



**Grant Agreement N°:** 101020259

**Topic:** SU-DS02-2020



# ARCADIAN-IoT

Autonomous Trust, Security and Privacy  
Management Framework for IoT

## D5.3: Use cases implementation

<b>Work package</b>	WP 5
<b>Task</b>	Task 5.3
<b>Due date</b>	TBD
<b>Submission date</b>	TBD
<b>Deliverable lead</b>	IPN
<b>Version</b>	1.0
<b>Partner(s) / Author(s)</b>	<p>IPN: Paulo Silva, Sérgio Figueiredo, Vitalina Holubenko, Rúben Leal</p> <p>UWS: Jose M. Alcaraz Calero, Qi Wang, Ignacio Martinez Alpiste, Antonio Matencio Escolar, Ignacio Sanchez Navarro, Julio Diez Tomillo, Pablo Benlloch Caballero,</p> <p>TRU: João Casal, Tomás Silva, Ivo Vilas Boas</p> <p>XLAB: Tilen Marc, Jan Antić</p> <p>UC: Bruno Sousa</p> <p>MARTEL: Giacomo Inches</p> <p>BOX2M: Alexandru Gliga, Ovidiu Diaconescu, Marian Macoveanu</p> <p>RISE: Alfonso Iacovazzi, Han Wang</p> <p>ATOS: Ross Little</p> <p>RGB: Ricardo Ruiz</p> <p>LOAD: Pedro Colarejo</p>

## Abstract

This public report constitutes the deliverable D5.3 of ARCADIAN-IoT, a Horizon 2020 project with the **grant agreement number 101020259**, under the topic **SU-DS02-2020**. The main purpose of the report is to document the implementation activities of the use cases.

The implementation of the different IoT application domain use cases includes primarily the application-specific artifacts, and their tailoring for integrating functionalities provided by ARCADIAN-IoT **Prototype 1 (P1)**; such process was led by the owners of the respective domains (Drone Guardian Angel, Grid Management Service and Medical IoT), with strong involvement of the technical partners responsible for the ARCADIAN-IoT components, both according to their involvement in the different use cases and the components readiness (i.e. availability for P1 or P2).

This document describes the implementation of the different use cases of each domain at M24. The final implementation of the presented (P1) use cases, as well as the remaining (P2) use cases is due in M30.

### Keywords:

ARCADIAN-IoT, Use cases, Drone Guardian Angel, Grid Management, Medical IoT, Identity, Security, Privacy, Trust, Recovery

## Document Revision History

Version	Description of change	List of contributors
V0.1	Table of Contents	RGB
V0.2	Reorganization. Second table of contents	IPN
V0.3	Inputs from domain owners	LOAD, BOX2M, RGB
V0.4	Use case description revision	Technical partners
V0.5	Formatting, complete revision and remaining sections	RGB
V0.9	Version forwarded to revision	All
V1.0	Final editing and cleaning	RGB, IPN

## Disclaimer

The information, documentation and figures available in this deliverable, are written by the ARCADIAN-IoT (Autonomous Trust, Security and Privacy Management Framework for IoT) – project consortium under EC grant agreement 101020259 and does not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

**Copyright notice:** © 2021 - 2024 ARCADIAN-IoT Consortium

Project co-funded by the European Commission under SU-DS02-2020		
Nature of the deliverable:	REPORT*	
Dissemination Level.		
PU	Public, fully open, e.g., web	√
CI	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to ARCADIAN-IoT project and Commission Services	

\* *R: Document, report (excluding the periodic and final reports)*

*DEM: Demonstrator, pilot, prototype, plan designs*

*DEC: Websites, patents filing, press & media actions, videos, etc.*

*OTHER: Software, technical diagram, etc*

## EXECUTIVE SUMMARY

The ARCADIAN-IoT framework proposes an integrated approach to managing identity, trust, privacy, security, and recovery across IoT devices, people, and services, based on specialized components distributed in vertical and horizontal planes. The project aims to demonstrate and validate the framework benefits in multiple use cases spread across three different IoT application domains: Emergency and vigilance using drones and IoT (Domain A), Secured early monitoring of grid infrastructure (Domain B) and Medical IoT (Domain C).

This Deliverable 5.3 (Use cases implementation) describes the status of implementation of the different use cases at M24. The material presented in this document mainly reflects the activities performed in the different IoT application domains (as part of Tasks 5.2, 5.3 and 5.4). Nevertheless, it builds on and integrates ARCADIAN-IoT functionalities provided in its first pilot (P1), which are both the result of the research outcomes resulting from the ARCADIAN-IoT horizontal and vertical planes (WP3, WP4) and the consequent framework integration activities (T5.1). All technical partners and domain owners have been involved in the iterative process of integrating ARCADIAN-IoT framework.

## TABLE OF CONTENTS

<b>EXECUTIVE SUMMARY.....</b>	<b>6</b>
<b>TABLE OF CONTENTS.....</b>	<b>7</b>
<b>LIST OF FIGURES.....</b>	<b>8</b>
<b>LIST OF TABLES .....</b>	<b>10</b>
<b>ABBREVIATIONS.....</b>	<b>11</b>
<b>1. INTRODUCTION .....</b>	<b>12</b>
1.1 Objectives and assumptions .....	12
1.2 Background .....	12
1.3 Structure the implementation status of each use case .....	12
1.4 Document Structure.....	13
<b>2. DOMAIN A .....</b>	<b>13</b>
2.1 Application domain context .....	13
2.2 Use Case A1 – Person Registration at DGA service.....	14
2.3 Use Case A2 – Person authentication at the DGA service.....	20
2.4 Use Case A3 – Person retrieving and editing personal data .....	23
<b>3. DOMAIN B .....</b>	<b>26</b>
3.1 Application domain context .....	27
3.2 Use Case B1 – New device registration.....	27
3.3 Use Case B2 – GMS IoT device data gathering and transmission process.....	31
<b>4. DOMAIN C .....</b>	<b>34</b>
4.1 Application domain context .....	34
4.2 Use Case C2 – MIoT capturing and sending vital signs and perceived health status.....	35
4.3 Use Case C3 – Personal data processing towards health alarm triggering .....	49
4.4 Use Case C4 – Monitor a patient and update a patient monitoring protocol.....	53
<b>5. CONCLUSIONS .....</b>	<b>59</b>
<b>REFERENCES.....</b>	<b>61</b>

## LIST OF FIGURES

Figure 1 - DGA participant entities.....	13
Figure 2 - Diagram for use case A1 .....	15
Figure 3 - UML Diagram A1 .....	19
Figure 4 - Diagram for use case A2 .....	20
Figure 5 - UML Diagram A2.....	23
Figure 6 - Diagram for use case A3 .....	24
Figure 7 - UML diagram A3 .....	26
Figure 8 - Use case B1 & B2 – configuration & debugging process high-level diagram .....	27
Figure 9 - B1 use case diagram.....	30
Figure 10 - B2 use case diagram.....	33
Figure 11 - MIoT participant entities .....	34
Figure 12 - UML diagram C2 .....	37
Figure 13 - MIoT App Login.....	40
Figure 14 - MIoT App home screen .....	41
Figure 15 - MIoT App profile screen .....	42
Figure 16 - MIoT App "new test" screen .....	43
Figure 17 - MIoT App ECG screen .....	44
Figure 18 - MIoT App NIBP screen.....	45
Figure 19 - MIoT App SpO2 screen.....	46
Figure 20 - Web service login page .....	48
Figure 21 - Web service patient's home page.....	48
Figure 22 - Web service patient's sessions page.....	49
Figure 23 - Web service patient's last session page .....	49
Figure 24 - UML diagram C3 .....	51
Figure 25 - Web service sending session with an alarm .....	52
Figure 26 - Web service show alarm .....	52
Figure 27 - UML diagram C4 .....	56



Figure 28 - Web service doctor's home page.....	57
Figure 29 - Web service doctor's patient's first page.....	57
Figure 30 - Web service doctor's patient's second page (C4 use case) .....	58
Figure 31 - Web service doctor's patient's third page .....	58

## LIST OF TABLES

Table 1 - Applications included in the telemedicine web service.....	47
--	----

## ABBREVIATIONS

3PP	3 <sup>rd</sup> Party Platform
ABE	Attribute Based Encryption
AI	Artificial Intelligence
BLE	Bluetooth Low Energy
CTI	Cyber Threat Intelligence
DGA	Drone Guard Angel
DB	Database
DID	Decentralized Identifiers
eSIM	Embedded Subscriber Identity Module
eUICC	Embedded Universal Integrated Circuit Card
GSMA	Global System for Mobile communications Association
HE	Hardened Encryption
IoT	Internet of Things
IPR	Intellectual Property Rights
OTA	Over-the-Air
RoT	Root of Trust
SIM	Subscriber Identification Module

## 1. INTRODUCTION

This section begins by establishing the objectives and assumptions of this deliverable, including its relationship with ARCADIAN-IoT Prototype 1 - P1). It is followed by a presentation of key background information related to each domain and associated use cases prepared for P1 and ends with the current status and future plans for the implementation of each use case in P1.

### 1.1 Objectives and assumptions

The main objective of this report is to document the implementation of the Use Cases of each of the 3 domains addressed by the project, within the scope described in the first prototype (P1) of the ARCADIAN-IoT framework. P1 provides preliminary ARCADIAN-IoT functionalities and associated embedded components, which are documented in Deliverable D5.1 [1] (ARCADIAN-IoT Framework Integration).

The specific Use Case implementation depends on the availability of use case artifacts (for example, service-specific software or hardware) that are being produced or adapted by the use case owners (as described in Task 5.2, Task 5.3, and Task 5.4).

### 1.2 Background

This deliverable builds, in the first place, on the work developed in T5.2 – T5.4, which includes the adaptation of IoT services for integrating ARCADIAN-IoT functionalities. It is based also on several different inputs from WP5 T5.1, taking into consideration the description of Components/Use Cases within P1 integration results included in the D5.1 deliverable and their availability. From WP5 T5.2 to T5.4, the Use Case Specifications applied are documented in deliverable D5.3, taking into account the subset of interactions available for P1, which lead to the definition of the scenarios for validation and evaluation in the document D5.4 [2] (Use Cases Validation and Legal Compliance).

In this deliverable, the different implementations of P1 are defined in each use case of the different domains, their current implementation status and possible future work within each use case.

This deliverable is intended to be more end user-centric, while D5.4 is more cybersecurity-centric.

### 1.3 Structure the implementation status of each use case

The core of this deliverable relies on providing the implementation status for each use case. For each of the domains, a common structure has been adopted, consisting of the main topics:

- Application domain context: this section provides an overview of the application objectives, its usage context, and associated technical challenges pertaining to security, trust or privacy.
- Per use case analysis: for each use case, a motivation for the importance of the use case in the overall IoT application context is initially provided; then, the detailed use case – with the necessary revisions with respect to D2.2 [1]- is presented. Finally, its implementation status, from the IoT service perspective, is presented, where the status on the implemented application artifacts and their current support (integration with) for ARCADIAN-IoT are focused.

## 1.4 Document Structure

The remainder of this document is presented as follows:

**Section 2** describes the use cases of **Domain A** within P1 and its implementation.

**Section 3** describes the use cases of **Domain B** within P1 and its implementation.

**Section 4** describes the use cases of **Domain C** within P1 and its implementation.

**Section 5** concludes the document with a summary of the main **conclusions** of the results obtained so far and provides an overview of the upcoming activities, which include the integration and general validation of the ARCADIAN-IoT framework.

## 2. DOMAIN A – EMERGENCY AND VIGILANCE USING DRONES AND IOT

### 2.1 Application domain context

Ensuring security and safety of citizens in urban environments is a complex subject that depends on the availability of considerable resources, with high costs, and, in many cases, the use and manipulation of sensitive data (e.g., when using street vigilance cameras communicating with centralized data centers). ARCADIAN-IoT domain A focuses on the use of IoT devices, in this case, drones, in novel efficient and citizen-centred urban vigilance services.

Illustrating a potential story of this IoT solution, high-level scenarios can feature a young lady, Ana, who is on her way home alone, after a dinner with friends. Using ARCADIAN-IoT Drone Guard Angel (DGA) app in her personal smartphone device, Ana requests vigilance services to escort her home. The service is available in her city and, to be registered and recognized by a DGA, Ana has supplied some personal data in the registration phase, like name, address and photos. When requesting the service, she needs to provide her initial and final location, to ensure that the service is available in both spots.



Figure 1 - DGA participant entities

After receiving the service request with Ana's data (e.g., location and identification), a drone parked in a specific place in the neighbourhood lifts off and arrives near her. The first thing it does is to validate the user through multiple criteria, which includes the recognition of Ana's smartphone and her physical characteristics (e.g., face recognition). After the successful identification, the

drone notifies her that it is ready to guard her home.

Ana starts walking home and the DGA is following her, aware of the surroundings for detecting any threat signal (e.g., rapid movements towards Ana, high speed vehicles or objects). If something abnormal is detected (e.g., an attempt of robbery), the drone can start an appropriate manoeuvre to scare or demotivate the robbers (blinking lights, emitting sounds, etc.), while it calls for rescue (police). If injuries are detected, a medical rescue team is also called. While the rescue team(s) is/are on its way, some details can already be sent, collected by the camera and appropriate sensors (e.g., GPS), to give precise location and provide the incident characteristics (e.g., number of people involved or type of injuries).

### *Trust, security and privacy management challenges*

DGA solution relies on an IoT device to provide its service to persons, who should have a personal device (smartphone) to use the service. Depends as well on the use of persons sensitive data, like location, address, and photos for facial recognition. In this sense, trust, security, and privacy management challenges arise, namely:

- Enable security and trust in the management of drones' and users' identification, ensuring protection to, e.g., impersonation attacks that could endanger the person physical and data security.
- Define trust evaluation models for the DGA devices, services and app, where the end-user is aware of the trustworthiness of the system components.
- Protect the users' and devices sensitive data (authentication credentials, location, course/path, photos) with hardened encryption mechanisms with recovery ability.
- Assess the integrity and detect anomalous behaviour on IoT devices (drones) and related services, which can indicate the presence of known or zero-day vulnerabilities or threats.
- In case of an incident with a drone or a personal device DGA ecosystem (e.g., app services and related data), have autonomous self-recovery mechanisms that allow to recover functionalities and data to pre-defined trust levels with reduced human intervention.
- Enable an automated and privacy-preserving cyber threat intelligence (CTI) approach for IoT threat information generation, sharing, analysis, storage, and consumption.

## **2.2 Use Case A1 – Person Registration at DGA service**

### **2.2.1 Summary**

Before requesting a DGA service, the user must previously be registered at the DGA system. The user must use its smartphone to collect very basic personal data and use the smartphone camera to collect images from its face in face different specific positions (front, front-left, front-right, front-top and front-bottom). The image collecting process is easily handled by the phone, which will automatically recognize the positions, store them and showing it to the user. After the user confirmation, all data is sent to the ARCADIAN-IoT platform.

