"5G for Drone-based Vertical Applications"

D4.1 – Integration Plan

Document ID:	**D4.1**
Deliverable Title:	**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:	**36 Months**
Contractual Delivery Date:	**M07**
Actual Delivery Date:	**February 8th, 2020**
Dissemination Level:	**Public (PU)**
Contributing Beneficiaries:	**UO, THA, ALE, INV, HEP, NCSRD, AU, COS, AIR, UMS, INF, NOK, RXB, EUR, DRR, CAF, FRQ, OPL, MOE, ORA**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857031.

DocumentID: D4.1
Version: V1
VersionDate: 08.02.2020
Authors: Pawel Korzec, Piotr Dybiec (DRR)

Security: Public (PU)

Approvals

	Name	Organization	Date
Coordinator	Jussi Haapola	UO	8.2.2020
Technical Committee	Pascal Bisson	THA	7.2.2020
Management Committee	Project Management Team Additional Reviewers	UO, THA, AU, AIR, UMS, FRQ CAF, COS	6.2.2020

Document History

Version	Contribution	Authors	Date
V0.10	**Initial version of integration plan**	**Paweł Korzec, Piotr Dybiec (DRR)**	**10.01.2020**
V0.11	**Draft version v1.1**	**Piotr Dybiec (DRR)**	**21.01.2020**
V0.20	**Version after reviewal**	**Piotr Dybiec (DRR), Paweł Korzec (DRR), Thomas Lutz (FRQ)** **Tomas Gareau (UMS)** **Adlen Ksentini (EUR)**	**31.01.2020**
V0.22	**Input from Demokritos and Pascal**	**Piotr Dybiec (DRR), Paweł Korzec (DRR), Thomas Lutz (FRQ)** **Tomas Gareau (UMS)** **Adlen Ksentini (EUR)** **Harilaos Koumaras (DEM)**	**3.2.2020**
V0.24	**Update from CAFA Tech, Oulu University, INFOLYSiS, UNMANNED LIFE, COSMOTE**	**Piotr Dybiec (DRR)** **Tanel Jarvet (CAF)** **Farid Benbadis** **Vaios Koumaras (INF)** **Nemish Mehta (UMS)**	**4.02.2020**

		Fotini Setaki (COS)	
V0.25	**Final comments from AIR, conclusions adjustment**	**Piotr Dybiec (DRR)** **Arthur Lallet (AIR)**	**5.02.2020**
V1	**Remarks and adjustments based on Involi comments and editorial corrections, editorial and table adjustments from Fotini Setaki, final editorial checking**	**Piotr Dybiec (DRR)** **Paweł Montowtt (INV)** **Fotini Setaki (COS)**	**6.02.2020**

Executive Summary

This deliverable provides an initial Integration Plan for the 5G!Drones project and is the product of Task 4.1" Software integration and 5G!Drones architecture validation". The Integration Plan, known also as deliverable D4.1, contains the integration methodology, key definitions and processes that will govern the integration activities. During the course of T4.1 task, it is foreseen that the plan shall be continuously enriched and optimized, and as a result this document will be superseded by deliverable D4.2 by M18 containing the updated final integration plan.

Main goal and scope of the task T4.1, following [2], is to deliver a fully-fledged trial system including all the necessary components at the UAV service and the infrastructure levels for the execution of the selected trials over 5G facilities. As part of this task, the 5G!Drones trial controller and 5G!Drones enablers, including UAV-service-related software and hardware shall be integrated. Given the size and complexity of the project, with lots of heterogeneous components that are to be implemented and integrated with existing ones in a manner compatible with the trial facilities, a detailed integration plan needs to be devised early in the course of the task to drive all integration activities in the project. This plan is expected to define the integration and testing procedures and environment which will manage how the software and/or hardware modules that will be progressively delivered shall be incrementally deployed and tested in the trial facilities. Thus, the main purpose of the Integration Plan (D4.1) is to describe actions, dependencies and responsible parties, as well as time and other necessary resources to perform integration activities.

The integration phase targets a fully functional 5G!Drones solution able to support the execution of desired UAV trial scenarios. As the 5G!Drones 5G platforms will grow and evolve, incrementally delivering required functionalities, integration and validation of project's resulting components must accord to this mode. The proposed method thus implements cyclic instantiation of functional packages (releases) of deliverables (solution components) according to required use case scenarios, facility capabilities and provided components' functionalities to ensure effective tests of planned trial use cases. The document starting from the identification of the components and interfaces that are in target of the integration, addresses the implications of the different level of maturity among the involved 5G!Drones facilities and concludes with the integration methodology addressing the Design-Build-Release Cycles, key release steps and actors.

It is proposed that the whole integration process shall consist of few iterations per facility site before reaching the required functionality. In essence, three integration iterations are planned

and one trial period towards the end of the project to support the final use cases executions and assessment. As an output of each iteration, feedback and validations reports shall be provided to WPs developing particular components and the project's team, so that to validate and further adjust the developments and the trial scenarios.

Description of the specific tools utilized, as well as detailed reference on target interfaces per release as well as testing infrastructure and procedures per facility and use case, shall also be presented in D4.2, to reflect the practical experience and assessment conclusions of the first integration cycle.

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 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
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
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 reference information regarding actions and steps with necessary resources that need to be performed to perform solution integration activities as required by project's task T4.1.

According to project's Grant Agreement [2], following actions are considered to take place within this task:
"[…]
- Incremental deployment and unit tests in a laboratory environment.
- 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. […]"

To address those needs, detailed planning, tools and methodologies are described hereafter.

1.2. Relation to other project work

The activities presented in this deliverable are complementary to the work done in WP2 and WP3 and cover the preliminary plans carried out to integrate the 5G!Drones framework for the validation and execution of the use cases and trials. Integration phase depends on deliverables from other work-streams (WP1, WP2, WP3), mainly: D1.1, D1.2, D1.3, D1.5, D2.1, D3.1 and D3.2. They are mentioned here to indicate dependencies, but detailed planning of their development is out of scope of the Integration Plan and as such should be delivered by respective responsible work-streams. On the other side it is assumed that outcomes from T4.1 shall provide continuous feedback to architecture and solution design streams within WP2 and WP3.

1.3. Structure of the document

This document is structured in four major chapters

Chapter 1, this one, presents the introduction, objectives and structure of the deliverable.

Chapter 2 contains description of 5G!Drones solution architecture, use cases description and reference information regarding system's key information and processes flows within the solution.

Chapter 3 focuses on description of release-based integration approach. Section 3.5 contains example, simple description of release prepared based on the process defined in section 3.4 to let readers better understand proposed release approach.

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 on the integration steps and methodology in order to be timely prepared for the actions necessary.
- 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 funding EC organization as committed by the Grant Agreement.

2. 5G!DRONES SOLUTION OVERVIEW, KEY FEATURES AND PROCESSES

The reference 5G!Drones description of solution, from integration perspective, consist of:

- Overall high-level architecture (provided in [1])
- Trial controller architecture (to be defined and implemented within WP2)
- Enablers for 5G Platform: 5G infrastructure level components (to be defined and implemented within WP3)
- Enablers for Drone coordination: UAV use case service components (as well to be defined and implemented within WP3)
- Use case scenarios enablement analysis providing information on resources and capabilities necessary to execute particular trial use case scenarios
- Information and data flow providing information on use case scenario execution actions

2.1. Overall high-level architecture

5G!Drones high-level architecture considers the design and implementation of experimentation tools and enablers that will facilitate the automation and execution of UAV/Drones-vertical experiments on top of 5G platforms that participate in the project.

Initial overall architecture of 5G!Drones is defined in D1.3 [1] and is also presented in the following figure (Figure 1).

5G!Drones architecture includes the design and development of a web portal and an open-API, which will be used by vertical experimenters in order to perform in an automatic way the experiments and trials of the use cases that have been described in D1.1 [3].

Core components of the 5G!Drones architecture involves the Trial controller module, which will be developed within WP2 activities. More specifically, the modules to be developed by WP2 are:

- **Web Portal**
- **Open Api** (Vertical Expansion of the facility).
- **Trial controller** (Trial scenario execution engine, Trial Architecture Management Plane, KPI assessment and data gathering)

- **U-Space Adapter** ("connector" module between trial controller and U-space systems/enablers)

Respectively, the 5G and U-space "enablers" will be developed as part of WP3, include two sets of components:

- 5G Platforms & UAV enablers
- Infrastructure federation (horizontal expansion of the facility).

Further details on the internal interfaces of the core components will be provided in the respective WP2 and WP3 deliverables.

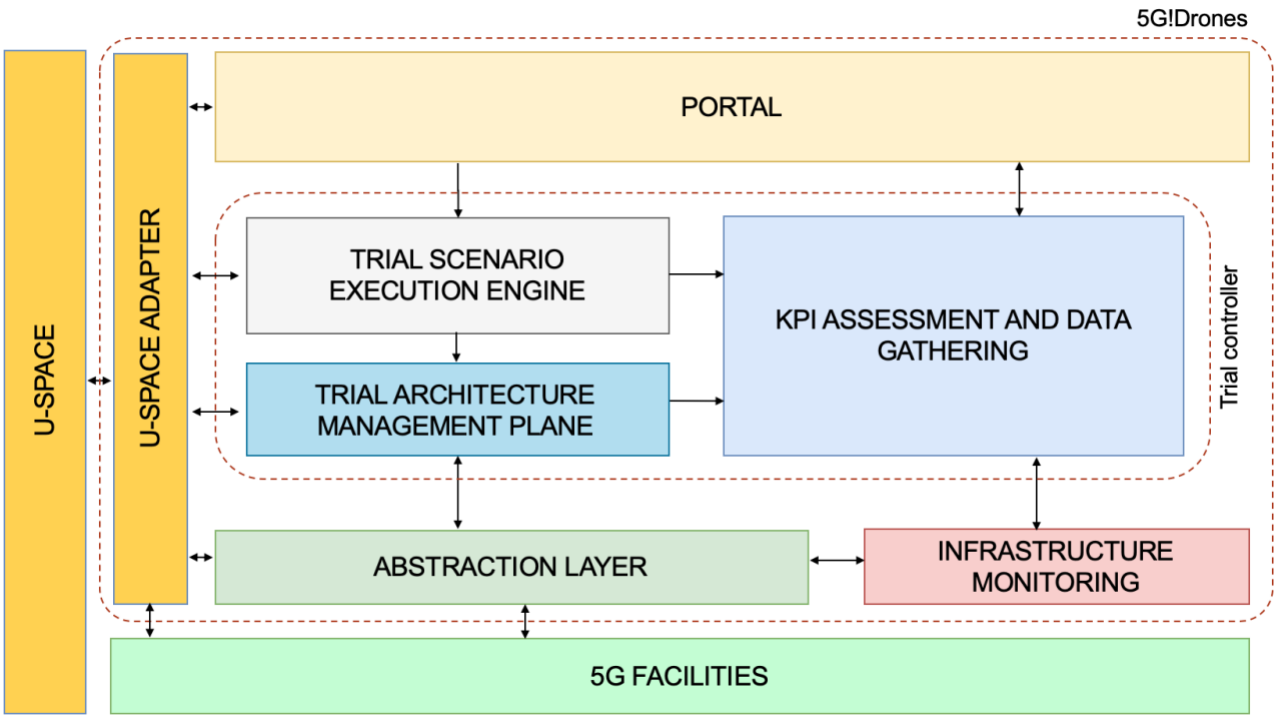


Figure 1 Overall 5G!Drones solution conceptual architecture diagram

In terms of integration planning, it should be clarified that 5G!Drones activities are supported by both ICT-17 and non-ICT-17 5G platforms. This differentiation means that different level of maturity exists among the 5G platforms that are going to support the 5G!Drones trials, meaning that ICT-17 facilities, and especially 5GENESIS, provide a vertical-agnostic experimentation layer, with components that plan and execute the trial on top of the platform. Therefore, the integration planning for these platforms will include a complimentary process of extending and adding any UAV-specific functionality and/or component that may be missing and is needed for the execution of the trial on top of the specific platform.

2.2. 5G!Drones Trial controller

Underlying architecture of Trial Controller is in scope of WP2 as well as the underlying components. Its integration is in scope of WP2 and it will be delivered to WP4 as a coarse-grained component to be integrated to others through external APIs it offers.

At this level, integration activities will rely on provided APIs as depicted on the above diagram: internal API between trial controller modules and external APIs (southbound and northbound) toward external systems (U-space and 5G facilities).

Part of T4.1 work is to plan and manage incremental deployment and tests of subsequent trial controller releases in the trial facilities, considering their different maturity levels.

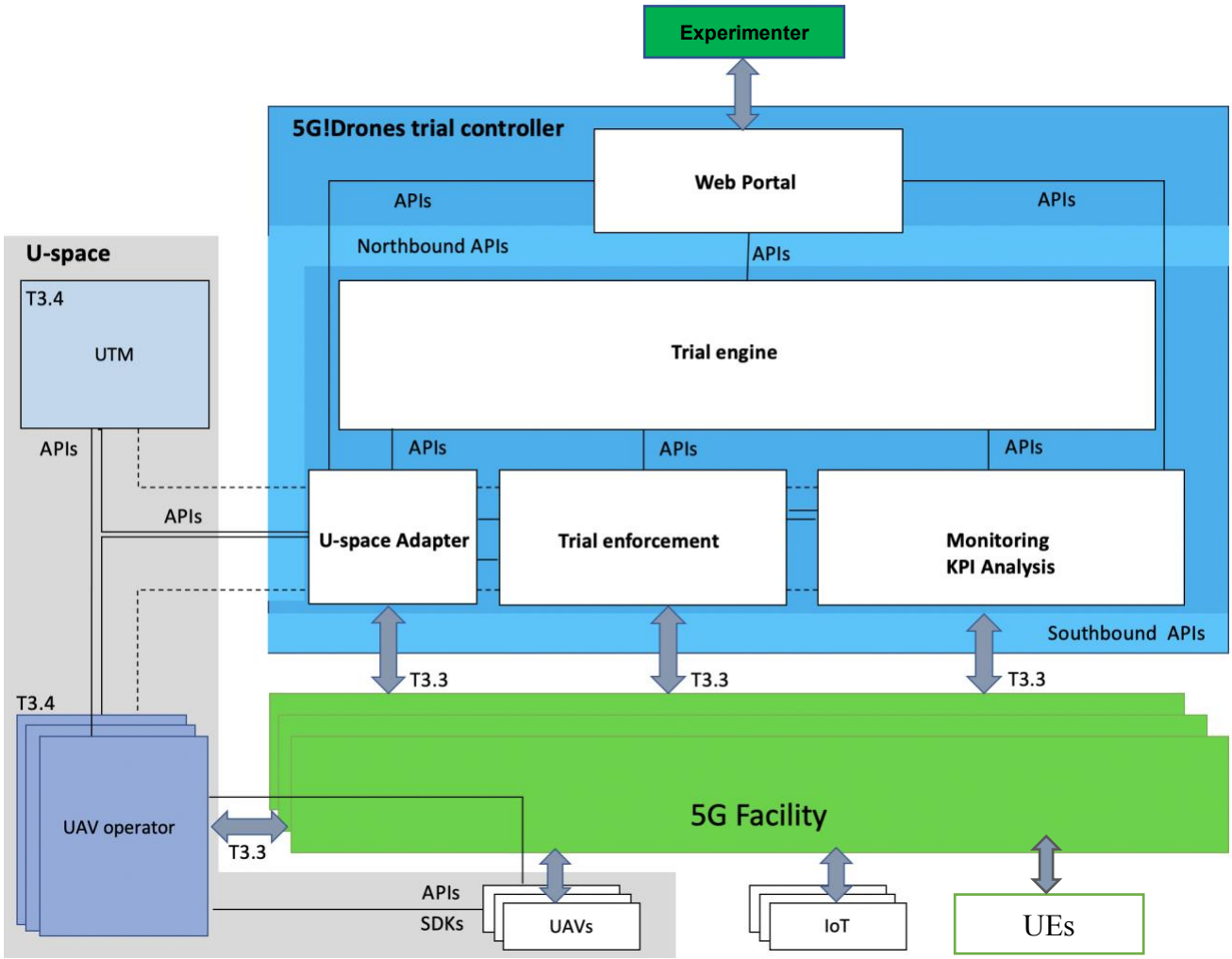


Figure 2 Trial controller architecture and APIs diagram

2.3. Enablers for 5G Platform

From the perspective of integration between trial controller and 5G facilities, two main deliverables play crucial role: D3.1 Report on infrastructure-level enablers for 5G !Drones and D3.3 5G!Drones Enablers Software Suite, which will provide the first software releases of the specific modules. This part includes abstraction layer for interfaces and communication within the Trial Controller and communication to the 5G facilities. As an ultimate goal the unified interfaces over heterogeneous facilities (abstraction layer enabler) should be here developed.

