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

D4.7 DSS Validation

Work package:	WP4 – Data aggregation, Analysis and Decision Support
Authors:	CNR
Status:	Final
Due Date:	31/10/2021
Version:	1.00
Submission Date:	30/10/2021
Dissemination Level:	PU

(*) Dissemination level.-PU: Public, fully open, e.g. web; CO:Confidential, restricted under conditions set out in Model Grant Agreement; CI: Classified, Int = Internal Working Document, information as referred to in Commission Decision 2001/844/EC.

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:	www.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	20/09/2021	Simona Panunzi (CNR)	ToC
0.20	27/09/2021	Simona Panunzi (CNR)	Sections 1, 2, 3.2 and 4.1.1, Annex III
0.30	6/10/2021	Christos Angelidis Kallia Angelaki (KT)	Sections 3.1, 4.2
0.40	10/10/2021	Simona Panunzi (CNR)	Sections 5,6, Annex II
0.50	20/10/2021	Marco Guerri (THALIT)	Review
0.60	25/10/2021	Simona Panunzi (CNR), Christos Angelidis (KT)	Minor Revisions to the document
0.70	27/10/2021	Patrik Karlsson (AiDEAS)	Review
0.80	30/10/2021	Christos Ntanos	Quality Control
1.00	30/10/2021	Christos Ntanos	FINAL VERSION TO BE SUBMITTED

Executive Summary

The aim of the present deliverable is to present the DSS validation framework, in all its steps, from the software verification to the establishment of a panel of experts identifying possible shortcomings or further user requirements that the stakeholders would declare to be necessary.

Several scenarios (simulated Uses Cases, described in Section 4) will illustrate the potentiality and usefulness of the S&R DSS platform from a user perspective. The proposed simulated Use Cases will be the object of the upcoming tabletop demonstration (the first one taking place before the execution of the real S&R Use Case 1: Victims trapped under the rubble, Italy).

The simulated Use Case Scenarios of Section 4 will therefore have the purpose to show all the DSS features implemented in the first version; feedback gathered from the panel of experts will help to fill the existing gaps to create an improved subsequent version of the component.

The present document also shows the way the DSS will be validated in the scheduled real S&R Use Cases and will provide a series of benchmarks against which the performance of DSS will be measured in a real situation. A User/Observer questionnaire will be adopted as a further instrument of validation.

The document is structured in the following Sections: in Section 2, Methodology, a background to a DSS validation approach is presented, with the description of the procedure set up for the establishment of the panel of experts. Section 3 presents a detailed description of the software verification; Section 4 presents the simulated Use Cases foreseen for the tabletop demonstration and Section 5 gives an overview of the real S&R Use Cases where the DSS will be validated with the presentation of the User /Observer questionnaire which will be administered at the end of the real demonstrations.

Finally, Section 6, Conclusions, summarizes the results achieved and their usefulness in relationship with future work to be performed.

Table of Contents

List of Tables

List of Figures

1 Introduction

The Search & Rescue project aims at designing, implementing and testing a highly interoperable, modular open architecture platform for first responders, by incorporating next generation R&D and COTS solutions for enhancing situation awareness, supporting rescue operations and improving the efficiency of the decision-making process. This is specifically important and of great usefulness in emergency health operations when emergency health service providers and decision makers have to face two great challenges: the disproportion between needs and available human/material resources and the time constraints typical of an emergency. The Search & Rescue framework is thought to provide decisional support instruments and monitoring systems to enable first responders to have an effective and unified vision of the dynamical changes occurring during the crisis event and of the available human and material resource capacity of the response system.

One of the Search & Rescue components devoted to assist the managers of a crisis or the first responders in the process of a fast and accurate decision making is the Decision Support System (DSS) module.

The module will assist and help the coordination of the response activities, using data from multiple, heterogeneous sources (historical and real-time data) and will be used at the strategic, operational and tactical level in order to address the following objectives:

1. optimal allocation of available EMS (Emergency Medical Sevice) units to incidents, depending on the estimated needs
2. optimal allocation of patients to transport vehicles and first receivers (hospitals), based on given order of evacuation and triage results from the occurred injuries
3. optimal allocation of tasks to available actors on the field, given the specific requests by the field commander
4. prediction of the evolution of the physiological status of the victim based on the occurred injuries, health measurements and treatment administration

The first three objectives are fulfilled by the implementation of three Services provided by the SOT (Situational Operational Tactical) DSS, the last one is addressed by the PHYSIO component of the DSS.

At the present stage of the project, the DSS (with its SOT and PHYSIO components) has been delivered in its first version (M14), this means that its functionalities will be enriched during the next year(s) of the S&R project. This enrichment process will be the result of new user requirements that should emerge from the DSS validation process, whose description constitutes the object of the present deliverable (D4.7, M16). The aim of the present deliverable, therefore, is that of presenting the DSS validation framework, in all its steps, from the software verification to the establishment of a panel of experts identifying possible shortcomings or further user requirements that the stakeholders would declare to be necessary. Several scenarios (simulated Uses Cases, described in Section 4) will illustrate the potentiality and usefulness of the S&R DSS platform from a user perspective. The proposed simulated Use Cases will be the object of the upcoming tabletop demonstration, the first one taking place before the execution of the real S&R Use Case 1 (Victims trapped under the rubble, Italy), planned for M22. The simulated Use Case Scenarios of Section 4 will have therefore the purpose to show all the

DSS features implemented in the first version; feedback from the panel of experts will help to fill the existing gaps to set up an improved subsequent version of the component. Suggestions and improvements will be included in a subsequent version of the DSS that will be then tested and validated in the real Use Case Scenarios. Since the second and final version of the component will be due at month 22, at the same time of the first Use Case, further improvements that should result to be necessary from its validation in a real setting will be validated in the Use case 5 and Use Case 7, planned at month 24 and between M28 and M29, respectively.

2 Methodology

2.1 Background to DSS validation

This deliverable deals with two aspects related to an effective and correctly implemented Decision Support System (DSS) release: the verification and validation of the DSS.

By convention, Verification is the process of checking whether the software system meets the specified requirements of the users; Validation is the process of checking whether the software system meets the actual requirements of the users [1].

Boehm [2] characterised the difference in the following two statements:

1) **Verification is building the system right**.
2) **Validation is building the right system.**

Both the SOT and the PHYSIO DSS rely on some hypotheses, that corresponds to a conceptual model for a simplified description of the reality. Simplifying, in the SOT DSS, which deals with the minimization or maximization of an objective function, the models (or hypotheses) at the basis of the developed modules are related to the construction of the objective function itself, while in the Physio DSS, the model is related to the representation of the victim physiology (see deliverable D4.3-D4.6 for details). The adoption of a simplified description of the reality leads to a more or less credibility of the outputs. The performance of the DSS will be initially tested by means of computer simulations and the credibility of the simulation results will depend on both the model correctness and the accurate formulation of the problem. A process of Verification and Validation (V&V) must be therefore employed with the initial presentation of simulated results [3].

Verification deals with the correctness from an implementation point of view, that is with the correctness of the transformations of the inputs to the outputs. Verification must guarantee that the system behaves as expected, according to the system specifications.

Validation deals with building the right model, that is with the correctness of the implementation from a domain of applicability point of view. Validation must answer the question if the hypotheses made are correct and if the system behaves consistently with the study objectives and end-user requirements.

The final aim of the V & V process are therefore [4]:

to detect bugs in the implementation, to search for defects in the implementation and to stop and resolve them;

(1) to detect problems related to the areas of applications or to highlight important missing parts (omissions) in the development.

Similarly, the U.S. Food and Drug Administration (FDA) guidelines report the following definition for validation and verification [5]:

Verification: "***Software verification provides objective evidence that the design outputs of a particular phase of the software development life cycle meet all of the specified requirements for that phase. Software verification looks for consistency, completeness, and correctness of the software and its supporting documentation, as it is being developed, and provides support for a subsequent conclusion that software is validated***".

Validation: "***confirmation by examination and provision of objective evidence that software specifications conform to user needs and intended uses, and that the particular requirements implemented through software can be consistently fulfilled.***"

The activities related to Verification include the software testing, whose objective is that of verifying and confirming that the outputs of the software are conform to the inputs; various static and dynamic analyses; code and document inspections; walkthroughs.

The activities related to software validation may be performed both during and at the end of the software development life cycle and are addressed to ensure that all requirements have been fulfilled.

The Verification and Validation process is therefore a complicated task which requires software testing, inspections, analyses, and other tasks that are performed at different stages of the software development life cycle.

Testing of device software functionality in both a simulated environment and in the user site (real environment) must be typically included in a validation framework for a software device.

The same concept is reported by Lamy [6] "***Before Decision Support systems are used in practice, these systems need to be extensively evaluated to ensure their validity and their efficiency***". Preece [4] classify the methods for DSS testing in:

- **static methods**, which in principle do not require the use of the DSS because they are focused on finding out syntactic, logical or semantic errors. These activities fall within the Verification process and are addressed to verify whether the DSS meets the technical and semantic user requirements;
- **dynamic methods** which require the use of the DSS in a set of test cases. This can be done by setting up specific test cases (the simulated Use Cases defined in Section 4), salient for testing purpose, and involving experienced people in the field. These experts should be required to evaluate the output by comparing the automatic results with what was expected or with results that would have been obtained without the use of the DSS.

It is worthwhile, however, to briefly discuss the difficulties encountered when validating a system. The topic has already been discussed in a previous work [7] aiming at describing the evaluation framework set up for the IMPRESS solution (FP7-SEC-2013-1 - Collaborative project).

