5G!DRONES

"5G for Drone-based Vertical Applications"

D4.2 – Integration status and updated integration plan

Document ID:	**D4.2**
Deliverable Title:	**Integration status and updated integration plan**
Responsible Beneficiary:	**DRR**

Topic:		**H2020-ICT-2018-2020/H2020-ICT-2018-3**
Project Title:	**Unmanned Aerial Vehicle Vertical Applications' Trials Leveraging Advanced 5G Facilities**	
Project Number:		**857031**
Project Acronym:		**5G!Drones**
Project Start Date:		**June 1st, 2019**
Project Duration:		**42 Months**
Contractual Delivery Date:		**M26**
Actual Delivery Date:		**July 30th, 2021**
Dissemination Level:		**Public (PU)**
Contributing Beneficiaries:		**AIR, ALE, AU, CAF, COS, DEM, DRR, EUR, FRQ, INF, INV, MOE, NOK, OPL, ORA, RBX, THA, UMS, UO**

DocumentID:	D4.2
Version:	V1.0
VersionDate:	27.07.2021
Authors:	Paweł Korzec (DRR), Piotr Dybiec (DRR), Wojciech Woźniak (DRR), Tanel Järvet (CAF), Maido Hiiemaa (CAF), Shvea Järvet (CAF), Oussama Bekkouche (AU), Hamed Hellaoui (AU), Tarik Taleb (AU), Antti Pauanne (OU), Jean Cerrillo Fernandez (OU), Syed Ali (OU), Juha Röning (OU), Joonas Pulkkinen (NOK), Riikka Valkama (NOK), Jyrkkä Johannes (NOK), Ilkka Känsälä (NOK), Karim Boutiba (EUR), Mohamed Mekki (EUR), Adlen Ksentini (EUR), François Despaux (ALE), Louis Viard (ALE), Gokul Srinivasan (RBX), Serge Delmas (AIR), Paweł Montowtt (INV), Gregor Mogeritsch (FRQ), Yvon Gourhant (ORA), Lechosław Tomaszewski (OPL), Pascal Bisson (THA), Ehsan Ebrahimi Khaleghi (THA), Farid Benbadis (THA), Harilaos G. Koumaras (NCSRD), George Makropoulos (NCSRD), Stavros Kolometsos (NCSRD), Anastasios Gogos (NCSRD), Pantelis Sarantos (NCSRD), Nemish Mehta (UMS), Pradyumna Vyshnav (UMS), Kim Clement (UMS), Vaios Koumaras (INF), Christos Sakkas (INF), Aggeliki Papaioannou (INF), Fotini Setaki (COS), Dimitris Tzempelikos (MOE), Ott Tahk (HEP)
Security:	Public (PU)

Approvals

	Name	Organization	Date
Coordinator	Jussi Haapola	UO	30.07.2021
Technical Committee	Farid Benbadis	THA	26.07.2021
Management Committee	Project Management Team and CAF	UO, THA, AU, AIR, UMS, FRQ, CAF	26.07.2021

Document History

Version	Contribution	Authors	Date
V0.1	Initial TOC	Piotr Dybiec (DRR)	22.06.2020
V0.2	Updated TOC/structure based on 4 releases	Piotr Dybiec (DRR)	30.11.2020
V0.3	First content	Piotr Dybiec (DRR)	28.02.2021
V0.4	New contributions	All	30.06.2021
V0.6	First content complete version	All	12.07.2021
V0.9	Version ready for review	Piotr Dybiec (DRR)	13.07.2021
V0.9b	Version reviewed by TM and internal reviewers	Farid Benbadis (THA), Pradyumna Vyshnav (UMS), Tanel Järvet (CAF), Gregor Mogeritsch (FRQ)	26.07.2021
V1.0	Final version	Piotr Dybiec (DRR)	27.07.2021

Executive Summary

This deliverable provides a status update on the integration procedures and a refined integration plan. It is based on D4.1 [1]. Concepts outlined in D4.1 document are developed in this deliverable.

The document is divided into four main chapters.

In the first "Introduction" chapter objectives of the document, target audience and relation to other documents are explained.

The most important conclusion is that WP4 consumes and consolidates products delivered by WP1 (overall architecture concept), WP2 (trial controller system) and WP3 (enablers for 5G facilities and U-space). Chapter 1 describes the dependencies.

The second chapter, "Updated Approach To Planning, Testing and Integration of 5G!Drones", contains detailed and updated description of release concept and plans, the testing approach with details of tools and methodologies, detailed descriptions of all integrations to be implemented as well as target deployment environments.

This is the only document, which provides detailed integration architecture schemas and information for all use cases, involved facilities, and used enablers.

The third chapter, "Updated Integration Plan", is dedicated to the description of integration planning. First part contains delivery plans of all enablers developed during the project. The second part provides detailed descriptions of all 4 releases: scopes, schedules, outcomes, etc.

It contains detailed development plans for all enablers through the release plan perspective, showing when enablers, with which functionalities, will be available for testing and integration.

Further sections of this chapter describe in details scope and plans for 4 major releases:
- The Integration validation release planned for period M22-M27 aimed on gathering and validating all the integration requirements of all system's components, as well as proofing of testing approaches and methodologies.
- The Trial Controller release planned for period M28-M31 aimed on integration testing of internal components of Trial Controller as well as end-to-end main process flows.
- The KPI release planned for period M32-M35 aimed on validating the tools and methodologies to be used for the purpose of KPI measurements gathering and processing.
- The Use case release planned for period M36-M37 aimed on final validation of all use case scenarios.

The last chapter, "Conclusions", summarizes the most important issues of task T4.1.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

3GPP   3rd Generation Partnership Project

5G   5th Generation Cellular Technology

ADS-B   Automatic Dependent Surveillance – Broadcast

API   Application Programming Interface

AR   Augmented Reality

BVLoS   Beyond Visual Line of Sight

CAA   Civil Aviation Authority

CC   Creative Commons

CONOPS   Concept of Operations

CoTS   Commercial Off-The-Shelf

DMP   Data Management Plan

EAB   External Advisory Board

eMBB   Enhanced Mobile Broadband

EPC   Evolved Packet Core

ETSI   European Telecommunications Standards Institute

FAIR	Findable, Accessible, Interoperable, Re-usable
FCT	Facility Coordination Team
FR	Financial Report
GA	General Assembly
GDPR	General Data Protection Regulation
GNSS	Global Navigation Satellite System
GOF	Gulf Of Finland
GPS	Global Positioning System
GUI	Graphical User Interface
ICAO	International Civil Aviation Organization
IMT	Innovation Management Team
IoT	Internet of Things
IPR	Intellectual Property Rights
IR	Internal Report
JSON	JavaScript Object Notation
KPI	Key Performance Indicator
LiDAR	Light Detection and Ranging
LADN	Local Area Data Network
LTE	Long-Term Evolution
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
mMTC	Massive Machine-Type Communications.
MoM	Minutes of Meeting
MS	Microsoft
PC	Project Coordinator
PCI	Physical Cell Id
PIA	Privacy Impact Assessment
PID	Persistent Identifier
PMT	Project Management Team
PSI	Public Service Identity
QMR	Quarterly Management Report

RAN	Radio Access Network
RRC	Radio Resource Control
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RSSI	Received Signal Strength Indicator
SNR	Signal to Noise Ratio
SINR	Signal-to-Interference-plus-Noise Ratio
SIM	Subscriber Identification Module
SORA	Specific Operations Risk Assessment
SSH	Secure Shell
SWIM	System Wide Information Management
TM	Technical Manager
ToC	Table of Contents
UAS	Unmanned Aerial Systems
UAV	Unmanned Aerial Vehicle
UE	User Equipment
uRLLC	Ultra-Reliable Low Latency Communications
UTM	UAS Traffic Management
VNF	Virtualized Network Function
VPN	Virtual Private Network
VR	Virtual Reality
WP	Work Package
WPL	Work Package Leader

1. INTRODUCTION

1.1. Objective of the document

The main purpose of this document is to provide the updated information regarding actions and steps with necessary resources needed to allow solution integration activities as required by 5G!Drones' task T4.1. The initial integration plan, defined within the D4.1 [1], is hereby refined and the integration procedures are updated. According to the plan, the integration and testing procedures as well as testing environment (including development and testing methodologies, tools, interfaces, and validation criteria) are defined, which will orchestrate the progressive delivery of software and/or hardware modules by WP2 and WP3 to incrementally deploy the 5G!Drones system and test it within the trial facilities.

The T4.1 responsibilities include the following activities:

- Coordination of deployment and individual component testing on the 5G facilities;

- Functional tests for the validation of the 5G!Drones architecture;

- Integration and testing of the UAV hardware in the target ICT-17 facilities and other supporting 5G facilities;

- Functional tests of the selected scenarios over the selected facilities.

The update on detailed planning, tools, and methodologies to fulfil these activities, is in scope of this document.

1.2. Relation to other project work

WP4 is a consumer of products delivered by the work packages WP1 (requirements, the overall architecture and workflows, and regulatory dependencies), WP2 (5G!Drones trial controller and cross-domain mechanisms, e.g., U-space adapter), and WP3 (enablers for network, MEC, aviation layer, and testbed specificity-abstracted mediation). The final integration is dependent on deliverables of these work packages, namely D1.1, D1.5 (as a successor of D1.2), D1.6 (as a successor of D1.3), D2.4 (as a successor of D2.1), D2.3, D3.1, and D3.2. In case of the updated documents and new ones (D2.3 and D3.2), a continuous information exchange was necessary to provide the mutual compatibility of the deliverables on which the parallel works were underway. While the detailed planning of the development of the WP2 and WP3 products is out of the scope of the Integration Plan, the timely delivery of the products according to the plan is the responsibility of these work packages. On the other side, the role of T4.1 was to provide continuous feedback to architecture and solution design streams within WP2 and WP3 to secure the end-to-end feasibility of the global 5G!Drones system.

1.3. Structure of the document

This document consists of four major chapters:

Chapter 1 hereby presents the introduction to the document, its objectives and structure.

Chapter 2 contains the updated description of 5G!Drones approach to planning, testing and integration, including the tests master plan, different level tests' guidelines, best practices and test design templates as well as general and test facility-specific integration architecture and interface issues.

Chapter 3 focuses on the description of delivery plans for the Trial Controller modules, test facilities' capabilities, and use case enablers as well as on the scope of the successive releases.

Finally, Chapter 4 concludes the document.

1.4. Target Audience

This deliverable is a public document and is mainly addressed to:

- The project consortium to establish a common understanding of the integration methodology, steps, and their scope in order to be timely prepared for the necessary actions.

- The research community, projects, and ICT professionals to present a concrete plan towards the preparation of the 5G!Drones platforms for subsequent execution of the use cases for the interest of other collaborations. The integration plan may also serve for better understanding of the final trials' results, from the point of view of the underlying environment used for the end-to-end trials.

- **The Industry**: All clusters that acquire an industrial professional background with high technical knowledge and specialization in the field, which may be working in relevant areas of 5G and drones such as:

  - o Technology Producers, Suppliers, Carriers, Vendors,

  - o SME's (e.g., application developers and third-party providers of 5G related services).

- **Academia and Research Institutions**: Institutions (consultancies) from the ICT sector, higher education institutions (universities and educational centres, including those for lifelong learning, i.e., vocational training, re-training of experts in new technologies and multimedia services and applications), as well as national, public, and private research institutes.

- **Public and Private Service Providers**: Current providers of 4G/5Gservices, network operators, verticals, and other relevant institutions that will upgrade to a 5G network system soon i.e., along with drone companies/providers (e.g., ANSPs).

- **The broadest possible technical audience:** This category covers the potential providers and users of 5G and drones' products and services as well as the general technical audience and scientific community which might be interested in these technological fields and advancements.

- **All 5G!Drones partners, collaborators, and stakeholders:** This document is addressed to the entire 5G!Drones consortium and serves as documentation of the integration status and updated integration plan to be applied for efficiently performing all planned use cases scenarios and trials.

- The funding EC organization, as it is committed by the Grant Agreement**.**

2. UPDATED APPROACH TO PLANNING, TESTING, AND INTEGRATION OF 5G!DRONES

In following sections, the updated approach to planning testing and integration is presented. The planning approach update provides more implementation details of release concept outlined in D4.1 document. Testing approach section focuses on methodologies and tools used to validate the integration of the system. The integration section describes in detail the interfaces, integration, and deployment architectures of the 5G!Drones system.

2.1. Release planning approach

There are in total 4 integration releases planned, which will assure that the Trial Controller system is ready to fulfil 5G!Drones requirements:
- Release 1: Integration validation release
- Release 2: Trial Controller release
- Release 3: KPI release
- Release 4: Use case release

The number of the releases is driven by the performance and capabilities of the consortium.

Focus of the first release, the Integration validation release, is to gather all the integration related requirements, validate testing methodology, perform selected API tests on available interfaces. Second release, the Trial Controller release, is supposed to be dedicated to tests of the Trial Controller. Third release, the KPI release, aims on validation and testing of KPI tools and methodologies. Final, the Use case release, is the release aimed on validating all end-to-end use case scenarios. More information about scope and content of each release is given in Section 3.2.

As the system is composed of more than 100 different components, the necessary tests will be performed according to the availability dates – so on top of the main topic of the release, additional tests of provided components/enablers will be performed.

Summary of the releases' scope is provided in Table 1.

Table 1 Major releases schedule and scope

	Integration validation release (Rel 1)	Trial Controller release (Rel 2)	KPI release (Rel 3)	Use case release (Rel 4)
Release date	M22-M27 (March-August 2021)	M28-M31 (September-December 2021)	M32-M35 (January-April 2022)	M36-M37 (May-June 2022)
Main goals	(1) To ensure that all Trial Controller components in scope are interconnected and required infrastructure is provided (2) To ensure, that all 3rd party external enablers' integration requirements are captured, fulfilled and validated (3) To propose/validate deployment architecture (4) To propose/validate (integration) testing tools/processes	(1) To perform end-to-end tests of key Trial Controller processes based on the assumed to be available fully functional releases of all Trial Controller components.	(1) To perform end-to-end tests (including facilities and UAVs) of all KPIs with designed KPI tools (e.g. dedicated scripts) and procesess to validate them and expected results.	Final release confirming full ability to perform trials/execute defined use case scenarios (1) To ensure all service level enablers are developed (2) To ensure all use cases are defined in their final form (3) To ensure all use case components are tested

Activities planned within each of those releases can be of 2 main types:
- Tests, which would require end to end testing environment availability including selected trial controller components, additional applications (enablers), facilities, UAVs, etc.
- Tests, which could be executed in simulated or limited capability environment in terms of required availability of UAVs able to flight, access to facilities, etc.

End to end tests, due to the logistic effort as well as limited resource and suitable time slots, will be synchronized with planned pre-trials, trials, and showcases, which is described in detail in D4.3 "Trials plan" document [2]. The other tests, which do not require end to end availability, will be planned to be executed in other suitable time.

2.2. Testing approach

Tests are the most important activities during system's validation process.

During the integration phase we focus on tests, which will provide the proof of interoperability between different components of the solution. It means, that we will not prepare nor perform other types of tests than those, which are directly related to integration issues.

There are many dimensions, how tests can be classified. One of the most basic distinction is diving tests activities into 2 generic categories:

- Functional tests

- Non-functional tests

In principle, functional tests cover tests that are aimed on validation, if defined business requirements are met. Sometimes, they are called "black-box tests," because they do not focus on the product's internals, but on the functional aspects of the interface layer. They provide the answer for the question "**What** the product does?" On the other hand, non-functional tests answer the question "**How** the product works?"

Examples of typical functional type of tests include unit tests, acceptance tests, smoke tests, integration tests, regression tests. To non-functional category of tests belong tests such as: performance tests, scalability tests, volume tests, load tests, stress tests, etc.

2.2.1. Test types in 5G!Drones project

During the course of the 5G!Drones project **only functional tests will be performed**. The non-functional tests are out of the scope of the project. From mentioned above different kinds of functional tests, following tests are planned to be performed:

- **Unit tests** – in case of dedicated development is performed; these tests are within sole responsibility of the development teams and are not covered/monitored in this document neither during the integration phase of the project.

- **Integration tests** – dedicated tests of interfaces/APIs; main validating activity during integration phase.

- **Acceptance tests** – functional, "end-to-end" tests; these tests will become part of the use case demonstrations activities described in D4.3.

As mentioned above, this document describes only testing approach specific to integration type of tests.

2.2.2. Integration Tests' documentation

For documenting tests, two main types of documents will be prepared:

- **Tests specifications**

- **Tests reports**

Tests specifications contain set of tests to be performed.

Test reports contain reports from the tests execution: list of performed tests and their results with eventual remarks and suggestions for improvements/corrections.

As part of the integration testing process, 5G!Drones introduces a Test Cases Descriptor template (see Table 2), to illustrate the critical parameters of each integration test case that is defined in order to

validate a specific component integration, or chain of components, against the functional and measurable objectives set per case.

Table 2 Integration test case descriptor template

Test case name here			
Test case Id	*Add Id number.*		
Test defined by	*Add name of the person that defined the test(s).*		
Test executed by	*Name of the person(s) executing the test(s).*		
Testing date	*When were the tests run?*		
Test purpose	*Provide a description of the purpose of the tests.*		
Test environment	*Describe the setup used for the integration tests. Can be a target 5G platform, a sandbox environment, or a specific simulation environment.*		
References	*Insert reference on underlying standards (e.g., 3GPP TR 22.829) and/or project's specification and reference points if any.*		
Validation target/metric	*Define the quantitative or qualitative result(s) expected from the execution of the test case.*		
Tested components	*List the HW and SW components under testing with version info.*		
	HW Components	*SW Components*	
Tested interfaces	*List the interface(s) and or API(s) that are tested.*		
Other interfaces	*List the interfaces that must be established to execute the test case.*		
Pre-test conditions	*List the pre-requisites that are necessary before the test execution, for example other test-cases that must be first completed. Note that this refers to the test case execution business and no other installation prerequisites that should be part of the deployment descriptors of each component/package.*		
Test tools	*Identify the test tool(s) used for the execution of the test.*		
Test repetitions	*<num>*	*List the number of iterations that are necessary to be run*	
Integration test steps Definition and results Log	**# Step**	**Description**	**Result**
	1	*List the sequence of integration actions to be performed. List pass criteria.* *List results for each action.Inform where measurement log can be found.*	DONE
	2		SKIP
	…		
Test verdict	PASS/ FAIL/ PENDIN G	*Describe the status, findings and extensive feedback provided to the next integration cycles. The status of the execution can be:* • *PASS: Results are as expected, and no further actions are necessary* • *FAIL: The targets set for the execution cannot be met due to functional problems that need new/corrected implementations expected in later integration cycles* • *PENDING: The tests cannot be executed due to missing environmental characteristics or other external constraints (ex. Availability of 5G SIM (Subscriber Identification Module) cards)*	
Change requests		*List here all change requests to the module if any.*	
Defects		*List here all the possible errors reported to the developer. Remember to include id.*	

Each such a test case will be named with appropriate, unique test id, e.g., WEB-IT-1. Test case descriptor describes test purpose, prerequisites, test scenario, results, evaluation, and conclusions.

