Specification of the HAMS

Deliverable 6.10

Lead Author: LINKS

Contributors: CSI, ASLTO5

Deliverable classification: PU

Co-funded by the Horizon 2020 Framework Programme of the European Union

Version Control Sheet

Title	*Specification of the HAMS*
Prepared By	*Francesco Lubrano / Giuseppe Varavallo / Fabrizio Bertone*
Approved By	
Version Number	*1.0*
Contact	francesco.lubrano@linksfoundation.com/ giuseppe.varavallo@linksfoundation.com/ fabrizio.bertone@linksfoundation.com

Revision History:

Version	Date	Summary of Changes	Initials	Changes Marked
0.1	29/07/2019	First Draft	LINKS	
0.2	27/08/2019	Internal review	CSI, PMS	
1.0	02/09/2019	Finalizing comments and remarks from internal reviewers; added JSON example for communication between HAMS and Data exchange layer	LINKS	

The research leading to these results has received funding from the European Union's Horizon 2020 Research and Innovation Programme, under Grant Agreement no 787002.

Contents

The SAFECARE Project 5

Executive Summary 6

1. Introduction 7
   1.1 Deliverable 6.10 7
2. State of the art 8
3. EDXL standard 11
   3.1 Oasis consortium 11
   3.2 Overview of EDXL standards 11
   3.3 HAVE standard 12
      3.3.1 HAVE elements 13
4. HAMS internal architecture 17
   4.1 Description of the back-end 18
   4.2 Description of the front-end 19
   4.3 Interactions with other SAFECARE modules 20
5. HAMS graphical user interface 21
   5.1 Login 22
   5.2 Dashboard 23
   5.3 Manual reporting 24

6. Users and different permission levels 27
7. Conclusion 28
References 29

LIST OF FIGURES

FIGURE 1 SAHANA DISASTER MANAGEMENT SYSTEM INTERFACE 8
FIGURE 2 DASHBOARD HERAMS IN GAZA STRIP 9
FIGURE 3 HAVE ELEMENT REFERENCE MODEL 14
FIGURE 4 HAMS INTERNAL ARCHITECTURE SCHEMA 18
FIGURE 5 HOME PAGE HAMS 21
FIGURE 6 HAMS - GENERAL INFORMATION OF HOSPITAL 22
FIGURE 7 AVAILABILITY OF DEPARTMENT RESOURCES 23
FIGURE 8 RESOURCE AVAILABILITY IN DEPARTMENT 24
FIGURE 9 FORM TO REPORT AVAILABILITY OF RESOURCE 25

LIST OF TABLES

TABLE 1 OVERVIEW OF EDXL STANDARDS 11

LIST OF ACRONYMS

API	Application Programming Interface
EDXL	Emergency Data Exchange Language
EMS	Emergency Management System
GUI	Graphical User Interface
HAMS	Hospital Availability Management System
HAVE	Hospital AVailability Exchange
OASIS	Organization for the Advancement of Structured Information Standards
XML	eXtensible Markup Language

The SAFECARE Project

Over the last decade, the European Union has faced numerous threats that quickly increased in their magnitude, changing the lives, the habits and the fears of hundreds of millions of citizens. The sources of these threats have been heterogeneous, as well as weapons to impact the population. As Europeans, we know now that we must increase our awareness against these attacks that can strike the places we rely upon the most and destabilize our institutions remotely. Today, the lines between physical and cyber worlds are increasingly blurred. Nearly everything is connected to the Internet and if not, physical intrusion might rub out the barriers. Threats cannot be analysed solely as physical or cyber, and therefore it is critical to develop an integrated approach in order to fight against such combination of threats. Health services are at the same time among the most critical infrastructures and the most vulnerable ones. They are widely relying on information systems to optimize organization and costs, whereas ethics and privacy constraints severely restrict security controls and thus increase vulnerability. The aim of this proposal is to provide solutions that will improve physical and cyber security in a seamless and cost-effective way. It will promote new technologies and novel approaches to enhance threat prevention, threat detection, incident response and mitigation of impacts. The project will also participate in increasing the compliance between security tools and European regulations about ethics and privacy for health services. Finally, project pilots will take place in the hospitals of Marseille, Turin and Amsterdam, involving security and health practitioners, in order to simulate attack scenarios in near-real conditions. These pilot sites will serve as reference examples to disseminate the results and find customers across Europe.

Executive Summary

This document is part of WP6: "*Integrated cyber-physical security solutions*" and contains the specification of the Hospital Availability Management System (HAMS) to be later implemented in continuation of Task 6.6.

The document is structure as follows:

Chapter 1 introduces the context of this deliverable and the objectives of the task.

Chapter 2 contains state of the art analysis for projects implementing similar HAMS services, with a focus on those using the EDXL-HAVE standard.

Chapter 3 lists and describes the set of open standards "EDXL" designed for the management of emergencies. Since hospitals typically hosts a wide range of different assets with different characteristics, a common description language needs to be used in order to exchange information with external actors. EDXL-HAVE standard was chosen to report availability information between hospitals, to emergency health care providers and other concerned actors (rescuers, etc).

Chapter 4 describes the internal architecture of the HAMS module and the interactions with the external modules developed within SAFECARE project.

Chapter 5 describes the graphical interface used to access the functionalities made available to different kind of users.

Chapter 6 lists and describes the user roles that can access the HAMS with different levels of available functionalities. The three classes of roles identified are: *Public*, *Privileged User* and *Administrator*.

Chapter 7 summarizes and concludes the deliverable.

1. Introduction

In a disaster or emergency, it is important for hospitals to be able to communicate with each other and with emergency health care providers about their availability and capacity. With this information, first responders are capable to better manage the flow of patients, improving the health services quality and the emergency management.

Effective management of emergencies and crises depends on timely information availability, reliability and intelligibility, therefore having a fast communication of incidents and a subsequent processing of availability is a key point, in order to provide relevant information as soon as possible, giving to emergency managers the possibility to take more accurate decisions as soon as possible.

The development of Hospital Availability Management System (HAMS) service in the context of SAFECARE will improve the resilience of health services and the communication of availability information among hospital staff and first responders.

Due to the wide and heterogeneous set of assets that can be found in a hospital, and the different users that should access this information, it's mandatory to find a common protocol/language to exchange data about availability. Concerning the SAFECARE project, for this task we identified a standard developed by OASIS (Organization for the Advancement of Structured Information Standards) consortium, called EDXL-HAVE (Emergency Data Exchange Language - Hospital Availability Exchange). This standard defines a common data model capable of describing all the assets, facilities, services and buildings in a hospital, in order to provide a complete overview of hospital status and availability.

Implementing the EDXL-HAVE standard, HAMS will be able to share data with any user/software compliant with the standard. HAMS will implement also a web interface in order to show availability data and relevant information also through a graphic interface.

1.1 Deliverable 6.10

The aim of Task 6.6 is to design and implement the HAMS, an innovative software that can improve health service resilience and availability, showing relevant information regarding availability of assets and services in hospital environment. HAMS will be completely integrated in SAFECARE environment, implementing interfaces to interact with the communication layer (T6.2 Data exchange layer) present in the global architecture.

This document, Deliverable 6.10, reports the specification of the HAMS and will be used as guideline during the implementation of the software. This document provides a description of similar projects and software in the state-of-the-art section. Then it describes the EDXL-HAVE, the standard on which the HAMS is based. At the end there are three chapters dedicated to the HAMS specification: the first related to the internal architecture of this module and the interconnection with other SAFECARE software; the second related to the description of the first version of the graphical interface of the HAMS; the third related to the users that will use this software and the ways in which different users could access availability data.

2. State of the art

This section will provide a description of other management systems that reports hospital availability, developed in similar use cases and projects.

In the following sections has been described some of most relevant projects that implements hospital availability management systems. Information about these projects has been gathered with CORDIS (Community Research and Development Information Service), a European platform that provides information about EU projects, in addition to traditional websites such as Google Scholar and Elsevier.

SAHANA Disaster Management System

One of the first software implemented with EDXL Standards is the SAHANA Disaster Management System (DMS) [1] [2]. Sahana DMS system was used in 2010, in the case of earthquake struck in the Caribbean nation of Haiti [3], in the city of Port-au-Prince. Tens of thousands of civilians were killed instantly, while thousands more were trapped beneath rubble, as well as leaving millions of civilians homeless. Figure 1 represents the principal interface of Sahana providing access to a collection of interconnected yet independent components. These components interact with each other via a set of shared databases to provide a range of Web-based information services.

