Decision Support Systems 55 (2013) 336–347

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Knowledge-centered design of decision support systems for emergency management

Daniela Fogli *, Giovanni Guida

Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Brescia Via Branze 38, 25123 Brescia, Italy

ARTICLE INFO

Article history:
Received 1 June 2012
Received in revised form 4 December 2012
Accepted 28 January 2013
Available online 8 February 2013

Keywords:
Decision-support systems
Knowledge-centered design
Knowledge engineering
Emergency management

ABSTRACT

This paper focuses on the design of decision support systems for emergency managers in charge of planning, coordinating and controlling the actions carried out to respond to a critical situation. A novel knowledge-centered design methodology is proposed and demonstrated through the application in a concrete case study in the field of pandemic flu emergency management. Knowledge-centered design is based on a rational and structured approach to the elicitation and modeling of the knowledge concerning the target environment, the application domain, the intended users, their tasks, and the specific activities that the decision support system is expected to provide. Our proposal aims at overcoming some of the limitations of user-centered and activity-centered design in the specific context of decision support systems. Knowledge-centered design is based on an iterative process that goes through four main phases, namely: target environment identification, domain understanding, user characterization, and functional analysis. The paper illustrates each phase in detail and discusses the application in the proposed case study.

1. Introduction

Emergency management encompasses a variety of activities, such as training and preparation, early signal detection, planning, mitigation, response, and recovery, which are usually carried out to cope with potentially catastrophic events caused by natural hazards or human behavior [18]. Indeed, an emergency may have a huge impact on every business, service provider, community and family in a nation, not just because of illness and loss of life, but also for the negative effects it exerts on workforce, availability of goods and services, and economic and social conditions in general. A structured and coordinated management is fundamental to be prepared and to minimize the consequences that an emergency may originate. Different approaches to emergency management can be found in current literature. They can be roughly divided in three classes: i) solutions based on social media [8,21,39], which aim at improving information exchange and sharing, and to favor citizens' participation; ii) emergency management information systems [2,9,42], designed to address the problem of enabling communication and coordination among independent institutions that may have different and competitive goals, and where a central control is absent; and iii) decision support systems [6,15,45], proposed to provide emergency managers with indications for selecting among alternative courses of actions in complex and uncertain situations.

By examining closer the third class of approaches, we can observe that *Decision Support Systems* (*DSS*) are usually applied in specific emergency situations – for example nuclear and radiological emergencies [34], earthquakes [12], health emergencies [20] – or to address specific aspects of emergency management, such as situation awareness [35], improvisation [24], and context awareness [25]. The principles on which the design of these systems is based are mostly domain-dependent; only rarely general and systematic approaches to the design and development of DSSs are proposed in literature. Among the few examples, we mention the knowledge-based framework proposed in [10], which exploits semantic web technologies to model the application domain and fuzzy cognitive maps to represent emergency plans; Ahmed et al. [3] present a scenario-driven, process-oriented DSS generator; finally, the work described in [25], even though focused on context-awareness, proposes a high-level design theory for real-time accident handling.

All such systems and approaches, even though representing important success cases, offer only a limited contribution to the general topic of DSS design for emergency management. Most of the works mentioned above focus on the conceptual and technical aspects of the DSS, but leave the design methodology and the relevant design processes in the background. Moreover, each system seems to adhere to a specific design practice, which can hardly be replicated in different contexts. The first goal of our work is, therefore, to face the issue of generality and to propose a methodology to DSS design that can fit the needs of a variety of application contexts and can be effectively applied in practical cases.

Going deeper into the design issue, the paradigm of *user-centered design* is advocated in recent literature about emergency management as one of the most sound and viable approaches to DSS design (see, for instance, [5,7,29]). User-centered design requires to involve users throughout the design process, from requirement and task analysis to usability testing; in this way, users can directly influence how the system takes shape [1,28]. Recently, however, Donald Norman has

* Corresponding author. Tel.: +39 030 3715666; fax: +39 030 380014.
E-mail addresses: fogli@ing.unibs.it (D. Fogli), guida@ing.unibs.it (G. Guida).

0167-9236/$ – see front matter
http://dx.doi.org/10.1016/j.dss.2013.01.022

warned that satisfying user requests should not be overestimated: “listening to customers is always wise, but acceding to their requests can lead to overly complex designs” [27]. Users do not necessarily know what is good for them. Moreover, different users may express different requirements, and thus satisfying the needs of a user may imply to neglect those of another. User-centered design, although important, might not be enough and in some cases might even turn out to be harmful. Therefore, Norman suggests that *activity-centered design* is superior to the mere user-centered approach and should be preferred in most cases [27]. Indeed, while user-centered design focuses on what a user considers good for him/her, activity-centered design advocates the need to take into account what is actually good for a user performing a given activity. The difference between these two standpoints is evident: while the former is more subjective and puts the design responsibility onto the user's shoulders, the latter is more objective and restores the designer in his/her original role.

A closer analysis of these approaches reveals, however, that users – with their needs and activities – are generally not the ultimate targets of a DSS. Both the user and the system are part of a larger environment – a process, an organization, a social context – that is expected to benefit from the availability of the DSS. Considering the case of emergency management, while it is important to consider the needs and preferences of emergency managers and the requirements emerging from the analysis of their activity, it is equally – or even more – important to take into account the actual needs of the population affected by the emergency and the goals of the institutions in charge of facing the emergency. Both user-centered and activity-centered design paradigms hardly take into account these aspects and might reveal weak in complex decision making situations.

The second goal of our work is, therefore, to propose a novel approach to DSS design that focuses not only on the users and their activity, but also on the knowledge about the environment where the users operate and the DSS will be employed. Our standpoint is that if the users can express their needs and preferences, and the analysis of their activity can reveal much of the actual tasks they are in charge of, only the consideration of a larger environment can allow identifying the real goals, constraints, processes, and rules of the decision making activity. We call this novel approach *knowledge-centered design*, since it puts knowledge in the center: knowledge about the users of the DSS, the activities they are supposed to carry out with the support of the DSS, and the environment where the DSS will be applied. Knowledge-centered design is based on an evolutionary concept, where a design can grow step by step towards a better design, through an incremental process of mutual adaptation where each involved party – the user, the system, and the environment – can impose something to the others and accept or reject the impositions received. Our knowledge-centered design methodology does not aim at substituting user- or activity-centered design, but incorporates these concepts in a wider framework, thus making the whole design process more sound and robust. Moreover, it strongly promotes a participatory approach to design [36], where different subjects – managers, process and organization specialists, domain experts, and operative personnel – are involved in the design process and bring their specific knowledge and experience.

The paper is organized as follows: Section 2 presents an overview of the knowledge-centered design methodology; Section 3 introduces the HEALTHREATS case study; Sections 4 to 7 focus on the individual phases of the methodology and present them in detail with reference to the case study considered; Section 8 illustrates the main features of the implemented DSS prototype, and Section 9 reports the main outcomes of DSS evaluation; and finally, Section 10 discusses the results obtained and outlines some issues for future research.

2. Knowledge-centered design: the methodology at a glance

2.1. The phases

The core of knowledge-centered design is the consideration of the different kinds of knowledge that are involved in the design of a DSS, namely: (i) knowledge about the target environment where the DSS will be used, (ii) knowledge about the user tasks, and (iii) knowledge about user profiles and interaction patterns. In order to satisfy this need in a structured and effective way, a *knowledge-centered design methodology* grounded on four main phases has been defined, as illustrated below (see Fig. 1):

1. *Target environment identification*. This phase has the goal of developing a deep understanding of the piece of real world where the DSS will be applied, including all the stakeholders that will be influenced, directly or indirectly, by the system, their work practices, the tools they are used to interact with, as well as the social and physical contexts in which they operate.
2. *Domain understanding*. Domain understanding aims at identifying, collecting and representing all the knowledge relevant to the specific domain of the DSS, including the basic entities and processes

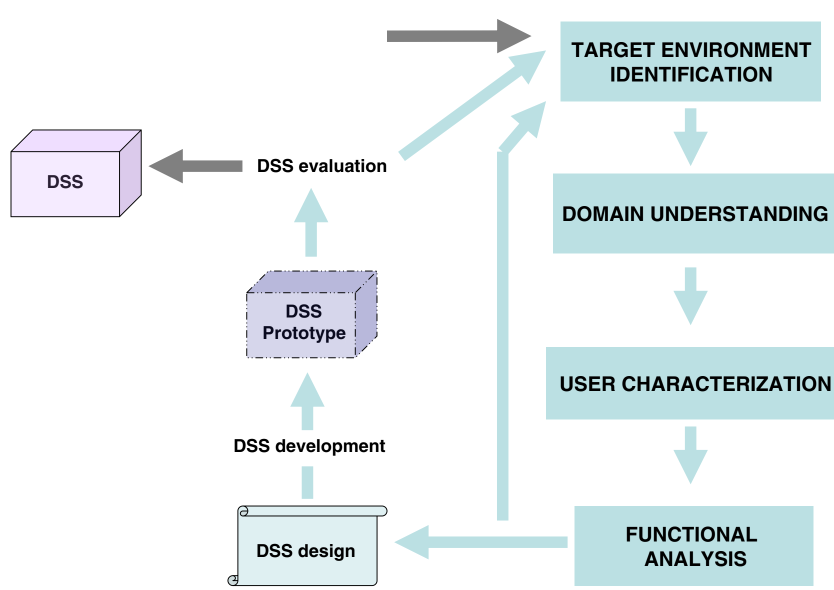

Fig. 1. The knowledge-centered design methodology.

that characterize the domain, the tasks carried out by emergency managers, and the knowledge necessary to operate correctly and effectively.
3. *User characterization*. This phase concerns all aspects of design that have an impact on the quality of interaction and, first of all, on usability. Therefore, the various classes of users that are expected to work with the DSS are identified, and their needs, goals, preferences, background, and social traits are brought to light. As a result, interaction requirements are then defined.
4. *Functional analysis*. Eventually, the specific functions that the DSS is expected to provide are analyzed and a detailed specification of system behavior is provided.

