D2.1 Setup and operation of 5G infrastructure

<table>
<thead>
<tr>
<th>Document Summary Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grant Agreement No</td>
</tr>
<tr>
<td>Full Title</td>
</tr>
<tr>
<td>Start Date</td>
</tr>
<tr>
<td>Project URL</td>
</tr>
<tr>
<td>Deliverable</td>
</tr>
<tr>
<td>Work Package</td>
</tr>
<tr>
<td>Contractual due date</td>
</tr>
<tr>
<td>Nature</td>
</tr>
<tr>
<td>Lead Beneficiary</td>
</tr>
<tr>
<td>Responsible Author</td>
</tr>
<tr>
<td>Contributions from</td>
</tr>
</tbody>
</table>

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Grant Agreement No 856691.
### Revision history (including peer reviewing & quality control)

<table>
<thead>
<tr>
<th>Version</th>
<th>Issue Date</th>
<th>% Complete</th>
<th>Changes</th>
<th>Contributor(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1.0</td>
<td>0</td>
<td>Initial Deliverable Structure</td>
<td>Fabrizio Moggio</td>
<td></td>
</tr>
<tr>
<td>V1.1</td>
<td>25%</td>
<td>Description of the integration of the LLs in 5G-EVE and 5G-VINNI.</td>
<td>Fabrizio Moggio, Silvia Canale, Min Xie, Christos Tranoris, Bader Mawasy</td>
<td></td>
</tr>
<tr>
<td>V1.2</td>
<td>13/07/2020</td>
<td>35%</td>
<td>Merge of chapters describing Northway and Patras</td>
<td>Fabrizio Moggio</td>
</tr>
<tr>
<td>V1.5</td>
<td>13/11/2020</td>
<td>40%</td>
<td>Chapter 3 update</td>
<td>Min Xie</td>
</tr>
<tr>
<td>V1.6</td>
<td>18/12/2020</td>
<td>50%</td>
<td>Chapter 4 revision, Chapter 5 contribution on LL2: Smart Energy</td>
<td>Silvia Canale</td>
</tr>
<tr>
<td>V1.6.2</td>
<td>10/05/2021</td>
<td>60%</td>
<td>Chapter 3.4 &amp; 4.3</td>
<td>Bader Mawasy</td>
</tr>
<tr>
<td>V1.6.3</td>
<td>04/06/2021</td>
<td>65%</td>
<td>Draft introduction</td>
<td>Fabrizio Moggio, Andrea Di Giglio</td>
</tr>
<tr>
<td>V1.6.4</td>
<td>12/08/2021</td>
<td>755%</td>
<td>Chapter 4 has been reviewed and integrated after completing the trials for Cycle 1 (July 2021)</td>
<td>Silvia Canale</td>
</tr>
<tr>
<td>V1.6.5</td>
<td>17/02/2022</td>
<td>80%</td>
<td>Updated Chapter4.3 after completing the trials for Cycle 1 (July 2021), CDSO API integrations</td>
<td>Nokia, Bader Mawasy</td>
</tr>
<tr>
<td>V1.6.6</td>
<td>30/03/2022,</td>
<td>85%</td>
<td>Executive summary draft</td>
<td>Fabrizio Moggio</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>V1.6.7</td>
<td>03/05/2022</td>
<td>85%</td>
<td>Executive summary update</td>
<td>Silvia Canale</td>
</tr>
<tr>
<td>V1.6.8</td>
<td>31/05/2022</td>
<td>85%</td>
<td>General review before sending to EC</td>
<td>Fabrizio Moggio</td>
</tr>
<tr>
<td>V1.6.9</td>
<td>15/11/2022</td>
<td>90%</td>
<td>Chapter 4 and 5(LL2) have been reviewed and integrated after completing the trials for Cycle 2 (July 2022). Annex 8.1 completed. Chapter 5(LL4) Media and Entertainment</td>
<td>Silvia Canale, Ran Rahav</td>
</tr>
<tr>
<td>V1.7</td>
<td>28/11/2022</td>
<td>90%</td>
<td>Fixed LL4 subchapters numbering 5.4.x</td>
<td>Fabrizio Moggio</td>
</tr>
<tr>
<td>V1.7.1</td>
<td>30/11/2022</td>
<td>90%</td>
<td>LL4 subchapters final review, Conclusions chapter</td>
<td>Ran Rahav, Fabrizio Moggio</td>
</tr>
</tbody>
</table>

1 According to 5G Solutions Quality Assurance Process:
1 month after the Task started: Deliverable outline and structure
3 months before Deliverable’s Due Date: 50% should be complete
2 months before Deliverable’s Due Date: 80% should be complete
1 months before Deliverable’s Due Date: close to 100%. At this stage it sent for review by 2 peer reviewers
Submission month: All required changes by Peer Reviewers have been applied, and goes for final review by the Quality Manager, before submitted
D2.1. Setup and operation of 5G infrastructure

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Description</th>
<th>Author(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>v1.7.2</td>
<td>05/12/2022</td>
<td>Conclusions chapter</td>
<td>Silvia Canale</td>
</tr>
<tr>
<td>v1.7.3</td>
<td>15/12/2022</td>
<td>Chapter 3 and 5 renaming and chapter 5 restructuring</td>
<td>Fabrizio Moggio</td>
</tr>
<tr>
<td>v1.7.4</td>
<td>09/01/2023</td>
<td>Chapter 5 completed. Indexing of figures and tables.</td>
<td>Silvia Canale, Ran Rahav, Fabrizio Moggio</td>
</tr>
<tr>
<td>v1.7.5</td>
<td>11/01/2023</td>
<td>Chapter 1. Executive summary for 5G-VINNI</td>
<td>Panagiotis Papaioannou</td>
</tr>
<tr>
<td>v1.7.6</td>
<td>12/01/2023</td>
<td>Review of Table 1</td>
<td>Andrea di Giglio</td>
</tr>
<tr>
<td>v1.7.7</td>
<td>16/01/2023</td>
<td>Review of chapter 4.1</td>
<td>Silvia Canale</td>
</tr>
<tr>
<td>V1.7.8</td>
<td>17/01/2023</td>
<td>100% Review of chapter 5.1. All comments in the document deleted. Chapter 7 review.</td>
<td>Panagiotis Papaioannou, Fabrizio Moggio</td>
</tr>
<tr>
<td>V1.7.9</td>
<td>18/01/2023</td>
<td>100% Quality Review</td>
<td>Christos Skoufis</td>
</tr>
</tbody>
</table>

Disclaimer

The content of the publication herein is the sole responsibility of the publishers and it does not necessarily represent the views expressed by the European Commission or its services.

While the information contained in the documents is believed to be accurate, the authors(s) or any other participant in the 5G Solutions consortium make no warranty of any kind with regard to this material including, but not limited to the implied warranties of merchantability and fitness for a particular purpose.

Neither the 5G Solutions Consortium nor any of its members, their officers, employees or agents shall be responsible or liable in negligence or otherwise howsoever in respect of any inaccuracy or omission herein.

Without derogating from the generality of the foregoing neither the 5G Solutions Consortium nor any of its members, their officers, employees or agents shall be liable for any direct or indirect or consequential loss or damage caused by or arising from any information advice or inaccuracy or omission herein.

Copyright message

© 5G Solutions Consortium, 2019-2022. This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. Reproduction is authorised provided the source is acknowledged.

Table of Contents

1 Executive Summary 7
2 Introduction 8
   2.1 Mapping Projects’ Outputs 8
   2.2 Deliverable Overview and Report Structure 9
3 5G-VINNI as supporting platform 10
   3.1 Integration process description 10
   3.2 5G-VINNI Portal 13
   3.3 UC deployment process description 14
      3.3.1 Deployment process overview 14
List of Figures

- Figure 1: General integration process ................................................................. 10
- Figure 2: Integrations between 5G Solutions and 5G-VINNI ....................................... 11
- Figure 3: 5G-VINNI portal description ................................................................... 14
- Figure 4: 5G-VINNI portal deployment in Norway and Patras .................................. 14
- Figure 5: TaaS with OpenTAP approach for vertical testing (Keysight Technologies, 2020) ............................................................................. 15
- Figure 6: Experiment ordering and executions (as of 5G-VINNI Norway) ................. 16
- Figure 7: 5G-VINNI SB structure .......................................................................... 21
Figure 8: Examples of 5G-VINNI SB attributes ................................................................. 21
Figure 9: UML representation of the 5G-VINNI SB .......................................................... 22
Figure 10: Testing and Monitoring tools that will be offered in 5G-VINNI ............................ 26
Figure 11: How customers order and consume testing services of 5G-VINNI ............................ 28
Figure 12: 5G-EVE Portal landing page ................................................................................. 33
Figure 13: 5G-EVE Portal - Section “Design Experiment” ......................................................... 35
Figure 14: 5G-EVE portal - Section “Descriptors” .................................................................. 35
Figure 15: “Activate” sequence diagram ................................................................................. 39
Figure 16: “Terminate” sequence diagram ............................................................................. 43
Figure 17: “Get Status” sequence diagram ............................................................................. 45
Figure 18: 5G-EVE Portal, Network KPIs Configuration .......................................................... 47
Figure 19: 5G Solutions KPI VS ........................................................................................... 48
Figure 20: Trial execution flow ............................................................................................... 50
Figure 21: Example of VNF use as part of UC4.5 ................................................................... 51

List of Tables
Table 1: Adherence to 5G Solutions GA Deliverable & Tasks Descriptions ................................. 8
Table 2: Three types of integration with 5G-VINNI .................................................................. 11
Table 3: Actors in 5G-VINNI and 5G Solutions participating in the integration .............................. 12
Table 4: Deployment process steps and participating actors ....................................................... 17
Table 5: Service exposure defined in 5G-VINNI .................................................................... 23
Table 6: Example KPIs collected by 5G-VINNI ...................................................................... 28
Table 7: 5G-EVE role and related 5G Solutions players ............................................................ 30
Table 8: 5G-EVE roles used in 5G Solutions ......................................................................... 34
Table 9: 5G-EVE login APIs ................................................................................................. 38
Table 10: Deploy API request Model ..................................................................................... 40
Table 11: “Execute” API request Model ............................................................................... 41
Table 12: “Terminate” API request Model ............................................................................. 44
Table 13: “Delete” API request Model .................................................................................. 44
Table 14: “Get Status” API request Model ........................................................................... 46
Table 15: Reference of Network KPIs in Cycle 1 ..................................................................... 46
Table 16: List of Vertical Service KPIs for all TCs .................................................................. 46
Table 17: Configuration of uplink-oriented slice for UC 4.3 ..................................................... 50
Table 18: Services used in each UC ....................................................................................... 51
### Glossary of terms and abbreviations used

<table>
<thead>
<tr>
<th>Abbreviation / Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LL</td>
<td>Living Lab</td>
</tr>
<tr>
<td>UC</td>
<td>Use Case</td>
</tr>
<tr>
<td>CDSO</td>
<td>Cross Domain Service Orchestration</td>
</tr>
<tr>
<td>SO</td>
<td>Service Orchestrator</td>
</tr>
<tr>
<td>TMF</td>
<td>TeleManagement Forum</td>
</tr>
<tr>
<td>TaaS</td>
<td>Testing-as-a-Service</td>
</tr>
<tr>
<td>MaaS</td>
<td>Monitoring-as-a-Service</td>
</tr>
<tr>
<td>NaaS</td>
<td>Network Slice as a Service</td>
</tr>
<tr>
<td>KPI</td>
<td>Key Performance Indicator</td>
</tr>
<tr>
<td>DB</td>
<td>Database</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>UE</td>
<td>User Equipment</td>
</tr>
<tr>
<td>LCM</td>
<td>Life Cycle Management</td>
</tr>
<tr>
<td>NS</td>
<td>Network Slice</td>
</tr>
<tr>
<td>VNF</td>
<td>Virtual Network Function</td>
</tr>
<tr>
<td>NFVO</td>
<td>Network Function Virtualization Orchestrator</td>
</tr>
<tr>
<td>NSD</td>
<td>Network Service Descriptor</td>
</tr>
<tr>
<td>eMBB</td>
<td>Enhanced Mobile Broadband</td>
</tr>
<tr>
<td>URLLC</td>
<td>Ultra Reliable Low Latency Communications</td>
</tr>
<tr>
<td>mIoT</td>
<td>Mobile Internet of Things</td>
</tr>
<tr>
<td>SST</td>
<td>Slice/Service Type</td>
</tr>
</tbody>
</table>
1 Executive Summary

This deliverable describes how the 5G infrastructure platforms (5G testbeds), namely 5G-VINNI and 5G-EVE, are used to support 5G Solutions. The focus of the deliverable is on how the Living Labs are integrated (such as on-boarding and deployment) onto and operated with those platforms.