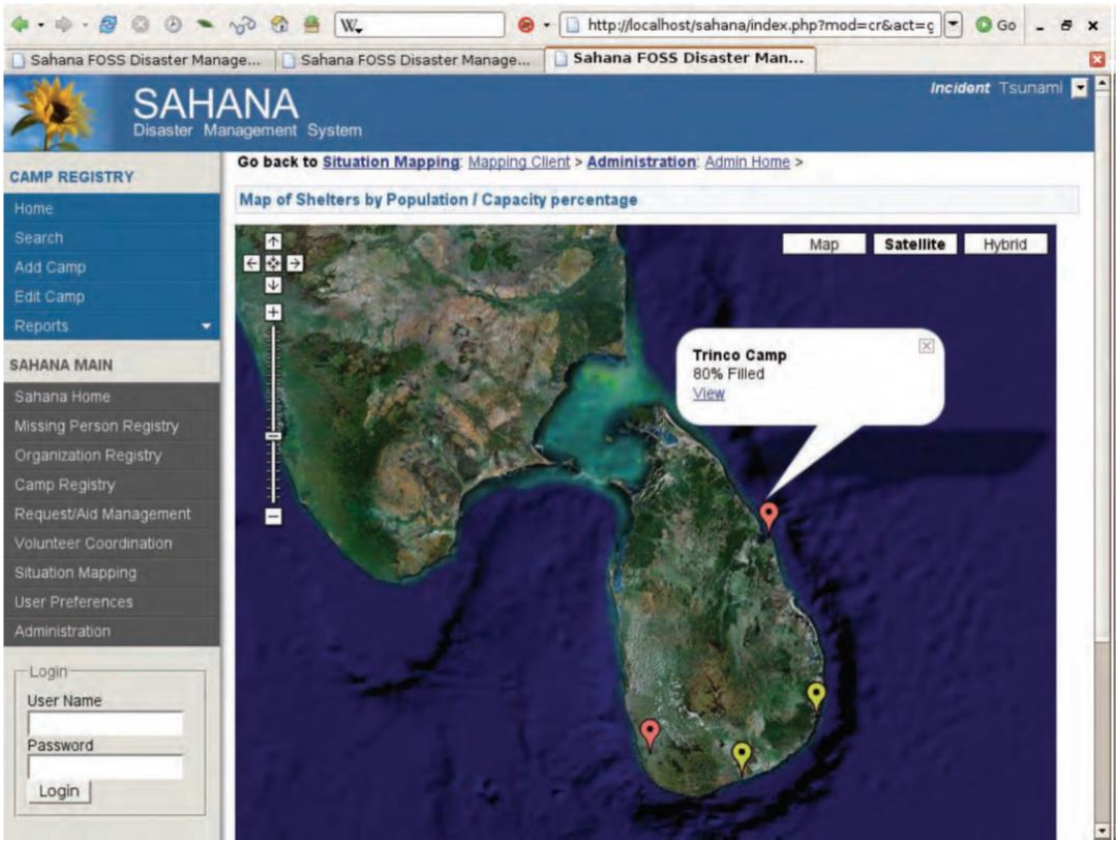

Figure 1 SAHANA disaster management system interface

The EDXL-HAVE protocol was used to structure data regarding hospitals availability to route the victims in Haiti. The data was received to Emergency managers from different hospitals using HAVE successfully.

National Disaster Medical System patient movement

In the description of HAVE standard, OASIS reports a use case. The HAVE standard has been used successfully in the United States to exchange information among independent systems at different geographical level: the emergency medical system (EMS) of the city of Memphis, the states of Maryland and Tennessee and the federal JPATS system. During a test done in 2010 for the National Disaster Medical System (NDMS) [4], HAVE standard was used to exchange data about patient movement between States. The data was transferred between the three EMS systems and the federal JPATS system in real time. All systems shared HAVE data from 3 hospitals, allowing emergency managers to better route patients among neighboring hospitals.

Health Cluster (WHO) - Health Resources Availability Management System (HeRAMS) in Gaza Strip

Another relevant project related to the management of hospital availability is the Health Cluster developed by World health organization in the occupied Palestinian territory. The Health Cluster [5] collectively prepares for and responds to humanitarian and public health emergencies to improve the health outcomes of affected populations through timely, predictable, appropriate and effective coordinated health action. Figure 2 shows the interface of Hospital availability management system implemented in the Gaza Strip. The Dashboard is rich of information about public hospitals, staff, beds, etc. It is also possible to interact and navigate in the dashboard visualizing data for each district and hospitals about staff, services, equipment, and drugs. The interface is very clear to understand in a short time the situation of public hospitals in each district and visualize information needed to mitigate emergency or crisis situation.

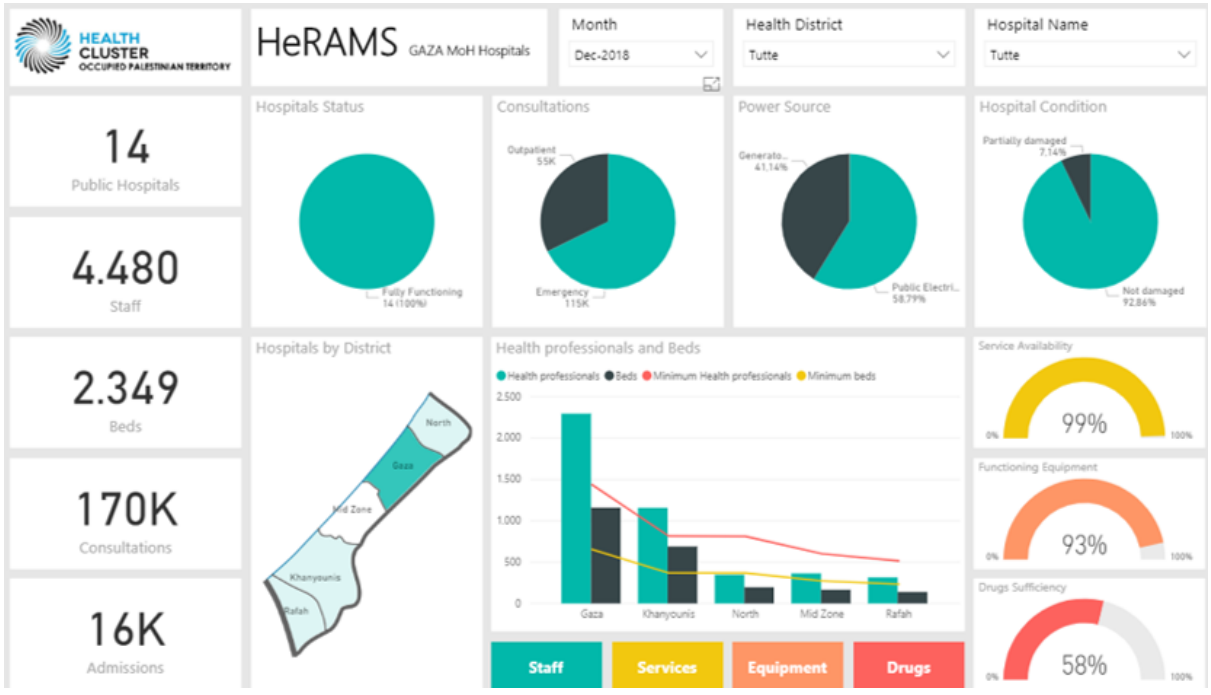

Figure 2 Dashboard HeRAMS in Gaza Strip

The IMPRESS (Improving Preparedness and Response of Health Services in major criseS) project

Regarding the CORDIS website, one of the most relevant EU projects in hospital availability management field is IMPRESS (2014-2017) project [6].

