H2020 – Secure societies - Protecting freedom and security of Europe and its citizens

SU-DRS02-2018-2019-2020– Technologies for first responders – Research and Innovation Action (RIA)

Emerging technologies for the Early location of Entrapped victims under Collapsed Structures & Advanced Wearables for risk assessment and First Responders Safety in SAR operations

D3.8 Situation Awareness Model – specification V2

Workpackage:	WP3 – Situation Awareness
Authors:	UBI
Status:	Final
Due Date:	31/12/2021
Version:	1.00
Submission Date:	31/12/2021
Dissemination Level:	PU

Search and Rescue Project Profile

Grant Agreement No.:	882897
Acronym:	Search and Rescue
Title:	Emerging technologies for the Early location of Entrapped victims under Collapsed Structures & Advanced Wearables for risk assessment and First Responders Safety in SAR operations
URL:	https://search-and-rescue.eu/
Start Date:	01/07/2020
Duration:	36 months

Partners

	NATIONAL TECHNICAL UNIVERSITY OF ATHENS (NTUA) Co-ordinator	Greece
	AIDEAS OÜ (AIDEAS)	Estonia
	SOFTWARE IMAGINATION & VISION S.R.L (SIMAVI)	Romania
	MAGGIOLI SPA (MAG)	Italy
	KONNEKT-ABLE TECHNOLOGIES LIMITED (KT)	Ireland
	THALES ITAIA Italia SPA (THALIT)	Italy
	ATOS IT SOLUTIONS AND SERVICES IBERIA SL (ATOS)	Spain
	ETHNIKO KENTRO EREVNAS KAI TECHNOLOGIKIS ANAPTYXIS (CERTH)	Greece
	UNIVERSITA DEGLI STUDI DI CAGLAIRI (UNICA)	Italy

	UKEMED GLOBAL LTD (UGL)	Cyprus
	PUBLIC SAFETY COMMUNICATION EUROPE FORUM AISBL (PSCE)	Belgium
	UNIVERSITA DEGLI STUDI DI FIRENZE (UNIFI)	Italy
	DEUTSCHES FORSCHUNGSZENTRUM FUR KUNSTLICHE INTELLIGENZ (DFKI)	Germany
	UNIVERSITA CATTOLICA DEL SACRO CUORE (UCSC)	Italy
	VRIJE UNIVERSITEIT BRUSSEL	Belgium
	SYNYO GmbH (SYNYO)	Austria
	UNIVERSITEIT HASSELT (UHASSELT)	Belgium
	SPOLECZNA AKADEMIA NAUK (SAN)	Poland
	GIOUMPITEK MELETI SCHEDIASMOS YLOPOIISI KAI POLISI ERGON PLIROFORIKIS ETAIREIA PERIORISMENIS EFTHYNIS (UBITECH)	Greece
Search and Rescue End-Users		

	ELLINIKI OMADA DIASOSIS SOMATEIO (HRT)	Greece
	ENOSI PTYCHIOYCHON AXIOMATIKON YPAXIOOMATIKON PYROSVESTIR OY SOMATEIO (EPAYPS)	Greece
	JOHANNITER-UNFALL-HILFE EV (JOHAN)	Germany
	JOHANNITER OSTERREICH AUSBLIDUNG UND FORSCHUNG GEMEINNUTZIGE GMBH (JOAFG)	Austria
	CONSIGLIO NAZIONALE DELLE RICERCHE	Italy
	POMPIERS DE L'URGENCE INTERNATIONALE (PUI)	France
	ASOCIATA CLUSTERUL ROAMN RENTRU PROTECTIE SI ECOLOGIE IN DOMENIUL MATERIALELOR CHIMICE, BIOLOGICE, RADIOLOGICE/NUCLEARE SI EXPLOZIVE (PROECO)	Romania
	SERVICIO MADRILENO DE SALUD (SERMAS)	Spain
	FUNDACIÓN PARA LA INVESTIGACIÓN E INNOVACIÓN BIOSANITARIA DE ATENCIÓN PRIMARIA (FIIBAP)	Spain
	ESCUELA ESPANOLA DE SALVAMENTO Y DETECCION CON PERROS (ESDP)	Spain

Document History

Version	Date	Author (Partner)	Remarks/Changes
0.10	27/10/2021	Iosif Angelidis (UBI) Elena Politi (UBI)	Table of Contents
0.20	05/11/2021	Iosif Angelidis (UBI) Elena Politi (UBI)	Initial draft of the document
0.30	16/11/2021	Elena Politi (UBI)	Enhanced version of the initial draft
0.40	25/11/2021	Iosif Angelidis (UBI)	Final version of the draft
0.50	01/12/2021	Iosif Angelidis (UBI) Elena Politi (UBI)	Document ready for Review
0.60	07/12/2021	Simona Panunzi, Fabio Cibella(CNR)	Quality and Technical Review
0.70	10/12/2021	Elena Politi (UBI)	Version ready for Quality Control
0.80	14/12/2021	Alexandru Dumitrascu(PROECO)	STM Quality Control
0.81	14/12/2021	Elena Politi(UBI)	Version addressing comments of STM Quality Control
0.90	16/12/2021	Christodoulos Santorinaios (NTUA)	Quality Control
1.00	31/12/2021	Christos Ntanos (NTUA)	Final version to be submitted

Executive Summary

This deliverable D3.8 "Situation Awareness Model – specification V2" presents the second iteration of the methodology and the actual technical work in the frame of the Situation Awareness model. Based on the previous iteration of the model, as well as the needs that have arisen in the project, we build upon the original model, expand it and provide an efficient, high-availability deployment. Furthermore, these needs have given a new dimension to the Situation Awareness model. More specifically, the model is now also responsible for:

- Real-time monitoring of devices paired with actors during S&R operations.
- Filtering of real-time metrics of the devices against specific rules which define the safety and danger thresholds per device/value.
- Timely notification of the actors, if some of their metrics indicate a health risk.
- Detailed notification in case of alert triggers with the symptoms experienced and the action which needs to be taken.

These aspects serve as the basis upon which we expand the SA model. Finally, the results of this deliverable serve as a thorough documentation of the provided services of the SA model, enabling its usage and adoption by all relevant services and components of the project.

Table of Contents

ACRONYMS

Abbreviation	Explanation
BIM	Building Information Modelling
CCO	Common Core Ontologies
CFO	Chief Financial Officer
D	Deliverable
DO	Disease Ontology
DSS	Decision Support Systems
EDXL	Emergency Data Exchange Language
EMT	Emergency Medical Technician
EU	European Union
KM	Knowledge Management
OWL	Ontology Web Language
PS/EM	Public Safety and Emergency Management ontology
RDF	Resource Description Framework
REST	RESTful Application Program Interface (API) that uses HTTP requests to access and use data.
SA	Situation Awareness
SAM	Situation Awareness Model
SnR	Search and Rescue
WGS84	World Geodetic System 1984

List of Figures

List of Tables

Introduction

1.1 Scope and Objectives

The purpose of this document is to describe the technical activities undertaken during the S&R project execution, towards the definition and description of an enhanced version of the Situation Awareness model, which will assist in providing situation awareness services to first responders in the field. The current document namely D3.8 "Situation Awareness Model – specification v2" is the second release of the deliverable D3.2 "Situation Awareness Model – specification v1" (released in M9 and the final outcome of Task T3.2 "Situation Awareness Model".

In this final iteration of the model, the work was focused on the requirements and input that was obtained from the Search and Rescue's consortium and particularly the partners involved in use cases, regarding technologies and devices that play an important role in the field. To this end, we are presenting the enhancements and extension of work performed for the model, in terms of situation awareness realization. Moreover, the current requirements of resources and the respective information will also be documented, for the model to be defined properly.

1.2 Relationship with Other Tasks and Deliverables

The S&R Deliverable D3.8 documents the technical activities undertaken in the frame of the second iteration of Task T3.2 "Situation Awareness Model" and, more specifically, the enhancement and update of the SA model which will include an extendable set of main concepts in Situation Awareness domain, as well as assisting in general Situation Awareness. The main objective of this document is to provide a coherent version in terms of technical specification of the SA model, by taking under consideration the input from the following S&R deliverables:

- D3.2 "Situation Awareness Model – specification v1"
- D3.3 "D3.3 - BIM based services and applications - review and service design"
- D7.3 "Component interface specifications for interoperability within S&R"
- D8.2 "S&R Use Case 1: Victims trapped under rubble (Italy) – Pilot Plan"
- D8.3 "S&R Use Case 2: Plane crash, mountain rescue, non-urban (Greece) - Pilot plan"
- D8.4 "S&R Use Case 3: Earthquake - Pilot plan"
- D8.5 "S&R Use Case 4 Forest fire - Pilot plan"
- D8.6 "S&R Use Case 5 Victims trapped under rubbles (France) - Pilot plan"
- D8.7 "S&R Use Case 6 Resilience Support for Critical Infrastructures (Romania) - Pilot plan"
- D8.8 "S&R Use Case 7Chemical substances spill (Spain) - Pilot plan"

1.3 Structure of the document

To address all the aspects relevant to the scope of D3.8, the present deliverable has been structured as follows:

- Section 2 describes the methodology and procedures taken into consideration when designing the second iteration of the model.
- Section 3 delves deeper into the use cases of the project and their needs, documenting the gathered requirements for the model, regarding Knowledge Management (KM), Decision Support Systems (DSS), as well as sensors with metrics, all of which set the requirements of the updated SA model.
- Section 4 presents the architecture of the updated SA model, documenting the incremental updates needed to accommodate the new needs of the component. Moreover, the changes and adaptations of the architecture in comparison to the first iteration of the model are presented, as well as user needs and means of verification of our proposed solution. Sample APIs of the interface of the SA model are also provided here.
- Section 5 presents the model specification, the entities, object properties, data properties of the updated model, as well as its functionality with a few small examples.
- Section 6 concludes the document and discusses future work.

2 Methodology

The SnR project aims to address the challenges of interoperability in crisis management that first responders and emergency teams face during a crisis. In this respect, the SA model offers situation awareness services by providing real -time semantic correlations of data coming from the devices and technologies integrated in the S&R use cases. The definition of the Situation Awareness model will be delivered in two releases. In the first version that was delivered in the D3.2 "Situation Awareness Model – specification v1" (released in M9), the current requirements of resources and the respective information were documented, in order for the model to be defined properly. The second release of the SA model is documented in the present deliverable (M18), where further enhancements of the model with respect to situation awareness are presented. Specifically, based on the specifications and requirements of the S&R technologies and devices that were identified in the aforementioned deliverables (see Section 1.2), the model will provide real-time alerts to involved actors in the field.

2.1 Working methodology

The current document presents the methodology that was followed for the final release of the SA model, towards the definition and description of services of the proposed situation awareness solution. To this end, further enhancements of the model will be documented, based on the emerging demands for real-time situation awareness during the course of the project, while possible deviations from the original plans of the SA model will be discussed.