While real-time simulated scenarios (tabletop sessions) represent the most immediate choice for evaluating the output of a DSS, it must be kept in mind that the input data and the outcomes are completely predictable and reflect the programmed situation (the simulated Use Case). The outcome therefore can be appropriately evaluated and measured: the simulation can be adapted by setting different intensity, severity, by choosing the number of people involved (victims and rescue personnel) and the simulation could represent an effective instrument for learning, exercising and training in different operational phases (command, control, communication) and each consequence of each decisions made can be identified and discussed. However, as Paul and Hlupic state [8], the real word is dynamic and testing a system (or a model) in a perfectly controlled situation is reductive and express the need to validate a system in a context that is as near as possible to the real situation which the system was created for.

This means performing testing in a framework where randomness, human error, and unexpected developments affect the course of the events. Only a live exercise could represent an effective framework where testing a system. In a live exercise, in fact, unexpected situations (random variables) could interfere with the operational procedures and expected results, and this becomes the right setting to demonstrate the real usability, usefulness and efficacy of a DSS projected for the real world.

However, when evaluating a DSS in real exercise, performing a rigorous statistical comparison of quantitative indicators, chosen to evaluate the performance of the DSS, becomes impossible. A rigorous approach would require the conduction of parallel, randomized, blind testing procedures to compare results obtained with and without the use of a DSS. It is obvious that for cost reasons and for the fact that no live exercise can be perfectly reproducible a such approach is impossible to implement.

In such a complex situation an ad-hoc methodology is necessary. Starting from the experience gained in IMPRESS, the Verification and Validation of the S&R DSS will occur in three steps:

1) an internal procedure for the software verification (for both the SOT and the PHYSIO DSS) is set up to find inconsistencies and bugs related to technical aspects
2) tabletop sessions where test cases (simulated Use Cases) will be presented to a panel of experts to test specific DSS functionalities (services provided by both the SOT and the PHYSIO DSS component) whose output will be benchmarked against best practice
3) execution of S&R Use Cases (in particular Use Case 1, Use Case 5 and Use Case 7) where the entire functionality of the DSS will be tested in a real situation.

Verification and Validation of the S&R DSS will be detailed in Sections 3 and 4, respectively.

2.2 Establishment of the panel of experts

As stated above, the role of the panel of experts is that of providing meaningful input and suggestions to evaluate and improve the outputs of the S&R DSS.

The procedure of constituting the panel of experts began by first requiring each partner to provide a name of a professional, along with his/her CV, with expertise in physiology or in crisis management. Table below reports the initial set of professionals suggested by the S&R partners, along with their expertise and country of origin:

Table 2.1: List of the professionals submitted to the vote

	Organization	expertise	Country of origin	Proposing partner
Radu Andriciuc	Ministry of Interior - ISOP	Crisis and Civil Emergency Management	Romanian	PROECO
Philippe Besson	French Firefighters	Rescue and fire – disaster preparedness – technological risks specialist – investigator for fire origin	French	PUI
Themistoklis Karafasoulis	Greek Fire Service	Critical control room digitization for communications and Command & Control	Greece	EPAYPS
Anastasios Kanavos				EPAYPS
Giovanni Magenes	University of Pavia	Biomedical Signal and Image Processing - Biomedical Engineering	Italy	UNICA
Rainer Thell	Medical Emergency Department, Donauspital Vienna	Managing Consultant - Emergency Department Specialist in Anesthesia and Intensive Care Medicine	Austria	JOAFG
Cristina Gómez Usabiaga	Servicio de Urgencia Médica de la Comunidad de Madrid – SUMMA112	Out of hospital Critical Care Mobile Unit (Advanced Life Support ambulance)	Spain	SUMMA112
Arturo de Blas	Samur-Protección civil	Managing operations of Emergency Service - Selection and implementation of technological devices at the Communications, Electromedicine and Informatics level, in	Spain	ESDP

		coordination with the Resources Department		
Daniele Gui	Policlinico Universitario "A. Gemelli"	Emergency and Trauma Surgery	Italy	CNR

After having received suggestions a poll was instituted among the partners to set up the final panel, according to the expertise. The elected members were: Themistoklis Karafasoulis, Cristina Gómez Usabiaga and Daniele Gui. They were immediately contacted to establish the date of the first meeting for the execution of the first tabletop which is scheduled for November/December 2021.

The Evaluation Committee, besides providing inputs to the S&R DSS, asking for highlighting possible critical points, will be invited to participate in the live S&R Use Cases. They will be administered an evaluation questionnaire and will be asked to compile the final evaluation report. Annex II reports the formal invitation, Annex III reports the Evaluation questionnaire.

3 DSS software verification

3.1 SOT DSS verification process

SOT DSS is an effective management system to allocate resources in case of a natural disaster. The development of the SOT DSS aimed in the decision-making procedures having the following functionalities:

- **Service 1**, recommendation of the most efficient allocation of resources to incidents,
- **Service 2**, recommendation of the optimal allocation of patients to transport vehicles and first receivers (hospitals), based on given order of evacuation and triage results for present injuries
- **Service 3**, recommendation of allocation of tasks to available actors on the field, given demand pre-defined by the field commander
- **Service 4**, the estimation of casualties and demanded resources (EMS units), given historical data on emergency incident recordings

From the above services, the first three use optimization techniques in order to recommend the allocation of supply resources based on given demand while the last one estimates the number of casualties and the required resources. Regarding Service 2, at this stage the evacuation order, the physiological score, and the location of the patient are used, and later on, the types of traumas will be utilized. Also, at the current stage the Allocation of Patients to EMS unit has been implemented. For Service 3 the roles will be defined based on the roles of the COncORDE platform. Roles, such as Rescuer, Retriever and Runner. However, in the first version of SOT DSS, this service was implemented with instances (e.g., R1, R2, R3), in case there is a new role need to be added in the 2nd version.

Regarding Service 4, the Advanced National Seismic System (ANSS) Comprehensive Earthquake Catalog (ComCat) is used. ComCat contains earthquake source parameters (e.g., hypocenters, magnitudes, phase picks and amplitudes) and other products (e.g., moment tensor solutions, macroseismic information, tectonic summaries, maps) produced by contributing seismic networks. Furthermore, through ComCat onePAGER product is provided. OnePAGER deliver information about an earthquake through an API, for instance the estimated fatalities, economic losses, and other useful information for the end-users. In this version of the deliverable, Service 4 will not be validated due to its early stage and it will be tested in the second version of the validation.

In the Deliverable 4.3 'Design of SOT DSS components' a detailed presentation of all the Services is presented. The Verification & Validation process of the SOT DSS aims to discover and minimize the deviations between the user (experts) perspective and the current state of SOT DSS. This process will lead to the next version of the SOT DSS based on the Knowledge and expertise of people in the field. In this process there will be a number of steps that will be followed.

1. The first step is the definition of Inputs and outputs
2. The acceptance of the inputs and outputs
3. The creation of specific simulated scenario Use Cases for each Service of the SOT DSS. In these simulated scenarios the SOT DSS will run and give results.
4. The validation of the results above from the experts and get feedback on how the SOT DSS operates from the users' perspective. The possible recommendations will be taken into consideration in order to improve the functionality of the SOT DSS.

In the 2nd Phase, another round of Validation in the real time Use Cases will be implemented where the next version of the SOT DSS will be validated.

The development of the SOT DSS components was implemented by using Python v3.8. It is well known that Python has a plethora of libraries which offer APIs for external services to support the service's functionality. For optimization problems, pyomo v5.7 is the main library which is able to give the right tools for symbolic modelling of linear programming problems. In order to endpoints and the web service of the SOT DSS to be establish Flask v2.0 was also used. Using dummy data as input for the algorithms, Jupyter Notebooks were created for the demonstration. Finally, the application has been containerized with Docker in order to be deployed independently on the server.

For the development of the SOT DSS specific packages were used. More specifically:

Table 3.1: Description of the packages

Package	Description
Pyomo[1]	A Python-based open-source software package that supports a diverse set of optimization capabilities for formulating, solving, and analysing optimization models.
TransportationProblem	A python package which contains modules required by the implementation of the transportation problem of linear programming using pyomo.
UTA	A python package which contains modules required by the implementation of the UTA method using pyomo.

[1] http://www.pyomo.org/

EMSAllocation	A python package which contains modules required by the implementation of the EMS allocation to incidents service.
PatientAllocation	A python package which contains modules required by the implementation of the patient allocation to transport vehicles and first receivers' service.
TasksAllocation	A python package which contains modules required by the implementation of the task allocation to actors' service.