2.4. Enablers for UAV/Drone coordination

This part includes definition of the abstraction layer for interfaces and communication within the U-space as general, which include U-space adapter definition, Operator communication and Drone C2/telemetry link.

For the purpose of integration between trial controller and U-space, U-space adapter concept is introduced. Specific U-space enablers, which need to be used/integrated for the purpose of trial use cases, will be described in D3.2 Report on vertical service-level enablers for 5G!Drones and implemented within the D3.4 UAV use case service components.

For the purpose of 5G!Drones project, the U-space abstraction layer, architecture and interfaces based on SESAR JU [1] projects (eg. Gulf Of Finland, GOF) may be used, as being most accurate and mature. Based on SWIM principles information exchange services are introduced to achieve interoperability.

Information exchange services facilitate data exchange for information provided and consumed by U-space and 5G services. They are described using a layered approach to decouple logical, technical and runtime aspects. Standardized document templates will be provided for formal documentation.

A microservice-oriented data exchange layer will provide standard protocols to connect various UTM services from different UTM service providers to and from 5G facilities. The value of standard protocols will be underlined by shuffling the capabilities of service provision during the demonstration.

The data exchange layer will use several data sources and harmonize them for further provision to the various services and stakeholders.

The following diagram (Figure 3) comprises multiple dimensions such as deployment aspects, actors, dataflows, services & capabilities.

[1] The Single European Sky Air Traffic Management (ATM) Research (SESAR) project was set up in 2004 as the technological pillar of the Single European Sky initiative. The SESAR Joint Undertaking (SESAR JU) was established in 2007 as a public-private partnership to harness the research and innovation expertise and resources of the entire ATM community. Founded by the European Union and Eurocontrol, the SESAR JU has 19 members, who together with their partners and affiliate associations represent over 100 companies working in Europe and beyond.

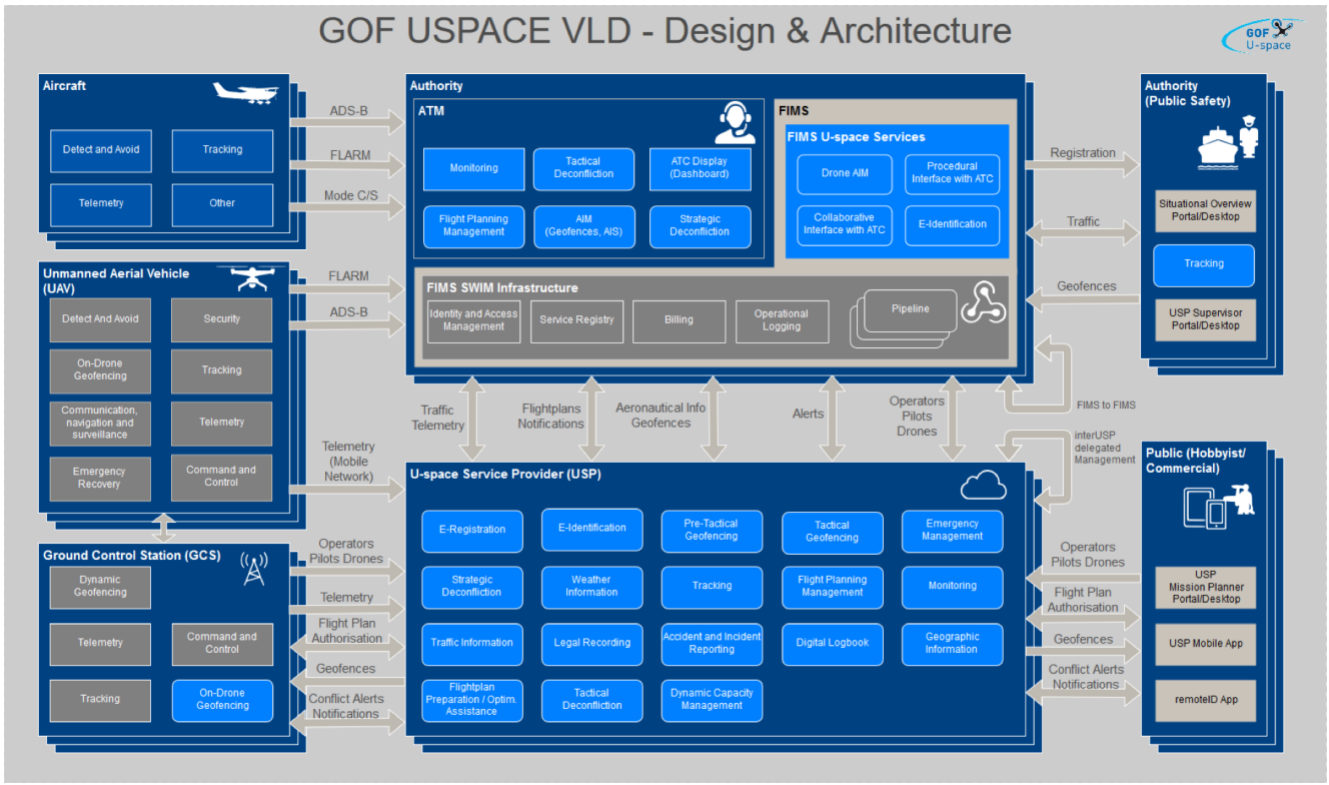


Figure 3 GOF USPACE VLD – Design & Architecture diagram

Following services will be supported:

- Traffic / Telemetry
- Flight plans / Notifications
- Aeronautical Info / Geofences
- Alerts
- Registration
- Ground Control Station Integration
- 5G KPI for mapping 4D RAN coverage in the airspace

The above services were identified based on the other trials, demonstrator expectation, consortium member experience and U-space services foreseen to be demonstrated.

The Information Exchange Services will be described in a technology agnostic way and will follow recommendations and standards, especially from communication security perspective, defined by SWIM framework as depicted on Figure 4 (source: https://www.sesarju.eu/newsroom/all-news/enabling-secure-information-exchange-atm).

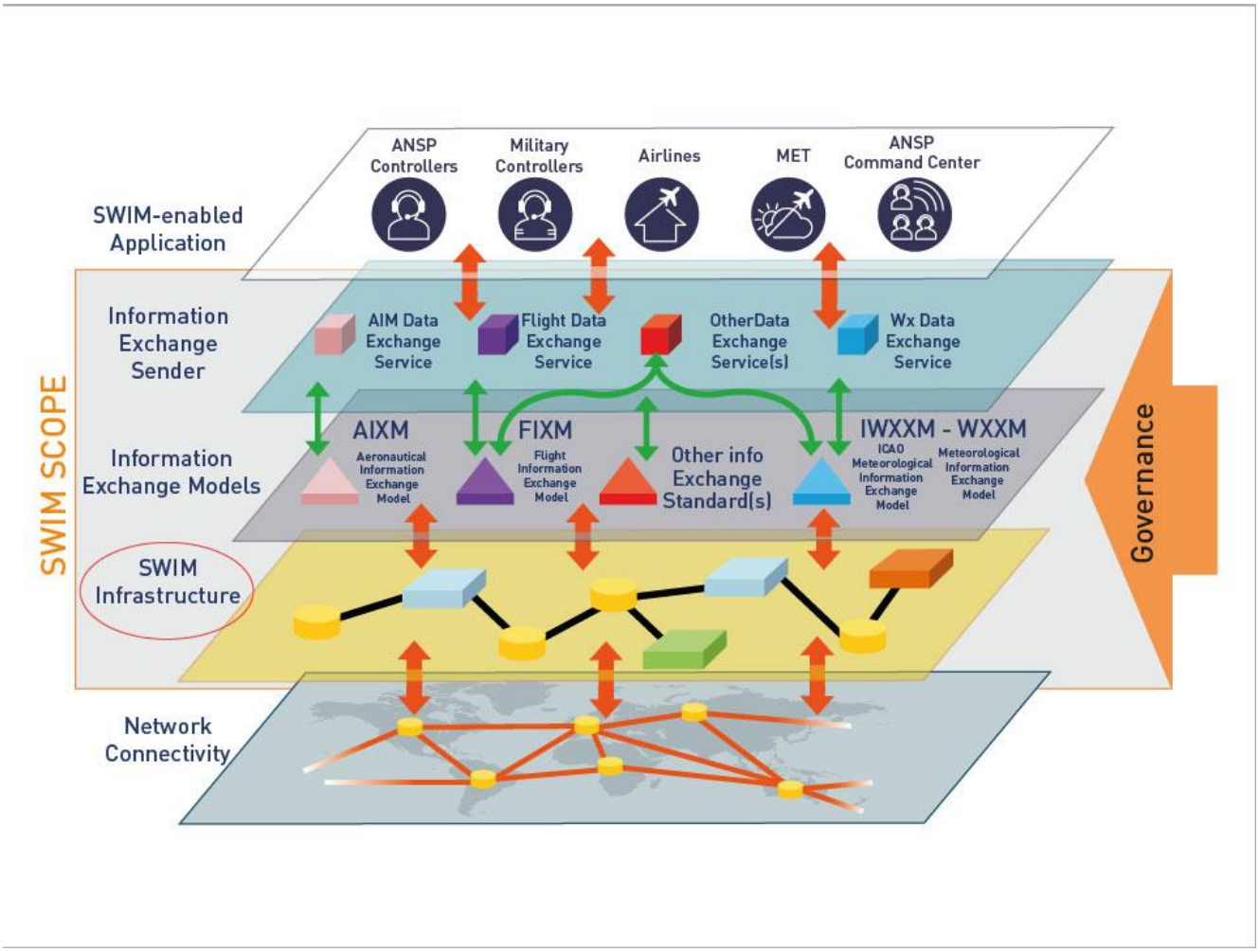


Figure 4 SWIM model of information exchange

From the integration perspective for U-Space, same rules/approach will be applied, as it is described for integration of 5G enablers.

2.5. Use Case Scenarios enablement

The ultimate goal of the Use Case Scenarios enablement is to perform correct match of the UC scenarios requirements with 5G facilities features, UAV enablers and KPI requirements. This information will be provided within D1.5 Description of the 5G trial facilities and use case mapping document by M12.

Initial requirements on use case enablement and required inventory can be derived from enablement's matrixes (Table 1).

No.	Use Case Scenarios	Owners	Facility	Use Case Scenario Enablements - UAV Focused																			
				Hardware on UAV																			
				Video streaming				5G connectivity		VR	Sensors												
				HD video camera	4K video camera	Thermal camera	Video streaming module	5G UE	gNodeB	VR headset	Network measurement sensor	Temperature sensor	Pressure sensor	CO^2 sensor	Spider camera	IoT camera	LIDAR	Bathymetric sensor	Water quality sensor	Humidity sensor	Gas sensor	U-space adapter	Positioning sensor
1	UC1:SC1	INV	5GEVE (EUR)		x		x	x		x												x	
2	UC1:SC2	UO	5GTN (UO)					x		x	x	x	x	x	x							x	
3	UC1:SC3	CAF	X-Network (AU)	x			x	x														x	
4	UC2:SC1	AIR	5GEVE (EUR)	x			x	x		x						x						x	
5	UC2:SC2	UMS	5GEVE (EUR)	x		x	x	x	x													x	
6	UC2:SC3	CAF	5GTN (UO)		x		x	x								x						x	
7	UC3:SC1:SSC1	CAF	5GTN (UO)					x								x						x	
8	UC3:SC1:SSC2	HEP	5GTN (UO)		x		x	x									x					x	
9	UC3:SC1:SSC3	ALE	5GTN (UO)					x									x	x	x			x	
10	UC3:SC2	AU	X-Network (AU)	x			x	x				x				x				x	x	x	
11	UC3:SC3	NOK	5GTN (UO)																			x	x
12	UC4:SC1	DEM	5GENESIS (DEM)	x			x	x	x													x	

No.	Use Case Scenarios	Owners	Facility	Applications on UAV					Applications on MEC/Edge											External Sensors + Items	
				C2 platform onboard UAV																	
				UL-ACE	CAF uGCS	Hepta uGCS	Alerion uGCS	Aalto uGCS	UL-CCP	CAFA CUP	Hepta UAS	Alerion UAS	Aalto UAS	Video processing application	Mission critical communications application	Air traffic awareness platform	DLN delivery software	3D mapping software	Alerion post-processing application	Delivery box IoT device	Delivery package
1	UC1:SC1	INV	5GEVE (EUR)	x	x				x	x				x		x					
2	UC1:SC2	UO	5GTN (UO)	x					x					x							
3	UC1:SC3	CAF	X-Network (AU)														x			x	x
4	UC2:SC1	AIR	5GEVE (EUR)		x					x				x	x						
5	UC2:SC2	UMS	5GEVE (EUR)	x					x					x	x						
6	UC2:SC3	CAF	5GTN (UO)		x					x				x							
7	UC3:SC1:SSC1	CAF	5GTN (UO)		x					x								x			
8	UC3:SC1:SSC2	HEP	5GTN (UO)			x					x										
9	UC3:SC1:SSC3	ALE	5GTN (UO)				x					x							x		
10	UC3:SC2	AU	X-Network (AU)					x					x								
11	UC3:SC3	NOK	5GTN (UO)		x					x											
12	UC4:SC1	DEM	5GENESIS (DEM)	x	x				x	x								x			

No.	Use Case Scenarios	Owners	Facility	Use Case Scenario Enablements - 5G Focused															
				Slices			MEC/Edge	Latency				Data rate				Error rate			
				uRLLC	eMBB	mMTC		C2	Telemetry	Application data	End-user data	C2	Telemetry	Application data	End-user data	C2	Telemetry	Application data	End-user data
1	UC1:SC1	INV	5GEVE (EUR)	x	x		x	<= 10 ms	<= 100 ms	<= 20 ms		256 kbps	800 kbps	30 Mbps		$< 10^{-5}$	$< 10^{-3}$	Typical values for HD or 4K video	
2	UC1:SC2	UO	5GTN (UO)				x	10 ms	<= 100 ms			60-100 kbps		50 Mbps					
3	UC1:SC3	CAF	X-Network (AU)	x	x	x	x	50 ms		50 ms		100 kbps		30 Mbps	0.5 Mbps	$< 10^{-3}$			
4	UC2:SC1	AIR	5GEVE (EUR)	x	x		x	<= 10 ms	<= 100 ms	~200 ms	~50 ms	256 kbps	800 kbps	50 Mbps	10 Mbps	$< 10^{-5}$	$< 10^{-3}$	Typical values in LTE	Typical values in LTE
5	UC2:SC2	UMS	5GEVE (EUR)	x	x		x	<= 10 ms	<= 100 ms	~200 ms	~50 ms	256 kbps	800 kbps	50 Mbps	10 Mbps	$< 10^{-5}$	$< 10^{-3}$	Typical values in LTE	Typical values in LTE
6	UC2:SC3	CAF	5GTN (UO)	x	x	x	x	50 ms		50 ms		100 kbps		30 Mbps	0.5 Mbps	$< 10^{-3}$			
7	UC3:SC1:SSC1	CAF	5GTN (UO)	x			x	50 ms		50 ms		100 kbps		30 Mbps	0.5 Mbps	$< 10^{-3}$			
8	UC3:SC1:SSC2	HEP	5GTN (UO)	x	x		x	<= 10 ms	<= 50 ms	<= 10 ms	~50 ms	256 kbps	800 kbps	120 Mbps	120 Mbps	$< 10^{-5}$	$< 10^{-3}$	$< 10^{-3}$	Typical values in LTE
9	UC3:SC1:SSC3	ALE	5GTN (UO)	x	x		x	<= 10 ms	<= 100 ms	~200 ms		256 kbps	256 kbps	50 Mbps		$< 10^{-5}$	$< 10^{-3}$	Typical values in LTE	
10	UC3:SC2	AU	X-Network (AU)	x	x	x	x	10 ms				60-100 kbps		50 Mbps		$< 10^{-3}$			
11	UC3:SC3	NOK	5GTN (UO)	x	x		x	10 ms		Typical values in LTE		60-100 kbps		50 Mbps					
12	UC4:SC1	DEM	5GENESIS (DEM)	x	x		x	10 ms	<= 100 ms	Typical values in LTE		60-100 kbps		50 Mbps	0.5 Mbps	$< 10^{-3}$			

Table 1 Use Case Enablement Mapping

The tables above provide information how each use case is decomposed into necessary enablers on both 5G and UAV side. Based on this information, each use case scenario can be depicted as unique to 5G facility/use case scenario set of necessary enablers.

