does not cover ALL the aspects stated in the sentence. The phrase needs to be revised accordingly. Also, the FUNCTIONAL architecture is the focus of the document, and not the TECHNOLOGICAL platform.	Updated description
9	Sections 2.1.1-2.1.4 are described in very high level. It requires further elaboration and detailed explanation for a complete understanding of the operations, and how they relate and they will be integrated in the IEP. E.g., "… MSE: such business entity has to be formally designed and all the business processes have to be properly defined in the servitization process" does not explain afterwards how it will be developed and integrated in the IEP.	Updated descriptions
10	Section 2.1.2. "…(to be defined in the following paragraph)". The definition does not appear in the text.	Updated description
11	It should be included a rational justification for the listed processes of the MSE Governance.	The rational about these process are related to task T25.1 and further information will be available in the relative deliverable.
12	Figure 3. "MSE Governance processes" seems to be more appropriate than "Ecosystem processes". The legend is missing (what's the meaning of the arrow in the background of the text ? The separation of Support processes and Primary activities is not clear and not explained properly in the text.	"Ecosystem processes" could fit better than "MSE Governance processes" because not all the processes in the image are governance processes e.g. "Innovation processes". Figure description updated.

13	Section 2.1.3. "The innovation process runs in parallel to operational and ecosystem-governance processes and…". The statement is not well justified. Why in parallel ? Why the operational and ecosystem-governance processes ?. This requires further elaboration for a clear understanding.	Added clarification in section 2.1.3
14	Figure 4. The text that addresses this figure, does not describe it properly. Also, the figure seems to be incomplete, without a clear and evident message from it intends to depict.	Updated figure (now picture 5)
15	Section 2.2. This section presents the most important use cases…". Why these are the most important use cases ? Nevertheless, the following sections provide description of the use cases in a very general way, without sufficient explanation why these use cases where selected. Also, the use cases themselves are not completely described, starting with an abstract high-level view, and then with the details in tabular form without appropriate description for a smooth understanding of the use case.	Added clarification in section 2.2 The description of the use cases is generic because the requirements collected in this first part of MSEE are generic and incomplete. The idea is to provide a general description of the UC (and the functionalities) of IEP in this deliverable and a more complete e detailed one in the next version D26.2 (due for M18).
16	Section 3. The IEP is depicted in Figure 8, but the rational for it is not presented, just focusing in the description of its components. Why this architecture and why these components needs to be justified and discussed.	Added clarifications in the section
17	Section 3.1, claims that the front-end interface to final users is one of the most important aims of the EIP. This strong statement is not justified, and it hard to understand why this is the case, whilst its functionalities would be core for the EIP operationally. Also, the list of modules proposed is not justified.	Added clarification in the section
18	Figure 12 is not enough explained in the text.	Description improved (now picture 13)
19	Section 3.4. TSM, TIM, BSM, SLMtoolbox are not described nor explained.	Added reference
20	Section 4. Should be "…technological products that could be USED to cover part…"	Misprint fixed
21	Table 39. It should be explained in detail what is expected to be developed by MSEE (and in which WP), to complement those "cells" that are not in green.	Table commented

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

ACRONYMS

API:	Application Programming Interface
BPM:	Business Process Modelling
IEP:	Innovation Ecosystem Platform
KPI:	Key Performance Indicators
MSE:	Manufacturing Service Ecosystem
SLM:	Service Lifecycle Management
SPx:	MSEE Sub Project x
GUI:	Graphical User Interface
USDL:	Unified Service Description Language
VME:	Virtual Manufacturing Enterprise
WM:	Washing Machine
WPx:	MSEE Work Package x

Executive Summary

The objective of WP26 is to design and develop the MSEE Innovation Ecosystem Platform (IEP) and a set of ecosystem-oriented services for collaboration and tangible/intangible assets management, in order to enable the execution of the business processes that characterize a Manufacturing Service Ecosystem (MSE). Those processes have been categorized as MSE Governance, for the management of the ecosystem and of its members, MSE Innovation, for fostering and stimulating open innovation actions among the ecosystem actors, and Virtual Manufacturing Enterprise (VME) Operations, for allowing VMEs in the delivering of the new value proposition (defined in the servitization process, see [1]) to the customer through collaborative processes.

This deliverable provides the first version of the IEP specification and functional architecture (the second version will be released at M18) and focuses on the three mentioned process categories in order to collect their main requirements and define the functional components that will allow their fulfilment.

In particular MSE Innovation processes have been mainly studied in WP21, which defines a formal innovation framework that should be adopted in Manufacturing Service Ecosystems with the support of tools and platforms of the MSEE IT System. The framework defines the innovation process as composed of the following phases: Idea Generation, Concept Development, Prototyping, Manufacturing, Marketing & Sales / Launch. The IEP has been designed to support the initial phases of Idea Generation and Concept Development (which constitute the Ideation Process) in order to feed the servitization process with innovation opportunities to be transformed into actual value.

VME Operation processes, which are mainly studied in WP52 through the MSEE use cases, and designed in the servitization process through tools developed in SP1 and SP4, are supported by the IEP through a set of BPM tools that allow the deployment, execution, monitoring and analysis of business processes that include both human-to-human collaboration and product-to-human/human-to-product interaction.

MSE Governance processes, currently under definition in WP25, enable the management of every aspect of an ecosystem. The IEP supports the MSE internal administrative processes and the monitoring of specific KPIs through a complete set of BPM tools, allows the management of value-added capabilities of the MSE members through the description and publication of tangible/intangible assets as Services on a dedicated marketplace (leveraging results of WP22 and WP23), and it includes the means to guide MSE members in the improvement of their servitization level and of their level of involvement in the ecosystem (reflecting the results of WP24).

The first part of the deliverable defines the role of the IEP in the overall MSEE IT System and outlines the use cases that the Platform has to support to enable its core processes. The second part presents the design of the functional architecture of the IEP, providing specific details of the single components and their dependencies. The final part provides a first evaluation of software platforms and technological components already available as Open Source and that can be leveraged to develop the specific components of the functional architecture.

1 Introduction

1.1 Objective of the deliverable

According to the Description of Work (DoW), the main objectives of WP26 are:

- To design the specific architecture of the Innovation Ecosystem collaborative platform
- To develop ecosystem-oriented distributed collaboration services
- To develop/integrate specific services for Tangible and Intangible Manufacturing Assets Mgmt

This deliverable is the first result of Task T2.6.1: *Innovation Ecosystem Platform specification and Design* (the second version of this deliverable is scheduled on M18 in order to take into account end-users evaluations and assessment) and its objective is to **provide the first version of the IEP specification and functional architecture**. The deliverable follows a structured approach that starts by clarifying the role of the IEP in the overall MSEE IT System identifying the type of processes and actors that it has to support. After that, the main use cases related to those processes are described. On this basis the functional architecture of the IEP and its components are defined in order to cover all the previously introduced use cases. Finally the deliverable provides a first assessment of possible concrete technologies and software that could be used to implement the different functional components of the IEP.

Therefore, the following objectives have been identified for the deliverable:

- to define the use cases of the Innovation Ecosystem Platform;
- to design the functional architecture of the IEP based on use cases descriptions;
- to evaluate a set of software and technologies to be used to implement the functional architecture.

1.2 Structure of the deliverable

The structure of the deliverable is organized into three major blocks, according to the objectives described in section 1.1:

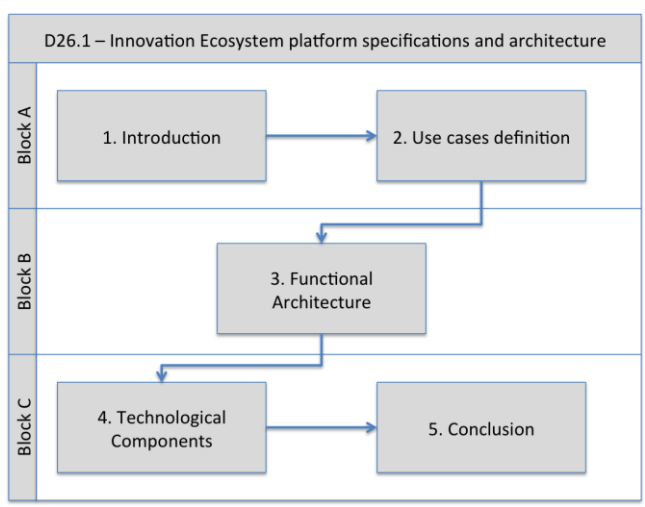

Figure 1 - Structure of the deliverable

Block A defines the role of the IEP in the overall MSEE IT System and outlines the use cases that the Platform has to support to enable its core processes.

Block B presents the design of the functional architecture of the IEP, providing specific details of the single components and on their dependencies.

Block C provides a first evaluation of software platforms and components already available as Open Source and that can be leveraged to develop the specific components of the functional architecture.

1.3 Relations with other work packages

Work Package 26 has the role of developing an integrated software platform that allows ecosystem members to run collaboration and innovation processes according to the conceptual and concrete outcomes of the rest of Sub Project 2. The following table shows the relations between the WP26 and the other work packages in SP2:

WP name	Relation
WP21 - Reference Framework for Innovation Ecosystems	WP21 provides the conceptual basis for the implementation of a tool for the management of innovation processes in the ecosystem.
WP22 – Intangible Assets Management	WP22 provides conceptual models and basic prototypes to model intangible assets aaS (as a Service). Those assets will be managed through dedicated tools in the IEP.
WP23 – Tangible Assets Management	WP23 provides conceptual models and basic prototypes to model tangible assets aaS (as a Service). Those assets will be managed through dedicated tools in the IEP.
WP24 - Maturity Models and Change Mgmt	WP24 provides models and processes to (i) assess the maturity level of an enterprise with respect to the servitization-level and the level of involvement of the ecosystem; (ii) define a roadmap that guides the enterprise in the change process toward the desired to be. Such models and processes have to be integrated in the IEP in order to facilitate their enactment in real-world scenarios.
WP25 - Service based Innovation Ecosystems	WP25 will provide processes and metrics to be implemented for ecosystem governance in the IEP.

Table 1 - Relations between WP26 and other SP2 work packages

1.4 Reference projects

Beyond dependencies and relations with internal MSEE work packages, the IEP will get inspiration and baseline resources from other projects that represent the reference point with respect to specific subjects, namely the COIN [2] Integrated Project and the FI-WARE [4] Project. On the one hand the IEP will be defined looking back at the results of COIN, taken as a synthesis of the achievements of the Enterprise Interoperability research stream; on the other hand FI-WARE will provide the necessary connection with the vision of the Future Internet and with the related current developments in terms of service models and related supporting technological environments.

1.4.1 COIN

COIN [2] is an integrated project in the European Commission Seventh Framework Programme . In the overall COIN Vision, networked enterprises are organized in supply chains, collaborative networks and business/innovation ecosystems. Their collaboration is often supported by IT Collaboration Platforms (CP), which provide utilities for social collaboration (e.g. wikis, blogs, communities, user profiling), knowledge collaboration (e.g. repositories, search engines, up- down-load facility, classification) and business collaboration (e.g. workflow, workgroup and business process management services)[3]. The philosophy of COIN about Collaboration Platforms is adopted by the MSEE IEP, which will leverage the results of COIN from the conceptual and technological perspectives. The functional architecture of the COIN CP has indeed inspired the one of the MSEE IEP in terms of strategies for integrating distributed resources, applications and services. Moreover collaboration and interoperability services developed in COIN and published on the COIN CP will represent a good foundation for further developments of services for the Manufacturing Service Ecosystems. Enrichment and evolutions of those services will, for example, provide specific means to fully leverage the value of enterprise assets through their publication as services on a dedicated marketplace.

1.4.2 FIWARE

The FI-WARE[4] project is part of the Future Internet Public Private Partnership (FI-PPP) program of the European Commission. Its main result will be the FI Core Platform, which comprises a set of technological “Generic Enablers” that are considered general purpose and common to almost any industry domain. Among the various technical aspects addressed by FI-WARE, MSEE is mostly interested in adopting and getting inspired by the infrastructure to create, publish, manage and consume FI services, addressing all technical and business aspects. In particular the use of USDL and of USDL-related Generic Enablers will allow the description of services and their management across their life cycle. The adoption and adaptation of those Generic Enablers to the necessities of MSEE (with the provision to FI-WARE of relevant feedback and, possibly, requirements) will represent one important result of SP2 since usage areas formally related to FI-WARE (eHealth and Wellbeing, Content, Smart Energy Grids, Utilities and Environment, Transport, Mobility and Logistics) do not include the manufacturing industry and Future Internet Enterprise Systems (see FInES Cluster), which are the focus of MSEE.

2 Use cases definition

2.1 The role of the IEP in the MSEE service system lifecycle

The role of the Innovation Ecosystem Platform (IEP) in MSEE is to enable the execution and the enactment of the business processes that characterize a Manufacturing Service Ecosystem (MSE). The scope of those processes must be defined in the context of the overall Service System Life Cycle (see Figure 2) that MSEE aims at supporting and guiding in the manufacturing industry.

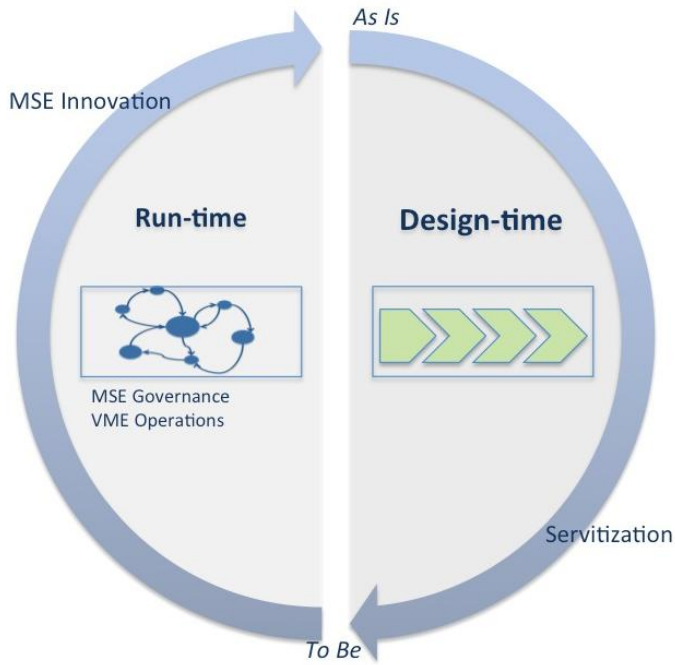

Figure 2 - MSEE Service System Life Cycle

Such a cycle is composed of two main phases:

- The design-time phase, which corresponds to the *servitization* process, on the basis of precise new business opportunities and after the set-up of a Virtual Manufacturing Enterprise (VME)[1] aims at grasping such opportunities, transforming the service system from its As-is situation to the most appropriated To-be in terms of operative processes, value-added services and through the leveraging of key assets of the VME. This process is responsible for the design of the VME's human, physical and IT

[1] a Virtual Manufacturing Enterprise is an autonomous and self-contained enterprise that uses capabilities and key assets of organizations "extracted" from a MSE and of organizations that are not part of that MSE, in order to exploit a specific business opportunity through ad-hoc processes and services that are designed and developed in the servitizaion process.

resources[2], leading to a new business model that switches the value offered from physical products to product-service "bundles".

- The run-time phase , that on the one hand allows the management of an MSE (MSE Governance), fostering and stimulating open innovation actions among the ecosystem actors (MSE Innovation), and on the other hand exploits the new techno-organizational configuration of VMEs (previously designed and developed in the design-time phase) in order to deliver the new value proposition to the customer through operational processes (VME Operations).

Figure 2 shows the two phases as separated and consequential: this can be considered only a logic view because in the real world they are performed at the same time influencing each other. Indeed the design-time phase is triggered by the MSE Innovation processes, which feed the servitization process with new potential business opportunities and new innovative ideas. When the servitization process starts, the Governance, Operative and Innovation processes continue to address their objectives (see following paragraphs) independently from the status of the design-time phase. At the end of the servitization process, the newly defined operative processes are deployed and executed by the owner VME until they are dismissed.
This deliverable, while taking into account the different technological Platforms and modules that support the Service System Life Cycle , has a specific focus on the functional architecture of the IEP, which is the technological platform that will support the run-time phase of the Service System Life Cycle.

The actual value-added of the IEP is given by its integration in the overall conceptual and technological framework that MSEE will provide to support the Service System Life Cycle. Taken alone, indeed, the three kind of processes addressed by the IEP, are already managed in the four MSEE Use Cases, but they all lack in terms of an end-to-end vision and enactment of the Service System Life Cycle. The use cases and functional components described in this deliverable are intended to guide the development of the IEP in order to allow a better integration of the three kind of processes and to make them compliant with the conceptual frameworks and methodologies conceived in the rest of the MSEE project.

The following sections describe more in detail the MSE Governance, MSE Innovation and VME Operations processes in order to highlight their goals and facilitate the reading of the

[2] A human resource can be an operator, a manager or any people performing an activity in the service; Machine type resource are devices and equipments that are needed to provide and deliver services. These components may include: robots, specialized machines and devices for producing, delivering, maintaining services, movers/transport means, as well as any kind of physical facilities used for the creation and consumption of servicee; IT type resources include mainly software used to perform activities in the service to be delivered

formal analysis of the use cases that the IEP should satisfy to support their execution and management (section 2.2).

2.1.1 VME Operations

VME Operations represent the set of business processes (operational processes from now on) that allow a VME to deliver the product-service bundle to the final user. More precisely, operational processes address the provision of value-added services that are bound to a physical product. Such processes typically involve both the VME and the final user (intended as a prosumer: indeed he/she contributes to the value creation), with the possibility to include the product itself as part of the process. The source for actually define the scope and collecting requirements of operational processes is WP52, which currently has issued one reference deliverable (D52.1) [6] that specifies the first version of a complete analysis of requirements and needs from the four MSEE Use Cases.
According to the studies already conducted in WP52 on the MSEE use cases, activities of operational processes span from human-to-human collaboration mediated by an ICT platform, to product-to-human/human-to-product interaction, in which the physical product is an active actor of the process (e.g. by triggering the start of a maintenance task or by feeding decisional processes with real-time information about their status).
Here we take one service from the Indesit use case in order to show how could a supporting operational process look like.

From D52.1, one of the services analysed in the Indesit use case:
Automatic soap/cleaner recharge*. This services package aim to simplify the daily use of the Washing Machine (WM) avoiding the soap charge phase, to facilitate the use of dedicated soap for the specific clothes and to automatically load the soaps and to inform the customer that recharge is needed. A specific feedback alarm can be visualized in remote systems of the users when the soap level is low. This information is useful to correctly manage the scheduled purchase of soap recharge. In addition, users can be informed about marketing actions or special offer with the scope to buy specific detergents for their needs and save time and money. Such a service could also be realized in collaboration with soap-cleaner producers or distributors as project partners with the final aim to take fruitful co-marketing actions.*

The services package described above must be properly designed and developed in the servitization process through dedicated methodologies and IT tools provided by MSEE (i.e by SP1 and SP4). The overall design and implementation of the service and of the service system (the to-be of Figure 2) will include the specification of the operational processes for the service provision.
The following is an example process for this package:

- the WM sends an alert to the mobile device of the user about the need of soap recharge. The alert includes a list of special offers and a list of the soap distributors that are in the geographical area of the user.
- the user chooses three soap distributors and sends them a doodle poll to arrange the delivery date and time;
- two of the three soap distributors respond to the doodle poll;

- the user chooses one of the soap distributors;
- at the agreed date and time the soap distributor delivers the detergent to the user;
- the user provides rating to the service and to the soap distributor.

