ANAIS PROCEEDINGS

CBSOFT * 2015

BRAZILIAN CONFERENCE ON SOFTWARE: THEORY AND PRACTICE

BELO HORIZONTE

SCrowd 2015

1st WORKSHOP ON CROWDSOURCING SYSTEMS

CBSOFT.ORG

Sponsors:

ThoughtWorks · AvenueCode · Google Brasil · take.net · RAROLABS · Avanti Negócios e Tecnologia · IBM

CNPq Conselho Nacional de Desenvolvimento Científico e Tecnológico · CAPES · FAPEMIG

Promotion:

SBC Sociedade Brasileira de Computação

Organizing Institutions:

UFMG · PUC Minas · CEFET-MG

SCrowd 2015

1[st] WORKSHOP ON CROWDSOURCING SYSTEMS
September 23[rd], 2015
Belo Horizonte – MG, Brazil

ANAIS | *PROCEEDINGS*

COORDENADORES DO COMITÊ DE PROGRAMA DO SCrowd 2015 | *PROGRAM COMMITTEE CHAIRS OF SCrowd 2015*
TRILHA DE TRABALHOS TÉCNICOS | *TECHNICAL RESEARCH TRACK*
Elisa Yumi Nakagawa (ICMC/USP)
Adolfo Duran (UFBA)

COORDENADORES GERAIS DO CBSOFT 2015 | *CBSOFT 2015 GENERAL CHAIRS*
Eduardo Figueiredo (UFMG)
Fernando Quintão (UFMG)
Kecia Ferreira (CEFET-MG)
Maria Augusta Nelson (PUC-MG)

REALIZAÇÃO | *ORGANIZATION*
Universidade Federal de Minas Gerais (UFMG)
Pontíficia Universidade Católica de Minas Gerais (PUC-MG)
Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)

PROMOÇÃO | *PROMOTION*
Sociedade Brasileira de Computação | Brazilian Computer Society

APOIO | *SPONSORS*
CAPES, CNPq, FAPEMIG, Google, RaroLabs, Take.net,
ThoughtWorks, AvenueCode, AvantiNegócios e Tecnologia.

APRESENTAÇÃO

Esta coletânea reúne os artigos selecionados para apresentação no I Workshop de Sistemas de Crowdsourcing (SCrowd2015). Nessa primeira edição, o foco principal édirecionado às aplicações de crowdsourcing relacionadas aogerenciamento de crises e emergências. O objetivo principal é explorar como o crowdsourcing pode ser utilizado, juntamente com diversas soluções computacionais, no gerenciamento de emergências e desastres, sobretudo em áreas com aglomeração de pessoas.

O SCrowd2015 ocorre em conjunto com o CBSoft2015, em Belo Horizonte, Minas Gerais. O CBSofté um evento promovido pela Sociedade Brasileira de Computação (SBC) e, nesta sexta edição, consolida-se como agregadordos eventos mais tradicionais organizados pela comunidade brasileira de desenvolvimento de software: o Simpósio Brasileiro de Engenharia de Software (SBES); o Simpósio Brasileiro de Linguagens de Programação (SBLP); o Simpósio Brasileiro de Métodos Formais (SBMF); e o Simpósio Brasileiro de Componentes, Arquiteturas e Reutilização de Software (SBCARS).

Emsua primeira edição, o SCrowd selecionou cinco artigos, sendo quatroartigos completos e um artigo resumido. A apresentação dos artigos está dividida em duas sessões técnicas. Os autores do melhor artigo do SCrowd2015 serão convidados a submeter uma versão estendida para publicação em edições especiais dos Workshops do CBSoft2015, a serem publicadas no Journal of Brazilian Computer Society (JBCS) ou no Journal of Internet Services and Applications (JISA), ambos editados pela SBC. Os autores do artigo classificado serão anunciados na sessão de encerramento do SCrowd2015.

Como palestrante convidado do workshop, temos o Professor Marcos R.S. Borges, professor titular da Universidade Federal do Rio de Janeiro (UFRJ), que iráproferir a palestra intitulada“Lidando com eventos inesperados na resposta a emergências: o valor do crowdsourcing”. Essa palestra explora as possibilidades do crowdsourcing como fonte inestimável de informações contextuais, visando aumentar a eficácia das operações de resposta às emergências.

Gostaríamos de agradecer a todos que contribuíram para a realização deste workshop. Somos especialmente gratos ao Professor Marcos R. S. Borges, que prontamente aceitou nosso convite para ser o palestrante convidado. Agradecemos imensamente aos autores dos artigos, aos participantes do painel, ao comitê de programa, ao comitê diretivo e aos organizadores do CBSoft 2015, que acolheram esta iniciativa e não pouparam esforços para tornar este workshopum sucesso. Em especial, agradecemosao Daniel Soares Santos e ao Marcelo Benites Gonçalves, ambos alunos do ICMC/USP, pelo apoio durante todo o processo de organização do workshop.

A realização deste workshop contou com o apoio financeiro do CNPq, por meio do projeto de pesquisa RESCUER. O RESCUER é uma parceria entre a União Europeia e o

Brasil que visa desenvolver uma solução de crowdsourcing para apoiar centros de comando em emergências e no gerenciamento de crises.

Desejamos a todos um excelente workshop!

Belo Horizonte, setembro de 2015.

Elisa Yumi Nakagawa (ICMC/USP)

Adolfo Duran (UFBA)

Organizadores do SCrowd 2015

FOREWORD

This collection gathers the selected papers for presentation at the I Workshop onCrowdsourcing Systems (SCrowd 2015). In this first edition, the main focus is the crowdsourcing applications related to crisis and emergencies management. The main objective is to explore how crowdsourcing can be utilized, alongside several computational solutions, in the management of disasters and emergencies,specially in areas with high concentrations of people.

SCrowd 2015 occurs within the CBSoft 2015, in Belo Horizonte, Minas Gerais. CBSoft is an event promoted by the Brazilian Computer Society (SBC) and, inits sixth edition, it consolidates itself as a force that brings together the most traditional conferences organized by the Brazilian community of software development: the Brazilian Symposium on Software Engineering (SBES); the Brazilian Symposium on Programming Languages (SBLP); the Brazilian Symposium on Formal Methods (SBMF); and the Brazilian Symposium on Components, Architectures and Software Reutilization (SBCARS).

In its first edition, SCrowd selected five papers, four of themas full papers and a fifth as short paper. The presentation of the papers is divided into two technical sessions. The authors of the best paper in SCrowd 2015 will be invited to submit an extended version for publication inspecial issues of CBSoft 2015 workshops, which will be published in the Journal of Brazilian Computer Society (JBCS) or in the Journal of Internet Services and Applications (JISA), both published by SBC. The best paper will be announced in the closing session of SCrowd 2015.

Our keynote speaker is Professor Marcos R. S. Borges, a full professor at Federal University of Rio de Janeiro (UFRJ). His talk is entitled “Dealing with unexpected events in emergency response: The value of crowdsourcing” and explores the possibilities of crowdsourcing as a valuable source of contextual information, aiming at increasing the efficiency of emergency response operations.

We would like to thank everyone who contributed to the organization of this workshop. We are especially thankful to Professor Marcos R. S. Borges, who readily accepted our invitation for the keynote speaker. We are also thankful to the papers’ authors, participants of the panel session, program committee members, and steering committee.We also thank the organizers of CBSoft 2015, which welcomed this initiative and worked restlessly to make this workshop a success. In particular, we thank Daniel Soares Santos and Marcelo Benites Gonçalvez, both of them students at ICMC/USP, for the support throughout the process of organizing thisworkshop.

CNPq provided financial support for this workshop through the RESCUER research project. RESCUER is a collaboration between Brazil and the European Union that aims to develop a crowdsourcing solution to support command centers in emergencies and in crisis management.

We wish everyone an excellent workshop!

Belo Horizonte, September 2015.

Elisa Yumi Nakagawa (ICMC/USP)

Adolfo Duran (UFBA)

SCrowd 2015 Organizers

COMITÊ DE ORGANIZAÇÃO | *ORGANIZING COMMITTEE*

SCrowd 2015 GENERAL CHAIRS
Elisa Yumi Nakagawa (ICMC/USP)
Adolfo Duran (UFBA)

CBSoft 2015 WORKSHOPS CHAIR
FábioMascarenhas (UFRJ)

WEBSITE AND SUPPORT
Daniel Soares Santos (ICMC/USP)
Marcelo Benites Gonçalves (ICMC/USP)

COMITÊ TÉCNICO | *TECHNICAL COMMITTEE*

COORDENADORES DO COMITÊ DE PROGRAMA DA TRILHA DE TRABALHOS TÉCNICOS | *TECHNICAL RESEARCH PC CHAIR*

Elisa Yumi Nakagawa (ICMC/USP)
Adolfo Duran (ICMC/USP)

COMITÊ DIRETIVO | *STEERING COMMITTEE*

Adolfo Duran (UFBA)
Elisa Yumi Nakagawa (ICMC/USP)
Karina Villela (Fraunhofer-IESE, Alemanha)
Vaninha Vieira (UFBA)

COMITÊ DE PROGRAMA | *PROGRAM COMMITTEE*

Agma Traina (ICMC/USP)
Cláudio Sant'anna (UFBA)
Eduardo Almeida (UFBA)
Lais Salvador (UFBA)
João Porto de Albuquerque (ICMC/USP)
José Fernandes Junior (ICMC/USP)
Juan Torres Arjona (UPM, Espanha)
Karina Barreto Villela (Fraunhofer-IESE, Alemanha)
Manoel Gomes de Mendonça Neto (UFBA)
Rafael Prikladnicki (PUCRS)
Renato Novais (IFBA)
Rita Suzana Pitangueira Maciel (UFBA)
Vaninha Vieira (UFBA)

PALESTRA CONVIDADA | *INVITED TALK*

Dealing with unexpected events in Emergency Response: The value of crowdsourcing

MarcosBorges (UFRJ)

Abstract: Rescue operations in most disasters are heavily dependent on accurate situational awareness. The environment can change dramatically and may be very difficult to plan an effective rescue operation without a situational awareness that indicates what and where to look for victims. Technology, such as satellite images, can help, but they can take time to retrieve and they lack important contextual data. That is where crowdsourcing can play an important role. We claim that "human sensors" with adequate support can be an invaluable source of contextual information. This presentation explores this issue, presenting some initiatives that attempt to make use of crowdsourcing information in order to increase the efficacy of the emergency response operations.

Marcos R.S. Borges is Full Professor of Computer Science at the Federal University of Rio de Janeiro, Brazil. He earned his doctorate in Computer Science from the University of East Anglia, UK in 1986. From 1994 to 1996, he was a visiting research scholar and a member of the Object Technology Laboratory at Santa Clara University, California, USA. Dr. Borges has also served as Visiting Professor at University of Paris VI (2001) and at Polytechnic University of Valencia, Spain (2004–2005). His research interests include CSCW, Group Decision Support Systems, Resilience Engineering and Collective Knowledge. Since 2004 he has been working in the emergency management domain and has published the latest results on this topic in journals and conferences such as: Computers in Industry, Journal of Network and Computer Applications, Journal of Decision Systems, Reliability Engineering and System Safety, Journal of Loss Prevention in the Process Industries and Information Systems for Crisis Response and Management (ISCRAM).

TRILHA DE ARTIGOS TÉCNICOS | *TECHNICAL RESEARCH TRACK PAPERS*

Crowd Steering in Emergencies and Crisis Situations

Ana Maria Amorim[1], Pedro Kislansky[1], Renato Novais[1,2], Vaninha Vieira[3]

[1]Fraunhofer Project Center for System Engineering and Systems
Parque Tecnológico, Salvador – BA – Brazil

[2]Department of Computer Science – Federal Institute of Bahia
Bahia, Brazil

[3]Department of Computer Science – Federal University of Bahia
Bahia, Brazil

{ana.amorim,pedro.kislansky,renato.novais,vaninha.vieira}@fpc.ufba.br

Abstract. *In emergencies and crisis, the main objective is to preserve the lives of involved people. Crowd steering is a key step to improve efficiency survivability. In the past, flaws in the process of crowd steering caused the loss of many lives. The purpose of this paper is identify the difficulties to guide people in emergencies and crisis, and present to the decision makers a way to communicate with a heterogeneous crowd. The proposed approach is an ongoing work of the RESCUER project. This project uses crowdsourcing as a source of information and bilateral communication. It focuses on emergencies in the scenario of industrial areas and large-scale events. The survey process was held in the Industrial Park of Linz, Austria, and Camaçari, Bahia.*

1. Introduction

In places with high concentration of people, an evacuation might safeguard the crowd in case of an emergency. Fast evacuation is vital to prevent casualties. Real evacuations in the past indicated that the process is slow and difficult to coordinate due to injured people or people searching for a way to make themselves safely (HOFINGER *et al.*, 2014). A sad example of crowd disaster occurred during a music festival on 24 July 2010, in Duisburg, Germany, where 21 people died and at least 510 were injured (VREUGDENHIL *et al.*, 2015).

