CROWDALENT: AN ANDROID APPLICATION FOR INCREASING THE AWARENESS AND RESPONSE INITIATIVES OF THE CITIZENS THROUGH CROWDSOURCING

John Benedict L. Bernardo*, Kathlyn M. Huavas, GremeirMitz O. Ociones, Ricky A. Pantuan and Jessa Mae P. Vasallo

Department of Information Technology
Mindanao University of Science and Technology
C.M. Recto Ave. 9000 Cagayan de Oro City, Philippines
jbl.bernardo@must.edu.ph

ABSTRACT
Crowdsourcing is a way of collecting information provided by the volunteers. This crowdsourced information has the capacity to increase the people's situational awareness in times of disasters. The research reflected in this paper strives to demonstrate the benefits of crowdsourcing during natural disasters and the ways of utilizing it for disaster response. Shared information regarding natural disasters from social media is often scattered as the inputs from these media are uncategorized. For this reason, the study aims to equip the citizens a medium that is solely intended for sharing and/or obtaining natural disaster-related information. Ergo, an android application was developed to gather and publicize these volunteered information. The capability of crowdsourcing and the effectiveness of the application were evaluated and the result shows overwhelming agreement that this study is indeed efficient in increasing the awareness and response initiatives of the citizens during natural disasters.

Keywords: Crowdsourcing, Natural Disasters, Mobile Application, Social Media, Awareness

1. Introduction

Natural disasters are catastrophic events that are caused by the natural processes of the earth. It is unpreventable and it can cause serious damages to lives and/or properties. Fortunately, the impact and damages these catastrophic events bring can be reduced by preparedness, awareness and response initiatives of the people.

Information and communications technologies (ICTs) largely contribute in increasing the people’s awareness, its use in disaster management has been considered more and more seriously because of the potentials it has in anticipating, communicating and organizing actions before, during and after disaster events (Narvaez, 2012). One of the most important contributions that digital technology can make to humanitarian operations is to ensure that the voices of people affected by disasters and complex emergencies are heard (Vinck, 2013); crowdsourcing is one way of doing this.

The basic idea of crowdsourcing is to obtain the needed services, ideas, or content by soliciting contributions from a large group of people. The term crowdsourcing is first coined by Jeff Howe in his June 2006 article in Wired Magazine. He defined crowdsourcing as the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call.
After a thorough study, Enrique Estellés-Arolas and Fernando González Ladrón-de-Guevara, researchers at the Technical University of Valencia, developed a new integrating definition of crowdsourcing. They defined crowdsourcing as a type of participative online activity in which an individual, an institution, a non-profit organization, or company proposes to a group of individuals of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task. The undertaking of the task the crowd participate entails a mutual benefit. The user will receive satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the crowdsourcer will obtain and utilize to their advantage that which the user has shared or provided.

Crowdsourcing can play a vital role in bridging the preparedness and response initiatives of the citizens. By viewing the citizenry as a powerful, self-organizing, and collectively intelligent force, ICT has the potential to play a remarkable and transformational role in the way society responds to mass emergencies and disaster (Palen et. al., 2010).

Social media such as Facebook and Twitter is extensively used by the netizens in sharing own experiences and situations in times of disasters. This is a good way of informing other people with the level of damage the calamity has already caused. In December 2012 the government of the Philippines turned to social media to help prepare for the oncoming Typhoon Bopha (also called Typhoon Pablo). Before the category 5 storm – the most severe – descended with 258 kph (160 mph) winds, flooding and mudslides, officials began alerting citizens via television, internet and radio; and they created a special Twitter hashtag for the storm, #PabloPH, and a mobile-friendly disaster information page that helped people locate disaster shelters and other assistance (Martin, 2012). Citizens voluntarily provided information of the current happenings during the storm and asked for rescue through the said hashtag.

The use of social media is helpful for sharing and/or obtaining disaster-related facts. These media can provide information, but identifying all these sites and separating the necessary information from the unnecessary is not easy and impractical. An instance of this is when Tropical Storm Debby hit Jackson, Florida on June 2012. Potential data about the storm was captured on twitter. The captured twitter traffic gathered 36,317 tweets matching the keywords “Debby”, “TSDebby”, “flwx” and “hurricane”. The data gained was promising but in further analysis, it was full of ‘noise’. The band One Direction, on that same week, tweeted a post with the word “Debby” in it which was retweeted by around 28,000 people. Over 75% of the traffic captured did not pertain to the storm rather majority of the data relates only to references to the fact that it is currently raining, or complaints about cable TV outages. Overall, majority of the data was noise and distraction thus, finding useful and actionable information was clearly a challenge (Merrick & Duffy, 2013).

Thus, having a medium which only contains the necessary facts would be very helpful and would save people’s time. Perhaps the latest wave of preparedness-oriented ICT applications lies in crowdsourcing. What takes place behind crowdsourcing activities are the efforts of individuals who not only facilitate and maintain the infrastructure that makes crowdsourcing possible but, increasingly, create a broader network of individuals and institutions seeing the use of crowdsourcing for disaster management in a more proactive manner (Narvaez, 2012).

