D6.1 Analysis of user applications, collection of requirements

Deliverable ID	D 6.1- Analysis of user applications, collection of requirements
Work Package Reference	WP6
Issue	1.0
Due Date of Deliverable	28/02/2023
Submission Date	28/02/2023
Dissemination Level[1]	PU
Lead Partner	KIT
Contributors	CSIC, KIT, IISAS, PREDICTIA, PSNC, MicroStep-MIS, WODR
Grant Agreement No	**101058593**
Call ID	**HORIZON-INFRA-2021-EOSC-01**

Funded by
the European Union

[1] **PU** = Public, **PP** = Restricted to other programme participants (including the Commission Services),
RE = Restricted to a group specified by the consortium (including the Commission Services),
CO = Confidential, only for members of the consortium (including the Commission Services)

Prepared by	Reviewed by	Approved by
Lisana Berberi, Ignacio Heredia, Valentin Kozlov, Elena Vollmer, Judith Sainz-Pardo, Juraj Bartok, Adam Fojud, Michal Blaszczak, Agnieszka Rausch, Khadijeh Fahimeh Alibabaei	Á. López, D. San Martín	Á. López, D. San Martín

Issue	Date	Description	Author(s)
0.1	10/12/2022	Initial version of the deliverable ToC	Lisana Berberi / Valentin Kozlov
0.2	05/01/2023	Content updates (Section 4)	Lisana Berberi
0.3	11/01/2023	Content updates (Section 3, summary)	Lisana Berberi
0.4	28/01/2023	Content updates (Section 3.1,3.2,3.3)	Elena Vollmer, Judith Sainz-Pardo, Juraj Bartok, Adam Fojud, Michal Blaszczak, Agnieszka Rausch, Khadijeh Fahimeh Alibabaei
1.0	31/01/2023	All sections are filled in, ready to be reviewed	Lisana Berberi
1.1	02/2/2023	WP6 internal review	Valentin Kozlov
1.2	09/2/2023	First external review	Pablo Orviz
1.3	20/2/2023	Second external review	Marcel Kvassay
1.4	21/02/2023	Addressing reviewers comments	Lisana Berberi
2.0	28/02/2023	Final review and acceptance	Alvaro López, Daniel San Martin

Table of contents

List of tables

List of figures

List of abbreviations

AI	Artificial Intelligence
BPMN	Business Process Model Notation
API	Application Programming Interface
CI	Continuous Integration

CD	Continuous Deployment
CSI	Characteristic Stability Index
CT	Continuous Testing
DoA	Description of the action
DL	Deep Learning
FAR	False Acceptance Rate
FL	Federated Learning
MSA	Multiple sequence alignments
ML	Machine Learning
MLOps	Machine Learning Operations
NDVI	Normalized Difference Vegetation Index
RNA	Ribonucleic acid
UAV	Unmanned Aerial Vehicle
UAS	Unmanned Aircraft System
XGBoost	Extreme Gradient Boosting

Executive summary

This document serves as a confirmation of the activities performed in T6.1 Initial Collection of User requirements.

The report describes how the requirement gathering process and results out of it are conducted and the support activities provided to the pilots applications to adequately co-design the architecture and platform of AI4EOSC.

Moreover, this report defines an initial implementation roadmap for the use cases and the preliminary plan on how to set up the leading by example approach task T6.2.

1.- Introduction

This deliverable describes the initial collection of user requirements and analysis from the three use cases of the project. This work is organized in collaboration with use cases representatives in WP6 (T6.1) and with technology representatives of WP3 (T3.1).

The AI4EOSC project bases its activities on the technological framework, the DEEP platform (provided on the EOSC marketplace), which was delivered by the DEEP-Hybrid-DataCloud H2020 project. AI4EOSC will enhance this platform to deliver new high-level services and functionalities. The project will continuously investigate the effectiveness and feasibility of the AI4EOSC concept and approach through the analysis of selected pilot use cases.

The methodology used in this task follows a bottom-up approach, starting with the identification of user requirements through the analysis of proposed use cases. This process involves defining user stories, personas, and epics linked to technical requirements. Information has been collected through dedicated interviews and translated into a "Technical Requirements dB" sheet, which is further described in Section 4.

Three use cases have been selected from different scientific disciplines for investigation, each with direct business impact. The use cases have been summarized in a tabular format covering various attributes such as motivation, target users, scientific domain, and application of ML/DL models.

In Section 5, we present insights from the requirements gathering process, which are illustrated through charts showing the defined metrics. In section 6, we outline the application development steps and release plan timeline.

Finally, in section 7 we draw the conclusions.

2.- Methodology used

T6.1 starts the process to gather user requirements by identifying the needs from the onboarded use cases (UC).
We used a bottom-up approach starting from the informal requirements identified as *user stories* (US), *personas* and *epics* associated with each use case. Once defined, they are linked to the technical requirements that are derived from them.

We compiled the collected information for each analyzed use case by conducting dedicated interviews. Then this information has been translated and transferred to the "Technical Requirements dB" sheet as referenced in [R1].

To give more context, the requirements elicitation methodology is thoroughly described in the "D3.1 State of the art landscaping and initial platform requirements specification" [R4]. The three aforementioned pieces of information: *personas*, *epics*, *user stories*, once extracted, help to identify the expected objectives and outcomes of the different stakeholders involved (i.e., customers/end users and development teams) and facilitate the communication among them.

We describe the requirement analysis process in more detail in Section 4.

3. - Summary of use cases

To investigate the effectiveness and feasibility of the AI4EOSC project concept and approach, we selected three use cases from several different real-world ones that originated from two distinct scientific disciplines, each having direct business impact.

In Table 1 we give a summary of the use cases described across these attributes: use case motivation, target users, scientific domain, Application of ML/DL model as a product, ML/DL Type/Algorithms and Performance Measurement.

Subject		UC1-Agrometeorology	UC2-Integrated Plant Protection	UC3-Automated Thermography
Use case motivation	*What is the problem you want to solve?*	Early warning of farmers before approaching thunderstorms using artificial intelligence techniques	Need for more precise solutions concerning mathematical models currently used in the plant protection service	Automation of feature detection in urban areas using thermal images and artificial intelligence techniques
	What strategic goal is this connected to?	Safety of farming	Reinforce the quality and quantity of food produced	Identifying energy losses to mitigate their effects and enable higher system efficiency
Target Users		Farmers, public administration, local governments	Farmers, public administration, local governments, scientific institutes and institutions responsible for monitoring hazards in agriculture in terms of plant protection	Urban planners, district heating network operators
Scientific domain		Agrometeorology	Agriculture	Energy (retrofitting / monitoring)
Application of ML/DL model as a product		Weather forecasting system	Recognizing plant diseases	Detection of thermal hotspots from thermal bridges and common urban features
ML/DL Type/Algorithms		Ensemble stacking, potential use of deep learning, vertical FL	Federated learning, composite AI	DL (potential use of decentralized learning techniques such as FL)
Performance Measurement	*How will you measure the accuracy of the predictions?*	By comparison to radar measurements, contingency table based scores like F1, POD, FAR, CSI[2]	Cross-validation with sets of photos marked by experts, comparison with mathematical models that calculate the risk (based on meteorological data)	By applying the models to annotated test datasets, set aside in advance. During the project, we shall try to identify suitable benchmark datasets and metrics for objective measurement of the quality of our models
	What is the minimum accuracy you expect?	POD higher than FAR in 30 minutes lead time	Accuracy of precise photos of plants at the level of 90%, accuracy of area photos at a level higher than 50%	Ca. 50% average recall for large thermal bridge detection on building rooftops, the rest remains to be determined.

