Disaster Communications System

By T.J. Muthig

Submitted to
the Faculty of the Information Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

University of Cincinnati
College of Applied Science

May, 2009

Disaster Communications System
By T.J. Muthig

Submitted to
the Faculty of the Information Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Technology

© Copyright 2009 Thomas J. Muthig III

The author grants to the Information Technology Program permission
to reproduce and distribute copies of this document in whole or in part.

___________________________________________________ __________________
T.J. Muthig                Date

___________________________________________________ __________________
Robert Schlemmer, Faculty Advisor     Date

___________________________________________________ __________________
Hazem Said, Ph.D. Department Head     Date
Table of Contents

Statement of Need............................................................................................................

Product Description and Intended Use............................................................................

User Profiles....................................................................................................................

Design Protocols.............................................................................................................

Deliverables.....................................................................................................................

Development....................................................................................................................

Budget.............................................................................................................................

Technical Aspects..........................................................................................................

Testing Plan.....................................................................................................................

Risk Management Plan..................................................................................................

Proof of Concept.............................................................................................................

Conclusion....................................................................................................................... 

Appendix........................................................................................................................
Statement of need

On September 13th, 2008, at 2:10 am, Hurricane Ike hit Galveston Texas. Ike caused floods, downed telephone and power lines, did $16 billion worth of damage, made thousands homeless, and cut power to about six million people (3, 4). In all, 126 individuals died and 202 more disappeared (6). In the aftermath, millions of people donated millions of dollars in cash, goods, and services. Several large agencies, including the Salvation Army, the American Red Cross, and Catholic Social Services, brought massive relief efforts to the region (5). As with other disasters, a crippling backlog of relief supplies developed, as supplies sat in warehouses and on trucks while volunteers figured out where to send them. Volunteers also experienced difficulty pinpointing where the supplies would be best utilized (2,3).

This situation, unfortunately, is an all-too familiar scenario. This scene played out with Ike, as well as in Katrina, Gustav, and in the Pacific tsunami. In the aftermath of a disaster, relief efforts and supplies come in, but communications and coordination leave something to be desired (2). The problem is that each agency brings its own system to the relief efforts. Bureaucracy and disorganization hamper communication and coordination even within organizations, and make communications between different organizations a complete debacle (7).

PROJECT DESCRIPTION AND INTENDED USE

The evidence makes it clear that a central communication system, shared by all the agencies in the area, would be of great benefit to disaster relief in the effected area. An integrated and open system would enable smooth communication between agencies, no matter their size or primary objective. A successful system could also incorporate private
volunteers, allowing these volunteers to not only find where they could be most helpful, but also to allow all the concerned agencies to examine the volunteers’ skill sets, allowing them to contact the ones they need most..

The software to enable this system is what will be the focus of this project. For such a system to be successful, it must include several features. A database must be built to hold all of the relevant information. The database would track information about the volunteers, along with information about relief sites, such as a blood donation site, and relief events, such as a cleanup project. This database would also interface with a Google Maps application that would show the relevant locations on a map easily accessible to the end users. Finally, there would need to be a message board to allow for rapid mass communication. This board would also allow individuals to bring problems to the attention of particular agencies. These systems will be designed so a user can access them not only from a laptop or desktop PC, but from a mobile device such as a cell phone or PDA.

USER PROFILES

The Disaster Communications System is aimed at several different types of users. The primary users will be the local county government and other authorities of the area affected, along with large aide organizations such as the Red Cross. They will be the users responsible for populating the database with rescue events and relief efforts. This will allow the organizations to be more transparent and coordinated with their efforts, so valuable time and resources are not wasted. Another large user population will be the individual volunteers. They will populate the volunteer database with their skills and availability, allowing relief organizations to quickly assemble a team to meet a specific
task. They will also be able to put their skills to use by searching the projects provided by
the relief agencies. The last group to use this project will be those affected by the disaster.
They will be put in touch with valuable information on where to go for relief and shelter.
All user groups will be able to come together and use the message board to communicate
with each other and with the authorities, and receive other general information. The use
case diagram in Figure 1 details how each user interacts the system.

(Figure 1) Use Case Diagram

![Use Case Diagram]

Figure 1: Use Case Diagram

**DESIGN PROTOCOLS**

The system consists of several distinct parts that will all work together in the
event of a major disaster. The central component of the system will consist of several
databases developed in SQL Server 2008. These databases will hold information on relief
sites, volunteer skills and other information, victim’s needs, etc. Users will be able to
navigate to data entry points from the message board, and fill in their relevant data. The volunteers and victims are able to go back and access the data created by the agencies, learning where they can help, get provisions or find any service they may need. Users interact through this part of the system with a Windows Form based C# application. This application works quickly and smoothly with the server, and allows the users quick access to the data they need, while denying them access to the data they do not. In addition to the Windows Form version of the software designed to run on a regular desktop, there is a mobile web version of the program, developed in ASP 2.0 and designed to run in a mobile web browser. It allows access to the same information as the desktop application in a more concise format for the mobile web.