As it can be noted, phases (1) to (4) have a progressively narrower scope and focus on different kinds of knowledge. Starting from the target environment where the DSS is supposed to be applied, they consider in turn the application domain, the intended users, and eventually the functions that the system is expected to provide. DSS design is then followed by the *development phase*, and, once a prototype of the DSS has been developed, an *evaluation phase* is necessary to assess its performance both in an experimental setting and in the target environment.

This methodology is intrinsically iterative and includes two cycles: the inner cycle involves all the four design phases and aims at assuring through a step-wise refinement process that the best possible design is obtained, both in terms of correctness and completeness; the outer cycle involves also the development and evaluation phases and has the goal of favoring the incremental improvement of the DSS by repeatedly iterating through design, system development and evaluation.

2.2. The techniques

Applying the methodology outlined in the previous section requires the availability of sound techniques. The four design phases are all grounded on a common, fundamental activity, namely the acquisition and modeling of knowledge [17]. To this purpose, a specific *knowledge acquisition methodology* has been adopted that can be applied, with the relevant customizations, for each of the above four phases. This methodology is organized in five steps, as illustrated below (see Fig. 2):

(i) *Elicitation* collects and analyzes the original mind and point of view of managers, domain experts and users, with the aim of making their knowledge, needs, and expectations explicit. This step focuses both on surface knowledge that can be obtained with simple interviews, and on deep knowledge [31], which requires more introspection and analysis to be brought to light and expressed in verbal terms. All state-of-the-art techniques for knowledge acquisition, including questionnaires, individual and group interviews, and naturalistic observation may be employed. Important information may be obtained also from available documents on the specific topics at hand and from more informal sources of information like personal notes, technical drafts, and especially logbooks of past emergencies.

(ii) *Modeling* synthesizes the knowledge obtained through elicitation and produces a draft design in a form that can be easily and unambiguously understood by all persons involved in the design process. It focuses both on factual knowledge, including entities, properties and relations of the domain, and on procedural knowledge concerning actions, activities and processes.

(iii) *Integration* merges knowledge collected through different acquisition sessions and aims at producing a coherent and as far as possible complete representation of the relevant design elements. One of the main problems that has to be faced in this step is the fusion of knowledge obtained from different sources, which might be scarcely consistent or even conflicting.

(iv) *Refinement* aims at resolving ambiguities, deepening concepts, looking for missing knowledge, dropping useless or misleading information, correcting errors, and improving the representation. Refinement stimulates the feedback of managers, domain experts and users, collects their critiques, and goes through a rehearsal of the design produced so far until a clear and shared version is achieved.

(v) *Validation* critically examines and scrutinizes the design developed to obtain a final approval from all involved parties. Validation is the last but perhaps the most crucial step of knowledge

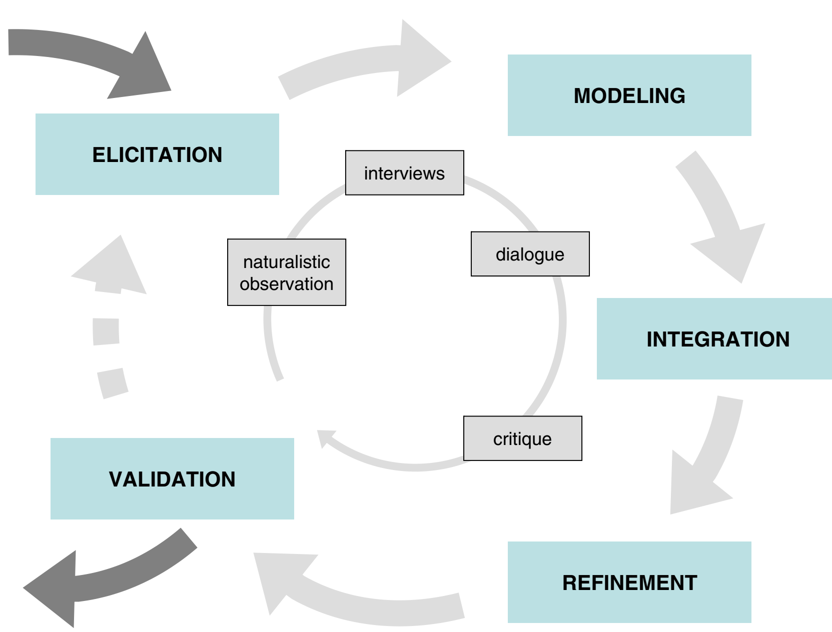

Fig. 2. The methodology for knowledge acquisition.

acquisition, since lacking a final assessment and a formal approval of the design may give rise to serious problems about the quality of the DSS to be developed.

Knowledge acquisition is a typical iterative process and must go through several cycles before a sound result is eventually obtained.

3. Experimenting knowledge-centered design

The knowledge-centered design methodology introduced in Section 2 was first proposed, in a preliminary version, in the frame of a one-year project [32] on decision support in emergency management funded by Regione Lombardia, Italy, in the years 2004–2005. In this context the fundamental traits of the methodology were outlined and practically experimented in the design of a DSS for assisting emergency managers in an earthquake case [16]. Two years later, the HEALTHREATS project [33], co-funded by the European Union in the years 2007–2010, offered the concrete occasion to develop the knowledge-centered design methodology to its full potential and to test it in a wider case concerning the issue of pandemic flu emergency. The HEALTHREATS project included eleven partners from five European countries, namely Italy, Romania, Slovenia, Portugal and Spain. The main goal was to develop an innovative DSS to assist the institutions in charge of public health in making strategic decisions underpinning the launch and management of operational interventions in response to pandemic flu. As a starting point of our project, the relevant prescriptions of the World Health Organization (WHO) have been assumed, as specified in the "WHO global influenza preparedness plan" [44], which provides national health authorities with priority goals and recommendations to develop preparedness plans. Of the six WHO pandemic phases, it was decided to focus on phases 4, 5 and 6, concerning the pandemic alert period and the pandemic period (see Table 1). The project activity was initially focused on the Italian case, namely on ASL Brescia (an important local health authority), which offered a suitable context for knowledge acquisition, DSS design, and evaluation. Afterwards, the DSS prototype was deployed to other partners of the HEALTHREATS project and further experimented in different practical cases. In Sections 4 to 7 we illustrate in detail the phases of the knowledge-centered design methodology applied to the design of the DSS prototype developed in the frame of the HEALTHREATS project.

Table 1
WHO pandemic periods and phases [44].

Period	Phase	Description
Interpandemic period	1	No new influenza virus subtypes have been identified in humans. An influenza virus subtype that has caused human infection may be present in animals. If present in animals, the risk of human infection is considered to be low
	2	No new influenza virus subtypes have been identified in humans, although a circulating animal influenza virus subtype poses a substantial risk of human disease
Pandemic alert period	3	Human infections with a new subtype, but no human-to-human spread or at most rare incidence of spread to a close human contact
	4	Small clusters with limited human-to-human transmission, but spread is highly localized, suggesting that the virus is not well-adapted to humans
	5	Larger cluster(s), but human-to-human spread still localized, suggesting that the virus is becoming increasingly better adapted to humans, but may not yet be fully transmissible
Pandemic period	6	Increased and sustained transmission in general human population
Postpandemic period		Return to interpandemic period

4. Identifying the target environment

Target environment identification includes several issues, largely depending on the specific application context considered. In our case we focused on three main points:

- modeling the emergency scenario and identifying the main stakeholders involved and their interactions;
- analyzing the emergency management task;
- identifying the decision support paradigm appropriate for emergency management.