The IEP has to allow the deployment, execution and monitoring of operational processes through appropriate workflow, BPM and decision support systems. The analysis of the four MSEE use cases leads to the definition of the use cases listed in section 2.2.1 for a system that supports the runtime of operational processes. It is to be noticed that such use cases and the derived functional components of the IEP (see section 3.4) are currently in a first version that aims at supporting a wide and general-purpose range of possible operational processes. This is due to the current lack of precise processes to be supported and specific needs to be satisfied. The second version of this deliverable will however be more detailed and will address more peculiar requirements after a deeper analysis of the four MSEE use cases.

2.1.2 MSE Governance

This section analyses the type of processes that a Manufacturing Service Ecosystem should follow to implement its activities. However, it is important to remark that these processes will be one of the majors' outputs of Task 25.1, consisting in the definition of the conceptual model of a MSE. Thus, as a first approach to the manufacturing service ecosystems processes in the sections will be presented an initial tentative list of these processes. A more complete analysis will be performed in D25.1 delivered after month 12 (September 2012).

It is also important to point out that the primary activities of the ecosystem would be related to innovation (see section 2.1.3 for more details on how innovation processes are to be addressed by the IEP). Meanwhile, the management processes would be considered as support processes of the ecosystem. The following picture is merely illustrative and represents the ecosystem processes: in the lower part of the image are indicated the innovation processes (described in section 2.1.3) that are primary activities performed in the ecosystem, considering that one of the main aims of the ecosystem is to stimulate and produce innovation. On the top are listed the ecosystem governance processes that can be considered support processes because they provide the functionalities needed to manage and control the whole environment: these projects will be described further on.

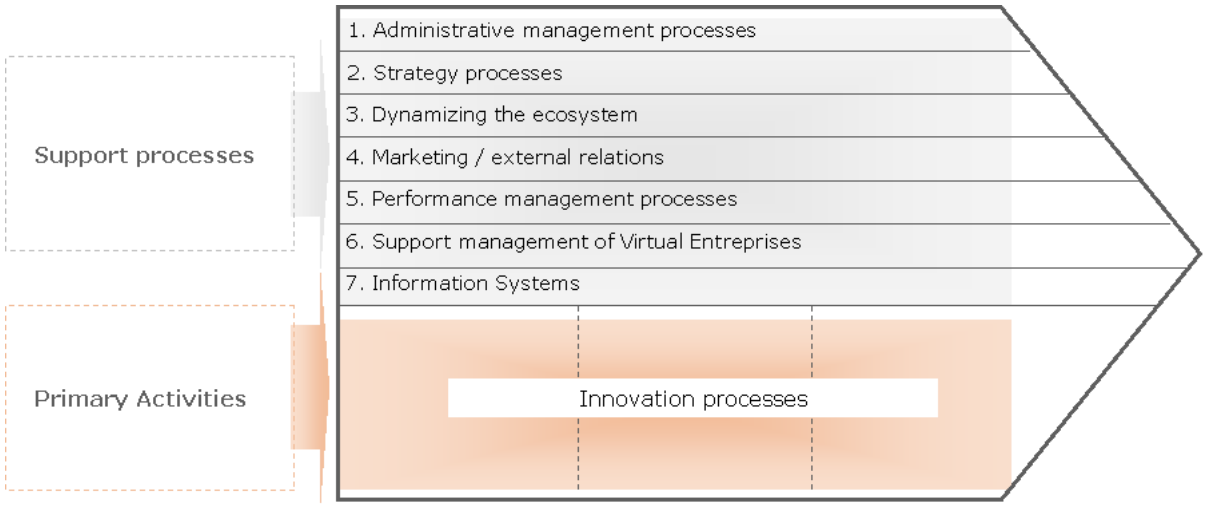

Figure 3 - Ecosystem processes

2.1.2.1 Ecosystem Administrative Management Processes

Administrative process includes the office tasks that are required to keep the ecosystem operating. This type of process might include the following sub-processes:

- **Member management**: some examples of tasks included in this process might be partners/agents' recruitment, leaving of partners, management of entry conditions (fees, etc.).
- **Legal processes**: examples of tasks that might be contained in this sub process are: establishment and satisfaction of collaboration agreements, NDA (confidentiality agreement), contracts review (i.e., with external suppliers), etc.
- **Intellectual property:** this sub process may include the creation, protection and exploitation of the intellectual property.
- **Finance management process**: some examples of tasks included in this sub process are membership fees management (if they exist), subsidies management, evaluation of financial health of members/potential members, accounting, etc.

2.1.2.2 Strategy Processes

It involves the definition and coordination of the ecosystem strategy in accordance with every member's objectives. Examples of activities included in this processes are: market surveillance to update the ecosystem strategy, coordination of members to implement the strategy, monitoring of KPIs, strategy dissemination, etc.

2.1.2.3 Dynamizing the ecosystem

It comprises the activities that are carried out to stimulate ecosystem members to exchange knowledge and create virtual manufacturing enterprises (VMEs). Examples of activities that might be included in this process are internal communication, promotion of networking and trust, implementation of knowledge management systems and activities, etc.

2.1.2.4 Marketing / external relations

It involves the definition and coordination aiming to optimize the relation of the ecosystem with its environment. This process might include other sub processes, as the identification of potential new members for the ecosystem, external communication activities and the institutional relations (or even, lobbying activities).

2.1.2.5 Ecosystem performance management processes

This process includes the controlling tasks that are required to maintain the quality and to improve the performance of the ecosystem. Therefore, it involves tasks as the definition of key performance indicators, assessment and improvement of processes and systems, partners' performance evaluation, etc.

2.1.2.6 Management support to "Virtual Manufacturing Enterprises"

The ecosystem will provide support for the creation, operation, metamorphosis and dissolution of VMEs. Accordingly, this process will consist of administrative process aiming to sustain and facilitate virtual enterprises activities throughout their lives.

2.1.2.7 Information systems processes

This process group includes all the sub processes and tasks related to the maintenance and improvement of the network of hardware and software that the ecosystem members will use to collect, filter, process, create and distribute data.

The general requirements that can be currently extracted for ecosystem-governance processes lead to the definition of the use cases listed in section 2.2.2.

2.1.3 MSE Innovation

The VME operations and MSE governance processes are strictly related: a VME cannot exist without the ecosystem and the related processes: also the operative process aimed to deliver product-service bundles are executed in the ecosystem. For this reason the VME operations and MSE governance process have to be considered two parallel processes groups.
Besides these, a key role it is also played by the innovation process, that includes all the activities aimed to promote new ideas to restart and continuously renovate the MSEE Service System Life Cycle. The innovation process involves different parties of the ecosystem with roles that must be well-defined in advance. Indeed, although innovation frequently stems from individual and informal initiatives/activities, its management has to follow a structured and commonly agreed procedure, in which ideas are discussed, merged, elaborated, evaluated, filtered, through different stages, and in which people are in charge of decisions at specific stages. Deliverable 21.1[7], part of WP21 ("*Reference Framework for Innovation Ecosystems*"), defines a formal innovation framework that should be adopted in Manufacturing Service Ecosystems, with the support of tools and platforms of the MSEE IT System.
The framework defines the innovation process with the following phases:

- **Idea Generation**: during this phase employees and customers submit ideas both responding to a campaign created by a campaign organizer or to an on-going theme for which companies of the ecosystem always require proposals. Ideas can be associated with some existing categories, tags and multimedia files.

- **Concept Development**: ideas created in the Idea Generation Phase are improved with votes, comments, more tags and attachments submitted by others employees and customers. Votes and comments contribute to the calculation of rating of the ideas. Ideas can be interlinked and merged in order to elaborate new concepts. At the end of this phase the best ideas are selected to feed the following phase.

- **Prototyping**: during this phase accepted ideas are transformed into prototypes in order to have an early evaluation of their business potential.

- **Manufacturing**: during this phase accepted prototypes are transformed to final product-service bundles

- **Marketing & Sales / Launch**: the product-service bundle is launched on the market and operational processes are enacted in the ecosystem in order to deliver it to the final users.

From D21.1: "*The considered process starts with the early idea generation (sometimes before a specific innovation project has even started) and result in product launch, marketing and sales. Therefore the following PIs can be regarded as an option to assess the "innovation capability" of an organisation. The performance of the innovation process has to be monitored and controlled to ensure that the activities make best use of the provided input and that the objectives regarding the output are achieved. To enable general conclusions the indicators have to regard the sum of on-going and finished projects.*"

The same deliverable proposes an initial set of PIs to be measured:

- "*Number of ideas submitted by employees in a defined period of time. Or number of ideas/builds collected.*
- *Number of ongoing experiments and ventures.*
- *Number of employees / "Authors" who submitted ideas. Alternatively: Ratio of these employees in relation to the overall number of employees.*
- *Number of ideas reviewed and concluded.*
- *Number of Ideas passed through to concept development*
- *Number of Event Visitors / Contributors*
- *Average time required from idea conceptualization to go-forward decision.*
- *Number of examined customer groups.*

- *Number of ideas collected.*
- *% of suppliers engaged in the collaborative design.*
- *% of ideas from outside.*
- *Number of Voice of Customer specifications."*

The aim of the Innovation Ecosystem Platform is to support the initial phases of Idea Generation and Concept Development (which constitute the Ideation Process) in order to feed the servitization process with innovation opportunities to be transformed into actual value.

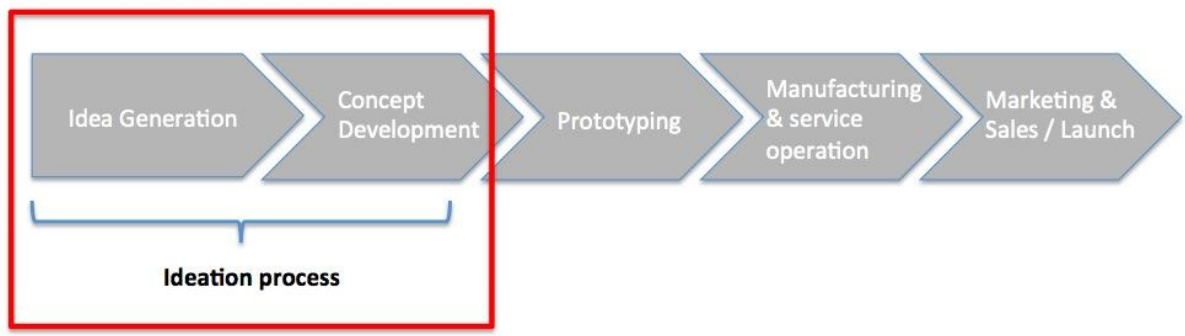

Figure 4 - Phases of innovation process

The ideation process must leverage collaboration tools to foster the generation of new ideas, collect and elaborate them and finally select those ideas that represent a viable opportunity for the ecosystem. In particular,

- ideas should be generated through ecosystem-wide idea contests
- ideas collected should be rated and commented by any members of the ecosystem
- members should discuss improvements to the ideas through collaboration tools
- ideas should be reviewed, ranked and finally selected by appropriate actors in order to feed the servitization process.

The general requirements extracted from D21.1 and from the MSEE scenarios lead to the definition of the use cases listed in section 2.2.3 for a system that supports the enactment of the ideation process.

2.1.4 Actors and roles

The Innovation Ecosystem Platform will provide functionalities for the different members of an ecosystem involved in the Innovation, Governance and Operational processes. This section introduces at a high level the roles that are played by the actors in the different processes. Further refinements (specializations) and details about such roles and their use cases are provided in the next sections. The set of roles should not be considered as definitive, but subject to further refinements in the following phases of the project.

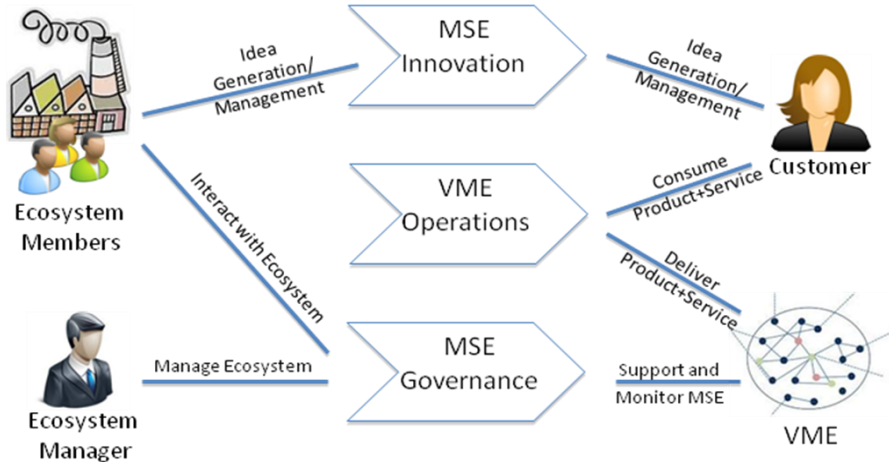

Figure 5 - Roles in the MSEE Innovation Ecosystem Platform

Any actor that is formally part of the ecosystem is an ***Ecosystem Member:*** an Ecosystem Member can be an organization (e.g individual enterprise, individual research center, etc), a person belonging to an organization that is part of the ecosystem, but also a **Virtual Manufacturing Enterprise** that is a composition of the previous ones. According to the specific characteristics and policies of the ecosystem, this role can be detailed in many other specialized sub-roles. One of those sub-roles is the ***Ecosystem Manager***, who is responsible for the definition and enactment of the processes related to the MSE Governance. Such processes involve both Ecosystem Members and the Virtual Manufacturing Enterprises. The former take part to processes such as the above introduced members management, legal processes and intellectual property management, whereas the latter must be monitored during their lifecycle by the Ecosystem Manager in order to check for their compliance to the ecosystem's policies and rules.

During its lifecycle a ***VME*** delivers its product-service bundles to the customer through operational processes (VME Operations). The specific roles of the VME's members cannot be exploded here since such level of detail has to be defined case-by-case (in the servitization process), for each particular operational process. It is to be noticed that the customer (which is the final user/consumer of the VME's value proposition) is involved in the VME Operations too. Finally the ideation process (i.e. the initial part of the Innovation Process) is concurrently participated by any member of the ecosystem (including VMEs, although not represented in the figure) and by customers, depending on the policies defined by the issuer of the idea campaign.

2.2 Use cases of the IEP

This section presents the most important use cases that the IEP has to support with respect to the three types of processes: operational processes, governance processes and innovation processes. The use case has been defined analysing the functionalities required in the MSEE Description of Work in conjunction with the initial user requirements gathered in SP5 and the outcomes of the technical meetings, performed in the first 6 month of the project, which involved the partners in charge of the other work packages related to the IEP.
For each use case it is provided a table with a general description and the actors involved.

2.2.1 Operational Processes

Operational processes are directly derived from the servitization process and involve the following specific roles:

- **IT Service Manager**: the actor in charge of the management of the IT infrastructures of the ecosystem and of the management of the lifecycle of the processes-instances in the operative and runtime environment.
- **VME Manager**: once a VME has been created, the role of VME Manager is responsible for its management from a strategic and tactical point of view.
- **VME Operator**: once a VME has been created, the role of VME Operator is responsible for its management from an operative point of view.
- **Ecosystem Manager** and **Customer** are defined in section 2.1.4.

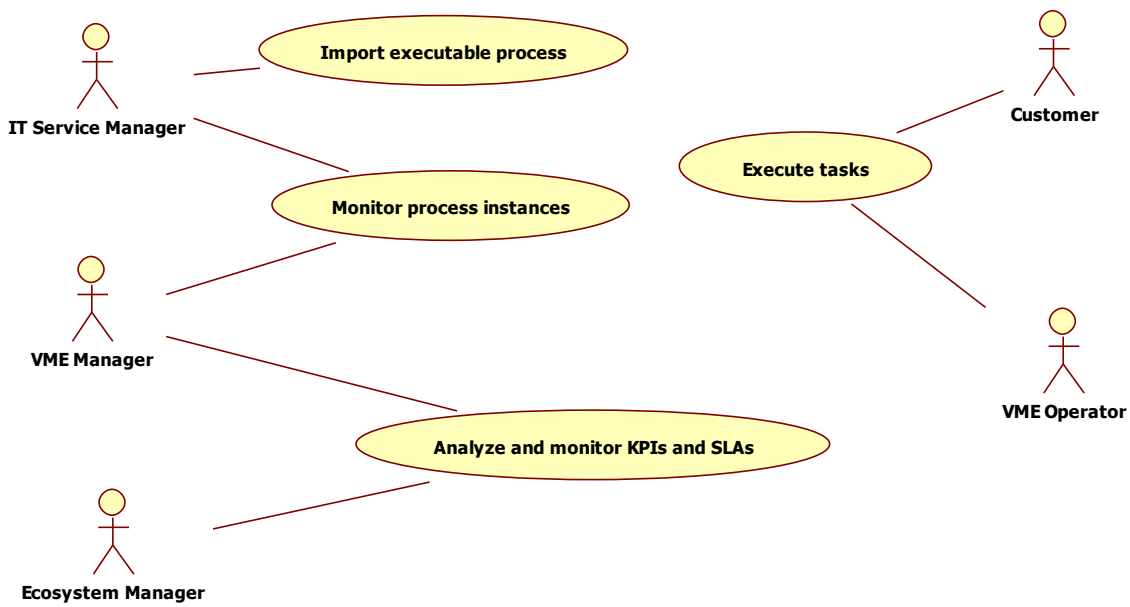

Figure 6 - Operational processes use cases

The following tables list and describe the main use cases identified for the management and execution of the operational processes.

Use Case ID	UC-OP-01
Name	Import executable process
Description	The executable model of an operational process is issued by the Generic Service Development Platform. This model is an extension of the BPMN 2.0 [8] XML serialization: such extension (TSM level) must be supported by runtime environment of the IEP.
Users	IT Service Manager

Table 2 - UC-OP-01

Use Case ID	UC-OP-02
Name	Execute tasks
Description	The executable model of the operational process (UC-OP-01) must be deployed in a run-time environment which is able to manage the state of different instances of the process. The actual execution of an end-to-end process requires the orchestration, in the proper order, of the defined human and automatic tasks. The runtime environment of the IEP has to support in a seamless way both the type of tasks. In particular human-oriented tasks of the process must provide a GUI to the proper user in order to collect his/her input. Each user who has a role in the operational process must be notified of the tasks that he/she is authorized/requested to accomplish. Automatic tasks of the process (e.g. web-services) must be invoked from any source of a private enterprise, of the ecosystem or of the Open Internet. Invocation of web-services from the different levels (private enterprise, ecosystem, Open Internet), must be subject to proper security and authorization rules.
Users	Customer, VME Operator

Table 3 - UC-OP-02

Use Case ID	UC-OP-03
Name	Monitor process instances
Description	Each instance of the operational processes (and each task that it is composed of) has a specific status (started, idle, running, completed, …). Processes and tasks need to be monitored both technical (e.g. technical problem of IT services) and business sides (e.g tasks to be completed by specific users) to prevent and handle unforeseen exceptions and/or bottlenecks. The monitoring of process instances includes the possibility to pause, stop, re-start and cancel any instance.
Users	VME Manager, IT Service Manager

Table 4 - UC-OP-03

Use case ID	UC-OP-04
Name	Analyze and monitor KPIs and SLAs
Description	The analysis of process logs and of any other data source through Business Intelligence techniques allows the managers of the VME and of the MSE to check for the compliancy of the operational processes to the KPIs and SLAs defined in the servitization process.
Users	VME Manager, Ecosystem Manager

Table 5 - UC-OP-04

2.2.2 Ecosystem Governance Processes

Ecosystem Governance Processes enable the management of every aspect of an ecosystem and involve the following specific roles: Ecosystem Member, Ecosystem Manager, VME and IT Service Manager. Those roles have been already briefly introduced in the previous sections.