Table 1: Summary of three use case implementations

[2] POD - Probability of Detection, FAR - False Alarm Ratio, CSI - Critical Success Index, F1 - F1 score

In conclusion, the use cases of UC1-Agrometeorology, UC2-Integrated Plant Protection, and UC3-Automated Thermography aim to solve important problems related to agriculture and energy, and to provide more precise solutions compared to traditional methods.

The use of ML/DL models in these use cases is expected to bring significant benefits to a diverse range of target users, including farmers, public administration, local governments, and urban planners.

The scientific domains covered by these use cases highlight the interdisciplinary nature of the use cases, which draw on expertise from fields such as agrometeorology, agriculture, and energy.

The use of cutting-edge ML/DL algorithms, such as ensemble stacking, deep learning, federated learning, and composite AI, demonstrates the potential of these use cases to drive innovation and bring significant benefits to the target users. The requirements of the use cases will also steer the platform development.

For AI model developers, the task of training a model with strong predictive capabilities is a challenge. However, the greater challenge lies in constructing a complete AI system and ensuring its seamless operation in production, not simply building the model itself. To accomplish this DevOps principles to ML systems (MLOps) have to be applied. In the fields of data science and machine learning, MLOps is becoming increasingly important [R5]. Teams need strong infrastructure to manage the massive amounts of data and computing power required for MLOps.

Existing open source MLOps frameworks have been investigated to what degree they comply and support the different categories/features as components of their workflow and workload management environment/system. They are reported in the ”D3.1 State of the art landscaping and initial platform requirements specification” [R4]. Furthermore, concepts to consider when setting up an MLOps environment for the data science practices, such as CI, CD, CT in ML are reported as well.

3.1 UC1 Agrometeorology

This UC is about usage of radar imagery together with in-situ measurements and numerical weather predictions (NWP) outputs to generate - utilizing AI - added value products for improving farmers activity, namely timely and precise warnings based on forecast of high impact weather for farmers - thunderstorms.

Target users are:

- farmers,
- public administration,
- local governments.

Thunderstorms can cause damage by a variety of means:

- high winds can break and damage buildings, equipment, material, plants
- hail can cause leaf damage reducing yield or destroying plants, machines, cars, buildings

- flash floods can endanger humans, animals, damage machines and equipment, can lead to loss of topsoil as well as damage to crops.

The farmer needs to check the meteorological forecast before planning any work activities and make sure to have a way of receiving weather information while working, especially at remote locations. In the current state, a farmer listens to the weather forecast about thunderstorms in the coming day. As an enhancement, he wants to know when exactly they will hit his area so that he can get everyone to safety. Nowcasting can provide localized warnings around 30 minutes ahead of thunderstorms.

Fig. 1: Approaching thunderstorm

Enhancements:

- AI - ensemble stacking to combine different models. If applied correctly, it is known to outperform any of the underlying models.
- AI - to grasp complex non-linear features of natural processes
- combination of large scale data (radar measurements) and local point measurements (as additional predictors), preserving the benefits of both approaches: preciseness from the ground based data and spatial coverage due to the radar imagery
- employment of numerical weather prediction (tailored short-range forecasts for the target sites) as a predictor.

3.2 UC2 Integrated plant protection

Use case 2 aims to enhance capabilities of currently used disease detection methods based on mathematical model calculations, with new possibilities of ML/DL-based models developed and scaled on AI4EOSC platform.

ML/DL will be based on a network of meteorological data from ground stations, the results of existing mathematical models, and ground observations. At the same time, they would be enhanced with greater terrain coverage and spatial precision by using satellite data. Current precision based on ground data is about 30 km. Supporting spatial data sets would be both photos in visible frequencies as well as other spectra and indices such as NDVI (normalized difference vegetation index) images. The enhancement is going to leverage solutions developed within AI4EOSC, e.g. federated learning and composite AI.

Fig. 2: Sample image taken during field observation. Zoom in to the leaves of a beet growing in the ground

The developed models are going to be integrated into existing national advisory platforms (eDWIN), operated by WODR and PSNC. This platform is dedicated both for advisors and farmers, which includes a network of meteorological ground stations (operated by 16 Regional Agriculture Advisory Centres), the Farm Management System - Virtual Farm application, and ground observations (visual observation stations and existing network of observers) of the occurrence of diseases and pests. The current solutions are based on predictive mathematical models. Platform in its first release enables individual risks of cumulative risk calculations for most common crops and related pests and disease:

- Potato blight
- Colorado potato beetle
- Archer's dart in beets
- Cereal leef beetle
- Cercospora in beet

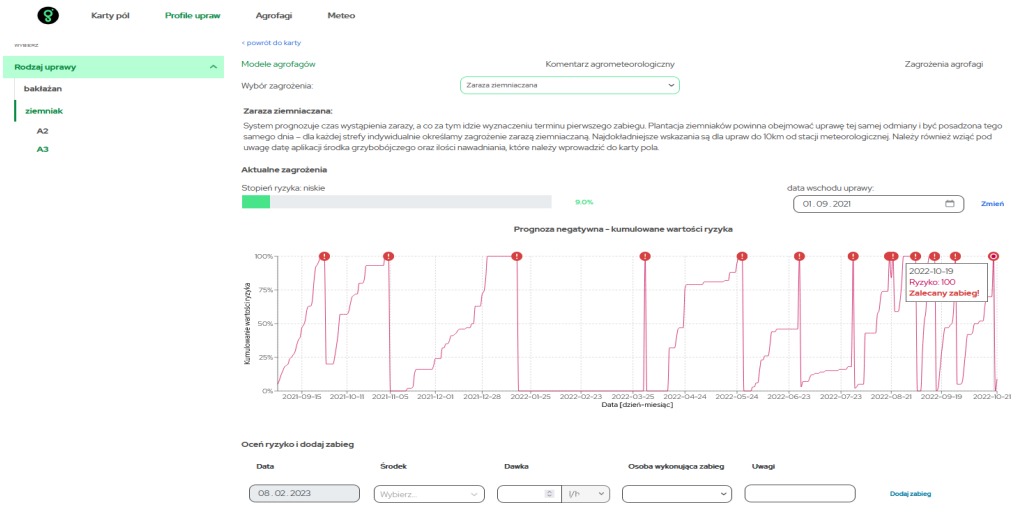

Fig. 3: eDWIN application view with disease calculation results from https://www.edwin.gov.pl/

Mathematical model results are being displayed to the user in the time domain up to the present day.

Target users are farmers, public administration, local governments, scientific institutes and institutions responsible for monitoring hazards in agriculture in terms of plant protection.

The planned number of the eDWIN platform users (where the outputs will be integrated) is around 100,000 in Poland (Farmers, Advisors). The results will be also integrated with other agriculture platforms in Poland. Developed AI models can be used in other countries and platforms. The important feedback is on improving the food production quality and safety by reducing usage of the pesticides.

3.3 UC3 Automated thermography

Use case 3 leverages thermal UAV-based imaging combined with artificial intelligence to identify “hot spots” (thermal anomalies) in urban settings and contributes to improving the efficiency of energy-related systems. In this instance, the general idea can be implemented within two scenarios:

1. Detecting thermal hotspots on building rooftops caused by thermal bridges. This supports urban planners and building owners in pinpointing retrofitting potential.
2. Detecting thermal hotspots on the ground caused by urban features (cars, manholes, streetlamps, etc.). This supports district heating network operators in their search for pipeline leakages by automatically removing common false alarms from the list of potential suspects

Currently, all named parties are unable to quickly and accurately pinpoint the location where heat losses occur and therefore struggle to maintain a high energy efficiency. In both scenarios, the overarching objective therefore lies in automating the detection of

thermographically salient heat losses to accelerate the implementation of necessary counter-measures and repairs.

AI can be used to this end as either a stand-alone tool (1.) or integrated into a pre-existing pipeline (2.).