This information will provide guidelines which components/functionalities needs to be integrated to support execution of the chosen trial scenario.

More information on planned trial use cases can be found also in [3].

2.6. Overall, End to end process and information workflow

The ultimate goal of the task dedicated to identification of process and Information flow is to create:

- Actors definition
- Service provisioning workflow
- Equipment inventory: Ensure required resources to execute exercise like: type of the drone, guaranteed End to end bandwidth between modules and other resources

- Business related aspects like plan the time to provision, cost estimation, maintenance of the website due to periodicity, etc

Above information allows experimenter to prepare and secure necessary resources like appropriate drones, service provisioning configurations, etc. On top of that information experimenter can plan exact steps that he/she would need to perform to execute the experiment.

In turn this step by step description of actions provides baseline for preparation of UAT tests: description of test scenarios, with exact steps, actions and expected results.

3. INTEGRATION PLAN

3.1. Process for adapting the 5G!Drones Framework to the 5G facilities

This section presents the process we followed for adapting the 5G!Drones Framework to the use cases. As 5G!Drones is 3-years project, the plan is to perform a number of iterations with specific objectives in each of them. Figure 5 graphically depicts the iterations timeline.

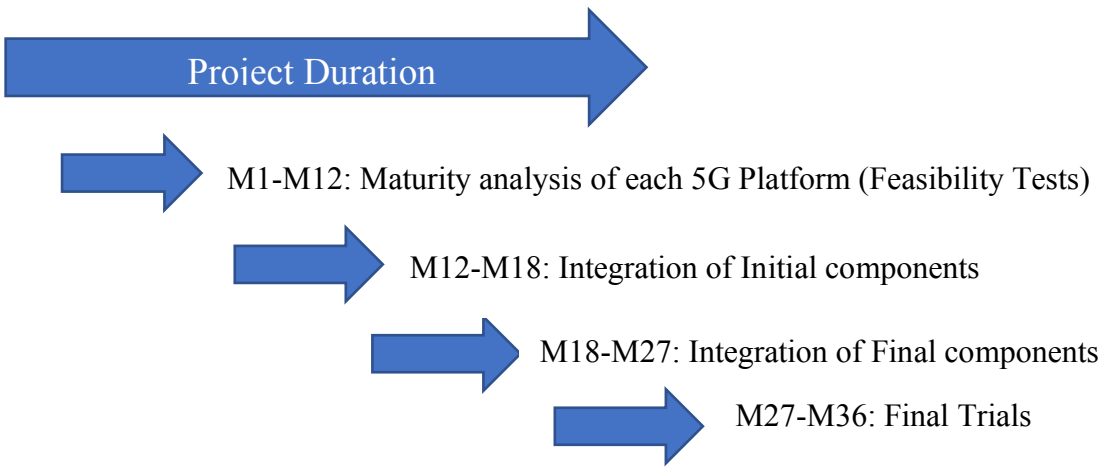


Figure 5 Iterations of integration of 5G!Drones tools

In essence, three integration iterations are planned and one trial period at the end, with the final goal of having all tools and internal components of 5G!Drones integrated into the 5G platforms according to the needs and specification of each use case. As the internal components are going to be developed after having an initial version of the architecture, it is assumed to work in parallel with the initial integrations of the tools.

Integration approach and respective plan presented on following pages reflects incremental development approach of main projects deliverables (trial controller, infrastructure enablers suite and UAV service components integrated in target facilities) and is aligned with main milestone dependencies – expecting that initial versions would be integrated by **M20 (as per MS3).**

Incremental development approach means, that integration activities will be performed with partially functional components delivered incrementally within defined releases. The main types of activities that shall be performed in each integration cycle are:

- **Component/integration tests**: the purpose is to validate the interface implementation but also to provide feedback to WP regarding potential discrepancies between designed/assumed interface and real UAV behavior (performance, capability, etc.).
- **Acceptance tests:** these tests will be performed for each 5G facility. It is assumed (and reflected in integration plan's Gantt chart –see Figure 7), that at least one end to end session for acceptance tests need to be planned. The initial, proposed dates are included in the Gantt's chart. Component/integration tests iterations will be planned according to the appropriate release and UAV availability dates. As those tests are end to end, they will cover both required "enablers": 5G infrastructure based and UAV based.

The functional scope of releases should be defined based on requirements derived from trial use case scenarios analysis selected as representative for the acceptance tests.

Below diagram illustrates this approach:

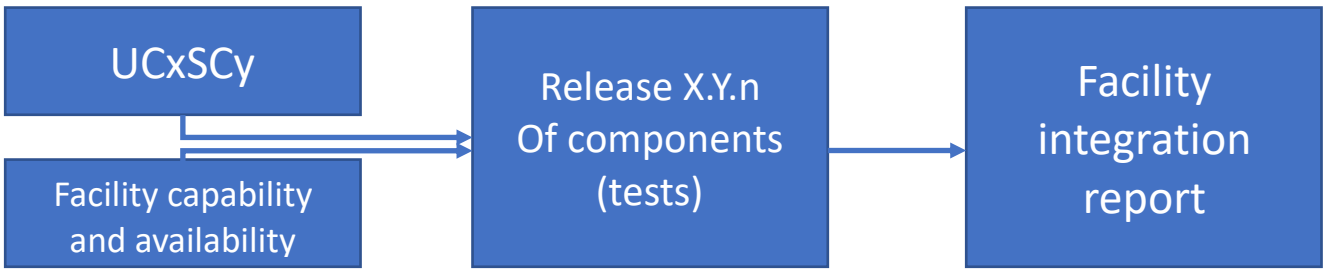


Figure 6 Incremental releases based on Use Case Scenarios and facility capabilities

On above diagram Release X.Y box represents deliverable package to be integrated and tested on selected facility. Contents of the package is determined by the requirements coming from selected use case (UC) scenario (SC) and respective facility requirements. Integration tests for Release X.Y are executed according to release cycle described in chapter 3.4. After each completed release cycle, facility integration report should be provided.

Dates and contents of the releases should be aligned with WP2 and WP3. As the schedule is tight, releasing deliverable packages is expected to start as soon as possible.

On the Gantt diagram below (Figure 7), example releases are marked with orange boxes. Provided dates are only for illustration purposes – actual releases must be agreed with WP2 and WP3, whenever they are ready to propose such dates. After integration is completed, internal milestone "ready for UAT" should be raised to confirm facility's readiness for final acceptance tests.

Note: It is already planned that first tests will be performed in April/May 2020 on selected facilities.

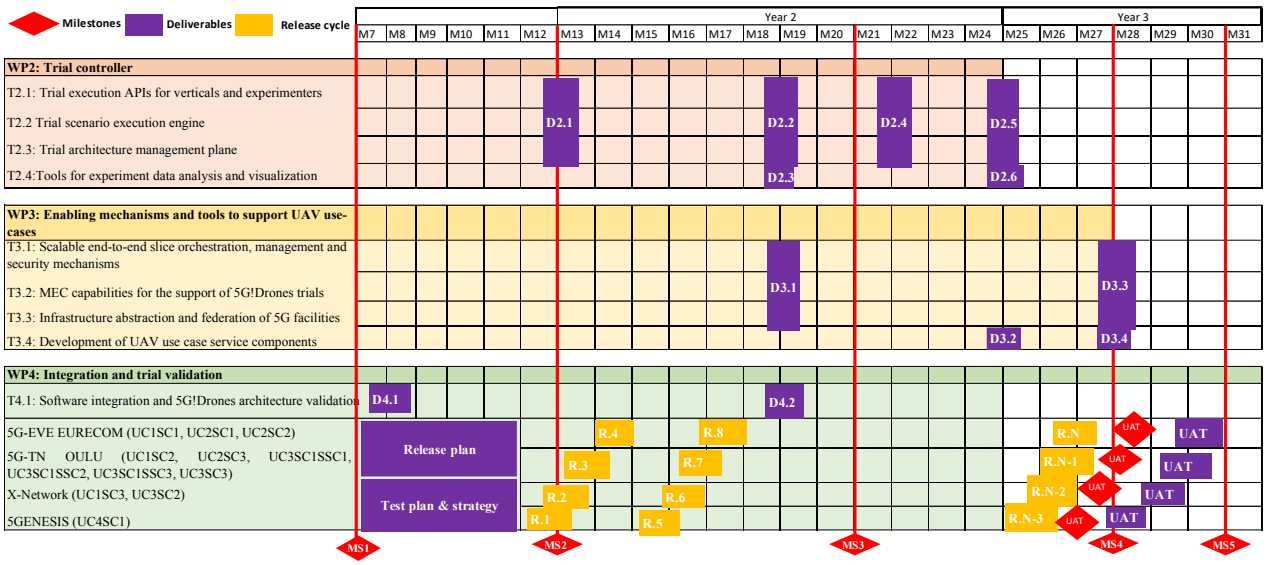


Figure 7 Gantt chart for integration activities

As indicated on the chart, the first weeks should be spent on detailed planning of test's strategy and release planning.

The initial test strategy plan was prepared and is documented in Appendix 3. Based on this document, further steps can be developed: test books, test tools and procedures, etc.

From the management perspective, list of key projects roles (like architects, project managers, test managers) were identified and assigned. As project progress, further contacts shall be provided. Furthermore, the responsibility matrix of roles and assignments has been created (Appendix 2).

3.2. Integration Phases/iterations

3.2.1. 1st iteration: M1-M12

The first iteration corresponds to the first year of the project. Having finalised (i) the requirement elicitation and analysis of each use case/trial, (ii) the initial architecture design and (iii) the design and definition of all provided tools, we will perform a feasibility test deployment of the use cases on top of the existing 5G infrastructures, without having integrated any of the 5G!Drones components, in order to check the feasibility of the use case execution, as well as the compatibility of the 5G equipment with the UAV systems.

Special interest to this feasibility check is on ICT-17 platforms, which are more mature in terms of experimentation in comparison to the independent 5G platforms that participate in the project.

By performing this feasibility check, we will have the chance to identify the tools which are necessary to be integrated for the execution of the trial, especially at the infrastructure layer,

and proceed to specification per platform the actual tools' integration order, bearing in mind the initial estimation of updates and refinements necessary in each case.

This information will be obtained by organising clustering activities involving the partners and platform owners that support specific use cases and trials. These clusters will help the platform owners and the tool providers to understand the technologies that the use cases are using and how much effort would take to extend the existing 5G facilities functionalities and adapt them to the 5G!Drones requirements.

3.2.2. 2nd iteration: M13-M18

The 2nd iteration is planned from M13 to M18. The main goal is for the target platforms to continue refining the tools already integrated and start integration of the ones planned, and for the tool owners to start integrating the common components of the 5G!Drones Framework.

The refinement of the existing tools in the 2nd iteration will be updated according to the comments of the use case partners and the feasibility check outcome of phase 1.

In addition, use case partners will start integrating the next planned set of tools in their 5G platforms. This process will have an easier start because all the available feedback and the common process set from the first iteration will be taken into consideration. Dedicated meetings between the tool owners and the use cases teams will be held, to support the deployment and configuration and through supportive material, such as videos and webinars, facilitate the successful integration. As a consequence, the tool owners will get extensive feedback on how to provide qualitative and distilled information to the use case partners as well as suitably support the testing and verification of the components under the phase 2 trials.

In order to support this work, monthly calls shall be planned, so that all use case partners can benefit from explanatory material and configuration clarifications that will be discussed.

Additionally, to this work where the tools and platform owners are updating and refining their facilities to help the integration by the use case partners, we will work also in internal components of the 5G!Drones Framework that had to be integrated.

3.2.3. 3rd iteration: M19-M27

The third iteration is planned from M19 to M27. This iteration focused in the refinement of the tools from the configuration phase (adaptation of the tools to the technical requirements of the use cases) and work in a more in-depth integration of the tools with the 5G!Drones Framework.

Regarding the first part, tools' leaders will have several meetings with use case partners for providing information of configuration and usage of the tools. The feedback will be compiled by each tool leader and prepared for when the trials start, as it will be the best way for evaluating how experimenter-friendly and useful all the material prepared will be and how fast can the tools adapt to the different technical and business needs of several different trials.

Additionally, to this work the tool leaders will work also in integrating more at low level their tools with 5G platforms and existing experimentation facility components. Therefore, tool leaders will work in providing APIs for accessing information of their tools that could be useful from a high-level point of view.

During the 3rd iteration, the Phase 2 preliminary trials shall be executed providing further feedback for the last iteration.

3.2.4 4th iteration: M28-M36

The last iteration is planned from M28 to M36 and refers to the period that the project trials and use case scenarios demonstrations, as designed and defined in D1.1, will take place.

3.3. Maturity of 5G facilities

The facilities to be used by 5G!Drones are provided/connected/supported by both ICT17 and non-INCT17 (independent) 5G platforms. The following existing 5G facilities will be integrated and used for trial execution:

- ICT-17 5GENESIS (https://5genesis.eu/)
- ICT-17 5GEVE (**https://www.5g-eve.eu/**)
- 5G-TN Oulu and
- X-Networks from Aalto University

Each facility will use a different set of 5G!Drones capabilities according to their existing built-in features. Below table (Table 2) summarizes the existing features' availability of each facility. Symbol 'X' identifies that facility has already its own relevant component. Symbol of the WP (or any other supplier) indicates, that the feature will be respectively delivered.

	5GGENESIS	5G-EVE	5G-TN	X-Network
Trial Engine	X	WP2	WP2	WP2
Trial Enforcement	X	WP2	WP2	WP2
Monitoring	X	X	X	X
KPI Analysis	X	WP2	WP2	WP2
Web Portal	X	WP2	WP2	WP2
Enablers	X/WP3	WP3	WP3	WP3
U-Space adapter	WP2		WP2	WP2
Additional rows for unforeseen components				

Table 2 Facility built-in and expected to be developed capabilities

The above table can be used for estimating release content for each facility, considering that the existing component already available at a specific platform can be used for the initial phases of the experimentation process, while extensions and improvements will be implemented in order to support better the UAV specific needs.

For example, in ICT-17 5GENESIS an experimentation portal is already available, which however does not currently support the interaction with the UTM system in order to be controlled and checked if there is the license to the drone to fly and therefore the experiment to be executed.

Integration activities will be performed independently for each mentioned above facility. It will consist of two main types of activities:

- Component/integration tests – iterative, more frequent, focused on testing new components' functionalities in connection with facility, mainly on API level
- Acceptance tests – focused on end to end user tests based on test scenarios derived from trial use cases

Component/integration tests will be performed in several iterations synchronized with subsequent releases of solution components (when delivered functionality of trial controller/enabler is in scope of use case to be implemented on respective facility). Its purpose is to validate the interface implementation but also to provide some feedback to WPs regarding potential discrepancies between design/assumptions and facility dependent implementation (e.g. performance, capability, etc.).