Figure 7 - Ecosystem Governance Process use cases

The following tables list and describe the main use cases identified for the management and execution of the ecosystem governance processes:

Use Case ID	UC-GP-01
Name	Manage tangible/intangible assets
Description	The participation of ecosystem members to VME is based on the tangible and intangible assets they can expose and leverage to deliver the product-service bundle to the customer. Tangibles and intangibles assets have to be virtualized (i.e. represented in a digital format) and then represented in a service-oriented interoperable language. The service-oriented representation has to be stored in a repository/registry that can be accessed by the authorized human and

	automatic agents through specific GUIs and APIs. Some of the ecosystem governance processes enabled by tangible/intangible assets aaS are: • Search for competencies and capacities for the creation of a new VME • Assessment of assets and potential gaps in the competencies that are available in the ecosystem
Users	Ecosystem Member
MSEE Project Cycle	First

Table 6 - UC-GP-01

Use case ID	UC-GP-02
Name	Model process
Description	The creation of new workflows that are needed to support the governance of the ecosystem has to be guided by the high-level description of the Ecosystem Manager. He/she must be enabled to model the activities of the workflow. The modelling language has to be compliant to those defined by SP1 in the SLMToolbox
Users	Ecosystem Manager
MSEE Project Cycle	First

Table 7 - UC-GP-02

Use case ID	UC-GP-03
Name	Develop process
Description	After the Ecosystem Manager has defined the high-level model of a workflow for supporting the management of the ecosystem, the IT Service Manager acts as a developer and completes the model in order to make it executable in the runtime environment of the IEP.
Users	IT Service Manager
MSEE Project Cycle	First

Table 8 - UC-GP-03

Use case ID	UC-GP-04
Name	Monitor process instances
Description	Each instance of the ecosystem governance processes (and each task that is composed of) has a specific status (started, idle, running, completed, …). Processes and tasks need to be monitored both

	technical (e.g. technical problem of IT services) and business sides (e.g tasks to be completed by specific users) to prevent and handle unforeseen exceptions and/or bottlenecks. The monitoring of process instances includes the possibility to pause, stop, re-start and cancel any instance.
Users	Ecosystem Manager, IT Service Manager
MSEE Project Cycle	First

Table 9 - UC-GP-04

Use Case ID	UC-GP-05
Name	Analyze and monitor KPIs
Description	The analysis of process logs and of any other data source through Business Intelligence techniques has to allow the manager of the MSE to check for the compliancy of the governance processes to the ecosystem-level KPIs defined in WP25.
Users	Ecosystem Manager
MSEE Project Cycle	Second

Table 10 - UC-GP-05

Use case ID	UC-GP-06
Name	Execute tasks
Description	The execution of the process is the orchestration, in the proper order, of the defined human and automatic tasks. The runtime environment of the IEP has to support in a seamless way both the type of tasks. This use case has the same functional and non-functional requirements of UC-OP-02
Users	Ecosystem member, VME
MSEE Project Cycle	First

Table 11 - UC-GP-06

Use Case ID	UC-GP-07
Name	Assess maturity level
Description	Assess the maturity level of an enterprise with respect to the servitization-level and the degree of involvement of the ecosystem.
Users	Ecosystem member
MSEE Project Cycle	Second

Table 12 - UC-GP-07

Use Case ID	UC-GP-08
Name	Manage change
Description	Define a roadmap that, after the assessment (as is) of the maturity level of an enterprise, guides the enterprise in the change process toward the desired to be (in terms of servitization level and level of involvement of the ecosystem)
Users	Ecosystem member
MSEE Project Cycle	Second

Table 13 - UC-GP-08

2.2.3 Innovation Processes

Innovation Processes allow ecosystems to elaborate new ideas and create new value. The focus of the IEP is on the initial part of the Innovation Process (Idea Generation and Concept Development), that is the Ideation Process, which is participated by the following roles:

- **Customer:** customers of an organization are involved in the idea management process. They participate by submitting new ideas and by commenting and voting existing ideas;
- **Ecosystem Member:** employees of an organization that is part of the ecosystem contribute to idea generation and improvement by submitting new ideas and by commenting or voting existing ideas;
- **Reviewer** – the reviewer can make some review about submitted ideas, like financial review or technical review.
- **Decision Maker**: this is the actor that looks at the rating, comments, reviews and statistics related to ideas and selects ideas that are valuable for a business initiative.
- **Campaign Organizer**: he sets up ideas campaign using simple on-line form or he promotes it via other means (e.g. e-mail)
- **Moderator**: a person given special powers to control the submissions and comments of other participants.

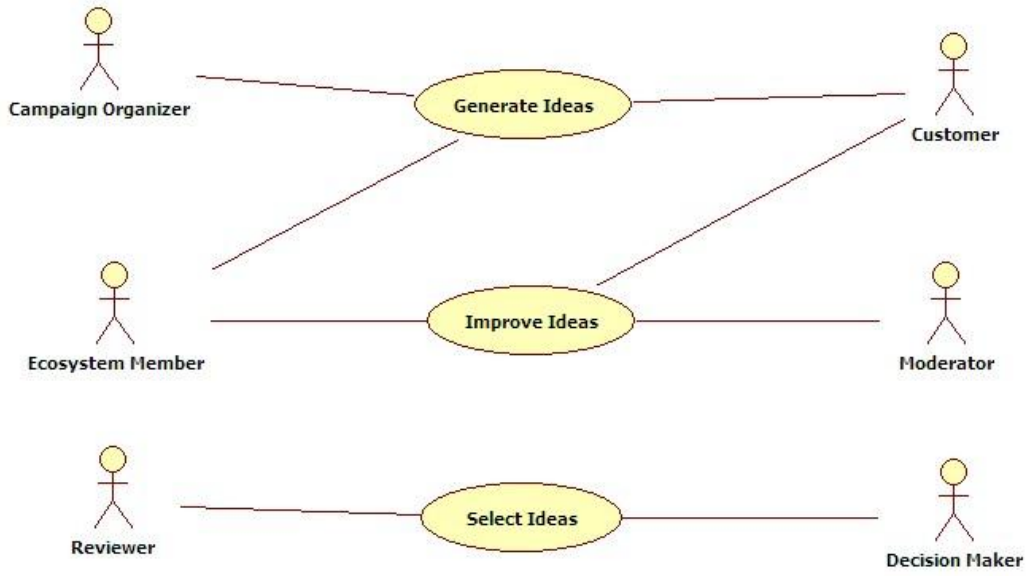

Figure 8 - Innovation Processes use cases

The following tables list and describe the main use cases identified for the management and execution of the innovation process.

Use Case ID	UC-IP-01
Name	Generate Ideas
Description	Ecosystem Members and Customers submits ideas both responding to a campaign created and managed by a Campaign Organizer or to an on-going theme for which the ecosystem always requires ideas. They can associate ideas with some existing categories or tags to help the overall management of the ideas.
Users	Campaign Organizer, Ecosystem Member, Customer
MSEE Project Cycle	First

Table 14 - UC-IP-01

Use Case ID	UC-IP-02
Name	Improve Ideas
Description	Ideas generated by Customers and Ecosystem members are improved with votes, comments, more tags and attachments submitted by other users. Votes and Comments contribute to the calculation of rating of the ideas. This phase is monitored by a Moderator, who enforces the compliancy of the community to specific policies and rules.
Users	Customer, Ecosystem member, Moderator
MSEE Project Cycle	First

Table 15 - UC-IP-02

Use Case ID	UC-IP-03
Name	Select Ideas
Description	Ideas are processed by Reviewers (e.g. Domain Experts), that can create reviews of ideas for better analyze some aspects like financial or technical ones. They can do some studies of ideas like SWOT analysis and attach results to ideas. They can also create interlinks between ideas to identify similar ideas, opposite ideas or extending ideas. The Decision Maker has access to ideas reviews and ratings and decides which ideas are accepted and which are refused.
Users	Reviewer, Decision Maker
MSEE Project Cycle	First

Table 16 - UC-IP-03

3 Functional architecture

In order to support the variety of use cases presented in the previous section, the functional architecture of the IEP (Figure 9) is here introduced showing the main interaction of its functional blocks with the rest of the MSEE IT System and with the outputs of the other work packages. The architecture, which will be described in the following section, implements through a set of components, the functionalities defined in chapter 2. Every component supports one o more use cases, and it is based on the contribution coming from other MSEE work package: the IEP, as depicted in the previous sections, supporting different phases of the MSEE Service System Life Cycle, can be considered the main entry point of the MSEE service system. The IEP integrates a set of SP2 tools, but also has to communicate with the SP4 platforms in order to support the operational processes developed by the Development Platform and published in the Delivery Platfom (for further details see [12]) .

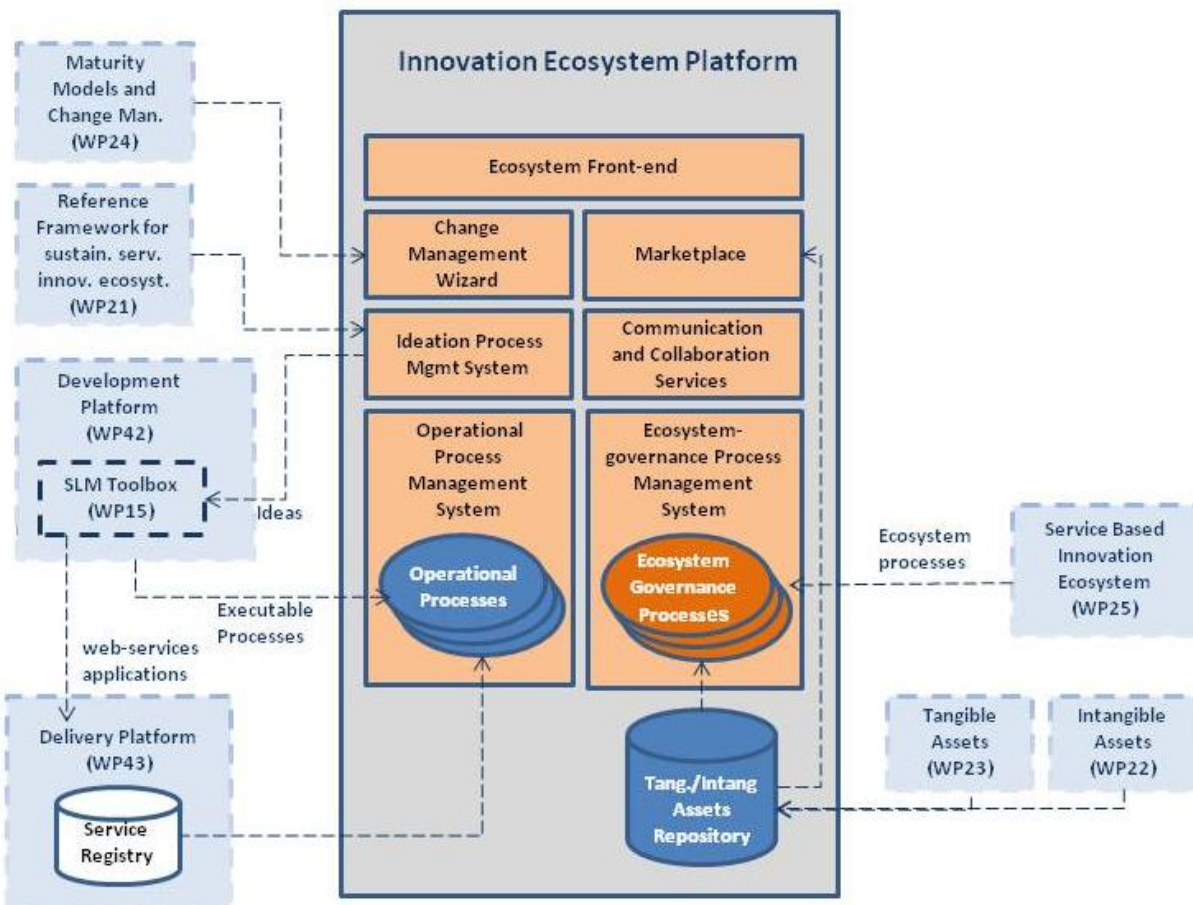

Figure 9 - Innovation Ecosystem Platform

The Innovation Ecosystem Platform is composed of six main components.

The Ecosystem Front-end is the single point of access to the Ecosystem for ecosystem members and it integrates in a single graphical environment all the other components.

The **Change Management Wizard**, represents a concrete tool to guide the enterprises through change management processes and toward a higher level of maturity with respect to their servitization level and their involvement in the ecosystem. This component can be considered a concrete implementation of the results coming from WP24 and is considered a tool to support the MSE Governance.

Ideation Process Management System, is a system for fostering and managing the creation of new and innovative ideas through cross-organizational and user-inclusive initiatives. The component provides to the SLMToolbox (which is part of the Generic Service Development Platform) a continuous and unstructured stream of suggestions, hints, ideas and potential opportunities to innovate the product-service value-proposition that the ecosystem offers to its customers.

The, **Collaboration Services** component, supports the three kind of processes through a coherent set of communication and collaboration services that allow members of the ecosystem to share contents and collaborate on their evolution, discuss and interact through web2.0 tools.

Marketplace and Tangible/Intangible Assets Repository, provides the means to ecosystem members to manage virtualized tangible and intangible assets and to offer their capabilities as services to the market (internal to the ecosystem or open to the Internet).

Operational/Ecosystem-governance Management System, components represent the front-end applications for the enactment and monitoring of operational and ecosystem-governance processes. Both provide a graphical interface that on one side allows members of the ecosystem to participate to those processes in specific human-oriented activities through ad-hoc defined web-forms, and on the other side presents the quantitative measures of Performance Indicators evaluated on the executed processes.

The run-time engine in which both operational and ecosystem-management processes are executed allows the orchestration of automatic (e.g. web services) and human-oriented (i.e. web-form enabled) activities. It has to be noticed that operational processes are modelled and engineered by the Development Platform, whereas ecosystem-governance processes are fully managed in the IEP. Moreover in the case of operational processes, automatic activities are retrieved and invoked through the Generic Service Delivery Platform; in the case of ecosystem-management processes, activities will mostly rely on services based on the virtualization of tangible/intangible assets of the ecosystem's members.

In the following chapters will be described in details the IEP components and the relative modules: for each of them are provided information about the functionalities they will support and about the MSEE project cycle in which the functionalities will be implemented: the first cycle corresponds to the first release of IEP prototype (M12), the second to the final release (M24).

3.1 Ecosystem Front-end

The Innovation Ecosystem Platforms integrates several tools and services aimed to support the whole MSEE Service System Life Cycle: in order to achieve this objective the IEP has to provide to final users (i.e. any actor that is involved formally or informally in the ecosystem) a consistent and simple front-end. The IEP frontend will consist in a web portal where actors can access the tools and functionalities that are granted to their role and can interact with others stakeholders through communication and collaboration services.
The IEP will integrate the different functional components and make them available to the final users through this single point of access.

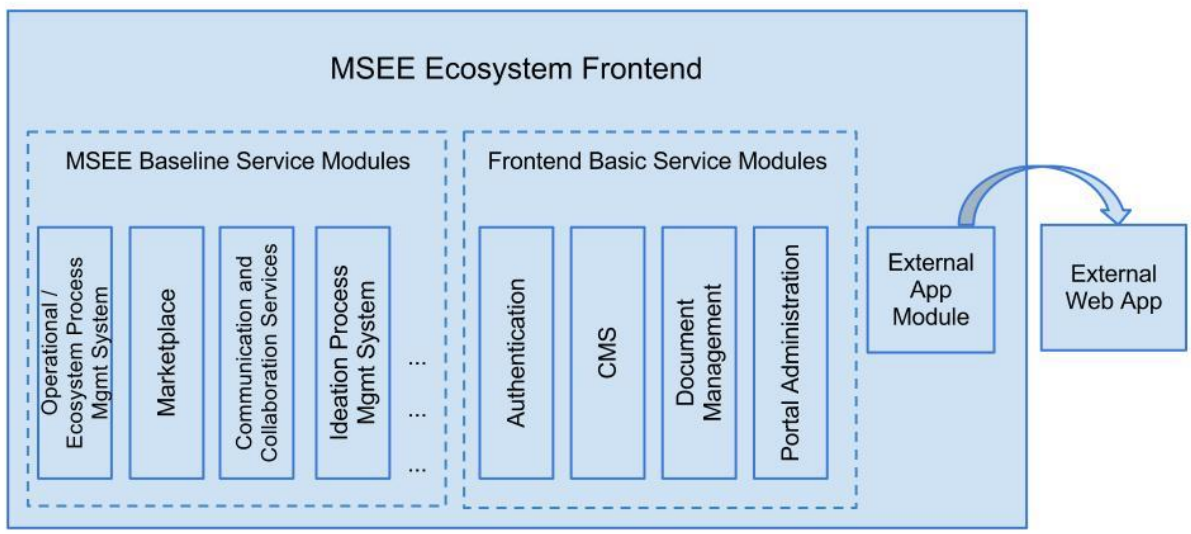

Figure 10 - Ecosystem Front-end

For the purpose of integrating different components, the Ecosystem Front-end will provide these modules:

- Frontend basic service modules, that are the main functionalities of the web portal that will host all the other MSEE components and tools :
  - Authentication Module: that allows the authentication of the users and it could be support several type of authentication based on different devices or security rules. It provides Single Sign On functionalities allowing to use a single user account for several services/applications.
  - Portal Administration Module: dedicated to the portal administrator, provides the functionalities related to the portal configuration: for example the portal administrator will be able to manage the user roles, the graphical templates, security policy etc.
  - Content management system (CMS): this module allows to create, upload and publish different types of contents on the portal such as textual news, and multimedia files. The contents can be restricted to specific users or groups.

- Document manager: through this component the users can upload/download, organize and find any type of files. The access to the files can be restricted to specific users or groups.

- MSEE baseline modules: these represent the integration modules of the services/applications specific for the MSEE context related to the objectives to be achieved by the Innovation Ecosystem Platform. These modules will provide access to the functional components described in the next sections.
- External app module: the frontend will be able to integrate the external applications, using API, if provided, or maintaining the same UI embedded in the portal (e.g. iframe approach).