The usage of 5G-VINNI and 5G-EVE platforms and the available tools and APIs are described. The adopted integration models and procedures are defined with roles and responsibilities. The integration process is defined in terms of service design, service preparation, on-boarding procedures, artefacts, and components management using the tools provided by the platforms. There are three overall platform instances, the 5G-EVE in Turin facilitated by TIM, the 5G-VINNI in Patras, facilitated by Univ. of Patras (VINNI-UoP), and the 5G-VINNI Norway (multiple sites) facilitated by Telenor (VINNI-NOR).

5G-VINNI facility in Patras, provides a laboratory in which various verticals can deploy and test their UCs end-to-end. The UCs can be developed either by UoP or the UC owners and the required artifacts can be uploaded in the appropriate portal deployed in Patras (Openslice). This allows fully automated execution of experiment including deployment, life-cycle management, configuration monitoring and tear down of deployed services. In Patras facility LL4 UCs were supported mainly focusing on media-oriented services. The facility was fully integrated with other partners services like CDSO which allowed communication and deployment and with AppArt providing the VS with all the required monitoring data. In a compatible manner 5G-VINNI Norway is offering and using similar tools and procedures.

The 5G-EVE facility offered a deployment framework based on web and application interfaces favouring a straightforward experiment preparation, design, deployment, and execution, with the possibility of integrating third party services. The experiments that are developed in the three use cases from the Living Lab Smart Energy in 5G Solutions have been easily prepared and designed by the 5G-EVE web portal, as well as the deployment and, after the experiment scheduling, the execution. Some issues have been encountered during the deployment phase due to the dynamic configuration of the entire chain of VNFs, but they were solved by re-configuring the experiment. As concerns third party services, in 5G Solutions the Cross Domain Service Orchestrator and the KPI Visualization System have been "integrated" in the sense that the two platforms were able to interact with suitable service in the 5G-EVE facility via the exposed Northbound APIs.
2 Introduction

The objective of this deliverable is to describe the integration process of the Living Labs into the supporting platforms from 5G-VINNI and 5G-EVE. This document describes, for each of the supporting platforms, the integration process and used tools. The supporting platforms provide a Web Portal that is described in this document. The goal is to address the usage of the Portals in the scope of the 5G Solutions Project. The deliverable describes the deployment process for the different Living Labs and the usage of the APIs to automate part of the process. The available APIs from the supporting platforms are described and their usage is documented. Examples on Living Labs actual deployment are provided.

2.1 Mapping Projects’ Outputs

Purpose of this section is to map 5G Solutions Grand Agreement commitments, both within the formal Deliverable and Task description, against the project’s respective outputs and work performed.

Table 1: Adherence to 5G Solutions GA Deliverable & Tasks Descriptions

<table>
<thead>
<tr>
<th>5G Solutions Task</th>
<th>Respective Document Chapter(s)</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task 2.1 - Interfacing and setup of ICT-17-2018 5G E2E facilities to enable LL operations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>This task entails the interfacing of the 5G-EVE Turin, 5G-VINNI Norway and 5G-VINNI Patra ICT-17 facilities with the Cross-Domain Service Orchestrator (CDSO) including configuration and setup in order to ensure the end-to-end readiness of the 5G infrastructure and measurement portals for the field trials. This task also includes the installation and the configuration of 3 remote 5G RAN stations by UOP, i.e. 2 in Ireland (one in IBM campus and the other in GLAN factory) plus another 5G RAN station in PBGS factory in Brussels. Localisation is necessary in order to perform real measurements and validate the performance capabilities of 5G especially for the Factories of the Future use cases, which required ultra-low latencies and high reliability. Two more 5G remote nodes will also be installed and setup by TNOR in Norway in order to enable the validation of the measurements for Industry 4.0, Smart Cities and Smart Ports use cases. For the procurement of these 5 additional 5G RAN nodes, budget has been allocated to both TNOR and UOP. Furthermore, the intention of TNOR is to enable 2 additional 5G RAN nodes for free in locations in Trondheim, Norway, to strengthen the Smart City IoT-based use cases.</td>
<td></td>
<td>Chapter 3 deals with 5G-VINNI, chapter 4 is about 5G-EVE and the deployment for living labs is reported in chapter 5.</td>
</tr>
</tbody>
</table>
2.1 Setup and operation of 5G infrastructure

The document describes the integration process of the Living Labs into the supporting platforms from 5G-VINNI and 5G-EVE, illustrating, for each of the supporting platforms, the integration process and used tools.

2.2 Deliverable Overview and Report Structure

This deliverable has been structured in such a way so that the reader can get a clear picture of the setup and operation of 5G infrastructure, based on 5G-VINNI and 5G-EVE. This deliverable is structured in two main sections, one describing the integration with 5G-VINNI and one describing the integration with 5G-EVE.

In more details, the complete structure of the document is:

- Chapter 2 “Introduction” introduces the reader into the document and reports how this deliverable fits in the more complex project framework.
- Chapter 3 “5G-VINNI as supporting platform” describes in detail the deployment process and the interaction with 5G-VINNI orchestrator
- Chapter 4 “5G-EVE as supporting platform” describes in detail the integration process, the usage of 5G-EVE portal and the interaction with the Project orchestrator.
- Chapter 5 outlines the relevant insight about the deployment of the 5G Solutions living labs, based on the platforms provided by ICT-17 projects
- Finally, a section of Conclusions (section 6) summarises the main outcomes
3 5G-VINNI as supporting platform

This section describes the general process of deploying Living Labs (LLs) in the 5G-VINNI facility sites in Norway (Oslo) and Greece (Patras), respectively. Section 3.1 briefly introduces the integration process applied to both Norway and Greece facility. Section 3.2 presents the web portal based on OpenSlice. Section 3.3 provides details on the actual LL deployment for the Norway facility and Greece facilities. Section 3.4 and 3.5 give details on the integration with CDSO and KPI-VS, respectively.

3.1 Integration process description

The 5G-VINNI project provides a network platform and a set of tailored network slices in the form of Network Slice as a Service (NSaaS) exposed to verticals (e.g., LLs in 5G Solutions) for testing and validating their use cases (UCs) that are developed by the vertical and deployed in the 5G-VINNI platform. An integration process takes place for LLs to design, develop, deploy their services, and run experiments for testing and validation in a DevOps way.

As shown in Figure 1, the integration between 5G-VINNI and 5G Solutions includes three parts:

1. Integrating the 5G Solutions LLs with the 5G network provisioned by 5G-VINNI for the 5G Solutions LLs to use the 5G network and serve their UCs;
2. Integrating the 5G Solutions CDSO with the 5G-VINNI E2E Service Orchestration (SO) of each facility site so that 5G Solutions CDSO can order and manage 5G network services offered by 5G-VINNI;
3. Integrating the 5G Solutions LLs, CDSO, and KPI-VS with the testing platform Testing-as-a-Service (TaaS) of 5G-VINNI:
   a. LLs design test cases based on the TaaS system
   b. CDSO orders and manages the test cases through the TaaS system
   c. KPI-VS retrieves testing and monitoring data collected from the TaaS system via a DB

Table 2 summarizes the three types of integrations and the corresponding participants from the 5G-VINNI and 5G Solutions side, respectively. The first and foremost step of integration is the interconnections between these participating actors/components. Figure 2 illustrates how the 5G-VINNI and 5G Solutions components are connected. Note that CDSO is hosted by the 5G-VINNI Norway facility. Therefore, the 5G-VINNI Norway facility is also responsible for providing and maintaining the connectivity between CDSO and other 5G Solutions
components, like KPI-VS and LL UC components. Detailed integration processes for CDSO and KPI-VS are described in Section 3.4 and 3.5, respectively.

Table 2: Three types of integration with 5G-VINNI

<table>
<thead>
<tr>
<th>Objective</th>
<th>Description</th>
<th>Integration type</th>
<th>5G Solutions actor(s)</th>
<th>5G-VINNI actor(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NSaaS provisioning</td>
<td>5G-VINNI provisions network slicing services to 5G Solutions UCs</td>
<td>5G network and service integration</td>
<td>UC owners</td>
<td>- E2E SO - MANO</td>
</tr>
<tr>
<td>Orchestration</td>
<td>5G Solutions CDSO orchestrates and manages UC services via 5G-VINNI</td>
<td>Service orchestration integration</td>
<td>- UC owners CDSO</td>
<td>E2E SO</td>
</tr>
<tr>
<td>Testing and monitoring</td>
<td>5G Solutions CDSO manages experiments run in the 5G-VINNI TaaS platform and 5G Solutions KPI-VS visualizes the data collected from the TaaS</td>
<td>TaaS integration</td>
<td>- UC owners CDSO - KPI-VS</td>
<td>- TaaS - Network-KPI DB</td>
</tr>
</tbody>
</table>

Note that the 5G-VINNI Patras facility uses OSM for MANO, which supports monitoring of infrastructure and VNFS. The monitoring data can also be retrieved by KPI-VS if KPI-VS is integrated with the OSM (or E2E-SO) at the Patras facility site.

The integration process is complex and involves several actors from 5G-VINNI and 5G Solutions. To prepare for the integration, we first summarize the role and responsibilities of participating actors in Table 3, which are reflected in the integration process later.
<table>
<thead>
<tr>
<th>Project</th>
<th>Actor</th>
<th>Description</th>
<th>Responsibilities</th>
</tr>
</thead>
</table>
| 5G Solutions     | UC owners      | Actor with the knowledge of the service to run, the experiments to execute, and the SLA to be required | - Design the service blueprint used by CDSO to order the NSaaS service  
- Prepare, design, and develop service components that are not part of the 5G services offered by 5G-VINNI by default, e.g., VNFs or application functions, UE devices, etc.  
- Integrate its own service components with CDSO and/or 5G-VINNI MANO  
- Prepare and design the test cases for experiments to collect network KPIs from 5G-VINNI TaaS  
- Prepare, design, and develop tools of collecting application KPIs  
- Integrate the application-KPI collection tools with CDSO and KPI-VS |
| CDSO             |                | Actor with the capability of orchestrating and managing the services offered to UC owners by 5G-VINNI and the experiments to be executed in the 5G-VINNI testing platform TaaS from the 5G Solutions perspective | - Integrate with the E2E-SO of 5G-VINNI  
- Order, start, and stop services for UC owners based on the service blueprint specified by the UC owners  
- Start and stop experiments for each test case of UC owners  
- Integrate with the TaaS of 5G-VINNI  
- Order, start, and stop test cases/experiments for UC owners based on the test scripts defined by the UC owners  
- Integrate with the KPI-VS of 5G Solutions to synchronize on the test cases |
| KPI-VS           |                | Actor with the capability of retrieving testing and monitoring data (e.g., network KPIs) produced by the 5G-VINNI testing platform TaaS | - Integrate with CDSO to synchronize on the test cases of each UC owner  
- Retrieve network KPIs from the network-KPI DB of 5G-VINNI based on the test cases of each UC owner  
- Retrieve application KPIs from the UC application-KPI collection tools based on the test cases of each UC owner |
| 5G-VINNI         | E2E-SO         | Actor with the capability of orchestrating and managing the network slicing services provisioned to 5G Solutions UC owners and interacting with CDSO for slicing service offering | - Integrate with CDSO to allow CDSO for ordering, starting and stopping NSaaS  
- Execute the requests from CDSO to order, start, and stop NSaaS  
- Exchange information of operating network slicing services with CDSO |
|                  | TaaS           | Actor/service with the capability of automating and unifying tests for 5G networks | - Integrate with CDSO to allow CDSO for ordering, starting and stopping test cases  
- Execute the requests from CDSO to order, start, and stop test cases by controlling and scheduling them  
- Integrate with Network-KPI DB to send testing data |
### 3.2 5G-VINNI Portal

A **Web Portal (customer portal)** based on OpenSlice is developed in 5GVINNI for vertical customers to define, request and order the network slicing services and (optional) experiment services from 5G-VINNI. It is developed to expose network slicing services to vertical customers via TMF Open APIs and allow customers to directly trigger operations related to NSaaS, such as service ordering, service catalog management and inventory management (as shown in Figure 3). In each facility site, a service orchestrator (SO) is deployed to implement the 3GPP network slice management functionality. SO is connected with a NFVO responsible for network orchestration via ETSI SOL005, in the form of RESTful APIs.