This study developed an android application to provide the citizens a medium that is exclusively intended for sharing and/or obtaining natural disasters-related information. It also targets to deliver instantaneous updates that are vital in attaining rapid situational awareness. The system was implemented in Cagayan de Oro City, Philippines.
2. Methodology

A program was designed and developed using appropriate languages, techniques and codes that met the user’s needs.

2.1 Analysis of the Android and Web Application

The desired outcome was attained by constituting the following features:

a. **Dynamic Reports.** The home tab is automatically updated when new reports are made without having to refresh the page.

b. **Real-time Capture.** To ensure that no old photos will be uploaded, users are not allowed to upload file from the local storage of their phones. Instead, when uploading a photo and making a report, they will be redirected to the camera (within the application) assuring that the photo captured happened in the present time.

c. **Information Integrity.** Crowdsourced information may have integrity issues. Flagging, Marking (as Important) and Commenting were added to improve the credibility of the reports made by users.

   - **Flagging** – A user may flag a post which he sees as fake or false. When that post gets flagged five (5) times, it will be automatically deleted from the application. Moreover, the user whose posts got deleted thrice (3) will be banned from using the privileges of a registered user; hence, he will only be allowed to view the reports – just like an unregistered user.

   - **Marking as Important** – A user may mark a post as important if he sees it as credible. When there is more than one user who made a report in the same location, the user’s post that got the most marks will be prioritized in displaying.

   - **Commenting** – Users may also add comments to the reports made.

d. **Registration** - Log in with Facebook. Users may register to the application through their facebook accounts, which makes the registration quick and easy. If a user wants to make a report but is not yet registered, he will not have to type much information and could immediately make a report in a few taps.

A **web application** was designed to allow users from different platforms to access the reports made from the android application. However, they will not be allowed to report, flag, mark and comment on the posted reports.

2.2 Design of the Android and Web Application

![Figure 1: Use Case Diagram of the Mobile Application](image-url)
Figure 1 and 2 are use case diagrams that show how the users interact with the application. In figure 1, the diagram comprises of two actors which represent the different user rights and privileges. The registered user, who has logged-in in the application via Facebook, has the rights to login or logout, view reports, post reports, mark the reports as important, flag other posts and comment. On the other hand, unregistered user can only view posted reports and comments. The interaction of the user to the web application is shown in figure 2 where users can only view posted reports from it.

Figure 3 shows both the mobile and web architecture of the application. The user has the access to both the mobile and web application. However, it differs in the privileges that are provided by the different media. In mobile application, the user has all rights and privileges provided by the application. While in the web application, the user can only view the reports provided by the user who uses the mobile application privileges. All information requested by the user will be retrieved from the database through the internet and will be provided back to the user.

2.3 System Development

2.3.1 Mobile Phones

The application was tested on android phones Samsung Galaxy s3 (Android version 4.3 - Jellybean), MyPhoneOceanlite (Android version 4.2 – Jelly Bean) and O+ 8.93 (Android version 4.4 – KitKat). Different devices were used to check if the application would still work efficiently on different android versions and screen sizes.


2.3.2 Android Application

The android application was made using the web technologies HTML, CSS and Javascript. Since this is not the native language of Android OS, this study made use of Apache Cordova for the application to communicate with the device’s Application Programming Interface (API) like the camera, geolocation and the internet. Android SDK was also installed to build the application’s Android Application Package (APK) through command line. The android application sends and receives data from CrowdAlertAPI, an API created by researchers. For the application’s user interface, this study used the Ionic Framework. Crosswalk, an open source project that allows you to specify a version of Chrome to use as your web browser in Android, was also used to properly render HTML and CSS for older devices because older devices make use of the older version of web view, which does not support all the latest feature of HTML and CSS.

2.3.3 Web Application

The web application receives data from the same API the android application uses – CrowdAlert API. For the web application’s user interface, this study used HTML, CSS and JavaScript.

2.3.1 Server

Three (3) virtual hosts were created using the Nginx web server running on Ubuntu 14.04 LTS with MySQL as the database. One virtual host serves the CrowdAlert API. Another virtual host is for the Node.js-based application for real-time broadcast of data to all connected devices. The last virtual host serves the web application. Nginx serves as the reverse proxy for the Node.js-based application.

2.4 Crowd Alert API

Using the Laravel framework, the researchers created their own API. The Crowd Alert API contains the main logic of the application. This is where the android application fetches and uploads the data for the reports, flags, marks and comments. This is also where the web application receives the information uploaded from the android application.
Figure 4 is a representation of the processes of CrowdAlert API. It shows the procedures when a user obtains and shares information to the application. In retrieving reports, CrowdAlert API queries the database, fetches the images from the disk and sends it back to the end users. While in uploading data, the API first checks if the user is registered or not, if not, he will be redirected to the Facebook’s login page. Facebook will then generate a token that would give the user access to the application. Facebook will also send the user’s information to the application such as name, Facebook ID, e-mail address and profile picture. After logging in, the registered user can proceed in uploading the report that contains the necessary data – photo, disaster type and description. These information will then be saved to the database and the image to the disk.