Fig. 4: Experimental setup for UAS flights in Karlsruhe, Germany, including the UAV itself (DJI Matrice 300) and portable control device

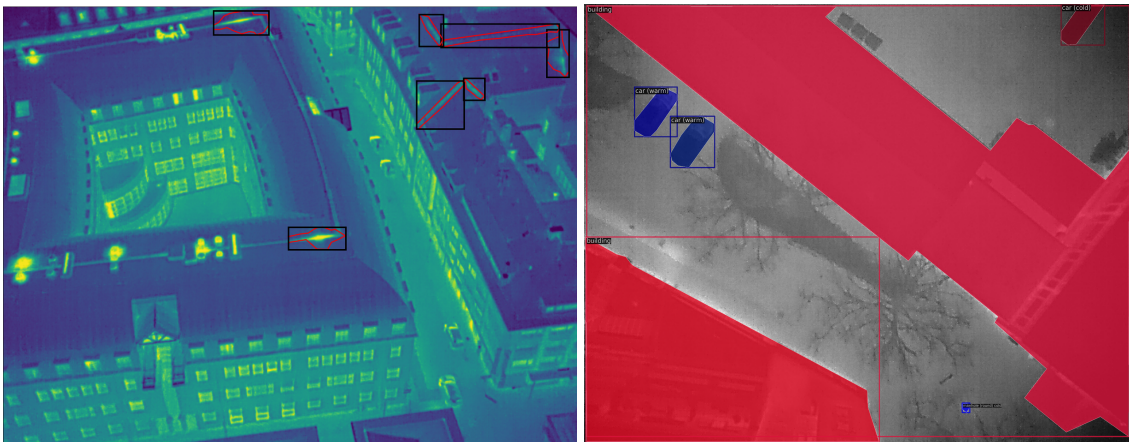

Fig. 5: Examples of annotated thermal images for (1, left side) thermal bridge detection and (2, right side) urban feature detection

Enhancements:

- implementing deep learning models for hotspot detection via e.g. instance segmentation in combined thermal and RGB image data
- creation of a cloud-based automated service leveraging best practices and technology advances of the AI4EOSC platform, potentially making use of decentralized learning techniques like federated learning by selecting each client according to the geographic location an image was taken.

3.4 Leading by example approach

In order to efficiently and effectively exploit all the capabilities of the AI4EOSC platform we proposed to use a "Leading by example" approach (T6.2) to guide and help use case representatives to implement their specific use cases by providing consultancy and knowledge exchange.

We identified two different scenarios: *classic* and *advanced,* from the phase of collecting and analysis of user requirements based on the technologies that use cases need in order to implement and deploy their AI/ML/DL applications.

3.4.1 Classic implementation scenario

The classic scenario consists of simple but representative steps and an AI/ML/DL application to guide the three use cases during their implementations.
The classic scenario will comprise the following steps:

- an interface to select the required computing and storage provisioning
- an interface to access own datasets used as inputs for model training
- an interface to deploy model training and/or as prediction service
- a CI/CD pipeline setup
- a step-by-step tutorial as user documentation

To serve as a demonstration, we have chosen a specific use case from the field of Biology and Medicine to be used as a leading by example approach. Ribonucleic acids (RNA (plays a crucial role in these fields both in determining life at the molecular level and in medical applications. The ability to predict RNA tertiary structure from its nucleotide sequence would greatly advance any study into its function and associated application.

Given the sparsity of RNA data, a group of researchers at KIT, FZJ and DLR has chosen prediction of RNA spatial adjacencies ("contacts") as a simplified proxy task for full RNA structure prediction. These researchers have implemented RNA [R6] Contact Map Prediction [R7] by Data Efficient Deep Learning RNA via a Deep learning model that received the multiple sequence alignments (MSAs) [R8] of the RNA as input .

The model consists of two parts, one upstream and one downstream. The upstream model is a self-supervised learning model that receives the augmented MSAs as input to the model. This model itself consists of two components, the backbone (preprocessing step plus attention layers), and the task heads. In the preprocessing step, some augmentation steps such as masking, jigsaw and others are performed on the MSAs, and then these augmented sequences serve as input to successive multi-head self-attention [R9] layers that extracts features from them. The task header for each preprocessing step is located after the attention layers and can be used to determine which preprocessing technique was employed in the first step. When only little labeled data is available, as in the case of the RNA dataset, the performance of the model can be improved by applying the techniques of self-supervised learning.The output of the attention layers is used to train a downstream model, which is either a

simple linear model or an Extreme Gradient Boosting (XGBoost) [R10] model for classification. The output of the downstream model is the contact map associated with the input MSA sequence.

At the beginning of the implementation of the model on the platform, the weights of the backbone model are frozen, and the user can input their MSAs into the model and extract the features from the backbone, select the downstream model and its hyperparameters, and re-train the downstream model on their own dataset.

3.4.2 Advanced implementation scenario

The advanced scenario consists of a more complex (AI/ML/DL) application to guide the use cases that will use the most promising artificial intelligence (AI) technologies like Federated Learning (FL) and/or composite AI (C-AI) techniques to run their inference.

The downstream model is a simple ML model such as the XGBoost model or a simple linear model. Training these models is simple and fast, and we do not need many resources. However, to get good results, the user dataset should be similar to the dataset used to train the backbone. Unless the model performs poorly with the user dataset.

Unlike the downstream model, the backbone model is a more complicated model that consists of multiple multi-headed attention layers. The backbone model is used to extract the features from the MSA sequences and to figure out which preprocessing is used in the first preprocessing step. For example in the case of jigsaw, the model is attempted to find out which permutation is used during the preprocessing step. In this case the input is the permuted MSA with size of (E,L) and the permute size of (E, D). The true label is the permutation that we used for permuting the original MSA. Since the number of trainable parameters of the backbone is much larger compared to the downstream model, training the backbone requires a large amount of memory. In a further development step, we will add the ability to train the upstream model along with the downstream model in the API. The user can use their own dataset and train the composition of the backbone and downstream model to get the best performance for their dataset.

The advanced scenario will comprise the following steps:

- an interface to select the required computing (CPU and/or GPU cores) and storage provisioning
- an interface to train over distributed datasets (from heterogeneous data sources) applying distributed learning techniques and algorithms (such as Federated Learning, Gossip Learning or Split Learning among others).
- an interface to access and combine multiple AI models to include in their workflows
- a step-by-step tutorial as user documentation

4.- Requirements Analysis

In order to adequately capture the requirements coming from use cases proposed in the DoA we designed a process model using BPMN notation [R3] as a de-facto standard for business processes diagrams. The diagram is depicted in Fig. 6.

We model this diagram as an orchestration between three different participants:, “T6.1 & T3.1 reqs collector & advisory” as a support team, “Use case pilots [WP6]” as stakeholders to express their requirements for implementing their DL application, and “Platform Team [WP4]” as resource providers that will add the requested computing/storage resources and manage them. They are represented graphically in three lanes in the single pool (depicted as a rectangular container) named “AI4EOSC Requirements Gathering Unit” to organize and categorize activities according to function or role and the interactions between the parties involved.

This process starts with two parallel activities (represented with a rectangular box) “Fill in the Epics-Personas-US Template” and “Fill in the Technical Aspects” being executed by the team of use case owners. The respective data output (A-doc in .docx and B-sheet in .xlsx format) are generated once the two activities are executed (respective template examples of the two data outputs are shown in Fig. 7 and Fig. 9 in section 4.1). Afterwards, the review phase begins and “Test cases and Value propositions” information are added to the first document Epics-Personas-US by the team of T6.1. Then, the content is revised and if there is still some missing information the use case representatives are asked to add it as an input, otherwise the document is published as a completed version and ready to be transferred to the database.

We prioritize the requirements derived from use case analysis by relevance to target users after proper transformation of collected information.

We collaborated as a team to avoid ambiguity in use case requirements collection through joint discussions during dedicated interviews. More information about the latter is available in Section 4.1.