For each 5G!Drones platform it is assumed (and reflected in integration plan's Gantt chart, Figure 6), that at least one end to end session for user acceptance tests (UAT) need to be planned. Initial, proposed dates are described in the phases above, which are in line with the the 5G!Drone's Gantt chart and the ICT-17 5GENESIS experimentation planning. Component/integration tests iterations will be planned according to the appropriate release and facility availability dates.

3.4. Design-Build-Test Release cycle

The entire release cycle used for the purpose of integration and testing of 5G!Drones solution components is presented on the below diagram (Figure 8).

The 5G!Drones integration cycle is split into three distinctive stages: the design and build stages are a unified development workflow that will be followed by WP2 and WP3 in order to deliver the instances scoped for the release. The test stage will be addressed in WP4 to provide a structured approach to testing and integrating the concrete implementations of components built in WP2 and WP3.

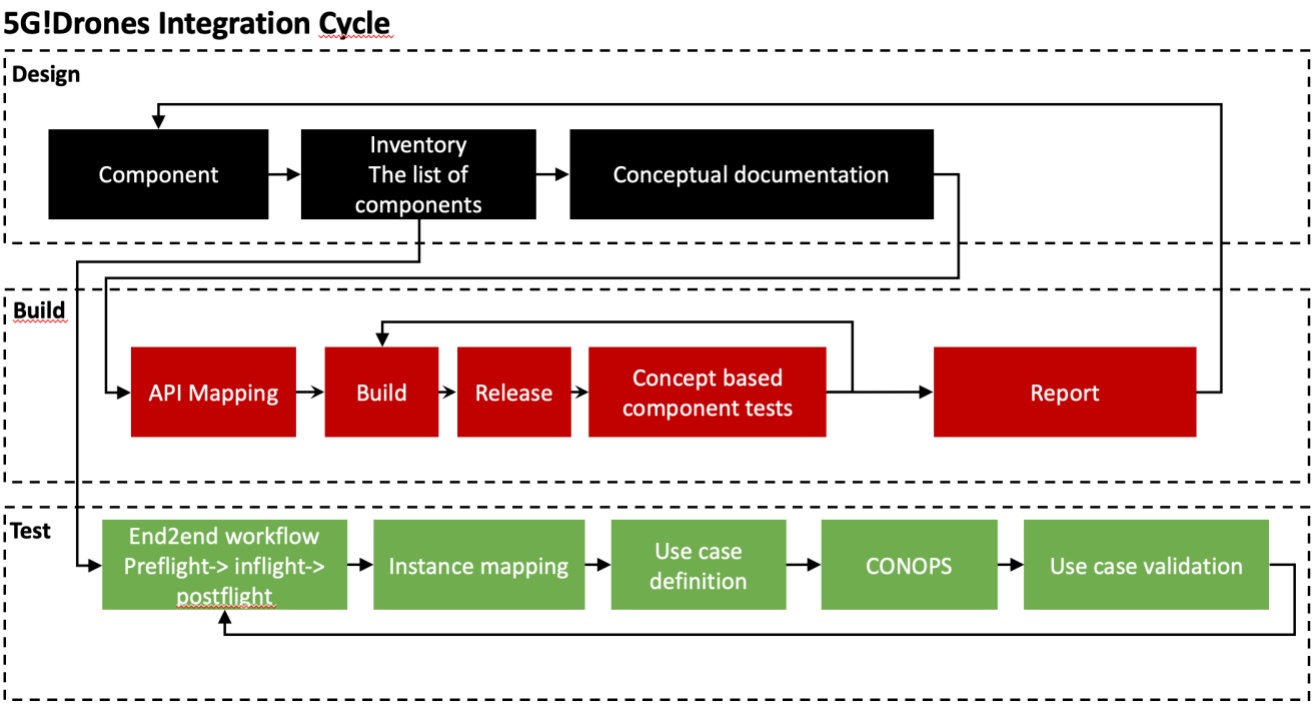


Figure 8 Release cycle for integration

Design and Build applies to every single component delivered within release in scope. Release might end up with Test stage, which consists of End to end user tests e.g. acceptance tests (UAT). Unit, development tests are assumed to be performed and eventually reported by delivering WP's development team. Integration cycle deals with component and integration tests. During the tests, tasks will be performed incrementally, as basic features will be available.

For the purpose of release tests, testing/staging environment 5G!Drones GitHub code repository will be created. Approach to testing and validation process is outlined in Appendix 1.

Below table (Table 3) contains more detailed description of each step depicted on the release lifecycle diagram.

5G!Drones Release Steps Definitions	
Component	Refers to a concrete functional component required by use-cases (e.g., Trial Engine, U-space adapter)
Inventory	The current set of available components
Component documentation	Component-specific documentation that details its expected behaviour on a conceptual level (e.g., Trial Engine must be able to ingest geographical data and ensure it does not interfere with no-fly-zones). This documentation will form the basis for integration tests
API mapping	In the API mapping, developers of a component identify the APIs that will be used to fulfill the requirements of the conceptual documentation of a given

	component
Build	During the build phase, developers work on a concrete implementation of a component
Release	A release is an incremental version of a component that is expected to satisfy a given list of features
Concept-based component tests	Unit tests written by module developers are run on released versions of components to test that they satisfy the conceptual documentation of the given component
Report	In this step, the implementation readiness of the given component is evaluated and reported to relevant stakeholders. Inputs of this report may inform the requirements of components that are yet to be built.
End-to-end workflow **Preflight -> Inflight -> Postflight**	The end-to-end workflow will identify the set of components (available in the inventory) that are required to enable a given use-case. A holistic approach to the Use Cases, requires a full understanding of process and workflows of all 5G project stakeholders. Due to the fact that tests are planned in three different locations, which until July 2020 are under the jurisdiction of different regulators, the test process will require individual approach. Also, it must be mentioned, that due to nature of the defined Use Cases, not all of them will be possible to be treated as a standard scenarios specified in Opinion No 05/2019. Therefore, members of the consortium responsible for unmanned systems and U-space, will define individually for each Use Case, the requirements of flight planning, required risk analysis, and obtaining necessary approvals. By connecting to the UTM system, communication between 5G facilities and systems relying on the U-space concept will be possible. Individual flow scheduled for each Use Case will be defined before testing and will be updated as consortium members gain practical experience from trial testing.
Instance mapping	During the instance mapping phase, specific instances of components are identified for use in evaluating the end-to-end workflow
Use-case definition	The use-case definition details the specific sequence of steps to execute the given end-to-end workflow with the identified instances
CONOPS	CONOPS definition as is: a document describing the characteristics of a proposed system from the viewpoint of an individual who will use that system such as a business requirements specification or stakeholder requirements specification. The CONOPS stage will output a document describing the characteristics of a proposed system from the viewpoint of an individual who will use that system
Use-case validation	The results of the trial execution are compared to the output of the CONOPS stage to evaluate the effectiveness of integrated components and identify areas for improvement – this analysis will drive (among other things) updates to the architecture, development of new components, iteration on existing components, and project-level reports

Table 3 Description of release steps

When the package is released, it may contain more than one component. In such a case described below design and build phases of the release cycle should be multiplicated accordingly (for each component in scope).

The table below (Table 4) provides an overview of the implementation readiness for a given component developed in WP2 or WP3 and can be considered as releases' roadmap. Table 3 is a product of the Release planning validated and updated by the Report step in above process. In the context of this table, "instance" refers to a concrete release version of a component (specific to a particular facility as required). It is possible for some component implementations to apply to multiple facilities.

Table 4 provides the flexibility to grow in the vertical axis as additional integration components are identified and/or developed by the implementation work packages. The list of these components and their estimated implementation dates will subsequently be used to drive the Design-Build-Test Integration Release cycles.

Facility NAME	WP	Instance A	Instance B	Instance C	…	…	…
Trial Engine	WP2	{DATE} + comments	{DATE} + comments	…	…		
Trial Enforcement	WP2	{DATE} + comments	…	{DATE} + comments	…		
Monitoring	WP2	{DATE} + comments	…	…			
KPI Analysis	WP2	{DATE} + comments	…				
Web Portal	WP2	{{DATE} + comments					
Enablers	WP3	{DATE}					
U-Space adapter	WP2						
Additional rows for unforeseen components, incl. expansion of "Enablers"	WP2/WP3						
Documentation	All						

Table 4 Release roadmap

As an example, the release roadmap for 5G-GENESIS facility is shown in the below table:

5GENESIS	WP	Instance A	Instance B	Instance C	…	…	…
Trial Engine	WP2	M12 (feasibility check of 5GENESIS component)	M18 (new version, if necessary)	M27 (new version, if necessary)			
Trial Enforcement	WP2	M12 (feasibility check of 5GENESIS component)	M18 (new version, if necessary)	M27 (new version, if necessary)			
Monitoring	WP2	M12 (feasibility check of 5GENESIS component)	M18 (new version, if necessary)	M27 (new version, if necessary)			
KPI Analysis	WP2	M12 (feasibility check of 5GENESIS component)	M18 (new version, if necessary)	M27 (new version, if necessary)			
Web Portal	WP2	M12 (feasibility check of 5GENESIS component)	M18 (new version, if necessary)	M27 (new version, if necessary)			
Enablers	WP3	M12 (feasibility check of 5GENESIS component)	M18 (new version, if necessary)	M27 (new version, if necessary)			
U-Space adapter	WP2	-	M18	M27			
UAV Special components	WP2/ WP3	-	M18	M27			
Documentation	All						

Table 5 Example of release roadmap for facility

3.5. Design-Build-Test Release cycle example

As an example of Design-Build-Test iteration of release cycle, the plan and release process of UC1SC1 is considered.

From the enablement matrixes table (Table 1 in section 2.5), it can be seen that:

- UC1SC1 is hosted by 5G-EVE
- This scenario, as it uses different slices, requires that trial enforcement module would be able to orchestrate the slices at 5G-EVE. The exact information about slices and the required parameters was determined on the basis of information gathered in Table 1, which is also instantly updated due to design development.

The basic integration approach for the specific case is presented below.

3.5.1 UC1SC1 Exemplary End-to-end Process and Information Workflow

3.5.1.1 Actors

As a first step the involved actors need to be identified, as noted in Table 6.

5G!Drones Actor	CONOPS Stakeholder
Aircraft	Aviation User
Unmanned Aerial Vehicle (UAV)	Drone Manufacturer, Drone Owner,
Ground Control Station (GCS)	Drone Manufacturer, Drone Owner, Drone Operator
U-Space Service Provider	U-space Service Provider
Authority	Civil Aviation Authority, Air Navigation Service Provider (ANSP), Aeronautical Information Management Provider (AIMP), (Airfield/Airport) Aerodrome operator (civil, Military), Surveillance Service Provider, CNS Infrastructure Service Provider, Communication Service Provider, Navigation Service Provider, Weather Data Service Provider, U-space Service Provider (Principal USSSP)
Authority (Public Safety)	Authority for safety and security (police, fire brigade, search and rescue orgs)
Telecommunication provider	5G facility

Table 6 Use case actors - example

3.5.1.2 Service provisioning workflow

Then the specific workflow steps that are necessary to be supported are identified:

Step 1. Delegated UTM service, including aeronautical restrictions will be implemented for a trial scenario

Step 2. Test user will register Drone and Drone pilot in UTM system

Step 3. User will initiate Drone Flight Plan via 5G Trial Engine Portal

Step 4. Flight Plan will be accepted/refused or modified by involved stakeholders based on aeronautical legislation requirements and 5G KPI (4D RAN SLA) via U-Space adapter

Step 5. Flight Plan will be submitted to all involved parties (stakeholders)

Step 6. Flight Plan will provision resources (Network Slices) via Trial Enforcement engine through UTM Adapter

Step 7. The 5G KPI monitoring service will be started

Step 8. Drone take –offs and is controlled via 5G C2 link

Step 9. Post analysis will be performed after the mission. KPI and Monitoring module will be updated

3.5.1.3 Equipment inventory

The checklist of all necessary tools, equipment, approvals, will be recorded in one report formula, which in future may be used as a supportive document for SORA analysis. This shall be constructed based on the analysis and specifications to be delivered by WP2 and WP3.

3.5.1.4 Business related aspects

Describes set of recommendations like potential costs of the service along with the estimated times of its implementation. Some expected KPIs with reference values e.g. maximum/minimum/average implementation time, etc. should be provided.

3.5.1.5 UC1SC1 scenario implementation

From the feature availability table (Table 2) in section 3.3, it can be read that for 5G-EVE there is no appropriate enabler and it should be delivered by WP3. Based on the above analysis, a new release R1.1.1 is planned (1.1 applies to UC1, SC1, last 1 identifies release number). This release will deliver Slice Orchestrator component to WP3.

For this release the instantiation of the integration cycle would be similar to the below table:

R1.1.1 Release Steps	
Component	Slice orchestrator
Inventory	{Slice orchestrator, trial enforcement module}
Conceptual documentation	The Slice Orchestrator will provide methods to CREATE, READ, UPDATE, and DELETE slices
API mapping	(At, e.g., 5G-EVE) We will allow users to CREATE a slice at a REST endpoint /createSlice that will respond with JSON with this schema...
Build	5G-EVE developers commit incremental changes to Github
Release	5G-EVE developers mark their codebase (after a given commit) as release-ready
Concept-based component tests	Unit tests are run that verify that the REST endpoints defined by 5G-EVE developers work as expected (without integration to other components)
Report	5G-EVE developers review the latest release and provide recommendations and a status update to WP3 and WP4
End-to-end workflow	An integration workflow is defined that will test the interfaces to 5G facilities from the trial enforcement module to the slice orchestrator
Instance mapping	The integration workflow defined above will use the 5G-EVE implementation of the slice orchestrator and the WP2 implementation of the trial enforcement module
Use-case definition	The trial enforcement module will use its interfaces to 5G services to create a slice, list that slice and confirm it exists, modify that slice, then delete that slice.
CONOPS	A specific series of steps will detail which actions will be generated by the trial enforcement module in order to test integration with the slice orchestrator
Use-case validation	WP4 integration managers will review the results of the integration test and provide feedback to 5G-EVE developers

Table 7 Release steps – use case example

After successful completion of release cycle, implementation readiness table for 5G-EVE can be updated accordingly:

5G-EVE	WP	Instance R1.1.1	Instance B	Instance C	…	…	…
Trial Engine	WP2						
Trial Enforcement	WP2	1/02/2020 Slice orchestrator					
Monitoring	WP2						
KPI Analysis	WP2						
Web Portal	WP2						
Enablers	WP3						
U-Space adapter	WP2						
Additional rows for unforeseen components, incl. expansion of "Enablers"	WP2/WP3						
Documentation	All components						

Table 8 Facility capabilities readiness - example

Above simplified example illustrates the idea behind the incremental development approach with integration release management that shall be applied to the 5G!Drones project.

