RAINBOW

Project Title	AN OPEN, TRUSTED FOG COMPUTING PLATFORM FACILITATING THE DEPLOYMENT, ORCHESTRATION AND MANAGEMENT OF SCALABLE, HETEROGENEOUS AND SECURE IOT SERVICES AND CROSS-CLOUD APPS
Project Acronym	RAINBOW
Grant Agreement No	871403
Instrument	Research and Innovation action
Call / Topic	H2020-ICT-2019-2020 / Cloud Computing
Start Date of Project	01/01/2020
Duration of Project	36 months

D6.1 – Evaluation Framework and Demonstrators Planning

Work Package	WP6 – Demonstrators and Performance Evaluators
Lead Author (Org)	Stefanos Venios (Suite5)
Contributing Author(s) (Org)	George Zafeiris (Suite5); Karthik Shenoy Panambur, Shantanoo Desai (BIBA); Marco Marchetti (CRF); Francesca Pacella (LINKS); Marco Rapelli, Claudio Casetti, Carla-Fabiana Chiasserini, Pramanik Somreeta (POLITO); Wieńczysław Plutecki, Jacek Piotrowski (MSP); George Amponis (K3Y); Moysis Symeonidis (UCY); Panagiotis Gouvas, Konstantinos Theodosiou (UBITECH); Thomas Pusztai (TUW); John Avramidis (INTRASOFT); Raphael Schermann (IFAT)
Due Date	30.06.2021
Actual Date of Submission	30.06.2021
Version	1.0

Dissemination Level

X	PU: Public (*on-line platform)
	PP: Restricted to other programme participants (including the Commission)
	RE: Restricted to a group specified by the consortium (including the Commission)
	CO: Confidential, only for members of the consortium (including the Commission)

European Commission

The work described in this document has been conducted within the project RAINBOW. This project has received funding from the European Union's Horizon 2020 (H2020) research and innovation programme under the Grant Agreement no 871403. This document does not represent the opinion of the European Union, and the European Union is not responsible for any use that might be made of such content.

Versioning and contribution history

Version	Date	Author	Notes
0.10	16.11.2020	Stefanos Venios, George Zafeiris (Suite 5)	Defined ToC
0.20	21.04.2021	Stefanos Venios, George Zafeiris (Suite 5), Karthik Shenoy Panambur, Shantanoo Desai (BIBA), Marco Marchetti (CRF), Francesca Pacella (LINKS), Marco Rapelli, Claudio Casetti, Carla-Fabiana Chiasserini, Pramanik Somreeta (POLITO), Wieńczysław Plutecki, Jacek Piotrowski (MSP)	Added sections 2, 3.1, 3.2 and 4
0.30	17.05.2021	Stefanos Venios, George Zafeiris (Suite 5), Karthik Shenoy Panambur, Shantanoo Desai (BIBA), Marco Marchetti (CRF), Francesca Pacella (LINKS), Marco Rapelli, Claudio Casetti, Carla-Fabiana Chiasserini, Pramanik Somreeta (POLITO), Wieńczysław Plutecki, Jacek Piotrowski (MSP)	Updated sections 2, 3.2.1, 4.2, 4.3 and 4.4.
0.70	08.06.2021	Stefanos Venios, George Zafeiris (Suite 5), Moysis Symeonidis (UCY), Panagiotis Gouvas, Konstantinos Theodosiou (UBITECH), Karthik Shenoy Panambur (BIBA), John Avramidis (INTRASOFT)	Added section 1. Updated sections 2.2.1, 4.1, 4.2.2 and 4.2.3.
0.80	18.06.2021	Stefanos Venios (Suite5), Raphael Schermann (IFAT)	Added Executive Summary and Conclusions (section 5). Updated section 2.2.1 (Security). Sent for internal peer review.
0.90	29.06.2021	Stefanos Venios (Suite 5)	Implemented internal peer review comments. Sent for submission to the EC.
1.0	30.06.2021	Christina Stratigaki (UBITECH)	QA Review; Final Version

Table of Contents

List of tables

List of figures

List of abbreviations

Term	Explanation
ADAS	Advanced Driver-Assistance System
AHED	Automatic Hazardous Events Detection
AI/ML	Artificial Intelligence/Machine Learning
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
CAM	Co-operative Awareness Message
CAN	Controller Area Network
CAP	Collision Avoidance & Prediction
CPU	Central Procesing Unit
DAA	Direct Anonymous Attestation
DENM	Decentralized Environmental Notification Message
DoA	Description of Action
ETSI	European Telecommunication Standards Institute
GCS	Ground Control Station
GPS	Global Positioning System
GUI	Graphical User Interface
HLN	Hazardous Location Notification
HMI	Human Machine Interface
HRC	Human-Robot Collaboration
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IMG	Inertial Motion Generator
IMU	Inertial Measurement Unit
IPC	Industrial PC
ISO	International Standardisation Organisation
KPI	Key Performance Indicator
LTE	Long Term Evolution
MAC	Media Access Control
MEC	Multi-access Edge Computing
MQTT	Message Queue Telemetry Transport
MTOW	Maximum Take Off Weight
N/A	Not Applicable
OBD	On-Board Diagnostics
OPC UA	Open Platform Communications United Architecture
PLA	Programmable Logic Array
PLC	Programmable Logic Controller
PLMC	Personnel Localization and Motion Capturing
PWPG	Personnel Walk Pattern Generator
QoS	Quality of Service
RHS	Road Hazard Signalling
RMT	Robot Motion Tracking
RMPG	Robot Motion Pattern Generator

Term	Explanation
RSU	Road Side Unit
RTSP	Real Time Streaming Protocol
SGW	Secure GateWay
SITL	Software-in-the-Loop
SLO	Service Level Objective
SWAG	Sigmoid Walk Angle Generator
TLS	Transport Layer Security
TPM	Trusted Platform Module
UI	User Interface
URL	Uniform Resource Locator
UWB	Ultra-Wide Band
V2I	Vehicle to Infrastructure
V2N	Vehicle to Network
V2V	Vehicle to Vehicle
V2X	Vehicle-to-Everything
WP	Work Package
3GPP	3rd Generation Partnership Project

Executive Summary

This is the first and pivotal deliverable in the context of WP6 – "Demonstrators and Performance Evaluators", which is responsible for the overall planning and management of the three RAINBOW demonstrators, aiming to evaluate both the scientific innovations and the business value proposition of the RAINBOW solution.
It commences by defining at a high level the evaluation framework, which will assess the actual RAINBOW platform against the objectives targeted, by means of probing and monitoring the impact of RAINBOW in the environments of the three demonstrators. The goal is to be able to systematically improve RAINBOW offerings, detect defects early enough within the project duration and prove that the anticipated impact can be measured and does provide the advertised benefits.
Then, it provides an evaluation methodology based on the well-known V model for the Verification & Validation of software. The evaluation methodology includes indicators to assess the offerings of a technological solution both from a technical excellence and a business impact perspective. Technical evaluation metrics based on ISO/IEC 25010:2011 "Product Quality" model and business impact metrics based on ISO/IEC 25010:2011 "Quality in Use" model are offered.
Following these, the descriptions of the three RAINBOW demonstrators are given. The environments set-up at each demonstrator are both virtual (i.e. set-up consisting of or based on labs infrastructure) and physical (i.e. set-up consisting of actual machinery and network infrastructure). They are described in detail. Then, the way to incorporate the RAINBOW tools and assets in the demonstration environment and prepare the solution for the tests is described. Finally, each demonstrator provides a set of specific business metrics, which measure the business impact as well, but are particular to the business domain and goals of this demonstrator.
In the next chapter, a structured approach is followed in order to ultimately prescribe an implementation plan for each demonstrator. First, the test scenarios are developed; these define the high level goals that each demonstrator has related to the evaluation of the RAINBOW solution. Then, those test scenarios are further fleshed out into test cases. To define the implementation plan, as it stands at M18 of the project and before the start of the evaluation effort, each test case is assigned to one or more test scenarios, and each test scenario is given a due date.
It has to be noted that the evaluation at each demonstrator will be implemented in two phases, which are aligned with the two stable releases of the RAINBOW platform, as foreseen in the project plan. The so-called "Early demonstrator" shall be based on the first release of the RAINBOW platform on M15 and shall conclude on M21; the "Advanced demonstrator" shall be based on the second release of the RAINBOW platform on M27 and shall conclude on M33.

1 Introduction

1.1 Scope and Objectives of the Deliverable

The present deliverable is the result of the work performed in Task 6.1; T6.1 aims to define an inclusive demonstrators' evaluation framework as well as a general guideline document to be used to monitor and align the demonstrators' phases. The goal of the task is to provide valuable remarks and conclusions about the viability and sustainability of the RAINBOW platform. All partners participating in WP6 have collaborated in the compilation of the deliverable; the technical partners who are responsible for the development of the RAINBOW platform have defined the technical metrics and reviewed the test scenarios and test cases of the demonstrators; the demonstrator partners have provided the business-specific KPIs, the descriptions and insights on the evaluation environments, the test scenarios and the test cases that will take place during the execution of the demonstrators.

1.2 Relationship with other RAINBOW WPs and deliverables

The input to "Evaluation Framework and Demonstrators Planning" consists of the requirements definition in WP1 and the technical implementation of the platform in WP2, WP3, WP4 and WP5. Deliverable D1.3 "RAINBOW Use-Cases Descriptions" describes the functionality offered by the RAINBOW platform, while the implementation & integration Work Packages, i.e. WP2, WP3, WP4 and WP5, build two stable releases of the platform, the first on M15 and the second on M27.
The present deliverable shall be used as input for the two main evaluation phases, to ensure that feedback from the evaluation process of the previous version is considered for the next version. Each cycle will conclude with the relative documentation:

- The first validation phase has started already on M15 and will be completed on M21 of the project. The results of this phase will be reported in deliverables D6.2, D6.4 and D6.6. for the three demonstrators respectively and in deliverable D6.8 for the RAINBOW platform as a whole.
- The second validation phase will be starting on M27 and will last till M33 of the project. The results of the second phase will be reported in deliverables D6.3, D6.5 and D6.7. for the three demonstrators respectively and in deliverable D6.9 for the RAINBOW platform as a whole.

1.3 Structure of the Deliverable

The remainder of this deliverable is structured as follows:
Section 2 defines the RAINBOW evaluation framework, a series of coordinated evaluation actions which will be performed in a unified manner across all demonstrators and lays down the evaluation guidelines;
Section 3 describes the three RAINBOW demonstrators and their integration with the RAINBOW platform and defines the execution plan of the evaluation phases at each demonstrator;

Section 4 defines the test scenarios and the test cases applicable to each demonstrator, as well as the implementation plan to be followed by each demonstrator; and
Section 5 draws the conclusions of the present deliverable.

2 The RAINBOW Evaluation Framework

The overall goal of the RAINBOW evaluation framework is to define a series of coordinated evaluation actions which will be performed in a unified manner across all demonstrators; the results shall be to provide the best possible feedback and support to the development of the RAINBOW platform, to ensure its viability and sustainability and ultimately foster its future success. The main aims are: (a) to ensure that the RAINBOW platform is built according to the requirements and generates the expected benefits for both the stakeholders as defined in [1] and the applications they build, and (b) to guide the continuous evaluation of the RAINBOW platform throughout the whole implementation phase of the project from M7 to M27. With the guidance of the evaluation framework, the activities across all demonstrator phases will be monitored and aligned in order to provide structured and actionable feedback to the development team of the RAINBOW platform.
In order to conduct as a holistic evaluation as possible of the RAINBOW platform, two different perspectives have been considered during the definition of the evaluation framework:

- The perspective of the demonstrator: it has been stated already in the DoA [2] that the success of the RAINBOW platform and the project as a whole is closely linked to the successful implementation and execution of the three project's demonstrations, which are expected to play the role of success stories for the project. The evaluation framework will include the demonstrators' perspective in the evaluation in order to assure that the expectations and requirements of the demonstrator-specific stakeholders are fully satisfied. Towards this direction, each demonstrator shall be involved in the definition of demonstrator specific KPIs and the formulation of test scenarios and test cases, as defined in sections 3 and 4.
- An industry generalised perspective: The success of the platform is not only related to the successful execution of the demonstrators, but also lies in fulfilling expectations and requirements of non-demonstrator-specific stakeholders, too. Therefore, the evaluation framework should include an evaluation method mix to enable learning as much as possible from the broader industry (e.g. relevant evaluation questions asked to the community, even at a high level, to get their feeling on what RAINBOW brings to the table).

The following sub-sections present the evaluation framework to be implemented, executed and monitored in the context of WP6.

2.1 The RAINBOW Evaluation Approach

The RAINBOW evaluation framework is based on the principles of the Validation and Verification (V&V) methodologies of software products so as to cover both verification, i.e. the discovery and elimination of defects, gaps in development and possible security

issues, and validation, i.e. the fulfilment of the stakeholders' needs and the generation of the expected benefits.
V&V methodologies, based on the V model approach, cover the whole development cycle of a product based on the active engagement of the project's demonstrators and respective users in the multiple demonstration iterations, exposing them to incremental versions of the platform services and APIs and generating feedback loops, allowing the developers to improve their components and the platform as a whole. The collaborative approach leads to the early diagnosis of defects and gaps in development, the improvement of development teams' performance and efficiency and consequently to cost savings. The application of V&V methodologies addresses (a) whether the software product / platform / system is built right (verification scope), and (b) if the right software product / platform / system is built (validation aspects). According to IEEE Standard 1012-2016[1] definition of V&V:

- Verification is the process of providing objective evidence that the software conforms to requirements (e.g., for correctness, completeness, consistency, accuracy) for all lifecycle activities during each lifecycle process (acquisition, supply, development, operation, and maintenance); meet standards, practices, and conventions during lifecycle processes; and successfully complete each lifecycle activity and satisfy all the criteria for initiating succeeding lifecycle activities (e.g., building the software correctly).
- Validation is the process of providing evidence that the software satisfies system requirements at the end of each lifecycle activity, solves the right problem (e.g., correctly models physical laws, implements business rules, uses the proper system assumptions), and satisfies intended use and user needs.

As such, the evaluation of the RAINBOW platform should be based on the following two questions:

- *Is RAINBOW platform operating according to its specifications?* This question concerns the technical validation of the project and has to be answered by conducting a quantitative technical evaluation, testing technical parameters of system availability, functionality, and performance. The baseline is the platform reference architecture as defined in D1.2 [3] and the documentation of the technical work performed in WP2, WP3, WP4 and WP5.
- *Does RAINBOW meet the defined objectives from the perspective of its users?* This question is closely related to business validation and product validation and for a successful reply to it, the demonstrator partners have to be involved in the different test scenarios. During business validation and product validation, one focuses on aspects like usefulness of the platform, user acceptance, user satisfaction and ease of use. The evaluation framework shall propose a qualitative evaluation approach to shed light on these aspects and create a feedback loop back to the development teams. The baseline is the use cases which have been defined

[1] https://standards.ieee.org/standard/1012-2016.html

in the DoA [2] and further elaborated in D1.3 [4], as well as the stakeholders identified in D1.1 [1].

The following RAINBOW evaluation framework is suggested to enable the success of the platform and to learn as much as possible from demonstrator users.

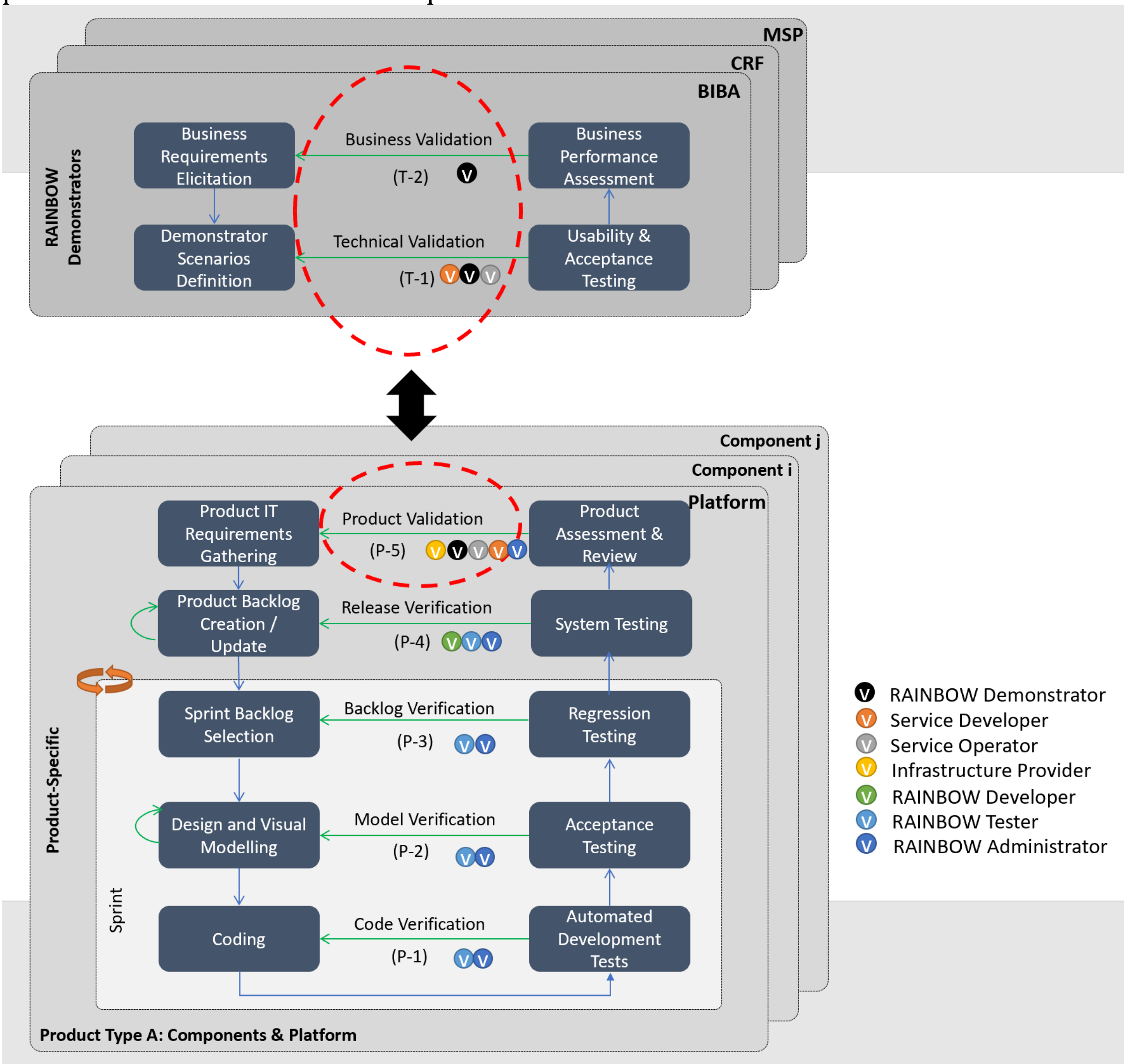

Figure 2-1 RAINBOW evaluation framework

The RAINBOW evaluation framework comprises two core phases, spanning over both the technical and the business perspectives:

- **Product-specific perspective** that concerns the RAINBOW platform and its individual components, including the following steps in an agile development approach: Code Verification (P-1) that ensures functionality, correctness, reliability, and robustness of code; Model Verification (P-2) that aligns design with requirements and design with code; Backlog Verification (P-3) that determines whether the requirements of the product after each sprint are met; Release Verification (P-4) that checks whether the requirements of each product release are met; and Product Validation (P-5) which investigates whether the platform as a whole satisfies intended use and user needs both from a technical and a business view and provides feedback during its operation. It needs to be noted that steps P-1, P-2, P-3, and P-4 are not in the scope of WP6. The project, however, fully covers all aspects of the V model; apart from the thorough collection of requirements and views of users and demonstrators collected in the context of WP1, the RAINBOW Deliverable D5.1 Technical Integration and Testing Plan has documented the integration and testing plan, as well as the code maintenance lifecycle and the continuous integration and delivery framework. Therefore, the RAINBOW evaluation framework in the context of WP6 shall focus mostly on the correct operation of the platform and more importantly on the impact that it generates.

- **Demonstrator perspective** that involves the RAINBOW demonstrators to evaluate the platform and the demonstrators' applications that are created on the platform depending on their scenarios, in the following steps: Technical Validation (T-1) to guarantee that the overall platform satisfies intended use and user needs from a technical and functional point of view only; Business Validation (T-2) to assess whether the overall platform eventually offers sufficient added value and has clear business benefits to the demonstrator, allowing it to operate more efficiently.

2.1.1 Technical Evaluation Background

For the technical validation part, the project has opted to build on top of suggestions provided by IEEE technical validation standards and to identify as a first step which of the proposed metrics should be monitored in order to ensure the smooth and unproblematic operation of the RAINBOW platform, based on its specifications.

The ISO 25010:2011 "Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models" proposes a set of models that better address the evaluation of the software quality. The ISO 25010:2011 standard practically proposes a range of eight (8) main characteristics with a widest range of thirty-one (31) sub-characteristics that capture all the fundamental aspects of a software evaluation. It includes, as stated in its description:

- A product quality model composed of eight characteristics (which are further subdivided into sub-characteristics) that relate to static properties of software and dynamic properties of the computer system. The model is applicable to both computer systems and software products.
- A quality in use model composed of five characteristics (some of which are further subdivided into sub-characteristics) that relate to the outcome of interaction when a product is used in a particular context of use. This system model is

applicable to the complete human-computer system, including both computer systems in use and software products in use.

This model proposes internal and external measures of software quality. Internal measures describe a set of static internal attributes which can be measured. External measures focus more on software as a black box and describe external attributes, which in turn can be measured. The model is applicable to both computer systems and software products.

Figure 2-2 ISO 25010:2011 - Product Quality Model[2]

In general, this model evaluates software quality from the following perspectives:

1. **Functional Suitability** - The degree to which the product provides functions that meet stated and implied needs when the product is used under specified conditions.
2. **Performance Efficiency** - The performance relative to the number of resources used under stated conditions.
3. **Compatibility** - The degree to which two or more systems or components can exchange information and/or perform their required functions while sharing the same hardware or software environment.
4. **Operability** - The degree to which the product has attributes that enable it to be understood, learned, used and attractive to the user, when used under specified conditions.
5. **Reliability** - The degree to which a system or component performs specified functions under specified conditions for a specified period.
6. **Security** - The degree of protection of information and data so that unauthorised persons or systems cannot read or modify them, and authorised persons or systems are not denied access to them.
7. **Maintainability** - The degree of effectiveness and efficiency with which the product can be modified.
8. **Portability** - The degree to which a system or component can be effectively and efficiently transferred from one hardware, software or other operational or usage environment to another.

[2] https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

Table 2-1 shows in detail the sub-characteristics of each category and indicates their suitability to the RAINBOW platform.

Table 2-1 Product Quality Model - Technical characteristics, sub-characteristics and relevance to the RAINBOW platform

Sub-characteristics	Definition	Suitability to RAINBOW platform
Functional Suitability		
Functional completeness	Degree to which the set of functions covers all the specified tasks and user objectives.	High
Functional correctness	Degree to which a product or system provides the correct results with the needed degree of precision.	High
Functional appropriateness	Degree to which the functions facilitate the accomplishment of specified tasks and objectives.	High
Performance Efficiency		
Time behaviour	Degree to which the response and processing times and throughput rates of a product or system, when performing its functions, meet requirements.	High
Resource utilisation	Degree to which the amounts and types of resources used by a product or system, when performing its functions, meet requirements.	High
Capacity	Degree to which the maximum limits of a product or system parameter meet requirements.	High
Compatibility		
Co-existence	Degree to which a product can perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product.	High
Interoperability	Degree to which two or more systems, products or components can exchange information and use the information that has been exchanged.	High
Usability		
Appropriateness recognisability	Degree to which users can recognize whether a product or system is appropriate for their needs.	High
Learnability	Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.	High

Sub-characteristics	Definition	Suitability to RAINBOW platform
Operability	Degree to which a product or system has attributes that make it easy to operate and control.	High
User error protection	Degree to which a system protects users against making errors.	High
User interface aesthetics	Degree to which a user interface enables pleasing and satisfying interaction for the user.	Medium
Accessibility	Degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use.	Low
Reliability		
Maturity	Degree to which a system, product or component meets needs for reliability under normal operation.	High
Availability	Degree to which a system, product or component is operational and accessible when required for use.	High
Fault tolerance	Degree to which a system, product or component operates as intended despite the presence of hardware or software faults.	High
Recoverability	Degree to which, in the event of an interruption or a failure, a product or system can recover the data directly affected and re-establish the desired state of the system.	High
Security		
Confidentiality	Degree to which a product or system ensures that data are accessible only to those authorized to have access.	High
Integrity	Degree to which a system, product or component prevents unauthorized access to, or modification of, computer programs or data.	High
Non-repudiation	Degree to which actions or events can be proven to have taken place, so that the events or actions cannot be repudiated later.	Medium
Accountability	Degree to which the actions of an entity can be traced uniquely to the entity.	Medium
Authenticity	Degree to which the identity of a subject or resource can be proved to be the one claimed.	Medium
Maintenability		
Modularity	Degree to which a system or computer program is composed of discrete components such that a	High

Sub-characteristics	Definition	Suitability to RAINBOW platform
	change to one component has minimal impact on other components.	
Reusability	Degree to which an asset can be used in more than one system, or in building other assets.	High
Analysability	Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.	Medium
Modifiability	Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality.	High
Testability	Degree of effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.	Medium
Portability		
Adaptability	Degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments.	High
Installability	Degree of effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment.	Low
Replaceability	Degree to which a product can replace another specified software product for the same purpose in the same environment.	Low

2.1.2 Business Evaluation Background

Business validation activities record all feedback of the demonstrator partners who use functionality of the RAINBOW platform to develop their demonstrator and verify that the produced software meets the criteria set by the end users at each organisation. In the deliverable at hand, evaluation test-cases are defined by the demonstrator owners which shall be subsequently executed by developers; the results of the execution of those test cases will help the development teams to identify and measure the impact and the usefulness of the platform to the end users. Test cases are in their core short user stories (narratives) developed by the demonstrator owners, describing how they intend to use the functionality of the platform to develop their demonstrator based on concrete requirements set by each organisation. For each of the three different demonstrators a set of such test cases is defined in section 4. According to the maturity of the RAINBOW platform at each testing round, it should provide all required functionality to allow the different stakeholders to run these evaluation cases.

Over the years, a number of frameworks that evaluate usage and impact to software users have emerged: first is the "Quality in Use" model of the ISO 25010:2011 standard that was mentioned in section 2.1.1, second there is TAM2 [5] which is a well-known model for technology acceptance evaluation, third we have HMSAM [6], another framework to evaluate systems that need to engage users, and finally UEQ [7] which is an evaluation framework that focuses on running fast and easy-to-fill, light questionnaires that cover a broad aspect of evaluation criteria.