Vertical customers can also directly interact with the SO via TMF Open APIs and receive services like service ordering and service catalog/inventory management. In both cases, vertical customers can i) directly select a SB available from the Customer Service Catalogue; or ii) create its own SB and add to the Customer Service Catalogue as one catalog item. Once these templates are added to the Customer Service Catalogue, the vertical customers can select and order a specific catalog item to deploy/activate their services. Service ordering is implemented via TMF OpenAPI service order.

OpenSlice is used to expose the Customer Service Catalogue. OpenSlice acts as a customer facing service (CFS) to verticals and offers TMF OpenAPIs for service ordering and catalog management, similar to the Web Portal with easy access. Specifically, the following services are available for customers:

- Create a service item (service specification)
- Access to the service catalogue
- Remove a service item
- Publish service/resource candidate to the catalogue
- Discover service/resource candidate from another service catalogue

Currently in 5G-VINNI, both the Norway and Patras facility site deploys OpenSlice although they have different E2E-SO and NFVO: Norway has Nokia’s NOrC (upgraded from FlowOne) as E2E-SO and NCOM (upgraded from CBND) as NFVO whereas Patras uses OSM for NFVO and develops an OSOM to orchestrate E2E services (Figure 4). Currently in Patras infrastructure both SOL005 and SOL006 are supported since OSMv10.1 is also deployed. Moreover, the vertical customers like 5G Solutions can order services from one facility site but offered by another facility site, e.g., CDSO can order a slice service provisioned by the Norway facility via the OpenSlice of the Patras facility. In this way, the integration between CDSO and 5G-VINNI facility site is simplified as the same integration flows developed for the Patras facility site can be reused for the Norway site.
### 3.3 UC deployment process description

#### 3.3.1 Deployment process overview

The UC deployments and experiments in 5G-VINNI includes the following steps/phases, which may be continuously iterated for optimization.

**Step 0**: preparation. Both UCs and 5G-VINNI facility sites participate to prepare the slicing services that are about to be deployed and onboarding. The preparation process includes:

- Prepare the 5G service: design the service in two parts.
  - First, decide the components such as the end devices, additional servers or functions attached to the network, additional 3rd-party VNFs that will be part of the network slice. For NFs that will be part of the network slice, UC owners are responsible for developing the VNF descriptor (VNFD) and the
corresponding VNF package (with support by the facility owner as needed). Then UC owners and the 5G-VINNI facility jointly design the network service descriptor (NSD) to contain the newly added VNF. Furthermore, the 3rd-party VNFs need to be validated before being deployed. If UCs expect edge clouds, then additional work is required to plan for the extra resources and procurement.

b. Second, decide how these components are connected and then set up the connections. Many UCs have application functions or servers in clouds or central office that should communicate with the end devices. The connections between such functions/servers and the 5G-VINNI core network should be established and tested before the corresponding slice service is activated.

- Prepare the experiments and test cases: each UC may run several test cases in the Testing platform (e.g. TaaS in 5G-VINNI Norway) to experiment different scenarios, settings, and KPIs. The preparation includes:
  a. Decide the testing scenarios, including the network components to be tested (the E2E network or special network domain like RAN or core network), the KPIs of interest, etc.
  b. Design the test scripts for each testing scenario. TaaS is a way to unify the testing functionalities for 5G and aims to automate tests. It requires UC owners to prepare test scripts and upload to the TaaS repository for execution (Figure 5).
  c. Select the testing tools. TaaS is built on top of OpenTAP, an open-source test automation platform. OpenTAP can accept various testing tools even though they are products of the OpenTAP developer Keysight. Note that if the selected tool has not been supported by TaaS, the plug-ins need to be developed and integrated with TaaS (Figure 5).
  d. Deploy the testing tools to the testing points. Usually, the testing tools are software that need to be installed at devices or infrastructure where the measurement data will be collected.

![Figure 5: TaaS with OpenTAP approach for vertical testing (Keysight Technologies, 2020)](image)

**Step 1**: Service ordering. Once the service is prepared, it is added into the service catalog of the 5G-VINNI E2E-SO that exposes the service catalog to OpenSlice, an open-source OSS with a web portal UI. CDSO can browse and order services from the OpenSlice via APIs. Note that if the UC orders services that do not belong the default slices provisioned by 5G-VINNI, a validation process will run internally in 5G-VINNI, prior to activating the services, in order to ensure that the slice can operate properly.

**Step 2**: Service activation and operation. The service order is passed down in 5G-VINNI from E2E-SO to MANO to activate the ordered slice service. Then the UC end devices are subscribed to the slice and allowed to use the slicing service. During this process, the UC may consider exchanging subscriber information with the 5G-VINNI facility site (via CDSO), such as the IP address assigned to the end devices. The exchanged information is used by UCs to establish E2E connections between the end devices (one end) and application functions/servers (the other end).

**Step 3**: Experiment ordering. After the service is activated and UC applications are up running, UC can request CDSO to order experiment services from TaaS. The prepared test scripts for the ordered test cases will be loaded and then executed on OpenTAP.
**Step 4:** Experiment activation and monitoring. OpenTAP activates and configures the deployed testing tools to test the target SUTs and collect measurements. The target SUTs could be the 5G network provisioned by 5G-VINNI, a particular network domain, or NFVI (see examples in Table 6: Example KPIs collected by 5G-VINNI).

**Step 5:** Retrieval and analysis of experiment results. The experiments usually run for a limited time period, e.g., few minutes.

- After the experiments finish, the In the 5G-VINNI Norway facility site, the collected measurements are stored in a Postgres SQL DB (network KPI DB in Figure 6). Note that if the UCs plan to use TaaS for their application KPIs, they design the corresponding test scripts and the collected application KPIs are also stored in the Postgres SQL DB.
- During the experiment execution in Patras facility, all related metrics (application, network, infrastructure etc.) are gathered and stored in a Prometheus TSDB. This Prometheus is available during the experiment duration to any interested parties like AppArt VS system, which can gather these metrics through appropriate endpoints provided by Patras infrastructure. Depending on the experiment the gathered metrics can be available for period of time after the experiment has been concluded.

![Figure 6: Experiment ordering and executions](image)

**Step 6:** service termination. If the UC no longer needs the service, CDSO requests to terminate the service so that the 5G-VINNI facility E2E-SO can execute service termination and release the corresponding resources, especially for slices dedicated to the UC.

The actions of each step are summarized in Table 4. Further details on some service deployment steps are presented in the following sections. See also Deliverable D1.4B [4] for more details on technical as well as business validation methodology, which should be seen as complementary to the stages, models and processes outlined here.
<table>
<thead>
<tr>
<th>Step</th>
<th>Target</th>
<th>Objective</th>
<th>5GSOLUTIONS participants</th>
<th>5GVINNI participants</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>LL or UC</td>
<td>CDSO</td>
</tr>
</tbody>
</table>
| 0    | Service | Service design | • Design the UC architecture and topology  
|      |        |           |   |   |       |       |   |
|      |        |           | • Fill in the 5G-VINNI SB | ✓ |   | ✓ |
| 0    | Test   | TC design | • Design the TC with SUT, tools, KPIs, and measurement points | ✓ |   | ✓ |
| 0    | Service | Service preparation | • Prepare VNFs and NSDs (TOSCA) if 3rd party VNFs are needed  
|      |        |           |   |   |       |       |   |
|      |        |           | • Prepare the slice template (per FlowOne)  
|      |        |           |   |   |       |       |   |
|      |        |           | • Plan and allocate resource  
|      |        |           |   |   |       |       |   |
|      |        |           | • Decide the resource placement | ✓ |   | ✓ |
| 0    | Test   | TC preparation | • Prepare the test scripts  
|      |        |           |   |   |       |       |   |
|      |        |           | • Validate the test scripts, e.g., syntax | ✓ |   | ✓ |
| 0    | Service | Service onboarding/ deployment | • Onboard and validate VNFs and NSDs  
|      |        |           |   |   |       |       |   |
|      |        |           | • Test/validate the interconnections  
|      |        |           |   |   |       |       |   |
|      |        |           | • Instantiate the slice | ✓ | ✓ | ✓ |
| 0    | Test   | TC onboarding/ deployment | • Onboard the TC script to the test catalogue  
|      |        |           |   |   |       |       |   |
|      |        |           | • Deploy the testing tools  
<p>| | | | | | | | |
|      |        |           |   |   |       |       |   |
|      |        |           | • Validate the TC, e.g., KPI generation and retrieval | ✓ | ✓ | ✓ |
| 0    | Service | Service (pre-activation) validation | • Validate the entire network slice instance | ✓ | ✓ | ✓ |
| 1    | Service &amp; Orchestration | Publicatio n of service catalogue | • Publish the validated service to the CFS catalogue | ✓ |   | ✓ |</p>
<table>
<thead>
<tr>
<th></th>
<th>Test &amp; Orchestrati on</th>
<th>Publication of TC catalogue</th>
<th>● Publish the TC to the test catalogue</th>
<th>✔</th>
<th></th>
<th>✔</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Service &amp; Orchestrati on</td>
<td>Service ordering</td>
<td>● UC selects and orders services ● CDSO places orders to E2ESO ● E2ESO initiates the ordering process</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>2</td>
<td>Service &amp; Orchestrati on</td>
<td>Service activation</td>
<td>● E2ESO configures and activates the service ● E2ESO reports the activation results to CDSO ● Service is up running</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Service &amp; Orchestrati on</td>
<td>UE attachments</td>
<td>● Onboard and attach the UE (5G SIM card) to the associated slice</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>2</td>
<td>Service &amp; Orchestrati on</td>
<td>UE activation</td>
<td>● UEs are activated and start generate application traffic</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>3</td>
<td>Test &amp; Orchestrati on</td>
<td>TC ordering</td>
<td>● UC sends TC orders to CDSO ● CDSO places orders to TaaS (or MaaS) ● TaaS or MaaS initiates TC ordering</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Test &amp; Orchestrati on</td>
<td>TC activation</td>
<td>● TaaS or MaaS activates the TC experiments ● TaaS or MaaS reports the experiment status to CDSO ● CDSO notifies KPI visualization (ready to collect data)</td>
<td></td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>5</td>
<td>Test &amp; KPI visualizatio n</td>
<td>KPI collection</td>
<td>● KPI visualization system sends request to TaaS/MaaS for data retrieval ● KPI visualization system starts retrieve data</td>
<td></td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Test &amp; Orchestrati on</td>
<td>TC termination</td>
<td>● TaaS/MaaS terminates the TC at the end of the experiment (automatically) ● (or) UC requests to terminate the TC ● TaaS/MaaS releases the resource (testing tools) and notifies CDSO</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Service &amp; Orchestrating</td>
<td>Service Decommissioning</td>
<td>UC requests to terminate the service via CDSO</td>
<td>CDSO sends the request to E2ESO</td>
<td>E2ESO terminates and decommissioning the service</td>
<td>E2ESO notifies CDSO of the decommissioning outcome</td>
<td>✓</td>
</tr>
</tbody>
</table>
3.3.2 Service Design, Preparation, and Onboarding

In 5G-VINNI, a lifecycle management (LCM) process of network slicing services has been planned and designed, including the service design, preparation and deployment. The process needs facilitations and support from a complete orchestration system, from customer-facing service orchestration (e.g., OSS/BSS) to service- and resource-facing service orchestration, as well as the integration between these orchestration components. In practice, some of the orchestration components have not been implemented, which caused delays in the actual service deployment of 5G-SOLUTIONS UCs. However, in the long-term, the following process is how/what we expect to onboard network services and application services for vertical customers.

3.3.2.1 Service Design

Service design mainly involves the design and specification of SB, in compliance with 5G-VINNI SB. 5G-VINNI SB is a model-based service template specifying a network slice service. It is used by 5G-VINNI as a reference to conduct service management at both commissioning (instantiation and deployment?) and run-time (operations) phase.

As a structured document, the 5G-VINNI SB contains a complete service description, including information on the service type and topology and the expected performance and behaviours. As displayed in Figure 7, four categories are defined in 5G-VINNI SB:

1. Slice Service type (SST): compatible with the 3GPP slice type definition (eMBB, URLLC, and mIoT), with an addition type SST4, customised slice service to allow for other slice types, e.g., V2X. For example, the Norway facility supports three SSTs: eMBB, mIoT and URLLC as well as the user equipment B2B2C CFS for end users once the corresponding slicing service is activated.