Each from the above package provides a main or utility functionality for the Services of the SOT DSS. All the scenarios were be executed using Postman. It is an HTTP client that tests HTTP requests, utilizing a graphical user interface, through which we obtain different types of responses that need to be subsequently validated. All the services will export an API which will be used in Postman given the inputs above and take the related output. The following figure summaries the whole verification process:

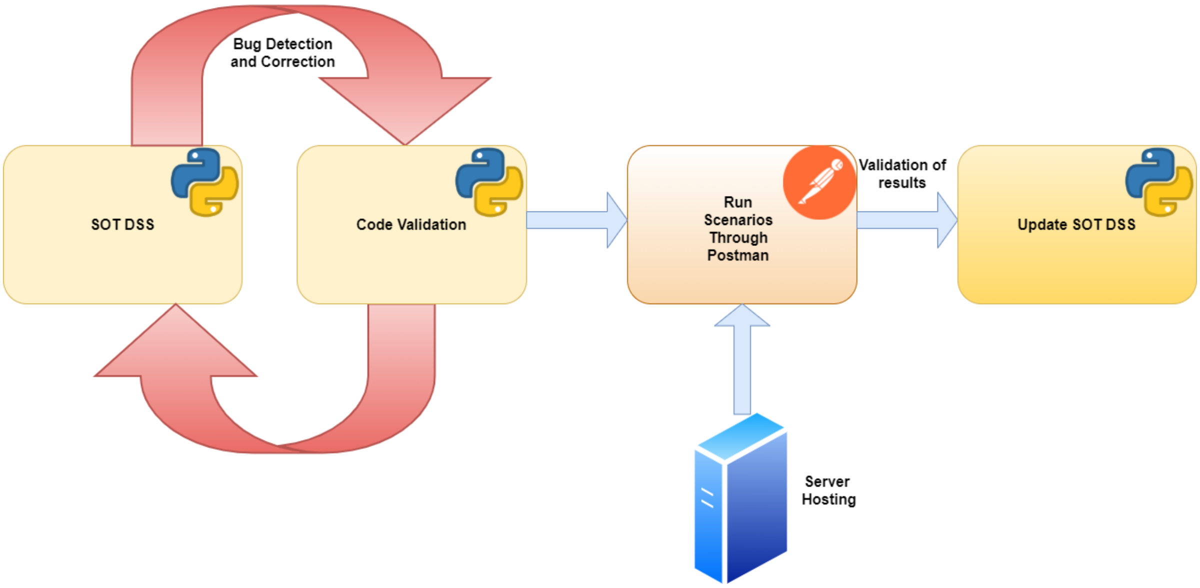

Figure 3.1 SOT DSS verification process

In this verification process the whole functionality of the DSS will be tested in order to find inconsistencies and unintended bugs (related to technical aspects) along with the operational aspects (users' perspective).

3.2 PHYSIO DSS verification process

The PHYSIO DSS is the second of the two components of the S&R Decision Support System. It includes a series of modules, functions and algorithms to support the services offered by the S&R DSS in order to

- provide suggestions about which resources to assign to a victim (for example based on the expected time to death corresponding to each possible treatment to deliver)
- provide a patient prioritization based on either the expected time to death or according to the values of computed scores and known triage algorithms.

To fulfill the above objectives the PHYSIO DSS provides the following functionalities:

- it may be called upon to simulate different Crisis Scenarios (number of victims, position of the victims with respect to event location, injuries affecting each victim);
- it describes the evolution of each (real or virtual) victim's physiological status, in terms of probability distributions, on the basis of health measurements from the field and of assigned treatments;
- it provides for each victim a specific score useful for victim prioritization: the Estimated Time to Death (ETD);
- it provides the automatic computation of relevant indices and scores for the assessment of the victim's impairment and triage.

The architecture of the PHYSIO component is constituted of the following elements:

1. **classes**: each class is a set of homogenous elements (Events class, Injuries class, Treatments class, Health resources class, Health measurements class);
2. **functions:** relationships between inputs and outputs. The functions can be external, that is functions that offer a named service and can be called by the S&R DSS non-PHYSIO modules for returning a result, or internal, that is hidden functions that are merely called internally by PHYSIO modules and that support some of the functionalities of the external functions;
3. **algorithms**: a series of rules, as for example triage algorithms.

One of the most important elements of the PHYSIO DSS is the PIE: an encoded long character string, including all the information about a victim (probability distributions of the Physiological State Variables (PSVs), gender, age, victim identification code). The PIE is the core packet for the exchange and sharing of information (see Deliverables D4.4 and D4.6).

We recall that the PSVs are the dimensions along which patients change their physiological conditions following injuries or therapy/treatment administration and that the evolution of the physiological state of a victim is described by mean of the distributions of values of the physiological dimensions as well as of their rate of worsening. Moreover, it is to be underscored that the PHYSIO DSS component is based on a client-server architecture delivered as Web Services located at the CNR-IASI Biomathematics Laboratory (https://biomatlab.iasi.cnr.it/SearchAndRescue/SearchAndRescue.wsdl), and exposes its functionalities via WSDL (Web Services Description Language). We refer to deliverables D4.4 and D4.6 for a detailed description of the PHYSIO DSS.

In order to set up the PHYSIO DSS verification process, a series of requirements have been defined:

R1. Definition of INPUTs and OUTPUTs for each function
R2. Definition of the range of acceptability of the inputs and outputs
R3. Definition of the system response
R4. Definition of possible errors and the way they must handled
R5. Definition of the user environment
R6. Definition of the way the user interacts with the software
R7. Definition of the performance requirements

Each function and algorithm of the PHYSIO DSS component has followed or is following a pre-defined development process. Deliverable D4.4 presents the design of the PHYSIO DSS. Here functions and algorithms are described in terms of input/output (requirements R1, R2) with the corresponding levels of release (internal/external).

Each function has been tested and the test design has been decided before the implementation phase was started. The development and test of the PHYSIO DSS Component make use of different programming languages, including MATLAB®, C++, XML and PHP (requirement R5). The use of more than one programming language and the involvement of different programmers allows the semantic and syntactic check of the implemented functions.

MATLAB® [9], being a mathematical interpreted language and a numerical simulation toolbox, is mainly used for prototyping functions and algorithms for an easier and more immediate semantic debugging, allowing to verify, by means of graphs and simulations, if the system responses are compliant with what expected.

C++ [10] is a middle-level compiled language employed for fast code execution. All the algorithms prototyped in MATLAB are translated to C++ and consistency of the outputs of the two implementations is verified for different sets of inputs.

PHP [11] is a scripting language operating on a Web Server. It is used to build a module, invoked on the web server, to call the compiled C++ executable files.

XML [12], Extensible Markup Language, is a markup language adopted for the construction of the Web Service Description, the public interface of the web service, and for a simplified data sharing and exchange according to the SOAP protocol [13].

For each function or algorithm, the design and test foresee, therefore, the implementation of 6 items. They are reported and described in the following table:

Table 3.2: List of the items for the development and testing PHYSIO DSS process

item	environment	description
א aleph	MATLAB	A Matlab function, that is a series of statements which take the inputs and return the outputs
ג gimel	MATLAB	a sample function in MATLAB which calls the corresponding aleph function locally and shows the results
ד dalet	MATLAB	a client function in MATLAB which calls remotely the webservice (which uses the server-located vav function) and returns the output
ו vav	C++	a function implemented in C++ which translates the corresponding aleph function into compiled executable code
ח chet	MATLAB	A MATLAB function calling both the function vav remotely (compiled in C++) and the MATLAB function aleph present in the local environment for a comparison of the results
PhysioService	PHP	A PHP module which calls the functions (vav items) located on the server

The implementation of the 6 items constitutes both the development and the verification phase of the PHYSIO DSS component.

To be noted that the MATLAB *chet* and *dalet* scripts can be replicated for any desired programming language on the client side, so that no constraints are imposed by the proposed architecture. The choice to provide the functionalities of the PHYSIO DSS as webservice addresses the requirement R6.

The figure below summaries the process.

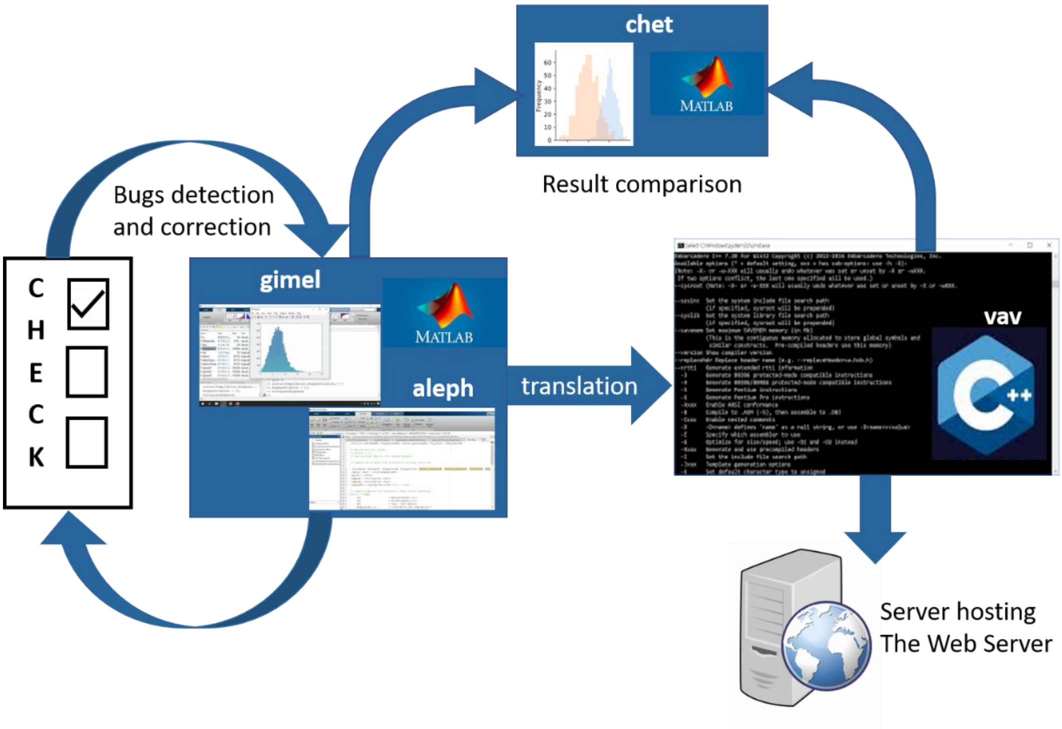

Figure 3-2: The PHYSIO DSS verification process

The functions developed in the Matlab design environment (aleph) are all equipped with an additional function (gimel) calling aleph itself. The function gimel allow to test the execution of aleph showing the results and verifying if the obtained output is compliant with the results defined in R3.