To face these issues six managers and three domain experts were involved, while three knowledge engineers were in charge of the analysis. Initially, according to the methodology illustrated in Section 2.2, a wide-scope knowledge acquisition activity has been carried out. Several naturalistic observation sessions have been performed to analyze experts' behavior in concrete emergency management tasks. Moreover, log books compiled in past health emergencies have been analyzed, which offered a deeper and detailed insight into the matter and provided new hints for further investigation. Traditional individual and group interviews have then been exploited to elicit knowledge about the typical problems faced by emergency managers and the relevant strategies adopted. Specific attention has been devoted to the emergency management unit as a whole and to the need for cooperation among its members; here, several collaboration issues were brought to light and their impact on DSS design was clarified. After each significant step of the analysis, managers were requested to check and possibly revise the model of the target environment under construction and eventually express their validation.

The main results obtained in this phase are illustrated in the following sections.

4.1. The emergency scenario

An *emergency* is a situation that, independently of being or not being foreseeable and independently of the level of preparedness, causes a sudden and strong modification in a specific context with a widespread impact on the environment and on the society, requiring a prompt and massive intervention by all parties involved, from public institutions to private citizens. In general terms, in any emergency five main classes of stakeholders are involved: i) the *impact population* including the persons directly affected by the events that determine the state of emergency; ii) the *context population* concerning the people related in any way to the impact population, but not directly affected by the occurred events, ranging from the relatives and friends of the persons belonging to the impact population to the people simply informed of the events occurred through newspapers, radio, TV or the web; iii) the *intervention staff*, including all professionals and volunteers engaged in the practical management of the emergency; iv) the *emergency management unit*, in charge of planning, coordination and control of the actions carried out by the intervention staff; and v) the *institutions* involved in the management of the emergency at local, governmental or world-wide levels.

In this context, several interactions occur, both internally to each class of stakeholders and among them. On the basis of these interactions, the emergency environment can be structured into four levels:

- the *population level* includes the subjects belonging to the impact and context populations and any interaction occurring between them;
- the *operation level* concerns the intervention staff and includes both the interactions occurring inside it to organize and manage the operations, and the interactions between it and the impact and context populations aimed at actuating the concrete measures necessary to face the emergency;
- the *management level* is about the emergency management unit and involves the various interactions internal to the emergency

management unit and between the emergency management unit and the intervention staff;
- the *institutional level*, finally, concerns the interactions between the institutions and the emergency management unit.

This scenario is illustrated in Fig. 3.

The need for a DSS specifically devoted to guide and support the activity of the emergency management unit is largely recognized [6,15,42]. Therefore, in the following, we will focus on the management level only.

4.2. The emergency management task

Emergency management is a complex and multifaceted task that involves a variety of problems and requires that the persons in charge, namely the *emergency managers*, carry out multiple reasoning paths in parallel. Emergency managers are not involved in operative activities, but – more importantly – have responsibility about situation assessment, action planning, and allocation of resources. They are in charge of a wide set of duties. First of all, they must acquire and validate information from the field about the events occurred, their consequences, the actions undertaken to face the emergency situation and their effects. On the basis of this information – generally chaotic, redundant, and partially inconsistent – emergency managers assess the situation at hand, analyze primary needs, identify the most appropriate intervention plans, and allocate the necessary resources to the activities to be carried out. They also have to interact with the intervention staff in charge of executing the intervention plans and manage effective communication with them. Finally, they must assure correct and timely communication with all the institutions involved in emergency management.

Emergency managers carry out all these duties in a context characterized by fatigue, pressure, stress, and responsibility overload. In their job, they have to comply with formal regulations, apply stated emergency plans, find solutions to new emerging problems, dynamically adapt to a rapidly changing situation, and promptly respond both to events occurring in the emergency field and to the requests arriving from the institutions. These tasks are typically knowledge intensive and require emergency managers to exploit all their skills and experience and – most importantly – to collaborate. In fact, emergency management is a cooperative activity, where several specialists must share information and participate in a highly interactive decision making process [5].

The kind of collaboration necessary among emergency managers, however, has not to be conceived as a free and unstructured process, whose dynamics is solely governed by the goals, skills, and initiative of the participants. Collaboration is strictly ruled by domain knowledge and is aimed at concretely supporting emergency managers in the execution of specific tasks. Therefore, it must be highly structured and sacrifice part of freedom and spontaneity of a more flexible concept of collaboration in favor of effectiveness and efficiency.

4.3. The decision support paradigm

Several approaches to decision support are possible [15], ranging from merely informative to strong normative paradigms. *Informative support* aims at improving the quality of human decisions by providing decision makers with more information that can help them analyze the current situation and assess alternatives. However, in the case of a large-scale emergency a merely informative approach would not be effective: decision support should not provide emergency managers with more data, but possibly with less, more focused information. *Normative support*, on the other hand, aims at directly helping decision makers through the suggestion of a final decision, possibly the best available one, ready for use. In a sense, it aims at substituting decision makers in all possible cases, solving the decision problem at the root. A strongly normative approach would hardly be applicable in emergency management since it is practically impossible to foresee all possible cases that might occur and to identify, for each of them, an intervention strategy suitable in any context. Moreover, a too strong normative support might raise acceptance problems, since, without a justification, emergency managers might be reluctant to adhere to the suggestions produced by a DSS.

A closer analysis of the behavior of emergency managers reveals that the majority of decisions they make are simple: in front of a problem, there is only one possible solution, well known to emergency management experts. All these decisions can be made on the basis of available domain knowledge, and mostly rely on official regulations and proven response protocols. On the other hand, only a small number of decisions are really success-critical and inherently complex, requiring a high level of attention by emergency managers. Most of them concern however only three aspects: (i) the selection among

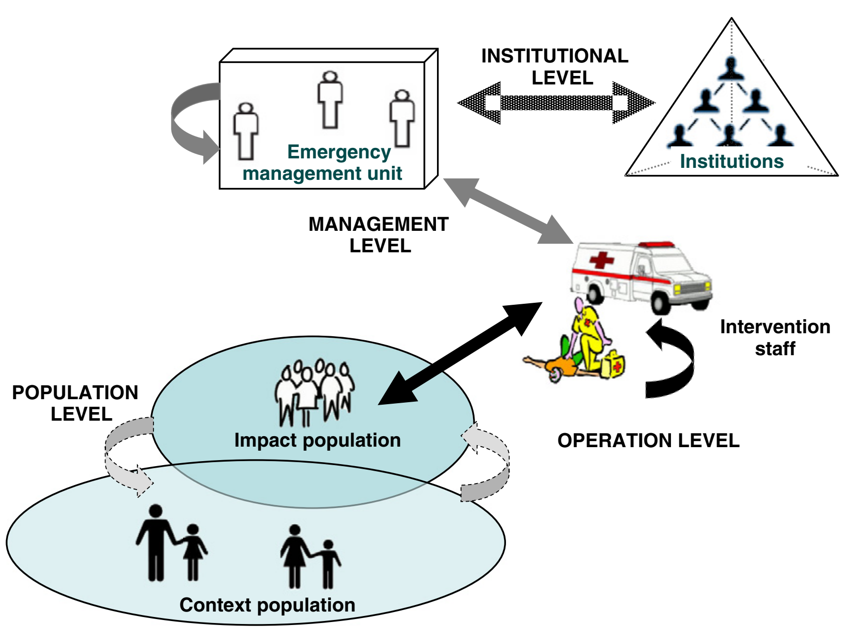

Fig. 3. The emergency scenario.

alternative intervention plans suitable to face a specific situation, (ii) the definition of a new plan in front of rare, atypical events for which a standard operative practice does not exist, and (iii) the allocation of resources, generally largely insufficient, to the activities to be carried out. According to these observations, a novel decision support paradigm is advocated, based on the following two main assumptions:

- for all non-critical decisions a *normative support* paradigm is appropriate; existing regulations and practical experience provide a rich and reliable knowledge base for decision making in all simple cases; such knowledge can be effectively coded into intervention plans that specify the best course of actions in front of an event; decision makers only need to approve – or reject – the intervention plans proposed by the DSS and their activity becomes simpler and faster;
- for critical decisions concerning the selection between alternative intervention plans, the definition of new plans, and the allocation of resources to the activities to be carried out, decision makers are left free to find the best solution resorting to their own knowledge and experience; in this case a *focused informative support* paradigm is appropriate: context-dependent and timely information can help decision makers improve the quality of their decisions and, at the same time, cope with improvisation [24].

The resulting approach to decision support is therefore a composite one, based on two different paradigms that apply to different decision making contexts according to their complexity, as suggested in [37].

5. Understanding the domain

Domain understanding aims at acquiring all the knowledge about the specific competence domain of the DSS necessary to:

- define the organizational structure of the collaboration context where the DSS will be used;
- identify the decision support objects, that is, the entities on which emergency managers work in their activity and that will constitute the data on which the DSS will operate.

Again the knowledge acquisition methodology illustrated in Section 2.2 has been applied with the support of two knowledge engineers. In the elicitation step, seven domain experts have been involved, representing the heterogeneous community of professionals in charge of managing a pandemic flu emergency, from veterinary and medical prevention, to primary health care and emergency assistance, home care and nursing service, socio-sanitary activities, and public safety. Two questionnaires have been first submitted to domain experts: the former focused on the needs and goals of domain experts and on the organization of the emergency management unit, the latter collected information on databases and documents that emergency managers usually exploit in their job. Afterwards, knowledge engineers focused on deep knowledge [31]; naturalistic observation and focus groups were of great help in this activity.