2. Service topology: represents the default topology of the specified SST. It defines how a slice is constructed from a logical point of view. It defines i) what service components (SC) (VNFs or NS) are enclosed; ii) how these SCs are interconnected. Each 5G-VINNI facility site provides a set of default topologies, on top of which the vertical customers can define more topologies, by flexibly extending the default topologies, e.g., adding their own VNFs and applications at the given attaching points (red dot in “service topology” of Figure 7).

3. Service attributes define the requirements of provisioned slicing services. These attributes are complied with GSMA Service Template (GST) and can be defined based on the SLAs between the vertical customers and the facility site. Details of the service attributes are available in (5G-VINNI, 2019)

4. Service monitoring and testing specifies the network capabilities required by the customers to execute use cases (UCs) experiments during run-time. There are two types of test cases (TCs) supported in 5G-VINNI: network validation (validating the fundamental network KPIs) and service validation (validating the slicing services provisioned to customers). The latter may include customer-specific applications and thus require substantial extra work for design.
A subset of attributes is exemplified in Figure 8. 5G-VINNI provides a UML representation of the SID-driven 5G-VINNI model, as shown in Figure 9, in which serviceTopology, serviceRequirements, serviceTesting and serviceMonitoring corresponding to the Category 2-4, respectively. The UML representation is created given the contents of the 5G-VINNI SB specified by customers who can configure the values of each attribute according to their actual service requirements. Then a customer’s service catalog is created and added into the service catalogue in Figure 3.

<table>
<thead>
<tr>
<th>Group</th>
<th>Parameter Name</th>
<th>Parameter ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>Peak data rate</td>
<td>P.PERF_1</td>
</tr>
<tr>
<td></td>
<td>User data rate</td>
<td>P.PERF_2</td>
</tr>
<tr>
<td></td>
<td>Area traffic density</td>
<td>P.PERF_3</td>
</tr>
<tr>
<td></td>
<td>5G QoS</td>
<td>P.PERF_4</td>
</tr>
<tr>
<td></td>
<td>Reliability</td>
<td>P.PERF_5</td>
</tr>
<tr>
<td></td>
<td>Availability</td>
<td>P.PERF_6</td>
</tr>
<tr>
<td></td>
<td>Service deployment time</td>
<td>P.PERF_7</td>
</tr>
<tr>
<td>Functionality</td>
<td>Deployment option</td>
<td>P.FUNC_1</td>
</tr>
<tr>
<td></td>
<td>Access technology</td>
<td>P.FUNC_2</td>
</tr>
<tr>
<td></td>
<td>Predominant device type</td>
<td>P.FUNC_3</td>
</tr>
<tr>
<td></td>
<td>Radio spectrum</td>
<td>P.FUNC_4</td>
</tr>
<tr>
<td></td>
<td>Isolation</td>
<td>P.FUNC_5</td>
</tr>
<tr>
<td></td>
<td>Support for value-added functionality</td>
<td>P.FUNC_6</td>
</tr>
<tr>
<td></td>
<td>3rd party VNF hosting</td>
<td>P.FUNC_7</td>
</tr>
<tr>
<td></td>
<td>Positioning</td>
<td>P.FUNC_8</td>
</tr>
<tr>
<td>Network Optimisation</td>
<td>Number of devices</td>
<td>P.NO_1</td>
</tr>
<tr>
<td></td>
<td>Device density</td>
<td>P.NO_2</td>
</tr>
<tr>
<td></td>
<td>Coverage profile</td>
<td>P.NO_3</td>
</tr>
<tr>
<td></td>
<td>Mobility profile</td>
<td>P.NO_4</td>
</tr>
<tr>
<td></td>
<td>Service lifetime</td>
<td>P.NO_5</td>
</tr>
</tbody>
</table>
3.3.2.2 Service Exposure Levels

As an effort for innovation, 5G-VINNI proposed a concept of service exposure that allows ICT-19 customers to independently manage their experimentation activities. 4 exposure levels are defined to differentiate the extent to which the management capabilities of each slice instance are exposed to customers. For example, most customers are only interested in collecting the service KPIs of their running UCs whereas some customers need deeper knowledge, e.g., the resource-layer details of their deployed slice instances. Table 5 describes the details of each exposure level. The customers can specify their preferred exposure level in the SB.

Two implications are raised by the service exposure model: (i) the higher the exposure level is, the more advanced capabilities the vertical can get; and (ii) the capabilities offered by a given level also include the ones offered by the level immediately above. Since higher-level service exposure is complex to realize and risks security and privacy, it is important for customers to carefully analyse their service requirements before selecting and specifying the service exposure level during the service design process (fill in the 5G-VINNI SB template).
D2.1. Setup and operation of 5G infrastructure

### Table 5: Service exposure defined in 5G-VINNI

<table>
<thead>
<tr>
<th>5G-VINNI is able to consume operations related to</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
<th>Level 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>E2E network slice application layer config &amp; management</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Network slice subnet / network function application layer config &amp; management</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Network slice subnet / network function virtualised resource layer config &amp; management -&gt; ETSI NFV network service (and VNF) orchestration</td>
<td>×</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Infrastructure resource control &amp; management -&gt; NFVI with optional EPA capabilities and infrastructural SDN control.</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>✓</td>
</tr>
</tbody>
</table>

#### 3.3.2.3 Service Preparation

The 5G-VINNI SB is a top-level service model facing the vertical customers. As a CFS (customer-facing service) descriptor understood by the vertical customers, it is not directly understood by underlying network management and orchestration and thus needs to be mapped to an RFS (resource-facing service) descriptor. This is the responsibility of **Service Preparation**. The RFS details how the slice can be deployed at the resource layer. Therefore, the mapping involves a translation from the abstract slice attributes into concrete NSDs and VNFDs with configuration information for instantiation (e.g., placement of NSs or VNFs and their resource allocation).

In the Norway facility, the following steps are suggested to prepare services:

- Prepare VNFDs for 3rd party VNFs if UCs bring their own VNFs to add into the slice: including the resource and configuration requirements. Each customer or VNF vendor should prepare a VNFD (VNF Descriptor) that is compatible with the VNFM/NFVO in the facility site. In Norway, the Nokia orchestrator accepts TOSCA-based VNFDs. A VNF package is generated as the outcome of this preparation step. It should include the image and VNFD, which contains the configuration information related to the VNFM and EMS (element management system), the latter of which is responsible for FCAPS (fault, configuration, accounting, performance, and security management). In Patras, the VNFDs are prepared in the SOL005 or SOL006 format and are onboarded to the respective OSM platform. Prepare NSDs: Including the resource and configuration requirements. Each customer should prepare a NSD that indicates how their VNFs are interconnected with existing VNFs provided by the facility site. Again, the NSD should be compatible with the NFVO in the facility site. In Norway, the Nokia NFVO (CBND) accepts TOSCA-based NSDs. A NS package is generated as the outcome of this preparation step, including the NSDs it references. The NSDs are onboarded to OSM and then exposed to Openslice. Prepare slice template: once the VNFDs and NSDs become available, the E2E orchestrator (FlowOne in Norway) design the slice specifications. Service chaining is carried out to compose the low-level RFSs (NSDs and VNFDs) into higher level CFSs (network slices). In Patras, the various VNFDs and NSDs are exposed to Openslice as RFSs and are then bundled together to be instantiated in parallel, according to the Use-Case. Dimension the required resources: the resource as pre-requisite for the ordered network slice instance should be provisioned in advance.
● Decide where to place these NS instances and VNF instances
● Publish from the E2E Orchestrator’s Service Catalogue to the Portal Customer Service Catalogue to be ordered by customers: once the service is successfully specified.

**Service Onboarding and Deployment:** During the service onboarding (deployment) process, all SCs of the slice are onboarded and deployed at the appropriate resource premises. In the Norway facility, the onboarding process aims to implement the following steps:

1. Onboard and validate the VNFDs: each VNF that is enclosed in the ordered slicing service, should be onboarded before being used to activate the network slice instance. If the customers use 3rd party VNFs, they need to upload the corresponding images and VNFDs that should be validated by the orchestrator VNF manager (VNFM), being the Nokia CloudBand Application Manager (CBAM) in the Norway facility site and OSM in the Patras facility site. The objective is to ensure that these VNFs are i) compatible with the facility’s capability and ii) secure and reliable. Once onboarded, the VNF is added into the NFVI and its catalogue. Note that this is an iterative process to continuously modify the VNFDs until the requirements are met.

2. Onboard and validate the NSDs: In Norway facility’s case, Nokia’s FlowOne (an E2E service orchestrator) uses its service catalogue to onboard NSDs from its NFVO, being the Nokia CloudBand Network Director (CBND). In Patras facility’s case, the artifacts are either onboarded directly on OSM platform’s catalogue and then discovered by Openslice automatically or onboarded on Openslice and propagated down to the respective OSM. The NSDs should be onboarded and validated such that the NSD is added to the E2E orchestrator’s catalogue. Note that this is an iterative process to continuously modify the VNFDs until the requirements are met.

3. Validate the interconnections, e.g., between VNF instances or between NS instances, to ensure that all SCs are properly interconnected.

4. Validate the entire network slice instance: the facility orchestrator validates if the slice can be properly orchestrated. Meanwhile, it is also planned to run pre-activation testing to validate the entire network slice instance. An iterative procedure may be needed to retune the attribute values based on the validation outcomes.

5. Onboard user equipment: if customers need to attach their own user equipment to a network slicing service, they have to wait until the slicing service has been ordered, instantiated, and activated. Then the 5G SIM cards of the equipment can be activated to the active network slicing service.

### 3.4 Integration of the 5G Solutions orchestrator with 5G-VINNI service orchestrator

Once the service is well prepared (design, preparation, and onboarding), it can be ordered, activated, (configured), and decommissioned by the 5G Solutions orchestrator CDSO. The 5G-VINNI E2E service orchestrator is responsible for the lifecycle management (LCM) of the network slicing services, including service creation, activation, modification, deactivation, and decommissioning. These orchestration flows can be made available to 5G Solutions orchestrator to manage and control the CFSs to 5G Solutions LLs, as a means to automate the LCM of the CFSs, in terms of configuration, activation and decommissioning.

The interactions between 5G-VINNI E2E SO and 5G Solutions CDSO may include:

- 5G-VINNI E2E SO exposes the customer service catalogue to 5G Solutions CDSO via TMF633 → 5G Solutions LLs can select services from the 5G-VINNI service catalogue
- 5G Solutions CDSO orders a service to 5G-VINNI E2E SO via TMF641 → 5G Solutions LLs order services via 5G Solutions CDSO
- 5G Solutions CDSO requests the status of the ordered services, e.g., with respect to the success of the service activation, etc. → 5G Solutions gains knowledge about the operational status of the ordered services.
Note that for some customer-specific VNFs and NSs, it may not be realistic to automate their LCM in the early stage since the LCM workflows need to be implemented.

3.5 Integration of the 5G Solutions with the 5G-VINNI testing and monitoring platform

Once the service is activated, UC owners can activate and validate their applications. The KPI validation process relies on proper KPI measurement and collections, which can be done either manually or automatically. In 5G-VINNI Norway, an automated Testing-as-a-Service (TaaS) platform is provided by Keysight to automate the ordering, scheduling and execution of test cases (TCs). The customers can order experiments in the form of TCs. Note that one UC in a LL may create multiple TCs to experiment different scenarios or different KPIs. Each TC can be executed multiple times to collect sufficient data in different environments (or network conditions). If the TaaS platform is integrated with CDSO, then customers can order the network services and testing services together via CDSO. Otherwise, they need to deploy and register the testing tools into the TaaS platform and then manually order and activate TCs.

The following is a description of the testing and monitoring services planned in 5G-VINNI. Note that the Monitoring-as-a-Service (MaaS) has not been developed and implemented due to the resource limitation.

3.5.1 Testing and Monitoring services

5G-VINNI offers two types of services for testing and validation, Testing-as-a-Service (TaaS) and Monitoring-as-a-Service (MaaS). The supported testing and monitoring tools are shown in Figure 10.