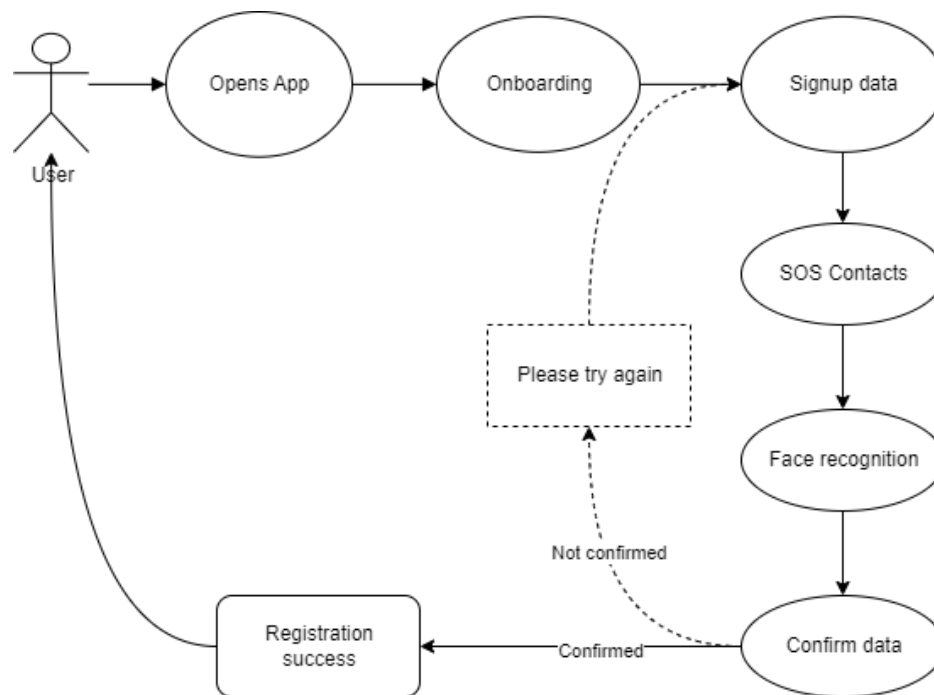


Figure 2 - Diagram for use case A1

## 2.2.2 Description

ARCADIAN-IoT Layers
<b>Vertical plane:</b> Identity; Trust; Recovery. <b>Horizontal plane:</b> Privacy; Security; Common.
Use Case Actors
Citizen / Person to guard.
Use Case Story
<p>The first step for using ARCADIAN-IoT Drone Guard Angel (DGA) is the person registration in the service. To do so, the person, e.g., a regular citizen, uses a <b>mobile app</b> previously downloaded and installed in his/her smartphone. The registration steps are:</p> <ol style="list-style-type: none"> <li>1. The person opens the DGA app and is presented with information from ARCADIAN-IoT framework referring security and privacy procedures included in the <b>service</b>. This means that, for example, the DGA service security <b>behaviour is continuously monitored</b>.</li> <li>2. Willing to proceed, the person is informed that public cryptographic material for encryption of his personal data will be obtained and saved in his/her personal device (<b>Hardened Encryption</b>). Additionally, a key pair (one private and one public) will be generated in the RoT of the mobile device, assuming that the device has an eSIM with an ARCADIAN-IoT profile (standard eSIM download procedure). The private key will be stored in the device RoT, ensuring tamper proof security for that information, while the public key will be sent to ARCADIAN-IoT key management system.</li> <li>3. After, the user proceeds with the registration procedure, filling a form with the personal data needed for the DGA services, part of which is used for generating his/her <b>self-sovereign identity – SSI</b> (e.g., decentralized identifiers - non-centralized credentials that will allow to</li> </ol>

identify and authenticate him/her in the DGA services). Alternatively, the user can present a well-accepted digital credential with the required information to form the base for the SSI. The SSI ensures that no centralized entity will own the full identify of the person, increasing thus its security.

4. The data provided by the person is encrypted with RoT information (the cryptographic material previously generated) and submitted to the DGAs services. The user is informed that an identifier for secure and private identification is going to be generated based on the data provided and that the generated identification credentials are encrypted and backed up in the users' SSI Wallet, allowing **credentials recovery**, in case of need. . Is also informed that has the right to request the deletion of all his data. Is also informed that has the right to request the deletion of all his data. Is also informed that has the right to request the deletion of all his data.

5. After the generation of the SSI, the DGA services request to store the private credentials in a secure ID wallet. The user is informed that the personal device network credentials, already at the device eSIM, will also be used as a **second secure identification/authentication mechanism** in DGA services.

6. To conclude the registration, and for the purpose of having **several simultaneous reliable identification mechanisms**, the DGA services ask the person to capture images to be used for **biometric identification** by authorized DGA drones. Images are encrypted with RoT material and sent to DGA services, where they are kept encrypted.

7. Finally, when the user identity is created, the device authenticates itself to the cryptographic key management service (**hardened encryption**) and obtains private decryption keys, that are saved to the personal device.

8. The person is informed in the app that the registration procedure is finished, and he/she can start using the service.

#### **Relation with ARCADIAN-IoT Objectives**

Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 3: Enable distributed security and trust in management of persons' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

Objective 5: Provide a hardened encryption with recovery ability.

#### **Use Case Priority**

*High: critical to several project objectives and without it some objectives could not be fulfilled*

*Average: Important, but other use cases have the same purpose*

*Low: Nice to have, but not critical for the project objectives*

High.

#### **Use case preconditions**

1. DGA services are registered on ARCADIAN-IoT.

2. User has a smartphone with eSIM and the DGA app installed.

3. Remote Attestation (its Verifier component) has received the Reference Values from the DGA Service Provider, enabling future attestation of the smartphone once the DGA app is installed and configured.



4. (starts with the user acceptance of the terms provided within this use case) ARCADIAN-IoT behaviour monitoring component monitors the interactions of the user, personal device, and third-party service to - in articulation with the CTI component - trigger any security action needed and update the trust knowledge. This may include dynamic reputation and authorization changes for the parties involved.

#### *Use case postconditions*

1. The user is registered in the system, has at least 3 strong identification mechanisms configured, being one of them decentralized, and the RoT on his/her personal device has information for performing hardened encryption/decryption of the private data. The user is able to securely log in and start using the services of the ARCADIAN-IoT third-party, the DGA services, with his/her sensitive data privacy ensured.

2. The ARCADIAN-IoT third-party (DGA services), has the necessary data for providing its service.

#### *ARCADIAN-IoT Entities (Person/ IoT device / Services)*

All.

#### *Data used and data flow*

1. Information for hardened encryption is generated at the RoT of the user personal device. Information for decrypting user data is managed by the HE key management, which securely provides limited access keys to services and persons. The person authorizes the services to access his/her data. Personal data not in use is always kept encrypted.

2. User is requested to download SSI Mobile wallet and provides personal data needed for the SSI (Verifiable Credentials / Decentralized Identifiers) and is issued with a Person Verifiable Credential (VC) with personal identifiable information. The registration flow will now start in the user personal device with the user presenting the Person VC to be authenticated and result in ID Token sent to the DGA service, where it is sent in a request to authorise ARCADIAN-IoT services to create an ARCADIAN-IoT identity with a radio network token added by the network authorizer. Registered ARCADIAN-IoT Services create the ARCADIAN-IoT Identity (aiotID) and associate it with the SSI Person Identity and the Network Identity to be used for future authentications. An event is published on successful registration of the new ARCADIAN-IoT Identity to be consumed by ARCADIAN-IoT services such as Reputation.

3. Biometric material is generated in the user mobile device, encrypted with RoT information and sent to the supporting ARCADIAN-IoT third-party services.

### 2.2.3 Implementation status

For prototype P1, the implementation concerned mainly in the interaction of the Mobile App with the ARCADIAN-IoT platform for registering DGA service users.

- A mobile App was created, including menus and functionalities for registration of the user, collecting login information, as well as minimal profile data (name, address, contacts) essential to the operation of all DGA features.
- The collected data is inserted in a WebView where the interaction with the Decentralised Identifiers and Verifiable Credential components is made. As result of this, a token ID is received by the DGA backend, which will forward it to the DGA App.
- The user is asked to confirm its registration in the AIoT platform.

- After the first step of registering personal credentials is made, the App presents a flow of screens to collect 5 photos of the users' face in 5 different positions and, after user's confirmation, these photos are sent to the Biometrics components, where they are validated as useful images and stored to be used later as authentication factor (described in use-case A2).
- Additionally, the DGA App integrated a Remote Attestation module, via two different implementations:
  - Ability to send Reference Values (e.g. App Signature) to the designated Verifier, via the framework's message bus (RabbitMQ)
  - Integration of Attester package in the DGA App, periodically collecting info about the smartphone operation.
- In each step of this flow, user-friendly and helpful feedback is given to the DGA App's user, not requiring any previous knowledge or training of the user regarding the actions to be done. Once the flow described above is completed and validated, the corresponding feedback is sent to the user.
- A DGA Guard Backend was also implemented to intermediate and optimize some of the actions regarding data circulating between the Mobile App and the ARCADIAN-IoT platform.

The integration of the above-mentioned feature and the ARCADIAN-IoT platform was implemented according to the sequence diagram presented below, describing the actions between use-case and the platform's components:

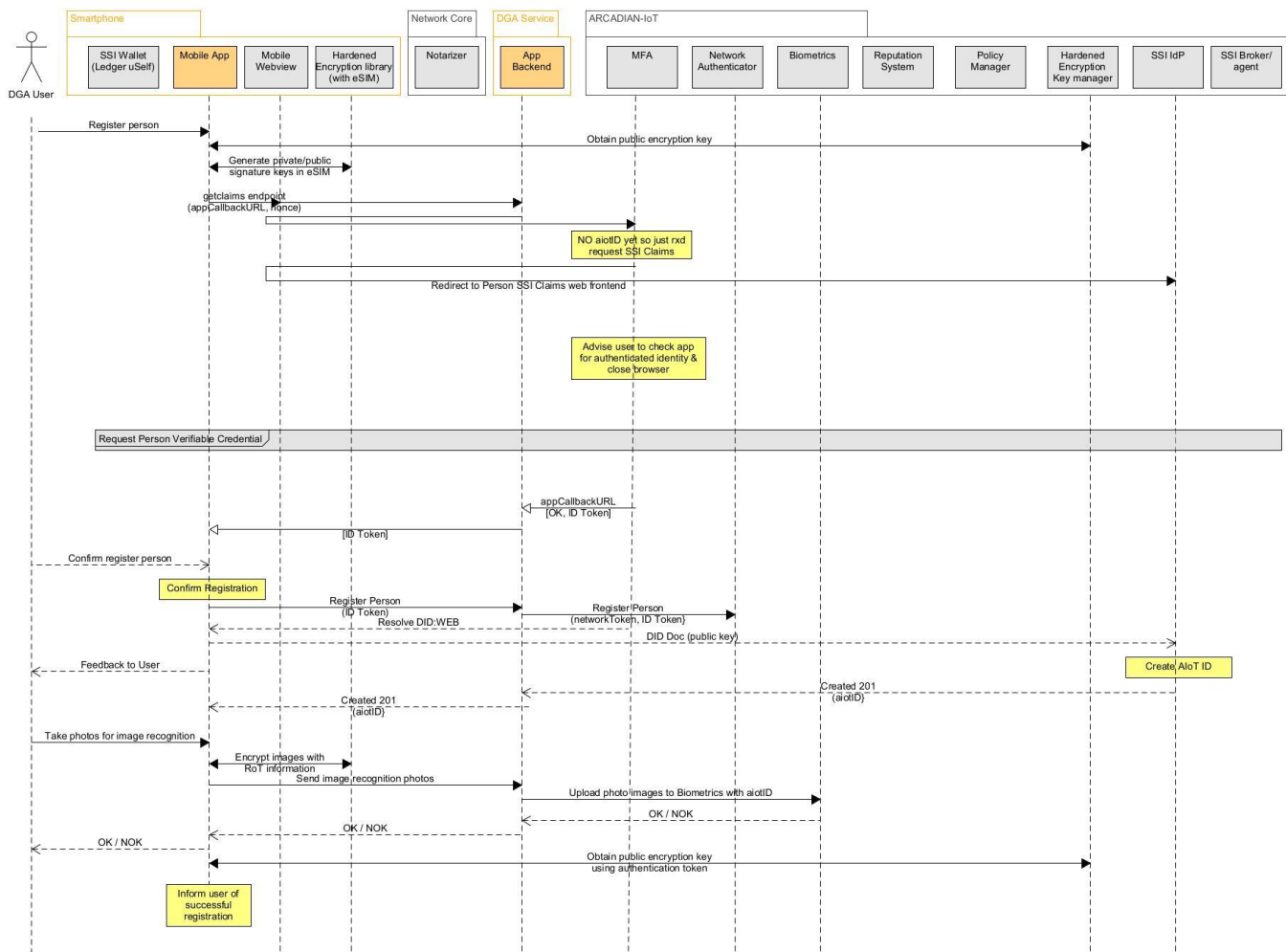


Figure 3 - UML diagram for use case A1

Note on the Diagrams: These are Diagrams with information related to the implementation where some interactions with Arcadian-IoT components have been hidden with a grey box. The domain components have been represented with orange background and the Arcadian-IoT components with grey background. Although components without interactions appear, they are components involved in the use case whose interactions have been hidden because they correspond to the integration and not to the implementation.

## 2.2.4 Future plans

The future actions for this use case focus the integration with components only planned for P2, such as Behaviour Monitoring or Hardened Encryption (via eSIM). Regarding the DGA Guard Mobile App the internal process is practically concluded, being special focus given to smoothly integrate the remaining cybersecurity ARCADIAN-IoT functionalities running in the smartphone, i.e., Behaviour monitoring, and Hardened Encryption, considering e.g., package integration efficiency or overhead aspects, as well as eventual improvements to the Multi-Factor Authentication, planned for P2.

## 2.3 Use Case A2 – Person authentication at the DGA service

### 2.3.1 Summary

When a user opens the Mobile App and intends to perform any action, and once the authentication token is expired, the authentication flow will be called, and this use-case will be invoked. It consists of a multi-factor authentication approach, consisting of biometric data, network identifier and SSI.

After the authentication process is done, the user will receive feedback of the success or unsuccess of the action.

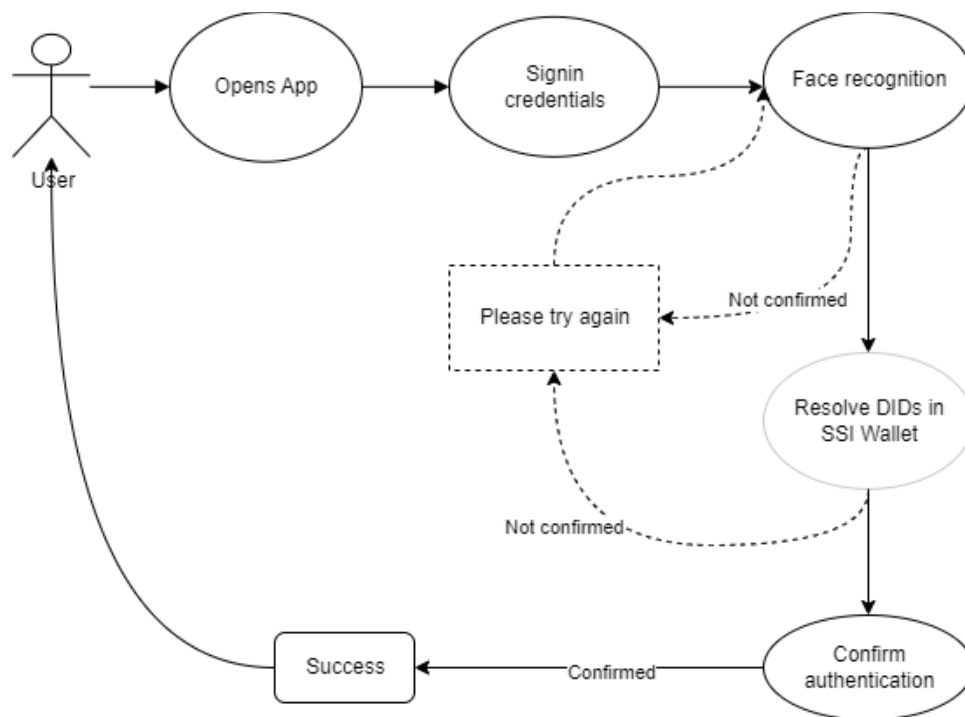


Figure 4 - Diagram for use case A2

*Description*

ARCADIAN-IoT Layers
<b>Vertical plane:</b> Identity; Trust. <b>Horizontal plane:</b> Security; Common.
Use Case Actors
Citizen / Person to guard.
Use Case Story
When registered, the user needs to authenticate himself/herself with at least <b>three robust identity mechanisms</b> to access ARCADIAN-IoT DGA services. The steps are: <ol style="list-style-type: none"> <li>1. The user opens the DGA app and selects the login form.</li> <li>2. The app informs that it will use the user SSI (<b>verifiable credentials</b>), <b>network credentials</b> (in the smartphone SIM/eSIM) and <b>biometrics</b> for authentication. To proceed, the person</li> </ol>

needs to have an SSI wallet installed and take a photo.

3. If the user agrees, the **three identifiers** mentioned are used to login the user in the DGA services. In this process, the user is requested to confirm its identity in its SSI wallet.

4. If the three authentication factors are verified, the login process is successful and the DGA app is securely informed that the user is allowed to proceed.

5. If not, the process is not successful and an error message is returned.

#### **Relation with ARCADIAN-IoT Objectives**

Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 3: Enable distributed security and trust in management of persons' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

#### **Use Case Priority**

*High: critical to several project objectives and without it some objectives could not be fulfilled*

*Average: Important, but other use cases have the same purpose*

*Low: Nice to have, but not critical for the project objectives*

High.

#### **Use case preconditions**

1. Use case A1.

2. ARCADIAN-IoT behaviour monitoring component monitors the behaviour of the device where the DGA is installed in order to trigger any security actions that are deemed necessary and update the trust knowledge. This may include system-level information (e.g., system calls), dynamic reputation and authorization changes for the device involved.

3. ARCADIAN-IoT Cyber Threat Intelligence (CTI) is running and receives threat alerts issued by the behaviour monitoring component.

#### **Use case postconditions**

1. The user is logged in in ARCADIAN-IoT DGA and may request a service.

#### **ARCADIAN-IoT Entities (Person/ IoT device / Services)**

All.