Table 2-2 Well-known usage and impact to software users' evaluation frameworks considered for RAINBOW

Methodology	Categories	Advantages
ISO 25010:2011 Quality in use model	Effectiveness Efficiency Satisfaction Freedom from risk Context coverage	Strict model that covers many technical aspects from the user perspective.
TAM2	Perceived usefulness Perceived ease of use Intention to use Usage behaviour	Popular and trusted model, can find correlation between user characteristics, intention and actual usage, if the correct answers are selected.
HMSAM	Perceived ease of use Perceived usefulness Curiosity Joy Control Immersion Behavioural intention to use	Related to social networks, media content, and generally projects that need to trigger users' interest
UEQ	Attractiveness Perspicuity Efficiency Dependability Stimulation Novelty	Fast, ready to evaluate, comprehensive, user-friendly

The most important criterion for selecting one of those frameworks as the best fit for the RAINBOW project was previous experience of the majority of the partners with them and their confidence to use them in practice. The preference has been given to the ISO 25010:2011 Quality in Use model which considers the user's point of view to measure the perception of the quality of the system. This system model is applicable to the complete human-computer system, including both computer systems in use and software products in use. The different characteristics and sub-characteristics of this model are derived from testing or observing the results of real or simulated use of the system.

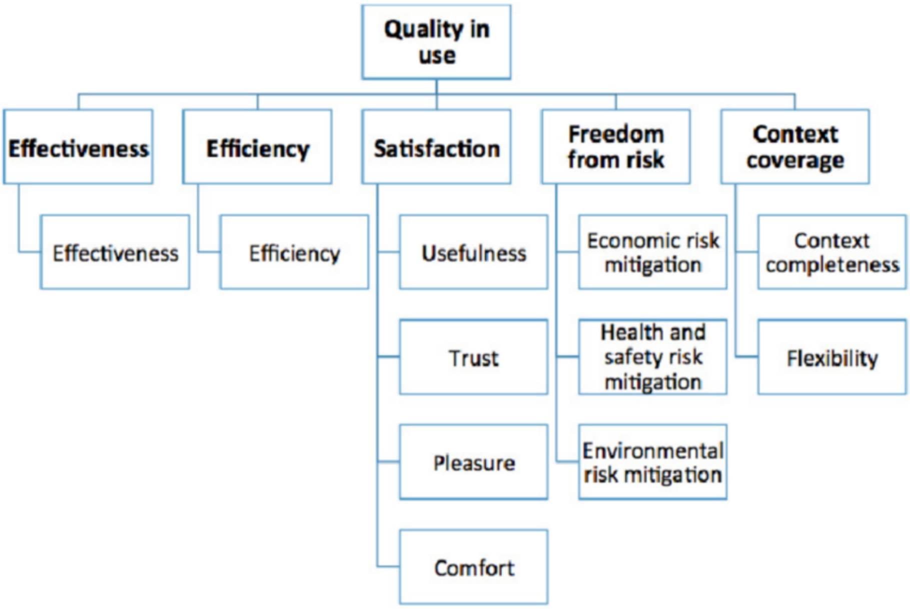

Figure 2-3 ISO 25010:2011 - Quality in Use Model[3]

As with the model selected for technical evaluation, this one assesses software quality (from a user point of view) using the following set of characteristics (each of them including one or more sub-characteristics):

1. **Effectiveness** - The accuracy and completeness with which users achieve specified goals.
2. **Efficiency** - The resources expended in relation to the accuracy and completeness with which users achieve goals.
3. **Satisfaction** - The degree to which users are satisfied with the experience of using a product in a specified context of use.
4. **Safety** - The degree to which a product or system mitigates the potential risk to economic status, human life, health, or the environment.
5. **Context coverage** - The degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in both specified contexts of use and in contexts beyond those initially explicitly identified.

Table 2-3 shows in detail the sub-characteristics of each category and indicates their suitability to the RAINBOW platform.

[3] https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en

Table 2-3 Quality in Use Model - Characteristics, Sub-characteristics and Relevance to RAINBOW platform

Sub-characteristics	Definition	Suitability to RAINBOW platform
Effectiveness		
Effectiveness	Degree of accuracy and completeness with which users achieve specified goals when using the system.	High
Efficiency		
Efficiency	Degree to which the users find that the software is efficiently covering its intended purpose.	High
Satisfaction		
Usefulness	Degree to which a user is satisfied with their perceived achievement of pragmatic goals, including the results of use and the consequences of use.	High
Trust	Degree to which a user or other stakeholder feel that they can trust the system and have confidence that a product or system will behave as intended.	High
Pleasure	Degree to which a user finds the software's functions a pleasure to use (emotionally).	Medium
Comfort	The degree to which users think that the system provides the comforts needed (physically)	Medium
Freedom from risk		
Economic risk mitigation	Degree to which a product or system mitigates the potential risk to financial status, efficient operation, commercial property, reputation or other resources in the intended contexts of use.	High
Health and Safety risk mitigation	Degree to which a product or system mitigates the potential risk to people in the intended contexts of use.	Low
Environmental risk mitigation	Degree to which a product or system mitigates the potential risk to property or the environment in the intended contexts of use.	Low
Context coverage		
Context completeness	Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in all the specified contexts of use	High
Flexibility	Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in contexts beyond those initially specified in the requirements.	Low

2.2 The RAINBOW Evaluation Framework

The standard ISO 25010:2011 was selected as a basis, as it is a well trusted framework that covers both technical aspects and actual usage. While it specifies the evaluation criteria, as presented in section 2.1 above, the specific list of indicators to measure them is left to the adopters. A set of specific KPIs has been devised, which will be used during the operation of the three project demonstrators, in order to identify defects and gaps in the implementation so as to steer the project towards a fully functional, reliable and stable environment that could serve the needs of the business users and have the potential to be exploited as a product in production operation status.
During the development of the evaluation framework, various factors have been taken into consideration. The particular characteristics of the three project demonstrators and their conformity and experience with the methodologies used; the need to evaluate both the technical characteristics of the platform (levels P-5 and T-1 in Figure 2-1) and the real usage through the pilots (level T-2 in Figure 2-1); the use cases which have been elaborated in D1.3 [4] and the stakeholders identified in D1.1 [1]; the platform reference architecture as defined in D1.2 [3] and the documentation of the technical work performed in WP2, WP3, WP4 and WP5.
This section presents the quantitative and qualitative Key Performance Indicators (KPIs) which will be used by the consortium to evaluate the performance of the RAINBOW platform. Following the main directions of the chosen standard, different elements and criteria have been selected and indicators specific to each element have been adapted appropriately to the scope and nature of the project in order to produce an evaluation framework that can be utilised for evaluating each one of project's assets.

2.2.1 RAINBOW Technical Validation - Product Quality Evaluation

Based on the criteria of high importance according to Table 2-1, the following list of technical KPIs has been devised in order to allow the technical assessment of the RAINBOW platform. It needs to be noted that due to the nature of the project and based on the operation conditions of the pilots, some of the below-mentioned indicators are considered optional, as their measurement might not be possible/not producing meaningful results.

Table 2-4 Technical Validation - Quantitative Evaluation Metrics selected for the RAINBOW platform

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
Functional Stability				
Functional completeness	Portion of completed requirements	(Completed requirements / Iteration Cycle requirements) * 100 %	100%	M (at the end of the cycle)
Functional correctness	Portion of requirements	(Completed requirements	>90%	M

Sub-characteristics	**KPIs**	**Calculation Type**	**Recommended Limit**	**Mandatory / Optional**
	without reported bugs, after tests	without bugs / Iteration Cycle User Stories) * 100 %		
Performance Efficiency				
Resource utilization	Mean % CPU Utilization	(CPU time in ns used by RAINBOW sidecar / overall available CPU time) * 100 %	<45%	M
	Mean % Memory Utilization	(RAM MB used by RAINBOW sidecar / overall available RAM MB) * 100 %	<45%	M
	Mean % Network Utilization	(Network received and transmitted bytes by RAINBOW sidecar / available bandwidth) * 100 %	<45%	M
	Mean % energy utilization	(Energy consumption occurred by RAINBOW sidecar / fog node's idle energy consumption) * 100 %	<45%	O
Time behaviour	SOMS device verification	Latency of the device verification process upon boot time	<1s	M
	Pod placement time	Pod creation timestamp –	<3s	M

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
		pod is scheduled timestamp		
	Cluster head election time	Starting process timestamp – timestamp of the new cluster head is elected event	<5s	O
	Monitoring API response time for current metric extraction	(Total Response Time) / (No. of Requests)	<=0.5s	M
	Monitoring API response time for historical data extraction	(Total Response Time) / (No. of Requests)	<=1s	O
	Detection time of SLO violation	timestamp of the violation event – event detected by the system timestamp	<20s	M
Capacity	Analytics engine max throughput	(Max No of datapoints / seconds) / No of Nodes	>5000	M
Compatibility				
Co-existence	Ability to Co-Exist	Can the RAINBOW platform operate in a shared environment? YES/NO	YES	O
Interoperability	Ability to Operate on various architectures	Can RAINBOW platform operate on edge devices with different architectures? (ARM, x64, etc.) YES/NO	YES	M

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
	% API coverage	(No. of Systems with API)/ (No. of Integrated Systems) * 100%	>95%	M
Usability				
Appropriateness recognisability	% Positive feedback on questionnaire	(No. of Positive opinions / No. of answers) * 100%	>70%	O
Learnability	% Coverage of Features with Learning documents	(Unique No. of help documents mentioning a feature / No. Of total Features) * 100%	50%	M
	Platform walkthrough availability	Is there a walkthrough available? YES/NO	YES	M
User error protection	System crash on user errors	Does the whole crash on user errors? YES/NO	NO	M
	% Coverage of input fields with error protection methods	(No. of error protected input fields / No. of total critical input fields) *100%	>80%	M
	Error message availability	Is there a comprehensive information message on errors? YES/NO	YES	M
	Warning message availability	Is there a warning message on critical actions? YES/NO	YES	O
	Bug report availability	Is there a way to report a bug?	YES	O

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
		YES/NO		
User interface aesthetics	Responsiveness	(No. of broken screens / No. of different screens) * 100%	<20%	M
Accessibility	Availability of Manual	Is there a RAINBOW manual available? YES/NO	YES	O
	Colour Blindness	Are colours compliant[4]? YES/NO	YES	O
Reliability				
Maturity	Max. Concurrent Users Supported	No. of Max. Concurrent Users Recorded	>100 users	M
	Simultaneous Requests	No. of Simultaneous Requests	>100	O
Availability	% Monthly Availability	1 - ((Downtown Time Minutes) / (Month Days*24*60))	>80%	M
	Error Rate	(No. of Problematic Requests / Total Number of Requests) * 100%	<10%	M
Fault tolerance	Number of Software problems identified without affecting the platform	No. of Non-Critical Software Errors	<20	M
	Number of Hardware problems identified without affecting the platform	No. of Non-Critical Hardware Errors	<10	M

[4] http://www.snook.ca/technical/colour_contrast/colour.html#fg=6CFF33,bg=333333

Sub-characteristics	**KPIs**	**Calculation Type**	**Recommended Limit**	**Mandatory / Optional**
Recoverability	Mean time to recover from software problems	(Total Recovering Time due to Software Issues) / (Total Software Issues resulting to recovery)	<2h	M
	Mean time to recover from hardware problems	(Total Recovering Time due to Hardware Issues) / (Total Hardware Issues resulting to recovery)	<24h	M
Security[5]				
Confidentiality	Secret information is protected from unauthorized disclosure	Number of recorded incidents	0	M
Integrity	Data cannot be modified in an unauthorized manner since it was created, transmitted or stored – Incidents of integrity/authentication breaches	Number of recorded incidents	0	M
Non-repudiation	Assurance that the owner of a signature key pair that was capable of generating an existing signature corresponding to certain data cannot	(Number of log reports / number of all system operations)*100%	>80%	O

[5] The security goals are estimated values. These values cannot be established in detail and further not evaluated in fine granularity. Here, a threat and risk analysis need to be performed. Additionally, the whole environment/setup must be pen-tested by an own pen-tester team.

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
	convincingly deny having signed the data – Log reports for activities			
Accountability	Assurance that the owner of a signature key pair that was capable of generating an existing signature corresponding to certain data cannot convincingly deny having signed the data – usernames included in the log entries	YES/NO	YES	O
Authenticity	Property of being genuine and being able to be verified and trusted	Identification of a subject or/and resource that it claims to be – YES/NO	YES	O
Maintainability				
Modularity	% Modularity (excluding backbone infrastructure)	(Number of microservices / Total number of components across all platforms) * 100%	>40%	O
Reusability	% of Reusable Assets	(No. of assets that be reused as is / Total number of assets) * 100%	>50%	O
Analysability	Level of Analysis	Can the changes in performance of the RAINBOW platform be	YES	O

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
		measured after each upgrade? YES/NO/Partially		
Modifiability	% of Update Effectiveness	(No. of updates performed without noticing operational problems / No. of updates performed) * 100%	>40%	M
Testability	Level of Testing	Are tests able to probe the RAINBOW platform behaviour? YES/NO/Partially	YES	M
Portability				
Adaptability	Mean No. of Errors per Hardware Change	(No. of Total Errors recorded) / (No of Total Hardware Changes)	<2	O
	Mean No. of Errors per Software Change	(No. of Errors recorded) / (No of Software Changes)	<2	O
Installability	Mean Installation Duration	(Total minutes recorded for installation) / (Total No. of Installations)	<240min	M
	% of Installation Errors	(No. of Installation containing Errors / Total No. of Installations) * 100%	<1%	M

Sub-characteristics	KPIs	Calculation Type	Recommended Limit	Mandatory / Optional
	Mean No. of Errors per Installation	(No. of Total Errors recorded during Installations) / (Total No. of Installations)	<2	O
Replaceability	Number of components that can replace existing RAINBOW components	(No. of products that can replace RAINBOW components / Total No. of distinct components) * 100%	>20%	O

2.2.2 RAINBOW Business Validation

2.2.2.1 Users' Evaluation

The users' evaluation part depends mostly on meeting criteria set by users, which will help them to identify and measure the impact and the usefulness of the RAINBOW platform in their operations. It starts from the test scenarios that have been described by and shall be performed at each of the three project demonstrators, which are further distilled in test cases (see section 4). The test scenarios stem from the use cases which have been elaborated in D1.3 [4] and the stakeholders identified in D1.1 [1].
During business validation, other significant characteristics of the platform which are related to its success and future traction are also measured as behavioral aspects of the users interacting with the platform. An exemplary workflow for running the behavioral, controlled tests is the following:

1. **Give a short description**: Users should land on the platform with a short description on what they can do with it. Detailed characteristics should not be disclosed, letting them free to explore what the platform does.
2. **Let them explore**: Users are free to navigate the platform and evaluate how well they perform each user story.
3. **Trigger them to explore more**: For stories not achieved by the users, the goal is given to them and their behavior is observed, e.g., "How would you create a new account, where would you look at?"
4. **Document failed stories**: Ask the users why they believe they failed to find any stories, even after help was provided. An informed user of the platform may present them the story, to see later if they like it or not.

5. **Report bugs**: Defects and bugs are written down, given the context where they occurred. Users should not be annoyed with bug reporting, e.g. asking them to capture a screenshot before continuing.
6. **Collect demographics**: Basic information about the user is collected. This step may take place in position 1 ahead of everything else.
7. **Run Questionnaire**: The answers to the created questionnaire are collected.
8. **Tell the complete story**: A more detailed view on what the platform will deliver, and what it aspires to achieve is given to the participant.
9. **Collect general feedback**: General feedback on the platform is collected based on the discussion that took place.

For each demonstrator, such evaluation information will be gathered in the context of tasks T6.2, T6.3 and T6.4 and documented in deliverables D6.2 through to D6.7. Table 2-5 presents the KPIs which correspond to the evaluation of the pilots' operation phase. KPIs are measured in a qualitative manner, either by measuring AS-IS and TO-BE values, or by using a Likert 1-5 scale.
During the users' evaluation, RAINBOW project aims to identify the customer segments that are most interested in the platform. According to TAM2, the user context differs each time during the evaluation process, thus the results should be examined for correlation based on the following user characteristics:

- **Experience**: how many years the user has been doing this job, or using similar systems;
- **Image**: how influential the user is considered;
- **Job Relevance**: the relevance of the user's job with the platform under test;
- **Output Quality**: how the user perceived the quality of the output in total, whether s/he is interested in the platform or not; and
- **Result Demonstrability**: if the user is willing to show the output to a peer or colleague.

For the RAINBOW platform business users and software teams of the demonstrator operators, a correlation with the users' characteristics can help the project identify what are the target segments of the platform.

Table 2-5 Business Validation - Evaluation Metrics selected for the RAINBOW Demonstrators Operation

Sub-characteristics	Related Questions
Behavioural Characteristics	
Effectiveness	
Effectiveness	**Do you think that RAINBOW platform increases the productivity of your business applications compared to the current status?** *Scale 1-5*

Sub-characteristics	**Related Questions**
	Do you think it is easier to have access to the usage of more advanced technologies via the RAINBOW platform compared to the current status? *Scale 1-5*
Efficiency	
Efficiency	**Do you think the RAINBOW platform covers its advertised purpose?** *Scale 1-5*
Satisfaction	
Usefulness	**Can you easily complete your business goals using the RAINBOW platform?** *Scale 1-5 (e.g. not at all / partially / most of them /almost all / all of them)*
Trust	**Do you trust the data and services (e.g. analytics results) relying on the RAINBOW platform?** *Scale 1-5*
Pleasure	**Do you use the RAINBOW platform with pleasure?** *Scale 1-5*
Comfort	**Do you feel that the UI and workflow of the RAINBOW platform are friendly to the user?** *Scale 1-5*
Safety	
Privacy guarantees	**How solid do you feel that your data and services relying on the RAINBOW platform are protected?** *Scale 1-5*
Trust feeling	**Do you believe that services based on the RAINBOW framework are trustworthy?** *Scale 1-5*
Usability	
Learnability	**How easy it was for you to learn how to use basic functionalities of the RAINBOW platform?** *Scale 1-5*
Flexibility	**How much do you believe that the RAINBOW platform can be used for applications other than the demonstrator ones?** *Scale 1-5*
Accessibility	**Do you believe that the RAINBOW platform can be accessed by disabled users (e.g. visual or hearing impairment)?** *Scale 1-5*
Business Value	
Clarity	**How clear it was to you what RAINBOW is about before engaging with the platform?** *Scale 1-5*

Sub-characteristics	Related Questions
Value	**How much do you feel that use of the RAINBOW platform increases the value of the product of your business?** *Scale 1-5*
Need Level	**How important is for your organisation the business need that the RAINBOW platform covers for you?** *Scale 1-5*
Urgency	**How soon after the end of the project do you expect RAINBOW to be fully functional?** *Scale 1-5*
Need Coverage	**In which degree does RAINBOW covers your need?** *Scale 1-5*
Innovation/Uniqueness	**How innovative do you find the idea of RAINBOW?** *Scale 1-5*
Virality	**How probable is it for you to recommend the use of RAINBOW platform to someone you know who works in the same domain as you?** *Scale 1-5*

2.2.2.2 RAINBOW Business KPIs Evaluation

RAINBOW platform shall contribute to different performance indexes of business interest, from direct costs (and time therefore) to other aspects such as perceived Quality of Service. Together with the demonstrator specific KPIs, to be defined in section 3, the following generic Business KPIs are proposed to be used to measure the benefits offered by the platform. It needs to be mentioned, that all KPIs presented in Table 2-6 might not be measurable simultaneously in every demonstrator. They should be considered as baseline KPIs to demonstrate the impacts of RAINBOW platform, when measured in a comprehensive and reproducible environment (see section 2.3.2).

Table 2-6 RAINBOW Generic Business KPIs

ID	Business Metric	Units	Description
RAINBOW-KPI-01	Deployment Time	minutes	The time it takes to deploy a new instance of the application in the RAINBOW infrastructure.
RAINBOW-KPI-02	Software Delivery Cycle	minutes	How long it takes to deliver a change in the software into production.
RAINBOW-KPI-03	Security Incidents	No. of Security Incidents / time	Number of security incidents recorded per unit of time.

ID	Business Metric	Units	Description
RAINBOW-KPI-04	Service Availability	Time services is up / Total time	Percentage of time the system is up and running.
RAINBOW-KPI-05	Cost Efficiency	number of HTTP requests faster than N ms / total cost of service in €	Service performance per unit of cost.
RAINBOW-KPI-06	Cloud Infrastructure Costs (OPEX)	€ / time	Total Cloud Infrastructure Cost (including edge/fog nodes, etc.) for running the service per unit of time.
RAINBOW-KPI-07	Energy Consumption Costs	€ / time	The cost for the energy needed for running the service per unit of time.
RAINBOW-KPI-08	User Satisfaction	% Perceived satisfaction of the customer	Distilled out of questionnaires that measure user satisfaction for the quality of the service
RAINBOW-KPI-09	Investments for developing fog computing services	Person-months	Number of person-months estimated for developing a system

2.3 Evaluation Guidelines

The RAINBOW evaluation framework has the dual goal of **verifying** the technical characteristics of the platform and evaluating the product quality, as well as **validating** the qualitative characteristics of the platform through realistic test scenarios performed in the context of the three demonstrators. In the current chapter, first we present the combination of the evaluation framework with the software development activities and explain where evaluation is positioned in the development lifecycle. Possible particularities in RAINBOW evaluation cycles are also presented and explained. Then, we describe a specific approach to ensure reproducibility in the demonstrators' testing conditions across two testing cycles and three different demonstrator set-ups, towards ensuring that objective measurements are collected during our tests. Finally, we describe the demonstrators, their particularities and their intended way to make use of the RAINBOW assets. Apart from targeting the technical excellence of the platform, testing activities shall also record the impact of the platform's innovations on our three demonstrators, through the definition (in the present deliverable) and measurement (in upcoming deliverables) of the demonstrator-specific KPIs.

2.3.1 RAINBOW Platform's Evaluation Execution Plan

A unified approach shall be employed to align the technical and the business evaluation of the platform towards producing meaningful results that can relate to each other and drive forward the necessary improvement activities from the development teams. The technical verification shall be performed in the technical development work packages and then the business evaluation and demonstrator-specific testing shall follow as suggested in Figure 2-4.

When the development of the first version of the RAINBOW platform is concluded on M15, with all the required unit tests and technical verification methods successfully executed during software development, it is time to test and validate by the platform users. Users' evaluation quality tests based on the "Quality in Use" model are performed to qualitatively evaluate the platform. Further defects, which the automated or simulated technical tests during software development might not capture, shall surface during the utilization of the platform from users; usefulness and ease of use defects typically belong here. Having concluded the use model quality tests, the demonstrator-specific tests should run; knowledge related to technical limitations of the platform may be already available already from the initial technical tests, e.g., the server may have crashed in extreme usage scenarios.

All tests should be concluded and documented properly, and then the responsible technical partners should come together to discuss the results. Initially, they should decide, based on received feedback, if the developed platform is actually a stable version that meets its goals and has acceptance in a group of users. If the version under testing is indeed validated, prioritization on improvements should be given. If not, the development team should reschedule, as soon as possible, deliveries of requirements and experiment with new ones in order to try to come up with a stable version covering the minimum functionality; in that case only blocking and crucial technical issues should be scheduled for fixing. Development moves forward, and new iteration is available.

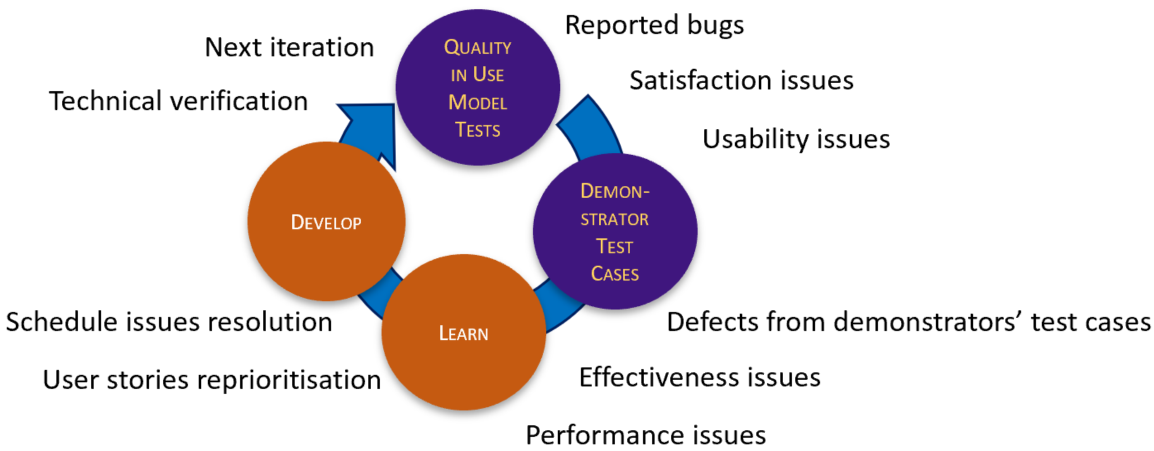

Figure 2-4 Combination of the RAIBOW evaluation framework with software development lifecycle

RAINBOW runs in two main iterations, therefore the overall validation and evaluation activities will be documented in two phases as described also in the DoA [2]. In Figure

2-5 we showcase how the evaluation cycle will run in each iteration of the platform development, to ensure that feedback from the evaluation process of the previous version is considered fort the next version. Each cycle will conclude with the relative documentation:

- The first validation phase has started already on M15 and will be completed on M21 of the project. The results of this phase will be reported in deliverables D6.2, D6.4 and D6.6. for the three demonstrators respectively and in deliverable D6.8 for the RAINBOW platform as a whole.
- The second validation phase will be start on M27 and will last till M33 of the project. The results of the second phase will be reported in deliverables D6.3, D6.5 and D6.7. for the three demonstrators respectively and in deliverable D6.9 for the RAINBOW platform as a whole.

