

Horizon 2020 – The EU Framework Programme for Research and Innovation  
Project Co-funded by the European Commission  
Contract number: 761145  
Call identifier: NMBP-22-2017  
Project Start Date: 1<sup>st</sup> January 2018



MANUSQUARE

MANUFACTURING ecoSYSTEM of QUALIFIED RESOURCES  
EXCHANGE

---

D6.1  
Deployment report

---

Dissemination Level	Confidential
Partners	HX, SUPSI, IBM, SINTEF, INESC
Authors	Caterina Calefato, Konstantinos Grevenitis, Alessio Lanini, Giuseppe Landolfi, Elias Montini, Radostin Tsvetanov, Audun Vennesland, Mario Amorim Lopes
Planned date of delivery	M42 – June 2021
Date of issue	25 <sup>th</sup> June 2021
Document version	Final 1.5



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 761145

**DISCLAIMER:**

The herewith information reflects only the author's view. The European Commission is not responsible for any use that may be made of the information herewith included.

## DOCUMENT HISTORY

Version	Issue date	Content and changes	Author
0.1	14.04.2021	Table of contents, introduction and methodology	Caterina Calefato (HX)
0.2	28.04.2021	Introduction	Caterina Calefato (HX)
0.3	07.05.2021	Services Deployment paragraph	Marko Vujasinovic (INNOVA); Mário Amorim Lopes (INESC); Audun Vennesland (SINTEF); Benjamin Mandler (IBM); Giuseppe Landolfi (SUPSI); Elias Montini (SUPSI); Radostrin Tsvetanov (SUPSI)
0.4	24.05.2021	First draft	Caterina Calefato (HX)
0.5	01.06.2021	Added § 4.3 and 4.6	Radostin Tsvetanov (SUPSI)
0.6	04.06.2021	Added § 4.7	Giuseppe Landolfi (SUPSI)
0.7	04.06.2021	Reviewed § 7	Giuseppe Landolfi (SUPSI)
0.8	07.06.2021	Added § 3.2.4	Elias Montini (SUPSI)
0.9	07.06.2021	Reviewed § 6	Giuseppe Landolfi (SUPSI)
1.0	08.06.2021	Complete draft	Caterina Calefato (HX)
1.1	14.06.2021	Updated document after first peer review	Caterina Calefato (HX)
1.2	16.06.2021	Finalization	Caterina Calefato (HX), Giuseppe Landolfi (SUPSI)
1.3	18.06.2021	Quality assurance	Andrea Bettoni (SUPSI)
1.4	22.06.2021	Table 1 with deployment history added	Alessio Laini (HX)
1.5	25.06.2021	Final check	Andrea Bettoni (SUPSI)

Role	Partner	Person
Reviewer 1	INNOVA	Linh Thai
Reviewer 2	IBM	Benjamin Mandler
Quality assurance	SUPSI	Andrea Bettoni

## TABLE OF CONTENTS

Document history .....	2
Table of contents.....	3
Figure List .....	4
Table list.....	4
1 Executive summary .....	5
2 Introduction.....	6
2.1 Aim and scope of Task 6.1 .....	6
2.2 Relationships of T6.1 with other tasks .....	6
2.3 Outline .....	6
3 Platform Deployment and Integration .....	7
3.1 Management of unexpected events.....	7
3.2 Overview of the deployment strategy .....	7
3.3 Deployment .....	8
3.3.1 Deployment stages.....	8
3.3.2 Platform Installation.....	8
3.3.3 Outcomes of the deployment .....	11
3.3.4 Testing, Quality Control and Issue Solved .....	11
4 Services Deployment.....	13
4.1 Semantic Infrastructure.....	13
4.2 Blockchain service.....	13
4.3 Distributed data sharing management.....	14
4.4 Platform interface and expansion SDK.....	14
4.5 Matchmaking service .....	14
4.6 Reputation mechanism.....	15
4.7 Sustainability assessment .....	15
4.8 Unified flow manager for ecosystem orchestration .....	15
4.9 Open innovation idea management tool .....	16
5 Platform Installation Guide .....	17
5.1 System requirements.....	17
5.2 Installing .....	17
5.3 Upgrading.....	17
5.4 Backup.....	17
6 Continuous Deployment .....	18
7 Conclusions.....	19