The modeling step included a detailed task analysis, to identify established procedures and best practices relevant to emergency management. Domain experts have been involved, through individual and group interviews, in identifying the significant events that might occur in each pandemic phase, the actors in charge of managing the emergency, their roles and responsibilities, as well as the plans to coordinate and manage the response to the emergency.

The integration step was carried out through focus groups, aimed at identifying and solving substantial or apparent divergences among domain experts, and to merge different perspectives into a coherent frame. Refinement included several rehearsal activities and, often, a return to elicitation, modeling or integration steps. Eventually, validation was carried out at the end of the knowledge acquisition process, when the domain model was complete, stable, and verified through an extended try-out activity.

5.1. Organizational structure

From the domain analysis carried out it emerged that a *hierarchical organization* is necessary for successful emergency management. Four issues support the need for a hierarchy:

- a rational assignment of roles and tasks to the various components of the emergency management unit;
- a clear separation of responsibilities;
- a precise definition of the information flow;
- an effective command chain between managers with different roles and, especially, between the emergency management unit and the intervention staff.

The analysis of the domain has shown that the management of a health emergency involves the following main levels:

- the *central level*, constituted by a task force, where all the departments and the directions of the health authority involved in emergency management are represented; on the basis of the events occurred, the task force decides the intervention plans to be carried out, follows their execution, and monitors the evolution of the emergency situation over time;
- the *territorial level*, represented by the branches of the health authority active in the geographical area involved in the emergency; this level is in charge of implementing the intervention plans suggested by the central level, by identifying the operative units and the resources necessary to carry out the individual activities prescribed by the plan; the territorial branches also manage the information flow from the field to the task force;
- the *operative level*, including the units in charge of carrying out the specific actions defined by the running intervention plans and of providing the relevant feedback to the territorial branches.

5.2. Decision support objects

Decision support objects are the abstract entities on which emergency managers work in their activity and, at the same time, the raw material on which the DSS operates. In general terms, decision support for emergency management can be grounded on three fundamental entities, as described below.

5.2.1. Events

An *event* is any fact happening at a certain instant and in a certain place in the emergency field, which is significant to the emergency management process. This concept includes not only the specific events that originate the emergency (an earthquake shock, an epidemic viral infection, a terrorist attack, etc.), but also their consequences (damages to a high risk factory, pollution of water, unavailability of communication networks, etc.) and the effects of the actions undertaken by the intervention staff in order to face the emergency (restoring an electric power facility, activating medical assistance to injured persons, executing safety checks to damaged buildings, etc.). In our case study 14 types of primary events have been identified, which represent the original events that occur in a pandemic emergency and that need to be managed. Examples of primary events are: suspected outbreak of avian influenza in a poultry farm, human case exposed to risk of contagion, suspected human case, presence of a suspected human cluster, and need for a mass vaccination.

5.2.2. Plans

A *plan* is a formal description of an intervention procedure that should be carried out in order to face a specific event. A plan specifies the *actions* that must be executed and the relevant control flow; actions can be executed sequentially or in parallel, can be controlled by the occurrence of conditions, and can be repeated until a certain condition is verified. Plans may also include information about the resources required to execute them and the expected completion time. Plans are intended to code all available knowledge on emergency management, including regulations, best practices and practical experience. In general, the relationship existing between events and plans is a partial many-to-many relation: several events may call for the execution of the same plan, while some events might be associated with more than one plan potentially appropriate to manage them; finally, some events might have no associated plan. Therefore, plans implement a form of mostly normative support, based on the best available knowledge, which applies to all cases where a proven solution is known. However, they leave the emergency managers completely free to deal with complex cases, for which no intervention plans exist or the availability of more than one possible plan calls for a choice. As a whole, 20 intervention plans in response to the 14 types of primary events have been identified. Examples of plans are: emergency vaccination plan, human resource shortage plan, vaccine virus distribution plan, plan for management of a suspected or confirmed human case, and nursing home emergency plan.

5.2.3. Resources

Each action of a plan requires the availability of specific resources to be executed. In general terms, a resource may be characterized by a *type*, a *place* where it is available, and a *quantity*. Moreover resources can be *consumable* (a specified quantity of the resource is consumed when it is used) or *reusable* (the resource may be reused as soon as it becomes available after use). Finally, resources can be classified into *human resources*, *material resources*, and *information resources*. In the HEALTHREATS project, the following main human resources have been identified: health authority staff, physicians, practitioners, nurses, and social workers. Material resources have been classified into several classes, including beds in hospitals and other healthcare structures, anti-viral drugs, vaccines, individual protection equipment, diagnostic kits, material for disinfection, and other drugs. Finally, information resources include two classes of objects: *reference documents* and *fill-in forms*. Reference documents are intended to provide general informative support to the users and include, for example, norms, official recommendations, and operative procedures. Fill-in forms are aimed at generating the technical reports necessary to assure a correct and timely communication with the involved institutions. Fill-in forms must reflect local reporting practices of the specific domain at hand and can be downloaded and completed by emergency managers whenever necessary. Both reference documents and fill-in forms are associated with the specific situations where they should be used, in such a way as to be presented to the users in the right moment and in the right context.

6. Characterizing the users

The third phase of the knowledge-centered methodology aims at identifying and characterizing the user classes that will utilize the DSS and at eliciting their interaction requirements. In the HEALTHREATS project, a team, including two knowledge engineers, two human–computer interaction specialists, eight representative users and seven domain experts, has carried out this phase.

First of all, seven classes of users were identified, as shown in Table 2. Then, the interaction scenarios between the users belonging to the identified classes and the DSS have been specified. Eight use cases, divided into five *normative use cases* and three *informative use cases*, have been designed; initially they have been documented in natural language according to a pre-defined template and later, after detailed rehearsal and refinement with users and domain experts, through formal UML diagrams [14]. Finally, attention has been focused on the interaction patterns between the users and the DSS; these aspects have been represented through five UML activity diagrams.

7. Analyzing the functions

Finally, the fourth phase of the knowledge-centered methodology focuses on the detailed specification of the functional requirements of the DSS, which constitute the fundamental reference for system development. This phase has been organized in two steps: the former devoted to the analysis of general requirements and the latter to the elicitation and modeling of knowledge about events and intervention plans.

7.1. General requirements

Two software designers with the collaboration of managers, domain experts and users have identified 77 functional requirements, divided into 8 areas: (i) management of events, plans, and actions; (ii) information about the emergency situation; (iii) management of material resources; (iv) management of human resources; (v) management of information resources; (vi) geographical mapping; (vii) management of knowledge and data bases; and (viii) system and user administration. Due to the high number of requirements identified and to the limited resources of the project, three classes of priorities have been defined: 'mandatory', 'advanced', and 'nice to have'.

7.2. Modeling of events and plans

Three knowledge engineers, seven domain experts, two managers and four users have collaborated to the elicitation and modeling of knowledge about events and plans already identified in the phase of domain understanding (see Sections 5.2.1 and 5.2.2). Initially, seven individual interviews with domain experts allowed knowledge engineers to analyze all possible events and to model the relevant intervention plans. Since intervention plans may refer to various types of documents (for example, operative procedures to be followed or report forms to be filled in) or to external data sources (typically, databases containing information about the emergency scenario), knowledge

Table 2
User classes and their main tasks.

User role	Organization level	Tasks
Event manager	Operative level	- Collecting input information for the DSS from the emergency field or from external sources
Central emergency manager	Central level	- Identifying and instantiating intervention plans - Requiring the execution of a new plan by a local emergency manager - Monitoring plan execution - Managing requests for the execution of a new plan submitted by local emergency managers
Local emergency manager	Territorial level	- Executing intervention plans - Monitoring plan execution - Requiring the execution of a new plan or stopping a running plan - Managing resources available in the relevant territorial branch
Field operator	Operative level	- Executing the actions prescribed in the running intervention plan - Providing the local emergency manager with the results of the actions undertaken
Resource allocation manager	Central level Territorial level	- Updating and monitoring allocation of resources in the various territorial branches
Knowledge administrator	Central level	- Managing DSS knowledge bases
System administrator	Central level	- Managing user accounts and user groups - Providing operative support during DSS operation

about all these kinds of external references has been elicited as well, and properly associated to the relevant plans. In a second iteration of the knowledge acquisition cycle, four individual interviews have been carried out with the aim of deepening knowledge about plans and completing their representation. Finally, two group interviews with domain experts, managers and users have been devoted to refine and validate the whole set of events and plans collected.