#### **Data used and data flow**

1. The DGA app requests the user to take a photo of his/her face and securely (e.g., using the **Hardened Encryption**) sends it attached to the authentication request.
2. Network identifiers are used according to the GSM standards. The novelty is that a protected network ID token is generated in the network core and attached to the authentication request.
3. Both the photo and the network ID token are verified to confirm if they are the expected identifiers for that person.
4. The SSI Person Verifiable Credential is presented from the user's SSI Wallet in the smartphone, and verified by the SSI Broker/Agent in the ARCADIAN-IoT platform with

- the obtained claims made available for the ID token.
5. A protected and signed ARCADIAN-IoT ID token is returned to the device for its authenticated operation.

### 2.3.2 Implementation status

For prototype P1, the implementation concerned mainly in the interaction of the Mobile App with the ARCADIAN-IoT platform when dealing with the authentication of a DGA Guard user.

An authentication module was added to the DGA Guard mobile App, which includes:

- The developed mobile App enables the process of user authentication, starting by presenting a login form to the user.
- After this first step, a screen using the smartphone camera is presented to capture the user's face in a front position.
- The data collected and the photo are sent to ARCADIAN-IoT Multi-Factor Authentication component, which will handle all the validation actions. The **Multi-Factor Authentication** component shares the person authentication results with other components. The **Reputation System** uses the positive – synonym of successful authentication, and negative results to update reputation accordingly.
- In each step of this flow user-friendly and helpful feedback is given to the App's user, not requiring any previous knowledge or training of the user regarding the actions to be done. Once the flow described above is completed and validated, the necessary feedback is done to the user.
- A DGA Guard Backend was also implemented to intermediate and optimize some of the actions regarding data circulating between the Mobile App and the ARCADIAN-IoT platform.

The integration of the above-mentioned feature and the ARCADIAN-IoT platform was implemented according to the sequence diagram presented below, describing the actions between use case and the platform's components (details on the current MFA implementation are in the public deliverable [3]):

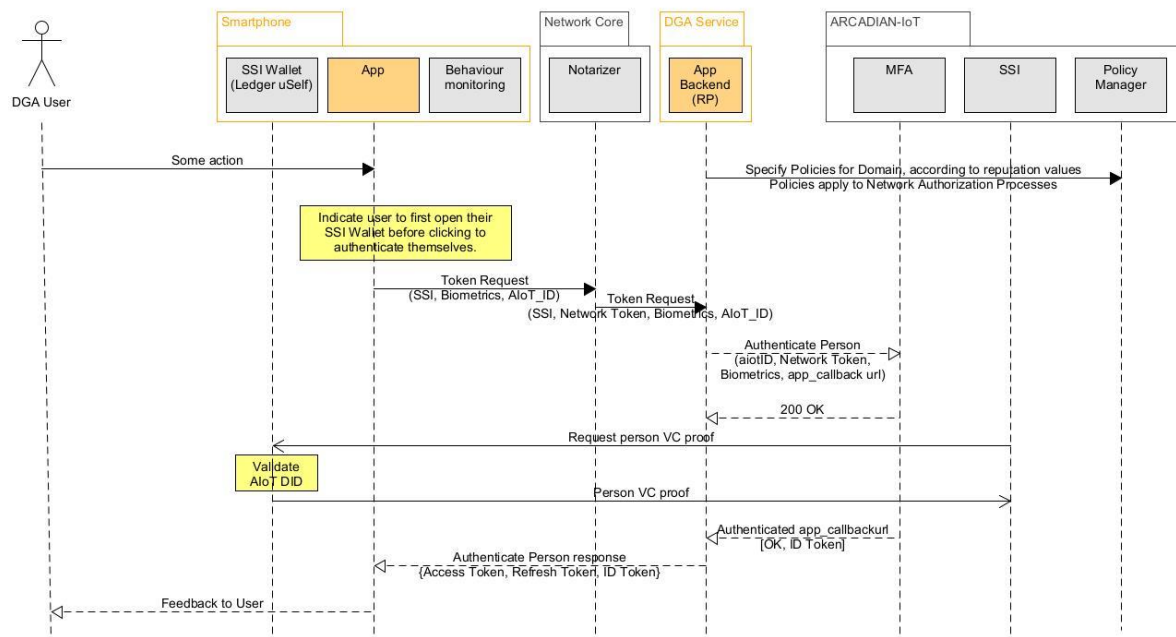


Figure 5 - UML diagram for use case A2

### 2.3.3 Future plans

The future actions for this use case will be focused on the prototype P2 implementation. Regarding the integration of the DGA Guard Mobile App in this particular use-case (A2, regarding authentication), the process is practically concluded, with minor adjustments to the ARCADIAN-IoT MFA suggested by the domain owner to be tackled. For P2, a special focus will be given to the direct interactions with **Behaviour monitoring** and **Hardened Encryption** and **Network-based authorization** enforcement.

## 2.4 Use Case A3 – Person retrieving and editing personal data

### 2.4.1 Summary

When a user opens the Mobile App and intends to check and/or edit its personal data, he need to authenticate first (using use-case A2 flow) and then navigate to the App Profile Area. The user can edit each one of its personal data fields, and then its face image, if wanted. Once the new data is collected it is sent to the DGA Guard backend, as well as to the ARCADIAN-IoT platform. Once the success/no success feedback is received form the backend, it is presented to the user.

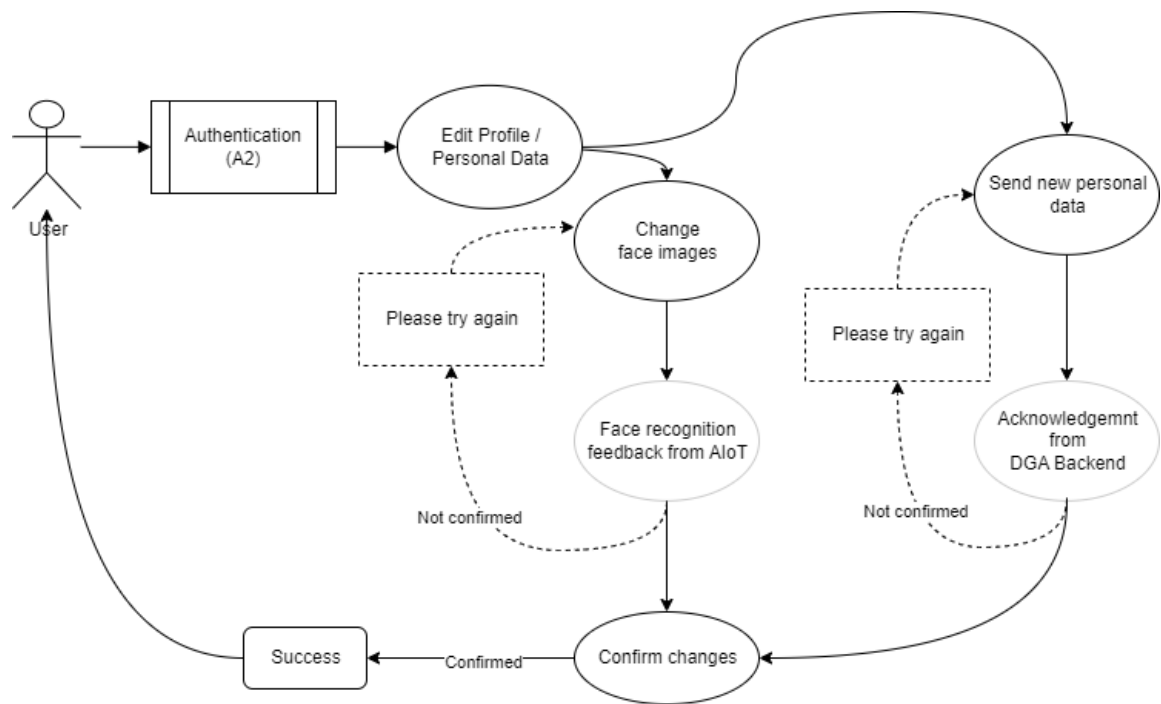


Figure 6 - Diagram for use case A3

## 2.4.2 Description

### ARCADIAN-IoT Layers

**Vertical plane:** Identity; Trust.

**Horizontal plane:** Privacy; Security; Common.

### Use Case Actors

Citizen / Person to guard.

### Use Case Story

The end-user can retrieve and edit his/her personal data (registered in the system) using DGA mobile app. In this case the story is:

1. When logged in the DGA app, the user requests to edit his/her personal data (e.g., name and address).
2. DGA services validate the requesting user, the requesting app and the personal device and assess the **reputation** of user, on which the request is performed. If the entities have the necessary **authorization**, the encrypted data is retrieved to the personal device.
3. The user decrypts it with his private ABE key (**Hardened Encryption**). The data is shown to the user.
4. The user edits the intended fields and requests the sending of the data, encrypted again, to the DGA service. A **Hardened Encryption** process happens using ABE for encryption and RoT signatures. The updated data is kept encrypted and only accessible by the user and third parties registered in ARCADIAN-IoT, authorized by the user.

### Relation with ARCADIAN-IoT Objectives



Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 3: Enable distributed security and trust in management of persons' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

Objective 5: Provide a hardened encryption with recovery ability.

### **Use Case Priority**

*High: critical to several project objectives and without it some objectives could not be fulfilled*

*Average: Important, but other use cases have the same purpose*

*Low: Nice to have, but not critical for the project objectives*

High.

### **Use case preconditions**

1. Use cases A1 and A2.

2. ARCADIAN-IoT behaviour monitoring component monitors the behaviour of the device where the DGA is installed in order to trigger any security actions that are deemed necessary and update the trust knowledge. This may include system-level information (e.g., system calls), dynamic reputation and authorization changes for the device involved.

3. The Remote Attestation (Verifier) must have reference values and evidence appraisal policies, against which to analyse the evidence received from the device / smartphone to attest. These can be retrieved either from manufacturers or IoT service providers when the IoT service is registered in ARCADIAN-IoT.

4. ARCADIAN-IoT Cyber Threat Intelligence (CTI) is running and ready to receive threat alerts issued by the behaviour monitoring component.

### **Use case postconditions**

1. Updated personal data stored, encrypted, in DGA services.

### **ARCADIAN-IoT Entities (Person/ IoT device / Services)**

All.

### **Data used and data flow**

1. If the requesting entities are trustable and authorized, personal data (e.g., name and address) is retrieved, encrypted, from DGA services to the requesting mobile device. It is decrypted with private cryptographic material. After editing the data is encrypted again and sent to DGA service. It is not stored decrypted anywhere (not at the device nor at the Cloud).

## **2.4.3 Implementation Status**

For prototype P1, the implementation concerned mainly in the interaction of the Mobile App with the ARCADIAN-IoT platform when dealing with the authentication of a DGA Guard user.

A profile management module was added to the DGA Guard mobile App, which includes:

- The user opens the App with the intention of changing its personal data and/or face image.
- After the authentication procedure (as described in use-case A2), the user can access the profile screen, when he/she can edit its personal data
- The user will also be able to change its face image, by going through the image capturing procedure, as he did when he registered the first time
- The data collected is sent to the DGA Guard backend and the photo set is then sent to the Biometrics component to update its records.
- DGA App integrated a Remote Attestation module in the DGA App, periodically collecting info about the smartphone operation.
- In each step of this flow a user-friendly and helpful feedback is given to the App's user, not requiring any previous knowledge or training of the user regarding the actions to be done. Once the flow described above is completed and validated, the necessary feedback is done to the user.

The integration of the above-mentioned feature and the ARCADIAN-IoT platform was implemented according to the sequence diagram presented below, describing the actions between use-case and the platform's components:

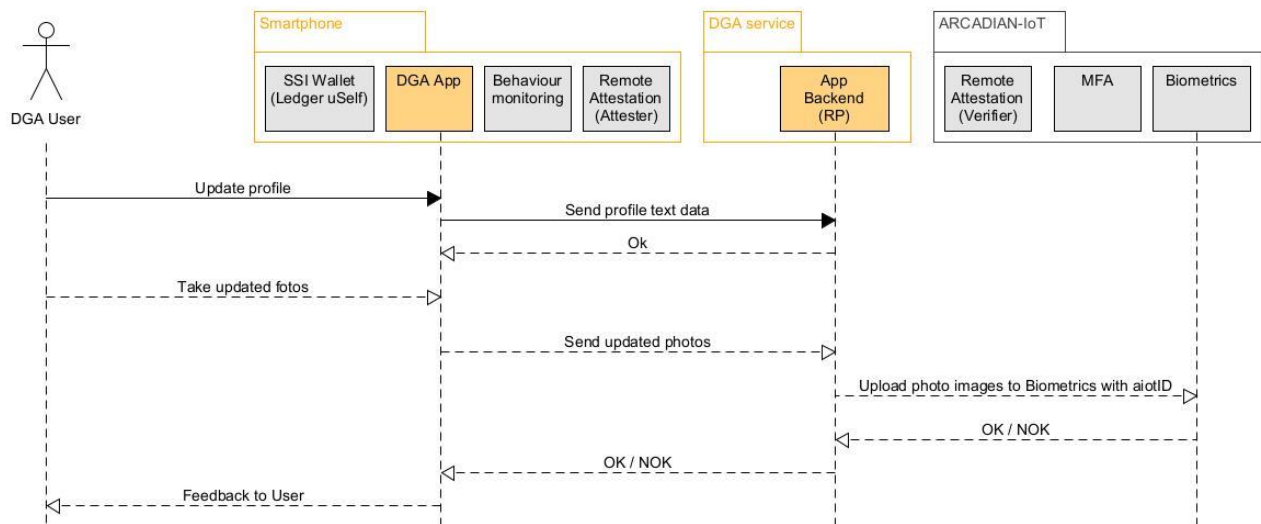


Figure 7 - UML diagram for use case A3

## 2.4.4 Future plans

The future actions for this use case will be focused on the prototype P2 implementation. Regarding the DGA Guard Mobile App, special focus will be done to the direct interactions with **Reputation System** and **Hardened Encryption**.

## 3. DOMAIN B – SECURED EARLY MONITORING OF GRID INFRASTRUCTURES

Grid domain is generically defined by critical infrastructures covering electrical energy, utilities and correlated environment. Monitoring of infrastructure elements (based on their role into architecture, typology or displacement into field) is either done by SCADA systems, or by local / isolated technologies or not at all. Industrial IoT technologies, operating with microcontroller

powered devices and cloud platforms is a good mitigation to monitor the unmonitored elements and retrofit the isolated / not modernized monitored elements. Due to sensitivity of data, it is highly demanded a cybersecurity technology matching both a solid state of data integrity, lack of exposure and limited computing capacity of MCU powered devices. So, it was designed a security system, hardware encryption base, which carry end-to-end data between sensors and managing IoT platform.

### 3.1 Application domain context

Use cases implementation will consist of provisioning phase, validation of provisioning details (using device CLI tool, Middleware SoC (Service Operations Center) tool), running live device and final integrated services validation (Encryption & Decryption, Behaviour Monitoring).

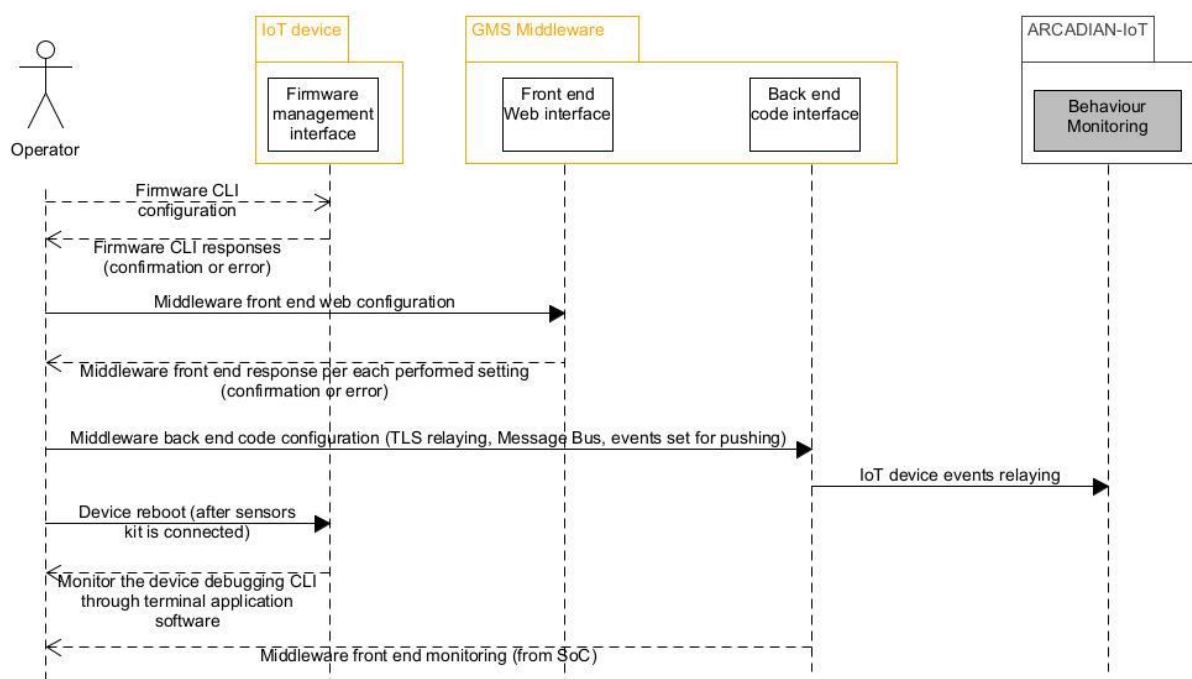


Figure 8 – Simplified UML diagram for use cases B1 and B2

### 3.2 Use Case B1 – New device registration

#### 3.2.1 Description

Into this use case, IoT device and middleware get provisioned for lifecycle operations. It is also provisioned accordingly the Behaviour Monitoring System for middleware docker status monitoring and devices monitoring. Once provisioned, when it is occurring a reauthentication (due to context), device and middleware will use the information already fulfilled. So, this use case treats both initial registration (when the system integrator onboards the Hardened Encryption System to customer) and regular registrations (after devices were onboarded).