In crises and emergencies, collaborative solutions as RESCUER are being proposed (VILLELA *et al*., 2013). The RESCUER project aims to develop an intelligent solution based on computer, to support emergency and crisis management, with a special focus on incidents that occur in Industrial areas or Large-scale events. The main source of information in RESCUER is the citizen, who, through their cell phone, communicates an emergency situation to the command centre. The command centre consists of a group of people responsible for evaluating the situation and making decisions in such cases. One of the desired results of the project is the reduction of damage to people's lives, and crowd steering is a key step in this process. Besides this, the project aims to improve the image of the organizations involved in an incident due to the use of modern technologies for effective and efficient management of emergency and crisis.

This paper reports the results of crowd steering requirements elicitation activities we conducted in the scope of the RESCUER. The objectives of this paper are: 1) identify the difficulties to guide people in emergencies and crisis situations; and 2) present to decision makers the RESCUER solution for crowd steering, a way to communicate with a heterogeneous crowd. The novelty is the use of a crowdsourcing system to help people at hazard condition. Crowd steering, in the context of this article, is the action of guide people to a safe location, taking into account the presence of crowded areas.

The paper is organized as follows: Section 2 introduces concepts for crowd steering in emergencies. Section 3 presents methods used for eliciting the requirements regarding crowd steering. Section 4 presents the RESCUER solution for crowd steering in emergencies and crisis situations. Section 5 discusses related works. Finally, Section 6 presents our final considerations and future work.

2. Basic Concepts

According to (MONTAGNA *et al.*, 2013), crowd steering is the action of "guide a group of people towards locations, along optimal paths and taking into account contextual information describing the presence of crowded areas which should be dynamically intercepted and circumvented".

The RESCUER is aware of an incident as soon as the crowd sends a report. In this context a **report** is a message that contains the description of a hazard situation depicting the emergency details from someone's point of view. The report corresponds to a spontaneous communication from the crowd to inform the occurrence of an incident and can be updated periodically. There are two types of report: simple and complete. A simple report is known as incident notification: with only one click the user of the mobile sends an incident notification. A complete report is known as incident report: besides photos and videos, it has some pre-defined questions answered by the user.

Incident is a natural or man-made occurrence that interrupts the normal procedure or behaviour in a certain situation. It may cause a critical or emergency situation that requires measures to be taken immediately to reduce adverse consequences to life and property. It can affect the image of the business and/or of the country. For the RESCUER system, an incident is a set of reports related to the same incident type (e.g. fire, explosion, gas leakage or environmental), occurring at the same area and time.

In the RESCUER context, as an incident is composed of a series of reports, likewise an emergency is formed by a set of incidents reported in different positions and time. **Emergency** is a critical situation caused by an incident, natural or man-made that requires measures to be taken immediately to reduce their adverse consequences to life and property. Examples of natural incidents are floods, wildfires, and snow storms, whereas examples of man-made incidents are fires, explosions and/or substance spills from oil platforms, ships, and factories.

In (BOIN; LAGADEC, 2000) a situation is defined as a crisis if something out of the ordinary happens. In RESCUER, the adverse and therefore undesired consequences of an emergency give rise to a crisis.

3. Methods Used for Crowd Steering Requirements Elicitation

In order to choose the most suitable elicitation requirements technique we must know the stakeholders of the system and their areas of expertise. For crowd steering requirements elicitation of the RESCUER, we have chosen a hybrid approach, composed of: ethnographic observation, workshop, and brainstorm. In addition, we have considered two main scenarios: industrial areas and large-scale events and we have divided each one into two contexts: Brazil and Europe.

The first step was the ethnographic observation (EO). EO study is a technique borrowed from social studies and has the goal of observing the interactions between users at their place of work. It is very effective when investigating collaborative work (ZOWGHI; COULIN, 2005). The EO was made only in Brazil for both scenarios. The first, during an evacuation simulation that occurred at the industrial park of Camaçari (Brazil), and the second was at the operational forces' command and control centre during the Football World Cup - June 2014. The EO aimed to understand the overall process.

Considering the industrial area scenario, we already participated in three simulations at Camaçari, Bahia. The first occurred in October 2014, in storage areas, simulating the leakage of liquefied petroleum gas (propane) with explosive cloud formation. In this event, there was a need for evacuation of workers in several nearby units, and blockage of access to the local roads. The other two simulations took place in November 2014, one in the city of Dias D'Ávila and the other in the city of Camaçari. In these opportunities, the main objective was the training of residents of local communities in a situation where there was a need for evacuation. During the simulations we could observe that the crowd steering process is currently done using phone or via radio. Several times we noticed that the messages exchanged between them were unclear and often confusing due to problems such as side conversations and information coming from various points.

In April 2015, we conducted a workshop with the end users of the project partners from Brazil and Europe. During this workshop, we held an activity that involved emergency specialists from both continents in order to understand the requirements for crowd steering. The activity followed five steps:

- Explain the meaning of "crowd steering" in the project context;
- Explain the activity that will be done;
- Perform a brainstorm about the information that has to be sent to each stakeholder related to crowd steering;
- Identify the stakeholder responsible for providing the information; and
- Ask them to link the information to the corresponding phase of emergency: detected, being combated, under control, and finished.

The explanation of the current status of the project had the purpose of establishing a common view and understanding about the project. Then the activity itself was explained to all specialists.

Afterwards, we used several paperboards. On those paperboards, with the help of specialists, we wrote the name of the stakeholders identified. At this point, we started the brainstorm; each specialist was invited to say freely what kind of messages and content they would like to send to the stakeholders (guidance) during an emergency.

After defining the messages we asked them to link the messages with four pre-defined emergency phases: detected, being combated, under control and finished. Additionally, we established with the specialists which stakeholder was responsible for providing the information.

Finally, after compiling the workshop's results, we sent them to the end-users for validation. The same process was repeated in Europe.

4. Crowd Steering RESCUER Solution

During an emergency situation, it is necessary to inform people in the affected area about what is going on and how to behave safely. The RESCUER solution for crowd steering is based in the communication from the command centre to the crowd with the goal of reducing the impact of the critical situation.

Differently from industrial park, in large-scale events it is important to consider that people generally are not prepared to face an emergency situation. In industrial parks, despite employees being extensively trained to deal with such situations, emergency-related tasks are not part of their daily work. Thus, in order to not worsen their situations, any information or guidance should be restricted to the minimum necessary and should contain clear, small, and simple actions.

In an industrial park scenario, we identified three target stakeholders: employees, visitors and neighbouring community. In a large-scale event scenario, we identified two target stakeholders: civilians and employees. At different phases of the emergency, it is important to communicate with those stakeholders. Table 1, for example, shows information to be sent to visitors in emergencies at an industrial park; and Table 2 shows information to be sent to civilians in a large-scale event scenario. The first column corresponds to the information that the system has to provide, and the second column corresponds to the emergency phase in which the information should be sent. According to the necessities, the command and control centre can select any of this information to send to a target stakeholder or a group of stakeholders.

Table 1. Industrial Park - Visitors Information

Information to Visitors	Phase
Go to the meeting point	Being Combated
Follow the evasion coordinator's instructions	Being Combated
Keep calm; do not run, walk	Being Combated
If necessary, use the escape mask	Being Combated
If you are driving, stop the car	Being Combated
Unit "X" in emergency	Being Combated
The situation is controlled	Under Control
The emergency is over	Finished
Return to your unit	Finished

Table 2. Large-scale Event - Civilians Information

Information to Civilians	Phase
Please, go to gate "X"	Detected and Being Combated
Keep calm	Detected and Being Combated
In case of mobility difficulties, follow the brigade orientation	Detected and Being Combated
Do not use elevator, use stairs	Detected and Being Combated
Follow the brigade orientation	Detected and Being Combated
Situation under control, keep calm, stay where you are	Under Control
Incident finished, wait for further orientation	Finished
Return to your place	Finished

Using the requirements elicitation methods previously described, we could identify difficulties that arise, during an emergency situation, when there is the necessity to guide people during an evacuation. The main difficulties identified were the following:

- identify groups of people that have to be guided;
- know which information must be given to each identified group of people;
- identify who is responsible to provide the information;
- identify the phase of emergency in which the information should be available.

We could also identify the difficulty that the emergency managers have to communicate with the crowd. Nowadays they use mainly radio services and the press. Often this type of communication was inefficient, causing misunderstandings and generating even more tumult among the people involved. With the RESCUER project we hope to provide, to each identified group of people, the information that they need at the right moment.

5. Related Works

Crowd steering is one of the major concern during an emergency situation, especially during evacuation process. There are many researches aiming to understand the crowd's behaviour and to improve the communication between operational forces and the crowd (MARSH *et al.*, 2014) (HOFINGER *et al.*, 2014) (CHALLENGER *et al.*, 2009) (RADIANTI *et al.*, 2013).

One theory seems to prevail over the others: Contagion and Predisposition. The first implies that an individual inside a crowd, during a crisis, loses his ability to decide rationally, then he is influenced by the crowd collective behaviour. The second complements the first by claiming that the crowd behaviour is a result of a pre-existing tendency (e.g. violence or coward behaviour) (RADIANTI *et al.*, 2013).

Moreover, several systems were created for the purpose of guiding people to safety. LifeBelt (FERSCHA; ZIA, 2009) is a wearable computer that can collect one's position and provide it to an emergency coordination system. The system guides the individual using an optimized escape plan and recommends the nearest exit via vibrations.

The Intelligent Evacuation Guidance System (IEGS), produced in China, is another example. The IEGS considers three parameters for deciding the optimal escape route: human behaviour, construction parameters, and the smoke state. During the guiding process [evacuation], the system receives other input data, such as [personnel] crowd density and speed; therefore, it can optimize the previous decisions dynamically (RAN *et al.*, 2014).

The Emergency Support System (ESS) uses opportunistic communication for exchanging packages at close range. This discards the need for a conventional infrastructure. In addition, it seeks for the shortest path to exits. The system uses fixed sensor devices previous located at the monitored building in a close range, so they can communicate among themselves and with the mobile devices. Because GPS information inside buildings are not reliable (visualization indoor), the mobile location is extract from triangulation of the pre-fixed sensors (KOKUTI; GELENBE, 2014).

The described solutions above are based on wearable devices or previously installed sensors. The RESCUER system does not need such infrastructure or previous preparation. It relies on a set of short and accurate information provided by the command and control centre at a specific time to a target stakeholder or group of stakeholders. The system seeks to mitigate the problem of contagion and predisposition, mentioned above. Our proposal suggests that once the crowd recognizes that the message comes from a trusted source and contains only concise and clear information, it behaves in a calm and organized manner.

6. Conclusion and Future Work

Evacuation planning, training and real time decision making are critical to safety. Fast smart techniques to provide on-line information about crowd density, crowd movements and the predicted duration of historical crowd steering planned process is very important for the evacuation decision making process.

This paper is therefore relevant as it presents useful findings following a knowledge elicitation study that was conducted across different fire stations in Europe and Brazil. The crowd is guided by predefined messages in different phases of an emergency. The purpose is to deliver the accurate information at the specific time, improving the communication between the crowd and the operational forces, and avoiding misunderstandings. To evaluate and improve our existing work, the guidance obtained by our approach can be incorporated in crowd simulations. Additionally, it is necessary to create a set of reliable metrics for evaluation plans. These items are part of our future work.

Acknowledgments

This work is supported by the RESCUER project, funded by the European Commission (Grant: 614154) and by CNPq/MCTI (Grant: 490084/2013-3).

References

BOIN, A.; LAGADEC, P. Preparing for the Future: Critical Challenges in Crisis Management. **Journal of Contingencies & Crisis Management**, v. 8, n. 4, p. 185, 2000.

CHALLENGER, R.; CLEGG, C. W.; ROBINSON, M. A. **Guidance and Lessons Identified**. [S.l: s.n.], 2009.