3.6 Overall alignment of Integration plan

The integration plan is expected to be in alignment with the global project's schedule as published in Grant Agreement [2]. As part of the current work, the project's Gantt chart has been enriched, to reflect specifics of the integration tasks identified, as shown in the following diagram:

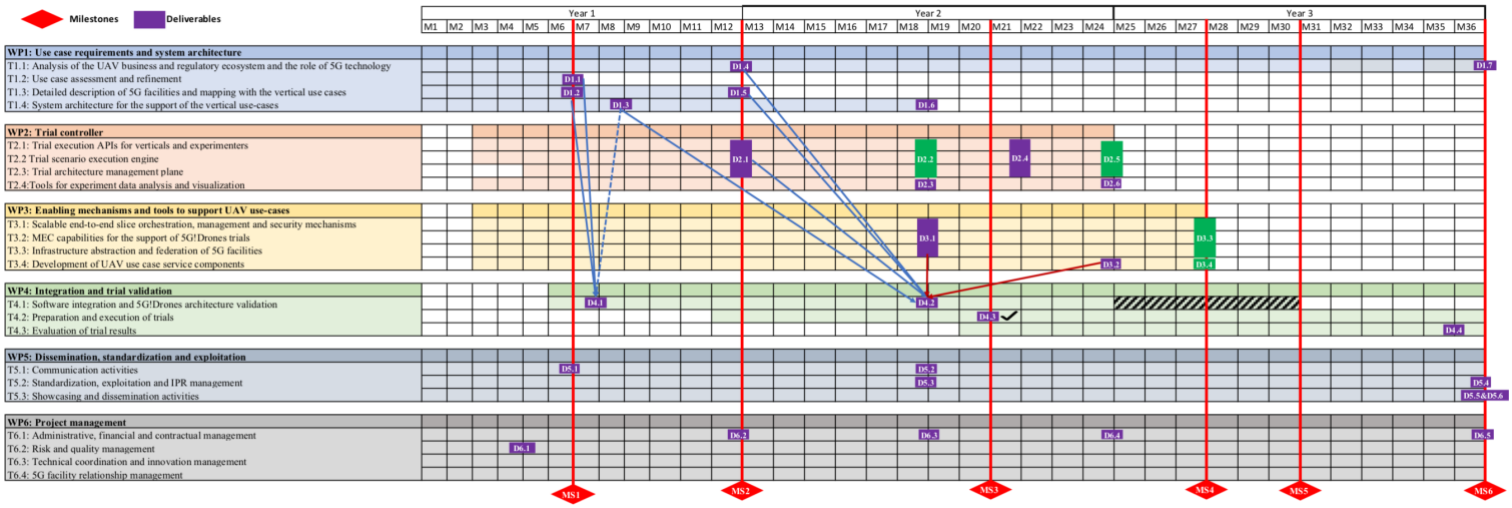


Figure 9 Project's Gantt chart

In detail, the following updates were performed:

- For planning purposes, illustration of D4.1 and D4.2 products dependencies and correlations with other project's deliverables were introduced: blue solid arrows represent identified relations, blue dashed lines potential dependencies and red solid arrows dependencies at risk (dependent deliverable is supposed to be released before precedent deliverable is ready)
- For implementation purposes, deliverables representing architecture elements to be integrated are marked in green color
- Potential extension of the period of duration beyond M24 (to be confirmed/validated if and for what period) of T4.1 is marked with hatched bar; this extension might be required to be able to perform required integration activities (e.g. unit, integration and acceptance tests) of particular solution elements (Trial controller – D2.5, 5G!Drones 5G enablers – D3.3 and UAV use case service components – D3.4)

4. CONCLUSION

This document has presented the initial integration plan of 5G!Drones and introduces an 4-phased iterative process of Build-Deploy-Cycles per facility. It has identified the basic steps and involved interactions for each integration activity.

The Initial Integration plan will be revisited and refined based on experience gained through first integration, to be delivered on M18 in the context of D4.2 "Integration status and updated integration plan". Work on specific tools utilized, as well details on target interfaces and testing procedures shall also be presented in D4.2. This document shall also detail the testing environments built, that are foreseen to include:

- Testing platform capable of hosting components delivered by WPs for component/integration testing integrated with all facilities
- Central repository e.g. git to handle code and documentation

- Tools for test and report automation

References

[1] D1.3 - System Architecture Initial Design
[2] Grant Agreement-857031-5GDrones
[3] D1.1 Use case specifications and requirements

Appendix 1 – Testing and validation process

The testing and validation process ensures that all components and interfaces necessary for the execution of the use cases conform to the design specifications and can be successfully deployed in the target environments. The output of the testing and validation process, apart from the Pass/Fail result is expected to provide quantitative and qualitative feedback to the subsequent integration cycles.

As a prerequisite, a dedicated integration and testing environment is assumed for the installation, integration and testing of the 5G!DRONES 'Release X' Components, that are delivered as part of the work carried out in WP2 and WP3 or already available at the member 5G platforms. The integration approach considers a Git-based methodology with semantic versioning (conveying information on time of creation, features, compatibility and major/minor/patch classification). Each Release is expected to include installation and configuration scripts that shall the effective deployment of the components in the diverse environments supported by the 5G!DRONES platforms.

As part of the Testing and Validation Process, 5G!DRONES introduces a Test Cases Descriptor template (see Table 9), 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. Upon the formulation of the final architecture, including the identification of the extensive list of interfaces and involved components, a set of common 5G!Drones test case descriptors shall be defined based on the descriptor template, to support the testing initiation and execution. The final set of identified test cases is expected to target the integration testing as well as final verification of the project results, and as such, is expected to include functional as well as KPI specific validation test cases' targets.

Each target platform is expected to select from the common set of Test Case Descriptors the most appropriate for the planned 5G!Drones instantiation, as well as to define its own, Use-Case specific ones, based on the scenarios to be demonstrated. Upon the execution of each test Case, the concluding results and observations shall be registered to provide feedback for the subsequent integration cycles. The set of Test Case Descriptors performed in each Integration Cycle per platform or Use Case, can be seen as the background to support the final evaluation of the trial and project results.

The proposed test cases template is provided in Table 9. The text in italics provides a brief description of the content expected per each attribute of the test case descriptor.

Test-Case-Id	**Test Case Name**	**# Integration Cycle**
Test Purpose	*Provide a description of the test purpose*	

Deployment	*Describe the deployment set up used for the integration test. Can be a target 5G platform, a sandbox environment or a specific simulation environment.*		
References	*Insert any reference on underlying standards (ex. 3GPP TR 22.829) and/or project's specification and reference points*		
Validation Target/Metric	*Define the quantitative or qualitative result(s) expected from the execution of the test case. In case KPI validation is involved in the test case (ex. Measure the end-2-end latency) it should be reported explicitly.*		
Involved Components	*List the architecture components and relevant versions that are involved in the test case execution*		
Involved Interfaces	*List the interfaces that must be established for the test cases execution (ex. Integration of the Slice Manager with the MANO for Service creation)*		
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 Tool	*Identify the test tool used for the execution of the test, can be a testing automation framework (e.x. Robot2, pyTest3, Watir4, jmeter5). Can also be Manual (Web browser, SSH client, Text editor).*		
Test Sequence	**# Step**	**Description**	**Result**
	1	*List the sequence of actions performed as part of the workflow/process under test. The result should make explicit the steps that are involved in the process under test but were skipped for any reason, as described in the final test verdict.*	DONE
	2		SKIP
	…		
	n		
Test Repetitions	*<num>*	*List the number of iterations that were necessary before concluding the test case execution and the corrective actions that were necessary before concluding the final verdict.*	

2 Robot – Online: http://robotframework.org
3 pyTest – Online: **https://docs.pytest.org/en/latest/index.html**, visited: 23.12.2019
4 Watir – Online: **http://watir.com**
5 jMeter – Online: **https://jmeter.apache.org**

Test Verdict	PASS/ FAIL/ PENDING	*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 test cannot be executed due to missing environmental characteristics or other external constraints (ex. Availability of 5G SIM cards)*

Table 9 Test Case Descriptor Template

Appendix 2 – RACI matrix for 5G!Drones Integration Plan

Role	Responsibility
WP4 Leader	The Work Package Leader (WPL) of Work Package 4 (Integration and trial validation), WPL4, is the responsible for managing the daily technical and administrative work of a work package. The work is further divided into tasks as a tool to help structure the activities performed and the task leaders have been appointed to support a WPL. The main responsibilities of a WPL are to: • Design and organise the work package activities. • Coordinate the technical work and check the progress of the work package. • Identify, coordinate, and harmonise deliverable content. • Verify that the work package objectives and targets are met.
T4.1 Leader	Task leader of T4.1 is responsible for managing the daily technical and administrative work of a task T4.1. The work is further divided into subtasks and actions as a tool to help structure the activities performed and the task leaders have been appointed to support a WPL. The main responsibilities of a T4.1 Leader are to: • Design and organise the work activities. • Coordinate the technical work and check the progress of the task. • Identify, coordinate, and harmonise deliverable content. • Verify that the task objectives and targets are met. • Transfer technical progress to the WPL, project coordinator and technical manager in monthly project management team (PMT) meetings and quarterly technical reports. • Schedule and chair task meetings and implement decisions.
T4.2 Leader	Task leader of T4.2 is responsible for managing the daily technical and administrative work of a task T4.2. The work is further divided into subtasks and actions as a tool to help structure the activities performed and the task leaders have been appointed to support a WPL. The main responsibilities of a T4.2 Leader are to: • Design and organise the work activities. • Coordinate the technical work and check the progress of the task. • Identify, coordinate, and harmonise deliverable content. • Verify that the task objectives and targets are met. • Transfer technical progress to the WPL, project coordinator and technical manager in monthly project management team (PMT) meetings and quarterly technical reports. • Schedule and chair task meetings and implement decisions.
FCT	The Facility Coordination Team (FCT) manages all the interactions with the facilities to schedule the usage of the facilities for trials and tests. It also manages the agreements in terms of IPR of the components to be used in the facilities. The FCT is composed by the project coordinator, technical manager of the project, the WPL and partners involved in the facilities (i.e. EUR, NCSRD, ORA, MoE).
Enterprise architect	Enterprise architect is responsible for E2E process and architecture design. He maintains overall architecture as well as data model across whole enterprise. He decides about development directions, performes impact analysis and makes final decisions, when changes in architecture and process flow is required.
Integration architect	Integration architect performs supportive function to enterprise architect, mainly in area of system's integration. There are many modules to be integrated to achieve final functionality, so this role is crucial during the trial implementation phase. He creates and maintains integration framework guidelines for all development teams in project.
Integration Manager	Integration manager is responsible for daily technical and administration work related to integration activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
Test Manager	Test manager is responsible for all testing and validation activities: planning, coordination, reporting, etc
Web Portal architect [INV, CAF]	Web portal architect is solution architect who is responsible for designing architecture of designated domain (Web portal).
Web Portal PM [INV, CAF]	Web Portal PM is responsible for daily technical and administration work related to Web Portal implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
U-space Adapter PM [UMS]	U-space Adapter PM is responsible for daily technical and administration work related to U-space Adapter implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
U-space Adapter Architect [UMS]	U-space Adapter architect is solution architect who is responsible for designing architecture of designated domain (UTM and UAV operator integration adapters).
Trial Enforcement Architect [ECM, CAF, UMS]	Trial Enforcement Architect is solution architect who is responsible for designing architecture of designated domain. (provisioning of MANO, MEC and Cloud)
Trial Enforcement PM [ECM, CAF, UMS]	Trial Enforcement PM is responsible for daily technical and administration work related to Trial Enforcement implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
UAV Repository PM [UMS]	UAV Repository PM is responsible for daily technical and administration work related to UAV Repository implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
UAV Repository Architect [UMS]	UAV Repository Architect is solution architect who is responsible for designing architecture of designated domain. (UAV repository)
5G Repository PM [????]	5G Repository PM is responsible for daily technical and administration work related to 5G Repository implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
5G Repository Architect [????]	5G Repository Architect is solution architect who is responsible for designing architecture of designated domain. (5G repository)
VNF Repository PM [AU]	VNF Repository PM is responsible for daily technical and administration work related to VNF Repository implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
VNF Repository Architect [AU]	VNF Repository Architect is solution architect who is responsible for designing architecture of designated domain. (VNF repository)
Lifecycle Manager PM [NOK]	Lifecycle Manager PM is responsible for daily technical and administration work related to Lifecycle Manager implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
Lifecycle Manager Architect [NOK]	Lifecycle Manager Architect is solution architect who is responsible for designing architecture of designated domain. (Lifecycle management)
Trial Translator PM [EUR]	Trial Translator PM is responsible for daily technical and administration work related to Trial Translator implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
Trial Translator Architect [EUR]	Trial Translator Architect is solution architect who is responsible for designing architecture of designated domain. (trial translator)
KPI Monitoring PM [FRQ, THA, RXB, ORA, AU]	KPI Monitoring PM is responsible for daily technical and administration work related to KPI Monitoring implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
KPI Monitoring Architect [FRQ, THA, RXB, ORA, AU]	KPI Monitoring Architect is solution architect who is responsible for designing architecture of designated domain. (data collection and reporting)
Trial Validator PM	Trial Validator PM is responsible for daily technical and administration work related to Trial Validator implementation activities: coordinates technical activities, plans, assure availability of necessary resources and performs daily basis governance activities (maintains reports, risk log, change management log, etc)
Trial Validator Architect	Trial Validator Architect is solution architect who is responsible for designing architecture of designated domain. (trial validation)
Communication and Business liaison [INF]	Analyzing WP4 activities for exploiting the achieved results from a business and communication perspective as input to tasks: T1.1 and T5.1
Trial Enforcement [DEM]	Trial automation by performing step by step the planned test caseupon appropriate interfacing with the Trial Engine

Role	Contact name	Contact email	Contact phone number
WP4 Leader [UMS]	Nemish Mehta	nemish@unmanned.life	+44 7521 514329
T4.1 Leader [DRR]	Pawel Korzec	pawel.korzec@droneradar.eu	48511230660
T4.2 Leader [CAF]	Tanel Järvet	tanel.jarvet@cafatech.com	37256911732
FCT (Funcional Tests) THA	Farid Benbadis	farid@benbadis.fr	+33141303929
FCT Coordinator X-Networks (Aalto)			
FCT Coordinator 5G-EVE			
FCT Coordinator 5G-TN			
FCT Coordinator 5GENESIS			
Enterprise architect	Thomas Lutz	thomas.lutz@frequentis.com	+43664608502363
Integration architect	Tanel Järvet	tanel.jarvet@cafatech.com	37256911732
Integration Manager DRR	Piotr Dybiec	piotr.dybiec@droneradar.eu	48501000601
Test Manager			
Trial Planning - Web Portal architect [INV, CAF]			
Trial planning - Web Portal PM [INV, CAF]	Pawel Montowtt	pawel.montowtt@involi.com	+48 602 408 878
Dashboard - Web Portal architect [INV, CAF]			
Dashboard - Web Portal PM [INV, CAF]	Pawel Montowtt	pawel.montowtt@involi.com	+48 602 408 878
U-space Adapter PM [UMS]			
U-space Adapter Architect [UMS]			
Configuration&Performance MEC - Trial Enforcement Architect [ECM, CAF, UMS]			
Configuration&Performance MEC - Trial Enforcement PM [ECM, CAF, UMS]			
Configuration&Performance MANO - Trial Enforcement Architect [ECM, CAF, UMS]			
Configuration&Performance MANO - Trial Enforcement PM [ECM, CAF, UMS]			
Monitoring&Management MEC - Trial Enforcement Architect [ECM, CAF, UMS]			
Monitoring&Management MEC - Trial Enforcement PM [ECM, CAF, UMS]			
Monitoring&Management MANO - Trial Enforcement Architect [ECM, CAF, UMS]			
Monitoring&Performance MANO - Trial Enforcement PM [ECM, CAF, UMS]			
UAV Repository PM [UMS]	Nemish Mehta	nemish@unmanned.life	+44 7521 514329
UAV Repository Architect [UMS]	Tomas Gareau	tomas@unmanned.life	+44 7521 514323
5G Repository PM [?????]			
5G Repository Architect [????]			
VNF Repository PM [AU]			
VNF Repository Architect [AU]			
Lifecycle Manager PM [NOK]			
Lifecycle Manager Architect [NOK]			
Trial Translator PM [EUR]			
Trial Translator Architect [EUR]			
Data Aggregator - KPI Monitoring PM [FRQ, THA, RXB, ORA, AU]	Gokul Krishna Srinivasan, Farid Benbadis	gokul.srinivasan@robots.expert, farid@benbadis.fr	3,58466E+11
Data Aggregator - KPI Monitoring Architect [FRQ, THA, RXB, ORA, AU]	Gokul Krishna Srinivasan	gokul.srinivasan@robots.expert, farid@benbadis.fr	3,58466E+11
Data Analysis - KPI Monitoring PM [FRQ, THA, RXB, ORA, AU]	Gokul Krishna Srinivasan, Farid Benbadis	gokul.srinivasan@robots.expert, farid@benbadis.fr	3,58466E+11
Data Analysis - KPI Monitoring Architect [FRQ, THA, RXB, ORA, AU]	Gokul Krishna Srinivasan	gokul.srinivasan@robots.expert, farid@benbadis.fr	3,58466E+11
Trial Validator PM	Saadan Ansari	Saadan.ANSARI@frequentis.com	+431811501425
Trial Validator Architect	Thomas Lutz	thomas.lutz@frequentis.com	+43664608502363
Communication and Business liaison [INF]	Vaios Koumaras	vkoumaras@infolysis.gr	+30 2103004250 - internal 12
Trial Enforcement (T2.3 Trial architecture management plane) [DEM]	Stavros Kolometsos	stkolome@iit.demokritos.gr	+30 21065031xx

Appendix 3 – General Test Strategy (initial draft)

"5G for Drone-based Vertical Applications"

D4.1 – General Test Strategy

Document ID:	**D4.1**
Deliverable Title:	**Integration Management Plan**
Responsible Beneficiary:	**UO**

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:	**36 Months**
Contractual Delivery Date:	**M04**
Actual Delivery Date:	**<Date of submission>**
Dissemination Level:	**Open Research Data Pilot (ORDP)**
Contributing Beneficiaries:	**AIR, ALE, AU, CAF, COS, DEM, DRR, EUR, FRQ, INF, INV, MOE, NOK, OPL, ORA, RBX, THA, UMS, UO**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857031.