The results of this gathering process we conducted are described in Section 5.

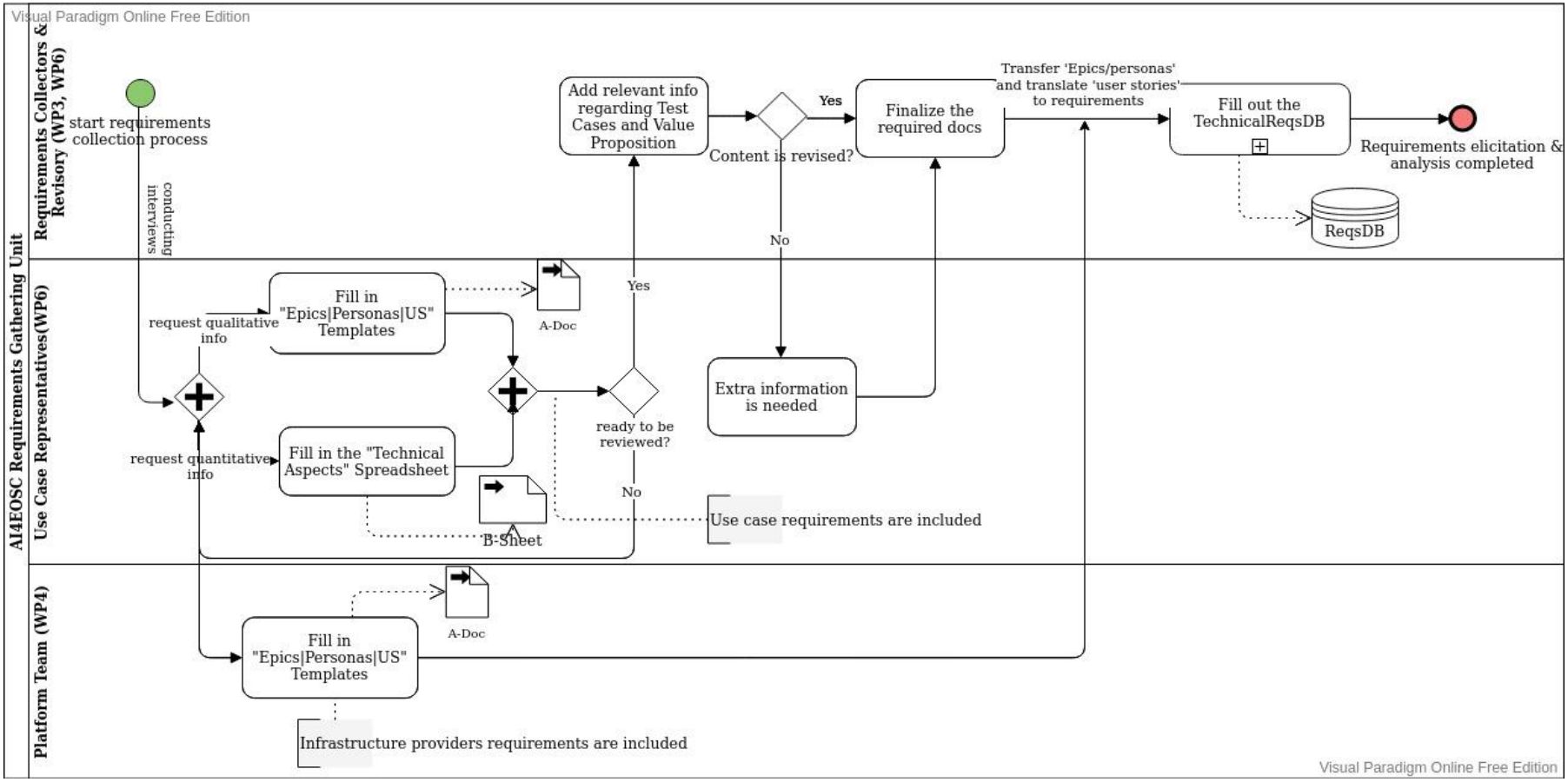

Fig. 6: Requirement gathering process BPMN diagram

4.1- Use Cases Interview

To elicit both quantitative and qualitative information from the use cases we conducted dedicated interviews over a period of three consecutive weeks (during November 2022), with one interview scheduled per week.

Two specific documents have to be filled in by the use cases in advance, so-called: Epics-Personas-US word document and Technical Aspects excel spreadsheet.

The template of Epics-Personas-US document has been inspired by the Requirement Analysis Methodology suggested in D3.1 and by the course "Agile Meets Design Thinking" from Coursera [R2] and proposed as best practices in design thinking to create valuable outcomes for the user. Therefore, we build something that users want by developing personas and problem scenarios. In this template there are 4 main sections associated to a persona:

Persona <*thinks*>: Refers to the beliefs or opinions of a character or persona.
Persona <*sees*>: Refers to the observations or perceptions made by the persona in a specific situation.
Persona <*feels*>: Refers to the emotions experienced by the persona in response to an event.
Persona <*does*>: Refers to the actions and behaviors performed by the persona..
A template example of the "Persona-E as a Model developer" for UC3 is shown in Fig. 7, whereas the problem scenarios and value propositions example in Fig. 8.

Model developer // expert

Persona-E is a use case representative with a background in mechanical engineering (M.Sc.) focused on energy technologies and robotics. As part of her PhD she is currently working on automating the thermal image analysis for anomaly detection to provide a feasible monitoring solution for district heating system operators. Her programming skills are of the learning-by-doing kind in Python (and Matlab). She has only very little experience with the development of AI models but believes that they can be helpful in improving her current analysis.
She is collaborating with an expert who has previously created a model using some of the thermal image data for thermal bridge detection.

Thinks	She (the model developer) faces many challenges in developing the reliable monitoring solution for the thermal image analysis: 1. She thinks that more data would help her in developing a more reliable model. However, drone flights require some organisations and other groups having similar data are not willing to share their raw data. 2. She thinks that the process of labeling the raw data (images) is very time consuming but very important for the model training. If it could be more efficient, better models could be developed faster. 3. She is currently using the HPC system, where a job is queued before the calculation starts. Therefore any bug in the code appears only after the code is executed. She thinks this can be improved by e.g. software testing practices.
Sees	She sees that: 1. While drone flights are not easy to order, people may profit from Federated Learning (FL) where the data are not shared among the different organizations but the trained parameters for the models are communicated to a central server which orchestrates the process. 2. Labeling of data can be improved when making use of the model results themselves (active learning), and then the dataset can be increased. 3. She heard about software testing, e.g. unit tests, functional tests, continuous integration and continuous delivery systems (CI/CD) allowing to automate testing and software building but never implemented herself and wants to learn and practice.
Feels	She feels that many things could be improved and tested but they may take a lot of time and effort for the limited time of her PhD.
Does	1. She contacts district heating network operators to see whether they are interested in such flights and organizes them in collaboration with a drone pilot. The flights are automatic so the actual acquisition isn't very challenging. 2. Data is annotated by hand, which is a very time-consuming endeavor. An active learning approach is planned to improve and partially automate this task. This would still be done locally and not include other network operators or images collected there.