The methodology that has been followed for the second version of the definition of the Situation Awareness model includes the following steps:

1. Revision of input and original methodology of the first iteration of the SA model.
2. Refactoring, corrections of the existing model.
3. Processing of the new needs of the project, which have occurred during the development process.
4. Definition of new main concepts which have been identified from technical workshops in the frame of WP3, WP6 and WP7, as well as from D7.3, "Component interface specifications for interoperability within S&R", where the interoperability of devices and services of SnR is documented.
5. Construction of the new ontology which will represent the SA model by interconnecting the new concepts among them, in order for the proper information flow to be established.
6. Integration between the existing/old ontology and the new one.
7. Development of a rules engine to support real-time inference of alerting events.
8. Provision of the SA model as a RESTful service.

2.2 Semantic Model Objectives

When compared to the first iteration of the SA model, a number of new objectives have been added. Although the original objectives remain goals of the model, the new objectives define its direction and purpose. More specifically, the new objectives include:

- Scheduled gathering of metrics from devices which monitor a number of values of actors in the field.
- Processing and efficient storage of aforementioned data.
- Filtering of gathered results and continuous real-time monitoring, in case values are in the danger threshold, in which case an alert is triggered.
- Efficient notification system, which notifies users or components in real-time about values in dangerous thresholds.
- Thorough messages towards the recipient, describing their experienced symptoms as well as the immediate action to be taken.

2.3 Model Construction

As explained in the previous section, the SA model now needs to also provide general real-time situation awareness, apart from merely semantically representing correlations among entities. It is a fact that a number of situation awareness components and services already exist in the SnR ecosystem. However, an area which needed coverage is the situation awareness which is related to the continuous monitoring of devices, which keep track of critical metrics of the actors paired with them during operations. It appears an undeniable fact that, in addition to efficiently allocating resources and monitoring the positions of potential victims, it is of equal importance to monitor the status of all actors and notify them in a timely fashion, if their biometrics are in dangerous thresholds. Furthermore, the SA model aims at providing thorough situation awareness, by letting the corresponding actors know, not only the symptoms they will be experiencing, but also what action is needed in order for a potentially fatal situation to be avoided.

Similarly, to the methodology originally proposed, we once again incorporate aspects of METHONTOLOGY, a methodology to build ontologies from scratch defined by Fernandez et al.[1]. Evolving our adapted version of METHONTOLOGY from the first iteration of the SA model to meet the needs of the current iteration, the development of the semantic model consists of the following steps:

1. Specification of the high-level requirements for the semantic model based on the feedback we gathered from D7.3, "Component interface specifications for interoperability within S&R", as well as on direct partner input from a number of workshops organized in the context of WP3, WP6 and WP7.

2. Definition of the actors and entities involved in the augmented situation awareness. Section 3 provides information about this topic.

3. Definition of simple use cases. Simple use cases are used to provide a first instantiation of the workflows the model (and corresponding RESTful service) is capable to support, based on the

defined use cases of the project. Sections 3 and 5 provide some simple usage examples of the service.

4. Development of the overall conceptual model to structure the domain knowledge, adapting the original model and integrating the new model into the existing one.
5. Formalisation of the definitions for the model through their implementation in Protégé (version 5.5.0), with the main classes, object and data properties specified. Section 5 presents this information in detail.

3 Description of Situation Awareness Services

One of the greatest challenges that first responders and emergency teams face during a crisis, deals with inadequate information access, semantic differences and inconsistency of heterogeneous data sources [2]. The SnR project aims to address the challenges of interoperability in crisis management. Even more relevant, due to the hazardous nature of the use cases covered by SnR, actors are frequently at risk. This implies that the project needs a component responsible for real-time monitoring of their biometrics and, based on a sophisticated inferencing engine of dangerous thresholds rules, notifies the actors (or responsible components of the ecosystem) about their symptoms and the immediate action(s) to be taken, in order to get out of a potentially fatal situation. The SA model fits this role perfectly, as it already serves an integration purpose via the storage of semantic correlations of entities.

Since the model already has correlations of information, it makes sense to also process and filter this information against a rule's engine. Every time a specific value is in a dangerous range, the system should immediately notify the actor paired to the device from which the dangerous value originates from, ensuring their safety as much as possible.

In this section, we are going to provide the foundation upon which we build in order to properly develop and augment our original SA model. More specifically, we are going to present:

- The use cases of the project and the devices which play a central role in each, as well as their contribution in situation awareness as a whole.
- The characteristics of each device, the biometric values they monitor, their danger thresholds, the symptoms experienced in each dangerous zone and the corresponding set of actions to be taken in each.
- A table which summarizes the devices and their thresholds for all relevant values.

3.1 Search and Rescue Use Cases

The purpose of the SA model is to provide general situation awareness services by monitoring the biometrics of actors in S&R operations, as well as alerting the first responders and rescue teams about critical events happening in the field and may threaten human lives. To achieve this, certain devices of the SnR project will be utilized as providers of real-time data, that will enhance the situation awareness capabilities of the SA model implementation. It should be noted that, not all the SnR devices can support the aforementioned fuctionalities, for example the devices that serve as data consumers and/or data aggregators were deliberately overlooked in this work, and their description and specific characteristics were therefore considered out of the scope of this deliverable. The complete description of component specifications and interfaces of the devices and technologies of the SnR project has been defined in D7.3, "Component interface specifications for interoperability within S&R".

The model will play a critical role within the SnR activities. We are going to provide a brief description of each of the seven (7) Use Cases of the project, as well as the devices involved, since the SA model will be processing their data frequently (the model will support all devices in bold):

- **Use case 1**: Victims trapped under rubble in Italy. This use case is going to test a real-life crisis scenario in which victims are trapped under rubble. Devices involved:

    - **Wearable GPS tracker** (UNICA)
    - **PHYSIO DSS** (CNR)
    - Wearable ECG (UNICA)
    - Wearable EMG (UNICA)
    - Wearable Strain Sensors (UNICA)
    - AI algorithms for recognizing objects from drones' images (AIDEAS)
- **Use case 2**: Plane crash, mountain rescue, non-urban in Greece. In this use case, a forced airplane landing will be simulated in a mountainous area which is isolated without road access. Devices involved:
    - **Smartwatch** (KT)
    - **Emergency Response Health Condition Monitoring Device** (CERTH)
    - Volunteer application (CERTH)
- **Use case 3**: Earthquake and heavy storms between Vienna Rail Station and Kufstein railway station in the cross-border between Austria and Germany. The simulation involves collapsed structures such as damaged buildings and communication break downs because of lack of power supplies. Devices involved:
    - **Smartwatch** (KT)
    - **Six Gas Hazmat Monitor** (UNICA)
    - Obstacle Detection and Avoidance System, ODAS (THALIT)
    - Rescue robots and autonomous vehicles (DFKI)
- **Use case 4**: Two-pronged threat between a forest fire which expands rapidly and a threat in a nearby industrial zone in the Attica region of Greece. Devices involved:
    - **Smartwatch** (KT)
    - Radiation Sensors, wearable (UNICA)
    - Chemical sensors – Rescue MIMS (NTUA)
    - Drones (UHASSELT)
    - Collaborative drones' platform (UHASSELT)
    - Rescue robots and autonomous vehicles (DFKI)
    - Obstacle Detection and Avoidance System, ODAS (THALIT)
- **Use case 5**: Victims trapped under rubbles in France. Devices involved:
    - **PHYSIO DSS** (CNR)
    - **COncORDE** (KT)
    - Chemical sensors – Rescue MIMS (NTUA)
- **Use case 6**: Resilience Support for Critical Infrastructures through Standardized Training on CBRNE (chemical, biological, radiological/nuclear, explosive) in Romania. In this scenario, specialized staff will undergo standardized training against CBRNE hazards management. Devices involved:
    - **Six Gas Hazmat Monitor** (UNICA)
    - Wearable Strain Sensors (UNICA)
    - Smart Glasses (SIMAVI)
- **Use case 7**: Chemical substance spill in Spain, after an accident in a factory that derives in chemical spill in a building, threatening the health of the workers of the factory. The Use case

involves real life simulation of S&R of victims in chemical risk emergency situation. Devices involved:

- **Wearable GPS tracker** (UNICA)
- **Smartwatch** (KT)
- **Six Gas Hazmat Monitor** (UNICA)

Based on the summary of devices per use case above, there are a lot of devices which do not feed the SA model for situation awareness. However, it should be noted that this is by no means a shortcoming of the model itself. Most of these devices are waiting for input themselves, do not offer real-time data (which defeats the purpose of real-time situation awareness the SA model is trying to realize), or monitor aspects of operations which are not directly related to biometrics of actors. The supported devices and their involvement in use cases is summarized in the table below (Table 1).

Table 1: Devices and use cases correlation.

	Use Case 1	Use Case 2	Use Case 3	Use Case 4	Use Case 5	Use Case 6	Use Case 7
COncORDE (KT)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Emergency Response Health Condition Monitoring Device	No	Yes	No	No	No	No	No
PHYSIO DSS (CNR)	Yes	No	No	No	Yes	No	No
Six Gas Hazmat Monitor (UNICA)	No	No	Yes	No	No	Yes	Yes
Smartwatch (KT)	No	Yes	Yes	Yes	Yes	No	Yes
Wearable GPS tracker (UNICA)	Yes	No	No	No	No	No	Yes

3.2 Characteristics of situation awareness devices

As explained in the previous section, since the SA model offers real-time data representation, it is essential to highlight that only some of the SnR technologies can support the situation awareness capabilities, thus only these will be considered in this work. The technologies which serve as data consumers and/or data aggregators do not necessarily comply with the capabilities of the SA model and, as such, cannot support the related functionalities.

As noted in previous sections, the complete description of component specifications and interfaces of the devices and technologies which participate in the SnR project has been defined in D7.3, "Component interface specifications for interoperability within S&R". In the next subsections, we are going to present the main devices with which the SA model will be communicating in order to realize general situation awareness.