**Vertical plane:** Identity; Trust; Recovery.

**Horizontal plane:** Privacy; Security; Common.

#### *Use Case Actors*

Grid infrastructure manager.

#### *Use Case Story*

This use case depicts the scenario of the service supplier configuring and registering a new IoT monitoring device that is supported by ARCADIAN-IoT<sup>1</sup> to gather and propagate information from sensors and actuators of a grid infrastructure, with security and privacy, to a monitoring tool and an IoT platform managing sensors data. The story steps are:

1. In the grid monitoring device assembling, besides the firmware adapted to each grid infrastructure needs, each device is setup with a **crypto chip** and an eUICC for receiving **eSIM** profiles. In this ARCADIAN-IoT domain, the crypto chip will be the device RoT that will have personalized cryptographic information for the **hardened encryption** of the private data collected in the grid infrastructure (generated at the crypto chip). In this case, the eSIM component will be used to provide cellular connectivity and to allow a network-based **authentication** of the device in the ARCADIAN-IoT GMS (extending current SoA, where network credentials are just used to authenticate devices in networks). The communication through the cellular network is also a key element of the domain because at the network core will live the ARCADIAN-IoT **authorization** enforcement component. This component will ensure that compromised devices don't communicate with services besides the ones for **recovery** from security or privacy incidents.

3. After the device robust identifiers are generated and provisioned to the device, and the device being ready for performing the hardened encryption (having the cryptographic material to encrypt the sensitive data and its firmware programmed to do so), it is registered and authenticates in the GMS middleware services. These services bootstrap the device processes of **Remote Attestation, Behaviour Monitoring** and **Reputation System** in ARCADIAN-IoT framework.

4. Through a reliable web interface, the GMS middleware informs the infrastructure manager<sup>2</sup> of the new IoT device registered for gathering and transmitting the data generated in their grid infrastructure. At this moment, the grid manager also selects (and authorizes) the device data to be forwarded to one or more specific third-party telemetry monitoring (such as OrchestraCities of Martel, standalone container of **self-aware data privacy**). The infrastructure owner will also be informed that ARCADIAN-IoT components will monitor the device and the related third parties' behaviour to ensure its data security and privacy.

5. If the infrastructure owner accepts the connection of the device to the IoT platform managing the sensors data and to ARCADIAN-IoT framework, the GMS setup is ready to securely connect grid infrastructure sensors and actuators to one or more web/mobile monitoring tools and IoT platform managing the sensors data (through the GMS IoT device and middleware). The monitoring tools themselves need to be registered at ARCADIAN-IoT<sup>3</sup> to be authorized to receive the cryptographic material that decrypts the private grid data sent.

<sup>1</sup> Definition of IoT devices compliant with ARCADIAN-IoT in D2.3

<sup>2</sup> For security purposes the grid infrastructure manager needs to comply with ARCADIAN-IoT identification procedures, having more than one robust identity mechanisms to access the GMS information.

<sup>3</sup> Definition of the requirements for a third-party to be compliant with ARCADIAN-IoT at D2.3

### ***Relation with ARCADIAN-IoT Objectives***

Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 3: Enable distributed security and trust in management of persons' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

Objective 5: Provide a hardened encryption with recovery ability.

### ***Use Case Priority***

*High: critical to several project objectives and without it some objectives could not be fulfilled*

*Average: Important, but other use cases have the same purpose*

*Low: Nice to have, but not critical for the project objectives*

High.

### ***Use case preconditions***

1. Operational needs and characteristics of the grid infrastructure considered in the manufacture of the IoT monitoring device (e.g., sensors and actuators secure communication with the IoT monitoring device).
2. To ensure the system security and have control over its privacy, the infrastructure owner needs to be registered in GMS services with identification and authentication mechanisms compatible with ARCADIAN-IoT.
3. (starts with the manager acceptance within this use case) ARCADIAN-IoT behaviour monitoring component oversees the interactions of the device and third-party monitoring service to trigger any security action needed and update the trust knowledge. This may include dynamic reputation and authorization changes for the parties involved.

### ***Use case postconditions***

1. GMS IoT device is supported by ARCADIAN-IoT, having a robust identity process set up, as well as the information for hardened encryption stored in the RoT.
2. Device is registered and able for being recognized by the respecting telemetry / grid infrastructure monitoring platform and start communicating data.
3. ARCADIAN-IoT Behaviour Monitoring component is monitoring the device behaviour in order to trigger any necessary security actions, and updating the related trust knowledge (which may include reputation and authorization changes).
4. The infrastructure owner is aware of where his data will start being sent, authorizing specific third-party monitoring tools.

### ***Entities/Scope (Person/IoT/Apps Services)***

IoT / Apps Services

### ***Data used and data flow***

1. The network-based identifier is provisioned to the device using the eSIM GSMA-SAS

standards and stored at the eUICC, which is a secure element.

2. The cryptographic material for hardened encryption is generated at the crypto chip.

3. After, if the infrastructure manager authorizes, the device authenticates in the ARCADIAN-IoT GMS services (with both authentication factors), its behaviour starts being monitored (behaviour monitoring component), its reputation and attestation are bootstrapped, and it is ready to start forwarding grid infrastructure data, encrypted to one or more monitoring tools and IoT sensors data management platform, which needs to be registered in the ARCADIAN-IoT platform.

### 3.2.2 Implementation Status

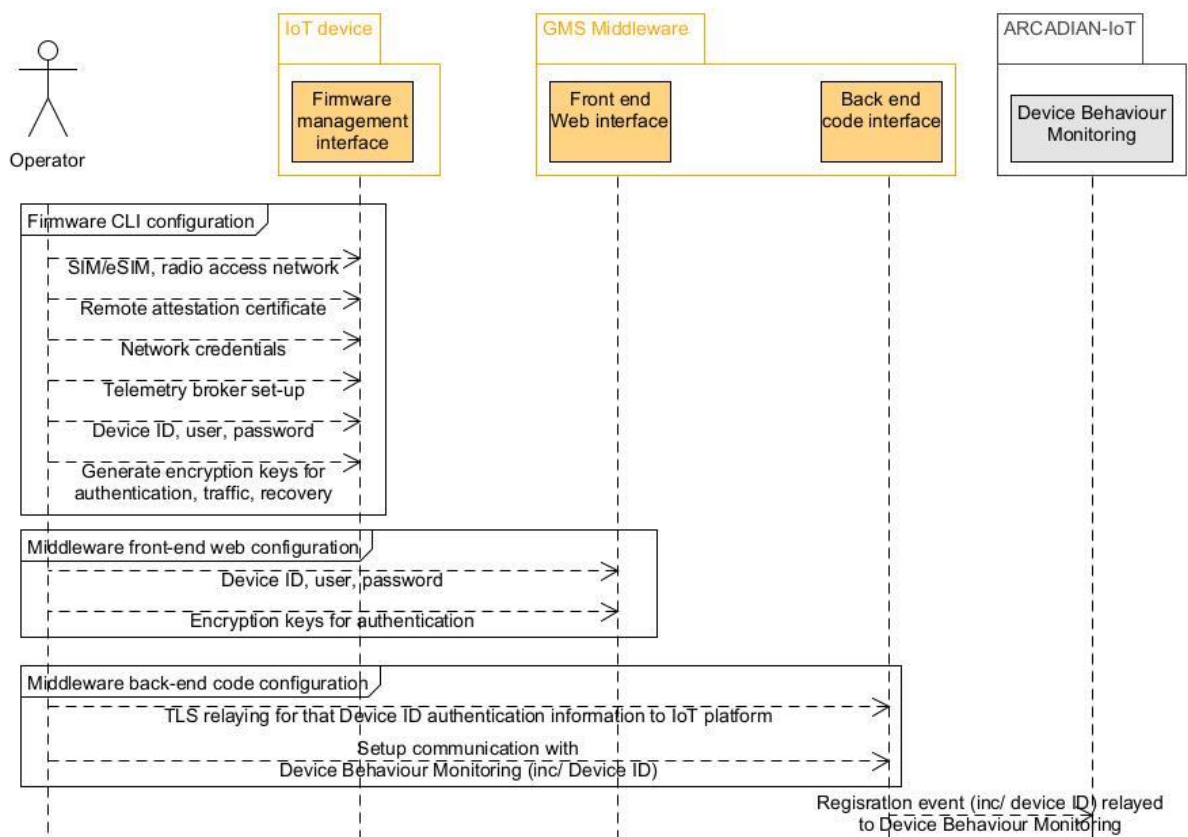


Figure 9 – UML diagram for use case B1

The approach for implementing the use case considered the following steps:

- prepare settings information
- connect to device by proper tool & method
- login into firmware CLI
- start provisioning of firmware settings
- start debugging function of CLI
- login to middleware front end
- start provisioning of middleware
- login to middleware back end
- perform changes into software code



- login to Behaviour Monitoring System
- start provisioning of Behaviour Monitoring System
- reboot the device
- monitor on CLI debugging, Middleware SoC (service operation centre) and Behaviour Monitoring System front end the authentication and traffic from IoT devices.

If there are intentionally applied (wrong) settings changes, these must be observed in the corresponding tool. Trial & error of authentication encryption fake keys, Device fake IDs and Device fake credentials is a main test of this use case. Device status changes events and middleware docker status changes are main tests for Behaviour Monitoring System reactivity.

### 3.2.3 Future plans

As future plans for B1 – device registration, there are couple of options towards strengthening the GMS service: IoT device extra authentication or authorisation performed with **Authentication** component, using the IMSI provisioned on SIM or eSIM or IMEI of the modem module part of IoT device; **Remote attestation** of IoT device, considering also the impact on device's reputation (i.e. integrating **Reputation System**). Another option, slightly more sophisticated, could involve the **Device Behaviour Monitoring**, which together with **Reputation System**, could deny authentication of a Device ID in case of an abnormal authentication failure patterns (i.e. considering number of attempts and spanned period).

## 3.3 Use Case B2 – GMS IoT device data gathering and transmission process

### 3.3.1 Description

In this use case, IoT device transmits sensors data, named also “data traffic”, in a secured way to a an authorized IoT platform. The Middleware (a cloud software component) must be capable of relaying the received encrypted traffic from IoT device to any IoT platform authorized to use sensors data. Relaying of traffic will take place after decrypting data with a correspondent key generated by the crypto chip (previously provisioned) and encrypting data by TLS before transmitting forward to IoT platform.

<i>ARCADIAN-IoT Layers</i>
<b>Vertical plane:</b> Identity; Trust. <b>Horizontal plane:</b> Privacy; Security; Common.
<i>Use Case Actors</i>
Grid infrastructure manager.
<i>Use Case Story</i>
<p>At this stage, the GMS IoT device, already registered and authenticated in GMS services, starts to run its local sensors reading cycle and transmitting the payload to the monitoring service. The main steps are:</p> <ol style="list-style-type: none"> <li>1. After getting and aggregating the data from the grid infrastructure sensors the GMS IoT device performs the <b>hardened encryption</b> of the payload with Root of Trust – crypto chip – information, which is the encryption key provisioned for device traffic.</li> </ol>

2. ARCADIAN-IoT **Behaviour monitoring**, oversees and interprets the IoT device behaviour through GMS middleware, adjusting the device security and privacy **reputation** and its **authorization** to access or be accessed by online services when needed. 3. GMS middleware **decrypt** the device traffic with corresponding provisioned key, and relay it **encrypted** by TLS to the IoT platform used for sensors data management.

3. According to the GMS monitoring rules and the infrastructure manager authorization, GMS middleware forwards the data to the compliancy monitoring tools (OrchestraCities of Martel, standalone container of **Self-aware data privacy**) by API. These third-party tools need to comply with ARCADIAN-IoT to be able to decrypt the GMS data<sup>4</sup>. TLS encryption will be used for API communication between GMS middleware and compliancy monitoring tool.

4. If, for some reason, the device is turned off, when it is turned on again it performs the authentication in GMS services with the Device ID, Device unique credentials (user, pass) and crypto chip allocated key and optionally with **network-based credentials**. **Once this authentication is done**, the device starts to run its local sensors reading cycle (by edge firmware agent) and encrypt the traffic with another crypto chip allocated key for this stage.

5. A re-**authentication** process (authenticate again according to a condition like the time since last authentication) may be applied for security purposes.

#### *Relation with ARCADIAN-IoT Objectives*

Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

Objective 5: Provide a hardened encryption with recovery ability.

#### *Use Case Priority*

**High:** critical to several project objectives and without it some objectives could not be fulfilled

**Average:** Important, but other use cases have the same purpose

**Low:** Nice to have, but not critical for the project objectives

High.

#### *Use case preconditions*

1. Use case B1.

2. ARCADIAN-IoT behaviour monitoring components oversee the interactions of the IoT devices involved to trigger any security actions that are deemed necessary and update the trust knowledge. This may include dynamic reputation and authorization changes for the parties involved.

#### *Use case postconditions*

1. IoT device is transmitting encrypted data to the GMS middleware, IoT platform managing sensors data and authorized IoT monitoring tools.

<sup>4</sup> Definition of compliant ARCADIAN-IoT third-party services available in D2.3



### Entities/Scope (Person/IoT/Apps Services)

IoT / Apps Services.

### Data used and data flow

1. The device identifiers and authentication material, cryptographic material for hardened encryption and TLS encryption and the data gathered from the grid infrastructure (payload) are used.
2. The payload gathered from the grid sensors and aggregated for communication, and the related timestamp is encrypted with RoT information and sent to the GMS middleware.
3. New authentication processes using the device identifiers, which are stored in the device secure element, are triggered if needed, according to security policies.
4. Communication events are captured by ARCADIAN-IoT framework (e.g., behaviour monitoring) to infer potential threats and act if needed.

## 3.3.2 Implementation Status

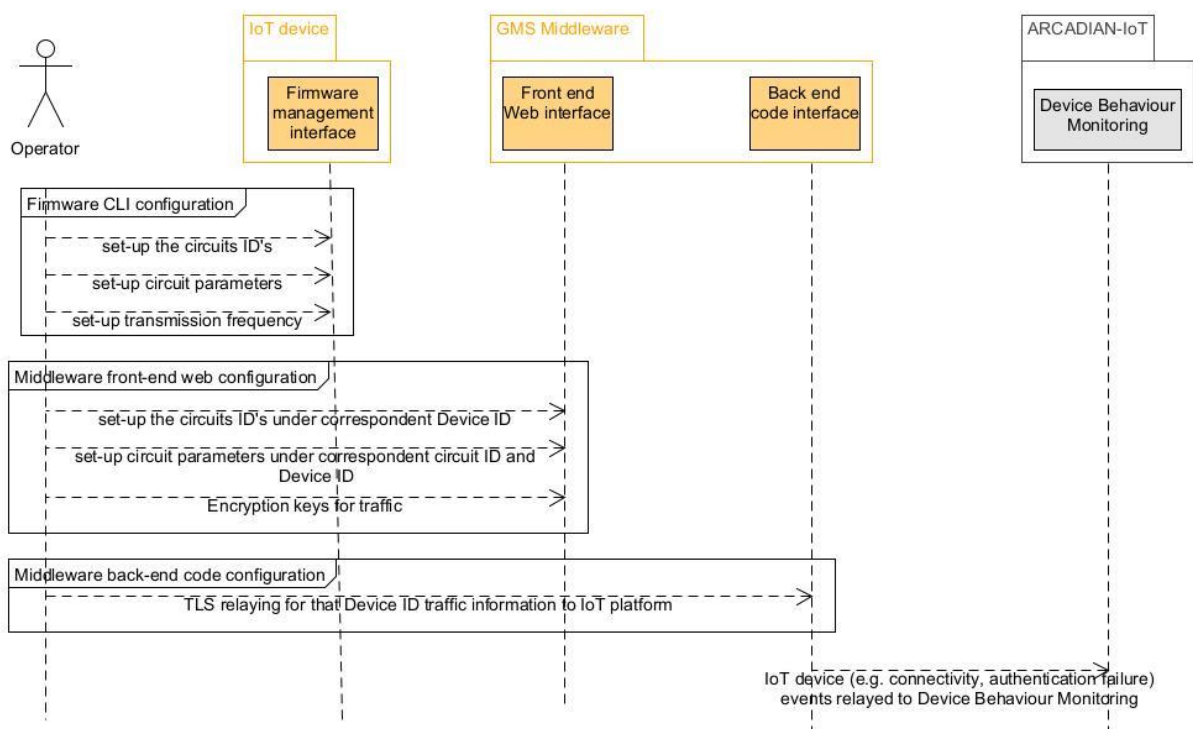


Figure 10 – UML diagram for use case B2

This use case implementation considered the following steps:

- prepare settings information
- connect to device by proper tool & method
- login into firmware CLI
- start provisioning of firmware settings
- start debugging function of CLI

- login to middleware front end
- start provisioning of middleware
- reboot the device
- monitor on CLI debugging, Middleware SoC (service operation centre) and Behaviour Monitoring System front end the traffic performed from IoT devices.