A set of test cases described according to above specification, defined by the scope of the specific release, defines the Test plan document. For integration testing such a document, "Integration Test plan" [3] was prepared for Release 1. Based on it also for further releases test plans would be prepared.

After tests execution, the corresponding test report is generated. It contains the list of performed tests with their results. Example test report is provided in Section 3.2.1.6.1.

2.2.3. Integration testing guidelines

The main purpose of the integration testing is to verify the interoperability between different, already tested in isolation, components of the system.

It might require good knowledge and understanding of the system architecture (white-box testing), when legacy components are tested.

When public APIs (Application Programming Interfaces) are tested, they can be performed with limited knowledge about the system's internal architecture (grey-box testing). The goal of the project is to design and build all the project's components with well documented APIs.

The output of the integration testing, apart from the Pass/Fail result is expected to provide quantitative and qualitative feedback to the subsequent integration cycles.

As opposed to unit tests, the integration tests focus on communication and interoperability aspects of the system. During integration tests, external systems/modules/application are tested extensively.

As the interfaces to be tested during this phase were deployed using RESTful APIs, the main tools supporting test execution were Postman (https://www.postman.com) and cURL (https://curl.se).

2.3. Integration approach

The 5G!Drones system has been designed as a system composed of separate components interconnected via defined APIs. Thus, the main goal of the integration activities is to set up connections between those components and to test one by one all functions of the APIs. The system will be deployed on shared infrastructure as described in section 2.3.3.

In current IT architectures typically two main techniques are used for systems' integration:

- REST (REpresentational State Transfer) architectural pattern, especially web service based RestFul pattern,
- SOAP (Simple Object Access Protocol) protocol

SOAP provides stateful mean of communication with asynchronous processing and subsequent invocation. It has higher level of reliability and security but requires more bandwidth because entirely it relies on XML files format.

On the other hand, REST permits many different data formats such as: Plain text, HTML, XML, JSON, etc. It requires limited resources and bandwidth. It is statelessness type of communication – it can be used mainly by the applications, which have no need to maintain a state of the information from one request to another.

For the 5G!Drones project, it was agreed between partners to use web services based RESTful approach, whenever it is feasible.

Moreover, the integration interfaces between components, to allow standardized testing, design, and documentation, were implemented according to the OpenAPI specification using Swagger toolset (https://swagger.io). Thanks to this approach each 5G!Drones module has its own well documented API.

2.3.1. General Integration architecture

The diagram in Figure 1 describes the general integration architecture – the connections between all components of the end-to-end implementation of 5G!Drones trial system. Each identified component represents separate, "standalone" module: software package, API, hardware/equipment, etc. Additional graphic symbols assigned to each of the component provide the information what type of the module it is: software package (usually in the form of deployable container), legacy software, hardware equipment, external API, etc.

All components have been grouped by following categories:

- Trial Controller components/enablers marked on the diagram with blue colour
- Facility components/enablers marked on the diagram with orange colour
- UAS/U-space components/enablers marked on the diagram with green colour
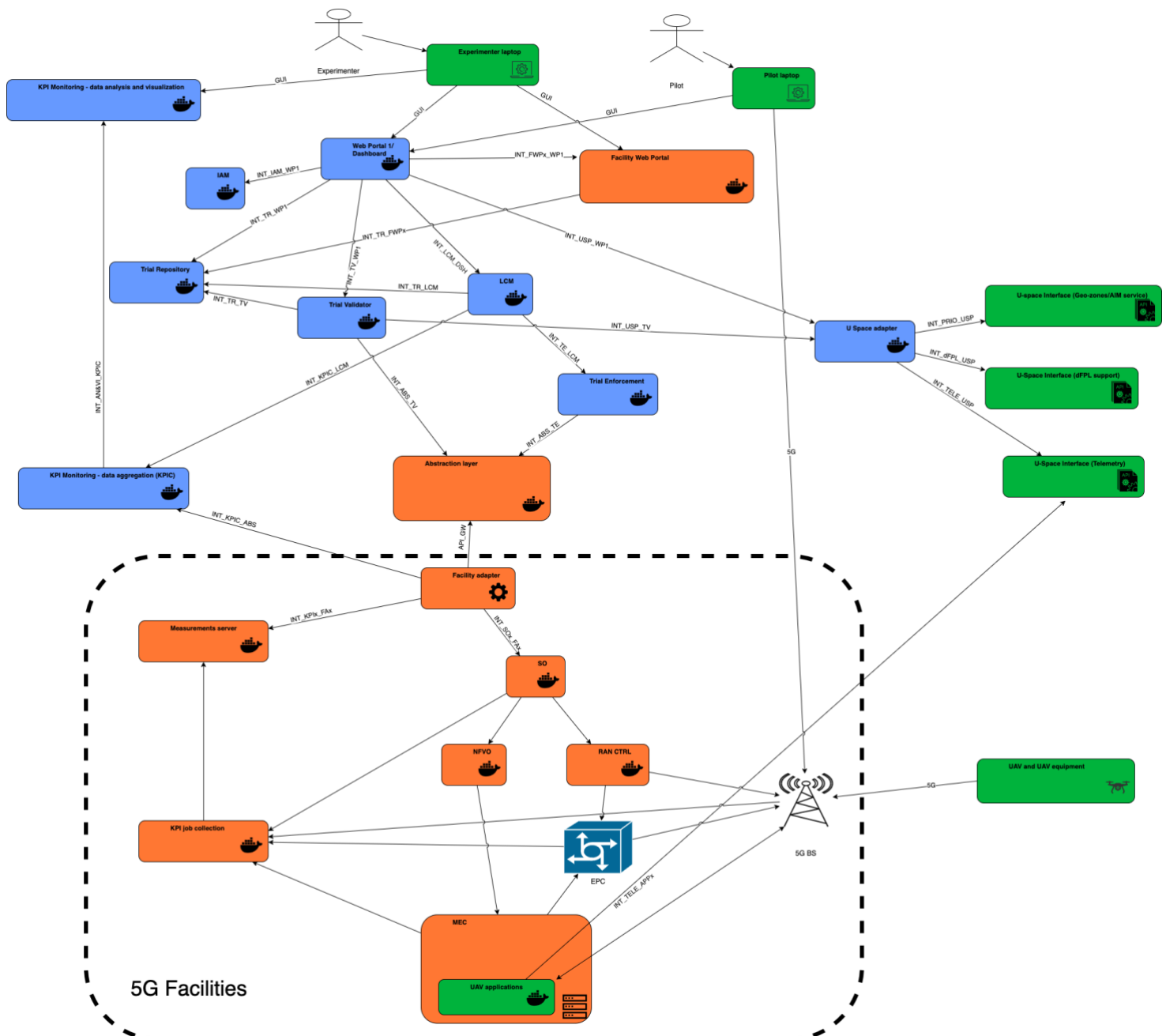
Figure 1 5G!Drones' components integration architecture

Each connection shown on the diagram represents the interface implemented between interconnected components. Interfaces are described in dedicated tables. Each row in the table represents a single interface. It is identified by an unique Interface id and is described with following information:

- Interconnected endpoints identified by component exposing API (API server) and component using API (API client)

- Type of the API – in majority cases it is, as mentioned earlier, RESTful API

- API description and its purpose

The information about Trial Controller internal interfaces is provided in **Error! Reference source not found.**. All mentioned interfaces are used to interconnect modules of the Trial Controller and in majority are implemented as typical web services (RESTful APIs). The Trial Controller is accessed by users (Experimenter, Pilot) through the web interface (GUI) via a web browser installed on dedicated laptops (Experimenter laptop, Pilot laptop). Web interfaces are exposed by Web Portal 1, KPI Monitoring, Trial Validator and Facility Web Portal. Users' laptops can be used also to provide access to other, use case specific, applications like backup GCS (emergency remote controller for UAVs), GUI for VNFs installed on edge servers, etc.

Table 3 List of Trial Controller's internal interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_TR_WP1	RESTful	Trial Repository	Web Portal 1/Dashboard	The Web Portal 1uses the trial repository to store and query information related to the operation flight plan	CRUD information related to the operational plan of the trials
INT_IAM_WP1	OAuth2.0	IAM	Web Portal1/ Dashboard	Keycloak application deployed for account management of users (https://app.swaggerhub.com/apis/INSPIRE-5Gplus/keycloak-admin_rest_api/1#free)	Provides authentication of users
INT_TV_WP1	Https redirection	Trial Validator	Web Portal1/ Dashboard	Web Portal 1 is redirecting to the Trial Validator, where it's possible to see the status of modules responsible for the Operational Flight Plan and ask for trial validation at the U-Space.	Asking for Operational Flight Plan validation and status verification.
INT_FWPx_WP1	Https redirection	Facility Web Portal	Web Portal1/ Dashboard	Web Portal 1 is redirecting to the Facility Web Portal of target facility to prepare NST specific to the trail requirements and facility capabilities	Preparation of NST
INT_USP_WP1	RESTful	U-space Adapter	Web Portal1/ Dashboard	Web Portal 1 receives from U-Space the updates regarding the drone telemetry, which can be used to update the status of on-going mission.	Telemetry presentation in Web Portal 1 for on-going mission.
INT_LCM_DSH	RESTful	LCM	Web Portal1/ Dashboard	The Life-Cycle Manager (LCM) starts and closes different services just before, during and after the trial. The LCM acts as a kind of state machine to handle the high-level execution of a trial lifecycle.	Starts engine components for trial based on pre-set trial time and takes care also trial component closing.
INT_TR_TV	RESTful	Trial Repository	Trial Validator	Trial Validator requests information about Trials from the Trial Repository to display and validate.	Trial validation and information
INT_USP_TV	RESTful	U-Space adapter	Trial Validator	The Trial Validator uses flight plan data received from Trial Repository against the U-Space adapter interface to validate the flight plan. This leads to the actual creation of the flight plan within the U-space components.	Validation of flight plan.
INT_ABS_TV	RESTful	Abstraction layer	Trial Validator	The Trial Validator uses trial dedicated facility configuration information to validate its availability on the facility side by requesting validation through abstraction layer	Validation of trial specific facility configuration
INT_TR_LCM	RESTful	Trial Repository	LCM	The LCM can retrieve information on the network services and KPIs of the trial to run	Retrieve trial information
INT_KPIC_LCM	RESTful	KPIC	LCM	The LCM can retrieve the current actual components from the KPIC that are reporting KPIs.	Retrieving KPI reporting components
INT_TE_LCM	RESTful	Trial Enforcement	LCM	TE allows to enforce network services and requests from measuring KPIs on the top of the target facilities	Enforce network services and PKI measurements
INT_ABS_TE	RESTful	Abstraction layer	Trial Enforcement	The abstraction layer provides common interfaces for exposing functionalities allowing to request network services and KPI measurements	Provide common interfaces for communication with the facilities
INT_KPIC_ABS	RESTful	KPIC	Facility Adapter	The KPI module exposes API for facilities for streaming of KPI data streams. Each facility connects to it via its own Facility Adapter.	Streaming KPI data from facility
INT_AN&VI_KPIC	RESTful	KPIC	KPI Analysis & Visualization	KPIC provides interface to enable reporting and analysis of KPIS.	KPI reporting and analysis

According to the design, Trial Controller has following external interfaces:

- Interface with U-space services
- Interface with 5G facilities

Interface with U-space services is implemented by U-space adapter component of the Trial Controller. U-space adapter connects directly to U-space systems like UTM through APIs exposed by those systems. On the diagramTable 4 there are three services mentioned:

- Interface for dFPL (accessed via INT_dFPL_USP), through which, at the validation stage, operational flight plan is submitted for validation
- Interface for telemetry (accessed via INT_TELE_USP), through which Trial Controller can get access to telemetry data from UTM systems (used for KPI measurements)
- Interface for the Geo-zones/AIM service enabler (accessed via INT_PRIO_USP), through which UTM system can send notification regarding geofenced areas and alerts/requests to adjust the slice configuration due to changes in airspace (e.g., emergency missions)

More details about mentioned interfaces and services are provided in the Table 4.

Table 4 Interfaces to U-space services used by the Trial Controller

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_PRIO_USP	RESTful	UTM (Geo-zones/AIM service)	U-space adapter	The main purpose of this enabler is to provide exchange and management (Drone AIM, Aeronautical Information Management) services of necessary situational awareness information to all operational nodes of U-space.	Provide information about geofenced airspaces (in pre-tactical and tactical phases)
INT_dFPL_USP	RESTful	UTM (dFPL support)	U-space adapter	The Drone Flight plan (dFPL) enabler provides the ability to collect and process information about all flight intents	The Drone Flight-plan enabler allows to submit the operational flight plan and receive its validation
INT_TELE_USP	RESTful	UTM (Telemetry)	U-space adapter	The Telemetry enabler receives surveillance data (position reports) from several data sources (radars, drone on-board position telemetry services, tracking services, etc.) and exposes this information to provide the information about the current situation in airspace	Generate and provide a common situational awareness in terms of current and up to date locations of all UAVs

Interfaces with different 5G facilities are abstracted by an Abstraction layer component. Thanks to this, the exact architecture of the facility is "hidden". Different facilities can be connected to this Abstraction layer through dedicated Facility adapter via API_GW interface. Such Facility adapter must support 2 types of the payload:

- Slice orchestration requests/operations and KPIs configuration (interfaces INT_KPIx_FAx and INT_SOx_FAx)
- Streaming of the KPI measurements data (INT_KPIC_ABS)

Separate connections must be provided to the UAV enablers. This category covers UAVs with their additional equipment (connected usually via the 5G radio network) as well as dedicated applications hosted on edge servers of the facility, which in turn would usually require the connection to UAV (via

the 5G core/access network) and often to external services/platforms (like UTM for telemetry transmission via INT_TELE_APPx interface). This is described in detail on a case-by-case basis in following sections dedicated to each of the facility/use case.

2.3.2. Facility specific integration architectures (use case based)

In following sections, integration architectures for each of the use case are presented. The Trial Controller components (marked in blue on the diagrams) as well as U-space services (marked in green) remains the same for all use cases and were described in previous section.

The use case specific integration's architecture information is grouped by facilities, which will host those use cases. Each use case section provides:

- brief description of the use case from the perspective of used components, their roles, and interactions,
- inventory table of interfaces to be deployed in the respective use case and
- overall, end-to-end integration diagram.

For each of the facility, the use case chapters are preceded by the section containing the table describing facility specific interfaces, which are common for all use cases based on this facility.

2.3.2.1. Aalto university facility (X-NETWORK)

Table 5 below lists all X-NETWORK facilities' interfaces to be deployed and integrated to fully support functionalities of 5G!Drones platform and specific use cases described in following subsections.

Table 5 X-NETWORK interfaces to be integrated with 5G!Drones platform

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_SOXNET_FAXNET	RESTful	Slice Orchestrator (X-Network)	Facility Adapter (X-Network)	Interfaces between the facility adapter and the SO	Adapt requests to the SO
INT_KPIXNET_FAXNET	RESTful	KPI measurements server (X-Network)	Facility Adapter (X-Network	Interfaces between facility adapter and KPI measurement	Adapt requests to the KPI measurement
INT_TR_FWPXNET	RESTful	Trial Repository	Facility Web Portal (X-Network)	Interface between the Facility web portal and the trial repository	To store the requested network services
INT_APIGW_FAXNET	RESTful	Abstraction layer	Facility Adapter (X-Network)	Interfaces between the abstraction layer the facility adapter	Route requests to the facility adapter

2.3.2.1.1. UC1:SC3

The purpose of this scenario is to demonstrate how UAVs thanks to 5G network capabilities can provide logistics solutions. The scenario is the delivery of a drug to a sick person with a drone. A sick person who cannot go to a pharmacy can receive his/her medicine through a personal delivery by subscribing through CAFA Field GIS C2 system for Drone Logistics. Then order will be forwarded to a logistics company and the drone operator creates a flight plan in UgCS C2 cloud-native application enhanced by CAFA Tech (CAFA CUP) The delivery is conducted by CAFA drone which is the drone built by CAFA Tech based on the PX4 Drone Vision Kit. When the drone reaches its destination, there are inaccuracies with GNSS (Global Navigation Satellite System) signals. Therefore, the operator of the drone logistics company (actually the CAFA Tech operator) takes over the control of the drone. The

operator uses the CAFA 4K video stream from the drone camera and uses the gamepad command function thorough CUP C2 system to conduct the landing of the drone to the Delivery Box. A gamepad will be added to the drone remote pilot laptop with 5G communication modem. Video transmission and drone remote control take place over a 5G network. When the drone lands on the Delivery Box the drone then releases paracetamol thanks to the drone´s electrical hook mechanism. During the flight virtual Air Traffic Control (ATC) operator monitors UTM system and CAF drone flight thanks to U-space integration via UgCS C2 cloud-native application enhanced by CAFA Tech.

Specific interfaces to be implemented are described in Table 6. The overall integration architecture is presented on Figure 2.