3.2.1 Smartwatch

The Smartwatch is a wearable device which enables heart rate monitoring of the person who wears it. While it is worn by first responders, it also allows them to take measurements of any person on the field, by simply placing the wristband on the victim's wrist. The main value of importance here for situation awareness is the heart rate, measured in beats per minute (bpm) and the body temperature measured in Celsius degrees. The device communicates with the SA model through a REST API service[1], so that the model can trigger specific alarms in case of a risk measurement. In such terms, we distinguish the following rules/danger thresholds:

- Heart rate (measured in bpm)
    - 40-50 bpm: The actor is experiencing chest pain, confusion or memory problems, dizziness or light-headedness, fatigue, shortness of breath, chest tightness or pain, dizziness or light-headedness, fainting. They need to immediately receive the message "Bradycardia: please request for medical support immediately".
    - 100-190 bpm: The actor is experiencing shortness of breath, light-headedness, rapid pulse rate, chest pain, fainting, heart palpitations (a racing, uncomfortable or irregular heartbeat or a sensation of "flopping" in the chest). They need to immediately receive the message "Tachycardia: please request for medical support immediately".
- Body temperature (measured in Celsius degrees)
    - 30.0-35.0°C: The actor is experiencing shivering, slurred speech or mumbling, slow and shallow breathing, weak pulse, clumsiness and lack of coordination, drowsiness or very low energy, confusion or memory loss, loss of consciousness. They need to immediately receive the message "Hypothermia: please request for medical support immediately".
    - 38.5-39.9°C: The actor is experiencing sweating, chills and shivering, headache, muscle aches, loss of appetite, irritability, dehydration, general weakness. They need to immediately receive the message "Fever/Hyperthermia: please request for medical support immediately".
    - 40-42°C: The actor is experiencing extreme sweating, extreme chills and shivering, extreme headache, extreme muscle aches, loss of appetite, extreme irritability, extreme dehydration, extreme general weakness. They need to immediately receive the message "Hyperpyrexia: please request for medical support immediately".

[1] Or through the aggregation pipeline where data are gathered in the Data Lake, aggregated and stored in COncORDE. As a matter of fact, the SA model will communicate with each device either directly via REST, or indirectly by obtaining aggregated data through COncORDE as needed.

3.2.2 Emergency Response Health Condition Monitoring Device

The Emergency Response Health Condition Monitoring Device is equipped with sensors which can measure critical vital signs to be used both in first responders and victims. In specific, it can measure heart rate, respiration rate, blood oxygen levels, and body temperature. The rescuer will be able to read the victims condition on a smartphone and transmit the position of the victim. Moreover, the device will provide the measured data through a REST API service to the SA model, so that the appropriate alarms can be triggered. We distinguish the following rules/danger thresholds:

- Heart rate (measured in bpm)
  - 40-50 bpm: The actor is experiencing chest pain, confusion or memory problems, dizziness or light-headedness, fatigue, shortness of breath, fainting. They need to immediately receive the message "Bradycardia: please request for medical support immediately".
  - 100-190 bpm: The actor is experiencing shortness of breath, light-headedness, rapid pulse rate, chest pain, fainting, heart palpitations (a racing, uncomfortable or irregular heartbeat or a sensation of "flopping" in the chest). They need to immediately receive the message "Tachycardia: please request for medical support immediately".
- Respiration rate (measured in bpm)
  - 8-12 bpm: The actor is experiencing a slow respiratory rate, possibly associated with mental confusion and lethargy. They need to immediately receive the message "Bradypnoea: please request for medical support immediately".
  - > 20 bpm: The actor is experiencing a fast respiratory rate, possibly due to severe alteration in pulmonary gas exchange. They need to immediately receive the message "Tachypnoea: please request for medical support immediately".
- Blood oxygen level (SpO2, measured as a percentage)
  - 93%-94%: The actor has concerning blood oxygen levels. They need to immediately receive the message "Concerning blood oxygen levels".
  - 50%-92%: The actor has low blood oxygen levels, indicating a severe alteration in pulmonary gas exchange. They need to immediately receive the message "Very low blood oxygen levels, please request for medical support immediately".
- Body temperature (measured in Celsius degrees)
  - 30.0-35.0°C: The actor is experiencing shivering, slurred speech or mumbling, slow and shallow breathing, weak pulse, clumsiness and lack of coordination, drowsiness or very low energy, confusion or memory loss, loss of consciousness. They need to immediately receive the message "Hypothermia: please request for medical support immediately".
  - 38.5-39.9°C: The actor is experiencing sweating, chills and shivering, headache, muscle aches, loss of appetite, irritability, dehydration, general weakness. They need to immediately receive the message "Fever/Hyperthermia: please request for medical support immediately".
  - 40-42°C: The actor is experiencing extreme sweating, extreme chills and shivering, extreme headache, extreme muscle aches, loss of appetite, extreme irritability, extreme dehydration, extreme general weakness. They need to immediately

receive the message "Hyperpyrexia: please request for medical support immediately".

3.2.3 Six Gas HazMat Monitor

The Six Gas HazMat Monitor is a portable single-unit gas monitor, which can be worn in a jacket or trouser pocket and can detect toxic or flammable gases that may be produced in the emergency area. The monitor can show real-time values of gas emissions in a multiple numerical display and can provide the measured data through a REST API service to the SA model, so that the appropriate alarms can be triggered. Important values monitored here include exposure to CO, CO_2 and CH_4. We distinguish the following rules/danger thresholds:

- CO measured in parts per minute (ppm)
    - 35-50 ppm: The actor is possibly experiencing mild headache. They need to immediately receive the message "Minor CO exposure detected. Try to avoid exposure as much as possible".
    - 50-80 ppm: The actor is experiencing mild headache, dyspnoea on exertion, cutaneous vasodilatation. They need to immediately receive the message "Mild CO exposure detected. Try to avoid exposure as much as possible".
    - 80-100 ppm (for 1-2 hours): The actor is experiencing irregular cardiac beats. They need to immediately receive the message "CO exposure detected. Try to avoid exposure as much as possible".
    - 100-200 ppm: The actor is experiencing headache, nausea, confusion. They need to immediately receive the message "Major CO exposure detected. Try to avoid exposure as much as possible".
    - 200-300 ppm: The actor is experiencing severe headache, irritability, fatigue, blurred vision, fatigue, blurred vision. They need to immediately receive the message "Severe CO exposure detected. Please request for medical support as soon as possible".
    - 300-500 ppm: The actor is experiencing headache, tachycardia, nausea, confusion, lethargy, collapse, disturbance of breathing rhythm. They need to immediately receive the message "Critical CO exposure detected. Please request for medical support immediately".
    - 500-600 ppm: The actor is experiencing the same symptoms as above, however if the value remains within this region after 2 hours the actor is in serious danger and in 2.5 hours, they will be dead. Both the actor and the command centre should be notified immediately with the message "Very critical CO exposure detected. Risk of danger if under exposure of over 2 hours. Risk of death if under exposure of over 2.5 hours".
    - 700-1200 ppm: The actor is experiencing coma, convulsions, respiratory and cardiac failure. At this point, the actors are unable to take action themselves, so the command centre should be notified immediately with the message "Critical

emergency. Prioritize medical support for this victim as they are reaching death status".
    - > 1900 ppm: The actor has possibly died. The command centre should be notified immediately with the message "Warning: Individual possibly dead".

- CO_2 measured in parts per minute (ppm)
    - 5.000-6.000 ppm: The actor is experiencing average exposure, which will be lethal after an 8-hour period. They need to immediately receive the message "Average exposure limit over an eight-hour period detected".
    - 6.000-30.000 ppm: The actor is under concerning exposure and should not stay for long. They need to immediately receive the message "Concerning exposure detected, please do not stay here longer than you absolutely have to".
- O_2 measured as percentage
    - < 18%: The actor is experiencing muscle coordination difficulties and increased respiratory rate. The command centre should be notified immediately with the message "Limited oxygen available in the area. Please move to an area with more oxygen".
- CH_4 measured in parts per minute (ppm)
    - 50.000-150.000 ppm: The actor is within a room under danger of exploding. They need to immediately receive the message "Potentially explosive room detected, please evacuate the room as soon as possible".
    - 500.000 ppm: The actor is experiencing asphyxia. The individual and the command centre need to immediately receive the message "Individual is experiencing asphyxia, please request for medical support immediately".
- H_2S measured in parts per minute (ppm)
    - <= 10 ppm for more than 10 minutes: The actor is experiencing irritation of the eyes, nose and throat, breath difficulties if having asthma. They need to immediately receive the message "Try to avoid exposure to H_2S".
    - > 500 ppm: The actor is sudden loss of consciousness. Both the actor and the command centre need to immediately receive the message "This individual is about to lose consciousness, please request for medical support immediately".

3.2.4 Physio DSS

The Physio DSS is responsible for providing modules, functions and algorithms, for the prediction of the evolution of the physiological status of the victims. Within the context of the SA model, the model will regularly gather data from Physio DSS and will describe a number of critical decisions (some of which trigger an alarm) through a set of Boolean and integer conditions. More specifically, we distinguish the following rules/danger thresholds:

- Glasgow Coma Scale: if this value is < 8, the command centre should immediately receive the message "Immediate intervention is required (Glasgow Coma Scale is < 8)".
- Jump start triage:

    - Black (0): if this value is set to 0, the command centre should immediately receive the message "Black code detected: Dead individual event emitted".
    - Red (1): if this value is set to 1, the command centre should immediately receive the message "Red code detected: Immediate intervention is required".
    - Yellow (2): if this value is set to 2, the command centre should immediately receive the message "Yellow code detected: Delayed event emitted".
- Start triage: if this value is set to true, the command centre should immediately receive the message "Immediate intervention is required (start triage condition met)".
- Sort triage: if this value is set to true, the command centre should immediately receive the message "Immediate intervention is required (sort triage condition met)".
- Sieve triage: if this value is set to true, the command centre should immediately receive the message "Immediate intervention is required (sieve triage condition met)".
- ETD pie (Estimated Time to Death): if the average value is less than 1 hour, the command centre should immediately receive the message "Estimated time to death of individual is less than an hour. Please send medical support as soon as possible".

3.2.5 GPS Tracker

The GPS Tracker is a module which can be embedded in a smartphone device in order to be used for the geolocation of the first responder. It will collect data such as position, timestamp, altitude, speed and bearing. In terms of situation awareness, the SA model will monitor the values of the device and, more specifically the speed of individuals paired with GPS trackers. If a speed of 0 is detected, both the actor and the command centre should immediately receive the message "Rescue dog and/or handler has stopped moving. Possible victim found in the emergency scene".

3.2.6 Summary of devices and alert triggers

The biometric values of importance per device, as well as their danger thresholds, symptoms and actions to be taken are summarized in the following table (Table 2).

Table 2: Devices with real-time metrics, danger thresholds, symptoms and actions needed.

