A Real-Time, Distributed and Context-Aware System for Managing Solidarity Campaigns

Ana Alves[a,b], Tiago Dias[a] and David Silva[a,1]

[a] Polytechnic Institute of Coimbra, Portugal

[b] Centre of Informatics and Systems, University of Coimbra, Portugal

aalves@isec.pt, a21160244@alunos.isec.pt a21170222@alunos.isec.pt

KEYWORD	ABSTRACT
Crowd sensing; distributed systems; system integration; solidarity campaings	*We present a project implemented on the field which has two separate strands, one refers on collecting crowd sensing data through mobile apps where context is (near) automatically induced, and another is related to a practical application of this method in a real time system to manage solidarity campaigns in collecting goods. Here, we cover both parts, we applied an experimental setup and obtained results and insights in a third sector institution, Caritas Diocesana of Coimbra[2], a non-profit organization part of Caritas[3]. As main contribution, we propose a distributed architecture for Mobile Crowd Sensing able not only to allow real time inventory through simultaneous campaigns but also to give feedback to volunteers in order to instantly acquire information about which categories of goods are more needed.*

1. Introduction

Crowd Sensing, also known as community sensing (Rheingold, 2002), participatory sensing (Ye *et al.*, 2011) or opportunistic sensing (Estrin *et al.*, 2006), ranges from the active involvement of users to contribute with data (e.g. when taking a picture, report the closing of a road, reporting an accident) until an autonomous data collection using sensors and with a minimum interaction (e.g. continuous collection of location data without explicit user action). Crowd sensing is sometimes classified depending by which way data is collected: from mobile devices (Mobile Crowd Sensing) or Web (Web Crowd Sensing). The typical scenario for Mobile Crowd Sensing is based on simple data collection from sensors included on mobile devices. Sometimes these data are locally processed to get useful information being sent to a service/server for aggregation and analysis.

---

[1] Present mail: david_nemesis@msn.com

[2] http://www.caritas.pt/site/nacional/ Portuguese Website (last visited in March 2016)

[3] http://www.caritas.eu/ (last visited in March 2016)

Web Crowd Sensing is voluntary data collection when proprietary data acquisition becomes extremely expensive (e.g., points-of-interest introduced into Foursquare[4], bar codes and descriptions of products being submitted to populate a universal database[5]).

Due to the social and economic context in the World, several national solidarity campaigns have emerged (Riches *et al.*, 2014), and for the first time a global campaign[6] was organized in recent years to collect food and other goods (books, toiletries, school supplies, etc.). These campaigns may last for a short period of time and can be promoted in the media supported by large commercial surfaces (e.g. Food Bank campaigns[7]) or may have a more permanent basis in gathering goods through Web sites and fixed collection points across a country (e.g. the Furniture Reuse Network[8]).

Despite these campaigns are extremely helpful, they raise two problems:

- Discrepancy between need and supply - Although all help is welcome, there are more priority goods. These priorities may change based on prior campaigns, time of year, natural phenomena (e.g.: floods, fires, earthquakes) or even as a function of the current campaign since there are themes of a particular type (e.g. feeding, toiletry, school supplies) but not restrict to a specific good. In the latter case, it is advantageous for the volunteer (for both the giver as to who is collecting) know in real time which amount of specific categories of goods have been donated so far (e.g. tons of rice, boxes of diapers, thousands of pencils).
- Geographic heterogeneity - Having people donating goods around a country, and taking into account the population dispersion, likely this donation is not homogeneous. Even in the regional level there is a great discrepancy in terms of quantity and type of donations, something if it could be possible to estimate, would allow better optimization of human and logistical resources

Having been successfully used to collect mobility data and services for urban planning (Liang *et al.*, 2009; Rodrigues *et al.*, 2013), this work aims to apply the concept of Mobile Crowd Sensing in real time to solidarity campaigns, a new field for a proof-of-concept. In addition to its Software Development side, this project will contribute to the inventory of goods, analysis of the data collected and innovates due the inherent social objective. Concretely, we proposed and implemented an easily adaptable platform that incorporates the concept of Mobile Crowd Sensing. The purpose behind this platform is to ease and centralize the inventory's process, in real time and easy to use, in order to encourage people to contribute in balanced portions, with what is really needed in a higher priority. This led to a collection of applications optimized in accordance with the objectives established by the institution Caritas Diocesana of Coimbra,