FERSCHA, A.; ZIA, K. LifeBelt: Silent directional guidance for crowd evacuation. **Proceedings - International Symposium on Wearable Computers, ISWC**, p. 19–26, 2009.

HOFINGER, G.; ZINKE, R.; KÜNZER, L. Human Factors in Evacuation Simulation, Planning, and Guidance. **Transportation Research Procedia**, v. 2, n. 0, p. 603–611, 2014.

KOKUTI, A.; GELENBE, E. Directional Enhancements for Emergency Navigation. 2014.

MARSH, K.; WILKIE, C.; LUH, P.; *et al.* Crowd Guidance in Building Emergencies: Using Virtual Reality Experiments to Confirm Macroscopic Mathematical Modeling of Psychological Variables. **Pedestrian and Evacuation Dynamics 2012**, v. 2012, p. 197–212, 2014.

MONTAGNA, S.; VIROLI, M.; FERNANDEZ-MARQUEZ, J. L.; MARZO SERUGENDO, G. DI; ZAMBONELLI, F. Injecting self-organisation into pervasive service ecosystems. **Mobile Networks and Applications**, v. 18, n. 3, p. 398–412, 2013.

RADIANTI, J.; GRANMO, O. C.; BOUHMALA, N.; *et al.* Crowd models for emergency evacuation: A review targeting human-centered sensing. **Proceedings of the Annual Hawaii International Conference on System Sciences**, p. 156–165, 2013.

RAN, H.; SUN, L.; GAO, X. Influences of intelligent evacuation guidance system on crowd evacuation in building fire. **Automation in Construction**, v. 41, p. 78–82, 2014.

VILLELA, K, V VIEIRA, M MENDONÇA, T FRANKE, J TORRES, and S GRAFFY. “RESCUER-Dow - Reliable and Smart Crowdsourcing Solution for Emergency and Crisis Management.” Munich, 2013.

VREUGDENHIL, B. J.; TORINO, P.; TOWNSEND, P. S. Using Crowd Modelling in Evacuation Decision Making. 2015.

ZOWGHI, D.; COULIN, C. Requirements elicitation: A survey of techniques, approaches, and tools. **Engineering and Managing Software Requirements**, p. 19–46, 2005.

Architecting Crowdsourcing Systems: Challenges and Solution Ideas from the RESCUER Project

Tassio Vale[1], Taslim Arif[2], Laia Gasparin[3], Vaninha Vieira[4]

[1]Federal University of Recôncavo da Bahia
Rua Rui Barbosa, 710, Centro - Cruz das Almas, Bahia, Brazil

[2]Fraunhofer IESE
Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany

[3]VOMATEC Innovations GmbH
Riegelgrube 7, 55543 Bad Kreuznach, Germany

[4]Federal University of Bahia
Avenida Luiz Viana Filho, KM 10, Salvador, Brazil

tassio.vale@ufrb.edu.br, taslim.arif@iese.fraunhofer.de,

laia.gasparin@vomatec-innovations.de, vaninha@ufba.br

Abstract. *The RESCUER system aims to create a smart and interoperable crowdsourcing solution to support emergency and crisis management, focusing on incidents in industrial areas and at large-scale events in Europe and Brazil. In this paper, we discuss about a set of key software architecture challenges for such a system considering different points of view (e.g. technical, functional, project settings, etc.). We believe our experiences will greatly benefit other organizations engaged in this kind initiatives of systems, since they might save valuable time and effort by discovering problems at a very early stage in the project.*

1. Introduction

Nowadays, most people use mobile devices and share their status and information about what is happening around them in real time. This phenomenon can help in an emergency situation, allowing a crowd with mobile devices to send detailed information to the command and control center. This information is important for the operational forces because it is possible to understand what is happening in real time, getting information from the place of the emergency. Aiming to fill the gap between mobile users and emergency coordinators, the RESCUER system aims to develop a smart and interoperable information system that provides support in an emergency situation using crowdsourcing information.

In this paper, we report a set of architectural challenges identified during the course of the RESCUER project. We have also identified a set of solution concepts that could potentially address these challenges. Reported architectural challenges are the key for successful software development projects, since money and effort are saved if appropriate measures are taken to tackle the architectural challenges. We hope that our findings will make the architects of similar systems aware of all those traps that we learned to avoid over time in the hard way.

This paper is structured as follows: Section 2 describes high-level architectural views of the RESCUER system. This is necessary to understand the context of the system and to understand the challenges better. In Section 3, we describe the architectural challenges and categorize them according to the architectural viewpoints. Section 4 presents the high-level solution concepts and how they are mapped to our challenges. Finally, Section 5, we conclude our findings.

2. RESCUER System

The RESCUER system intends to provide faster and more accurate management in emergency and crisis situations by achieving: improved time to collect information regarding an emergency situation; decreased time and effort to analyze emergency data; improved and reliable information provided to different stakeholders within the shortest possible time; context-aware interaction with different stakeholders; minimized effort among various workforces; and efficient management of the emergency through smart crowd steering and effective management of the workforces on site.

In order to understand the architectural requirements, we identified the key RESCUER stakeholders from the architectural point of view: civilians at large events, employees of industrial parks, workforces, affected people, command and control center, authorities, and press. They are located within the incident area (on-site), very close to the incident area (on-site-nearby), or far from such places (off-site). The characteristics of the stakeholders as well as their current location influence the decisions to mitigate the emergency and should be considered in order to develop a suitable system.

The goals and requirements defined in conjunction with the project stakeholders, which guided the construction of the RESCUER architecture. We adopted the Fraunhofer ACES approach [Keuler et al. 2011] as the software architecture construction process. As result, the architecturally significant requirements were: development-related requirements, integration requirements, availability, robustness, scalability, reliability, performance, usability, security, safety, operability, upgradeability, auditability, and variability.

Aiming to give an overall understanding of RESCUER and its main architectural components, we present the sub-systems perspective (Figure 1). The system is divided into five sub-systems: Mobile Solution, Communication Infrastructure, Data Analysis, Emergency Response Toolkit, and Integration Platform. shows the interaction among stakeholders and sub-systems.

The Mobile Solution sub-system explores the use of mobile devices to gather information from the crowd in an emergency situation and to support follow-up interactions in an optimized and context-sensitive way. The resulting mobile application interacts appropriately to avoid cognitive overload for the users and to get people engaged in using the RESCUER system.

Aiming to support the information flow between the crowd and the command center, the Communication Infrastructure includes a server for receiving, synchronizing, organizing, and storing crowdsourced data from the users' mobile devices. In addition, this sub-system provides a solution for delivering messages to the users' mobile devices in a personalized, location- and situation-sensitive way. This sub-system is also responsible for providing peer-to-peer communication by using the built-in Wi-Fi capability of mobile devices if no Internet connection is available during the emergency.

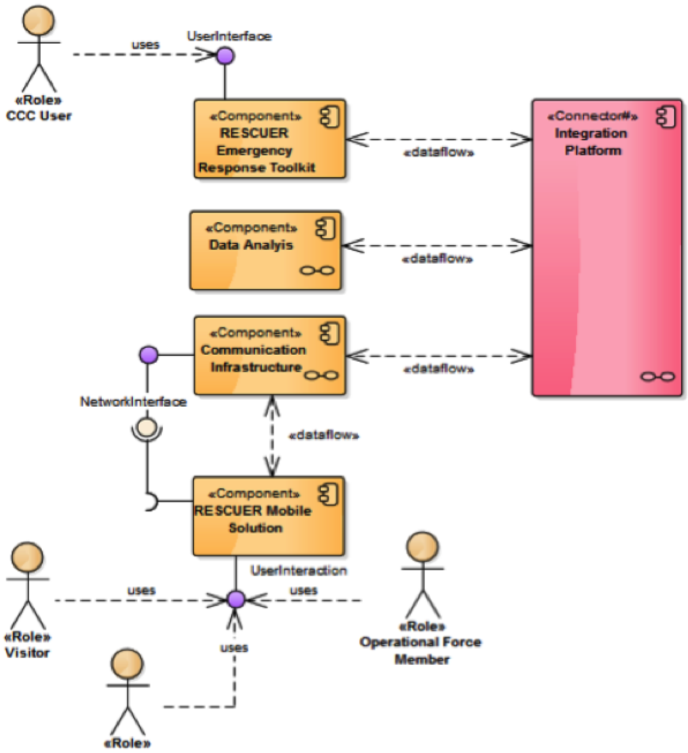


Figure 1. Sub-systems perspective.

Automatic data analysis is especially relevant for emergencies in large-event scenarios, where emergency reports from thousands of people are sent to the command center. The Data Analysis sub-system receives data to be fused and filtered in order to obtain an enriched collection of data about the emergency situation and thereby enable a more robust and efficient analysis.

The Emergency Response Toolkit supports decision-making, coordination of responses, and communication with stakeholders. This sub-system provides appropriate data visualization mechanisms through an intuitive, concise, but resourceful dashboard with modern solutions to map an incident scenario. It also includes a semi-automated solution for communication with the community to provide timely, coordinated, and accurate information about the nature and status of an emergency situation.

In order to assure consistent and efficient interaction among the other sub-systems, the Integration Platform provides a communication protocol, storage, and technology to handle message (called topics) exchanges. In general, the sub-systems interact by publishing and subscribing appropriate topics to the Integration Platform, which enables normal system execution.

2.1. Development Settings

The RESCUER project fosters cooperation among companies, research institutions, and universities from Brazil, Germany, and Spain. In addition, this project has partners from industrial parks in Brazil and Austria to validate the proposed solution in a real-world scenario. Software development is being performed by the following members: MTM, DFKI, VOMATEC, Universidad PolitÃ©cnica de Madrid (UPM), University of SÃ£o

Table 1. Development task allocation among partners.

Development team	Responsibilities
MTM	Ad-hoc P2P Network, Sensor Data Recorder, Sensor Data Receiver, Sensor Data Analysis, SMS Data Receiver and Text Analysis.
VOMATEC	Emergency Response Toolkit, Combined Analysis, and Integration Platform.
UPM	Video Analysis.
USP	Image Analysis.
Fraunhofer	Infrastructure tasks (requirements engineering, software architecture specification, etc.)
UFBA	Social Media Connector (SMC), Legacy System Connector (LSC), and infrastructure tasks.

Paulo (USP), and Federal University of Bahia (UFBA). Table 1 presents the responsibilities for each RESCUER development team.

3. Architectural Challenges

A sound architecture is the key to success for any software-intensive system. It is better to know architectural challenges early so that architects can address them properly. The later such challenges are known, the more must be spent on fixing them, as architectural refactoring cannot be done in one afternoon. From the experience with the RESCUER project, we have identified a number of crucial architectural challenges and have clustered them according to several viewpoints and perspectives of the system.

3.1. Context Viewpoint Challenges

- *AC_01 - Unclear Context Interaction*: It is of utmost importance to define the boundary of the system. It is always a challenge to identify what is inside the system and what is outside of it. One key Litmus test would be to identify whether we have any influence as architects or whether we can change anything. If the answer is yes, it is in the system scope; otherwise, it is in the context. Using this approach, it has been possible to clearly delineate the context. But the challenge we faced mostly regards the business view of the system. There was a requirement to integrate the legacy systems of the operational forces with the RESCUER system. But due to the closeness of these systems, it was not possible to integrate them with our proposal. Operational forces cannot foresee the future scenarios of the system and therefore cannot provide input on what should be integrated or how a system could be integrated. Due to these vague requirements, it is extremely hard for architects to design this challenge. If the issue were merely to import/ export data exchanges between the systems, it would not be so architecturally significant. Integration could be done in real time and two systems could influence and use each other. If the latter is the case, the architecture has to be significantly different to support it;

3.2. Functional Viewpoint Challenges

- *AC_02 - Not possible to compete with industry*: Expectations from the stakeholders regarding this kind of project are huge. This is mostly because they see how big companies are doing cool analyses and they want to have this, too. Everybody wants to be on top of what is available. They often forget how much money and effort those organizations spend to get to this stage. The amount of knowledge and know-how these companies have cannot be gained instantly. This expectation management is a big challenge;
- *AC_03 - Requirements and analysis meeting points*: Defining functionality can be done in two ways - making a wish concerning the system or thinking about what is possible and, based upon that, defining the functionalities. In some cases, it happens that two groups wait for each other. The technical team may say,_ "Give us the requirement and we will see whether it is possible to implement it", while the requirements people may say, "Tell us what is possible and then we will define requirements based on that". As a result of this, requirements become vague and the architecting input is poor;

3.3. Information Viewpoint Challenges

- *AC_04 - Handling various kinds of data*: Several kinds of data need to be handled in these kinds of systems. For example: (1) sensor data that is coming from the crowd's mobile devices in a continuous stream of data (2) reports that are sent from mobile devices, which usually contain structured information and text; (3) multimedia data - images and videos are taken by the crowd and received by the RESCUER backend. Because of the different characteristics of the data, the architecture should handle them differently. One communication channel is not optimal for all sorts of data in the system;
- *AC_05 - Deciding on central or distributed data storage*: It is easy to implement a component when you have a central repository of data and the component accesses it whenever necessary. We have quite a few components that need to be built. In the case of central data storage, if we change the schema, all components will be affected. It is natural for this kind of systems to modify the schema over time along with a deeper understanding of the domain and the requirements. However, if no direct access to the data is allowed, but the components get data through some message broker, these components might need to create a local data storage. Replication and synchronization of these data stores are then additional issues;
- *AC_06 - Creating reliable output data*: It is of utmost importance for an emergency management system to ensure that whatever it provides as output has to be completely trustworthy. Otherwise, human lives might be endangered or damage might be caused to the environment. The system needs to make sure that all the information that is coming from the crowd is analyzed and presented such that it reflects the reality. Moreover, the sources have to be made secure with appropriate mechanisms;

3.4. Development Viewpoint Challenges

- *AC_07 - Each team has its own tools and history*: Every development team has expertise with different technologies and has a different history of development.