Technology/ device	Physical quantity	Alarming values	Symptoms	Notification action needed
Emergency Response Health Condition Monitoring Device	Heart rate (bpm)	40-50	chest pain, confusion or memory problems, dizziness or light-headedness, fatigue, shortness of breath, fainting	Bradycardia: please request for medical support immediately
		100-190	shortness of breath, light-headedness, rapid pulse rate, chest pain, fainting, heart palpitations (a racing,	Tachycardia: please request for medical support immediately

			uncomfortable or irregular heartbeat or a sensation of "flopping" in the chest)	
	Respiration Rate (bpm)	8-12	slow respiratory rate	Bradypnoea: please request for medical support immediately
		20-50	fast respiratory rate	Tachypnoea: please request for medical support immediately
	Blood Oxygen Level (SpO2 %)	93-94	concerning blood oxygen levels	Concerning blood oxygen levels
		50-92	low blood oxygen levels	Low blood oxygen levels, please request for medical support immediately
	Body temperature (°C)	30.0-35.0	shivering, slurred speech or mumbling, slow and shallow breathing, weak pulse, clumsiness and lack of coordination, drowsiness or very low energy, confusion or memory loss, loss of consciousness	Hypothermia: please request for medical support immediately
		38.5 or 39.9	sweating, chills and shivering, headache, muscle aches, loss of appetite, irritability, dehydration, general weakness	Fever/Hypertherm ia: please request for medical support immediately
		40.0- 42.0	extreme sweating, extreme chills and shivering, extreme headache, extreme muscle aches, loss of appetite, extreme irritability, extreme dehydration, extreme general weakness	Hyperpyrexia: please request for medical support immediately
Smartwatch	Heart rate (bpm)	40-50	chest pain, confusion or memory problems, dizziness or light-headedness, fatigue, shortness of breath, fainting	Bradycardia: please request for medical support immediately
		100-190	shortness of breath, light-headedness,	Tachycardia: Please request for

			rapid pulse rate, chest pain, fainting, heart palpitations (a racing, uncomfortable or irregular heartbeat or a sensation of "flopping" in the chest)	medical support immediately
	Body temperature (°C)	30.0-35.0	shivering, slurred speech or mumbling, slow and shallow breathing, weak pulse, clumsiness and lack of coordination, drowsiness or very low energy, confusion or memory loss, loss of consciousness	Hypothermia: please request for medical support immediately
		38.5-39.9	sweating, chills and shivering, headache, muscle aches, loss of appetite, irritability, dehydration, general weakness	Fever/Hyperthermia: please request for medical support immediately
		40.0-42.0	extreme sweating, extreme chills and shivering, extreme headache, extreme muscle aches, loss of appetite, extreme irritability, extreme dehydration, extreme general weakness	Hyperpyrexia: please request for medical support immediately
Six Gas HazMat Monitor	CO (ppm)	35-50	mild headache	Minor CO exposure detected. Try to avoid exposure as much as possible
		50-80	mild headache, dyspnoea on exertion, cutaneous vasodilatation	Mild CO exposure detected. Try to avoid exposure as much as possible
		80-100 exposure for 1-2 hours	irregular cardiac beats	CO exposure detected. Try to avoid exposure as much as possible.
		100-200	headache, nausea, confusion	Major CO exposure detected. Try to avoid exposure as much as possible

		200-300	severe headache, irritability, fatigue, blurred vision, fatigue, blurred vision	Severe CO exposure detected. Please request for medical support as soon as possible
		300-500	headache, tachycardia, nausea, confusion, lethargy, collapse, disturbance of breathing rhythm	Critical CO exposure detected. Please request for medical support immediately
		500-600	same symptoms as above, however if the value remains within this region after 2 hours the actor is in serious danger and in 2.5 hours they will be dead	Very critical CO exposure detected. Risk of danger if under exposure of over 2 hours. Risk of death if under exposure of over 2.5 hours
		700-1200	coma, convulsions, respiratory and cardiac failure	Critical emergency. Prioritize medical support for this victim as they are reaching death status
		> 1900	death	Warning: Individual possibly dead
	CO_2 (ppm)	5.000-6.000	average exposure limit over an 8-hour period	Average exposure limit over an eight-hour period detected
		6.000-30.000	professional exposure limit for 8h exposures	Concerning exposure detected, please do not stay here longer than you absolutely have to
	O_2 (percentage)	< 18%	muscle coordination difficulties and increased respiratory rate	Limited oxygen available in the area. Please move

				to an area with more oxygen
	CH_4 (ppm)	50.000-150.000	potentially explosive	Potentially explosive room detected, please evacuate the room as soon as possible
		500.000	asphyxia	Individual is experiencing asphyxia, please request for medical support immediately
	H_2S (ppm)	<= 10 ppm for more than 10 minutes	irritation of the eyes, nose and throat, breath difficulties if having asthma	Try to avoid exposure to H_2S
		> 500	sudden loss of consciousness	This individual is about to lose consciousness, please request for medical support immediately
Physio DSS	Glasgow_Coma_Scale_computation (double)	< 8	critical Glasgow Coma Scale detected	Request medical support immediately
	Jump_Start_Triage (integer)	0, 1, 2	black (death), red (immediate intervention) or yellow code (delayed event) received	Black code detected: Dead individual event emitted| Red code detected: Immediate intervention is required| Yellow code detected: Delayed event emitted
	Start Triage (boolean)	true	start triage condition met	Immediate intervention is required (start triage condition met)
	Sort_Triage (boolean)	true	sort triage condition met	Immediate intervention is required (sort triage condition met)
	Sieve_Triage (boolean)	true	sieve triage condition met	Immediate intervention is required (sieve

				triage condition met)
	ETD_PIE (long)	< 1hour	critical expected time of death threshold	Average estimated time to death of individual is less than an hour. Please send medical support as soon as possible
Wearable GPS tracker	speed	0	No speed for dog or dog handle	Rescue dog and/or handler has stopped moving. Possible victim found in the emergency scene

4 Situation Awareness model as a service and architecture

In this section, we are going to present the proposed architecture of the SA model. More specifically, we are going to illustrate a high-level view of the architecture, describe the internal components in detail, the technological stack required for the realization of the solution, as well as the deployment process. Furthermore, we are going to discuss the incremental updates and adaptations which were necessary in order to incorporate the newest features of the model into the old implementation. Finally, we are going to discuss the required user needs and the means of verification of the model in its current iteration.

4.1 High level view

First of all, it is necessary to provide the operation context, through which the SA model functions. However, before describing the SA model itself, we need to describe the integration and interaction points with other parts of the SnR ecosystem:

- **Devices**. The devices serve as sensors and endpoints (e.g., Physio DSS) which process data and produce information which needs to be processed, forwarded and/or stored. The devices of relevance here are the Smartwatch, the Six Gas Hazmat Monitor, the Physio DSS, the wearable GPS tracker and the Emergency Response Health Condition Monitoring device. These devices interact with the ecosystem through RESTful APIs and either offer data passively, or actively forward them to necessary components in general.
- **Data aggregation pipeline**. This pipeline refers to the general processing of sensor data throughout the SnR ecosystem. Data originating from sensors and historical sources are gathered into the Data lake for coherence and maintenance, while the data are then undergoing aggregation operations, resulting in their storage in aggregated form inside COncORDE. Depending on the needs of the rules engine of the SA model, data from the relevant devices are obtained either directly from the sensors, or indirectly via their aggregated equivalents in COncORDE.
- **Actor or component**. This is a placeholder indicating any actor or component of the entire ecosystem with the potential of querying the SA model for data and information, or a general entity which the SA model is responsible for notifying in a timely fashion in case of an alert trigger. In the vast majority of cases, this placeholder refers to either the actors who are paired with devices which trigger events in dangerous thresholds, or the command centre itself, because there are certain cases where the endangered actor is unable to get out of danger themselves.

Generally speaking, all external integration and interaction points of the SA model interact with it in one of the following ways:

- **Sending data to the SA model or the SA model itself pulling their data through REST**. The devices themselves and the interaction with COncORDE DB through the aggregation pipeline interact with the SA model this way.
- **Querying the SA model for specific information**. Actors who are involved in an alert trigger interact with the SA model this way. However, it makes more sense for actual

components which control S&R operations (e.g., the command centre) to use this kind of interaction with the SA model.

- **The SA model pushes alert messages/notifications to involved actors and components**. The SA model pushes important alerts and messages to actors of relevance and important components of the ecosystem such as the command centre.

Having established the interaction points of the SA model, we can now safely proceed with the presentation of its inner modules, which are going to be briefly mentioned here, but thoroughly documented in the following subsections:

- **Semantic database**. The semantic database of the SA model, which is responsible for keeping all stored data for future reference and for queries. In order to exploit semantic context of the stored information, RDF4J is utilized as a triple/RDF store. Additionally, in cases where performance is crucial, or the queries asked against the service require complex computations, ElasticSearch will be used as an additional database, due to its ability to better handle such cases.
- **Scheduler**. The scheduler is responsible for periodically communicating with the devices participating in general situation awareness, in order to filter the biometrics of subscribed actors in case of an alert trigger event. Due to the importance of the scheduler, and to avoid stale content, the scheduler adjusts the time period per check dynamically.
- **Rules engine**. The rules engine is responsible for monitoring the values of the data obtained by the scheduler against a set of sophisticated rules, in order to determine whether one or more values trigger an alert. If so, a proper message depending on the alert trigger is generated, stored for historical data and is then sent to the relevant actors.
- **Push notification service**. The service of the SA model responsible for properly notifying relevant actors in cases of alert triggers. It keeps track of subscribed actors and components per actor paired with devices and notifies them in a timely fashion. Messages are kept in an event queue to ensure they are never lost in cases of failures.
- **RESTful interface**. The RESTful layer of the service. Following the REST principles in a strict fashion, this interface supports POST/GET operations. The data format for all exchanges is JSON.

The overall high-level architecture of the SA model is shown in the figure below (Figure 2), including a memo for clarifications regarding the types of interactions (Figure 1).