---

[4] http://www.foursquare.com (last visited in March 2016)

[5] http://upcdatabase.org/ (last visited in March 2016)

[6] News about worldwide campaigns (http://food.caritas.org/pope-urges-catholics-to-join-caritas-week-of-action-to-end-hunger/) (last visited in March 2016)

[7] Companies who support Food Bank campaigns in Europe (http://www.eurofoodbank.eu/get-involved/companies-foundations) (last visited in March 2016)

[8] http://www.frn.org.uk/ (last visited in March 2016)

the host of this project. To achieve this goal, these applications use the latest technologies such as Android, Web Services and non-relational databases.

Leveraging urban technology for social purposes, such as volunteer participation and community engagement, has been steadily gaining interest in the scientific community and in the third sector. Essentially, citizens can be empowered to participate and engage the city better through the use of modern technology. Examples of these technologies are mobile phones, public displays, sensor networks, art installations, or any other type of urban technology. A national social study was made in USA (Chen *et al.*, 2013) where results show that the diversity of mobile phone use and the frequency of relational mobile communication are positively associated with mobile donation. Internet donation and mobile donation complement rather than compete with each other.

Mobile Crowd Sensing until now, as far we know, was never used in this context to help a Solidarity Institution in the real time sensing of the adherence in campaigns for collecting goods. Besides the fact we are implementing this pilot in a regional coverage, we believe this project can be adopted by others NGO in a broader sense.

Caritas is an Institution of Social Solidarity supporting transversely communities in social, health, education and pastoral care. Implemented since the 50s, Caritas has always tried to follow and respond to the problems of the communities, using a methodology that favors dialogue, cooperation and networking. Currently it focuses its intervention in the search for innovative and economically sustainable strategies, which enable the provision of answers with quality, suitable to emerging needs while maintaining focus as humanism, professionalism and technical and scientific rigor. With an increasing technology adoption, Caritas has been involved in the development of some European projects (CITEK, 2014), being an enthusiastic collaborator in the proposed architecture. Thus, it will be possible to integrate in a near future the Mobile Crowd Sensing platform with the intranet which actually exists in that institution.

Context is defined by different aspects that can be sensed in a given situation: identity, activity, time and location (Ryan *et al.*, 1999). These context types not only answer the questions of who, what, when, and where, but also act as indexes into other sources of contextual information. The goal of context-aware computing is to use context as an implicit cue to enrich the impoverished interaction from humans to computers, making it easier to interact with a computational system.

The remainder of this paper comprises 3 sections. Section 2 is devoted to the proposed architecture with each module described in detail. Section 3 presents the online implemented system and some scenarios of use. And finally, section 4 is dedicated to some conclusions and future work.

2. Distributed Architecture

One way to address the problem of the growing number of users and the need to provide more and more information efficiently passes precisely by how it is processed and made available. Exactly on these subjects, high performance and high availability, we built our platform, being a requirement to serve a high number of users and great amount of information. Figure 1 presents two possible configurations of our system depending on the availability of resources: a minimal and a maximal topology. In the latter, geographically distant servers were tested and the database was also maintained in a distributed and synchronized parallelism.

This project was developed following an agile methodology, with cyclical definition phase of requirements, architecture design, development, testing and presentation of release's result, as a way to get feed-

back and allow a more efficient development of the project. Its main components are: a mobile application with the ability to read EAN bar codes; a very responsive and high available Web service which serves as interface to a non-relational Database Management System (DBMS), and finally a back office website for administration purposes.
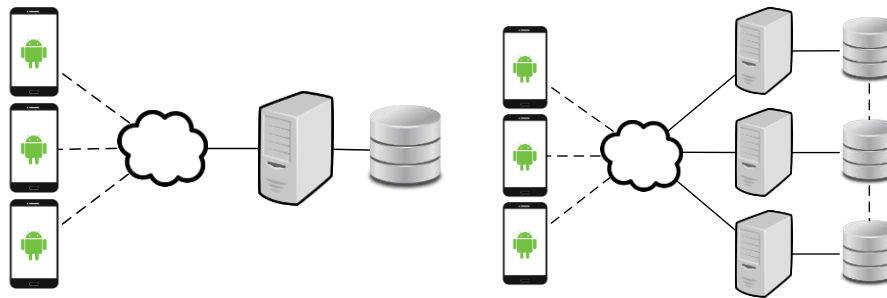