The IMPRESS (Improving Preparedness and Response of Health Services in major criseS) project aims to advance the preparedness of emergency medical services (ambulance dispatch centers,

hospitals, volunteer communities, etc.) in numerous ways, including planning for all-hazards, increasing surge capacity, tracking the availability of beds and other resources using electronic systems, and developing systems that are interoperable with other response teams. The IMPRESS general objective is to provide preparedness and response capabilities through guidelines and tools where ultimately, the routine use of these capabilities will sharpen their application in larger disaster scenarios. In the IMPRESS project was implemented the Hospital Availability with EDXL-HAVE like plugin in a sub-system through which hospitals or other health care institutions can exchange general facility and resources availability information. This information exchange is of primary importance during mass casualty incident situations by assisting decision making of medical personnel about patient's evacuation. Data about the hospital availability are entered by the hospital operators that report the bed availability, staff availability and service available to the crisis center and first responders.

In this case, operators usually receive a request from another hospital or emergency call center and answer the request reporting the availability of the hospital.

3. EDXL standard

3.1 Oasis consortium

OASIS [7] is a nonprofit organization that promotes and produces a worldwide standard for cloud computing, security, Internet of Things, energy, content technologies, emergency management, and other areas. OASIS is organized in technical committees and the standard produced is open. The goal is to bring people and organizations to exchange information on intelligent ways by adopting standards.

3.2 Overview of EDXL standards

EDXL is a group of standards adopted by OASIS to manage the entire emergency lifecycle. It is designed to exchange and share information easily between different emergency systems.

The following are the most relevant EDXL standards:

EDXL standard	Key features
Common Alerting Protocol (EDXL-CAP)	emergency alerts notifications public warnings
Distributed Element (EDXL-DE)	routing of any emergency information
Resource Messaging (EDXL-RM)	collect data manage logistic information
Hospital Availability Exchange (EDXL-HAVE)	hospital status service availability resource availability
Tracking of Emergency Patients (EDXL-TEP)	emergency patient data EMS tracking information

Table 1 Overview of EDXL standards

EDXL-CAP [8] helps different system to communicate in real-time the alert messages, this protocol facilitates the task of notification in the case of emergency. EDXL-DE [9] organizes the content of messages (as a container for structured or unstructured data) from other standards to identify the senders and targeted recipients of the message. EDXL-RM [10] collect data and organize emergency logistics information in a standard XML dictionary and includes pre-defined messages that cover the fields of logistics-related resource messaging- in the request-response-report pattern. EDXL-HAVE [11] enables the communication of the status of Hospital, its services, and its resources. This covers bed capacity per department status, available services and the status of a hospital's facility and operations. Have supported emergency dispatchers and managers to make logistics decisions on where route patients to ensure the receiving Hospital is prepared to take care of them. EDXL-TEP [12] provide a standard dictionary to collect data

emergency patient data to share it to emergency managers for tracking and receiving hospitals. This information is distributed in real-time so the emergency responders, coordinating organizations, and care facilities can organize better the chain of resource and transport.

Thanks to the EDXL Standards, first responders communicate efficiently in emergencies with operational centers, which can provide them a higher level of situational awareness, more resource for the response, and the ability to mobilize the citizens with an alert or warning helping to make better decisions.

3.3 HAVE standard

EDXL-HAVE (HAVE) [13] is a XML messaging standard developed by OASIS in the context of the emergency management. This type of standard provides information about the availability of hospitals and health networks. In particular, the information exchanged is about facility services, bed counts, capacities, operations and resource, so that the emergency managers, hospitals, first responders and care facilities can have a complete view of each other's availability of health system resources.

The HAVE standard was created to share information about the status of hospital for day-to-day use and in emergency case. The first version of HAVE 1.0 [14] was focused only on hospitals, while in HAVE version 2.0 the functionalities have been extended to the whole health network, including temporary clinics and urgent care clinics.

In emergencies or crises, it is important for the hospitals to share information with members of the emergency network. The capacity to exchange data regarding hospitals bed counts, status, services, and capacity allows the hospitals to manage the emergencies and crisis with greater efficiency. This information about availability helps the emergency manager to decide better where to route patients or victims and determining automatically which hospitals have the needed services. Usually, this kind of information is already available in the hospitals that uses a self-developed information technology that enables to publish this information in a region, Emergency operation center, 911 centers. The issue of such systems is that the data might not be recorded in a standardized format.

The HAVE standard was conceived to solve this issue: share information with a standard format with different hospitals and health network, including urgent care clinics, and temporary centers.

HAVE messages are designed to be payloads of several messaging and/or delivery systems. Messaging system such as EDXL-DE can use HAVE message as a payload. Furthermore, non-message-based system (e.g. RESTful web service) can deliver HAVE messages just as easily. An individual facility may provide an up-to-date report via a web service. An aggregator could poll that facilities that are of interest for a particular reason, or in a publish/subscribe scenario, subscribe to the facilities of interest.

The principles that guided the design of the HAVE include:

- **Interoperability** – The HAVE message should provide an interoperable mechanism to exchange healthcare organization information among different domains and among multiple system.

- **Multi-Use Format** – The HAVE message must be designed such that it can be used in everyday events, during mass disasters and for incident preparedness.
- **Flexibility** – The design structure must be flexible such that it could be used by broad range of applications and systems to report status and availability information.

3.3.1 HAVE elements

A HAVE schema consists of a root element that uniquely identifies the organization that is responsible for the reporting facilities. Each facility is described through a list of sub-elements that allow a complete description of hospital departments, services and resources. Figure 3 shows the Element Reference Model and the elements of EDXL-HAVE.

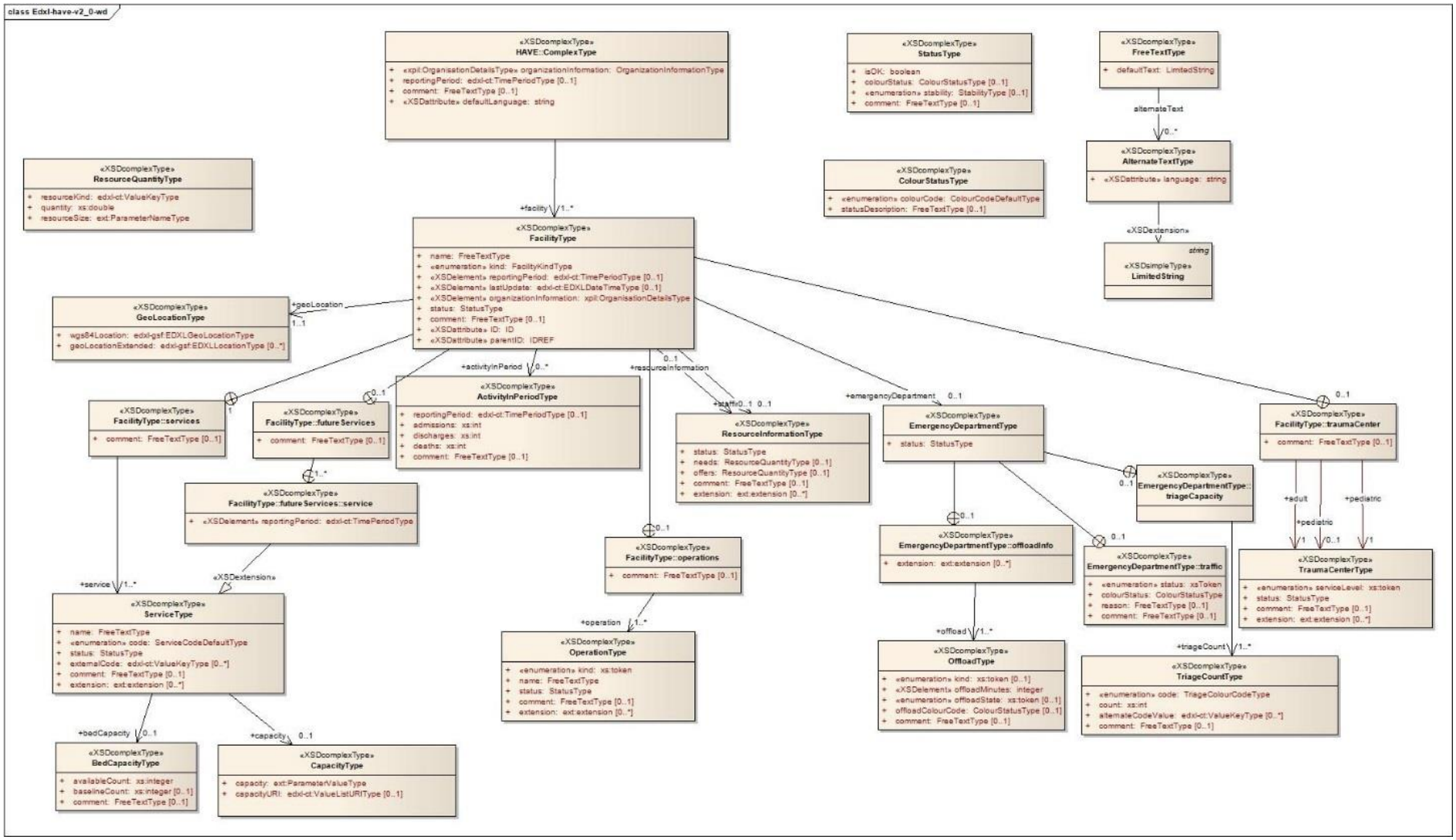

Figure 3 HAVE Element Reference Model

Below EDXL HAVE core elements are listed and described. This description of elements aims at providing to the reader a general knowledge of HAVE standard. Particular reference has been made to those elements involved in the development of HAMS application.