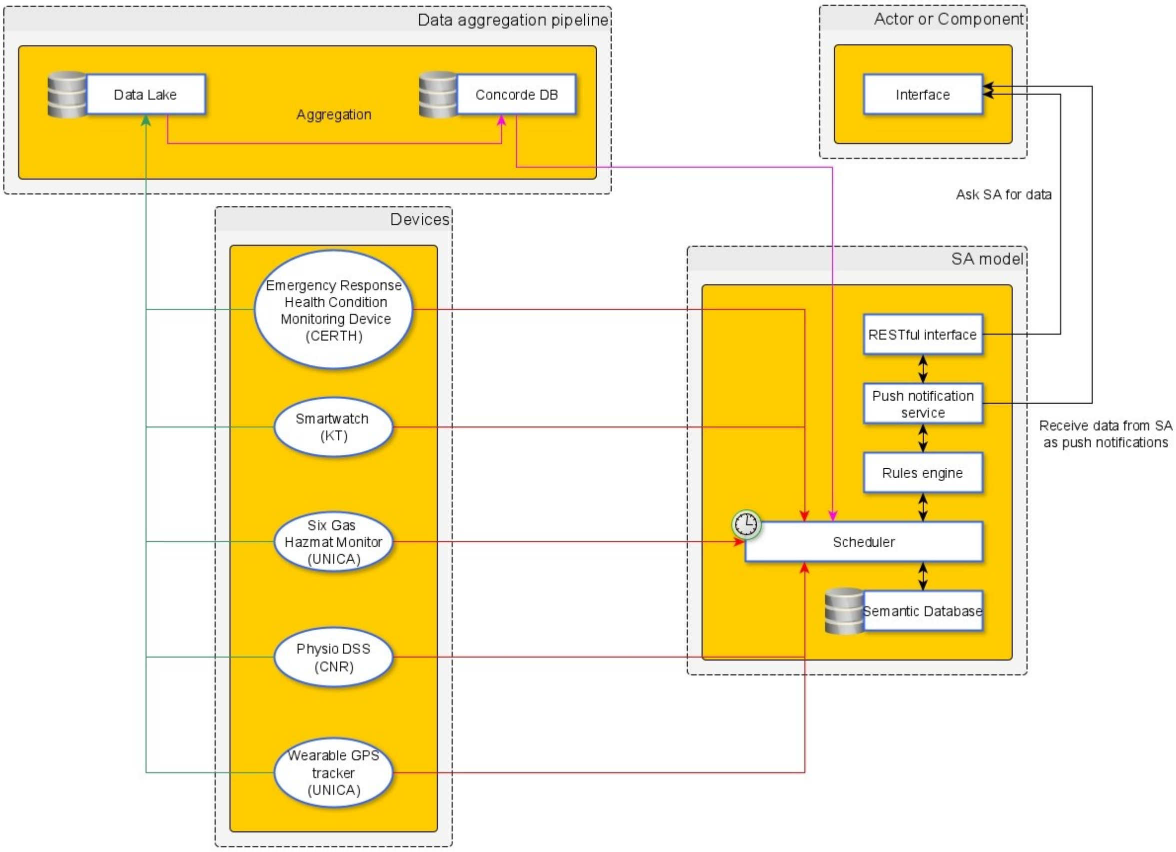

Figure 1: High level architecture of the SA model.

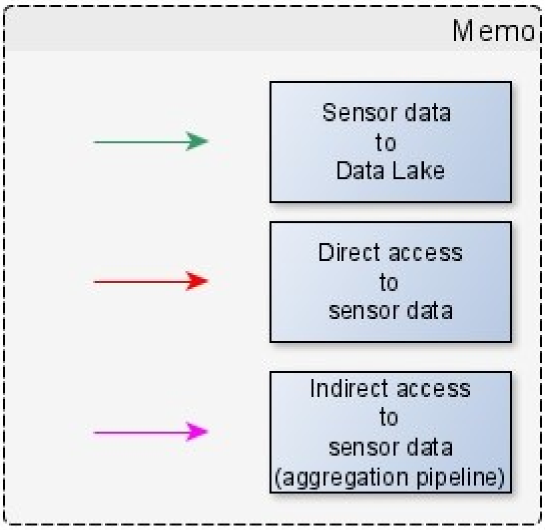

Figure 2: High level architecture of the SA model memo.

4.1.1 Core entities

Due to the need for situation awareness support, we need to properly and formally define the necessary core entities of the solution. These entities are going to be semantically represented and correlated with each other through specific properties (the exact specification of which is provided in Section 5). The goal of the definition of the core entities is to allow maximal flexibility and potential for extensions in the future as needed, while also ensuring the complexity and cardinality of the entities themselves is minimal. To that end, we have identified the following core entities:

- **Actor**: An entity which is an actor can be anyone or anything participating in the SnR project and can be uniquely identified. Entities which can be considered actors are: first responders, drones, robots, the command centre, a component of the SnR ecosystem. In the general case, an actor is paired with one or more devices, a fact which correlates an actor with the specific biometrics observed in the devices they are paired with.
- **Alert trigger**: An alert event. The SA model will be closely monitoring the devices paired with each actor and the status of the devices themselves will be filtered against the rules engine of the model. If dangerous thresholds are detected and an alert condition is triggered, the system will generate the corresponding alert, correlating it with the actor it is meant to reach as well as the command centre itself in particularly deadly situations where the actor is unable to get out of danger by themselves (e.g., a first responder faints). Naturally, these alerts will also be stored into the model's database for future reference and as historical data.
- **Device**: As already explained, the SA model will support general situation awareness by communicating with a number of devices, all of which have been identified in detail in Section 3. The general entity device serves as the root/template of all subsequent devices (Smartwatch, GPS tracker etc.) by defining the specification all of them need to conform to as a common basis. Subcategories of the general device entity include the Smartwatch, the Six Gas Hazmat Monitor, the Physio DSS, the wearable GPS tracker and the Emergency Response Health Condition Monitoring Device.

4.1.2 Technologies utilized

The semantic model is provided as a service through a RESTful API. The technical details of our proposed solution are as follows:

- The implementation is written in Java and, more specifically, the Spring Boot ecosystem.
- All provided APIs are documented via OpenAPI (previously known as Swagger). Swagger also lets developers experiment with specific calls, inspect the specification of required parameters and request structures etc.
- No frontend is required for the service itself.
- Caching is used where possible, to improve performance on hot queries.
- Scheduling libraries are utilized to monitor the devices supported, as well as for periodic communication with COncORDE.
- Once an alert event is detected, sophisticated messages are sent to components directly, or through Android push notifications.
- RDF4J is used as the database/technology of choice for storing semantic data. However, depending on the strict needs for real-time monitoring, alerting in a timely fashion and nature

and overall complexity of the queries posed against the service itself, ElasticSearch will be utilized instead.

- Alert events are detected by a sophisticated rules engine (custom implementation adapted to the needs of the project).
- The entire solution is provided as a set of docker containers, which can be deployed as a singular ecosystem via docker-compose.

4.1.3 Deployment

In order to stay as close to the original specification of the model as possible, as well as ensuring an easy yet efficient deployment, our entire solution is deployable via a set of containers. These containers include:

- **Semantic database**: The container of the database of the service. A container with an RDF4J deployment with a Tomcat instance will be provided for the semantic database. This container also provides a frontend for interacting with the semantic database through a WAR file deployed in Tomcat.
- **Alternative database**: The container of the alternative database of ElasticSearch, which might be necessary in cases of strict time constraints and performance.
- **Kibana**: Kibana is a well-known frontend for ElasticSearch databases. It is production-ready and utilized widely in the industry. It offers a number of real-time analytics, health status monitoring, read-write performance, advanced queries against the database etc.
- **Backend backbone**: A container including the full Java implementation. The corresponding Spring Boot implementation, the scheduler, the rules engine, the push notification service, the RESTful interface and the OpenAPI specification will all be deployed inside this container. Although this is done in order to reduce the number of containers deployed, these modules are strictly separated and well-defined internally.

Although containers make deployment of the solution much easier, less error prone and much more environment agnostic, they still require a number of steps in order to successfully deploy each one and then connect them through ports/networks. In order to simplify this process, we additionally offer a docker-compose file, which takes care of all of these details, effectively making the whole procedure much easier. In fact, all one needs to do in order to bootstrap the entire solution is merely running **docker-compose up -d**.

Moreover, we also make sure to ensure data persistence through the deployment. Docker containers eliminate a number of problems when deploying a solution in multiple environments etc., but it is also crucial to ensure data persist after failures, container restarts etc. Imagine for example a scenario where a runtime exception occurred in the backbone solution which would force it to stop. In that case, a restart without persistence could very easily delete all the data stored in the database(s), which is unacceptable, especially given the nature of the situation awareness (monitoring biometrics in real-time and ensuring first responders and/or the command centre are notified in a timely fashion to prevent deaths).

4.1.4 Unit testing

An often ignored, but very important nevertheless aspect of software development, is the inclusion of unit tests during the development process, as well as the bootstrapping of the deployed solution in order to ensure correct health status. In order to address this aspect, we have written proper unit tests for each provided API interface (at least 2) of the model. The tests involve the validation of both correct input and incorrect input, with the aim of ensuring that even in cases of problematic data, the system will be able to process them properly without risking runtime errors etc.

Additionally, mock-up services were set up during the development process, in order to simulate the production of data from devices and their timely consumption and processing by our system. The mockups designed uphold the principles and specifications of the interfaces of their corresponding real counterparts.

Furthermore, we utilize these mock-up services in a category of units testing which deals with stress testing. We simulate harsh conditions such as limited bandwidth, plenty of data arriving at the same time etc. The purpose of this type of unit testing is to ensure a high quality of performance of the service.

Finally, another couple of minimal unit tests serve the purpose of a family of tests to ensure correct health status during bootstrap time. Simply put, they ensure the smooth functionality of the system in the deployed environment, only allowing it to continue operating if it can uphold the quality standard set by this family of tests.

4.2 RESTful interface

The RESTful interface is the topmost module of the system and the one communicating with all external components of the SnR ecosystem. REST is chosen due to its compatibility with the other components, as well as due to its well-established specification and wide usage in general.

Conforming to the aforementioned specifications, each resource of our system can be interacted with, through POST, GET, PUT, DELETE requests. However, apart from the base resources representing the entities of the semantic model, we also provide interfaces and operations on resources which result from the core entities. Such examples include aggregated results, results filtered by time ranges etc.

Furthermore, all data sent to our system, as well as data returned by it, will be in JSON. Due to the nature of the interactions with the SA model, it can be considered as a data producer as well as a data consumer. Generally speaking, the provided interface offers interactions of the following types:

- Raw data submission. Requests sending raw data into our system for processing and storage. This set of interfaces covers devices which actively forward their data to the SA model.
- Raw data retrieval. Requests asking the SA model for raw data, which are already stored as historical data. These requests cover cases where data from core entities are required as is.
- Aggregated/filtered data retrieval. Requests asking the SA model for aggregated data from various raw types for specific needs. For example, such requests can include the filtering of a specific time range for which aggregated data for a device are needed, data for specific biometrics, or alerts of a specific type generated for a specific actor.

All provided APIs are thoroughly documented via OpenAPI (previously known as Swagger). However, for the sake of completeness, we are going to provide a representative sample for each of the three types of interactions mentioned above. These samples are presented in the table below (Table 3):

Table 3: Sample APIs of the RESTful interface

Interface type	Endpoint	Parameters	Body	Response	Example
Raw data submission	**POST** /smartwatchData	-	{timestamp: 24353, smartwatchBodyTemperatureCelsius: 36.0, smartwatchHeartRateBPM: 90}	Request status (200 for successful incorporation to the triple store, or error code in case of an error)	**POST** /smartwatchData
Raw data retrieval	**GET** /smartwatchLatestData	deviceId	-	{timestamp:24353, smartwatchBodyTemperatureCelsius: 36.0, smartwatchHeartRateBPM: 90}	**GET** / smartwatchLatestData?deviceId=sw1
Aggregated/filtered data retrieval	**GET** /smartwatchData	deviceId, dateStart (optional), dateEnd (optional)	-	[{timestamp:123456, smartwatchBodyTemperatureCelsius: 36.0, smartwatchHeartRateBPM: 90}, {timestamp: 123457, smartwatchBodyTemperatureCelsius: 36.0, smartwatchHeartRateBPM: 90}]	**GET** / smartwatchData?deviceId=sw1&dateStart=123456&dateEnd=123458

4.3 Push notification service

The push notification service is responsible for the timely forwarding of detected alert events and corresponding messages to relevant actors and/or components. In order to achieve this, actors and components of interest need to subscribe to specific devices, actors or biometrics values, for which they wish to be notified. Alternatively, this can also be achieved by external components communicating with the SA model to query on alert triggers for a specific device, actor or actor-device pair. Each alert contains two vital pieces of information:

- **Experienced symptoms**. The symptoms the actor is currently experiencing. This helps them understand the condition they are in and what they should do to get out of danger.
- **Action message**. This message contains detailed instructions about what the actor needs to do in order to ensure their safety. In cases where their vitals render them unable to act on

their own (e.g., the actor faints), the command centre needs to be notified with this information, in order to send help to the victim.