Fig. 1. Minimal e maximal topology of the proposed architecture

2.1. Web Service

High performance frameworks use a different strategy from common thread-based or worker-process-based services, taking advantage of an event-driven architecture and a non-blocking API, usually coupled with the reactive software pattern. In practice this means to have a single thread that is cyclically waiting for new events, these may be a new application or the appointment of new requested data, thus reducing the effect of the wait cycles due to hardware access delays, reducing memory use and sharing best processor time for each client. Also, this strategy allows a big increase of the number of requests accepted without congestion or depletion of resources in process data access.

In order to choose a framework for this project, priority was given to frameworks prepared for high availability, reliability and performance, preferably linked to successful cases and recommended by the respective communities for similar projects.

In this line, based on the results of tests performed over various frameworks (ribs29, Vert.x10, Netty11, Tomcat12, EventMachine13, Node.js14), it was decided to opt for Vert.x which presents good re-

---

[9] https://github.com/Adaptv/ribs2 (last visited in March 2016)

[10] http://vertx.io/ (last visited in March 2016)

[11] http://netty.io/ (last visited in March 2016)

[12] http://tomcat.apache.org/ (last visited in March 2016)

[13] https://github.com/eventmachine/eventmachine (last visited in March 2016)

[14] http://nodejs.org (last visited in March 2016)

sponse time and a very low error rate, and additionally, it allows both server and client side being developed in the same language (Java).

The chosen solution was to use the Vert.x library, which allows creating some Verticles that, together, implement a REST server. Figure 2 shows a schematic view of this structure and the dataflows expected between the Internet, the receiver Verticle, all the specialized Verticles and the Database Module. The received requests start by being redirected by the RestServer Verticle to a specialized Handler Verticle, according the request. All this communication between Verticles occurs on the EventBus, a very fast and intuitive communication mechanism, build-in on Vert.x, which implement the Publisher-Subscriber pattern. These Verticles use this same EventBus to communicate with each other to perform extra functions (e.g. see if the device you are requesting statistics of a campaign is blocked) or simply send or request data to the module responsible to communicate with the database server (a MongoDB15 DBMS as will be discussed later).
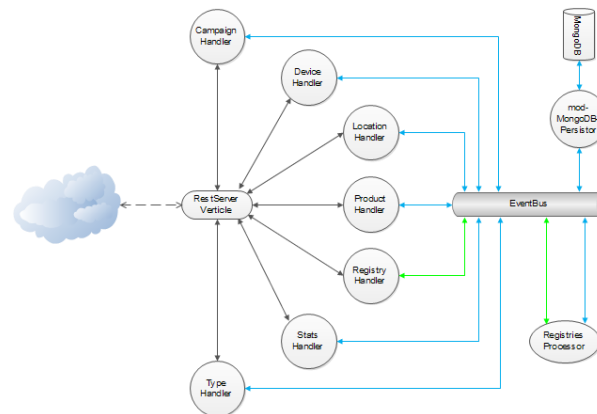
Fig. 2. Service structure

Another key point of this technology is its ability to be distributed, either by multiple processors, machines or even in different geographic locations, allowing having multiple servers around the level of a Wide Area Network (WAN) as a way of redundancy and more closeness to the points where there is greater flow of traffic, increasing overall throughput. Thus mobile devices can only communicate with a server/service, through a communication protocol. Since the client applications will work on mobile devices, it is expectable that the connection to the server can occasionally be completely absent or occasionally with a high degree of instability. It is also worth noting that this same connection can be chargeable. It is therefore important to choose simple protocols which do not require constant connection, having low overhead and a minimum of reliability. In addition it is also important that there are already tools and libraries available to support these communication protocols.

We therefore considered various client-server communication solutions, typically used in Web Services, since RPC (Remote Procedure Call) to more generic protocols (JSON-RPC, REST, CORBA, WebSockets, Eventsource, XML-RPC, SOAP). Through data obtained in tests and the knowledge gained

---

15 http://www.mongodb.org (last visited in March 2016)

from past projects, the REST protocol with a JSON structure was chosen in order to optimize as much as possible peering, facilitate debugging operations and increase the range of libraries for handling messages.