## FIGURE LIST

Figure 1 The Tweet the OVH company wrote to communicate the fire incident .....	7
Figure 2 Container Registry .....	9
Figure 3 Bugs & Fix tracking document .....	12
Figure 4 Relationship between deployment and adoption activities.....	18

## TABLE LIST

Table 1 Deployment history .....	11
Table 2 Deployment and configuration needs.....	13
Table 3 Semantic Infrastructure - deployment and configuration needs .....	13
Table 4 Blockchain service - deployment and configuration needs.....	14
Table 5 Distributed data sharing management - deployment and configuration needs .....	14
Table 6 Platform interface and expansion SDK - deployment and configuration needs.....	14
Table 7 Matchmaking service - deployment and configuration needs.....	14
Table 8 Reputation Mechanism - deployment and configuration needs.....	15
Table 9 Sustainability assessment - deployment and configuration needs .....	15
Table 10 Unified flow manager for ecosystem orchestration - deployment and configuration needs.....	15
Table 11 Open innovation idea management tool - deployment and configuration needs.....	16

## 1 EXECUTIVE SUMMARY

This deliverable is the last technical report within the Work Package 6 and it corresponds to *Task 6.1 MANU-SQUARE platform customization and deployment*. It summarizes the configuration and deployment activities of the platform as a whole and of the tools developed in the technical WPs (WP2-WP3-WP4).

T6.1 aim was to organize deployment activities among technical partners, in order to carry out the tasks required to manage the platform and its evolution.

This process included issues regarding rolling out new versions of the platform and tools, bug fixing, code evolution, and service support. During the platform operation, some needs emerged and required to integrate them in newer releases of the platform (for details, please ref. to § 3.3.4).

In the services deployment section, service owners have described the deployment and integration of their services. Each service includes a brief description of the deployment from a technical point of view, a description of the customization process (e.g., specific system configurations), a description of the operational tasks performed and a list of the major issues solved, if any.

This report consists of the following chapters:

- § 2 introduces the aim and scope of task 6.1, the relationships with the other tasks and the document's outline.
- § 3 describes the phases of the MANU-SQUARE platform deployment and integration.
- § 4 describes the services deployment (ref. to WP2, WP3, WP4)
- § 5 provides an installation guide for the MANU-SQUARE platform, specifying also the hardware resources needed.
- § 6 describes the MANU-SQUARE continuous deployment activity
- § 7 concludes the report.

Although the conclusions chapter presents some lesson learnt, the evaluation of the platform and its services is documented in *D6.6 Final evaluation of results*.

## 2 INTRODUCTION

### 2.1 Aim and scope of Task 6.1

The aims of D.6.1 within WP6 - Demonstration pilots in the machine tools and silk sectors are to:

- describe the MANUSQUARE platform deployment and operation;
- describe the MANUSQUARE services deployment;
- provide a platform installation guide.

According to the envisaged project architecture, the MANU-SQUARE Marketplace Platform (MMP) acts as the core managing system of the whole solution and, with its components provides functions at three different levels: data management (WP2-WP3), service provision (WP4) and integration (WP6).

### 2.2 Relationships of T6.1 with other tasks

T6.1 collected and integrated the technical work from the IT point of view in the MANU-SQUARE project, hence T6.1 encompasses the description of deployment activities of all the technical WPs, namely:

- WP2 – Ecosystem ontological representation
- WP3 – Blockchain-based platform infrastructure
- WP4 – Service – providing tools for platform expansibility

### 2.3 Outline

This report is organized into 6 chapters. The first chapter, § 1, is an executive summary. § 2 provides introduction, explains the aim and scope of the report, relationships with other tasks. § 3 describes the MANU-SQUARE platform deployment integration. § 4 provides the description of the services deployment. § 5 provides an installation guide for MANU-SQUARE platform. § 6 describes the continuous deployment activity. The final chapter, § 7, concludes the report.

### 3 PLATFORM DEPLOYMENT AND INTEGRATION

This section provides an overview of the deployment and integration process carried out during the last phase of the project to guarantee, on the one hand, continuous integration of the platform services as soon as new versions were released and, on the other hand, a stable service level.

To ensure this two primary goals, a task force has been established including developers from all IT partners (HX, SUPSI, IBM, INESC-TEC, INNOVA, SINTEF). The objective of this task force was focused on delivering a satisfactory service level for the end-users and working towards the enhancement of the platform through the fixing and managing new feature requests, while keeping the platform operational. By including all partners in the task force under the guidance of the HX, it was possible to speed up operations in a continuous evolutionary process. Involved partners kept almost daily communication active in order to smoothly made available new services and maintain the platform.