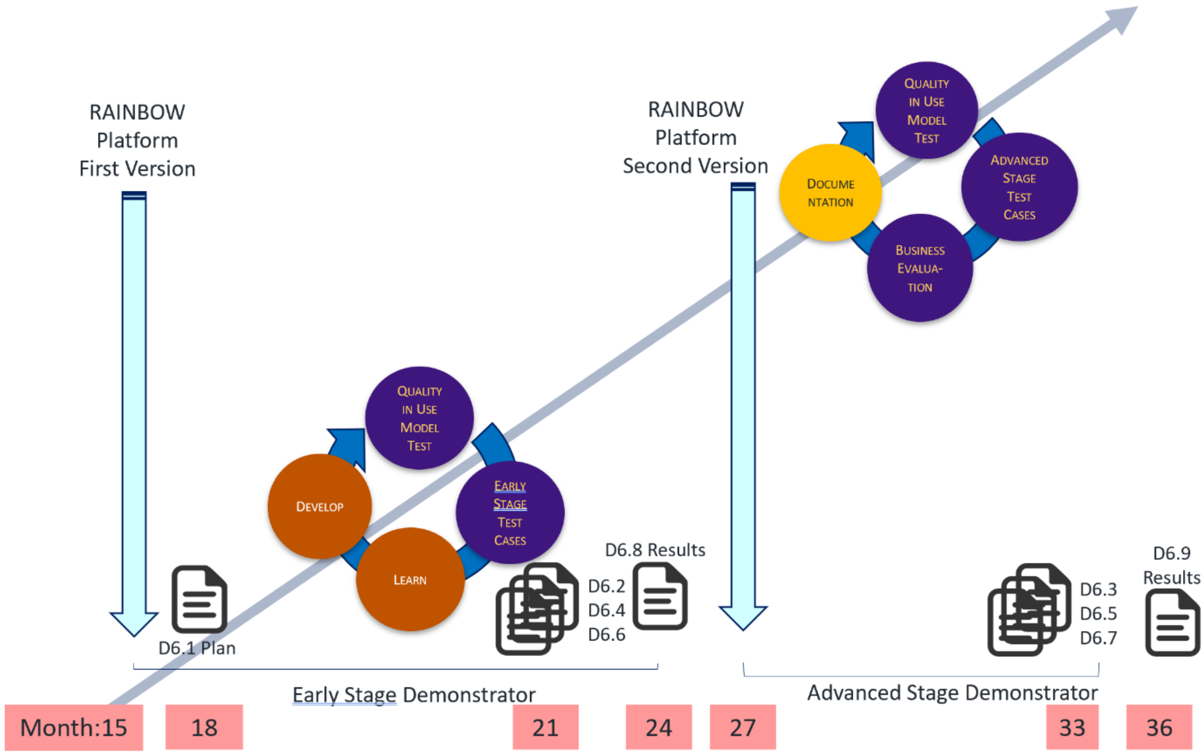

Figure 2-5 RAINBOW evaluation cycles

2.3.2 Ensuring Reproducibility in the Demonstrators' Testing Conditions

Reproducibility is the term encompassing the steps taken to guarantee that a test method is able to replicate the same conditions for acquiring results, at the same granularity and while being comparable across separate test runs. We plan to employ it in our project as a first step in the evaluation to specify the conditions that will allow to properly measure the impacts of the platform across the three different demonstrators.
As explained above, the impacts are measured by a set of KPIs, which compare an existing (AS-IS) situation and a new (TO-BE) situation enabled by RAINBOW. In order to guarantee equal terms between the comparison of those results, and extract as many

objective readings as possible, there is a need to specify the preconditions that should be present in both those situations, thus creating a "reproducible" environment of the existing (AS-IS) situation, where the KPIs of the new (TO-BE) situation enabled by RAINBOW can be measured.
The suggested approach proposes the most essential aspects that should remain stable when trying to measure specific indicators between an existing (AS-IS) situation and a new (TO-BE) situation. The "Execution Phase" considers the operation of an application/service. The focus should remain on the soft and hard infrastructure being used, as well as the operational conditions, which are the main dimensions that should be kept stable when trying to reproduce the conditions that led to a previous result.
As such, the following checklist allows for the identification of some existing conditions that apply to the existing (AS-IS) situation, which need to be also present in the new (TO-BE) situation enabled by RAINBOW, without allowing modification.

Table 2-7 Checklist for Trials Preconditions

Item	Traditional case	RAINBOW powered case	Notes
Project Team Characteristics			
Project Leader (Name)			*Same Project Leader is Recommended*
No. of Operators			
No. of same Operators present in both situations			*100% match of team is recommended*
Distribution of Team's skills			
No. of Junior Operators			*Same No. of Junior operators recommended*
No. of Medium-Skilled Operators			*Same No. of Medium-Skilled operators recommended*
No. of Expert Operators			*Same No. of Expert operators recommended*
Effort Dedicated			
Total Effort of Junior Operators			
Total Effort of Medium-Skilled Operators			
Total Effort of Expert Operators			
Execution Phase			
Platform Conditions			
Cloud Provider / VM characteristics			*Same provider/VM recommended*

Item	Traditional case	RAINBOW powered case	Notes
Type of Cloud used (Public/Private/Hybrid)			*Same Cloud Type recommended*
Network Infrastructure Characteristics			
Operational Conditions			
Type of Load			*Use the same type of load for the tests or, if possible, a reusable load scenario.*
Max Load Recorded (per Timeframe)			
Min Load Recorded (per Timeframe)			
Average Load (per Timeframe)			

3 Demonstrators' Evaluation Execution Plan

In addition to the business KPIs in section 2.2.2, each demonstrator operator shall define their own list of KPIs to better measure the business impact of the RAINBOW solution in the context of their implementation. The KPIs in section 2.2.2.2 above should be considered the "RAINBOW KPIs list", while the KPIs created by the demonstrators below should be considered the "Business Case Specific" ones.
This set of KPIs shall be recorded using the following template that defines each KPI and why this is business-critical for each use case.

Table 3-1 Business KPIs Identification Template

#	Values
ID	#id
KPI Title	KPI Title
KPI Type	RAINBOW KPIs list / Business Case Specific
Objective / Subjective	Objective when measured with data that are objective / Subjective when measured though subjective data or estimations
Need Relevant to KPI	Description of the Business Need that is relevant to the KPI
RAINBOW's Contribution	How RAINBOW serves towards covering this need
KPI Measurement Formula	Formula for measuring the KPI (or simple metrics to be measured)
Current Value (without RAINBOW)	Value of the AS-IS Scenario. If not measurable, say why, or provide estimation
Expected target value (with RAINBOW)	Estimated target in the RAINBOW powered scenario

3.1 Evaluation in Demonstrator #1 – BIBA

3.1.1 Physical Demonstrator

At a high-level, an HRC system is a collision prediction and avoidance system aimed at reducing risk of accidents involving Personnel and Robots in an indoor environment. The following information is required continuously in a time-deterministic manner:

1. Personnel's current 3D Coordinates and motion dynamics
2. Robot's current 3D Coordinates and motion dynamics

Using the above information, predictions on collision are made a-priori. Based on the probability of collision, the collision prediction and avoidance system sends control messages to slow down or stop the Robot, thus avoiding the collision between Personnel and Robot.
The demonstrator setup for RAINBOW evaluation is shown in Figure 3-1**.** The demonstrator consists of 2 workplace areas: "Area-1" and "Area-2". Each of the work-

place areas consists of a robotic arm controlled by an industrial PC and a PLC using PROFINET field bus. An IoT Gateway collects the telemetry data of the Robot (joint angles, velocity, etc.) from the industrial PC using PROFINET and forwards this data to RMT service running on fog devices.
Additionally, if a collision is predicted by CAP services running on a fog device, then a control signal is sent to the IPC via the IoT Gateway for stopping or slowing down the Robot.
On the other hand, Personnel localization and motion dynamics data from mobile node device units are received by the data aggregator wirelessly over UWB. The received data is published to MQTT broker hosted within the data aggregator. PLA services running on fog devices subscribe to this data using MQTT Client.

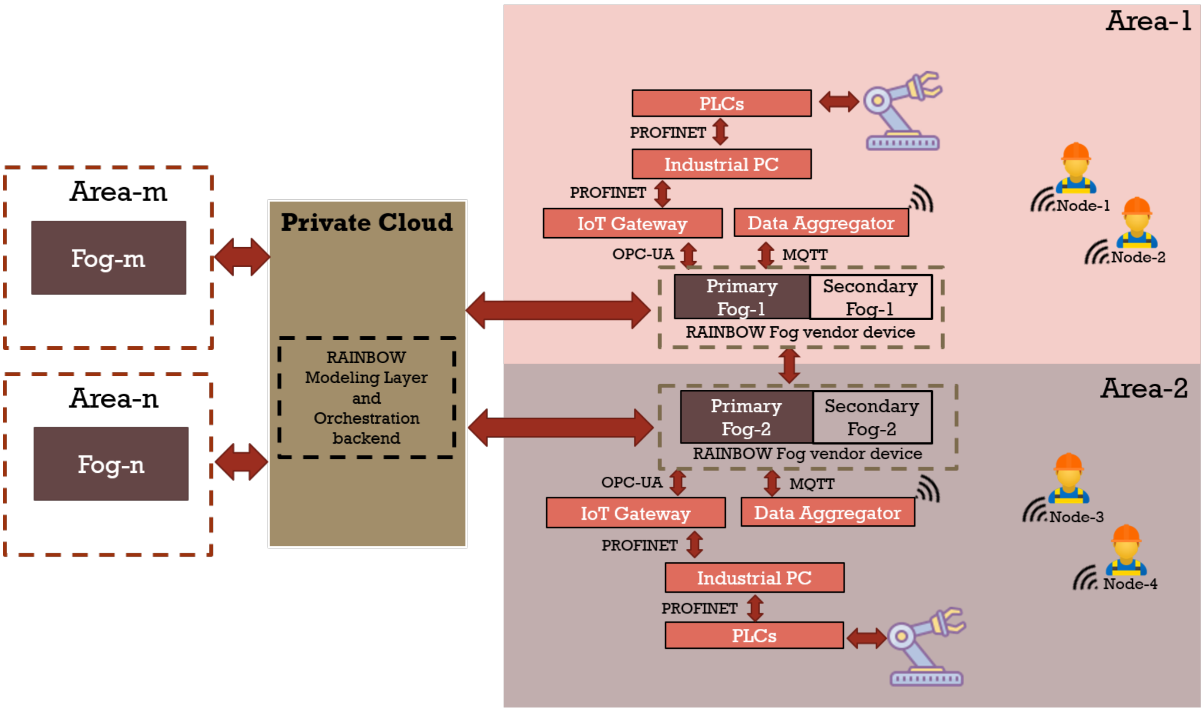

Figure 3-1 BIBA physical demonstrator setup for RAINBOW evaluation

Below is the description of each block represented in Figure 3-1:

Industrial robotic arm

Used for lifting heavy parts for assembly (e.g., transformer). The robotic arm can be either manually guided or be programmed to carry out particular tasks.

Industrial PC (IPC) and Programmable Logic Controller (PLC)

Industrial PC serves as the control unit and communicates control signals to the robot arm via PLC. In the particular demonstrator, the industrial PC will be from Siemens and runs applications on Microsoft Windows Operating system. Here, PROFINET field bus is used for communication.

IoT Gateway

Acts as an adapter to communicate information between IPC (using PROFINET) and CAP services (using MQTT, OPC UA).

Node

Each Personnel member on the shop floor carries a node device unit all the time. A node device essentially provides instantaneous 3D position coordinates and motion dynamics of personnel. Each node device consists of motion sensors, Ultra-Wide Band (UWB) tag/s, a microcontroller unit and a battery pack to support mobility as shown in Figure 3-2. Typically, these node devices are embedded in the personnel's vest. Motion sensors capture the motion dynamics of the personnel. UWB tag/s capture the 3D coordinates of personnel with respect to the stationary UWB anchors (stationary device mounted on walls, roof, or any fixed support structure in the area). The microcontroller performs digital signal processing (filtering) on extracted data from motion sensors and tag/s before sending the processed data to the data aggregator.

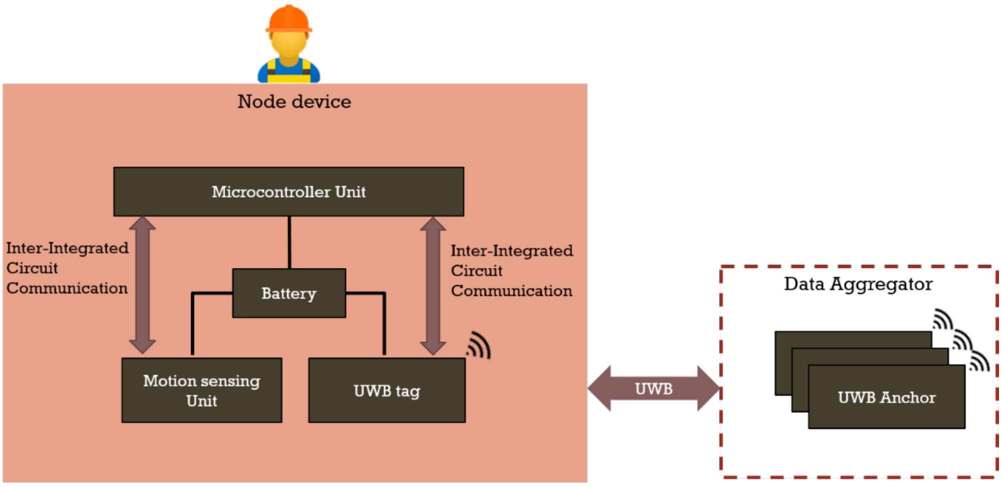

Figure 3-2 Node device description

Data Aggregator

Primary purpose of the data aggregator is to receive telemetry data from multiple node devices and send these received data to subscribed devices using MQTT (Message Queue Telemetry Transport) with TLS enabled. A work-place area can contain one or more data aggregators. The actual number of data aggregators is decided based upon the maximum number of personnel allowed in work-place area simultaneously.

Node devices uses UWB for distance ranging and telemetry, due to superior channel characteristics in noisy industrial environment. As UWB physical layer is not inherently supported by IoT Gateway or fog devices, there is the need of a data aggregator as an interface between node devices and IoT Gateway/fog devices. The data aggregator consists of UWB Anchors (distributed over the workspace) and an MQTT Broker as shown in Figure 3-3.

The UWB anchor serves a two-fold purpose: first, it acts as a reference beacon in distance ranging for node devices in a workspace area. Second, it receives the telemetry messages containing motion sensing and 3D coordinates information from all node device units. These received messages are published to the local MQTT broker, present in the same data aggregator device. The MQTT broker, in turn, provides the telemetry data to the

MQTT client (typically running on other devices such as IoT Gateway/fog device) subscribed to relevant MQTT topics.

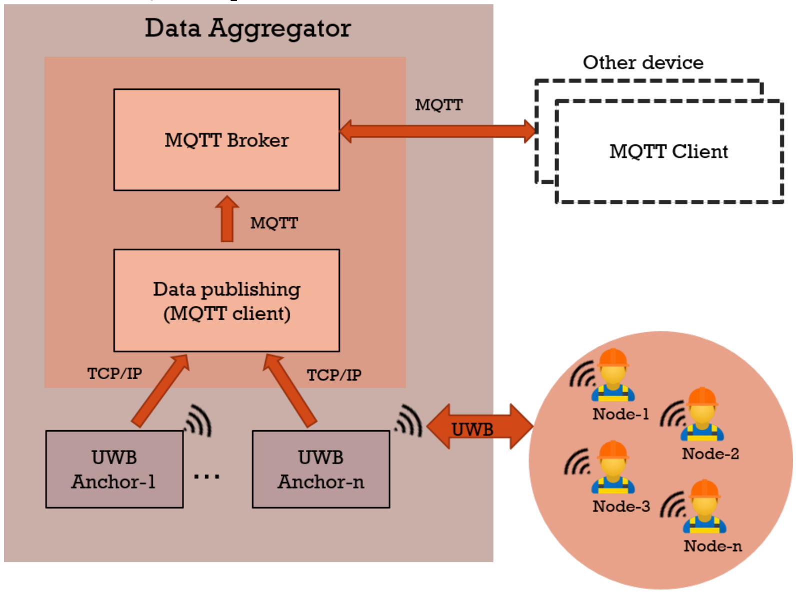

Figure 3-3 Data Aggregator description

Fog device

It is a high-performance 64-bit multi-core processor hardware with Linux OS capable of running multiple instances of each of the below services as shown in Figure 3-4:

1. Personnel Localization and Motion Capturing service (PLMC);
2. Robot Motion Tracking service (RMT); and
3. Collision Avoidance and Prediction service (CAP).

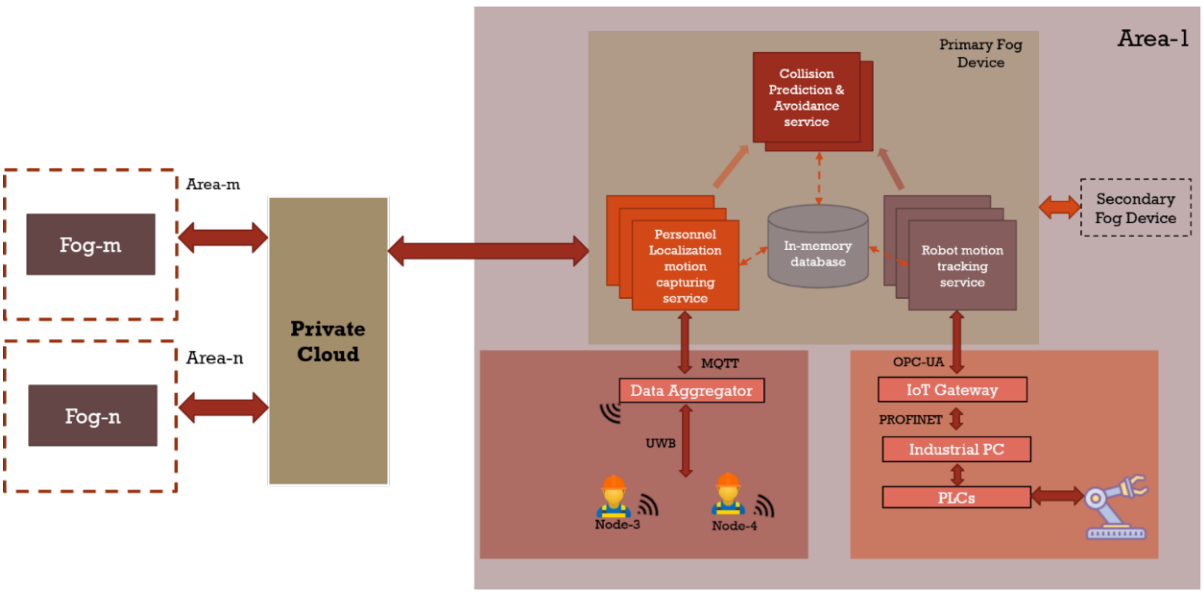

Figure 3-4 Fog device description

Personnel Location and Motion Capturing Service (PLMC)

This service provides optimal estimates of personnel's 3D coordinates and predicts their future motion trajectory time ahead and with a certain confidence level in different regions. Figure 3-5 shows different components of a PLMC.

One instance of this service is assigned to exactly one Personnel member in the workspace. This service provides the following information:

- Optimal estimate of personnel's instantaneous position
- Estimate of personnel's future position/region of presence, ahead of the actual time when the person will be in the position/region
- Monitor Node device QoS (Quality of Service) Parameters

The data obtained from localization sensing are often noisy and are affected by environmental factors such as interference, multi-path fading, etc. Noise interferences are unknown and are stochastic in nature. Extracting the exact measurement from a noisy measurement is not feasible. Instead, it is possible to acquire an estimate of measurements at a given time. To obtain the best possible estimates of coordinates we use Coordinate Estimation Algorithms, for example Robust Adaptive Linear Quadratic estimator, multi-model adaptive Kalman filters, Marginalized-Particle filter, etc. Adding to this, using Motion Predict Algorithm it is also feasible to predict future occupancy coordinates/regions time ahead with a certain confidence. These algorithms are computationally intense and are required to run in hard real-time constraints. Thus, there is the need to run these algorithms on powerful multi-core processors. Both Personnel coordinates estimates and predicted occupancy coordinates are updated in an in-memory database and are published to the subscribed Collision Avoidance and Prediction services.

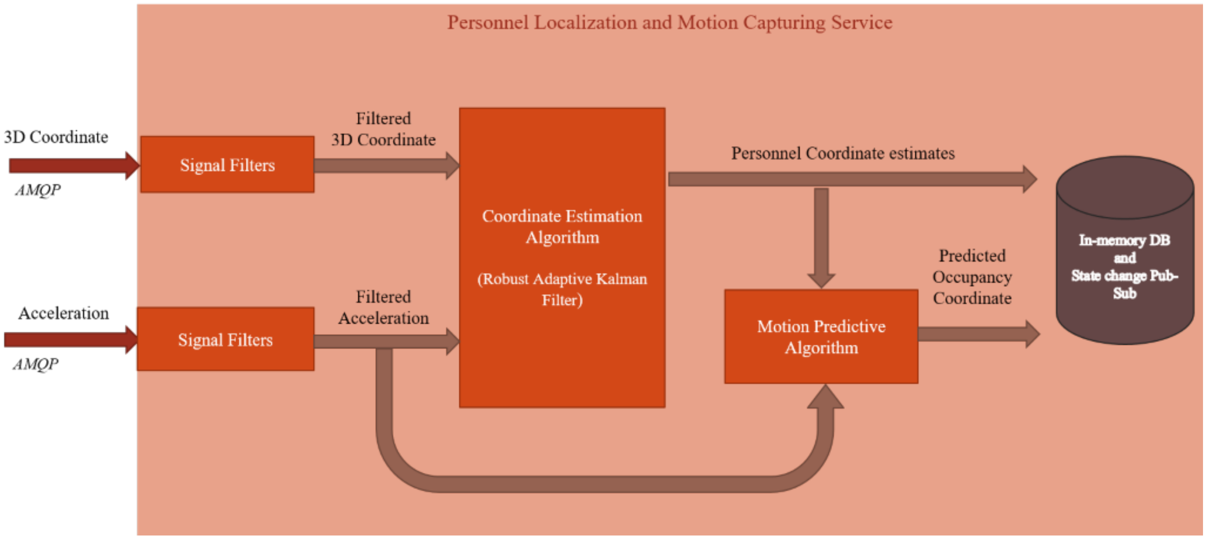

Figure 3-5 Personnel Location and Motion Capturing Service (PLMC) components

Robot Motion Tracking Service (RMT)

This service tracks robot arm movement and also provides future motion path. Figure 3-6 shows the different components of RMT.

One instance of this service is assigned to exactly one Robot in the workspace. The service provides the following:

- Instantaneous 3D Coordinate of the robot joints
- Future motion path of robot joints ahead of time

As the industrial PC provides instantaneous joint angles of the robot arm, this service performs Forward Kinematics operations to extract the 3D coordinate position of the end-effector. Since the Robot motion planning unit knows ahead of time about future motion attributes (like joint angles) of the robot. These attributes are obtained to predict future coordinates/regions of presence. Both current and next end-effector coordinates are updated in an in-memory database and are published to the subscribed Collision Avoidance and Prediction services.

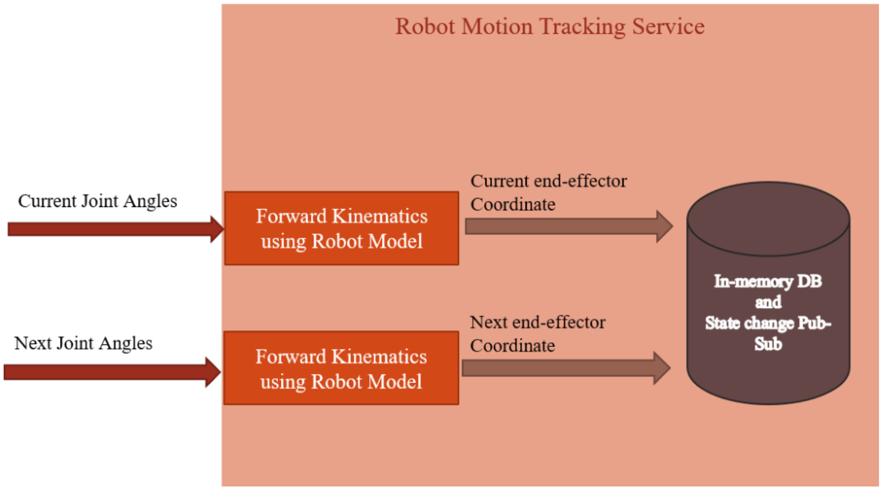

Figure 3-6 Robot Motion Tracking Service (RMT) components

Collision Avoidance and Prediction service (CAP)

CAP subscribes for the current and next end-effector coordinates of Robots in the workspace from corresponding instances of RMT service. Also, CAP subscribes for the Personnel coordinate estimates and predicted occupancy coordinates of Personnel in the workspace from corresponding instances of PLMC service. Then, CAP combines information from these services and uses probabilistic algorithms to predict the probability of collision between a given personnel and robot in a workspace area time ahead. If the possibility of collision is detected, based on likelihood, safety distance and velocities of approaching Personnel and Robot, CAP service either slows down the robot or stops the Robot by sending the appropriate control signal to PLC via IPC.

One instance of CAP service is assigned to a group of Robots and Personnel in a workspace. This service provides the following:

- Probability of collision between given Personnel and Robot time ahead
- Stop or slow Robot based on the likelihood, safety distance and velocities of approaching Personnel and Robot
- Calculate safety region/distance required between Robot and Personnel

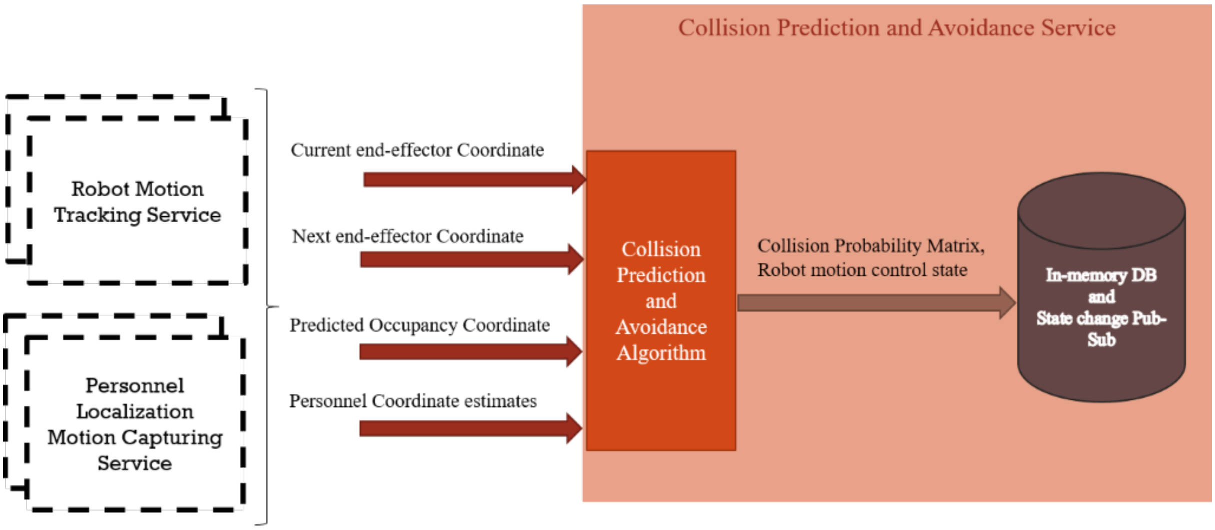

Figure 3-7 Collision Avoidance and Prediction Service (CAP)components

3.1.2 Virtual Demonstrator

Need for Virtual demonstrator over Physical demonstrator

The Human-Robot Collaboration use case aims to evaluate the RAINBOW platform and its benefits in industrial manufacturing. As a physical demonstrator is limited in terms of reproducibility, repeatability and setting up of a variety of complex test scenarios, to successfully evaluate various features and components provided by RAINBOW an approach of virtual and physical demonstrator is opted for.