<HAVE> is the top-level container element for Hospital Availability Exchange (HAVE) message. HAVE element has the following set of sub-elements:

- the **<organizationInformation>** element provides basic information about the name and location of the organization for which the status and availability is being reported; each organizationInformation element is an instance of **OrganizationInformationType**;
- the **<reportingPeriod>** element provides information about the period to which the report refers. If this element is left blank, the assumption is that the file refers to the last 24h.
- the **<facility>** element consist of a list of facilities related to the hospital or organization.

Each **<facility>** element is an instance of **FacilityType** and contains a set of information that defines a facility. This information is represented by the following sub-elements:

- the **<name>** of the facility;
- the **<kind>** of facility (e.g. Hospital, Long Term Care, Senior Residence, Temporary Clinic);
- the **<organizationInformation>** reports administrative and organizational information about the Facility;
- the **<reportingPeriod>** applicable for this facility. If left blank the HAVE element reporting period will be assumed also for the facility;
- the **<geoLocation>** provides accurate geospatial information about facility location;
- the **<status>** reports the overall status of the facility from the perspective of the person responsible for it. Each status is an instance of the **StatusType,** composed by four sub-elements (isOK, colourStatus, stability and comments);
- the **<services>** is a container element of all elements of service coverage, including both the necessary staff and facilities. Each element in this container is an instance of **ServiceType**.

Each **<service>** is represented by a set of sub elements listed below:

- the **<name>** of the service;
- the **<code>** uniquely defines and represents the service;
- the **<status>** reports the status of the service and is an instance of **StatusType**;
- the **<bedCapacity>** reports bed capacity of a service. It is an instance of the **BedCapacityType**, that contains three sub-elements:
    - **<availableCount>** contains the number of vacant/available beds;
    - **<baselineCount>** contains the total amount of beds;
    - **<remarks>** provides context for the **BedCapacityType;**
- the **<capacity>** is a container to report information of further resources. It is an instance of **CapacityType;**

Additional sub-elements that are inside the **FacilityType** are:

- the **<resourceInformation>** reports information of resource state (status, needs, offers);
- the **<staffing>** reports information of staffing status;
- the **<emergencyDepartment>** reports information related to the status of emergency department. It is an instance of **EmergenyDepartmentType**, that contains following sub-elements:
    - status;
    - offloadInfo, information about the offload state of various modes of transport such as ambulance, air ambulance;
    - traffic, contains additional information related to the emergency department triage;
    - triageCapacity, the number of each triage patient type the hospital can accept;
- the **<traumaCenter>**, the type of the trauma center for the organization;

4. HAMS internal architecture

Considering the need to have a module that interact with the SAFECARE ecosystem and at the same time an application with a graphical interface that can run on a wide range of different devices, HAMS module has been designed as a web application. A web application is an application designed and developed following the client-server paradigm, providing communication interfaces over widely supported standard web protocols.

The server side runs in a remote server/datacenter and exposes some external APIs, allowing the interaction with multiple user clients at the same time. The client side, that consists in the user interface and client-side logic, runs in a web browser or other compatible applications.

In the Figure 4 HAMS internal architecture schemais depicted a high-level overview of the internal structure of the HAMS module and how HAMS is interconnected with the others SAFECARE modules through the Data exchange layer, a communication layer that allows module developed in SAFECARE project to exchange information each other and interact with the Central database. Detail and specification of the Data exchange layer will be reported in the deliverable D6.2. HAMS will use the Central database to get data about hospital assets and resources. Indeed, the Central database will store all the static data collected in the three demonstration sites of the project in addition to storing dynamic data such as incidents, impacts, etc. Details and specification about Central database will be reported in the deliverable D6.4.

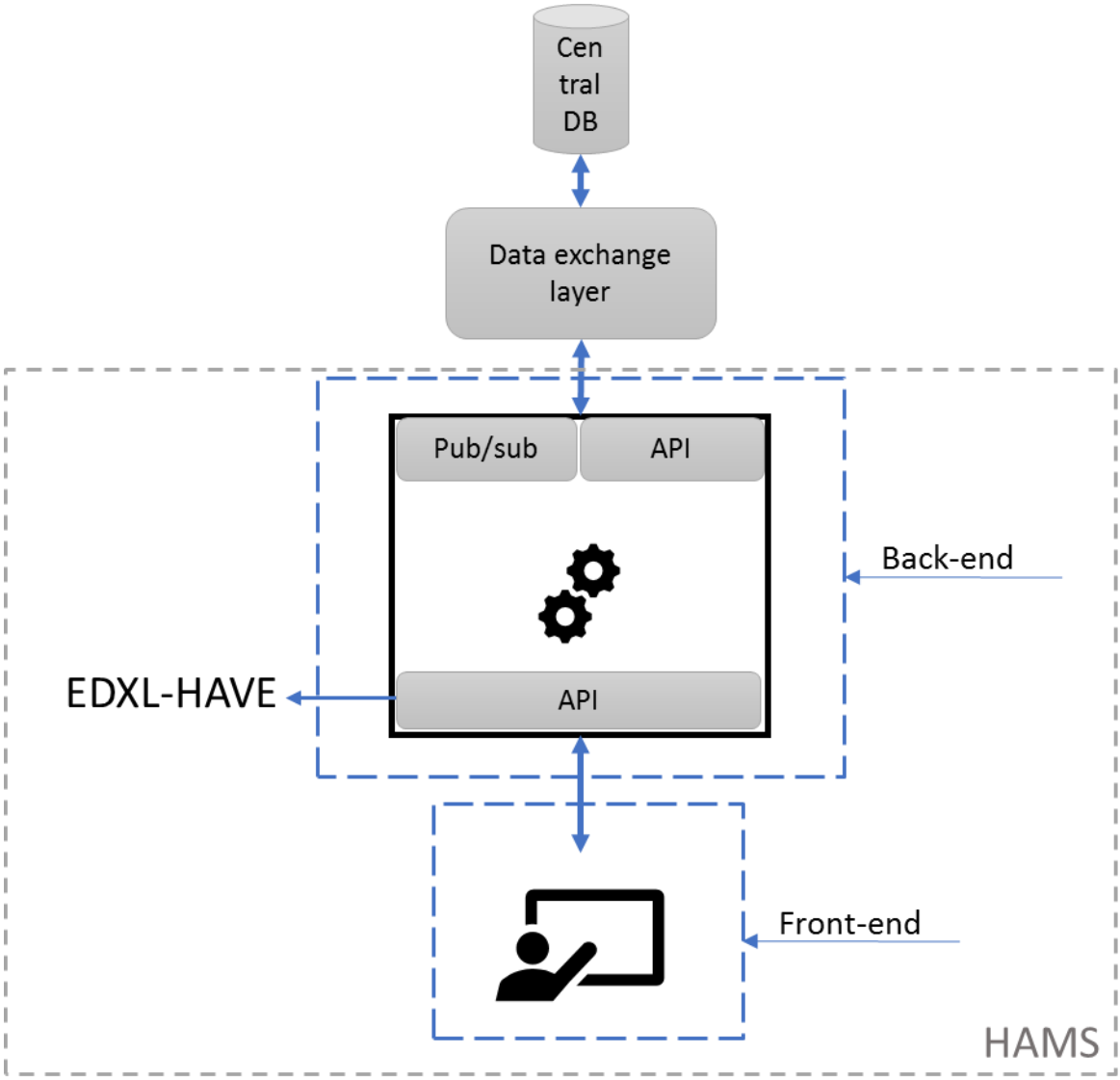

Figure 4 HAMS internal architecture schema

4.1 Description of the back-end

The server side of the HAMS module is called back-end. As stated above, the back-end is in charge of interact with the SAFECARE world through the Data exchange layer, that is devoted to providing the communication between modules.

