



6G-SANDBOX

Supporting Architectural and technological Network evolutions through an intelligent, secureD and twinning enaBled Open eXperimentation facility

1st Open Call

Deliverable D3.1

Description of functionalities/infrastructures and user manual

6G-LoRaGRAN project: Integration of the University of Granada's LoRaWAN network in the 6G SANDBOX connectivity infrastructure



University of Granada

Date: 29/02/2024

Version: 1.0



Grant Agreement	101096328
Document number	D3.1
Document title	Description of functionalities/infrastructures and user manual
Lead Beneficiary	University of Granada
Editor(s)	Jorge Navarro-Ortiz
Author(s)	Jorge Navarro-Ortiz
Dissemination level	Public
Contractual date of delivery	29/02/2029
Status	Initial version
Version	1.0
File name.	6G-LoRaGRAN_D3.1_V1.0.pdf

Document History

Version

V1.0 Initial version of the deliverable



TABLE OF CONTENTS

Abstrac	t
Keywor	ds4
Glossar	y4
1 Inti	oduction5
1.1	Objectives and scope of this document5
1.2	Document structure5
2 Des	cription of functionalities/infrastructure6
2.1	6G-LoRaGRAN infrastructure6
2.2	6G-LoRaGRAN Functionalities9
3 Use	er manual10
3.1	Register a new user10
3.2	Create an experiment
3.3	Create traffic patterns
3.4	Create LoRaWAN network slices14
3.5	Assign traffic patterns and network slices to specific devices15
3.6	Enable or disable the LoRaWAN gateways15
3.7	View status of devices, gateways and connectivity towards pilot sites16
3.8	View LoRaWAN frames in the ChirpStack platform17
3.9	Subscribe to an MQTT broker to receive LoRaWAN frames in real-time
3.10	Use the REST API to check, verify or modify the settings of the experiment in real time 19
4 Cor	clusions
Referen	nces



Figure 1. PyCom FiPy and TTGO LoRa32 nodes.	6
Figure 2. CISCO gateway at the roof of ETSIIT-UGR	7
Figure 3. Pair IMST Lite and Pycom Pygate gateways (example of gateways located at CSIC-UGR)	7
Figure 4. Location of 6G-LoRaGRAN gateways.	8
Figure 5. Rack cabinet with 6G-LoRaGRAN cluster	8
Figure 6. 6G-LoRaGRAN testbed architecture.	9
Figure 7. a) Login and b) registration pages	11
Figure 8. a) Registration complete pages and b) mail sent to the user	11
Figure 9. Home web page	12
Figure 10. a) Navigation bar and b) profile details.	12
Figure 11. Creation of a new experiment.	13
Figure 12. Creation of a new traffic pattern.	14
Figure 13. Creation of a new network slice.	14
Figure 14. Assignment of traffic patterns and network slices to devices.	15
Figure 15. Configuration of LoRaWAN gateways	16
Figure 16. Status of devices.	16
Figure 17. ChirpStack – Status of gateways	17
Figure 18. ChirpStack – LoRaWAN frames of one device in the application server.	18
Figure 19. Instructions on the home page to subscribe to the user's MQTT topic.	19
Figure 20. REST API – Authorization.	20
Figure 21. REST API methods for 6G-LoRaGRAN.	22
Figure 22. REST API – GET /pilotsites/all/status.	23
Figure 23. REST API – GET /experiments/all.	24
Figure 24. REST API – POST /slices/{slice_id}	25



ABSTRACT

The 6G-LoRaGRAN project, granted to the WiMuNet research group (code TIC-235 of the Scientific Information System of Andalusia) of the University of Granada (Spain), is intended to integrate the University of Granada's LoRaWAN network with the 6G-SANDBOX pilot sites. Additionally, it enables researchers from 3rd parties to conduct experiments using this infrastructure.

The objective of this deliverable is to provide a comprehensive description of the project's functionalities, encompassing the infrastructure used and the user manual.

Keywords

6G-SANDBOX, 6G-LoRaGRAN, LoRaWAN, N3IWF, 5GC, Network Slicing

GLOSSARY

5GC	5G Core network
AMF	Access and Mobility Management Function
BW	Bandwidth
CR	Coding Rate
eNB	3GPP 4G evolved Node B
IKEv2	Internet Key Exchange version 2
IPSec	Internet Protocol Security
ISM	Industrial, scientific, and medical
gNB	3GPP 5G next Generation Node B
LoRa	Long Range modulation
LoRaWAN	Long Range Wide Area Network
MQTT	Message Queuing Telemetry Transport
N3IWF	Non-3GPP InterWorking Function
N6	Interface between the Data Network (DN) and the UPF
NWu	Reference point between the UE and N3IWF for establishing secure tunnel(s)
UPF	User Plane Function
RAN	Radio Access Network
REST API	REpresentational State Transfer Application Programming Interface
SF	Spreading Factor
UE	User Equipment
VPN	Virtual Private Network



1 INTRODUCTION

The 6G-LoRaGRAN project [1], granted to the WiMuNet research group [2] (code TIC-235 of the Scientific Information System of Andalusia) of the University of Granada (Spain), integrates the University of Granada's LoRaWAN network into the 6G-SANDBOX connectivity infrastructure.

6G-LoRaGRAN enables researchers to conduct experimental campaigns using the University of Granada (UGR) LoRaWAN infrastructure, comprising a LoRaWAN Radio Access Network (RAN) with gateways and nodes, as well as a LoRaWAN backend with network and application servers. These components are integrated with the 5G cores situated at one of the 6G-SANDBOX pilot sites, facilitating the transmission of all LoRaWAN traffic across the 5G network. In this way, a mobile network operator would be able to offer LoRaWAN (a non-3GPP network) services to their customers. For that purpose, 6G-LoRaGRAN equipment can connect to the 5G cores 1) via the N3IWF (Non-3GPP InterWorking Function) entity (using a NWu emulator) or 2) by emulating a UE/gNB, depending on the 6G-SANDBOX pilot sites framework.

To enable third parties to conduct experiments on our testbed, we have developed a web UI for our Experiment Manager (EM). Within this interface, researchers can reserve time slots for their experiments and customize the required parameters. This includes the ability to modify node traffic patterns and create network slices. Nodes will transmit frames based on a customizable frame arrival pattern and with a customizable frame size distribution.

Additionally, researchers are able to create various network slices, each comprising one or more channels that are not shared between slices. Slice assignment to the nodes is initially assigned through the web interface but it can be modified in real-time using a REST API (REpresentational State Transfer Application Programming Interface). This enables adjustments on the channel assignment based on real-time metrics, thus allowing researchers to implement their own assignment strategies.