#### 3.1 Management of unexpected events

The deployment task force has been working on a critical and totally unexpected event: the fire incident at the data centre hosting the project platform, the OVH data centre in Strasbourg<sup>1</sup>.



Figure 1 The Tweet the OVH company wrote to communicate the fire incident

Upon instruction from the Coordinator, the Dissemination Manager redirected the call-to-actions on the project website to a temporary web page to inform users that the service was unavailable. Luckily, the servers hosting MANU-SQUARE development and production were not damaged, hence towards the end of March 2021 the MANU-SQUARE platform was restarted, without information loss. This notwithstanding, an action plan had already been put in place to tackle the case in which the service could not be restarted by the cloud provider. That relied on a series of backed up data to restore the platform to its original content and user profiling.

#### 3.2 Overview of the deployment strategy

Currently the platform and services are deployed in two physical servers located in OVH data centres. The domains of the MANU-SQUARE servers are:

- **Development server:** manusquaredev.holonix.biz
- **Production server:** platform.manusquare.eu

The project opted for having two different servers (development and production). The development server allows preliminary development, deployment and testing activities, while the production server hosts the finalized services. New versions of services are first deployed to the development environment and only after being successfully tested they are

<sup>1</sup> <https://twitter.com/olesovhcom/status/1369478732247932929>

promoted to the production environment. The development environment contains test data while the production environment is kept clean.

### 3.3 Deployment

The main goal of the deployment activities was to make the platform and associated services easy to adopt and even to replicate. Focusing on that, the technical tasks associated with the deployment process of the functional platform were reduced to a few steps in order to simplify the future maintenance and adoption by the partners who are interested in the platform exploitation after the project ends. The reduction of the tasks has been carried out thanks to the selected technical tools and the designed platform architecture. Architectural decisions at the platform level, such as adopting a microservices approach and making use of docker containers has transformed the deployment process into a simplified procedure.

#### 3.3.1 Deployment stages

The deployment process is broken down into four main steps. This is an iterative process that was performed since the very first release of the platform and for the further releases. The main steps are:

1. **Configuration of the host virtual machine:** in order to install the platform, a host VM has to be set up.
2. **Platform installation:** referred to the installation and configuration of the MANU-SQUARE platform.
3. **User management:** this step consists of different administrator tasks such as users' registration and server account configuration.
4. **Deployment of services:** this stage includes tasks such as the uploading of new services to the MANU-SQUARE platform, according to the scheduling of each WP.

#### 3.3.2 Platform Installation

Since the whole system is based on docker containers, its installation is simplified by the use of scripts. (Refer to § 5 Platform Installation Guide for a detailed explanation of the installation procedure of the platform). In addition, description files of the deployment for Docker (Docker compose) and snapshots of the development are stored in a Gitlab registry. The MANU-SQUARE's Gitlab repository is: <https://gitlab.com/manusquare>

Figure 2 shows the container registry.

## Container Registry

10 Image repositories

With the GitLab Container Registry, every project can have its own space to store images. [More information](#)

Filter results	Q	Updated	↓
manusquare/ufm4eo/ Root image			
11 Tags			
manusquare/fileserver/ Root image			
7 Tags			
manusquare/reputation-manager/ Root image			
8 Tags			
manusquare/manusquare.semanticmatching.api/ Root image			
12 Tags			
manusquare/semantic-infrastructure/matchmaking_api			
1 Tag			
manusquare/idea-manager/ Root image			
20 Tags			
manusquare/backup-manager/ Root image			
1 Tag			
manusquare/gui/ Root image			
64 Tags			
manusquare/semantic-infrastructure/ Root image			
19 Tags			
manusquare/edm/ Root image			
40 Tags			

Figure 2 Container Registry

For details, please refer to the development docker compose link: <https://gitlab.com/manusquare/manusquare-core/-/blob/master/docker-compose-dev.yml>

The production docker compose is available here: <https://gitlab.com/manusquare/manusquare-core/-/blob/master/docker-compose-prod.yml>

Access to this registry is managed by:

- SUPSI, for the development server;
- HX, for both development and production servers.