The two items (aleph and gimel) are therefore double checked by other two programmers (who did not take part in the production process) for a syntactic and logic check of the implemented procedures. Possible errors and the way they can be solved are noted and all the bugs are corrected before the next step is implemented. Once the two items (aleph and gimel) have been checked and corrected for possible errors, the aleph function undergoes the "translation" process in C++ (vav) to improve the performance of the services offered (requirement R7). Since from a semantic point of view the aleph item is correct, the test of the vav item consists in the comparison of the outputs generated by the aleph and vav items (chet). Since most of the functions are not deterministic but contain different elements of randomness (see deliverables D4.4 and D4.6), comparisons between functions implemented in different programming languages are tackled with a statistical approach.

4 DSS validation 1° phase: tabletop demonstration

4.1 Introduction to Use Cases

A **Use Case** in Testing is a description of a particular use of the software application or system by an actor or user. Use cases are planned by considering the actions a user can perform and the software application's responses to user actions. The test cases involve the interactions between users and software applications and are used to identify gaps in the software application both from a development point of view and from the point of view of the responses that the system provides. Use cases assume therefore a relevant importance in the phases of the Software Development Life Cycle. They must include a description of the Actions performed by the Actor/User and the corresponding Behaviour of the System to the User 'Actions'. In the present document we use a simplified version of the Use Case Template proposed by Alistair Cockburn (https://www.cs.otago.ac.nz/coursework/cosc461/uctempla.htm):

Table 4.1: Use Case definition Template

Use Case *number*	***name***
Objective	description of the objective that the system, or a subsystem of it, must achieve
Preconditions	what we expect is already the state of the world
Success End Condition	the state of the world upon successful completion, that is successful ways the use case might end.
Failed End Condition	the state of the world if goal abandoned, that is unsuccessful ways the use case might end.
Primary Actors	a role name or description for the primary actor. It is the actor that initiates this use case
Main success scenario	A description of events for the most likely termination outcome. It is a list of the actions the actor does and how the system responds (steps of the scenario from trigger to goal/result delivery)
Extension	Write the extensions (a list of use cases that are associated with the current use case), each referring to the step of the main scenario: *Description of the altered condition causing branching* *Altered step 1: action or sub.use case description* *Altered step 2: action or sub.use case description*

	…
Use case notes and open issue	Information that is not directly part of this use case but that the solution developer needs to consider and/or a list of issues on this use cases pending any decision

The test Use Cases presented in the following subsections will show the functionalities of the system to the panel of experts and will help to identify those requirements that have not yet been addressed by the first version of the system.

4.2 Validation of the SOT DSS through simulated scenario Use Cases

The validation of the SOT DSS will be executed through the implementation of simulated scenarios for each of the 3 services. Each Service uses a number of specific variables as inputs and gives back an output. All inputs and outputs are presented in the following Table.

Table 4.2 Inputs and outputs of DSS services

Services	Inputs	Outputs
Service 1	id : Unique id of the EMS Unit	Number of EMS units from an EMS station to incident
	Fleet Size of EMS unit from an EMS station	Reduce demand: Boolean variable with True if the demand was reduced, False otherwise
	Location (Latitude, Longitude) of the EMS Unit	
	Location (Latitude, Longitude) of the Incident	
	Demand for EMS units from the Incident	

Service 2	id : Unique id of patient	The id of the EMS and the id of the patient that should be transferred
	Physiological score of the patient	
	Evacuation order	
	is_child: Binary variable that define if patient is child or not	
	Location of the patient	
	id: Unique id for the EMS unit	
	Location: (Latitude, Longitude) The location of EMS unit	
Service 3	id : The unique id of the actor	The id of the actor and the id of the task that is assigned to the actor
	role: The role of the actor	
	Physiological score of the actor	
	Location: (Latitude, Longitude) Location of the actor	

	id: Unique id of the task	
	Role: The necessary role required by the task	
	demand: The demand of actors in the task	
	Location: (Latitude, Longitude) The location of the task	

4.2.1 Use case 1: Allocation of available EMS units to incidents

In this section a Use Case 1 scenario about the Service 1 from the SOT DSS is analyzed. The main success scenario is described as follows:

Use Case 1	***EMSAllocation***
Objective	Allocation of available EMS units to incidents, depending on estimated needs. Use of the **EMSAllocation** package
Preconditions	Earthquake event. Let assume the existence of six EMS stations (hospitals in Italy): Vittorio Emanuele III, Neuropissichiatria Infantile, Presidio Ospedaliero Maria Immacolata Longo, Ospedale Ingrassia, Pronto Soccorso, John Paul II and assume that two areas Poggioreale and Alcamo, have been involved in the incident:

<table>
<tr><td>Success End Condition</td><td>The demand from the two sites has been satisfied and the available resources have been assigned to the accident sites, satisfying all or part of the request.</td></tr>
<tr><td>Failed End Condition</td><td>The allocation is based on specific requirements. Although, the results will be available, it is not known if the results correspond to reality. Thus, the role of validation is essential for the optimal operation of the SOT DSS.</td></tr>
<tr><td>Primary Actors</td><td>High Commander

Field Commander

PSAP (Public safety answering point)</td></tr>
<tr><td>Main success scenario</td><td>The incident resulted in total demand less than total supply (no excess demand)

<u>Step 1</u>. The user enters the data problem: a list of EMSStation objects and a list of Incident objects.

<u>Step 1.1</u>: data for each EMSStation object have been correctly inserted (the fleet size and the EMS station location)

(Example)
<table>
<tr><th>NAME</th><th>FLEET SIZE</th><th>LAT</th><th>LON</th></tr>
<tr><td>Vittorio Emanuele III</td><td>7</td><td>37.824</td><td>12.8038</td></tr>
<tr><td>Neuropsichiatria Infantile</td><td>2</td><td>37.6785</td><td>12.7927</td></tr>
<tr><td>Presidio Ospedaliero Maria Immacolata Longo</td><td>6</td><td>37.5642</td><td>13.7307</td></tr>
<tr><td>Ospedale ingrassia</td><td>5</td><td>38.0604</td><td>13.537</td></tr>
<tr><td>Pronto Soccorso</td><td>7</td><td>38.0088</td><td>12.9261</td></tr>
</table></td></tr>
</table>

John Paul II	4	37.522	13.0762
Total Supply	31		

Step 1.2 data for each *Incident* objects have been correctly inserted (the *demand* and the incident *location)*

(Example)

NAME	DEMAND	LAT	LON
Poggioreale	8	37.7621	13.0343
Alcamo	10	37.9766	12.9627
Total Demand	18		

Step 2: The allocation problem is solved.

Step 2.1: the *EMSAllocation* class uses the *allocation()* method for the computation of the cost matrix;

Step 2.2: the *TransportationProblem* class is used to solve the transportation problem for the optimal allocation;

Step 2.3: the allocation solution is returned;

Step 2.4: The Boolean variable, indicating if the demand has been reduced or not, is FALSE.

(Example)

FROM	TO	EMS UNITS
Vittorio Emanuele III	Poggioreale	4
Vittorio Emanuele III	Alcamo	3
Neuropsichiatria Infantile	Poggioreale	2
Pronto Soccorso	Alcamo	7
John Paul II	Poggioreale	2

		Reduce Demand: False
Extension	The total demand is greater than the total supply	

Reduce Demand	False

Extension

The total demand is greater than the total supply

Step 1.2.a: data for the Incident objects are provided where the demand for resources exceeds the availability

(*Example*)

NAME	**DEMAND**	**LAT**	**LON**
Poggioreale	8	37.7621	13.0343
Alcamo	10	37.9766	12.9627
Castelvetrano	21	37.6785	12.7918
Total Demand	39		

Step 2.2.a: the *TransportationProblem* reduces the demand for each node (accident site) by approximately an amount equal to

$$\left(1 - \frac{Total\ Supply}{Total\ Demand}\right) \times 100$$

Step 2.2.b: the *TransportationProblem* computes the reduction percentage 20.51%

Step 2.2.c: the *TransportationProblem* computes the new demand and take the integer part

Step 2.2.d: the *TransportationProblem* checks if total demand is less than total supply. If yes, find the minimum demand and add 1 Else return result.

FROM	**TO**	**EMS UNITS**
Vittorio Emanuele III	Castelvetrano	7
Neuropissichiatria Infantile	Castelvetrano	2
Presidio Ospedaliero Maria Immacolata Longo	Poggioreale	3

	<table><tr><td>Presidio Ospedaliero Maria Immacolata Longo</td><td>Castelvetrano</td><td>3</td></tr><tr><td>Ospedale ingrassia</td><td>Poggioreale</td><td>5</td></tr><tr><td>Pronto Soccorso</td><td>Alcamo</td><td>7</td></tr><tr><td>John Paul II</td><td>Castelvetrano</td><td>4</td></tr><tr><td>Reduce Demand</td><td colspan="2">True</td></tr></table>
Use case notes and open issue	End users must evaluate the results and the inputs which were considered in the objective function of the TransportationProblem module.