Finally, researchers can subscribe to LoRaWAN metrics and data from the application server, which includes radio statistics, transmission parameters, transmitted data, etc. As stated before, this information can be used for real-time resource assignment algorithms or for performance evaluation analysis.

1.1 OBJECTIVES AND SCOPE OF THIS DOCUMENT

In accordance with the terms of the call, the objective of this deliverable is the description of the functionalities and infrastructure made available to researchers through the 6G-LoRaGRAN project and the provision of a user manual.

1.2 DOCUMENT STRUCTURE

The rest of this document is structured as follows:

- Section 2: Overview of the functionalities and infrastructure of the 6G-LoRaGRAN project. This section describes in detail the infrastructure in the University of Granada for 6G-LoRaGRAN, which can be used by researchers.
- Section 3: User manual. This section provides a detailed description of how a researcher can carry out experimentation using the UGR LoRaWAN network integrated with the 6G-SANDBOX pilot sites.



• Section 4 draws the main conclusions from this document.

2 DESCRIPTION OF FUNCTIONALITIES/INFRASTRUCTURE

The following subsections describe the infrastructure and the functionalities developed during the 6G-LoRaGRAN project.

2.1 6G-LORAGRAN INFRASTRUCTURE

As described in Section 1, the 6G-LoRaGRAN infrastructure comprises a LoRaWAN Radio Access Network, a LoRaWAN backend and an Experiment Manager. Both the LoRaWAN backend and the Experiment Manager are executed on a Kubernetes [3] cluster to simplify management and improve reliability. In addition, another computer provides connectivity towards the 5G cores from the 6G-SANDBOX sites using a UE/gNB emulator or a NWu client.

The RAN includes:

 LoRaWAN nodes: 25 Pycom Fi-Py [4] and 15 TTGO LoRa32 [5] devices. These nodes connect via Wi-Fi to the EM to receive their transmission parameters (e.g. channel, spreading factor, slice) and traffic patterns to accordingly generate LoRaWAN traffic. Their firmware includes watchdog timers to automatically recover after non-expected failures and can be updated over-the-air (OTA) to simplify updates.





Figure 1. PyCom FiPy and TTGO LoRa32 nodes.

- LoRaWAN gateways:
 - One outdoor gateway. As part of this project, UGR has deployed a CISCO IXM-LPWA-800-16-K9 [6], a carrier-grade product specifically designed for outdoor environments. It has been deployed to enhance coverage around the School of Technology and Telecommunications Engineering (ETSIIT-UGR) and the Research Center for Information and Communication Technologies (CITIC-UGR).







Figure 2. CISCO gateway at the roof of ETSIIT-UGR.

Several indoor gateways. Several IMST Lite Gateway [7] and Pycom Pygate [8] gateways have been deployed in the WiMuNet laboratory as well as in different offices of the ETSIIT-UGR and CITIC-UGR buildings (see an example in Figure 3). They are placed as pairs for redundancy.



Figure 3. Pair IMST Lite and Pycom Pygate gateways (example of gateways located at CSIC-UGR).



The location of the gateways deployed is summarized in Figure 4, and is included in the home page of the web UI for reference.



Figure 4. Location of 6G-LoRaGRAN gateways.

Regarding our LoRaWAN backend, we employ the ChirpStack platform [9], which has been curstomized to be executed on a Kubernetes cluster, to automatize deployment, scaling, and management of the containerized applications as well as to provide high availability. Kubernetes also manages data persistence and automatic recovery. The cluster is composed of three Intel NUCs with i7-10710U processors and 16 GB of RAM, in addition to the required Ethernet switches.



Figure 5. Rack cabinet with 6G-LoRaGRAN cluster.

We also employ another Intel NUC, acting as N3IWF (Non-3GPP InterWorking Function) client or gNB/UE emulator, to provide connectivity towards the 6G-SANDBOX sites. Finally, an Intel Core i3 desktop provides connectivity from our private network to the Internet.

Figure 6 provides an overview of the 6G-LoRaGRAN testbed architecture. As previously mentioned, the Radio Access Network (RAN) is attached to a computer, which then directs LoRaWAN traffic through a 5G core to the cluster housing the LoRaWAN backend. This integration seamlessly combines both LoRaWAN and 5G functionalities within the 6G-LoRaGRAN framework.



Figure 6. 6G-LoRaGRAN testbed architecture.

2.2 6G-LORAGRAN FUNCTIONALITIES

The functionalities achieved within this project can be categorized as:

- 1) Connectivity to 6G-SANDBOX pilot sites.
- 2) Deployment and operation of the LoRaWAN network.
- 3) Execution of experiments

Regarding the connectivity to 6G-SANDBOX pilot sites, the objective is to connect the LoRaWAN RAN to the LoRaWAN backend via the 5G cores in the different sites. Since LoRaWAN is a non-3GPP network, it can be connected 1) via N3IWF or 2) emulating a UE/gNB, depending on the selected 5G core. The web UI of the Experiment Manager allows the researcher to select the 6G-SANDBOX pilot site to integrate with.

Regarding the deployment and operation of the LoRaWAN network, we employ the ChirpStack platform [9] in a Kubernetes cluster. Our implementation, using Kubernetes and the ChirpStack API, automatizes the deployment of the ChirpStack platform, the automation to the creation/modification of security credentials, and the automation of the configuration of the LoRaWAN network (including the definition of tenants, applications, gateways, nodes, and LoRaWAN network slices).

Regarding the execution of experiments, 6G-LoRaGRAN allows researchers to perform campaigns of experiments using UGR's LoRaWAN testbed. The Experiment Manager includes a web UI which allows users to define, modify and delete the required parameters for the experiments. This includes the booking of a time slot for the experiment, selecting a 6G-SANDBOX pilot site to be integrated into. For the sake of fairness, researchers are initially allocated a maximum of 48 hours for experimentation, meaning they have to wait for the experiments to finish before being able to reserve more time. A user can contact the 6G-LoRaGRAN team (loragran@ugr.es) if he/she requires further resources.

During an experiment, traffic generation can be customized. This is done by defining traffic patterns for the nodes. Nodes will generate one frame following a frame arrival distribution (currently uniform and normal distributions are supported) with a given frame size (also following a uniform or a normal distribution).