3. Results and Discussion

The main purpose of this study is to come up an android and web application to provide the citizens a medium that is exclusively intended for sharing and/or obtaining natural disasters-related information.
3.1 Graphical User Interface of the Mobile Application

Figure 5 shows the home tab where the reports made within 24 hours are seen. The reports can be sorted by most recent or most important. In most recent option, latest reports are shown first while in most important, reports with most marks will be prioritized in displaying. Also, the home tab will be automatically updated once a new report has been made, the user doesn’t need to refresh or reload the page. Figure 6 shows the individual report where a user has the option to view the map, mark a report as important, flag a report and view and make a comment. Figure 7 shows the disaster tab where reports are categorized according to the type of natural disaster. Each disaster has a counter which pertains to the number of reports made within 24 hours. A user may choose what type of disaster report he wants to see.

3.2 Graphical User Interface of the Web Application

Figure 8 shows the home page of the web application.
The web application is for viewing purposes only. It has three (3) tabs: home tab, disasters tab and archive tab. Any user can access the application without logging in via facebook. This allows the users from different platforms to still view disaster-related information that are being reported from the android application. Figure 8 display the home page of the web application’s interface. Each of these tabs has the same functionalities with the mobile application.

3.3 System Evaluation

The application was evaluated based on its functionalities, design, usability and performance. Random citizens from Cagayan de Oro as respondents to test and evaluate the application’s functionalities, design and usability. Different mobile devices were used to test the mobile application’s performance. As for the web application, the performance was evaluated by load testing the application with different numbers of concurrent users.

Table 1. Mean Responses in terms of Functionalities, Design and Usability for Android Application

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Average Mean</th>
<th>Adjectival Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionalities</td>
<td>4.7</td>
<td>Strongly Agree</td>
</tr>
<tr>
<td>Design</td>
<td>4.6</td>
<td>Strongly Agree</td>
</tr>
<tr>
<td>Usability</td>
<td>4.7</td>
<td>Strongly Agree</td>
</tr>
</tbody>
</table>

These are the tabulated results of the functionalities, Design and Usability of the application assessed by the respondents. Table 1 shows the application in terms of functionalities has gained a positive feedback with an average of 4.7 indicating that the respondents agreed that it has accomplished in providing efficient functionalities. The result of how the respondents rate the applications interface shows that the respondents strongly agreed that the design is responsive and is satisfactory with an average of 4.6. The overall usability of the application was rated with an average of 4.7, imply that the application is effective being a medium in sharing and/or obtaining disaster-related information.

Table 2. Android Application Resource Usage Test Result

<table>
<thead>
<tr>
<th>Android Devices</th>
<th>Samsung Galaxy S3 Android Jellybean</th>
<th>O+ 8.37 – Android Jellybean</th>
<th>MyPhone Agua Ocean Lite – Android Jellybean</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM usage</td>
<td>74.9MB</td>
<td>93MB</td>
<td>73.6MB</td>
</tr>
<tr>
<td>CPU Usage</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Storage Usage</td>
<td>69.03MB</td>
<td>42.32MB</td>
<td>42.32MB</td>
</tr>
</tbody>
</table>

System Monitor Application was used to track the resource usage of the application: RAM usage, CPU usage and Storage usage. Table 2 shows that the application consumes large amount of space in the memory but has minimal to 0% CPU usage which means that it has low demand of processing power. Based from the tests done from different android phones, the application’s performance varies depending on the type of phone it runs.
Table 3. Web Application Load Test Result

<table>
<thead>
<tr>
<th># Of Concurrent Users</th>
<th>Average Response Time</th>
<th>Response Count</th>
<th>Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Success</td>
<td>Timeout</td>
</tr>
<tr>
<td>50</td>
<td>262ms</td>
<td>40</td>
<td>0</td>
</tr>
<tr>
<td>150</td>
<td>241ms</td>
<td>150</td>
<td>0</td>
</tr>
<tr>
<td>250</td>
<td>251ms</td>
<td>249</td>
<td>0</td>
</tr>
</tbody>
</table>

Loader.io was utilized to stress test the capacity of the web application to handle concurrent connections. Table 3 shows the result of the Load Test conducted to test the Web Application’s performance based on different numbers of concurrent users. It shows the web application’s response time without timeout and network congestion. It also shows the uploading and downloading capacity of the application.

4. Conclusion

The study shows the use of crowdsourcing in disaster management. The application, acting as a medium for disaster-related information, allows individuals to take part in such issues. Thus, empowering the citizens by involving themselves to share critical data, as well as to gain information to have a situational awareness. The application’s effectiveness was tested and gained positive feedbacks from the respondents. It also passed the performance test conducted using the testing tools. Results show that the application was effective in providing a common platform that is solely intended in propagating and/or collecting information that could help in disaster management. Crowdsourcing is not one hundred percent reliable, but with the features of the application, the credibility of the reports is fortified.

References