3.2 Change Management Wizard

Enterprises that are part of a Manufacturing Service Ecosystem and that want to fully leverage the potential opportunities that the ecosystem could provide in the mid-long term, should be supported in finding their way toward an effective servitization of their value-delivery model and toward a fruitful collaboration with other members of the ecosystem. Enterprises should thus be helped in assessing their maturity level with respect to such two objectives and should be guided through a concrete roadmap toward the desired to-be with the support of the right technological and methodological tools (i.e. tools developed in MSEE or other).
The Innovation Ecosystem Platform, in the second cycle of the project, will integrate a set of components to support such requirements. The technological components will reflect the methodologies and the models defined in WP24 in order to provide concrete means to guide the enterprises through change management processes that may vary from simple and short-term modifications of the technological infrastructures, to complex and long-term switch of the business model. At this stage it is only possible to outline the high-level functionalities of the components, which will be refined in the second version of this deliverable (D26.2)

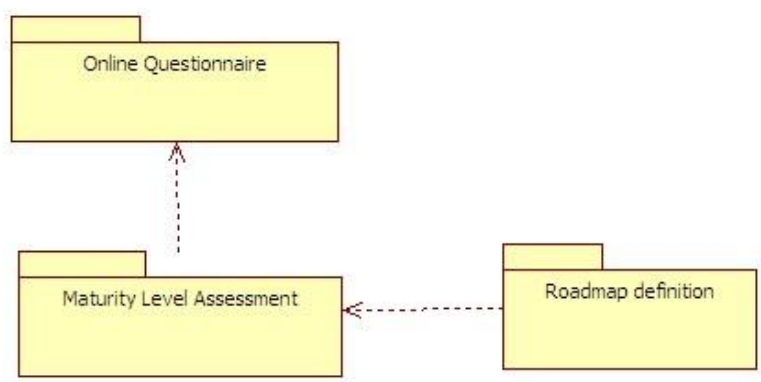

Figure 11- Change Management Wizard modules

Name	**Online Questionnaire**
Description	This component represents the means to collect information from the ecosystem members about their level of involvement in the ecosystem and about their servitization level.
Functionalities / services	• creation and management of online surveys with open and closed questions • answering questionnaires
Dependencies	
User	The Ecosystem Manager defines, creates and manages online questionnaires to be used by ecosystem members.
MSEE Project Cycle	Second

Table 17 - Online Questionnaire

Name	**Maturity Level Assessment**
Description	This component takes as input the data produced by the online questionnaire and automatically computes the maturity level of the enterprise that responded to it. The procedures and the models to be used in this component will be defined in WP24.
Functionalities / services	• definition of the maturity level of the enterprise with respect to the involvement in the ecosystem and to the servitization of its techno-organizational structures. • adaptation of the computation procedures with respect to the type of input provided by the online questionnaires
Dependencies	This component depends on the data provided by the Online Questionnaire component
User	The Maturity Level Assessment component is used by any ecosystem member to evaluate its maturity level in the ecosystem and in the servitization process.
MSEE Project Cycle	Second

Table 18 - Maturity level assessment

Name	**Roadmap Definition**
Description	On the basis of the maturity level (as is) and of the desired to be, this component creates a roadmap for the enterprise in order to be guided through the change management process. The roadmap will provide actions to be enacted, decisions to be taken, methodologies to be adopted and tools to be used. In particular the roadmap will suggest which MSEE tools should be used to optimize the change management process.
Functionalities / services	• definition of the desired to-be status, in terms of servitization level and level of involvement in the ecosystem • creation of a roadmap to guide the enterprise from the current maturity level to the target one • identification of actions, tools and means to support the change management process
Dependencies	This component depends on the Maturity Level Assessment component which provides the initial status of the enterprise.
User	The Roadmap Definition component is used by any ecosystem member who needs guidelines to improve its maturity level in the MSE.
MSEE Project Cycle	Second

Table 19 - Roadmap definition

3.3 Ideation Process Management System

The Ideation Process Management System is a component devoted to identify, select and improve, in an iterative manner, the ideas that stimulate the servitization cycle. For this reason the Ideation Process Management System can be used by different ecosystem actors in every phase of the servitization process: at the beginning, to produce ideas for new product/services, during their development, contributing to refine needs and requirements, but also at the operative phase in the evaluation of the results achieved. The Ideation Process Management System is based on the concept of "ideas": an idea can be a solution, request or proposal about a product, service or process innovation. An idea can be described by a title and a textual or graphical description. Therefore an idea can be classified and associate with the following concepts:

- **Campaign** – an idea can be triggered by idea campaign or context and incentives that come with;
- **Comment** – users can comment ideas proposed by others. An idea can have one or more comments;

- **Vote** – users can vote ideas;
- **Rating** – votes and comments of an idea contributes to the calculation of the rate of an idea. Ideas with best rating are candidates for acceptance and realization;
- **Revision** – ideas can be reviewed by a specific review team or group of experts with knowledge in the field;
- **Idea State** – an idea can be related to a state that represents the position of the idea in the workflow of the idea management process. The state of an idea can be managed by a decision maker or changed by an event or trigger. Some possible states of an idea can be: DRAFT for a new end incomplete idea, UNDER REVIEW for an idea well defined that has to be reviewed by reviewers under some aspects like financial or technical aspects, and ACCEPTED or REFUSED.
- **Tag and Category** – an idea can be associate with some tags or categories that helps users to classify and search ideas,
- **Attachment** – an idea can be related to some attachments that contain details for better idea description. An attachment can be a document, an image or a link related to an idea.
- **Interlinking** – some ideas can be interlinked because of some relationship like opposite ideas, similar ideas or extending ideas.

The Ideation Process Management System is composed by three modules: the Ideation Process Collaboration that provides the web GUI to access the functionalities of the ideation process; the Ideation Process Analysis that provides a series of reports, supports the Decision Maker in the final selection of the more valuable proposals. These two modules are related with the Ideation Process repository that contains all the information produced during the ideation process.

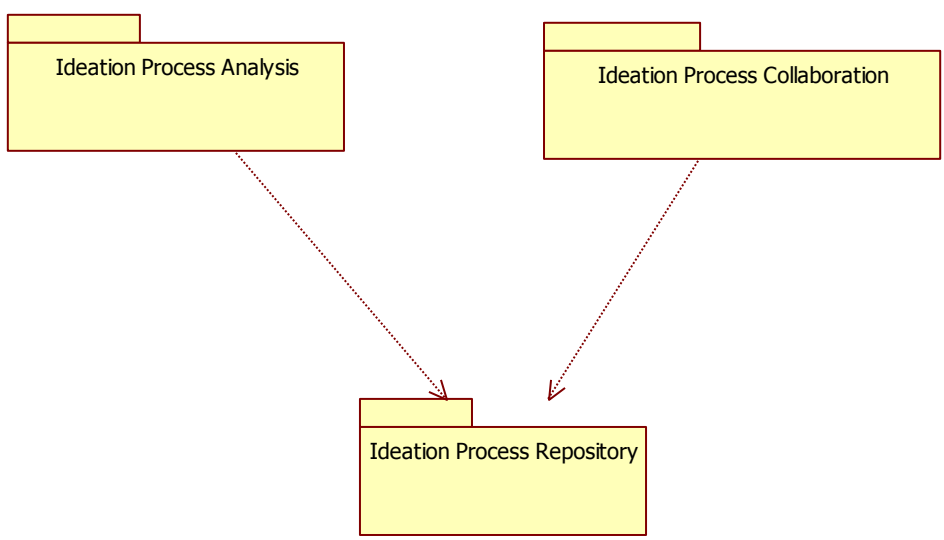

Figure 12 - Ideation Process Management System modules

Name	**Ideation Process Repository**
Description	The Ideation Process Repository contains the data produced by Ideation Process Management System and it provides the functionalities to access them exposing a set of API to the other modules.
Functionalities / services	• Store, get, delete and modify data in the Ideation Process Repository
Dependencies	The Ideation Process Analysis and Ideation Process Collaboration System modules access to the Ideation Process Repository.
User	Ideation Process Repository is not used directly by users but by the above-mentioned modules through specific API.
MSEE Project Cycle	First

Table 20 - Ideation Process Repository

Name	**Ideation Process Collaboration**
Description	The Ideation Process Collaboration is the core module of the whole system: it provides to the users, through a web UI, all the functionalities needed in the ideation process, such as creation and managing of ideas and campaign, comment, review and voting.
Functionalities / services	• **Create and Manage Campaign** – campaign are created by an Idea Campaign Organizer to stimulate employees or clients to submit ideas about a theme. • **Create and Manage Idea** – Customers and Ecosystem Members can create ideas in response of a campaign or to an on-going theme for which the company always requires ideas. • **Search Ideas** – with this function it is possible to search ideas by its name, description or other properties and to view ideas filtered by some aspects, like state or category. • **Vote Idea** – this function allows to vote ideas. Users can vote ideas during the Idea Improvement phase. Votes and comments of ideas contribute to the calculation of the idea ranking. Ideas with highest ranking are candidates for

	acceptance and production. • **Comment Idea** – this function allows to comment ideas during the Idea Improvement process. • **Attach file to Idea** – an idea can be better explained with the help of an image or a file related to some specific domain application. This function permits to attach files to ideas, like image files, spreadsheets or presentations. • **Assign State to Idea** – assigning a state to the idea it is possible to trace the current position of the idea in the idea management process. The first state of an idea can be Draft, then Under Review, Accepted, Rejected or Implemented. • **Tag Idea** - this function permits to tag ideas to better describe it and allows it to be found again by browsing or searching. • **Review Idea** -. This function enables reviewers and domain experts to refine ideas adding more authoritative and structured information like financial reviews or technical review. • **Interlink Ideas** - this function allows users to add links between ideas during the selection phase to point out aspects among ideas relationships like similarity, opposite and extension. It is then possible to link two ideas as Similar Ideas, Opposite Ideas or Extending Ideas.
Dependencies	This module accesses to Ideation Process Repository to store and retrieve the information related to the ideation process.
User	All the Ecosystem Members can use the Ideation Process Collaboration System: in particular Customer, Campaign Organizer, Moderator, Reviewer and Decision Maker.
MSEE Project Cycle	First

Table 21 - Ideation Process Collaboration

Name	**Ideation Process Analysis**
Description	All the campaigns and the ideas produced and the relative information and attributes can be visualised using a set of predefined reports (e.g. statistics, charts etc.) provided by the Ideation Process Analysis module. This module will help the Decision Maker to select the more valuable ideas taking in account different parameters (e.g. number of votes, typology of the proposers etc.)
Functionalities / services	• Generation of statistics overview about the information produced in the Ideation Process Management System, such as average amounts of ideas and comments, number of users involved, time statistics etc. • Generation of charts and graphical reports about ideas, contests, reviews and comments;
Dependencies	This module accesses to Ideation Process Repository to retrieve the information related to ideation process for the statistics and report generation.
User	This module will support the Decision Maker in the analysis and selection of ideas.
MSEE Project Cycle	First

Table 22 - Ideation Process Analysis

The following figure shows an activity diagram of a typical ideation process that involves, in the different tasks, the actors mentioned in the previous section. In particular it is shown that the ideation process can start from a new campaign created by a Campaign Organizer or directly from an idea proposed by an Ecosystem Member or a Customer that can enrich its description in different ways (tagging, voting commenting the idea etc). After that the Reviewer or Moderator can refine the ideas collected making a selection or adding new information. Finally the Decision Maker, with support of Reviewer can rank and select the more interesting ideas that satisfy specific requirements.

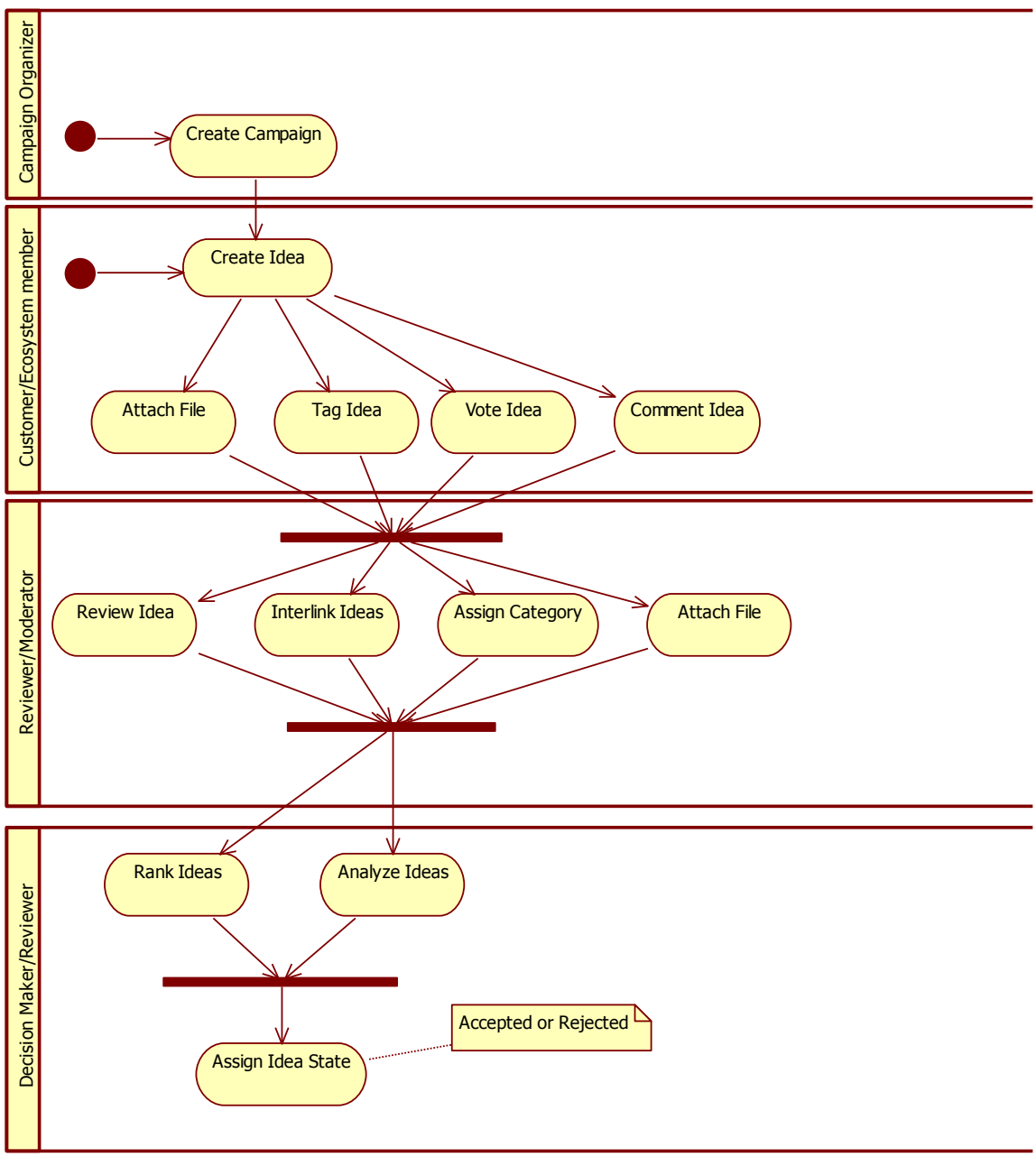

Figure 13 - Idea Management Process Activity Diagram

3.4 Operational/ecosystem-governance process management system

The use cases outlined in the previous sections for the management and execution of operational and ecosystem-governance processes highlighted some relevant overlapping and similarities. In particular, all the requirements related to the orchestration, monitoring and analysis of human-oriented and automatic tasks can be fulfilled by a common set of functional components. The only remarkable difference among business-process related requirements for operative and ecosystem-governance use cases is related to the source of the process models that have to be executed. In the case of operational processes, the executable model is the result of the servitization process and is thus provided by the Generic Service Development Platform. Such model (that is at TSM-level) is the customization (for the process-engine specific needs) of a BPMN representation (at BSM-TIM level) originally developed through the SLMToolbox (definitions about BSM-TIM-TSM can be found in [1]). In the case of ecosystem-governance processes, the modelling and the development of the processes/workflows is done directly by means of tools (Process Editor) that are part of the Innovation Ecosystem Platform.

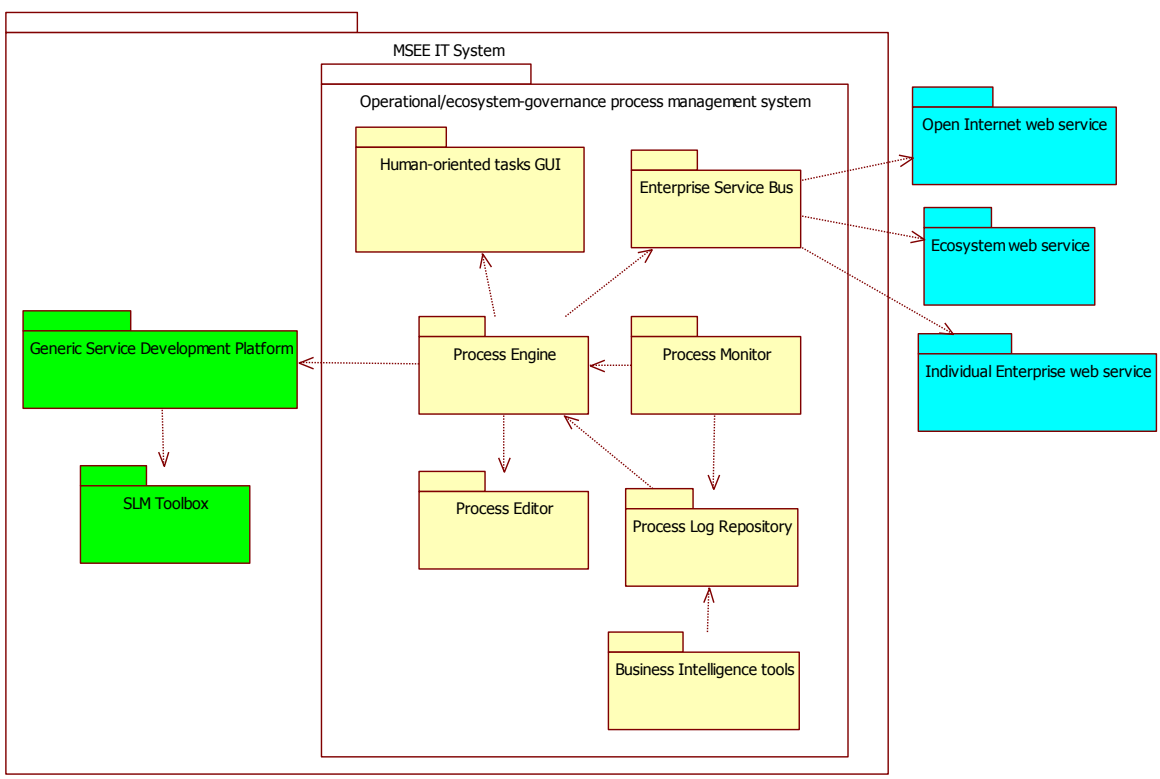

Figure 14- Operational/ecosystem-governance process management system modules

This section describes the functional components of the Innovation Ecosystem Platform that will support the following use cases:

- UC-OP-01: Import executable process
- UC-OP-02/ UC-GP-06: Orchestrate tasks
- UC-OP-03/ UC-GP-04: Monitor process instances
- UC-OP-04: Analyze and monitor KPIs and SLAs
- UC-GP-02: Model process
- UC-GP-03: Develop process
- UC-GP-05: Analyze and monitor KPIs (which partially overlaps with RQ-OP-04)