To get access to the GitLab repository, please write to coordinator Andrea Bettoni: [andrea.bettoni@supsi.ch](mailto:andrea.bettoni@supsi.ch).

## D6.1 – Deployment report

The development platform was installed for the first time in April 2020. This version was accessible to the IT partners only, and not by the general public. It has been upgraded several times thanks to the continuous improvement and bug fixing tasks performed during the project lifetime.

The first integration of the available services (according to WP2-WP3-WP4 scheduling) on the production server was released in May 2020. When the piloting phase in WP6 started in M25, the production platform was iteratively frozen and new releases for integration and testing were moved to the development platform. The production server was updated once new stable versions of the platform were ready. The production platform was upgraded with the last new release in May 2021 to include a series of improvements and patches identified during the adoption phase. Some further refinements on the look and feel of the GUI were released in June 2021.

The full history of the deployment process is described in Table 1.

Date	Services released
May 2020	Edm(MongoDb, Swagger Api):1.3.0, Semantic-infrastructure:1.0, Manusquare.semanticmatching.api:.1, Manusquare-gui:0.4.2, Backup-manager:1.0, Idea-manager (MySQL) :0.6.6, Blockchain
Oct 2020	Edm(MongoDb, Swagger Api):1.7.1, Semantic-infrastructure:1.5, Manusquare.semanticmatching.api:0.5 Manusquare-gui:0.5.0, Reputation-manager:0.3.1, Idea-manager (MySQL) :0.6.8, Fileserver:0.3, Nginx, Backup-manager:1.0, Blockchain
Nov 2020	Edm(MongoDb, Swagger Api):1.7.1, Semantic-infrastructure:1.5, Manusquare.semanticmatching.api:0.5 Manusquare-gui:0.5.5, Reputation-manager:0.3.1, Idea-manager (MySQL) :0.6.9, Fileserver:0.3, Manusquare-ufm4eo, Manusquare-wiki:(Mariadb), Nginx, Portainer-ce, Backup-manager:1.0, Blockchain
Dec 2020	Edm(MongoDb, Swagger Api):1.7.1, Semantic-infrastructure:1.5, Manusquare.semanticmatching.api:0.5 Manusquare-gui:0.5.5, Reputation-manager:0.3.1, Idea-manager (MySQL) :0.7.1, Fileserver 0.3, Manusquare-ufm4eo, Manusquare-wiki (Mariadb), Nginx, Portainer-ce, Backup-manager:1.0, Blockchain
Feb 2021	Edm(MongoDb, Swagger Api):1.7.1, Semantic-infrastructure:1.5, Manusquare.semanticmatching.api:0.5

## D6.1 – Deployment report

	Manusquare-gui:0.7.7, Reputation-manager:0.3.1, Idea-manager (MySQL):0.7.1, Fileserver:0.3, Manusquare-ufm4eo, Manusquare-wiki (Mariadb), Nginx, Portainer-ce, Backup-manager:1.0, Blockchain
Mar 2021	Edm (MongoDb, Swagger Api):1.7.1, Semantic-infrastructure:1.5, Manusquare.semanticmatching.api:0.5 Manusquare-gui:0.8.0, Reputation-manager:0.5, Idea-manager (MySQL):0.7.1, Fileserver:0.3, Manusquare-ufm4eo, Manusquare-wiki (Mariadb), Nginx, Portainer-ce, Backup-manager:1.0, Blockchain
May 2021	Edm (MongoDb, Swagger Api):1.7.1, Semantic-infrastructure:1.8, Manusquare.semanticmatching.api:1.3 Manusquare-gui:0.8.3, Reputation-manager:0.5, Idea-manager (MySQL):0.7.1, Fileserver:0.3, Manusquare-ufm4eo, Manusquare-wiki (Mariadb), Nginx, Portainer-ce, Backup-manager:1.0, Blockchain

Table 1 Deployment history

We adopted an iterative deployment process for each release. This process was characterized by a first deployment on the development server. After a testing phase, the release was moved to the production server.

