ARTICLE IN PRESS

International Journal of Disaster Risk Reduction xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

International Journal of Disaster Risk Reduction

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

Development of a mobile post-disaster management system using free and open source technologies

Yuanrong He[a,b], Dejian Zhang[a,*], Yihui Fang[c]

[a] College of Computer and Information Engineering, Xiamen University of Technology, Xiamen 361024, Fujian, China
[b] Key Laboratory of Urban Environment and Health, Institute of Urban Environment, Chinese Academy of Sciences, Xiamen 361021, Fujian, China
[c] Information Management & Engineer Department, Fujian Commercial College, Fuzhou 350012, Fujian, China

ARTICLE INFO

Keywords:
Disaster
MDMS
Mobile GIS
Open source
Post-disaster management

ABSTRACT

Portability, accessibility and usability in extreme situations during or after disasters are essential requirements for a disaster management system to work at full capacity. Affordability is another concern that should be highlighted, especially in underdeveloped countries. For these purposes, a mobile-based post-disaster management system (MDMS) mainly used for collecting, sharing and disseminating disaster-induced damages/risks was successfully developed under a combination of native and web application technologies (so-called hybrid technologies) using various open source and free software such as GeoServer, Openlayers, Cordova, and jQuery Mobile. The MDMS was then assessed using a case study with a class of 45 students who were asked to report typhoon-induced damages/risks via the MDMS and complete a questionnaire concerning the portability, accessibility and usefulness of the system. In general, the functionality provided by the MDMS was well recognized, as most of the surveyed students were satisfied with the mobile and web-based technologies. The cross-platform and offline-work capabilities as well as the portability of the system were considered the most valuable features in facilitating post-disaster management. Therefore, it was concluded that an MDMS based on free and open source software constitutes an affordable, portable and cross-platform solution for post-disaster management. Nevertheless, as a prototype system, the MDMS will require further improvements to provide better user experiences and maximize its functionality.

1. Introduction

Disasters such as floods, tsunamis, typhoons, earthquakes and landslides have resulted in serious damage to disaster-stricken areas [1,2]. Up-to-date and reliable information about these disasters, including their location, duration, the number of people affected, the extent of the damage, and the location of aid resources, is essential for responding to these emergencies [3]. Nevertheless, the availability and accessibility of this type of information may be limited in some cases. This can pose a serious problem for disaster management efforts that rely on timely spatial data to estimate damages and coordinate relief activities, especially in rural areas [4].

In the past few decades, the emergence of new information technologies, such as Geographical Information Systems (GIS), Remote Sensing (RS) and Global Positioning Systems (GPS), have proved to be effective tools for collecting and communicating the spatial distribution of risks associated with particular areas [5–10]. Many GIS-based and RS-based disaster management solutions were devised for disaster data collection and dissemination. For example, the NGO Development Workshop in France conducted a GIS hazard mapping project for reproducing the magnitude and extent of past flood disasters and supporting decision-making regarding flood risk reduction activities in central Vietnam in 2005. Andrews Deller [11] used color Advanced Spectral and Thermal Emission Radiometer (ASTER) imagery draped over a DEM to produce three-dimensional views of hazardous terrain in Ethiopia, and the reproduced imagery was readily understood by people with no prior experience of image interpretation or map-reading. Chi et al. [12] introduced a collaborative framework of an interactive WebGIS platform integrated with a multi-criteria evaluation tool to support the engagement of different stakeholders and the encouragement of a collaborative, decision-making process for flood and landslide management. More examples of GIS-based and RS-based disaster management solutions for delivering and communicating disaster information can be found in research by Kelmelis et al. [13]; Parvaiz et al. [14]; Taubenböck et al. [15] and Srivihok et al. [16]. While these solutions have largely increased the efficiency of disaster data collection and enabled the data to be shared with a wider range of communities, they may still require a significant amount of time to

* Corresponding author.
E-mail address: djzhnag80@163.com (D. Zhang).

http://dx.doi.org/10.1016/j.ijdrr.2017.08.007
Received 22 November 2016; Received in revised form 10 August 2017; Accepted 11 August 2017
2212-4209/

Table 1
Key functionalities identified through the requirement analysis.

Component	Functionality	Goal
Backend services	Automatic geodata acquisition	To collect geodata from third-party map services and relieve the overhead and time in producing geodata.
	OGC Web Feature Service and Tile Map Service	To disseminate geodata in an OGC standards-compliant way.
	OGC Web Feature Service-Transaction	To edit vector geodata over the Internet.
	Authorized mechanism	To enhance the reliability of the obtained data.
Mobile and web client	Report disaster-induced damages	To save disaster-induced damage to the spatial database via the WFS-T service and pictures about the disaster-induced damage via the HTTP protocol
	Searching nearby facilities/risks	To retrieve nearby facilities/risks via the WFS service
	Visualization of the geodata	To display the base map and overlaid risks, damages and/or facilities.
	Geolocation service	To retrieve user location information, such as latitude and longitude
	Working in offline mode (mobile client only)	To enhance the usability in extreme conditions
	Access to the camera (mobile client only)	To take a photo using device cameras or retrieve a photo from the image gallery.

collect field data as well as to process and geo-reference the secondary data. Solutions using aircraft or satellite images may also provide a more efficient means of mapping and monitoring hazards before, during and after a disaster. Nonetheless, the quality of aircraft or satellite imagery is largely subject to the quality of the image sensors, meteorological conditions and so on. Furthermore, risks and damage that occur inside buildings cannot be documented through these methods.