Name	**Process Engine**
Description	The process engine is responsible for instantiating and executing process models. Some activities are automatic (e.g. sending an email or invoking a web service), other activities are manual and they needs human actors to be executed (e.g forms to be completed with specific information). The process engine manages and persists the state of the process instances and of its tasks during their entire lifecycle.
Functionalities / services	• Importing and deploy of executable process models • Orchestration of human-oriented and automatic tasks • Management of multiple process instances
Dependencies	The process engine is the runtime environment for both operational and ecosystem-governance processes/workflows. Their execution requires the previous import of the executable model from the Generic Service Development Platform (for operational processes) or from the Process Editor (for ecosystem-governance processes)
User	The process engine is not directly used by users but offers its functionalities through API to the other process management system modules.
MSEE Project Cycle	First

Table 23 - Process Engine

Name	**Process Editor**
Description	Ecosystem-governance processes, unlike operational ones, are not specified and implemented in the servitization process. They need a dedicated and light-weight set of tools in order to be rapidly described and implemented respectively by Ecosystem Managers and IT Service Managers. The Process Editor module is responsible for supporting such actors in any activity required to transform an high-level definition of a business model to the correspondent executable version for the chosen Process Engine module.
Functionalities / services	• Modelling of business processes through BPMN graphical notation • Detailed design of the process in terms of low-level properties of the activities in order to make it executable in the Process Engine module • Graphical design of web-forms for human tasks
Dependencies	The generation of the executable process model (i.e. the specific format of the output of this module) must be compliant to the formats accepted by the Process Engine module. The language used for the high-level description of the ecosystem-governance process model must be aligned to the ones adopted in SP1 and used in the SLMToolbox (i.e. BPMN).
User	The Ecosystem Manager uses this module to describe at a high-level the process to be implemented The IT Service Manager defines the technical details of each activity in order to make the process executable.
MSEE Project Cycle	First

Table 24 - Process Editor

Name	**Process Monitor**
Description	Both operational and ecosystem-governance processes have to be monitored in their execution to verify the different tasks performed assuring a correct error management. The Process Monitor module provides functionalities to directly monitor the different instances of processes through a dedicated graphical console that allows the

	users to analyse the whole execution of them. The user will be able to view the status of the processes (started, idle, running, completed etc.), of the single tasks, information about the users involved in the manual tasks and to manage exceptions and errors that can occur.
Functionalities / services	• Status Monitoring for process instances and tasks • Monitoring of task assignments and responsible users • Error and exception handling
Dependencies	The information reported by the Process Monitor is based on the data produced by the Process Engine and stored in the Process log Repository
User	The Ecosystem Manager and VME Managers use this module to monitor the status of processes (respectively for ecosystem-governance and operational processes) and tasks and to manage error and exception. The IT Service Manager can also use this module in order to manage the possible technical problems reported by the web console.
MSEE Project Cycle	First

Table 25 - Process Monitor

Name	**Process Log Repository**
Description	The Process Log Repository provides the functionalities to access to the logging data produced by the Process Engine during the operational and ecosystem-governance processes execution. The logs are information at different levels: status of tasks, process instances, IT services called in the process executions, users involved in the task, error reports etc.
Functionalities / services	• Store, get, delete and modify data in the Process Log Repository
Dependencies	The information managed by the Process Log Repository are produced by the Process Engine and used by the Process Monitor
User	The Process Log Repository is not used directly by users but by the above-mentioned modules through specific API.
MSEE Project Cycle	First

Table 26 - Process Log Repository

Name	**Human-oriented tasks GUI**
Description	Operational and ecosystem-governance processes can be composed by automatic or human-oriented tasks. In order to perform these manual activities the users will use a specific graphical interface (e.g forms). The Human-oriented task GUI consists in a dynamic web user interface to be filled by the user with the information needed to complete the specific tasks.
Functionalities / services	• Worklist: list of tasks that the user is allowed or required to perform. • Web forms and graphical elements to support users in manual tasks
Dependencies	The Process Engine activates this module for the tasks that require direct actions by the users to be completed.
User	This module can be used by a generic Ecosystem Member involved in a manual task of operational or ecosystem-governance processes
MSEE Project Cycle	First

Table 27 - Human-oriented tasks GUI

Name	**Enterprise Service Bus**
Description	The Enterprise Service Bus is the module that allows the communication and interaction among services and application involved in the operational or ecosystem-governance processes orchestrated by the Process Engine. Services based on different technologies and communication protocols can interact through the Enterprise Service Bus that translates messages from the consumer to the provider of the service: in order to perform this translation the Enterprise Service Bus uses some adapters for the different languages/technologies to be supported (Java, SOAP, BPEL etc).
Functionalities / services	• Message-processing transformation and routing • Validation against schemas for sending and receiving messages • Support for synchronous and asynchronous transport protocols, service mapping • Adapters for supporting integration with different

	technologies and legacy systems,
Dependencies	This module depends directly from the Process Engine that orchestrates the whole process demanding the invocation of the services to the ESB
User	The Enterprise Service Bus is not used directly by users but by the Process Engine through specific API.
MSEE Project Cycle	First

Table 28 - Enterprise Service Bus

Name	**Business Intelligence Tools**
Description	The Business Intelligence Tools perform advanced analysis on the process log data and on other data sources that may be of interest: while the Process Monitor reports information about processes and tasks status and errors, the Business Intelligence Tools allow to define and monitor high level Key Performance Indicators and SLA based on specific business objectives related to the ecosystem and the operational processes
Functionalities / services	• Define and monitor Key Performance Indicators and SLA • Graphical reports of the analysis
Dependencies	The Business Intelligence tools perform their analysis on the data provided by Process Log Repository
User	The Ecosystem Manager uses this module to define and monitor Key Performance Indicators The IT Service Manager uses this module to define and monitor technical SLA
MSEE Project Cycle	Second (the complete development and integration of the BI tools depends on the requirements defined in WP25, which are due at M12 in a first version and M18 in a final step)

Table 29 - Business Intelligence Tools

3.5 Marketplace and tangible/intangible assets repository

This section describes in detail the two functional blocks *Marketplace* and *Tangible/Intangible assets repository* introduced at the beginning of this chapter. They are described together because their coupling allows the complete support of the complex use case UC-GP-01 (Manage tangible/intangible assets aaS (as a Service)) in the context of the Manufacturing Service Ecosystem. It is to be noticed that tangible/intangible assets will be managed by the IEP as services published through the USDL [9] [10] format (or an "extended" version of USDL, customized for the needs of the manufacturing industry). Such services are published by ecosystem members after that the proper pre-processing procedures (defined in WP22 and WP23) have been performed (e.g. identification and prioritization of the assets, extraction of their relevant information from the proper data-sources, transformation of the information into the USDL format, integration of the missing data into the USDL description of the service, …). The actual focus of D26.1 is on the USDL Repository and on the Marketplace that leverages its capabilities. Nevertheless software components that will support the pre-processing procedures will be taken into account since the USDL Repository will be populated through their usage.

The following figure represents the main functional components (and their dependencies) that are part of the process which leads to the publication of tangible/intangible assets as a service through the Marketplace, starting from the output of the aforementioned pre-processing procedures.

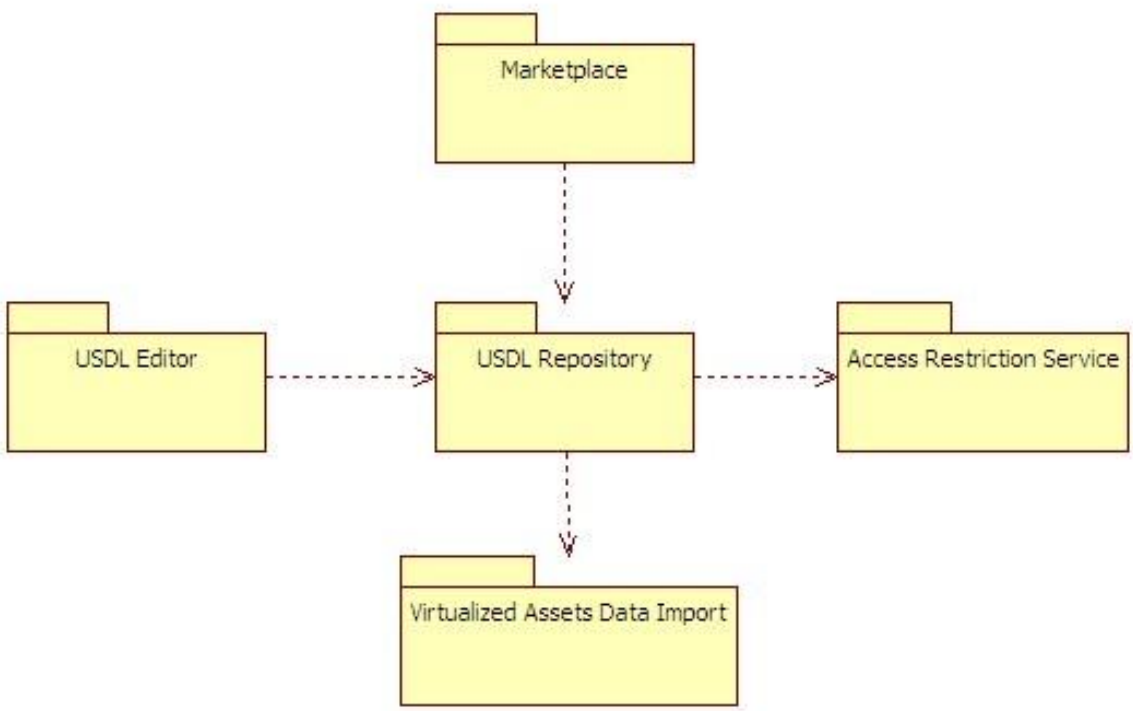

Figure 15 - Marketplace and tangible/intangible assets repository modules

Name	**Marketplace**
Description	The marketplace component has the aim of providing a point of contact among offers and demands of services built upon tangible/intangible assets. Functionalities provided by the Marketplace are meant to allow the dynamic and flexible creation of ad-hoc VMEs to respond to business opportunities through the search and identification of the right capabilities (i.e. tangible and intangible assets aaS) available in the MSE. It is to be noticed that the functionalities of the Marketplace have to be accessible both to human users and to automatic systems through web applications and through APIs respectively. The functionalities listed hereafter represent a first, high-level view on the component, based on the specifications available from FI-WARE project [11]. The second version of this deliverable (D26.2, due at month 18) will provide a refined specification of the functionalities thanks to a deeper analysis of MSEs specific requirements.
Functionalities / services	• management of marketplace participants (providers, consumers, …) • management of offerings (creation, modification, deletion) • searching & listing of services that match specific criteria (e.g. price range, service level, technical characteristics, functionality, …) • compare services along multiple attributes (e.g. price, service level, …) • management of feedback and ratings about services
Dependencies	The Marketplace component depends primarily on the USDL Repository, which is the logical place where the concrete descriptions of the published services are maintained.
User	On one hand ecosystem members, acting as providers, can publish their services and manage offers by specifying supplementary information like their price model and their category; on the other hand, acting as consumers, they can search, compare and rate services according to their necessities and to the requirements coming from the market.
MSEE Project Cycle	Second

Table 30 - Marketplace

Name	**USDL Repository**
Description	The USDL Repository component is responsible for maintaining the USDL description of services that are based on tangible and intangible assets. Its functionalities consist of simple operations that allow the creation, update, retrieval and deletion of service descriptions. Additional functionalities are related to the listing of descriptions accessible to the user and the searching/querying of the descriptions with respect to their properties. It is to be noticed that this component will support distributed architectures by allowing service descriptions maintained in a specific repository to refer to descriptions maintained by other repositories. At this moment WP22 and WP23 are working on the elaboration of a clear strategy to extend USDL toward the necessities of MSEE, with particular reference to the description of tangible/intangible assets aaS. The development of the Repository will be carried on in parallel with the definition of such strategy, which will be reflected by the data-model underpinning the repository.
Functionalities / services	• Create new service descriptions • Read service descriptions or parts of service descriptions. • Update parts of service descriptions • Delete service descriptions • List all the service descriptions accessible to the user • Filter specific attributes of the service description with respect to the access-level (individual enterprise, ecosystem, open internet) • Query/Search service descriptions
Dependencies	This component depends on • the *Access Restriction Service* component for the management of the rules for filtering the service description attributes with respect to the access-level. • the *Virtualized Assets Data Import* component for the description of services based on information extraction and transformation from tangible/intangible data sources.

User	The USDL Repository is indirectly used by ecosystem members, acting as service consumers, through the Marketplace to search for competencies and capabilities inside the ecosystem. Ecosystem members act also as service providers when they create new service descriptions to be managed on the USDL Repository.
MSEE Project Cycle	First

Table 31 - USDL Repository

Name	**USDL Editor**
Description	This component has the aim of allowing service providers to describe their service through the USDL formalism.
Functionalities / services	• searching and retrieval of service descriptions from in the USDL Repository • editing of service descriptions • storing of service descriptions in the USDL Repository
Dependencies	This component depends on the USDL Repository for storing and retrieving USDL descriptions.
MSEE Project Cycle	Second

Table 32 - USDL Editor

Name	**Virtualized Assets Data Import**
Description	This component is responsible for extracting, transforming and making available relevant data about tangibles and intangibles to be published as services on the USDL Repository. Procedures implemented by this component are defined in WP22 and WP23 and may be fully automatic, semi-automatic or fully manual.
Functionalities / services	• identifying and extracting relevant data about tangible/intangible assets from different data sources (enterprise systems, paper-based documents, machine-tools, …) • transforming the extracted data into a format compatible with the USDL representation
Dependencies	No dependency is currently considered

User	This component is used by ecosystem members who want to provide tangible/intangible assets as services and that have the necessity to retrieve the relevant information about such assets from different and heterogeneous data sources.
MSEE Project Cycle	Second

Table 33 - Virtualized Assets Data Import

Name	**Access Restriction Service**
Description	Information about tangible and intangible assets and about their characteristics may have different degrees of privacy requirements and constraints. For example the rate of malfunctions of a certain machine tool and the real cost of a specific employee (whose capabilities are to be published as services) are important information for the enterprise that actually owns such assets, but they should never be accessed by external actors (e.g. potential consumers of the service). MSEE defines three architectural levels for the publication of services: Individual Enterprise, Manufacturing Service Ecosystem, Open Internet. At the Individual Enterprise level all the information are available for the members of the single enterprise. At the Ecosystem level, members of the ecosystem can access a sub-set of the information available at Individual Enterprise level (i.e. sensible and privacy-constrained data are filtered). Finally at the Open Internet level (public level) further information are obscured to actors that are not formally part of the MSE. The Access Restriction Service component has to provide the means to manage and enforce the rules for accessing the asset-related information in the USDL description.
Functionalities / services	• fine-grained management of access rules to attributes reported in the USDL description • enforcement of the access restriction rules
Dependencies	No dependency is currently considered
User	This component is used by the ecosystem member that, acting as a service provider, wants to restrict the access to certain information related to the tangible/intangible assets.
MSEE Project Cycle	Second

Table 34 - Access Restriction Service

3.6 Communication and Collaboration services

One of the most important values of an ecosystem is related to the degree on relationship and interactions among the entities taking part in it. The Innovation Ecosystem Platform will foster and support these interactions through a set of services that will allows companies and customer to communicate, collaborate and make business. These services will be provided by two modules:

Figure 16 - Communication and collaboration services modules

Name	**Communication**
Description	The aim of this module is to give to the ecosystem entities a wide range of tools to communicate using different medias and devices, but also to take advantage from the social networking. The communication could be private or public in the ecosystem or also published outside of the ecosystem. This type of services would be used for example to promote products, to provide support to customers etc.
Functionalities / services	• messaging services (e.g. chat, instant messaging, mailing list), • forum, wikis • social network integration (facebook, twitter etc) • support for mobile communication
Dependencies	
User	This module will be used by any ecosystem member and other external actors (e.g. Customers)
MSEE Project Cycle	First

Table 35 – Communication module

Name	**Collaboration**
Description	Collaboration module provides a set of services and tools aimed to foster cooperation among the ecosystem members in order to create new business opportunity and facilitate the creation of VME. The knowledge sharing will be one of the important targets of this module: it will be achieved providing shared document environments. Also some collaboration tools will be based on collaborative software developed in COIN project (e.g. collaborative project management, collaborative production plan).
Functionalities / services	• Knowledge sharing environment • Tools for collaborative tasks execution
Dependencies	The collaboration module uses the functionalities provided by communication module
User	This module will be used by any ecosystem member
MSEE Project Cycle	First

Table 36 - Collaboration module

The following table resumes the list of the Innovation Ecosystem Platform modules showing the expected deadline for their implementation: as said in the previous sections some modules will be provided for the first MSEE cycle (M12), others will be implemented in the second cycle (M24) after requirements refinement and new contribution from relative work packages.

Innovation Ecosystem Platform Modules	**First Cycle (M12)**	**Second Cycle (M24)**
Ecosystem Frontend	**X**	
Change Management Wizard		
Online Questionnaire		**X**
Maturity Level Assessment		**X**
Roadmap Definition		**X**
Ideation Process Management System		
Ideation Process Collaboration	**X**	
Ideation Process Analysis	**X**	
Ideation Process Repository	**X**	
Operational/ecosystem-governance process management system		
Process Engine	**X**	
Process Editor	**X**	
Process Monitor	**X**	
Process Log Repository	**X**	
Human-oriented tasks GUI	**X**	
Enterprise Service Bus	**X**	
Business Intelligence tools		**X**
Marketplace and tangible/intangible assets repository		
Marketplace		**X**
USDL Repository	**X**	
USDL Editor	**X**	
Virtualized Assets Data Import		**X**
Access Restriction Service		**X**
Communication and Collaboration services		
Communication	**X**	
Collaboration	**X**	

Table 37 - IEP modules in the MSEE cycles

4 Technological components

In this section we will describe some technological products that could be used to cover part of the functional requirements presented in the previous chapters. The described products are related to the modules to be implemented in the first MSEE cycle. This analysis will support the evaluation of the best technologies to be used in the next weeks/months for the implementation of the IEP.

4.1 Ecosystem Front-end

4.1.1 Liferay

The Liferay Portal [13] contains a rich list of features, many of which are made available through the use of AJAX. This technology allows user to setup a customized homepage by dragging and dropping the available functionalities (portlets and gadgets). This task can be easily carried out by any internet user without the need of any specific technical skill. In the same way, users can customize the Portal look and feel by choosing one the different themes made available by administrators.
Liferay offers a complete range of functionalities to manage each aspect of the portal, from simple changes to more complex operations such the addition of a new portlet. Administrative tasks are managed through an AJAX developed interface which allows to drag and drop the necessary objects.

The list below shows a summary of the main features offered by Liferay.

- Support to the major J2EE Application Server, database and operating systems (about 700 combinations of deployment).
- Customization layout through themes.
- Services Content Management System (CMS) and Documental based repository standard JSR 170 (Apache Jackrabbit).
- Collaborative Services .
- Availability of over 100 portlets.
- Availability of AJAX interface.
- Multilingual (over 34 languages already available).
- Support for SSO by CAS.
- Availability development of tools for the main IDE (Eclipse).
- Among the available features, DAWLATI will publish the following basic services offered by Liferay:
- Content management (CMS)
- Collaboration services: This component provides:
- Messaging: through a portlet similar to a mail client, users can manage and exchange messages with other registered users.
    - Address Book: support service to user information to which users can send a message. The information will be managed through LDAP.