4.2.2 Use case 2: Allocation of patients to transport vehicles and hospitals

Use Case 2	***PatientAllocation***
Objective	Allocation of patients to EMS Units, depending on estimated needs. Use of the **PatientAllocation** package
Preconditions	Earthquake event. It is assumed that there are six Patients with the following characteristics: id, physiological score, order of evacuation if the patient is a child or not, and its location(lat,lon). Furthermore, it is assumed that there are three EMS Units each of them characterized by an id and the location(lat,lon).
Success End Condition	Optimal allocation of Patients to the EMS Units
Failed End Condition	The allocation is based on specific requirements. Although, the results will be available, it is not known if the results correspond to reality.

	Thus, the role of validation is essential for the optimal operation of the SOT DSS.
Primary Actors	High Commander Field Commander PSAP (Public safety answering point)
Main success scenario	The allocation of the patient is implemented by using the UTA method

Step 1. The user enters the data problem: a list of *Patients* objects and a list of *EMS UNITS* objects.

Step 1.1: data for each *Patient* object have been correctly inserted

(Example)

ID	SCORE	ORDER	IS CHILD	LAT	LON
1	5	4	1	38.0107	23.6022
2	2	6	1	38.0154	23.5981
3	3	5	0	38.0059	23.5958
4	1	1	1	38.0115	23.6011
5	1	3	0	38.0135	23.5942
6	2	1	0	38.003	23.6028

Step 1.2 data for each *EMS UNIT* object have been correctly inserted *(Example)*

ID	LAT	LON
1	38.0005	23.5941
2	38.0107	23.6042
3	38.0123	23.6022

Step 2: the allocation problem is solved.

Step 2.1: Using the *PatientAllocation* class and the UTA method, the allocation is the following:

EMS UNITS	PATIENTS

	<table><tr><td>1</td><td>4</td></tr><tr><td>2</td><td>1</td></tr><tr><td>3</td><td>6</td></tr><tr><td>-1</td><td>[2,3,5]</td></tr></table>
Extension	The allocation of the patient by using the Patient cost formula $$cost_{ij} = \frac{r_j \cdot \sqrt{1+d_{ij}}}{t_j^{\;2+c_j}}, \qquad i = 1,2,\dots,n \quad j = 1,2,\dots,m$$ *Step 2.1a*: Using the *PatientAllocation* class and the Patient cost formula, the allocation is the following: <table><tr><th>EMS UNITS</th><th>PATIENTS</th></tr><tr><td>1</td><td>3</td></tr><tr><td>2</td><td>6</td></tr><tr><td>3</td><td>1</td></tr><tr><td>-1</td><td>[2,4,5]</td></tr></table>
Use case notes and open issue	End users must evaluate the results and the inputs which were considered in the objective function of the TransportationProblem module.

4.2.3 Use case 3: Allocation of tasks to available actors on the field

Use Case 3	*TaskAllocation*
Objective	Allocation of tasks to available actors on the field. Use of the **TaskAllocation** package
Preconditions	Earthquake event. It is assumed that there are six actors each of them characterized by an id, their role, its physiological score and its location (lat, lon). Furthermore, it is assumed that there are three tasks by id, the demanded role and its location

Success End Condition	Optimal allocation of Task to available actors in the field.
Failed End Condition	The allocation is based on specific requirements. Although, the results will be available, it is not known if the results correspond to reality. Thus, the role of validation is essential for the optimal operation of the SOT DSS.
Primary Actors	Field Commander Actors

Main success scenario

The total supply is 6 (number of actors) and the total demand is 7 (sum of demand), so in this case there is excess demand. Thus a dummy supply node will be added.

Step 1. The user enters the data problem: a list of *Actors* objects and a list of *Tasks* objects.

Step 1.1: data for each *Actors* object have been correctly inserted

(Example)

ID	ROLE	SCORE	LAT	LON
1	R11	3	38.0237	23.9022
2	R23	5	38.0164	23.5915
3	R31	2	38.0059	23.5957
4	R12	6	38.0136	23.6322
5	R22	4	38.0138	23.5946
6	R11	1	38.413	23.6037

R11, R23, etc. represent the COncORDE roles, such as Runner, Retriever, Rescuer. The roles will be specified in 2nd version.

Step 1.2 data for each *Task* object have been correctly inserted *(Example)*

ID	ROLE	DEMAND	LAT	LON
1	R11	4	38.0255	23.5321
2	R23	1	38.0207	23.7042
3	R22	2	38.0423	23.5022

<table>
<tr>
<td></td>
<td>
<u>Step 2</u>: The allocation of tasks to available actors is implemented by using the UTA and there is excess demand.
<u>Step 2.1a</u>: Using the TaskAllocation class and the UTA method, the allocation is the following:
<table>
<tr><th>ACTOR</th><th>TASK</th></tr>
<tr><td>1</td><td>1</td></tr>
<tr><td>2</td><td>3</td></tr>
<tr><td>3</td><td>1</td></tr>
<tr><td>4</td><td>2</td></tr>
<tr><td>5</td><td>3</td></tr>
<tr><td>6</td><td>1</td></tr>
<tr><td>-1</td><td>1</td></tr>
</table>
</td>
</tr>
<tr>
<td>Extension</td>
<td>
<u>Step 2.1b</u>: Using the TaskAllocation class and Task Cost Formula method,

$$cost_{ij} = \varphi^2(r_i, r_j) \cdot s_i \cdot \sqrt{1 + d_{ij}}, \qquad i = 1, 2, \dots, n \quad j = 1, 2, \dots, m$$

the allocation is the following:
<table>
<tr><th>ACTOR</th><th>TASK</th></tr>
<tr><td>1</td><td>1</td></tr>
<tr><td>2</td><td>2</td></tr>
<tr><td>3</td><td>1</td></tr>
<tr><td>4</td><td>1</td></tr>
<tr><td>5</td><td>3</td></tr>
<tr><td>6</td><td>1</td></tr>
<tr><td>-1</td><td>3</td></tr>
</table>
</td>
</tr>
<tr>
<td>Use case notes and open issue</td>
<td>End users must evaluate the results and the inputs which were considered in the objective function of the TransportationProblem module.</td>
</tr>
</table>

4.3 Validation of the PHYSIO DSS through simulated scenario Use Cases

In the following subsections a series of Use Cases have been prepared to show the different functionalities of the PHYSIO DSS component.

4.3.1 Use case 4: generation of a-priori distribution for victims of an earthquake

Use Case 4	*Event generation*
Objective	Generation of an event using the **scenario_generator** function and utility functions for showing the results. The objective of the function is the generation of a certain number of victims each one presenting with several lesions which, on the basis of their severities, determine a defect on the physiological dimensions of the victims.
Preconditions	The test environment has been set up
Success End Condition	The function returns a matrix of PIEs (a PIE is a long vector including all the information about a victim), with the a priori distributions of the victims' physiological state. The Use case shows a series of plots summarizing the output of the function.
Failed End Condition	Wrong functioning of the function
Primary Actors	The Matlab user A series of functions internally called by the main function
Main success scenario	Generation of an earthquake of medium severity **Step 1.** The user calls the function in the testing environment with the exact input values: • the scenario ID (the type of incident) • the dimension of the event (the extension in terms of meters from the event location expressing the extent of the involved area) • the severity of the incident (MINOR = 1, MEDIUM = 2, MAJOR = 3, CRITICAL = 4) (Example) • the scenario ID = 2 (earthquake)

	• the dimension of the event = 100 meters • the severity of the incident = 2 (MEDIUM) **Step 2.** The function takes input values from some of pre-defined classes and calls internal and external functions of the PHYSIO DSS component. *Step 2.1*: SCENARIO_GENERATION loads the input matrices • event_class (including the minimum and maximum number of bystanders) • event_to_lesions_modal_severity (modal severity for each lesion in correspondence of each event) • event_to_lesions_max_probability (maximal probability of occurrence for each lesion in correspondence of each event) *Step 2.2*: SCENARIO_GENERATION generates the number of affected people and, for each generated victim, randomly assigns a longitudinal and a latitudinal deviation from the event location (*for each victim the a priori distribution of his/her physiological status is computed*) *Step 2.3*: SCENARIO_GENERATION calls the compute_a_priori function and for each victim the PIE is generated *Step 2.3.1*: compute_a_priori calls the DEFECT_GENERATION function. The matrices lesions_to_variables_delta and lesions_to_variables_alpha are loaded as input. They express the defects that each lesion provokes on the ten physiological variables in terms of instantaneous defect and rate of worsening **Step 3**. Use a series of utility functions to show the following results: • Plot of the location of the generated victims around the earthquake epicentre

	• Plot of the distribution of the distances of the generated victims from the event location • Plot of the severity distribution of the different injuries for a sample of victims (the closest to the event location and the furthest away from the event location) • Plot of the average severity distribution of the injuries for the closest victims to the event location and for the furthest away from the event location • Plot of the a-priori distribution of the ten physiological variables for victims located at the shortest distance from the event locations, for victims located at medium distance and for victims positioned at the maximum distance from the epicentre
Extension	No extension has been foreseen for this use case
Use case notes and open issue	End users must evaluate the reliability of the results and their suggestions will be used to modify the input matrices: *event_to_lesions_modal_severity, event_to_lesions_max_probability, lesions_to_variables_delta and lesions_to_variables_alpha*