Document ID: D4.1
Version: V1
Version Date: 16.12.2019
Authors:

Security: Open Research Data Pilot (ORDP)

Approvals

	Name	Organization	Date
Coordinator			
Technical Committee			
Management Committee			

Document History

Version	Contribution	Authors	Date

Executive Summary

To be completed

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Software in Subject		This term generally refers to the integrated software solution, which will be delivered by the Beneficiaries to 5G!Drones and is the main subject of this document
5G!Drones	5G!Drones Test Manager	In most of the cases this term will refer to the responsible person, defined as Single Point of Contact for any test management related issues from 5G!Drones's side. In some cases depending on context, it will refer to the 5G!Drones's Team in general.
5G!Drones	5G-TM, 5G!Drones Test Manager	In most of the cases this term will refer to the responsible person, defined as Single Point of Contact for any test management related issues from 5G!Drones side. In some cases depending on context, it will refer to the 5G!Drones Project in general.
5G!Drones software modules		Part of the 5G!DRONES project where the following deliverables are considered: WP 4.1, 4.2, 4.3 Deliverables
Implementation phase		This term refers to the sub-phase of 5G!DRONES Project, during which the actual development and accompanying test activities will take place.
Acceptance phase		This term refers to the sub-phase of the 5G!DRONES Project, during which the ready solution will be delivered to Test Bed facilities premises and will be validated for correspondence to its requirements
Defect	software defect, software issue	Any discrepancy between the expected behavior of Software in Subject, defined in the requirement documents (business, marketing, functional, technical or program), or documents derived from them (Test Objectives, Test Cases etc.) and actual behavior of the Software in Subject
Test Case	Test Procedure	Description of initial state of the Software in Subject, set of instruction to manipulate it, and description of final state of the Software in Subject, used to test its behavior in conditions, described in requirement documents
Requirements/Bug Tracking software		Piece of software, incorporating some or all of the following functions: requirements management, software architecture, computer programming, software testing, software maintenance, change management, project management, and release management. Bug Tracking system to be known after 5G!Drones Team choice

Test Levels		
FAT	Factory Acceptance Testing	- Refer to Chapter TEST LEVELS
INT/AFT	Integration Testing & Acceptance Functional Testing	- Refer to Chapter TEST LEVELS
Re & NFT	Reliability and Non Functional Testing	- Refer to Chapter TEST LEVELS
Data Migration	Data migration tool testing & Dry Run	Refer to Chapter TEST LEVELS
UAT	User Acceptance Testing	- Refer to Chapter TEST LEVELS
Disaster Recovery	Disaster Recovery Testing	Refer to Chapter TEST LEVELS

3GPP	3rd Generation Partnership Project
5G	5th Generation Cellular Technology
5G-PPP	5G Infrastructure Public Private Partnership
ADS-B	Automatic Dependent Surveillance – Broadcast
API	Application Interface
AR	Augmented Reality
BVLoS	Beyond Visual Line of Sight
CAA	Civil Aviation Authority
CC	Creative Commons
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
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
SORA	Specific Operations Risk Assessment
SSH	Secure Shell
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	Virtualised 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 purpose of this document is to provide general overview of planned testing activities within WP 4.1 Deliverable on every stage of integration phase for 5G!Drones Project. Implementation and acceptance phase, targeted on assuring the sufficient quality of delivered solution, describe the scope and required deliverables on every stage of testing, specify entry and exit criteria of every stage and ways to measure the quality of delivered solution.
This document shall be used as a basis for creation of Master Test Plan and all underlying Level Test Plans as specified below in Test Levels section.

This document will cover:

1. Overview of fundamental test process and its adaptation to the 5G!Drones Project
2. High-level description of the test activities on all test levels
3. Interdependencies between the test activities
4. Entry and exit criteria of each test level
5. Deliverables of every test level activities
6. Description of high-level test design approach
7. Description of the defect management process and defect lifecycle

This document will not cover:

1. Detailed planning of the test activities
2. Detailed description of test design techniques used on each test level (subject of Level Test Plans)
3. Training needs
4. Environmental needs (subject of Level Test Plan)
5. Tools and additional software needs (subject of Master Test Plan)

Following activities to take place within this task:

Incremental deployment and unit tests in a laboratory environment.
Deployment and individual component testing on the 5G facilities.

- Web portal (northbound API)
- UTM (southbound API)
- UAV operator (southbound API, UAV API/SDK, 5G Facility service access)
- MANO (southbound API)
- MEC (southbound API)
- UAV Repository – Fleet Management Trial Facilities (internal)
- 5G Repository (internal)
- VNF Repository (internal)
- Lifecycle Manager (internal)
- Trial Translator (internal)

- U-space Adapter (internal)
- Trial enforcement (internal)
- KPI monitoring (internal)
- Trial Validator (internal)
-

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.

UAT tests in Aalto University
UAT tests in 5GEVE EURECOM
UAT tests in 5GTN Oulu

Functional tests of the selected scenarios over the selected facilities

Task to work in close synergy with WP2 and WP3

- Provide continuous feedback from the integration activities for the refinement of the designed trial architecture and enablers

- Deliverables:

Integration Plan (D4.1): M07
Integration Status and Updated Integration Plan (D4.2): M18

2. 5G!DRONES KEY OBJECTIVES

To be completed

2.1. Project objectives

To be completed.

3. PROJECT OVERVIEW

3.1. Members of the Project

Table 1: Members of The 5G!Drones Consortium

Participant No.	Part. Short name	Participant organisation name	Country
1 (Admin. Coordinator)	UO	OULUN YLIOPISTO	Finland
2 (Tech. Coordinator)	THA	THALES SIX GTS FRANCE SAS	France

3	ALE	ALERION	France
4	INV	ONESKY SARL (INVOLI)	Switzerland
5	HEP	Hepta Group Airborne OÜ	Estonia
6	NCSRD	NATIONAL CENTER FOR SCIENTIFIC RESEARCH “DEMOKRITOS"	Greece
7	AU	AALTO KORKEAKOULUSAATIO SR	Finland
8	COS	COSMOTE KINITES TILEPIKOINONIES AE	Greece
9	AIR	AIRBUS DS SLC	France
10	UMS	UNMANNED SYSTEMS LIMITED	United Kingdom
11	INF	INFOLYSIS P.C.	Greece
12	NOK	NOKIA SOLUTIONS AND NETWORKS OY	Finland
13	RXB	ROBOTS EXPERT FINLAND Ltd	Finland
14	EUR	EURECOM	France
15	DRR	DRONERADAR Sp z o.o.	Poland
16	CAF	CAFA TECH OÜ	Estonia
17	FRQ	FREQUENTIS AG	Austria
18	OPL	ORANGE POLSKA SPOLKA AKCYJNA	Poland
19	MOE	MUNICIPALITY OF EGALEO	Greece
20	ORA	ORANGE SA	France

3.2. Project Organizational Structure

3.2.1. Project Governance

The goal of this structure is to secure proper management and governance of the WP 4.1 from both Technical and Business point of view by direct involvement of the executive committee, assignment of a dedicated WP’s team (PMO) and definition of work streams with appointed stream leaders and functional/technical experts inside each stream. The structure ensures accurate communication flows from/to selected Beneficiary to/from WP’s leader, and the internal WP communication

The definition of the roles and their respective level of accountability, responsibility, participation and information is described in chapter 3.3.

The structure of the governance of the program is described below in figure 3:to be updated

The 5G!Drones project management structure is depicted in **Error! Reference source not found.**. The roles of each element of the management hierarchy are described in the following subsections.

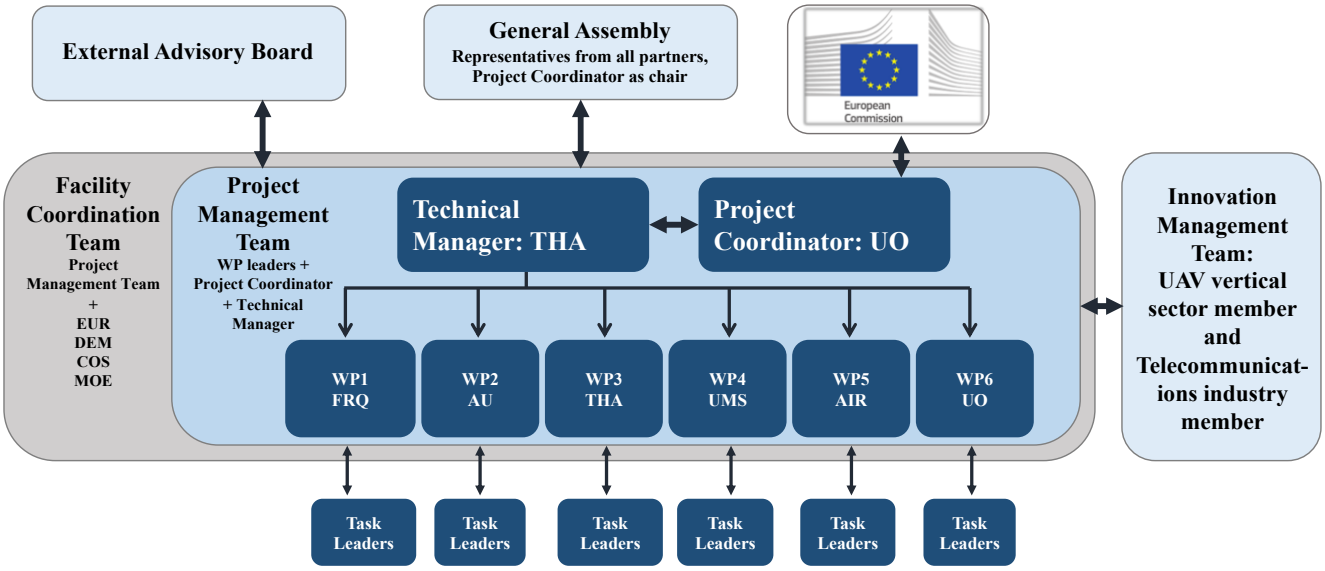


Figure 1: 5G!Drones management structure schematic.

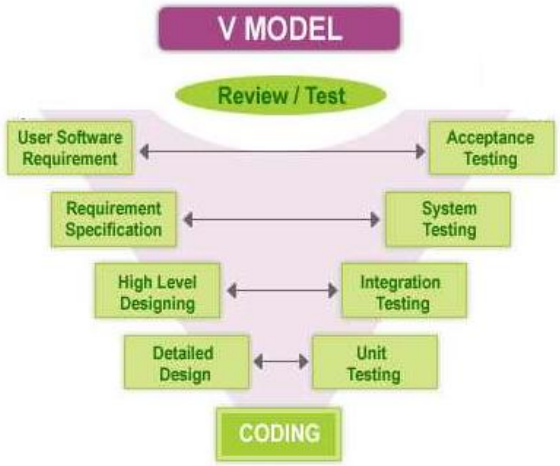
3.2.2. Roles and Responsibilities

The roles and responsibilities for both PMO members and all other internal and external stakeholders are described in the tables below. For any personnel and Program organization changes the PMP will be updated accordingly.

4. FUNDAMENTAL TEST PROCESS

4.1. Test Methodology

General testing methodology used in 5G!DRONES Project is based on V-model of software development lifecycle/process. V-model assumes that each part of the delivery – documentation and also the delivered system – is verified, validated and tested in various phases, to recover potential problems and defects as soon as possible. All planed phases of software development lifecycle includes corresponding phase of testing.
Testing consists of analysis of input documents, process analysis, test documents preparation, test data preparation, tests execution and reporting.



4.2. Main goals of testing

The purpose of Integration and E2E tests is to provide the end users with confidence that the system will function according to their expectations. Reasoning from this fact, the scope of testing will base on marketing and business requirements.

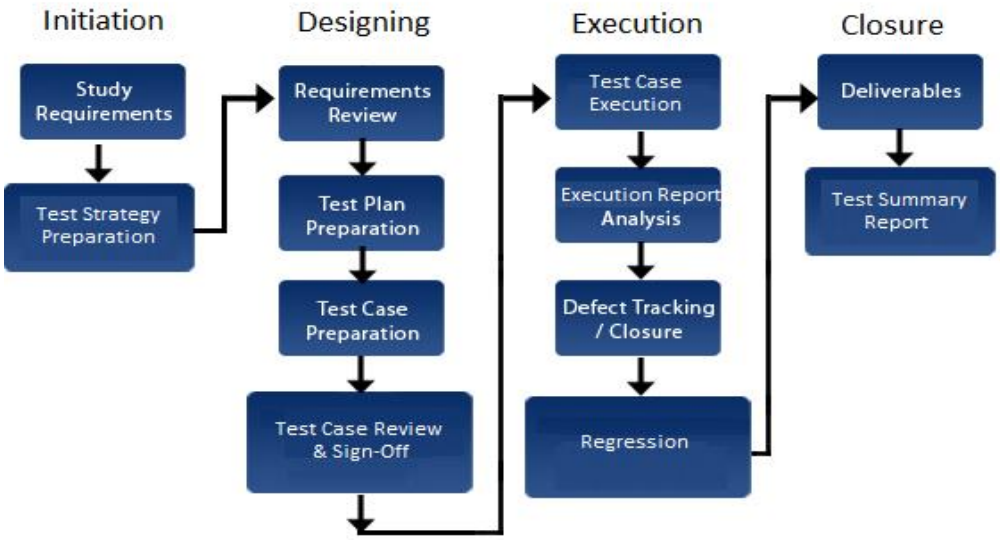
Typical E2E Test activities included in any Testing Project include –

- Test Planning
- Test Design
- Test Execution
- Monitoring & Control
- Evaluation & Closure

Irrespective of Test phase involved above sequence will be followed.

4.3. Test process

The fundamental test process comprises test planning and control, test analysis and design, test implementation and execution, evaluating exit criteria and reporting, and test closure activities.



Test process is divided into ***four consequent phases***. Quality gates define if all conditions are fulfilled to advance to the next phase. Every phase is composed by tasks/activities which are assigned to certain roles (see **RACI matrix** in chapter 1.0, ref.2).