Ensuring some level of homogeneity is therefore a very difficult task. it has to be ensured that these heterogeneous technologies work together and, most importantly, that they can evolve at the same speed;

- *AC_08 - Lack of control over individual teams*: In the case of distributed development not done within one company, it is hard to have authority over all teams. Without this, it is hard to impose architectural principles - which is, however, necessary to ensure end-to-end quality. Each team wants autonomy and if this is not managed properly, it is a threat to the overall success of the project;
- *AC_09 - Different plan for commercialization afterwards*: A practical constraint we faced were the different plans for commercialization or evolution of the individual components. Every team wants to have a piece that they will be able to enhance afterwards without the help from other teams. This creates a barrier to an optimal overall architecture. These kinds of systems have a number of important quality requirements and these quality requirements are hard to achieve in these kinds of settings;
- *AC_10 - Integration in distributed setting*: Integration is the key to success for any software system development. In general, integration is one of the painful parts of the software delivery pipeline. In the case of distributed settings, it becomes even more complex. Creating an environment for integration testing, packaging, building, and deploying all the components requires huge coordination effort;
- *AC_11 - Knowledge propagation is harder in iterative development (in the case of distributed settings)*: Knowledge propagation is important to create a common understanding among all the teams. Moreover, efficient knowledge propagation is required for an efficient delivery and deployment pipeline. In the case of iterative development, knowledge propagation becomes a continuous task. It requires a lot of effort to disseminate knowledge effectively. On the other hand, without it, the whole development pipeline becomes chaotic very quickly; and
- *AC_12 - No integrated toolchain to support development*: Setting up a toolchain that supports distributed settings is an absolute necessity to gain speed. Setting up such a toolchain requires effort. Moreover, individual teams have experience with different tools. The toolchain also has to be flexible enough. In addition to that, some teams have restrictions on sharing their code. So the toolchain needs to consider this as well. Despite all these challenges, if we do not have any kind of tool support to facilitate distributed development, testing, and deployment, there will be a lot of trouble in the project.

4. Solution Ideas

We present as follows a set of solution ideas to deal with the aforementioned challenges, and Table 2 shows the relationship among them.

- *SI_01 - DevOps toolchain*: Incorporating DevOps concepts [Bass et al. 2014] would greatly help in our scenario. It would facilitate continuous integration, continuous testing, and continuous delivery with appropriate tool support;
- *SI_02 - Minimum viable product*: Thinking about the overall system makes the architecture definition complicated. The key part of the system that makes it viable needs to be identified, designed, and developed first. For the rest, divide and conquer could bring more focus to architecting and would make the architecting process more efficient;

- *SI_03 - First evaluation, then commercialization*: Designing the system for industrial adoption is a very ambitious goal. The domain is new and a lot of experiences are required to arrive at some reasonable solution. Therefore, designing for commercialization makes the architecture definition even more complicated. There is always a chance for improvement. It is reasonable to know the future requirements but it is important to focus on the evaluation or demonstration scenarios only.
- *SI_04 - Continuous feedback from end-users*: As the domain is not yet well understood, it is not possible to elicit all requirements in the first attempt. Therefore, it is important to continuously show the prototypes or partially implemented system to the end-users. Refining the requirements and making the system acceptable to the community is not possible without continuous feedback;
- *SI_05 - Knowledge propagation mechanism*: Distributed development heavily suffers from the lack of consistent knowledge about the status of project. It is important to identify the bottlenecks of information propagation. Appropriate tool support and other mechanisms are necessary to propagate knowledge;
- *SI_06 - Plan for creating development/test environment*: It is important for such kinds of projects to plan for IT provisioning support. Individual development environments provided by each team are not enough for doing sound integration and acceptance testing;
- *SI_07 - Commitment from operational forces management*: Just like any new software system, this kind of system also needs to overcome barriers in order to be accepted. Of course, for operational forces it is a bit of a change, but they need to see the positive side of the new system. It is not easy to get such cooperation from operational forces. Therefore, commitment from management is crucial. This facilitates fruitful cooperation between end-users and development teams;
- *SI_08 - Twin pics model*: Requirements and architecture activities need to be done in parallel. It is important for these two teams to collaborate closely. Once higher-level requirements decisions are made, architects need to find higher-level solutions. The requirements team needs to refine the requirements with respect to the architectural decisions made so far. In this way, these two teams should continue refining requirements and solutions;
- *SI_09 - Wiki for architecting*: Besides all other knowledge propagation and collaboration tools, it is important that the architecture is accessible to everyone in the team and well understood. To get continuous feedback, it is also important that everyone can comment on what is not working properly. The architecture should be at the center of all communication and feedback. Any tool, such as a wiki, is therefore very helpful;
- *SI_10 - Decouple Components*: Every component needs to be decoupled as much as possible from the rest of the system. A component should provide specific services but it should not be aware of how the service will be consumed;
- *SI_11 - Continuous Testing*: To learn about the current status of the development, it is important to set up an environment and a process for continuous integration; and
- *SI_12 - Context-aware Middleware*: This kind of system needs to use modern analysis and visualization techniques. To do an effective and efficient analysis, it is important to know the context of the current state of the system. It is therefore important to build a context-aware middleware that keeps track of the context and

Table 2. Architectural and solution ideas mapping.

Architectural challenges	Solution Ideas
AC_01 – Unclear context interaction	*SI_04*, *SI_07*, *SI_08*
AC_03 – Not possible to compete with industry	*SI_04*, *SI_07*
AC_04 – Requirements and analysis meeting points	*SI_08*
AC_05 – Handling various kinds of data	*SI_01*, *SI_02*, *SI_09*
AC_06 – Deciding on central or distributed data storage	*SI_03*
AC_07 – Creating reliable output data	*SI_01*, *SI_04*, *SI_06*
AC_08 – Each team has its own tools and history	*SI_01*, *SI_05*
AC_10 – Lack of control over individual teams	*SI_01*, *SI_05*
AC_11 – Different plan for commercialization afterwards	*SI_02*, *SI_03*
AC_12 – Integration into distributed setting	*SI_05*, *SI_06*
AC_13 – Knowledge propagation is harder in iterative development	*SI_01*, *SI_05*, *SI_08*, *SI_09*
AC_14 – No integrated toolchain to support development	*SI_01*, *SI_06*, *SI_09*

supports all other components with the context information whenever necessary. The whole system should be built around this context-aware middleware.

5. Concluding Remarks

In this paper, we presented the challenges from the RESCUER crowdsourcing system. We classified the challenges according to architectural viewpoints. High-level solution concepts were also presented corresponding to each challenge. As a future activity, we will continue refining the solution concepts and applying them in the RESCUER setting and strive to obtain evidence with respect to our ideas. We hope that this will also benefit other organizations that would like to build such a system and in such a distributed setting, allowing them to prepare themselves well with respect to the challenges and solutions described here and thus greatly reduce the cost and effort for development.

This paper presented a qualitative and quantitative approach to investigate influencing factors of code smells based on a software developer point-of-view. Participants opinions point code smells and the quantitative information around them are important to improve software maintainability. Furthermore, we identified a set of factors that influence the emergence of code smells and possible strategies to mitigate them.

As future work, we intend to have a better understanding on mitigation strategies for code smells. In addition, despite of applying the research protocol in different scenarios, it needs to be applied in other different contexts to improve external validity, since we considered only systems implemented in the Java language.

References

Bass, L., Weber, I., and Zhu, L. (2014). *DevOps - A Software Architect's Perspective*. Addison-Wesley Professional, 1 edition.

Keuler, T., Knodel, J., and Naab, M. (2011). Fraunhofer aces: Architecture-centric engineering solutions. Technical Report 079.11/E, Fraunhofer IESE, Kaiserslautern, Germany.

Challenges in Crowd Communication for Emergency Management

Jorge Pereira[1], Renato Novais[1, 2], Vaninha Vieira[1, 3,] , Karina Villela[4], Manoel Mendonça[2]

[1]Centro de Projeto Fraunhofer para Engenharia de Software e Sistemas
Parque Tecnológico, Salvador – BA – Brasil
[2]Departamento de Ciência da Computação – Instituto Federal da Bahia (IFBA)
Salvador – BA – Brasil
[3]Departamento de Ciência da Computação – Universidade Federal da Bahia (UFBA)
Salvador – BA – Brasil

[4]Fraunhofer Institute for Experimental Software Engineering (IESE), Germany

{jorge.pereira,vaninha.vieira}@fpc.ufba.br,renato@ifba.edu.br,
karina.villela@iese.fraunhofer.de, manoel.mendonca@ufba.br

Abstract. *The public communication during an emergency is a key step on emergency management. The maintenance of a clear communication with the involved parties is essential for avoiding panic or misguidance. One challenge is to find out who are the targets such communication which are relevant information to them and which communication channels they can be achieved. Existing public communication solutions were not designed thinking in the variability of message according the interested, concerned only with the disclosure of a single message independent of relevance of your content for each stakeholder. However, variability in the message content is important for ensuring that each stakeholder receives relevant information for him, respecting the good principles of communication in crisis situations. This paper presents the challenges of dissemination of public communication to different targets on the crowd during an emergency on industrial areas.*

1. Introduction

Emergencies are critical situations caused by **incidents** (e.g. flooding, wildfire, earthquake) [United Nations Department of Humanitarian Affairs 1992]. The adverse and undesired consequences of an emergency give rise to a **crisis**. Crisis management [Quarantelli 1986] involves: a) *evaluating the severity* of the adverse consequences of an emergency; b) *coordinating required measures* to avoid, control, and/or mitigate those consequences; and c) *establishing a communication strategy* with and among the involved parties. The challenge is to filter contextual information regarding the incident and provide it with appropriate timing to the right people [Engelbrecht et al. 2011].

The public communication team must establish clear communication between the emergency managers and the general public, ensuring precise, reliable and real-time contextual information about the emergency. The absence of official information feeds the creation of false rumors that can may provoke panic or misguidance. To be effective, the public communication must consider some essential principles [Steven Venette 2006]: communicate repeatedly; be clear (use simple language and do not use technical terms, statistics or probabilities); communicate by different tools, media and communication channels; transmit consistent information; and only provide relevant information. Existing public communication solutions were not designed thinking in the variability of message according the interested, concerned only with the disclosure of a single message independent of relevance of your content for each stakeholder. However, variability in the message content is important for ensuring that each stakeholder (e.g. press, public authorities and civilians) receives relevant information for him, respecting the good principles of communication in crisis situations

For example, consider a fire in an industrial park provoked by a chemical product leakage; *employees* from companies in the park may want to know in what company the fire occurred and the status of the emergency; while a *politician* may also demand information about the actions performed by the operational forces, and if there were casualties; additionally, *environmental departments* may also be interested in what chemical material was released during the incident. In addition, the employees can be informed by an emergency management *mobile app*, the politician may receive a release by *email*, and the *environmental department* may see the information in the industrial park *website*.

The challenge here is to establish a good strategy for communication with different parties that should be informed, providing **only** the information they require *(to be simple)*, but **all** information they need *(to be precise)*, and considering multiple communication channels *(to be safe)*. This is a key step on crisis management. If well conducted, it can help involved people and organizations to handle the crisis situation.