Table 6 Specific UC1:SC3 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC1:SC3					
INT_CUP_UGCS	C2 over 5G	CUP C2 client	CAFA UgCS container	Docker container of CAFA CUP	Sends C2 information to drone and receiving telemetry data from the drone.
INT_CUP_LAPTOP	C2 over 5G	CUP C2 client	CAFA CUP server	Client GUI for planning, monitoring, and controlling drone flights.	To create flight plan, monitor it, take over if needed and teleoperation using gamepad.
INT_TELE_UGCS	Telemetry	U-space Interface API (Telemetry)	CAFA UgCS container	FRQ telemetry plugin integrated to CAFA CUP.	Forwards telemetry data to FRQ system and receives alerts from U-space
INT_4K_VIDEO_ STREAM	RTSP video	Videoserver	CAFA pilot	4K Video streaming from drone to MEC/edge server and forwarding video feed for end users and drone pilot	To support drone control teleoperation i.e., remote control via 5G and thanks to near real time video from the drone
INT_CAFA_FIELD	C2 over 5G	CAF field laptop	CAFA pilot	CAFA Field C2 system is running in CAF server. Drone pilots see on the laptop (using web browser) Field map and new delivery order.	To show delivery order on the map. Then pilot can plan delivery flight.
INT_LATENCY_ MEASUREMENT	End-to-end latency	AU server	AU server	Latency measurement script is installed in drone onboard computer and AU edge server, and it is using timestamps.	Drone operator understand the latency of C2 data.

Figure 2 Integration architecture of UC1:SC3

2.3.2.1.2. **UC3:SC2**

The use case scenario UC3:SC2 aims to trial UAV-based IoT data collection, where drones equipped with IoT sensors are deployed to collect various measurements. UC3:SC2 is described in D1.1, while the associated service-level enablers are detailed in D3.2. Two service-level enablers are considered for this scenario which are: C2 service (UAV control platform) and IoT service (IoT as a service platform).

Specific interfaces to be implemented are described in Table 7. The overall integration architecture is presented on Figure 3.

Table 7 Specific UC3:SC2 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_XNET_C2	RESTful	C2 service	UAV control platform (client)	Interfaces between C2 client and the C2 service at the edge	Control UAV
INT_XNET_IoT	RESTful	IoT service	IoT as a service platform (client)	Interfaces between IoT client and the IoT service at the edge	Request and collect IoT measurements
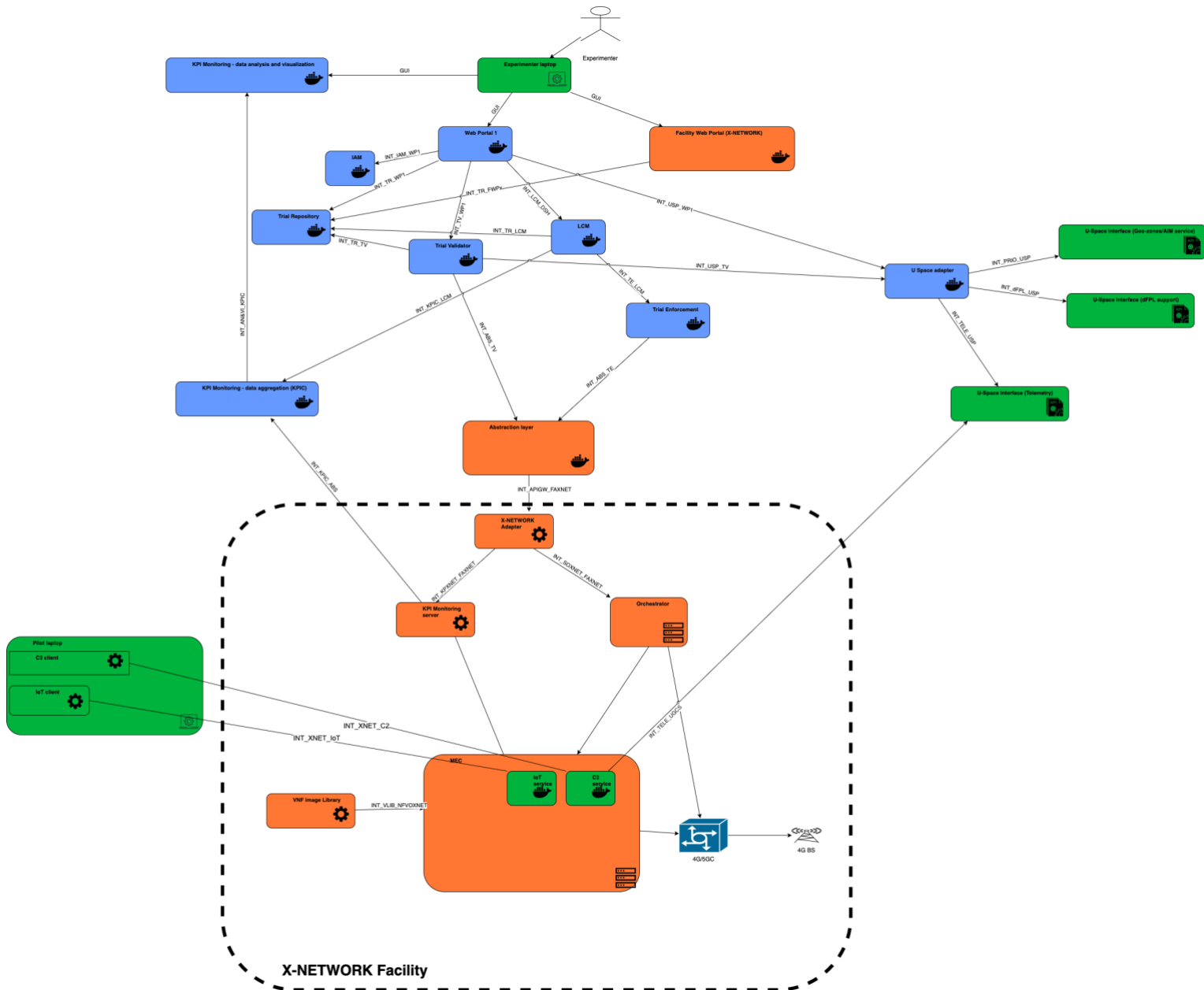
Figure 3 Integration architecture of UC3:SC2

2.3.2.2. University of Oulu facility (5GTN)

Table 8 lists all 5GTN facilities' interfaces to be deployed and integrated to fully support functionalities of 5G!Drones platform and specific use cases described in following subsections.

Table 8 5GTN interfaces to be integrated with 5G!Drones platform

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_SO5GTN_FA5GTN	Internal, implemented in Python	Slice Orchestrator (5GTN)	Facility Adapter (5GTN)	Creates, deletes, and gets information about slices	Managing slice life cycle
INT_KPI5GTN_FA5GTN	RESTful	KPI measurements server (5GTN)	Facility Adapter (5GTN)	Create measurement job. Get and push KPI measurement data.	Deliver measurement results to Trial Controller
INT_VLIB_NFVO5GTN	File	VNF image library/repository/hub	NFVO (5GTN)	Stores the VNFDs in OSM	Local application image repository
INT_TR_FWP5GTN	RESTful	Trial Repository	Facility Web Portal (5GTN)	Sends Trial Id and NST to Trial Repository	Initiate trial
INT_TR_TT5GTN	RESTful	Trial Repository	Trial Translator (5GTN)	Sends Trial Id and NST to Trial Repository	Initiate trial
INT_APIGW_FA5GTN	RESTful	Abstraction layer	Facility Adapter (5GTN)	Network slice and KPI management interface	Delivers needed Slice and KPI info to 5GTN facility
INT_QOSIUM	Qosium internal interface	Qosium probe	Qosium Scope UI	Manage Qosium network measurements	Monitor and collect Qosium network measurement results

2.3.2.2.1. UC1:SC2

The use case scenario of UC1:SC2 focuses to demonstrate, how to control drone from the Virtual Reality (VR) UI. In addition, used cable type of drone includes different sensors to update existing virtual reality space based on real-time changes in actual environment. The scenario was originally planned to be performed in an indoor OU test facility, so there is no reliable access to global navigation satellite system (GNSS). However, due to Covid-19 there has been limited possibility to instrument indoor infra the use case will be realized outdoors in the Science Garde of the University of Oulu. The original limitations will be followed. So, no GPS is assumed. Due to this limitation, during the first LoS (Line of Sight) control flight 5G UE and network related plus multiple other sensory information is collected. In the later flights the 5G UE location application (in local MEC computing unit) can analyse the UAV location and share that information to the UAV controller.

Specific interfaces to be implemented are described in Table 9. The overall integration architecture is presented on Figure 4.

Table 9 Specific UC1:SC2 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
US1SC2					
INT_5GTN_POS	RESTful	5GTN Position server	Positioning client UI	Provide positioning information to interested parties via secure connection	Positioning analysis application
INT_5GTN_VIR_FLI _CONT	RESTful	5GTN virtual flight controller application	Virtual Reality UI	Controls drone from Virtual Reality UI	virtual flight controller application
INT_5GTN_DATA_ TO_VIR	RESTful	5GTN Data collection and mapping for virtual flight controller	Virtual flight controller application	Data collection and mapping	Data collection and mapping for virtual flight controller

Figure 4 Integration diagram of UC1:SC2

2.3.2.2.2. US2SC3

This use case will demonstrate how remotely piloted UAV and video analytics can be used for police tasks, including C-UAS activities using 5G communication. Following scenario: The police is preparing for a VIP visit. The police also uses a drone that automatically flies and streams video to the video

analyzer software and the command center. The video analyzing software, CAFA Lyzer, installed on the MEC, uses videos and photos provided by the drone. As a part of the VIP visit, a temporary No Fly Zone (NFZ) and restricted ground area are established.

During the visit, the police drone flies on autonomous mode. The police drone streams continuously the video feed to the MEC server where at the same time, computer vision software CAFA Lyzer analyzes the stream. CAFA Lyzer detects a suspicious activity (person or drone on restricted area). The police operator then changes drone control from automated mode to teleoperation mode and takes over the drone control and flies near the suspicious person and affects offender to stop the illegal activity.
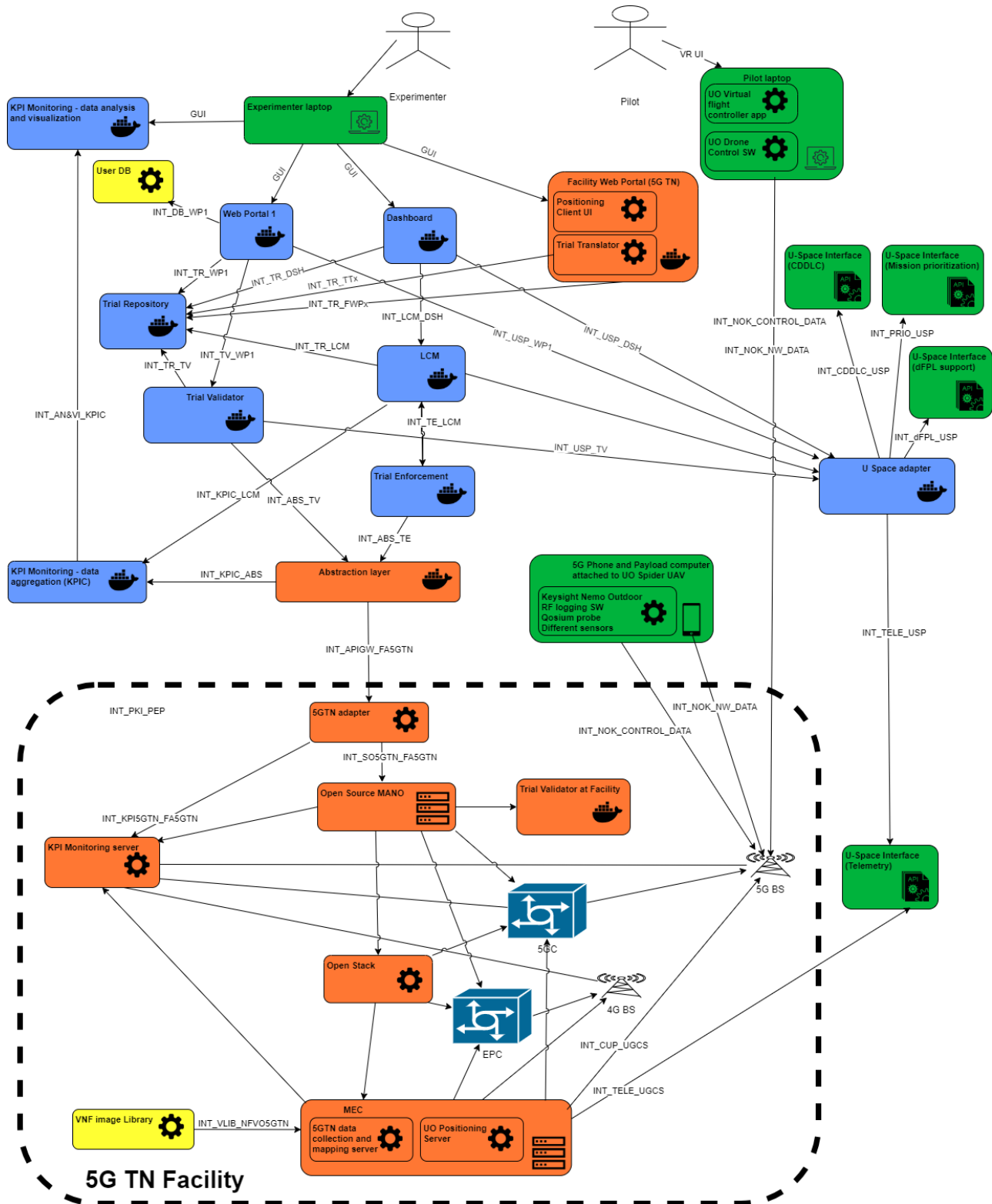
Specific interfaces to be implemented are described in Table 10. The overall integration architecture is presented on Figure 5.

Table 10 Specific UC2:SC3 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC2:SC3					
INT_CUP_UGCS	C2 over 5G	CUP C2 client	CAFA UgCS container	Docker container of CAFA CUP	Sends C2 information to drone and receiving telemetry data from the drone.
INT_CUP_LAPTOP	C2 over 5G	CUP C2 client	CAFA CUP server	Client GUI for planning, monitoring, and controlling drone flights.	To create flight plan, monitor it, take over if needed and teleoperation using gamepad.
INT_TELE_UGCS	Telemetry	U-space Interface API (Telemetry)	CAFA UgCS container	FRQ telemetry plugin integrated to CAFA CUP.	Forwards telemetry data to FRQ system and receives alerts from U-space
INT_4K_VIDEO_ STREAM	RTSP video	Video server	CAFA pilot	4K Video streaming from drone to MEC/edge server and forwarding video feed for end users and drone pilot (Police operator)	To support drone control teleoperation i.e., remote control via 5G and thanks to near real time video from the drone
INT_CAFA_FIELD	C2 over 5G	CAF field laptop	End user (Police)	CAFA Field C2 system is running in CAF server. Drone pilots see on the laptop (using web browser) Field map and incident location.	To show Police operational area and information on the map.
INT_LATENCY-MEASUREMENT	End-to-end latency over 5G	UO server	End user	Latency measurement script is installed in drone onboard computer and AU edge server, and it is using timestamps.	Measuring end-to-end latency drone operator understand the latency of C2 data.
INT_VIDEO-LYZER	Server application	UO server	End user (Police)	VideoLyzer is a Docker container running in UO edge server and analysing drones' videos and photos.	To analyse videos and photos from drone camera automatically.

Figure 5 Integration diagram of UC2:SC3

2.3.2.2.3.   UC3:SC1:sub-SC1

This scenario will demonstrate how 5G QoS mapping is done using 5G MEC based software for measuring 5G QoS. The communication company ordered from a drone company the 3D mapping of 5G QoS. At first, the drone operator takes 50-80 photos which are then processed to a point cloud. The point cloud is used for creating a 3D map. Then the 5G base station icon will be placed on the 3D map. 3D map processing is a manual process and takes place outside UO facility during the preparation phase.

The CAFA cellular drone then carries a 5G modem to measure the quality of 5G coverage from various positions with 3D coordinates (x, y, z) and timestamps. During the flight the operator and safety pilot monitor the automated flight. Measuring results are transferred to the server and then results are visualised on the CAFA 3D Analyzer.

Specific interfaces to be implemented are described in Table 11. The overall integration architecture is presented on Figure 6.

Table 11 Specific UC3:SC1:sub-SC1 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC3:SC1:sub-SC1					
INT_CUP_UGCS	C2 over 5G	CUP C2 client	CAFA UgCS container	Docker container of CAFA CUP	Sends C2 information to drone and receiving telemetry data from the drone.
INT_CUP_LAPTOP	C2 over 5G	CUP C2 client	CAFA CUP server	Client GUI for planning, monitoring, and controlling drone flights.	To create flight plan, monitor it, take over if needed
INT_TELE_UGCS	Telemetry	U-space Interface API (Telemetry)	CAFA UgCS container	FRQ telemetry plugin integrated to CAFA CUP.	Forwards telemetry data to FRQ system and receives alerts from U-space
INT_3D_MAP_QOS _VISUALISING	GUI	UO server	End user	CAFA Analyzer 3D map graphical interface running on Chrome browser and visualising 5QoS measurements.	To visualise 5G QoS in 3D environment
INT_LATENCY-MEASUREMENT	End-to-end latency over 5G	UO server	End user	Latency measurement script is installed in drone onboard computer and AU edge server, and it is using timestamps.	Measuring end-to-end latency drone operator understand the latency of C2 data.

Figure 6 Integration architecture of US3:SC1:sub-SC1

2.3.2.2.4. UC3:SC1:sub-SC2

Following components/enablers are specific for this use case scenario:
- Hepta's data cloud
- Hepta's ground control station
- Mapping LiDAR payload
- Interface with autopilot
- Sensor data streaming
- Data processing

Specific interfaces to be implemented are described in Table 12. The overall integration architecture is presented on Figure 7.