2.2. Crowd Sensing Database Management System

In addition to framework's choice, the method to store data is also prepared to support large volumes of data in a distributed manner using specific DBMS for each end in the architecture. Importantly already mature relational databases and especially the new NoSQL databases that have been a hotly debated issue, both replication capacity, distributed processing and (case of NoSQL) supporting an unfixed data structure.

The Android framework itself already has support for SQLite, which has proven to have excellent performance in low concurrency and in very low resource devices, these were the main reasons to automatically opt for this technology on the client.

Contrary to the client part, the service/server needs a database that supports plenty of concurrent accesses and that can effectively take advantage of available resources. Since there are various options, it was decided to refine the search with greater relevance to free databases, free use and good performance and reliability in parallel with the chosen framework. We decided also to investigate the use of NoSQL databases, which has lately been the subject of much discussion because of its proven more efficient ways to store information and better redundancy to prevent failure.

Centered on the concept of storing documents, Document model is designed precisely to save encapsulated data according to a standard, usually XML, JSON, BSON and YAML. Each document is assigned a unique ID throughout the database and sometimes some extra processing can be made, such as sorting and aggregations.

Within NoSQL databases (MongoDB and Cassandra) it has been found advantageous to opt for MongoDB, mainly due to the flexibility of data, using JSON interface (such as the framework chosen), easy to use and ideal for applications prototyping process. On the other side, we excluded Cassandra due to its higher complexity and because it is a DBMS optimized for a much bigger data volume and infrastructure.

2.3. Context-Aware Mobile Application

The client-side of this platform (Figure 3) allows a volunteer to register its device in a campaign already created on the server side. After that, the device is ready to submit new readings of donated goods by its barcode. In this way, no time or manual introduction is required, timestamp and geographical position are locally recorded. As soon Wi-Fi or 3G access is detected (or after a predefined time), the application transmits these records to the service, populating the server database. As output, any volunteer who is participating in the same campaign can also see which amount of goods was donated so far. This is completely different from the old scenario where Caritas, only in the end of a week could account, and with no guarantees of accuracy, the success of a given campaign.

Specifically in the retail area, there is a known regulator GS1[16], in order to define the type of barcode for each application and reserve blocks of barcodes in order to prevent collision of codes of different products. Although there are no technological restrictions, throughout this project the barcode EAN-13 format has been the focus of study, since it is the most commonly used and easiest to test in Portugal. This standard is public and present in all products sold in national retail outlets, identifying a type of product (not working as a serial number mechanism). Thus, as we are using a global format, which is public, it is clear that there will be a set of currently available tools that allow the reading and recognize these codes.
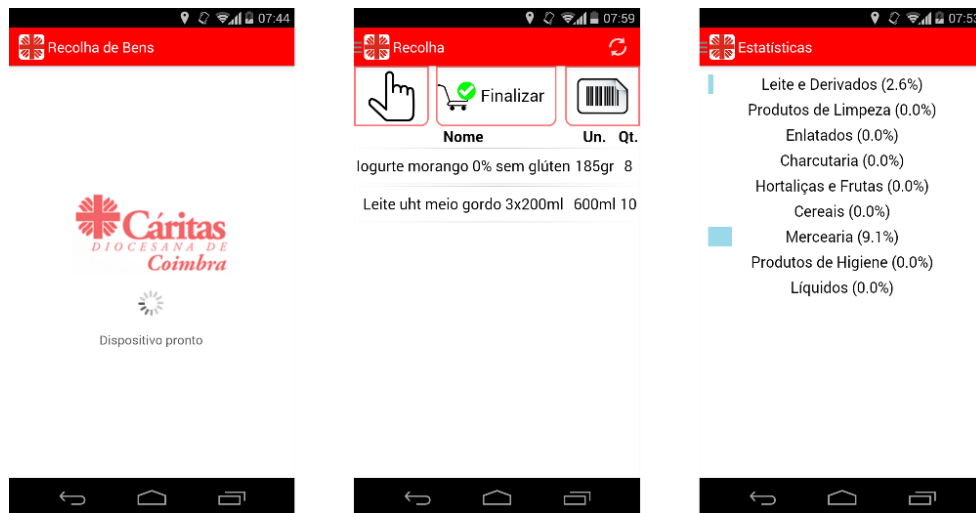
Fig. 3. Screenshots of the mobile application