If there are intentionally applied (wrong) settings changes, these must be observed in the correspondent tool. Trial & error on traffic encryption fake keys is a main test of this use case.

### 3.3.3 Future plans

As future plans for B2 – device data gathering and transmission, once integration with **Reputation System** is performed, this one will be capable to deny a data transmission, if the wrong keys are used by a Device. It will be possible to integrate via API with **Self Aware Data Privacy**, to forward the device's data to a 3<sup>rd</sup> Party Platform (3PP) IoT Platform.

## 4. DOMAIN C – MEDICAL IOT

### 4.1 Application domain context

Monitoring patients at their homes, when possible, is important for the sustainability of health systems and for the comfort of the monitored persons. IoT systems, namely body sensor networks, provide solutions that make this possible. However, the use of IoT solutions for medical purposes raise concerns, such as those related to the patient's data privacy and security.

ARCADIAN-IoT Medical IoT (MIoT) scenario can be described as follows: The tumour was first removed from Maria (5 years old) in Ecuador, and she is now being treated in Madrid. It is a very rare cerebral sarcoma with a poor prognosis, associated with DICER, a rare genetic disorder that predisposes individuals to multiple cancer types. The paediatrics radio-oncology treatment is based on a proton medical device that generates, during the required number of sessions (e.g., 30), intensive radiotherapy by sending large amounts of proton to the brain tumour. Almost all patients receive radiotherapy and chemotherapy, but each of them undergoes different treatments and is treated in a personalized manner.

Considering the demanding volume of treatment sessions, reducing the number of consulting sessions for assessing the patient's well-being is very beneficial. For this purpose, a telemonitoring system is well accepted by the medical staff - team of doctors and nurses - and by the patients. Both see the solution as more comfortable and able of automatically providing an evolutionary record of the patients' status, and potentially getting the medical staff attention to relevant readings. This is the context for the application of ARCADIAN-IoT MIoT solution.

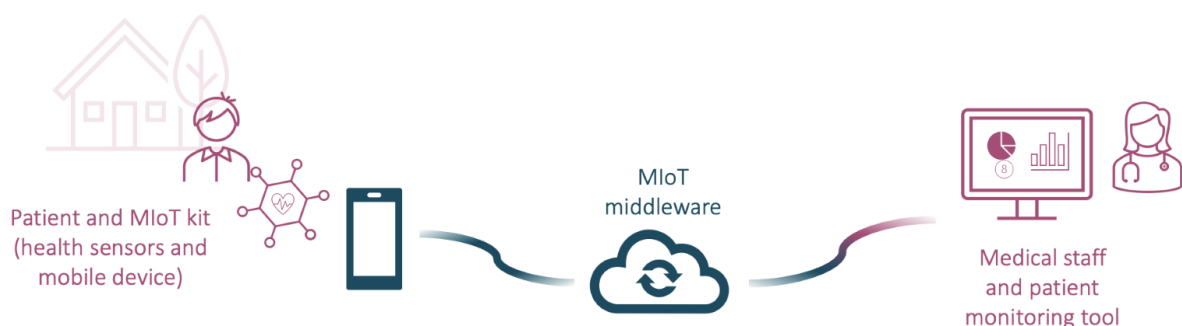


Figure 11 - MIoT high-level representation

According to medical experts (UNAV, ARCADIAN-IoT partner), to be effective, MIoT needs to be able to monitor patients considering a treatment protocol (readings frequency, medication, and other medical recommendations). It needs to collect, store, and present the evolution of vital signs, namely in the cardiac area, the heart rate, temperature, SpO2 and blood pressure, captured with the medical sensors and timely provide alerts for medical decision support. To complement these parameters, it should be possible for the patient to enter perceived symptoms in a mobile app, such as levels of fatigue, sweating, diarrhea, or others that can describe a symptom intensity.

To fulfil these requirements, the solution will rely in a MIoT kit that comprises a set of medical sensors, and a smartphone that will be used as gateway for the sensing devices and as interface for the patient to enter his/her perceived well-being. This kit is provided to the patients at the hospital. The solution will also include a MIoT middleware service for distributing securely the patients' data and generating health alerts; and a monitoring tool for the medical staff to check the patient's well-being, alerts and to change the monitoring protocol when needed.

## 4.2 Use Case C2 – MIoT capturing and sending vital signs and perceived health status

### 4.2.1 Description

<i>ARCADIAN-IoT Layers</i>
<b>Vertical plane:</b> Identity; Trust; Recovery. <b>Horizontal plane:</b> Privacy; Security; Common.
<i>Use Case Actors</i>
Patient.
<i>Use Case Story</i>
<p>At this stage, the patient is at home with the sensors placed in the body and synced with the smartphone. The MIoT app, authenticated in MIoT services, starts to run the health sensors reading cycle. When needed or according to the health monitoring plan, the patient also provides the perceived health status in the app. The main steps are:</p> <ol style="list-style-type: none"> <li>1. After getting and aggregating the data from the health sensors, the app in the smartphone, which works like a gateway between the sensors and the MIoT services, performs the <b>Hardened Encryption</b> of the payload with RoT – <b>eSIM</b> - information. The same happens if the patient uses the app form to inform about his/her perceived health status. The information is encrypted with RoT information in the smartphone before being sent.</li> <li>2. The smartphone sends the encrypted payload to the MIoT middleware services. In the case there is no connectivity available, the device stores the encrypted data locally. When communication is back in service, the stored data is sent to MIoT services with a timestamp assigned to the sensors data / perceived health status.</li> <li>3. With the patient authorization, MIoT middleware forwards the data to the supporting hospital</li> </ol>

monitoring tools. These third-party tools need to comply with ARCADIAN-IoT<sup>5</sup> to be able to decrypt the MIoT data. Decryption only happens with the patient authorization to a specific medical professional (which can be given in the hospital – C1) – enforced by the **self-aware data privacy** via the **Hardened Encryption** libraries and encryption settings.

5. If, for some reason, the smartphone is turned off, when it is turned on again it requests the patient to do the authentication in MIoT services with the **network-based credentials** (stored at hardware level) and the **SSI** (decentralized) again and starts to run the health sensors reading procedure. A biometric identification to access the app can apply as well (can be configured in the hospital when the smartphone is given to the person).

6. A re-**authentication** process (authenticate again according to a condition like the time since last authentication) may be applied for security purposes.

### **Relation with ARCADIAN-IoT Objectives**

Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 3: Enable distributed security and trust in management of persons' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

Objective 5: Provide a hardened encryption with recovery ability.

Objective 7: Enable proactive information sharing for trustable Cyber Threat Intelligence and IoT Security Observatory.

### **Use Case Priority**

*High: critical to several project objectives and without it some objectives could not be fulfilled*

*Average: Important, but other use cases have the same purpose*

*Low: Nice to have, but not critical for the project objectives*

High.

### **Use case preconditions**

1. Use case C1.

2. Sensors are well placed (app can help by informing when data is not being well received).

3. ARCADIAN-IoT behaviour monitoring and CTI components oversee the interactions of the IoT device (the IoT gateway) and services involved in order to trigger any security actions that are deemed necessary and update the trust knowledge. This may include dynamic reputation and authorization changes for the parties involved.

### **Use case postconditions**

1. MIoT app is transmitting encrypted data to the MIoT middleware, which can be forwarded, encrypted as well, to authorized IoT monitoring tools.

### **Entities/Scope (Person/IoT/Apps Services)**

<sup>5</sup> Requirements for a third-party to be compliant with ARCADIAN-IoT in D2.3

All.

### *Data used and data flow*

1. The data used in this use case are user identifiers and authentication material (from device, app and person), and the data gathered from the health sensors (payload).
2. The payload gathered from the health sensors and aggregated for communication, and the related timestamp, are encrypted with RoT information and sent to the MIoT middleware. If no communication is available, the payload is stored encrypted in the device until the transfer is possible.
3. New authentication processes using the device identifiers, which are stored in the device secure element and/or are decentralized, are triggered if needed, according to security policies.
4. Communication events are captured by ARCADIAN-IoT framework (e.g., behaviour monitoring, CTI) to infer potential threats and act if needed.

## 4.2.2 Implementation Status

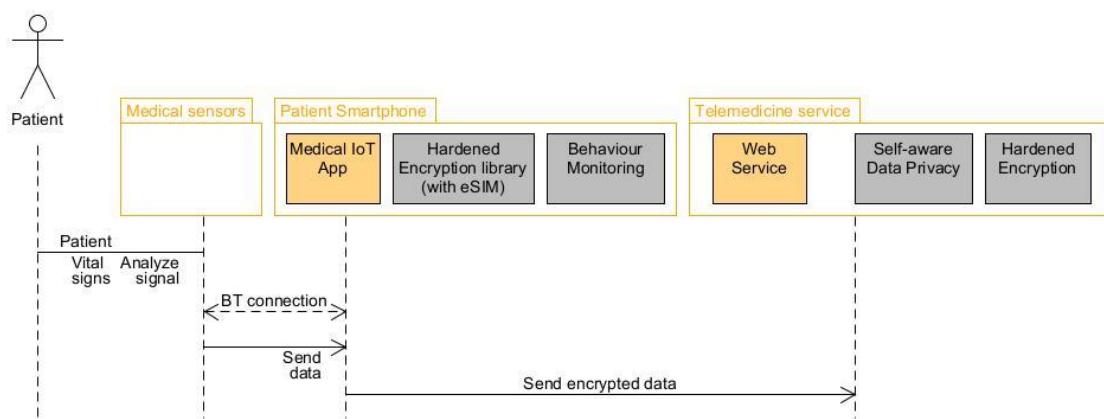


Figure 12 - UML diagram for use case C2

The overall implementation status of use case C2 is as follows:

- For achieving the “Sending data to the web platform” functionality, the following steps have been implemented:
  - o Integration of the MIoT app with **Hardened encryption**, to be able to store and transmit information securely.
  - o Reception of encrypted data by the web platform.
  - o Integration with the **Self-aware Data Privacy**, so that data was decrypted and encrypted again taking into account the policy associated to the user. Thus, giving access to the data only to the users (doctor/nurses) who are allowed access.
  - o Storage of encrypted data in the web platform database.
  - o Additionally, the MIoT app integrated **Remote Attestation** for increasing the trust over the smartphone and MIoT service integrity, via two different threads:
    - The implementation of the ability to transmit Reference Values (e.g. MIoT app signature) to the designated Verifier, via the framework’s message bus (RabbitMQ)
    - The integration of MIoT with the Attester package (which periodically

- collects claims regarding the smartphone operation)
- For enabling the “Perceived health status” functionality, the following was implemented or performed:
  - Web platform that allows the visualization of the stored data.
  - Visualization of the data after its correct decryption.

In order to implement this use case, it was necessary to implement three independent components: the Telemedicine modules, a MIoT App, and a web service. The associated implementation is detailed below.

### Telemedicine modules

RGB has developed a series of **telemedicine modules** for monitoring electrocardiogram (ECG), pulse oximetry (SpO2) and non-invasive pressure (NIBP) with Bluetooth BLE communication (AES-CCM cipher with 128-bit key length).

The possible signals to transmit are the following:

1. The physiological signal waveform collected by the different modules is:
  - a. SpO2. The signal collected is the alternating component of the light collected in the photodiode of the probe corresponding to the infrared channel (signal sampling every 15ms).
  - b. ECG. The signals collected are the available leads (signal sampling every 5 ms with a resolution of 1024 points per mV), that is, one lead in the case of the module with 2 electrodes and two leads in the case of 4 electrodes.
  - c. NIBP. The signal collected in this case has no physiological value. Its usefulness is to evaluate the accuracy of the measurement obtained, since the measurement may not be accurate in patients in movement or with cardiac arrhythmias. The signal collected is the pressure in the cuff (signal sampling every 20 ms). The alternating component of this signal is due to pressure oscillations in the cuff caused by the patient's arterial pulse, and its amplitude is what is analyzed in the oscillometric method to derive the patient's blood pressure measurements.
2. Measurements derived from the physiological signal. Signal processing allows the derivation of clinically relevant measurements:
  - a. SpO2. The measurements derived are oxygen saturation in arterial blood (range between 0-100%), heart rate (range between 30-250 ppm) and perfusion level (data about blood circulation with a range between 0 -128). In special cases the R-R period can also be measured (range between 200-2000 ms).
  - b. ECG. The derived measurement is the heart rate (range between 50-250 bpm). In special cases the R-R variability can also be measured (range between 200-2000 ms).
  - c. NIBP. The measurements derived are the patient's systolic, diastolic, and mean blood pressure (range 10-250 mmHg) and heart rate (range 30-250 bpm).
3. Status information. The information available is related to the technical alarm conditions that prevent the capture or processing of the signal and the information related to the battery charge level.

For communication with the telemedicine module, a Bluetooth-type wireless link is used in such a way that it does not limit the patient's movements. The information transmitted by the telemedicine module is collected and archived by a Gateway which, in turn, transmits this information to an external system for processing. The duration of patient monitoring can be variable depending on the application. For the definition of the protocol, it has been considered that the monitoring period can last several hours.

The Bluetooth technology used is the BLE (Bluetooth Low Energy). Its protocol is characterized by low electrical power consumption and by being a wireless data transmission technology. The



objective of this protocol is the transmission of data for a long period of time. The connection between the MIoT application and the telemedicine module uses AES-CCM encryption with a 128-bit key length to provide encryption and data integrity over the wireless link. In this way, we protect the connection between the telemedicine module and the MIoT application.

To allow for greater flexibility, the telemedicine module can function in two different modes:

- Transmission in real time. The module transmits the collected information continuously in real time. In case of loss of the Bluetooth connection, the data collected during the disconnection period is lost.
- Transmission by blocks in deferred mode. The use of the sensor module ubiquitously does not guarantee the maintenance of the Bluetooth connection permanently. This working mode allows information not to be lost. Information is continuously stored in non-volatile memory and is transmitted each time a memory sector is full. If the transmission fails, it is tried again the next time.

The structure of the frame to be used for communication is as follows, from most to least significant bytes:

1. Command. Byte that identifies the type of the frame.
2. ID. A byte that is used for a double function: identification of the module type and identification of the measured session.

The least significant byte corresponds to the type of module, being 0001 the pulse oximetry module, 0010 ECG module with 4 electrodes, 0011 ECG module with 5 electrodes and 0100 non-invasive pressure module.

The most significant byte identifies the measurement session, from 0 to 16.

3. Data. It consists of a variable number of bytes.
4. CKS. It consists of an integrity check byte of the received frame, its calculation is based on the checksum of the previous bytes in the frame. The formula is the following:

$$\text{CKS} = \text{not}(\text{Command} + \text{Id} + [\text{Data0}] + \dots + [\text{DataN}] + 1) \& 0x7F$$

The integrity of the frame is checked with the following calculation:

$$((\text{Command} + \text{Id} + [\text{Data0}] + \dots + [\text{DataN}]) + \text{CKS}) \& 0x7F = 0$$

5. FT. Byte indicating the end of the frame, 0xFF is used.

All the bytes that make up the frame can take all possible values. To distinguish a data byte from a control one, the value 0xFE is used to precede each of the data bytes of the frame whose value coincides with a control one. According to these definitions, the control bytes are: 0xFF, frame indicator and 0xFE data byte indicator. Therefore, when the value of a data byte matches these two values, this byte will be preceded by 0xFE.

It is important to emphasize that the CKS byte must be calculated on the original frame, therefore, the 0xFE control bytes must be eliminated to verify that the CKS byte is correct.