  - Forum: system for managing on-line forums. Each discussion group (Forum) consists of "arguments" (Topic) and each topic from a set of messages (Thread).
  - Shared calendar: tool for managing a shared calendar appointments between a working group.
  - Wiki: portlets for managing content so shared.
- Documents sharing
- Security: Authentication and Authorization System integrated with LDAP.

4.1.2 Drupal

Drupal [14] is an open source content management system. It is a modular application written in PHP. It is compatible with different SO like Linux, Windows and Mac OS X and web servers like Apache or IIS. For data storing it can work with MySql or PostgreSQL database management systems.

Drupal has a modular design, composed by a core module that provides some basic functionalities, and others optional models and template that are in continuous development and provided new functionalities and customization.

The functionalities provided by the core module are:

- Creating, modifying and categorization of generic contents,
- Content searching,
- Inserting comments to content,
- Forum capabilities,
- Creating and participating to a survey,
- Collaborative writing projects,
- User and profile management, user accounting and registration,
- Communication among users and between users and administrators,
- Changing the site template using existing templates or creating new templates,
- Creating menus with a multi layer structure,
- Internationalization,
- RSS feeds,
- Granular access control management, configuring detailed permission rules related to single users or groups,
- Site usage statistics.

Drupal extension is done using Modules and Themes.

Modules extends Drupal providing new functionalities, like image galleries, WYSIWYG editors, mailing list, e-commerce. In the Drupal website it is possible to access to almost 16500 free modules.

Templates can change the default Drupal look-and-feel. They can be created using common design tools like PHPTemplate and XTemplate. The concept of template allows to separate user interface programming (HTML and CSS) from logic (PHP). Like in the case of modules also many themes are provided freely.

4.2 Ideation Process Management System

4.2.1 G2MO Idea Management

Gi2MO IdeaStream [15] is an Open Source web application developed during the research on Idea Management Systems in the GI2MO project, an initiative of GSI group of the Universidad Politecnica de Madrid. Ideastream is an Idea Management System build through a set of modules that customizes the popular CMS Drupal. Ideastream delivers the basic functionalities of an Idea Management System that are:

- Create, rate and comment ideas,
- Track idea's workflow status, like draft, under review, accepted etc,
- Organize idea with Idea Contests, Categories, Tags,
- Submit Idea Reviews that can be categorized by different criteria and supplied with rating,
- Browse and analyze review statistics,
- Customize access permissions to functionalities using different roles like administrator, manager, moderator, reviewer, inventor.

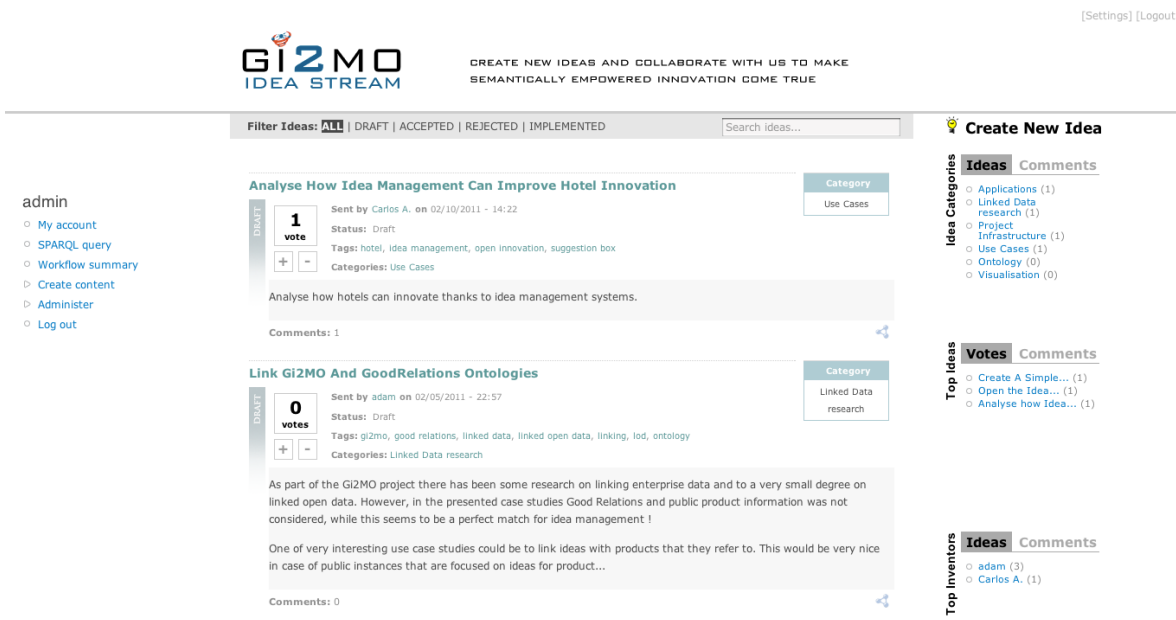

Figure 17 - G2MO Idea Management

As we can see in Figure# the IdeaStream frontend is composed of some sections:

- **Main section**: where an user can see ideas present in the system, filtered by idea workflow status, with some details: title, description, author, tags, categories, votes and comments. A generic user can also vote an idea and add a comment.
- **Create new Idea Link**: this link leads to a page where the user can submit a new idea.
- **Idea Categories**: this section shows idea categories. For each category the user can see the number of ideas, and, clicking on the category name, the ideas of the selected

category. In this section categories are ranked according to the number of ideas or to the number of comments.

- **Top Ideas**: this section shows the top ideas ranked according to the number of ideas or to the number of comments.
- **Top Inventors**: in this sections inventors are ranked. Also here ranking is done according to the number of ideas created by the inventor, or to the number of comments if inventor's ideas.

As mentioned before IdeaStream extends Drupal CMS. In particular the extension is made by two modules and a theme:

- The **IdeaStream Module**, that is the core part of the application, reconfigures Drupal creating data necessary for Idea Management,
- The **IdeaStream Elements Module**, that is a supplement module that delivers custom functionality that extends Drupal basic ones,
- **IdeaStream Theme**, that is a custom theme that reorganize the look & feel of Drupal, on order to present to the user some custom pages for idea management.

During the GI2MO project the GSI group developed other interesting modules that work together with the IdeaStream module, acting ad an axtension. These modules are:

- **IdeaStream Analytics**: that delivers a number of analyses and data views that improve the workflow of a manager or system administrator during their idea management processes. These views are:
  - Summary Statistics: that presents a general overview of all data in the Idea Management System, like count of posts per each content type: maximal, minimal and average amounts of comments or reviews per idea; time statistics,
  - Charts: that shows timeline statistics for ideas, contests, reviews and comments; and bar charts with breakdown of posts per each category type.
  - Idea, Contest and Review Management: that presents table views allowing to browse the content submitted by users, view basic statistics and filter this data based on various criteria
- **IdeaStream Similarity**: that delivers the ability to create and manage idea relationships in an Idea Management System. This module add functionalities that help users in duplicate detection, and in defining relationship between ideas, like opposite ideas, extending ideas or similar ideas. In detail it permits to:
  - Select relationship for newly created idea: during the idea creation phase this module can detect and suggest similar ideas,
  - Manage relationship: this functionality allows moderators to browse, add, delete and change all relationships of an idea, and approve or reject requests for idea similarity made by users
  - Create new ideas that extend existing ones: this functionality allows users to create ideas that extends an existing idea
  - Submit reports of idea similarity: this functionality allows users to submit reports for existing ideas to link them with other, and these requests can be approved automatically or by a moderator.

4.2.2 Open ideaL

Open IdeaL [16] is an open source Idea Management System consisting of a web application based on Drupal.

The system offers the clients community a tool for expressing their views on various issues and for running conversations around these issues. The tool also allows social change organizations to collect ideas from a large public, and develop a vivid and engaged conversation.

The heart of the conversation is usually an idea which the clients wish to see fulfilled by the organization. These ideas may be, for example, the development of a new product, enhancements of existing products, changing size or shape of packages, handling of customers with special needs.

Open ideaL includes tools for community managers which allow them to identify successful ideas (those with high chance of becoming a popular product), and to transmit them to professional treatment within the organization. Community members gain points for activity on the website, and their overall score grows particularly if their ideas become popular, or if the discussions in which they participate are "hot". This allows to reward good community members, and to use them to lead more discussions.

Open IdeaL basic components are:

- Idea Creation: for creating an idea with title, description and tagging
- Idea Presentation: for presenting an idea
- Idea Editing: this function allows idea's owner to change idea's information and administrator to manage ideas for example changing it's state.
- Comments presentation: for showing idea comments made by other users
- Idea tagging: for tagging an idea in order to find ideas according to tags
- In-Process projects: to take trace of the ideas workflow
- Trends and popularity identification
- Mail notifications
- Voting
- Friend scores
- Abuse flags: for ideas, comments and users
- Bookmarks
- Social network sharing
- RSS
- Blog
- Anti-spam protection

The pages for interacting with the Idea Management System are:

- Home Page
- Registration and login page
- Idea creation
- Idea presentation , that shows idea details including comments
- Category and Tag page
- Popular ideas

- Recent ideas
- In-Process Ideas
- Profile page
- Search for ideas
- Search for members
- Project page
- Analytics page

Other optional components are:

- Selected Ideas: this function allows administrator to select an idea and publish it in the home page
- Community Leaders: this function present, in the home page, the community members with the highest score
- "1 on 1": this function allows choosing two competing ideas and letting members vote for them, in a dedicate section in the home page,
- Challenge: this function allows the creation of a special page related to a challenge, for a certain category or tag. In this page community members are called to propose ideas for that specific category

4.3 Operational/ecosystem-governance process management system

4.3.1 Process Maker

ProcessMaker [17] is an open source business process management (BPM) and workflow software web application. This application allows organizations to design, automate and deploy their processes and workflows. ProcessMaker is completely web based and accessed via any web browser, making it simple to manage and coordinate workflows throughout an entire organization, including user groups and departments. It can also interact with other applications and systems such as ERP, business intelligence, CRM and document management.

This application allows business users and process experts, with no programming experience, to design and run workflows, automating processes across the systems, including human resources, finance and operations. With ProcessMaker it is possible to:

- **Design processes**: through a process designer that allows business analysts to design processes using an intuitive drag-and-drop interface easy to use,
- **Run processes**: through a web based application with profiling rules that allows users to interact with form-driven processes,
- **Extract reports from processes and Optimize processes**: through a dashboard that shows to managers KPI reports.

For the design phase ProcessMaker presents these functionalities:

- **Process Map Designer**: that is a graphical tool (depicted in FIGURE) allowing business analysts to create workflow maps. It is possible to define tasks, to connect tasks and define routing rules, to edit task properties. This tools offers also version control and debugging functions.
- **Dynaform Builder**: this functionality allows business analysts to design custom forms for human tasks. It is possible to add fields, whit a drag-and-drop interface, like text boxes, check boxes, drop downs, grids, date picker or file upload fields. Forms can be enhanced by technical users adding css or javascript improving form appearance and behavior.
- **Business Rules Engines**: this functionality allow business analysts to describe the logic to determine how a process should flow down one particular path instead of another in a decision gateway of the business process.
- **Output Document Builder**: this functionality is provided by a document editor that allows business managers to create electronic receipts, letters, confirmations, invoices, contracts or any other type of printable document, that can be added to tasks that automatically generate printable output with auto-filled information.

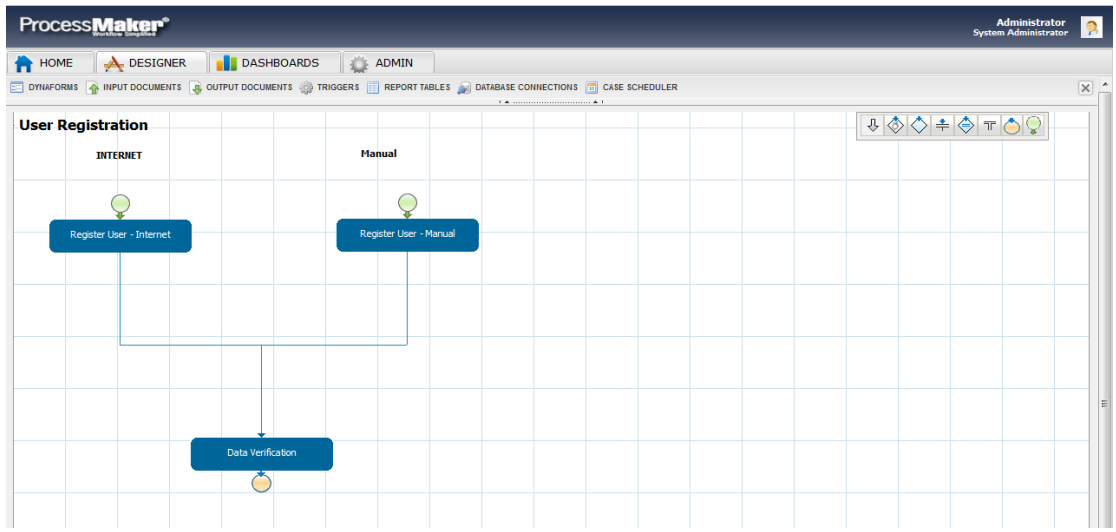

Figure 18 - Process Maker Cases Inbox

For the process execution phase ProcessMaker offers these functionalities:

- **Cases Inbox**: that is the main interface (depicted in Figure 18) for the end user interacting with processes. This is an area where users can create new process instances, track the progress of requests they have initiated or requests that require their input through. This interface presents an inbox similar to E-Mail inbox, where users can show new task handling requests. It also allows supervisors to review and reassign cases.
- **User management**: this functionality allows administrators to create roles, groups and departments to customize the solution to the organization structure. In this way users have different privileges according to their roles or groups or departments.
- **Document Management**: this functionality allows users to store and retrieve files that have been uploaded to ProcessMaker or automatically created by a process. These files are stored by a complete document management system included in ProcessMaker that allows to store files but also to manage custom tag clouds, custom folder paths and version control.
- **Process Monitoring**: this functionality is realized by a dashboard that provides real time monitoring efficiency and productivity of processes and users executing those processes. The dashboard is composed of some graphical indicators, called dashlet (depicted in Figure 19), that can be created by the administrator. Dashlets measure indicators over a predefined period, that can be Today, Yesterday, ThisWeek, Previous Week, Previous Month, This Month, This Year, and Previous Year.

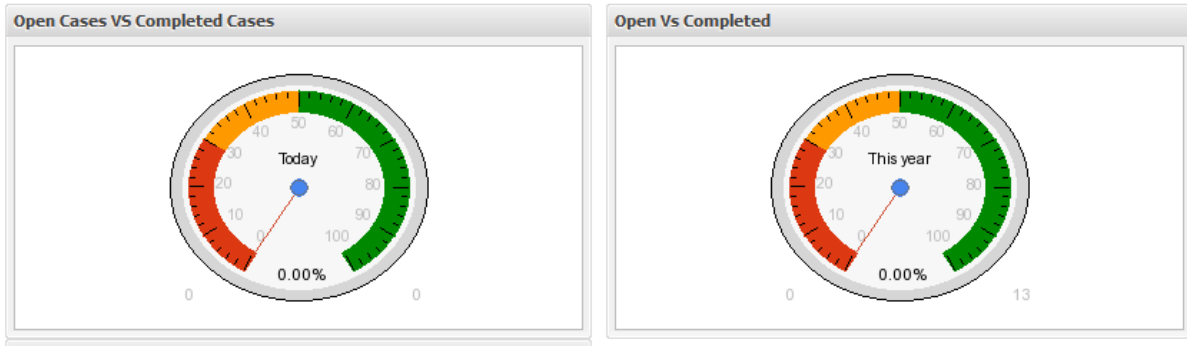

Figure 19 - Process Maker Dashlets

4.3.2 Aperte Workflow

Aperte Workflow [18] is an Open Source (GNU LGPL) BPM solution realized putting together some stable and mature frameworks. The frontend is a web application where users can design, deploy, run and monitor processes.

Aperte Workflow is composed of these modules:

- Integration Module: supporting persistence in accordance to JPA standard and using OSGi framework for plugin managements,
- Basic GUI: that is implemented using Vaadin framework,
- BPMN Engine: supporting some BPM implementations like jBPM and Activiti,
- Liferay Portlets: that form the user interface for interacting with the processes.
- Aperte Modeler: that is a BPMN2.0 editor, for process design and deploy.

The key featurs of this sytem are:

- It runs on Liferay, taking advantage of it's powerful functions,
- It supports various BPM process engines implementations like JBPM and Activiti,
- It is based on OSGi service platform, taking advantage of its modularity,
- It has a relational data model easily extendable,
- It presents a rich user interface,
- It contains a visual web based process editor,
- Supports enterprise integration with an embedded ESB like Mule ESB.

Technologies used from Aperte Workflow are:

- Liferay and Vaadin: for the implementation of the user interface,
- OSGi framework: for the plugin support,
- Hibernate: for data persistence,
- JTA: for distribute transaction support,
- jBPM and Activiti: as BPM engine implementation.
- Signavio Core Components: open source library for business process modeling

The tree principal phases in BPM process systems are modeling, running and monitoring.

In Aperte Workflow modeling is done using the web application Aperte Modeler, shown in FIGURE. Using the modeler it is possible to draw BPMN2.0 process diagrams, define human or automatic activities, create user interfaces for human activities, associate services to automatic activities and then deploy service bundles to the OSGi framework.

Process running is performed by the BPM implementation engine, and human tasks of running processes are accessed through liferay portlets.
These portlets are:

- New Processes: that allows users to start new process that are compatible with their profile,
- Task Queues: that shows to users their new tasks, recent tasks and other custom queues like subscriptions requests,
- Search: that allows users to search for processes instances,
- Task main portlet: that shows human tasks forms that users have to fill.

Process Monitoring is performed by tree functions in the liferay control panel that are:

- Aperte Workflow plugin manager: that shows installed plugins in the OSGi framework,
- Aperte Workflow Process Definitions: that show all deployed processes, and enable/disable processes,
- Aperte Workflow Process Instances: that allows administrators to track running and terminated process instances and their status. For each process instance it is also possible to show a process map that represents the process end tracks the current status, like in Figure 21.