- **Test planning phase (see chapter 6)**
  - Test plan creation
  - Test tools definition
  - Test team creation
  - Meetings & Escalation definition
  - Reporting & documentation definition
- **Test preparation phase(see chapter 6)**
  - Test book preparation in (tools selection ???)
  - Test sets & cycles preparation in (tools selection ???)
  - Test schedule
  - Test environment preparation
  - Test data analysis and preparation
  - User access / rights granted
  - Working instructions defect resolution
- **Test execution phase(see chapter 6)**
  - Test execution tracking and reporting
  - Defect tracking, reporting and escalation
  - Risk evaluation, prioritization, replanning
  - SLA tracking, escalations
  - Bug fix delivery and retest
  - Test exit report
- **Process improvement phase (see chapter 6)**
  - Lessons Learned
  - Test Summary Report

5. RESPONSIBILITY

The General Test Strategy (GTS) is approved by the Work Package Leader (WPL) and used as a guideline by all activities within WP 4.1. The GTS will be progressively elaborated by updates through the course of the 5G!Drones project. The **DRR** Officer will be responsible for updating the GTS.

6. TEST ACTIVITIES

Typical E2E Test activities included in any Testing Project include –

- Test Planning
- Test Design
- Test Execution
- Monitoring & Control
- Evaluation & Closure

Irrespective of Test phase involved above sequence will be followed.

6.1. Test Planning

Test Planning activity is one of the most important activity of Test Managers of 5G!Drones project. Test Planning lies in the area of direct responsibility of Test Manager and will be performed under D4.1 WorkStreamLeader supervision.
This activity presumes developing the list of tasks and milestones for each test phase to track the progress against, as well as defining the shape and size of test effort and forecasting the prerequisites and dependencies of each test phase.
The main document, produced as a result of test planning activity is Master Test Plan. It should provide the high level definitions of all planned test activities, sufficient to enable global project scheduling and effort allocation. Master Test Plan should be provided at early stage of the project and before the start of any test activities.
The detailed overview of every test phase will be covered in the Level Test Plans, which should be delivered by Test Manager and approved by D4.1 WSL before the start of corresponding test phase.

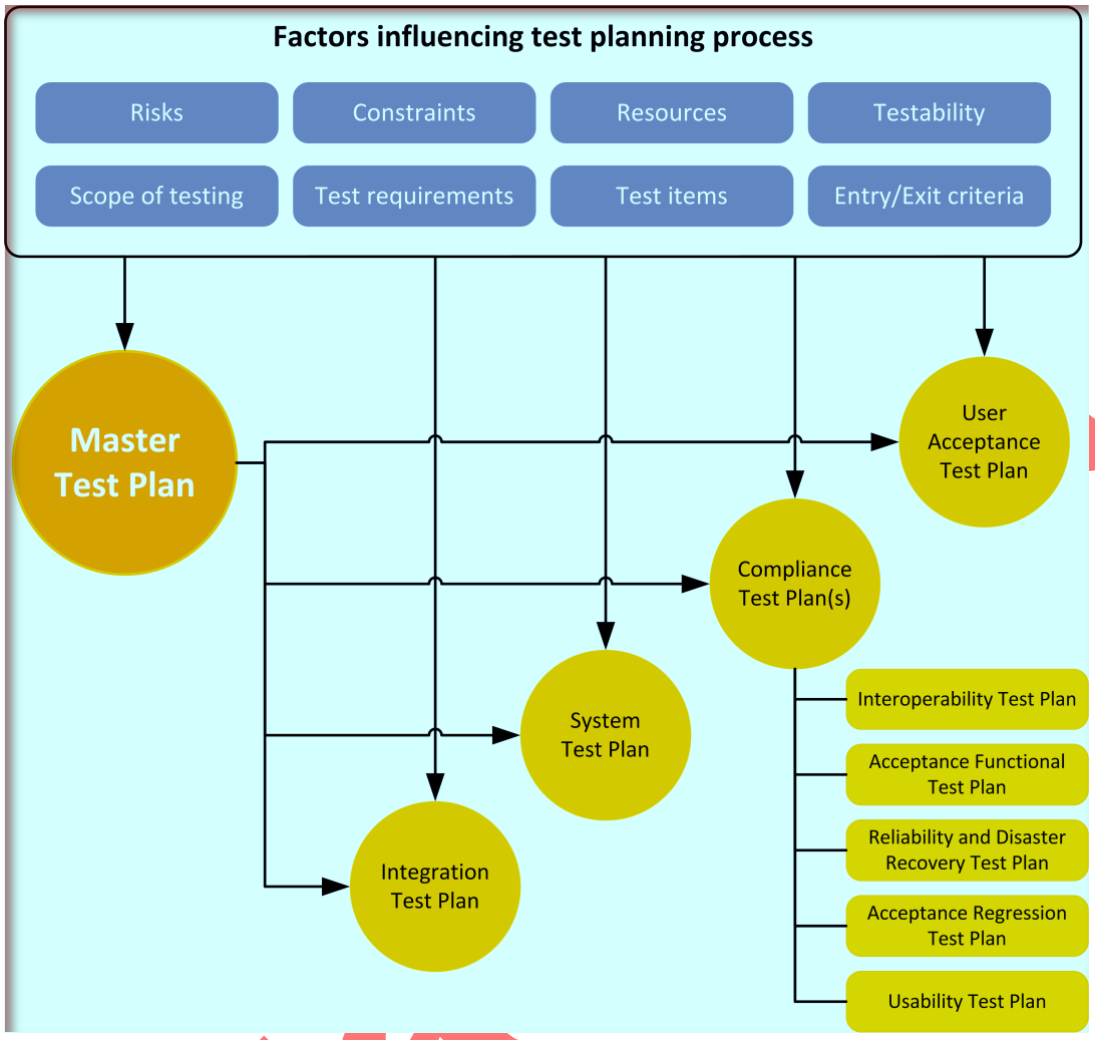


Figure 4. Test Planning

The Master Test Plan and all Level Test Plans delivered by 5G!Drones should correspond to the **IEEE 829**, the Standard for Software Test Documentation, and contain the items, listed in Table 1 (but not necessarily be limited to them)

ID	Heading	Description
1	Test plan identifier	A unique identifying reference such as 'Doc ref XYZ v2'
2	Introduction	A brief introduction to the document and the project for which it has been produced
3	Test items	A test item is a software item that is the object of testing. A software item is one or more items of source code, object code, job control code, or control data. This section should contain any documentation references, e.g. design documents.
4	Features to be tested	A feature is a distinguishing characteristic of a software item (e.g. performance, portability, or functionality). Identify all software features and combinations of features and the associated test design specification.
5	Features not to be tested	Identify all software features and significant combinations and state the reasons for not including them.
6	Approach	Details the overall approach to testing; this could include a detailed process definition, or could refer to other documentation where these details are documented, i.e. a test strategy
7	Item pass/fail criteria	Used to determine whether a software item has passed or failed its test
8	Suspend/resume criteria	Suspension criteria define criteria for stopping part or all of the testing activity. Resumption criteria specify the requirements to resume testing.
9	Test deliverables	The documents that testing will deliver, e.g. according to IEEE 829, these should include: - test plans (for each test level) - test specifications (design, case and procedure) - test summary reports
10	Testing tasks	All tasks for planning and executing the testing, including the intertask dependencies
11	Environmental needs	Definition of all environmental requirements such as hardware, software, PCs, desks, stationery, etc.
12	Responsibilities	Identifies the roles and tasks to be used in the test project and who will own them
13	Staffing and training needs	Identifies any actual staffing requirements and needs any specific skills and training requirements, e.g. automation
14	Schedule	Document delivery dates and key milestones
15	Risks and contingencies	High-level project risks and assumptions and a contingency plan for each risk
16	Approvals	Identifies all approvers of the document, their titles and the date of signature

17	Input Test Data	Data which has been specifically identified for use in tests (definition same to Initial Conditions)
18	Expected Result	The result which are expected after executing the software. According to the requirements what a customer has asked for in the software.
19	Actual Result	Result was given at the end of an tests.
20	Test Log	Document which consists of information about the test cases. Means whether the test case is Passed or Failed.

Table 1. Test Plan sections according to IEEE 829

The Testing Plan is structured as follows:

- Testing Team - Resources
- Main Testing Activities
- Key Milestones and Deliverables

6.1.1. Testing Team – Resources

This section is a reference to Project Specification document.

Information referred here is the following:

- Team Structure - Describe the test organization. Include all participating teams as well as other organizational groups that have a role in the planning and execution of testing or may have an approval role.
- People requirements

6.1.2. Main Testing Activities

For 5G!Drones, which is being a technology project the testing will include below testing activities:

- Planning of Testing activities
- Test Cases preparation
- Test Data preparation
- Inter-Operability Testing
- Regression test where needed
- Reliability & Non-Functional Testing

6.2. TEST PROGRESS MONITORING

In the process of testing is necessary to ensure control testing. The purpose of monitoring is to provide testing feedback and visualization of the testing process. Necessary for the control of information is collected and used to assess the status and decision-making, such as coverage (e.g., coverage requirements or code tests) or exit criteria (for example, the criteria for the testing).

During every stage Test Report with Metrics agreed will be provided by Test Manager to D4.1 WS Task Leader. The frequency of this activity is different for every stage and is defined below.

Refer to Master Test Plan for Reporting techniques and details.

Test Process Controlling and Monitor Summary –

Report	Frequency	Who
Factory Acceptance Testing Report	End of the phase	Created:Software Developer/ Supplier Approved: 5G!Drones test manager
Interoperability & Acceptance Functional testing status Report	Daily	Created: test manager Approved: D4.1 WS Task Leader
Reliability & Non Functional Testing status Repot	Daily	Created: Supplier test manager Approved D4.1 WS Task Leader
UAT Testing status Report	Daily	Created: test manager Approved: 5G!Drones Project Coordinator Informed: Supplier test manager

6.3. TEST CONTROL

5G!Drones Test Manager shall ensure below in confines of Test Control -

- Making sure the Test Plan satisfies the needs of the project and approved by all stakeholders.
- Making sure the test activities are correctly incorporated in the global project plan, all dependencies and contingency risks are taken into account.
- Making sure the testing tasks are distributed between corresponding project participants, the test schedule is feasible and test activities are executed in correspondence with it.
- Making sure the test activities are executed to high-enough standard, giving the required level of confidence in the results.
- Evaluating entry and exit criteria of each test stage. In case the criteria are not met, liaise with the involved project members to resolve the dependencies and put the project back on track.

- Continuously monitor the progress of all test activities and take action in case of delays, which can potentially affect the overall project schedule
- Putting in place comprehensive metrics and reporting, providing a well-rounded project status overview from testing perspective.
- A daily/weekly test report (Test &Defect report) must be shared with D4.1 WS Task Leader and Project Coordinator
- Before test starts initial testing conditions must be validated

6.4. TEST DESIGN

Classic black-box design techniques will be used to transform Test Requirements into sets of Test Cases.

For functional tests:

1. equivalence partitioning
2. boundary value analysis
3. decision tables
4. state transition diagrams

For UAT:

1. use case testing (should be provided via Level Test Plan)

Black box testing is a Testing, either functional or non-functional, without reference to the internal structure of the component or system. So in this method internal structure of program is not considered, tester should provide input set to the program and test whether the program is giving expected output or not.

This method is called as black box because, tester is not aware of the software program. Software program is like a black box; inside which tester cannot see.

It should be specifically noted, that Test Analysis and Design stage is the last stage of the project, when Supplier can challenge the requirements and request update/change of their contents. All unclear and ambiguous points in requirements should be explicitly clarified at this point, and all missing details requested by Software or Hardware Supplier should be added by appropriated Project participants

Final result of Test Analysis and Design activity – Sets of Test Cases – should satisfy the requirements towards Test Case Design, as specified in **IEEE 829**:

1. Unique identifier
2. Test Requirements reference
3. Baseline Date and Actual Date
4. Test steps and input data

5. Expected results description and output data forecast
6. Actual results
7. Status (including executed, Suspended, Passed, Fail, Suspended, Resume …)
8. Priority (based on priority of covered requirements)
9. Impact (Test Plan)

Final Test Case library should provide 100% coverage of corresponding requirements, including negative scenarios.

The test design is performed as part of the detailed test preparation. Test Design will produce:

- Test Scenarios
- Test Conditions
- Test Cases / Scripts

It is required that all (non-unit) test cases will be traced back to specific requirements (i.e. Requirements Traceability Matrix).

With respect to this specific project, i.e. 5G!Drones, we will have the following Test Books designed as a part of the Testing Delivery:

- Module 1 FAT Test Book
- Module 2 FAT Test Book
- Module 3 FAT Test Book
- Module 1 IOT/AFT Test Book
- Module 2 IOT/AFT Test Book
- Module 3 IOT/AFT Test Book
- Module 1 UAT Test Book
- Module 2 UAT Test Book
- Module 3 UAT Test Book
- Reliability & Non-functional Test Book

6.5. TEST EXECUTION

Test execution phase shall cover these activities and requirements

- Testers
  - All testers shall be familiar with the system and test tools
  - Collaboration of developers/integrators in knowledge transfer is assumed
- Access
  - All persons involved in testing shall have access in all tools and systems necessary for proper test execution

- Check list of these systems will be prepared

- Test data preparation
  - Test data is appropriately planned and arranged on time for Execution
- Test Execution
  - Testers are executing the Test Scenarios according the plan and day to day assignments
  - Received results are compared with expected results; incident with lower priority shall be raised in case of differences
  - Incident with higher priority shall be raised in case of failure, malfunction or other unexpected behavior
- Defect handling
  - Defect handling process is described in separated chapter
  - In general every incident reported by testers shall be given priority and SLA
  - Each incident has to be analyzed and assigned to the responsible party
  - Developers/Integrators shall take care for these incidents along with test designers
  - All fixed defects will be properly tested and released according agreed release process

The Test Scenario can be in following statuses

Status	Description
Not started	Test Scenario has not been started yet
Not completed	Test Scenario is ongoing and waiting for next action
Passed	Test Scenario was successfully passed
Failed	Test Scenario failed and defect was raised
N/A	Test Scenario is not possible to run, e.g. due to change of business requirement
De-scoped	Test Scenario is not relevant and was removed from the scope based on the mutual agreement
Blocked	Test Scenario cannot be executed due to opened defect(s) or feature unavailability

6.6. TEST EVALUATION

The main purpose of this activity is to assess, whether project at certain point meets the initially defined exit criteria and can be moved forward to the next test level/project stage. In case when exit criteria are not met, a decision shall be taken, whether more tests are needed or the specified exit criteria need amending. As decision should be taken on project management level and can affect general project schedule, it should be approved by Program Management Team

6.7. Acceptance phase

As entering the acceptance phase (Compliance Tests, UAT) presumes that a ready for acceptance testing product functionality has been delivered a continuous testing activity will take place, which can be interrupted only if suspend criteria are met (e.g. execution is blocked by a showstopper defect).

Defects, discovered during the acceptance phase will be immediately evaluated by Test Management and Work Stream Owners, and reported to Software Developers/Supplier, and fixed according to agreed timelines. After each defect-fix drop, defect fix validation and regression tests on corresponding functional area will be executed at highest priority (scope of regression tests will be evaluated individually by test management in collaboration with development team).

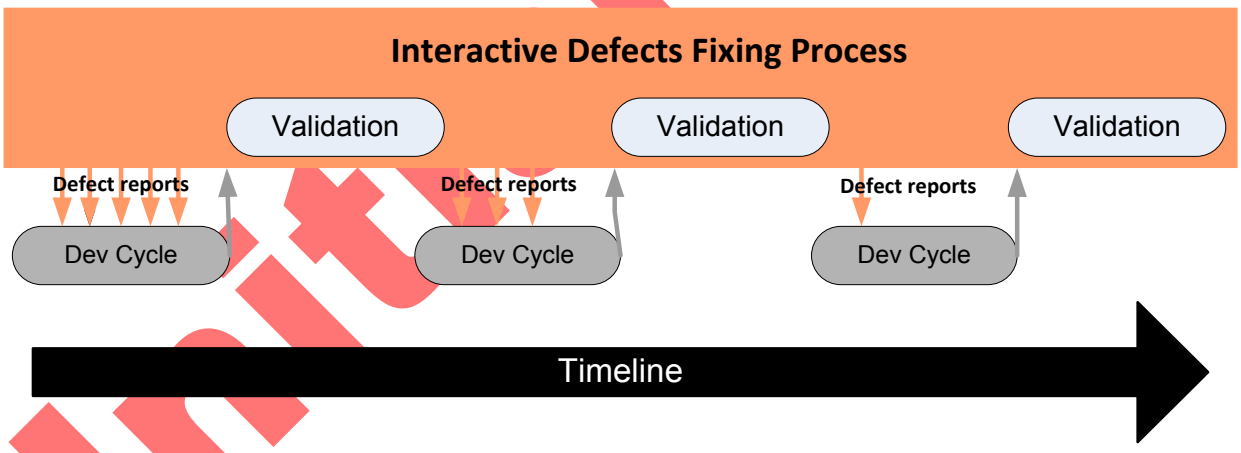


Figure 7. Interactive Defects Fixing Process

Dev Cycle = is a process of problem analysis and defect fixing. Validation = is a process of test execution including non-regression tests.