Another resource available to the users is SQL reporting services, or SSRS for short. This web-based service allows users to query data stored on the SQL server from their browser. Reporting services allows end users to search for specific data about relief centers and events around the area. System Administrators can also use reporting services to search the volunteer database and compile lists of potential workers, based on the skills they have, to fill the needs of the area.

A web-based application integrating the Google Maps Web API was also developed. This application allows users to see the physical location of important locations, represented as pins on a Google Map, which the system draws from the database discussed above to track information about them. End users are able to locate relief locations and events. Relief agencies and local government will also be able to use this part of the system to get a picture of where their equipment is deployed across the area.
Another important piece of the system is the online message board, which all
users can access through a web browser. The message board is where the users,
particularly the individual volunteers and disaster victims, will first interact with the
system. The message board provides links to other parts of the system, allows users to
ask general questions and allow agencies to publish important information that needs to
reach everyone quickly. It is also the tool which users utilize to create their accounts
needed to interact with certain parts of the system, including volunteers creating profiles.

**DELIVERABLES**

1) A database system that keeps track of volunteers, relief events, relief locations, the
location and condition of remote access points in SQL Server 2008.

2) A desktop based C# application to administer the databases.

3) A Windows mobile based C# application to allow users access to the databases.

4) A C# application using Google Maps that will display the information on a map. This
will show where the carts are, and where aid events and relief locations are based.

5) A message board, based in PHP, which will be used to distribute information and
coordinate efforts.

6) An .ASP webpage to direct people to the message board and allow them to download
the applications.

**DEVELOPMENT**

The timeline depicted in Figure 8 lays out how the project was completed. After
the research and proposal were completed, construction on the database that serves as the
backend for all the applications was started. After that the Google maps application was
created. Next, the message board was created. Then the SQL reporting services pages
were created, along with several custom reports. Lastly, the mobile ASP web applications
were created. After each component is completed, it will be put through the alpha testing process. After all of these components were completed, testing of the project began.

Documentation was also a heavily emphasized part of the project. In early February work on the design freeze will begin. February 16th the rough draft was turned in and reviewed, with the final design freeze turned in March 10th. After design freeze, preparation began for Tech Expo, which was held May 7th. After Tech Expo, the project was presented to the faculty and students.

![Figure 2: Timeline](image)

**BUDGET**

The costs incurred by this project were little to nothing in terms of actual financial resources. The hardware needed for the project was either already owned or purchased at an economically conservative price. Thanks to the University of Cincinnati’s agreement with Microsoft and the MSDNAA, most of the necessary software was obtained free of charge. Other software utilized in the project, specifically PHPBB and the Google Maps API were open source. Table 1 on the next page details each item used in the project. Retail costs are listed along with costs actually incurred in creating the project.
### Table 1: Project Costs

<table>
<thead>
<tr>
<th>ITEM</th>
<th>DETAIL</th>
<th>RETAIL COST</th>
<th>INCURRED COST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dell Laptop</td>
<td>Use for programming, and running software during presentations</td>
<td>$300</td>
<td>$300</td>
</tr>
<tr>
<td>Visual Studio 2008</td>
<td>Obtained through MSDNAA</td>
<td>Varies Between $600 and $1,000(8)</td>
<td>$0</td>
</tr>
<tr>
<td>SQL Server 2008</td>
<td>Obtained through MSDNAA</td>
<td>Varies Between $3,000 and $12,000(8)</td>
<td>$0</td>
</tr>
<tr>
<td>PHPBB</td>
<td>Open Source software obtained from PHPBB.com</td>
<td>$0</td>
<td>$0</td>
</tr>
<tr>
<td>Lexar 512 MB Jump Drive</td>
<td>Given to me as a gift</td>
<td>$30</td>
<td>$0</td>
</tr>
<tr>
<td>Western Digital 120 GB External USB Hard Drive</td>
<td>Will be used as a backup to the laptop and Jump drive</td>
<td>$70</td>
<td>$70</td>
</tr>
<tr>
<td></td>
<td>RETAIL TOTAL: $4,000-$13,400</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ACTUAL COST: $370</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### TECHNICAL ASPECTS

#### Hardware
- Dell D600 Laptop
  Most of the programming and testing will be done on this laptop.
- OCAS Computer Lab
  Any programming not done on the laptop will be done in the OCAS computer lab in room ADMIN 401.
- Lexar Jump Drive
  512 Mb jump drive. Will be used as a backup and to store important documents related to the project.
- Western Digital 120 GB External Harddrive
  120 Gb USB external hard drive. Will be used as a backup to both the laptop and the USB drive, and to store any large files.