Table 12 Specific UC3:SC1:sub-SC2 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC3:SC1:sub-SC2					
INT_UAV_GCS	C2 over 5G	Hepta UAV	UAV operator VNF	Using mavlink protocol	C2 data and telemetry forwarding
INT_UAV_GCS2	C2 over 5G	Hepta UAV	GCS software (laptop)	Using mavlink protocol, via UAV operator VNF	Receiving telemetry, controlling the drone
INT_TELE_ UAVNVF	Telemetry	U-space Interface API (Telemetry)	UAV operator VNF	Relevant data is extracted from the telemetry stream and sent to UTM	Sending telemetry data to UTM, receiving data from UTM
INT_UBIRD_ CAMERA	Video over 5G	Data Cloud (HEPTA)	Camera	Images are recorded to the on-board computer and sent to uBird via API	Automatically uploading captured images
GUI_VNF	C2 over 5G	UAV operator VNF, Data processing (Point cloud mapping)	UI for VNFs	User interfaces for configuring/controlling UAV operator VNF, point cloud processing	For monitoring and controlling subsystems
GUI_UBIRD	Data over 5G/https	Data Cloud (HEPTA)	UI for uBird	Browser based UI – images attached to towers on a map view and annotations displayed on images	Viewing captured images & processing results
GUI_POINT_ CLOUD_SW	Data over 5G	Data processing (Point cloud mapping)	UI for Point cloud visualisation	Near real time 3D visualisation and colouring based on chosen channels/dimensions	For visualizing the scanned objects for the operator
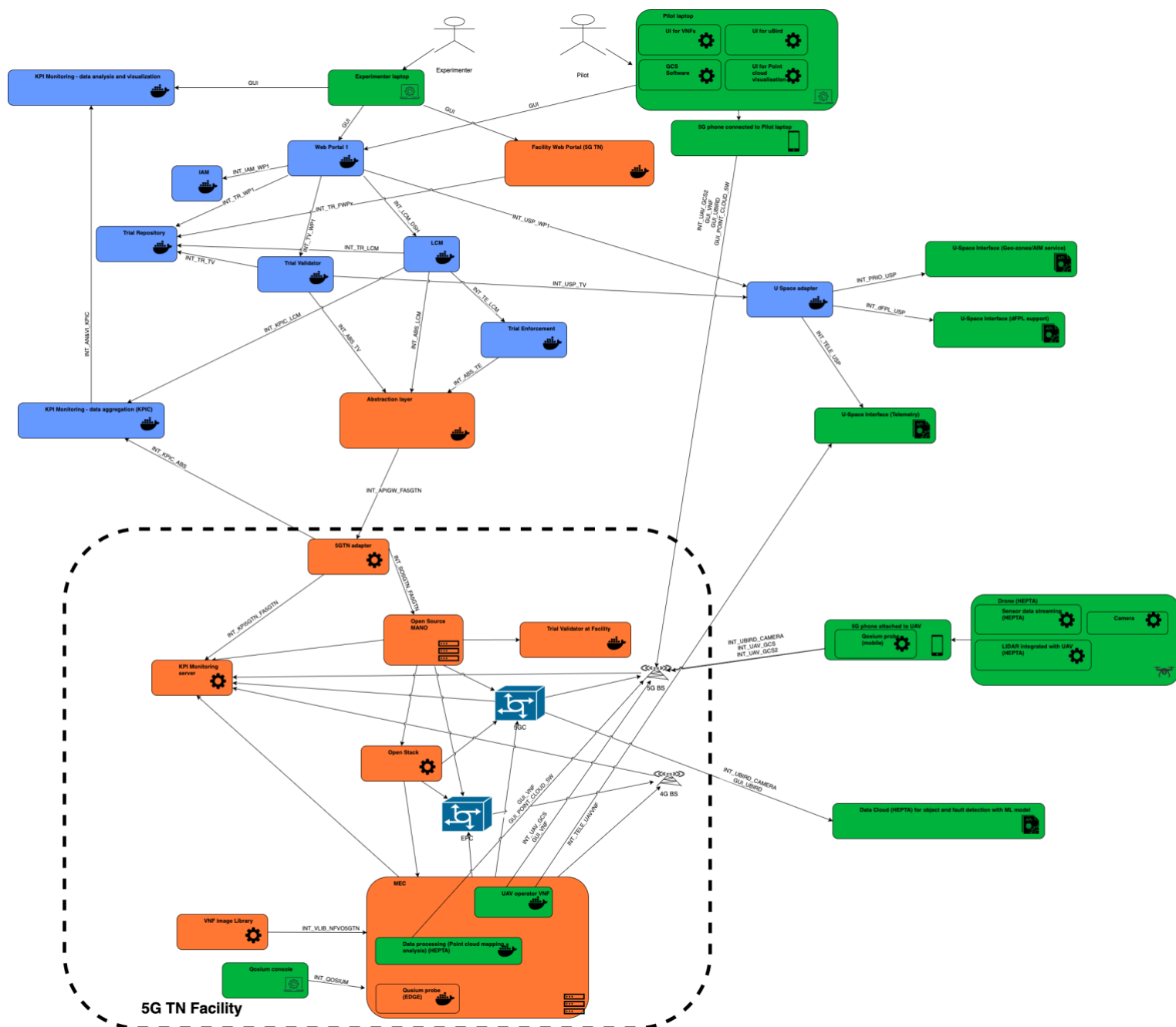
Figure 7 Integration architecture of UC3:SC1:sub-SC2

2.3.2.2.5. UC3:SC1:sub-SC3

The sub-scenario 3 in UC3SC1 aims at assisting search and rescue mission over a large body of water. This scenario is further described in deliverable D1.1. All following Alerion's enablers are specific to this sub-scenario; a description of these enablers can be found in deliverable D3.2.

- Hydradrone: Alerion's amphibian UAV, equipped with a bathymetric sensor
- Data Processing: an application processing raw bathymetric readings and emitting processed bathymetric values
- Sensor Data Streaming: an application handling sensor data acquisition and streaming
- ALE GCS: a Ground Control Station handling C2, payload control, and data visualization

Specific interfaces to be implemented are described in Table 13. The overall integration architecture is presented on Figure 8.

Table 13 Specific UC3:SC1:sub-SC3 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC3:SC1:sub-SC3					
INT_HYDRA_GCS	C2 over 5G	Hydradrone	ALE GCS	MAVLink protocol/ ØMQ	Telemetry data and payload control
INT_SENSOR_DATA _STREAMING	Sensor data over 5G	Hydradrone	Data Processing (edge)	MAVLink protocol/ ØMQ	Raw bathymetric readings to be processed.
INT_TELE_HYDRA	Telemetry	U-space Interface API (Telemetry)	ALE GCS	API used to submit constantly location information (telemetry data).	Sends telemetry to UTM.
INT_DATA_VISUAL	Processed data	Data Processing at the edge	ALE GCS	Once the data is processed at the edge, it will be sent to ALE GCS for visualisation purposes	Provides processed data that will be used for 3D bathymetric map rendering.

Figure 8 Integration architecture of UC3:SC1:sub-SC3

2.3.2.2.6. UC3:SC3

The Use Case 3 Scenario 3 Location of UE in non-GPS environments aim is to collect 5G data from environment and provide an estimation of UE positioning based on that information. Following components/enablers are specific for this use case scenario:

- UO positioning engine
- Keysight Nemo Outdoor data collector

Specific interfaces to be implemented are described in Table 14. The overall integration architecture is presented on Figure 9.

Table 14 Specific UC3:SC3 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC3:SC3					
NT_NOK_NW_DATA	CSV file	Offline tool	Keysight Nemo Outdoor	Collects data for positioning data analytics	Data collector
INT_NOK_CONTROL _DATA	RESTful	Pilot laptop	Nokia Drone	Controls and share information of the Nokia Drone	Controls Nokia Drone

Figure 9 Integration architecture of UC3:SC3

2.3.2.3. Eurecom university facility (5G-EVE)

Table 15 below lists all EUR facilities' interfaces to be deployed and integrated to fully support functionalities of 5G!Drones platform and specific use cases described in following subsections.

Table 15 5G-EVE interfaces to be integrated with 5G!Drones platform

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_KPIC_ABS	RESTful	KPIC	Abstraction layer	https://5gdrones-kpi-endpoint.utm-labs-frequentis.com/swagger-ui/	Push KPIs to KPI Management module.
INT_TR_FWP5 GEVE	RESTful	Trial Repository	Facility Web Portal & Trial Translator (5GEVE)	https://5gdrones.mosaic-lab.org/repositories/swagger/	Update the Network Slice Template (NST) field for the targeted Trial.
INT_APIGW_ FA5GEVE	RESTful	Abstraction Layer	Facility Adapter (5GEVE)	https://gitlab.5gdrones.droneradar.eu/wp3/abstraction-layer/-/blob/master/swagger-description.yaml	Create/Update/Stop/Delete a Network Slice instance. Create/delete measurement jobs

2.3.2.3.1. UC1:SC1

The main focus of UC1:SC1 tests is the services provided by U-space, Trial Controller and Facility to the Ground Control Station deployed by UAV Operator in the EDGE server. The intent is to assure that everything is under control before and during a mission. The plan is to simulate the problems, which can appear, like the addition of the Geozone (as forbidden zone), incoming bad weather alert or connectivity problem. In case of such events, the mission may continue or may be terminated, according to the defined rules. The most important requirement is that everything is happening in a controlled manner.

It is planned to add the INVOLI LEMAN tracker, which will report independently of the on-board drone system its position, direction, altitude, and speed. This is the additional safety backup, which is considered to be required in the real BVLOS operations in medical delivery missions. The INVOLI K1090 receiver connected either through 5G test network or 4G commercial network will deliver to the U-space service the actual airspace traffic in the area of the test.

Detailed integration architecture for UC1:SC1 is presented on Figure 10.

Following components/enablers are specific for this use case scenario:
- CAFA drone T1 with: Pixhawk flight controller, CUP C2 client, Latency measurement script, INVOLI KIVU LEMAN tracker (using 4G network), attached 5G Phone with Latency measurement script, Ookla speed test and Nperf
- INVOLI K-1090 receiver (attached/connected through 4G network)
- INVOLI Central Server (via API exposed through the internet)
- Additional Pilot's laptop with: CAFA CUP C2 server container and client application, attached 5G phone
- VNFs: UgCS C2 cloud-native application enhanced by CAFA Tech (CAFA CUP), Latency measurement script

List of required interfaces to be implemented is provided in Table 16.

Table 16 Specific UC1:SC1 enablers' interfaces

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
US1SC1					
INT_CUP_UGCS	C2 over 5G	CUP C2 client	CAFA UgCS container	Docker container of CAFA CUP	Sends C2 information to drone and receiving telemetry data from the drone.
INT_CUP_LAPTOP	C2 over 5G	CUP C2 client	CAFA CUP server	Client GUI for planning, monitoring, and controlling drone flights.	To create flight plan, monitor it, take over if needed
INT_TELE_UGCS	Telemetry	U-space Interface API (Telemetry)	CAFA UgCS container	FRQ telemetry plugin integrated to CAFA CUP.	Forwards telemetry data to FRQ system and receives alerts from U-space
INT_INVCS_KIVU	Tracking over 4G	INVOLI Central Server	INVOLI KIVU LEMAN Tracker	https://www.involi.com/ products/kivu-drone-tracker	Provides drone's location
INT_INVCS_K1090	ADS-B over 4G/Ethernet	INVOLI Central Server	K-1090 receiver	https://www.involi.com/ products/k-1090-receiver	Receives ADS-B information
INT_LATENCY-MEASUREMENT	End-to-end latency over 5G	UO server	End user	Latency measurement script is installed in drone onboard computer and AU edge server, and it is using timestamps.	Measuring end-to-end latency drone operator understand the latency of C2 data.

Figure 10 UC1:SC1 integration architecture

2.3.2.3.2. UC2:SC1

Main enabler in this so-called "Monitoring a Wildfire" use case is the Critical Collaboration Platform implementing 3GPP Mission Critical Services MCS and deeply evolved on the one hand to leverage 5G networks features and on the other hand to integrate UAV as Drone as a Service. MCS clients are sharing real time and non-real time multimedia data and streams between them to acquire a common situational awareness between them (field users and command centres). Drones' data and streams are also federated in this set. In particular, C2 application is injected on MCS backbone and shared through the platform.

For more details about UC1:SC1, please look at D1.1 "Use case specifications and requirements", chapter 3.4.1.1 – UTM command and control application.

Specific interfaces to be implemented are described in Table 17. The overall integration architecture is presented on Figure 11.

Table 17 Specific UC2:SC1 enablers' interface

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
UC2:SC1					
INT_CUP_UGCS	C2 over 5G	CUP C2 client	CAFA UgCS container	Docker container of CAFA CUP	Sends C2 information to drone and receiving telemetry data from the drone.
INT_CUP_LAPTOP	C2 over 5G	CUP C2 client	CAFA CUP server	Client GUI for planning, monitoring, and controlling drone flights.	To create flight plan, monitor it, take over if needed
INT_4K_VIDEO_ STREAM	RTSP video	Video server	CAFA pilot	4K Video streaming from drone to MEC/edge server and forwarding video feed for end users and drone pilot (Police operator)	To support drone control teleoperation i.e., remote control via 5G and thanks to near real time video from the drone
INT_MCS	3GPP MCS	MCS Critical Collaboration platform	Mission Critical mobile app	Interface supporting Multimedia secured collaboration platform dedicated to Public Safety End Users and integrating Drone as a Service	Sharing and augmenting situational awareness among field users and between field users and command centres

Figure 11 Integration architecture of UC2:SC1

2.3.2.3.3. UC2:SC2

This use-case scenario is a "disaster recovery" simulation in which UAVs are used to provide on-demand network connectivity and video footage of the affected area simultaneously and autonomously.

Additional information and detailed description of the scenario can be found in D4.3 [2].

Specific interfaces to be implemented are described in Table 18. The overall integration architecture is presented on Figure 12.

Following components will be used:
- The UL-Autonomous Control End Node (UL-ACE): This computing unit provides higher processing power than the flight controller and enables higher-level autonomous control by sending C2 commands to the flight controller. The autonomous control endpoint also communicates with a central command platform (UL-CCP) that orchestrates the individual actions of a group of autonomous vehicles, hosted on a local server, on the network edge, or in the cloud. This communication can take place over Wi-Fi, 4G, or 5G.
- The UL-Central Command Platform (CCP): for an UMS deployment, UL-CCP is responsible for gathering data from connected UL-ACEs to create a single context for devices in the fleet. UL-CCP integrates and orchestrates all the systems involved in the solution, processed via the Edge and/or cloud enabling the cloud-based Autonomy-as-a-Service operation.
- The UL-Video Analytics (VA): Typically, this module is located on the edge with the UL-CCP. The focus is on the sensor type and analysis methods that can be applied for different system inspections. E.g., visual inspection will be camera based and will use AI techniques to analyse the camera data to perform object detection. Depending on the training data used during the learning process, the algorithm will be able to detect different types of objects. The modules can be custom made or a third-party plug-in to the UL-CCP.

Table 18 Specific UC2:SC2 enablers' interfaces

Interface ID	Type/Protocol	API Server	API Client	Description	Purpose
UC2:SC2					
INT_ACE_CCP	C2 over 5G	UL-ACE	UL-CCP	The UL- Autonomous Control End Node (UL-ACE) is a computing unit installed on a drone in addition to a flight controller.	Command and control
INT_TELE_CCP	Telemetry	U-space Interface API (Telemetry)	UL-CCP	The UL-Central Command Platform (CCP) is a containerized application which is comprised of software modules and acts as the "central brain"	Orchestration of robotic systems
INT_VA_CAMERA	Video over 5G	Unmanned Life Video Analysis (UL_VA)	Video Camera	The UL-Video Analytics (VA) module collects and processes the sensor data streamed from the drone for computer vision tasks.	Video processing and analysis
INT_LIFEX_USP	REST - Telemetry	LifeX System	U-space adapter	The Frequentis LifeX solution integrates with the U-space adapter to consume positions of drones.	Integration between Public Safety organizations and U-space components
INT_LIFEX_USER	SIP	LifeX System	End User	The LifeX solution provides communication capabilities to the end user.	Public Safety Management and Communication
INT_LIFEX_DISP	WebSocket, SIP	LifeX System	LifeX Dispatcher Client	The LifeX solution provides data as well as communications and management capabilities to the LifeX dispatcher client.	Public Safety Management and Communication
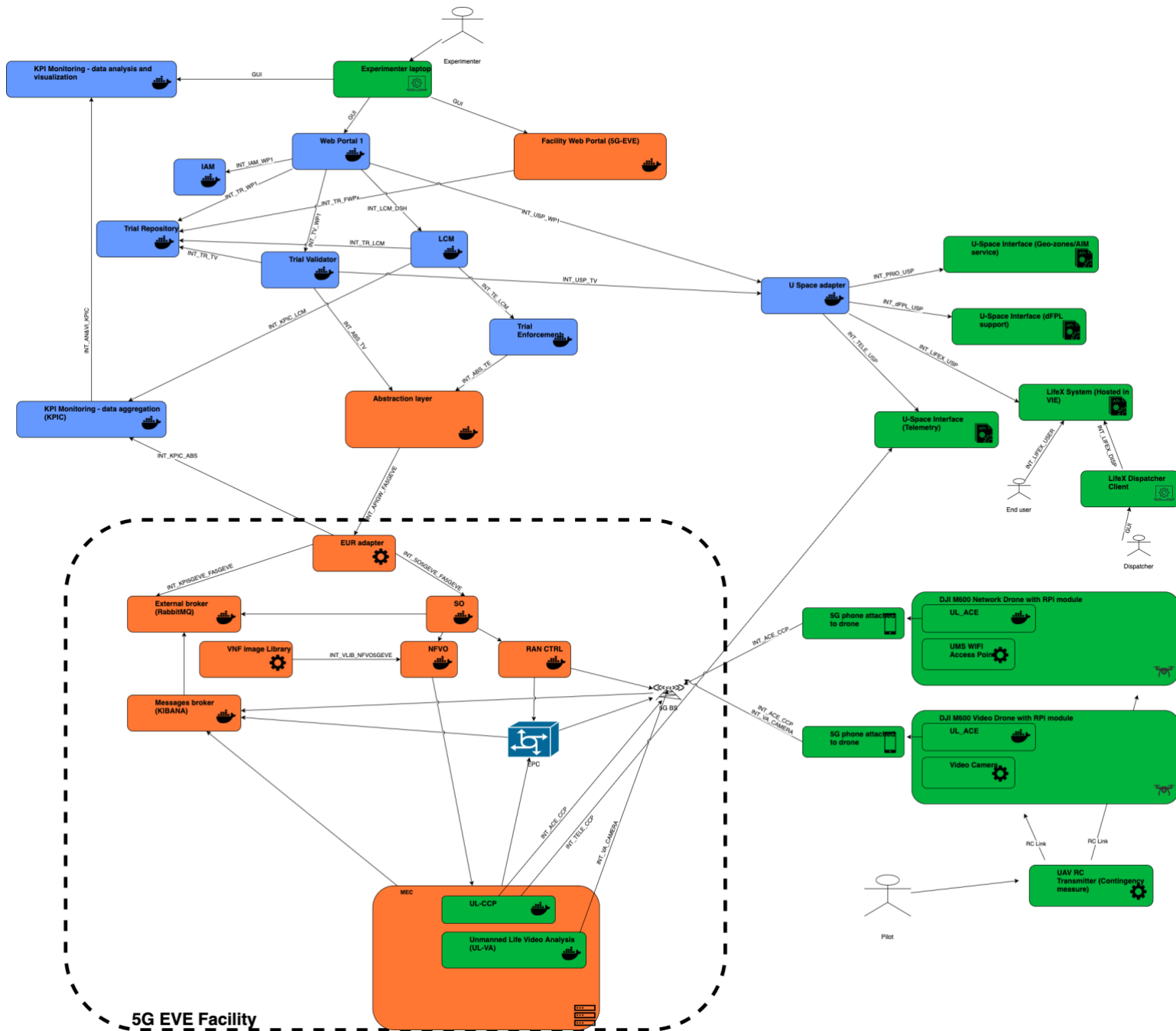
Figure 12 Integration architecture of UC2:SC2

2.3.2.3.4. UC4:SC1 and 5GENESIS facility

5GENESIS is an ICT-17 experimentation facility that provides full automation for the design and execution of a 5G experiment, supporting vertical industries in benchmarking their services and applications.