The latest version of the MANU-SQUARE platform in the production environment can be accessed via the following link: <https://platform.manusquare.eu/>

### 3.3.3 Outcomes of the deployment

The main features of the production server are:

- a production environment composed by 16 containers with the various components of the platform among the different services.
- Portainer Service (<https://www.portainer.io/>): it is an open-source tool for managing containerized applications. It works with Kubernetes, Docker, Docker Swarm, Azure ACI in both data centres and at the edge. Portainer removes the complexity associated with orchestrators so anyone can manage containers. It can be used to deploy and manage applications, observe the behaviour of containers and provide the security and governance necessary to deploy containers widely. It was introduced to remotely handle the docker instances.

### 3.3.4 Testing, Quality Control and Issue Solved

Tests were carried out firstly by the development team coordinated by SUPSI and supported by HX, INNOVA, SINTEF, INESC and IBM. Then, end-users inside the consortium have been involved. Relevant support has been provided by JPM,

## D6.1 – Deployment report

P-TECH and CSEM. More than 300 fixes and improvements have been identified and collected into the bug tracker document, depicted in Figure 3 which supported bugs and improvements tracking, but also planning and execution of the development activities.

ID	Type	Where	Who	Bug/Improvement	Priority	In the filed	Comments
300	GUI	Innovation capabilities	Elias	Adding a skill which is not between suggestion and changing from one interface to IM capabilities, skills table appears cutted with a Java error.	4	Solved	
301	GUI	Project	Elias	Alignment and spaces between the table and the drop down menu field. It should be as much possible similar to resources	2	Solved	
302	GUI	RFP	Davide	I add problem solving as skills in the Inno capabilities. However it is not suggested in the skills suggestion in RFP creation	3	Solved	
303	GUI	RFP	Rado	Info "Select how you want to choose supplier candidates": to remove RFQ and add RFP		Solved	
304	GUI	Profile company		Please remove from company's attribute "Region" and add "Revenue [€]"	3	Solved	
305	GUI	Profile company		Add as company's attribute: "Organisation type" with values as drop down list (Research & Technology Organization; Large company; SME cluster; Association; Small Medium Enterprise; Start-up) and Ownership with values as dropdown list (Private; Public)	1	Ongoing	
306	GUI	1 proposal creation	Davide	Request documentation -> Request's attached documentation	2	Solved	
307	GUI	1 proposal creation and every time we have a file	Elias	Add an icon before the file name.	3	Solved	
308	GUI	Innovation Capabilities	Elias	Reduce spaces and distances between tables and fields	3	Solved	
309	GUI	1 proposal selection	Elias	Remove action column	3	Solved	
310	GUI	Profile company	Rado	Alignment Company's certifications list e company's attribtues	2	Solved	
311	GUI	General	Elias	The widgets that show project name etc.. should be text widgets and not input widgets	1	Ongoing	
312	GUI	Final proposal creation	Elias	Milenstones cannot be after the project end.	4	Solved	
313	GUI	2nd round proposal selection	Davide	Custom NDA link is not in the expected font.	2	Solved	
314	GUI	Innovation Manager details	Elias	Fix spaces and distances. Increase size of past experiences. Align right border of past experiences.	2	Solved	
315	GUI	Innovation managers selection for 1° round proposal	Elias	Values of matching score in % and with right-alignment. Reputation with right alignment.	3	Solved	
316	GUI	First proposal candidates selection	Elias	If no candidates are selected, it is still possible to continue clicking on next	4	Solved	

Figure 3 Bugs & Fix tracking document

The Bugs and Fix tracking file is available as a shared document here: [https://docs.google.com/spreadsheets/d/1knRpkOOH1Zaq3T\\_pKiwZe4k04Gd6ahJViR\\_V92NoBk/edit#gid=397433977](https://docs.google.com/spreadsheets/d/1knRpkOOH1Zaq3T_pKiwZe4k04Gd6ahJViR_V92NoBk/edit#gid=397433977)

Continuous updates were in place mainly focused on critical issues identified in the production environment by both internal and external users. A specific procedure was in place to address these issues:

1. a user identifies and communicates an issue/bug;
2. the user is notified that the issue is going to be analysed and the updates about its evolution will be provided;
3. the SUPSI team carries out a preliminary analysis, investigating the issue/bug and trying to replicate and classify it;
4. the SUPSI team delivers the collected information to the responsible team(s), depending on the type of issue/bug identified after the preliminary analysis, involving the needed developers according to the MANU-SQUARE components;
5. the involved team(s) evaluates the required effort to solve the issue/bug and its relevance;
6. if the relevance is high and the effort is low, the issue is given the highest priority;
7. the SUPSI team is notified that the issue has been solved, and communicates its resolution to the user;
8. in case the issue has not been solved due to lack of information, a notification is provided in any case to the user.