All alerts to be pushed are timestamped with the time the alert was generated and this time will be represented as a UNIX timestamp (long). The reason for this is to avoid timezones conversion among components and services.

4.4 Rules engine

The rules engine plays a central role in the realisation of proper general situation awareness. After the data retrieved from the devices are properly processed, they are aggregated per device and/or per actor, feeding the rules engine. Then, the rules engine queries the historical data already stored in the Semantic Database in order to begin checking the biometrics of an actor against the predefined set of sophisticated rules.

The historical data retrieval is a crucial part of the process, because certain rules trigger an alert only if a biometric value is observed to be in dangerous thresholds for a certain amount of time. Each biometric value is carefully checked, both independently and in correlation with the rest and all conditions which trigger alerts are identified. In case contradictory conditions are met, the one having the most priority (closest match reflecting the situation at hand) will be kept. Finally, the generated alerts are stored for future reference and are also forwarded to the relevant actors and/or components by the push notification service.

4.5 Scheduler

The scheduler is responsible for the continuous data retrieval from the devices involved. It is configured to retrieve data periodically in default time intervals in the general case. However, once a critical situation takes place and closer monitoring of the actors' vitals is required, the time intervals can be dynamically tweaked as needed. More importantly, the scheduler offers maximum flexibility in terms of dynamic monitoring, because it is capable of fine-grained scheduling at the following levels:

- Actor level. At this level of monitoring, the scheduler has separate time intervals for each actor. This is useful when having different kinds of first responders in the field. Some of them might be in more dangerous situations and, as such, need closer monitoring to ensure their safety.
- Role of actor level. This is similar to the above, the main difference being that this level of monitoring allows scheduling intervals per role. Its usage is almost equivalent to the actor level, albeit a bit more abstract.
- Device level. At this level of monitoring, each unique device has a separate time interval scheduled.
- Device type. This is similar to the device level, although instead of scheduling for each unique device, scheduling is now at a device type level. This is useful when some devices check more crucial vitals than others in certain situations. For example, when entering a room with hazardous gases, the Six gas hazmat monitor device needs closer inspection compared to the smartwatch.

4.6 Semantic Database

All data gathered throughout the uptime of the system will be stored into the semantic database(s). Due to the semantic capabilities of the RDF specification, the main database of choice will be based on

RDF4J, as it was the case in the first iteration of the SA model. Although RDF is very verbose, the semantic and inferencing capabilities it offers are significant, especially when trying to identify latent correlations among pieces of information. However, the verbosity of RDF itself comes with several shortcomings:

- Having so much verbosity means more bandwidth is required by the network to send data back and forth.
- Other components and services are not required to be compliant to RDF. Since the vast majority of services of the SnR ecosystem communicate with JSON and XML, conversion between standards would be costly.
- While the inferencing capabilities of RDF are impressive, they serve their purpose at the cost of speed. Usually, inferences take a long time to produce the desired outputs and require lots of resources (in terms of processing power) to do so.

To that end, we overcome these shortcomings by incorporating a second database into our solution, Elasticsearch. General advantages of Elasticsearch which assist in overcoming the shortcomings mentioned above include the following:

- Elasticsearch stored data in JSON format, which means the only data sent through the network are the ones which are truly necessary. This addresses both the shortcoming of bandwidth occupation and compatibility with the rest of the SnR ecosystem (since data is in JSON).
- While Elasticsearch doesn't support the inferencing capabilities of RDF, it supports sophisticated indexing mechanisms. In fact, its performance in specific indexed queries against an Elasticsearch backend is generally considered very fast. To achieve this level of efficiency, the data structures stored in it need to be carefully planned, but the result is definitely worth the effort.

Both databases serve a purpose and they complement each other and, consequently, both will be part of our proposed solution. In cases where inferencing and analytics are required, data will be retrieved from RDF4J, while where there is a need for real-time monitoring and timely alerts, ElasticSearch will be providing the corresponding data.

Furthermore, depending on the evolution of needs of the project and the general situation awareness the SA model is aiming to achieve, a hybrid storage of data could also be realised. In that context, data duplication will be minimized and part of a coherent data structure will be stored in each of the two databases. However, it should be emphasized that this approach will only be considered if performance of the existing solution does not meet the users' needs.

4.7 Deviations compared to the old architecture

In this subsection, we are going to discuss the changes and adaptations of the architecture, when compared to the first iteration. It should be noted that, due to the SA model presented originally in D3.2, "Situation Awareness Model – specification V1" and the BIM model presented in D3.3, "BIM based services and applications - review and service design" having a co-dependent overall architecture, the adaptations documented here affect the outcomes and direction of both deliverables.

The stack of components presented here does not contradict the old architecture. This was an intentional choice, in order to minimize the required changes of the implementation as much as possible. The old architecture involved the additional modules of the Data transformation handler and the Semantic Data model interceptor. While both modules' functionality and role are documented in D3.3, for the sake of completeness we also provide here a brief description of their role:

- The Data transformation handler is responsible for transforming data from heterogeneous sources into a uniform and processable structure, so that the data can be stored and queried against in the future.
- The Semantic Data model interceptor is responsible for correlating information between the SA and BIM models, when such correlations are required for queries and data retrieval.

These two modules can be seen as intermediate modules in the overall system comprised of the two models (SA and BIM). This means that the components presented here do not contradict the old architecture at all. Rather, they merely expand it for the SA model implementation.

Therefore, taking all of the above into consideration, a general workflow/component stack can be seen as follows:

- Data are retrieved in timed intervals via the dynamic **Scheduler**.
- Data reaches the system.
- Data are transformed via the **Data transformation handler**.
- If necessary, the data is correlated with more data from other sources through the **Semantic Data model interceptor**.
- The data is filtered through the **Rules engine** in order to check whether an alert event is triggered.
- If an alert is triggered, the **Push notification service** will make sure the alert reaches relevant actors and/or components in a timely fashion.
- The data will be stored in the **Semantic Database** (SA database of the old architecture) for future reference and as historical data.
- External entities can interact with the system through the **RESTful interface**.

The overall stack of components, taking both the old and the new architecture into consideration is illustrated in the figure below (Figure 3):

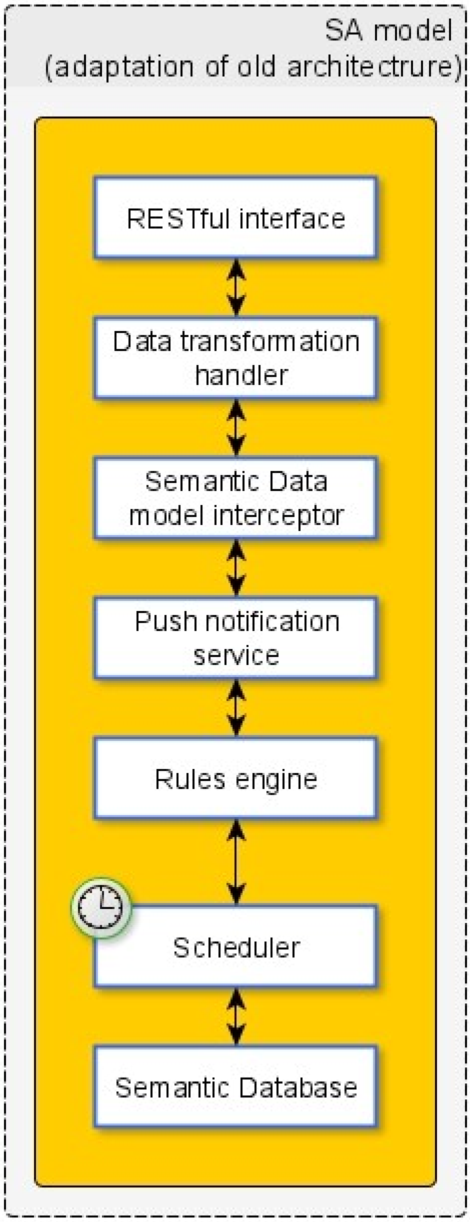

Figure 3: Components stack of the old and new architectures combined.

4.8 User needs

As explained already, the model has its purpose shifted slightly, when compared to its first iteration. This implies that, while the old user needs will still be satisfied, we need to place emphasis and thoroughly document the user needs which arise in the current stage of the development process. Due to supporting general situation awareness and closely interacting with a number of devices to achieve such a goal, all of them are paired with specific actors and those correlations feed the rules engine in order to send sophisticated messages to relevant actors/components, in case an alert is triggered. Because of the central role of the SA model in alerting events detection from critical biometrics' values, our component needs to provide the following core functionalities:

- **Scheduled pinging** of devices for biometrics aggregation and storage. The scheduling itself should be dynamic in nature, adapting to the needs of specific SnR operation and, perhaps, even at a device-specific level (for example, smartwatches getting pinged more often than GPS trackers).
- **Interfaces** to provide aggregated data about specific devices, actors, date ranges etc.
- Sophisticated **rules engine** which filters biometrics and occasionally triggers alarm events.
- **Storage** of alarms for future reference as historical data.
- **Timely notification** and signalling of relevant parties about alerts.
- **Real-time updating** of information as needed. For example, a device paired with a first responder suddenly changing its paired actor, once the first responder places the device on a victim to measure their vitals.
- Thorough **documentation** of the provided infrastructure. In order to ensure maximal usability of the system, it is imperative to describe its functionality and behaviour in detail. Additionally, first responders should be allowed to verify the conditions which trigger alerts, as well as tweak the generated messages for alerts.
- **Decision Support**. While not strictly a DSS component, the SA model plays a central role in general situation awareness, by monitoring devices and alerting actors and components on time.

4.9 Situation Awareness model Objectives and Means of Verification

While the purpose of our component has slightly deviated from the first iteration, the original objectives and means of verification remain immutable. In this subsection, we will describe the new objectives and corresponding verification means which are required by the current status of the proposed solution. That being said, the model's new aims include:

- Proper aggregation of biometrics and alerts per device, per actor.
- Dynamic scheduling at device/actor level to ensure efficient real-time monitoring of the devices, avoiding redundant pinging (a behaviour which would burden the SnR ecosystem).
- Filtering of biometrics' values against rules, which trigger conditional alerts.