This paper fits in the scope of a larger research project named **RESCUER**[1], a joint Brazil-Europe initiative, involving nine research and industry organizations in four countries (Brazil, Austria, Germany and Spain). RESCUER aims at developing an interoperable solution to support command centers (both in Brazil and in Europe) in quickly managing emergencies and crisis, based on reliable and intelligent analysis of crowdsourcing information mashed up with open data. Two application scenarios are under investigation, considering emergencies in: a) *Industrial areas*, such as chemical parks; and b) *Large-scale events*, such as the Olympic Games. This contribution focuses on emergencies and crisis in the **industrial areas** scenario.

The goals of this research are: 1) to identify the best practices on public communication during an emergency; 2) to elicit existent users and contextual information needs in an emergency; 3) to define the relevant information adaptation according to different target audiences and phases of an emergency; and 4) to analyze the channels used to communicate with the public.

The paper is organized as follows: Section 2 introduces concepts and best practices for public communication in emergencies. Section 3 discusses related works. Section 4 presents the main contribution of this paper that is raising the challenges public communication in crisis situations, targeted to different audiences in the crowd. Finally, Section 5 presents our final considerations and further work.

---

[1] www.rescuer-project.org

2. Crisis and Emergency Risk Communication (CERC)

In this study, we adopt the concepts and best practices defined in the CERC Manual [Centers for Disease Control and Prevention 2014; Veil et al. 2008], created by the CDC (Centers for Disease Control and Prevention, USA). CERC means a set of principles that aim to guide emergency managers to know what to say, when to say, how to tell and thus preserve or win the public's confidence during a crisis [Centers for Disease Control and Prevention 2014]. Its basic principle of communication is: *be first*, *be right*, and *be credible*.

Two concepts are essential to understand the principles of CERC: a) Crisis communication, and b) Risk communication. The former refers to communication activities of an organization facing a crisis and is associated with the emergency management and the effort involved in alerting and keeping the public informed about an incident [Centers for Disease Control and Prevention 2014]. *Risk Communication* [Glik 2007] is the "information exchange about health risks caused by environmental, industrial, or agricultural, processes, policies, or products among individuals, groups, and institutions". CERC arises from the application of risk communication principles in crisis communication [Reynolds and W Seeger 2005]. Despite a crisis is a random event and totally unexpected [Centers for Disease Control and Prevention 2014], it presents patterns that can help communicators to anticipate problems and give immediate response. A crisis is divided in five phases [Centers for Disease Control and Prevention 2014; Reynolds and W Seeger 2005], each demanding different actions for public communication, as follows:

- **Pre-crisis**: the goal is educational and seeks to prepare the public to know how to be positioned in an emergency, and to test the communication systems.
- **Initial**: the priority is to inform the general public and affected people about the occurrence of the incident [Reynolds and W Seeger 2005] in a quick, yet reliable, way. It aims to ensure that the public: a) has reassurance, reducing emotional turmoil; b) understands the roles of involved organizations; and c) knows where to get more information. The message should be simple, credible, accurate, consistent, and on time [Centers for Disease Control and Prevention 2014].
- **Maintenance**: seeks to ensure that the public: a) understands the risk involved in the incident and how to prevent it; b) is instructed about misunderstandings or rumors; c) receives guidance on protective actions to be followed; and d) be kept informed about actions taken to control the incident.
- **Resolution**: guarantees continuous information to the public about the incident, showing the measures taken to overdue the damages caused by the incident.
- **Evaluation**: starts after the crisis is over; includes the analysis of the performed communication, actions to improve it, and documentation of best practices.

Some principles are essential for effective communication with the public during emergencies [Centers for Disease Control and Prevention 2014; Glik 2007; Reynolds and W Seeger 2005]: Communicate repeatedly; Be clear (use simple language and do not use technical terms, statistics or probabilities); Communicate by different tools, media and communication channels (never trust on a single method of communication); Transmit consistent information; Provide only relevant information.

3. Related Works

In this section, we discuss existing solutions for public communication. The **Twitter Disaster Alerts**[2] provides, for authorities, a new way to disseminate information about an emergency. The crisis reports are disseminated via SMS and tweets, with a specific visual identity. To receive such notifications, the Twitter users must subscribe to this service. This solution is a means of message dissemination in large scale. However, it does not support generating or sending messages to specific people in specific areas.

The **Google Public Alerts**[3] is a platform to disseminate relevant emergency alerts to users when and where they are searching for. The users are warned by: a) automatic reports sent through Google Now, and b) customized search results when the user is looking for the situation of a specific emergency in the Google search engine.

The **Cell Broadcast Emergency Alerts** is a communication channel where messages can be issued to people in a specific area [One2Many 2012]. To receive reports, the user's mobile device must support the technology and the cell phone provider should make the cell broadcast infrastructure available. Unlike SMS, solutions based on cell broadcast are free of network congestion, since such messages use an exclusive channel. There are government applications being developed for emergency communication using cell broadcast. For example, in USA, the Commercial Mobile Alert System (CMAS) and the Wireless Emergency Alerts (WEA) [FCC 2014], and, in the Netherlands, the NL-Alert [Conict 2006].

Developed by the Israeli government, the **National Message** solution [Weiss 2011] aims to "establish a nationwide warning system that disseminates selective alerts and guidance messages to the population in real time, based on immediate control of all available and relevant channels in Israel". It has a module for public communication called Personal Message that allows the notification of an emergency occurrence to the population in a specific area.

Alerts4All [Alert4All 2014] is a project developed in cooperation by 12 European partners. Its goal is to create a complete communication framework for public alerts. It supports authorities during the preparation of emergency messages and provides a specific communication protocol to transmit such messages. Therefore, it is necessary that TV, cell voice message, SMS, radio, GPS Navigator and sirens manufacturers implement the communication protocol created in the project.

Table 1. Analysis of Public Communication Solutions

Solution	Communication Technology	Semi-Automatic Messages?	Reports for Specific Areas?	Group-Target Personalization?
Twitter Alerts	Mobile Application, Website and SMS	☒	☒	☒
Google Alerts	Mobile Application and Website	☒	☑	☒
Cell Broadcast	Cell Broadcast	☒	☑	☒
Israel National Message	Cell Broadcast, Pager, TV, Radio, Email, Website, ETWS and Sirens.	☑	☑	☒
Alerts4All	Cell voice message, SMS, TV, Radio, GPS Navigator and Sirens.	☑	☑	☒

[2] https://about.twitter.com/products/alerts
[3] https://suport.google.com/publicalerts/

Table 1 presents characteristics of those solutions, comparing communication technologies, if they support semi-automatic messages, if reports (messages) can be sent to specific areas (location), and if they support customized messages to different target groups. Existent solutions are not designed to create personalized messages for different target groups.

4. Challenges in crowd communication for emergency management

The goals of this work (presented in Section 1) are aligned to the challenges in crowd communication. The first one was discussed in Section 2, aiming to identify public communication challenges coming up in an emergency. To pursue the other goals: 2) to *elicit existent users and contextual information needs in an emergency;* 3) to *define the relevant information adaptation according to different target audiences, and different phases of an emergency;* and 4) to *analyze the channels used to communicate with the public*; we conducted a **workshop with real end users** to learn from their experience how public communication occurs, in practice, during an emergency. Representatives from two partner organizations in the RESCUER project participated in this workshop: **COFIC**[4] (Brazil) and **FIRESERV**[5] (Austria), both with large experience in dealing with emergency situations in industrial parks. We wanted to guarantee that our solution was valid both for emergencies in Brazil and in Europe.

The workshop was guided by the brainstorm technique with the goal to answering the following questions: i) What are the types of users presented in an emergency situation, and what users should be considered for public communication? ii) What is the relevant information to be presented for each type of user in a public communication? and iii) How the information presentation should be adapted, considering the different phases in an emergency? Section 4.1 presents our findings for Question i), while Section 4.2 presents results for Questions ii) and iii). Section 4.3 analyzes differences in communication with the crowd.

4.1. Types of Users in an Emergency

As illustrated in Figure 1, an emergency situation involves different people that are located in different places. We classify the locations between two main types: **On-site** (where the incident took place) and **Off-site**. The on-site is further divided into two regions: *on-spot area* and the *area nearby*. In the *on-spot area*, we can find various types of users, such as *Civilians* (e.g. Eyewitnesses, Affected People, Employees, Visitors), and *Operational Forces* (e.g. Medical, Fire and Police Services). In the *nearby area* we have Commanders and Staff of the *Command and Control Center*, responsible for managing the emergency. The off-site region is divided into two groups based on user type: *Operational Forces* and *Others* (e.g. Politicians, Media, General Public, Organizations). Operational Forces communicate directly to the Command and Control Center. The Command and Control Center must guarantee that public communication is sent to the users represented on green rectangles shown on Figure 1. Regarding **Question i)**, the end-users in our workshop agreed with the following types of users for public communication of emergencies in industrial parks: *Employee; Visitor; Neighbor Community; Environmental Department; Politician;* and *Press.*

---

[4] http://www.coficpolo.com.br
[5] http://www.fireserv.at

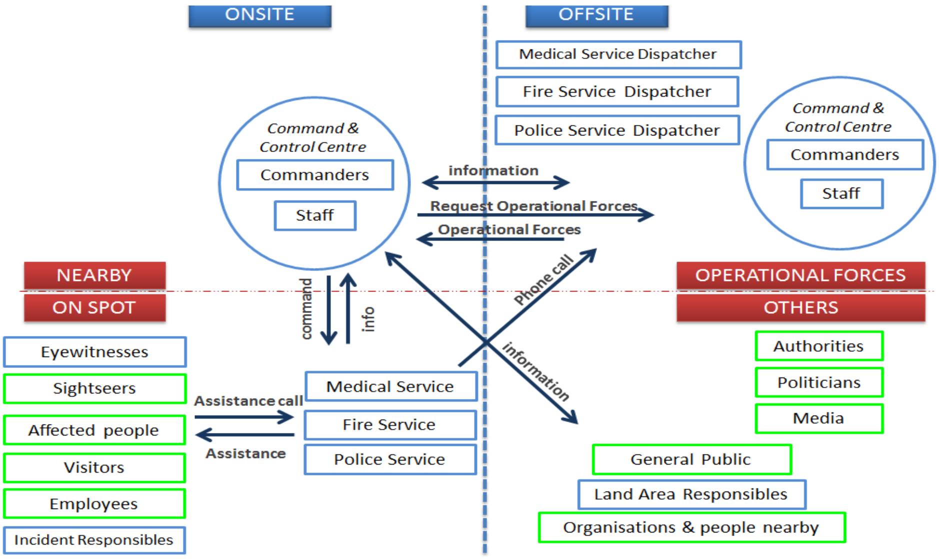


Figure 1 – Locations and Users Involved in an Emergency

4.2. Contextual Information Adaptation for Public Communication

Public communication implies transmitting information from the on-site location to people located both on-site and off-site the incident place. In our workshop with end-users, on answering **Question ii),** they agreed that the following set of information is relevant in industrial parks: *Incident Location, Incident Type, Occurrence Time, Consequences (physical, material, financial, etc.), Providences taken to control the Emergency, Status of Emergency, Injured People, Fatalities (or not), Schedule of press conferences and Type of chemical released.* Yet for **Question ii)** and also for **Question iii)**, we asked them to indicate how we should consider **information adaptation**, both to **different users** (which information is relevant for each type of user) and to **different phases** of an emergency. The end-users agreed that *Press* and *Politician* have the same set of relevant information. Similarly, *Visitor, Neighbor Community* and *Employee* share the information needs. The Table 2 is divided according to different users' groups and the information they need. For example, the Incident Location, Incident Type and Occurrence Time are relevant for all users while the *Environmental Department* is also worried about the Type of chemical released.

Table 2 - Stakeholder x Emergency Contextual Information

Emergency Contextual Information	Stakeholder		
	Visitor, Employee and Neighbor Community	Environmental Department	Press and Politician
Visitor/ Neighbor Community and Employee			
Incident Location	☑	☑	☑
Incident Type	☑	☑	☑
Occurrence Time	☑	☑	☑
Consequences (physical, material, financial etc..)	☑	☑	☑
Providences taken to control the Emergency	☑	☑	☑
Status of Emergency	☑	☑	☑
Injured People	☑		☑
Fatalities (or not)	☑		☑
Type of chemical released		☑	
Schedule of press conferences			☑

We can see that the communication of an emergency to the public varies according to the type of user. In addition, sending public information for unauthorized people is prohibited. Therefore, determining which information will be sent for each specific user is essential.