After simple tests over two libraries (ZXing17 and Scandit Barcode18), small prototypes were built to validate and evaluate the process and the ease of integration of those libraries in a real mobile application. Since the early research stage, the ZXing library proved to be the best accepted by most Android developers and since it is an entirely open source library and supports a wide range of 1D and 2D codes. The majority of Android applications require this library to read barcodes. The Scandit Barcode Scanner library is not open-source and has a much narrower range of supported codes (especially in the free version), however it provided better reading speed on less favorable luminosities and lower resolution cameras. Also there was the possibility of reading an EAN 13 code even with large inclinations of the camera and poorly focused image. With this, we managed to catalog a big amount of products, previously shorted by category and weight (reading one product by category + weight), adjust quantities and submit the data in just 8 minutes, a time much shorter than before.

---

[16] http://www.gs1.org/ (last visited in March 2016)

[17] https://play.google.com/store/apps/details?id=com.google.zxing.client.android (last visited in March 2016)

[18] https://play.google.com/store/apps/details?id=com.scandit.demoapp (last visited in March 2016)

Beyond the mobile application performance, the user interface was also an issue to improve through feedback of volunteers. An experimental set of tests was performed during the last Caritas Diocesana of Coimbra's Christmas' campaign which donors were invited to give food items in order to compose Christmas baskets for the most deprived people. In this context some adjustments were performed. On the last release (Alves *et al.*, 2015), we presented a system capable of register big quantities of donated products (mostly edible ones), using an external application to read bar codes or by manual introduction from a category/weight screen, where all the data are stored on a network of servers, developed with reactive frameworks and NoSQL databases. It proved to be usable, but all the information could only be accessed via mobile app.

In a new release everything has a bigger focus on Caritas Diocesana of Coimbra needs, so many changes occurred, which the most relevant are probably:

The web service and database system has been migrated to PHP (with Epiphany framework) and MySQL, respectively. This could reduce the system throughput, but it is cheaper to implement and integrate with the services already in use by the institution;

Changes on the REST API to allow a faster startup and synchronization;

The data is stored in just one powerful machine. This could be changed again to a distributed server network if the usage level worth it;

The concept of product category and campaign changed. Visually it has no notable effects, but it will allow other types of campaigns (e. g. scholar and monetary campaigns), each one with a completely different list of categories and in the meantime, it will reduce the storage used by the mobile application;

Statistics are much more intuitive, with better filters, more informative graphs and an export functionality even easier to use if an administrator intent to do external offline statistics.

With so many improvements, a future release with these new functionalities is intended to be released on the Google Play Store.

2.4. Back office Administration

Complementing the mobile application, and by focusing on the process control, in this section we present a Back office tool with a web interface[19] to assist the institution in the management processing of collecting goods, campaigns, users and give instant access to results. This contributes to greater efficiency in the response to deprived people, as well as greater transparency and speed in the dissemination of results through employees, volunteers and donors.

The main requirements that the Back office must accomplish were:
- Support the creation to campaigns of collecting goods and the association of volunteers that complements the mobile application and the web service;
- Facilitate and make faster the access to the results of the campaigns, also allowing for greater transparency and speed in the dissemination of who cooperates (volunteers) and who contributes (donors);

---

[19] http://projectos.isec.pt/deis/mis/a21170222/backoffice/ (last visited in March 2016)

- Distinct entities (Caritas of different cities in Portugal) can access the same system administered by Caritas Diocesana of Coimbra;
- Reduce costs with the process of registration and management of volunteer teams, which currently is still done mostly on paper.