However, 5GENESIS facility is vertical agnostic, which means that special components needed for the assessment of vertical services are not included in its architecture. Therefore, in the framework of 5GDRONES is given the opportunity to showcase the modular nature of the 5GDRONES trial controller by integrating to the 5GENESIS facility only the components that are needed to assess the vertical

apps under test. In the case of 5GDRONES and 5GENESIS integration, the U-space-specific components, such as the Web Portal 1, UTM etc. are integrated in from the trial controller in order to complement the 5GENESIS functionalities with vertical specific requirements.

The Figure 13 and Figure 14 below projects a common and revised 5G!DRONES-Enhanced 5GENESIS integration architecture, optimized in order to encapsulate and reuse the existing capabilities of the 5GENESIS platform. The specific architecture primarily illustrates the components of -in 5GENESIS terms- Coordination layer and considers some capabilities of the MANO Layer as necessary to collect and maintain key monitoring & measuring metrics for the subsequent KPIs validation.

More specifically, for both use-cases the provided figures illustrate, how the complementarity between the 5G!Drones and the 5GENESIS experimentation frameworks can be fulfilled, providing a direct mapping of the 5G!Drones components, whose functionality is already offered by Open5GENESIS Suite.

For example, as we can observe, the 5G!Drones KPI monitoring components are already covered by the Open5GENESIS monitoring module. As regards to Experiment Analytics, it aims to provide a full and reliable assessment of 5G KPIs, in the form of a) statistical analysis of the KPIs, and b) Machine Learning (ML)-based analysis. The latter takes as input not only the performance KPIs (collected by the PM probes), but also the infrastructure-related parameters (monitored by the IM probes).

Similarly, the Facility web portal is realized by the 5GENESIS Web Portal, which for the needs of the 5G!Drones trials, is interfaced with the Web Portal 1. The 5GENESIS Portal provides a Web interface that experimenters can use for defining new experiments and request their execution, providing real time information about the execution status, including execution logs. Experimenters can also use the Portal for accessing to the results generated by any previous execution, including the visualization of raw results (based on Grafana dashboards) or the download of such results as CSV files for further processing offline, and access, in general, to advanced features. Finally, the 5GENESIS Portal provides an interface that can be used for onboarding new Network Services in the platform. After the onboarding process is completed these Network Services can be used as part of an experiment.

The experiment lifecycle will be controlled by the LCM and Trial Enforcement entities, which together with the trial validator functionality are realized by the ELCM of the 5GENESIS facility. The Experiment Lifecycle Manager is the entity that oversees the execution of an experiment from start to completion, coordinating the different components in the platform. The ELCM also coordinates the execution of different experiments so that the resources required by each of them are available, allowing the execution of experiments in parallel only if they cannot interfere with each other. This management is performed by the Scheduler sub-component of the ELCM.

The SO functionality is realized by the Slice Manager component. The Slice Manager is a centralized software component that provides an interface for creating, modifying, monitoring, and deleting end-to-end network slices. Through the North Bound Interface (NBI), the Slice Manager interacts with a coordination layer from which receives the Network Slice Template (NEST) for creating network slices and provides the API for managing and monitoring them. Based on the on-boarded NEST, Slice Manager has to make the mapping between the available data plane resources and the described slice requirements, providing the proper placement of each Network Service. Through the South Bound Interface (SBI), Slice Manager talks to the other components of the Management and Orchestration Layer (MANO).

Interface ID	Type/ Protocol	API Server	API Client	Description	Purpose
INT_KPIC_ABS	RESTful	KPIC (5GEN)	Facility Adapter (5GEN)	The Open5GENESIS suite is exploited for the assessment of the KPIs which include a visualization tools of the experiment results	The scope of this KPI assessment is the performance benchmarking between 5G and other systems
INT_SO5GEN_FA5GEN	RESTful	Slice Orchestrator (5GEN)	Facility Adapter (5GEN)	Open5GENESIS Suite is equipped with Katana slice manager, which exposes NBIs in order to allow easy integration with vertical industries for experimentation.	The scope of using the slice manager is the selection of the appropriate slice type in order to provide the required QoS at the service under test.
INT_KPI5GEN_FA5GEN	RESTful	KPI measurements server (5GEN)	Facility Adapter (5GEN)	Open5GENESIS suite stores the KPU measurements to a time-series database, where can be further exploited for visualization purposes.	Storing the KPI measurements reassures the reusability of them for further analysis and elaboration towards the KPI evaluation.
INT_VLIB_NFVO5GEN	RESTful	VNF image library/repository/ hub	NFVO (5GEN)	OSM has been selected as NFV orchestrator of the 5GENESIS platform. The communication to the NFVO happens via the Slice Manager.	The orchestration of VNFs is important for the deployment of novel services
INT_APIGW_FA5GEN	RESTful	Abstraction layer	Facility Adapter (5GEN)	Open5GENESIS suites exposes NBIs in order to allow the further integration of the facility with vertical industries that require custom experimentation.	The use of the Open API provides an abstraction layer to the underlying 5GENESIS facility experimentation features.

| INT_TR_FWP5GEN | RESTful | Trial Repository | Facility Web Portal (5GEN) | The Open5GENESIS Suite includes the experimentation portal, which lists various test-cases that the experimenter can select and deploy. | The scope of the pre-defined list of test-cases facilitates the experimenter to select and proceed fast with popular and pre-defined experiments. |
| INT_TR_TT5GEN | RESTful | Trial Repository | Trial Translator (5GEN) | The selected-testcase is mapped to specific automated actions for the execution of the experiment via the dispatcher module. | The translation process that takes place allows the expandability of the experimentation facility with additional test-cases and experimentation components. |

Table 19 Specific UC4:SC1 enablers' interfaces

Figure 13 Integration Architecture UC4:SC1 – Variant 1

Figure 14 integration Architecture UC4:SC1 – Variant 2

2.3.3. Testing and deployment architecture of 5G!Drones platform

In the previous sections, the integration perspective view of all use cases was provided: all required components with their interconnections were listed (as named interfaces) and explained.

As the Trial Controller, U-space services and facilities are common and shared across all scenarios, it was decided on consortium level, that for the purpose of performing all tests, trials executions and demonstrations, there will be one persistent, shared integrated environment of the 5G!Drones platform as presented on Figure 15.
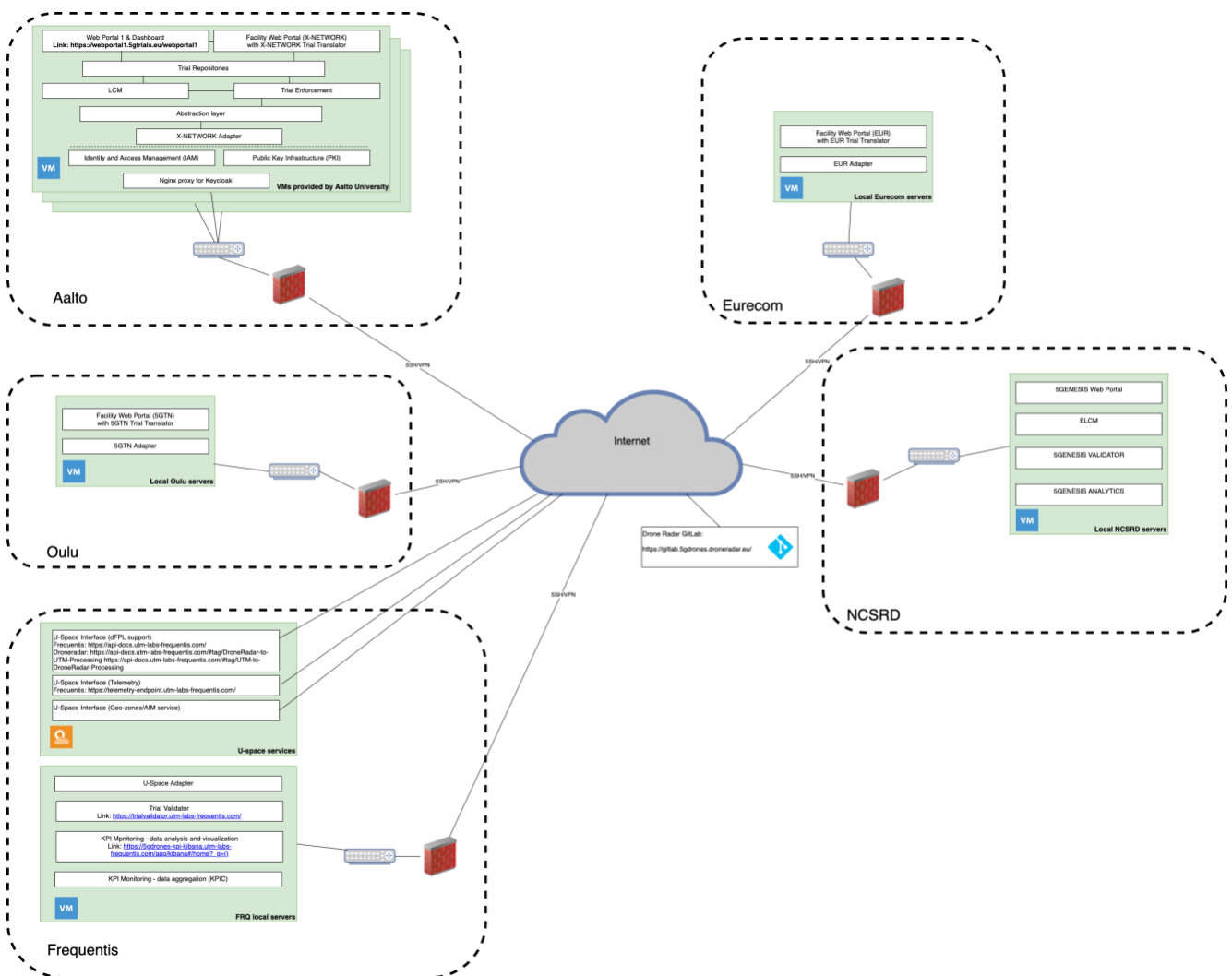
Figure 15 Deployment architecture of the 5G!DRONES

The Aalto University offered to provide the hosting environment for solution's components software modules. Majority of the Trial's Controller components, as shown on the diagram, as well as all other required components (e.g., security components of PKI, IAM, etc.) were deployed on those premises. There are few (U-space adapter, KPI Monitoring module, Trial Validator) components, which are hosted on infrastructure provided by Frequentis. Frequentis is also hosting and exposing all required U-space services. Also, each facility is connected to this Trial Controller instance via Facility adapter components. Each of the facilities hosts also its own, dedicated instance of Facility Web Portal with

Trial Translator. All the components, regardless, where they are hosted, are interconnected via secure connections over the internet.

The described platform constitutes the core of the 5G!Drones system. On top of and in addition to it, whenever required by specific use case requirements, additional components are added. Most typically this would mean hosting use case specific applications as VNFs on facilities' edge servers and eventually integrating additional specialized components/platforms. All such details are provided in respective sections dedicated to each of the use cases earlier in this document.

To allow to any of the project's partners to build and host its own copy of the environment, all the required software packages are stored in the project's GitLab repository.

3. UPDATED INTEGRATION PLAN

The updated integration plan part of the document consists of two main sections:
- The captured individual delivery plans of the enablers and facilities which are being developed during the course and for the purpose of the project
- The prepared plans for four main releases planned for the implementation

3.1. Delivery plans

In following subsections implementation details for all enablers, that are being developed during the project are provided. The tables contain list of enablers with implementation timeframes aligned with major releases periods thus allowing to plan when which enabler can be tested. The lists are grouped by partners and facility owners.

3.1.1.   Delivery plan for Trial Controller Modules

Trial Controller Modules	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
Web Portal	Secure login and users' management, UAV – new/edit/info/del, dFP creation -new/edit/info/del, redirection to Web Portal 2	Redirection to Trial Validator and back, templates, enhancement of map planning, integration with LCM	Adding KPIs analysis, detailed dashboard, other requests identified after Releases 1 and 2	Maintenance and bug fixes
Trial Validator	First initial release for Backend and Frontend with only Hardcoded configuration.	Retrieve Trial Config from Trial Repository. Showing FPL status.	Validation of dFPL against U-space.	Status of 5G Facility component. Bugfixes.
Trial repository	Full content release	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration
Life Cycle Manager	Released as individual component with all features	Full content release	Maintenance release based on integration test feedback	Maintenance release based on integration test feedback
Trial Enforcement	Release of Slice Management capabilities	Full content release and integration with Abstraction Layer	Integration with LCM and rest of the platform	Maintenance based on the feedback from the integration
KPI Monitoring	Initial REST interface for KPIs.	Additional KPI markings supported.	Added request to provide KPI reporting components for LCM integration.	Bug fixing, additional improvements
U-space adapter	Telemetry	GeoZones	Flight plans	Additional improvements

3.1.2. Capabilities delivery plan for Facilities

3.1.2.1. 5G-EVE

5GEVE	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
MEC	Full functionality	Full functionality	Full functionality	Full functionality
Facility Web Portal	Full functionality	Full functionality	Full functionality	Full functionality
Facility Adapter	Full content release	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration
KPI collection	Full functionality	Full functionality	Full functionality	Full functionality

3.1.2.2. 5GTN

5GTN	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
vMEC	Full functionality	Full functionality	Full functionality	Full functionality
Abstraction Layer 5GTN Parser	Full functionality	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration
KPI Measurements REST API	Basic automated measurement job creation	Full functionality	Maintenance based on the feedback from the integration	Maintenance based on the trials
Facility Web Portal (OU)	Secure login and users' management, slice creation and deletion capability.	Redirection from wen portal 1 to web portal 2	Integration with Trial Repository and KPI analysis	Maintenance and bug fixes

3.1.2.3.   X-Network

X-Network	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
MEC/Edge		Full functionality	Full functionality	Full functionality
Facility Web Portal (AU)	Full content release	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration
KPI collection		Full functionality	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration
Facility adapter		Full functionality	Maintenance based on the feedback from the integration	Maintenance based on the feedback from the integration

3.1.2.4.   5GENESIS

5GENESIS	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
5G NSA	5G NSA is available in Rel.1 with some commercial UEs to support this	5G NSA will be further complemented with open 5G UEs that can be programmed.	5G NSA will be further complimented with 5G CPEs capable of providing WIFI6 and other network interfaces	All the previous NSA UEs will be available in the final release of the facility
5G SA	5G SA is available in Rel.1 with some commercial UEs to support this	5G SA will be further complemented with open 5G UEs that can be programmed.	5G SA will be further complimented with 5G CPEs capable of providing WIFI6 and other network interfaces	All the previous SA UEs will be available in the final release of the facility
EPC	Amarisoft and Athonet EPC for NSA deployment is available	Athonet EPC for NSA deployment with LBO capability is available	Both EPCs are available. LBO capability is supported by Athonet EPC.	Both EPCs are available. LBO capability is supported by Athonet EPC.
5G-Core	Amarisoft 5G-Core for SA deployment is available	Amarisoft 5G-Core for SA deployment is available	Amarisoft 5G-Core for SA deployment is available	Amarisoft 5G-Core for SA deployment is available

5G-NR	Amarisoft 5G-NR is available for deployment	Nokia 5G-NR Airscale is available for deployment	Both 5G-NRs are available for deployment	Both 5G-NRs are available for deployment
MEC	Far-edge deployment served as MEC	MEC deployment with NOKIA airscale and ATHONET EPC	MEC deployment with Athonet EPC is available	MEC deployment with Athonet EPC is available
Slice Manager	Slice Manager available with for multiple shared NSSIs among slices	Slice Manager supports scripting for building, starting, testing, and packaging	Slice manager added CI/CD pipeline	Slice manager is available.
Web Portal 1	Web portal 1 has been added but not integrated with Facility web portal.	Web portal 1 has been integrated with the Facility web portal.	The experimenter can design via Web Portal 1 the flight plan and upon receiving approval the system forwards him to the Facility Web portal.	Web portal 1 is available.
Facility Web Portal	5GENESIS platform offers the experimentation portal, through which test-cases could be selected	Facility web portal will be complemented with UAV specific functionalities via Web Portal 1.	Facility web portal will offer test-cases that include the flight plan as part of the experiment.	Final version of web portal facility is available.
Trial Validator and translator	Open5GENESIS experimentation layer includes the dispatcher component that performs validation and mapping of the experiment to underlying experimentation features.	Extensions will be included to the component in order to feasible to tackle with UAV specific test-cases.	The full functionality will be ready.	The full functionality will be available.
Life Cycle Manager	Experiment LCM will be in place for performing experiments without including flight as part of the experiment.	Experiment LCM will be extended to include flight as part of the experiment.	The full functionality will be available.	The full functionality will be available.