Communication with Data exchange layer

Data exchange layer will provide communication through two different approaches:

- publish/subscribe mechanism, provided by the Data exchange layer to support real time communication of incidents and impacts. Through the implementation of an internal client, HAMS will be able to subscribe to relevant topics in order to receive incidents and impacts and evaluate new services and assets availability. If needed, HAMS will also be able to publish messages;
- REST APIs, provided by the Data exchange layer to allow modules to interact with the Central database. HAMS will use these APIs to get information related to assets and to

store or to update data in the database. HAMS will allow administrator users to manually report the availability of some relevant assets in the hospital. This functionality needs to store or update data in the database, so a dedicated API will be created in order to satisfy this requirement. This API will be used also to automatically update the status of assets involved in an incident.

The main function of the HAMS back-end consists on receiving messages containing incidents or impacts from the Data exchange layer and evaluate the availability of assets involved in the incident. Each incident is related to a set of involved assets. HAMS will automatically report the availability of an asset checking the previous status and updating it in the central database. Incidents can involve a large number of different assets and HAMS will report availability only for assets like departments, health services, medical devices.

Communication with HAMS front-end

Therefore, HAMS back-end is in charge to provide information to the front-end. For this reason, it will expose some REST APIs for the following functionality:

- providing availability data for each asset in order to visualize it
- managing user authentication
- receiving asset availability status from the front-end in order to update asset status in Central database

Export of HAVE document

Finally, HAMS back-end will expose a dedicated API to export the status of the hospital in the HAVE format. Internally the HAMS back-end will implement classes to represent the objects present in the HAVE schema (see section 3.3.1). In this way HAMS will be able to manage internally the data related to hospital assets availability and, if requested, will build and share a document compliant to the EDXL-HAVE standard that represents the status of the hospital.

4.2 Description of the front-end

The front-end is the part of HAMS devoted to user interaction and data visualization. This part consists of multiple graphical views that show information related to hospital general status and availability status of each relevant asset. This part has been designed as a web interface that is connected to the back-end. As stated in section 4.1, the back-end exposes a set of APIs that can be reached by the front-end in order to get and send data. This means that the HAMS front-end can fetch data from the back-end in order to visualize it through more appropriate graphical means such as tables, charts, etc.

To do that, the front-end will have an internal logic capable to manage and reorganize data, before visualization. This internal logic will manage the internal status of the front-end.

The web interface will also provide the possibility to manually report the status of an asset through a dedicated page. Information entered by the user will be properly formatted before sending them to the back-end.

Considering the importance of the information visualized and the disposition of elements in the page, the graphical interface has been designed taking into account comments and remarks of health practitioners.

4.3 Interactions with other SAFECARE modules

As depicted in Figure 4, HAMS is connected to the SAFECARE ecosystem through the Data exchange layer. This interconnection is of primary importance; indeed, it allows HAMS to interact with the Central database and to receive messages from other modules. In particular, HAMS will receive incidents, the output of monitoring systems developed in the work package 4 and work package 5. Incidents report a security event associated to one or more assets, specifying the type of incident, the severity level, the timestamp and the list of assets confirmed to be impacted by the incident, HAMS will update the availability of involved assets, if they are a department or a service. HAMS will also receive the output of the *Impact propagation model and decision support model*, in order to update the availability of assets indirectly involved in an incident.

Once the HAMS module detects a change on the status of one or more asset, it will send a request to the data exchange layer in order to update stored data in the Central database. Below an example of the JSON that the HAMS can send to the data exchange layer. NB. This is only an example, the final JSON structure could be different.

```
{
  "RequestID": "abcd1234",
  "timestamp": 1567432951,
  "resources": [
    {
      "id": "0000001",
      "type": "asset",
      "status": "available"
     },
     {
      "id": "0000005",
      "type": "department",
      "status": "available",
      "bed_availability": 10,
      "staff_availability": 5
     }
  ]
}
```

5. HAMS graphical user interface

The graphical user interface (GUI) of HAMS consists different views and elements. The core elements of the GUI are:

- map to visualize the location of involved hospital buildings;
- dashboard to visualize the availability of departments, medical devices, beds, staff;
- form to insert or update data related to the status of departments, medical devices, beds, staff.

Figure 5 shows the home page of Hospital Availability Management System. The interface is accessible to all users, showing a general map, the status of the Hospital, descriptive information, and resources. At this step, it is possible to select the Hospitals to check availability.

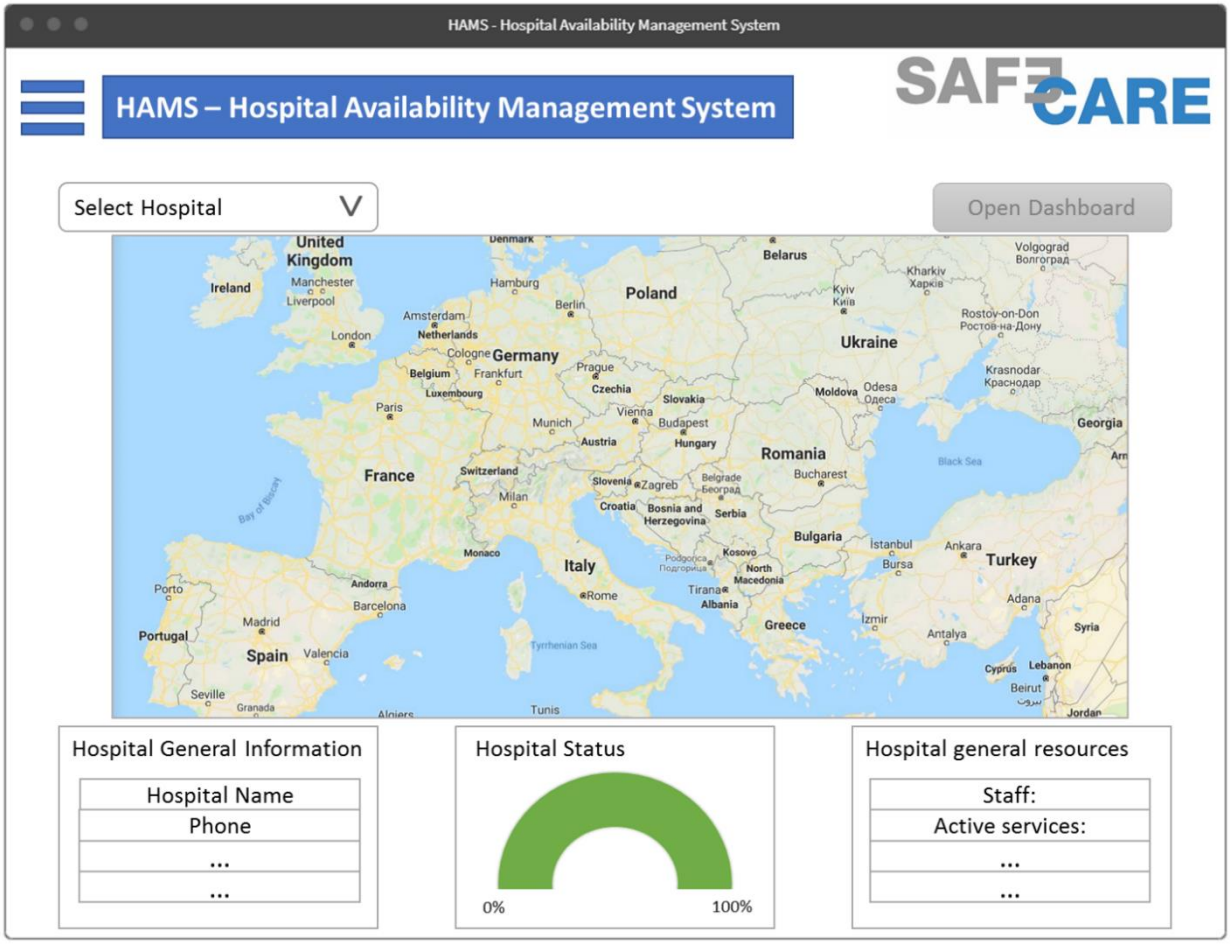

Figure 5 Home page HAMS

Once the user selects a hospital, through the drop-down menu, the map automatically updates to show location of selected hospital buildings (Figure 6).