4.3. Communication Channels with the Crowd

One of the basic principles of CERC is to never trust in a single channel for disseminating communications. This is because in case of failure, crisis managers would be unable to send news to the crowd. In order to avoid this situation, there is a need to ensure that multiple communication channels are available to the crisis managers can issuing communication during an emergency.

Another point to be considered in communication with the crowd is the need to disseminate to specific areas. For example, in the case of an emergency in the industrial park, you may need to send a warning statement for a specific neighboring community but not for other communities. In this case, you need to use communication channels that allow receiving messages filtered by geographic location (e.g. Mobile App, Cell Broadcast, Google Alerts) or channels that can use knowledge about the community to send information (e.g. Voice Messages, SMS or Email)

However, in many cases it is necessary for the news disseminated to be as wide and public as possible. This is important to prevent any misunderstandings, preserve the organization's image and sustain public confidence. For these cases, the news must be published in mass media, i.e., on social networks, website of organization responsible for crisis management or sent by email to the editorial offices of the main communication media.

5. Conclusions and Future Work

Public communication is a key step on crisis management and helps involved people and organizations in handling crisis situations. A key point is to guarantee that the right people will receive the relevant information they need at the right time. This paper identified possible variabilities in the content of messages according to different users and emergency phases. We also analyze communication needs with the crowd giving examples of possible media that can meet such demands.

The result of this paper it is being used for develop a context-sensitive solution for dissemination of public communication to the crowd during an emergency. This solution was called RESCUER News and it is being developed inside of Rescuer Project. The main objective of Rescuer News is help the coordinator of public communication in the task of creation and dissemination of public statements adapted for each stakeholder. Besides, we will improve our approach to include support for crowd steering communication, when the Command Center not only notifies people but also gives them precise instructions to follow during the emergency.

Acknowledgments

This work is supported by the RESCUER project, funded by the European Commission (Grant: 614154) and by CNPq/MCTI (Grant: 490084/2013-3), and by the FPC-UFBA, a joint initiative of Fraunhofer Society and UFBA, with support from the Bahia State Government.

References

Alert4All (2014). Alert4All. http://www.alert4all.eu/, [accessed on May 20].

Centers for Disease Control and Prevention (2014). 2014 Edition B E F I R S T . B E R I G H T . B E C R E D I B L E . 2014 Edition.

Conict (2006). NL-ALERT and Cell Broadcast.

Engelbrecht, A., Borges, M. R. S. and Vivacqua, A. S. (2011). Digital tabletops for situational awareness in emergency situations. In *Proceedings of the 2011 15th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2011*.

FCC (2014). Federal Communications Commission: Wireless Emergency Alerts.

Glik, D. C. (2007). Risk communication for public health emergencies. *Annual review of public health*, v. 28, p. 33–54.

One2Many (2012). Cell Broadcast Emergency Alerts.

Quarantelli, E. L. (1986). *This of New. In *International Conference on Industrial Crisis Management*.

Reynolds, B. and W Seeger, M. (2005). Crisis and emergency risk communication as an integrative model. *Journal of health communication*, v. 10, n. 1, p. 43–55.

Steven Venette (2006). Risk and Crisis Communications. *Communication," Journal of Applied Communications Research,* p. 229–231.

United Nations Department of Humanitarian Affairs (1992). *Internationally agreed glossary of basic terms related to disaster management*.

Veil, S., Reynolds, B., Sellnow, T. L. and Seeger, M. W. (2008). CERC as a theoretical framework for research and practice. *Health promotion practice*, v. 9, n. 4 Suppl, p. 26S–34S.

Weiss, G. (2011). National Message – The Israeli Public Alert Warning System.

Uma visão sobre a adoção do Crowdsourcing para desenvolvimento de software no Brasil

Leticia Machado[1], Graziela B. Pereira[1], Rafael Prikladnicki[1], Cleidson R.B.de Souza[2], Erran Carmel[3]

[1]Faculdade de Informática – Pontifícia Universidade Católica do RGS (PUCRS)
Av. Ipiranga, 6681– 90619-900 – Porto Alegre – RS – Brazil

[2] Faculdade de Computação – Universidade Federal do Pará (UFPA)- Belém, PA – Brasil

[3]Information Technology Kogod School of Business - American University
Washington, D.C.

{leticia.smachado,grazibpereira}@gmail.com, rafaelp@pucrs.br, cleidson.desouza@acm.org, carmel@american.edu

Abstract. *Crowdsourcing means outsourcing to the crowd and can be used to support software development activities. In this paper, we present crowdsourcing, a global trend and show how Brazil is inside this context. We also introduce a preliminary study about the opportunities and challenges faced by Brazilians in the three crowdsourcing perspectives – buyers, platforms and crowd.*

Resumo. *Crowdsourcing significa a terceirização para a multidão e pode ser usado para apoiar diferentes atividades inclusive atividades de desenvolvimento de software. Neste artigo, apresentamos o crowdsourcing como uma tendência mundial e mostramos como o Brasil está posicionado neste cenário. Também introduzimos um estudo preliminar sobre as oportunidades e os desafios enfrentados na perspectiva brasileira dos três elementos do crowdsourcing – clientes, plataformas e multidão.*

1. Introdução

Crowdsourcing (ou simplesmente *CS*) é um modelo de solução de problemas distribuído. O termo "crowdsourcing" foi cunhado por Jeff Howe (Howe, 2006) quando discutiam sobre como os negócios estavam efetivamente utilizando a Internet para terceirizar o trabalho para diversos indivíduos. Em nosso estudo adotamos a definição amplamente aceita, CS é o ato de uma organização terceirizar seu trabalho para uma indefinida rede de trabalho utilizando uma chamada aberta para participação.

Este artigo apresenta os resultados iniciais de um estudo empírico sobre *CS* para o desenvolvimento de software a fim de investigar a sua adoção pelo mercado de indústrias brasileiras e os efeitos deste novo fenômeno no mercado de trabalho.

Buscando compreender como e quem está utilizando *CS* para desenvolvimento de *software* no Brasil e os principais desafios na sua adoção realizou-se uma pesquisa de campo com pessoas e empresas através de entrevistas e com a aplicação de um questionário. Os resultados preliminares estão descritos na Seção 4 deste artigo. Antes é apresentada uma breve definição de *CS* (Seção 2) e os detalhes sobre os métodos de pesquisas e amostra de dados utilizados (Seção 3). Finalizando o artigo estão as conclusões e os próximos passos do estudo (Seção 5).

2. *Crowdsourcing*

A evolução das tecnologias de mídia da *Web* 2.0 potencializou a colaboração, refletindo no fenômeno de *crowdsourcing,* no qual um grande e indefinido grupo de participantes globalmente distribuídos, pode ser alocado sob demanda, contribuindo de maneira online para solucionar problemas e resolver tarefas.

Várias áreas de conhecimento estão utilizando *CS* para obter escalabilidade, promover inovação, reduzir custos e tempo. A área de Engenharia de Software (ES) tem recentemente explorado o modelo de *CS* para tarefas de desenvolvimento de software (requisitos, *design*, codificação e testes) buscando soluções coletivas para resolução de problemas dentro do contexto da ES (Stol, & Fitzgerald, 2014). Entretanto, embora esse paradigma indique uma tendência na área de ES existem desafios e preocupações para as empresas adotarem o CS para o desenvolvimento de *software* no Brasil.

2.1 Elementos Básicos do *Crowdsourcing*

No *CS* há três atores (Figura 1). A plataforma é o intermediário da comunicação entre as outras duas partes: o cliente, que tem a demanda ou necessidade a ser resolvida e a *crowd* (multidão), comunidade dispersa globalmente que realizará o trabalho (Prikladnicki et al. 2014). Exemplos de plataformas de *crowdsourcing* incluem TopCoder[1], Amazon Mechanical Turk[2] entre outras tais como, Crowdtest[3] e WeDoLogos[4], estas duas brasileiras.

Figura 1: Modelo básico de *Crowdsourcing*

3. *Crowdsourcing* no Brasil

Este trabalho se propôs a investigar este fenômeno emergente de terceirização e como ele está sendo adotado no Brasil. Foram realizadas entrevistas, pessoalmente e por voz, com executivos de grandes empresas e com trabalhadores da área de Tecnologia da Informação (TI).

No estudo inicial, buscou-se identificar clientes que adotam *CS* para desenvolvimento de *software* no cenário brasileiro, quais plataformas são utilizadas, qual o conhecimento dos profissionais em relação ao termo *CS* e também compreender quais fatores inibem as empresas a submeterem tarefas para serem solucionadas através do *CS*.

---

[1] www.topcoder.com

[2] www.mturk.com/ mturk/welcome

[3] www.crowdtest.me

[4] www.wedologos.com.br

3.1 Participantes

Um total de 24 entrevistas foram conduzidas. Os participantes eram principalmente do Brasil com vários *backgrounds*, perfis e habilidades como: gerentes de TI, mídia e empresas de internet, executivos de plataforma de *CS* do Brasil, desenvolvedores de *software* e gerentes de projetos. Nesse conjunto de dados, há duas plataformas pioneiras de CS no Brasil – Crowdtest e WeDoLogos. Essas empresas são duas das maiores plataformas de *crowd testers* e *crowd designers* na América Latina.

Para identificar se o Brasil segue a tendência de crescimento global do *CS*, a partir da perspectiva dos gerentes de projeto, também foi realizada uma pesquisa piloto, que recebeu 363 respostas e foi submetida eletronicamente no formato de questionário.

3.2 Procedimento

Os participantes de cada entrevista foram convidados separadamente a seguir um roteiro semiestruturado de perguntas com o intuito de obter informações sobre as suas experiências com o CS no Brasil. Foram questionados diferentes aspectos sobre *CS*: (1) iniciativas de *CS*, (2) plataformas de *CS*, (3) tarefas e projetos de *CS*, (4) premiação/pagamento de *CS*, (5) impacto no negócio, e (6) futuro do *CS*. Enquanto que na pesquisa aplicada aos gerentes de projeto constavam perguntas sobre o uso, experiência, desafios e recomendações para o gerenciamento de projetos de *CS* para o desenvolvimento de *software*.

3.3 Análise

As entrevistas tiveram uma média de 30 min e 1h de duração. Também foram visitadas empresas, desenvolvedores e gerentes foram entrevistados pessoalmente, via vídeo e *email*. Algumas conversas não foram gravadas por questões de confidencialidade das empresas. Os resultados foram analisados utilizando técnicas de codificação de *grounded theory* para elicitar os desafios e oportunidades além, do agrupamento de dados oriundos do questionário.

4. Resultados Preliminares

Os resultados preliminares após as entrevistas, sobre oportunidades e desafios sob a perspectiva dos 3 atores de *CS* (cliente, plataforma e *crowd*) podem ser consultados na Tabela 1.

A análise dos resultados possibilitou concluir que o Brasil possui um número considerável de membros ativos em plataformas de *CS* nacionais e internacionais como *crowd*. A situação é diferente para as inciativas de *CS* por parte das empresas de TI ou de outros segmentos. Há ainda um baixo número de clientes brasileiros utilizando o modelo de *CS*. Por outro lado, as plataformas brasileiras de *CS* estão emergindo para diferentes domínios (criação, produção de informação em pares, teste de *software*, etc.). Através da análise quantitativa do resultado do questionário aplicado aos gerentes de projetos ficou evidente que o Brasil não está acompanhando o rápido crescimento de *CS* mundial. Os dados coletados mostraram que muitos participantes desconheciam o termo *crowdsourcing*. Apenas 35% conheciam, enquanto 65% desconheciam completamente.

Apenas 7% dos entrevistados já tiveram alguma experiência com *CS*, confirmando que o mercado brasileiro é imaturo nessa área.