## MIoT App

This application has been programmed for mobile phones with the Android operating system. Below we will show the different screens that we have implemented for a correct operation of the use case:

- The first thing is the login screen. As you can see in the image, the login screen is a conventional screen, where the user is asked for the email and password fields to access their account. If you do not have an account already created, you need to create one. Clicking on "New User?" navigate to the corresponding screen to create an account. The operation of "Login" and that of "registry" will be implemented and described in detail during use case C1 within P2.

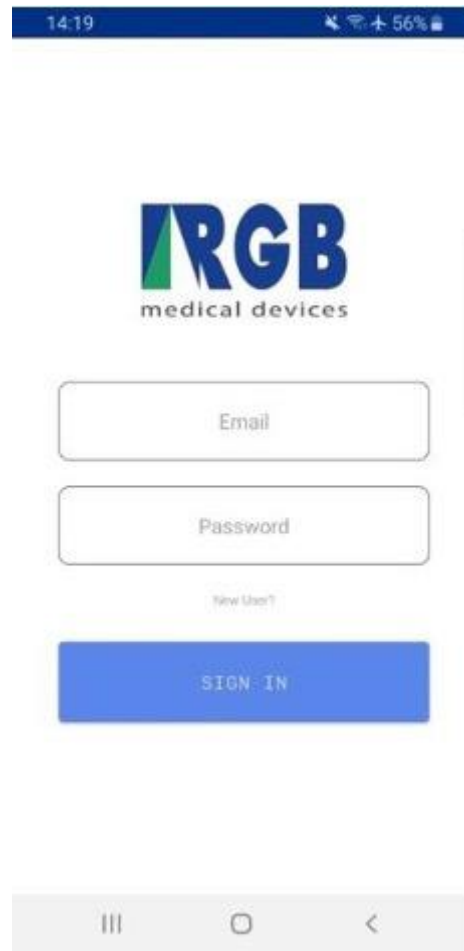


Figure 13 - MIoT App Login

- In the following figure, it represents once the user enters his account with his credentials. This screen is designed so that the user can see the most relevant information at that moment, that is, what tests must be carried out in the future. In this way, a calendar is displayed where the days that the user must carry out a test are marked, by clicking on each day you can see the different tests to be carried out. All these tests are programmed by the doctor from the web (use case C3), then the data is uploaded to the database and the MIoT application reads this data in this format.



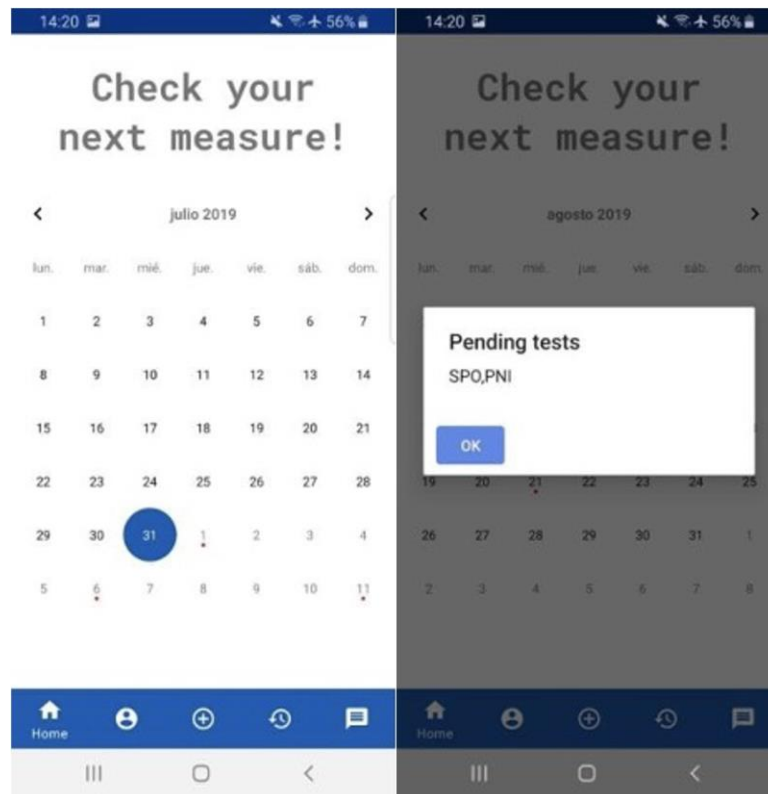


Figure 14 - MIoT App home screen

- The following figure shows the user profile screen. This section shows the personal information of the user, from here you can view and edit all the information. It also includes a button to be able to log out of the account.

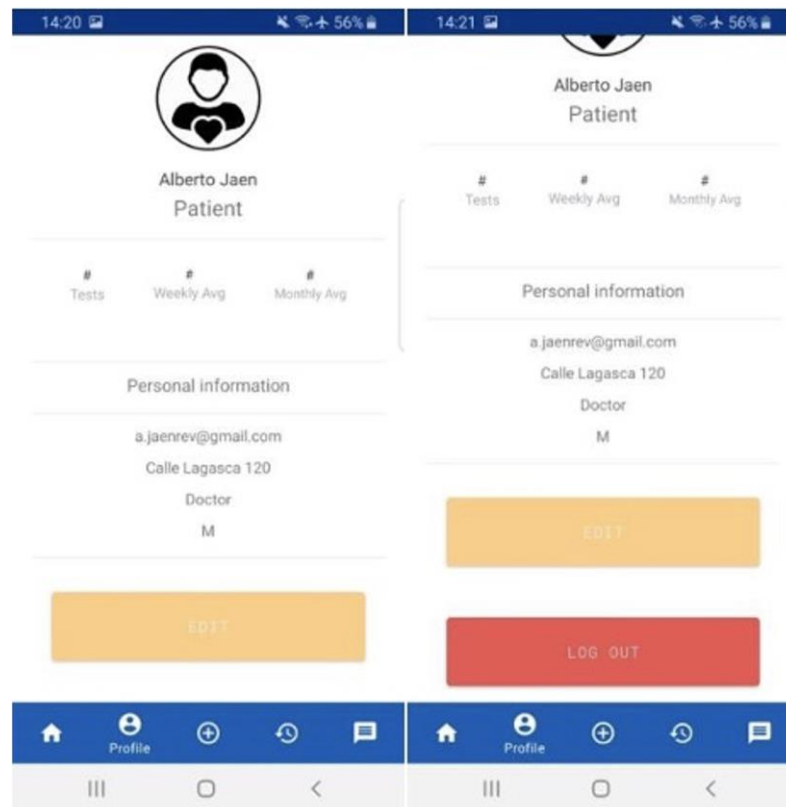


Figure 15 - MIoT App profile screen

- The following figure shows the screen dedicated to taking vital sign measurements. This screen is the main screen of the MIoT application, in which the user manages the connections with each device and can see the data being saved in real time.

As can be seen in the following figure, it has been implemented in such a way that the management of connected and disconnected devices is separated to facilitate interaction. The grey icon represents a nearby unconnected BLE device, in this case it represents an SPO2 device. It is the first element of a horizontal bar where the rest of the non-connected devices would be placed.

Below this element there is a vertical list where all the connected devices are located. It has been implemented in such a way that the information on this screen is smaller compared to the information provided on the main screens of this device. The intention is not to overwhelm the user with too much information.



Figure 16 - MIoT App "new test" screen

- The screen shown in the following figure is responsible for representing all the information obtained through the ECG2 module (electrocardiogram with two electrodes). The information represented is based on the beats per minute, located in the upper right corner, and the electrocardiogram graph with one lead. You can also view all the information related to the device and its status, such as the device type, connection status, and a warning in case of an error. In the case of clicking on the alarm icon, a window will be displayed to describe the type of error.

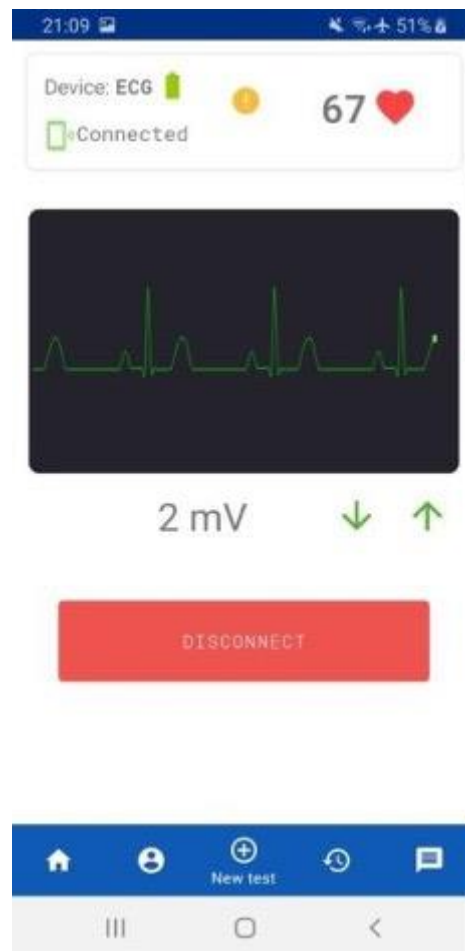


Figure 17 - MIoT App ECG screen

- The screen shown in the following figure matches the previous one in terms of device information (type, battery level, and connection status) but differs in terms of the data represented (non-invasive blood pressure NIBP). The information represented is the systolic pressure, the diastolic pressure and the beats per minute.

It is important to note that this device has an extra button compared to the others, the “Start” button. This button is necessary since the measurement and filling of the pressurized air cuff does not start until this button is pressed. Once pressed the text on this button will change to “Stop” to allow the user to stop filling the cuff if necessary. Cuff filling will stop automatically when measurements have been taken correctly.



Figure 18 - MIoT App NIBP screen

- The screen shown in the following figure is the same in terms of connection status information as the previous ones. Regarding the information represented related to vital signs, the percentage of oxygen in the blood is represented, the pulses per minute, the perfusion bar and a graph with the pulses detected by the sensor.



Figure 19 - MIoT App SpO2 screen

### Web service

For this implementation, we have used Django. Django is a high-level Python web framework that enables rapid development of secure and maintainable websites.

For this implementation, six applications have been integrated, all of them referring to the most important issues that the platform must address. In this way, the structure is as follows:

```

WEB_TELEMEDICINE/
    alerts/
    charts/
    chat/
    events/
    users/
    webapp/
    telemedicine/
    manager.py
  
```

Within the WEB\_TELEMEDICINA/ directory you can find the manager.py file together with

seven folders. The following table describes the fields that are referenced by each of the application folders:

directory	field
alerts/	Application that manages the alerts derived from specific monitoring sessions
charts/	Application used to represent the signals collected in the monitoring sessions
chat/	Application that manages the platform's messaging service
events/	Application that manages patient monitoring sessions
users/	Application that manages platform users
webapp/	Application that manages all views of the platform

Table 1 - Applications included in the telemedicine web service

The seventh directory, telemedicine/, refers to the Python package, which is a fundamental folder within the system structure.

Finally, we will explain the different views that we have implemented in the web service. In this section, we will show the generic views as when explaining the MIoT app and the modules, and also specific views of the C2 use case:

- The first thing is the login screen. As you can see in the image, the login screen is a conventional screen similar to the MIoT app, where the user is prompted for the email and password fields to access their account. Similar to the MIoT app, if you don't already have an account created, you need to create one. Clicking on "New user?" will navigate to the corresponding screen to create an account. The "Login" operation and the "register" operation will be implemented and described in detail during use case C1 within P2.

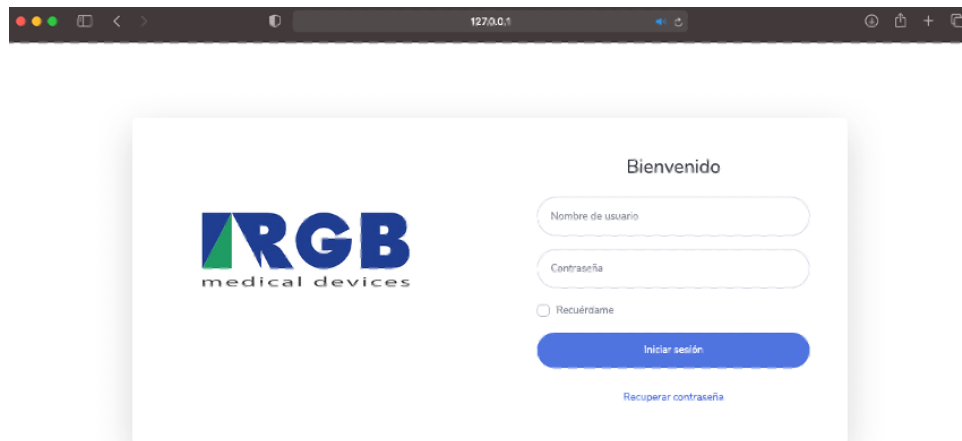


Figure 20 - Web service login page

- Once the login is done, there are two possibilities. Either a patient or a doctor/nurse has logged in. The main view we are interested in for use case C2 is patient. First of all, it is required to show if the patient has any new messages from the assigned doctor. To do this, it is necessary to make a comparison between the dates of the patient's last connection and the last chat message. If the chat message is more recent, the artifact that indicates to the user in the interface that he has unread messages must be activated. Next, the patient should be told if he has a session soon, for which it is necessary to compare the session start date with the date of his next monitoring session. Continuing with the monitoring sessions, the patient will have at their disposal a calendar on their main page that will allow them to see both previous sessions (use case C2) and future ones. Finally, the patient receives notifications in case he has triggered an alert in one of his monitoring sessions.

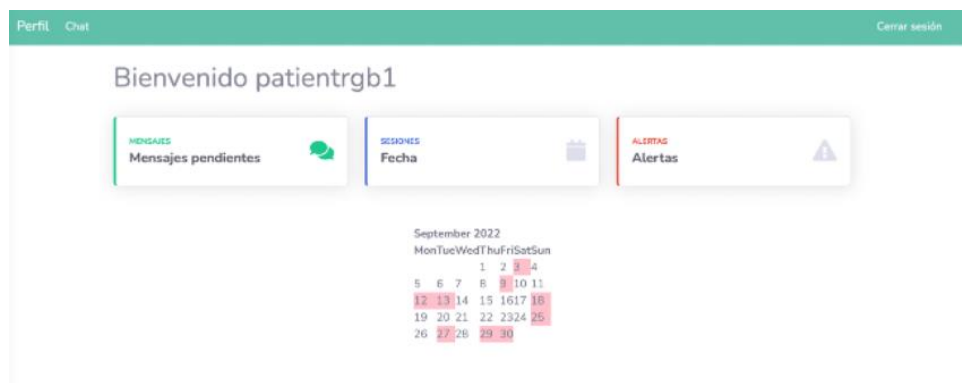


Figure 21 - Web service patient's home page

- In the C2 use case, it is required to be able to visualize the data that we have sent from the MIoT app.



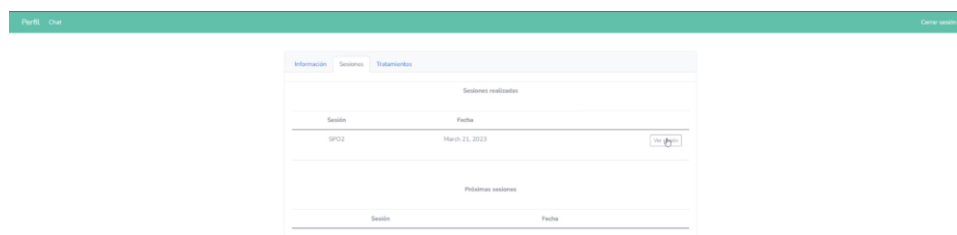


Figure 22 - Web service patient's sessions page

- After selecting the session, the web platform requests the information from the database, decrypts it and displays it on the screen as in the image below:

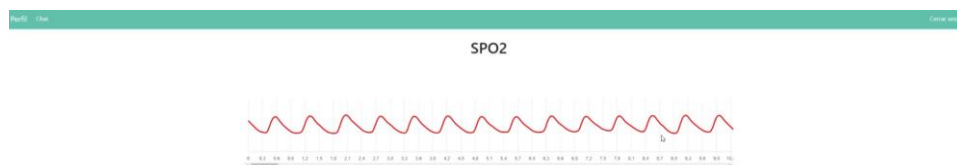


Figure 23 - Web service patient's last session page

### 4.2.3 Future plans

Within the C2 use case there are several components that will be integrated for P2. One possible integration will be to use **Network-based authorization enforcement** for user to log in from the MIoT App via Arcadian-IoT. This will allow access depending, in addition to the credentials, on the reputation of the device used within the **Reputation Systems**, which will modify its values based on what is sent by **Behavior Monitoring**.

Regarding the components integrated in P1, we maintain a policy of continuous improvement that will make it possible that there are modifications and/or updates between P1 and P2 that we can report in P2, although we do not have a specific plan to improve specific aspects within this C2 use case.