After exit criteria for specific acceptance activity are met, a Test Report will be elaborated by Supplier, highlighting main features of the test activity in subject, outcomes of the test activity, and project status in general.

This report will be subject of Test Evaluation (see V.5 Test Evaluation), and, as a result of evaluation process, either new Development Cycle or Test Termination process for current activity will be initiated.

6.8. TEST CLOSURE

Test closure activity presumes that testing on current level is finished. Scope of this activity is to make sure that everything is tidied away, reports written, defects closed, and those defects deferred for another phase clearly seen to be as such.

Test is considered closed if both Supplier and all stakeholders agree on the test closure, based on test evaluation.

6.9. TEST DELIVERABLES & RESPONSIBILITY MATRIX

The table below gives an overview of the testing deliverables.

Note: Test cases noted in the deliverables column below can be expanded where necessary to match tests type.

Deliverables/ Outcomes	Pre-Requisites
Test Strategy Document	Solution Requirements Specifications & HLD documents of the proposed Solution.
Master Test Plan	Test Strategy, Use cases
FAT Test Cases	Test Plan, HLD, LLD, Use cases & Requirement Traceability Matrix
IOT/AFT Test Cases	Test Plan, Use cases
Reliability Test Cases	Test Plan, Use cases
Migration - TBD	Test Plan, Use cases
Disaster Recovery Test Cases	Test Plan, Use cases
Final Test Report	• Test Management Tool ready for use • All test cases executed and execution statuses defined • All defects documented and loaded into the tool.

Following is the Responsibility Matrix (RACI: Responsible, Accountable, Consulted, and Informed) for D4.1 Deliverable Task for the Testing activities

Responsibility Description	Test Team	Participiant
Test Strategy Document	C,I	R,A
Test Strategy Document Sign Off	R,A	C,I

Master Test Plan Document	C,I	R,A
Master Test Plan Document Sign Off	R,A	C,I
FAT Testing	C,I	R,A
IOT/AFT Test Book Preparation	C,I	R,A
IOT/AFT Test Execution	C,I	R,A
Reliability & Non-Functional Test Book	C,I	R,A
Reliability & Non-Functional Test Execution	C,I	R,A
UAT Test Book Preparation	R,A	C,I
User Acceptance Execution	R,A	C,I
UAT Testing Support (Bug Fixing)	C,I	R,A
UAT Sign off	R,A	C,I

6.10. DEFECT MANAGEMENT

Effective process of defect management is crucial and can decide if the testing phase will be able to pass all tests in the given time frame.

- Defect management has to secure correct understanding and interpretation of Test Scenario results
- Define collection of necessary inputs (logs, traces, snapshots) for proper troubleshooting
- Manage list of opened defects and simplify and automate workflows and defect lifecycle
- Secure prompt reaction of developer/integrator
- Secure fast communication among involved project members (architects, developers, integrators, test designers, testers)
- Along with configuration and release management provide reliable and fast way to deploy fixed bugs
- Provide reports, KPIs and metrics to measure overall performance and efficiency

6.10.1. DEFECT SEVERITY

Severity	Description	Indicative Examples
Severity 1 **(Critical)**	Critical meaning that Customer cannot use the Customer Solution at all or that the error in the Customer Solution alone severely impacts Customer's operations and/or business. The situation has	Test environment not functioning and testing cannot proceed i.e. Log-in failure

Severity	Description	Indicative Examples
	one or more of the following characteristics: • Customer Solution Outage • Critical functionality in the Customer Solution is not available; Disruption of the functionality of the Customer Solution to the extent that it cannot be used at all.	
Severity 2 **(High)**	The entire application(s), components or business functionality will not work, but a business workaround is available. Testing is impacted, but can proceed. The business impact of the defect is high. This importance is based on factors such as the business units impacted, estimated number of impacted users, or if Customer regulatory compliance is compromised. Users are hindered from being able to utilise the system and/or their productivity is lowered.	Vital user-interface not working; however can perform function via native system interface. "Add" functionality works, but "Delete" functionality does not. "Edit" functionality does not work, but can use "Delete/Add" instead.
Severity 3 **(Medium)**	Medium impact meaning that the error in the Software causes loss of service which results in only a negligible or no business impact to Customer. The business impact of the defect is medium. The user is partially hindered from being able to utilise the system but also has some kind of a work around.	Help function does not work. Time-out failures on minor and infrequently used functionality.
Severity 4 **(Low)**	The function does not perform as expected, however business functionality is not compromised. The business impact of the issue is low.	Cosmetic, navigational or similar issues exist.

Severity	Description	Indicative Examples
	A defect that does not impair users from utilising the system.	

6.10.2. DEFECT PRIORITY

Priority	Description	Indicative Examples
Priority A (Critical)	A problem that means Testing cannot continue	Vital user interface not working, and no access to native system
Priority B (High)	A problem where testing of a significant application component or function cannot continue	Vital user-interface not working; however can perform function via native system interface
Priority C (Medium)	A problem that is not severe. The test cases for a functional matrix cannot be tested, but testing can continue in other areas of the function being tested	Failures with minor and infrequently used functionality
Priority D (Low)	A minor problem. Some of the function tested using a test case will not work as expected but testing can continue	Cosmetic, navigational or similar issues exist

6.10.3. DEFECT LIFE CYCLE

Defect Flow irrespective of Tool implementation is as below -

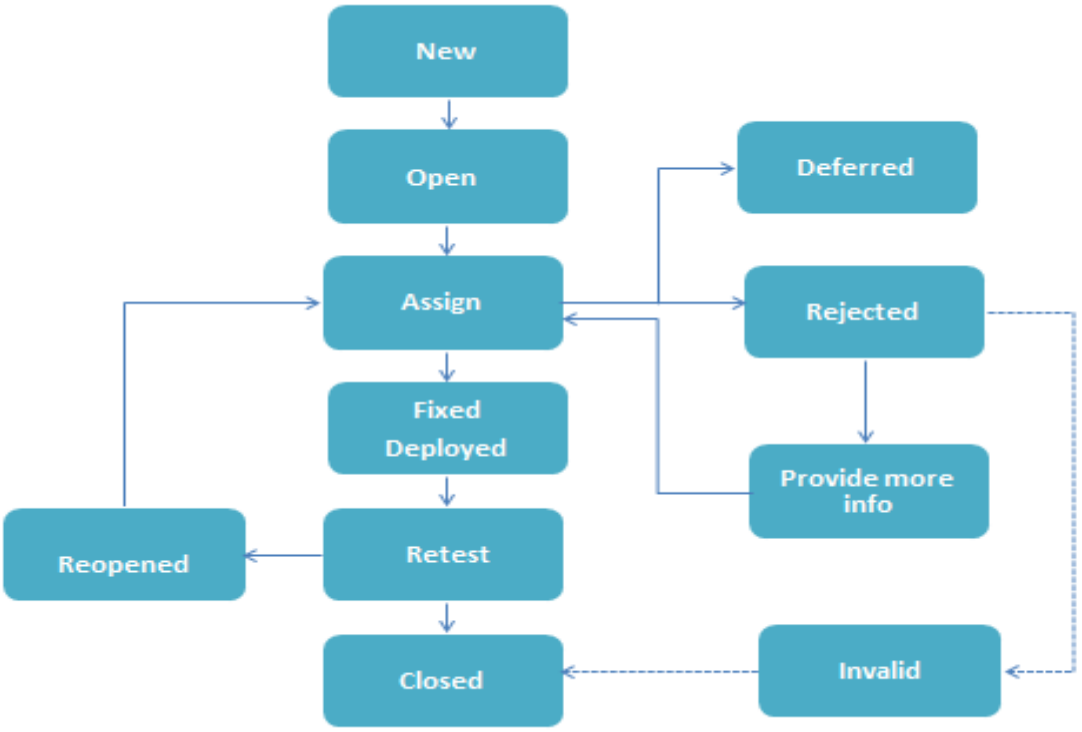


***Figure 4**: Defect Lifecycle*

- Tester to open the defect and assign to SPOC with status Open
- SPOC assigns to developer to analyze
- Developer can fix the defect (Fixed status), reject and add reason for that (Rejected – reasons could be Duplicated, Invalid with explanation) or put in Pending status when fix can't be provide (core fix pending, 3rd party pending)
- Defect manager will decide when fix will be installed and then move to Ready for Test once installed

The main difference between Softwer or Hardware Suppliers (Program Participients) and 5G!Drones 4.1 Deliverable Team tests in terms of defect life cycle control is the additional layer added. This layer presumes including all discovered defects in so-called Punch List. After they are reviewed and accepted by Supplier, they will follow standard life cycle workflow through bug tracking system. All status changes of the defect will be reflected in the Punch List. The closure of the defect is followed by a comment with the reason for closure.

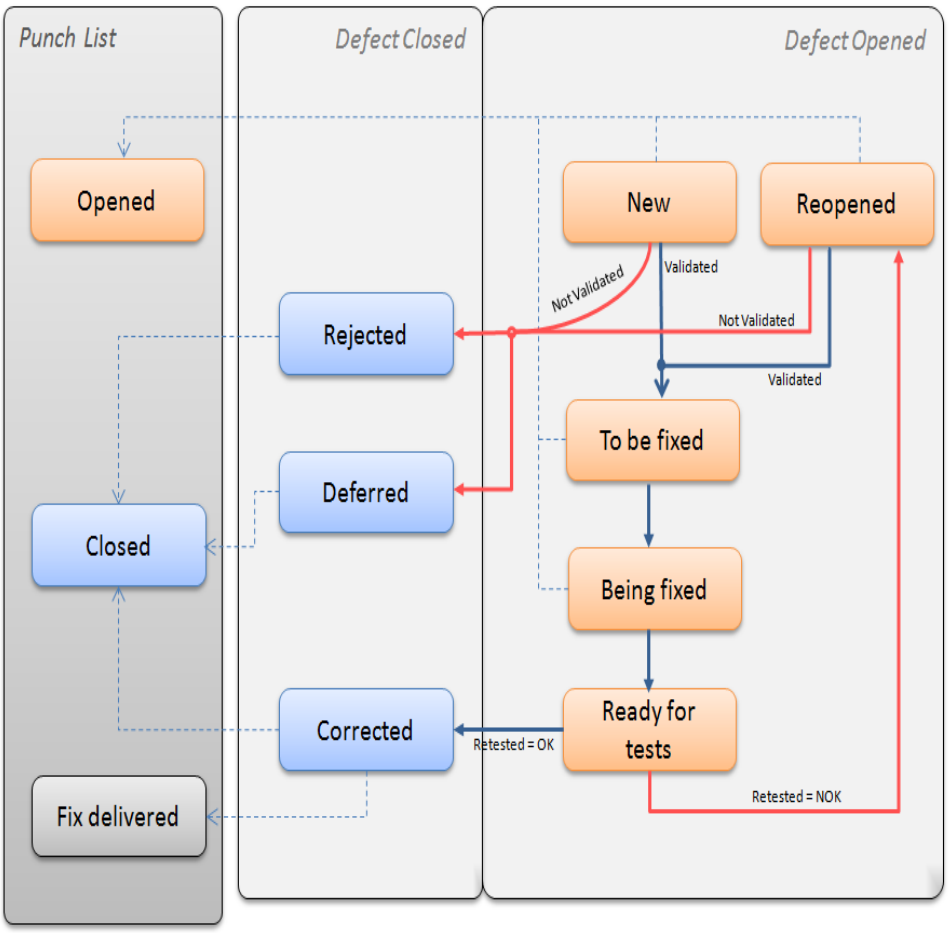


***Figure 5**: Punch List Flow*

As bug tracking system for the Punch-list will be used RedMine tool.

Tester re-tests the scenario and can reopen if test fail (SPOC again will assign to right developer) or change to closed status.

7. TESTING TOOLS

5G!Drones project Implementation primarily needs to have Test & Defect Management Tools for Quality Management.

7.1. TEST MANAGEMENT TOOL

For example It could be: Test Link (open source) or ALM HP (HP Proprietary) Test Management Tool used by DDR

7.2. DEFECT MANAGEMENT TOOL

Red mine is the Proprietary Test Management Tool (open source) used by DDR

8. TESTING TECHNIQUES

Testing techniques will be based on black-box techniques such as equivalence partitioning, boundary value analysis, state transition testing and use case testing. The black-box test design will define the test cases, test conditions and test data from the requirements.

Equivalence partitioning – inputs to the software or system are divided into groups that are expected to exhibit similar behavior, so they are likely to be processed in the same way. Equivalence partitions are to be defined for both valid data (values that should be accepted) and invalid data (values that should be rejected).

Boundary value analysis - Behavior at the edge of each equivalence partition. This technique is an extension of equivalence partitioning.

State Transition – the state of the system or object under test dependent on inputs or events that trigger state change (transitions). The states are separate, identifiable and finite in number.

Use case testing – test that are derived from use cases. Each use case has precondition which need to be met and post conditions which are observable results and final state of the system after the use case has been completed.

9. QUALITY CONSIDERATIONS

Process of Quality Gates has been incorporated as a Quality Control Procedure. The purpose of the Quality Gate is to define the criteria that are to be met before the FAT, IOT/AFT and UAT Test Phase commences. Three proposed Quality Gates for FAST Project -

- FAT - Entry Gate for FAT Test Phase
- IOT/AFT - Entry Gate for IOT/AFT Test Phase

The Checklist for the above Quality Gates will be captured in the QG Checklist.

Review Process: Any Deliverable Document will have two levels of review.

- Internal Review: As a first round of review, the document will be reviewed by D4.1 Task Leaders Team. Review comments will be captured in the review Template document and then accepted comments will be incorporated.
- 4.1Deliverable Review: Internally reviewed document will be shared with Team members for the comments. Review comments will be captured in the review Template document and then accepted comments will be incorporated and base lined after sign off.

9.1. TASK ITERATION POLICY (RETESTING, REGRESSION TESTING)

Any increase in retesting or iterations of the problem has a direct impact on the timing of the task of testing and completion dates respectively as a certain level of testing and phase (implementation / acceptance) and as a result in the project budget - need to identify criteria which are the basis for retesting tasks or for re-regression test functionality.

Any problem that requires repeat of one or more tests of the test cases causes an increase in the iterative time-consuming task. Retesting may be due to two main reasons:

Finding problems having All High / Critical / Showstopper severity and High / Critical priority necessitates additional regression tests covering related technological operations. Finding problems Low / Medium severity with Low / Medium priority involve only repeated retesting test, in which the problem is detected. In this case, the process is completely determined by
Defect Life Cycle

Unsatisfactory quality test: Can be initiated either by Supplier or Customer. Determined by, for example, clear deviation of the test log step-by-step from the test cases, if specified in the Test Report test actions ran by employee conducting the tests do not cover test cases explained in task.

9.2. DEVIATION POLICY

Changes in approved Master Test Plan (including documents of a lower level, for example, Level Test Plan), during the period for which D4.1 Task Leaders approval has already been given, may not be initiated without review and approval of other Deliverables for WP2 & WP3 except when necessary to eliminate apparent immediate hazards to the Subject. The initiator deviation (Test Manager level and higher) must submit and receive approval from the both WP2 & WP3 Task Leaders before initiating any changes to a testing procedure described in Master Test Plan.