- Norway
  - TaaS: is a set of testing tools and an automation framework that allow the customers to either execute standard/default TCs to actively verify the NS (or network slice) or create and execute customized TCs that can be successfully integrated into the LCM of the NSD. It is possible to enclose customer-specific applications into the TC as the TaaS system is capable of onboarding specific drivers.
  - MaaS: is developed for observation and maintenance of the network slicing services. It contains two types of services:
    - Network Monitoring that produces the traditional overview of the traffic traversing through the network, with special attention to the visibility of special critical points. For network monitoring, the customers define the visibility scope and provide the tools for analysis if needed. 5G-VINNI will provide a set of monitoring tools, as illustrated in Figure 10:
      - Network tap: that can sniff (North-South and East-West) traffic with simple filtering and re-routing (to a specified destination). It can be deployed in specific points of the network and described in NSDs
      - Packet broker: that can perform a preliminary sorting of the selected traffic flows. It allows for more advanced traffic filtering, aggregation, and re-routing (e.g., to an analysis tool or a traffic recording server).
    - Telemetry that focuses on providing the health and performance of the individual NS (or network slice) or application components. It can collect metrics exposed by individual SCs (e.g., VNFs) that are either actively pushed or passive collected.
TaaS supports several testing tools that can be deployed, configured and even automated through offered web services, e.g., traffic generators that emulate real traffic and protocols. In 5G-VINNI, the testing services offered by TaaS include:

- Onboard specific drivers for automating vertical applications and UCs
- Create and execute test scripts to automate the tests or experiments
- Create and execute test campaigns, i.e., a batch of test scripts that can be executed on multiple target infrastructures
- Visualize logs and results through the TaaS visualization system

MaaS is resource-consuming (computation and networking) and therefore is not always suggested unless necessary.

- Greece: In University of Paras infrastructure a MaaS is also deployed. It contains four types of telemetry data:
  - Infrastructure monitoring: Data is gathered from the physical servers that make up the infrastructure. Metrics, such as CPU, memory and disk load are made readily available to telemetry data aggregators at higher levels.
  - Network Service monitoring: Telemetry that focuses on providing the health and performance of the individual NS (or network slice) or application components. These metrics are gathered by OSM’s integrated monitoring platform (integrated Prometheus).
  - Application monitoring: Metrics can be gathered from individual VNFs in two ways. Either by adding lightweight Netdata probes to the VNFs, or by using OSM’s Juju charms to gather metrics periodically, without affecting the actual VNF. This offers application-level metrics, that can be customized according to the application and the Use-Case.
  - 5G monitoring: Metrics can be gathered by the 5G core, regarding various network KPIs, such as the number of connected users (UEs) or the Cell’s bandwidth.

### 3.5.2 Integration process

In a sense, the TCs are a special class of services available to customers. Therefore, the integration process for TCs is similar to that of the CFSs. In general, it includes the following steps:

- TC design: produces a detailed design plan/scheme for the experiment, by considering the following aspects
  - what is the objective of the experiment, e.g., the target SUT or the target KPIs
  - what KPIs are interesting
where are these KPIs measured, i.e., the measurement points (UEs, VNFs, or network ingress/egress interfaces)

- how are these KPIs measured, i.e., via 5G-VINNI testing and monitoring platform (e.g., 5G-VINNI provides Testing-as-a-Service (TaaS) and Monitoring-as-a-Service (MaaS) that offer network-side KPIs) vs. 5G Solutions LL testing and monitoring functionality (application-specific KPIs may need to be measured by 5G Solutions UC owners)

- what tools are needed to measure these KPIs, 5G-VINNI testing and monitoring tools vs. 3rd party monitoring tools, etc.

- how possible is it to automate the experiment orchestration process

TC preparation and validation: in most cases, the TC needs KPIs from the 5G-VINNI facility, at least for the network-side data. Then a TC descriptor should be prepared. 5G-VINNI provides Testing-as-a-Service (TaaS) to customers. In TaaS, all experiments or TCs are defined by test scripts that are to be written by customers (UC owners). These scripts contain information on:

- the testing and monitoring tools involved,
- the workflows to execute the experiments, and
- the KPIs to be measured and collected,

- (probably) the policies of how the KPIs are collected and stored (e.g., sampling period, etc.). The scripts need to be validated, e.g., syntax checking, etc.

TC deployment: includes two parts:

1) Onboard the drivers specific for automating the TCs
2) Deploy the testing and/or monitoring tools demanded by each TC (instantiation and configuration) prior to the activation of the TC. The deployment could be manual or automated, depending on how the tools are supported by TaaS or MaaS. It is also expected that the deployed TC is validated to ensure that:
   - The required KPIs are measured and collected
   - The required KPIs can be properly retrieved by the KPI visualization system

TC ordering: customers (UC owners) can order TC services via 5G Solutions CDSO that passes the ordering request to 5G-VINNI TaaS

TC activation: once the orders are validated by 5G-VINNI TaaS, the TC services are activated to run the corresponding experiments and collect the KPIs of interest.

TC data collection: the measurement data (KPIs) are stored in 5G-VINNI TaaS and can be retrieved by 5G Solutions KPI visualization system for visualization and/or further processing and analysis.

Note that pre-activation tests can be run by the Norway facility via TaaS to validate the slicing service before activating it while there is no application traffic. These tests require minimal configurations and can be enclosed as part of the service deployment of CFS (as built-in or default tests?). The objective is to ensure proper instantiation of VNFs, NSs and network slices, and proper measurement and extraction of the required KPIs.

The UC-specific TCs include UC applications, which onload application traffic to the activated network slice.

3.5.3 Integration Implementation

The TaaS (and MaaS) can be consumed in two ways (see Figure 11):

- APIs (left side of Figure 11): a set of service validation TCs can be ordered via the APIs to BSS/OSS or MANO and activated and executed once the network slicing service is activated. These validation TCs are designed to validate the health or performance of the network slicing service. This process can be automated by programatically calling the APIs to order TCs. This option is recommended as it ensures consistency and repeatability of the results.

- Web portal (right side of Figure 11): more exploratory TCs can be ordered and executed via a web portal in a manual way.
From the perspective of testing and validation, the integration of 5G Solutions with 5G-VINNI contains two parts:

1. Integration with the CDSO to orchestrate experiments. (In theory), the TCs describe their test details in VNFD/NSD and thus the orchestration of TCS is expected to be done by CDSO?
2. Integration with the KPI visualization system to collect KPIs. The testing results (TaaS and MaaS) are stored in heterogeneous data stores. At present, there is no clear definition about the data stores as they depend on the collected KPIs and used tools. The specific details of how the interested parties can consume the collected KPIs were discussed earlier.

3.5.4 Example KPIs from 5G-VINNI

The 5G-VINNI facility provides network KPIs by default that measure the capabilities of the 5G network provisioned by the facility infrastructure. As shown in Table 6, these KPIs are divided into:

- E2E network performance: that validates the E2E network service performance by using handsets and CPEs from end to end.
- NFVI network performance: that verifies the connectivity capability of the infrastructure
- NFVI compute resource performance: that verifies if the NFV HW profile is capable of supporting the NFV architecture

<table>
<thead>
<tr>
<th>Category</th>
<th>KPI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>E2E Network performance</td>
<td>UL Max throughput (Mbps)</td>
<td>mmWave, 27.3-27.5GHz, 4T4R, 1 stream</td>
</tr>
<tr>
<td></td>
<td>DL Max throughput (Mbps)</td>
<td>mmWave, 27.3-27.5GHz, 4T4R, 40 streams</td>
</tr>
<tr>
<td></td>
<td>UL latency (ms)</td>
<td>3.6GHz, low foot-print traffic profile with 100Kbit/s bandwidth.</td>
</tr>
<tr>
<td></td>
<td>DL latency (Ms)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UL jitter (ms)</td>
<td></td>
</tr>
<tr>
<td>NFVI network performance</td>
<td>Max throughput with 0 frame loss</td>
<td>Two compute nodes, DPDK was configured in the test environment.</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>Latency between VMs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DL jitter (ms)</td>
<td>mmWave, 27.3-27.5 GHz, 4T4R, low foot-print traffic profile with 100Kbit/s bandwidth.</td>
<td></td>
</tr>
<tr>
<td>UL frame loss (%)</td>
<td>3.6GHz, low foot-print traffic profile with 100Kbit/s bandwidth.</td>
<td></td>
</tr>
<tr>
<td>DL frame loss (%)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NFVI compute resource performance</th>
<th>CPU benchmarking score</th>
<th>Yardstick CPU/Memory/storage tests</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Memory read latency</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Memory r/w bandwidth</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Storage r/w IOPS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Storage r/w latency</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Storage r/w bandwidth</td>
<td></td>
</tr>
</tbody>
</table>
4 5G EVE as supporting platform

This section reports in chapter 4.1 a brief description of the process foreseen by 5G-EVE external project integration. In the following chapters, it is explained how 5G Solutions uses the tools and basic services offered by 5G-EVE to deploy a use case developed in LL. The experiment workflow described in this Section has been completely executed for trials in Cycle 1 concerning two use cases in Living Lab Smart Energy, namely Industrial Demand Side Management (UC2.1) and EV Smart Charging (UC2.2). Related results and lessons learned through the integration of the Use Cases in the testbed have been reported in “D5.6-D5.3A: LL performance evaluation and lessons learned (v1)”, see Section 4.1.2 and 4.2.2 for UC2.1 and UC2.2, respectively.

In Section 5.2) the experiment workflow will be detailed to accommodate individual UC specificities during the trial process.

4.1 Integration process description

The 5G-EVE project offers a platform to Verticals (e.g., Living Labs in 5G Solutions) to test and validate an Experiment (e.g., a Use Case in 5G Solutions) developed by the Vertical and deployed on the 5G-EVE platform.

The reference documentation for the 5G-EVE integration consists of two main documents [1] and [2]. The reference documentation describes the complete use case on boarding and configuration workflow from the Use Case Developer’s point of view.

The 5G-EVE approach for Experiment deployment is to provide to the owner of the Experiment a dedicated Web Portal ([3]) to define the use case of interest and, accordingly, to request a deployment in terms of new VNFs and network blueprint. 5G-EVE defines different actors in [1] that have different roles in the experiment life cycle management. The following table maps those roles in the scope of 5G Solutions.

<table>
<thead>
<tr>
<th>Actor in 5G-EVE</th>
<th>Description</th>
<th>Target actions</th>
<th>Actor in 5G Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anonymous user</td>
<td>User not authenticated to the 5G-EVE Web Portal</td>
<td>No actions are allowed</td>
<td>End user (if applicable)</td>
</tr>
<tr>
<td>Vertical</td>
<td>Actor with the knowledge of the service to be tested, including SLAs and service components</td>
<td>● Support in the definition of the vertical service blueprints and related experiments.</td>
<td>5G Solutions Use Case responsible</td>
</tr>
</tbody>
</table>
| Vertical’s VNF provider | Actor who provides the VNF packages for the vertical applications. | ● Uploading of VNF packages in 5G-EVE portal.  
● Issuing of requests (via 5G-EVE portal) for VNF packages on-boarding to specific sites. | 5G Solutions VNF producer for the specific Use Case + 5G Solutions Orchestrator |
| Experiment developer | Actor responsible for specifying the blueprints associated to an experiment, as well as the associated NFV network services descriptors. This user has the knowledge about the 5G-EVE infrastructure and expertise about NFV network service modelling. Moreover, it interacts with the vertical to receive information | ● Modelling and definition of blueprints for vertical services, experiment contexts and experiments.  
● Modelling and definition of NFV Network Service Descriptors (NSD) associated to the experiment blueprints, as well as related translation rules.  
● Definition and development of mechanisms for collecting | 5G Solutions Use Case developer partner |
### D2.1. Setup and operation of 5G infrastructure

<table>
<thead>
<tr>
<th>Role</th>
<th>Responsibilities</th>
<th>Tools</th>
<th>Notes</th>
</tr>
</thead>
</table>
| Experimenter | Actor responsible for the request of an experiment and the assessment of its results. He/she defines characteristics of an experiment starting from its blueprint, requests the deployment of related virtual environment and experiment execution and analyzes results and KPIs. | ● Selection of blueprints for the target vertical service and experiment, via 5G-EVE Portal.  
● Configuration of the experiment, according to the blueprint parameters, via 5G-EVE Portal.  
● Issuing of requests, via 5G-EVE Portal, for scheduling the execution of the experiment in the 5G-EVE facilities.  
● Monitoring of the experiment execution, via 5G-EVE Portal.  
● Visualization of experiment measurements, KPIs and results, via 5G-EVE Portal. | 5G Solutions Use Case responsible |
| Site manager | Actor responsible for the management of the infrastructure and the orchestration systems in a particular site. | ● Pre-provisioning and configuration of the physical infrastructure in the managed site.  
● Validation and on-boarding of VNF packages.  
● Preparation of resources needed to deploy an experiment. | (External) 5G-EVE Site Manager |
| System administrator | Actor who has access to all tools provided by the 5G-EVE project. | All actions, including management functions (e.g., configuration of users and related permissions, visualization of system logs, etc.)  
Note: System administrator may have a limited access to the management system of each single site facilities. | (External) 5G-EVE System administrator |

For the testing and validation phase of the Experiment, 5G-EVE offers different tools integrated into the Web Portal.