## 4.3 Use Case C3 – Personal data processing towards health alarm triggering

### 4.3.1 Description

#### ARCADIAN-IoT Layers

**Vertical plane:** Trust.

**Horizontal plane:** Privacy; Security; Common.

#### Use Case Actors

Patient.

### Use Case Story

This use case refers to the health data processing, in the cloud (in a data processing unit of MIoT middleware), with the purpose of detecting and triggering health alarm conditions in the hospital monitoring tool. The key story aspects are:

1. Having the encrypted data that results from C2 in a database of the MIoT service, the processing unit of MIoT wishes to analyse the data and trigger alerts if needed. This MIoT processing unit needs to be registered in ARCADIAN-IoT and the patient needs to authorize the processing of his/her data for this purpose (**self-aware data privacy**). If the patient revokes the grant for processing his/her data, new policies are applied to the encryption of data, so that this unit cannot access the data.
2. To ensure the patient privacy and keep him/her anonymous in the processing unit of MIoT services, the decryption techniques applied will just decrypt the payload (sensor data), keeping the **person identity encrypted/anonymized** at all times. If needed, the payload that the processing unit receives should include aspects like age or gender, and pathological data, to be able to infer the alarm conditions without identifying the individual.
3. ARCADIAN-IoT **behaviour monitoring**, and **CTI**, oversee and interpret this MIoT service behaviour, adjusting its trustworthiness **reputation** and its **authorization** to continue receiving health data from patients, which is revoked if the service is found not to be trusted.
4. According to medical protocols or related health patterns learned from other patients (all anonymous) the MIoT processing unit detects alarm conditions. When detected, these alarm conditions are encrypted and merged with the encrypted identification of the patient.
5. With the patient authorization, MIoT middleware forwards the encrypted data that includes the alarms to the supporting hospital monitoring tools. To be able to decrypt the alarms, these third-party tools need, as all that comply with ARCADIAN-IoT, to have robust identity and authentication mechanisms, allow security behaviour monitoring and need to authenticate in ARCADIAN-IoT MIoT services to receive the cryptographic material to decrypt the data sent to them. Decryption, by the medical staff, only happens with the patient authorization (which can be given in the hospital – C1).

### Relation with ARCADIAN-IoT Objectives

- Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.
- Objective 2: Enable security and trust in the management of objects' identification.
- Objective 3: Enable distributed security and trust in management of persons' identification.
- Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.
- Objective 5: Provide a hardened encryption with recovery ability.
- Objective 7: Enable proactive information sharing for trustable Cyber Threat Intelligence and IoT Security Observatory.

### Use Case Priority

**High:** critical to several project objectives and without it some objectives could not be fulfilled

**Average:** Important, but other use cases have the same purpose

**Low:** Nice to have, but not critical for the project objectives

High.

### *Use case preconditions*

1. Use case C2.
2. ARCADIAN-IoT behaviour monitoring and CTI components monitor the interactions of the devices (and when applicable, services) involved in order to trigger any security actions that are deemed necessary and update the trust knowledge. This may include dynamic reputation and authorization changes for the service.
3. Appropriate cryptographic material distributed to the alert processing unit of MIoT.

### *Use case postconditions*

1. When relevant, patient health alarms are triggered and sent to the hospital encrypted.

### *Entities/Scope (Person/IoT/Apps Services)*

Services.

### *Data used and data flow*

1. Patient identification and health data are used in this use case.
2. When arriving to the processing unit of MIoT middleware, coming from MIoT kit (health sensors and gateway), the part of the health data is decrypted, keeping the patient identification encrypted (the service does not have authorization to decrypt patient identification).
3. The health data is kept in a time series associated with that patient (who is anonymous to the system) and processed according to health rules to infer alarm conditions.
4. If an alarm condition is detected, it is encrypted and sent to the monitoring tool to be seen by authorized medical staff.

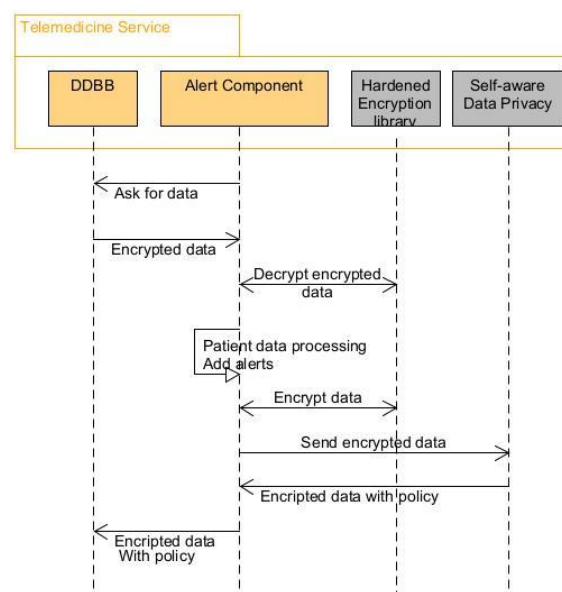


Figure 24 - UML diagram for use case C3

### 4.3.2 Implementation Status

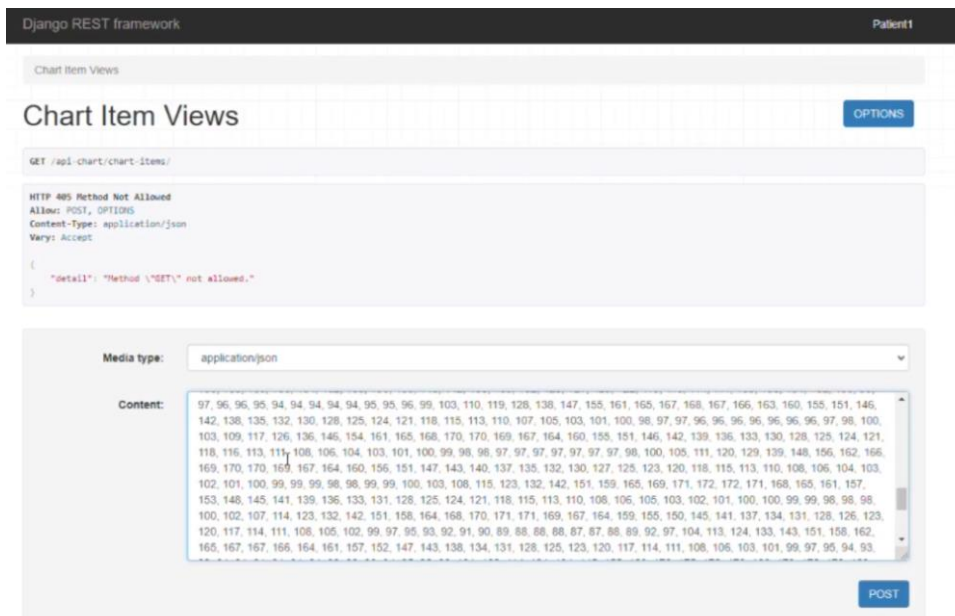
The implementation status of use case C3 is as follows:

Based on the implementation described in use case C2 and specifically within the web platform, a module has been created that can request the decryption of medical data (The encrypted data in the database is sent to the **Self-aware data privacy** proxy which decrypts the data) to check if there are alerts that must be shown to the doctors/nurses in charge of monitoring the patient. This module does not have access to the personal information of the specific user, it only analyzes medical data anonymously in search of alarms.

These alarms, if they exist, are added to the medical record to be viewed by the doctor or nurse in charge.

Below, we show some screenshots in which we can see how, when sending a session with data outside the limits that are considered safe and, therefore, with the need to show the alarm for the doctor or nurse to analyze it, the module detects it and displays the corresponding alarm.

1. Sending a patient session with a low saturation measure:



Django REST framework Patient1

Chart Item Views

Chart Item Views

OPTIONS

GET /api-chart/chart-items/

HTTP 405 Method Not Allowed  
 Allow: POST, OPTIONS  
 Content-Type: application/json  
 Vary: Accept

```
{
  "detail": "Method \"GET\" not allowed."
}
```

Media type: application/json

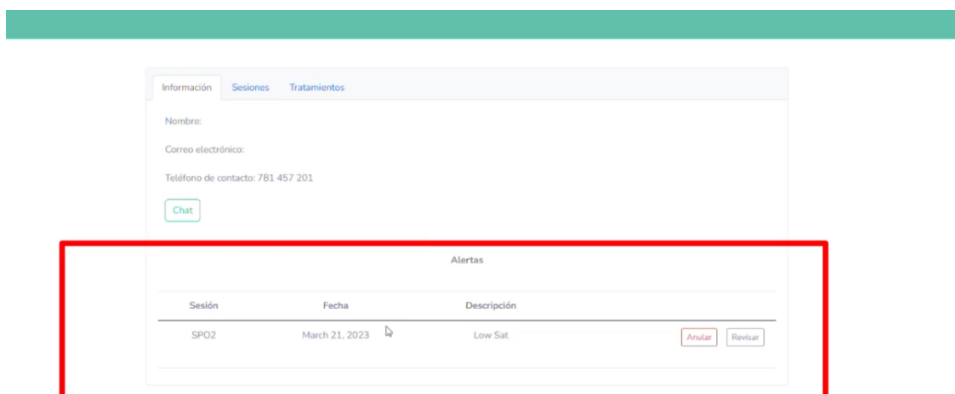
Content:

97, 96, 96, 95, 94, 94, 94, 94, 95, 95, 96, 99, 103, 110, 119, 128, 138, 147, 155, 161, 165, 167, 168, 167, 166, 163, 160, 155, 151, 146, 142, 138, 135, 132, 130, 128, 125, 124, 121, 118, 115, 113, 110, 107, 105, 103, 101, 100, 98, 97, 97, 96, 96, 96, 96, 96, 96, 97, 98, 100, 103, 109, 117, 126, 136, 146, 154, 161, 165, 168, 170, 170, 169, 167, 164, 160, 155, 151, 146, 142, 139, 136, 133, 130, 128, 125, 124, 121, 118, 116, 113, 111, 108, 106, 104, 103, 101, 100, 99, 98, 98, 97, 97, 97, 97, 97, 97, 98, 100, 105, 111, 120, 129, 139, 148, 156, 162, 166, 169, 170, 170, 169, 167, 164, 160, 156, 151, 147, 143, 140, 137, 135, 132, 130, 127, 125, 123, 120, 118, 115, 113, 110, 108, 106, 104, 103, 102, 101, 100, 99, 99, 99, 98, 98, 99, 99, 100, 103, 108, 115, 123, 132, 142, 151, 159, 165, 169, 171, 172, 172, 171, 168, 165, 161, 157, 153, 148, 145, 141, 139, 136, 133, 131, 128, 125, 124, 121, 118, 115, 113, 110, 108, 106, 105, 103, 102, 101, 100, 100, 99, 99, 98, 98, 98, 100, 102, 107, 114, 123, 132, 142, 151, 158, 164, 168, 170, 171, 171, 169, 167, 164, 159, 155, 150, 145, 141, 137, 134, 131, 128, 126, 123, 120, 117, 114, 111, 108, 105, 102, 99, 97, 95, 93, 92, 91, 90, 89, 88, 88, 88, 87, 87, 88, 89, 92, 97, 104, 113, 124, 133, 143, 151, 158, 162, 165, 167, 166, 164, 161, 157, 152, 147, 143, 138, 134, 131, 128, 125, 123, 120, 117, 114, 111, 108, 106, 103, 101, 99, 97, 95, 94, 93.

POST

Figure 25 - Web service sending session with an alarm

2. Low Sat alert is generated and shown to the doctor/nurse:



Información Sesiones Tratamientos

Nombre:

Correo electrónico:

Teléfono de contacto: 781 457 201

Chat

Alertas

Sesión	Fecha	Descripción
SPO2	March 21, 2023	Low Sat

Cancelar Repasar

Figure 26 - Web service show alarm

### 4.3.3 Future plans

Within the C3 use case there are several components that will be integrated for P2. One possible integration will be to use **Network-based authorization enforcement** for medical personnel to log in from their browser via Arcadian-IoT. This will allow access depending, in addition to the credentials, on the reputation of the device used within the **Reputation Systems**, which will modify its values based on what is sent by **Behavior Monitoring**.

Regarding the components integrated in P1, we maintain a policy of continuous improvement that will make it possible that there are modifications and/or updates between P1 and P2 that we can report in P2, although we do not have a specific plan to improve specific aspects within this C3 use case.

## 4.4 Use Case C4 – Monitor a patient and update a patient monitoring protocol.

### 4.4.1 Description

<i>ARCADIAN-IoT Layers</i>
<b>Vertical plane:</b> Identity; Trust. <b>Horizontal plane:</b> Privacy; Security; Common.
<i>Use Case Actors</i>
Medical professional.
<i>Use Case Story</i>
<p>Even though it depends on the data collected in the medical IoT devices, monitoring patients is one of the most relevant functional use cases of the whole medical IoT domain. The system is thought to monitor patients in an efficient and secure way, while they are at home. Related to the patient monitoring, which can lead to decisions to change medical protocols, is the sending data/commands to the medical IoT devices. An example can be the request to change the devices reading frequency, or just request the patient to say how he/she feels more often (through the app). In this case, a use case story with ARCADIAN-IoT participation is:</p> <ol style="list-style-type: none"> <li>1. A medical professional authenticates in the MIoT hospital platform with a <b>strong multi-factor authentication</b>, where one of the factors is an <b>SSI</b>. MIoT services validate the requesting user and the app (MIoT hospital platform) identity and assess its reliability (<b>reputation</b>) to grant him/her access to the services. Using the ARCADIAN IoT ID token the medical professional obtains private cryptographic keys needed to access the data of the patients that authorized him/her to do so.</li> <li>2. When successfully logged in, the medical professional selects a patient to whom he/she has access to, or requests access to a new patient. The medical professional can also have a dashboard for monitoring several patients that authorized the access to their data to that professional, and this dashboard can include a section with health alerts related with those patients.</li> <li>3. Patient's data is kept encrypted until being requested by an authorized medical professional. At this moment it is decrypted with cryptographic material provided by ARCADIAN-IoT <b>hardened encryption</b> key management component directly at the medical professional.</li> </ol>

4. If the medical professional wants to change the monitoring protocol of a given patient, it requests MIoT services to send commands to that patient MIoT app, encrypted with hardened encryption.
5. If the medical professional and the services he/she is using have the necessary **authorization**, the MIoT app configuration data is retrieved from the MIoT App, and sent to the MIoT hospital platform encrypted.
6. Having the necessary authorization, the medical professional using the web interface decrypts the data sent by the patient app. The user edits the intended fields and requests the sending of the new data to the MIoT app, **encrypted** again. The updated data is kept encrypted in all the flow and only accessible (decrypted) by the MIoT app, in the device (smartphone) it is being sent to.
7. To be able to receive the new commands, the smartphone needs to be on, connected and the patient **securely authenticated with more than one factor** in ARCADIAN-IoT MIoT middleware.
8. When the patient device and the MIoT app receive the encrypted request, it decrypts it. If the data is successfully decrypted it means that the command was really directed to that device, and it shall be executed by the app. If the device cannot decrypt the request, it informs MIoT middleware and discards it.

#### *Relation with ARCADIAN-IoT Objectives*

Objective 1: To create a decentralized framework for IoT systems - ARCADIAN-IoT framework.

Objective 2: Enable security and trust in the management of objects' identification.

Objective 3: Enable distributed security and trust in management of persons' identification.

Objective 4: Provide distributed and autonomous models for trust, security and privacy – enablers of a Chain of Trust.

Objective 5: Provide a hardened encryption with recovery ability.

Objective 7: Enable proactive information sharing for trustable Cyber Threat Intelligence and IoT Security Observatory.

#### *Use Case Priority*

*High: critical to several project objectives and without it some objectives could not be fulfilled*

*Average: Important, but other use cases have the same purpose*

*Low: Nice to have, but not critical for the project objectives*

High.

#### *Use case preconditions*

1. Use case C1.

2. Medical professional, user of the MIoT hospital platform registered and with an identification matching an ARCADIAN-IoT person identification.

3. Medical professional authorized by a patient (or more) to decrypt his health data for well-being monitoring purposes.

4. ARCADIAN-IoT behaviour monitoring and CTI components monitor the interactions of the devices (and when applicable, services) involved in order to trigger any security actions that

are deemed necessary and update the trust knowledge. This may include dynamic reputation and authorization changes for the service.

#### ***Use case postconditions***

1. Authorized medical staff is able to monitor a patient health and update the medical protocol.
2. In the case of a medical protocol change, the MIoT kit has new information to update the medical monitoring routines.

#### ***Entities/Scope (Person/IoT/Apps Services)***

All.