Tabela 1. Perspectiva dos elementos de *Crowdsourcing*

	Oportunidades	**Desafios**
Crowd	Remuneração extra	*Feedback* fraco
	Flexibilidade de horário	Domínio de outro idioma
	Compartilhar conhecimento	Pouca colaboração
	Curiosidade	Conhecimento de tecnologias
Cliente	Escalabilidade	Conhecimento específico do negócio
	Reduzir tempo e custo	Baixa qualidade dos serviços entregues
	Criatividade	Maturidade dos fornecedores (*crowd*)
	Novos caminhos para fazer o mesmo	Decomposição das tarefas
Plataforma	Entrega rápida	Confidencialidade de dados
	Redução de Custos	Conhecimento específico do negócio
	Diversidade de soluções	Leis e taxas envolvidas

5. Conclusão e próximos passos

Nossos dados empíricos iniciais mostram que os brasileiros, especificamente, as indústrias de TI do Brasil, ainda não têm conhecimento do conceito de *CS*. Os profissionais não tem certeza de quão confiável é o modelo de *CS*. Além disso, o Brasil é um país que normalmente recebe uma grande demanda de *outsourcing*, mas, não está acostumado a solicitar serviços terceirizados (o Brasil na maioria das vezes é fornecedor e não consumidor). As empresas não entendem muito bem a força do trabalho distribuído, dentro de um sistema colaborativo online, e também, possuem forte preocupação com questões sobre propriedade intelectual e exposição de suas regras de negócios.

Como trabalhos futuros pretende-se continuar a condução de estudos qualitativos para explorar outros desafios e futuras oportunidades enfrentadas pelos três atores de *CS* no Brasil – clientes, plataformas e a multidão.

6. References

Howe, “Crowdsourcing: A definition”http://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_a.html, June 2006.

Kaganer, E., Carmel, E., Hirscheim, R., and Olsen, T. (2013), “Managing the Human Cloud”. MIT Sloan Management Review, 54(2), 23-32.

Prikladnicki, R., Machado, L., Carmel, E., and de Souza, C. R. B. (2014) “Brazil Software Crowdsourcing: A First Step in a Multi-year Study”, 1st International Workshop on CrowdSourcing in Software Engineering (CSI-SE). Collocated with the 36th International Conference on Software Engineering (ICSE), Hyderabad, India.

Stol, K. J., & Fitzgerald, B. (2014) “Two's company, three's a crowd: a case study of crowdsourcing software development”, In Proceedings of the 36th International Conference on Software Engineering (pp. 187-198). ACM.

Evaluation of a Crowdsourcing System: An experience report

Daniel Soares Santos[1], **Brauner R. N. Oliveira**[1], **Elisa Yumi Nakagawa**[1]

[1]ICMC, Department of Computer Systems
University of São Paulo, São Carlos, SP, Brazil.

{danielss,brauner}@usp.br, elisa@icmc.usp.br

Abstract. *Crisis and emergency management systems can be very useful to provide to the responsible authorities (e.g., firefighter and police) measures to efficiently act in case of an emergency situation. These systems refer to crowdsourcing systems, where information is gathered from a group of people involved in an emergency situation. In this context, several information about an incident can be provided by eyewitness through the use of mobile applications. Considering the risky and stress situation, it is important to ensure the quality of these systems. This paper presents the evaluation of the quality of crowdsourcing mobile application that takes part of a crisis and emergency management systems.*

1. Introduction

In the context of crisis and emergency management systems, crowdsourcing refers to the use of experience and information from groups of people that are attending to an event or are nearby, to enable operational forces to adequately act in case of a crisis or an emergency (e.g., fire, explosion, turmoil, and gas leak). In this paradigm, relevant information is gathered from the event area or attendees and sent to the command and control centers, allowing a more efficient response to the situation. In this context, RESCUER is an international research project[1] that addresses the development of a computer-based solution that relies on collection, combination, and aggregation of crowdsourcing information, to provide command and control centers with real-time contextual information related to an emergency situation in risky industrial areas and large-scale events. One of the main systems that compose this solution is the Mobile Crowdsourcing Solution (MCS) that implements suitable context-sensitive mechanisms for eyewitnesses and operational forces carrying mobile devices to provide the command and control centers with information about emergency situations.

In this type of systems, it is important to assure high level of quality considering their use in this critical application domain. Emergency situations trigger very basic human instinctive behavior, making people overwhelmed and confused. Therefore, it is important to deeply understand how these situations affect human behavior. A mobile device in an emergency situation offers the possibility for promptly asking for help. A quality evaluation of this type of system was reported by Ariffin et al. [Ariffin et al. 2014] considering existing mobile applications that use crowdsourcing to report crime related incidents. Considering the lack of studies reporting this kind of experience and that it is a valuable source to understand how to evaluate crowdsourcing systems, this paper aims to

[1]http://www.rescuer-project.org/

present an experience in evaluating the first version of MCS concerning five quality characteristics: (i) user interface aesthetics; (ii) learnability; (iii) usefulness; (iv) trust; and (v) effectiveness. These characteristics were picked from a quality model built for the entire solution [Santos et al. 2015] and established through the selection of quality characteristics presented by the ISO/IEC 25010 standard [ISO/IEC 2011]. Through this evaluation, it was possible to verify if the current version of the system met its quality requirements and which improvements must be implemented in the next system version.

The rest of this paper is organized as follows: Section 2 describes the MCS in more details. Section 3 presents the evaluation plan, which defines strategies and guidelines used to conduct the evaluation. Section 4 shows the main results and finally, Section 5 presents our conclusions.

2. Mobile Crowdsourcing Solution

The MCS supports communication of eyewitnesses and official first responders (police, fire fighter, etc.) with the command and control centre. Eyewitnesses and first responders will be equipped with the MCS for the following purposes [Villela et al. 2013]:

- Eyewitnesses use MCS on their mobile devices to provide multimedia information about an incident that has occurred. The goal is to benefit as much as possible from information that can be provided by mobile devices without any explicit action of their users, but taking into consideration the user's privacy; and
- First responders should first focus on rescuing victims, providing medical care, and dealing with hazards, so their profile for iteration with the command centre is very similar to the eyewitnesses' profile. They will mainly use mobile devices, such as smart phones and wearable devices, equipped with MCS to keep the command centre informed about the evolution of the situation.

Based on that, two types of information can be gathered from people carrying mobile devices at the place of an emergency situation: (i) information that can be extracted from mobile devices without user interaction with those devices, e.g., GPS position, movement speed, movement trails, and number of devices at a specific location; and (ii) information provided by users through an explicit interaction with their mobile device, e.g., videos of the incident, text message with the incident description, and photos of damages.

Figure 1 presents the user interface of MCS, which was used in the evaluation presented in this paper. In short, users start the report of an incident by notifying the command and control center. The users can select the incident type, such as explosion, fire, human crush or other (on the first screen). By pressing one of the options, the reporting process is immediately triggered and sensor based information is send to the server. After this, the user can continue the interaction and send a standard report or call back the notification, in the case of a false alarm. In a standard report (on the second screen), the user is confronted with his/her own position (automatically detected) represented in terms of a pin displayed on the map and the incident position, which can be redefined by holding and dragging the point of the incident on the map. In addition, this screen offers the possibility of specifying the severity of the incident, if there are injured persons, and possibility of taking photos/videos of the incident. By navigating in the section "Describe the fire", the user get a new screen (on the third screen) with several attributes that are relevant for the incident classification.

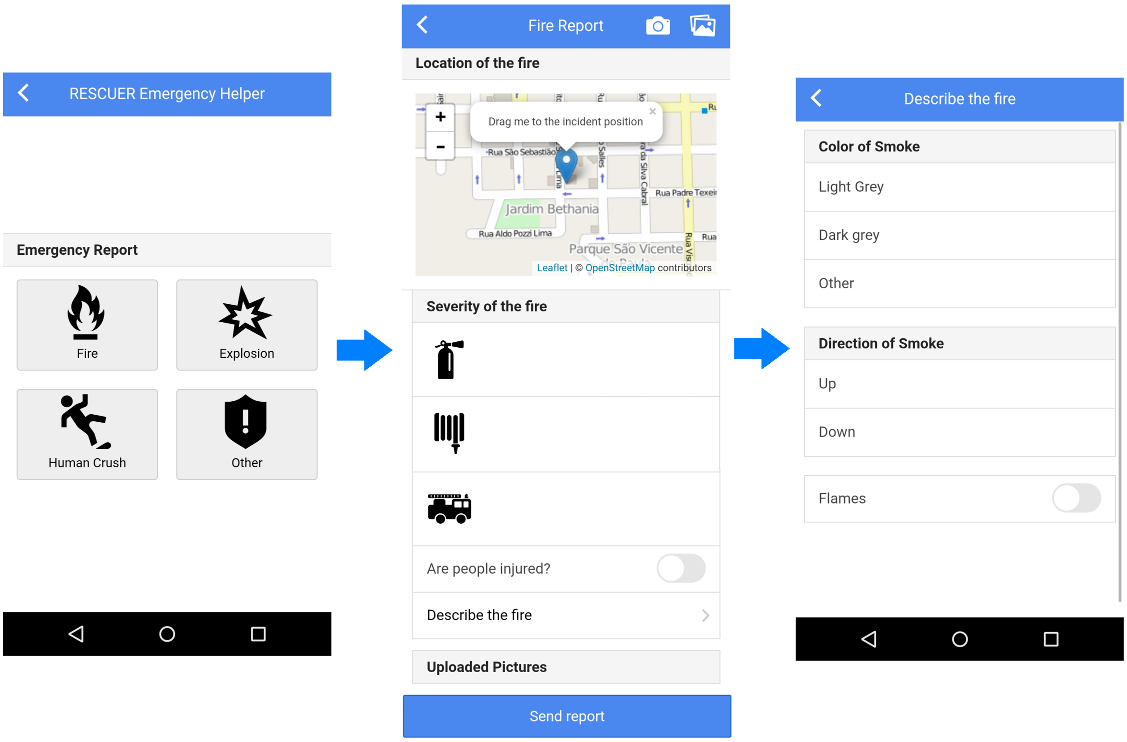


Figure 1. MCS user interface

3. Evaluation Plan

In order to evaluate the MCS, a specific evaluation plan was prepared. This plan aims to define strategies, guidelines, artifacts, scenarios, and participants that will be considered to obtain information for the quality metrics measurement. An important source for the quality evaluation was the quality model established in [Santos et al. 2015]. This quality model was based on ISO/IEC 25010 standard, which proposes a general quality model with several quality characteristics, which are defined and hierarchically organized, to support the specification of quality requirements and quality evaluation of software systems. For each one of the five characteristics, a set of metrics was defined to measure their presence on the mobile application. All of these metrics were selected and adapted from the ISO/IEC 9126-2 standard [ISO/IEC 2003] and applied to the current version of the system. A summary of the quality metrics considered in this evaluation is presented in Table 1. In short, these quality attributes were evaluated through the two complementary strategies: (i) evaluation of MCS in use; and (ii) interviewee opinion and characterization through a survey. These strategies are described in more detail.

3.1. Evaluation of the Application in Use

In this strategy, the evaluation was performed considering the use of MCS by potential eyewitnesses in the scenarios of large events and industrial parks (these scenarios will be detailed in Section 3.3). In summary, potential users were asked to perform a predefined set of tasks in the context of an emergency situation such as a fire in a stadium. In the meanwhile, an evaluation team guided the tasks execution and observed the participants behaviour. The tasks asked for participants to perform were:

1. Report that you see an incident

Table 1. Quality Metrics

Quality characteristics	Quality sub characteristics	Metric	Purpose of the metric	Method of application
Usability	**User interface aesthetics**	MUS2	How attractive is the interface to the user?	Measure the score assigned by users to defined criteria in the applied questionnaire.
	Learnability	MUS3	What proportion of user can operate successfully a function without a demonstration or tutorial?	Count the number of users that adequately operated the functions without a demonstration and compare with the total number of users in the evaluation who did not have a demonstration or tutorial.
		MUS4	What proportion of user can operate successfully a function after a demonstration or tutorial?	Count the number of users that adequately operated the functions after the demonstration and compare with the total number of users in the evaluation who had a demonstration or tutorial.
Effectiveness		QUES1	What proportion of the tasks are adequately completed by users?	Count total number of tasks adequately completed by users and compare with total number of tasks performed by users.
Satisfaction	**Usefulness**	QUUS1	How useful is the system for the user?	Measure the score assigned by users to usefulness criteria defined in the applied questionnaire.
	Trust	QUTR1	How Trust is the system to the user?	Measure the score assigned by users to trust criteria defined in the applied questionnaire

2. Inform that the incident is in the other side of yours
3. Inform that you see injured people
4. Describe the properties of the incident
5. Inform the severity of the incident
6. Take a photo of the incident