In the life-cycle of an Experiment, we can distinguish four major phases:

(a) experiment design and definition  
(b) experiment preparation  
(c) experiment execution and monitoring  
(d) experiment results analysis

The experiment design and definition phase is the initial phase where the experiments for a given vertical service are technically defined, planned and formalized, with the collaboration and off-line interaction of verticals, experiment developers, VNF providers and 5G-EVE Site Managers. This phase has the objective of identifying the main components of the service, their interactions, the target execution environment(s) and condition(s), the elements to be monitored and the relevant KPIs, as well as the detailed steps to run the experiment.
As result of this phase, the blueprints of vertical services (Vertical Service Blueprint – VSB), contexts (Context Blueprint – CB), test cases (Test Case Blueprint – TCB) and experiments (Experiment Blueprint – ExpB) are produced and physically on-boarded in the system.

The blueprints are a sort of structured “template” that define high-level features of services, execution environments and experiments, respectively. Such blueprints include also several variable parameters characterizing each single Experiments that can be specified by the experimenters during the experiment preparation phase for tuning the specific experiment instance. An experiment blueprint filled with concrete values assigned to its variable parameters is called Experiment Descriptor (ExpD) and it corresponds to a deployment model for the instantiation of an experiment, summarizing the characteristics of the target virtual environment where the experiment needs to run.

Together with the blueprints, the experiment design and definition phase also produce a set of VNF Packages implementing the virtual applications of the Vertical Service and one or more NFV Network Service Descriptors (NSD) that describe potential deployments of the experiments in target 5G-EVE facilities. VNF Packages are on-boarded on target 5G-EVE sites, subject to a preliminary validation of the packages by the site owners. NSDs are also on-boarded in the system, namely on the 5G-EVE Catalogue, together with “translation rules” that describe how to automatically map Experiment Descriptors into an equivalent NFV Network Service instance.

The following experiment preparation phase involves mostly experimenters and site managers and has the objective of preparing an environment in the 5G-EVE facility suitable for the execution of a given experiment. At the beginning, the experimenter uses 5G-EVE portal to select and configure target ExpB, specifying all needed parameters to generate a valid related ExpD. The ExpD is then on-boarded in the portal and used as baseline for the request of the experiment. Since in most of cases the execution of an experiment requires manual setting and configuration of the physical infrastructure at 5G-EVE sites, as well as the reservation of dedicated resources, the experiment execution needs to be scheduled in a specific time period that is agreed between experimenter and involved site manager(s). Once the experiment is scheduled, the Site Manager(s) is responsible to guarantee that the target physical infrastructure(s) will be able to host the experiment as planned. This typically involves some off-line actions where the site manager configures the settings of the site facility.

The most important phase is related to the experiment execution and monitoring. The initial step of the experiment execution is the instantiation and configuration of the virtual environment where the experiment will be dynamically set up and, accordingly, will run. The following one concerns monitoring the experiment progress and accessing graphs describing metrics, logs and KPIs for the final phase.

After experiment execution, diagnostics tools will help experimenters in the evaluation of the service, checking reference statistics and comparing KPIs from different experiment settings during the experiment results analysis phase.

The experiment design, definition and preparation are summarized in chapter 3.1 (High-level interactions between 5G-EVE users and 5G-EVE platform) of [1] and detailed in chapter 3.2 (Internal workflows of 5G-EVE platform for experiments management) of [1] and chapter 2.4 (Portal GUI) and 4 (Service handbook) of [2].

4.2 Usage of the Web Portal for UC configuration

The 5G-EVE Web Portal URL is: https://portal.5g-eve.eu/ ([3]).

The 5G-EVE Web Portal provides a web-based, unified access point for verticals and experimenters to all the functionalities made available by the 5G-EVE platform. In the context of 5G Solutions the portal is used for the entire onboarding and deployment activities to support the execution of each single test case developed in the Living Lab for a given Use Case of interest, while the experiment execution and status monitoring are performed via the 5G Solutions Orchestrator using the 5G-EVE north-bound APIs.
A user of the portal must first register himself/herself to the portal specifying his/her role. A 5G Solutions user typically covers more than just one role so, after the first registration, he/she must open a ticket in the portal asking the 5G-EVE Administrator to update his/her profile also the other roles of interest. The roles in scope of 5G Solutions are: Experimenter, Vertical, ExperimentDeveloper and VNFDeveloper (please refer to Table 8 for corresponding role in 5G Solutions).

The Portal includes several tools that facilitate the design and the definition of the experiment. These tools are used by 5G Solutions actors to deploy, to execute and to validate the experiment, since the Portal also offers tools for the execution, monitoring and evaluation of experiments.

The 5G-EVE Web Portal tools are summarized in [1], see Chapter 2.3 “5G-EVE experimentation tools for experiment definition and monitoring”. A more detailed overview of the tools is in [1], see Chapter 4 “Experimentation tools, including intent-based interfaces”.

The 5G-EVE Web Portal is used by the 5G Solutions Use Case Responsible, playing the role of Experimenter (see Table 8), to request the deployment of the UC itself. Specifically, the Portal is used to deploy the Use Case of interest for the vertical LL2 Smart Energy. LL2 is composed by three use cases:

- UC2.1: Industrial Demand Site Management (Use Case responsible IREN)
- UC2.2: Electrical Vehicle Smart Charging (Use Case responsible IREN)
- UC2.3: Electricity Network Frequency Stability (Use Case responsible Enel X Way)
All Use Cases in LL2 Smart Energy are deployed over the 5G-EVE Italian Site located in Turin. The roles played in each UC is reported in the following table.

<table>
<thead>
<tr>
<th>Role in 5G Solutions</th>
<th>UC2.1</th>
<th>UC2.2</th>
<th>UC2.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>5G Solutions Use Case responsible</td>
<td>IREN</td>
<td>IREN</td>
<td>Enel X Way</td>
</tr>
<tr>
<td>5G Solutions VNF producer</td>
<td>Ares2t</td>
<td>Ares2t</td>
<td>Ares2t</td>
</tr>
<tr>
<td>5G Solutions Use Case developer partner</td>
<td>Ares2t</td>
<td>Ares2t</td>
<td>Ares2t</td>
</tr>
</tbody>
</table>

To deploy a UC, as an Experiment in the 5G-EVE platform, the VNFs for specific UC and some artifacts must be prepared. The VNFs depend on the specific tests to perform within each UC. All VNFs developed in LL2 are detailed in D5.3A. In general, each UC in LL2 will rely on a Remote Controller that will be implemented as VNF in the 5G-EVE facility. The 5G Solutions Use Case developer partners, for each Use Case, need to upload the VNFs using the “VNF Storage” tool on the 5G-EVE Web Portal. The tool is described in [2], see Chapter 1.4.1 “VNF Storage”. The 5G Solutions Use Case developer partner is expected to upload one zip file containing the proper VNFDs, the corresponding images and the NFV Network Service Description (NSD) containing information on how to map the NFV Network Service instances. He can include a list of sites indicating where the uploaded VNFs should be onboarded. In the case of vertical LL2 Smart Energy, the one location is the 5G-EVE Italian Site in the metropolitan area of Turin.

The VNF package typically includes the image with the developed application, the VNFD and several scripts as well as metadata for the management of the application. It should be noted that the format of VNF packages and VNFDs are dependent on the specific site where the VNFs will be deployed.
From Figure 12, the first phase of the deployment procedure is the upload of the blueprints/descriptors in the Section “Design Experiment”. As shown in Figure 13, the Use Case Developer needs to on-board the proper files in the sections “Vertical Service Blueprints”, “Context Blueprints”, and “Test Case Blueprints”. In particular, the VSB, VSB NSD, CTX, CTX NSD and TCB must be inserted in the portal. Finally, the Section “Experiment Blueprints” is responsible for the creation of an Experiment Blueprint which gathers all the information stated in the other configuration files. This tool is thoroughly described in chapter 1.4.3 (Experiment Blueprint Builder) of [2]. This tool is organized in steps where the Experimenter selects the components of the Experiment Blueprint, defines relevant parameters (e.g., metrics and KPIs) and on-boards the associated EXP NSD.

At this stage, the experiment created in the previous section, can be scheduled, and on-boarded from the Section “Request Experiments” (see Figure 12 and Figure 14).

In particular, the Experiment Descriptor is on-boarded by filling some parameters as:

1. Number of devices
2. Incoming Traffic Load

These values can be set according to the preferences of the Experimenter, in order to test different network backgrounds for the experiment. Finally, the Experimenter is asked to choose a proposed date time to execute the experiment. The whole procedure for the on-boarding of the Experiment Descriptor is also described in [1], chapter 4.4 (Intent Based Interface).

Consequently, the Use Case Developer asks the Site Manager to accept the request for deployment. Finally, all the VNFs can be deployed in the Section “Manage Experiment”. Once the deployment is executed successfully, the experiment can be activated via Portal or via Cross-Domain Service Orchestrator (CDSO).

4.3 Integration of the north-bound APIs with the 5G-Solution orchestrator

The 5G-EVE north-bound APIs can be used by the 5G Solutions Orchestrator to automate the experiment execution. The APIs are described in [2].

To use those APIs the 5G Solutions Orchestrator must authenticate to the 5G-EVE portal backend as described in chapter 2.1.1 (Role-Based Access Control) of [2].
The API to request the experiment execution is described in chapter 2.1.2 (Experiment Lifecycle Manager) of [2]. The REST interface is described in chapter 1.2.2 (Experiment Lifecycle Manager Interface) of [2].
4.3.1 Activate and Terminate Experiments from CBND UI.

CDSO (CBND) enables Activation and Termination of experiments through Deploying and Termination of "Network Service" instance that can be created from "NS Package".

NS Package is an artefact containing the definition of interfaces and workflows described below (4.4.2, 4.4.3).

It is important that Experiment already prepared in eve portal by the UC owner, and below steps taken place:

- Design and define experiment – manual process [UC Owner]
- create experiment in 5G-EVE portal and upload all descriptors and required hardware, pre-conditions [UC Owner]
- Wait until experiment is Ready for execution [EVE]
  - after manual preparation is ready, status will move to READY (may take few days)

4.3.1.1 Preparing the artefact

Artefact (provided by NOKIA CBND team) should be uploaded for the first time from the catalog menu, to do that:

1- Login to the CBND UI using the provided credentials:
   a. Username: {REALM\USER}
   b. Password: {user password}
2- Go to Catalogue, "Create new package" of type "NS Package" and follow instructions in the wizard to complete the package details and upload the artefact file.

4.3.1.2 Activate Experiments

In order to Activate new experiments:

1- login to CBND UI
2- In the Catalogue menu, search the "5gEVE" package with the relevant version.
3- Ensure the package is in "active" status in order to create a new network service
4- Create a new Network service, providing the required input fields
5- Deploy the newly created Network service

Deployment of the networks service will execute the described Workflow (see the below sequence diagram), Which "Deploys" and "Executes" while providing the required configurations of the experiment in the 5GEVE system
4.3.1.3 Terminating Experiments

In order to Terminate an existing experiment in the 5GEVE system:

1. login to the CBND UI
2. go to Network Service\NS Instance
3. locate the intended Network Service (experiment) for termination
4. and select the terminate action

Terminating of the Networks Service will execute the described Workflow (see the below sequence diagram), which terminates and deletes the experiment from the 5GEVE system.