Fig. 7: Persona-E as a model developer

Problem Scenarios / Jobs-to-be-Done	Current Alternative	Value proposition
In some cases he/she does not have a large enough amount of data available for training the deep learning models and it will be necessary to jointly use data obtained from different locations to train the models and achieve an adequate level of robustness. However, this is not always possible due to privacy restrictions. In the other words, she wants to expand her dataset but data from other groups or network operators may not be shared.	Train models by centralizing data from different locations (when privacy conditions allow it) or train models individually in each location (with a small amount of data that may lead to inaccurate models). The only way to expand data is to order a new drone flight, which is not always easy.	AI4EOSC will offer the option of federated learning with a central server to aggregate trained parameters from various learning agents and share the aggregated weights back. This would also allow greater collaboration in cases where privacy restrictions prevent it. • The prediction app has to include a preprocessing stage. It might be a part of the application itself or we talk about deploying a whole pipeline, meaning AI4EOSC should support ML/DL pipelines. • Learning agents will communicate to the central server offered by the platform the weights or parameters obtained when training the model with their own data.
She wants to improve the data labeling process in order to enlarge the training dataset(s).	Done manually	• ToDo: search for a data labeling practices and tool (collaborative labeling, expert verification) • Missing functionality?
She wants to expand computing resources by exploring cloud resources.	Currently runs on the HPC system available locally.	AI4EOSC will allow easy access to cloud resources for the model development and training (based on DEEP solutions) through the Dashboard and cloud resources of partners.
She wants to keep track of trained models and store the best ones in some stores.	Models are stored locally as files.	AI4EOSC will offer: • Means to track ML development cycle and models (e.g. DVC, MLflow). • a model store solution for its users.
She wants to automate some stages of her ML pipeline, e.g. automatic model retraining once new training data is available.	All ML steps are executed manually.	MLOps, e.g. retraining triggered on an event like a new dataset (e.g. public) becomes available or when a new federated learning agent is registered.

Fig. 8: Problem scenarios and value propositions of Persona-E

The revised and completed template documents for all UCs are shared in a Google Drive: UC1 is [R11] , UC2 in [R12] and UC3 in [R13].

In the same document epics, user stories are defined as well. More information about them is reported in the following section.

In order to collect technical aspects on use case needs like Nr. of CPU cores, RAM memory allocation, etc. we introduced another sheet like the one presented in Fig. 9 where only a part of it is shown due to page space limitation. The complete information on the technical aspects of all the three use cases may be accessed from [R14]. The sheet is organized as a questionnaire with structured questions covering various elements of an ML lifecycle.

Questions-1	UC1: Agrometeorology	Question-2	UC1: Agrometeorology	Questions-3	UC1: Agrometeorology
	Data source UC1.DSo1				Response
How data is acquired?	by meteorological radar	*Do you need to label collected data?*	no	*Do you have training dataset(s) already?*	No
What type of data is collected (images, digits, text, ...)?	2D data and 3D fields of radar reflectivity usually in HDF5 format	*What is used to label data?*	current measurement is truth for previous forecast	*What is the size (GiB, TB) of your minimum training dataset?*	Not known yet, but 2-3 TB I guess
Where is the data stored (type of storage)?	tape storage	*Does data require anonymisation ?*	no	*Any constraints on the dataset access (e.g. fast access, privacy, ..)?*	no
Can the data be transfered to a 3d party cloud?	probably yes	*Do you need to augment data?*	probably not	*What deep learning framework is used (Tensorflow, PyTorch, ...)?*	We use both Tensorflow and PyTorch
At what rate the data is collected (e.g. 100GiB/year)?	300 GiB / year	*Any time constraints for data preparation?*	no more than 1 month seems practical time	*Any tools to track your modelling experiments (e.g. MLflow, Tensorboard, DVC, ...)?*	starting to work with DVC, will consider other options as well
How big storage may be required for the time of the project (3 years)?	2TB for data	*Storage requirements to store prepared data?*	2-3 TB	*Do you see a use of Federated Learning?*	most probably yes
Data Sources		**Data Preparation**		**Data Modelling**	

Fig. 9: An Example of structured questions covering various elements of an ML lifecycle (from UC1)

4.1.1- Epics-Personas-User Stories

We follow the “Epics-Personas-User Story” approach suggested in D3.1 as a method to gather requirements for use cases. This approach involves defining high-level goals or epics, creating personas to represent the target users, and writing user stories to describe the actions and goals of the personas.

This approach helps to ensure that the requirements for a use case are aligned with the needs and perspectives of the target users, leading to more effective and user-centered solutions.

In Table 2, we present an excerpt of the collected epics, personas, user stories for each use case. The epics, personas, and user stories have been prioritized based on their relevance to the target users and the desired outcomes of the use case.

Use Case	Epic ID	Title	Description	Personas	User Story	Prioritize
UC1-Agrometeorology	UC1.E01	Productive data science	To be productive, data scientists, data engineers, AI developers need to enhance the efficiency of their typical data science tasks, e.g. data processing and visualization, build, deploy AI/ML models, code testing etc. This means that these tasks need to be performed at a higher speed and accessing any of the available hardware/software resources from the established platform	Data Scientist	UC1.E01.US01: As a data scientist I want to have more computing resources so that I can test alternative models and find the best one faster	Must have
	UC1.E02	Extensible and integrable forecasting product	The end-user application as a final product should have the relevant functionalities to allow to integrate it with other existing systems in production, e.g. a forecasting system and to extend by adding extra features to its original application	Product Manager	UC1.E02.US03: As a Product Manager I want to be able to check the model performance so that I am sure that the application does not degrade in performance	Must have
	[...]	[...]	[...]	[...]	[...]	[...]
UC2-Integrated plant protection	UC2.E01	Input data management	Input data management for an AI application involves several key steps, including: Data collection: This step involves gathering the data that will be used to train and test the AI model; Data cleaning and preprocessing: This step involves preparing the data for use in the AI model; Data storage and management; Data Annotation	Model Developer	UC2.E01.US04: As a Model Developer, I have to preprocess the images and prepare the data so it can be used as an input for the AI model.	Must have
	UC2.E02	Building and Deploying a ML Model/System in Production	Building and Deploying a ML Model/System in Production involves several key steps, including: Model development: This step involves developing the ML model using the collected and preprocessed data; Model validation; Model deployment; Model monitoring and maintenance	Model Developer	UC2.E02.US01: As a Model Developer, I want to train a DL/ML model in order to be able to efficiently predict the detection of pests disease	Must have
	[...]	[...]	[...]	[...]	[...]	[...]

Use Case	Epic ID	Title	Description	Personas	User Story	Prioritize
UC3-Automated Thermography	UC3.E01	Train an AI model in application for automated thermography (AT)	Train an AI model to efficiently detect thermal anomalies in thermographic images acquired via UAV which need to be preprocessed, labeled and stored for long-term use	Model Developer	UC3.E01.US01: As the model developer, I have to acquire thermal images via UAV and share / store them for long-term use so that I can process them for preparing a training dataset.	Must have
	UC3.E02	Provide an integrable prediction tool for end users	The application as a product will allow non-expert end users to detect thermal anomalies in thermographic images. There is the option to integrate the AT service with an image analysis program of a data scientist or build additional services on top of it, so that it becomes a part of a larger service/program	End-user (e.g. Urban Planner or Network Heating Operator)	UC3.E02.US02: As the end-user I want to analyze my UAV based images using the AT service/application so that I can identify potential problems (e.g. heat bridges, leaks)	Should have
	UC3.E03	Allow external contributions	To improve the models that would be obtained after training with the data available from a single center or institution, collaboration with others with similar data is key. This can be achieved by applying a federated learning approach, i.e. that the model can be improved by using the data of other parties without sharing them.	Model Developer	UC3.E03.US01: As a model developer, I want to apply federated learning so that the model can be improved by using the data of other parties without sharing them.	Should have
	[...]	[...]	[...]	[...]	[...]	[...]
Resource Provider Use case	UCX.E01	Providing computing resources to the AI4EOSC platform or deploy platform on-premises	A resource provider can offer computing power to the AI4EOSC platform in the form of virtual machines or cloud-based services. This allows the platform to access additional resources as needed, scaling up or down as necessary to meet the demands of the use case applications.	Infrastructure Provider	As an infrastructure provider, I want to be able to easily add my computing resources to the AI4EOSC platform, so that I can support the R&D of artificial intelligence within the scientific community and monitor the usage of my resources.	Must have
				Platform Administrator	As an AI4EOSC platform administrator, I want to be able to install and administer the platform on my own premises, so that I have full control over the infrastructure and can ensure compliance to any necessary regulations or security protocols.	Must have

Table 2: An excerpt of epics/personas and user stories for each use case and for AI4EOSC Resource Provider as an extra use case

As a summary we identified 35 user stories, 7 personas and 8 epics from the 3 use cases. Finally, the aforementioned info is transformed to technical requirements that will be addressed and completed from the architecture design and platform provider teams as reported in the following section.