4.3.2 Use case 5: computation of a-posteriori distributions from measurements from the field

Use Case 5	***Compute a posteriori distribution from Health measurement***
Objective	Computation of the posteriori distributions of the physiological state variables given the values of one or more health measurements from the field *Example: a first responder takes care of the first victim. The victim is positioned x meters away from the event location. The first responder communicates this very first information to the system. The system computes the victim PIE, including the a priori distributions of the PSVs. The first responder detects vital signs and takes some health measurements. The first responder takes the first measure, for instance the heart rate, and the system compute the updated PIE, including the*

	posteriori distributions of the PSVs. The first responder then observes other parameters, for example the O2 Oxygenation, and the system computes another updated PIE.
Preconditions	A PIE or a set of PIEs have been constructed and some health measurements on the victims are made available to the system
Success End Condition	The function returns the updated PIE which includes the a posteriori distribution of the victim physiological state The Use case shows a series of plots summarizing the output of the function.
Failed End Condition	Wrong functioning of the function
Primary Actors	The Matlab user A series of functions internally called from the main function
Main success scenario	Computation of the a posteriori distributions of the physiological variables on the basis of some health measurements on the victim **<u>Step 1</u>**. The user calls the function compute_a_priori in the testing environment with the exact input values: • idpat (the identification number) • demographic (gender, age, weight, height) • longitudinal_deviations (deviation from the event site in the longitudinal direction) • latitudinal_deviations (deviation from the event site in the latitudinal direction) • event_type (the scenario ID) • event_dimension (the extent, in meters, of the event dimension) • date (date in posix format) and returns the victim's first PIE containing the a priori distributions of the physiological variables **<u>Step 2</u>**. The user calls the function compute_a_posteriori_given_health_meas with the exact input values: • PIE (the last available PIE) • the ID of the health measurement;

	• the value related to the health measurement ID; *Step 2.1*: the compute_a_posteriori_given_health_meas function loads the input matrix: • Dependence_Health_Meas_PSV (the matrix which set the dependencies between each pair of PSV and health measurement) *Step 2.2*: the compute_a_posteriori_given_health_meas function calls the internal function conditional_probability_hm_given_psv for the computation of the conditional distribution of the health measurement variable given the value of the PSV (the computation of the distribution is necessary to apply a full Bayesian approach) *Step 2.2.1*: conditional_probability_hm_given_psv loads two matrices: • health_meas_to_variable_level_rate_LowerBound • health_meas_to_variable_level_rate_UpperBound The two matrices express the strength of association between the PSVs and each health measurement (*for each victim the a posteriori distribution of his/her physiological status is computed*) Step 3. Use of a series of utility functions to show the following results: • Plot of the a priori distributions of the physiological state variables of the victim on the basis of the first few observations (distance of the victim from the event site) • Plot of the a posteriori distributions of the physiological state variables of the victim on the basis of the first recorded health measurement value
Extension	The use case starts directly from **Step 2** with the last available PIE (which is the last updated PIE)

	Step 3a. Use of a series of utility functions to show the following results: • Plot of the updated posteriori distributions of the physiological state variables of the victim on the basis of the subsequent recorded health measurement values

Use case notes and open issue	• End users must evaluate the reliability of the results and their suggestions will be used to make changes to the input matrices:
	  • Dependence_Health_Meas_PSV health_meas_to_variable_level_rate_LowerBound
	  • health_meas_to_variable_level_rate_UpperBound

4.3.3 Use case 6: computation of a-posteriori distributions from treatment administration

Use Case 6	computation of a-posteriori distributions from treatment administration
Objective	Update of the posteriori distributions of the physiological state variables given the administration of one or more treatments *Example: after a first responder or a physician in the field has met a victim, assessed his/her vital signs, made some measurements and received additional information about his/her physiological conditions (as output from the PHYSIO DSS), he/she can ask how the victim's distributional physiological status appears in corresponding with the administration of different treatments.*
Preconditions	A PIE or a set of PIEs have been constructed and the first responder or physician has to administer some treatment to restore the victim's physiological impairment or to stabilize the patient before final and definitive care in some health facility

Success End Condition	The function returns the updated PIE The Use case shows a series of plots summarizing the output of the function.
Failed End Condition	Wrong functioning of the function
Primary Actors	The Matlab user A series of functions internally called by the main function
Main success scenario	Computation of the updated distributions of the physiological variables after treatment administration **<u>Step 1</u>**. The user calls the function compute_a_priori in the testing environment with the exact input values: • idpat (the identification number) • demographic (gender, age, weight, height) • longitudinal_deviations (deviation from the event site in the longitudinal direction) • latitudinal_deviations (deviation from the event site in the latitudinal direction) • event_type (the scenario ID) • event_dimension (the extent, in meters, of the event dimension) • date (date in posix format) and returns the first PIE of the victim containing the a priori distributions of the physiological variables **<u>Step 2</u>**. The user calls the function compute_distribution_given_treatment with the exact input values: • PIE (the last available PIE) • Treatment_ID (the ID of the administered treatment) • Treatment_Value (the quantity of the administered treatment, a proportion between 0 and 1, now set to 1) *<u>Step 2.1</u>*: the compute_distribution_given_treatment function loads the input matrix • TreatmentDeltaRate sheet “delta” • TreatmentDeltaRate sheet “rate" The two matrices report the effect of each treatment on the values and rates of improvement of the physiological variables

	(*for each victim the updated distribution of his/her physiological status is computed*) **<u>Step 3</u>**. Use of a series of utility functions to show the following results: • Plot of the a priori distributions of the physiological state variables of the victim on the basis of the first few observations (distance of the victim from the event site) • Plot of the updated distributions of the physiological state variables of the victim on the basis of the administered treatment
Extension	After **<u>Step 1</u>** the use case proceeds by executing **<u>Step 2</u>** of the Use Case 3 **<u>Step 2.a</u>**: The user calls the function compute_a_posteriori_given_health_meas with the exact input values: • PIE (the last available PIE) • the ID of the health measurement • the value related to the health measurement ID *<u>Step 2.a.1</u>*: the function *compute_a_posteriori_given_health_meas* loads the input matrix: • Dependence_Health_Meas_PSV *<u>Step 2.a.2</u>*: the function *compute_a_posteriori_given_health_meas* calls the internal function *conditional_probability_hm_given_psv* for the computation of the conditional distribution of the health measurement variable given the value of the PSV *<u>Step 2.a.2.1</u>*: the function *conditional_probability_hm_given_psv* loads two matrices: • health_meas_to_variable_level_rate_LowerBound • health_meas_to_variable_level_rate_UpperBound (*for each victim the a posteriori distribution of his/her physiological status is computed*)

	Step 2 **Step 3.a**: Use of a series of utility functions to show the following results: • Plot of the a priori distributions of the physiological state variables of the victim on the basis of the first few observations (distance of the victim from the event site) • Plot of the posteriori distributions of the physiological state variables of the victim on the basis of the recorded health measurement • Plot of the updated distributions of the physiological state variables of the victim on the basis of the administered treatment
Use case notes and open issue	• End users must evaluate the reliability of the results and their suggestions will be used to modify the input matrices: • TreatmentDeltaRate sheet "delta" • TreatmentDeltaRate sheet "rate"

4.3.4 Use case 7: computation of the Expected Time of death

While all the previous Use Cases shows the functionalities of the PHYSIO DSS services in a testing environment, the following Use Case aims at showing the ETD service by exploiting the PHYSIO DSS webservice architecture, with a remote call of the Expected Time to Death function. MATLAB is used to generate the ד(dalet) function, that is the client function which calls remotely the webservice. The webservice uses the server-located ו(vav) function (implemented in C++) and returns the output. For more details refer to Deliverable D4.6.

Use Case 7	computation of the Expected Time of death
Objective	Computation of the Expected Time to Death *Example: after a first responder or a physician in the field has met a victim, has evaluated his/her vital signs, has taken some measurements and has received additional information on his/her physiological conditions (as output from the*

	PHYSIO DSS), he/she can ask about his/her Expected Time to Death
Preconditions	A PIE or a set of PIEs have been constructed The remote correct call to the function (client side) has been implemented. The call must include all the variables and parameters necessary for the correct connection to the webservice as well as all the input parameters required by the function
Success End Condition	The call to the function returns the expected output which is stored in a cell array (return_string) which is then parsed. After the numerical output is returned the Use case shows the numerical results and a series of plots summarizing the output of the function.
Failed End Condition	• Wrong functioning of the implemented functions • Failure in the connection to the Server • Incorrect format of the call and return of a warning message from the server
Primary Actors	The Matlab user (client-side) The webservice
Main success scenario	Computation of the Expected Time to Death given a victim PIE. **Step 1**. The user (the client) run the function CallSearchAndRescueCNR which call the ETDPIE function residing in the development environment with the exact input values: • Obj = an object of the class CNR_services, including the addresses where the wsdl and the "services" (functions) of the PHYSIO DSS are located • callString = the string including the name of the "service" (function) to call (in this case ETDPIE) with the appropriate input (the last available PIE) and returns the output into the variable return_string which, once parsed, provides the distribution of the Expected Time to Death

	Step 2. Use of a series of utility functions to show the results: • Plot of the distributions of the values of the physiological dimensions and of the rate of worsening • Plot of the corresponding ETD distribution
Extension	No extension is foreseen
Use case notes and open issue	• End users must evaluate the reliability of the results and their suggestions will be used to improve the model assumptions and will lead towards a more realistic physiological modelling.