APIs Swagger can be found under this repository

https://github.com/nextworks-it/experiment-portal/blob/master/API/ExperimentLifecycleManager.json

4.3.2 Login

All the requests received at the (Experiment Lifecycle Manager) ELM NBI are authenticated and authorized through the RBAC component, according to the actions permitted for the different 5GEVE roles, as defined in API specifications (D4.1). To do login and get the access token first call the below APIs:

Table 9: 5G-EVE login APIs

<table>
<thead>
<tr>
<th>HTTP method</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td>/portal/rbac/register</td>
<td>Endpoint to create a new user at the portal</td>
</tr>
<tr>
<td>POST</td>
<td>/portal/rbac/login</td>
<td>Allows the creation of a new session for a specific user who is already registered at the portal.</td>
</tr>
<tr>
<td>POST</td>
<td>/portal/rbac/refreshtoken</td>
<td>Allows users to obtain a new access token once the one they have is expired</td>
</tr>
<tr>
<td>GET</td>
<td>/portal/rbac/logout</td>
<td>Closes a user session</td>
</tr>
</tbody>
</table>

POST /portal/rbac/login

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Parameter type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Body (JSON)</td>
<td>email</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td>password</td>
<td>String</td>
</tr>
<tr>
<td>Response body</td>
<td>access_token</td>
<td>String</td>
</tr>
<tr>
<td>Response body</td>
<td>Refresh_token</td>
<td>String</td>
</tr>
</tbody>
</table>

Successful response HTTP code: 200 OK
Error response HTTP code: 400 BAD REQUEST, 500 INTERNAL ERROR

4.3.3 Activate
To activate a new experiment, first it should be created and then deployed in the 5G-EVE portal.

**Sequence Diagram**

**Figure 15:** “Activate” sequence diagram
4.3.3.1 Deploy

To Deploy the already created Experiment execute the below API using the experimentId selected in placement stage in CDSO.

Table 10: Deploy API request Model

<table>
<thead>
<tr>
<th>POST /portal/elm/experiment/{expId}/action/deploy</th>
<th>Parameter name</th>
<th>Parameter type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request body</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td>URI Variables</td>
<td>expId</td>
<td>String</td>
<td>ID of the experiment to be deployed.</td>
</tr>
<tr>
<td>Response body</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

Successful response HTTP code: 202 ACCEPTED
Error response HTTP code: 400 BAD REQUEST, 403 FORBIDDEN, 404 NOT FOUND, 409 CONFLICT, 500 INTERNAL ERROR

Example: To Deploy Experiment {expId}

```
$> curl -X POST https://{5GEVE.url}/portal/elm/experiment/{expId}/action/deploy -H 'Authorization: bearer ${token}''
```
4.3.3.2 Execute experiment request Model

To Deploy the just created Experiment execute the below API using the experimentId returned from the create step.

“Execute” API request Model

Table 11: “Execute” API request Model

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Parameter type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>executionName</td>
<td>String</td>
<td>Value that is given on creation of the NS instance in CDSO</td>
</tr>
<tr>
<td>experimentId</td>
<td>ExecutionTimeSlot</td>
<td>Experiment Id that is selected in deployment stage in CDSO</td>
</tr>
<tr>
<td>perfDiag</td>
<td>List&lt;EveSite&gt;</td>
<td>Value that is given on creation of the NS instance in CDSO</td>
</tr>
<tr>
<td>testCaseDescriptor</td>
<td>String</td>
<td>Json which is Configuration that can be passed to the Experiment execution process, value can be given in the NS instance creation stage in CDSO:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>{</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp1&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp1&quot;: &quot;string&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp2&quot;: &quot;string&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp3&quot;: &quot;string&quot; },</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp2&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp1&quot;: &quot;string&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp2&quot;: &quot;string&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp3&quot;: &quot;string&quot; },</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp3&quot;: {</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp1&quot;: &quot;string&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp2&quot;: &quot;string&quot;,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;additionalProp3&quot;: &quot;string&quot; }</td>
</tr>
</tbody>
</table>

URL path param expId string Experiment Id that is selected in deployment stage in CDSO

Successful response HTTP code: 201 CREATED

Error response HTTP code: 400 BAD REQUEST, 403 FORBIDDEN, 404 NOT FOUND, 500 INTERNAL ERROR

Example: To Deploy Experiment {expId}
$> curl -X POST https://${5GEVE.url}/portal/elm/experiment/{expId}/action/execute -H 'Authorization: bearer ${token}' -d {
  "executionName": "string",
  "experimentId": "string",
  "perfDiag": true,
  "testCaseDescriptorConfiguration": {
    "additionalProp1": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    },
    "additionalProp2": {
      "additionalProp1": "string",
      "additionalProp2": "string",
      "additionalProp3": "string"
    }
  }
}
4.3.4 Terminate

To terminate an existing Experiment, it should be terminated and then deleted from 5G-EVE system.

Sequence Diagram

Figure 16: “Terminate” sequence diagram
4.3.4.1 Terminate

"terminate" action should be executed through the API,
It terminates the virtual environment where the experiment has been executed (i.e. terminates the associated NFV network service). The experiment must be in instantiated state.

Example:

```bash
$> curl -X POST https://${5GEVE.url}/portal(elm/experiment/{expId}/action/terminate -H 'Authorization: bearer ${token}"
```

4.3.4.2 Delete

"Delete" action should be executed through the API, Removes an experiment and its record from the system. The experiment must be in refused, terminated, or failed state.

Table 13: “Delete” API request Model

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Parameter type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>expId</td>
<td>String</td>
<td>ID of the experiment to be deleted.</td>
</tr>
<tr>
<td>Response body</td>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

4.3.5 Get Status

Sequence Diagram
Figure 17: “Get Status” sequence diagram
4.4 Usage of the Validation tools

In Cycle 1, Vertical Service and Network KPIs have been set for both UC2.1 and UC2.2. The Network KPIs are common for all the TCs (see Table 15). Instead, the Vertical Service KPIs are specific for each TC. This is summarized in Table 16.

### Table 15: Reference of Network KPIs in Cycle 1

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latency Userplane RTT</td>
<td>Network</td>
<td>ms</td>
</tr>
<tr>
<td>Availability</td>
<td>Network</td>
<td>%</td>
</tr>
<tr>
<td>Reliability</td>
<td>Network</td>
<td>%</td>
</tr>
<tr>
<td>User Data Rate Downlink</td>
<td>Network</td>
<td>Mbps</td>
</tr>
<tr>
<td>User Data Rate Uplink</td>
<td>Network</td>
<td>Mbps</td>
</tr>
</tbody>
</table>

### Table 16: List of Vertical Service KPIs for all TCs

<table>
<thead>
<tr>
<th>Name</th>
<th>TC</th>
<th>Type</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuation Time</td>
<td>2.1.1</td>
<td>Vertical Service</td>
<td>s</td>
</tr>
<tr>
<td>Disconnection Time</td>
<td>2.1.3</td>
<td>Vertical Service</td>
<td>ms</td>
</tr>
<tr>
<td>Aggregated charging power reference curve deviation</td>
<td>2.2.3</td>
<td>Vertical Service</td>
<td>kWh</td>
</tr>
<tr>
<td>State of charge</td>
<td>2.2.3</td>
<td>Vertical Service</td>
<td>kWh</td>
</tr>
<tr>
<td>Sampling time</td>
<td>2.2.3</td>
<td>Vertical Service</td>
<td>s</td>
</tr>
</tbody>
</table>

An important distinction between these two types of metrics (Network and Vertical Service) is their origin. Precisely the Network KPIs are computed and registered in the 5G-EVE facility and consequently collected and filtered by the 5G Solutions KPI VS. The Vertical Service KPIs are, instead, computed in the VNFs deployed...
dynamically and eventually sent to the 5G Solutions KPI VS through the dedicated APIs exposed by the platform (see D5.3A, Section 3.1).

In particular, the Network KPIs are configured during the experiment preparation phase (see Section 4.1). Indeed, once all the configuration files are uploaded in the portal (VSB, VSB NSD, TCB, CTX, CTX NSD), an experiment blueprint must be created in the “Design Experiment” section (see Figure 12). At this stage, the Use Case Developer can add a set of Network KPIs as shown in Figure 18. During the experiment execution, the registration of these metrics is activated through the CDSO which is responsible for the experiment activation on the 5G-EVE portal.

![Figure 18: 5G-EVE Portal, Network KPIs Configuration](image)

Finally, all the metrics of a specific experiment can be visualized and analysed on the 5G Solutions KPI VS. An example is shown in Figure 19.
4.5 Deployment and execution guide

A first idea of how to deploy and execute the experiment via the 5G-EVE portal is in chapter 3 ("Service handbook") of [2].

For technical details on how to prepare the experiment look at the Annex section of [1].

1. Virtual Machine preparation (Network interfaces setup and python2 installation) and snapshot of the instance.
2. Interaction with EVE site manager to define the details for the VNF package. Upload the VNF package, a zip file containing the proper VNFD and the corresponding snapshot ids.
3. Blueprints and Descriptors preparation
4. Upload of the Blueprints/ descriptors and creation of the experiment through the portal
5. Experiment Scheduling
6. Usage of 5G-EVE Ticketing tool to ask for VNF onboarding
7. Confirmation by the Site Manager. VNF ready to be deployed.
8. VNF deployment through the portal.
9. Experiment Execution via the 5G Solutions Orchestration or via Cross-Domain Service Orchestrator (CDSO)
10. Monitor Experiment progress and result via the 5G Solutions Visualization System.
5 Insight on LLs deployment on the supporting platforms

With a reference to chapter 3 and 4 where the supporting platforms for 5G-Solutions are detailed, this chapter provides the relevant insights related to the actual usage of those platform to support the deployment of the Living Labs. Because more Living Labs use the same features from the supporting platforms, the following subchapters report examples from different living labs covering the relevant aspects related to the actual usage of those features in 5G-Solutions.

5.1 Insight on LLS deployment on 5G-VINNI

Living Lab 4 is mainly using the 5G-VINNI network platform at university of Patras.

The six UCs in LL4 are utilizing the tailored network slices in the form of Network Slice as a Service (NSaaS) for testing and validating that are developed for the needs of media and entertainment. UoP along with other WP6 partners facilitate the integration process for UCs design, development, deployment of the different streaming services and running of experiments for testing and validation in a DevOps way.

The following general steps were executed in order to achieve end to end testing of the desired trial scenario:

1. [In most UCs] CDSO starts the scenario by sending a service order which will enable the required services and, in most cases, start the Monitoring services as well.
2. The order is sent from CDSO to the UoP Service Orchestrator, OpenSlice, which is the entry point for every interaction with UoP infrastructure.
3. Depending on the order specifics, this order is propagated either to OSM, which will deploy and configure any VNFs or PNFs required, or to other internal services in UoP infrastructure. This step will usually configure the network slices that are going to be used.
4. The relevant service according to the UC start streaming content on the designated network slice
5. CDSO starts the monitoring data capture on the Visualization System to collect the necessary network metrics
6. Metrics and KPIs are collected in the Visualization System. VS gets these metrics from the appropriate endpoint provided by UoP.

An example taken from UC4.1:
The network configuration was set according to the different needs of the UC.

Here is an example from UC 4.3 configuration which illustrate the configuration of uplink-oriented slice:

Table 17: Configuration of uplink-oriented slice for UC 4.3

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Generic Configuration</th>
<th>UL oriented Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freq</td>
<td>3.6GHz (n78)</td>
<td>3.6GHz (n78)</td>
</tr>
<tr>
<td>BW</td>
<td>50 MHz</td>
<td>50 MHz</td>
</tr>
<tr>
<td>DL</td>
<td>2 × 2 (MIMO)</td>
<td>2 × 2 (MIMO)</td>
</tr>
<tr>
<td>UL</td>
<td>1 × 1</td>
<td>2 × 2 (MIMO)</td>
</tr>
<tr>
<td>DL / UL slots</td>
<td>6:3</td>
<td>2:7</td>
</tr>
<tr>
<td>Modulation</td>
<td>qam64 / qam64</td>
<td>qam256/ qam256</td>
</tr>
</tbody>
</table>

In cycle 3, we extended the configurations from cycles 1 and 2 by using the following parameters:

1. Increasing the number of services per UC
2. Generating higher load on the network to simulate real life scenarios
3. Using generic versus different network configurations in conjunction with CDSOs
4. Triggering a ZTA for several use cases

A number of services are available in UoP that can be used depending on each UC. Some of these services are VNF based and are dynamically deployed for each UC to be tested and others are always operational and ready to be accessed by the UC through the CDSO.
D2.1. Setup and operation of 5G infrastructure

The following tables shows these services and how they can be used in each UC.