4.2- Requirements Gathering

Once we recorded all the Epics-Personas-User Stories for each use case, we translated them to Requirements, each with a unique identification in the format UC#.Req#. Further attributes associated with a requirement are the Title, Priority (Options: must have, should have, nice to have), the associated Category, (e.g. Computing, storage, etc.), a Description (e.g. what are the resources, tools needed to address the requirement and if this is a critical, moderated or optional aspect), Status (e.g. Defined, In Progress, In Testing, Implemented, Canceled).

As a next step we will associate each requirement to a Component/Tool of the AI4EOSC Platform that will be defined in the D3.2 "Initial high-level architecture specification". Other important attributes that will be recorded into the database are: Acceptance Criteria (e.g. 32GB RAM and 1 GPU, the specific conditions that must be met for a requirement to be considered complete and accepted), Supporting materials (reference sources/links to show that the requirement is met), Tentative scheduling (propose a datetime when the Requirement will be completed), Requester (who requested it), Owner (who is in charge of it) and Version (e.g. 1.0, a number that keeps track of the state of it).

UC#.Req#	Title	Priority	Required for (User Story)	Category	Description	Status
UC1.Req01	Provision/customize the required computing resources to optimize the AI/ML/DL training process	Must have	UC1.E01.US01, UC2.E02.US01, UC3.E01.US03	Computing resources	Proper selection of computing resources must be done in order to speed up the AI/ML training process. *Appropriate computing resources are available (hardware is available): 32GB GPUs, enough storage (c) *Computing resources information is easily available for model developers (m) *Appropriate computing resources can be requested/selected by a user via the Dashboard or CLI: number of CPUs, amount of storage, TYPE OF GPU (e.g. GPU memory is important!), Number of GPUs (c)	Defined
UC1.Req02	Organize and track all training experiments	Must have	UC1.E01.US03, UC2.E02.US04, UC3.E01.US05	Data version control enabled	*The model developer can log parameters of interest;can compare multiple runs (c) *It is possible to reproduce and re-run training experiments (c)	Defined

UC#.Req#	Title	Priority	Required for (User Story)	Category	Description	Status
UC1.Req15	Check/Run data, software and user application (service) quality	Must have	UC1.E01.US02, UC2.E03.US02, UC3.E01.US04	SQA installed	*SQA tool criteria	Defined
[..]	[..]	[..]	[..]	[..]	[..]	[..]
UC2.Req01	Store and share raw image data to be processed and prepared for a testing dataset	Must have	UC2.E01.US01, UC2.E01.US05, UC3.E01.US01	Storage Data Transfer Solution	*Acquiring images via a mobile application(is being developed) *data sources: Stationary cameras *Experts mobile cameras *Users mobile cameras (not regular) meteorological stations *External storage has enough volume for raw data for the time of the project(c) *External storage can be accessed via Internet(c)	Defined
UC2.Req05	Preprocess and prepare input data sets to be used for training and evaluation	Must have	UC2.E01.US04	Required Tools installed and configured	*Preprocess:Initial filtering from nonvalue objects, test reports etc *Computing resources for preprocessing *Storage has to be synchronized with the platform (c) Platform has running endpoint where flagged data can be pushed (flagged data is automatically copied to destination link) (m)	Defined
UC2.Req06	Training a DL model efficiently to predict the occurrence of pest diseases (astrophage)	Must have	UC2.E02.US01	Computing resources | Storage | DL Frameworks	*The required Computing resources available *Access to input data sets from different data sources *Access to the selected data sets is given (c) *Necessary amount of computing and storage resources is allocated (c) *It is possible to combine different data sources (acquired images from the domain specialists and radar/satellite data) to increase the prediction rate/accuracy (composite AI case) (m)	Defined
[..]	[..]	[..]	[..]	[..]	[..]	[..]

UC#.Req#	Title	Priority	Required for (User Story)	Category	Description	Status
[..]	[..]	[..]	[..]	[..]	[..]	[..]
UC3.Req06	Add a user-friendly GUI for my application to ease access for non-IT persons	Should have	UC3.E02.US04	Web frameworks	*The model developer has an option to add in a simple way a GUI to the application, e.g. based on the Gradio framework (good practices examples are provided) (o)	Defined
UC3.Req09	Containerize the application(s)	Must have	UC3.E02.US03	Docker frameworks	*The final application features a clearly documented API (c) *The prediction service is easily deployable either in the AI4EOSC platform or on-premises (m)	Defined
UC3.Req11	Monitor the model performance in production	Must have	UC3.E03.US02	Monitoring via dashboard	*The data quality tests are implemented (m) *The model key metrics is identified (m) *Operational parameters (e.g. CPU, memory, latency) to monitor are defined (m) *The monitoring system to log data, model and operational parameters is available (m) *The end-user can assess results of the prediction and submit a report (o)	Defined
[..]	[..]	[..]	[..]	[..]	[..]	[..]

Table 3: An excerpt of use case requirements

As a summary, we identified 44 Requirements for the 3 use cases. In Table 3, we present a sample of the collected requirements from the full list, which is stored in our database in a spreadsheet format[R1]. The complete set of entries can be accessed through the database. Furthermore, we (in collaboration with WP3 team) will *monitor requirements* by updating their status attribute, which will progress from the initial value of "Defined" to "In Progress" once processing begins, "In Testing" during the testing phase, "Implemented" once it has been addressed, and “Canceled” if it has been canceled.

More metrics and graphics are described in the following section.

5.- Insights from Requirements Results

The successful identification and categorization of use case requirements through aggregated results are illustrated in charts showing the defined metrics.

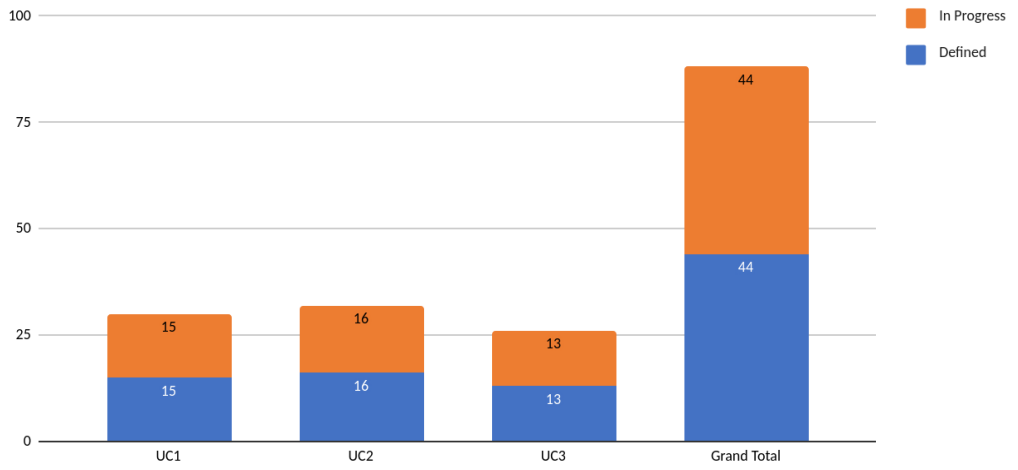

Fig. 10: Number of requirements per each Use Case

From this stacked column chart, it can be inferred that the total number of requirements for all use cases (UC1, UC2, and UC3) is 44, with each use case having 15, 16, and 13 requirements, respectively.
Additionally, all requirements for all use cases have a defined status and currently “In Progress”.