#### ***Data used and data flow***

1. The data used in this use case includes the medical professional identifiers and authentication material, third-party/monitoring tool identifiers, patient identifiers, his/her medical data, generated alarms, and cryptographic material for decrypting the person health data in the monitoring tool. The current medical monitoring protocol and a new one may be used as well, and the related cryptographic material for encrypting/decrypting the new protocol.
2. Regarding the patient health monitoring in the hospital MIoT tool, a medical professional, after being authenticated (according to the ARCADIAN-IoT compliance protocol described in D2.3), requests the access to a patient data record. If he/she is authorized (trustworthy) to access patients' data, and if that patient has authorized that medical professional to access his/her data, and the service he/she is using is trustworthy, the data is retrieved and decrypted. Data includes the health readings (shown in a relevant format) and health alarms. After the professional logs out from the system, the patient decrypted data and related cryptographic material is deleted.
3. Also, in the MIoT monitoring tool, for changing a given patient medical protocol, the same authorized professional with access to that patient data, requests the change. Being authenticated and having the necessary authorization, the information is retrieved encrypted from the patient device. It is decrypted in the web interface and editable for the medical professional to change it. The new commands are encrypted and sent, through the MIoT middleware, to the patient device. At the device, the commands are decrypted and, if successful, applied. If the device is unable to decrypt the data, informs MIoT middleware and discards it. ARCADIAN-IoT components monitor the event and acts accordingly.



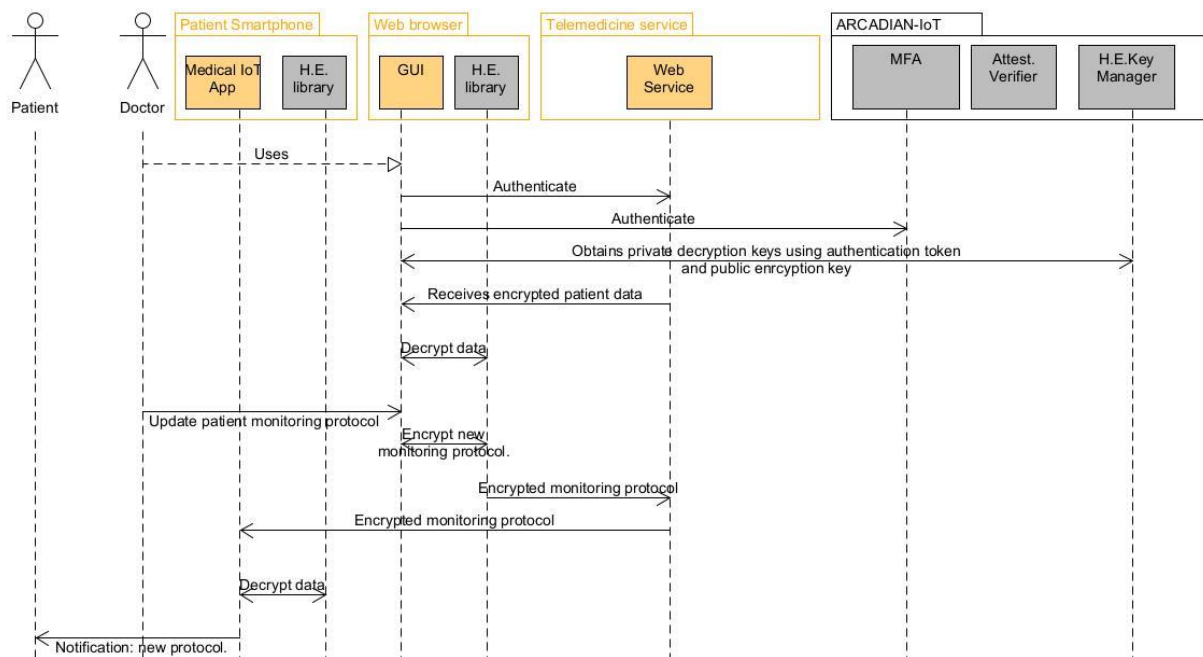


Figure 27 - UML diagram for use case C4

#### 4.4.2 Implementation Status

The implementation status of use case C4 is as follows:

The doctor or nurse in charge of a specific patient can enter their monitoring protocol and make the pertinent modifications. For this process, the corresponding module has been created within the web platform. For this process, the doctor or nurse must be authorized to access the patient's data and, therefore, be included in the patient's encryption policy.

Based on the implementation described in the C2 use case and specifically within the web platform, we are going to describe the views that are displayed when the user used to authenticate belongs to a doctor or nurse:

- The first thing we are going to show is the home view of these users. In this case, the screen shows a list of patients, those assigned to them at that moment, in a specific order. To do this, it is necessary to launch a query to the database. In the event that the user is of the medical type, knowing their identifier (a unique unsigned integer for each patient), the patients under their care are obtained, since the patients have an attribute in the database table data that refers to the identifier of the doctor that they have assigned and have accepted. Regarding the order of the list, firstly, the patients with alerts that have not been reviewed appear. In addition, from the list the doctor/nurse has the possibility of accessing the profile of the selected patient. Below, the patients who do not have any active alerts at that time appear and the doctor can also access their profile.



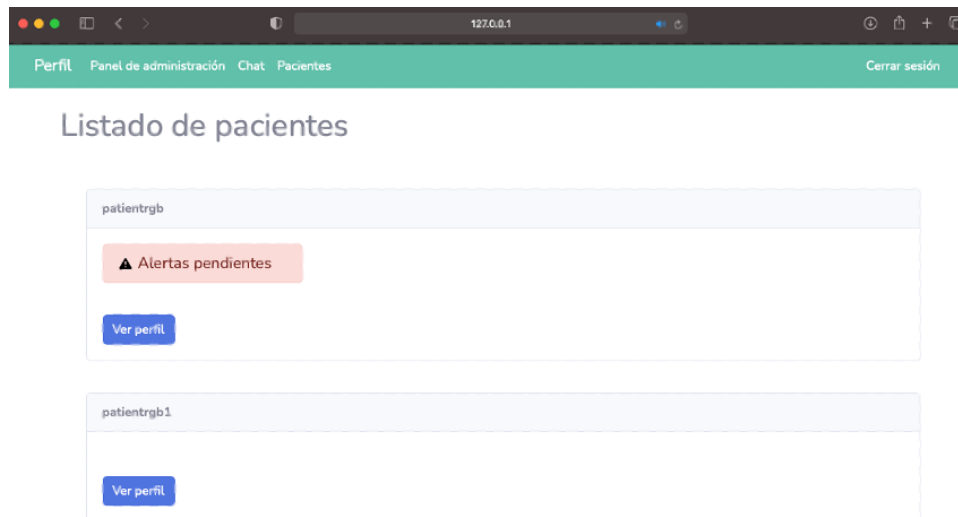


Figure 28 - Web service doctor's home page

Many functionalities of the application are found in the patient profile view. It is divided into three pages:

- In the first, the patient's personal information is displayed (name, first surname, mobile phone and email), which is obtained through a query to the database, knowing the patient's identifier and selecting the desired attributes. In addition, this first window shows the alarms that the patient may have, indicating the type of session in which it occurred, the date and a description of the alert.

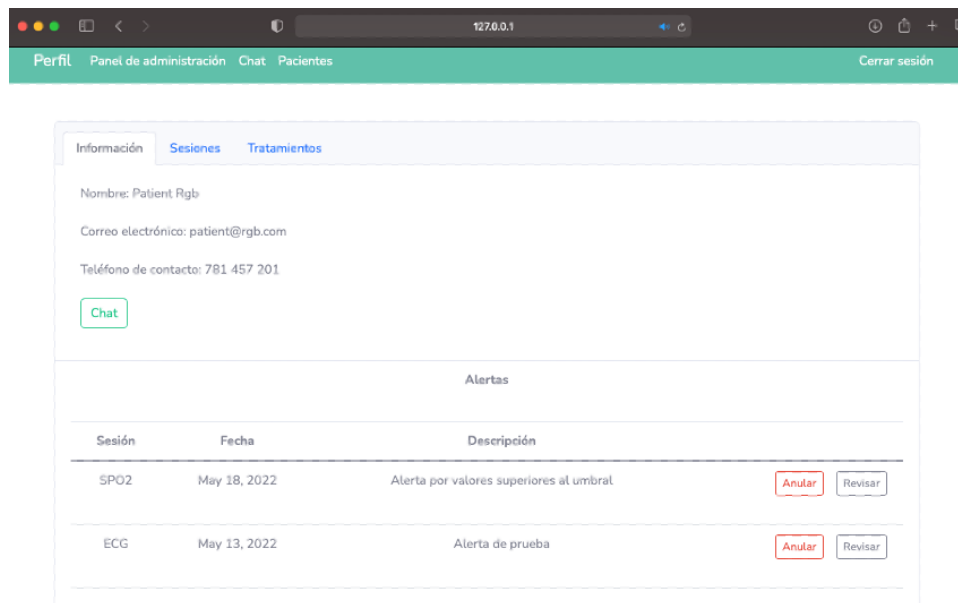


Figure 29 - Web service doctor's patient's first page

- On the second page is the one related to the C4 use case, which shows the information regarding the patient's sessions and allows you to modify your monitoring protocol. In the first place, there is the form to add a new monitoring protocol, for which it is necessary to enter the type of session and the date on which it is to be carried out. Next, the patient's past sessions are shown, ordered from most to least recent, indicating the date of the session and being able to access the view that shows the monitoring signals that were collected in that session. Finally, the next sessions of the patient are displayed.

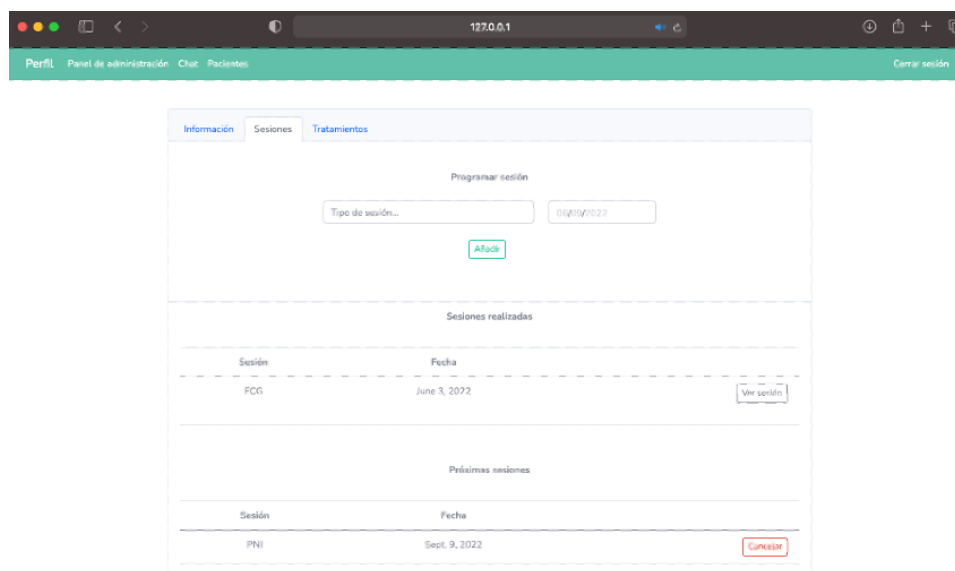


Figure 30 - Web service doctor's patient's second page (C4 use case)

- In the third and last window of the view, are the treatments that the patient may have.

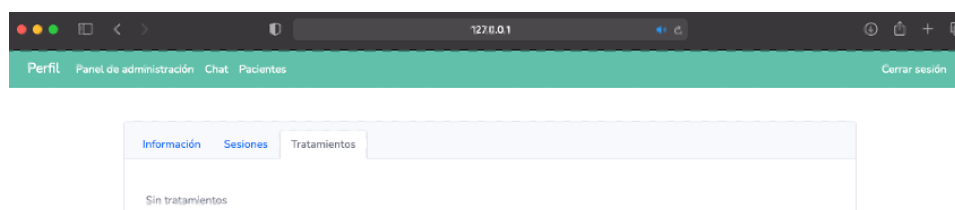


Figure 31 - Web service doctor's patient's third page

#### 4.4.3 Future plans

Although in this C4 use case we also maintain a policy of continuous improvement on the implementations made that will make enable the possibility for updates between P1 and P2; particularly, an important update planned for P2.

During P1, we have been able to implement a functional C4 use case with the essentials described initially. Unfortunately, we came across the unplanned need for web authentication, needed for the doctor/nurse to successfully authenticate through ARCADIAN-IoT. Having planned the use case C1, which includes user registration and authentication, to be ready only in P2, the authentication functionalities are not yet implemented, and therefore we cannot securely confirm the authentication of the doctors/nurses when entering to update the patient monitoring protocol.

In addition to the above, the integration of several components is planned for P2. One possible integration will be to use **Network-based authorization enforcement** for medical personnel to log in from their browser via ARCADIAN-IoT. This will allow access depending, in addition to the credentials, on the reputation of the device used within the **Reputation Systems**, which will modify its values based on what is sent by **Behavior Monitoring**.

The web platform decrypts the data through the **Self-aware data privacy** proxy, encrypts it with the **Hardened Encryption** library and sends it to the medical staff's browser. In the browser it will be decrypted with the same Hardened Encryption library, and it will be shown to the doctor that, after making the required modifications, it will be re-encrypted with the **Hardened Encryption** library and sent to the **Self-aware data privacy** proxy to encrypt it and the web platform stores it

in the database.

## 5. CONCLUSIONS

This document reports the implementation status for each of the 3 IoT application domains addressed in ARCADIAN-IoT, depicting as well how they are currently leveraging ARCADIAN-IoT (i.e. its first prototype - P1).

Regarding Domain A, led by LOAD and scoping the support of the DGA service, the use cases planned for P1 (A1, A2 and A3) were completely implemented, leveraging the expected ARCADIAN-IoT features (with emphasis on the Identity and Trust planes). Namely, the user registration via the smartphone (use case A1) is now ready, including the insertion of basic personal data, the generation of an ID Token, and the usage of the smartphone camera to collect and send facial images from different specific positions - to be used later in the authentication process. The authentication process is also supported (Use case A2), where a user authenticates using the DGA App via its smartphone fully leveraging ARCADIAN-IoT's multi-factor authentication approach (including biometric data, network identifier and Decentralized Identifiers). The checking and/or editing of personal data data – including facial data - is also supported (use case A3). Some adjustments are yet to be made in these use cases, particularly regarding integrations planned for P2 (e.g. taking advantage of ARCADIAN-IoT Security or Privacy plane features). As next steps, use cases involving the Drone component, as well as privacy or security incidents, will be the main focus of the future actions towards prototype P2.

Regarding Domain B, led by BOX2M and addressing the grid management service, the use cases planned for P1 (B1, B2) were implemented as planned. In the scope of supporting the use cases, BOX2M had to design and develop the industrial IoT device, as a set of 11 plugin boards, with dedicated category functions (e.g. motherboard, extension boards for sensors interfacing, extension boards for communication), and designed and deployed a firmware structure adapted to contain and run all use cases requirements. Additionally, BOX2M designed and deployed a Middleware software application, correlated with device firmware, and built an IoT device fleet simulator (with both hardware probes and software images of these). These developments, together with B1 and B2 implemented use cases **were essential for Hardened Encryption with crypto chip system validation** and consist of the main run live customers scenarios. B1 and B2 demonstrate the resolution for a global cyber security problem, which resides in the cost of monitoring versus security requirements for grid elements. There are some adjustments yet to be made in these use-cases, particularly towards taking advantage of ARCADIAN-IoT Trust and Recovery Planes. The remaining Domain B use cases, targeted for P2, are focused on the OTA (over the air, actually reverse traffic, command type, from IoT platform to IoT device) and data recovery after incidents (e.g. in case of sabotage, abnormal device behaviour or law enforcement requirements).

Regarding Domain C, led by RGB and scoping Medical IoT, the use cases planned for P1 (C2, C3, C4) were mostly implemented according to planned, being possible to reach a sufficient and functional implementation. The support for capturing and transmitting patient's vital signs (use case C2) was achieved via a complete implementation with all the functions described initially. It has been possible to integrate encryption and decryption functionality for medical information. In use case C3, it was possible to implement the necessary component for the analysis of medical

data and display of alerts, taking into account role-based privacy policies. In the C4 use case, we have been able to implement the necessary functions, except for the authentication, which will be resolved during the P2 implementation. The use cases planned for P2 include the support for patient registration and facing different incidents (e.g. at the patient medical IoT device or in the Cloud / web service).

## REFERENCES

- [1] ARCADIAN-IoT, “D2.2: Use case specification,” 2021.
- [2] ARCADIAN-IoT, D5.4: ARCADIAN-IoT Use Cases Validation and Legal Compliance, 2023.
- [3] ARCADIAN-IoT, “D4.2: ARCADIAN-IoT Vertical Planes - 2nd version,” 2022.