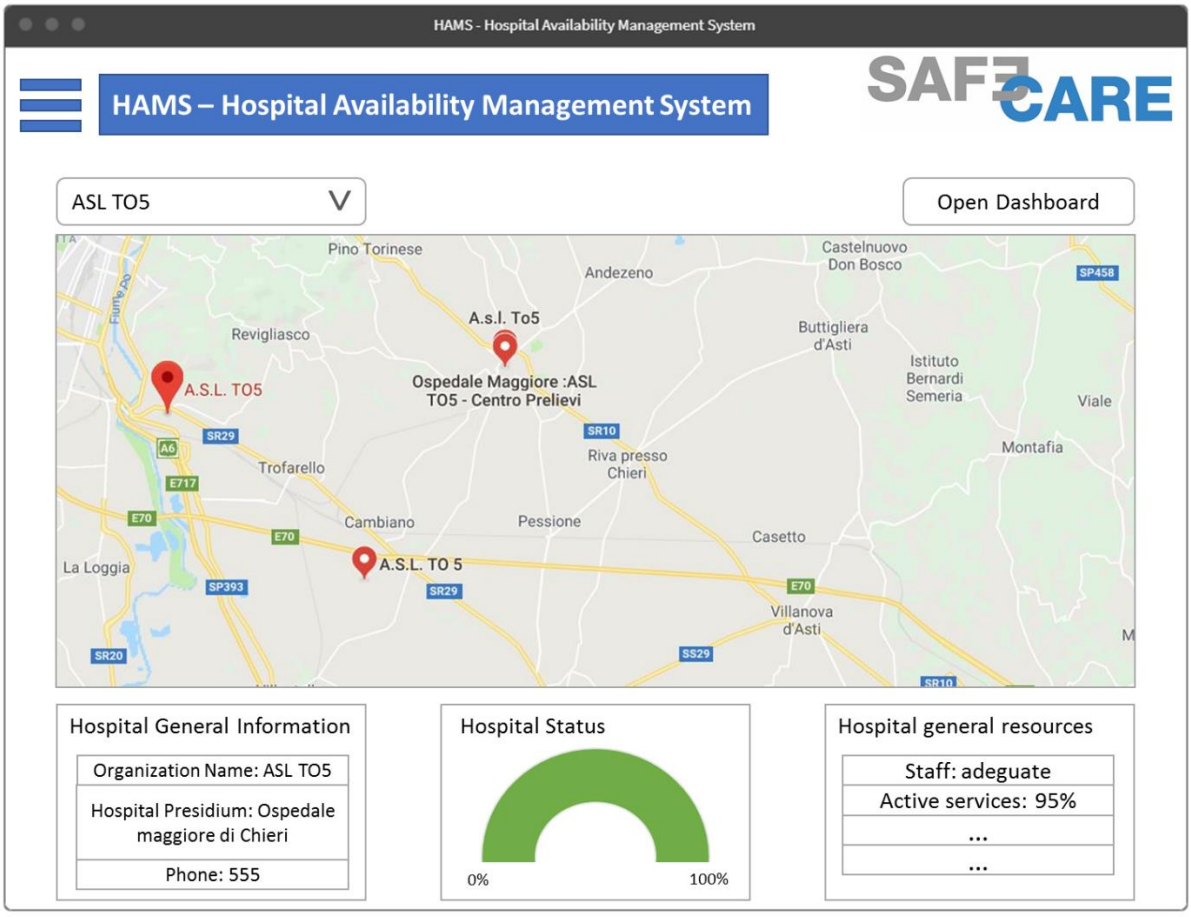

Figure 6 HAMS - General information of Hospital

General information is shown (name of Hospital, address, phone number), the status of the Hospital (in percentage) and the general resources of the hospital (staff, percentage of active services, etc.). Hospital status graph is a composite indicator that will show the general status of the hospital, made aggregating information of several assets.

In this view it is possible to click on "Open Dashboard" to login and access to the dashboard.

5.1 Login

The login page will appear automatically when unauthenticated users will try to access to the dashboard.

The login page is used to authenticate authorized users, allowing them to request, access and/or update non-public information from the HAMS back-end system. Users insert assigned credentials, the front-end take care of contacting the authentication service exchanging the authentication information. In case of success, the user is authorized to access the additional private sections of the HAMS interface, otherwise an error message is shown to the user. More details about different users are described in section 6.

5.2 Dashboard

The dashboard represents the main index page for accessing the information of a specific hospital. This page shows a high-level view of the availability status of all relevant resources in the hospital. This is a read-only reporting interface and it is mainly used to report updated information to authorized users.

Dashboard reports status of following elements:

- Department name and status
- Services
- Bed availability
- Staff availability
- Resources per department

The department name will report information about the single department and its status in terms of availability. The services will report the overall status of the elements to support the services of departments like availability of beds, staff and other resources.

The dashboard in Figure 7 reports information about the department availability. In details, the bed availability reports information about the overall beds (baseline) per department and the number of beds available. The staff availability follows the same process, there is the overall staff (baseline) per department and the number of staff available.

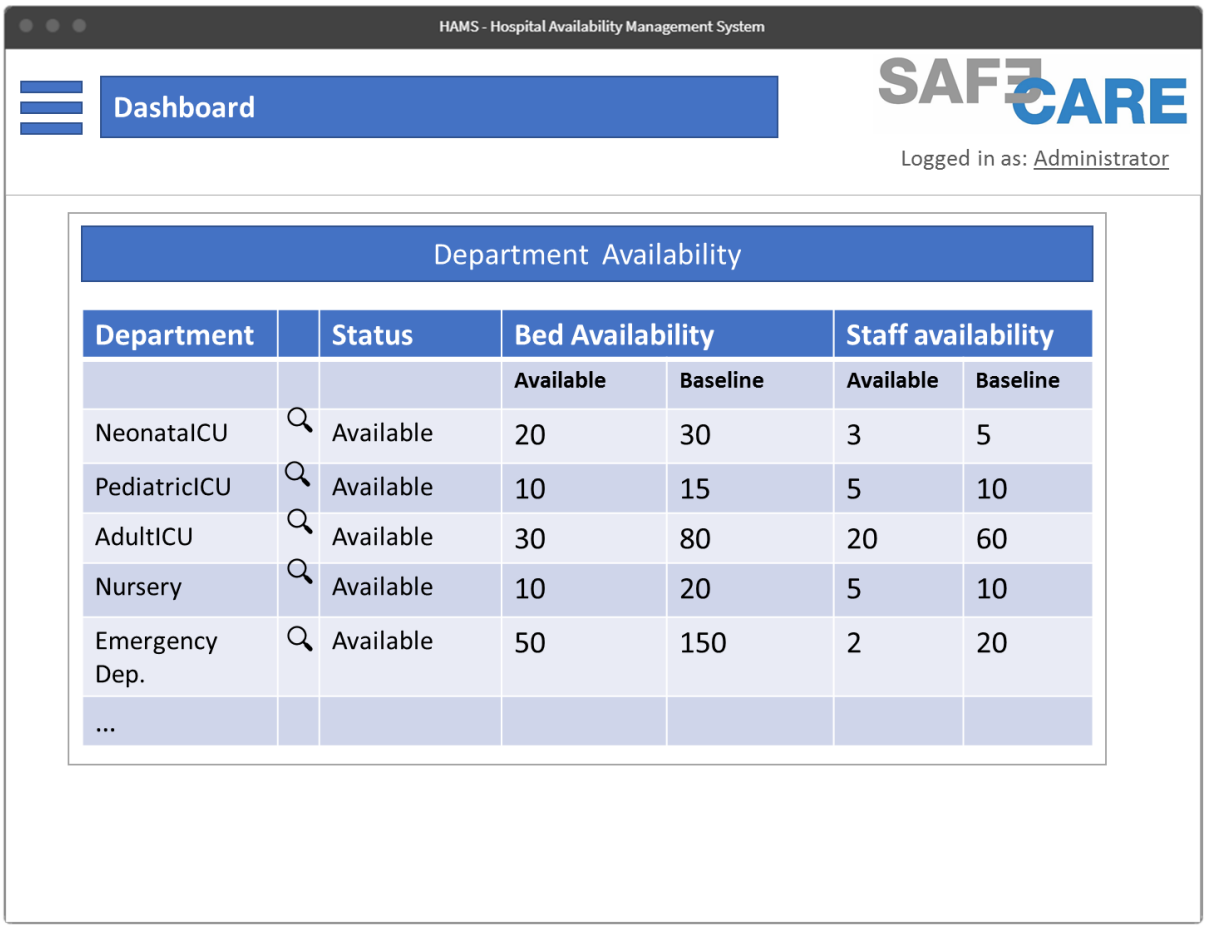

Figure 7 Availability of department resources

The button represented by the magnifying glass per each department helps to investigate the availability of medical devices and other types of resources. By clicking on the magnifying glass it is possible to view the status of the technical resources of the individual departments (Figure 8).