Also, LoRaWAN network slices are used during the experiment. In the 6G-LoRaGRAN platform, network slices are defined as a set of channels that cannot be shared between slices. The slice assignment to the nodes, as well as the initial channel and spreading factor, is done via the web UI. A



REST API is also available which allows researchers to modify the slices and the channel assignment in real-time based on metrics. These metrics can be obtained by subscribing to the MQTT (Message Queuing Telemetry Transport) broker [10] executed along the ChirpStack platform, which publishes LoRaWAN frames with metadata information (reception time, RSSI, SNR, channel, spreading factor, frame counter, etc.).

The following section summarizes the guidelines to use the 6G-LoRaGRAN platform.

3 USER MANUAL

This section provides a step-by-step guide for users that want to use the 6G-LoRaGRAN platform. As a summary, the next subsections will introduce how to:

- a) Register a new user
- b) Book a time slot for experimentation, i.e. create an experiment
- c) Create traffic patterns for the LoRaWAN devices
- d) Create network slices for the different service operators
- e) Assign traffic patterns and network slices to specific devices
- f) Enable or disable the LoRaWAN gateways
- g) Check the status of devices, gateways and connectivity towards pilot sites
- h) View LoRaWAN frames (including radio link conditions, transmission parameters, transmitted data) in the ChirpStack platform
- i) Subscribe to an MQTT broker to receive LoRaWAN frames in real time
- j) Use the REST API to check, verify or modify the settings of the experiment in real time

3.1 REGISTER A NEW USER

Open a browser and connect to <u>https://loragran.ugr.es</u>. If you have not registered/logged in before, it will show the login page (Figure 7a). There, you shall select "Create an account", which will forward to the registration page (Figure 7b). It will ask for a username, an e-mail address, a tenant name (which may be shared between researchers from the same team), an organization and a password. All these settings can be later modified along with other related information such as first/last names, address, phone, avatar picture, etcetera.



	👻 🤪 6G-LoRa C 🗙 💿 Nueva pe 🗙 + 🛛 🗖	×								
	← → C 😄 loragran.ugr.e 🖙 ☆ 🏠 🔲 🌒	:								
$\begin{array}{c c c c c c c } & & & & & & \\ \hline \bullet & $	6G-LORAGRAN PROJECT									
	Welcome to the 6G-LoRaGRAN Experiment Manager!									
	Please register a new account									
6170	Username									
6G-LORAGRAN PROJECT	Jorgenavano									
	E-mail									
Welcome to the 6G-LoRaGRAN	Jorgonatano@agneo									
Please sign-in to your account	UGR									
E-mail	Organization									
Password 🗞	Password &									
Remember Me Forgot Password?	✓ I agree to privacy policy & terms									
SIGN IN	SIGN UP									
New on our platform? Create an account	Already have an account? Sign in instead									
Know more about the 6G-LoRaGRAN project here	Know more about the 6G-LoRaGRAN project here									

Figure 7. a) Login and b) registration pages.

An e-mail (Figure 8a) will be sent to verify that the user owns that address.



Figure 8. a) Registration complete pages and b) mail sent to the user.

Once the user has clicked the link in the mail (Figure 8b), the user will be notified that the administrator has to approve the request. This is to avoid spam or unwanted behaviors.



When the administrator approves your request, you will be notified by mail, and you will be able to login using the login web page (<u>https://loragran.ugr.es/login.php</u>). The home page will be shown (Figure 9).



Figure 9. Home web page.

The home page includes 1) a menu on the left, 2) a navigation bar at the top right and 3) the web content. The pages included in the menu are categorized in "Home", "Admin" (only for administrators), "Experiment Manager", "ChirpStack", "API" and "About", which will be described in the next subsections. The navigation bar allows the user to log out or to modify the user's profile (Figure 10a). The user can also delete his/her account on this page.

	Profile Details					
	UPLOAD NEW PHOTO Allowed JPG, GIF or PNG. Max size of 2M					
	First Name	Last Name				
13:11:16 UTC	- Username Jorgenavarro	E-mill jorgenavarro@ugr.es				
	- Tenant- UGR	Organization Universidad de Granada				
jorgenavarro@ugr.es Universidad de Granada	-Country Spain	Granada				
	Phone Number +34958241000, ext. 200422	Address ETSI Informatica y Telecomunicacion, c/ Periodista 2Daniel Saucedo Aranda s/n, 18071 Grs				
🛎 My Profile	Password					
ப் Log Out	SAVE CHANGES					

Figure 10. a) Navigation bar and b) profile details.

3.2 CREATE AN EXPERIMENT

To book a time slot to experiment with the 6G-LoRaGRAN platform, the user has to select "Experiment" in the menu. There, the user will see his/her experiments as well as those from other users (anonymized), allowing the user to determine when the platform is busy. An "instructions" button will summarize what the user can do in this page, which is also available in all the pages related



to the Experiment Manager. If the user presses "Add new experiment", he/she can select the start and stop times of the experiment and the pilot site to be integrated into, as well as its title and its description (Figure 11).

👻 😂 6G-LoRaGRAN Experiments 🗙 +									σ×
← → Ø 🔄 loragran.ugr.es/experiments.php							\$	0 0	0 1
6G-LORAGRAN							8:33:46 UTC		
- HOME	Experiments /	JGR							
🔂 Home									
EXPERIMENT MANAGEN	Experiments								
A Experiments									
O Silces			Experiment 13		×				
🗠 Traffic Patterns	ADD NEW EXP	ERIMENT	tenant.	experiment					
(+) Devices	ACTIONS	EXPERIMENT		title		START TIME (UTC)	STOP TIME (UTC)		
곱 Gateways		12	BERLIN -	title of experim	ient	2024/02/26 08:33	2024/02/26 08:35		
∏ Status			26/02/2024 09:00	stop time 26/02/2024 09	:59 🗖	2024/02/26 07:10	2024/02/26 07:14		
			description			2024/02/26 07:05	2024/02/26 07:09		
CHIRPSTACK		9	description of experiment			2024/02/26 07:00	2024/02/28 07:04		
ChirpStack		8		CANCEL	AVE CHANGES	2024/02/25 22:01	2024/02/25 23:59		
				UNITOLE		2024/02/25 21:56	2024/02/25 22:00		
API		6				2024/02/25 21:50	2024/02/25 21:55		
MOTT		5				2024/02/25 17:10	2024/02/25 17:15		
REST		4				2024/02/22 20:54	2024/02/22 23:59		
ABOUT		3	direct connection	title	description	2024/02/21 23:19	2024/02/21 23:59		
A About				*******	*******	2024/02/18 09:20	2024/02/18 09:25		
						2024/02/18 00:40	2024/02/18 00:45		

Figure 11. Creation of a new experiment.