Plans have been represented in BPMN (Business Process Modeling Notation) [43], which proved to be suitable both for domain experts and knowledge engineers, thus constituting an effective communication and knowledge sharing tool. BPMN includes all the control primitives necessary to model complex processes, such as sequence, selection, iteration, and parallel execution; moreover, it can be compiled into executable computer code, thus making the implementation of the DSS easier. For example, Fig. 4 shows the "Plan for the surveillance of human subject exposed to risk of pandemic virus" associated with the primary event "Human case exposed to risk of contagion". The start event of the plan is represented through a small circle (at the left in the example), while each rectangle denotes either an action to be carried out or a new plan to be activated (in the example, the "Plan for the management of a suspected or confirmed human case"). A diamond with a cross represents an exclusive gateway, that is a choice among a set of possible alternatives (in the example, the left-most exclusive gateway asks to verify the presence in the emergency environment of children attending public schools: if this turns out to be true, the action "Apply measures to restrict presence of children in the emergency environment" has to be carried out, otherwise the plan continues with the next action). Other primitives (not included in the example) allow representing inclusive decisions and parallel executions (with fork and join symbols). Actions may refer to documents or to external data sources; to denote such references, colored data objects are used; in particular, red data objects (as shown in the example) refer to procedures, standards, or official regulations, to be applied or taken into account during the execution of an action.

8. The DSS prototype

On the basis of the design developed according to the knowledge-centered process described in the previous sections, a web-based DSS prototype has eventually been developed. Its three-layer architecture is shown in Fig. 5.

The *user interface layer* is in charge of generating the web pages that allow users to access the functionalities of the DSS made available by the application layer.

The *application layer* is composed of four modules. The *data access module* and the *document access module* include the logical components in charge of implementing informative support. The *administration module* provides the functions necessary to manage all data, document, and knowledge bases of the DSS. The *DSS core module* is the heart of the DSS and is grounded on a knowledge-based reasoning engine. It is in charge of providing the specific normative support necessary to aid emergency managers in their job; more precisely, in front of events occurring in the emergency field, it suggests the most suitable intervention plans and offers support to their correct and effective execution (see [13] for more details).

The *knowledge and data layer* includes three components: the *data base* comprising the available resources and their allocation, the statistical data about the emergency, and the geographical data concerning the emergency environment; the *document base* containing all the documents relevant to emergency management; and the *knowledge base* that stores the domain knowledge on which the operation of the DSS core module is based, namely: plans, actions, event–plan associations, and resource–action relations.

From a technical point of view, the DSS has been implemented on the Java Enterprise Edition (Java EE) platform. The user interface layer is build upon the Apache web server that manages the request and generation of web pages. The application layer includes the application logic running in the JBoss container. The knowledge and data layer consists of a set of MySQL relational databases.

To develop DSS user pages, widely-accepted human–computer interaction design patterns for web applications have been adopted [41], with the aim of presenting support information – both informative and normative – in a clear and as far as possible efficient way, thus facilitating interaction and enhancing usability. While different pages are necessary for the various user roles and tasks, a common basic structure has been replicated in all user pages to ensure consistency and support ease of use. A generic page includes the following fundamental elements, highlighted in Fig. 6 that shows the home page of the DSS: (1) the page header with the logo of the DSS, which also constitutes a link to the home page; (2) the main menu bar, which is different for each user role; (3) the "signed in as" area, with the username of the user logged in; (4) the page title; (5) a brief subtitle describing the page content; (6) a welcome message (just for the home page);

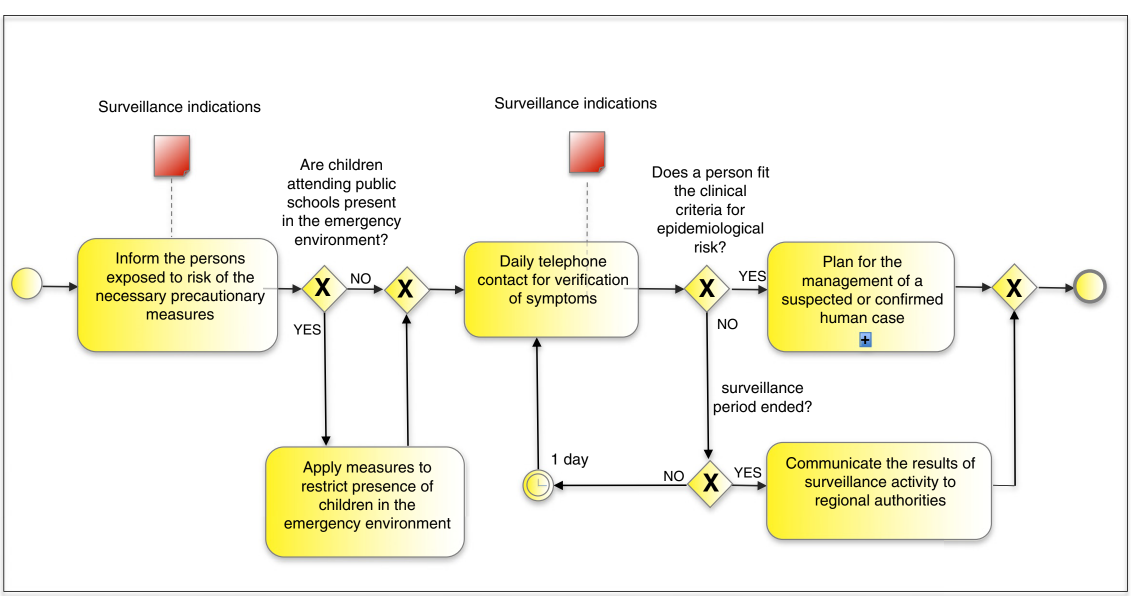

Fig. 4. Plan for the surveillance of human subject exposed to risk of pandemic virus.

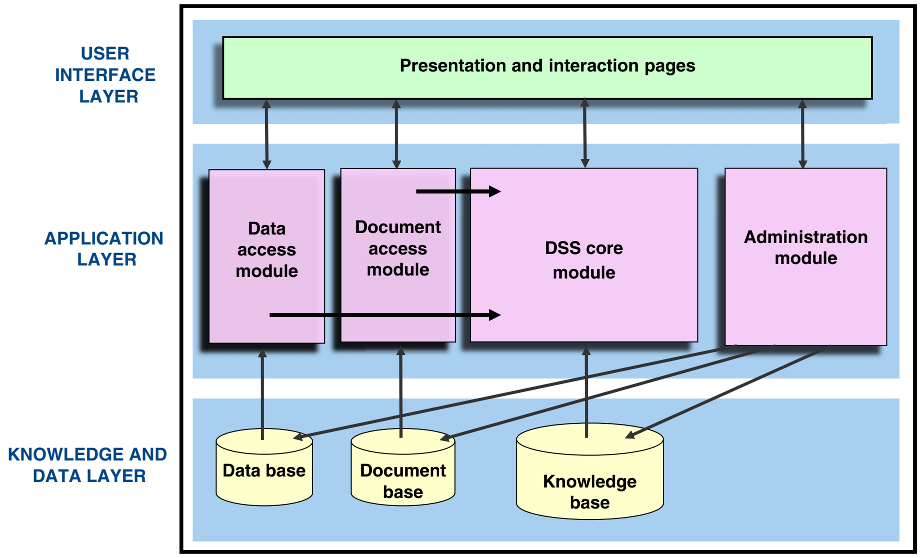

Fig. 5. The DSS architecture.

(7) the main area, which depends on user role and system state; and (8) the footer, with the system version and the link to the project web site.

9. DSS evaluation

9.1. The quality characteristics

After DSS development, evaluation with end users was carried out, focusing both on the quality of user interaction with the DSS and on the capability of the DSS of supporting collaboration among users.

As to the evaluation of user–DSS interaction, attention has been centered on *usefulness* and *usability*, generally considered the milestones for system acceptability [26] with a direct impact on the satisfaction of user needs. *Usefulness* is concerned with the question whether the system meets user expectations. Specifically, it assesses implemented DSS functions against the *intended* requirements of the users. *Usability* faces the question whether users can perform their tasks easily and effectively by exploiting the interaction tools offered by the DSS.

To evaluate user-to-user interaction through the DSS, only one global characteristic has been considered, namely *collaboration effectiveness*, which represents the degree to which the DSS can support effective and efficient collaboration among the persons involved in emergency management.

9.2. Evaluation methods

9.2.1. Assessing usefulness

The following methodology, an extension of questionnaire-based methods [23], has been adopted to carry out usefulness assessment:

1. An *evaluation team* was first defined, including a set of real users in charge of assessing the implemented DSS functions.
2. A set of *test cases* related to the implemented requirements was selected and assumed as a reference for the evaluation.