<table>
<thead>
<tr>
<th>Table 18: Services used in each UC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>VNF</strong></td>
</tr>
<tr>
<td>FNET</td>
</tr>
<tr>
<td>UC4.1</td>
</tr>
<tr>
<td>UC4.2</td>
</tr>
<tr>
<td>UC4.3</td>
</tr>
<tr>
<td>UC4.4</td>
</tr>
<tr>
<td>UC4.5</td>
</tr>
<tr>
<td>UC4.6</td>
</tr>
</tbody>
</table>

Example of VNF use as part of UC4.5:

5.2 Insight on LLS deployment on 5G EVE

Living Lab 2 (Smart Energy) is the one using the 5G EVE experimental platform in the Italian Site located in the City of Turin.

The three UCs in LL2 exploited the 5G-EVE experimental platform in C1 (fully integration and exploitation during trial activities) and in C2 and C3 (partially because of the unavailability of some functionalities of the platform since January 2022). In both situations, the NR RAN was preserved, with same performances, and the trials achieved with a number of options to reduce the dependencies of use case execution from the 5G-EVE platform.
As described in Section 4, the following workflow was executed in order to achieve end to end testing of the desired trial scenario in Cycle 1:

1. Virtual Machine preparation (Network interfaces setup and python2 installation) and snapshot of the instance.
2. Interaction with EVE site manager to define the details for the VNF package. Upload the VNF package, a zip file containing the proper VNFD and the corresponding snapshot ids.
3. Blueprints and Descriptors preparation
4. Upload of the Blueprints/ descriptors and creation of the experiment through the portal
5. Experiment Scheduling
6. Usage of 5G-EVE Ticketing tool to ask for VNF onboarding
7. Confirmation by the Site Manager. VNF ready to be deployed.
8. VNF deployment through the portal.
9. Experiment Execution via the 5G Solutions Orchestration or via Cross-Domain Service Orchestrator (CDSO)
10. Monitor Experiment progress and result via the 5G Solutions Visualization System.

Since January 2022, the 5G EVE web portal ([https://portal.5g-eve.eu/](https://portal.5g-eve.eu/), ref. Section 4.2) necessary for steps 1-8 was not reachable at the beginning of November 2022. On the 17th, the services have been restarted by one partner in 5G EVE project. From that day, 5G-EVE web portal is reachable, but no access is allowed with previous working login data. New access was requested by not granted by partner in 5G-EVE project.

Since January 2022, the North-Bound APIs exposed by the 5G-EVE platform necessary for steps 9-10 were not available (ref. Sections 4.3 and 4.4).

Therefore, starting from February 2022, the implemented options to avoid dependency on the 5G-EVE experimental platform in C3 were:

1. Manually operated experiment onboarding and static VNF configuration to avoid interaction with the 5G-EVE web portal and network service orchestrator via North-Bound APIs by the 5G-EVE platform for steps 1-8
2. Vertical Service KPIs and suitable network KPIs collected by the Vertical Service and sent to the 5G-Solutions KPI Visualization System to avoid interaction with the 5G-EVE Monitor Experiment via North-Bound APIs by the 5G-EVE platform for step 10.
6 Conclusions

The integration of 5G Solutions with the supporting platforms was achieved and described in this document. The progresses in the different phases of the project (Cycles) are tracked. In each trial cycle the Living Labs developed 5G based solutions in different areas with respect to related business needs and deployed the entire chain of VNFs by means of a deployment framework depending on the 5G facility. In this document, the specific on-boarding and deployment framework and features are illustrated with respect to the two main 5G facilities, namely the supporting platforms 5G-VINNI and 5G-EVE, as well as the suitable integration activities to support the use case experiment.

The deliverable provides a complete reference on all the integration done with the supporting platforms in one document to help understanding the complexity and to compare the used tools.
7 References


8 Annex

8.1 Vertical Service Blueprints for Use Case integration in the 5G-EVE platform in LL2

- Use Case “Industrial Demand Side Management” (UC2.1) – Use Case Developer by Ares2t
  VSB_5G_Solutions_v1_UC21.json by Ares2t

```json
{
    "blueprintId": "vsb_UC2_1",
    "version": "1.0",
    "name": "vsb_UC2_1",
    "description": "Vertical Service Blueprint for UC 2.1",
    "atomicComponents": [
        {
            "componentId": "experiment_server",
            "serversNumber": 1,
            "endPointsIds": [
                "cp_experiment_server_ext_mobile",
                "cp_experiment_server_data",
                "cp_experiment_server_mgmt"
            ]
        }
    ],
    "parameters": [
        {
            "parameterId": "number_of_devices",
            "parameterName": "Number of devices",
            "parameterType": "number",
            "parameterDescription": "Number of Devices",
            "applicabilityField": "num devices"
        }
    ],
    "endPoints": [
        {
            "endPointId": "cp_experiment_server_ext_mobile",
            "external": true,
            "management": false,
            "ranConnection": true
        },
        {
            "endPointId": "cp_experiment_server_data",
            "external": false,
            "management": false,
            "ranConnection": false
        },
        {
            "endPointId": "cp_experiment_server_mgmt",
            "external": false,
            "management": true,
            "ranConnection": false
        },
        {
            "endPointId": "sap_experiment_server_ext_mobile",
            "external": true,
            "management": false,
            "ranConnection": true
        }
    ]
}
```
Use Case “Electric Vehicle Smart Charging” (UC2.2) – Use Case Developer by Ares2t
D2.1 Setup and operation of 5G infrastructure

VSB_5G_Solutions_v1_UC22.json by Ares2t

```json
{
  "blueprintId": "vsb_UC2_2",
  "version": "1.0",
  "name": "vsb_UC2_2",
  "description": "Vertical Service Blueprint for UC 2.2",
  "atomicComponents": [{
    "componentId": "charge_advisor",
    "serversNumber": 1,
    "endPointsIds": [
      "cp_charge_advisor_ext_mobile",
      "cp_charge_advisor_data",
      "cp_charge_advisor_mgmt",
    ]
  },{
    "componentId": "master_control_agent",
    "serversNumber": 1,
    "endPointsIds": [
      "cp_master_control_agent_data",
      "cp_master_control_agent_mgmt",
    ]
  },{
    "componentId": "mqtt_broker",
    "serversNumber": 1,
    "endPointsIds": [
      "cp_mqtt_broker_ext_mobile",
      "cp_mqtt_broker_data",
      "cp_mqtt_broker_mgmt",
    ]
  }],
  "parameters": [
    {
      "parameterId": "number_of_devices",
      "parameterName": "Number of devices",
      "parameterType": "number",
      "parameterDescription": "Number of Devices",
      "applicabilityField": "num devices"
    }
  ],
  "endPoints": [{
    "endPointId": "cp_charge_advisor_ext_mobile",
    "external": true,
    "management": false,
    "ranConnection": true
  }, {
    "endPointId": "cp_charge_advisor_data",
    "external": false,
    "management": false,
    "ranConnection": false
  }, {
    "endPointId": "cp_master_control_agent_data",
    "external": false,
    "management": false,
    "ranConnection": false
  }, {
    "endPointId": "cp_mqtt_broker_ext_mobile",
    "external": true,
    "management": false,
    "ranConnection": true
  }]
}
```
D2.1. Setup and operation of 5G infrastructure

```
"endPointId": "cp_charge_advisor_mgmt",
"external": false,
"management": true,
"ranConnection": false
},
{"endPointId": "cp_master_control_agent_data",
"external": false,
"management": false,
"ranConnection": false
},
{"endPointId": "cp_master_control_agent_mgmt",
"external": false,
"management": true,
"ranConnection": false
},
{"endPointId": "cp_mqtt_broker_ext_mobile",
"external": true,
"management": false,
"ranConnection": false
},
{"endPointId": "cp_mqtt_broker_data",
"external": false,
"management": false,
"ranConnection": false
},
{"endPointId": "cp_mqtt_broker_mgmt",
"external": false,
"management": true,
"ranConnection": false
},
{"endPointId": "sap_experiment_server_ext_mobile",
"external": true,
"management": false,
"ranConnection": true
},
{"endPointId": "sap_experiment_server_mgmt",
"external": false,
"management": true,
"ranConnection": false
},
"connectivityServices": [
  {
    "endPointIds": [
      "sap_ext_mobile",
      "cp_charge_advisor_ext_mobile",
      "cp_mqtt_broker_ext_mobile"
    ],
    "external": true,
    "management": false,
    "name": "vl_ext_mobile"
  }
]
```
D2.1. Setup and operation of 5G infrastructure

```json
{
    "endPointIds": [
        "cp_charge_advisor_data",
        "cp_mqtt_broker_data",
        "cp_master_control_agent_data"
    ],
    "external": false,
    "management": false,
    "name": "vl_data"
},
{
    "endPointIds": [
        "sap_experiment_server_mgmt",
        "cp_experiment_server_mgmt",
        "cp_mqtt_broker_mgmt",
        "cp_experiment_server_mgmt"
    ],
    "external": false,
    "management": true,
    "name": "vl_mgmt"
}
```

- **Use Case “Energy Network Frequency Regulation” (UC2.3) – Use Case Developer by Ares2t**

  VSB_5G_Solutions_v1_UC23.json by Ares2t

  ```json
  {
    "blueprintId": "vsb_UC2_3",
    "version": "1.0",
    "name": "vsb_UC2_3",
    "description": "Vertical Service Blueprint for UC 2.3",
    "atomicComponents": [{
        "componentId": "local_control_agent",
        "serversNumber": 1,
        "endPointIds": [
            "cp_local_control_agent_ext_mobile",
            ...
        ]
    }]
  }
  ```
"cp_local_control_agent_data",
"cp_local_control_agent_mgmt",

"componentId": "master_control_agent",
"serversNumber": 1,
"endPointsIds": [
  "cp_master_control_agent_data",
  "cp_master_control_agent_mgmt",
]

"componentId": "mqtt_broker",
"serversNumber": 1,
"endPointsIds": [
  "cp_mqtt_broker_ext_mobile",
  "cp_mqtt_broker_data",
  "cp_mqtt_broker_mgmt",
]

"parameters": [
  {
    "parameterId": "number_of_devices",
    "parameterName": "Number of devices",
    "parameterType": "number",
    "parameterDescription": "Number of Devices",
    "applicabilityField": "num devices"
  }
],
"endPoints": [
  {
    "endPointId": "cp_local_control_agent_ext_mobile",
    "external": true,
    "management": false,
    "ranConnection": true
  },
  {
    "endPointId": "cp_local_control_agent_data",
    "external": false,
    "management": false,
    "ranConnection": false
  },
  {
    "endPointId": "cp_local_control_agent_mgmt",
    "external": false,
    "management": true,
    "ranConnection": false
  },
  {
    "endPointId": "cp_master_control_agent_data",
    "external": false,
    "management": false,
    "ranConnection": false
  },
  {
    "endPointId": "cp_master_control_agent_mgmt",
    "external": false,
    "management": true,
"ranConnection": false,
"endPointId": "cp_mqtt_broker_ext_mobile",
"external": true,
"management": false,
"ranConnection": false
},

"endPointId": "cp_mqtt_broker_data",
"external": false,
"management": false,
"ranConnection": false
},

"endPointId": "cp_mqtt_broker_mgmt",
"external": false,
"management": true,
"ranConnection": false
},

"endPointId": "sap_experiment_server_ext_mobile",
"external": true,
"management": false,
"ranConnection": true
},

"endPointId": "sap_experiment_server_mgmt",
"external": false,
"management": true,
"ranConnection": false
}},

"connectivityServices": [{
"endPointIds": [
"sap_ext_mobile",
"cp_local_control_agent_ext_mobile",
"cp_mqtt_broker_ext_mobile"
],
"external": true,
"management": false,
"name": "vl_ext_mobile"
},

"endPointIds": [
"cp_local_control_agent_data",
"cp_mqtt_broker_data",
"cp_master_control_agent_data"
],
"external": false,
"management": false,
"name": "vl_data"
},

"endPointIds": [
"sap Experiment Server Mgmt",
"cp Experiment Server Mgmt",
"cp Mqtt Broker Mgmt"
"cp_experiment_server_mgmt"
},
"external": false,
"management": true,
"name": "vl_mgmt"
}
],
"compatibleContextBlueprint": [
  "ctx_delay"
],
"applicationMetrics": [{
  "metricId": "UC23_topic_id",
  "name": "topic_id",
  "metricCollectionType": "GAUGE",
  "unit": "%",
  "interval": "1s",
  "metricGraphType": "LINE"
}],
"compatibleSites": [
  "ITALY_TURIN"
]