Table 3-2 Benefits of using virtual demonstrator over physical demonstrator

Test attributes	Virtual Demonstrator	Physical Demonstrator
Setting-up variety of test scenarios	Supports simple and complex test scenarios with tens of workspace and hundreds of personnel	Supports simple test scenario with only 2 workplaces and less than 10 personnel
Repeatability of tests	The tests scenarios can be repeated any number of times	The tests scenarios are to repeat many times
Reproducibility of tests	The tests scenarios can be reproduced completely	The tests scenarios cannot be reproduced completely

The virtual demonstrator consists of virtual workspace and fog emulator as shown in Figure 3-8:

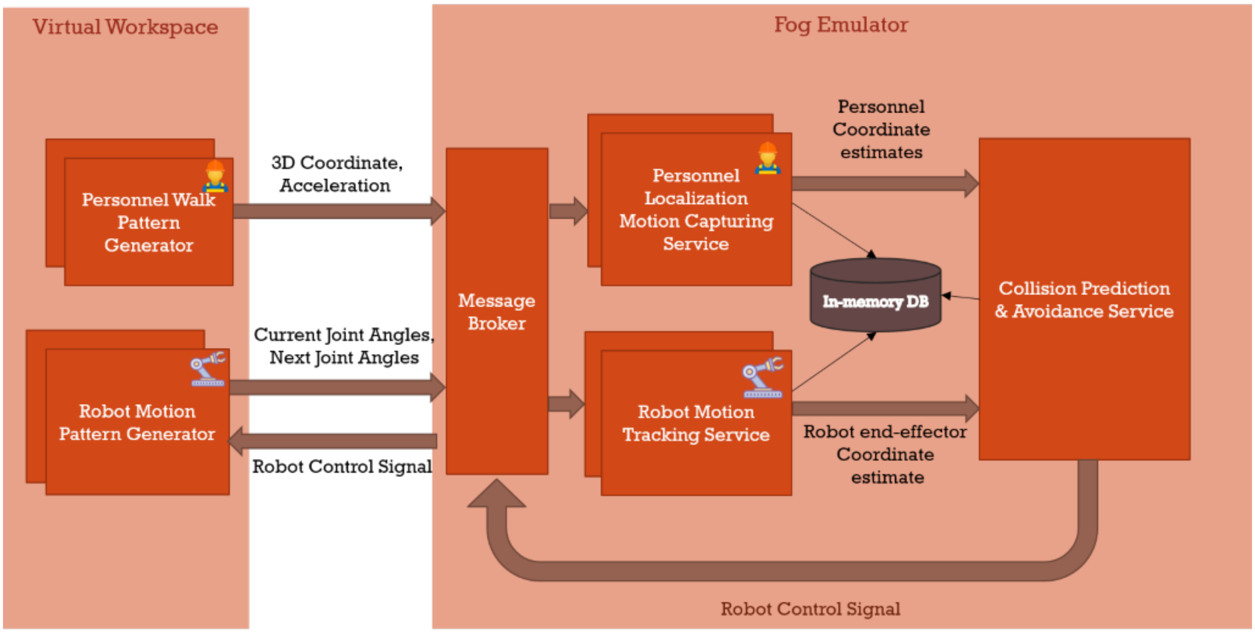

Figure 3-8 BIBA virtual demonstrator

Virtual Workspace

The virtual workspace consists of Personnel Walk Pattern Generator (PWPG**)** and Robot Motion Pattern Generator (RMPG). PWPG simulates a pseudo-random walk of a Personnel member within the workspace by considering Personnel acceleration, velocity and walking model. RMPG simulates the motion pattern of the robot standing still or performing a task in the workspace. Both PWPG and RMPG use information from floor plan (a floor plan contains information regarding robot placements, and static obstacles like cage, walls etc.) to obtain information about the static obstacles and environment in the workspace. Using these fundamental components, it is possible to build complex test scenarios which simulate a large number of workspace and personnel interactions.

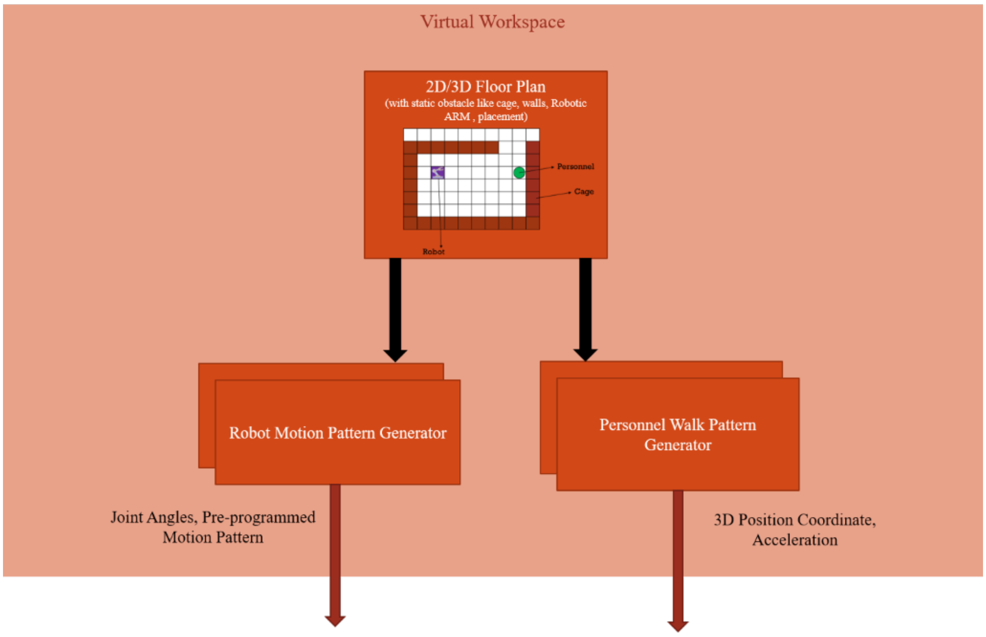

Figure 3-9 Virtual Workspace components

Personnel Walk Pattern Generator (PWPG)

Simulates positioning tags mounted on personnel using Inertial Motion Generator (IMG) and Sigmoid Walk Angle Generator (SWAG) to produce Noisy 3D Acceleration and noisy 3D Positional coordinates which are published periodically to the Message Broker using Message Client. Figure 3-10 shows the different components in PWPG.

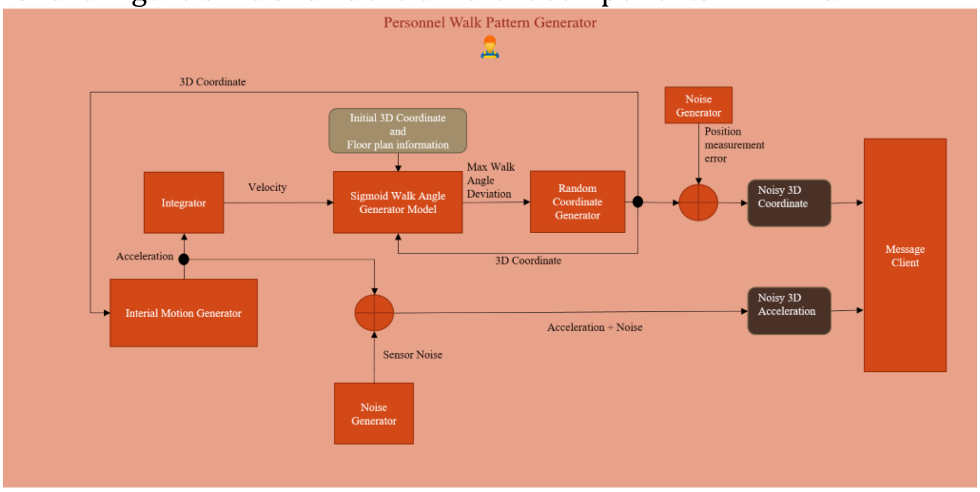

Figure 3-10 Personnel Walk Pattern Generator (PWPG) components

Inertial Motion Generator (IMG)

Generates acceleration values (true value without noise) from current and previous 3D position coordinates. This acceleration value is added with Gaussian noise using a noise generator, to simulate actual sensors noisy 3D acceleration output. Adding to this, acceleration values from IMG is further integrated to get velocity which is then given to Sigmoid Walk Angle Generator (SWAG).

Sigmoid Walk Angle Generator (SWAG)
Generates Max Walk Angle Deviation given velocity and current 3D positional coordinates of personnel along with floor plan information for identifying static obstacles in the workspace. Max Walk Angle Deviation can be defined as maximum sideward movement that personnel can take at a given velocity without falling over. Max Walk Angle Deviation can have a wide range of values and can change from personnel to personnel depending on the body weight distribution, etc. For simulation purposes, a generalized Sigmoid model is assumed for Velocity Vs Max Walk Angle Deviation (see Figure 3-10).
The Max Walk Angle Deviation is a range of values a person's position is allowed to take. From this range, a value of Walk Angle Deviation is picked randomly from a Gaussian distribution with previous Walk Angle Deviation as mean value. Later, new coordinates are generated using previous coordinates and new Walk Angle Deviation. To simulate real-world position tag, a Gaussian Noise is added to new Positional Coordinates thus producing Noisy 3D Coordinates.

Robot Motion Pattern Generator (RMPG)
Simulates the working movement of a physical robotic arm in the workspace by generating motion pattern for Robots (both operational and non-operational). Figure 3-11 shows the different components in RMPG.
For Robots, the motion patterns for end effectors are stored in Lookup. This information is used to obtain joint angles of the robot using Robot model and applying inverse kinematic approach. As the motion pattern is fixed and stored up in lookup, thus it is possible to obtain next Joint Angles ahead of time.

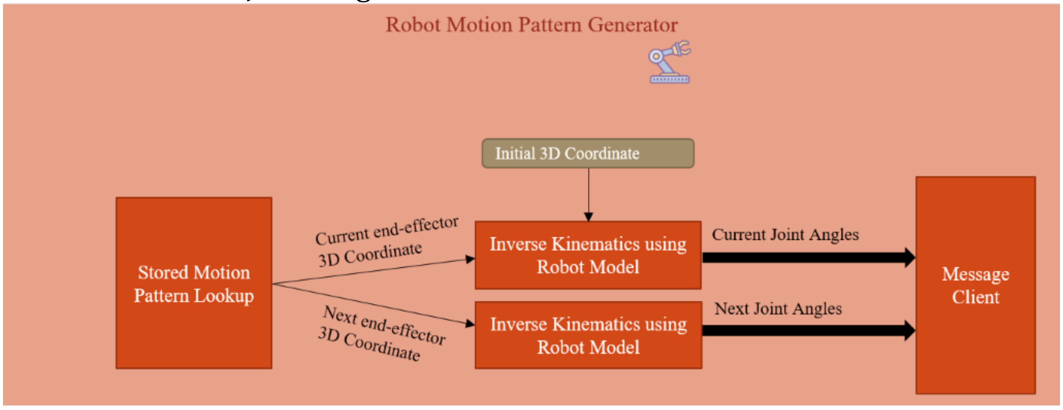

Figure 3-11 Robot Motion Pattern Generator (RMPG) components

3.1.3 Usage of RAINBOW Tools and Assets

Here an attempt is made to introduce the role played by the RAINBOW platform in the use case.
<u>Service Developer</u>
Below services are developed as cloud-native components by the Service Developer:
- Personnel Localization and Motion Capturing Service (PLMC)
- Robot Motion Tracking Service (RMT)

- Collision Avoidance and Prediction Service (CAP)

The service developer creates application templates of PLMC, RMT, CAP cloud-native components using the Service Graph Editor in the Modelling Layer. Thus, this provides information about service topology, resource dependencies and constraints. Following are few high-level constraints.

- Deployment constraints:
    - Robot type, make supported
    - Max-Robot-Stop time
    - Processor architecture of fog device
- Operation constraints:
    - Max-jitter
    - Max-delay
    - Minimum-bandwidth required
    - Max system reaction time
- Resource constraints:
    - Storage
    - CPU usage
- Security constraints:
    - Data Integrity
    - Data sharing with respect to other service entities

<u>Service Provider and Data Analyst</u>
Using Policy Editor in Modelling layer, the Service Provider provides run-time constraints such as:

- Max-jitter
- Max-delay
- Minimum-bandwidth required
- Max system reaction time
- Storage
- Dynamic sharing of resources between Fog devices in case one of the devices lack resources. Here amount of load to move and Max system latency for moved load are described
- Switch to secondary Fog device if primary Fog device fails in a workplace area.

To view operational stats, the Service Provider uses the Dashboard UI. The Service Provider should be able to create a new dashboard and view application-level monitoring metrics in a graphically intuitive and interactive manner. Adding to this, Service Provider and Data Analyst use analytics editor for creating queries which are optimized on distributed database to get data necessary for high level analytics. Following are some high-level analytics performed by the Data Analyst:

- Activity recognition and activity synchronization between Personnel and Robot

- Unintended Service/Process downtime
- Localization, prediction accuracy
- Process optimization analysis
- Human Ergonomics analysis

The RAINBOW platform enables this use case demonstrator to use a scalable micro-service-based architecture by providing:

1. Scalable and secure deployment of micro-services based on deploy-time constraint evaluation: This is achieved with the help of components such as the Service Graph Editor in the modelling layer. Also, with components such as pre-deployment constraint solver, deployment manager, orchestration lifecycle manager, resource manager in centralized orchestration backend and RAINBOW mesh stack on the fog device.
2. Run-time application monitoring, periodic run-time constraint evaluation and resolution as described in policies (when not met): This is achieved with the help of components such as policy editor in the modelling layer. Also, with components such as orchestration lifecycle manager, resource manager, resource application-level monitoring in centralized orchestration backend and multi-domain sidecar proxy in RAINBOW mesh stack.
3. Resource-sharing enables handling failures and short-term fluctuation in processing load: This is achieved with the help of components such as orchestration lifecycle manager, resource manager, resource application-level monitoring in centralized orchestration and multi-domain sidecar proxy in RAINBOW mesh stack.
4. Data management and high-performance queries across distributed databases for data analytics: This is achieved with the help of components such as the analytics editor in the modelling layer. Also, with components such as the analytics engine in centralized orchestration backend and RAINBOW mesh stack.

3.1.4 Demonstrator Specific KPIs

#	Values
ID	BIBA-KPI-01
KPI Title	Deterministic System Latency for collision prediction and avoidance
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Deterministic System Latency to predict and avoid fatal collision between personnel and Robot
RAINBOW's Contribution	System Latency can be divided into 4 parts • **Data Acquisition Latency (T_{DA})** is time taken for acquiring Robot motion data, Personnel localization and motion data from a remote

	node and sending the data to respective services in Fog devices. • **Data Processing latency (T_{DP})** is time taken by CAP, RMT, PLMC services for processing the data. • **Robot reaction time (T_R)**: Time taken for stop signal which is generated by CAP service in Fog device to be received by Industrial PC (IPC) of the Robot • **Robot stop time (T_{stop})**: Time taken to stop the Robot, once stop signal is acted upon. This latency depends on operating speed, type and manufacturer of Robot. • **Jitter (T_{jitter})**: is time jitter in receiving the packets due to network condition, topology, routing mechanism etc. In an event of predicted collision, the service operator expects the System Latency to be **System Latency = $T_{DA} + T_{DP} + T_R + T_{stop} \pm T_{jitter}$** *$T_{DA}$,T_{DP} ,T_R ,T_{stop} are application case specifics and changes based on Robot manufactured, algorithm, Processing IT infrastructure, and protocol used.* System Latency must have jitter of range less than 200 milliseconds.
KPI Measurement Formula	System Latency = $T_{DA} + T_{DP} + T_R + T_{stop} \pm T_{jitter}$ [T_{DA} , T_{DP} , T_R , T_{stop} , T_{jitter} are expressed in milliseconds]
Current Value (without RAINBOW)	System Latency is not determinstic due to network jitter.
Expected target value (with RAINBOW)	System Latency is deterministic within the range of 200 milliseconds.

#	**Values**
ID	BIBA-KPI-02
KPI Title	Reliable hand-off of data for personnel mobility scenario
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Device to device reliable hand-off of data in fog mesh for personnel mobility scenario
RAINBOW's Contribution	In case, when personnel move from one workspace area to another area, there must be reliable transfer of data from one fog device to other.

#	Values
	Adding to this, appropriate new instances of services must spin in new workspace area fog device. This data migration and service instantiation must happen within 1000 milliseconds(ms) after receiving workplace transition signal. Also, it is important the service in old workplace fog device must be terminated subsequently.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Data migration + service instantiation ≤ 1000 ms

#	Values
ID	BIBA-KPI-03
KPI Title	Up-Scaling of Personnel Localization Motion Capturing (PLMC) services
KPI Type	Business Case Specific
Objective / Subjective	Subjective
Need Relevant to KPI	Up-Scaling of Personnel Localization Motion Capturing (PLMC) services running in Fog device, when new personnel enter a workspace.
RAINBOW's Contribution	When personnel (P1) enter a collaborative workspace area (W1), upon discovering the presence of P1's positioning tags. A new instance of PLMC service must be spun with the help of RAINBOW in Fog device (FD1) which has enough resources to accommodate new personnel's PLMC service in the assigned Workspace (W1).
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Service count of PLMC service must be increased by one in FD1 and service must subscribe to localization and motion data from P1. Time taken to spin new instance of service must be less than 1000 ms

#	Values
ID	BIBA-KPI-04
KPI Title	Down-Scaling of Personnel Localization Motion Capturing (PLMC) services
KPI Type	Business Case Specific
Objective / Subjective	Subjective
Need Relevant to KPI	Down-Scaling of Personnel Localization Motion Capturing (PLMC) services running in Fog device, when personnel leave a workspace.
RAINBOW's Contribution	When personnel (P1) leave a collaborative workspace area (W1), upon discovering the absence of P1's positioning tags. The instance PLMC service corresponding to personnel must be terminated in Fog device (FD1)
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Service count of PLMC service must be decreased by one in FD1 and service must unsubscribe to localization and motion data from P1.

#	Values
ID	BIBA-KPI-05
KPI Title	Up-Scaling of Collision Avoidance and Prediction (CAP) services
KPI Type	Business Case Specific
Objective / Subjective	Subjective
Need Relevant to KPI	Up-Scaling of Collision Avoidance and Prediction (CAP) services running in Fog device, when the number of personnel and/or Robots installed within the workspace area (W1) exceed a prescribed group size.
RAINBOW's Contribution	When number of personnel and/or Robots installed within the workspace area (W1) exceed a prescribed group size (say for example group size is 10 which includes count of Personnel, Robots combined) for one CAP service. The Fog device (FD1) that has enough resources to accommodate new instance of CAP in W1. A new instance of CAP service must be spun by RAINBOW in FD1 assigned to W1.
KPI Measurement Formula	N/A

#	Values
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Service count of CAP service must be increased by one in FD1 and service must subscribe data from newly added Personnel and/or Robots Time taken to spin new instance of service must be less than 1000 ms

#	Values
ID	BIBA-KPI-06
KPI Title	Down-Scaling of Collision Avoidance and Prediction (CAP) services
KPI Type	Business Case Specific
Objective / Subjective	Subjective
Need Relevant to KPI	Down-Scaling of Collision Avoidance and Prediction (CAP) services running in Fog device, when the number of personnel and/or Robots installed within the workspace area (W1) recede a prescribed group size.
RAINBOW's Contribution	When number of personnel and/or Robots installed within the workspace area (W1) recedes a prescribed group size (say for example group size is 10 which includes count of Personnel, Robots combined) for one CAP service. The instance CAP service with zero reference to personnel must be terminated in Fog device (FD1)
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Service count of CAP service must be decreased by one in FD1

#	Values
ID	BIBA-KPI-07
KPI Title	Up-Scaling of Robot Motion Tracking (RMT) services
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Up-Scaling of Robot Motion Tracking (RMT) services running in Fog device when a new Robot is installed a workspace.

#	Values
RAINBOW’s Contribution	A new Robot (RA1) is installed within the workspace area (W1). The Fog device (FD1) that has enough resources to accommodate new Robot in the workspace, must create a new instance of RMT service. This Service instance must be spun by RAINBOW in FD1 assigned to W1.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Service count of RMT service must be increased by one in FD1 and service must subscribe to motion data from RA1.

#	Values
ID	BIBA-KPI-08
KPI Title	Down-Scaling of Robot Motion Tracking (RMT) services
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Down-Scaling of Robot Motion Tracking (RMT) services running in Fog device, when a robot is uninstalled in a workspace.
RAINBOW’s Contribution	When Robot (RA1) is uninstalled in a collaborative workspace area (W1), upon discovering the absence of P1's positioning tags. The instance RMT service corresponding to the robot must be terminated in Fog device (FD1)
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Service count of RMT service must be decreased by one in FD1 and service must unsubscribe to robot motion data from RA1.

#	Values
ID	BIBA-KPI-09
KPI Title	Monitoring and evaluation of Service Level Objective's (SLO's)
KPI Type	Business Case Specific
Objective / Subjective	Objective

#	Values
Need Relevant to KPI	Continuously monitor and evaluate mentioned SLO's against run time policies set by Service operator. Take suitable action if SLO's are not met.
RAINBOW's Contribution	Continuously monitor and evaluate mentioned SLO's against run time policies set by Service operator. • Max-jitter in the network • CPU and memory Usage • Operational status of micro-service • Customizable application specific metrics If SLO's are not met. then suitable actions as defined by Service Operator in run time policies are to be taken.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Continuously monitor and evaluate mentioned SLO's against run time policies set by Service operator. If SLO's are not met then suitable actions as defined by Service Operator in run time policies are to be taken. In this use case, if SLO's are not met the Robots are stopped immediately to prioritize human safety.

#	Values
ID	BIBA-KPI-10
KPI Title	Data sharing
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Data sharing with registered user for monitoring and analytics
RAINBOW's Contribution	Registered users are allowed access RAINBOW Platform only upon user authentication process. Un-registered user must register first before using and for successful user registration process approval from Service Operator is needed.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.

#	Values
Expected target value (with RAINBOW)	Only Users registered using valid credentials and access permission are allowed to access data and other RAINBOW platform specific features.

#	Values
ID	BIBA-KPI-11
KPI Title	Security and Attestation
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	On-boarding of new fog device
RAINBOW's Contribution	In an event of on-boarding of new fog device in the mesh network. The new fog device must adhere to attestation policies by providing verifiable evidence on their configuration integrity and correctness.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	On-boarding new fog device must adhere to attestation policies.

#	Values
ID	BIBA-KPI-12
KPI Title	Data Synchronization
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Periodic data synchronization from distributed databases with central database for analytics, data sharing.
RAINBOW's Contribution	Periodically Synchronize data from all distributed databases present in each of the Fog with Central database present in the premises for maintaining long time data persistence.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Periodically Synchronize data from all distributed databases present in each of the Fog with Central database.

#	Values
ID	BIBA-KPI-13
KPI Title	Analytical query
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Support queries on Distributed database across Fog Device Mesh Network
RAINBOW's Contribution	Support for customized queries requiring to fetch data from a Distributed database across Fog device mesh network.
KPI Measurement Formula	N/A
Current Value (without RAINBOW)	In the current demonstrator, this feature is not supported because services are not scalable and run-on dedicated server.
Expected target value (with RAINBOW)	Support optimized queries requiring to fetch data from Distributed database across Fog device mesh network. For this a user-friendly query language, yet optimized in terms of CRUD operation is required.

Table 3-3 RAINBOW Business KPIs (section 2.2.2.2) applicable to Demonstrator #1

ID	KPI	AS-IS Value	TO-BE Value
RAINBOW-KPI-01	Deployment Time	N/A	< 1000 msec
RAINBOW-KPI-03	Security Incidents	N/A	< 1 / year
RAINBOW-KPI-04	Service Availability	N/A	> 99%

3.2 Evaluation in Demonstrator #2 – CRF

3.2.1 Physical Demonstrator

The Digital Transformation of Urban Mobility use case aims to demonstrate how RAINBOW will contribute to develop a real-time geo-referenced notification system about a hazardous situation for vehicles travelling in urban areas. The RAINBOW platform will act also in the vehicle communication field, by providing a reliable and decentralized approach to safely handle the exchange of messages.

Application Layer
The use case we address is the Automatic Hazardous Events Detection (AHED) which refers to Road Hazard Signaling (RHS) application category, defined in the Vehicle-to-Everything (V2X) application standard [8]. The notification system will be designed to collect signals issued by entities in urban areas (Vehicles, Road Side Units and Vulnerable

Users); automatic notifications may also be triggered by Road Side Units, or MEC (Multi-access Edge Computing) where AI/ML algorithms can infer alert conditions that should be reported with different priorities based on the vehicle distance from the hazardous situation.

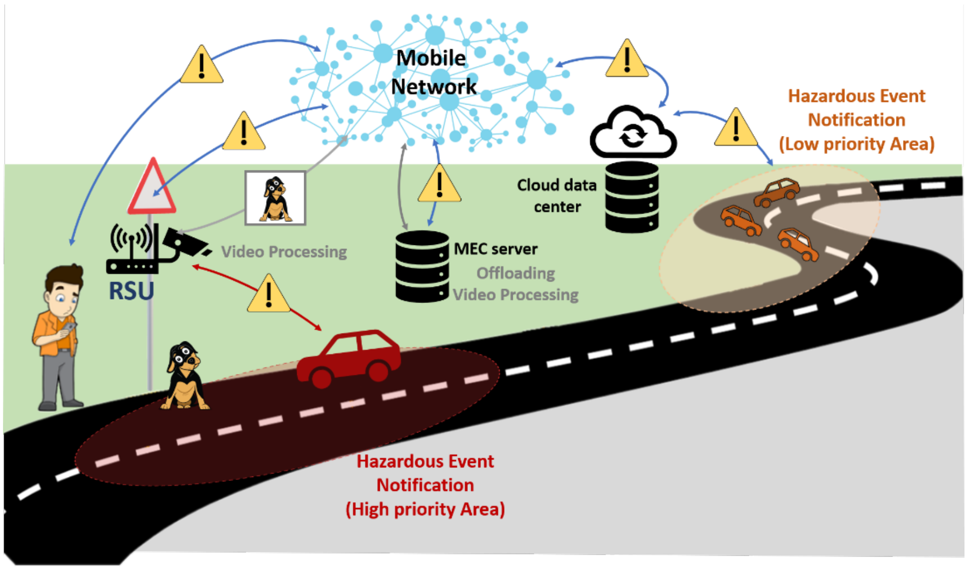

Figure 3-12 CRF physical demonstrator

The message priority is defined to support the vehicle safety and in the AHED use case the levels "Warn" and "Inform" are managed:

- Act: Use the Vehicle on-board sensors to instantly react at a risk of collision (ADAS Advanced Driver-Assistance System)
- Warn: Use the Short-range communication (V2I: Vehicle to Infrastructure, V2V: Vehicle to Vehicle) to receive collision risk alerts (High Priority Area)
- Inform: Use the Long-range communication (V2N: Vehicle to Network) to receive collision risk alerts (Low Priority Area)

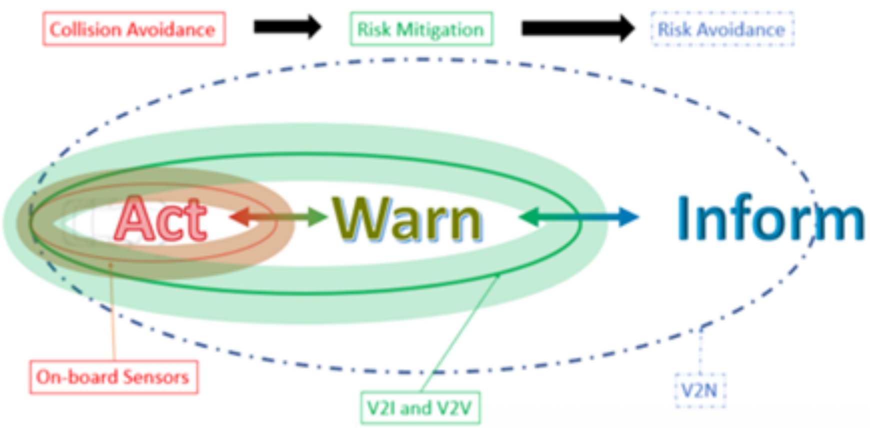

Figure 3-13 Understanding vehicle safety [9]

Messages-Facilities Layer

C-V2X messages used in this use case are chosen according to the European standard for vehicular communications [10]. In particular, we use the Co-operative Awareness Message (CAM) type, defined in [11] and the Decentralized Environmental Notification Message (DENM) type, defined in [12].