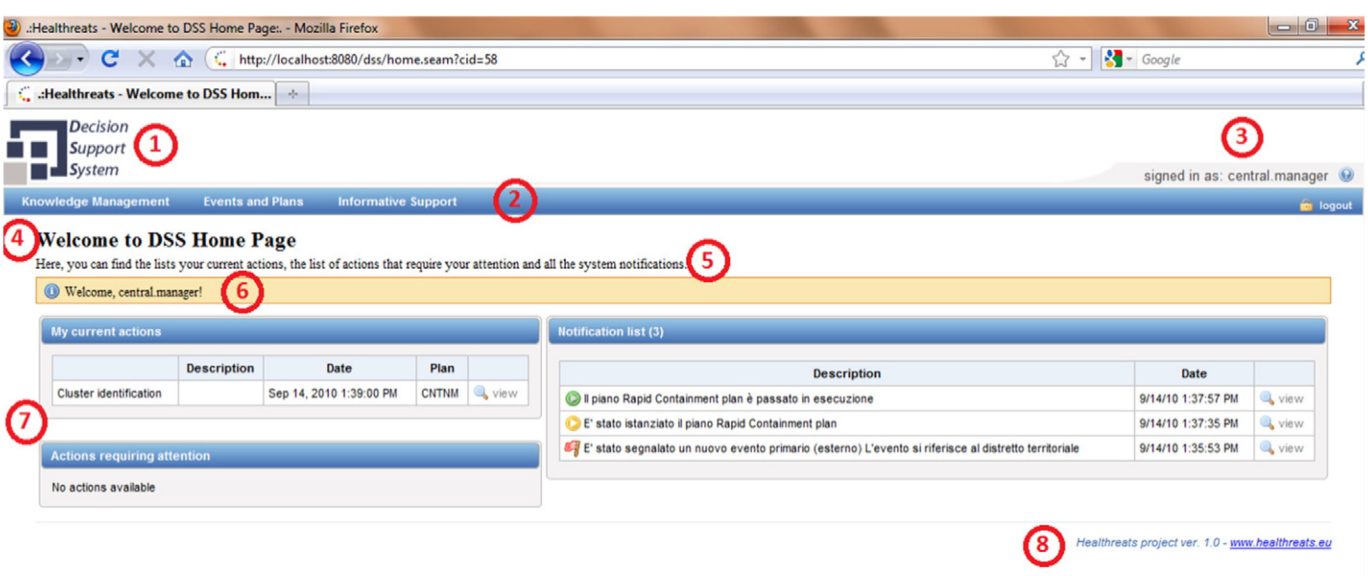

Fig. 6. The home page of the DSS.

3. An *indicator* for usefulness was then defined as a triple $\underline{\text{USF}} = \langle$USF/man, USF/adv, USF/nth$\rangle$, where USF/man, USF/adv, and USF/nth represent the average values of the assessment expressed by the evaluation team about 'mandatory', 'advanced', and 'nice to have' requirements respectively. Users were instructed to express their judgment, for each test case, through a qualitative value in the ordered set {*unacceptable*, *partially acceptable*, *adequate*, *satisfactory*, *completely satisfactory*} according to the degree to which the implemented DSS functions meet their intended requirements. These values were then mapped onto the integer scale {1, 2, 3, 4, 5}.
4. A *metric* for USF was finally defined to aggregate the three components of $\underline{\text{USF}}$, namely USF/man, USF/adv, and USF/nth, into a global qualitative evaluation:

   $\text{USF} = \text{EXCELLENT iff } (\text{USF/man} \geq 4 \ \underline{\text{and}}\ \text{USF/adv} \geq 3 \ \underline{\text{and}}\ \text{USF/nth} \geq 3)$
   $\text{USF} = \text{GOOD iff } (\text{USF/man} \geq 3 \ \underline{\text{and}}\ \text{USF/adv} \geq 2 \ \underline{\text{and}}\ \text{USF/nth} \geq 2 \ \underline{\text{and}}\ \underline{\text{not}}\ \text{EXCELLENT})$
   $\text{USF} = \text{ACCEPTABLE iff } (\text{USF/man} \geq 3 \ \underline{\text{and not}}\ (\text{GOOD}\ \underline{\text{or}}\ \text{EXCELLENT}))$
   $\text{USF} = \text{POOR otherwise.}$

9.2.2. Assessing usability

To assess usability we have considered *robustness*, *efficiency* and *memorability*, which constitute a subset of the usability dimensions proposed in [26], and that were considered meaningful for the case at hand. The methods for measuring such dimensions are quite standard in the specific usability literature and are based on measuring the time and the error rate for completing a selected set of tasks [26]. The following methodology has then been adopted:

1. An *evaluation team* was defined, including a set of real users in charge of assessing DSS usability and a set of reference users perfectly trained and fully familiar with the DSS.
2. A set of *test tasks*, where a task is intended as a structured set of activities required to achieve a given goal [11], was selected and assumed as a reference for the evaluation.
3. An *indicator* for usability was then defined as a triple $\underline{\text{USE}} = \langle$USE/eff, USE/rob, USE/mem$\rangle$, where USE/eff, USE/rob and USE/mem represent the ratio between the average values of the assessment of robustness, efficiency and memorability recorded with the real users of the evaluation team while executing the set of test tasks defined (*user values*) and the values obtained with the same set of test tasks by the reference users (*reference values*). In particular, during the execution of a test task, the time spent to carry out the task and the number of errors made were counted to assess efficiency and robustness respectively; memorability was then evaluated by repeating the execution of the same tasks after a period of inactivity with the system and by measuring the difference in efficiency observed with respect to the first experimental session.
   The values assigned to USE/eff, USE/rob, USE/mem were then mapped onto an integer scale from 1 to 5 with the following semantics:

   1: user value > 1.8 × reference value
   2: 1.4 × reference value < user value ≤ 1.8 × reference value
   3: 1.2 × reference value < user value ≤ 1.4 × reference value
   4: reference value < user value ≤ 1.2 × reference value
   5: user value ≤ reference value
4. A *metric* for USE was finally defined to aggregate the three components of $\underline{\text{USE}}$, namely USE/eff, USE/rob, and USE/mem, into a global qualitative evaluation:

   $\text{USE} = \text{EXCELLENT iff } (\text{USE/eff} \geq 4 \ \underline{\text{and}}\ \text{USE/rob} \geq 4 \ \underline{\text{and}}\ \text{USE/mem} \geq 3)$
   $\text{USE} = \text{GOOD iff } (\text{USE/eff} \geq 3 \ \underline{\text{and}}\ \text{USE/rob} \geq 3 \ \underline{\text{and}}\ \text{USE/mem} \geq 2 \ \underline{\text{and}}\ \underline{\text{not}}\ \text{EXCELLENT})$
   $\text{USE} = \text{ACCEPTABLE iff } (\text{USE/eff} \geq 2 \ \underline{\text{and}}\ \text{USE/rob} \geq 2 \ \underline{\text{and not}}\ (\text{GOOD}\ \underline{\text{or}}\ \text{EXCELLENT}))$
   $\text{USE} = \text{POOR otherwise}$

9.2.3. Assessing collaboration effectiveness

The issue of evaluating the collaboration effectiveness of a computer-based system aimed at supporting task-oriented interaction between persons in a specific working environment is only poorly considered in current literature. Some approaches simply focus on usability of collaboration systems (see, for example, [30,38]), while others, even if specifically considering effectiveness, do not provide objective measures, but only gather comments and suggestions from the users through workshops or focus groups (see, for instance, [4]). Therefore, a new heuristic approach has been proposed, based on three dimensions: *pertinence*, considered as the property of user-to-user interactions to be focused on the topic at hand, avoiding out-of-scope or immaterial dialogs; *correctness*, intended as the property of pertinent interactions to convey correct information, avoiding misleading or even wrong messages; and *utility*, which accounts for the property of correct interactions to concretely contribute to the effectiveness and efficiency of the collaborative task.

The following methodology, purposely designed for the case at hand, has been adopted to carry out the assessment of collaboration effectiveness:

1. An *evaluation team* was first defined, including a group of real users with the role of assessing DSS effectiveness.
2. A (simulated but realistic) *test scenario* was defined, where the evaluation team was requested to carry out a complete *emergency management job*.
3. An *indicator* for collaboration effectiveness was then defined as a triple $\underline{\text{CEF}} = \langle$CEF/per, CEF/cor, CEF/uti$\rangle$, where CEF/per, CEF/cor, and CEF/uti represent the average values of the assessment of pertinence, correctness and utility expressed by the users of the evaluation team while taking part in the emergency management job defined. In particular, during the execution of their job, users were requested to record the total number of user-to-user interactions occurred, by distinguishing between the interactions that are considered pertinent, correct or useful. CEF/per, CEF/cor, and CEF/uti were then mapped onto an integer scale from 1 to 5 with the following semantics:

   1: $\text{Ni} < 0.6 \times \text{Nref}$
   2: $0.6 \times \text{Nref} \leq \text{Ni} < 0.7 \times \text{Nref}$
   3: $0.7 \times \text{Nref} \leq \text{Ni} < 0.8 \times \text{Nref}$
   4: $0.8 \times \text{Nref} \leq \text{Ni} < 0.9 \times \text{Nref}$
   5: $\text{Ni} \geq 0.9 \times \text{Nref}$

   where:
   - for Ni = Nper, Nref = Ntot; for Ni = Ncor, Nref = Nper; and for Ni = Nuti, Nref = Ncor;
   - Ntot is the total number of interactions, and Nper, Ncor and Nuti are the numbers of pertinent, correct or useful interactions respectively.
4. A *metric* for CEF was finally defined to aggregate the three components of $\underline{\text{CEF}}$, namely CEF/per, CEF/cor, and CEF/uti, into a global qualitative evaluation:

   $\text{CEF} = \text{EXCELLENT iff } (\text{CEF/per} \geq 4 \ \underline{\text{and}}\ \text{CEF/cor} \geq 4 \ \underline{\text{and}}\ \text{CEF/uti} \geq 4)$
   $\text{CEF} = \text{GOOD iff } (\text{CEF/per} \geq 3 \ \underline{\text{and}}\ \text{CEF/cor} \geq 4 \ \underline{\text{and}}\ \text{CEF/uti} \geq 3 \ \underline{\text{and}}\ \underline{\text{not}}\ \text{EXCELLENT})$
   $\text{CEF} = \text{ACCEPTABLE iff } (\text{CEF/per} \geq 3 \ \underline{\text{and}}\ \text{CEF/cor} \geq 3 \ \underline{\text{and}}\ \text{CEF/uti} \geq 3 \ \underline{\text{and not}}\ (\text{GOOD}\ \underline{\text{or}}\ \text{EXCELLENT}))$
   $\text{CEF} = \text{POOR otherwise}$

9.3. Experimental setting and assessment results

For the assessment of usefulness and usability, two human–computer interaction specialists have organized and conducted an experiment with five users recruited from ASL Brescia. A set of 34 cases, covering most of the DSS requirements and comprising the main activities users have to carry out during an emergency situation, was selected and used both as test cases for the assessment of usefulness and as test tasks for the assessment of usability. Experimental sessions have then been carried out under laboratory conditions in which the users could operate in an interruption-free environment.

For the assessment of collaboration effectiveness, a test scenario offered by ASL Brescia (District No. 4 — Valle Trompia) was assumed. The emergency team created for DSS experimentation included one event manager, two central emergency managers, three local emergency managers, and four field operators; separate working rooms have been prepared to simulate a distributed setting. The simulation was planned to last for a whole working day. In this period three events occurred – namely “Need for mass vaccination”, “Suspected human case”, “Need for local emergency vaccination” – which triggered seven intervention plans. During the execution of the simulated emergency, a human–computer interaction specialist provided the necessary support for the correct execution of the experimentation, observed the activities of the team, and collected user judgments about collaboration effectiveness.

The final assessment of the DSS has provided the results presented in Table 3.

In addition to the evaluation reported above, informal opinions have been gathered from the users during the assessment activities. They indicate that:

- all users judged the DSS useful for their work, not only for the management of emergency situations but also as a personal assistant in their daily activity; the possibility to monitor the execution of plans in a simple and structured way was especially appreciated;
- usability was considered good, with the exception of some complaints for the long time required to carry out some specific operations;
- some users pointed out how the DSS can prevent that different people simultaneously make incoherent or even contradictory decisions;
- the majority of the members of the emergency team agreed that the DSS can simplify person-to-person interactions and make them more effective and focused on the goal; moreover, they observed that new management processes induced by the adoption of the DSS make operators feel more involved in a complex organization.

10. Discussion and conclusion

The research and experience reported in this paper provide a contribution to the state of the art in collaboration and emergency management from three perspectives. First of all, from the point of view of design methodologies, it proposes a novel knowledge-centered approach to DSS design which is a step further both with respect to user-centered design promoted in emergency management literature [7,29] and DSS research [22,40], and to activity-centered design proposed by Norman [27] for the development of complex interactive systems. In comparison with other approaches to DSS design recently proposed in literature, which are mainly concerned with the development of general frameworks (see for example, [3,10]) or high-level design theories (e.g., [25]), our methodology is based on a rational and structured approach, which exploits sound knowledge engineering methods to carry out all the phases and tasks of the design process. Besides emergency management, it is applicable in any decision-support context characterized by the availability of a variety of knowledge sources and by the need of balancing between informative and normative support. The practical application of the methodology is demonstrated with reference to HEALTHREATS project.

Second, the experience reported in the paper has shown the validity – at least in emergency management applications – of the new decision support paradigm adopted, which combines normative and focused informative support. The advantages of this approach are many. First of all, emergency managers can concentrate only on critical cases that definitely require their personal competence and experience, relying for all simple decisions on a normative support. This way, their cognitive resources are not dispersed among a number of trivial choices, but can be focused on a small set of crucial aspects. Moreover, for critical decisions, emergency managers do not receive a generic informative support, which might give rise to the undesired result of increasing their information overload, but receive a focused support that provides them with synthetic information specifically tailored to the decision problem at hand.

Third, a multifaceted approach to the evaluation of a DSS has been proposed, which considers not only the quality of user–DSS interaction, including usefulness and usability, but also the level of user-to-user interaction through the DSS, namely collaboration effectiveness. This represents a substantial advancement with respect to existing literature that usually focuses only on usability [19] and misses to consider the degree to which a system meets the concrete requirements of an application context. Moreover, the proposed approach has been developed in all technical details necessary for practical implementation and has been actually experimented in the case study considered.

Finally, the present work has disclosed several promising research directions. An important topic, mentioned by more than one user during the experimentation, concerns the capability of the DSS to provide explanations and justifications for the intervention plans proposed, showing the goals that have inspired the design of the plan, the reasons behind the actions suggested, the alternatives discarded, and the expected benefits. This point might greatly contribute both to increase the level of system acceptance and to support its use as a training tool for improving emergency preparedness.

Another issue concerns the possibility of providing the DSS with a new functional module that enables users, under specified and controlled conditions, to modify existing plans or to edit new plans at run-time in face of unforeseen situations. These plans might then be reconsidered after the emergency is concluded and, after critical revision, might become part of the permanent knowledge base of the system or discarded if they revealed to be inappropriate or ineffective. This might expand the flexibility of the DSS, stimulate a more active user participation, and set the background for continuous improvement; moreover, the DSS might support this way the incremental construction of a valuable repository of emergency management knowledge. A last, but not less important, direction for future activity is the extension and experimentation of the DSS to face other types of emergencies; a project in the area of safety in work environments will most probably be launched shortly.

Acknowledgments

The authors would like to thank all the colleagues who contributed to the definition of the knowledge-centered methodology, to the design of the DSS prototype, and to the experimental assessment, in particular Pietro Baroni, Massimiliano Giacomin, and Loredana Parasiliti

Table 3
Assessment results.

Usefulness			*Usability*			*Collaboration effectiveness*		
USF/man	USF/adv	USF/nth	USE/eff	USE/rob	USE/mem	CEF/per	CEF/cor	CEF/uti
4.05 USF = EXCELLENT	4.40	4.83	3.66 USE = GOOD	4.61	4.15	4.12 CEF = GOOD	4.89	3.45

Provenza. The contribution of all partners of the HEALTHREATS project is also recognized.