5 DSS validation 2° phase: real demonstration

Deliverables D8.2 to D8.8 report a first description of the scenarios that will be simulated during the executions of the planned S&R Use Cases. To date, the organization of the Use Cases is in a very early state and a lot of work will be done in the coming months to set up the first Use Cases. The organization of each Use Case will be carried out by considering the S&R components that will be tested.

In the following sections we give a description of the three S&R Use Cases in which the DSS will be evaluated, and a first version of Key Performance Indicators (KPIs) is presented. It should be noted that the list of KPIs is not final and will be finalized with the detailed planning of the Use Cases.

5.1.1 S&R Use Case 1: Victims trapped under the rubble (Italy)

The first S&R Use Case (S&R UC1) will take place in Poggioreale, a small community in Sicily, Italy. On January 15th, 1968, Poggioreale was hit by a terrible earthquake where about 900 people died and ten towns and villages were damaged. Given the current status of the city, it represents a perfect scenario where simulating an earthquake. The S&R UC1 will be organized by the CNR in strict collaboration with the Regional Department of Civil Protection of Sicily (DRPC) and with the participation of the Regional System of Civil Protection, the Fire Department, and the Municipality of Poggioreale. The S&R UC1 will have a duration of about 6/8 hours and at the end of the demo a debriefing session will be organized to discuss the main difficulties encountering in the use of the tested S&R components and where the questionnaires will be administered to the users, experts and external invited people (possible users and stakeholders).

Several situations will be simulated to test the different S&R components, as for example

- people trapped under the rubble and / or in premises not reachable as a consequence to the earthquake;
- the release of gases and / or other toxic substances;
- blocked roads preventing traditional vehicles to reach the area

While a detailed plan of the Use Case must be still defined, the UC1 will foresee a series of mock victims presenting with pre-specified health conditions, with symptoms to be detected and with values of certain parameters. This information will be used to feed the DSS in order for example to estimate the distribution of the values of the victims' physiological state and their evolution over time or to compute the Expected Time to Death, with the aim of guiding the manager and/or the SOT DSS in the victim prioritization (service 3) or in the choice of the best treatment to administer to the patient.

A series of Kee Performance Indicators (KPIs) will be defined to help the evaluation of the DSS in terms of its efficacy and performance. Some have already been reported in Deliverable D8.2 and a selection of those, strictly related to the DSS is, listed below:

1) degree of accordance/discordance of the decision-maker and the DSS recommendations (e.g., how many EMS unit the High Commander would actually dispatch to the specific Incident);
2) time that is required by the DSS to consume a message, process it and return the send a response back.
3) time until the first ill/injured victim has been triaged in the field;
4) time until first treatment was performed;
5) time until victim is evacuated from scene;
6) time until victim arrives to the Emergency Department;

7) time to notification of the first appropriate staff person who assumes medical management coordination role;
8) time to arrival of the first EMS ambulance on scene;
9) time to transportation/evacuation of the last ill/injured survivor from the scene;
10) time until last triage assessment in Emergency Department;
11) average time spent by victims on the scene;
12) number of victims that receive first triage;
13) number of victims transported to emergency department (first triage);
14) number of victims transported to emergency department.
15) Other indicators could be recorded and analysed with the scope of validation.

5.1.2 S&R Use Case 5: Victims trapped under the rubble (France)

Use Case 5 is the second pilot where the DSS will be tested and, as the Use Case 1, is centred on rescue operations for «Victims trapped under rubbles». It will be held on June 2022 in the town La Souterraine, in a site which represents the International Emergency Firefighters' (PUI) training facility and which meets the conditions to simulate an earthquake. The area indeed is characterized by several collapsed or damaged buildings, some of them created to reproduce a site hit by an earthquake.

The Use Case will be organized by the Pompiers de l' Urgence Internationale (PUI France) and will see the participation of other International Firefighters. It will last from the morning to the evening and as for the UC1 it will be attended by the personnel involved in the exercitations and by invited people, experts of the panel and stakeholders from local organizations. After the Use Case a post-exercise session (live or remote) will be held where questionnaires will be administered to get feedback on the effectiveness and usefulness, as well as on shortcoming and deficiencies, of the S&R DSS.

During the UC different situations will be simulated to allow the test of several S&R components, and the activities particularly connected to the S&R DSS validation are:

- the research and localization of the victims from the K9 team and electronic devices
- the control of gas absence or dangerous or radioactive materials
- the care of the victims
- the coordination of operations by a command post integrated into an operations base
- the medical post activated in the operations base.

A more time-oriented first version of KPIs has been defined and those related to the DSS validation are reported below:

1) Notification and coordination time
2) Time for crisis notification call issues by the Operations Centre to reach user's communication devices
3) Effectiveness in supporting the communication and coordination between first responders on the field and the Operations Centre
4) Effectiveness in monitoring and taking care of the health status of victims, first responders and volunteers
5) Total evacuation time for victims
6) Average time spent by victims on the scene before transportations in appropriate facilities for definitive cares
7) Time for rescuers to reach the first/last affected/injured person on scene

Once the definitive list of DSS related KPIs has been assessed for the UC1, it will be used in UC5 and possibly integrated with further indicators which could prove necessary.

5.1.3 S&R Use Case 7: Chemical substances spill (Madrid)

The S&R Use Case 7 is organized by SUMMA 112 and ESDP and is focused on a CBRN (Chemical, Biological, Radiological and Nuclear) incident with two separate scenarios. The pilot will be allocated to The National School of Civil Protection, Rivas Vaciamadrid, and will take place between October and November 2022. The school provides several scenarios and allows first responders to train in aquatic rescue, in collapsed structures, in fire, explosions, CBRN incidents, in transport of dangerous goods, etc.

The main objective of the UC is that of delimiting the working zones according to the existing risks and toxicity levels, to guarantee the safety of the first responders, rescue dogs, as well as of the victims. Since different events can occur, environmental or accidental, a zone declared safe can become dangerous and the S&R platform must be able to detect the occurrence of changes (by means of chemical sensors) and to promptly alert the first responders.

The first scenario will simulate an earthquake with collapsed structure of a residential building. At the beginning no gas risk is detected so the rescue operations will take place and several trapped victims will be assisted and cared. Several minutes after, the six-gas hazmat monitor will detect the presence of flammable gas and consequently, the gas leak from the first scenario will produce a deflagration (second scenario). The explosion will cause a gaseous leak of the ammonia.

The pilot will last 24-36 hours and pre- and post-sessions will be organized to train participants in the use of the system technologies and functionalities and to get feedback on the S&R DSS by means of questionnaires and open discussions.

The DSS will be validated for efficient real-time resource allocation, for the estimation of the number of expected victims/patients in such an incident as well as for the estimation of the number of necessary ambulance units, for the allocation of available EMS units to the incident and for the assignment of tasks to the rescue personnel. The PHYSIO DSS will support the DSS for a better victim prioritization, by estimating the expected time to death and the evolution of the physiological status.

The provisional list of KPIs is the one reported in the description of the Use Case 5.

5.1.4 The User/Evaluation questionnaire

The DSS evaluation has a twofold objective:

- it aims to minimize the gaps between user requirements and the design and implementation of the first version of S&R DSS solutions as resulting from tasks T4.5 and T4.6 of work packages 4.
- It aims to demonstrate the functionalities of the DSS to possible end-users during a real exercise.

The evaluation methodology is a long process since iterative in its nature, where improvements and modifications at each step will lead to improved but not final versions, converging towards a configuration that satisfies most of the users' needs.

The design of the DSS evaluation framework includes tabletop sessions and real exercises allowing the demonstration of the DSS functionalities to the end-users in their real environment and giving end-users the opportunity to assess the added value of using the S&R DSS.

The validation of the DSS in a real environment will be useful in particular to test technical aspects related to the communications, testing the flow of information from the field to the DSS passing through other S&R components, and to get feedback about the usefulness of the DSS (more related to the interaction user-platform). This second aspect is relevant for improving the usefulness of the DSS. Regarding validation, the goal of the live S&R Use Cases will be to assess the relevance to the end users of the functionalities that the DSS component aims to provide.

Taking advantage of the fact that the first version of the DSS has already been delivered, the validation during the live S&R Use Cases will be based on the following elements:

1. A group of high-level experts (constituting the panel of experts) has been identified, and they will be invited to express a judgement and to make suggestions to the questionnaire (Annex II) to be administered post-exercise. They will be also asked to fill in the questionnaire.

2. A number of professionals, from different organizations, who will participate in the S&R Use Cases and who will have the opportunity to exploit the functionalities of the DSS will fill out the questionnaire immediately after the end of the demo.

3. A set of quantitative, KPIs has been defined. They will be automatically recorded during the execution of the Use Cases.

The questionnaires to be administered will gather information about

1. The capability of the DSS to improve the performance of important functions during the crisis (for example in improving information on the physiological status of the victims)
2. Usability of the DSS and its compatibility with the standard procedures
3. Its utility in supporting the rescue operations and its ability to resolve problems encountered during a crisis
4. Open questions on ways of improvement, shortcomings, limitations and dysfunction

The questionnaire will be the main tool for gathering user feedback and another important opportunity to gather opinions and suggestions from the members of the panel of experts.