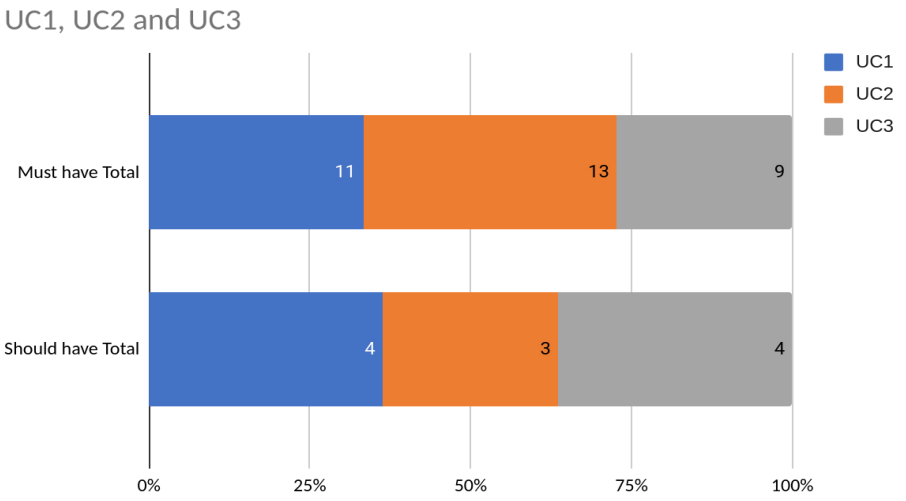

Fig. 11: Use Case Requirements classified by priority

Fig. 11 shows the requirements for three use cases (UC1, UC2, and UC3) classified by priority and status using a stacked bar chart. The "*Must have*" column indicates the number of requirements that must be met for each use case, the "*Should have*" column indicates the number of requirements that are desired but not essential.
For UC1, 11 out of 15 requirements are considered "Must have" (73%) and 4 are "Should have" (27%). For UC2, 13 out of 16 requirements are "Must have" (81%) and 3 are "Should have" (19%). In UC3, 9 out of 13 requirements are "Must have" (69%) and 4 are "Should have" (31%).
Overall, it can be seen that the majority of requirements for all use cases are considered "Must have".

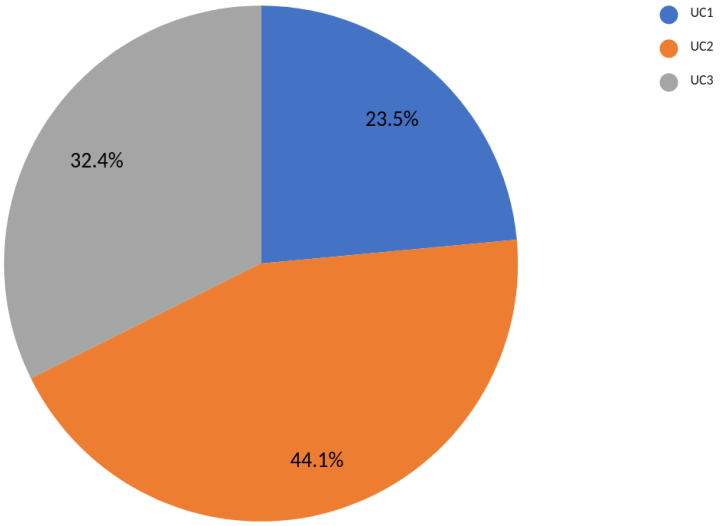

Fig. 12: Number of User Stories for each Use Case

The pie chart in Fig. 10 shows the distribution of user stories among the three use cases. For example, UC1 would take up a 23.53% slice of the pie (8 / 34), UC2 would take up 44.12% (15 / 34), and UC3 would take up 32.35% (11 / 34).

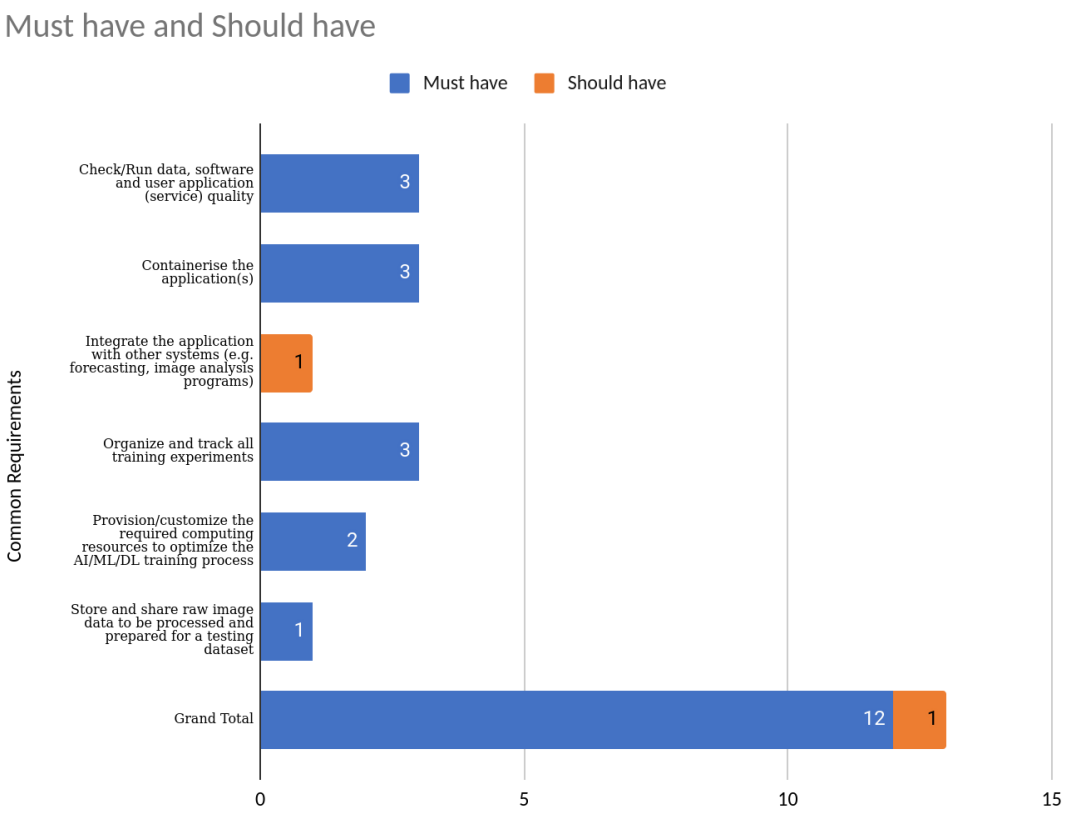

Fig. 13: Common Requirements and their prioritization

Fig. 13 displays common requirements across 3 use cases, and their prioritization as either "Must have" or "Should have". The metrics shown is the COUNTA of the requirement, which counts the number of times each requirement appears in the table.
The stacked column chart shows that the majority of the common requirements are considered "Must have" (12/13), and only one common requirement is considered a "Should have" (1/13).
The most frequent requirement with a "Must have" priority is "Check/Run data, software and user application (service) quality" with 3 occurrences, followed by "Organize and track all training experiments" and "Containerise the application(s)" with 3 occurrences each.
The number of requirements per use case, user stories, and common requirements provide a clear picture of the scope of the project and the priorities for development. The classification of requirements by priority helps in making informed decisions on prioritizing the development of features that are most important to the target users. Overall, the results of the requirements analysis provide a solid foundation for the successful implementation of the project.

6.- Implementation Timeline (Planning) for use cases

The application development and release plan outlines the steps and timeline for creating and launching a software application. This plan helps to coordinate the efforts of development teams and ensures that the application is delivered on time and with the desired features and functionalities.