With these needs in mind a web site was developed where Campaigns, users and statistics data can be administrated through a web administration panel, which uses the most recent HTML5 oriented technologies (CodeIgniter, jQuery and Bootstrap), as shown in Figure 4. In order to normalize the authentication process, both web interface and mobile application use the same login system. This ensures that only validated and authorized volunteers can use system functionalities or access nonpublic information.
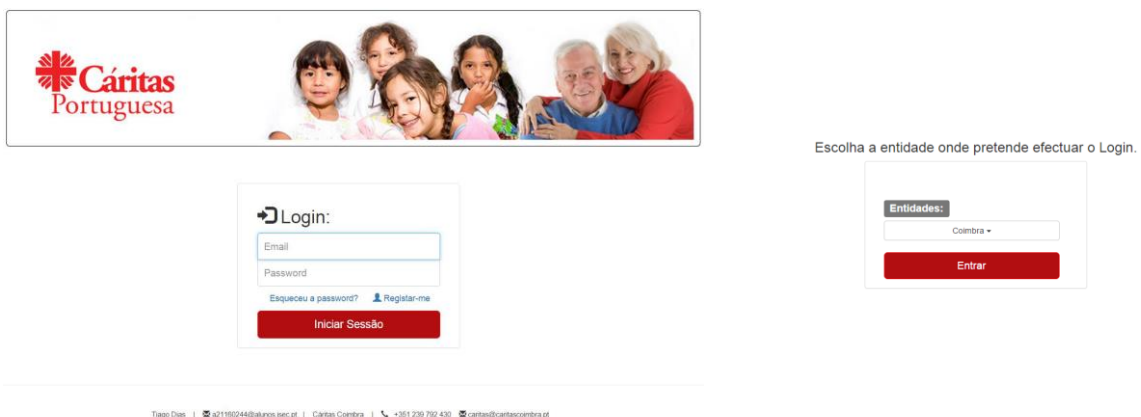
Fig. 4. Back office login page and the identification of the entity where this login must take place.

Different user profiles (Global Administrator, Entity Administrator, Employee, Volunteer) are associated to a given user depending on the entity he is authenticating on (Figure 5 presents the access of a volunteer in the web site). Thus, the same user can be employed in Caritas Diocesana of Coimbra, and a volunteer in Caritas Diocesana do Porto[20]. The user hierarchy is supported in the database and depending on the user profile when logged in, different permissions and functionalities are presented by the web site, these usage scenario are fully described on the next section.

---

[20] http://www.caritas.pt/porto/ (last visited in March 2016)

Fig. 5. Back office home page after successful login.

3. Usage Scenarios

This section presents a typical scenario from the initial step – creation of a new solidarity campaign; until the real use where goods are collected both directly by registered volunteers in a supermarket front-door using the mobile application, or when donations are sent to the headquarters of Caritas Diocesana of Coimbra and recorded in the Back office website. This project is being implemented in cooperation with Caritas as a proof-of-concept and its applicability to a larger extent beyond region of Coimbra.,

Figure 6 illustrates the access of a privileged user (Application administrator or Entity administrator) creating a new campaign. The attributes needed indicate: campaign name, period, type (e.g. food, school supplies), responsible (employee that manages this campaign), description and objectives (which volume of donations are expected for each category on that type of campaign).

Fig. 6. New campaign scenario created by the Application administrator where some attributes are required.

Fig. 7. Campaigns that are presently created in the Back office website.

Different campaigns may simultaneously be active in the system and only authorized/registered volunteer can participate in each one, as presented in Figure 7. The Application administrator or the campaign administrator is allowed to associate users to participate as volunteers in a given campaign. All users registered in Back office and posteriorly approved by the Entity administrator are eligible to be associated in a campaign. Figure 8 shows this process of association of a new volunteer in the new campaign created before. From now, when a campaign begins, each associated user is allowed to register new donations in a given campaign through 2 different ways: using the mobile application (Figure 8) or the Back office website (Figure 9). In this last option (Back office), a volunteer could also register products through the web browser using a serial or Bluetooth bar code reader to identify products even faster.

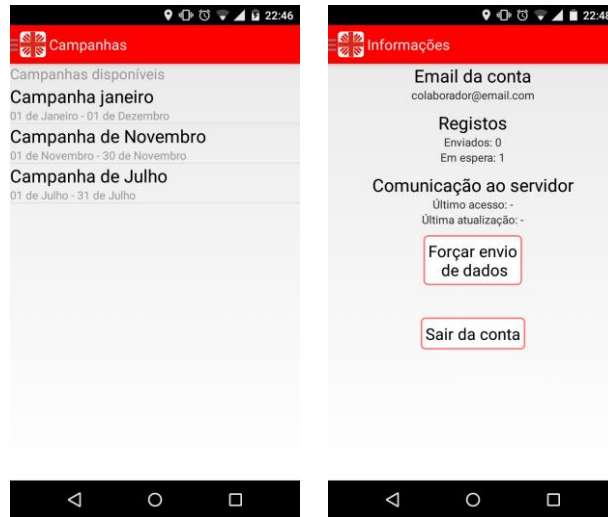**Fig. 7.** Associating a volunteer to be able to register products in a given solidarity campaign.