Taking all of the above into account, we present how these goals can be achieved by conceptualizing the domains described in Subsection 4.8 in Table 4. For each such domain, we also provide the conceptual entities playing a central role, which will be further described in Section 5.

Table 4: Objectives and Means of Verification

Objective	Domain of SA requirements	Concepts
Proper aggregation of biometrics and alerts per device, per actor	Interfaces	• Actor • Alert Trigger • Device
	Storage	• Actor • Alert Trigger • Device
	Decision Support	• Alert Trigger • Device
	Scheduled pinging	• Actor • Device

Dynamic scheduling at device/actor level to ensure efficient real-time monitoring of the devices, avoiding redundant pinging	Real-time updating	• Actor • Device
	Storage	• Actor • Alert Trigger • Device
Filtering of biometrics' values against rules, which trigger conditional alerts	Rules engine	• Alert Trigger • Device
	Decision Support	• Alert Trigger • Device
	Timely notification	• Alert Trigger • Device

5 Semantic Data Model Specification

In this section, we are going to present the overall semantic data model specification. More specifically, we are going to describe the core entities and concepts of the model, their relation and hierarchy, their properties etc. Moreover, we provide relevant terminology which was provided in D3.2, "Situation Awareness Model – Specification v1", where needed (for the sake of completeness and to make the flow of reading easier). Furthermore, we will present the SA model in action, with two (2) examples of general usage throughout the SnR ecosystem.

5.1 Vocabularies and Ontologies

As already explained, in this iteration of the SA model's development, we need to take a slightly different direction. However, this indicates enrichment and augmentation of our previous work, not a completely new implementation. To that end, our old ontology and semantic model will remain as is, this being the case for the interlinking of our ontology with the state-of-the-art models/ontologies of POLARISCO, IMPRESS, FOAF and GEONAMES (see D3.2 "Situation Awareness model – Specification V1"), which we updated and expanded for our needs. The new ontology of the SA model builds upon the same concepts and follows a rather similar structure. This fact will become more evident as the reader continues with the subsections following immediately after.

5.2 Search and Rescue Model

Based on the semantic model requirement analysis, we are going to present the updated Search and Rescue model. It should be noted that we are going to present only the new additions in the next subsections. The definitions and specification of the Semantic Data Model of the previous iteration of the SA model (D3.2, "Situation Awareness Model – specification V1") remain as is.

Generally speaking, there are three main concepts throughout the model:

- **Classes**, which represent the core entities of the model.
- **Object properties**, which conceptually link the model's classes, aiming to represent relations.
- **Data properties**, which accompany each class of the model. Each class is defined to have a specific set of properties correlated to it and this information is explicitly stated in the described ontology instead of it being implied by an inferencing engine.

5.2.1 Model representation in OWL

In this subsection, we are going to justify the reasons which led us to design the provided model/ontology in its current state. More specifically:

- We will showcase subtyping relations and the need for additional entities insertion in the ontology.
- We will showcase subtyping relations between object properties, as well as which semantic attributes they should have (functional, inverse functional, reflexive, irreflexive, transitive, symmetric, asymmetric).
- We will discuss data properties which have a complex range type.

Further on, we will describe attributes which carry semantic elements and meaning, as well as terminology used throughout the ontology's description. The main elements defining an (our) ontology are described in the next subsections, alongside respective data and object properties.

More specifically, we should take into consideration the following properties attributes (which have already been presented in D3.2 but are placed here once more for the sake of completeness):

- **Functional**: An object property is defined as functional if for each instance x of a specific class, there can be at most one individual instance y of another class, such that x is linked to y via this object property.
- **Inverse functional**: An object property is defined as inverse functional if for each instance x of a specific class, there can be at most one individual y of another class, such that y is linked to x via this object property.
- **Reflexive**: An object property is defined as reflexive if an instance is linked to itself via the object property.
- **Irreflexive**: An object property is defined as irreflexive when no instance is linked to itself via the object property.
- **Transitive**: If an instance x of a specific class is linked to an instance y of another class, which in turn is linked to an instance z of yet another class (both links have the same object property), x is also linked to z via the same object property. In this case, the property is a transitive one.
- **Symmetric**: If an instance x of a class is linked to an instance y of another class via this object property, then y is also linked to x via the same property. In this case, the property is a symmetric one.
- **Asymmetric**: If an instance x of a class is linked to an instance y of another class via this object property, then y cannot be linked to x via the same property. In this case, the property is an asymmetric one.

5.2.1.1 Entity classes

In order to supplement the understanding of our proposed ontology, the entities/classes of the model, as well as their object property relations are presented in a grid illustrating their hierarchy (left to right indicates topmost to bottom classes) and are displayed in Figure 4.

The main/core entities of the model are the following:

- **Actor**: An entity representing an actor/component. Actors are first responders in most cases and they are paired with one or more devices.
- **Alert Trigger**: An entity representing an alert trigger (generated by the SA model).
- **Device**: An entity representing a general device. **Device** is a top-level entity, which further divides into:
  - **ERHCMD**: The Emergency Response Health Condition Monitoring Device.
  - **Physio DSS**: The result generated by PhysioDSS representing a state and specific triggers.
  - **Six Gas Hazmat Monitor**: The six gas hazmat monitor device.
  - **Smartwatch**: This smartwatch device.
  - **Wearable GPS Tracker**: The GPS tracker device.

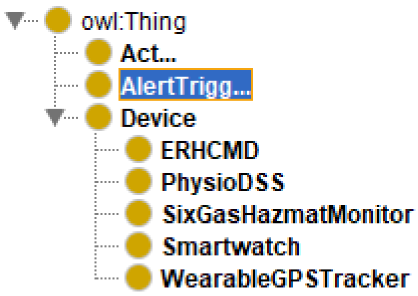

Figure 4: Search and Rescue ontology's classes.

5.2.1.2 Object properties

Leveraging subtyping relations among the ontology's entities, the defined object properties are as flexible as possible with respect to the entity type of their domain and range. This means that each object property tries to link higher-level classes as much as possible, since this implies that lower-level classes having the same parents can also utilize these properties themselves.

For the most part, the object properties of the ontology carry the following attributes:

- Due to the relations being one-to-many, the direct connections are functional, while the reverse ones are reverse-functional.
- They are not transitive, because in general the properties comprising the model do not carry any transitive semantics conceptually.
- They are irreflexive, because their domain and range differ.

The object properties of the model are the following (Figure 5):

- **Actor Has Device**: A relation representing the fact that an actor belongs to an actor group.
- **Device Paired with Actor**: A relation representing the fact that an actor group has access rights.
- **Alert Triggered By Device Status**: A relation representing the fact that an actor has a means of communication available.
- **Device StatusTriggersAlert**: A relation representing the fact that an actor is related to health impact.

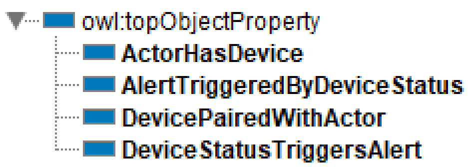

Figure 5: Search and Rescue ontology's object properties.

5.2.1.3 Data properties

Regarding data properties, the vast majority have an intuitive data type. The aim of the data properties is a simple, yet extensible representation, which can be provided in JSON format. That being said, depending on the needs of the actors and components the SA model will be sending data to, it is possible that a need for specifically structured messages arises. In that case, the symptoms inferred by the rule's engine, as well as the message generated will be converted into an XML string. The reason for this is to keep structured information in a serialized format, in order to better assist services consuming such information from the system. For example, services might need to store into the model a structured JSON message indicating a plan of action to be taken by first responders and then, another service will get this information from that system in order to forward it further. If the two services have already agreed upon the message's representation, then they can just deserialize this information and use it directly, instead of being forced to reconstruct it manually by parsing each field individually. Naturally, throughout Search and Rescue's development, this design is subject to change, in order to better facilitate the needs of the components and the end users.

It should also be noted that we will take into consideration the EDXL standards, when representing messages, similarly to the first iteration of the model. As a quick reminder, EDXL is a suite of XML-based messaging standards that facilitate emergency information sharing between government entities and the full range of emergency-related organizations. EDXL standardizes messaging formats for communications between these parties. Implementation of EDXL standards aims to improve the speed and quality of coordinated response activities by allowing the exchange of information in real time.

The data properties of the model are the following (Figure 6):

- **Actor ID**: Property of **Actor**. The unique identifier of an **Actor**. String type.
- **Actor Role**: Property of **Actor**. The role they own. Obviously, an actor can have more than one role (in semantic terms and RDF this means a single **Actor** instance can have more than one **actor Role** data properties). String type.
- **Alert Actor ID**: Property of **Alert Trigger**. The equivalent of **actor ID**, stored internally in an alert, so that the alert can be correlated with a specific actor. String type.
- **Alert ID**: Property of **Alert Trigger**. The unique identifier for each alert generated by the SA model. Long type.
- **Alert Message**: Property of **Alert Trigger**. The exact message to be passed when pushing the alert notification. String type.
- **Alert Symptoms**: Property of **Alert Trigger**. The exact symptoms experienced during an alert. The symptoms message is as detailed as possible to describe what the actor is currently experiencing. String type.
- **Device ID**: Property of **Device** (and, therefore, also property of **ERHCMD**, **PhysioDSS**, **Six Gas Hazmat Monitor**, **Smartwatch**, **Wearable GPS Tracker**). The unique identifier of each device. String type.
- **erhcmdBloodOxygenLevelSpO2Percentage**: Property of **Emergency Response Health Condition Monitoring Device** (**ERHCMD**). The percentage of blood oxygen level (SpO2) measured. Double type.
- **erhcmdBodyTemperatureCelsius**: Property of **Emergency Response Health Condition Monitoring Device** (**ERHCMD**). The degrees in Celsius of the body temperature measured. Double type.
- **erhcmdHeartRateBPM**: Property of **Emergency Response Health Condition Monitoring Device** (**ERHCMD**). The measured heart rate in beats per minute (bpm). Integer type.