The Analysis of the questionnaires and of the collected KPIs will be discussed in the second version of this deliverable (D4.13)

6 Conclusion

This document describes the steps and procedures established to conduct the Verification and Validation of the S&R Decision Support System. In this initial phase of the project, Task 4.6 focused in particular in the Verification process of the first version of the DSS (both the SOT and the PHYSIO), which was released in month M14, with the aim of verifying that the software system meets the specified technical requirements. The deliverable also presented the steps taken for the subsequent Validation phase, with the establishment of the panel of experts and with the presentation of the tabletop Uses Cases and live S&R UCs where the DSS will be evaluated. This validation phase will allow to collect feedbacks and suggestions to highlight shortcoming and ways of improvements and to verify whether the system meets the actual needs of users and stakeholders. A questionnaire was also developed to be administered to experts, users and potential stakeholders during the post-exercise sessions. The results of the tabletop demonstrations and of the analysis of the questionnaires will be the subject of the second version of the deliverable associated with the DSS Validation task (D4.13) and due to month M26 and will form the basis for the design and development of the second version of the S&R DSS (tasks T4.5 and T4.6).

Annex I: References

[1] P. Alun, «Building the Right System Right: Evaluating V&V Methods in Knowledge Engineering,» *AAAI Technical Report ,* pp. 38-45, 1998.

[2] B. W. Boehm, «Verifying and Validating Software Requirements and Design Specifications,» *IEEE Software,* vol. 1, pp. 75-88, 1984.

[3] O. Balci, «Validation, verification, and testing techniques throughout the life cycle of a simulation study,» *Ann Oper Res,* vol. 53, p. 121–173, 1994.

[4] P. Pierce, «Software verification and validation,» in *IEEE Technical Applications Conference. Northcon/96. Conference Record*, 1996, pp. 265-268.

[5] U.S. Food & Drug, «General Principles of Software Validation, Guidance for Industry and FDA Staff,» Center for Devices and Radiological Health, Center for Biologics Evaluation and Research, 2002.

[6] A. E. J. N. A. V. a. J.-D. Z. ean-Baptiste Lamy, «Testing Methods for Decision Support Systems,» in *Decision Support Systems*, IntechOpen, 2010, pp. 87-98.

[7] F. Cibella, S. Panunzi, V. Cusimano e A. De Gaetano, «Decision support for medical disasters: Evaluation of the IMPRESS system in the live Palermo demo,» *International journal of disaster risk reduction,* vol. 50, p. 101695, 2020.

[8] R. J. Paul e V. Hlupic, «The CASM environment revisited,» *Proceedings of WinterSimulation Conference,* pp. 641-648, 1994.

[9] A. P. King e P. Aljabar, MATLAB Programming for Biomedical Engineers and Scientists, Elsevier, 2017.

[10] B. Johnson, Professional Visual Studio 2017 (english Edition), Wrox, 2017.

[11] R. Lerdorf e K. Tatroe, Programming PHP, O'Reilly Media, 2002.

[12] R. Richard, Pro PHP XML and Web Services (Books for Professionals by Professionals), APRESS, 2006.

[13] W3C, «Simple Object Access Protocol (SOAP) 1.1,» [Online]. Available: https://www.w3.org/TR/2000/NOTE-SOAP-20000508/.

Annex II: Invitation Letter

Dear ……………,

I am writing to you in your quality of internationally recognized expert in crisis management.

As you know, you were invited to participate in the European Search & Rescue (S&R) project in the role of expert evaluator of the S&R Decision Support System (DSS).

On behalf of the S&R Consortium, I hereby formally invite you to be part of the Panel of Experts. Your expertise will be very useful in the process of identifying important shortcomings and means of improvement in the S&R DSS, which aims to provide support in the decision-making procedures on the recommendation of the most efficient allocation of resources to incidents, of the optimal allocation of patients to means of transport and health facilities, of the task assignment to available actors on the field. Furthermore, the DSS embodies a component dedicated to the prediction of the evolution of the physiological state of the victims (a model-based module), whose assumptions must be validated. The validation procedures of the DSS will follow several steps. In your quality of expert you will be invited to participate in one or more (remote) tabletop sessions and in the live S&R Use Cases where the S&R DSS will be validated:

- S&R Use Case 1: "Victims trapped under the rubble (Italy)"
- S&R Use Case 5: "Use Case 5: Victims trapped under the rubble (France)"
- S&R Use Case 7: "Chemical substances spill (Madrid)"

The evaluation of the S&R DSS will depend on the opinion of the Panel of Experts in which you are formally invited to participate. You will also be asked to comment and provide suggestions on the definition of the list of Key Performance Indicators to be used in the aforementioned S&R Use Cases and to answer the post-exercise evaluation questionnaire. We would be honored if you could confirm your participation in this Committee: if the answer is yes, you will be contacted over the next few days to arrange a first meeting to start the DSS validation process. For your convenience, you will find the leaflet of the projects attached to this invitation.

Thank you in advance for your attention and best regards,

Simona Panunzi

Annex III: DSS Evaluation Questionnaire

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

DSS Evaluation Questionnaire

Specify your role in the S&R Use Case

- ☐ USER
- ☐ OBSERVER

Specify to which Organization/Agency you belong to

- ☐ Emergency Medical Service
- ☐ Medical (Hospital)
- ☐ Military
- ☐ Police
- ☐ Fire Fighters
- ☐ Public Institution
- ☐ Rescue organization
- ☐ Academic Institution
- ☐ Civil Protection
- ☐ Other (specify ____)

Please, specific your role in your Agency ___________________________________

Gender

- ☐ Male
- ☐ Female

Age

- ☐ Less than 30 years
- ☐ Between 31 and 40 years
- ☐ Between 41 and 50 years
- ☐ Greater than 50 years

Contribution of S&R Decision Support System (DSS)

Please rate from 1 (strongly disagree) to 5 (strongly agree) to express your opinion on the extent to which the S&R DSS has improved the performance of the following functions compared to similar situations where a DSS is not used (NA=Not Applicable)

Rate	**1**	**2**	**3**	**4**	**5**	**NA**
S&R DSS improves the EMS allocation	☐	☐	☐	☐	☐	☐
S&R DSS improves the patient allocation	☐	☐	☐	☐	☐	☐
S&R DSS improves the task allocation	☐	☐	☐	☐	☐	☐
S&R DSS improves the victim prioritization	☐	☐	☐	☐	☐	☐

S&R DSS improves the management of health resources coming from different agencies	☐	☐	☐	☐	☐	☐
S&R DSS improves the management of human resources coming from different agencies	☐	☐	☐	☐	☐	☐
S&R DSS improves coordination and control	☐	☐	☐	☐	☐	☐
S&R DSS improves victim information (patient status, victim physiological evolution, worsening/improvements)	☐	☐	☐	☐	☐	☐
S&R DSS improves victim triage	☐	☐	☐	☐	☐	☐
The productivity was improved by using the S&R DSS	☐	☐	☐	☐	☐	☐
The functions of the S&R DSS satisfied my needs	☐	☐	☐	☐	☐	☐
I am satisfied with the S&R DSS's capabilities	☐	☐	☐	☐	☐	☐
The S&R DSS worked as I expected	☐	☐	☐	☐	☐	☐

S&R DSS helped in the execution of the usual activities during the execution of the Use Case.

1	**2**	**3**	**4**	**5**	**NA**
☐	☐	☐	☐	☐	☐

The use of the S&R DSS allowed the normal course of the emergency plans, policies and procedures of your institution

1	**2**	**3**	**4**	**5**	**NA**
☐	☐	☐	☐	☐	☐

Response of the S&R DSS are effective, understandable, contain the expected information

1	**2**	**3**	**4**	**5**	**NA**
☐	☐	☐	☐	☐	☐

Please suggest one or more functionalities that should be added to improve the S&R DSS:

____

____

____

____

Compared to your usual tools S&R DSS helped in "Saving Time"

1	**2**	**3**	**4**	**5**	**NA**
☐	☐	☐	☐	☐	☐

The use of S&R DSS helps users in adequately fulfilling their tasks (only for Observers)

1	**2**	**3**	**4**	**5**	**NA**
☐	☐	☐	☐	☐	☐

S&R DSS represents an efficient support to the rescue activities during the Use Case (only for Users)

1	**2**	**3**	**4**	**5**	**NA**
☐	☐	☐	☐	☐	☐

S&R DSS capability solved partially/totally the problems addressed during the Use Case

- ☐ Yes
- ☐ No

Please, explain why

____

____

____

Questions about the usability of S&R Tool

Please rate from 1 (strongly disagree) to 5 (strongly agree) to express your opinion about the following statements

Rate	**1**	**2**	**3**	**4**	**5**	**NA**
The system was ready to use	☐	☐	☐	☐	☐	☐
The system was easy to learn	☐	☐	☐	☐	☐	☐

The labels describing the functions are easy to understand	☐	☐	☐	☐	☐	☐
The information was clear	☐	☐	☐	☐	☐	☐
It is necessary the support of a technical person to use this system	☐	☐	☐	☐	☐	☐
I was able to do my job easily while using the system	☐	☐	☐	☐	☐	☐
The functionalities are well organized	☐	☐	☐	☐	☐	☐
There were inconsistencies in the system	☐	☐	☐	☐	☐	☐
Learning and using the system take too much time in relation to the benefits of using it	☐	☐	☐	☐	☐	☐

OPEN QUESTIONS

Have you identified any issue during the execution of the Use Case? If yes, please report them and for each one rate it as high, medium or low priority.

Issue	**High**	**Medium**	**Low**
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐
	☐	☐	☐

Based on the executed Use Case, list suggestions for the improvement of S&R DSS.

____

____

____

____