If the user's time slot overlaps with other experiments, an alert will be shown, allowing the user to modify the start and stop times.

This page also allows the user to modify or delete his/her existing experiments. Note that, to allow fair use of the platform, each user can create experiments for a maximum duration of 48 hours. This applies to both current and future experiments, excluding those already completed.

3.3 CREATE TRAFFIC PATTERNS

By default, a traffic pattern with a uniform distribution for the frame arrivals (min 300 seconds, max 600 seconds) and a uniform distribution for the frame size (min 10 bytes, max 20 bytes). This default traffic pattern is not modifiable. But the user can create, modify and delete other traffic patterns by selecting the "Traffic Patterns" menu and then pressing the "Add new pattern" button. The user can select uniform or normal distribution for frame arrivals and for frame sizes. For the uniform distribution, param1 is the minimum value and param2 is the maximum value. For the normal distribution, param1 is the mean and param2 is the standard deviation.



👻 😺 6G-LoRaGRAN Traffic Patterns 🗙 +							- 0	×
← → ♂ 🖾 loragranugr.es/trafficpatterns.php						☆ ひ		8 1
GG-LORAGRAN					1	8:34:07 UTC	1	
HOME Traffic Patterns / UGR								
ехревниент маладен								
A Experiments	New Pattern		83	×				
O Silces	terant	patte	m :					
Traffic Patterns	UGR	2		ERAME SIZE DISTRIBUTION	PADAMI	DADAMO		
(+) Devices		Frame arrival distrib	ition	uniform				
🔓 Gateways i 1	NORMAL -	param1 300	param2 20	uniform				
I Status								
© 2023, powered by ThemeSelectio		Frame size distribut	param2 10					
ChirpStack								
		CANCE	SAVE CHANGES					
MOTT								
REST								
8 About								

Figure 12. Creation of a new traffic pattern.

3.4 CREATE LORAWAN NETWORK SLICES

As described before, the platform allows the users to create network slices (Figure 13), each comprising one or more channels that are not shared between slices. To define a slice, we employ the concept of channel mask, which is a binary mask in which each bit represents one channel of the eight channels used for LoRaWAN in the in the 868 MHz band. For example, a channel mask of 15 (00001111 in binary representation) means that the slice includes channels 1, 2, 3, and 4; and a channel mask of 255 (11111111 in binary representation) means that all the channels are included in the slice. Note that it is up to the researcher whether slices share or not channels. To achieve isolation, slices shall not have common bits set to 1.

👻 😂 6G-LoRaGRAN Slices 🗙 +							2	- 0	×
← → C S loragran.ugr.es/slices.php						\$	Ð		8 1
6G-LORAGRAN						8:34:22 UTC	¢?		
Home	Slices / UGR								
EXPERIMENT MANAGER	Slices								
	ADD NEW SLICE	New Slice	stor	× 340	IANNEL MASK				
🖀 Gateways	i.	UGR channel mask 15	2	:					
CHINPETROK	© 2023, powered by ThemeSelectio		CANCEL SAVE C	HANGES					
API MOTT REST ADOUT 8 About									

Figure 13. Creation of a new network slice.



3.5 Assign traffic patterns and network slices to specific devices

Once the user has created and customized traffic patterns and network slices, he/she has to assign them to specific devices for a particular experiment (Figure 14). For that, the user can select the "Devices" menu, which will show the configuration of the devices for future experiments (past experiments are not shown). There, the user can select the slice and traffic pattern for each device or select several devices and apply the same configuration (traffic pattern and network slice) for the selected devices. The user can also select the spreading factor to be employed for each device. Similarly, the user can select the channel used for transmission or "periodic", which means that the device will hop between the channels included in the selected slice.

👻 😝 6G-LoRaGRAN Devices 🗙 +											- 0	×
← → C 🛱 loragran.ugr.es/devices.php									\$	Ð		8 3
6G-LORAGRAN	Devices	/ UGR										
номе	Config	uration of device	5									
☆ Home	INST	RUCTIONS										
EXPERIMENT MANAGER	EXPE	RIMENT 13 -										
A Experiments												
O Slices	SLICE		TRAFFIC P	ATTERN	CHANNEL		SPREADING FACTOR	1				
🗠 Traffic Patterns	1	•	2 -		4 -		7 -					
(H) Devices	APPL	Y CONFIGURATION TO	SELECTED DEVIC	ES	periodic							
🔓 Gateways	_				2							
👔 Status		NAME	MODEL	LOCATION	FW 3		SLICE	TRAFFIC PATTERN	CHANNEL			
CHIRPSTACK		loragran-pycom01	Pycom FiPy	ETSIIT 2nd floor (WiMuNet lab)	1.0 4 5	.9449D5B370	0 -	0 -	PERIOD	IC		
ChirpStack		loragran-pycom02	Pycom FiPy	ETSIIT 2nd floor (WiMuNet lab)	1.0 ⁶ 7	9849D5B370	1 •	2 -	4 -			
API		loragran-pycom03	Pycom FiPy	ETSIIT 2nd floor (WiMuNet lab)	1.0. 8	9349D5B370	1+	2 -	4 +			
REST		loragran-pycom04	Pycom FiPy	ETSIIT 2nd floor (WiMuNet lab)	1.0.11	CB93089549D5B370						
ABOUT		5 11-11-1							PERIOD			
A About		loragran-pycom05	Pycom FiPy	ETSIIT 2nd floor (WiMuNet lab)	1.0.0	36E9129049D5B370	0 -	0 -	PERIOD	IC.		
javascript.modifySelectedParameterForSelectedDevices('UGR', 13, 3, 5);		loragran-pycom07	Pycom FiPy	ETSIIT 2nd floor (WiMuNet lab)	1.0.0	D7D7059949D58370	0 -	0 -	PERIOD	IC		

Figure 14. Assignment of traffic patterns and network slices to devices.

Please note that these device settings are for a specific experiment, so the user has to repeat this procedure for each experiment (unless the user wants all the devices to use the default traffic pattern and network slice).

3.6 ENABLE OR DISABLE THE LORAWAN GATEWAYS

By default, all the LoRaWAN gateways belonging to the 6G-LoRaGRAN platform are enabled during the experiment. These gateways are located in the School of Technology and Telecommunications Engineering (ETSIIT-UGR) of the University of Granada and in the Research Centre for Information and Communications Technologies (CITIC-UGR), as shown in Figure 4.