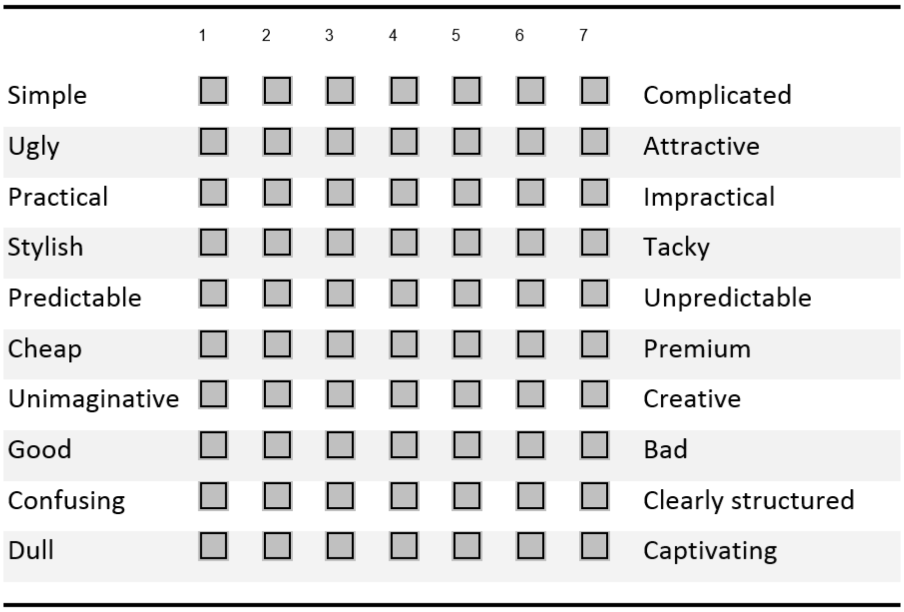
Each evaluation team was composed by two people, one moderator, and one observer, each one with the following responsibilities: Moderator was responsible for addressing participants, presenting the application and an emergency scenario, supporting participants during the test, and applying the survey in the final of the evaluation; Observer was responsible to observe if the users performed each task in an expected way filling up the observation sheet and collecting evaluation cards used in the survey. In addition, in each evaluation, one person was responsible for supporting and supervising the activities performed by evaluation teams.

For each task set, users were randomly divided into two groups: (i) users that performed the tasks without previous demonstration of MCS; and (ii) users that performed the tasks after demonstration. This division into groups was very important, as it allowed us to evaluate the learnability of MCS and to identify which aspects and characteristics of the application can influence its usability. It was expected that the difference between the results for these groups would be minimal, what would indicate that the application is usable enough for the users to complete all tasks without difficulties, doubts or questions even in an emergency situation.

3.2. Survey

After using the application, the users were asked to answer questions in a survey to provide their opinion about the usability of the application. They were also asked questions regarding the general acceptance of the application and their personal characteristics.

The questionnaire applied to obtain feedback from users regarding the usability of the application was defined based on the AttrakDiff questionnaire [Väätäjä et al. 2009]. This is an established evaluation tool that addresses evaluations of user experience and has already been used for evaluating mobile systems [Väätäjä et al. 2009]. As can be seen in the Figure 2, this questionnaire consists of pairs of contrasting attributes that may be applied to the application, such as Simple and Complicated, Ugly and Attractive, Confusing and Clearly structured, among others. In this questionnaire, squares between the attributes represent gradations between the opposites. The user can express his/her agreement with the attributes by ticking the square that most closely reflects his/her impression.



	1	2	3	4	5	6	7	
Simple	☐	☐	☐	☐	☐	☐	☐	Complicated
Ugly	☐	☐	☐	☐	☐	☐	☐	Attractive
Practical	☐	☐	☐	☐	☐	☐	☐	Impractical
Stylish	☐	☐	☐	☐	☐	☐	☐	Tacky
Predictable	☐	☐	☐	☐	☐	☐	☐	Unpredictable
Cheap	☐	☐	☐	☐	☐	☐	☐	Premium
Unimaginative	☐	☐	☐	☐	☐	☐	☐	Creative
Good	☐	☐	☐	☐	☐	☐	☐	Bad
Confusing	☐	☐	☐	☐	☐	☐	☐	Clearly structured
Dull	☐	☐	☐	☐	☐	☐	☐	Captivating

Figure 2. AttrakDiff based questionnaire

In order to identify the acceptance of the application, the following questions in the questionnaire: (i) Would you use this application to help workforces if an emergency situation like this occurs during a big event? (ii) Would you use this application to safe yourself if an emergency situation like this occurs during a big event? In addition, the users were asked to answer some personal information, such as (i) Gender; (ii) Age; (iii) If they own a smartphone; and (iv) If they have experience with emergency situation. Through this questionnaire, it was also possible to obtain specific recommendation of the users in order to become the interface more intuitive as possible.

3.3. Evaluation Scenarios

This evaluation was performed in two application scenarios: large-scale events and industrial areas/Chemical parks. Large-scale events are public events attended by a lot of people. Public events address every interested visitor who intends to attend to the activities that are offered, for example: musical performance, sports, or other social activities. Considering this scenario, the evaluation was performed in Salvador - Brazil, São Carlos - Brazil, and Kaiserslautern - Germany. In particular, the evaluation happened during the FIFA World Cup 2014, which is one of the biggest sport events of the world and was used as main scenario of MCS evaluation.

Chemical parks are industrial areas in which, among other things, chemicals are stored and processed. In case of an incident in these industrial areas, chemicals can harm employees, civilians in the affected community, and the environment. In Brazil, chemical companies are usually placed in industrial parks. These industrial parks have several companies that must be according to the Brazilian laws for emergency management. Considering this scenario, the evaluation was performed in Camaçari Industrial Complex, which is the largest integrated industrial complex in the Southern Hemisphere. It is comprised of over 90 chemical and petrochemical companies, in addition to other production facilities, such as cellulose, copper metallurgy, textiles, automobiles, beverages, and services.

4. Results

In the large-scale event scenario, 50 evaluations were conducted in Kaiserslautern-Germany, 35 in Salvador-Brazil, and 27 in São Carlos-Brazil. In total, 112 users participated of the evaluations, 55 without demonstration and 57 with demonstration of MCS. In the industrial park, a total of 60 evaluations were conducted in Camaçari-Brazil, 28 without demonstration and 32 with demonstration.

The results were grouped according to the tasks performed by each participant with and without demonstration of the application. For each task presented in the Section 3, the number of users that performed it successfully was measured. A successfully performed task means that no question was asked and the participant behavior was as expected. In addition, the usability of the application can be derived from the slight difference between the results of those who received and did not receive a demonstration. This means that, as lower the influence of the demonstration in the comprehension of the application and effectiveness of the users, the level of usability and learnability of the application will be greater. Table 2 shows a summary of the successfully performed task regarding the kind of demonstration and the evaluation scenario. As it can be observed, the tasks "2. Inform that the incident is in the other side of yours", "3. Inform that you see injured people", and "6. Take a photo of the incident" had a lower level of effectiveness. Beside this, the difference of the effectiveness regarding the users that received and not received demonstration was very significant. This allow us to identify key points of the application that impacted the usability of the application in these tasks.

Table 2. Tasks Successfully Performed Percentage

Tasks	Large scale events		Industrial Park	
	With Demo	Without Demo	With Demo	Without Demo
1. Report that you see an incident	95%	80%	88%	82%
2. Inform that the incident is in the other side of yours	59%	38%	86%	61%
3. Inform that you see injured people	80%	67%	80%	76%
4. Describe the properties of the incident	82%	72%	80%	60%
5. Inform the severity of the incident	81%	58%	91%	89%
6. Take a photo of the incident	62%	49%	72%	61%

The feedback of the participants regarding the usability of the application, as early mentioned, was captured through the use of an AttrakDiff questionnaire. In general, the feedback of the participants was positive. On average, the score obtained was 6 of a maximum of 7. In addition, the application was well received by the general public. More than 85% of participants said they would use MCS to save themselves, which indicates a clear acceptance. When asked if they would use the application to help operational forces, more than 90% responded positively. With these results, it is possible to conclude that most people would probably use this application in an emergency situation.

Using the information obtained from the evaluation of MCS (in use and through the survey), the set of six metrics, presented in Table 1, defined for the measurement of three quality attributes was calculated. After this, the results were compared to the assessment criteria established in the requirement document of MCS. These assessment criteria were used to decide whether the metric results were satisfactory or not, considering the expected results. In general, these criteria are numerical thresholds or targets used to determine the need for action or further investigation.

Table 3 presents the metric results regarding the quality attributes and the scenario where the evaluation was performed (i.e., large scale events or industrial park). In order to better understanding how the metrics results were obtained during the evaluations, some clarifications are provided.

Table 3. Metric Results

Quality Characteristic	Quality Sub-Characteristic	Metric	Assessment criteria	large-Scale Events	Industrial Park	Total Measure	Total Result
Usability	User Interface Aesthetics	MUS2	0.7	0.84	0.87	0.86	yes
	Learnability	MUS3	0.6	0.57	0.70	0.64	yes
		MUS4	0.7	0.73	0.80	0.77	yes
Effectiveness		QUES1	0.55	0.66	0.76	0.71	yes
Satisfaction	Usefulness,	QUUS1	0.6	0.94	0.97	0.96	yes
	Trust	QUTR1	0.6	0.87	0.93	0.90	yes

The metric "MUS2" was calculated from the participants' feedback about the usability of the application. In particular, the average score for all aspects defined in the AttrakDiff questionnaire was used as input for the measurement of this metric. In this sense, 0.7 is the assessment criteria defined for this metrics; the values 0.84 and 0.87 are the average score obtained in the AttrakDiff questionnaire for large-scale events and industrial parks context respectively; 0.86 is the average of the results obtained in both contexts; and finally, column "Total Results" shows that the total measure was sufficient when compared to the assessment criteria. The metrics "MUS3" and "MUS4" were calculated from the number of all tasks performed by the participants with and without demonstration of MCS, respectively. The metric "QUES1" was calculated from the number of tasks successfully performed by the participants. The metrics "QUUS1" and "QUTR1" regarding the user satisfaction characteristic were measured considering the answers obtained in the application acceptance evaluation. Specifically, answers to the question "Would you use this application to safe yourself?" were used to measure the degree to which a user has confidence in the application, whereas answers to the question "Would you use this application to help operational forces?" were used to measure the degree to usefulness of the application perceived by the users. In short, as it can be observed, all the results were satisfactory considering our expectations for now. This means that, taking into consideration the average evaluation results, the metric values are higher than the values of the assessment criteria.

5. Conclusion

In the context of crisis and emergency management systems, the quality of crowdsourcing applications need to be ensured in order to allow their utilization in an emergence situation, where the users can be overwhelmed and confused. In this sense, this paper presented an evaluation experience of a mobile crowdsourcing system addressed to crises and emergencies management. As main result, the evaluation results were quite satisfactory considering our expectations for now. Despite this, it was observed the existence of room for improvement in order to achieve a higher quality for the next evaluation iteration and to make the application as intuitive as possible. In addition, we intend to perform two more evaluations concerning additional quality attributes such as performance, portability and functional suitability, besides to consider increased assessment criteria for each metric. All the improvement points identified will be carefully considered and analyzed in order to develop a better version of the system. To finish, we believe that the evaluation planning and our experience presented in this paper can be reproduced in similar application contexts.

Acknowledgments

RESCUER: Reliable and Smart Crowdsourcing Solution for Emergency and Crisis Management. This work is supported by Brazilian funding agency CNPq (Grant: 490084/2013-3) and European Commission (Grant: 614154).

References

Ariffin, I., Solemon, B., and Wan Abu Bakar, W. (2014). An evaluative study on mobile crowdsourcing applications for crime watch. In *International Conference on Information Technology and Multimedia (ICIMU)*, pages 335–340.

ISO/IEC (2003). ISO/IEC 9126 - Software engineering - Product quality - Part 2: External metrics.

ISO/IEC (2011). ISO/IEC 25010 - Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models.

Santos, D., Oliveira, B., Duran, A., and Nakagawa, E. (2015). Reporting an experience on the establishment of a quality model for systems-of-systems. In *27th International Conference on Software Engineering and Knowledge Engineering (SEKE)*, pages 1–6.

Väätäjä, H., Koponen, T., and Roto, V. (2009). Developing practical tools for user experience evaluation: A case from mobile news journalism. ECCE '09, pages 23:1–23:8, VTT, Finland, Finland. VTT Technical Research Centre of Finland.

Villela, K., Vieira, V., Mendonça, M., Franke, T., Torres, J., and Graffy, S. (2013). RESCUER-Dow - Reliable and Smart Crowdsourcing Solution for Emergency and Crisis Management.