- **erhcmdRespirationRateBPM**: Property of **Emergency Response Health Condition Monitoring Device** (**ERHCMD**). The measured respiration rate in beats per minute (bpm). Integer type.
- **pdssEDTPIE**: Property of **PhysioDSS**. The measured estimated average time of death (EDT). Long type.
- **pdssGlasgowComaScaleComputation**: Property of **PhysioDSS**. The measured Glasgow Coma Scale computation. Double type.
- **pdssJumpStartTriage**: Property of **PhysioDSS**. The jump start triage status, generated by Physio DSS. Integer type (0, 1, 2, 3).
- **pdssSieveTriage**: Property of **PhysioDSS**. The sieve triage status, generated by Physio DSS. Boolean type.
- **pdssSortTriage**: Property of **PhysioDSS**. The sort triage status, generated by Physio DSS. Boolean type.
- **pdssStartTriage**: Property of **PhysioDSS**. The start triage status, generated by Physio DSS. Boolean type.
- **sghmCH4ExposurePPM**: Property of **SixGasHazmatMonitor**. The measured exposure to CH_4 in parts per minute (ppm). Integer type.
- **sghmCO2ExposurePPM**: Property of **SixGasHazmatMonitor**. The measured exposure to CO2 in parts per minute (ppm). Integer type.
- **sghmCOExposurePPM**: Property of **SixGasHazmatMonitor**. The measured exposure to CO in parts per minute (ppm). Integer type.
- **sghmH2SExposurePPM**: Property of **SixGasHazmatMonitor**. The measured exposure to H_2S in parts per minute (ppm). Integer type.
- **sghmO2ExposurePPM**: Property of **SixGasHazmatMonitor**. The measured exposure to O_2 in parts per minute (ppm). Integer type.
- **smartwatchBodyTemperatureCelsius**: Property of **Smartwatch**. The degrees in Celsius of the body temperature measured. Double type.
- **smartwatchHeartRateBPM**: Property of **Smartwatch.** The measured heart rate in beats per minute (bpm). Integer type.
- **timestamp**: Property of AlertTrigger and Device (and, therefore, also property of **ERHCMD**, **PhysioDSS**, **SixGasHazmatMonitor**, **Smartwatch**, **WearableGPSTracker**). The UNIX timestamp the entity was received (for devices data) or generated (for alerts).
- **wgpstSpeed**: Property of **WearableGPSTracker**. The measured speed of the device. Double type.

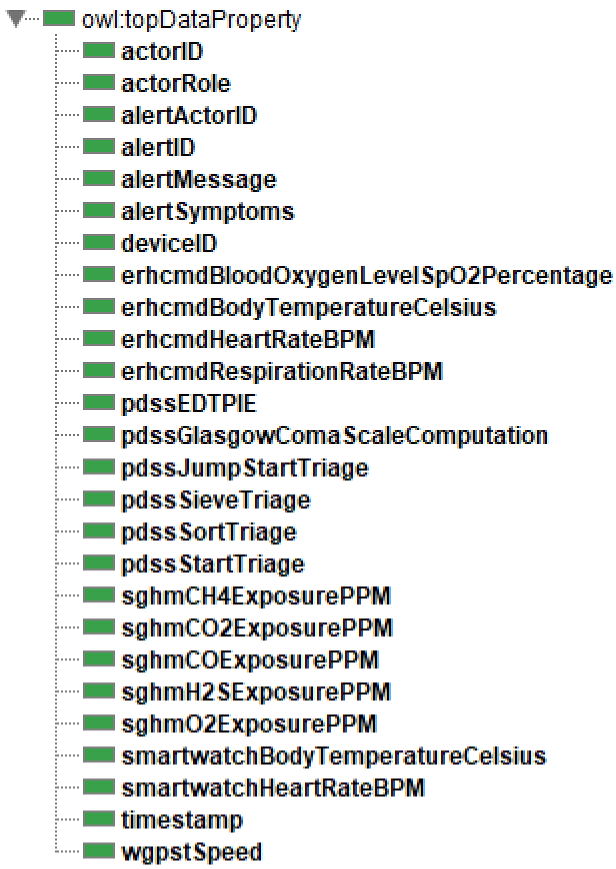

Figure 6: Search and Rescue ontology's data properties.

5.2.1.4 Ontology classes as a graph

In order to supplement the understanding of our proposed ontology, the entities/classes of the model, as well as their object property, relations are presented in a grid illustrating their hierarchy (left to right indicates topmost to bottom classes) and are displayed in Figure 8.

For the sake of completeness, we provide how the classes of the ontology are connected together to form the model's semantic graph, in Figure 8.

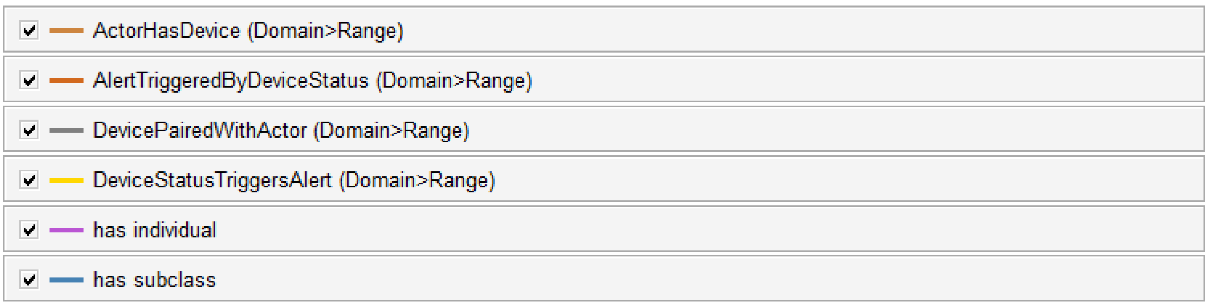

Figure 7: Search and Rescue's ontology in semantic graph representation's memo.

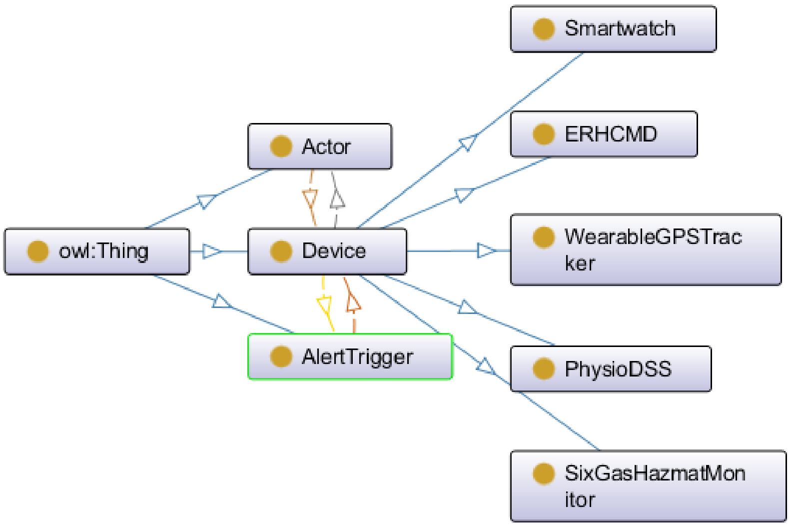

Figure 8: Search and Rescue's ontology in semantic graph representation.

5.3 Examples of semantic representation and usage

In this subsection, we are going to illustrate the usage of the SA model and service, in the context of two (2) examples: an alert event in which the SA model needs to notify the actor and a component querying the SA model for historical data/alerts.

5.3.1 Example 1: Push notification from alert event

For the sake of this example, we are going to assume that a first responder with id=a1 is paired with a six gas hazmat monitor device with id=s1. The SA model will periodically communicate with the REST API of the device to obtain values of interest. For the sake of this example, let's assume that the last 10 requests (spaced out in increments of 6 minutes for a total history of one hour) returned a CO value in the range of 80-100 ppm. Based on the rules engine, this condition triggers an alert event, because the first responder is in risk of irregular cardiac beats, if the value remains as is for 1-2 hours. Therefore, the SA model will push a notification to the device. The object will have a structure like the one below:

```
{
  alertActorID: "a1",
  alertID: "alert1",
  timestamp: "67876234",
  alertSymptoms: "irregular cardiac beats",
  alertMessage: "CO exposure detected. Try to avoid exposure as much as possible."
}
```

It should be noted that this example is a very special case, due to the time constraint it has as an additional condition. Not only does the value need to be in a dangerous threshold, it also needs to remain like this for 1-2 hours. This means that the SA model is supposed to check all previous values of the devices for this first responder and ensure the condition is met fully.

5.3.2 Example 2: Querying the SA model

For the sake of this example, we are going to assume that a first responder with id=a1 is paired with a smartwatch with id=s1. Let's also assume that a component which generates analytics for the health status of first responders needs to gather all values observed in the smartwatch, as well as occasional alerts which might have been generated by the SA model. By providing a REST API for this exact requirement, the SA model is capable of filtering its stored data for all smartwatch values for the smartwatch with id=s1 and correlated (paired) to a1. Furthermore, the SA model will also gather all alerts which involve both s1 and a1. As a final step, the SA model will sort the gathered results by timestamp, to accommodate time-series processing as well. The final result will be a JSON array, composed of JSON objects, one for each time step. The JSON objects can be one of two types:

Monitored values: Objects of this type will have a structure like the one below:

{
  deviceId: "s1",
  deviceType: "smartwatch",
  timestamp: "344365879",
  smartwatchHeartRateBPM: 85,
  smartwatchBodyTemperatureCelsius: 36
}

Alert: Objects of this type will have a format similar to the response shown in Example 1.

6 Future work and conclusion

6.1 Future work

The outcome of the current documentation is to establish an excellent point of reference for the functionality of the SA model. While everything is documented as thoroughly as possible and, based on our proposed methodology, the SA model will be continuously developed, adapted and expanded upon, taking into consideration state-of-the-art approaches during this process. Additionally, we will continuously monitor the SA model in order to ensure maximal coverage of the project's needs, especially the use cases' requirements. Furthermore, as the development of the model continues, we will keep calibrating the model with the goal of maximizing usability, compatibility of the model with well-known industry standards, increase the credibility of Search and Rescue as a whole, as well as minimizing its complexity.

6.2 Conclusion

The current deliverable documents the second iteration of the semantic model, which will constitute the Situation Awareness model of Search and Rescue. In this context, the working methodology in designing the second iteration of the provided ontology was described, while specific devices which play a key role to the general situation awareness the SA model is aiming to achieve are described, as well as critical biometrics' values, symptoms and actions to be taken. Furthermore, the document presented the overall architecture of the second iteration of the model, its main components, its differences and evolution from the first iteration, as well as the user needs it is currently addressing and the latest means of verification. Moreover, a thorough specification of the model, vocabulary, entities and properties linking the entities was provided, while the important concepts involving the attributes accompanying properties of the model were also briefly explained. The model's usage in action was shown through two (2) representative examples throughout the SnR ecosystem. Finally, future work on how to better improve the model during Search and Rescue's development was presented.

Annex I: References

[1] "METHONTOLOGY: from ontological art towards ontological engineering", Accessed: Dec. 01, 2021. [Online]. Available: https://www.researchgate.net/publication/50236211_METHONTOLOGY_from_ontological_art_towards_ontological_engineering

[2] G. Burel, L. Piccolo, K. Meesters, and H. Alani, "DoRES - A Three-tier Ontology for Modelling Crises in the Digital Age," *undefined*, 2017.