Recently, the evolution of mobile devices and geospatial technologies has ushered in a new phase of geographic-information application use in the practice of disaster management. Among these technologies, Mobile GIS, which extends the capability of traditional GIS to a higher level of portability, usability and flexibility, is one of the most vital technologies for the efficient and effective data collection, sharing and dissemination [17–19]. Examples of Mobile GIS use in disaster management are ubiquitous. For instance, Aanensen et al. [20] proposed a framework for epidemiology, ecology and community data collection by taking advantage of the open source nature of Android technology and open development tools for web applications (such as Google Maps and Google Earth). In this framework, users can submit and retrieve data to and from the central database as well as view and analyze the data with Google Maps both via the web and on mobile phones. Rafoss et al. [21] successfully implemented a new application to support risk and crisis management in confronting the threat that invasive species pose to global crop production and biodiversity. In this application, the open geospatial standards Web Feature Service-Transaction (WFS-T) was implemented for GPS-enabled mobile phones communicating with a GeoServer back end to track and eradicate disease outbreaks and susceptible host plants. Tsai et al. [22] constructed an application called Mobile Escape Guidelines (MEG) by integrating geographical information and augmented reality techniques to help people evacuate from nuclear accident sites. Kalabokidis et al. [23] developed an integrated wildfire prevention and management platform (AEGIS) to manage wildland fire hazards in Greece that uses an innovative smartphone application, the AEGIS App, to operate in conjunction with the web-based version of the system. These specifically designed applications have significantly improved the quality of disaster management. Nevertheless, the general accessibility (i.e., access from multi-platforms and multi-devices) and usability of such applications in extreme situations either during or after disasters, which represent an essential requirement for better disaster management, is not well addressed. Affordability is another concern that should be addressed, especially in the interest of underdeveloped regions [24,25].

In this work, a mobile-based post-disaster management system (MDMS) is presented by using open source and free software, such as GeoServer, OpenLayers, Cordova, and JQuery Mobile, for risk and damage information collection, sharing and dissemination. Unlike conventional solutions, this study focused on building the MDMS by using a combination of native and web application technologies (so-called a hybrid technologies) [26] for the first time in mobile-based disaster management solutions. Furthermore, this study also stressed valuable features in facilitating post-disaster management such as offline-access capabilities and portability. To assess the MDMS and gather information for further improvement, this system was applied to collect and disseminate disaster-induced damages/risks during typhoon Meranti, one of the strongest typhoons ever to make landfall in Amoy since the 1980s. In this case study, a class of students was asked to report typhoon-induced risks and damage via the MDMS and complete a questionnaire concerning the portability, accessibility and usefulness of the system. Finally, the performance of the MDMS was assessed according to the responses provided by the surveyed students.

2. End-user requirement analysis

Prior to the development of the MDMS, an analysis was conducted to gather end-user requirements for the information and desired functionalities of the system from key stakeholders as well as documentation of similar solutions or products. The essential use cases and key functionalities (Table 1) of the system were identified through this analysis. In general, the stakeholders wanted to develop the MDMS into a dedicated system for collecting, sharing and disseminating disaster-induced damages/risks. In other words, some of the system functionalities, such as the reporting disaster-induced damages functionality, were designed for use by specific groups of authorized users only for the reliability of the obtained information. Nevertheless, other functionalities were available to all users for sharing obtained data with a wider range of communities. The most essential and representative use cases identified were as follows: submitting disaster-induced damages/risks and searching nearby facilities/risks. A larger number of interactions and information communications between clients and back end services were involved in these use cases. Data exchanges include map tiles, pictures of disaster-induced damage, and locations and attribute data of reported damages. For a high level of portability, the stakeholders also required the system to be implemented in an Open Geospatial Consortium (OGC) standards-compliant way; thus, the data exchanges (except for the pictures of disaster-induced damage) between the clients and backend services must follow the OGC standards, namely, the Web Feature Service (WFS), WFS-T and Tile Map Service (TMS).

3. Architecture overview

It is an essential requirement for a disaster management system to be accessible and operational under extreme situations in which reliable Internet connectivity or speed are not normal. In addition, the ability to work across multiple platforms with affordable cost is beneficial to expend its user community. With these requirements in mind, a viable generic 4-tier system architecture is outlined (Fig. 1). The system is a conglomeration of multiple free and open sources tools, resources