- **CAM messages** are periodically sent by vehicles (both in broadcast via PC5 interface and towards the AMQP Broker via Uu interface) to inform other vehicles and RSUs of their presence. Among all the standard fields, vehicles share their position, heading angle and speed.
- **DENM messages** are notification messages triggered by a hazardous event detection. They are generated by the RSU as it detects a road hazard and they reach vehicles in a high priority area (vehicles directly interested in the event) in a direct way via PC5 interface. DENM messages are also sent by RSU to the AMQP Broker server on the cloud node, which forwards them to vehicles in a low priority area (vehicles far from the hazard) via Uu interface. Among all the standard fields, the DENM message provides a description of the event being reported through a "Cause code". Of interest to us is Cause code 11, defined as "Hazardous location - Animal on the road" by the standard [13].

The target for the KPI will be the value from the pertinent standard ETSI TR 103 300-1 V2.1.1 [14].

Network-Data Link Layers

The Vehicle architecture implements the C-V2X standards to support the AHED use case and has two complementary cellular technology communication modes (Direct and Network) which is typically a single, fully integrated chipset solution. The features of the two modes as the following:

- Direct (Sidelink) for V2V, V2I, and V2P
    - Communications take place within the network coverage as it is a short-range direct-communication mode. The radio base station manages the radio resources to maximize radio performance (Mode 3)
    - Operating in ITS bands (e.g., ITS 5.9 GHz) independent of cellular network
    - Short range (<1 kilometer), location, speed Implemented over 3GPP's PC5 interface
    - The protocol is connection-less and no IP layer encapsulation is used
    - Messages are sent directly over the MAC layer that handles source and destination address resolution
- Network (Up/Downlink) for V2N
    - This interface connects the devices with the 5G radio base station (eNB), thus operating as in normal communications and also being able to perform V2N type communications.
    - It operates in traditional mobile broadband licensed spectrum.

- o Long range (>1 kilometers), e.g., accident ahead Implemented over 3GPP's Uu interface

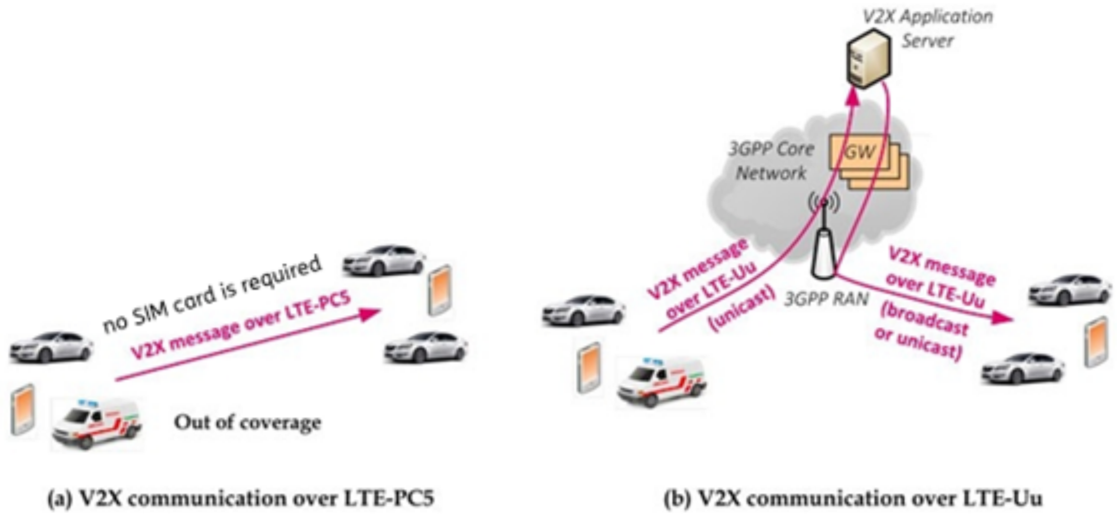

Figure 3-14 V2X communication through PC5 and Uu [15]

3.2.2 Virtual Demonstrator

Consists of three labs (CRF, Links and Polito) that emulate a real-time geo-referenced notification system. **CRF lab** makes available an equipped vehicle that can receive DENM messages and send CAM messages. **LINKS lab** is tasked with controlling the RSU, the edge node and the AHED instance whose resources are managed between RSU and edge node thanks to RAINBOW capabilities. **Polito lab** will provide the cloud node tasked with managing the communication between RSU and vehicles in a low priority area. The cloud node is also responsible for the visualization of the events on the City Aggregator online platform.

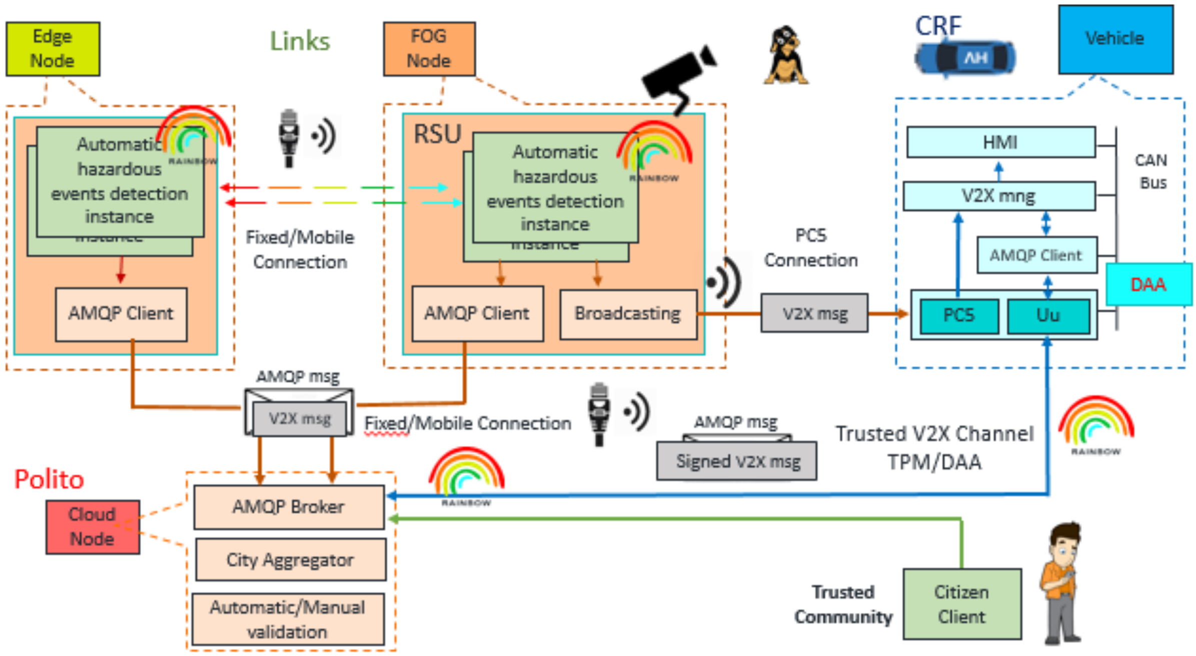

Figure 3-15 CRF Virtual Demonstrator

CRF Vehicle

The Vehicle architecture implements all the C-V2X standards (Physical, Data Link, Transport, Facilities Layers) to support the AHED use case and has the two complementary communication modes (Direct, Network). For safety purposes and for the sake of experimental setup, the Vehicle has 2 CAN busses; one is the standard Vehicle production CAN bus and one is the experimental ah-hoc “RAINBOW bus” for the AHED use case implementation. Both busses are connected to the Secure Gateway (SGW).

The architecture core component consists of a 5G Telematics Box Prototype equipped with Uu and PC5 interfaces with a C-V2X client and an AMQP client. The Telematics Box sends and receives ETSI messages (CAM, DENM) on both interfaces. When an AMQP message comes from the Uu interface, it is decoded and the C-V2X payload message is extracted; after the parsing, the C-V2X message is converted into a CAN bus message to transmit over the CAN bus to other on-board Units. From the PC5 interface, the messages are in C-V2X standard so they can be directly converted in a CAN bus message to transmit over the CAN bus.

The Head Unit is the on-board unit that contains the Radio features and the HMI system, all messages from the Telematics Box are received by the Head Unit and, after the decoding, displayed on the HMI. The Car PC contains all additional and experimental features to support the RAINBOW Trusted features TPM and DAA.

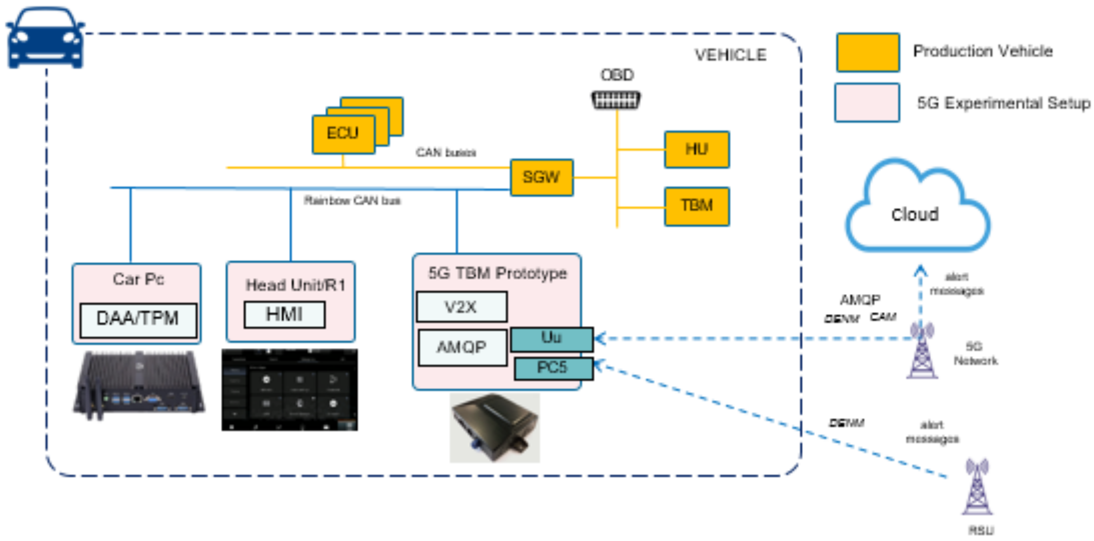

Figure 3-16 CRF Vehicle Setup

CRF Vehicle Setup:

- R1 High Radio/Head Unit
    - Vehicle Head Unit and Radio for HMI implementation
- Prototypal 5G Telematic Box
    - Uu and PC5 radio interfaces
    - GPS
    - AMQP and C-V2X Client
- Car PC: System embedded Core i7 based on 9th/8th generation
    - PC Card for CAN BUS Connection
    - DAA/TPM Modules

LINKS lab

The **Fog Node** consists of a Road-Side Unit equipped with the following components:

- IP Network Camera 5MP @30fps. The IP camera can be accessed through web browsers, RTSP players and 3GPP-compatible mobile devices. The IP camera is connected to the internet and its RTSP video stream can be accessed from outside the Fog Node.
- NVIDIA Jetson AGX Xavier with following characteristics:
    - GPU 512-Core Volta GPU with Tensor Cores
    - CPU 8-Core ARM v8.2 64-Bit
    - Memory of 32 GB
    - Linux kernel-based operating system (Ubuntu 18.04.5 LTS (Bionic Beaver)

  The platform runs computer vision algorithms on GPU able to detect the presence of animals. the RSU is in charge of broadcasting hazardous event notification through ETSI DENM messages (via PC5); it also acts as an AMQP Client, publishing

messages on the hazardous notification to the cloud so they can be accessible everywhere (and not only for vehicles in the RSU coverage area).
- PC5 interface for V2X direct communication between RSU and incoming vehicles.
- Interface for internet access (optical fiber or 4G/5G) to allow communication RSU-MEC, RSU-Cloud Node.

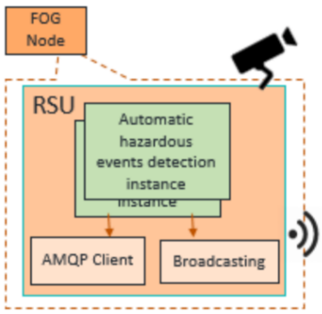

Figure 3-17 Fog Node Setup

The **MEC Node** is a Linux-based server. The MEC server will be "simulated" using a cloud server that will be geographically near to the Road Side Unit. A real MEC server would be available only with the participation of an MNO or of a road operator that is not foreseen in the project. Additional information on the configuration of the server will be provided.

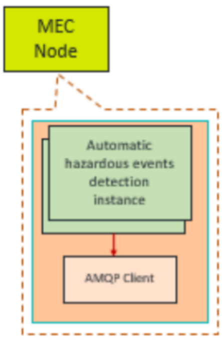

Figure 3-18 MEC Node Setup

Polito lab

The **Cloud Node** is implemented on a server, located in the Polito lab, which provides real-time communication between RSU and vehicles, an online platform for visualization of events on a map and a validation process through a simulated citizen application. In particular:

- The AMQP Broker is the module intended for real-time communication. AMQP (Advanced Message Queuing Protocol) is a publish/subscribe based protocol (described in detail in the following paragraph) for forwarding real-time messages with a geo-referenced tag. In our scope, the AMQP broker acts as Consumer from the RSU, i.e., it listens to incoming DENM messages. It acts both as a Producer towards CRF vehicles, i.e., forwarding DENM messages from RSU in real-time, and as a Consumer from CRF vehicles, i.e., listening to periodic CAM messages.

- The City Aggregator is an online platform reachable via URL for the visualization of vehicle positions and of hazardous events on a map tile. It decodes DENM and CAM messages from the AMQP Broker and extracts geo-referenced information to be visualized on the map.
- The automatic/manual validation process performs a data-fusion analysis to validate hazardous events notifications both from the RSU and from pedestrians near the high priority area, which can send a notification to the cloud node via the Citizen application.

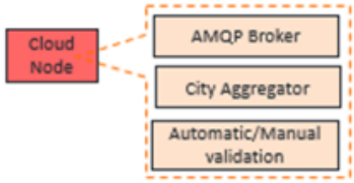

Figure 3-19 Cloud Node Setup

The **Citizen App (location)** is intended as a support for the system and allows pedestrian near the high priority area to inform the cloud node of a potential hazardous situations. For the purpose of this demonstrator, the citizen app will not be developed as a smartphone application, but it will be emulated by a module on a standalone computer in the Polito Lab capable of generating a flow of notification messages destined to the AMQP Broker.

Figure 3-20 Citizen App

<u>Communication protocols</u>

In this demonstrator we will have two types of communication protocols. A fixed/mobile connection between RSU and MEC nodes is intended as a service management channel to allocate resources between RSU and/or MEC node thanks to RAINBOW capabilities through RTSP protocol. Communications between RSU/MEC nodes and cloud node as well as between cloud node and vehicles are managed through the AMQP protocol. In particular:

- The Real Time Streaming Protocol (RTSP) is a network control protocol designed to control streaming media servers. It is used to monitor audio/video transmission between two endpoints and facilitate the transportation of low-latency streaming content across the internet. The RTSP communication is used to access the camera video stream from the Xavier component embedded in the RSU, and from the MEC server.

- AMQP is a message-oriented, open standard application layer protocol. It relies on a publish/subscribe message pattern, where users may subscribe to some topics of interest and receive real-time information on subscribed topics. Messages are incapsulated in a geo-referenced topic. In this way, messages are delivered to subscribed users only if they are located in an area where messages are relevant. When messages are produced and sent to the AMQP Broker server, they are stored in a first-in-first-out topic. If one or more user subscribed for that topic are present, messages are pushed in real-time towards all users located in a relevant area for the information. In our use case we use AMQP v1.0 with the ActiveMQ v5 broker implementation and the QPid-proton library.

Early Stage Demonstrator

The Early Stage Demonstrator consists of a Lab-to-Lab integration to demonstrate a simple service chain starting from a simulated Lab Hazardous Situation detection to a notification received from the in-lab emulated vehicle through the cloud. The main steps of the service chain are:

- [LINKS Lab] In-lab AHED service orchestration through animal detection using the RSU - MEC with emulated camera and AI/ML algorithm
- [LINKS Lab] Simple DENM HLN (Hazardous Location Notification) message generated from the RSU
- [LINKS Lab] Encapsulation of V2X DENM message into a AMQP standard message and publication on the Polito Cloud AMQP broker
- [Polito Lab] DENM management and publication of DENM on AMQP Vehicle queue
- [Polito Lab] Hazardous event notification displayed on the City Aggregator
- [CRF Labs] DENM reception and AMQP message parsing
- [CRF Labs] De-encapsulation of V2X HLN message
- [CRF Labs] Transformation of V2X alert message to a CAN message and dispatch to the simulated CAN network
- [CRF Labs] Reception (and read receipt) of the CAN message, elaboration and visualization on Vehicle HMI

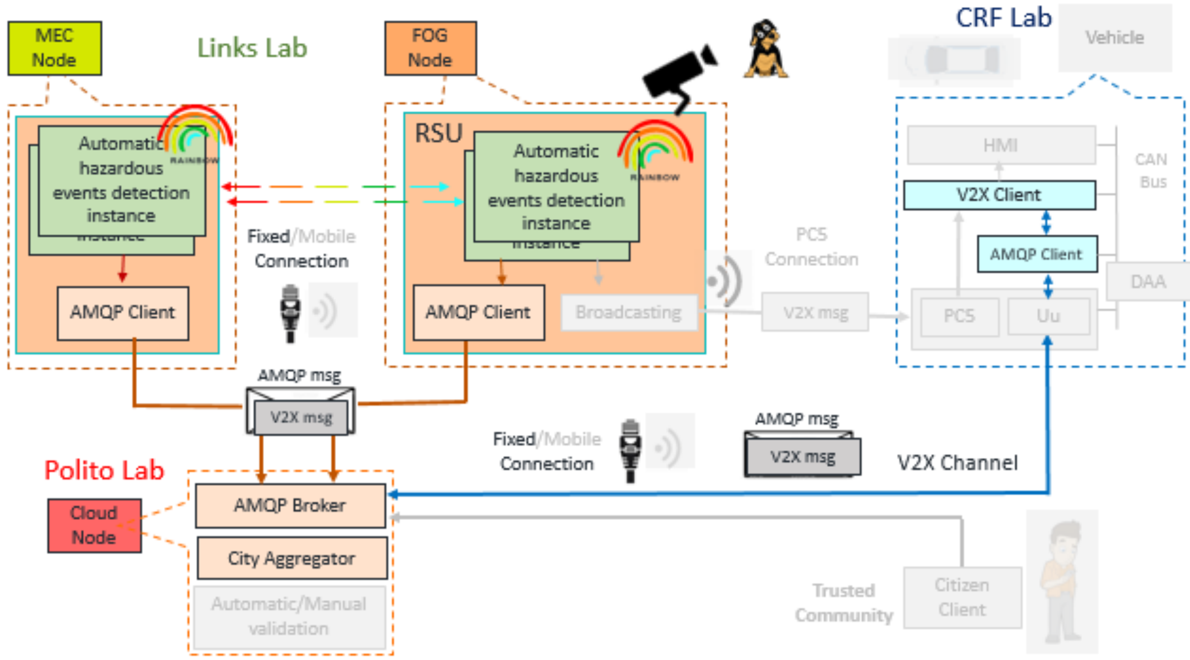

Figure 3-21 CRF Early Stage Demonstrator flow

<u>CRF Early Stage lab (CRF Vehicle)</u>

The architecture implements in the lab all the C-V2X standards (Physical, Data Link, Transport, Facilities Layers) to support the AHED use case and has only the network connection (Uu). The Setup is based on a CAN bus emulator on which a Car PC and an OBD port (on-board diagnostics) are connected. On the OBD port is connected an OBD dongle that communicates via Bluetooth with a Tablet. The Tablet has the emulation software of the Head Unit Radio system and displays alerts messages from the CAN BUS. The Car PC is also connected to a raspberry module where an AMQP Client is executed, using a GPS. The raspberry manages also the Uu connection through a 5G Modem.

When a new DENM alert is published on the Cloud AMQP broker:

- The AMQP client on the raspberry downloads and parses the AMQP message
- The raspberry extracts the V2X HLN message as payload of the AMQP message and generates a simple V2X Message which is sent to the Car PC
- The Car PC parses the V2X message and transforms it to a CAN message; then sends it to the simulated CAN bus
- The CAN message alert is read by the Tablet through the OBD dongle
- The Radio Emulation Software on the Tablet decodes the CAN bus and displays the result on the emulated HMI

The Car PC together with the Raspberry and the GPS are an emulation of the future 5G Telematic Box without the PC5 connection. For the scenario simulation there are two options, one is the GNSS signal emulator (GPS track) and the other is to use the GPS integrated inside the 5G Router (fixed position).

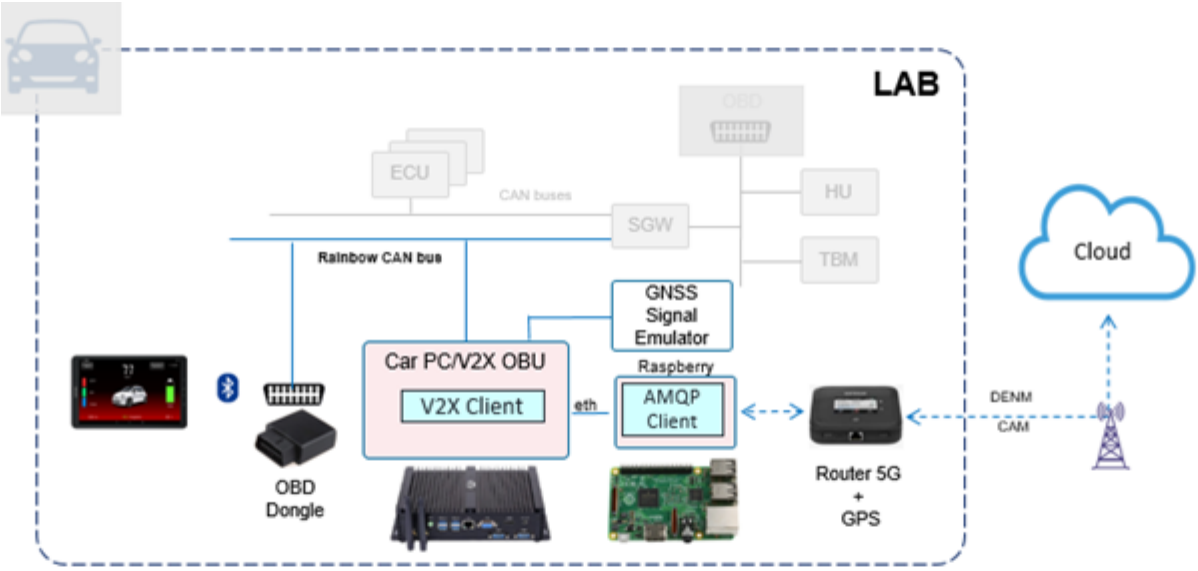

Figure 3-22 CRF Early stage lab

CRF Early Stage Vehicle Setup:

- Car PC: System embedded Core i7 based (or V2X On Board Unit)
    - C-V2X client (TBM emulation)
    - PC Card for connection to CAN BUS
- Tablet
    - Vehicle Head Unit and Radio emulation for HMI implementation
- OBD Bluetooth Dongle
- Raspberry
    - AMQP Client
- CAN Case simulation: CAN/LIN Network Interface with high-speed transceivers
- Router 5G for network connection and GPS

LINKS Early Stage lab

The LINKS lab early-stage demonstrator is reported in Figure 3-23:

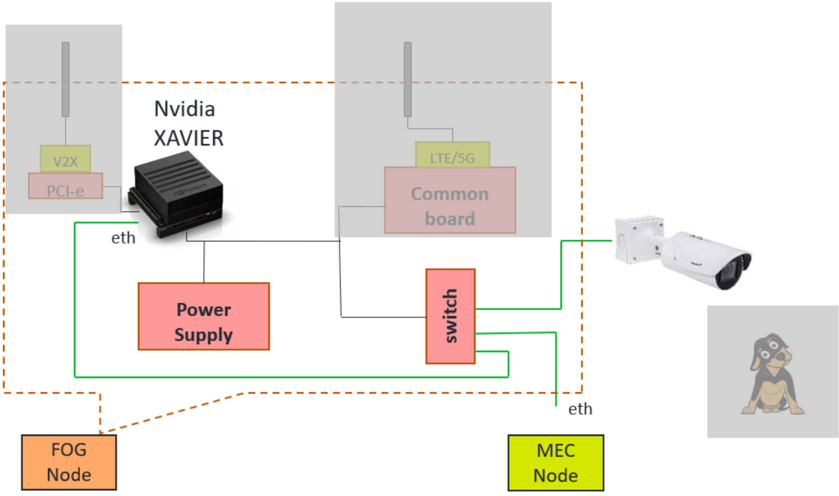

Figure 3-23 LINKS Early stage lab

This lab demonstrator includes both the RSU acting as a fog node and the MEC server. Both the MEC node and the RSU will host the AHED service, and both will forward the alert message to the Cloud node through the AMQP Client.
Since the early-stage demonstrator will be a lab demonstration, the V2X short range communication with the vehicle won't be used. Moreover, since both RSU and MEC server will be within the same lab, communication among them will be through cable and not wireless.
Finally, the occurrence of a dangerous situation (dog crossing the road) will be simulated by reproducing a video of that situation and the camera will be pointed toward a screen showing the video.

<u>Polito Early Stage lab</u>

Since Cloud node is supposed to work as a remote node even in the physical demonstrator, there are few differences between Early and Advanced stages from the Cloud node point of view. In particular, in Early stage:

- Message flows from Citizen app are not simulated
- Validation of received DENM messages is not performed from notifications via Citizen app

3.2.3 Usage of RAINBOW Tools and Assets

The first challenge for the demonstrator is to obtain, by means of the RAINBOW Smart Orchestrator, the best balancing between MEC and Fog Node in terms of energy consumption, bandwidth occupancy, accuracy and latency. The hazardous situations are detected through a Road-Side Unit (RSU) equipped with multiple IP cameras directly connected with it. The AHED service can identify the presence of animals on the road, which can represent a dangerous situation for drivers, by means of AI computer vision algorithms running on the video stream coming from the IP cameras.