3.1.3. Delivery plans for Use Case Enablers

3.1.3.1. AALTO service enablers

AALTO	Functionalities Delivered within a Release Cycle			
	Release 1	**Release 2**	**Release 3**	**Release 4**
UAV platform	Full functionality	Full functionality	Full functionality	Full functionality
IoT as a service platform	Full functionality	Full functionality	Full functionality	Full functionality

3.1.3.2. AIRBUS service enablers

AIRBUS	Functionalities Delivered within a Release Cycle			
	Release 1	**Release 2**	**Release 3**	**Release 4**
Mission Critical Services server	Integration of Cloud native critical collaboration platform with a completely new architecture and implementation optimized for 5G infrastructures paradigms and drones' applications (from 5G!Drones collected requirements) first release on 5G!Drones facility (Eurecom)	Integration of evolutions from release 1 with a new breakdown in micro-services and optimizing Mission Critical Video for application on 5G!Drones infrastructures after pre trials.	Integration of other Mission Critical Services functionalities to Release 3 MCPTT and MCDATA. Fully stateless approach releases functional.	Integration of Mission Critical Services interfaces fully available and functional for 3rd parties
Mission Critical Services client	Integration of Mobile client to MCS server release 1 fully functional for pre-trial tests and embedding latency and throughput performance measurements assets	Integration of Mobile client evolved for MCS server release 2. Integration of Web browser application available for integrating COTS drones into trials.	Integration of Final mobile and Web browser client release available. Integration of PC application available for embedding MCS client directly in drones.	

3.1.3.3. ALERION service enablers

ALERION	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
Alerion GCS	First version will be delivered during the first Release. Updates on this enabler (if any) will be delivered in future releases.			
Data Processing		This component is currently in dev. We expect to deliver one of the modules (data processing) in the 2nd release.	The second module (3D visualisation) is expected for the 3rd release.	
Sensor Data Streaming	First version of this component will be available for the first Release.	Improvements on this component will be delivered in R2		
Hydradrone	ALE will deliver a first version of its Hydradrone during the first release. Improvements are expected and thus, new "versions" of the drone should be available in future releases.			

3.1.3.4. CAFA TECH service enablers

CAFA TECH	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
UgCS C2 cloud native application enhanced by CAFA Tech (CAFA CUP)	All functions except receiving U-space data	All functions except receiving U-space data	All functions except receiving U-space data	Fully functional
CAFA cellular drone (Pixhawk platform based) supporting onboard 5G commands via 5G module	In development phase	All functions except video packing and delivery electrical hook mechanism	All functions except video packing	Fully functional
4K Video streaming system from drone to MEC/edge server	In development phase	All functions except video packing and adaptation	All functions except video adaptation	Fully functional
CAFA Video analyzer (VideoLyzer)	In development phase	In development phase	Ability to detect predefined objects	Fully functional
3D map for visualising QoS of 5G (CAFA Analyzer)	In development phase	3D map functions except visualising QoS	Initial visualisation QoS points except different colouring based on QoS.	Fully functional
CAFA Field GIS C2 system for Police operations and Drone Logistics	In development phase	In development phase	All functions except drones' real time locations	Fully functional

3.1.3.5. DRONERADAR FREQUENTIS service enablers

DRONERADAR FREQUENTIS	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
Telemetry Enabler	First version (legacy API) – integration with CAFA Tech	Integration with UMS (legacy API)	Integration with ALE (legacy API)	Full version (standardized API)
Geo-zone/AIM service enabler		First version (exchange of information between UTMs)	Exposing Geo-zones /AIM information for pre-tactical phase	Exposing Geo-zones for tactical phase
Flight Plan enabler		Receiving flight plans	First version with flight plan validation	Full version

3.1.3.6. HEPTA service enablers

HEPTA	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
Hepta's data cloud	-	Importing images to uBird via API, triggering ML processing.	Refinements & bugfixes if needed	-
Hepta's GCS	-	User interfaces for VNFs, GCS software, payload & on-board equipment control	Refinements & bugfixes if needed	-
Hepta's drone with tether	Tether integration for Hepta's heavy lift helicopter drone H19 – flight tests, determine payload capabilities, test tether datalink, tether length optimal	Mounts for equipment to be lifted (if needed)	-	-
Payload – LiDAR for 3D mapping	Lidar hardware setup – VLP-16, interface box, on-board computer, IMU, GPS, power setup, mounting to UAV. Lidar software setup – ROS + needed libraries & packages.	-	-	-
Interface with autopilot	Controlling the UAV from GCS via this interface, interfacing with UTM – send telemetry	Logging telemetry, interface for 3rd parties, encrypted communication	Refinements & bugfixes if needed	-
Sensor data streaming	Stream LiDAR scanned points over any internet connection, dynamically change point density.	Refinements & bugfixes if needed	-	-
Data processing	-	Point cloud mapping & conversion	Refinements & bugfixes if needed	-

3.1.3.7. INVOLI service enablers

INVOLI	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
Networked Remote ID (telemetry)	Available in Rel. 1 as commercial product (LEMAN)			
Broadcast Remote ID	Development of broadcast function – defining the requirements, software development	Prototype ready for tests with different receiving equipment, bug fixes and adjustments	Product ready for use	
Reception of ADS-B, Mode A/C, Mode S and Flarm signals and Interface with ATC and drone operator	Available in Rel. 1 as commercial product			

3.1.3.8. NOKIA service enablers

NOKIA	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
DJI Matrice 600 payload adapter	Done, ready for use			
5G Modem integration		Under development		Ready
UWB based drone positioning system	Ready, but due to Covid-19 this technology is not used in trials as trials are performed outdoor environment			
5G smartphone & camera holder	Done, ready for use			
Data Collector	Done, ready for use [https://github.com/nokia/5GDrones-data-collector/]			
Nokia Drone				Ready

3.1.3.9. ORANGE service enablers

ORANGE	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
Geo-location of an UAV based on signal strength from 1 BS				A feasibility is study is under way by getting appropriate measures to inject in the algorithm. We expect to deliver it as an application running on UAV device.
Optimal trajectory of UAV Base Station				A feasibility is study is under way by getting appropriate measures to inject in the algorithm. We expect to deliver it as an application running on UAV device.
Drone Attestation (security enabler)		This enabler is being developed. We expect to deliver a component that can be illustrated on an application.		

3.1.3.10. OULU service enablers

OULU	Functionalities Delivered within a Release Cycle			
	Release 1	Release 2	Release 3	Release 4
5GTN Position server and UI		First release	Fully functional release	Maintenance release
5GTN virtual flight controller application		First release	Fully functional release	Maintenance release
5GTN Data collection and mapping for virtual flight controller		First release	Fully functional release	Maintenance release
Qosium Network Measurement Tool	Monitoring and measurement of network metrics (e.g., throughput, delay, jitter, packet loss).	Additionally, monitoring and measurement of RSRQ, RSRP, RSSI, and SINR for Android mobile phones.		
Nemo Air Interface Measurement Tool	Monitoring and measurement of radio coverage metrics of the mobile devices air interface (full functionality).			

3.1.3.11. UNMANNED SYSTEMS service enablers

UNMANNED SYSTEMS	Functionalities Delivered within a Release Cycle			
	Release 1	**Release 2**	**Release 3**	**Release 4**
UL-ACE	Integration of hardware and installation of companion computer software to enable them for autonomous control operations, Waypoint navigation, Mission functionalities (play, pause, RTH, emergency stop)	Integration of for end-to-end implementation, Tests in real-life environment for validation of all system functionalities, Development of APIs to interact with 5G facilities/edge	Integration with Drone from Hepta & installation of necessary payload	
UL-CCP	Creation of behaviour tree nodes, Creation of a behaviour tree for swarm orchestration, Integrating the output of the object recognition tool with the behaviour tree, integration with 5G edge infrastructure, integration with UTM	Integration of for end-to-end implementation, Tests in real-life environment for validation of all system functionalities, Development of APIs to interact with 5G facilities/edge	Development of UAV interfaces API to interact with Trial controller, UL-ACE Mission monitoring, APIs to integrate with U-Space Adapter	Software tools to measure KPIs, Development of APIs for KPI measurement and monitoring, Development of software front-end and end-user interface (UL-WEB)
UL-VA	Streaming video from the drone to the UL-CCP for object recognition, Video analysis and image processing of pre-recorded video feed	Integration of for end-to-end implementation, Tests in real-life environment for validation of all system functionalities, Development of APIs to interact with 5G facilities/edge	Multi-image analysis with live video feed, Integration with 5G edge infrastructures, integration with UTM	
Wi-Fi AP	Hardware setup, Network configuration			
Simulation Testbed	Design and development, System validation, Preliminary communication tests of over 5G networks	End-to-end tests with pre-defined KPI measurements, Integration with third-party systems		

3.2. Releases

Following subsections provide detailed planning of each release.

3.2.1. Release 1: Integration validation release (M22-M27, March-August 2021)

3.2.1.1. Introduction

The Release 1 focus is to validate integrations between Trial Controller platform and external systems: facilities, U-space services, and other enablers. Its goal is also to identify and define tools and methodologies, how integration tests for this and remaining release will be defined and executed. The aim of the activities performed during the period of this release was also to align on design, build and set up the runtime environment of 5G!Drones platform for all of the tests, trials, and demonstrations.

3.2.1.2. Scope of Integration validation release

The scope of the Integration validation release related to enablers covers:
- gathering the integration requirements and prerequisites for all enablers,
- defining integration testing test cases and procedures,
- validating of those procedures by performing selected integration tests.

Scope of the integration validation related to the Trial Controller covers:
- gathering the specifications of APIs of all internal components of the Trial Controller,
- defining integration testing test cases and procedures,
- validating of those procedures by performing selected integration tests.

3.2.1.3. Deliverables of Integration validation release

Integration of the Trial Controller and use cases enablers
- List of all Trial Controller modules and all use case enablers with their interfaces – information is gathered in **Error! Reference source not found.** (Trial Controller) and in Section 2.3.2 (enablers)
- Integration test specifications – test specifications and results are described in [3]
- Results of integration tests execution according to test specifications – report from remote tests performed in Eurecom in June 2021 is summarized in Section 3.2.1.6.1

Defining deployment and testing tools and architecture
- Deployment approach is summarized in Section 2.3.3
- Testing methodology is described in Section 2.2

3.2.1.4. Schedule of the Release 1

Schedule of the Release 1 is presented on Figure 16.

	M21	M22	M23	M24	M25	M26	M27
Release 1 (Integration validation release) Plan							
1. Collect all the required information: related to Trial Controller interfaces and integration of enablers							
2. Perform analysis of gathered information and prepare recommendations for tools/methodologies related to: integration framework, deployment, testing							
3. Prepare test specifications							
4. Prepare tests environments, e.g., align and synchronize with Pre-trials tests							
5. Perform required installations, configurations, tests							
6. Prepare and distribute feedback							

Figure 16 Schedule of the Release 1

It is important to understand, that mentioned on Figure 16 task 1 will span across all releases (until all enablers are classified). Tasks 3,4,5 and 6 are common for all releases. Testing related activity (Task 5) will be executed separately for each facility, according to the synchronized plans of the on-site trials planned to be performed within the T4.2 (more details on this planning are in deliverable D4.3).

3.2.1.5. Testing scope of the Release 1

Tests of interfaces between the Trial Controller and external/key enablers:

- Trial Controller <-> Facility: tests of integration of Abstraction Layer (with Facility adapter) and facility Slice Orchestration (OSM, SO, etc.)
- U-space adapter <-> U-space: tests of the operational flight plan handling, telemetry data streaming
- VNF library <-> Edge/MEC server: tests of onboarding process of VNFs
- Tests of Trial repository API

Scope of tests:
- Trial controller<->Facility:
  - tests of NS enforcement,
  - tests of KPI stream receiving and forwarding to KPI Monitoring (KPIC)
- U-space adapter<->U-space:
  - Tests of operational plan submission and response receiving
  - Tests of streaming telemetry data through U-space adapter
- Tests of Trial repository API
- Integration tests of all API calls, validating completeness of API implementation from various components perspective (Web Portals, LCM, Trial Enforcement, etc.)

Test sequence of integration tests is presented on Figure 17. It is prepared for 5G-EVE facility, but remaining facilities would have the same set/sequence of tests (there will be only differences in facility specific configurations of tests). In total there are 23 specific test cases defined and described in Test Plan [3], which fully validate the implementation of integration architecture planned for Release 1.
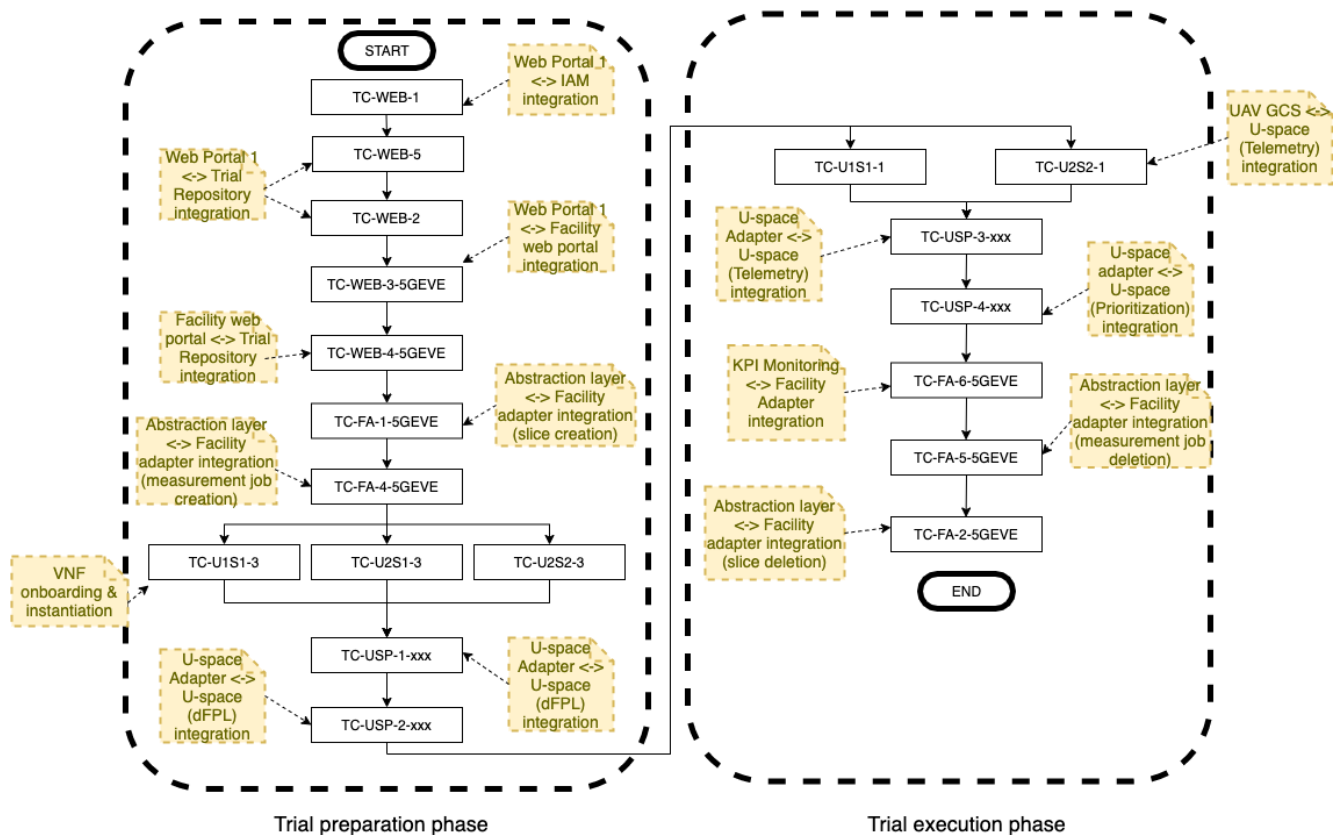

Figure 17 Release 1 test sequence (5G-EVE drop)

3.2.1.6. Status update of the Release 1

In following sub-sections information on current integration testing activities is provided.

3.2.1.6.1. 5G-EVE Drop

On the 17th of June 2021 series of trial and integrations tests with Eurecom facility were performed. The complete report from executed tests in provided in Table 20.

Table 20 List of integration tests performed in Eurecom (17th of June 2021)

Nr	Index of test cases	Result
1	TC-WEB-1 – Login into Web Portal 1	Completed successfully
2	TC-WEB-5 – Save/retrieve/modify UAV to/from Trial Repository (Web Portal 1)	Completed successfully
3	TC-WEB-2 – Save/retrieve trial configuration to/from Trial Repository (Web Portal 1)	Completed successfully
4	TC-WEB-3-5GEVE – Redirect from Web Portal 1 for NST preparation	Completed successfully
5	TC-WEB-4-5GEVE – Save/retrieve trial configuration to/from Trial Repository (Facility web portal)	Completed successfully
6	TC-FA-1-5GEVE – Create slice (5G-EVE)	Completed successfully
7	TC-FA-4-5GEVE – Create measurement job (5G-EVE)	Completed successfully
8	TC-U1S1-3 – Onboard and instantiate CAFA Tech UgCS Platform (CUP)	Completed successfully
9	TC-U2S1-3 – Onboard and Instantiate CAFA Tech UgCS Platform (CUP) and Airbus MCS (Mission Critical System) Platform	Completed successfully
10	TC-U2S2-3 – Onboard and Instantiate UL-CCP	Not used on June 17
11	TC-USP-1-DRR – Submit dFPL	Not used on June 17
12	TC-USP-1-FRQ – Submit dFPL	Not used on June 17
13	TC-USP-2-DRR – Receive res/req from U-space	Not used on June 17
14	TC-USP-2-FRQ – Receive res/req from U-space	Not used on June 17
15	TC-U1S1-1 – Send telemetry data from CAFA Tech UgCS to U-space	Not used on June 17
16	TC-U2S2-1 – Send telemetry data from UL-CCP to U-space	Not used on June 17
17	TC-USP-3-DRR – Receive telemetry data from U-space	Not used on June 17
18	TC-USP-3-FRQ – Receive telemetry data from U-space	Not used on June 17
19	TC-USP-4-DRR – Receive new Geofencing notification from U-space	Not used on June 17
20	TC-USP-4-FRQ – Receive new Geo-fencing notification from U-space	Not used on June 17
21	TC-FA-6-5GEVE – Receive KPI data	Completed successfully
22	TC-FA-5-5GEVE – Delete measurement job	Completed successfully
23	TC-FA-2-5GEVE – Delete slice	Completed successfully

Due to the ongoing development, some of the integration tests with U-space service enablers (11-20) were moved to be performed during August drops (with 5GTN and X-NETWORK). All the other planned tests were successfully executed.