References

[1] C. Abras, D. Maloney-Krichmar, J. Preece, User-centered design, in: W. Bainbridge (Ed.), Encyclopedia of Human–Computer Interaction, Sage Publications, Thousand Oaks, CA, USA, 2004, pp. 763–767.
[2] I. Aedo, P. Díaz, J.M. Carroll, G. Convertino, M.B. Rosson, End-user oriented strategies to facilitate multi-organizational adoption of emergency management information systems, Information Processing and Management 46 (1) (2010) 11–21.
[3] D.M. Ahmed, D. Sundaram, S. Piramuthu, Knowledge-based scenario management — process and support, Decision Support Systems 49 (2010) 507–520.
[4] J.E. Bardram, Activity-based computing for medical work in hospitals, ACM Transactions on Computer–Human Interaction 16 (2) (2009) 10:1–10:36.
[5] P. Baroni, D. Fogli, M. Giacomin, G. Guida, L. Parasiliti Provenza, M. Rossi, M. Bohanec, M. Žnidaršič, Supporting DSS acceptability through a user-centered design methodology: experiences in emergency management, in: A. Respìcio, et al., (Eds.), Bridging the Socio-technical Gap in DSS, IOS Press, Amsterdam, Nederland, 2010, pp. 87–98.
[6] F. Burstein, G.R. Widmeyer, Decision support in an uncertain and complex world, Decision Support Systems 43 (2007) 1647–1649.
[7] L. Carver, M. Turoff, Human–computer interaction: the human and computer as a team in emergency management information systems, Communications of the ACM 50 (3) (2007) 33–38.
[8] L.M. Collins, J.E. Powell, C.E. Dunford, K.K. Mane, M.L.B. Martinez, Emergency information synthesis and awareness using E-SOS, Proceedings of the 5th ISCRAM Conference, Washington, DC, 2008, pp. 618–623.
[9] M. Couturier, E. Wilkinson, Open Advanced System for Improved crisiS management (OASIS), Proceedings of the 2nd ISCRAM Conference, Brussels, Belgium, 2005, pp. 283–286.
[10] C. De Maio, G. Fenza, M. Gaeta, V. Loia, F. Orciuoli, A knowledge-based framework for emergency DSS, Knowledge-Based Systems 24 (8) (2011) 1372–1379.
[11] D. Diaper, N. Stanton (Eds.), The Handbook of Task Analysis for Human–Computer Interaction, Lawrence Erlbaum Ass, Hillsdale, NJ, USA, 2003.
[12] H. Engelmann, F. Fiedrich, Decision support for the members of an emergency operation centre after an earthquake, Proceedings of the 4th ISCRAM Conference, Delft, NL, 2007, pp. 317–326.
[13] D. Fogli, G. Guida, Enabling collaboration in emergency management through a knowledge-based decision support system, in: A. Respìcio, F. Burstein (Eds.), Fusing Decision Support Systems into the Fabric of the Context, IOS Press, Amsterdam, Nederland, 2012, pp. 291–302.
[14] M. Fowler, UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3rd ed. Pearson Education Inc., Boston, MA, USA, 2004.
[15] S. French, M. Turoff, Decision support systems, Communications of the ACM 50 (3) (2007) 39–40.
[16] M. Giacomin, G. Guida, M. Rossi, A. Viola, A knowledge-based decision support system for crisis management in the field of public health, Proceedings Workshop on Intelligent Decision Support in Process Environment, Siena, Italy, 2005.
[17] M. Greenwell, Knowledge Engineering for Expert Systems, Halsted Press, New York, NY, USA, 1988.
[18] G.D. Haddow, J.A. Bullock, Introduction to Emergency Management, Butterworth-Heinemann, Amsterdam, Nederland, 2003.
[19] K. Hornbæk, Current practice in measuring usability: challenges to usability studies and research, International Journal of Human Computer Studies 64 (2006) 79–102.
[20] J. Jenvald, M. Morin, T. Timpka, H. Eriksson, Simulation as decision support in pandemic influenza preparedness and response, Proceedings of the 4th ISCRAM Conference, Delft, NL, 2007, pp. 295–304.
[21] A. Krakovsky, The role of social networks in crisis situations: public participation and information exchange, Proceedings of the 7th ISCRAM Conference, Seattle, WA, USA, 2010.
[22] S. Lepreux, M. Abed, C. Kolski, A human-centred methodology to decision support system design and evaluation in a railway network context, Cognitive Technology Work 5 (2003) 248–271.
[23] A.M. Lund, Measuring usability with the USE Questionnaire, STC Usability SIG Newsletter 8:2, available at http://www.stcsig.org/usability/newsletter/0110_measuring_with_use.html 2001, (last visited May 2012).
[24] D. Mendonça, G.E.G. Beroggi, W.A. Wallace, Decision support for improvisation during emergency response operations, International Journal of Emergency Management 1 (1) (2001) 30–38.
[25] E.W.T. Ngai, T.K.P. Leung, Y.H. Wong, M.C.M. Lee, P.Y.F. Chai, Y.S. Choi, Design and development of a context-aware decision support system for real-time accident handling in logistics, Decision Support Systems 52 (2012) 816–827.
[26] J. Nielsen, Usability Engineering, Academic Press, San Diego, CA, USA, 1993.
[27] D. Norman, Human-centered design considered harmful, Interactions 12 (4) (2005) 14–19.
[28] In: D.A. Norman, S.W. Draper (Eds.), User-centered System Design: New Perspectives on Human–Computer Interaction, Lawrence Erlbaum Ass, Hillsdale, NJ, USA, 1986.
[29] T. Onorati, A. Malizia, P. Diaz, I. Aedo, Interaction design for web emergency management information systems, Proceedings of the 7th ISCRAM Conference, Seattle, WA, USA, 2010.
[30] D. Pinelle, C. Gutwin, Task analysis for groupware usability evaluation: modeling shared-workspace tasks with the mechanics of collaboration, ACM Transactions on Computer–Human Interaction 10 (4) (2003) 281–311.
[31] M. Polanyi, The Tacit Dimension, Rouledge & Kegan Paul, London, UK, 1967.
[32] Project 196320, An integrated decision support system for crisis management, supported by Regione Lombardia, Italy, FSE, Misura D4, November 2004–December 2005.
[33] Project 2006203, HEALTHREATS — Integrated Decision Support System for Health Threats and Crises Management, Supported by the European Union, Executive Agency for Health and Consumers, May 2007–September 2010.
[34] W. Raskob, V. Bertsch, J. Geldermann, S. Baig, F. Gering, Demands to and experience with the decision support system RODOS for off-site emergency management in the decision making process in Germany, Proceedings of the 2nd ISCRAM Conference, Brussels, Belgium, 2005, pp. 269–278.
[35] C. Sapateiro, P. Antunes, An emergency response model toward situational awareness improvement, Proceedings of the 6th ISCRAM Conference, Goteborg, Sweden, 2009.
[36] D. Schuler, A. Namioka (Eds.), Participatory Design, Principles and Practice, Lawrence Erlbaum Ass, Hillsdale, NJ, USA, 1993.
[37] D.J. Snowden, M.E. Boone, A leader's framework for decision making, Harvard Business Review 85 (11) (November 2007) 68–76.
[38] M. Steves, E. Morse, C. Gutwin, S. Greenberg, A comparison of usage evaluation and inspection methods for assessing groupware usability, Proceedings Int. Conf. on Computer-Supported Cooperative Work (CSCW 2001), ACM Press, New York, 2001, pp. 96–105.
[39] J. Sutton, L. Palen, I. Shklodvski, Backchannels on the front lines: emergent uses of social media in the 2007 Southern California wildfires, Proceedings of the 5th ISCRAM Conference, Washington, DC, 2008, pp. 624–632.
[40] K.A. Thursky, M. Mahemoff, User-centered design techniques for a computerized antibiotic decision support system in an intensive care unit, International Journal of Medical Informatics 76 (2007) 760–768.
[41] J. Tidwell, Designing Interfaces. Patterns for Effective Interaction Design, O'Reilly, Sebastopol, CA, USA, 2005.
[42] M. Turoff, S.R. Hiltz, C. White, L. Plotnick, A. Hendela, X. Yao, The past as the future of emergency preparedness and management, International Journal of Information Systems for Crisis Response and Management 1 (1) (2009) 12–28.
[43] S.A. White, D. Miers, BPMN Modeling and Reference Guide, Future Strategies, Wallsend, UK, 2008.
[44] WHO — World Health Organization, Global Influenza Preparedness Plan (WHO/CDS/CRS/GIP/2005.5), 2005. (Geneva, Switzerland).
[45] M.H. Zack, The role of decision support systems in an indeterminate world, Decision Support Systems 43 (2007) 1664–1674.

Daniela Fogli received the Laurea degree in Computer Science from the University of Bologna, Italy, in 1994 and the Ph.D. degree in Information Engineering from the University of Brescia, in 1998. From 1998 to 2000, she was a post-doc grant holder at the Joint Research Centre of the European Commission. Since 2000, she is assistant professor at the Department of Information Engineering, Faculty of Engineering, University of Brescia, Italy. She has published more than 70 scientific papers. Her research interests include methods for designing complex interactive systems, meta-design, end-user development, web usability and accessibility, and system personalization through artificial intelligence techniques. She performed her research activity in collaboration with scholars of many Italian universities and of the National Research Council. She also collaborated with the members of the Center for LifeLong Learning & Design (L3D), University of Colorado at Boulder, USA, where she has been a visiting scholar several times. She served as short papers co-chair of the ACM Conference Advanced Visual Interfaces (AVI'06) and doctoral consortium co-chair for the 3rd International Symposium on End-User Development (ISEUD 2011). She is member of SIGCHI Italy (the Italian Chapter of ACM SIGCHI — ACM's Special Interest Group on Computer–Human Interaction).

Giovanni Guida received the Laurea degree (Dr.Ing.) in Electronic Engineering (major in computer science) from Politecnico di Milano, Italy, in 1975.
He is currently a full professor of Computer Science at the Faculty of Engineering, University of Brescia, Italy and a principal investigator of the “Knowledge Engineering and Human–Computer Interaction” research group at the Department of Information Engineering of the University of Brescia.
Previously, he taught at the Politecnico di Milano (1984–1986) and at the University of Udine (1979–1983, 1987–1990), where he founded and directed a research group in the area of artificial intelligence for ten years (1980–1990). He has been the director of the computing center of the University of Udine (1984–1985, 1989–1990) and of the University of Brescia (1991–1993).
His present research interests include knowledge-based systems, decision support, uncertain reasoning, argumentation theory, human–computer interaction, knowledge engineering methodologies and applications, collaboration and cooperation, and web site design. Formerly, he has carried out research activity in the fields of formal languages, programming languages and systems, problem solving and searching, natural language processing, and intelligent information retrieval.
Dr. Guida is active as an independent consultant in the area of information technology applications and knowledge management. He teaches advanced courses and conducts executive seminars in the field of emerging information technologies.