Fig. 8. Accessing the distributed system using the mobile application. Only campaigns where this volunteer was associated to are visible.

Fig. 9. . Registering products using the Back office website. This can be done in two ways: describing in general the product received (above) and its amount; or introducing its barcode (below).

4. Conclusions and Future Work

In this paper we presented a Mobile Crowd Sensing platform in the context of solidarity campaigns. As active contribution from volunteers the platform receives in real time, recordings related to donations of goods that are in their majority identified by EAN barcodes. Thus, the mobile application is easy to operate and it allows seeing anytime the total amount received in a given campaign. The architecture proposed favors high availability and concurrency, managing different campaigns even geographically disperses. This project has just been set up in Caritas Diocesana of Coimbra, a Solidarity Institution and data are going to be collected through different campaigns. Nevertheless, this platform is of free use to other NGOs and is publicly available for open use[21]. We hope with this application being used by Institutions and final users we can collect intensive spatial and temporal data in order to recognize patterns in the giving process. We also plan to extend the platform and the mobile application to cover monetary donations since Caritas promote these national campaigns on the streets. This new mode of engagement offers opportunities to NGOs for reaching new donors under new circumstances as messages spread virally through friend networks.

5. Acknowledgements

A special acknowledgement is dedicated to Caritas Diocesana of Coimbra, namely to its President for all collaboration in the development of this work, Priest Luís Costa.

6. References

Alves, A., and Silva, D., 2015. Mobile CrowdSensing for Solidarity Campaings, in 6th International Symposium on Ambient Intelligence (ISAmI 2015), 125-133. doi: 10.1007/978-3-319-19695-4_13.

Chen, W., and Givens, T., 2013. Mobile donation in America. Mobile Media & Communication, 1(2), 196-212. doi: 10.1177/2050157913476028.

CITEK, 2014. Lyon, France. [Online]. http://www.yourinnovationday.eu/wp-content/uploads/2014/10/Pr%C3%A9sentation-CARITAS.pdf (last visited in March 2016).

Estrin, D., Hansen, M., Parker, A., Ramanathan, N., Reddy, S., and Burke, M., 2006. Participatory sensing, in Workshop on World- Sensor-Web, ACM SenSys.

---

[21] https://play.google.com/store/apps/details?id=com.scandit.demoapp (last visited in March 2016)

Liang, A., Biderman, A., Ratti, C., Pereira, F., Oliveirinha, J., Gerber, A., and Vaccari, A., 2009. A holistic framework for the study of urban traces and the profiling of urban processes and dynamics, in 12th Intl. IEEE Conference on Intelligent Transportation Systems, 2009.

Rheingold, H., 2002. Smart Mobs: The Next Social Revolution. New York: Basic Books.

Riches, T., and Graham, T., 2014. First World Hunger Revisited: Food Charity or the Right to Food? 2nd Edition: Palgrave Macmillan. http://dx.doi.org/10.1057/9781137298737

Rodrigues, F., Alves, A., Polisciuc, A., Jiang, S., Ferreira, J., and Pereira, F., 2013. Estimating disaggregated employment size from Points-of-Interest and census data: From mining the web to model implementation and visualization," International Journal on Advanced Intelligent Systems, p. Vol. 7.

Ryan, N., Pascoe, J., Morse, D., 1999. Enhanced Reality Fieldwork: the Context Aware Archaeological Assistant, in: Dingwall, L., S. Exon, V. Gaffney, S. Laflin and M. van Leusen (eds.), Archaeology in the Age of the Internet. CAA97. Computer Applications and Quantitative Methods in Archaeology. Proceedings of the 25th Anniversary Conference, University of Birmingham, April (BAR International Series 750). Archaeopress, Oxford, pp. 269-274.

Ye, F., and Ganti, H., 2011. Mobile crowdsensing: current state and future challenge, IEEE Communications Ma.