3.2.1.6.2. 5GTN Drop

The scope and specification of tests for 5GTN facility would be the same as for described above 5G-EVE. Some specific configurations of the facility will be adjusted. On-site tests are planned for August 2021.

3.2.1.6.3. X-NETWORK Drop

The scope and specification of tests for X-NETWORK facility would be the same as for described above 5G-EVE. Some specific configurations of the facility will be adjusted. On-site tests are planned for August 2021.

3.2.1.6.4. 5GENESIS Drop

The scope and specification of tests for 5GENESIS facility still need to be prepared. It will be based on the same scope but need to be adjusted to 5GENESIS specificity. On-site tests are planned for October 2021.

3.2.2. Release 2: Trial Controller release (M28-M31, September-December 2021)

3.2.2.1. Introduction

The Release 2 focus is to integrate 5G!Drones Trial Controller. The trial controller consists of several components, which are defined in D1.6, D2.2 and D2.4. After Release 2 there should be the first functional version to plan and execute trials in all four 5G facilities. The experimenter can plan flights and needed services from a 5G facility. During the trial execution phase, the Trial Controller can execute trials with existing 5G facility services. After the execution phase Trial Controller can close needed services.

Definition of release scope:

- All Trial Controller components based on maturity level
- The Abstraction Layer with facility parsers based on maturity level
- All 5G facility service enablers based on maturity level

3.2.2.2. Scope of Trial Controller release

The scope of this release is to confirm full ability to perform all trial scenarios:
1. Ensure all Trial Controller components are developed
2. Ensure that all Trial Controller components are integrated and working with all four 5G Facilities services

Following items are not part of the Release 2 content:
- Detailed KPI data collection and analytics – this is the scope of the Release 3
- UAV specific enablers
- All use case specific enablers and applications
- Final UTM integrations
- Final Trial Controller or Abstraction Layer content as this is the first feature release of Trial Controller. The following releases will be including needed feature updates and bug fixes
- All 5G Facility service enablers
- All security requirements and functionalities integration, testing, and validation. If single components have in release security related features and tests, those are integrated, tested, and validated.

The list of components to be included in Release 2:
- The Trial Controller components
    - Web Portal 1
    - 5G Facility portals (all four 5G Facility portals)
    - Dashboard
    - Trial Engine
        - Trial Translator

- Trial Validation
- LCM (Lifecycle Manager)
- Trial Enforcement
  - o Repositories
    - VNF Repository
    - UAV Repository
    - Predefined Trials Repository
  - o KPI monitoring
  - o U-space adapter
- Abstraction layer
  - o The Front-end, which is a generic part API
  - o 5G-EVE translator
  - o 5GTN translator
  - o X-NETWORK translator
  - o 5GENESIS translator
- 5G Facility Enablers
  - o The basic functionality or SW stubs to test above listed component. These enablers are listed on facility specific Drop Chapter below.

3.2.2.3. Release 2 implementation schedule

1Schedule of the Release 2 is presented on the Figure 18.

Release 2 (Trial Controller release) Plan	M27	M28	M29	M30	M31
1) Define and collect required information of all Trial Controller Release Component interfaces and functionalities – process is triggered during Release 1 and components are documented to D1.6, D2.2 and D2.4					
2. Perform analysis of gathered information and update recommendations for tools/methodologies released to integration framework, deployment, and testing					
3) Investigate and finalize possible data types and formats, classify data (critical + payload) analyze.					
4) Prepare and update test specifications - what and how the release content will be tested					
5) Prepare the test environment to all four 5G test facility environment. Perform configurations, installations, deployments etc. of the release content before actual integration tests.					
6) Perform final configurations, installations, deployments, make the release integration tests. If necessary, repeat the tests with modified configurations					
7) Analyze results, prepare the report and share it among project members					

Figure 18 Schedule for Release 2

3.2.2.4. Outputs of Release 2 (Trial Controller)

There will be following topics in Release 2 report:
- Updated list of all released components with their interfaces, interconnected modules, types/structures of exchanged data, communication categories (multi-/uni-/broadcast), data traffic characteristics, etc.

- If necessary, proposal of integration framework (rules/practices) and infrastructure to integrate components with different communication needs – based on the analysis
- Updated integration test specifications for selected cases according to methodologies
- Results of performed integration tests according to test specifications
- Feedback to development teams (change requests, issues, etc.) based on testing results
- Feedback to 5G Facilities from trial teams
- Feedback from testing deployments like collected/updated requirements, dependencies, deployment, and configuration instructions for all the enablers in scope deployment methodology guidelines, recommendations how modules should be prepared/packaged for further deployment using selected CD tools and hosting infrastructure configuration tips etc.
- Results of performed integration tests according to test specifications
- Feedback from integration and testing sessions
- Recommendations and lessons learn to the next releases

3.2.2.5. Release architecture and sequence

See the Figure 19. below to see the release integration test flow in a high-level. The detailed Trial Controller sequences and requirements are listed in D1.6 and D2.4.
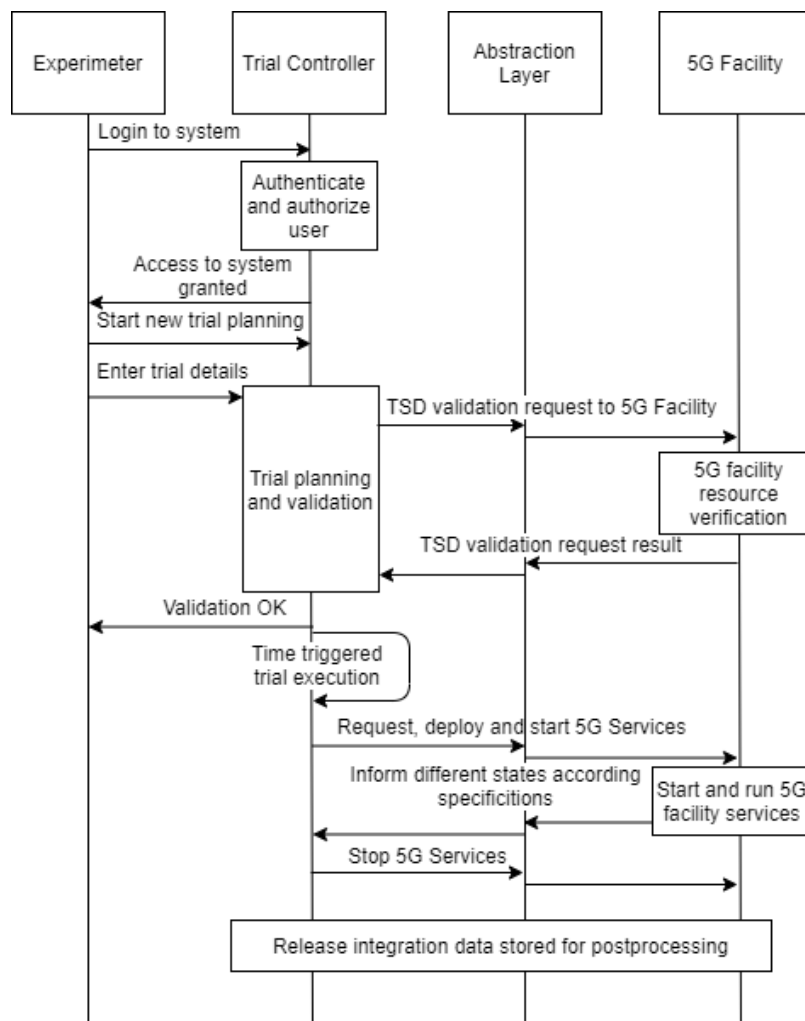

Figure 19 Release 2 Trial Controller integration tests flow

3.2.2.6. Release 2 integration testing

The release integration tests requirements and specifications are similar to respective tests of the Release 1. For example, requirements to testing interfaces between the Trial Controller, Abstraction Layer, 5G Facilities and external/key enablers are same. In Release 2 special focus in testing should be used for the Trial Controller internal interfaces related test cases. Thus, it is assumed, that those tests, as the Trial Controller is not facility specific, would not need dedicated facility drops. This is still under evaluation and whenever necessary, the appropriate activities will be planned to perform on-site tests during trials planned for August and October.

3.2.3. Release 3: KPI release (M32-M35, January-April 2022)

3.2.3.1. Introduction

For this release step, we have almost all modules of the Trial Controller in place, integrated and working together – the experimenter can plan, submit for validation, and execute the tests. They are performed to confirm the assumptions and measure in practice the theoretical calculations. The results should be stored, and the experimenter should be able to see, manipulate and analyse the received data in the most convenient way – in the form of raw data in excel, chart, graph, or table. This 3rd release will be focused on the validation of KPIs acquired from all the required sources, their monitoring while the experiment is being performed and post-test analysis. It is assumed that in the previous releases data was acquired and stored from different sources (facilities, trial controller, UTM, etc.), but in this release, the intent is to validate data itself and confirm, that we are collecting all the interesting data, in suitable format – regarding the periodicity, accuracy and synchronisation (timestamping) of the streams arriving from the different places. Mentioned synchronization is particularly important data quality measure, as bad correlation of the streams can ruin the whole effort invested in the experiment.

3.2.3.2. Sources of KPIs

During the tests, the data will be collected in multiple places (see Figure 20). Following sources are identified:

- Facilities IT infrastructure, including MEC server – KPIs like RAM usage, CPU/GPU load, throughput on some interfaces, etc.), RNIS data
- 5G KPIs collected in the gNodeB and core network
- 5G KPIs collected in the UE or modem carried by drone
- Data collected by IoT sensors on-board the drone
- Data collected in the 5G!Drones Trial Controller
- Telemetry data from UAV and other data shared by UTM service
- Video signals or pictures taken by the drone and transmitted to the off-drone systems

During some missions, the UAS operations can create massive amounts of data. Most of that volume can be considered as the payload, which is important for the customer requiring the UAV service, but irrelevant for UAV operation or safety. We should prioritize the safety and security related data (e.g., C2 payload), but the measurements or other data payloads types (e.g., video transmission) are also important, because without them, the mission is not justified.
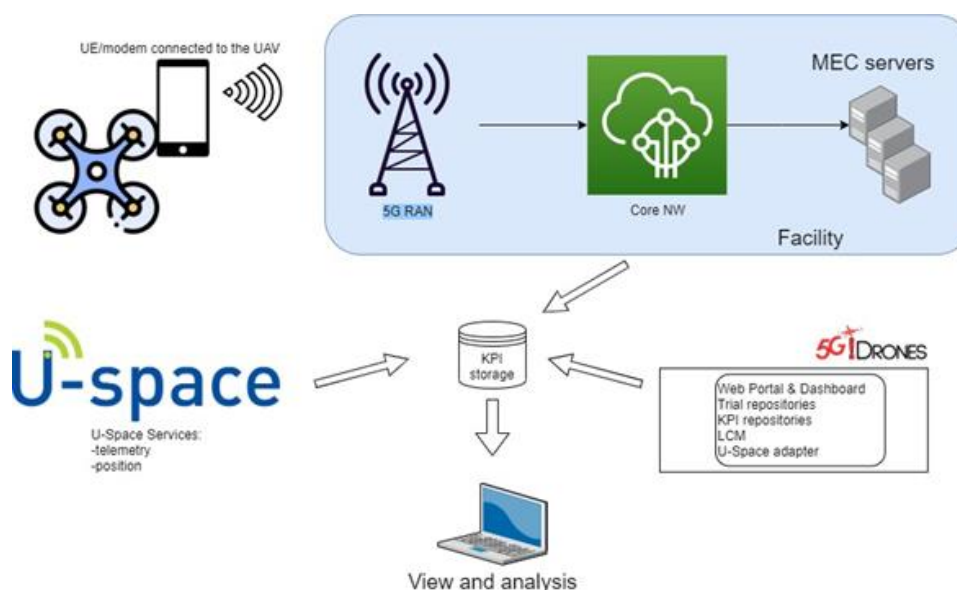
Figure 20 KPIs sources

3.2.3.3. Implementation schedule

This release is planned for January – April 2022, but preparation work can be carried out during Release 1 and 2 (see the Figure 21), like definition of KPIs and related data sources (task 1 on schedule, which starts earlier than KPI Release implementation but is strictly related to the scope of this release).

	08'2021	09'2021	10'2021	11'2021	12'2021	01'2022	02'2022	03'2022	04'2022
Release 3 Plan : KPIs release	M27	M28	M29	M30	M31	M32	M33	M34	M35
1) Define all sources of collected KPIs - process can be triggered during Rel. 1 and 2									
2) Agree on the storage: location, required space, storage expansion etc. - should be finalised before Rel.3									
3) Investigate the possible data types and formats, classify data (critical + payload) - should be finalised before Rel.3									
4) Prepare test specifications - what and how we want to test - is it facility/use case specific? - when rel. 2 is on-going									
5) Prepare the test environment - investigate the possibility to make simulations for stress/data load - at the end of rel.2 we should have majprity of the test environment for KPIs ready									
6) Perform configurations, make the tests, if necessary repeat the tests with modified configurations									
7) Analyse results, prepare the report and share it among project members									

Figure 21 Schedule for Release 3

3.2.3.4. Testing scope

The general scope description is presented here. The details of tests for Release 3, per each facility and use case is presented in the next sub-chapters.

1) The KPI collection in Trial Controller and Facilities will be based on API and verification of their functioning will be the part of the tests.

Tests of API for initiation of the KPI collection:
- For Trial Controller it is LCM, which is initiating the setup and automatic KPI collection:
  - o Facilities will expose the interfaces or links, where data is available
  - o U-space will provide the telemetry through U-space adapter

Additionally, the KPI storage should be in place to accommodate the incoming data streams.

2) For drone – manual or automatic initiation of data collection. Data will be probably collected locally, and later downloaded to the location accessible for all interested parties. It consists of:

- Data and logs from the board of the drone
- Data from the modem/phone connected to the drone
- Data from additional sensors (IoT) connected to the drone

3) Applications running in the MEC
- KPI and logs collected in the GCS
- Other specific applications (VNF) KPIs, logs and data

Test of accurate synchronisation of all KPI data streams
- The accuracy limit should be settled and verified. Verification can be performed by triggering some events, which can be later observed in the different recorded streams and the time can be compared

Tests of KPI collection termination – at the end all processes should be properly terminated.

Tests of KPI retrieval and analysis – to see how the data can be accessed, retrieved, and analysed. After that, the modifications to the existing processes can be proposed – it may require iterative steps to make some modifications.

3.2.3.5.   5G-EVE Drop

From UC1:SC1 perspective:
1. Availability of RNIS measurements in MEC
2. Availability of data from INVOLI central server for the area of the EUR location
3. Availability of the different UAV positioning data – from U-space telemetry and from INVOLI tracker

Presence of time alignment between different modules (Trial Controller, Facility, VNFs, UAV), i.e., using GPS, NTP or other.

3.2.3.6.   5GTN Drop

Definition of release scope:
- Release Scope: KPI measurements data collection during the trials to provide them to the trial controller system of 5G!Drones project.
- Enablers: Abstraction Layer (API Gateway routing).
- Trials Controller: Web Portal, LCM, Trial Repository, Trial Enforcement, and KPI Monitoring component (KPI Endpoint REST API, Elasticsearch and Kibana).
- Facility components: KPI Measurements REST API, Qosium and Nemo tools, 5G network, network slices, and UAV systems.

Implementation process of KPI Measurements REST API is described in Table 21.

Table 21 Implementation of KPI Measurements API

No.	Implementation Process
1	KPI measurements REST API Release 1: Basic Measurement Job
2	KPI measurements REST API Release 2: DB connection
3	KPI measurements REST API Release 3: DB consumption
4	KPI measurements REST API Release 4: Multiple DBs working for KPIs measurements and measurement jobs
5	KPI measurements REST API Release 5: Routines to Get and Post Qosium Measurements
6	KPI measurements REST API Release 6: Token authentication implementation to send KPI data to the KPIC Endpoint (Frequentis)
7	KPI measurements REST API Release 7: Collection of Nemo measurements at DB server
8	KPI measurements REST API Release 8: Routines to Get and Post Nemo measurements
9	KPI measurements REST API Release 9: Post Measurements to KPI Component Endpoint (Trial Controller)
10	KPI measurements REST API Release 10: Unit and software Integration Tests for connection and consumption of DBs (Qosium & Nemo)
11	KPI measurements REST API Release 11: Unit Tests for automated routines to Get and Post KPI measurements
12	KPI measurements REST API Release 12: Software Integration Tests for automated routines to Get and Post KPI measurements

For the integration of the KPI measurement mechanism at 5GTN, it is required to complete the following Integration prerequisites and dependencies:

- Integration of the main Trial Controller components such as Web Portal, LCM, Trial Repository, and Trial Enforcement.
- Network Slice(s) creation at 5GTN facility by the Web Portal. Once the Network Slice is created, the Web Portal obtain as a response NS-ID (Network Slice ID).
- The Trial Enforcement must provide the NS-ID and KPI list (requested by the vertical user at the Web Portal) to the Abstraction Layer to create the measurement job.
- The LCM and Trial Enforcement must provide to the Abstraction Layer and thus to the KPI Measurements REST API the auth credentials (host URL, password, port, and username), interval, KPIs, and NS to create the measurement job, start the measurements, get, and post measurements.
- Established and secure communication among the Trial Enforcement and Abstraction Layer provides relevant information to start the measurement jobs driven by the user request posed at Web Portal.
- Established and secure communication among the Abstraction Layer (Aalto University) and KPI Measurements REST API (University of Oulu).
- Integration of the 5GTN parser to create and manage Network Slices at 5GTN facility as well as to provide the NS-ID to the trial controller.
- Integration of KPI Measurements REST API at 5GTN facility to publish the measurements each second to the Trial Controller.
- The API Gateway of Abstraction Layer must route Trial Enforcement related requests to the KPI measurements REST API at the 5GTN facility.
- Secure communication among the 5GTN Parser and Abstraction Layer to provide the NS-ID to the Trial Controller components.
- Secure and authorized communication from the KPI Measurements REST API to the KPI Monitoring Endpoint REST API (Frequentis) to provide the measurements collected from the 5GTN facility.