As starting configuration, the RSU is supposed to be connected with a wireless or wired connection to the MEC node and to be powered by the electricity grid. In a non-RAINBOW situation, the AHED service can run or on the MEC node or on the RSU node. In the standard situation, the network bandwidth for connecting the RSU and the MEC node is enough to forward the video streams at low resolution and low fps on the MEC server. In this way, the RSU itself runs in low-power mode, leaving the most power-consuming activities on the MEC node where the energy management can be optimized. A drawback of such a set-up is that even if the frame image resolution is enough to detect dangerous situations, it might not be enough to ensure a reliable placing of the event, while the additional latency video stream forwarding and results communication from MEC node to RSU node has to be kept in consideration.
The other possible scenario refers to the AHED service continuously running on the RSU node. In this case, the direct communication between RSU and IP camera allow to work directly on high-resolution image frame, ensuring a higher level of accuracy and reducing the latency to deliver the information to all incoming vehicles. On the other hand, RSU hardware capability would require a high-power profile to manage all required computation, increasing the power consumption and requiring an additional cooling system.
With the introduction of the RAINBOW platform, the AHED service can be migrated backwards and forwards between MEC node and RSU fog node, to keep power consumption, bandwidth usage and performances always in a safe range. These polices constraints are defined at the RAINBOW modelling layer and continuously monitored by the RAINBOW orchestration functionality.
A second challenge for the demonstrator is to obtain, by means of the RAINBOW platform a secure data collection and distribution of V2X messages in order to obtain secure end-to-end architecture that enables a scalable bidirectional communication. To obtain this goal are needed two functionalities:

- The RAINBOW secure enrollment based on remote attestation enablers, for the registration of edge devices (vehicles) and of fog devices (RSUs)
- The RAINBOW privacy preserving exchange of messages through the integration of the DAA protocol.

3.2.4 Demonstrator Specific KPIs

#	Values
ID	AHED-KPI-01
KPI Title	AHED service orchestration power consumption
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Useful for managing power usage of the road infrastructure dedicated to the RSU node. Power consumption dedicated to road infrastructures has to be minimized on every RSU. Minimizing the power consumption, also the RSU temperature is reduced, guaranteeing better performances.

#	Values
RAINBOW's Contribution	RAINBOW platform allows to decrease the power consumption by moving the AHED service on the high-power fog node only when required, while using the MEC server (lower power consumption) during the rest of the time.
KPI Measurement Formula	RSU provides different power mode profiles to look at. By observing these, one can extrapolate the power consumption details.
Current Value (without RAINBOW)	Average power consumption: 30W more or less
Expected target value (with RAINBOW)	Average power consumption: 15W (Estimation) Average power consumption: <30W (Expected)

#	Values
ID	AHED-KPI-02
KPI Title	AHED service orchestration bandwidth occupancy
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Bandwidth occupancy management is crucial in order to provide a good video resolution for identifying the road hazard. Moreover, the more frames per seconds (fps) are guaranteed, the faster is the DENM sending process.
RAINBOW's Contribution	RAINBOW platform allows to move the AHED service from the MEC server to the RSU based on the bandwidth occupancy.
KPI Measurement Formula	Probing the network and measuring if the available bandwidth is enough for the video stream
Current Value (without RAINBOW)	fps not under control $0 < fps < 30$
Expected target value (with RAINBOW)	fps under control $10 < fps < 30$

#	Values
ID	AHED-KPI-03
KPI Title	AHED C-V2X Alerts delivery latency
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	In this use case, two types of communications are ensured. A direct V2I communication via PC5 interfaces for sending DENM messages in a high priority area and a V2N communication via AMQP Broker in order to inform vehicles (through Uu interfaces) in a low priority area of the hazard ahead. Both those transmissions have to be considered as security communications and for

	this reason the end-to-end latency of **every message** has to follow the standard requirements for security C-V2X communications.
RAINBOW's Contribution	RAINBOW platform adds DAA and TPM assuring a secure and trust layer between the dispatching of the alert and its reception at the vehicle. Security protocols provided by RAINBOW are indispensable, nevertheless security headers might produce higher end-to-end latencies. Standard end-to-end value has to be ensured despite the RAINBOW security layer delay.
KPI Measurement Formula	Probing end to end latency of V2X message from scene recognition to the alert on the vehicle.
Current Value (without RAINBOW)	300 milliseconds between the RSE and the vehicle. (KPI from ETSI TR 103 300-1 [14] "Non equipped VRUs crossing a road" as referred in 3.2.2.1).
Expected target value (with RAINBOW)	Less or equal to 300 ms

#	**Values**
ID	AHED-KPI-04
KPI Title	AHED Number of C-V2X Events managed
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	As soon as a hazard situation is recognized, a flow of DENM messages is sent towards the interested car (via PC5) and towards approaching cars (via Uu). It is important to send a flow of DENM messages and not a single one cause a single packet might be lost in transmission, received with errors or delayed due to network traffic. The frequency of messages in the DENM flow is set in order to have the reception **of the first DENM** on the vehicle within the standard latency value.
RAINBOW's Contribution	RAINBOW platform adds DAA and TPM assuring a secure and trust layer between the dispatching of the alert and its reception at the vehicle. Security protocols provided by RAINBOW are indispensable, nevertheless security headers might produce higher end-to-end latencies. Standard end-to-end value has to be ensured despite the RAINBOW security layer delay.
KPI Measurement Formula	C-V2X messages frequency from the start of the alert recognition to the end of the alert.
Current Value (without RAINBOW)	V2X exchange messages are broadcasted at a frequency between 1 and 10 Hz. (KPI from ETSI TR 103 300-1 [14] "Non equipped VRUs crossing a road").

Expected target value (with RAINBOW)	Less or equal to 10 Hz

Table 3-4 RAINBOW Business KPIs (section 2.2.2.2) applicable to Demonstrator #2

ID	KPI	AS-IS Value	TO-BE Value
RAINBOW-KPI-01	Deployment Time	N/A	< 1 min
RAINBOW-KPI-04	Service Availability	N/A	> 99%
RAINBOW-KPI-09	Investments for developing fog computing services	N/A	< 1 personmonth

3.3 Evaluation in Demonstrator #3 – MSP

3.3.1 Physical Demonstrator

Currently use of drones for power line monitoring is very ineffective due to the lack of coordination between them. The current system depends on an operator (or independently working teams) handling an individual drone via a Ground Control Station (GCS) (see figure below) for the facilitation of powerline surveillance.

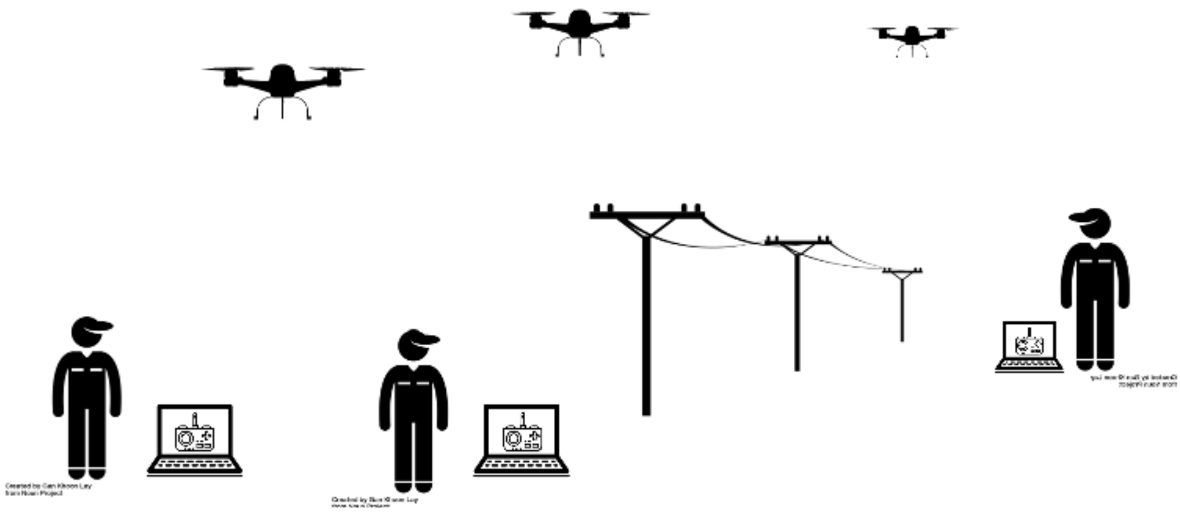

Figure 3-24 State of the powerline surveillance

The demonstrator will implement a distributed GCS that will govern a swarm of drones to optimize their operations and increase the swarm's range and autonomy. Due to legal requirements, the drone must operate within its radio link range and the line sight of an operator. The system will lift those limits and extend the drone's usable flight distance because it will be possible to automatically pass the control over a drone from one GCS node to another. Effectiveness will be further raised thanks to the data quality assessment in real-time. The GCS will automatically generate tasks for individual drones without human intervention taking into consideration the current state of the overall mission. The role of operators is limited to setting-up the GCS nodes and servicing drones. After the drones finish operations in a given area, this specific GCS node can be relocated, while others are still operating. The figure below, visualizes the architectural evolution of the swarm network, with the utilization of the RAINBOW platform.

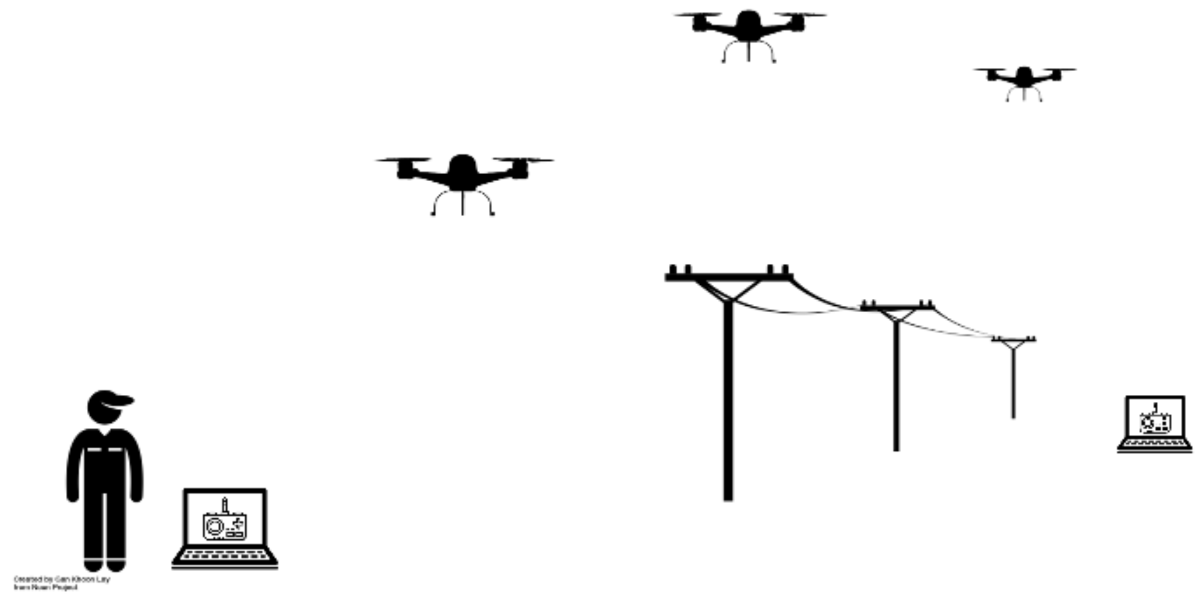

Figure 3-25 RAINBOW-enabled distributed powerline surveillance and GCS handover

The role of the physical demonstrator is to simulate real conditions during inspection missions along power lines. In such a case there is neither any established spatial organization of the system, nor any predetermined locations of system elements. Under those random conditions the operators shall follow a few general rules each time, such as: the power line predominantly determines the flight routes, GCS's are located along the power line in a manner that ensures the continuity of the radio connection between each other and between the flying drone and any of the GCS, as well as the line of sight between the flying drone and the operator. The range of communication between GCS's depends strongly on the terrain configuration and the presence of local radio interference from other sources, which factors are not possible to measure a priori. The adoption of safe distances must rely on the experience of the drone operator. The tests simulate real-world conditions and the above distance considerations do not have to be literally applied, i.e. the GCS's can be positioned closer together. However, it is important that drones can fly to the maximum of their flight time capabilities.
According to the above, the physical demonstrator will be organized as presented in the below diagram.

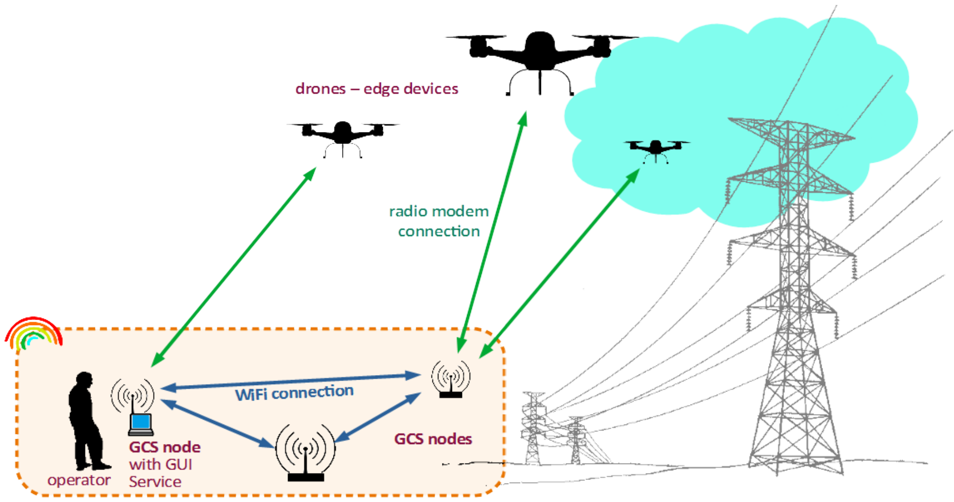

Figure 3-26 MSP Physical demonstrator – a general view

The drone – an edge device

The drone – an adge device - serves as the sensor carrier. A quadrocopter geoHOVERFLY (MSP's model) will be used (MTOW 1.6 kg, 20 min flight time). Three copters will be used and they will perform flights in different configurations (flying alone, flying together, flying interchangeably) realizing various scenarios.

A photogrammetric camera - the sensor – is installed on the geoHOVERFLY. For the purpose of the project, a **photo quality control module** will be developed and implemented, which, based on the **camera orientation angles, copter angular velocities and flight speeds according to GPS and IMU**, in real time will evaluate the correctness of the conditions for photos acquiring (result: correct photo, incorrect photo).

The drone is controlled by an **autopilot** based on the ArduCopter software. Its basic task is to navigate the quadrocopter autonomously based on a programmed flight plan. On the other hand, the autopilot **can receive commands from GCS** to change the course and route, including the return and landing commands. After deployment and launching, the drone automatically connects to the nearest GCS and registers itself as an edge device in the system. It also provides **data on the position, speed and altitude of the copter**, which are **transmitted within the telemetry data to the GCS (via a radio link, MAVLink protocol**) in a continuous mode and which are used by the camera trigger and photo quality control module.

In the project, the autopilot software will be extended with the possibility of changing the GCS from the one, which currently controls the flight of the copter, to another. The module will meet the requirements related to the safety and speed (short time) of switching operations between GCS's. This functionality will be used in the event of a decrease in the signal level from the current GCS to a level that may result in a loss of communication.

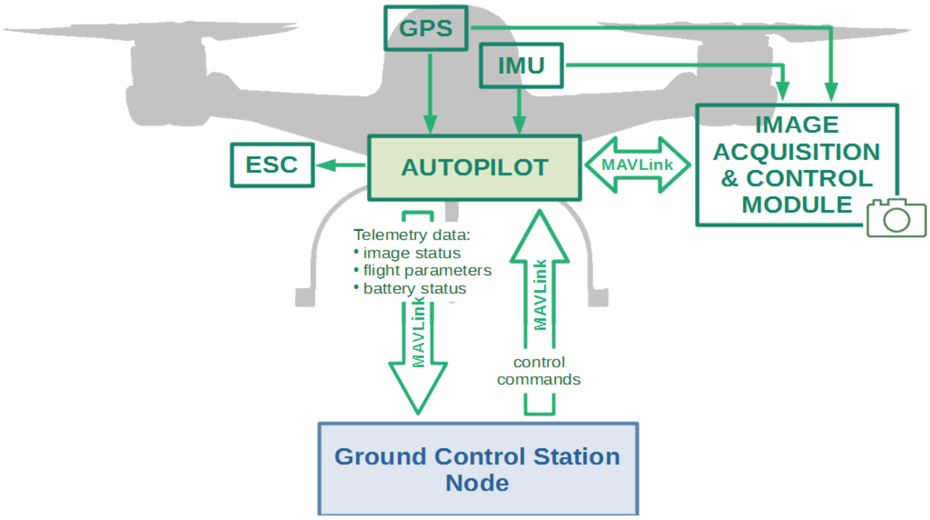

Figure 3-27 The drone, an edge device diagram

GCS - a node

The nodes of the ground flight control station will be based on a single board computer and a laptop (the latter in case an operator is present at a given node). They are accompanied by a communication system with a multicopter using a **radio modem and the MAVLink protocol**, and with other GCS nodes using **WiFi**. In addition, it can have an optional **internet connection** via LTE.

The GCS implementation will be based on open-source projects: **QGrondControl** (GUI) and **MAVProxy** (service part). The GCS node, in cooperation with other nodes and with the appropriate RAINBOW services, performs several functions:

- prepares the master mission plan (in the case of the node with the Mission Guidance service)
- based on the master mission and current data from the implementation of the missions of individual drones, as well as on the current results of the photo quality control, it plans an individual flight plan for the copter (in the case of the node on which the Mission Guidance service operates)
- uploads flight plan for the drone before its taking-off or transmits modifications of the flight plan to the drone being in the air
- maintains the connection and provides control over more than one drone
- receives telemetry data from the controlled drones
- maintains the exchange of data on the status of the entire mission and its implementation, including the status of all logged in drones with all GCS nodes
- is able to transfer control over the drone to another GCS or take the control over

GCS nodes can be freely placed in the field. New nodes may be introduced at any time, and they become involved in the system of controlling the entire operation on an equal basis. The node can also change its position at any time, also by turning it off (subject to prior transfer of control over the drone to another GCS), moving to a new position and re-joining the system.

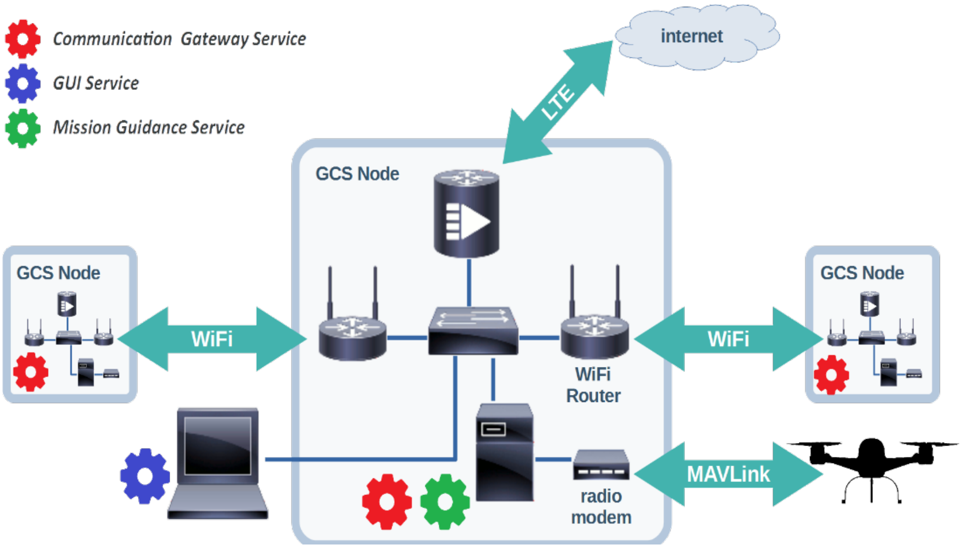

Figure 3-28 GCS node diagram

As part of the GCS implementation, three services will be prepared and implemented, as already described in D1.3 [4]:

- **Communication Gateway** - responsible for communication with the drone via a radio modem,
- **Mission Guidance Service** - responsible for master mission planning and individual flight plans planning
- **GUI Service** - the interface for the operator to handle the GCS

3.3.2 Virtual Demonstrator

At the early stage of the evaluation, a virtual demonstrator will be prepared. It will use the Software-in-the-Loop (SITL) approach to testing where the drone firmware controls a virtual drone that flies in a simulated 3D world. This allows testing various scenarios in a repetitive manner without risking damage to any equipment, without a need to go to the test field and without being dependent on weather conditions. This should greatly increase development speed as tests can be performed in minutes instead of days. Virtual environment allows to test every software functionality needed for the third use case of the project. The most important difference is that in the virtual world weather conditions do not change randomly. It is especially hard to simulate the effects of winds on battery drain. Also, the radio links will not be present – all components will communicate over virtual network within Docker. This means that it will not be possible to simulate unstable connections and fading radio signal strength. These aspects will be tested at a later stage using a physical demonstrator.

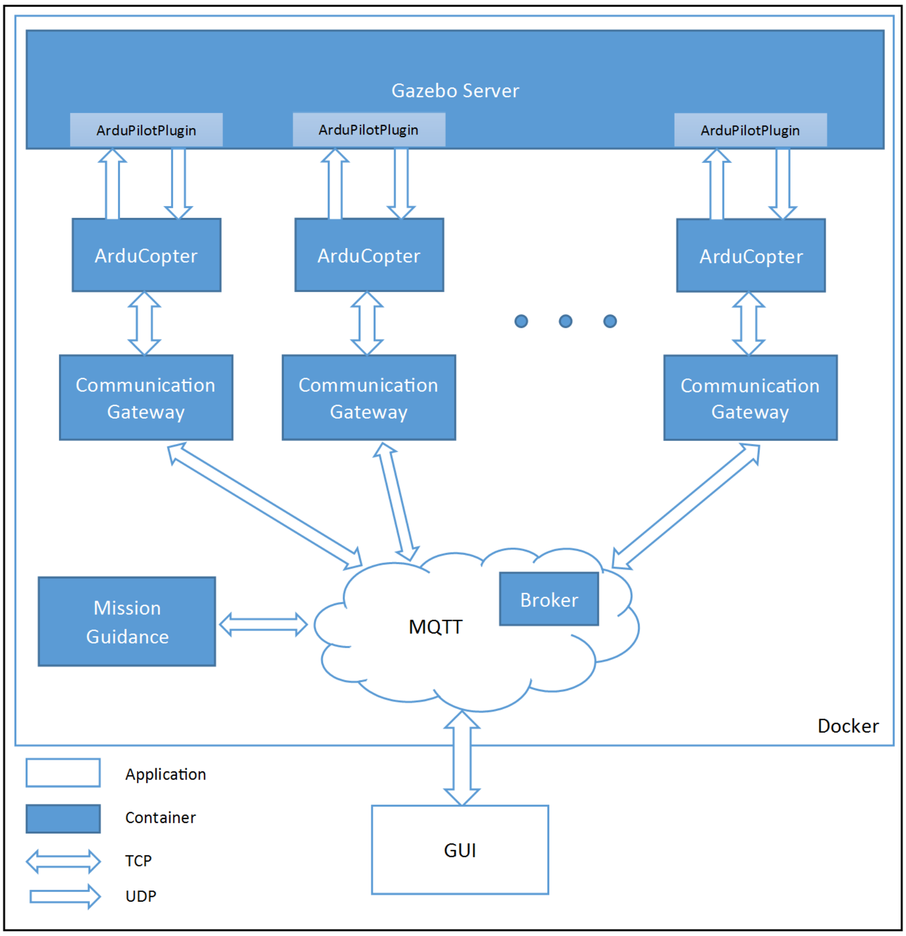

Figure 3-29 MSP virtual demonstrator diagram

The initial test environment will be prepared using Docker Compose and a set of containers – one for each service. This will allow for easy migration to RAINBOW at a later stage. Radio links will be replaced with TCP connections. The drone firmware (ArduCopter) will be compiled in SITL simulator mode and configured to connect to the Gazebo server through a dedicated plugin. Gazebo allows to simulate a 3D virtual world with accurate physics. ArduPilotPlugin receives motor control commands from firmware and transforms them into forces acting on the drone model in Gazebo. Changes in position and orientation of the model are then transformed into simulated sensor readings and sent back to the firmware. Firmware in SITL mode allows to programmatically introduce random errors to these readings. The other components do not require any modifications in relation to the physical demonstrator.

3.3.3 Usage of RAINBOW Tools and Assets

The demonstrator depends on the RAINBOW mesh networking stack to provide reliable communication, better range and more efficient coverage. The system will be deployed in a plain field where no established infrastructure is present. Antennas will be erected on temporary masts and connections might be unstable due to wind. Given the requirement for rapid relocation of the ground station nodes' components required for the acquisition of reliable and high-quality data, the RAINOW mesh network autoconfiguration will prove to be of utmost importance. Network autoconfiguration will enable the efficient and quick establishment of links between GCSs and drones, thus significantly reducing setup effort and down-time. Additionally, the RAINBOW mesh network autoconfiguration component will enable an efficient and flawless drone handover during the exchange of GCS "jurisdiction" mid-flight.

Since power lines are critical infrastructure and any damage might have fatal consequences, the system will depend on security primitives offered by the RAINBOW to ensure that no rogue stations can connect and hijack the drone.

The system will heavily depend on the RAINBOW data store to ensure that nodes can share data easily to coordinate the drone operations and preserve their state while they are powered off for relocation.

RAINBOW orchestrator policies will allow implementing a single service that controls the swarm. The orchestrator will ensure that there is always a single instance of this service running. Since this is a core service and it needs to communicate with all other services to assign tasks to drones, the orchestrator can also choose the node where this service will be running in such a way, that the latency to the furthest node is minimized. Such an approach eliminates the need for distributed algorithms that would have to be implemented if the task for drones would be generated collectively by a group of services. This substantially reduces implementation effort.