The 6G-LoRaGRAN platform allows the users to select which gateways are enabled or disabled for a given experiment (Figure 15). This increases the possibilities of experimentation by being able to vary radio conditions based on whether closer or more distant gateways are enabled. This can be customized by selecting the "Gateways" menu, selecting the required location (which can be "All" to enable all the gateways) and pressing the "Activate gateways" button. The user can also enable/disable each gateway individually.



🗸 😂 6G-LoRaGRAN Gateways 🗙 +						
← → ♂ III loragran.ugr.es/gateways.php						\$
6G-LORAGRAN						8:35:37 UTC
HOME	Gateways / UGR					
EXPERIMENT MANAGER	Configuration of gate	eways				
A Experiments	INSTRUCTIONS					
Slices	EXDEDIMENT 12 -					
🗠 Traffic Patterns	EXPERIMENT 13 *					
(+) Devices	ACTIVATE ONLY GATEWAY	S AT THIS LOCATION				
🛱 Gateways	CITIC 2ND FLOOR -					
7 Status	ACTIVATE GATEWAYS					
CHIRPSTACK						
ChirpStack	NAME	MODEL	LOCATION	FW VERSION	GATEWAYEUI	ENABLED
API	loragran-imst04	IMST Lite Gateway	CITIC 2nd floor	1.0.0	B827EBFFFE3F06BD	TRUE -
MQTT	loragran-pygate04	Pycom Pygate	CITIC 2nd floor	1.0.1	840D8EFFFE1177D0	TRUE -
♥ REST ABOUT	loragran-imst01	IMST Lite Gateway	ETSIIT 2nd floor (WiMuNet lab)	1.0.0	B827EBFFFE09D416	FALSE -
A About	loragran-imst02	IMST Lite Gateway	ETSIIT 2nd floor (WiMuNet lab)	1.0.0	B827EBFFFEEA9756	FALSE -

Figure 15. Configuration of LoRaWAN gateways.

3.7 VIEW STATUS OF DEVICES, GATEWAYS AND CONNECTIVITY TOWARDS PILOT SITES

The 'Status' menu allows to check that the devices (Figure 16. Status of devices.) and gateways are working properly (tabs 'Devices' and 'Gateways'). In case there is some failure, the user can contact the LoRaGRAN team (<u>loragran@ugr.es</u>) for assistance. Similarly, the user can check that the 6G-LoRaGRAN platform is connected to the 5G cores from the different 6G-SANDBOX pilot sites ('Pilot Sites' tab). This allows to confirm the correct behavior of the platform during your experiments (for this, the user can also use the REST API, see Section 3.10).

👻 😝 6G-LoRaGRAN Status 🛛 🗙 🕂								-	σ×
← → ♂ ♯ loragran.ugr.es/info.php							☆ Ď	* 🛛	e :
6G-LORAGRAN							9:40:30 UTC	2	ĺ
HOME	Status / UGR								
EXPERIMENT MANAGER	DEVICES GATEWAYS	PILOT SITES							
囚 Experiments	Devices								
Slices									
Z Traffic Patterns	INSTRUCTIONS Sorted by name								
(+) Devices	NAME	MODEL	DEVEUI	FIRMWARE VERSION	LOCATION	LAST SEEN	LAST RESET		
😤 Gateways	loragran-pycom01	Pycom FiPy	8F96DE9449D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:40:25+00	2024-02-26 09		
- Ctatus	loragran-pycom02	Pycom FiPy	E1F7C09849D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:36:21+00	2024-02-26 09		
E Status	loragran-pycom03	Pycom FiPy	F047669349D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 08:33:59+00	2024-02-26 07		
CHIRPSTACK	loragran-pycom04	Pycom FiPy	CB93089549D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:36:14+00	2024-02-26 09		
ChirpStack	loragran-pycom05	Pycom FiPy	36E9129049D5B370	1.0.0	ETSIIT 2nd floor (WiMuNet lab)				
	loragran-pycom07	Pycom FiPy	D7D7059949D5B370	1.0.0	ETSIIT 2nd floor (WiMuNet lab)				
API	loragran-pycom11	Pycom FiPy	CA0E789949D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:39:28+00	2024-02-26 09		
MQTT	loragran-pycom13	Pycom FiPy	F1EEC59B49D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-22 17:27:29+00	2024-02-22 15:		
REST	loragran-pycom14	Pycom FiPy	7DD7719449D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:39:05+00	2024-02-26 09		
10017	loragran-pycom15	Pycom FiPy	8CAB2A9149D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:38:43+00	2024-02-26 09		
0	loragran-pycom16	Pycom FiPy	ED00589649D5B370	1.0.0	ETSIIT 2nd floor (WiMuNet lab)				
X About	loragran-pycom17	Pycom FiPy	0807E59E49D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:38:58+00	2024-02-26 09		
	loragran-pycom18	Pycom FiPy	4E7E839349D5B370	1.0.11	ETSIIT 2nd floor (WiMuNet lab)	2024-02-26 09:40:26+00	2024-02-26 09		

Figure 16. Status of devices.



3.8 VIEW LORAWAN FRAMES IN THE CHIRPSTACK PLATFORM

The 6G-LoRaGRAN platform uses ChirpStack as the LoRaWAN backend, which includes the functionalities of the network and application servers, among others. Users can open the ChirpStack platform web page by selecting "ChirpStack" in the menu (or directly <u>https://loragran.ugr.es:8443/</u>). If it is the user's first time, he/she can log in by using the 6G-LoRaGRAN username and password that he/she selected in the registration (Section 3.1).

After the start time of the experiment, the platform will automatically configure the pilot site, the traffic patterns and the network slices for the devices and enable/disable the required gateways. Devices and gateways are automatically assigned to the user's tenant in ChirpStack, which allows he/she to see the LoRaWAN frames in the gateways (ciphered) and in the application (deciphered).

For example, users can see whether the gateways are active or not (Figure 17 was taken with IOT-ETSIIT gateway enabled and other gateways disabled). Or the LoRaWAN frames received by the application server (Figure 18).