#### Software
All of the software for this project was either provided by the MSDNAA (Microsoft Development Network Academic Alliance) or is open source under the GNU license.

- **Microsoft Visual Studio 2008**  
  This will be used to program the database applications in C#, and also for the Microsoft reporting services portions of the project.

- **Microsoft SQL Server 2008**  
  This program will be used to create and manage the databases in this project.

- **PHPBB**  
  This is an Open source program that will be used to create the message board component of the project.

- **Google Maps API**  
  Javascript API used to create the DisasterMaps application.

**TESTING**

In the second quarter of senior design, large emphasis was placed on the importance and the different methods of product testing. It is important to test the project in order to find defects and bugs. Removing these in the early stages of the project will not only make the project completion and implementation easier, but will allow for a smoother and more worry-free project support phase. It is also important to do user-testing, to ensure user’s are happy with and will use the product.

The first type of testing performed was the technical testing on the software components of the system. Queries and stored procedures were tested to ensure they perform the operations they are intended to (and only those functions). Programs were run on several different machines with different operating systems to ensure consistency in performance. This process will also be used on the web-based components, which will also be tested on a variety of popular browsers.
The second and arguably most important part of the testing was the user testing. Friends and colleagues, who are at all ends of the spectrum in terms of computer proficiency, were asked to test the usability of the application. The application was tested to ensure the instructions for the user are clear, and that the system will easily do what the user wants it to. Efforts were also be taken to ensure the system does not do anything to confuse or frustrate the user, which will contribute to user buy-in. Comments, questions and suggestions were collected from the users and the system was revised.

Security is always a main concern with any software, and testing is imperative to ensure the security and integrity of the data. In addition to the internal tests that were performed, the system was also be placed in front of several knowledgeable colleagues who tested the system by trying break or “hack” the system. They played the roles both of the blundering illiterate user, and the disruptive and malicious “griever” trying to disrupt the system. Holes they found in the system were patched up and the system will be tested again.

Testing the Disaster Recovery Communications System in a deployment environment was also conducted. In addition to the local environment testing, the system was tested on the WAN and was designed to ensure the two systems work together smoothly. The WAN testing was accomplished by placing the project on the senior server and using a computer in the lab as a client. The project was also deployed on the DisasterComm group’s equipment, to ensure it would work in a field environment.

Testing of the Disaster Recovery Communication System will be ongoing and iterative. As each component of the system becomes operational, it will enter the appropriate testing phases. Once the system completes a phase, revisions will be made
and it will be tested again. The goal of this method is to produce the highest quality project possible.

Figure 3: Field Testing

**RISK MANAGEMENT**

All projects must account for any risks posed to them, and the Disaster Recovery Communications System is no exception. Being a database focused project, the primary risk will be to protect the integrity and security of the data at all levels. The data will only be assessable to a few chosen individuals, overseen by one administrator. The system requires a password anytime a user wants to access it. Data validation also takes place at the client level, and again at the database level. When data is transmitted, it is encrypted to ensure security. Finally, only non-sensitive data will be collected from end-users. No social security numbers, credit card numbers, or other sensitive information will be collected.
Another significant risk is a lack of user acceptance of the system. To ensure that users will accept the system, user testing will be conducted on each component of the system, with the revisions then retested for ease of use and acceptance. In short, I will work with the users each step of the way to ensure the best level of user buy-in possible.

Hardware failure is always a problem lurking in the background with any IT project. Certain precautions, such as backing up to multiple sources while the project is in development will be implemented. A regular backup plan to preserve the data will be designed, and will be implemented once the system is deployed. Table 2 details some of the risks identified in the project, the likelihood of them occurring within the project’s lifetime, and the measure’s taken to mitigate them.

<table>
<thead>
<tr>
<th>RISK</th>
<th>PROBABILITY</th>
<th>MITIGATION ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal security breach: Employee will purposely misuse sensitive data in the system.</td>
<td>LOW</td>
<td>Only a few select admin’s will have access to all the data. Data will be password protected and encrypted.</td>
</tr>
<tr>
<td>Failure of user buy-in</td>
<td>MED to HIGH</td>
<td>Users interviews and testing to ensure the system meets the needs of the user in function and usability.</td>
</tr>
<tr>
<td>Hardware/other failure resulting in loss of data/infrastructure.</td>
<td>MED</td>
<td>Backup systems will be put in place to preserve data and functionality in case of a hardware failure.</td>
</tr>
<tr>
<td>Outside security breach/attack</td>
<td>MED</td>
<td>Access to parts of the system will be limited and Password protected. Once deployed, network monitoring software could be used to protect the system.</td>
</tr>
</tbody>
</table>

Table 2: Risk Assessment
PROOF OF CONCEPT

The client based C# program will be one of the main ways the users enter and retrieve data from the database. All users, regardless of their status, will begin by logging into the system, as show in Figure 2.