## 4 SERVICES DEPLOYMENT

The goal of T6.1 was to configure and deploy the MANU-SQUARE platform and tools to make them accessible by the end-users during the validation activities. It focused on the operative activities of preparing the environment where the services are deployed, identifying in advance, with the support of the end users, any barriers and requirements that could interfere with the deployment. In the following sections, we will describe the configuration and customization features for each service. Developer partners have been asked to fill Table 2 in order to clearly specify the digital datasheets of their components for the deployment purpose.

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM
CPU
Other

  

CUSTOMIZATION

Table 2 Deployment and configuration needs

Deployment needs are defined in the “Minimum hardware requirements for the deployment” section of the table, while configuration needs are defined in the “Customization” section of the table.

### 4.1 Semantic Infrastructure

The following table summarizes the deployment of this service. The semantic infrastructure can be deployed as a docker container. The docker image is available at [registry.gitlab.com/manusquare/semantic-infrastructure](https://registry.gitlab.com/manusquare/semantic-infrastructure) by means of docker build commands.

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM – 2.0 GB RAM
CPU – 1
Other: Min 2 GB of disk space Java SE Development Kit 8 Min Java heap size 512MB No firewalls policy Runs on port 8080

  

CUSTOMIZATION
At the moment, the source code is not publicly available, but access to the GitLab project and code repository can be granted when requested (by emailing such request to Innova Srl).

Table 3 Semantic Infrastructure - deployment and configuration needs

### 4.2 Blockchain service

The following table summarizes the deployment of this service. The blockchain service can be deployed as a docker container. The docker images are available at: <https://gitlab.com/manusquare/blockchain>. Blockchain support components run as docker containers. The code repository can be found in: <https://gitlab.com/manusquare/blockchain>.

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
2.5 GB RAM to enable scaling; can be reduced for small installations.
CPU - 1
Other: none.

  

CUSTOMIZATION

n/a
-----

Table 4 Blockchain service - deployment and configuration needs

### 4.3 Distributed data sharing management

The following table summarizes the deployment of this service. The Ecosystem Data Manager (EDM) can be deployed as a docker container. The docker images are available at: <https://gitlab.com/manusquare/edm>

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
4 GB RAM
CPU – i3
Other: none.

  

CUSTOMIZATION
API adaptation for data collection

Table 5 Distributed data sharing management - deployment and configuration needs

### 4.4 Platform interface and expansion SDK

The following table summarizes the deployment of this service. The platform interface and expansion SDK can be deployed as a docker container. The docker image is available at: [registry.gitlab.com/manusquare/gui](https://registry.gitlab.com/manusquare/gui) by means of docker build commands. Source code is available at: <https://gitlab.com/manusquare/gui>

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM – 2.0 GB RAM
CPU – 1
Other: Jetty 9.4 No firewall policy Runs on port 8080

  

CUSTOMIZATION
n/a

Table 6 Platform interface and expansion SDK - deployment and configuration needs

### 4.5 Matchmaking service

The following table summarizes the deployment of this service. The matchmaking service can be deployed as a docker container. The docker image is available at:

[https://gitlab.com/manusquare/manusquare.semanticmatching.api/container\\_registry/1041443](https://gitlab.com/manusquare/manusquare.semanticmatching.api/container_registry/1041443)

Source code for the Matchmaking Service is currently publicly available from: <https://github.com/PeterHaro/manusquare>

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM – 2GB
CPU – 1
Other: OpenJDK:8-jre-alpine Recommended Java heap size 1024MB Min 2GB of disk space No firewall policy Runs on port 7000

  

CUSTOMIZATION
n/a

Table 7 Matchmaking service - deployment and configuration needs

#### 4.6 Reputation mechanism

The following table summarizes the deployment of this service. The reputation mechanism can be deployed as a docker container. The docker image/other is available at: [registry.gitlab.com/manusquare/reputation-manager:latest](https://registry.gitlab.com/manusquare/reputation-manager:latest) by means of docker build commands.

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
2GB RAM
1 vCPU (x86)
Other: none.
CUSTOMIZATION
n/a

Table 8 Reputation Mechanism - deployment and configuration needs

#### 4.7 Sustainability assessment

The following table summarizes the deployment of this service. The sustainability assessment service is a separate jar library.