3.3.4 Demonstrator Specific KPIs

#	Values
ID	MSP-KPI-01
KPI Title	KPI Title: time to pass the control over the drone from one GCS to another GCS
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	At the moment that GCS that controls the drone is switched off (for any reason) or the drone flies away beyond the radio link range, the control over the drone has to be passed to another GCS
RAINBOW's Contribution	RAINBOW mesh network provides communication over unreliable links, which allows services to execute drone handover from one GCS to another in atomic[6] fashion
KPI Measurement Formula	time of confirmed connection with GCS_2 – time of break of connection with GCS_1
Current Value (without RAINBOW)	not measurable in the "as is" scenario – no such situation occurs
Expected target value (with RAINBOW)	4 sec

#	Values

[6] atomic: either passes the control or not

ID	MSP-KPI-02
KPI Title	KPI Title: increase of productive flight distance per drone
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	The drone does not need to return to the starting point for landing – the control over a drone can be handed over to another ground station node and land near it. This reduces the length of unproductive parts of the mission.
RAINBOW's Contribution	RAINBOW allows a group of services to share data to automatically plan drone operations in a more efficient manner in comparison to the typical "set of individual drones" scenario.
KPI Measurement Formula	$(FD_2-FD_1)*100/FD_1$ where: FD1 – average flight distance of a single drone in the "as is" scenario FD2 – average flight distance of a single drone in the RAINBOW supported scenario
Current Value (without RAINBOW)	0%
Expected target value (with RAINBOW)	50%

#	**Values**
ID	MSP-KPI-03
KPI Title	KPI Title: reduction of data acquisition time per kilometer of power line.
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	In contrast to single drone operations, a group of coordinated drones is able to inspect power lines in a more efficient way, resulting in shorter time needed for the inspection of an average 1 km of power line.
RAINBOW's Contribution	RAINBOW allows for a quick deployment of distributed GCS's nodes along the power line. Automated coordination allows for more efficient data acquisition.
KPI Measurement Formula	$\|T_2 - T_1\|*100/T_1$ where: T_1 – an average time of inspection in the "as is" scenario T_2 – an average time of inspection in the RAINBOW supported scenario
Current Value (without RAINBOW)	0

Expected target value (with RAINBOW)	50%

#	**Values**
ID	MSP-KPI-04
KPI Title	KPI Title: reduction of overlaps between individual flight routes
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	Coordination of flights of drones within the group tends to monitor and optimize flight phases related to finishing operation of one drone and starting operation of the next drone in the way that assures a minimum required overlap of subsequent flight during inspection. This will be possible owing to accurate and consistent monitoring and use of recorded coordinates to plan subsequent flights.
RAINBOW's Contribution	RAINBOW allows GCS nodes to share data about mission completion status and data quality in real time.
KPI Measurement Formula	$\|OD_2 – OD_1\|*100/OD_1$ where: OD_1 – an average overlap distance between subsequent flights in the "as is" scenario OD_2 – an average overlap distance between subsequent flights in the RAINBOW supported scenario
Current Value (without RAINBOW)	0
Expected target value (with RAINBOW)	-75%

#	**Values**
ID	MSP-KPI-05
KPI Title	KPI Title: efficiency of battery usage for a productive phase of the drone flight
KPI Type	Business Case Specific
Objective / Subjective	Objective
Need Relevant to KPI	The drone can use the energy in a more efficient way, using more of it for flight and decreasing safe margin for landing maneuvers
RAINBOW's Contribution	RAINBOW allows GCS nodes to implement the drone handover in an atomic fashion over unreliable network links, which allows for shorter returns paths - drones can land at the nearest GCS node.
KPI Measurement Formula	$\|BU_2 – BU_1\|*100/BU_1$ where:

	BU_1 – an average battery usage (in mAh) in a productive flight phase in the "as is" scenario BU_2 – an average battery usage (in mAh) in a productive flight phase in the RAINBOW supported scenario
Current Value (without RAINBOW)	0
Expected target value (with RAINBOW)	55%

Table 3-5 RAINBOW Business KPIs (section 2.2.2.2) applicable to Demonstrator #3

ID	KPI	AS-IS Value	TO-BE Value
RAINBOW-KPI-01	Deployment Time	N/A	< 2 sec
RAINBOW-KPI-03	Security Incidents	N/A	< 1 / year
RAINBOW-KPI-04	Service Availability	N/A	> 99%
RAINBOW-KPI-08	User Satisfaction	N/A	> 70%

3.4 Evaluation of RAINBOW Assets

The diversity of the different demonstrators in RAINBOW allows for testing the platform and its offerings in its whole; however, not every single demonstrator will be able to evaluate all the tools of RAINBOW, or at least not in their full potential, as the usage and the intensity of their usage is highly dependent on the scope of each demonstrator.
The following tables provide a high-level view on which specific components per architecture layer [3] of the overall platform will be evaluated by each demonstrator, using a three-mark ladder approach:

- **-** = Not Relevant, none or very low component usage by the demonstrator
- **+** = Low Usage by the demonstrator, component not in the core of the development or of the business need
- **++** = High Usage, component will be used to serve the development and business needs

Table 3-6 Usage of Modeling Layer components

Demonstrator	Service Graph Editor & Analytics Editor	Policy Editor
BIBA	++	++
CRF	+	++
MSP	-	++

Table 3-7 Usage of Orchestration Layer components

Demon-strator	Predeploy-ment Constraint Solver	Deploy-ment Manager	Orches-tration Lifecycle Manager	Resource Manager	Resource & Application -level Monitoring
BIBA	++	++	++	++	++
CRF	-	-	+	++	++
MSP	-	++	+	++	-

Table 3-8 Usage of Mesh Layer components

Demonstrator	Mesh Routing Protocol Stack	Multi-domain sidecar Proxy	Security Enablers
BIBA	++	++	+
CRF	-	-	++
MSP	++	+	++

Table 3-9 Usage of Data Management & Analytics Layer components

Demonstrator	Data Storage & Sharing	Analytics Engine
BIBA	++	++
CRF	-	-
MSP	++	-

3.5 Critical Issues, Obstacles and Mitigation

Table 3-10 Identified risks and mitigation measures

Risk Description	Severity Level (1-3)	Mitigation Measures
[Demonstrator #1] System latency is not deterministic increasing risk with regards to personnel safety in workspace	3	Provide fences and slow down robot operations, increase safety distance between personnel and robot.
[Demonstrator #2] End-to-end latency depends on both single components' latency and external network overload. While the first one is under our control,	2	The connections used will be cabled where possible. Moreover, single components will minimize their internal latencies.

Risk Description	Severity Level (1-3)	Mitigation Measures
the latter is unpredictable		
[Demonstrator #3] unstable communication caused by interference from external sources or terrain configuration or the presence of objects that interfere with the propagation of electromagnetic waves, which may result in the activation of failsafe procedures and problems with initializing edge devices and / or nodes, or control transferring between nodes	2	Choosing the right places for locating GCSes (open places, elevations), using efficient antenna systems, using effective radio links, using other methods of data transmission and communication (LTE, 5G), locating GCSes at reasonable distance between them (not approaching expected max local wifi range).
[Demonstrator #3] possibility that the flying drone will not be able to find a communication with the next GCS after losing range with the previous GCS	1	Adhere to the rules of the correct GCS deployment, appropriate programming of failsafe procedures allowing to return to the previous GCS.

4 Demonstrators' Implementation Plan

4.1 Demonstrators' Planning and Evaluation Phases

Following the foreseen release schedule of the RAINBOW platform as a whole (first release on M15 and second release on M27), the three demonstrators will execute "early stage" and "advanced stage" demonstrator's evaluation. Such division of the evaluation in "early' and "advanced" is driven by the maturity level of the platform and the readiness level of the demonstrator to uptake the use of the platform.
The result of this division, with the information available at M18 of the project, has been laid out as an implementation plan per demonstrator in this section; test scenarios which shall be performed during the "early stage" (i.e. between M18 and M21) and test scenarios which shall be left for the "advanced stage" (i.e. between M30 and M33). Obviously, test scenarios which shall be performed during the first test run could be executed again during the second test run, i.e. this division is not exclusive, rather a planning effort for year 2 and year 3 of the project.
Furthermore, each demonstrator has provided several test cases which shall guide the testing effort. The identified test cases have been also linked to the identified test scenarios, in an effort to further prescribe the testing effort in year 2 and year 3 of the project. Our intention is to indicate which test cases should be (at least) executed in order to conclude the test scenario. However, the reader should keep in mind that at the time of the writing of the present deliverable this is a first effort, which will be further refined per demonstrator in the specific deliverables which shall report the evaluation of the RAINBOW platform at each demonstrator (D6.2, D6.4 and D6.6 respectively for the first version of the platform and D6.3, D6.5 and D6.7 respectively for the second version of the platform).

4.2 Demonstrator #1 – BIBA

4.2.1 Scenarios

The test scenarios identified for the first demonstrator are the following:

Table 4-1 Demonstrator #1 Scenarios

ID	Scenario	Scenario's Scope
BIBA-TS-1	Horizontal Up-scaling and Down-scaling of Personnel Localization and Motion Capturing (PLMC) Service.	Scalability of Services
BIBA-TS-2	Horizontal Up-scaling and Down-scaling of Robot Motion Tracking Service (RMT).	Scalability of Services
BIBA-TS-3	Horizontal Up-scaling and Down-scaling of Collision Prediction and Avoidance (CPA) Service.	Scalability of Services

ID	Scenario	Scenario's Scope
BIBA-TS-4	Deterministic System Latency within prescribed maximum tolerance for jitter.	System Latency
BIBA-TS-5	Continuously monitor and evaluate mentioned SLO's against run time policies set by the Service operator. If SLO's are not met, then suitable actions as defined by Service Operator in run time policies are to be taken. In this use case if SLO's are not met the Robots are stopped immediately to prioritize human safety.	Monitoring and evaluation of Service Level Objective's (SLO's)
BIBA-TS-6	Only registered user with valid Credentials and access right must be able to get access to data from the RAINBOW platform. Also applicable for data sharing with 3rd parties.	Data sharing
BIBA-TS-7	Secure on-boarding of new fog device by adhering to attestation policies by providing verifiable evidence on their configuration integrity and correctness is allowed to join with other fog devices in the RAINBOW mesh network.	Security and Attestation
BIBA-TS-8	Dynamic resource allocation from nearby fog device due to resource unavailability.	Dynamic resource allocation
BIBA-TS-9	Support for Data analyst to express complex, customized query in user-friendly and intuitive way and the result is returned to the dashboard/editor.	Data Analytics

4.2.2 Test Cases

The test cases identified for the first demonstrator are the following:

Table 4-2 Demonstrator #1 Test Cases

Test Case ID	HRC-TC-01
Test Case Title	Horizontal upscaling of Personnel Localization and Motion Capturing Service (PLMC)
Actors	• Centralized backend Orchestrator • Personnel Localization and Motion Capturing Service (PLMC)
Pre-conditions	• Personnel (P1) entering a collaborative workspace area (W1). • The Fog device (FD1) has enough resources to accommodate service corresponding to new personnel in the workspace (W1).
Post-conditions:	• A new instance of PLMC service corresponding to the P1 is spun in FD1 within 1000 milliseconds of position tag discovery by Data Aggregator.
Normal Flow	1. P1 enters W1. 2. P1's positioning tags discovered by Data Aggregator. 3. A new instance of PLMC service corresponding to the new personnel entered is spun in FD1 within 1000 milliseconds of position tag discovery.
Pass Metrics	• PLMC service corresponding to P1 is instantiated in FD1. • The new PLMC service subscribes to localization and motion data from that P1. • Time taken to spin a new instance of PLMC service corresponding to P1 has been less or equal to 1000 millisecond.
Fail Metrics	• PLMC service corresponding to P1 is not instantiated in FD1. • The new PLMC service fails to subscribe to localization and motion data from that P1. • Time taken to spin a new instance of PLMC service corresponding to P1 has been greater than 1000 millisecond.
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Scalability of Services

Test Case ID	HRC-TC-02
Test Case Title	Horizontal downscaling of PLMC service
Actors	• Centralized backend Orchestrator • Personnel Localization and Motion Capturing Service (PLMC)
Pre-conditions	• Personnel (P1) recognized and registered by fog device in collaborative workspace area (W1) moves out of the workspace area.
Post-conditions	• Data Aggregator losses wireless signal from P1's Positioning Tag. • PLMC service unsubscribes from the localization and motion data from P1. • PLMC service corresponding to P1 is terminated.
Normal Flow	1. When P1 moves out of W1. 2. Data Aggregator recognizes wireless signal link loss of P1's positioning tags. 3. The respective instance of PLMC service must be terminated in FD1.
Pass Metrics	• PLMC service unsubscribes from the localization and motion data from P1. • PLMC service corresponding to P1 is terminated.
Fail Metrics	• PLMC service fails to unsubscribe from the localization and motion data from P1. • PLMC service corresponding to P1 not terminated.
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Scalability of Services

Test Case ID	HRC-TC-03
Test Case Title	Upscaling of Robot Motion Tracking Service (RMT)
Actors	• Centralized backend Orchestrator • Robot Motion Tracking (RMT) Service

Pre-conditions	• A new Robot (RA1) is powered-on within the workspace area (W1). • The Fog device (FD1) has enough resources to accommodate the new Robot in the workspace.
Post-conditions	• A new instance of RMT service corresponding to the RA1 is spun in FD1 within 1000 milliseconds of discovery of RA1's presence by IoT Gateway.
Normal Flow	1. RA1 is powered-on in W1 2. RA1's presence is discovered by IoT Gateway. 3. A new instance of RMT service corresponding to the RA1 is spun in FD1 within 1000 milliseconds of position tag discovery.
Pass Metrics	• A new instance corresponding to RA1 is spun in FD1 • New instance of RMT service subscribes to motion data from RA1. • Time taken to spin new instance of RMT service corresponding to RA1 has been less than or equal to 1000 millisecond
Fail Metrics	• A new instance corresponding to RA1 is not spun in FD1 • New instance of RMT service fails to subscribe to motion data from RA1. • Time taken to spin new instance of RMT service corresponding to RA1 has been greater than 1000 millisecond
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Scalability of Services

Test Case ID	HRC-TC-04
Test Case Title	Downscaling of Robot Motion Tracking Service (RMT)
Actors	• Centralized backend Orchestrator • Robot Motion Tracking (RMT) Service
Pre-conditions	• A Robot (RA1) is powered-off from the workspace area (W1).
Post-conditions	• A instance of RMT service corresponding to the RA1 is terminated in FD1 when the IoT gateway loses motion data signal from RA1.

Normal Flow	1. When RA1 is power-off in W1 2. IoT Gateway loss motion data from of RA1. 3. PLMC service corresponding to RA1 unsubscribes for motion data from RA1. 4. PLMC service corresponding to RA1 is terminated in FD1.
Pass Metrics	• RMT service unsubscribes from the motion data from RA1. • RMT service corresponding to RA1 is terminated.
Fail Metrics	• RMT service fails to unsubscribe from the motion data from RA1 • RMT service corresponding to RA1 not terminated
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Scalability of Services

Test Case ID	HRC-TC-05
Test Case Title	Upscaling of Collision Avoidance and Prediction (CAP) Service
Actors	• Centralized backend Orchestrator • Collision Avoidance and Prediction (CAP) Service
Pre-conditions	• Number of Personnel and/or robots (here on called as participants in the context of CAP service) powered-on within the workspace area (W1) exceed a prescribed group size (say group size is 10 which includes count of Personnel and Robots combined) for one CAP service. • The Fog device (FD1) has enough resources to accommodate new instance of CAP in W1.
Post-conditions	• Number of participants in group exceeds the prescribed group size in W1. • New instance of CAP service is spun corresponding to new participants added to the new group. • Time taken to spin new instance of CAP service must be less than or equal to 1000 milliseconds. • New instance CAP service subscribes for data from corresponding participants PLMC and RMT service instances.

Normal Flow	1. Number of participants in group exceeds the prescribed group size in W1. 2. New instance of CAP service is spun corresponding to new participants added to the new group. 3. Time taken to spin new instance of CAP service must be less than or equal to 1000 milliseconds. 4. New instance CAP service subscribes for data from corresponding participants PLMC and RMT service instances.
Pass Metrics	• New instance of CAP service is spun corresponding to new participants added to the new group. • Time taken to spin new instance of CAP service must be less than or equal to 1000 milliseconds. • New instance CAP service subscribes for data from corresponding participants PLMC and RMT service instances.
Fail Metrics	• New instance of CAP service is not spun corresponding to new participants added to the new group. • Time taken to spin new instance of CAP service must be greater than 1000 milliseconds. • New instance CAP service fails to subscribe for data from corresponding participants PLMC and RMT service instances.
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Scalability of Services

Test Case ID	HRC-TC-06
Test Case Title	Down-scaling of Collision Avoidance and Prediction (CAP) Service
Actors	• Centralized backend Orchestrator • Collision Avoidance and Prediction (CAP) Service
Pre-conditions	• Number of Personnel and/or robots (here on called as participants in the context of CAP service) powered-off within the workspace area (W1) recede a prescribed group size (say group size is 10 which includes count of Personnel and Robots combined) for one CAP service.

Post-conditions	• Number of participants in group recedes in W1. • Instance of CAP service is terminated corresponding to group with zero personnel in the group.
Normal Flow	1. Number of participants in group recedes in W1. 2. Instance of CAP service is terminated corresponding to group with zero personnel in the group. 3. Instance of CAP service terminated must unsubscribes for data from corresponding participants PLMC and RMT service instances.
Pass Metrics	• Instance of CAP service is terminated corresponding to group with zero personnel in the group. • Instance of CAP service terminated must unsubscribes for data from corresponding participants PLMC and RMT service instances.
Fail Metrics	• Instance of CAP service is not terminated with zero personnel in the group. • Instance of CAP service is to be terminated fails to unsubscribes for data from corresponding participants PLMC and RMT service instances.
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Scalability of Services

Test Case ID	HRC-TC-07
Test Case Title	System latency and jitter
Actors	• RAINBOW Mesh stack • Robot Motion Tracking (RMT) Service • Collision Avoidance and Prediction (CAP) Service • Personnel Localization and Motion Capturing (PLMC) Service

Pre-conditions	• Personnel (P1) and Robot (RA1) must be in collision course in workspace (W1). • Position tag of P1 must be wirelessly sending data to Data Aggregator of W1. • RA1 data is received by IoT Gateway of W1. • RMT, CAP, PLMC service corresponding P1 and RA1 must be running.
Post-conditions	• RA1 must be stopped with system latency being as follows ○ System Latency < ($T_{DA} + T_{DP} + T_R + T_{stop} \pm T_{jitter}$) • Given use case specific parameters T_{DA} , T_{DP} , T_R , T_{stop} . These parameters are specific to robot manufacture, algorithm and task program given to the robot, speed of the robot performing the task. • T_{jitter} must be less than 200 milliseconds.
Normal Flow	System Latency can be divided into 4 parts • **Data Acquisition Latency (T_{DA})** is time taken for acquiring Robot motion data, Personnel localization and motion data from remote node and sending the data to respective services in Fog devices. • **Data Processing latency (T_{DP})** is time taken by CAP, RMT, PLMC services for processing the data. • **Robot reaction time (T_R)**: Time taken for stop signal which is generated by CAP service in Fog device to be received by Industrial PC (IPC) of the Robot • **Robot stop time (T_{stop})**: Time taken to stop the Robot, once stop signal is acted upon. This latency depends on operating speed, type and manufacturer of Robot. • **Jitter (T_{jitter})**: is time jitter in receiving the packets due to network condition, topology, routing mechanism etc. In an event of predicted collision, the service operator expects the System Latency to be **System Latency = $T_{DA} + T_{DP} + T_R + T_{stop} \pm T_{jitter}$** *$T_{DA}$,T_{DP} ,T_R ,T_{stop} are application case specifics and changes based on Robot manufactured, algorithm, Processing IT infrastructure, and protocol used.* System Latency must have jitter of range less than 200 milliseconds.
Pass Metrics	• **System Latency < $T_{DA} + T_{DP} + T_R + T_{stop} \pm T_{jitter}$** • T_{jitter} less than or equal to 200 milliseconds

Fail Metrics	• **System Latency > T_{DA} + T_{DP} + T_R + T_{stop} ± T_{jitter}** • T_{jitter} greater than 200 milliseconds
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	System Latency

Test Case ID	HRC-TC-08
Test Case Title	Continuous monitoring and evaluation of Service Level Objective's (SLO's) of services running in fog devices.
Actors	• Policy editor • Resource Application-level Monitoring in Centralized Orchestration Backend • Multi-domain sidecar proxy in RAINBOW Mesh stack
Pre-conditions	• Service Operator using Policy editor defines SLO’s to monitor and set of measures to be taken if SLO’s are not met. • RMT, CAP, PLMC service corresponding Personnel and robots in workspace W1 must be running. • RAINBOW mesh stack along with cluster head must be functional.
Post-conditions	• Continuously monitor and evaluate mentioned SLO's against run time policies set by Service operator. • If SLO's are not met then suitable actions as defined by Service Operator in run time policies are to be taken. In this use case if SLO's are not met the Robots are stopped immediately to prioritize human safety.
Normal Flow	1. Service Operator using Policy editor defines SLO’s to monitor and set of measures to be taken if SLO’s are not met. 2. Continuously monitor and evaluate mentioned SLO's against run time policies set by Service operator. 3. When SLO's are not met then suitable actions as defined by Service Operator in run time policies are to be taken. In this use case if SLO's are not met the Robots are stopped immediately to prioritize human safety.

Pass Metrics	• SLO's are monitored continuously. • If SLO's are not met then suitable actions as defined by Service Operator in run time policies are to be taken. In this use case if SLO's are not met the Robots are stopped immediately to prioritize human safety.
Fail Metrics	• SLO's are not monitored. • If SLO's are not met then suitable actions as defined by Service Operator in run time policies failed to be executed taken. In this use case this cause compromise in safety of personnel in the workspace.
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Monitoring and evaluation of Service Level Objective's (SLO's)

Test Case ID	HRC-TC-09
Test Case Title	Data sharing within organization and 3rd parties
Actors	• Policy editor • Service editor • Analytical editor • Dashboard
Pre-conditions	• User must be registered with the RAINBOW platform and have a valid credentials. • User must have access rights to the data.
Post-conditions	• Only registered user with valid Credentials and access right must be able to get access to data.

Normal flow	1. Request to login from service operator. 2. Service operator provides Credentials to user and also set access right settings for the requested user. 3. User Logins using Credentials provided. 4. Credentials are validated. 5. If access rights are given for the data the data is provided to requested user. 6. User within organization with valid credential and access must be able to query data using RAINBOW query editor/dashboard. 7. Data sharing with 3rd party using API tokenization must be supported.
Pass Metrics	• Only registered user with valid Credentials and access right must be able to get access to data.
Fail Metrics	• A registered user is not allowed access to the RAINBOW platform. • Un-registered or registered user with incorrect credentials and/or access right is able to get access to data. • RAINBOW platform is offline. • API tokenization is not functional.
Notes and Issues	• An un-registered user must be redirected to the register page/area first. • For successful user registration, approval from the Service Operator is needed.
Major RAINBOW Use Cases Tackled	Data sharing

Test Case ID	HRC-TC-10
Test Case Title	Secure on-boarding of new fog device
Actors	Security and Attestation enablers
Pre-conditions	• New fog device is added in the infrastructure. • Service operator defines attestation policies using policy editor.

Post-conditions	• New fog device that adheres to attestation policies by providing verifiable evidence on their configuration integrity and correctness is allowed to join with other fog devices in RAINBOW mesh network.
Normal Flow	1. New fog device is added in the infrastructure. 2. Service operator defines attestation policies using policy editor. 3. Fog devices provides verifiable evidence on their configuration integrity and correctness. 4. Fog device is added to RAINBOW mesh network.
Pass Metrics	• New fog device that adheres to attestation policies by providing verifiable evidence on their configuration integrity and correctness is allowed to join with other fog devices in RAINBOW mesh network.
Fail Metrics	• New fog device that does not adheres to attestation policies is allowed to join with other fog devices in RAINBOW mesh network. • New fog device that does adheres to attestation policies is not allowed to join with other fog devices in RAINBOW mesh network.
Notes and Issues	
Major RAINBOW Use Cases Tackled	Security and Attestation

Test Case ID	HRC-TC-11
Test Case Title	Dynamic resource allocation
Actors	• Centralized backend Orchestrator • Personnel Localization and Motion Capturing (PLMC) Service

Pre-conditions	• A personnel (P1) entering a collaborative workspace area (W1). The Fog device (FD1) of respective workspace area (W1), does not have enough resources to accommodate P1's localization and motion capturing service. On other hand a Fog Device (FD2) in the immediate neighboring workspace (W2) has enough hardware and software resources. • Adding to this it is assumed suitable policies are set by Service Operator and Service Level Objectives for Dynamic resource allocation are met.
Post-conditions	• When P1 enters a workspace W1. Upon discovering the presence of P1's Tag. FD1 of W1 looks for availability of resources. As in this case scenario, FD1 does not have enough resources. Based on the policies set and satisfaction of Service Level Objectives, a new instance of PLMC need to be spun by the help of RAINBOW platform in FD2 .
Normal Flow	• Service operator sets suitable policies using policy editor to support Dynamic resource allocation. • FD1 does not have enough resources to accommodate P1's localization and motion capturing service. • FD2 in the immediate neighboring workspace (W2) has enough hardware and software resources. • P1 enters the W1. • Based on the policies set and satisfaction of Service Level Objectives, a new instance of PLMC need to be spun by the help of RAINBOW platform in FD2
Pass Metrics	• New instance of PLMC service corresponding to P1 is spun in in FD2 • New instance of PLMC service corresponding to P1 must subscribe to localization and motion data from P1 position tag. • Time taken to spin new instance of PLMC service corresponding to P1 must be less than 1000 milliseconds.
Fail Metrics	• New instance of PLMC service corresponding to P1 is not spun in in FD2 • New instance of PLMC service corresponding to P1 fails to subscribe to localization and motion data from P1 position tag. • Time taken to spin new instance of PLMC service corresponding to P1 must be less than 1000 milliseconds.
Notes and Issues	N/A

Major RAINBOW Use Cases Tackled	Dynamic resource allocation

Test Case ID	HRC-TC-12
Test Case Title	Data query for performing data analytics
Actors	• Analytical engine • Query editor • Dashboard
Pre-conditions	• Editor and query language to express data query in more user-friendly and intuitive way • Analytical engine optimizes user queries to fetch data from Distributed database across Fog device mesh network • Supports for high level functions in query language
Post-conditions	• Data analyst is able to express complex, customized query in user-friendly and intuitive way and result is returned to the dashboard/editor.
Normal Flow	1. Data analyst is able to express complex, customized query with high level functions (such as sliding window averaging, periodic querying etc.) in user-friendly and intuitive way 2. Editor checks for syntax validity of the query. If query is valid forwards, it to analytical engine. 3. Analytical engine optimizes user queries to fetch data from Distributed database across Fog device mesh network. 4. Query result is displayed on dashboard/editor.
Pass Metrics	• Editor must only allow query with valid syntax and structure only. • Analytical engine optimizes user queries to fetch data from Distributed database across Fog device mesh network. • Query result is displayed on dashboard as per user setting in dashboard.

Fail Metrics	• Editor only allow query with invalid syntax and structure only. • Analytical engine does not allow user queries to fetch data from Distributed database across Fog device mesh network. • Query result is not displayed on dashboard.
Notes and Issues	N/A
Major RAINBOW Use Cases Tackled	Data Analytics

4.2.3 Implementation Plan

The implementation plan for the first demonstrator; test scenarios which shall be performed during the "early stage" (i.e. between M18 and M21) and test scenarios which shall be left for the "advanced stage" (i.e. between M30 and M33).
Furthermore, the test cases linked to each identified test scenario, in an effort to further prescribe the testing effort in year 2 and year 3 of the project (*further refinement per demonstrator expected in deliverables D6.2 and D6.3*).