👻 👙 6G-LoRaGRAN Gateways 🛛 🗙 🐼	ChirpStack LoRaWAN ® Networ	× +				- o ×
🗧 🔆 🔿 🖸 🖾 longranugr.es.843/4/tenants/3333606-ab88-4e79-8174-51078e0e5678/gateways 🏠 🗖 🔮						
ChirpStack				Search	۹ ?	A, jorgenavarro@ugr.es ∨
UGR v	Tenants / UGR / Gat Gateways	eways			-	dd gateway Selected gateways
Regions		Last seen	Gateway ID	Name	Region ID	Region common-name
 Tenant Dashboard 	Online	2024-02-20 14:15:25	70d379fffe8e31c0	IOT-ETSIIT	eu868	EU868
A Users	Offline	2024-02-20 14:15:30	b827ebfffe09d416	loragran-imst01	eu868	EU868
ρ API Keys	Offline	2024-02-20 14:15:16	b827ebfffeea9756	loragran-imst02	eu868	EU868
Device Profiles	Offline	2024-02-20 14:15:28	b827ebfffe4cdb39	loragran-imst03	eu868	EU868
중 Gateways ■ Applications	Offline	2024-02-20 14:14:21	b827ebfffe3f06bd	loragran-imst04	eu868	EU868
	• Offline	2024-02-20 14:15:29	b827ebfffe4ea01a	loragran-imst05	eu868	EU868
	Offline	2024-02-20 14:15:30	807d3afffe947cc4	loragran-pygate01	eu868	EU868
	Offline	2024-02-20 14:14:21	840d8efffe1228b4	loragran-pygate02	eu868	EU868
	Offline	2024-02-20 14:15:28	840d8efffe1177d0	loragran-pygate04	eu868	EU868
	• Offline	2024-02-20 14:15:29	807d3afffe948df4	loragran-pygate05	eu868	EU868
	Offline	2024-02-20 14:15:16	840d8efffe1185c8	loragran-pygate06	eu868	EU868
	Offline	2024-02-20 14:15:25	807d3afffe948778	loragran-pygate08	eu868	EU868
						< 1 > 50 / page V

Figure 17. ChirpStack – Status of gateways.



ChirpStack			٩. ?	A jorgenavarro@ugr.es v
UGR 👻	Tenants / UGR / Applications / UGR-app / Devices / tor loragran-pycom18 device aud 4e7e839349d5b370	agran-pycom18	 phy_payload: () 3 keys mhdr: () 2 keys m_type: "UnconfirmedDataDown" 	
Regions Tenant	Dashboard Configuration 0744 lass Activa	ton Queue Events LoRaWAN frames	major: U 4 terns 0. 220 1: 215 2: 122	
Dashboard A Users			s: 128 payload: 0 3 keys f_port: 0 findr: 0 4 keys devaddr: "00000012"	
P API Keys	2024-02-20 14:09:17 (UnconfirmedDataD	Com DevAddr: 00000012 DevEUI: 4e7e859340d5b370 Bateway ID: B07d3af	f_cnt: 0 f_cnt: 0 f_ctrb: () 6 keys ack: faise	
Device Profiles	2024-02-20 14:09:17 (Q. UnconfirmedDataU	DevAddr: 00000012 DevEUI: 4476839349d56570	adr. true adr_ack_req: false class_b: false	
Gateways Applications	2024-02-20 14:07:24 (@ UnconfirmedDataD	Own DevAddr: 00000012 DevEuI: 4076839349d5b570 Gateway ID: 807d5af	f_opts_len: 0 f_pending: false f_opts: [] 0 items	
	2024-02-20 14:07:24 @ UnconfirmedDataU	DevAddr: 00000012 DevEuI: 4676039349d56570	frm_payload: [] 6 items 0: () 1 key NewChannelReq: () 4 keys	
	2024-02-20 14:06:19 (0, UnconfirmedDataD	Own DevAddr: 00000012 DevEul: 4e7e539349d5c570 Gateway ID: 807d3af	rres ch_index: 3 freq: 867100000 max_dr: 5	
	2024-02-20 14:06:18 @: UnconfirmedDataU	DevApar: 00000012 Dev8UI: 4e7e830349d8b870	min_dr: 0 1: 0 1 key NewChannelReq: 0 4 keys	
	2024-02-20 14:05:04 (0: UnconfirmedDataD	Own Devider: 00000022 DeviuI: 4e7630340650370 Gateway ID: 80763a4	rres ch_index 4 freq: 867300000 max_dr: 5	
	2024-02-20 14:05:04 @ UnconfirmedDataL	DevAddr: 00000012 DevEUT: 4676830549d5b370	min_dr: 0 2: () 1 key NewChannelReq: () 4 keys	
	2024-02-20 14:03:27 @ UnconfirmedDataD	Comm DevAdor: 00000011 DevEUI: 4e7e839340d50370 Gateway ID: 807d3a4	ch_index: 5 freq: 867500000 max_dr: 5	
	2024-02-20 14:03:26 Q: UnconfirmedDataU	DevAdor: 00000012 DevEUI: 4474630349050370	min_dr: 0 3: "DesStatusBen"	

Figure 18. ChirpStack – LoRaWAN frames of one device in the application server.

For more information about the functionalities of the ChirpStack platform, please refer to its web page [9].

3.9 SUBSCRIBE TO AN MQTT BROKER TO RECEIVE LORAWAN FRAMES IN REAL-TIME

The ChirpStack platform [9] includes the publication of LoRaWAN frames received by the application server (which includes the radio link conditions and the transmission parameters) to an MQTT broker [10]. In our case, it publishes these frames onto the MQTT broker of the 6G-LoRaGRAN platform, which has been customized so that users will only see the frames from your experiments.

Users can use any MQTT client to subscribe (receive) to these frames. For that, they will have to download the MQTT broker certificate at <u>https://loragran.ugr.es/libs/ca.crt</u> and configure their MQTT client with their username and password from the 6G-LoRaGRAN platform. As an example, users could use the following commands in Ubuntu:

```
$ sudo apt-get update
$ sudo apt-get install mosquitto-clients
$ mosquitto_sub -h loragran.ugr.es -p 31883 -u <username> -P <password>
    -t "application/<application ID in ChirpStack>/#" --cafile <path to ca.crt>
```

This information, including user's application ID, is included on the home page (Figure 19). Last 10 MQTT messages are also included.





Figure 19. Instructions on the home page to subscribe to the user's MQTT topic.

With these frames, users can check the transmission parameters (e.g. the channel, the spreading factor, etc.) employed by each device and its radio link conditions (e.g. RSSI, SNR, etc.). You may use this information for resource allocation algorithms considering network slicing or for any other research, for example, by implementing a Python program with the *paho-mqtt* library [11]. In the next subsection we will explain how to obtain information from the 6G-LoRaGRAN platform and modify its parameters in real-time.

3.10 Use the REST API to check, verify or modify the settings of the experiment

IN REAL TIME

The 6G-LoRaGRAN platform includes a REST API which allows users to request the status and configuration of the Experiment Manager and to modify its parameters. This REST API can be tested directly using the "REST API" menu or by sending HTTP requests from, for example, a Python program (which may receive MQTT data as commented).