This is illustrated using *GANTT* charts in Fig. 14, with the first release phase starting in February and ending in December of 2023. The first release will encompass the three use cases concluding in December 2023 (each represented with a colored diamond box). Whereas, the second release will commence in January 2024 and is expected to be completed by the end of April 2025.

The 1st UC release will include several important milestones: completing a thorough requirements analysis, becoming familiar with the platform, and designing, coding, building, and testing a UC (use case) solution.

Whereas, the 2nd US release will focus on reviewing and updating the roadmap, as well as implementing federated learning (FL) and/or composite AI techniques as outlined in the advanced scenario. This release is expected to build upon the first release and further enhance the capabilities of the platform.

During both releases, continuous feedback monitoring will be implemented to ensure that we can gather feedback from users and make necessary improvements.

The AI4EOSC platform is expected to have two releases. The first release is scheduled for February 2024, on M18, while the second release is planned for May 2025, on M33.

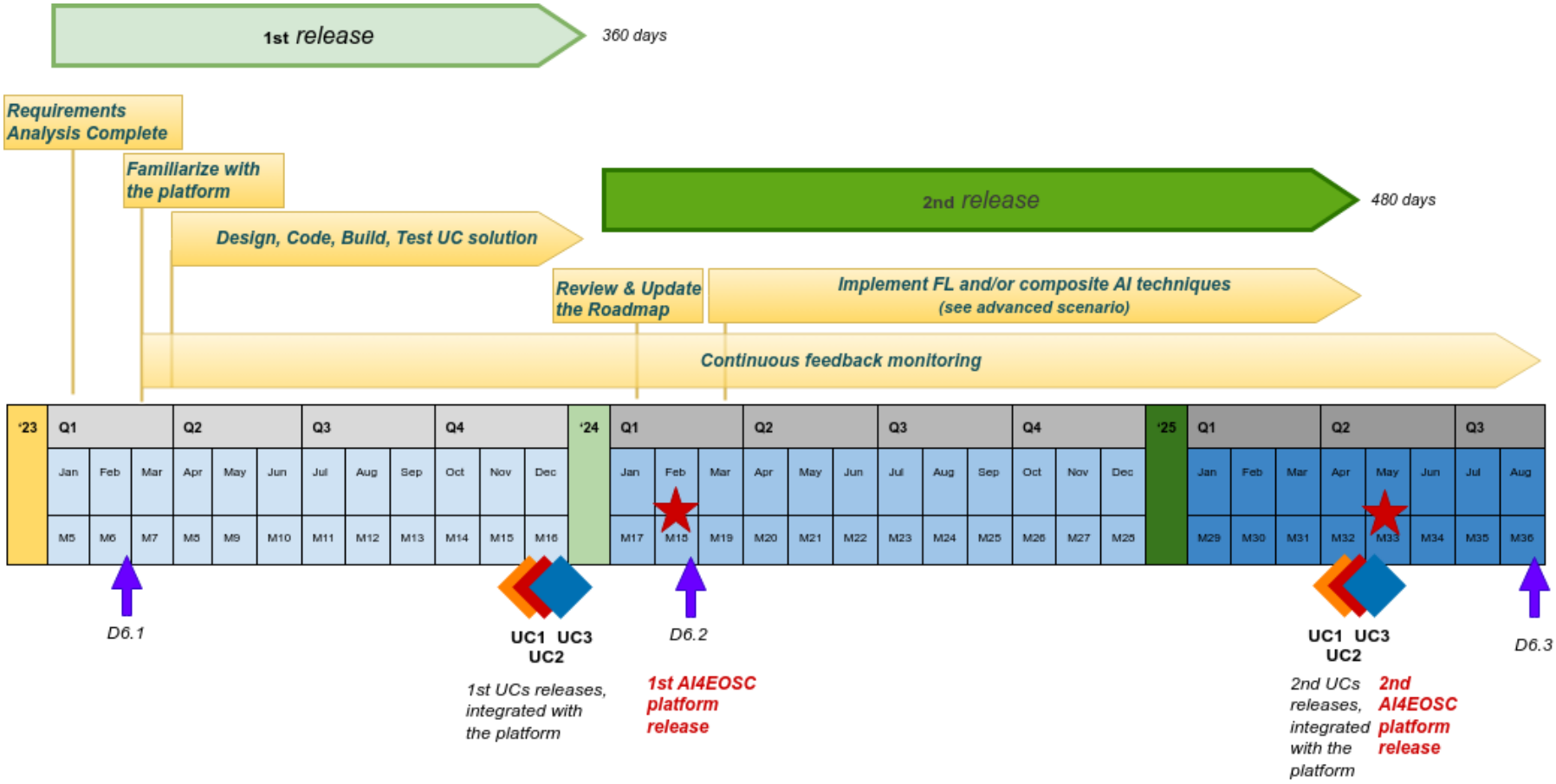

Fig. 14: Application development and release plan for the three use cases

7.- Conclusions

In this deliverable, the initial collection and analysis of user requirements from three proposed use cases were described. The methodology used was a bottom-up approach starting from informal requirements identified as user stories. The collected information was gathered through dedicated interviews and translated into the Technical Requirements database. The report also defines an initial implementation roadmap for the use cases. Following the Agile principles, the roadmap is going to be reviewed and adjusted accordingly in the next deliverable D6.2 “Intermediate status report about integration of pilot applications”.

The use cases selected represent two scientific disciplines and have direct business impact, providing a comprehensive evaluation of the AI4EOSC project concept and approach.

Links

[R1] “AI4EOSC-RequirementsDB” sheet, https://docs.google.com/spreadsheets/d/1wFe3YKH3hZTegTwaOR8rfFh-6dIFy6-G/edit#gid=1342414375

[R2] “Agile Meets Design Thinking” course, offered by University of Virginia. https://www.coursera.org/learn/uva-darden-getting-started-agile

[R3] Business Process Model Notation http://www.omg.org/spec/BPMN/2.0/ (Accessed: Dec 2022)

[R4] D3.1 “State of the art landscaping and initial platform requirements specification”, https://docs.google.com/document/d/1EY343S3b1QXDw8IqT6u-pNZpjHRADrSK/edit?rtpof=true

[R5] MLOps: Continuous delivery and automation pipelines in machine learning, https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning

[R6] https://en.wikipedia.org/wiki/RNA

[R7] Pucci, Fabrizio, et al. "Evaluating DCA-based method performances for RNA contact prediction by a well-curated data set." RNA 26.7 (2020): 794-802.

[R8] Multiple sequence alignment (MSA): https://en.wikipedia.org/wiki/Multiple_sequence_alignment

[R9] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017). https://arxiv.org/abs/1706.03762v5

[R10] Chen, Tianqi, and Carlos Guestrin. "Xgboost: A scalable tree boosting system." Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016.

[R11] “Personas, EPICs, User stories” document, UC1 “Agrometeorology” https://docs.google.com/document/d/12EqqZWxQhs7oQtHrHJpQ8SVYDpOVtRWdSWznpGwUmng/edit

[R12] “Personas, EPICs, User stories” document, UC2 “Integrated Plant Protection” https://docs.google.com/document/d/1fMqsjsHcU9KzcH0f09iy825C6VM6sdmFvNNW02CWVyM/edit#

[R13] “Personas, EPICs, User stories” document, UC3 “Automated Thermography” https://docs.google.com/document/d/1SC7jOV7u5FxBgbvPaDk_9--UrAghdO-nId3KGIsm_w4/edit

[R14] “Gathering Technical Aspects” sheet, https://docs.google.com/spreadsheets/d/1t2QAWXmhty_A8y3FSV4fjq48caC8orz7HQ8gf5v0BVc/edit#gid=730060320