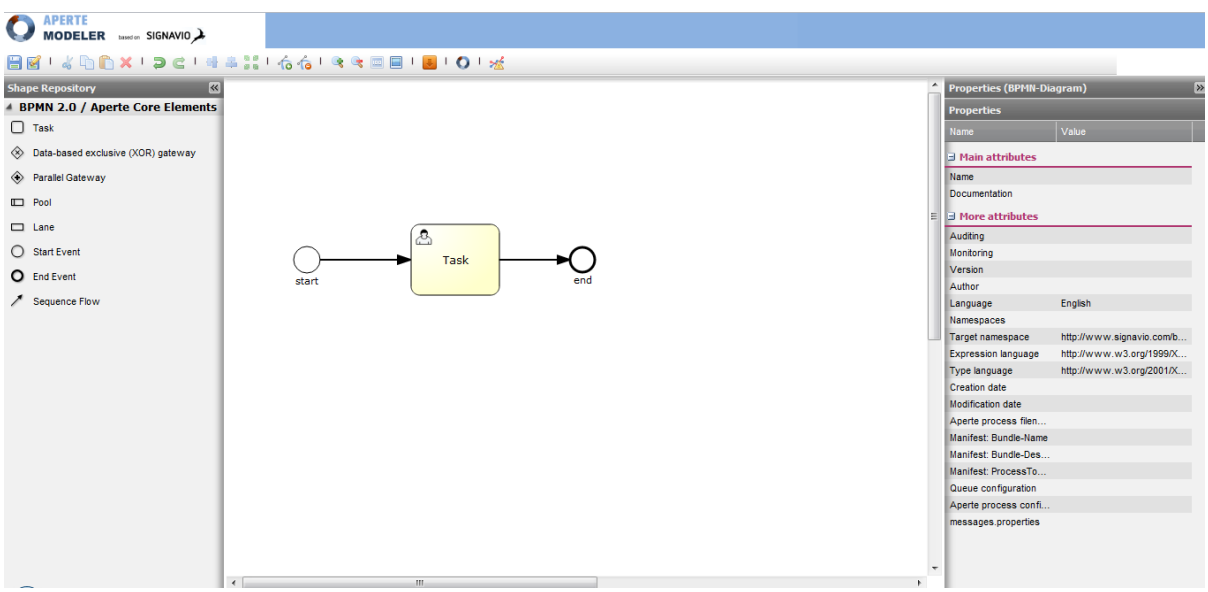

Figure 20 - Aperte Modeler

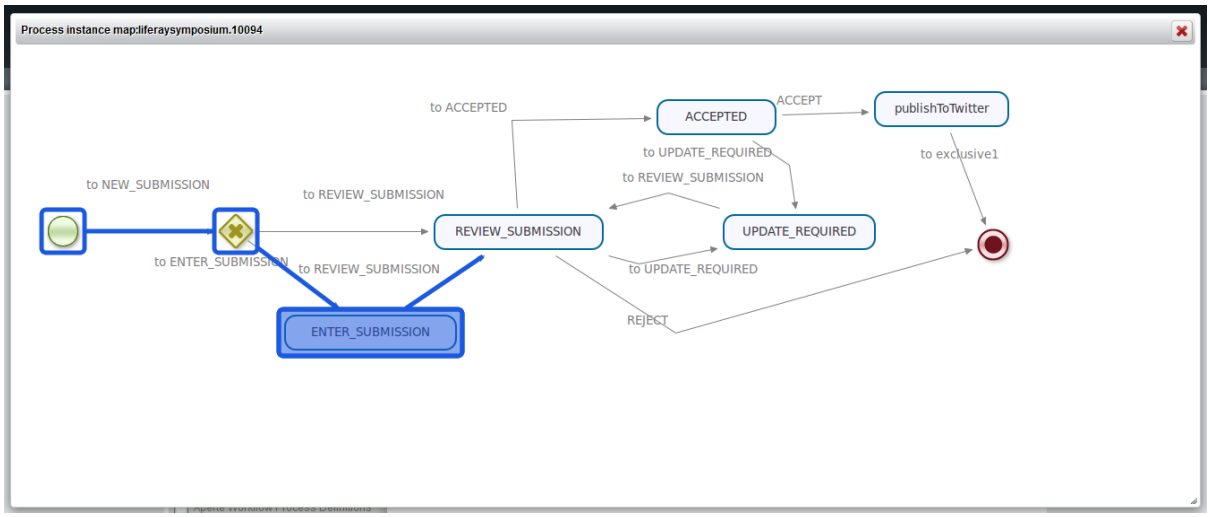

Figure 21 - Aperte process map

4.3.3 Bonita Open Solution

Bonita Open Solution [19] is an open source BPM and workflow suite. The application is composed of three major components:

- **Bonita Studio**: that allows users to graphically design business processes in BPMN2.0 standard notation. The studio is an Eclipse RCP application, that contains the core of Eclipse, some other Eclipse projects and Bonita specific plugins. It exports process to the Bonita Execution Engine. With this tool it is possible also to connect processes to external information systems like ERP, ECO anda databases. The studio includes a graphical form designer in order to create forms for human tasks. These forms are showed to users in a web context via the Bonita User Experience application. The studio makes Bonita Open Solution very integrable because it provides some import and exprort functionalities to the principal BPM formats. Indeed it provide import from BAR (Bonita Archive), but also from BPMN2.0, XPDL and jBPM3, and export to BAR and BPMN2.0
- **Bonita User Experience**: that is a web portal that provides user interface for interaction with process. It has an interface similar to a webmail, with a area in which the user can see new tasks to handle (inbox), favorite tasks, my tasks. This application therefore allows administrator to manage the process, handling processes, cases, categories, users. It includes a report area that shows KPI information. The web application is based on GWT, and is very modular. It supports some reporting systems like BIRT and JasperReports. Human task forms are created in the Bonita Studio and exported as xml file, these files are rendered in the Bonita User Experience using GWT.
- **Bonita Execution Engine**: is a Java API that allows to interact programmatically with processes.

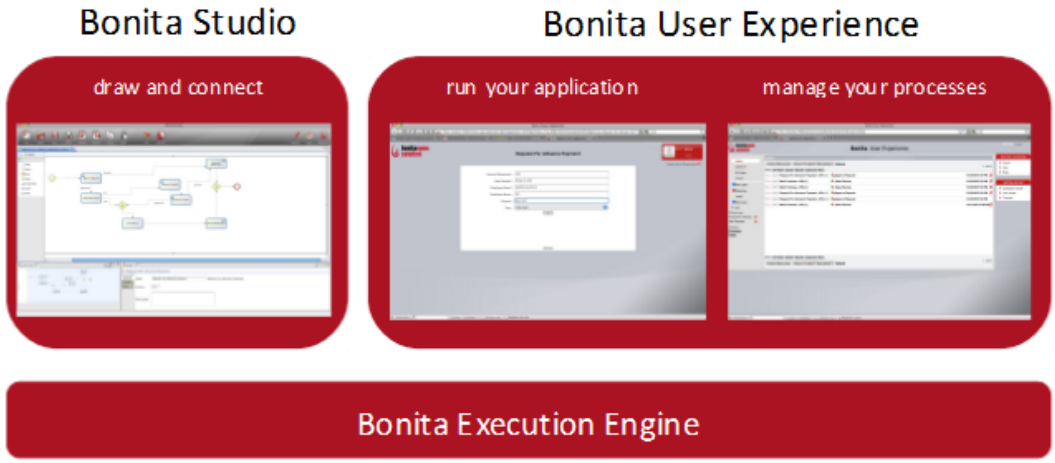

Figure 22 - Bonita open solution

As a BPM framework, the Bonita Open Solution presents three major functionalities, modeling, execution and monitoring, and for each of them presents some features:

- **Modeling**: this functionality is provided by the Bonita Studio, that allows user to model processed in a standard BPMN2.0 notation. The modeling can be done in a collaborative way, sharing model between business analysts, stakeholders and developers. The tool supports the versioning of processes, that allows to save and manage different versions of a process. It supports also user profiling, in order to expose to users only those features related to their profile's role. As mentioned before users can import model from external tools describer in some languages, and export models. It is also available a function that allows to simulate process execution, tracking parameters like cost, duration and resource consumption. After a process I modeled, developers can create forms for human tasks and link automatic tasks to many built-in connectors provided by the studio they also can create custom connectors helped by a connector development wizard. Finally processes are deployed to the Bonita Execution Engine.
- **Execution**: the execution functionality is handled by the Execution Engine and provided to the users through the Bonita User Experience web application. Like mentioned before this application offers an intuitive user experience with "inbox" interface. Users log in applications and see in the in box new tasks they have to handle according to their profile. The application provides also a function that allows to delegate tasks to other users when the task actor is unavailable. The User Experience has a multilingual support that includes some default languages, that can be increased through the Babili community translation tool.
- **Monitoring**: this functionality is provided to the administrators by the Bonito User Experience application, in order to track process instances and monitor some indicators. It is available an advanced dashboard that can be customized, showing statistics and reporting. It is also possible to define and monitor some KPIs. Administrators can also manage the process lifecycle, enabling, disabling or archiving processes. they also can

manage users, roles and groups. They also can manage active tasks suspending or resuming it.

4.3.4 Spagic

Spagic open source solution [20] is the solution used to meet both ESB/BPM middleware requirements and governance needs such as monitoring services and tools for development and deploy.

Spagic offers a Universal Middleware with an innovative approach to governance and to highly modular and configurable SOA solutions around a kernel that is based on OSGi (Open Service Gateway initiative) model. Spagic, thanks to OSGi modality, enables to produce solutions that are extensible and truly adaptable to different application domains, providing the following features:

- platform independence
- code reusability for different purposes and in different containers (e.g. J2EE, TP, ESB, BPM)
- optimum management of heterogeneous components life-cycle, both of business and infrastructure nature
- configuration and activation of single modules in order to get the most suitable solution for the specific project context.

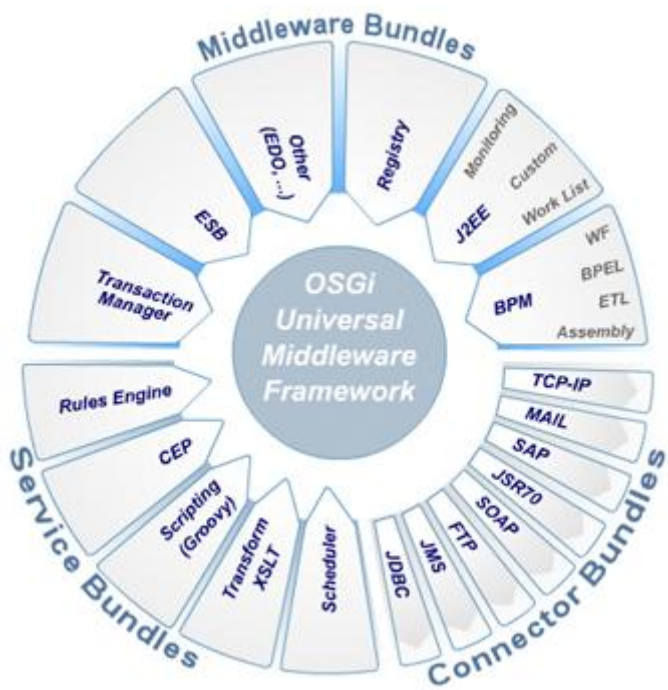

Figure 23 - Spagic configurable and activable modules

Spagic provides full support to the governance of SOA projects: tools for modeling, service definition, form creation for user activities, deploy control, connectors, BPM (Business Process Management) engines, infrastructure services and containers and monitoring environment.

The following figure depicts the different Spagic components.

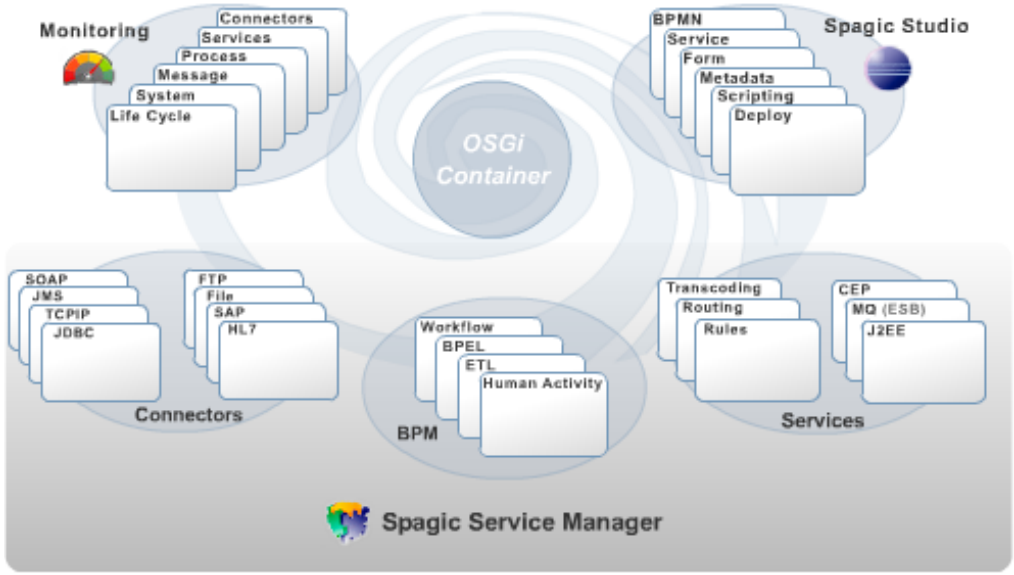

Figure 24 - Spagic components

Spagic Service Manager is a dynamic and adaptable Universal Middleware solution. The OSGi model modularity features of such solution assure a wide adaptability in different contexts, such as: Enterprise Middleware, Lite Middleware, installation of OSGi container in previously existent applications or in specific systems (mobile, custom hardware etc). The OSGi R4 framework, upon which Spagic is based, is Eclipse Equinox (http://www.eclipse.org/equinox/).

Spagic Service Manager components are:

- Connector: for cooperating with external services through HTTP, SOAP, REST, TCP/IP, JDBC, SAP, JMS, Mail. During the preliminary analysis activity before the beginning of the project, the necessary connectors will be individuated and in case new special connectors will be defined to be developed from scratch for the peculiar context.
- Middleware bundles: it includes infrastructure services such as
    - MQ, for asynchronously managing of messaging through queues. Apache ActiveMQ is set by default, but it is still possible to use other JMS provides such as the ones in Jboss Application Server.
    - BPM, for implementing a workflow process (JBoss JBPM) or BPEL (Apache ODE) with both manual and automatic tasks. Interaction with processes can happen both through a web service like a work-list and through calls from already existing applications (API or shared form execution)
    - Role Engine BRMS, through Jboss Drools open source solution

  - Registry, for registering services. Spagic relies on the Eclipse Equinox registry inside the OSGi framework. However it is possible to rely on other standard solution such as UDDI (Apache JUDDI)
- Service Bundles: for managing infrastructure services such as:
  - XSL/XSLT transcoding,
  - Scripting, through Groovy open source solution,
  - Batch scheduler, through Quartz open source solution.

Spagic Studio, based on Eclipse IDE, provides tools for design, development and deploy.

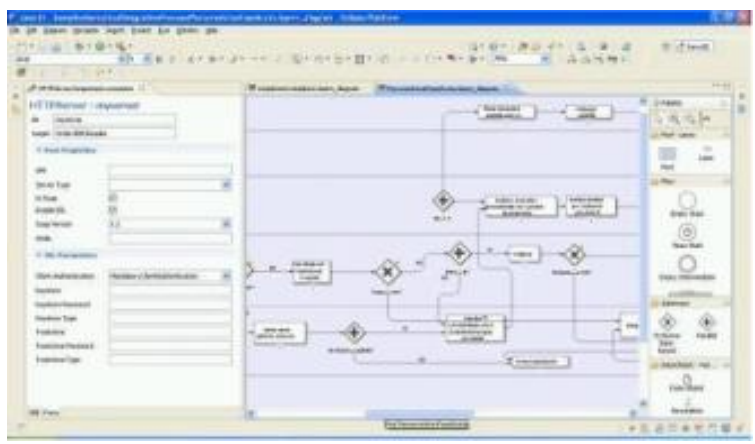

Figure 25 - Spagic Studio based on Eclipse IDE

The Spagic Studio main functions are:

- Service Designer for visual definition of services and publication through connectors
- BPMN process designer as an extension of open source Eclipse plugin for BPMN.
- XML/XSLT support
- XML Schema support
- WSDL Editor
- XML Message Editor with the possibility to generate file for Scribus: one open source editor for PDF Forms.
- Deploy management
- Support to Java development and Groovy script.
- Integration with SVN for version control

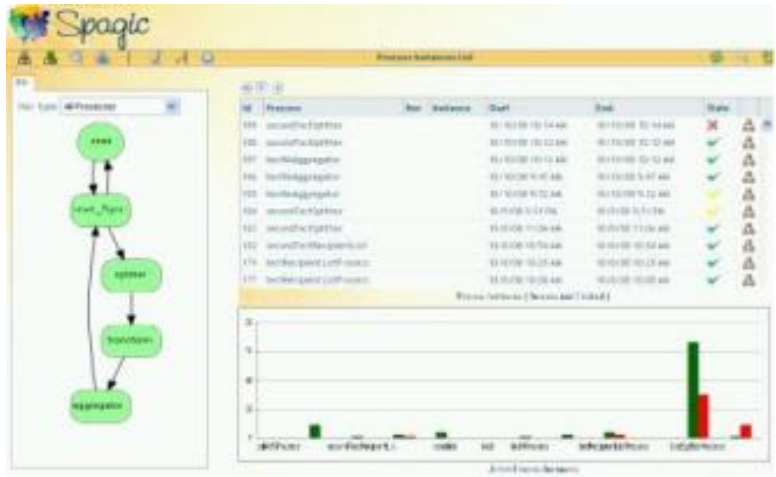

Figure 26 - Spagic web-based monitor

Spagic Monitor provides services for monitoring and managing the life-cycle of single module described in the application architecture.
In particular, through Spagic Monitor, it is possible to stop, pause and re-start a middleware service or process. In short, main functions are:

- monitoring of single services
- monitoring by processes
- monitoring of OSGi platform
- search by messages
- search by variable or relevant values
- managing of life-cycle
- logging.

All information is stored into Spagic MetaDB, that defines the data schemas described in the application architecture, complying with Eclipse EMF standard and implemented by Eclipse STP IM project, which Engineering is contributor of.

4.3.5 Petals ESB

Petals ESB [21] is an open source Enterprise Service Bus develop by the OW2 Consortium. It is implemented in java language and based on JBI (JSR 208) specification. It supports some standards like BPEL and SCA, and supports EDA capabilities. As an enterprise service bus it offers many connectors and processing components or services. Some connectors are File system connectors, FTP or SFPT, JMS, HTTP, mailing (SMTP/POP/IMAP) database connectors (JDBC or SQL). Some services are BPEL engine, XSLT transformation, RMI, Java scripting, CSV transformation and KPI notification. A key feature of Petals is its distributed, in the sense that a single bus can be constituted by several Petals servers.

Key features of Petals ESB are:

- it is distributed: as said before an ESB build with Petals can be composed of some server instances that communicate with each other over the network;
- it grantee the message delivery: if a message is sent to a server that in not available, for example is off-line, the message is stored pr sent to an equivalent service;
- it allows message routing: message routes are decided runtime according to some rules;
- it is robust: in case of incident the platform can restore its state and resend not delivered messages;
- it allows monitoring: it is possible to take trace of content and properties of messages that transit in the bus;
- it is extensible: because of it is component based, therefore it is possible to create new components to extend or modify the platform
- it is completed by several tools, for development, governance and monitoring of the processes.

the major tools that complete Petals ESB and that enhances its potentiality, are:

- **Petals Studio**: it is a graphical editor based on Eclipse IDE, that allows to create processes like BPEL and SCA improving productivity and accelerating Petals ESB configuration through automation wizards.
- **Petals Master**: that extends the capabilities of the Petals ESB allowing the governance of services, like indexing and documenting services.
- **Petals View**: that is a web application allowing to monitor data exchanges between services deployed by Petals ESB. Petals View is able to generate pertinent meta-data (KPIs) to drive management reporting that can be interpreted easily by business stakeholders.