![User Login Page](image)

Figure 4: User Login Page

The logon information will be encrypted and verified by the server. The user will then be allowed into the system. In order to mitigate the risk of data being stolen or corrupted, each user will have a security level which determines which parts of the system they may access, what data they may view from the system, and what data they can give to the system. Once the user logs in, they will be presented with a menu, its choices determined by their security level, as shown in Figure 3.
Users who have just created their accounts will only be able to change their passwords. Once they have changed the password, they will be able to edit information about themselves, and view information about relief events and locations. To become a higher level user, an account will need to be tagged by an administrator. Higher-level users, who work for the county, will also be able to use this system to manage the network carts providing connectivity out in the field (See Figure 4).
The data for this system will be held on a server running SQL Server 2008. This database administration program is both powerful and secure when used properly. It will provide scalability to the system as the load of users and volume of data increases.

The system will collect data about the physical locations of each cart, relief event and aid location. This data will then be used to construct a web accessible application integrated with Google Maps to give a graphical representation of important locations within the scope of the disaster area (see Figure 5). This will provide end users and administrators with an overhead shot of the current situation on the ground. The page interfaces with Google Maps through an API, which is programmed in Javascript. The page itself will be developed in ASP and be hosted on an Internet Information Services (IIS) server. The data for the map points will come from an XML file.
Another important tool that will be utilized is SQL reporting services. A variety of reports will be created for each user level. A report publishing a list of relief locations (ex. Blood Bank or Shelter) and relief events (ex. sandbagging) will be open to the public, as shown in Figure 6. Administrators will also be able to run reports on the status of the network carts, along with being able to search lists of volunteers by their availability and by their skills. This will allow the authorities to put together the best possible team for a particular event.
A message board was also constructed to serve the needs of the area (See figure 7). The message board is constructed using the PHPBB tool, which relies on the PHP programming language. The message board also uses the previously mentioned SQL Server as a back end. Users will create one central login account using the message board. In addition to using this account on the message board, users will use the account for all the other components of the system. The message board will allow the end users to communicate questions and concerns to the administrators and to each other, and allow the administrators and authorities to disseminate general information to the public.
In addition to the desktop based applications, a mobile web-based application was also developed using ASP technology. This site was designed to allow users access to important information when in the field, using a mobile phone or other mobile web enabled device. The mobile web application includes most of the features of the desktop based application, and interacts with the same database.
Conclusion:

There is clear need for a communication system that can be deployed quickly and utilized by all the “stakeholders” in a disaster situation. The scenarios that have played out in previous disasters show that a system like this could not only save money and time, but possibly lives as well. Once created, this project would easily be adopted and deployed by an appropriate agency. This project would be the perfect opportunity to contribute something positive and helpful to the local, and possible national community.
Bibliography

1: “Texas TERT Activation For Hurricane Ike A Success”, September 24\textsuperscript{th}, 2008, \url{http://www.medicalnewstoday.com/articles/122720.php}

2: “Communications Lessons Learned Along Gulf Coast”, October 1\textsuperscript{st}, 2008, \url{http://www.satellitetoday.com/via/ednote/24711.html}

3: “Traffic chaos as locals try to get into Galveston”, September 17\textsuperscript{th}, 2008, \url{http://www.msnbc.msn.com/id/26754232/}


5: “Red Cross Responds to Hurricane Ike”, September 7\textsuperscript{th}, 2008 \nl \url{http://www.redcross.org.uk/news.asp?id=85506}

6: “Texas begins mass post-Ike rescue”, September 15\textsuperscript{th}, 2008, \url{http://news.bbc.co.uk/2/hi/americas/7616973.stm}

7: “FEMA / Red Cross Katrina Response a Disaster”, September 27\textsuperscript{th}, 2005 \nl \url{http://www.gulfcoastnews.com/Katrina/GCNOpinionFEMAKatrina.htm}


9: PHPBB website, accessed 11/12/08, \url{http://www.phpbb.com/}
Appendix

Figure 1: Use Case Diagram: ................................................................. Page 6
Figure 2: Timeline: ............................................................................. Page 9
Figure 3: Field Testing: ..................................................................... Page 13
Figure 4: User Logon: ....................................................................... Page 15
Figure 5: User Menu: ......................................................................... Page 16
Figure 6: Equipment tracking: .............................................................. Page 17
Figure 7: DisasterMaps: ................................................................. Page 18
Figure 8: Sample SSRS report: ...................................................... Page 19
Figure 9: PHP MessageBoard: ............................................................. Page 19
Figure 10: DisasterMobile: ............................................................... Page 21

Table 1: Budget Plan ................................................................. Page 10
Table 2: Risk Management Detail ...................................................... Page 14