Table 4-3 Demonstrator #1 Implementation Plan

Related Scenario	Test Cases	Due by
BIBA-TS-1	HRC-TC-01 HRC-TC-02	Early (M21)
BIBA-TS-2	HRC-TC-03 HRC-TC-04	Early (M21)
BIBA-TS-3	HRC-TC-05 HRC-TC-06	Early (M21)
BIBA-TS-4	HRC-TC-07	Advanced (M33)
BIBA-TS-5	HRC-TC-08	Advanced (M33)
BIBA-TS-6	HRC-TC-09	Advanced (M33)
BIBA-TS-7	HRC-TC-10	Advanced (M33)
BIBA-TS-8	HRC-TC-11	Advanced (M33)
BIBA-TS-9	HRC-TC-12	Advanced (M33)

4.3 Demonstrator #2 – CRF

4.3.1 Scenarios

The test scenarios identified for the second demonstrator are the following:

Table 4-4 Demonstrator #2 Scenarios

ID	Scenario	Scenario's Scope
CRF-TS-1	Lab-to-lab AHED service orchestration	Demonstrate how RAINBOW allows to migrate the AHED service to keep bandwidth occupancy and energy consumption within a predefined value range
CRF-TS-2	Lab-to-lab AHED service orchestration and alert dissemination	Demonstrate how RAINBOW allows to migrate the AHED service from the MEC server to the RSU and vice versa based on the hazardous event detection. Alerts are also disseminated through the AMQP broker and received by simulated vehicle.
CRF-TS-3	City Aggregator online platform visualization	Demonstrate how City Aggregator demodulates DENM messages and shows hazardous area on the online platform
CRF-TS-4	On the road Vehicle Alerts from PC5 interface	Demonstrate how a real vehicle on the road and AHED service authenticated, subscribes for AHED service and receives DENM messages on PC5 interface when an upcoming alert occur (High priority Alert).
CRF-TS-5	On the road Vehicle Alerts from Uu interface	Demonstrate how a real vehicle on the road can subscribe to the AHED service and receive, though the RAINBOW Trusted enabler, DENM messages on Uu interface when Alerts occur (Low priority Alert).
CRF-TS-6	Alert reception via Citizen App	Demonstrate how City Aggregator displays alerts from the Citizen App through AMQP Broker and it forwards DENM messages to vehicles in hazard area on vehicles' Uu interface.

4.3.2 Test Cases

The test cases identified for the second demonstrator are the following:

Table 4-5 Demonstrator #2 Test Cases

Test Case ID	AHED-TC-01
Test Case Title	AHED orchestration (network monitoring based)
Actors	**MEC Node, RSU Fog Node**
Pre-conditions	• AHED service running or on the MEC node or on the RSU node • No control on the MEC – RSU communication bandwidth occupancy
Post-conditions	• AHED service running on the MEC node and on the RSU node by continuously migrating between the two nodes, based on policy constraints defined at development stage • Continuously monitoring of network status
Normal Flow	1. AHED service start running on the MEC node 2. Network bottleneck occurs and video stream fps is no more enough to ensure reliable signaling of alert 3. RAINBOW migrate the AHED service on the RSU fog node, changing the RSU power mode from normal to max power (from 15W to 30W)
Pass Metrics	• service migration occurs
Fail Metrics	• service migration does not occur
Notes and Issues	
Major RAINBOW Use Cases Tackled	• The network status is monitored, and migration allows to meet video stream fps required by the service

Test Case ID	AHED-TC-02
Test Case Title	AHED orchestration (energy consumption-based)
Actors	**MEC Node, RSU Fog Node**
Pre-conditions	• AHED service running or on the MEC node or on the RSU node • No control on the RSU energy consumption

Post-conditions	• AHED service running on the MEC node and on the RSU node by continuously migrating between the two nodes, based on policy constraints defined at development stage • Continuously monitoring of energy consumption
Normal Flow	1. AHED service start running on the RSU node 2. Energy consumption rises up to the pre-defined threshold 3. RAINBOW forward the video stream from the RSU to the MEC node 4. RAINBOW migrate the AHED service on the MEC node, changing the RSU power mode from max power to normal (from 30W to 15W)
Pass Metrics	• service migration occurs
Fail Metrics	• service migration does not occur
Notes and Issues	
Major RAINBOW Use Cases Tackled	• The RSU power consumption is monitored, and migration allows to improve service energy efficiency

Test Case ID	AHED-TC-03
Test Case Title	AHED orchestration (event based) and alert dissemination
Actors	**Vulnerable Road User (VRU), MEC Node, RSU Fog Node, Cloud node**
Pre-conditions	• AHED service running or on the MEC node or on the RSU node • No event-based action
Post-conditions	• AHED service running on the MEC node and on the RSU node by continuously migrating between the two nodes, based on the occurrence of the alert detection
Normal Flow	1. AHED service starts running on the MEC node 2. The AHED service detect the presence of an animal on the road 3. RAINBOW migrate the AHED service on the RSU fog node to reduce latency and increase results reliability, changing the RSU power mode from normal to max power (from 15W to 30W) 4. Alert is disseminated through the Cloud message broker

Pass Metrics	• service migration occurs in time • Improvement on latency and accuracy of detected animal during alert period
Fail Metrics	• Service migration does not occur in time • No improvement on latency and accuracy of detected animal during alert period
Notes and Issues	
Major RAINBOW Use Cases Tackled	• The AHED service result is monitored, and event-based migration allows to improve service performances

Test Case ID	AHED-TC-04
Test Case Title	AHED authenticated and privacy protected access
Actors	**Vehicle, AHED service, RAINBOW Orchestrator**
Pre-conditions	• The vehicle is without credentials and Keys to access at AHED service • The vehicle is not subscribed to the AHED service
Post-conditions	• DAA cert (cred) + DAA key are stored in the telematic box • The Vehicle is authenticated and its correct state is verified
Normal Flow	1. Setup: Registration and secure enrolment of the vehicle's telematic box to the RAINBOW Orchestrator 2. Join: The vehicle engages with the RAINBOW Orchestrator so that it can create the Attestation Identity Credential (AIC) and ECC-based DAA key that will then be used for anonymously signing all the subsequent exchange of messages. This DAA key is stored and managed in the attached TPM key hierarchy 3. Sign/Verify: the communication between the vehicle and the 4. AHED service provider is secured through the RAINBOW DAA Sign and Verify commands
Pass Metrics	• DAA cert (cred) + DAA key are stored in the vehicle that is authenticated to AHED service
Fail Metrics	• The vehicle fails the authentication to AHED service

Notes and Issues	
Major RAINBOW Use Cases Tackled	• UrbanMob.US.4: The vehicle is authenticated and its correct state is verified before being allowed to access the AHED service

Test Case ID	AHED-TC-05
Test Case Title	High priority VRU (Animal) crossing a road
Actors	**VRU (Animal), Vehicle, RSU**
Pre-conditions	• The RSU and vehicle have their C-V2X stack activated • Animal detection algorithm is running • There is an VRU (Animal) on the road in the vehicle trajectory recognized by the RSU
Post-conditions	• RSU and Vehicle establish a secure connection through a PC5 channel • The RSU send Alerts through PC5 channel (DENM messages) with a 10 Hz frequency • The Vehicle send CAM messages and receives and evaluates DENM messages from RSU • The HMI show the Alert to the driver
Normal Flow	1. RSU is detecting, via CAMs reception, the arrival of a vehicle at a relatively short distance (e.g. < 100 m to be adjusted according to vehicle velocity) 2. The RSU detects one VRUs (animal) starting to cross the road. 3. The RSU broadcasts DENMs, signalling the presence of VRUs on the road 4. The vehicle analyse the relevant DENMs for collision risk analysis 5. If a risk of collision is detected, the vehicle triggers an alert to the driver HMI
Pass Metrics	• High Priority alert is showed on the vehicle HMI
Fail Metrics	• High Priority alert doesn't showed on the vehicle HMI before the hazardous situation

Notes and Issues	
Major RAINBOW Use Cases Tackled	• UrbanMob.US.5: RSU and Vehicle are mutually recognized and exchange messages securely and with respect of the privacy

Test Case ID	AHED-TC-06
Test Case Title	Low priority VRU (Animal) crossing a road
Actors	**RSU/MEC Node, VRU (Animal), Vehicle, Cloud Broker**
Pre-conditions	• Vehicle have C-V2X stack activated and HLN Alerts subscribed • Vehicle has C-V2X stack activated and HLN Alerts subscribed • AHED service is running or on the MEC node or on the RSU node • AMQP Broker is running • Animal detection algorithm is running on RSU • There is an VRU (Animal) on the road in the vehicle trajectory recognized by the RSU
Post-conditions	• The AMQP Broker receives DENM messages from the AHED service running on the MEC node or the RSU node • The AMQP Broker forwards DENM messages to the vehicle • The flow of DENM messages (from AHED service to vehicle) is kept active via the AMQP Broker with a 10 Hz frequency (until the alert is finished) • The Vehicle send CAM messages and receives and evaluates DENM messages from AMQP Broker • The HMI show the Low priority Alert to the driver
Normal Flow	1. The RSU detects one VRUs (animal) starting to cross the road. 2. The RSU publish DENM Alerts to the AMQP broker, signaling the presence of VRUs on the road with a 10 Hz frequency (until the alert is finished) 3. AMQP Broker receives DENM messages 4. Geo-position information is extracted from DENM messages 5. AMQP Broker forwards DENM messages to vehicles in the referred area 6. The vehicle analyzes the relevant DENMs for collision risk analysis

	7. If a risk of collision is detected, the vehicle triggers a low priority alert to the driver HMI
Pass Metrics	• Low Priority alert is showed on the vehicle HMI
Fail Metrics	• Low Priority alert doesn't show on the vehicle HMI before the hazardous situation
Notes and Issues	
Major RAINBOW Use Cases Tackled	• UrbanMob.US.5: AMQP Broker and Vehicle are they mutually recognized and exchange messages securely and with respect for privacy

Test Case ID	AHED-TC-07
Test Case Title	City Aggregator online platform visualization
Actors	**AMQP Broker, City Aggregator**
Pre-conditions	• AHED service is running or on the MEC node or on the RSU node • AMQP Broker is running
Post-conditions	• DENM messages are decoded and geo-position information is extracted • Hazardous area is displayed on the City Aggregator
Normal Flow	1. DENM messages are sent from the AHED service running on the MEC or the RSU node 2. DENM messages are received by AMQP Broker 3. AMQP Broker decodes DENM messages and extracts geo-position information 4. Hazardous area is displayed on the City Aggregator
Pass Metrics	• Hazardous area spotted by the MEC or the RSU node is displayed on the City Aggregator online platform
Fail Metrics	• Incorrect or incomplete information is displayed on the City Aggregator

Notes and Issues	
Major RAINBOW Use Cases Tackled	• DENM messages sent from the AHED service result in a highlighted hazardous area displayed on the City Aggregator online platform

Test Case ID	AHED-TC-08
Test Case Title	Citizen App integration
Actors	**Citizen App, AMQP Broker, City Aggregator, Vehicle**
Pre-conditions	• Citizen App is developed • AMQP Broker is active • City Aggregator correctly displays information • Vehicle has C-V2X stack activated and HLN Alerts subscribed
Post-conditions	• Messages from the Citizen App are decoded and geo-position information is extracted • Hazardous area is displayed on the City Aggregator • Vehicles in the referred area are informed of the hazard with DENM messages
Normal Flow	1. Citizen client observes a hazardous situation and rises an alert on the Citizen App 2. Citizen App messages are received by the AMQP Broker 3. AMQP Broker decodes and extracts the geo-position information from the messages 4. Hazardous area is displayed on the City Aggregator online platform 5. AMQP Broker generates DENM messages 6. DENM messages are sent to all vehicles in the hazardous area
Pass Metrics	• Hazardous area spotted by the Citizen App is displayed on the City Aggregator • DENM messages are delivered to vehicles
Fail Metrics	• Incorrect or incomplete information is displayed on the City Aggregator • DENM messages don't reach the vehicle or arrive not in time

Notes and Issues	
Major RAINBOW Use Cases Tackled	• Citizen clients which observe a hazardous situation can rise an alert via the Citizen App. The alert is displayed on the City Aggregator online platform and DENM messages are sent to vehicles in the hazardous area

4.3.3 Implementation Plan

The implementation plan for the second demonstrator; test scenarios which shall be performed during the "early stage" (i.e. between M18 and M21) and test scenarios which shall be left for the "advanced stage" (i.e. between M30 and M33).
Furthermore, the test cases linked to each identified test scenario, in an effort to further prescribe the testing effort in year 2 and year 3 of the project (*further refinement per demonstrator expected in deliverables D6.4 and D6.5*).

Table 4-6 Demonstrator #2 Implementation Plan

Related Scenario	Test Cases	Due by
CRF-TS-1	AHED-TC-01 AHED-TC-02	Early (M21)
CRF-TS-2	AHED-TC-03	Early (M21)
CRF-TS-3	AHED-TC-07	Early (M21)
CRF-TS-4	AHED-TC-05	Advanced (M33)
CRF-TS-5	AHED-TC-04 AHED-TC-06	Advanced (M33)
CRF-TS-6	AHED-TC-08	Advanced (M33)

4.4 Demonstrator #3 – MSP

4.4.1 Scenarios

The test scenarios identified for the third demonstrator are the following:

Table 4-7 Demonstrator #3 Scenarios

ID	Scenario	Scenario's Scope
MSP-TS-1	Flight of drone passing several GCSs in a virtual environment	Three GCSs are deployed in a random order along the power line. A drone is deployed next to the first GCS, takes off and flies passing GCS's and lands near the

ID	Scenario	Scenario's Scope
		last GCS. The control over it is transferred between GCS's during the flight. The scenario is simulated in a virtual environment, as the example of a basic drone's long-range inspection operation.
MSP-TS-2	Multidrone operation in a virtual environment.	Three GCS's are deployed in a random order along the power line. Three drones are deployed, each of them next to one of the GCS. Drones perform individually designed routes and land. Within the scenario, all aspects of the multi-drone operation can be tested, including route planning, control transfer, drone do GCS reconnecting. The scenario is performed in a virtual environment
MSP-TS-3	Flight of drone passing several GCS's in a real environment	The scenario similar to the 1st one: one drone is flying along the power line, passing by three GCS's, with the control over it transferred between sequential stations. The scenario carried out in a real environment, with physical GCS's and drones and in connection to RAINBOW components.
MSP-TS-4	Multidrone operation in a real environment.	The scenario similar to the 2nd one: three drones flying individually between three GCSs simulating multi-drone operations. The scenario carried out in real conditions, with drones and ground control stations deployed in a field, with possibility to test route planning, control transfer, drone do GCS reconnecting.

4.4.2 Test Cases

The test cases identified for the third demonstrator are the following:

Table 4-8 Demonstrator #3 Test Cases

Test Case ID	MSP-TC-01
Test Case Title	Ground control stations deployment and master mission planning
Actors	Drone operators
Pre-conditions	• 3 GCSs are freely positioned within a few hundred meters of each other
Post-conditions	• Defined master route is received by the Mission Guidance Service
Normal Flow	1. After the nodes are switched on, RAINBOW automatically configures the network and GCSs are connected with the mesh network 2. The operator designs the master mission route 3. The master mission route is received by the Mission Guidance Service
Pass Metrics	• Possibility to automatically deploy GCS node, with human role reduced to setting up equipment and connecting the power - achieved
Fail Metrics	• possibility to automatically deploy GCS node, with human role reduced to setting up equipment and connecting the power – not achieved
Notes and Issues	User stories confirmation: • As a drone operator I want to deploy a GCS node in the field • As a drone operator I want to define a master mission through the GCS GUI

Major RAINBOW Use Cases Tackled	• FT2[7] Containerized application packaging • FT4 Application deployment over fog realms • FT5 Application lifecycle management • FT8 Reactive routing • FT10 Zero-touch security fog node configuration • FT11 Fog node "smart" storage

Test Case ID	MSP-TC-02
Test Case Title	A drone carries out the mission beyond a standard communication range
Actors	Drone operators
Pre-conditions	• RAINBOW system running • GCS's nodes are deployed • Master mission designed • Drone is deployed and assigned to one of the GCS's
Post-conditions	• Drone successfully completes the mission with the support of RAINBOW
Normal Flow	1. Mission Guidance Service plans a route for the drone 2. Mission Guidance Service sends a route plan for the drone via Communication Gateway 3. The drone starts the mission 4. When flying the drone reports flight parameters, battery status, acquired images quality parameters 5. When the link quality with the GCS decreases, another GCS with a strong communication link is searched for 6. A new GCS immediately overtakes control over the drone 7. The drone continues the mission without any interruption 8. If the drone cannot connect to another GCS, a failsafe procedure is introduced (e.g. return home)

[7] As laid down in Table 2 "RAINBOW key features as extracted from the RAINBOW Reference Architecture" of deliverable D1.3 [4].

Pass Metrics	• automatic assignment of flight routes to drones – achieved • possibility for a drone to automatically reconnect to other GCS in case of lost radio link instead of executing fail-safe procedure – achieved • ability to pass the control over the drone to another GCS to overcome the constant visibility (VLOS)/radio link range (BVLOS) constraint - achieved
Fail Metrics	• automatic assignment of flight routes to drones – not achieved • possibility for a drone to automatically reconnect to other GCS in case of lost radio link instead of executing fail-safe procedure – not achieved • ability to pass the control over the drone to another GCS to overcome the constant visibility (VLOS)/radio link range (BVLOS) constraint – not achieved
Notes and Issues	For the purpose of the test case, the link quality between GCS and flying drone is intentionally impaired by e.g. moving the flight path away from the GCS position. User stories confirmation: • As a drone operator I want to deploy a new drone • Communication Gateways agree to pass control over the drone from one to another in atomic[8] fashion
Major RAINBOW Use Cases Tackled	• FT3 High-level analytics query editor and job compiler • FT7 Fog-optimized distributed data processing • FT8 Reactive routing • FT9 Adaptive monitoring • FT10 Zero-touch security fog node configuration • FT11 Fog node “smart” storage

Test Case ID	MSP-TC-03
Test Case Title	Several drones carry out the mission
Actors	Drone operators

[8] atomic: either passes the control or not

Pre-conditions	• RAINBOW system running • GCS's nodes are deployed • Master mission designed • two drones deployed, assigned to the GCS's, with automatically designed and delivered route plans
Post-conditions	• Successfully designed and assigned route for the drone, based on tracked parameters
Normal Flow	1. Two drones carry out missions, report flight parameters and battery status 2. One drone completes the mission successfully (all images acquired correctly, the whole mission distance realized) and lands near the operator 3. The second drone must stop the mission earlier (due to the unfavourable conditions, eg. strong wind), returns home and lands 4. Mission Guidance designs a supplementary route 5. The drone take off and carries out the supplementary flight
Pass Metrics	• automatic assignment of flight routes to drones – achieved • optimization of flight paths based on tracked parameters: battery status, overlaps minimization, photo control – achieved • prediction of the copter's flight range depending on the current weather conditions (wind) - achieved
Fail Metrics	• automatic assignment of flight routes to drones – not achieved • optimization of flight paths based on tracked parameters: battery status, overlaps minimization, photo control – not achieved • prediction of the copter's flight range depending on the current weather conditions (wind) – not achieved
Notes and Issues	User stories confirmation: • a drone executes a task that has been assigned to it • the drone finishes its flight and lands near an operator. • Mission Guidance service executes the master mission

Major RAINBOW Use Cases Tackled	• FT1 Constraint and policy editor • FT2 Containerized application packaging • FT4 Application deployment over fog realms • FT5 Application lifecycle management • FT6 Underlying resource and application runtime adaptation • FT7 Fog-optimized distributed data processing • FT8 Reactive routing • FT9 Adaptive monitoring • FT11 Fog node "smart" storage

Test Case ID	MSP-TC-04
Test Case Title	A next drone is being prepared for the flight and starts the next mission
Actors	Drone operators
Pre-conditions	• RAINBOW system running • GCS's nodes are deployed • Master mission designed • two drones deployed, assigned to the GCS's, with automatically designed and delivered route plans • the 3d drone is deployed and waits for the route
Post-conditions	• Successfully designed and assigned route for the drone, based on tracked parameters and photo control results
Normal Flow	1. Two drones take off and one of them reports several incorrectly acquired images during the flight 2. Mission Guidance service designs a route for the third drone, based on the current mission status (one mission completed, the second interrupted) 3. Supplementary mission is assigned to the third drone that starts the mission 4. All three drones land after their flights
Pass Metrics	• real-time monitoring of data quality with corrective actions applied during the mission – achieved • automatic assignment of flight routes to drones - achieved • optimization of flight paths based on tracked parameters: battery status, overlaps minimization, photo control - achieved

Fail Metrics	• real-time monitoring of data quality with corrective actions applied during the mission – not achieved • automatic assignment of flight routes to drones - not achieved • optimization of flight paths based on tracked parameters: battery status, overlaps minimization, photo control – not achieved
Notes and Issues	User stories confirmation: • a drone executes a task that has been assigned to it • the drone finishes its flight and lands near an operator. • Mission Guidance service executes the master mission
Major RAINBOW Use Cases Tackled	• FT1 Constraint and policy editor • FT2 Containerized application packaging • FT4 Application deployment over fog realms • FT5 Application lifecycle management • FT6 Underlying resource and application runtime adaptation • FT7 Fog-optimized distributed data processing • FT8 Reactive routing • FT9 Adaptive monitoring • FT11 Fog node “smart” storage

Test Case ID	MSP-TC-05
Test Case Title	Carry out full missions over a test distance with RAINBOW support to measure metrics KPIs
Actors	Drone operators
Pre-conditions	• designated 3 km of the test distance of linear object • RAINBOW services running • 3 drones ready to fly • 3 GCS’s deployed in random order along the test objective
Post-conditions	• Full mission is performed with RAINBOW support • KPI values are measured

Normal Flow	1. Master Mission is planned 2. Two drones are deployed, individual routes are designed automatically and assigned to the drones and they start for their flights 3. After any of the drones lands, the route for the 3d drone is designed, assigned and it starts for its flight 4. For all drones that land – batteries are replaced, new routes are designed and assigned automatically and drones start for their new flights 5. At least one GCS is relocated during the test case, to cover the full test distance with the radio link 6. When the full test distance is covered, KPI values are measured and compared with the reference values
Pass Metrics	• increased productive flight distance per drone – min. 150% • reduced data acquisition time per kilometer of power line – 50% in relation to current being 100% • reduced overlap between individual flight routes – (-)75% in relation to current being 0%
Fail Metrics	• increased productive flight distance per drone – below 115% • reduced data acquisition time per kilometer of power line – 80% or more in relation to current being 100% • reduced overlap between individual flight routes – (-)40% or less in relation to current being 0%
Notes and Issues	• The reference values for KPIs estimations are acquired by performing several reference missions over the test distance carried out in a traditional way (1 operator = 1 drone = 1 flight) • test case can be repeated several times if necessary
Major RAINBOW Use Cases Tackled	

4.4.3 Implementation Plan

The implementation plan for the third demonstrator; test scenarios which shall be performed during the “early stage” (i.e., between M18 and M21) and test scenarios which shall be left for the “advanced stage” (i.e. between M30 and M33).

Furthermore, the test cases linked to each identified test scenario, in an effort to further prescribe the testing effort in year 2 and year 3 of the project (*further refinement per demonstrator expected in deliverables D6.6 and D6.7*).

Table 4-9 Demonstrator #3 Implementation Plan

Related Scenario	Test Cases	Due by
MSP-TS-1	MSP-TC-01 MSP-TC-02	Early (M21)
MSP-TS-2	MSP-TC-03 MSP-TC-04	Early (M21)
MSP-TS-3	MSP-TC-01 MSP-TC-02	Advanced (M33)
MSP-TS-4	MSP-TC-03 MSP-TC-04 MSP-TC-05	Advanced (M33)

5 Conclusions

The objective of the present deliverable was to document the work in the context of Task 6.1; the planning and coordination of the three demonstrators of the project towards the successful execution of the evaluation at their environments, after integrating with the RAINBOW solution.

The result is an evaluation framework which covers both the technical evaluation and the business evaluation, based on test cases, combined in test scenarios, and measured with the help of KPIs. The technical evaluation was built based on ISO/IEC 25010:2011 "Product Quality" model and the business evaluation was built based on ISO/IEC 25010:2011 "Quality in Use" model. Quantitative evaluation of the platform will be performed based on the list of KPIs and the demonstrator specific KPIs, while a qualitative aspect shall be also provided through the provision of the appropriate questionnaires and demonstrator specific questionnaires, which shall be defined in the upcoming deliverables of WP6.

For each demonstrator a comprehensive description of the pilot that will be executed is presented. First, the virtual demonstrator is described. A virtual demonstrator is a set-up which consists mainly of or depends greatly on lab infrastructure and usually can offer more testing capabilities than a physical demonstrator (i.e. simulating unlimited number of instances for a certain scenario). Most of the "early" demonstrations will take place using the virtual set-up across all three demonstrators. Then, the physical demonstrator is described. A physical demonstrator consists of actual machinery, infrastructure and real network links and infrastructure and could be usually limited by objective conditions (e.g. a physical set-up for the manufacturing use case can include only one robot). The "advanced" demonstrations shall be executed using a physical set-up. Finally, the integration of each demonstrator with the RAIBOW solution is explained.

In the final section, a list of test scenarios which will be executed is presented for each demonstrator, along with the corresponding test cases for each scenario. Within the context of each test case the required information for the proper execution is documented indicating the actors involved, the required pre-conditions, the execution flow of the test case, the expected post-conditions and finally the metrics for the evaluation of the test case. Following this, the implementation plan of each demonstrator is laid down, documenting the test scenarios and test cases which are expected to be executed while in the "early" demonstration phase and those test scenarios and test cases which shall be left for the "advanced" demonstration" stage. It is not an exclusive division, meaning one test case could be executed in the context of more than one test scenarios, while at the same time one test scenario could be necessary for both the "early" and the "advanced" demonstration.

The reader should consider the whole process of demonstrator evaluation and feedback as a living process, which shall be further fleshed out at the level of each demonstrator in the upcoming deliverables of WP6, namely D6.2 "Human-Robot Collaboration Demonstrator – Early Release", D6.4 "Digital Transformation of Urban Mobility Demonstrator – Early Release" and D6.6 "Power Line Surveillance Demonstrator – Early Release", all due on M21 of the project.

References

[1] RAINBOW Consortium (2020) Deliverable D1.1 "RAINBOW Stakeholders Requirements Analysis"
[2] RAINBOW Consortium (2020) Description of Action
[3] RAINBOW Consortium (2020) Deliverable D1.2 "RAINBOW Reference Architecture"
[4] RAINBOW Consortium (2020) Deliverable D1.3 "Use-Cases Descriptions"
[5] Venkatesh, V. & Davis, F. (2000). A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies. Vol 46(2), pp. Management Science 200046:2, 186-204
[6] Hedonic-motivation system adoption model (HMSAM) https://is.theorizeit.org/wiki/Hedonic-motivation_system_adoption_model_(HMSAM)
[7] User Experience Questionnaire. https://www.ueq-online.org
[8] European Telecommunication Standards Institute, ETSI Technical Specification, ETSI TS 101 539-1: Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification, 2013
[9] Cellular-Vehicle-to-Everything: today and next steps. Maxime Flament, CTO of 5GAA
[10] European Telecommunication Standards Institute, ETSI Technical Specification, ETSI TS 102 637-1: Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 1: Functional Requirements, 2010
[11] European Telecommunication Standards Institute, ETSI European Standard, ETSI EN 102 637-2: Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Co-operative Awareness Basic Service, 2014
[12] European Telecommunication Standards Institute, ETSI European Standard, ETSI EN 302 637-3: Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service, 2014
[13] European Telecommunication Standards Institute, ETSI Technical Specification, ETSI TS 102 894-2: Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary, 2014
[14] European Telecommunication Standards Institute, ETSI Technical Report, ETSI TR 103 300-1 V2.1.1: Intelligent Transport System (ITS); Vulnerable Road Users (VRU) awareness; Part 1: Use Cases definition; Release 2, 2019
[15] 3GPP TR 21.914 Release description; Release 14