To use the web interface for the REST API ("REST API" menu, or directly <u>https://loragran.ugr.es:9443/docs</u>), a user has first to "authorize" its usage, i.e. press the "Authorize" button and enter his/her username and password from the 6G-LoRaGRAN platform (Figure 20). The authorization token required for the REST API can also be obtained using the "GET /token" request with the user's username and password. This is required for any other REST API request.



👻 👙 6G-LoRaGRAN Admin	× Ø REST API for the 6G-LoRaGRAN × +			- o x
< → ♂ 😫 loragran.ug	r.es:9443/docs			👓 🖈 🖸 🔲 🥮 🗄
	REST API for the 6G-Lo	oRaGRAN project an and		
	1	Available authorizations	X Authorize	
	Users Post /teken Orl Autoritation Taken OCT /users/me/ Orl User Details	Scopes are used to grant an application different levels of access to data on behalf of the end user. Each API may docker one on more scopes. API requestes the biothysics scopes. Scalar which ones you want to grant to Smagper UI. OAuth2PasswordBearer (OAuth2, password) Titem URL: totals Plore passions username:		기 기
	Pilot sites OLT /pilotsites/sil/status Get The Convolution OLT /pilotsites/(id)/status Get The Convolution	Deservoit Deservoit Client credentials location: Authorization header ∨ client_isi:		
	Devices GET /devices/all/status Out The Current Da GET /devices/{deveu}/status Out The Cu	clent_secret:	^ ۵	 ▼ ▼
	Gateways		^	

Figure 20. REST API – Authorization.

After that, the user can select any of the following methods:

- Users:
 - GET /token (get authentication token)
 - GET /users/me (get user details)
- Pilot sites:
 - GET /pilotsites/all/status (get the current status of the connectivity towards all pilot sites)
 - GET /pilotsites/{id}/status (get the current status of the connectivity towards one pilot site)
- Devices:
 - o GET /devices/all/status (get the current status of all devices)
 - GET /devices/{deveui}/status (get the current status of one device)
- Gateways:
 - o GET /gateways/all/status (get the current status of all gateways)
 - GET /gateways/{gatewayeui}/status (get the current status of one gateway)
- Experiments:
 - GET /experiments/all (get all experiments data)
 - o GET /experiments/current (get the current experiment)
 - o GET /experiments/{experiment} (get one experiment data from this tenant)
 - GET /experiments/{experiment} (get the slices and traffic patterns of all devices for a given experiment)
 - GET /experiments/{experiment}/devices/{deveui} (get the slices and traffic patterns of one device for a given experiment)
 - POST /experiments/{experiment}/devices/{deveui}/slices/{slice_id} (set the slice for one device during a given experiment, and set channel to periodic (0))



- POST /experiments/{experiment}/devices/{deveui}/channel/{channel} (set the channel for one device during a given experiment (0 to periodically hop between all channels))
- POST /experiments/{experiment}/devices/{deveui}/trafficpatterns/{pattern_id} (set the traffic pattern of one device for a given experiment)
- Traffic patterns:
 - GET trafficpatterns/all (get all traffic patterns data from this tenant)
 - GET /trafficpatterns/{pattern_id} (get one traffic pattern data from this tenant)
 - POST /trafficpatterns/{pattern_id} (create or modify one traffic pattern for this tenant)
 - DELETE /trafficpatterns/{pattern_id} (delete one traffic pattern from this tenant)
- Slices:
 - GET /slices/all (get all slices data from this tenant)
 - GET /slices/{slice_id} (get one slice data from this tenant)
 - POST /slices/{slice_id} (create or modify one slice for this tenant)
 - DELETE /slice_id} (delete one slice from this tenant)

Figure 21 presents these methods in the web UI.



REST API for the 6G-LoRaGRAN project and asso

	Authorize
Users	^
POST /token Get Authentication Token	
GET /users/me/ Get User Details	• ∽
Pilot sites	^
CET /pilotsites/all/status Get The Current Status Of The Connectivity Towards All Pilot Sites	- →
CET /pilotsites/{id}/status Get The Current Status Of The Connectivity Towards One Pilotsite	≜ ∨
Devices	^
GET /devices/all/status Get The Current Status Of All Devices	
GET /devices/{deveui}/status Get The Current Status Of One Device	• ∽
Gateways	^
CET /gateways/all/status Get The Current Status Of All Gateways	≜ ∨
CET /gateways/{gatewayeui}/status Get Current Status Of One Gateway	- →
Experiments	^
CET /experiments/all Get All Experiments Data	- €
GET /experiments/current Get Current Experiment	≜ ∨
GET /experiments/{experiment} Get One Experiment Data From This Tenant	≜ ∨
GET /experiments/{experiment}/devices/all Get The Silces And Traffic Patterns Of All Devices For A Given Experiment	
CET /experiments/{experiment}/devices/{deveui} Get The Slice And Traffic Pattern Of One Device For A Given Experiment	
POST /experiments/{experiment}/devices/{deveui}/slices/{slice_id} Set The Slice For One Device During A Given Experiment	
POST /experiments/{experiments/devices/{deveui}/channel} Set The Channel For One Device During A Given Experiment (0 To Periodically Hop Between All Channels)	
POST /experiments/{experiment}/devices/{deveui}/trafficpatterns/{pattern_id} Set The Traffic Pattern For One Device During A Given Experiment	
Traffic patterns	^
GET /trafficpatterns/all Get All Traffic Patterns Data From This Tenant	• ∽
GET /trafficpatterns/{pattern_id} Get One Traffic Patterns Data From This Tenant	
POST /trafficpatterns/{pattern_id} Create Or Modify One Traffic Pattern For This Tenant (Returns The Number Of Rows Updated In The Database)	
DELETE /trafficpatterns/{pattern_id} Delete One Traffic Pattern Data From This Tenant (Pattern 0 Cannot Be Deleted)	- ↓
Slices	^
CET /slices/all Get All Silces Data From This Tenant	• ∽
GET /slices/{slice_id} Get One Slice Data From This Tenant	≜ ∨
POST /slices/{slice_id} Create Or Modify One Slice For This Tenant (Returns The Number Of Rows Updated In The Database)	
DELETE /slice_id} Delete One Slice Data From This Tenant (Slice 0 Cannot Be Deleted)	• ∽

Figure 21. REST API methods for 6G-LoRaGRAN.