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM – 2GB
CPU - 1
Other: this library has to be added as a dependency library which provides functionalities used by the by-product sharing service.
CUSTOMIZATION
Uses a set of predefined processes. Can be extended with others by importing them as .xml from Ecoinvent V2

Table 9 Sustainability assessment - deployment and configuration needs

#### 4.8 Unified flow manager for ecosystem orchestration

The following table summarizes the deployment of this service. The Unified flow manager for ecosystem orchestration service can be deployed as a docker container. The docker image is available at: [https://gitlab.com/manusquare/ufm4eo/container\\_registry/1398070](https://gitlab.com/manusquare/ufm4eo/container_registry/1398070).

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM – 2GB
CPU - 1
Other: Spring embedded application running on Tomcat 8.5.43 No firewall policy Runs on port 8080
CUSTOMIZATION
Source code is available at: <a href="https://gitlab.com/manusquare/ufm4eo">https://gitlab.com/manusquare/ufm4eo</a>
It uses the IBM CPLEX Optimization Java Library available at: <a href="https://www.ibm.com/analytics/cplex-optimizer">https://www.ibm.com/analytics/cplex-optimizer</a> . For research purposes, this library has been obtained under academic license. This implies that a subscription to the CPLEX library will be needed to use such a service for commercial purposes. In order to successfully run the UFM4EO service, it is important to specify the path folder where CPLEX native libraries are located: java -jar ufm4eo-1.0.jar -Djava.library.path=/Applications/CPLEX_Studio1210/cplex/bin/x86-64_osx
More details about the UFM4EO and its setup have been widely described at Deliverable D4.4.

Table 10 Unified flow manager for ecosystem orchestration - deployment and configuration needs

#### 4.9 Open innovation idea management tool

The following table summarizes the deployment of this service. The Idea Manager tool can be deployed as a docker container. The docker image is available at: <https://gitlab.com/manusquare/idea-manager>.

MINIMUM HARDWARE REQUIREMENTS FOR THE DEPLOYMENT
RAM: 4 GB
CPU: Intel Core Processor
Other: none.

  

CUSTOMIZATION
It is advisable to install SSL certificate. The tool can run also in restricted environments where the certification is not needed. This is the reason why the SSL certificate is not mandatory

Table 11 Open innovation idea management tool - deployment and configuration needs

## 5 PLATFORM INSTALLATION GUIDE

### 5.1 System requirements

The advised system requirements are:

- Linux platform Redhat
- Hard disk space: 60 GB
- RAM: 16 GB

### 5.2 Installing

At first, Docker ([www.docker.com](http://www.docker.com)) has to be installed, then Docker Compose, too. At this point, SSL required certificates have to be installed before proceeding with the configuration of the Docker Compose file. Then it is possible to start the containers of the MANU-SQUARE services by the Docker Compose file.

### 5.3 Upgrading

To implement the upgrades of the MANU-SQUARE services, it is necessary to upgrade the Docker Compose file with the new released versions and restart the related script.

### 5.4 Backup

MANU-SQUARE already provides an internal service for the backup of Idea Manager data, EDM data, Semantic infrastructure database, and the WIKI data and WIKI data DB. This service does not have a user interface.

MANU-SQUARE infrastructure is compliant also with any advanced third-party backup services.

## 6 CONTINUOUS DEPLOYMENT

As duly referred in *D6.7 Platform final integration*, continuous improvements have been carried out since the very beginning of the development activities, targeting both bugs fixing, new features and usability improvement. This activity has been implemented through several tests of the MANU-SQUARE platform involving different users. According to the development roadmap, testing activities involved first the basic features of the platform including registration, profiling and company characterisation. Then, all the features related to the Capacity Sharing service, such as those related to resource characterisation and the whole business process, including testing of all the integrated tools (e.g. matchmaking, reputation mechanism, etc.), have been tested. Finally, in the same way, Innovation Management and By-Product Sharing services have been tested.

Following each test and thanks to the CD/CI (Continuous Deployment/Continuous Integration) provided by Gitlab that offers specific configuration modes, it was possible to launch an automatic deployment every time a new version of a component was available.

Related to the continuous deployment activity, Figure 4 gives a clear overview of the multiple and interconnected tasks that have been managed from M25 to the end of the project. Of course, for development and testing reasons, some tasks never ended: such as deployment on DEV server, monitoring and management of the overall development and deployment activities.