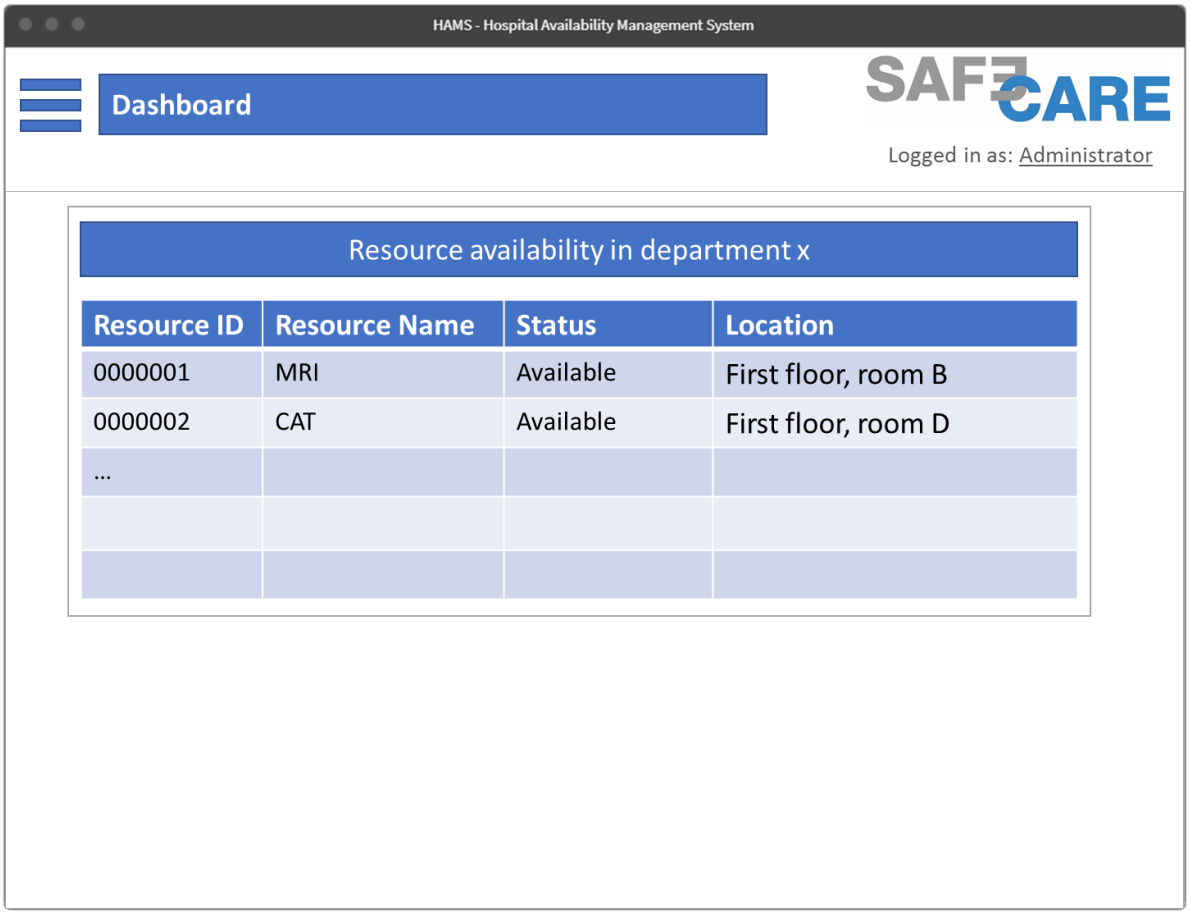

Figure 8 Resource availability in Department

5.3 Manual reporting

This section of the dashboard allows authorized users to update the status of resources. It is only accessible after login and just for users with a specific role (such as administrator).

To manually update an element, the user can choose a department or a resource from the HAMS dataset. The HAMS dataset consists of all the hospital resources present in the Central database.

Figure 9 shows the form to report the availability of departments. In particular, the status (Available / Not Available) and the number of beds and staff are reported manually to compare it with the baseline.

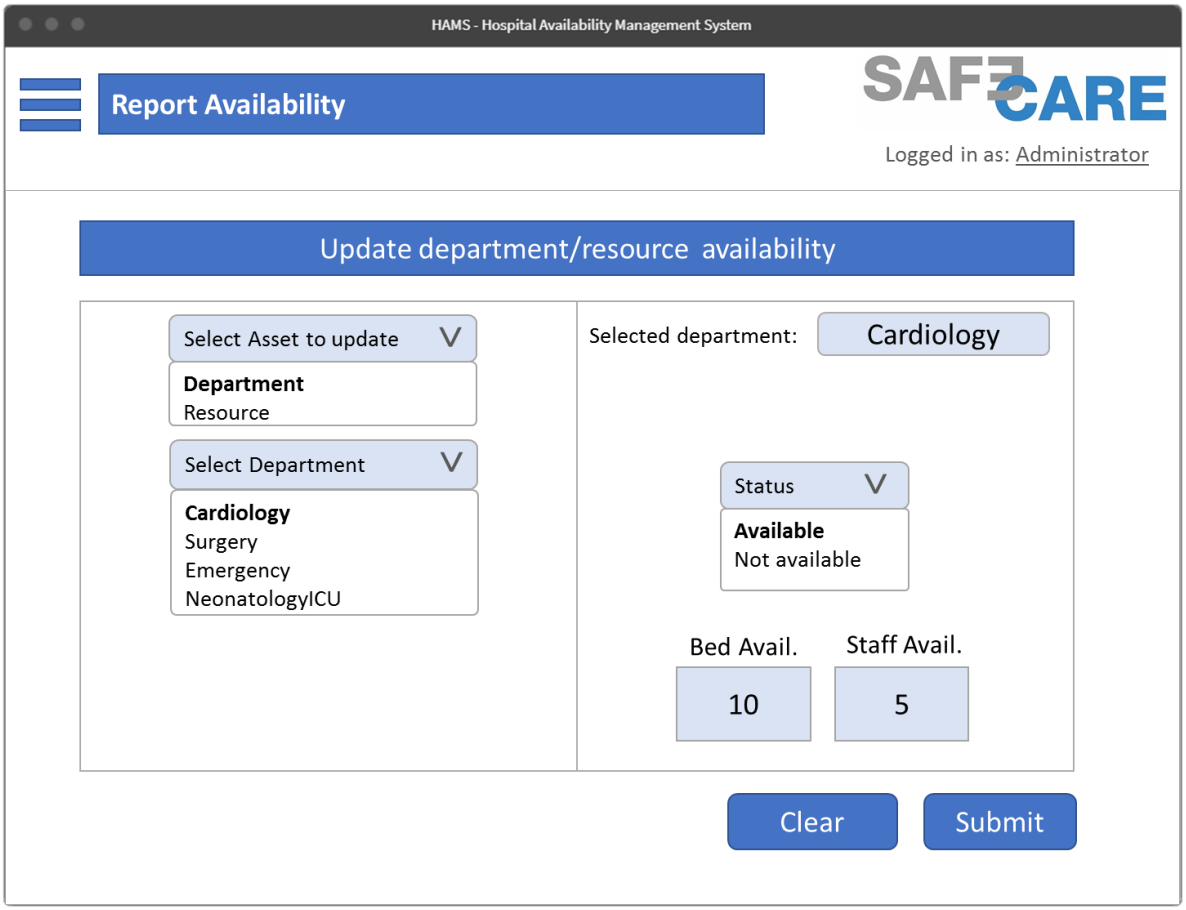

Figure 9 Form to report availability of department

Figure 10 shows the form to report the availability of resources belonging to a department. Once the resource option is selected, further fields become available to insert the new information into the system.