As examples, Figure 22 shows the status of the connectivity towards the supported pilot sites (direct connection, Berlin and Malaga). As shown, the web UI includes the HTTP command (using the *curl* tool) to receive this information from, for example, a Python program. Similarly, Figure 23 shows the data (pilot site, start and stop times) of all the experiments (anonymized if the tenant is different).



Finally, Figure 24 shows how to modify the channels assigned to a network slice in real-time using the REST API.

Pilot si	tes	^
GET	/pilotsites/all/status Get The Current Status Of The Connectivity Towards All Pilot Sites	۵ ۸
Parameter	rs	Cancel
No parame	zters	
	Execute	Clear
Response	s	
Curl curl -X ' 'https: -H 'acc -H 'Aut	GET' \ //loragram.ugr.es:9443/pilotsites/all/status' \ ept: application/json' \ horization: Bearer eyJhbGci0iJIUzIINIIsInR5cCIGIkpXVCJ9.eyJzdWIi0iJqb3JnZW5hdmFycm8iLCJleHAi0;	iE3MDhaMThaMzh9.kg4_vm0dUP8ER00-xQUec1cR8VEPWY6_ccVr5NK9sHQ*
Request UR	ll	
Server resp	onse	
Code	Details	
200	Response body	
	<pre>{ "Filot sites status": [</pre>	B Download
	Response headers connection: Keep-Alive content-ingth: 211 content-type: application/json date: lue_2A feb 2024 AB:38:00 GMT keep-alive: imeeut-5_max=99 server: unicon	
Responses		
Code	Description	Links
200	Successful Response	No links

Figure 22. REST API – GET /pilotsites/all/status.



Experin	nents	^
GET	/experiments/all Get All Experiments Data	â ^
Parameters	5	Cancel
No paramet	ers	
	Execute	Clear
Responses		
curl -X 'G	ET' \ //orancan.unr.es:9443/avneriments/all' \	
-H 'acce -H 'Auth	yste application/json`4 pt: application/json`4 orization: Bearer eyJhbGciOiJIUzIINiIsInR5cCI6IKpXVCJ9.eyJzdWIiOiJqb3JnZW5hdmFycm8iLCJ1eHAiO	jE3MDkwNTkwHzh9.kg4_vnDdUP8ER09-xQUec1cR8VCPNY6_ccVr5NKPsHQ'
Request URL	- oragran.ugr.es:9443/experiments/all	
Server respo	nse	
Code	Details	
200	Response body	
	<pre>"plot_site" ************************************</pre>	Bownioad ,
0	connection: Keep-Alive content-length: 639 content-type: application/json date: Tue,20 Feb 2024 18:38:21 Keep-Alive: Linecout-5,max-100 server: wvicon	
Responses Code	Description	Links
200	Successful Response	No links





Slices			^
GET	/slices/all Get All Slices Data From This Tenant		≜ ∨
GET	/slices/{slice_id} Get One Slice Data From This Tenant		≜ ∨
POST	/slices/{slice_id} Create Or Modify One Slice For This Tenant (Returns The Number Of Row	s Updated In The Database)	<u>ه</u> ۸
Parameters		Cancel	Reset
Name	 Description		
<pre>slice_id * integer (path)</pre>	required 3		
Request bo	ty	applicat	tion/json v
{ } "channe	T 7		ħ
	Execute	Clear]
_			
Responses			
Curl curl -X 'P 'https:// -H 'accb -H 'Auch -H 'Cont -d '{ "channel }' Request URI https://Lu Server respondent	STT \ loragram.ugr.es:9443/slices/3' \ t: application/json \ mitrican: Bearer eyshbacioiIIUzIINIIsInR5cC16IkpXVCJ9.ey3zdkIi0iJqb33nZd6hdmfycs8ilC3leHAi nt-Type: application/json' \ mask*: 7 ragram.ugr.es:9443/slices/3 150	3jE3MDKwMThwHzh9~kg4_vn0dUP8ER00-xQUec1cR8VCPWY6_ccVr5NKPsHQ* \	8
200	Details		
200	Response body ("Updated rows": 1)		Download
	Response headers connection: Keep-Alive content-length: 18 content-lyne: application/json date: imc_jb teb 2024 18:33:53 GWT keep-blve: LinexUe-5,axx=100 server: micron		
Responses	Description		Linke
ooue	econipacii		

Figure 24. REST API – POST /slices/{slice_id}.

4 CONCLUSIONS

This document offers an overview of the infrastructure and functionalities integrated into the 6G-SANDBOX project's portfolio through collaboration with the 6G-LoRaGRAN project. Furthermore, it includes a comprehensive user manual for researchers interested in utilizing the 6G-LoRaGRAN platform.

REFERENCES

- [1] Public web page for the 6G-LoRaGRAN project, WiMuNet team. Available online: <u>https://wimunet.ugr.es/projects/6gloragran.php</u> (accessed on 25th February 2024).
- [2] WiMuNet research group, University of Granada. Available online: <u>https://wimunet.ugr.es/</u> (accessed on 25th February 2024).
- [3] Kubernetes, an open-source system for automating deployment, scaling, and management of containerized applications. Available online: <u>https://kubernetes.io/</u> (accessed on 25th February 2024).
- [4]FiPydatasheetv1.0,Pycom.Availableonline:https://docs.pycom.io/datasheets/development/fipy/ (accessed on 25th February 2024).
- [5] TTGO LoRa32, LiLyGo. Available online: <u>https://www.lilygo.cc/products/lora-v1-0-kit</u> (accessed on 25th February 2024).
- [6] Cisco Wireless Gateway for LoRaWAN Data Sheet, CISCO, 2021. Available online: <u>https://www.cisco.com/c/en/us/products/collateral/se/internet-of-things/datasheet-c78-</u> 737307.html (accessed on 25th February 2024).
- [7] Lite Gateway datasheet v1.5, WiMOD, 2017. Available online: <u>https://wireless-solutions.de/products/lora-solutions-by-imst/development-tools/lite-gateway/</u> (accessed on 25th February 2024).
- [8] Pygate datasheet v1.0, Pycom, 2019. Available online: https://docs.pycom.io/datasheets/expansionboards/pygate/ (accessed on 25th February 2024).
- [9] ChirpStack, open-source LoRaWAN[®] Network Server. Available online: <u>https://www.chirpstack.io/</u> (accessed on 25th February 2024).
- [10] Eclipse Mosquitto, an open source MQTT broker. Available online: <u>https://mosquitto.org/</u> (accessed on 25th February 2024).
- [11] Eclipse Paho MQTT Python client library. Available online: <u>https://pypi.org/project/paho-mqtt/</u> (accessed on 25th February 2024).