The chart shows how the deployment of the services on the Production environment and the adoption phase was managed in a consistent way, along the second half of the project. Development (on the top) and adoption (on the bottom) activities are highlighted in blue. Whenever a service was available, the related adoption activities were started.

Monitoring and management activities, highlighted in green, followed the whole duration of the deployment process.

TASKS	2020												2021					
	January	February	March	April	May	June	July	August	Sept	Octob	Novem	Decemb	January	February	March	April	May	June
	M25	M26	M27	M28	M29	M30	M31	M32	M33	M34	M35	M36	M37	M38	M39	M40	M41	M42
Deployment DEV server																		
Deployment PROD server (capacity sharing)																		
Deployment PROD server (innovation mng)																		
Deployment PROD server (by-product sharing)																		
Deployment PROD server (idea-manager)																		
Adoption (capacity sharing)																		
Adoption (innovation mng)																		
Adoption (by-product sharing)																		
Adoption (idea-manager)																		
Monitoring																		
Management																		

Figure 4 Relationship between deployment and adoption activities.

## 7 CONCLUSIONS

Over the last 18 months of the project, the MANU-SQUARE platform and its services were continuously deployed and improved, according to the monitoring feedbacks assessing their performance. The platform performance has been analysed from different perspectives including the technical point of view but also the business feedback from direct users (for the whole report, please refer to deliverable *D6.6 Final evaluation of results* that presents the overall evaluation as outcome of the demonstration phase, carried out in *T6.5 Evaluation and improvement of results*).

Some lessons learnt were produced as a result of the monitoring process undertaken during this period and can be summarised in the following aspects:

1. **Deployment of complex platforms requires an iterative approach.** MANU-SQUARE deployment strategy followed an iterative approach: instead of releasing new versions of the whole platform we decided to release services one by one, but taking into account dependencies. This means that sometimes the deployment of a new release of a service implied the deployment of connected services. In this way, we are able to start collecting feedbacks regarding the use of the released services, and in parallel, to work on the development of others or on the implementation of the feedbacks gathered iteratively. ***This represents a positive strategy that will be applied also in future projects.***
2. **Relying on dockerized packages helps in case a rollback is needed.** The continuous integration work relied on tagged docker versions. In this way we were able to perform rollbacks when needed. At every push action on Gitlab we automatically had a new image ready to be deployed. ***This represents a positive strategy that will be applied also in future projects.***
3. **Defining clear decoupling points between tools is key for integration.** The separate management of design and development of tools (WP4) and of their integration (WP6) allowed independent deployments. More in details, we were able to split the core development of each component from its interactions with the other components, by means of quite stable integration interfaces. The integration interfaces define how each component interacts with the others. ***This represents a positive strategy that will be applied also in future projects.***
4. **Keeping separated environments for each activity makes things easier.** We had two different deployment environments: one for development and testing activities (named “DEV” environment) and one for end-user adoption and promotion toward the general public (named “PROD” environment). Every time we did training, and demo activities, we worked in the PROD environment, considering the better stability of the platform. Nevertheless, this type of activities (i.e., testing, training, and demo) often needs the entering of dummy data that we needed to manually remove afterwards. This was an uncomfortable practice. ***This represents a strategy to be improved in future projects.*** Next time the best choice might be to have three different environments: 1) DEV (development): accessible only to IT partners, totally dedicated to development and testing activities, mainly working on dummy data; 2) PRE (pre-production): this environment will be used for training and demo activities, periodically synchronizing it with the PROD; 3) PROD (production): is the latest stable and debugged version of the whole platform, to be used just for piloting and adoption activities. No dummy data can be entered in the PROD version. In this way there is no need to clean the PROD environment from dummy data.
5. **Tagging released code should be a standard to ensure quick identification of versions.** The tagging mechanism for the deployed versions was adopted in MANU-SQUARE toward the end of the development work, in an experimental way, only for the GUI component. We assessed that it is very useful to distinguish the deployed version and to activate specific pipelines according to the tags. ***This represents a strategy to be improved in future projects.*** Next time the best choice in a project will be to adopt from the beginning of the development activity the tagging of the deployed versions. This type of indication shall be delivered to all the IT partners once the project Gitlab repository is available.