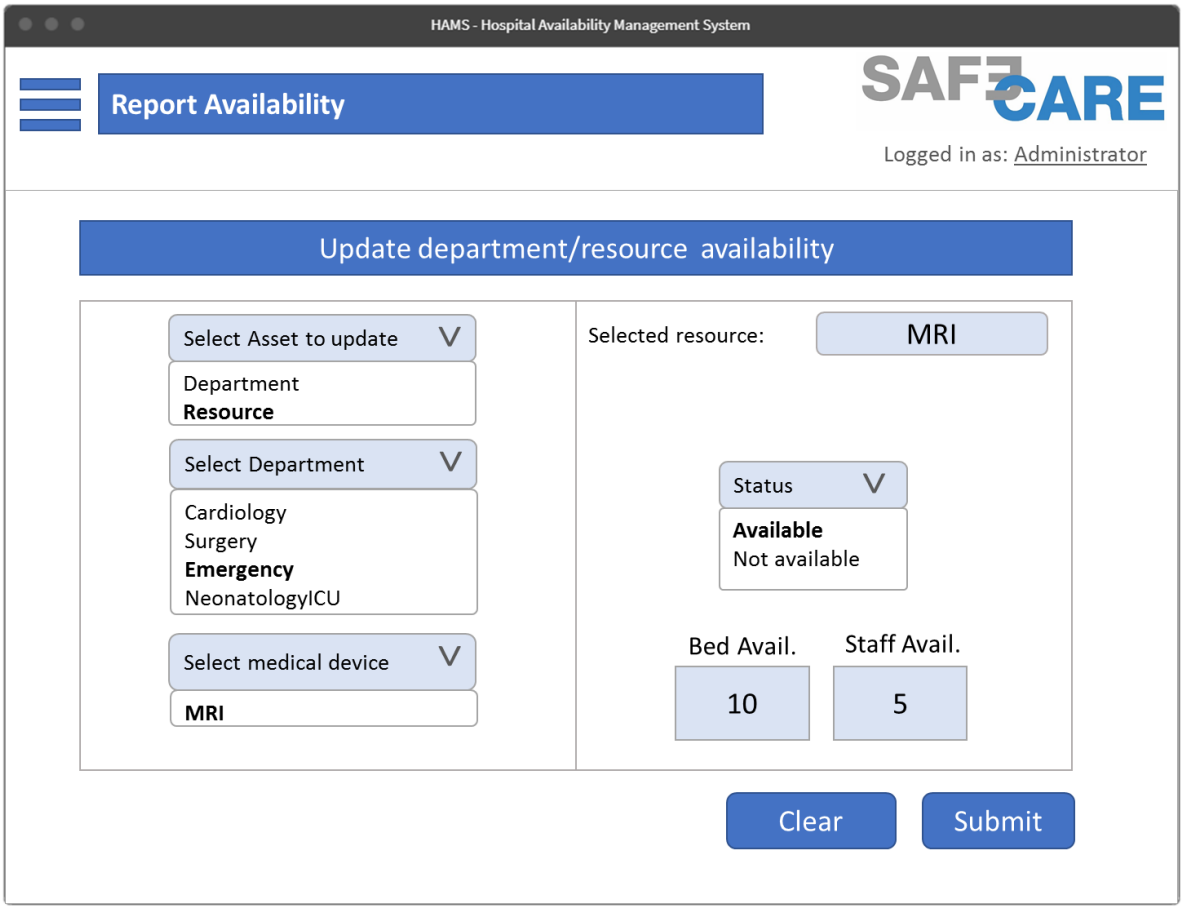

Figure 10 Form to report the availability of a resource

This data will be updated manually by administrators responsible of the HAMS. Once the operator clicks on submit button, the information entered will be sent to the Central database in order to update stored data. HAMS will also update the data reported by the GUI accordingly to new availability of assets. This section has been designed to allow operators to enter manually data related to availability of each asset stored in Central DB. Therefore, status of hospital assets can be updated automatically, when incidents or impacts are received by the HAMS, and manually, through the depicted interface.

6. Users and different permission levels

Considering its role, HAMS module is different from all the other modules that are being developed in SAFECARE project. HAMS is directly aimed to public users, (potential) patients and health practitioners. For this reason, particular importance was given to the design of graphical interface. Taking into account the requirements of HAMS, it has been identified 3 categories of different users that will provide different user experience and access level:

- Public
- Privileged user
- Administrator

The HAMS interface will have a home page (Figure 6), publicly accessible, that reports some general information about the hospital. Unauthenticated users will be able to access to this page and see general data of hospital, such as the name, the phone number, the location, etc.

Detailed information regarding assets, departments or resources is restricted to authenticated users.

For first responders and healthcare practitioners HAMS will provide a dashboard that reports all the relevant information about departments, services and resources available in the hospital after an authentication procedure. This is the case of privileged users, that will be created for anyone that has the rights to access to hospital detailed information.

HAMS interface will offer also a dedicated page where user can modify the status of each assets in order to update information stored in the Central database. This particular page will be accessible only to hospital staff in charge of update this information. These users are identified as administrators of the HAMS application.

7. Conclusion

In the SAFECARE architecture, the HAMS module is in charge of report the availability of hospital assets. This allows the optimization of information exchange between health practitioners and first responders during emergency. As result, HAMS improves the health service resilience and provides updates about assets and service availability.
This document reports the description of specification and functionality of the HAMS, providing also a comparison among other similar software.
Furthermore, a mockup of the graphical user interface has been designed, in order to give a first overview about the interface and the functionality.

This document represents a guideline for the development of the HAMS software and for the development of the other modules that will interact with the HAMS, such as the Data exchange layer.

References

[1] [Online]. Available: https://sahanafoundation.org/.

[2] M. Prustalis, P. Bodduluri, C. de Silva, D. Bitner, F. Boon, D. König e G. Treadgold, «The Sahana Software Foundation response to the 2010 Haiti Earthquake: A New Standard for Free and Open Source Disaster Data Management Systems,» 2010.

[3] N. Rencoret, A. Stoddard, K. Haver, G. Taylor e P. Harvery, «Haiti Earthquake Response – Context Analysis,» 2010.

[4] J.-Y. Ahn, K.-M. Ha e S.-H. Park, «National Disaster Medical Systems in the United States and Korea,» 2010.

[5] W. H. Organization, «Health Cluster Mission,» [Online]. Available: http://healthclusteropt.org/pages/1/what-is-health-cluster.

[6] I. INTERNATIONAL, «IMproving Preparedness and Response of HEalth Services in major criseS,» 2017. [Online]. Available: https://cordis.europa.eu/project/rcn/185510/reporting/en.

[7] Oasis. [Online]. Available: https://www.oasis-open.org/.

[8] G. Trott e E. Jones, «Emergency Data Exchange Language (EDXL) Common Alerting Protocol (CAP) v1.2 Australia (AU) Profile Version 1.0.,» 2013.

[9] J. Waters., «Emergency Data Exchange Language (EDXL) Distribution Element Version 2.0.,» 2013.

[10] OASIS, «Emergency Data Exchange Language Resource Messaging (EDXL-RM) v1.0».

[11] D. O'Donnell, B. Wilkins, R. Brooks e a. S. M. Robertson, «Emergency Data Exchange Language (EDXL) Hospital AVailability Exchange (HAVE) Version 2.0.,» 2019.

[12] A. W. Joerg e P. Iles, «Emergency Data Exchange Language (EDXL) Tracking of Emergency Patients (TEP),» 2018.

[13] E.-H. 2.0. [Online]. Available: https://docs.oasis-open.org/emergency/edxl-have/v2.0/cs02/edxl-have-v2.0-cs02.html.

[14] OASIS, «Emergency Data Exchange Language Hospital Availability Exchange (EDXL-HAVE) 1.0,» 2009.