Release architecture for KPI measurements mechanism at the 5GTN in shown on Figure 22.
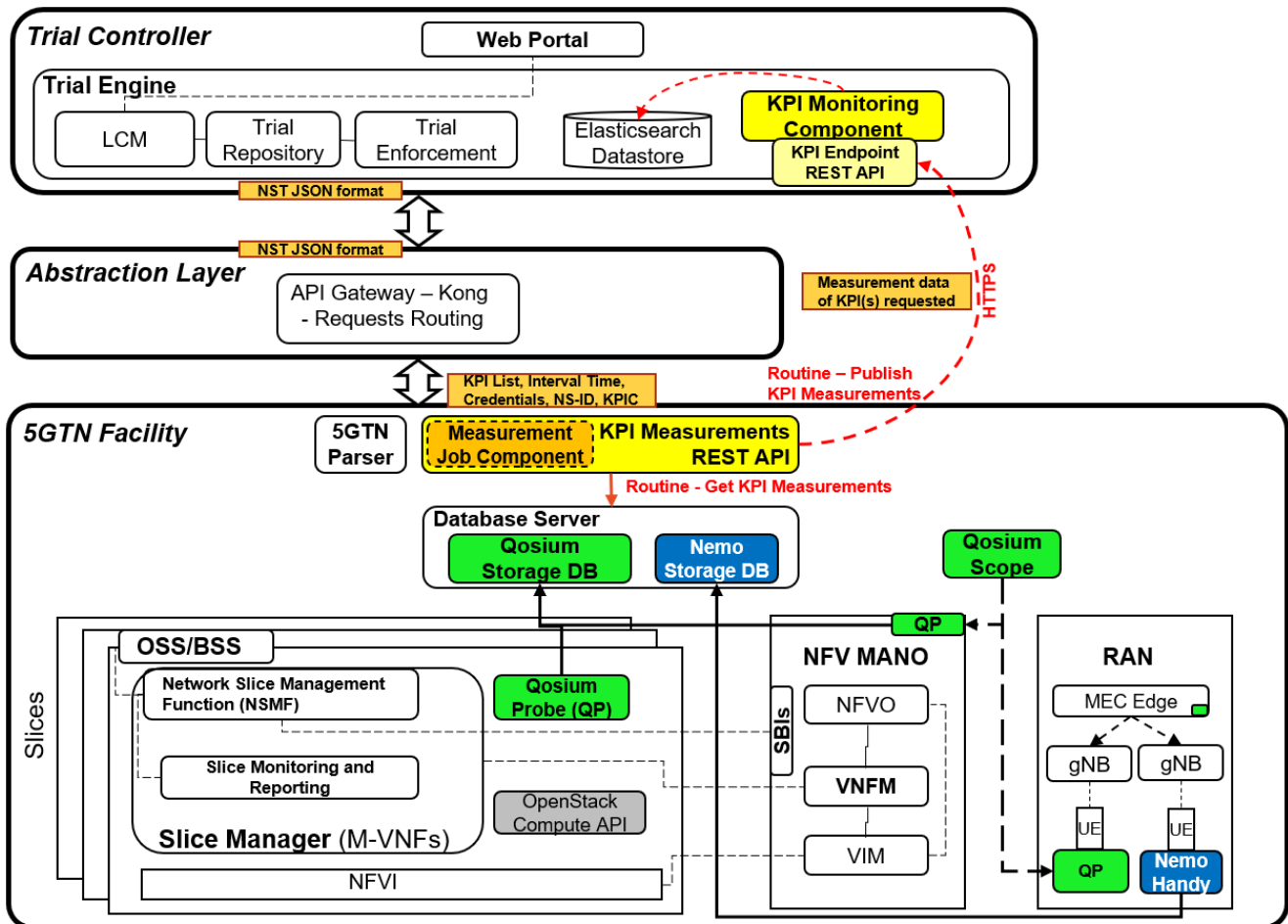
Figure 22 Architecture of KPI Measurements Mechanism Release

Integration procedures – KPI measurements mechanism set-up/deployment procedures:

- Definition of metrics according to the required KPIs, traffic type, measurement tools, measurement points, and KPIs target for each use case scenario.
- Enabler's installation and configuration for Qosium and Nemo tool on Linux or Android-based systems of the planned measurement points like the mobile phone on UAV, NSI, MEC server, 5G network core, and receiver machine of UAV's video streaming.
- Installation and configuration of tool systems like Qosium Scope, Qosium Probe, Qosium Storage, and Nemo Handy to perform the network KPIs monitoring and measurements of chosen measurement points.
- Database's implementation to store the measurements monitored and collected by the Qosium and Nemo systems at the 5GTN facility.
- Measurements starting manually by using the user interface of Qosium and Nemo just when arriving at creation requests of measurement job at the KPI Measurements REST API.
- KPI Measurements REST API Get measurements of specific KPIs and Post such measurements each second to the KPI Endpoint REST API (at the Trial Controller).
- Access and depiction of KPI measurements at dashboard at Web Portal using the KPI Measurements data stored at Elasticsearch datastore (Trial Controller).

Integration process of KPI measurements mechanism is shown in Table 22.

Table 22 Integration process of KPI measurements

No	Integration process per release
Integration of KPIM Mechanism Release 1	
1	Integration start
2	Communication testing among Abstraction Layer (Aalto University) and KPI Measurements REST API (University of Oulu)
3	Test communication among KPI Measurements REST API and DBs (Qosium & Nemo)
4	Deployment of NEMO parser to store coverage measurements at the centralized DB Server
5	Deployment of Views, Procedures, and Triggers for DBs (Qosium & Nemo)
6	Docker deployment of 1st release of KPI Measurements REST API to Get measurement in routines of 1 second
7	Docker deployment of 1st release of KPI Measurements REST API to Post measurements in routines of 1 second
8	Integration tests of the deployed docker with 1st Release of KPI Measurements REST API
9	Enable communication from KPI Measurements REST API (OU) to the KPI Component at Trial Controller (Frequentis)
10	Testing publication of measurements from KPI Measurements REST API to KPI Endpoint REST API (Frequentis)
11	Validation of KPI measurements stored on the Elasticsearch datastore of the KPI Component at Trial Controller
12	Enable Web Portal access to Kibana and Elasticsearch datastore of KPI measurements from the KPI Component at Trial Controller
13	Testing of deployed KPIM REST API (Release 1) at 5GTN during use-case scenarios
Integration of KPIM Mechanism Release 2	
1	Integration start
2	Docker deployment of the KPI Measurements REST API (Release 2) with Unit and integration Tests for DBs consumption (Qosium & Nemo)
3	Integration tests for connection and consumption of databases (Qosium and Nemo)
4	Testing of deployed KPIM REST API (release 2) at 5GTN during use-case scenarios
Integration of KPIM Mechanism Release 3	
1	Integration start
2	Docker deployment of the KPI Measurements REST API (Release 3) with Unit and Integration Tests for Get and Post routines to local KPI Monitoring Component
3	Integration of local KPI Monitoring Component Endpoint as part of the Trial Controller
4	Integration tests for Get routines of KPIs Measurements from the DB Server
5	Integration tests for Post routines of KPIs measurements towards the local KPI Monitoring Endpoint of Trial Controller
6	Testing of deployed KPIM REST API (release 3) at 5GTN during use-case scenarios

Testing scope – list and description of Use Case (UC) Scenarios (SC), and KPIs to be tested and measured during the trials at the University of Oulu:

- UC1:SC2 – 3D mapping and supporting visualization/analysis software for UTM: This use case studies the possibilities of using Virtual Reality (VR) for drone operation and real-time visualization. Among the main KPIs collected in this use-case, there are:
  - Throughput (bits/s)
  - Delay (ms)
  - Throughput (bits/s) (video streaming data rate to MEC server for processing)

- UC2:SC3 – Police, including counter-UAS: The purpose of this scenario is to demonstrate how remotely piloted UAV and video analytics can be used for police tasks, including C-UAS activities using 5G communication. Among the KPIs collected in this use-case, there are:
  - Throughput (bits/s)

- o Delay (ms)
- o RSRP (dBm)
- o RSRQ (dB)
- o RSSI (dBm)
- o SINR (dB)

- **UC3:SC1:sub-SC1 – 3D Mapping of 5G QoS:** The main purpose of this use case is to control the propagation of the 5G signal at a high height of buildings (i.e., up to 50 m). To plan base stations in 3D and for the best 5G coverage and measuring signal propagation at such heights and over, drones are a very effective tool. Among the main KPIs collected in this use-case, there are:
  - o Throughput (bits/s)
  - o Delay (ms)
  - o RSRP (dBm)
  - o RSRQ (dB)
  - o RSSI (dBm)
  - o SINR (dB)

- **UC3:SC1:sub-SC2 – Long-range power line inspection:** The purpose of this sub-scenario is to demonstrate how UAVs could be used in well-connected 5G urban areas for power line inspection and fault detection. This is an essential and time-critical service, which can benefit from the advantages of 5G networks. Among the KPIs collected in this use-case, there are:
- Delay (ms) (C2 latency)
- Throughput (bits/s) (collision avoidance LIDAR data rate)
- Delay (ms) (collision avoidance LIDAR latency)
- Throughput (bits/s) (payload video streaming data rate)
- Throughput (bits/s) (payload LIDAR streaming data rate)

- **UC3:SC1:sub-SC3 – Inspection and search & recovery operations in a large body of water:** This sub-scenario aims at demonstrating how 5G can support and greatly enhance the efficiency of daily UAV inspection operations. More specifically, the operations that this sub-scenario will demonstrate are water survey operations using a hybrid air and water drone. The goal of the operation is to monitor the state and the evolution of water bodies such as rivers, streams, and lakes. Among the main KPIs collected in this use-case, there are:
  - o Delay (ms) (C2 Latency)
  - o Throughput (bits/s) (sensor control data rate)
  - o RSRP (dBm)
  - o RSRQ (dB)
  - o RSSI (dBm)
  - o SINR (dB)

- **UC3:SC3 – Location of UE in non-GPS environments:** In this use case will be performed 3D positioning analysis and sensors measurement processing at different MEC slices before further transmission to the operating centre. Among the main KPIs collected in this use-case, there are:
  - o Throughput (bits/s)
  - o Delay (ms)
  - o Throughput (bits/s) (video streaming data rate to MEC server for processing)

More information about KPIs, methodology, and tools used for monitoring and measurements on each use-case scenario can be found in the section 3.4 of D4.3 [2].

3.2.3.7. 5GENESIS Drop

5GENESIS planned enhancements for the Release 3:

- 5G NSA: 5G NSA will be further complimented with 5G CPEs capable of providing WIFI6 and other network interfaces
- 5G SA: 5G SA will be further complimented with 5G CPEs capable of providing WIFI6 and other network interfaces

5GENESIS prerequisites for the Release 3:
- EPC (Evolved Packet Core): Both EPCs are available. LBO (Local Breakout) capability is supported by Athonet EPC.
- 5G-Core: Amarisoft 5G-Core for SA deployment is available
- 5G-NR: Both 5G-NRs are available for deployment
- MEC: MEC deployment with Athonet EPC is available
- Slice Manager: Slice manager added CI/CD pipeline
- Web Portal 1: The experimenter is able to design via Web Portal 1 the flight plan and upon receiving approval the system forwards him to the Facility web portal.
- Facility Web Portal: it will offer test-cases that include the flight plan as part of the experiment.
- Trial Validator and translator: The full functionality will be ready.
- Life Cycle Manager: The full functionality will be available.

3.2.4. Release 4: Use case release (M36-M37, May-June 2022)

3.2.4.1. Introduction

Prior to this release step, all modules of the Trial Controller and facility enablers should be integrated and functional. In this release, final components needed to run the use case scenarios in full scope will be released and tested. These components are primarily the vertical service-level enablers reported and developed within work package WP3, task T3.4. Although most of the vertical service-level enablers are planned to be released before release 4, it is expected that some minor works and end-to-end functional tests remain.

3.2.4.2. Release scope

The scope of this release is to confirm full ability to perform all trial scenarios:
1. Ensure all service level enablers are developed
2. Ensure all use cases are defined in their final form
3. Ensure all use case components are tested

The following enablers are planned to be released after release 3 by the consortium partners for the use case scenarios, deployed and tested by release 4:
- Geo-location of a UAV based on signal strength from 1 base station (ORA)
- 4K video streaming system from drone (CAF)
- CAFA video analyser VideoLyzer (CAF)
- 3D map for visualising QoS of 5G CAFA Analyzer (CAF)
- CAFA field GIS C2 system for police operations and drone logistics planning (CAF)

Additionally, the functional testing of all previously released enablers that have had updates after release 3 will be covered in release 4.

3.2.4.3. Release implementation schedule

The implementation of the release is planned for May-June 2022. Preparations for the release will start after release 2. Detailed planning for Release 4 is provided in Figure 23.

	10'2021	11'2021	12'2021	01'2022	02'2022	03'2022	04'2022	05'2022	06'2022
Release 3 Plan : Use case release	M28	M29	M30	M31	M32	M33	M34	M35	M36
1) Collect data on development status and plans of use cases									
2) Schedule and plan end-to-end testing									
3) Iprepare test specifications and documentations									
4) Conduct tests									
5) Analyse results, prepare reports									

Figure 23 Schedule for Release 4

3.2.4.4. Testing scope

Specific vertical service level enablers will be functionally tested by the owners of the enablers before the end of release 4. Final end-to-end testing can be done in conjunction with the use case trials. If not already done so before the final trials, the following enablers will be displayed to function as mentioned in Table 23.

Table 23 List of enablers

Owner	Enabler
AU	AU IoT HAT
AU	AU IoT HAT driver
AU	AU IoT-as-a-Service (IoTaaS) platform
AU	AU Virtual flight controller
ALE	Alerion's GCS
ALE	Data processing
ALE	Sensor data streaming
ALE	Hydradrone
DRR, FRQ	Telemetry
DRR, FRQ	Drone Flight Plan
DRR, FRQ	Geozone information (Mission Prioritization) support
HEP	Hepta's Data Cloud
HEP	Hepta's GCS
HEP	Hepta's drone with tether
HEP	Payload – Lidar for 3D mapping
HEP	Interface with AP
HEP	Sensor data streaming
HEP	Data processing
INV	INVOLI Central Server
INV	INVOLI's KIVU/LEMAN tracker
INV	K-1090 receiver
NOK	Nokia Drone
NOK	UWB based drone positioning system
NOK	5G Modem
NOK	5G smartphone & camera holder
NOK	DJI Matrice 600 payload adapter
ORA	Geo-location of an UAV based on signal strength from 1 BS
ORA	Attestation
ORA	Optimal trajectory of UAV Base Station
UO	5G Smartphone
UO	virtual flight controller
UO	Positioning analysis application
UO	Data collection and mapping
UMS	UL-ACE
UMS	UL-CCP
UMS	UMS Video Analysis
UMS	UMS Wi-Fi Access Point

CAF	UgCS C2 cloud-native application enhanced by CAFA Tech (CAFA CUP)
CAF	4K Video streaming system from drone
CAF	CAFA Video analyser (VideoLyzer)
CAF	3D map for visualizing QoS of 5G (CAFA Analyzer)
CAF	CAFA Field GIS C2 system for Police operations and Drone Logistics planning
CAF	DJI Mavic drone
CAF	CAFA cellular drone (Pixhawk platform based) supporting onboard 5G commands via 5G module

All the use case scenarios defined within the project are to be trialed by the end of release 4. These scenarios are defined in deliverable D1.1 Use Case Specifications and Requirements and updated in deliverable D4.3 Trial Plan [2].

From the facilities' perspective all components are planned to be released by release 3 the latest, no additional components are released here.

4. CONCLUSION

This document develops the concepts and approaches for integration of 5G!Drones project components as introduced in deliverable D4.1. It contains the descriptions of integration and testing procedures as well as definitions for testing and deployment of the environment. All this work was developed in close cooperation with WP2 and WP3 partners.

This document complements other project's architectural deliverables D1.6, D2.2 and D2.4 in the way that it contains complete inventory of interfaces to be integrated and all components that need to be implemented to fulfil all use cases' requirements. It provides a clear end to end view of the integration architectures of all use cases.

From the testing perspective it provides the full testing approach, describing what and how should be tested, providing test case descriptor templates, detailed release scope definitions and selections of tools to be used.

This document contains also complete implementation/delivery plans of all enablers used in the project. The release approach as outlined in D4.1 is fully developed in this document. This document defines purpose of each release, covers the detailed scope and implementation plans. The four defined releases are:
- Integration Validation Release, Release 1
- Trial Controller Release, Release 2
- KPI Release, Release 3
- Use Cases Release, Release 4

The sequence, timeframes and scope of mentioned releases were carefully aligned with development and delivery plans of WP2 and WP3.

Finally, this document also contains first reports from integration tests – in this case the report from trial and tests in Eurecom in June 2021.

References

[1] 5G!Drones: "D4.1 – Integration Plan", H2020 5G!Drones project, 2020.
[2] 5G!Drones: "D4.3 – Trial plan", H2020 5G!Drones project, 2021.
[3] 5G!Drones: "Integration Tests Specifications for Release 1", H2020 5G!Drones project, 2021.