Figure 27 - Petals ESB

4.3.6 Mule ESB

Mule ESB [22] is an open source Enterprise Service Bus framework. As a ESB it provides interactions between heterogeneous systems and services enabling developers to easily build high-performance, multi-protocol interactions between these systems. It presents a flexible and architecture, because it can be used as a stand-alone application, like an operating system daemon, or within an application server. It provides many functionalities for connecting different systems:

- It provides many transport connectors: like Email, FTP, HTTP, TCP, SOAP and others,
- data transformation services: XSLT, XQuery
- it handles many data formats, like Atom, CSV, HTML, XML, JSON and Java Objects,
- it provides cloud connectors to interact with Amazon, Twitter, Google, Facebook Whois,
- it is compatible to soma application servers like Tomcat, Jboss, Geronimo, Jetty and Websphere,
- it is compatible to many OS like Linux, Windows, Solaris,Mac OS,
- it can interact with many services types and standards like REST, WSDL, WS-Security
- it can manage processes like Bpel or jBPM.

Figure 28 - Mule ESB

As represented in FIGURE Mule ESB allows system integrators to easily interconnect many information systems, like proprietary applications, web services, cloud and social media

platforms, but also databases and files. It is possible to configure a message flow between these different systems in order to obtain a unique big integrated system.
The framework includes an editor, based on Eclipse, that allows users to graphically define flows between blocks and to deploy this flow that can be executed by the system. These blocks falls into these categories:

- **Endpoints**: that exchange messages to and from external applications. Endpoints can be inbound, like an http or FTP listener, or outbound such as SMTP endpoint to send email, or JDBC to insert some date to a database.
- **Scopes**: that combines block of different nature like components or messages so that they can be processed as a single unit by other Message Processors in the flow.
- **Components**: that are blocks that provide business logic services and functionalities, like logging, scripting components, web service components. For example the scripting Java component allows to select a java class and invoke it in the application flow.
- **Transformers**: that transform information data that goes through the flow in order to allows communication among different block. An example is transformer that receives a payload in an HTTP request and convert it into a JDBC format to store it in a database.
- **Filters**: that uses various criteria to select messages for further processing. For example a filter can ignore incomplete or invalid credential data submitted by an user before sending it to the authentication module of an application.
- **Flow Controls**: that allows to define criteria to route messages in the flow.
- **Error Handling**: that add exception strategies to normal flows.
- **Cloud Connectors**: that allows to communicate with some cloud platforms like CMS (CMIS), e-commerce applications (Magento), CRM and social networks (Twitter).

4.4 Tangible/Intangible assets repository

As stated in the description of the USDL Repository functional component, the repository has to support the management of service descriptions based on the USDL formalism or on an extension of its metamodel that allows the description of specific characteristics of tangible/intangible assets.

USDL went through several iterations since its conception in 2007. It was initially defined in an informal way and then formalized through an XML Schema. The current version of the complete USDL metamodel is USDL version 3.0 milestone M5 [23] which has been developed in the Eclipse Modeling Framework (EMF) [24]. The EMF representation has been discontinued and currently the FI-WARE project (together with the USDL community) is introducing Linked USDL[25], a new representation (based on the RDF data model), which keeps the original semantics and content of USDL version 3.0.

Figure 29 - USDL roadmap

As to the resources available for further development by the MSEE project, two options are under evaluation:

- reuse and customize the USDL models and tools (editor, marketplace) available as Open Source which are based on the EMF formulation. This option has the drawback that those resources are no longer maintained by the USDL community, which is now fully committed to the new Linked USDL version.
- adopt the new Linked USDL models and, if possible, reuse and customize tools from the FI-WARE project. The choice of Linked USDL would foster simplification and reuse of existing schemas on the basis of Linked Data principles and technologies. Although Linked USDL is sponsored by FI-WARE, the Linked USDL community has been established in order to foster an open development as well as a broad and world-wide adoption.

In the first case (EMF-based USDL) the following resources are available:

- **USDL Editor for Experts**: the editor is based on Eclipse development environment and realized as a Rich Client Platform project;
- **USDL Editor light for Novices**: web-based editor that is intended to be used by non-technical people. Right now this editor is work in progress. Not every USDL element is shown and only a few new elements can be created.
- **USDL Marketplace**: web-based application, still to be tested for its compliance with functionalities required for the second cycle of MSEE
- **USDL Model and XML Schema**: data model for the USDL editor. It can be used to manipulate USDL files, i.e., CRUD features of properties within the USDL document.

In the second case (Linked USDL), the Linked USDL community currently provides the following RDF vocabularies:

- **Linked-USDL Core**: "This vocabulary provides the core module of Linked USDL. It therefore covers the main concepts and relationships characterising services, leaving more specific aspects regarding some particular dimensions such as technical interfaces, licensing or security aside. These aspects will be covered in additional modules, or in some cases, like the technical interfaces, by existing vocabularies. This first draft includes most of the original USDL specification with some useful simplifications. The vocabulary is, however, likely to be reorganised and better modularised as we progress."

- **USDL Service Level Agreements**: "Simple vocabulary for describing service level agreements. This vocabulary is based on the work on USDL in the TEXO project. Service level module of USDL. Service Level Agreements (SLAs) are a common way to formally specify such functional and non-functional conditions under which services are or are to be delivered. However, SLAs in practice are specified at the top-level interface between a service provider and a service customer only. Customers and providers can use top-level SLAs to monitor whether the actual service delivery complies with the agreed SLA terms. In case of SLA violations, penalties or compensations can be directly derived. The USDL Service Level Module allows modeling such information. The module is derived from our research as part of a multilevel SLA management platform. This vocabulary defines a simple subset of the original USDL Service Level Module."
- **USDL Pricing Vocabulary**: Simple pricing model vocabulary for USDL
- **USDL-SEC Vocabulary**: very simple security model vocabulary for USDL

Moreover the FI-WARE is currently developing specifications and reference implementations for Generic Enablers (Repository, Marketplace, Editor, …) for the management of USDL service descriptions. Implementations are not open but the specifications are available to the public, thus we will consider the possibility of realizing ad-hoc and bare-bone implementation of such Generic Enablers.

4.5 Matching of the technological components with the functional components

Functional component	Technological component	Description
Ideation Process Management System		
Ideation Process Collaboration System	Gi2MO IdeaStream; Gi2MO IdeaStream Similarity	The combination of this two technological components cover all the functionalities required.
	Open ideaL	This technological component provides the basic functionalities required for the Ideation Process Collaboration System
Ideation Process Analysis System	Gi2MO IdeaStream Analytics	This technological component, in conjunction to Gi2MO Idea Stream, provides all the required functionalities
	Open ideaL	Open Ideal has a report page that contains report charts for ideation analysis.
Ideation Process Repository	Gi2MO IdeaStream	Gi2MO IdeaStream integrates the Drupal Knowledge Base in order to create a complete repository for the management of ideas, campaigns and of all the entities related to the ideation process
	Open ideaL	Open Ideal stores data in the Drupal Knowledge Base.
Operational/ecosystem-governance process management system		
Process Engine	ProcessMaker	Full support of the required features through a proprietary engine
	Aperte Workflow	Full support of the required functionalities through the inclusion of some well-known BPM implementations like jBPM and Activiti
	Bonita Open Solution	The Bonita Execution Engine fully supports the required features through a Java API that allows to interact programmatically with processes
	Spagic	Spagic includes execution engines such as jBPM and Apache ODE (BPEL Engine) for supporting both manual and automatic tasks
	Petals ESB	Petals ESB includes a BPEL service engine component (Petals-SE-BPEL).
	Mule ESB	Mule ESB contains an integrated JBPM engine that can interact with a Mule Flow.

Process Editor	ProcessMaker	The Process Map Designer is a web-based editor that allows the modelling of BPMN2.0 diagrams. Dynaform Builder allows business analysts to design custom forms for human tasks
	Aperte Workflow	The Aperte Modeler is a web-application that supports BPMN2.0. Liferay and Vaadin are used for the implementation of the user interfaces.
	Bonita Open Solution	Bonita Studio, an Eclipse RCP application, allows users to graphically design business processes in BPMN2.0 standard notation. The studio includes a graphical form designer in order to create forms for human tasks.
	Spagic	Spagic Studio, based on Eclipse IDE, provides tools for design, development and deploy of BPMN-based processes and of simple web-forms for human-oriented tasks
	Petals ESB	Petals Studio is a tool based on the Eclipse IDE that embed the BPEL Designer, a modified version of the official Eclipse BPEL Designer. No support for human tasks
	Mule ESB	Mule Studio offers a graphical designer that allows to edit process called Mule Flows. However these flows are not business processes, but mostly a way to orchestrate services.
Process Monitor	ProcessMaker	Process Maker has a dashboard that provides real time monitoring efficiency and productivity of processes and users executing those processes. But advanced dashboard functionalities are commercial.
	Aperte Workflow	Aperte Workflows provides a Liferay control panel function, called Aperte Workflow Process Instances, that allows administrators to track running and terminated process instances and their status. For each process instance it is also possible to show a process map that represents the process end tracks the current status.
	Bonita Open Solution	Bonita User Experience application allows to track process instances and monitor some indicators. It is available an advanced dashboard that can be customized, showing statistics and reporting.
	Spagic	Spagic Monitor is a web application for monitoring and managing the life-cycle of single

		services or processes. It is possible to track the state of a process instance, and also to show messages exchanged between services.
	Petals ESB	Petals View is a web application allowing to monitor business process instances and data exchanged between services. It is also able to generate KPIs to drive management reporting.
	Mule ESB	Mule ESB Management Console is a web-based interface to centrally monitor, manage, and administer the runtime aspects of the enterprise service bus. But this functionality is only available in the commercial Enterprise version.
Process Log Repository	ProcessMaker	ProcessMaker stores service instances information in its database and exposes web services for interacting with processes. One of these services, called CaseList(), allows to retrieve the list of process instances, with information about them.
	Aperte Workflow	AperteWorkflow is released in two versions, one using jBPM engine and one using Activiti Engine. Both engines log process data in a database table that can be accessed directly or using some APIs. Aperte Workflow doesn't provides a custom log repository manager.
	Bonita Open Solution	Bonita Open Solutions REST APIs allows to query process logging information present in its database.
	Spagic	Spagic runs business processes using the jBPM engine implementation. Logging information are available accessing jBPM database tables or using jBPM APIs.
	Petals ESB	No Process Log Repository
	Mule ESB	No Process Log Repository
Human-oriented tasks GUI	ProcessMaker	In ProcessMaker users can interact with processes using the Cases Inbox web page frame. This frame is similar to a mail inbox, where users can see new, draft or participated cases. Cases can be also filtered, sorted or searched. In the main page frame users can see a list of cases, and select one of them for handling the active task filling a form.
	Aperte Workflow	Aperte Workflow user front-end is integrated in Liferay. User interact with processes using four

		portlets: New Processes portlet that allows to start new processes, Task Queues portlet that allows to trace active tasks, Search portlet for searching tasks, and a task main portlet thet shows task forms that users have to fill.
	Bonita Open Solution	Bonita User Experience is a web module of Bonita Open Solution suite that provides user interface for interaction with processes. It has an interface similar to a webmail, with a area in which the user can see new tasks to handle.
	Spagic	Spagic TaskList is a web application provided by Spagic suite where users can manipulate forms defined for manual tasks.
	Petals ESB	No Human-oriented tasks GUI
	Mule ESB	No Human-oriented tasks GUI
Enterprise Service Bus	ProcessMaker	ProcessMaker doesn't provide ESB functionalities
	Aperte Workflow	Aperte Workflow doesn't provides ESB functionalities, but it is integrated with the Mule ESB.
	Bonita Open Solution	BES provides more than 100 connectors that provides ESB functionalities.
	Spagic	Spagic contains Connectors and Services. Connectors provide cooperation functions with external services, Services provide some functionalities like data transformation, role engine and scripting.
	Petals ESB	Petals ESB is an Enterprise Service Bus that supports several communication protocols like SOAP, Mail, FTP and standards like Web Services and XSLT.
	Mule ESB	Mule ESB is an Enterprise Service Bus that provides many transport connectors like email, FTP, SOAP; data transformation services like XSLT or XQuery; cloud connectors like Amazon or Google ones; and services interaction functionalities for interact with REST or WSDL services.
Business Intelligence tools	ProcessMaker	ProcessMaker Advanced Dashboards allows users or administrators to monitor KPIs, but this is a commercial functionality.
	Aperte Workflow	No Business Intelligence Tools.
	Bonita Open Solution	Bonita Open Solution provides functionalities for

		defining KPIs and populating them with easy to use graphical tools, but these functionalities are also in commercial Subscription Packs.
	Spagic	No Business Intelligence Tools
	Petals ESB	In Petals ESB it is possible to define KPI configurations with the Petals Studio, that acts as probes that collect and filter information from the bus and its services. The collected data are stored in a database and displayed by Petals View.
	Mule ESB	Mule ESB provides a module called Business Event Analyzer that provides a place to monitor business transactions and KPIs, but this module is only available in commercial version.
Tangible/intangible assets management		
USDL Repository	EMF-based USDL	There is no ready-to use repository for USDL models that are defined through the EMF data model. However EMF provides a set of tools that would help the development of the repository. Among those tools: CDO and Teneo
	LinkedUSDL	LinkedUSDL is not currently supported with a reference implementation of a repository. However RDF stores and generic Content Management Systems could be adapted to provide the required functionalities.
USDL Editor	EMF-based USDL	The EMF-based USDL version is supported by two editors: • **USDL Editor for Experts**: the editor is based on Eclipse development environment and realized as a Rich Client Platform project; • **USDL Editor light for Novices**: web-based editor that is intended to be used by non-technical people. Right now this editor is work in progress. Not every USDL element is shown and only a few new elements can be created.
	LinkedUSDL	LinkedUSDL does not have currently any publicly available editor for modelling USDL service descriptions.
Ecosystem front-end		
Web portal	Liferay	Liferay provides authentication, document

		management, content management and portal administration functionalities. MSEE Baseline modules can be realized as custom portlets and external applications can be integrated using iframe portlet.
	Drupal	Drupal provides authentication, document management, content management and portal administration functionalities. Web modules can be integrated in Drupal creating new Drupal Modules and Themes.

4.5.1 Functionalities coverage by technological products

The following tables aggregates tools by the categories corresponding to the IEP components: the tables map functionalities to technological products, indicating for each one the degree of coverage of the requirements using different colors.

- Green colour indicates that requirements are almost fully covered,
- Yellow colour indicates that requirements are only partially covered (functions are not present, they have commercial license etc),
- Red colour indicates that requirements are not covered.

	Ideation Process Collaboration System	**Ideation Process Analysis System**	**Ideation Process Repository**
G2MO Idea Management			
Open ideaL			

Table 38 - Ideation Process Management System

	Process Engine	**Process Editor**	**Process Monitor**	**Process Log Repository**	**Human-oriented tasks GUI**	**Enterprise Service Bus**	**Business Intelligence tools**
Process Maker							
Aperte Workflow							
Bonita Open Solution							
Spagic							
Petals ESB							
Mule ESB							

Table 39 - Operational/ecosystem-governance process management system

	USDL Repository	**USDL Editor**
EMF-based USDL		
Linked USDL		

Table 40 - Tangible/Intangible assets management

	Ecosystem Front-end
Liferay	
Drupal	

Table 41 - Ecosystem frontend

It is easy to see that several functionalities are not (or partially) covered by any analysed product: for this reason in the next development phase, in order to deliver a complete version of the Innovation Ecosystem Platform it will be necessary both integration of different existing products (among the previous listed or new) and custom implementation of new components from scratch.

5 Conclusion

This deliverable reports the current state of the architectural design and specification of the Innovation Ecosystem Platform, which is the technological environment that hosts and supports the processes and services of a Manufacturing Service Ecosystem and of the related Virtual Manufacturing Enterprises. Key processes that are supported by the IEP have been categorized as MSE Governance processes (for managing the ecosystem and its members), MSE Innovation (for fostering innovation actions among the members of the MSE) and VME Operations (for enacting and monitoring value-delivering processes that are conceived, designed and developed in the servitization process).

In order to enable those key processes, the IEP will use web-portal technologies to integrate the different functional components and to make them available to the final users through a single point of access. In particular: (i) innovation processes will be guided and managed through collaboration tools that allow the generation, development and analysis of ideas in the MSE community; (ii) MSE Governance processes will be supported by BPM tools and by instruments for managing USDL service descriptions enriched with tangible/intangible related information; (iii) VME Operations processes will be enacted and monitored through a workflow environment.

For each key process, the main use cases of the Innovation Ecosystem Platform have been defined, as well as the necessary functional components for their realization. Technological solutions have been studied and evaluated to guide the next phase of development of the IEP.

The IEP will be released in two cycles, the first is based on this specification and is due at month 12, the second will be based on D26.2 (M18) and is due at month 24. This first version of the IEP design builds on the MSEE use cases requirements specification collected until now in SP5 and on the conceptual frameworks and results of SP2. Given the on-going activities and partial results of the input deliverables that have been produced in such Sub Projects, some functional components of the IEP (e.g. BPM tools, workflow environment) have been defined with general-purpose characteristics, which will allow the platform to host and support any kind of service and process that will be actually needed in the next months. Feedback from final users and further requirements analysis will allow to define the final specifications and implement/refine the proper technological components.

References

[1] MSEE Deliverable D11.1 "Service concepts, models and method: Model Driven Service Engineering"
[2] http://www.coin-ip.eu/
[3] Sitek P, Gusmeroli S, Conte M, Jansson K, Karvonen I, The COIN Book - Enterprise Collaboration and Interoperability. Verlagsgruppe Mainz GmbH Aachen, 2011
[4] http://www.fi-ware.eu/
[5] http://ec.europa.eu/information_society/activities/foi/lead/fippp/index_en.htm
[6] MSEE Deliverable D52.1 "User Requirements Analysis for Virtual Factories & Enterprises in MSEE"
[7] MSEE Deliverable D21.1 "State of the Art report for requirements and innovation principles"
[8] http://www.bpmn.org/
[9] Cardoso, J., Barros, A., May, N., Kylau, U., "Towards a Unified Service Description Language for the Internet of Services: Requirements and First Developments" in IEEE International Conference on Services Computing 2010.
[10] Cardoso, J., Winkler, M., Voigt, K., "A Service Description Language for the Internet of Service"" in Published in: Alt, R., Fähnrich, K.-P., Franczyk, B. (Eds.), Proceedings First International Symposium on Services Science ISSS'09, Logos, Berlin, 2009.
[11] http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/FIWARE.ArchitectureDescription.Apps.Marketplace
[12] MSEE Deliverable D41.1 "MSEE Service-System Functional and Modular Architecture"
[13] http://www.liferay.com/
[14] http://drupal.org/
[15] http://www.gi2mo.org/apps/ideastream
[16] http://www.linnovate.net/ideal
[17] http://www.processmaker.com
[18] http://www.aperteworkflow.org
[19] http://www.bonitasoft.com
[20] http://www.spagic.org/
[21] http://petals.ow2.org
[22] http://www.mulesoft.org
[23] http://www.internet-of-services.com/fileadmin/IOS/user_upload/pdf/USDL-3.0-M5-Archive.zip
[24] http://www.eclipse.org/modeling/emf/
[25] http://www.linked-usdl.org/